OPEN CHANNEL FLOW

Numerical Methods and Computer Applications

Roland Jeppson

CRC Press
Taylor & Francis Group

OPEN CHANNEL FLOW

Numerical Methods and Computer Applications

OPEN CHANNEL FLOW

Numerical Methods and Computer Applications

Roland Jeppson

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20150505

International Standard Book Number-13: 978-1-4822-8216-0 (eBook - PDF)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Contents

List of Figures

List of Tables

Preface

This book was developed over several years while teaching courses in open channel flow to graduate students. Initially, while on the quarter system, two open channel graduate-level courses were taught: the first dealing with steady-state flow was a four-credit course, including one credit for a laboratory; and the second dealing with unsteady flow, that is, numerical solutions of the St. Venant equations, was a three-credit course. When the university switched to the semester system, these two courses were both expanded into three-credit graduate-level semester courses, and the amount of material covered was essentially that in the current book. Before undertaking the second course, most of the students had also taken a course dealing with numerical methods in engineering. While the material was developed and intended to complement lectures in this subject area, it should also be useful to the practicing engineer. There are numerous example problems throughout the book that elucidate principles, formulate and set up problems, and/or apply techniques of problem solutions. Thus, the book is also intended for self-study for those who have taken courses in fluid mechanics and hydraulics.

The basic principles of conservation of mass, energy, and momentum are emphasized in the hope that this will help students master this important subject rather than just learn routine techniques in solving the large host of open channel applications. In so doing, students will enhance and enlarge their understanding of the fundamental principles of fluid mechanics and apply them in solving complex real problems. This emphasis is accomplished by devoting an entire chapter (Chapter 2) to the energy principle as it applies to open channel flow. Chapter 3 is devoted to the momentum principle, but since energy and conservation of mass have been covered previously, all three principles are used in setting up and solving problems. (Since the principle of conservation of mass is relatively easily implemented in solving open channel problems, a separate chapter is not devoted to it.) Many of these equations are nonlinear, and therefore numerical means for solving them are covered in addition to the open channel hydraulics. Real channels generally do not consist of a single channel of constant size, but rather a series of channels with different sizes and control structures, and/or parallel systems. Therefore, in dealing with these principles, they are applied repeatedly to link the equations together, which must be solved simultaneously to obtain depths, velocities, and flow rates throughout channel systems. Again, numerical means for accomplishing the solution for a system of nonlinear equations are covered, and the techniques for accomplishing such solutions are documented through computer codes and program listings. The progression from a single channel to a multichannel system is a distinguishing feature that sets this book apart from other books on this subject.

Seldom is the flow in real channels uniform, that is, the depth varies with position along the channel. Except near control structures, these variations in depth can be handled as gradually varied flow, that is, the flow is assumed to be one-dimensional, or the dependent variables are only a function of the position along the channel. Such gradually varied flows are described by an ordinary differential equation (ODE), for which closed-form solutions are only possible using very restrictive assumptions and, therefore, seldom apply in practice. The longest chapter in the book, Chapter 4, deals with gradually varied flows. It begins by deriving the general gradually varied flow equation, and documents numerical methods for solving this first-order ODE. Computer codes based on mathematical numerical methods rather than the traditional standard step method for solving a single ODE are provided, and these are then applied to solve a variety of problems associated with upstream and downstream controls, side weirs, etc. Again, as in previous chapters, after the student is thoroughly familiar with how gradually varied flow in single channels can be solved, he or she is shown how to set up and solve gradually varied flows in a system of channels.

This instruction involves numerical methods for solving systems of ordinary differential and nonlinear (with some linear) algebraic equations simultaneously. To assist in the instruction, computer codes and/or computer programs are provided. Following the line-by-line instructions for the computer to follow is in fact a most effective means of learning how such complex problems can be solved. The setting up of simultaneous algebraic equations in Chapters 2 and 3 provides the basis for setting up the equations that govern gradually varied flows in channel systems. The extension is that now, not only are there algebraic equations involved, but also ODEs. Commonly available software packages are not capable of solving combined systems of ODEs and algebraic equations without "add-in." Thus, much of the material in Chapter 4 is not available in other textbooks on this subject.

The variety of these complex problems is almost unlimited, but in an attempt to provide the student with the tools needed to set up and solve a particular problem that he or she may encounter, a large number of example problems are provided as an integral part of the text. The solutions to these example problems generally contain numerous computations and therefore require the use of a computer. Thus, many of the example problems contain computer programs. There are also a large number of homework problems at the end of the chapters. These problems provide the student not only with experience to solve problems somewhat similar to the example problems, but require him or her to also apply the principles to solve problems that expand upon the text material.

The material covered in Chapters 1 through 4 assumes that the channel's geometry is rectangular, trapezoidal, circular, or can be defined by simple parameters such as bottom width, side slope, diameter, etc. Chapter 5, entitled "Common techniques used in practice and controls," describes how the geometry of natural channels can be defined using a table of xy values for its bottom shape, and how quantities such as areas, perimeters, and top widths can be obtained from this data rather than just by solving an equation. It then covers water measurements in open channels, gates, and transitions, and concludes with a section dealing with total least cost design of channels.

The last two chapters, Chapters 6 and 7, deal with unsteady flow. Chapter 6 derives the various forms of unsteady flow equations for one-dimensional flow, that is, the St. Venant equations, and describes their characteristics. By assuming that the difference between the slopes of the energy line and channel bottom is the same, these unsteady flow equations can be solved along characteristic lines; this concept can then be used to obtain solutions to a variety of problems with upstream and downstream controls. Such simplified solutions provide the student with a good understanding of unsteady channel flow, and become almost indispensable in setting up the complete unsteady flow equations for a variety of problems, as described in Chapter 7. Initially, the material also included a follow-on chapter dealing with solutions of the two- and three-dimensional unsteady flow equations, but that has now been deleted. These two- and three-dimensional equations are now only derived.

I would like to express my heartfelt gratitude to Dr. Oulhaj Ahmed at the Institut Agronomique et Véterinaire Hassan II, Rabat, Morocco, who has translated this book into French and used it as course material for years now. He had translated material years ago, and recently updated that translation to include this book. The French version of this book is available at the Institut Agronomique et Véterinaire Hassan II, Rabat, Morocco.

An electronic "User's Manual" is also available on the CRC Press Web site (www.crcpress.com) that contains the solutions to the homework problems that are located at the end of each chapter. To obtain these solutions, please contact the publisher (Taylor & Francis Group).

I wish to thank the many students who have participated in this course and in other courses. The satisfaction and joy associated with teaching are truly enormous, and the enthusiasm of students adds much thereto. It is difficult to think of any profession that is as rewarding as teaching at a university. I sincerely hope that this book will contribute to the important subject of open channel flow and the use of numerical methods in engineering practice.

For MATLAB® and Simulink® product information, please contact

The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA, 01760-2098 USA
Tel: 508-647-7000
Fax: 508-647-7001
E-mail: info@mathworks.com
Web: www.mathworks.com

GUIDELINES FOR STUDYING THIS BOOK

The details of learning and internalizing a new subject until it becomes an integral part of your working knowledge cannot be described here. Rather, as a university student, or a professional, by now you should know how you learn best. Unless you are genius, it is unlikely that you will learn the subject of open channel flow by just reading, as you would read a newspaper, a magazine such as *National Geographic*, or a novel. Rather you will have to acquire the ability to set up and solve problems. Therefore, in these guidelines, I will suggest objectives and goals that you should set, and establish a means for measuring whether these goals are being systematically met. In other words, you will not really understand the subject of open channel flow unless you are able to set up mathematical equations that properly describe an open channel flow system and then solve those equations. To help in this process, this book contains numerous example problems, and an even larger number of homework problems. Some of these problems are what one might call routine, for example, problems that simply deal with solving a specified equation for a specified unknown variable. Other problems require that you identify the basic principles that apply, and how the proper application of these principles produces the system of equations that needs to be solved simultaneously to get the numerical values that describe how the channel system will perform under given conditions. It is hoped that this course will help you appreciate how the mathematics and mathematical methods that were developed mainly for purely theoretical reasons in your previous courses on linear algebra, calculus, and differential equations, suddenly assume great importance when dealing with engineering problems.

In your previous math courses, practically all of the learning was associated with linear equations and differential equations for which "closed-form" solutions are possible. Real engineering problems are, most frequently, governed by nonlinear and differential equations that must be solved using numerical techniques. This is certainly true for open channel flow. In brief, engineering mathematics comes down, ultimately, to numerical results, for example, numbers that define a problem's properties, such as depth, velocity, flow rate, force, etc., and the variations of these quantities in space and time. Consequently, much of what you will learn during this course deals with numerical methods. You will find that the material in the appendixes, especially Appendixes B and C, will need to be studied in detail, and fully mastered. Since numerical methods require a large amount of number crunching, beyond what can be practically accomplished by hand, it will be vital that you use a computer to solve many problems. Thus, plan on making your computer a heavily relied upon workhorse in studying this subject. It is for good reason that the title of this book contains not only open channel flow, but also numerical methods and computer applications.

Throughout the text there are listings of computer programs in Fortran, C (or C++), and math applications software such as Mathcad and TK-Solver that are used to solve example problems and to illustrate concepts. A folder on the CD-ROM on the back cover of this book contains MATLAB® programs that accomplish the same tasks of many of the Fortran programs. You might not be thoroughly familiar with these languages and/or applications for obtaining the numerical solutions, but it is relatively easy to gain sufficient understanding of Fortran (or C++) to follow the logic and computations needed to implement solutions. You will discover that it is often much easier to fully understand how principles and equations are used to solve problems by studying these programs than by just reading the text. The listings of these programs will be a very important tool for you to accomplish the goals (as listed below) that you should establish for yourself during the courses you take using this book.

Most of our bodies of knowledge in engineering, and its practice, are based on, relatively, a few fundamental principles and/or laws. In open channel flow, the three underlying principles are (1) conservation of mass, (2) conservation of energy, and (3) conservation of momentum. A thorough and comprehensive understanding of these principles is vital to properly understanding open channel flows and to solving complex channel systems. The proper application of these relatively simple concepts often entails considerable insight; therefore, devote time and effort in understanding these principles completely, and how they are used in setting up the equations that provide solutions to a wide variety of open channel situations. The problems solved in Chapters 1 and 2 use only the first two of these principles and gets you acquainted with equations such as Manning's equation and Chezy's equation, which describe how energy is dissipated due to fluid friction. Chapter 3 adds the momentum principle to your working tools. Chapter 4 and Appendix C devote much space to solving first-order ODEs, because the vast majority of steady-state open channel flows are mathematically described by such equations. The ability you acquire in numerically solving ODEs related to open channel flows will enhance your ability to cope with many other engineering problems, since ODEs are a most important body of knowledge in engineering. The first five chapters assume that flow conditions do not change with time, that is, they deal with the subject of steady-state open channel flow. The last two chapters are devoted to unsteady flows in open channels. Initial conditions, or what the flow consists of at time zero, require that steady-state solutions be obtained. Thus, you will need to have a good understanding of Chapters 1 through 5 before beginning Chapter 6. Since there is a semester's amount of study (or more) in Chapters 1 through 5, to study the material in this book will take at least two semesters of graduate-level course work. In fact, the last time I taught the last two chapters to PhD students in Fluid Mechanics Hydraulics Program at Utah State University, Logan, Utah, it took me two semesters, the second at the request of the students, so they could solve more general unsteady open channel problems.

Now, let us define goals you should set for yourselves. Below, only the goals for the first portion of the book will be outlined, that is, for steady-state flows. After studying this portion of the book, you should be able to set your goals associated with solving unsteady open channel flow problems. (You should repeatedly read these goals and assess your progress in completing them.)

Goal 1: Develop computer software that will solve for any of the variables associated with Manning's equation for (a) rectangular channels, (b) trapezoidal channels, (c) circular channels, and (d) natural, or irregularly shaped, channels. A vital part of this goal is to understand why the Newton method works and how it is implemented to solve for variables that cannot be placed on the left side of the equal sign by manipulating an equation, for example, solving implicit equations.

Goal 2: Develop a similar software that solves the combined Chezy and Chezy-C equations. The simultaneous solution of these two equations can be considered a more fundamentally sound approach to open channel flow than using the empirical Manning's equation. A similar comparison in pipe flow is use of the Darcy–Weisbach equation (with the friction factor therein being a function of the relative roughness of the pipe wall, and the Reynolds number associated with the flow) versus use of the empirical Hazen–Williams equation.

Goal 3: Develop computer software that will solve the energy equation (and energies equate at two different positions) for any of the variables associated therewith for (a) rectangular channels, (b) trapezoidal channels, (c) circular channels, and (d) natural, or irregularly shaped, channels.

Goal 4: Develop computer software that will solve the critical flow equation for any of the variables associated therewith for (a) rectangular channels, (b) trapezoidal channels, (c) circular channels, and (d) natural, or irregularly shaped, channels.

Goal 5: Develop computer software that will solve the momentum equation (and momentum functions equated at two different positions) for any of the variables associated therewith for (a) rectangular channels, (b) trapezoidal channels, (c) circular channels, and (d) natural, or irregularly shaped, channels.

Goal 6: Combine the solution capabilities of Goals 1 through 5 into a single software package that is easy to use by allowing you to select the type of problem you are solving, and what variable(s) is (are) to be solved.

Upon completing these first six goals, you will have developed a program similar to program CHANNEL that is available on the CD-ROM on the back cover of this book. You might wish to use your software to solve the problems given at the end of Chapter 3 "Problems to solve using program CHANNEL."

In addition to, or in conjunction with, the achievement of the above goals you should establish and complete the following goals:

Goal 7: Using the energy equations and/or critical flow equations associated with channel systems that consist of branched and parallel channels, write out the system of equations that describe the flow rates, velocities, and depths throughout the system.

Goal 8: Use linear algebra in combination with the expanded Newton method to solve the system of equations from Goal 7.

Goal 9: Use the momentum principle, in addition to the energy and critical flow equations, to define and solve problems involving branched and parallel channel systems that have controls such as gates that cause hydraulic jumps to occur.

The above goals deal with flow situations in which the depth does not vary with position along the channel, except in the immediate position of control structures, such as gates, intakes, etc., or, in other words, uniform flows occur. Similar goals need to be set to handle gradually varied flows (GVFs), or situations in which depths, velocities, and possibly flow rates vary with position along the channel. These latter types of flows are governed by ODEs, if steady state, or, if unsteady, by partial differential equations.

Goal 10: Become thoroughly familiar with the general ODE that defines GVFs, which allows for lateral inflow/outflow and changing channel size and shape, and how this equation simplifies depending upon the conditions.

Goal 11: When possible solve the GVF equation by numerical integration, otherwise learn and obtain numerical solutions of this equation using techniques designed to solve ODEs, and apply these numerical methods to solving GVF problems in single channels. Part of this goal should be to develop computer software that implements the solutions.

Goal 12: Be able to write out the system of equations that define channel systems that involve both algebraic equations and ODEs.

Goal 13: Solve these combined systems of algebraic equations and ODEs using the Newton method in combination with numerical solutions of ODEs. Again, part of this objective is to develop computer software to obtain the solutions. This later software (or computer program), because of the variety of equations involved, will not be a general program, but will need to be modified to handle different given situations.

Chapter 4 is devoted to helping you achieve Goals 10 through 13. Because the subject of GVF is more complex, and the type of problems more varied, Chapter 4 is longer than Chapters 1 through 3 combined.

Computer Programs with Listings of Code

1 Dimensions, Terminology, and Review of Basic Fluid Mechanics

1.1 INTRODUCTION

Open-channel flow is distinguished from closed-conduit flow by the presence of a free surface, or interface, between two different fluids of different densities. The two most common fluids involved are water and air. The presence of a free surface makes the subject of open-channel flow more complex, and more difficult to compute commonly needed information about the flow, than closed-conduit flow, or pipe flow. In pipe-flow problems, the cross-sectional area of the flow is known to equal the area of the pipe. In open-channel flow, the area depends upon the depth of flow, which is generally unknown, and must be determined as part of the solution processes. Coupling this added complexity with the fact that there are more open-channel flows around us than there are pipe flows, emphasizes the need for engineers, who plan work in water-related fields, to acquire proficiency in open-channel hydraulics. The wide use of computers in engineering practice reduces the need for graphical, table lookup, and other techniques learned by engineers who received their training a decade ago.

Technical fields apply very specific meanings to words. With a knowledge of these meanings it is possible for individuals educated in that field to communicate much more effectively and, with fewer words, convey clear and concise information. Often these words have a more general, less concise, meaning in their general use and, therefore, have a less concise meaning for the public at large. Other words are coined especially for a technical discipline. This chapter introduces the terminology used in open-channel flow. It is important that some terminology be fully mastered to effectively read and understand the remainder of this book, and to converse verbally, or in writing with open-channel hydraulic engineers.

1.2 ONE-, TWO-, AND THREE-DIMENSIONAL FLOWS

The dimensionality of a flow is defined as the number of independent space variables that are needed to describe the flow mathematically. If the variables of a flow change only in the direction of one space variable, e.g., in the direction along the channel x, then the flow is described as **one dimensional**. For such flows, variables such as depth Y and velocity V are only functions of x, i.e., Y(x) and V(x). If the variables of the flow change in two directions, such as the position along the channel x, and the position from the bottom of the channel y, or the position across the channel z, then the flow is described as **two dimensional**. For two-dimensional flows the mathematical notation of variables contains two arguments such as V(x,y) and Y(x,y), or V(x,z) and Y(x,z). If the variables of the flow change in three directions, such as the position along the flow, with the vertical position within the flow, and the horizontal position across the flow, then the flow is **three dimensional**. If a Cartesian coordinate system with axes x, y, and z (note here that lower case y is not the depth of flow, Y) is used, then three-dimensional flows are described mathematically by noting that the variables of the flow are a function of all three of these independent variables, or the velocity, for example, is denoted as V(x,y,z), to indicate that its magnitude varies with respect to x, y, and z in space, and since velocity is a vector, its direction also depends on x, y, and z. If the velocity also changes with time, this additional dependency will be denoted by V(x,y,z,t).

A two-dimensional flow that changes in time would have its velocity described as V(x,y,t) or V(x,z,t), and thus depends upon three independent variables and, from a mathematical point of view, is three dimensional. However, in fluid mechanics such flows are called **two dimensional, unsteady**. A flow with V(x,z) is called **two dimensional, steady**. The notation for the variables of a **one-dimensional, unsteady** flow consists of Y(x,t), V(x,t), etc.

Since the computations needed to solve a one-dimensional problem are much simpler than a two-dimensional steady problem, we wish to define the flow as one dimensional, if assuming this does not deviate too much from reality. The equations describing one dimensional, non-time-dependent or steady-state flows, are either just algebraic equations, or ordinary differential equations, whereas equations needed to describe two-dimensional flows are almost without exception partial differential equations. A one-dimensional flow that does change with respect to time must be described mathematically by partial differential equations also. Furthermore, because of the complex nature of the two- and three-dimensional flow equations that describe real flows, a very small number of closed form solutions to the mathematical boundary value problems governed by these partial differential equations are available for theoretically simple flows. Therefore, it is necessary to resort to approximate numerical methods to solve general two- and three-dimensional flows.

The term **hydraulics of open channel** flow is often used for one-dimensional free surface flow. The assumption made, that allows the flow in an open channel to be defined as one dimensional, is that the average velocity at a cross section can be used, and it is not necessary to be concerned with variations of the velocity with depth, or position across the flow. Based on this assumption the velocity, V(x) at any position along the channel equals the flow rate Q at this section divided by the cross-sectional area. Since the velocity varies from the bottom of a channel to the top at any position, and this velocity distribution may vary across the channel, correction factors are sometimes utilized to provide more accurate values of the kinematic energy per unit weight, and the momentum flux when the average velocity is used in the appropriate formula. However, the assumption is retained that the flow is one dimensional. These correction factors are discussed later in this chapter.

1.3 STEADY VERSUS UNSTEADY FLOW

A fluid flow is steady if none of the variables that can be used to describe the flow change with respect to time. Mathematically, steady flow is described by having partial derivatives of such variables as the depth of flow, the velocity, the cross-sectional area, etc., with respective to time all equal to zero, e.g., $\partial Y/\partial t = 0$, $\partial V/\partial t = 0$, $\partial A/\partial t = 0$.

A fluid flow is unsteady if any of the variables that describe the flow changes with respect to time. Thus a flow is unsteady if the depth at a given position in an open-channel flow changes with respect to time. Mathematically, a flow would be determined to be unsteady if any of the partial derivatives of any variable that describe the flow such as the depth, the velocity, the cross-sectional area, etc., with respect to time is different from zero. Generally, if one variable changes with respect to time all variables do.

Since for steady flows the variables that describe the flow are not a function of time, the independent variable t is not included in describing the flow, mathematically. If the flow is one dimensional in space, then the variables of the flow are only a function of that space variable. Thus the depth varies only as a function of the position variable x, or Y(x). If the flow is unsteady then the dependent variable Y varies with x and t, and this is denoted as Y(x,t). For a three-dimensional time-dependent flow, the depth is a function of the Cartesian coordinate system x,y, and z as well as time, and therefore the depth is denoted mathematically as Y(x,y,z,t).

1.4 UNIFORM VERSUS NONUNIFORM FLOW

A flow is **uniform** if none of the variables that describe the flow vary with respect to position, x along the channel. Therefore, the flow is uniform if the depth, velocity, and cross-sectional area are all constant. When dealing with one-dimensional open-channel hydraulics, uniform flows are not dependent on x. In fact, there is not such a thing as one-, two-, or three-dimensional **uniform flow** since the variables of the flow do not change with respect to x, y, or z. In theory, there could be a time-dependent uniform open-channel flow, but in practice such flows never actually exist. To have a uniform unsteady flow, the depth would have to increase (or decrease) at the same rate throughout the entire length of channel so that at all times neither the depth nor the velocity changes with respect to position, but is constantly changing with time. Mathematically, a flow is uniform if $\partial Y/\partial x = 0$, $\partial V/\partial x = 0$, $\partial A/\partial x = 0$, etc., throughout the flow. Setting partial derivatives with respect to the position x equal to zero, and having this zero occur throughout the flow is synonymous with stating that there is no dependency upon x.

A flow is **nonuniform** if any of the variables of the flow vary from position to position. Thus, in a one-dimensional channel flow, if the depth either increases or decreases from one position to another in a channel, the flow is nonuniform. A nonuniform flow may be one, two, or three dimensional, and may be either steady or unsteady. The flow over the crest of a dam's spillway is nonuniform but steady if the flow rate does not change. If, however, the flow rate is either increasing or decreasing with time then the flow is nonuniform and unsteady.

Nonuniform flows will be further subdivided into **gradually varied**, **rapidly varied**, and **spatially varied**. When the radius of curvature of the streamlines is large, e.g., the streamlines are nearly straight, such that the normal component of acceleration can be ignored, a nonuniform flow will be referred to as **gradually varied**. In a gradually varied open-channel flow the pressure will increase in the vertical direction just as it does in the same fluid for a uniform flow. This variation of pressure with depth is hydrostatic. A **rapidly varied** flow occurs when the change in depth is too rapid to ignore the normal acceleration component of the flow, and the pressure distribution is not hydrostatic.

Another way of looking at the difference between gradually varied and rapidly varied flow is that it is possible to use one-dimensional hydraulic equations for gradually varied flows, but rapidly varied flows are two dimensional, or three dimensional. Because of the complexities involved in solving two- and three-dimensional open-channel flows, often rapidly varied flows are handled by utilizing one-dimensional open-channel equations that are modified by experimental coefficient that account for the deficiencies in the one-dimensional assumption. This utilization of experimental coefficients distinguished **one-dimensional hydraulics** from pure fluid mechanics.

There is no parameter of measurement of the radius of curvature, or other characteristic of the flow with a threshold value that separates a gradually varied from a rapidly varied flow. Rather the distinction is subjective. Almost everyone would agree that the flow over a dam's spillway crest is rapidly varied, whereas the flow in the channel upstream from the dam is gradually varied. Likewise the flow immediately downstream from a sluice gate, where the flow is contracting rapidly from the gate height, is rapidly varied whereas the flow both upstream and further downstream from the gate is gradually varied until the position downstream from the gate where a hydraulic jump occurs, should this be the case. The flow through the hydraulic jump is rapidly varied, again. Whether a flow through a transition between two channels of different sizes is rapidly or gradually varied may be debatable. The classification will depend upon how rapidly the transition changes the channel's cross section, and how accurate it is necessary that the computed results correspond to the actual flow characteristics. An abrupt enlargement or an abrupt contraction will cause a small section of rapidly varied flow to occur. Whereas the solution of one-dimensional hydraulic equations for gradually varied flow may represent an accurate method for solving the variation of depth across a 50 ft long smoothly formed transition. If so, the flow is gradually varied.

Spatially varied flows are those portions of the main channel flow over which either lateral inflow or lateral outflow occurs. Therefore, in a spatially varied flow, the flow rate changes with position along the channel. Strictly speaking, the joining of two channels creates a spatially varied flow for a short distance. Generally, however, spatially varied flows occur where the lateral inflow or outflow is over some length of channel. A side weir that runs parallel to the direction of the channel over which discharge occurs creates a section of spatially varied flow. In this case spatially varied flow has distributed outflow from the channel, and the flow rate decreases in the direction of the main channel flow. Water accumulating from rainfall over a roadway surface and flowing into the gutters along the sides of the roadway causes a spatially varied open-channel flow in the gutters. In this case the channel flow in the gutter has a lateral inflow, and the flow rate increases in the direction of flow. As this gutter flow crosses the grates of a storm drain a spatially varied outflow occurs in the gutter, but in the storm drain that receives the flow from the grates, a spatially varied inflow occurs. If all the gutter flow can enter the storm drain then the channel flow in the gutter terminates at the end of the spatially varied flow.

1.5 PRISMATIC VERSUS NONPRISMATIC CHANNELS

Definitions that are closely associated with uniform and nonuniform flows, but apply to the channel rather than the flow in the channel, are **prismatic** and **nonprismatic** channels. A prismatic channel has the same geometry throughout its length. This may consist of a trapezoidal section, a rectangular section, a circular section, or any other fixed section. If the shape and/or size of the section changes with position along the channel, the channel is referred to as a **nonprismatic** channel. In theory, it is possible for a natural channel created by nature to be prismatic. However, in practice natural channels are nonprismatic.

1.6 SUBCRITICAL, CRITICAL, AND SUPERCRITICAL FLOWS

An open-channel flow is classified according to how the average velocity, V, of the flow compares with the speed, c, of a small amplitude gravity wave in that channel. If V is less in magnitude than c, then the flow is **subcritical**. If V is greater in magnitude than c, then the flow is **supercritical**, and if V = c, then the flow is **critical**. The speed of a small amplitude gravity wave is given by

$$c = \sqrt{\frac{gA}{T}} = \sqrt{gY_d} \tag{1.1}$$

where
g is the acceleration of gravity
A is the cross section of the flow
T is the top width of the flow
Y_d is the hydraulic depth, A/T

Subcritical flows behave differently from supercritical flow because in a subcritical flow the effect of downstream changes are noted by the fluid and it adjusts in anticipation of that downstream occurrence. Thus, if an obstruction exists in a subcritical flow the depth will gradually increase to the depth needed to pass by the obstruction. This signal that something exists downstream is propagated continuously to the upstream flowing fluid by gravity waves. These gravity waves can travel upstream because the velocity of flow is smaller than their speeds.

In supercritical flows the effect of changes cannot travel upstream because the velocity in the channel exceeds the propagation speed of gravity waves. Therefore, flow does not adjust itself for downstream conditions. For example, if a channel containing a supercritical flow ends abruptly, the depth at the end of the channel at the free overfall will be the same as if the channel had continued. If the flow were subcritical, the depth would decrease toward critical depth at the end of a free overfall, however.

As a consequence of whether gravity waves can move upstream or not, subcritical flows have their control "downstream," whereas supercritical flows are "upstream controlled." An example of both downstream and upstream control exists at a gate in a channel. Upstream from the gate the flow must be subcritical because the gate, which is downstream, controls the depth, velocity, area, etc., of the flow. Downstream from the gate the flow will be supercritical, and the gate determines the magnitude of the variables of the flow. If the gate is lowered, for example, it will decrease the downstream depth while increasing the downstream velocity, and has the opposite effect on the upstream flow.

The Froude number, F_r, is the ratio of the velocity in a channel divided by the speed of propagation, or celerity of a small amplitude gravity wave c, or

$$F_r = \frac{V}{c} = \frac{V}{\sqrt{gA/T}} = \sqrt{\frac{Q^2T}{gA^3}} \tag{1.2}$$

Therefore, the determination of whether a flow is subcritical, critical, or supercritical is commonly accomplished by computing the Froude number of the flow. If this value is less than unity then the flow is subcritical. If the Froude number is exactly equal to one, then the flow is critical, and if the Froude number is larger than unity, then the flow is supercritical. It turns out that the Froude number is also the ratio of inertia to gravity forces. A more in-depth treatment of subcritical and supercritical flows and their associated Froude numbers is given in subsequent chapters.

1.7 TURBULENT VERSUS LAMINAR FLOW

The Reynolds number, or the ratio of inertia to viscous forces, is used to distinguish whether a flow is laminar or turbulent. For open-channel flows the Reynolds number is defined by using the hydraulic radius, R_h, or the cross-sectional area A divided by the wetted perimeter P as the length variable. Thus the Reynolds number is defined by

$$R_e = \frac{VR_h}{\nu} = \frac{Q}{\nu P} = \frac{\rho VR_h}{\mu} \tag{1.3}$$

where
- V is the average velocity of the flow
- Q is the volumetric flow rate
- μ is the absolute viscosity
- ν is the kinematic viscosity of the fluid

If the Reynolds number is less than 500, the flow is laminar. Otherwise, the flow is turbulent. Often $4R_h$ is used as the length parameter in Reynolds number for channel flows because this is equivalent to the diameter of a pipe, e.g., the hydraulic radius of a pipe is $R_h = (\pi D^2/4)/(\pi D) = D/4$. When using this latter definition, the numerators on the left side of Equation 1.3 should be multiplied by 4.

Laminar flows are rare in open channels if the fluid is water. Examples of laminar flow might be the sheet flow over the surface of a watershed produced by precipitation, or the lateral flow over the crest of a roadway as it moves toward the side gutter. To have laminar water flow in an open channel, the depth generally has to be very small, in conjunction with a not too large velocity, since the kinematic viscosity of water is about 1.2×10^{-5} ft^2/s.

1.8 REVIEW OF BASIC FLUID MECHANICS PRINCIPLES

The rest of this chapter provides basic theory upon which the book is based. The presentation assumes that you are familiar with fluid mechanics, and therefore the remainder of this chapter should be considered a review; but this review is slanted toward open-channel hydraulics. Books dealing with engineering fluid mechanics will contain a more thorough treatment of this subject material.

The application of fluid mechanics in solving engineering problems involves a thorough understanding of the following four related items: (1) Physical properties of fluids, e.g., density, specific weight, viscosity, surface tension, and how these cause pressures to change, resistance to motion, etc. (2) The conservation of mass, or the continuity principle. (3) The conservation of energy and its dissipation into non-recoverable forms. (4) Utilization of momentum fluxes as vector quantities to deal with external forces on fluid in motion. A section for each of these four important subjects follows as the rest of this chapter. The specific application of these subjects to the flow of water in open channels constitutes the remaining chapters of this book. The review in this chapter will introduce the symbols that will be used through the rest of the book, and subsequent chapters are written assuming that you are acquainted with these symbols and their meanings. The problems at the end of this chapter have been listed under four similar headings. Problems given under subsequent headings generally also require an understanding of the principles involved in the previous headings.

1.9 PHYSICAL PROPERTIES OF FLUIDS AND THEIR EFFECTS ON OPEN-CHANNEL FLOWS

In solid mechanics, since the object being dealt with generally stays together its mass or weight is used when dealing with the effects it has on its environment. With fluids, however, total mass or total weight, generally, have no significance since these totals are directly related to the length of time the flow has been occurring. Rather, mass per unit volume or weight per unit volume are used. These quantities are the density ρ and the specific weight γ, or the fluid respectively. In the SI (International System of Units) the density ρ is given in kilograms per cubic meter, e.g., ρ (kg/m^3). In ES units (English System of Units) the density is in slugs per cubic foot, ρ (Slug/ft^3).

Specific weight γ is related to density through Newton's second law of motion,

$$\text{Force} = \text{Mass} \times \text{Acceleration}.$$

Weight is a force due to resisting gravity, and therefore $\gamma = \rho g$ in which g is the acceleration of gravity and equals 32.2 fps^2 in ES units or equals 9.81 m/s^2 when using SI units. (With more digits of precision g = 32.174049 fps^2 in ES units, and g = 9.80685 m/s^2 in SI units.) In SI units the specific weight is generally given as kilonewtons per cubic meter, e.g., γ (kN/m^3) and in ES units γ is given in pounds per cubic foot, e.g., γ (lb/ft^3). The density and specific weight for water vary moderately with its temperature as shown in Table 1.1.

The weight of a fluid causes pressure to increase with depth. If z is taken as the vertical coordinate, positive upward against gravity, from a selected datum and there is no motion in the fluid, then the pressure varies according to the hydrostatic law,

$$\frac{dp}{dz} = -\gamma = -\rho g \tag{1.4}$$

and if the fluid is incompressible (which is another way of indicating that γ and ρ are constant values), then this equation integrates to

$$p_2 - p_1 = \gamma(z_2 - z_1) = \rho g(z_2 - z_1) \tag{1.4a}$$

When the fluid is a liquid with a free surface it is convenient to use a coordinate h (e.g., independent variable) that has its origin on this surface and is positive downward. h is related to z by dh = −dz, and, therefore, Equations 1.4 and 1.4a become as following with p considered a function of h:

TABLE 1.1
Properties of Water Related to Temperature

Temperature		Density, ρ (Mass/Vol.)		Specific Weight (Weight/Vol.)		Abs. Viscosity, μ		Kinematic Viscosity, ν		Vapor Pressure, p_v	
						$\mu \times 10^3$	$\mu \times 10^5$	$\nu \times 10^6$	$\nu \times 10^5$		
°C	°F	kg/m³	Slug/ft³	kN/m³	lb/ft³	(N·s/m²)	(lb·s/ft²)	(m²/s)	(ft²/s)	kN/m² (abs)	psia
0	32.0	999.8	1.940	9.805	62.42	1.785	3.728	1.785	1.922	0.61	0.09
5	41.0	1000.0	1.940	9.807	62.43	1.518	3.170	1.518	1.634	0.87	0.13
10	50.0	999.7	1.940	9.804	62.41	1.307	2.730	1.307	1.407	1.23	0.18
15	59.0	999.1	1.939	9.798	62.37	1.139	2.379	1.140	1.227	1.70	0.25
20	68.0	998.2	1.937	9.789	62.32	1.002	2.093	1.004	1.080	2.34	0.34
25	77.0	997.0	1.934	9.777	62.24	0.890	1.859	0.893	0.961	3.17	0.46
30	86.0	995.7	1.932	9.764	62.16	0.798	1.667	0.801	0.863	4.24	0.61
35	95.0	994.1	1.929	9.749	62.06	0.720	1.504	0.724	0.780	5.62	0.82
40	104.0	992.2	1.925	9.730	61.94	0.653	1.364	0.658	0.708	7.38	1.07
45	113.0	990.2	1.921	9.711	61.82	0.596	1.245	0.602	0.648	9.58	1.39
50	122.0	988.0	1.917	9.689	61.68	0.547	1.142	0.554	0.596	12.33	1.79
55	131.0	995.7	1.932	9.764	62.16	0.504	1.053	0.506	0.545	15.73	2.28
60	140.0	983.2	1.908	9.642	61.38	0.466	0.973	0.474	0.510	19.92	2.89
65	149.0	980.5	1.902	9.615	61.21	0.433	0.904	0.442	0.475	25.00	3.63
70	158.0	977.8	1.897	9.589	61.04	0.404	0.844	0.413	0.445	31.16	4.52
75	167.0	974.9	1.892	9.561	60.86	0.378	0.789	0.388	0.417	38.53	5.59
80	176.0	971.8	1.886	9.530	60.67	0.354	0.739	0.364	0.392	47.34	6.87
85	185.0	968.6	1.879	9.499	60.47	0.333	0.695	0.344	0.370	57.58	8.35
90	194.0	965.3	1.873	9.466	60.26	0.315	0.658	0.326	0.351	70.10	10.17
95	203.0	962.2	1.867	9.436	60.07	0.294	0.614	0.306	0.329	79.36	11.51
100	212.0	958.4	1.860	9.399	59.83	0.282	0.589	0.294	0.317	101.33	14.70

$$\frac{dp}{dh} = \gamma = \rho g \tag{1.4b}$$

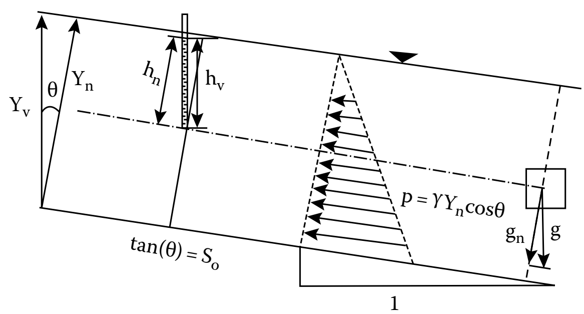
and

$$p = \gamma h = \rho g h \tag{1.4c}$$

In the last equation, the pressure p assumes that atmospheric pressure is the reference base pressure, and therefore p is gage pressure. Atmospheric pressure must be added to gage pressure to get absolute pressure.

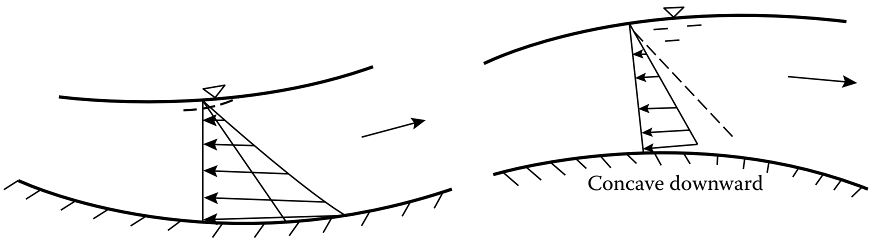
A few example problems follow that illustrate the use of Equations 1.4; however, it is worthwhile to examine how the pressure varies with depth in a channel with a steep bottom slope that contains a flow, before leaving the subject of pressure within a fluid caused by gravity. If the angle that the channel bottom makes with the horizontal is denoted by θ (note that the $\tan\theta = S_o$ the slope of the channel bottom), then the normal depth Y_n is related to the vertical depth Y_v by the cosine of this angle, or $Y_n = Y_v\cos\theta$, as shown in the sketch below.

Since gravity acts on fluid elements in the vertical direction, the height water would rise in a piezometer with its opening pointing downward, and located at any position will equal $\gamma h_n\cos\theta = g\rho h_n\cos\theta = \gamma h_v\cos^2\theta$, where h_n is the normal distance from the water surface down to the point being considered, and h_v is the vertical distance from the water surface to the point as shown on the sketch. Thus, the pressure at the bottom of the channel equals $\gamma Y_n\cos\theta = \gamma Y_v\cos^2\theta$.



An alternative, to the above method for determining the pressure distribution, is to note that the acceleration of gravity g acts in the vertical direction on any fluid element. The component of the acceleration in the normal direction is $g_n = g\cos\theta$, and therefore $p = \rho g_n h_n = \rho g h_n\cos\theta = \gamma h_v\cos^2\theta$.

Should the bottom of the channel be curved instead of having a constant bottom slope, then the normal acceleration in the fluid due to the curvature of the channel bottom will affect the pressure distribution as illustrated in the sketches below. The fluid at the very bottom of the channel will have a normal acceleration equal to v^2/r and will add to the gravitational acceleration when the curvature is concave upward and subtract from g when the curvature is concave downward. Note that since the fluid is not free falling under gravity's acceleration, upward accelerations add to g and downward accelerations subtract from g. The component of acceleration on a fluid particle in the direction of the radius of curvature equals $g\cos\theta + v^2/r$, where θ now is the angle between the vertical and the radius of curvature, and therefore at any point in the fluid $dp/dr = \rho(g\cos\theta + v^2/r)$. Since the radii of curvature of streamlines other than the bottom streamline will not, in general, equal the radius of curvature of the channel, it is not possible to determine the pressure distribution without making assumptions related to the radius of curvature between the channel bottom and the free surface.



The bulk modulus E_v is a similar quantity in dealing with fluids, as is the modulus of elasticity when dealing with solids. It has the same units as pressure and is the reciprocal of the compressibility. The bulk modulus equals the change in pressure needed to cause a decrease in volume divided by that decreased volume divided by the volume of fluid involved, or

$$E_v = -\frac{\Delta p}{\Delta \forall / \forall} = -\frac{dp}{dv/v} = \frac{dp}{d\rho/\rho} \tag{1.5}$$

The second and third parts of Equation 1.5 are obtained by taking one unit mass of a fluid so $\forall$ becomes the specific volume (volume/unit mass) v which is the reciprocal of the density ρ, and therefore $dv/v = -d\rho/\rho$. The bulk modulus of pure water equals 320,000 psi.

A small amount of free air entrained in the water can significantly reduce the bulk modulus of the mixture. If x is taken as the fraction of free air (not dissolved) mixed in water, then the density of the air–water mixture is given by $\rho_m = x\rho_a + (1 - x)\rho_w$ in which ρ_a and ρ_w are the densities of air and water, respectively. To obtain the bulk modulus of this mixture note from Equation 1.5 that $\Delta\forall = -\forall\, \Delta p/E_v$. The change in volume of the mixture $\Delta\forall_m$ will be the sum of the changes in volume of the air and the water or, $\Delta\forall_m = \Delta\forall_a + \Delta\forall_w$. By taking an original volume of a unit amount ($\forall = 1$), this last expression for the change in volume becomes the following upon substituting for the $\Delta\forall$'s:

$$-\frac{\Delta p}{E_m} = -\frac{x\Delta p}{E_a} = -\frac{(1-x)\Delta p}{E_w} \quad \text{or} \quad E_m = \frac{E_a E_w}{(1-x)E_a + xE_w}$$

where

E_a is the bulk modulus of air

E_w denotes the bulk modulus of water

Since the air mass in a water–air mixture generally is an extremely small fraction of the water mass, the air's temperature will remain constant when it is compressed. The compression of a gas at constant temperature is referred to as an isothermal process. The bulk modulus of a gas undergoing an isothermal process equals its absolute pressure, or $E_a = p_{abs}$. Therefore, the bulk modulus for a small fraction x of air entrained in water becomes

$$E_m = \frac{p_{abs} E_v}{(1-x)P_{abs} + xE_v} \tag{1.6}$$

where the symbol E_v has been used for the bulk modulus of water again instead of E_w. The table below indicates how the bulk modulus is affected by a small fraction of air. Note that a small amount of air entrainment reduces the density of the mixture very modestly, but reduces the bulk modulus by orders of magnitude. These values assume air at atmospheric pressure of 14.7 psia, and at a temperature of 60°F, so its density equals 0.00237 slugs/ft^3.

Fraction of air x	0.0000	0.0001	0.0005	0.001	0.005	0.01	0.10
Density of mixture ρ_m	1.940	1.940	1.939	1.938	1.930	1.921	1.746
Bulk modulus of mixture $E_m \times 10^{-2}$ (psi)	3,200	1,007	26.927	7.115	2.913	1.463	0.147

EXAMPLE PROBLEM 1.1

A sensitive pressure transducer is used to record the pressure at the bottom of a river that carries a sediment load that causes the density to increase linearly from 1.945 slug/ft_3 on the surface at a rate of 0.012 slug/ft^3/ft of depth. Develop the equation that gives the depth of flow from this pressure reading. What is the error if the sediment load is ignored, and the density taken equal to 1.94 and a pressure of 3.5 psi is recorded?

Solution

The problem is solved by defining $\rho = 1.945 + 0.012h$, substituting this into Equation 1.4b, separating variables, and integrating. The result is

$$p = g(1.945h + 0.006h^2),$$

where
- p is in pound per square foot
- h is in feet

Applying the quadratic formula gives the depth h as a function of the pressure reading as

$$h = -162.083 + 83.333\left(3.783 + 0.024\frac{p}{g}\right)^{1/2}.$$

Substituting $p = 3.5 \times 144$ into the above equation gives $h = 7.856$ ft. If $\rho = 1.94$ (constant), then $p = 62.4h$, or $h = 8.077$ ft, or an error of +0.221 ft.

EXAMPLE PROBLEM 1.2

Water is being drawn from a well whose water level is 1500 ft below the ground surface. Determine the significance of the compressibility of water in determining the head the pumps must supply if the friction loss in the well pipe is 25 ft, and the pump must supply 80 psi of pressure at the ground surface. Assume the bulk modulus for water remains constant and equal to 320,000 psi.

Solution

First, it is necessary to determine the relationship between density and pressure since the effects of increasing density of the fluid are to be taken into account. The definition of a fluid's bulk modulus $E_{\underline{V}}$ provides this relationship since $E_{\underline{V}} = -\Delta p/(\Delta\underline{V}/\underline{V}) = \rho dp/d\rho$, in which $\underline{V}$ is fluid volume and $\Delta\underline{V}$ is the change in this volume due to the pressure increase Δp. Separating variables in this equation and integrating the density from ρ_o (the density at atmospheric pressure, which will be taken as 1.94 slugs/ft^3) to ρ and integrating the pressure from 0 (atmospheric) to p gives

$$\frac{\rho}{\rho_o} = \mathrm{Exp}\left(\frac{p}{E_{\underline{V}}}\right)$$

(An alternative to integrating between the two limits is to just integrate and add a constant to the resulting equation. This constant can then be determined from a known condition. In this case the known condition is that the density equals $\rho_o = 1.94$ slugs/ft^3 when the pressure, $p = 0$.)

Substituting this expression into the hydrostatic Equation 1.4b gives

$$\frac{dp}{dh} = g\rho = g\rho_o\mathrm{Exp}\left(\frac{p}{E_{\underline{V}}}\right)$$

Again separating variables and integrating gives

$$p = -E_{\underline{v}} \mathrm{Ln}\left(1 - \frac{g\rho_o h}{E_{\underline{v}}}\right)$$

The value of h to use in this equation is the sum 1500 ft, the pressure head needed at the surface, $80 \times 144/(32.2 \times 1.94) = 184.41$ ft, and the frictional loss of 25 ft, or h = 1709.41 ft. Substituting this value for h in the above equation gives a pressure of 742.41 psi at the pump in the well. If the compressibility of the water is ignored, the pressure is obtained from $p = g\rho h/144 = 741.55$ psi. This small difference of about 1 psi points out that the compressibility of water is not very significant for most engineering applications.

An exception is whenever the speed of pressure waves are concerned, because for such applications a small amount of free air in water can dramatically change the speed of this wave as noted above.

EXAMPLE PROBLEM 1.3

Water is flowing at a rate of 800 cfs in a 6 ft wide rectangular channel at a constant depth of 4 ft. The bottom of the channel changes slope by means of a circular arc of radius 50 ft. The depth of flow through this arc remains constant at 4 ft. Assuming that the streamlines of this flow are concentric circles (have the same center of curvature), determine the pressure along a radial line at the beginning of the arc where it connects to the straight upstream channel that has a bottom slope $S_o = 0.15$, and also determine the pressure distribution along a vertical radial line.

Solution

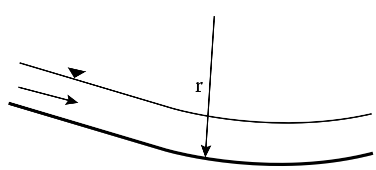
Using the upstream bottom slope $\theta = \arctan(0.15) = 8.531°$. The velocity in the channel equals 33.333 fps and therefore,

$$\frac{dp}{dr} = \rho\left\{g\cos(8.531°) + \frac{(33.333)^2}{r}\right\}$$

or after integrating p from 0 to p as r is integrated between 46 and r, this results in

$$p = \rho\left\{31.844r + 1111.11\ \mathrm{Ln}\left(\frac{r}{46}\right) - 1464.81\right\}$$

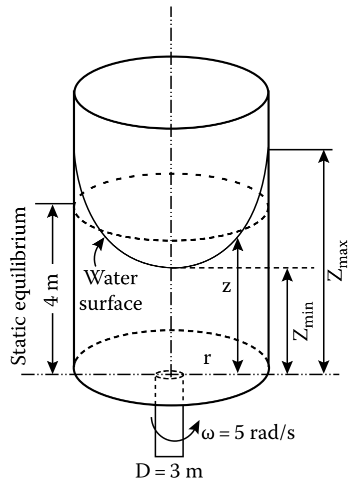
to give the pressure distribution as a function of r. On the bottom r = 50, and the pressure here is p = 426.75 psf = 2.964 psi, at the channel bottom, where r = 50 ft. Ignoring the added pressure due to the radius of curvature, the pressure at the bottom of the channel is $p = 4\gamma/144 = 1.73$ psi. This latter amount is 1.234 psi too small.



Flip-bucket spillways cause high-velocity water to have a considerable normal component of acceleration, as illustrated in this example. To find the added forces on such structures caused by changing the direction of the fluid can be handled easier through the use of the momentum principle that will be discussed in detail in Chapter 3.

EXAMPLE PROBLEM 1.4

A circular tank with a 3 m diameter (or radius $r_o = 1.5$ m) initially containing water to a depth of $h_o = 4$ m is rotated at an angular velocity $\omega = 5$ rad/sec for a long time until the water is brought into solid body rotation. Determine the shape of the water surface in the tank, and its depth at the center and outside walls of the tank.



Solution

In this problem the fluid (water) is being accelerated. The equation for fluid statics $\partial p/\partial z = -\rho g$ (with $\partial p/\partial x = 0$ and $\partial p/\partial y = 0$) which accounts only for gravitational acceleration in the z-direction (vertical) can be generalized to the following: $\partial p/\partial z = -\rho(g + a_z)$, $\partial p/\partial x = -\rho a_x$, $\partial p/\partial y = -\rho a_y$. When applied to the rotating tank, with z as the vertical coordinate, and r the radial coordinate, these equations become $\partial p/\partial z = -\rho g$ and $\partial p/\partial r = -\rho a_r = \rho r\omega^2$. From the definition of a differential $dp = (\partial p/\partial r)dr + (\partial p/\partial z)dz$. Along any constant pressure surface, such as the free surface, $dp = 0$, and therefore the slope of the water surface at any radial position r is $dz/dr = -a_r/(g + a_z) = r\omega^2/g$. Separating variables and integrating gives the following parabolic relationship between z and r for the water surface:

$$z = \frac{r^2\omega^2}{2g} + C$$

The constant C can be evaluated by noting that the same amount of water exists in the tank after rotation as before, or

$$\int_0^{r_o} 2\pi rz\,dr = 2\pi\int_0^{r_o} r\left(\frac{\omega^2 r^2}{2g} + C\right)dr = 2\pi\left(\frac{\omega^2 r_o^4}{8g} + \frac{r_o^2}{2}C\right) = \pi h_o r_o^2$$

or solving for C

$$C = h_o - \frac{r_o{}^2\omega^2}{4g} = 4 - \frac{(1.5\times 5)^2}{4(9.81)} = 4 - 1.433 = 2.567\,\text{m}.$$

At the center axis of the tank the water depth will be $z_{in} = C = 2.567$ m, and at the outside $z_{max} = C + (r_o \times \omega)^2/(2g) = 2.567 + 2.867 = 5.433$ m. As a check on the computations, etc., we might note that if the volume of the cylinder formed by z_{max} as its height has the volume of the paraboloid subtracted from it, then the original volume of the cylinder with a height h_o should occur. The volume of a paraboloid equals 1/2 the area of the base time the height. Therefore, $Ah_o = A\{z_{max} - 0.5(z_{max} - C)\} = A(z_{max} + C)/2$, or $h_o = (z_{max} + C)/2 = (5.433 + 2.567) = 4$ m.

Before leaving this problem, it is worth noting that the Bernoulli equation cannot be applied across the streamlines in this rotation tank, e.g., from the inside radius to the outside. If this were done then the total head on the free surface would be the sum of the elevation and velocity heads (since p = 0) or

$$z + \frac{V^2}{2g} = z + \frac{(r\omega)^2}{2g} = H$$

or

$$z = H - \frac{r^2\omega^2}{2g},$$

but this equation has a constant minus the velocity head whereas the above equation indicated that z was equal to a constant plus the velocity head. The reason is that the Bernoulli equation is based on irrotational flow, whereas this is rotational flow. A free vortex represents irrotational flow, but its equation indicates that the transverse component of velocity v_t = Constant/r. The forced rotation in the tank gives $v_t = r\omega = r \times \text{Constant}$.

EXAMPLE PROBLEM 1.5

Water is flowing in a natural channel, and the flow rate Q is to be determined by measuring the velocity on the surface and the depth of flow at various positions x across the channel. These measurements have produced the values in the table below.

Position, x (m)	0.0	4	8	12	16	20	24	28	32	36
Depth, Y (m)	0.0	4.7	8.3	11.2	12.7	13.3	13.4	12.8	12.0	11.8
Velocity, Vs (m/s)	0.0	0.35	0.45	0.47	0.49	0.50	0.49	0.50	0.48	0.48
	40	44	48	52	56	60	64	68	72	75
	11.5	11.7	12.3	13.3	14.5	15.0	14.8	12.7	8.0	0.0
	0.48	0.48	0.48	0.49	0.50	0.50	0.45	0.35	0.20	0.0

It has also been determined that the velocity varies according to the same dimensionless profile from the bottom of the channel at any position x according to the data given in the table below. (In this table y′ = y/Y is the dimensionless depth, with y beginning at the bottom at this position x, and likewise v′ = v/Vs is the dimensionless velocity, in which Vs is the velocity on the surface at this position x.)

Dimensionless depth y′	0.0	0.04	0.08	0.12	0.20	0.28	0.40	0.52	0.64	0.72	0.80	0.90	1.0
Dimensionless velocity v′	0.0	0.22	0.375	0.500	0.675	0.815	0.955	1.040	1.070	1.068	1.06	1.03	1.0

Solution

There are a number of methods that could be used to solve this problem, including plotting the data from the above tables and determining area under the curves. However, since in this book we wish to emphasize the use of the computer to do numerical computations, the problem will be solved by writing a computer program. First, however, let us note that a dimensionless flow rate per unit width q′ can be obtained by integrating the dimensionless velocity profile given in the second table, or $q' = \int v'dy'$, with limit from 0 to 1. The flow rate per unit width q equals the dimensionless q′ multiplied by the depth Y and the surface velocity at any position, or $q = \int v\,dy = VsY\int v'dy' = VsYq'$. Thus, the flow rate Q, which is the integral of the unit flow rate q times dx, or $Q = \int q dx$, can be determined from $Q = q'\int VxY\,dx$, with the limits of this integration from 0 to the total width of the section, or 75 m. These integrations will be accomplished by using the subroutine SIMPR, which is described in Appendix B, and implements Simpson's rule to numerically evaluate integrals. To provide the integrand as a continuous function of the variable being integrated, a cubic spline function will be used, i.e., subroutine SPLINESU, also described in Appendix B is utilized. The program EXPRB1_5 (both in FORTRAN and C) is given below to provide the solution. The following are the key components of this program: (1) The first two READS store the data from the above two tables in arrays, with arrays YP and VP storing the dimensionless velocity profile data in the second table, and arrays X, Y, and VS storing the data in the first table. (2) The cubic spline function SPLINESU is called three times to provide (a) the second derivatives d^2v'/dy'^2, (b) the second derivatives d^2Y/dx^2, and (c) the second derivatives d^2Vs/dx^2 corresponding to the points in the tables. (3) The subroutine SIMPR is called next to integrate the dimensionless velocity profile. It calls on function subprogram VPROF to provide v' corresponding to any dimensionless depth y', and accomplishes this by using the d^2v'/dy'^2 supplied by the first call to SPLINESU. (4) The subroutine SIMPR is called again to provide the integral $\int(VsY)dx$, and subprogram Dq provides the arguments for this integration. (5) Finally by multiplying q' (the result from the first integration) by the result from the second integration the flow rate Q is printed out. (Note that the C program contains the function that supplies what is integrated as argument 1, as mentioned in Appendix B if the name is not equat used; thus allowing the two different arguments vprof (for v') and qp (for Y*Vs).)

Listing of program EXPRB1_5.FOR

```
      EXTERNAL VPROF,Dq
      REAL DUM(30)
      COMMON YP(20),VP(20),X(30),Y(30),VS(30),
     &D2VP(20),D2VS(30),D2Y(30),I1,I2,NP,NX
      I1=1
      I2=2
      READ(2,*) NP,(YP(I),VP(I),I=1,NP)
      READ(2,*) NX,(X(I),Y(I),VS(I),I=1,NX)
      CALL SPLINESU(NP,YP,VP,D2VP,DUM,0)
      CALL SPLINESU(NX,X,Y,D2Y,DUM,0)
      CALL SPLINESU(NX,X,VS,D2VS,DUM,0)
      CALL SIMPR(VPROF,0.,1.,qPRIM,1.E-6,20)
      WRITE(*,*)' Integral of dimensionless'/
     &' velocity profile=',qPRIM
      I1=1
      I2=2
      CALL SIMPR(Dq,0.,X(NX),Q,1.E-4,20)
      WRITE(*,*)' Flowrate, Q =',Q*qPRIM
      END
      FUNCTION Dq(XX)
      COMMON YP(20),VP(20),X(30),Y(30),VS(30),
     &D2VP(20),D2VS(30),D2Y(30),I1,I2,NP,NX
1     IF(XX.LT.X(I2) .OR. I2.EQ.NX) GO TO 2
      I1=I2
      I2=I2+1
      GO TO 1
2     IF(XX.GE.X(I1) .OR. I1.EQ.1) GO TO 3
      I2=I1
      I1=I1-1
      GO TO 2
      END
      FUNCTION VPROF(YY)
      COMMON YP(20),VP(20),X(30),Y(30),VS(30),
     &D2VP(20),D2VS(30),D2Y(30),I1,I2,NP,NX
1     IF(YY.LT.YP(I2) .OR. I2.EQ.NP) GO TO 2
      I1=I2
      I2=I2+1
      GO TO 1
2     IF(YY.GE.YP(I1) .OR. I1.EQ.1) GO TO 3
      I2=I1
      I1=I1-1
      GO TO 2
```

```
3     DYP=YP(I2)-YP(I1)
      A=(YP(I2)-YY)/DYP
      B=1.-A
      VPROF=A*VP(I1)+B*VP(I2)+((A*A-1.)*A*D2VP(I1)+
     &(B*B-1.)*B*D2VP(I2))*DYP**2/6.
      RETURN
3     DX=X(I2)-X(I1)
      A=(X(I2)-XX)/DX
      B=1.-A
      AA=A*(A*A-1.)*DX*DX/6.
      BB=B*(B*B-1.)*DX*DX/6.
      DEPTH=A*Y(I1)+B*Y(I2)+AA*D2Y(I1)+BB*D2Y(I2)
      VSURF=A*VS(I1)+B*VS(I2)+AA*D2VS(I1)+BB*
     &D2VS(I2)
      Dq=DEPTH*VSURF
      RETURN
      END
```

Listing of program EXPRB1_5.C

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float yp[20],vp[20],x[30],y[30],vs[30],d2vp[20],d2y[30],d2vs[30];
int i1,i2,np,nx;
extern float simpr(function equat,float xb,float xe,\
 float err,int max);
extern void splinesu(int n,float *x,float *y, float *d2y,\
 float *d,int ity);
float vprof(float yy){float a,b,dyp;
 while((yy>=yp[i2])&&(i2<np−1)){i1=i2;i2++;}
 while((yy<yp[i1])&&(i1>1)){i2=i1;i1−−;} dyp=yp[i2]−yp[i1];
  a=(yp[i2]−yy);b=1.−a;
 return a*vp[i1]+b*vp[i2]+((a*a−1.)*a*d2vp[i1]+(b*b−1.)*b*d2vp[i2])\
  *dyp*dyp/6.;
}  // End of vprof
float qp(float xx){float a,b,dx,aa,bb,depth;
 while((xx>=x[i2])&&(i1<nx−1)){i1=i2;i2++;}
 while((xx<x[i1])&&(i1>1)){i2=i1;i1−−;}
 dx=x[i2]−x[i1];a=(x[i2]−xx)/dx;b=1.−a;
 aa=a*(a*a−1.)*dx*dx/6.;bb=b*(b*b−1.)*dx*dx/6.;
 depth=a*y[i1]+b*y[i2]+aa*d2y[i1]+bb*d2y[i2];
 return depth*(a*vs[i1]+b*vs[i2]+aa*d2vs[i1]+bb*d2vs[i2]);
}  // End of qp
void main(void){FILE *fili; char filnam[20];int i;
 float dum[30];qprim,q;
 i1=1;i2=2; printf("Give input file name\n");
 scanf("%s",filnam);
 if((fili=fopen(filnam,"r"))==NULL){
  printf("Cannot open file\n");exit(0);}
 fscanf(fili,"%d",&np);
 for(i=0;i<np;i++)fscanf(fili,"%f %f",&yp[i],&vp[i]);
 fscanf(fili,"%d",&nx);
 for(i=0;i<nx;i++)fscanf(fili,"%f %f %f",&x[i],&y[i],&vs[i]);
 splinesu(np,yp,vp,d2vp,dum,0);splinesu(nx,x,y,d2y,dum,0);
 splinesu(nx,x,vs,d2vs,dum,0);
 qprim=simpr(vprof,0.,1.,1.e−6,20); i1=1;i2=2;
```

```
 printf("Integral of dimensionless velocity profile =%f\n",qprim);
 printf("Flowrate, Q =%f\n",qprim*simpr(qp,0.,x[nx-1],1.e-4,20));
}
```

The following is the input file needed to solve the problem (EXPRB1_5.DAT):

```
13 0. 0. .04 .22 .08 .375 .12 .5 .2 .675 .28 .815 .4 .955 .52
 1.04 .64 1.07 .72 1.068 .8 1.06 .9 1.03 1 1
20 0 0 0 4 4.7 .25 8 8.3 .45 12 11.2 .47 16 12.7 .49 20 13.3 .5
 24 13.4 .49 28 12.8 .48 32 12. .48 36 11.8 .48 40 11.5 .48 44
 11.7 .48 48 12.3 .48 52 13.3 .49 56 14.5 .5 60 15. .5 64 14.8
 .45 68 12.7 .35 72 8. .2 75 0. 0.
```

The solution produced is

Integral of dimensionless velocity profile = 8.726063e−01

Flowrate, Q = 342.505700

Viscosity is the fluid property that defines its resistance to motion. By definition, the absolute viscosity μ of a fluid is defined as the coefficient that relates the internal shearing stress τ within the fluid to the velocity gradient at this point in the flowing fluid, or

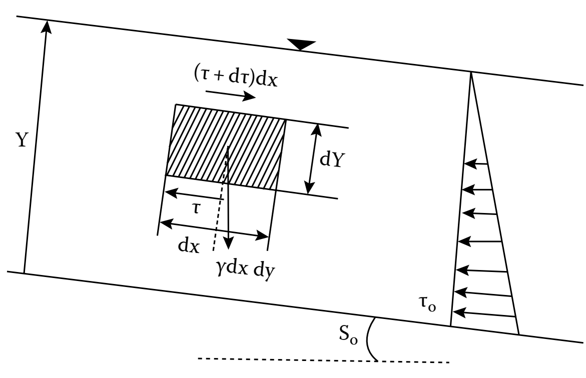
$$\tau = -\mu\left(\frac{\partial v}{\partial n}\right) \tag{1.7}$$

where n is a direction normal to velocity v. Fluids, such as water, whose absolute viscosities are not dependent upon the rate of shear, $\partial v/\partial n$, are referred to as **Newtonian fluids**. The viscosity of water does depend upon its temperature, as shown in Table 1.1. If the viscosity of a fluid does change with the rate of shear (μ is a function of $\partial v/\partial s$ or τ, i.e., $\mu(\partial v/\partial s)$ or $\mu(\tau)$) then the fluid is referred to as a **non-Newtonian** fluid. Silly putty is a non-Newtonian fluid that behaves similar to an elastic solid if sheared rapidly (i.e., bounces as a rubber ball), but flows as a very viscous fluid if given sufficient time to do this. Mud and debris flows, which are open-channel flows, of a fluid that is a mixture of water and soil (and possibly debris from the channel) that have mobilized are other examples involving non-Newtonian fluids.

The kinematic viscosity v equals the absolute viscosity divided by the fluid density, or

$$v = \frac{\mu}{\rho} \tag{1.8}$$

Effects that viscosity has on fluid motion can be examined by considering a small section of uniform, steady flow in a channel in which an element of fluid with a differential length dx, and a height dy is taken as shown in the sketch below.



The channel has a bottom slope equal to S_o and is wide enough that only the shearing stress on the bottom is important (the side shears can be neglected). Since the flow is steady and uniform, the summation of forces on the differential fluid element in the x-direction must equal zero. This summation produces the following:

$$-\tau dx + (\tau + d\tau)dx + S_o\gamma\, dx\, dy = 0$$

In this equation the slope of the channel bottom is small so that the $\sin\theta = \tan\theta = S_o$. Simplifying gives $d\tau = -S_o\gamma\, dy$, which can be integrated between the bottom of the channel to any depth y, or

$$\int_{\tau_o}^{\tau} d\tau = -\gamma S_o \int_{0}^{Y} dy$$

which gives

$$\tau = \tau_o - S_o\gamma y$$

An examination of the section of this flow from the bottom to the top with a length dx shows that $\tau_o = \gamma Y S_o$. For any arbitrary section in which the shear stress on the side is considered, it can be shown that

$$\tau_o = \frac{\gamma A S_o}{P} = \gamma R_h S_o \tag{1.9}$$

where
- A is the cross-sectional area
- P is wetted perimeter
- R_h is the hydraulic radius, A/P

which reduces to the above for the wide section being considered here. Substituting for τ_o gives the following equation that defines the shearing stress as a function of the position from the bottom y:

$$\tau = \gamma S_o (Y - y) \tag{1.10}$$

which shows the shearing stress varies linearly from $\gamma S_o Y$ at the bottom to zero at the top of the channel flow. Substituting Equation 1.10 into Equation 1.7 allows for the velocity distribution from the bottom to the top of the channel to be determined. Making this substitution gives

$$dv = \left(\frac{\gamma S_o}{\mu}\right)(Y - y)dy$$

which integrates to give the parabolic relationship,

$$v = \frac{\gamma S_o}{\mu (Yy - y^2/2)} \tag{1.11}$$

It turns out that Equation 1.11 is correct only for laminar flows in which the viscous forces dominate over the inertia forces. For a turbulent flow the fluid near the walls has its velocity increased, whereas the velocity near the top of the flow away from the solid boundaries has its velocity decreased from the velocity given by Equation 1.11 by mixing of the flows. In other words, the large velocity portion of the flow is continually being mixed with the small velocity portion of the flow tending to make the velocity distribution more nearly the same throughout the flow than that described by Equation 1.11. This mixing is referred to as momentum transfer within the flow, and results in the need to modify Equation 1.7 for turbulent flows. This momentum transfer causes the inertial forces to play a more important role in determining the distribution of the velocity with depth. For turbulent flows it is necessary to multiply the velocity gradient in Equation 1.7 by an additional eddy viscosity term ε to get the shearing stress, or μ is replaced by $(\mu + \varepsilon)$. This eddy viscosity varies with y in a manner that cannot be defined easily for different flow situations, and therefore has limited practical value.

Even though Equations 1.7 and 1.11 hold only for laminar flows, Equation 1.10 is valid for turbulent flows as well.

EXAMPLE PROBLEM 1.6

A trapezoidal channel with a bottom width b = 3 m, and a side slope m = 2 has a bottom slope $S_o = 0.0008$ and is 5000 m long. This channel contains a uniform flow of Q = 20 m³/s at a depth Y = 1.6 m. Determine the total shear force on the sides of this channel caused by the flowing water.

Solution

The shear stresses on the boundaries of this channel are given by Equation 1.9. Computing the area and wetted perimeter gives A = (b + mY), Y = 9.920 m², P = 10.155 m. Substituting into Equation 1.9 gives

$$\tau_o = 9800\left(\frac{9.920}{10.115}\right)(0.0008) = 7.658 \text{ N/m}^2.$$

The area over which this shear stress acts equals the perimeter times the length of channel or 10.155 × 5000 = 50,775 m². Therefore, the total shear force equals

50,775 × 7.658 = 388,848 N.

Alternatively, this shear force equals the component of the weight in the channel in the direction of the flow, or weight XS_o. This computation gives 9800(9.92 × 5000)(0.0008) = 388,848 N. It should be noted in this problem that the shear force does not involve using the flow rate. In actuality, it is the flow rate and the channel roughness that determines the depth of flow.

EXAMPLE PROBLEM 1.7

Water at a rate of Q = 25 m³/s flows around a circular bend in a rectangular channel with a width of 4 m. The inside radius of this bend is 4 m. If the depth upstream from the bend is $Y_o = 3$ m, determine the velocity distribution through a radial line through this bend under the following assumptions: (1) an average velocity through the depth of flow can be used so that the velocity can be considered only a function of r (distance from the center of the bend); (2) the shear stress is proportional to the velocity gradient dv/dr; and (3) the shear stress is zero along the circle midway between the two channel walls and varies linearly to τ_o at the two walls.

Solution

The solution starts by noting from the problem's description that $|\tau| = a(dv(r)/dr) = \tau_o(r-6)/2$, in which a is a constant. Integrating this equation and evaluating the constant of integration gives the following equation for the velocity distribution:

$$v = C\left\{\left(\frac{6r - r^2}{2}\right) - 16\right\} \text{ (Note that v = 0 when r = 4 and r = 8)}$$

To evaluate the constant C, the integral of vYdr is equated to the flow rate Q, or

$$Q = \int vY\,dr = 25$$

If the depth is assumed constant at Y = 3 m across the circular portion of the channel then

$$25 = 3C\int\left[6r - \frac{r^2}{2} - 16\right]dr = 3C\left[\frac{6r^2}{2} - \frac{r^3}{6} - 16r\right]\Bigg|_4^8 = 3C[5.33], \quad \text{or} \quad C = 1.563$$

The assumption of constant depth is not good since the centrifugal forces of the flow around the bend will cause the depth at its outside to be larger than at the inside. The slope of the water surface will be normal to the total acceleration vector, or $dY/dr = v^2/(gr)$. Substituting for v and integrating gives

$$Y = \left(\frac{C^2}{g}\right)\left\{\frac{r^4}{16} - 2r^3 + 26r^2 - 192r + 256\,\mathrm{Ln}(r)\right\} + C_1$$

Since the depth is assumed to equal 3 m when r = 6 m (the mid-radius of the bend), C_1 can be evaluated as

$$C_1 = 3 + \left(\frac{C^2}{g}\right)(108.31).$$

The above expressions for Y and v are again substituted in $25 = \int vY\,dr$ and this integrated from 4 to 8 with the result,

$$25 = 16C - 0.015982C^3$$

The three roots of this equation are 1.5663, –32.395, and 30.828. The first root is the desired one, i.e., C = 1.5663. The depths and velocities through the bend are given in the table below.

r	4	4.5	5	5.5	6	6.5	7	7.5	8
Y	2.800	2.808	2.847	2.917	3.000	3.077	3.131	5.157	3.162
V	0.000	1.371	2.350	2.937	3.133	2.937	2.350	1.371	0.000

1.10 CONSERVATION OF MASS, OR CONTINUITY EQUATIONS

A basic principle of the mechanics is that mass is conserved. When this principle is applied under steady-state flow conditions to one-dimensional fluid flows it is generally expressed that the mass flow rate or flux past a second section equals the mass flow rate past the first section plus the mass flow rate entering between the two sections, or

$$(\rho VA)_2 = (\rho VA)_1 + (\rho VA)_{1-2} = (\rho VA)_1 + \left(\int \rho vY\,dx = \int \rho q\,dx\right) \tag{1.12}$$

where

- the Vs represent the average velocities at the designated positions
- the As represent the corresponding cross-sectional areas normal to these velocities

When dealing with an incompressible fluid the density ρ is constant and can be divided out of Equation 1.12 to give a volumetric flow equation. The volumetric flow rate with dimensions of L^3/t will be given the symbol Q, so Equation 1.12 can be written in either of the following two forms:

$$Q_2 = Q_1 + \int q(x)dx$$

or

$$(VA)_2 = (VA)_1 + \int q(x)dx \tag{1.13}$$

where q(x) is the lateral inflow (negative for outflow) in units of flow rate per unit length (L^2/t) being equal to v_iY, and the integrations are over the length between the two sections 1 to 2. If the lateral inflow is constant, then the last term in Equation 1.13 can be simplified to qL_{1-2} where L_{1-2} is the distance between section 1 and 2.

The average velocity V in Equations 1.12 and 1.13 represents the point velocity integrated over the cross-sectional area of the flow divided by this area, or

$$V = \frac{\int v\,dA}{A} \tag{1.14}$$

When dealing with two- or three-dimensional flows, or an unsteady one-dimensional flow, the continuity equation becomes a partial differential equation rather than one of the above algebraic equations. The general equation that applies for three-dimensional unsteady flow of a compressible fluid is

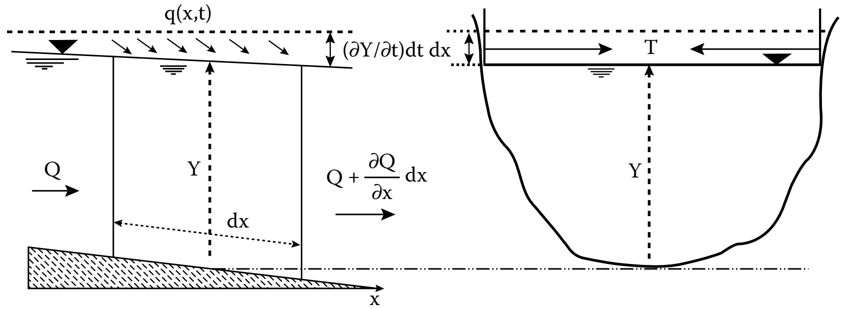
$$\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = -\frac{\partial(\rho)}{\partial t} \tag{1.15}$$

where u, v, and w are the velocity components in the x, y, and z coordinate directions, respectively. For either steady flow, or the flow of an incompressible fluid the right-hand side of this equation becomes zero. For an incompressible fluid the continuity equation reduces to

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{1.16}$$

The continuity equation for an unsteady one-dimensional flow in an open channel needs to apply for the entire cross section of the channel rather than at a point as is the case with Equations 1.15 and 1.16. To develop this equation consider a differential length of a channel flow as shown in the sketch below whose depth, etc. changes with time, as well as the position x along the channel. The principle of conservation of mass requires that the following, in relationship to this control volume, CV, be true:

$$(\text{Vol. in CV in time } \Delta t) - (\text{Vol. leaving CV in time } \Delta t) = \text{Vol. change in } \Delta t,$$



or substituting the appropriate quantities into this equation gives

$$\left[Q - \left\{Q + \frac{\partial Q}{\partial x}dx\right\} + q(x,t)dx\right]dt = T\left(\frac{\partial Y}{\partial t}\right)dt\,dx$$

Simplifying and dividing by dxdt gives

$$\frac{\partial Q}{\partial x} - q(x,t) + T\frac{\partial Y}{\partial t} = 0 \tag{1.17}$$

This equation, coupled with a second partial differential equation of motion, will be the basis for solving unsteady channel flows in Chapter 6. It is worthwhile noting that if $\partial Q/\partial x$ is of the same positive magnitude as the magnitude of lateral inflow q(x,t) at all points x and at all times, then $\partial Y/\partial t = 0$, i.e., the flow is steady state, which means at any position x the depth and velocity are constant. However, it is important to note that q and $\partial Q/\partial x$ represent two different quantities. q(x,t) is the lateral inflow that can in general vary along the channel as well as vary with time. It is generally known and not part of what is solved for. Should lateral outflow occur, then q(x,t) is a negative quantity. The term $\partial Q/\partial x$ represents the change in flow rate in the channel direction x for a given instant in time, and in general varies from instant to instant. Sometimes an additional subscript is applied to partial derivatives to denote what is being held constant. In Equation 1.17 the two independent variables are x and t. Therefore, it is understood that t is constant when dealing with partial derivatives with respect to x and x is held constant when the partial derivative is with respect to t. Q is unknown and therefore its variation with x and t is a part of the solution.

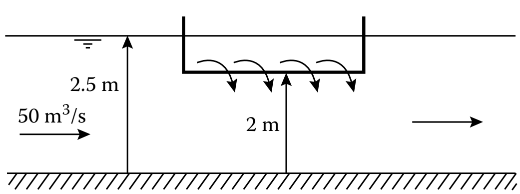
Since the volumetric flow rate Q equals the average velocity V times the cross-sectional area A, the first term in Equation 1.17 can be expanded to give the following alternative continuity equation for one-dimensional unsteady flow in a channel:

$$A\frac{\partial V}{\partial x} + V\frac{\partial A}{\partial x} - q(x,t) + T\frac{\partial Y}{\partial t} = 0 \tag{1.17a}$$

Equation 1.17 generally cannot be solved by itself because it involves two unknowns, or dependent variables, Q and Y. The principle that as many independent equations must exist as there are unknowns, dictates that another equation must be obtained. This second equation comes from Newton's second law of motion, and will be given later in Chapter 6. Equation 1.17 and this second equation are often referred to as the St-Venant equations that govern one-dimensional unsteady open-channel hydraulic problems.

EXAMPLE PROBLEM 1.8

A 10 m wide rectangular channel has a 10 m long section of side weir that discharges a volumetric flow rate per unit length as given by the equation $q = 0.5/2g\,H^{3/2}$ in which H is the depth of water in the channel above the weir crest. The weir crest is 2 m above the channel bottom, and the depth in the channel at the beginning of the side weir is 2.5 m, and the flow rate here equals 50 m³/s. If the depth of flow through the side weir is such that $V^2/2g + Y =$ constant, determine the flow rate in the channel over the length of the side weir.



Solution

Letting Q_o and Y_o be the flow rate and depth, respectively, at the beginning of the side weir the constant referred to can be determined from constant $= Q_o^2/(2gA_o^2) + Y_o = (50/25)^2/19.62 + 2.5 = 2.704$. Thus, the following implicit equation provides the depth Y:

$$Y = 2.704 - \frac{Q^2}{19.62(10Y)^2}$$

This equation will converge with Q given if the Y computed on the left of the equal sign is iteratively used for the Y on the right of the equal sign. After determining Y the lateral outflow can be determined by

$$q = 2.215(Y - 2)^{3/2}$$

and the flow rate determined from Equation 1.13, or

$$Q = 50 - \int q\,dx$$

Since this integration cannot be completed in closed form, it must be carried out using numerical techniques. The small FORTRAN program listed below does this under the assumption that if small enough steps are used for dx then the Q at the current position x can be used to compute the depth, etc.

Techniques designed to solve ordinary differential equations (ODEs) provide a better alternative for solving this problem than the crude numerical integration used in the above program. The ODE for this problem is $dQ/dx = 0.5/2g(Y - 2)^{3/2}$. Numerical methods for solving ODEs will be described in Chapter 4 and are extensively utilized in solving gradually varied flow problems.

Listing of FORTRAN program that numerically integrates the above equation in the solution for Example Problem 1.8

```
      Q=50.
      Y=2.5
      WRITE(6,100) Q,Y
100   FORMAT(' x',7X, 'Q',7X, 'Y',/,19('_'),
     &/,' 0',2F8.3)
      DO 30 I=1,1000
10    Y1=2.704-(Q/Y)**2/1962.
      IF(ABS(Y1-Y).LT. .000001) GO TO 20
      Y=Y1
      GO TO 10
20    Q=Q-.02215*(Y1-2.)**1.5
      IF(MOD(I,100).EQ.0) WRITE(6,110) I/100,Q,Y1
110   FORMAT(I3,2F8.3)
30    CONTINUE
      END
```

Solution from Progaram

X	Q(cfs)	Y(ft)
0	50.000	2.500
1	49.208	2.500
2	48.397	2.515
3	47.569	2.523
4	46.723	2.530
5	45.859	2.537
6	44.977	2.545
7	44.078	2.552
8	43.161	2.559
9	42.227	2.556
10	41.275	2.573

1.11 ENERGY PRINCIPLE

The equations of motion are of extreme importance to all mechanics, whether dealing with fluids or solids. The equations of motion are based on Newton's second law, that states that force = mass × acceleration. In this section the scalar application of this fundamental law of mechanics is discussed, as it applies to fluids in motion. The section is entitled "Energy Principle" because this scalar application gives equations similar to the energy equation in Thermodynamics. In the next section, the vector application of Newton's second law to fluid motion is discussed under the heading "Momentum Principle." The application of Newton's second law to a general three-dimensional unsteady fluid flow results in the **Navier Stokes** equations, whose development can be found in books dealing with fluid mechanics. Applying the same law to an ideal fluid that is inviscid (viscous shear stresses are ignored) produces the **Euler** equations of motion. Neither of these equations is presented herein. Rather simplified versions of these equations are given for one-dimensional and special flows, and these are most useful in handling problems of one-dimensional open-channel hydraulics.

It will be necessary to define acceleration in fluid motion, first. Acceleration is a vector that represents the time rate of change of the velocity vector, or **a** = d**V**/dt. Since velocity in a fluid is a function of the position as well as time in general, the full derivative d**V**/dt is obtained by the chain rule in calculus, or using Cartesian coordinates,

$$\frac{dV}{dt} = \frac{DV}{dt} = \frac{\partial V}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial V}{\partial y}$$

The quantity d**V**/dt is called the substantial derivative and is often denoted by D**V**/Dt to make it clear that the partial derivative with respect to time is not intended. Since dx/dt, dy/dt, and dz/dt equal the velocity components u, v, and w in the x-, y-, and z-directions, respectively, the above equation becomes

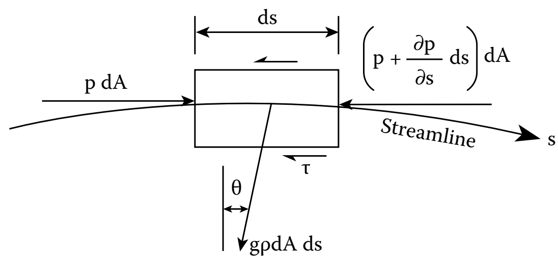
$$\frac{dV}{dt} = \frac{DV}{dt} = u\frac{\partial V}{\partial x} + v\frac{\partial V}{\partial y} + w\frac{\partial V}{\partial z} + \frac{\partial V}{\partial t}$$

The terms u∂**V**/∂x, v∂**V**/∂y, and w∂**V**/∂z are called the **convective** accelerations and can be visualized as the acceleration that occurs as streamlines within a flow converge together. The term ∂**V**/∂t is the **local** acceleration, and represents the variation in velocity that occurs at a point in the fluid with time. Also note that the above equations are vector equations since every term contains the velocity vector **V**.

If the flow is one dimensional, then the magnitude of the velocity is only a function of the coordinate in that direction and time, if the flow is unsteady, or V(s,t). s will be used for the space coordinate rather than x, because the one-dimensional flow will be allowed in the curved direction along a streamline, and not restricted to just the direction of the channel. For such one-dimensional flows the magnitude of the acceleration is given by

$$a = \frac{dV}{dt} = V\frac{\partial V}{\partial s} + \frac{\partial V}{\partial t}$$

Consider an element of fluid of length ds and cross-sectional area dA along a streamline in the direction s, as shown in the sketch below. Applying Newton's second law $\sum F_s = ma_s$ to this element results in (note that only magnitudes are used since only one direction s is involved)



$$p\,dA - \left(p + \frac{\partial p}{\partial s} ds\right) dA - \tau dP\,ds - \rho g\,dA\,ds \times \sin(\theta) = \rho\,dA\,ds\left(V\frac{\partial V}{\partial s} + \frac{\partial V}{\partial t}\right)$$

where P is the perimeter of the element. Noting that $\sin\theta = \partial z/\partial s$, dividing by $(-\rho\ dA\ ds)$ and simplifying results in

$$\frac{1}{\rho}\frac{\partial p}{\partial s}+\frac{\tau}{\rho}\frac{dP}{dA}+g\frac{\partial z}{\partial s}+V\frac{\partial V}{\partial s}+\frac{\partial V}{\partial t}=0$$

It is helpful in understanding equations to follow their dimensions. In the equation immediately above, the dimensions are force divided by mass, since every term started out as a force which was divided by the $\rho dAds$, or the mass of the differential element.

If flow is steady then $\partial V/\partial t = 0$, and the variables are only a function of s and, therefore, the above equation can be integrated along a streamline to give

$$\frac{p}{\rho}+\int\frac{\tau\, dp}{\rho\, dA}ds+gz+\frac{v^2}{2}=\text{constant}$$

Now the dimensions of the equation are force times length divided by mass. The product of force and length is work or energy. Thus, upon integrating along the streamline each term in the equation represents a form of fluid energy per unit mass. Dividing by g results in the Bernoulli Equation, that has units of energy per unit weight of fluid, or

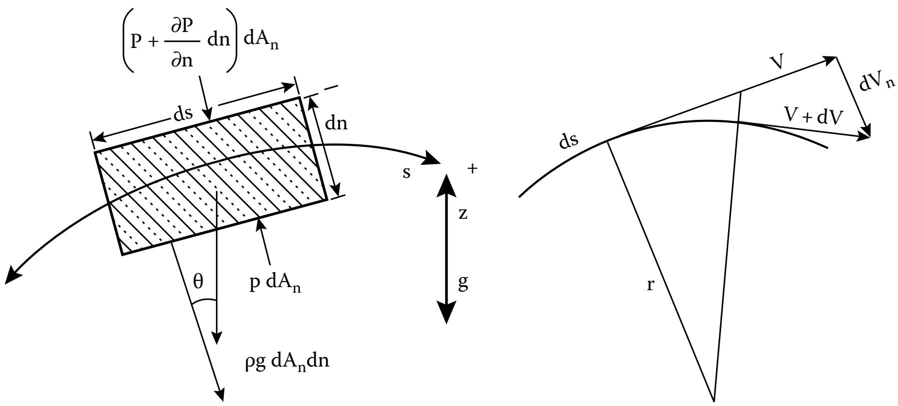
$$\frac{p}{\gamma}+z+\frac{v^2}{2g}+\int\frac{\tau\, dp}{\gamma\, dA}ds=H(\text{constant}) \tag{1.18}$$

The integral in Equation 1.18 cannot be evaluated in general since τ is not a known function of s, and therefore its value over the distance between a couple of points in the flow is designated as the headloss h_{L1-2} and Equation 1.18 is commonly applied between the two points in the following form:

$$\frac{p_1}{\gamma}+z_1+\frac{V_1^2}{2g}=\frac{p_2}{\gamma}+z_2+\frac{V_2^2}{2g}+h_{L_{1-2}} \tag{1.18a}$$

Each term in Equation 1.18 is considered a head because they all have the dimensions of length. In ES units this length is given in feet, and in SI units this length is in meters. It is important to note that terms are actually energy per unit weight of fluid. That is, when using ES units each term represents some form of energy in ft-lbs that each pound of fluid contains, and when using SI units each term is this energy in N-m that each Newton weight of fluid contains. p/γ is referred to as the pressure head, and represents the potential energy due to pressure per unit weight. z is the elevation head and is the potential energy due to the vertical position of the fluid per unit weight, and $V^2/2g$ is the kinetic energy per unit weight. The sum of p/γ and z is called the piezometric head.

To visualize these energies first consider the elevation head z. If a unit weight of fluid were a distance z above a datum then it would be capable of doing z times its weight of work in dropping under gravity to the datum. Since energy is the ability to do work, the potential energy that this unit weight of fluid has equals z. To visualize the potential energy due to the pressure of a fluid consider an infinitely large reservoir of liquid. At the bottom of this reservoir there is a cylinder that can extract work from the fluid due to its constant pressure here. This pressure acts over the area of the piston within the cylinder, and if the piston is permitted to move through a distance L, the amount of work done on it by the fluid equals pAL. The weight of fluid involved in doing this work equals γAL. Therefore, the energy per unit weight of this fluid is $pAL/(\gamma AL) = p/\gamma$. Note from the hydrostatic equations that $p/\gamma = h$ the height of the liquid above the cylinder. Thus, at the top of the reservoir the fluid contains elevation head, $z = h$, and at the bottom the liquid contains pressure head and at neither position does the liquid possess any velocity head since it is not moving, but these sum to a constant value. To understand that the velocity head $V^2/2g$ is the kinetic energy per unit weight, note that kinetic energy equals 1/2 the mass times the velocity squared, or $\rho(\text{Vol})V^2/2$. Dividing this by the weight of this mass of fluid gives K.E./(unit weight) = $\{\rho(\text{Vol})V^2/2\}/\{g\rho(\text{Vol})\} = V^2/2g$.



Some applications in open-channel flow call for looking at what happens across streamlines. Flow over the crest of a dam is one such application. Therefore, consider applying Newton's second law normal to the streamlines, as shown in the sketch. If forces on this element due to viscous shear are ignored then the summation of forces in the direction of an outward normal equated to the mass times outward normal acceleration results in

$$pdA_n - \left(p + \frac{\partial p}{\partial n} dn\right) dA - \rho g\, dA_n \cos(\theta) = \rho dA_n \left(\frac{V^2}{r} + \frac{\partial V_n}{\partial t}\right)$$

The $\cos\theta = \partial z/\partial n$ and the last quantity within parenthesis is the acceleration in the normal direction. To see the latter, note that $V_n = f(s,t)$, and therefore by the chain rule,

$$a_n = \frac{\partial V_n}{\partial s} V + \frac{\partial V_n}{\partial t} \quad \text{where } V = \frac{\partial s}{\partial t}$$

From the small vector diagram in the sketch above it can be seen from similar triangles that

$$\frac{dV_n}{V} = \frac{ds}{r} \quad \text{and therefore } \frac{\partial V_n}{\partial s} = \frac{V}{r}$$

dividing the above force equation by $\rho dA_n dn$, and simplifying gives

$$\frac{1}{\rho}\frac{\partial p}{\partial n} + g\frac{\partial z}{\partial n} + \frac{V^2}{r} + \frac{\partial V_n}{\partial t} = 0$$

Special cases are worth examining. If the flow is steady and streamlines are straight then the last two terms in the last equation are zero and upon integrating in the n-direction, the following is obtained:

$$\frac{p}{\gamma} + z = \text{constant}$$

which indicates that the pressure distribution in the normal direction is hydrostatic in a moving flow provided the streamlines are straight. Since this is the assumption made to classify flows as one dimensional, pressure distributions are generally considered hydrostatic unless the channel is steep as discussed previously.

Before the above equation can be integrated with curved streamlines, it is necessary to have a relationship between V and r. Two special cases will be considered; irrotational flow, and solid body, or forced vortex, rotational flow. Real flows are between these two cases. Fluid viscosity must be absent for a flow to be irrotational, whereas the rotation of a forced vortex follows the solid body rotation law that the velocity increases with the radius of rotation, $V = r\omega$.

In general, a flow is irrotational if the vorticity or cross-product of the differential del operator and the velocity vector equal zero, or $\xi = \notin X\mathbf{V} = 0$ where (using the Cartesian coordinate system)

$$v = \frac{\partial}{\partial x}\mathbf{i} + \frac{\partial}{\partial y}\mathbf{j} + \frac{\partial}{\partial z}\mathbf{k}$$

and i, j, and k are unit vectors in the x-, y-, and z-directions respectively.

$$\nabla = \frac{\partial}{\partial x}\mathbf{i} + \frac{\partial}{\partial y}\mathbf{j} + \frac{\partial}{\partial z}\mathbf{k}$$

and i, j, and k are unit vectors in the x, y, and z.

The component of vorticity for the plane of the curved flow above in the natural coordinate system is $\partial V_n/\partial s - \partial V/\partial n$ and if this flow is irrotational then this quantity equals zero. Since $\partial V_n/\partial s = V/r$, the term V^2/r in the above equation can be written as $V\partial V/\partial n$, permitting the equation to be integrated in the normal direction, giving

$$\frac{p}{\gamma} + z + \frac{v^2}{2g} = H\,(\text{constant}) \tag{1.18b}$$

or identically the same as Equation 1.18, without the viscous loss term; but since viscous forces were not included one would expect this. Therefore, if the flow is irrotational the Bernoulli equation applies across, as well as, along streamlines. The usual concept is that the Bernoulli equation is applied for the entire section of a one-dimensional flow. If the radius of curvature of the streamline is constant, it can be shown that for an irrotational flow the velocity times the radius is constant, or Vr = constant; thus, as r becomes zero V must become infinite.

For a forced vortex, the vorticity equals twice the actual rotation about an axis, or

$$\frac{\partial v_n}{\partial s} - \frac{\partial v}{\partial n} = 2\omega = \frac{2v}{r}$$

Therefore, $\partial V/\partial n = -V/r$ and integration of the above equation in the normal direction gives

$$\frac{p}{\gamma} + z - \frac{v^2}{2g} = H\,(\text{constant}) \tag{1.18bc}$$

(Note the minus sign in front of the velocity head.)

A centrifugal pump causes the flow within the housing of the pump to nearly follow this forced vortex law. The exception is that there must be a radial component of velocity for the fluid to pass onto and leave the pump impeller. This radial component will be governed by the usual Bernoulli equation and, therefore, the head produced by a pump will be approximated by (with $z_i = z_o$)

$$h_p = \frac{p_2 - p_1}{\gamma} + \left(\frac{V_i^2 - V_o^2}{2g}\right)_r + \left(\frac{V_o^2 - V_i^2}{2g}\right)_t$$

where subscripts i and o denote inlet and outlet, respectively, and subscripts r and t denote radial and tangential directions, respectively.

1.11.1 Kinetic Energy Correction Coefficient, α

The Bernoulli equation developed above is generally applied for an entire section of an open-channel flow, and the average velocity for the cross section is used to compute the velocity head, or kinetic energy per unit weight. Since the velocity is not constant throughout the entire section, the actual or exact kinetic energy per unit weight, K.E./Wt, can be defined by multiplying the velocity head by a correction coefficient α, or

$$\frac{\text{K.E.}}{\text{Wt}} = \alpha\left(\frac{V^2}{2g}\right)$$

The evaluation of α involves determining the exact kinetic energy and dividing this by the velocity head. The exact, or correct, kinetic energy K.E., passing a cross section is the integral over the area of the kinetic energy through a differential area dA, or

$$\text{K.E.} = \frac{\int \rho v^2 dQ}{2} = \frac{\int \rho v^3 dA}{2}$$

Dividing K.E. by $g\rho Q = g\rho VA$ gives K.E./Wt and, therefore,

$$\alpha = \frac{\text{K.E./Wt}}{v^2/2g} = \frac{\int v^2 dQ}{v^2 Q} = \frac{\int v^2 dA}{v^2 A} \tag{1.19}$$

For a natural channel, or when it is not possible to define v and A as functions of the depth, the integral sign is replaced by a summation, and Equation 1.19 becomes

$$\alpha = \frac{\text{K.E./Wt}}{v^2/(2g)} = \frac{\sum v^2 \Delta Q}{V^2 Q} = \frac{\sum v^2 \Delta A}{V^2 A} \tag{1.19a}$$

The value of α is only slightly larger than unity for typical turbulent open-channel flows. Furthermore, when the flow is subcritical, as it always is in natural rivers and streams, the velocity head will be smaller than the other terms in the Bernoulli equation, and therefore it is common practice to take α as 1.

EXAMPLE PROBLEM 1.9

Water flows over the crest of a small dam that is 4 m high at a rate of 48.2 m^3/s. The depth of water over the dam's crest is 6 m, and the channel is rectangular and 10 m wide. Determine the depth and velocity in the channel downstream from the dam. Ignore any frictional losses. Solve the problem twice; first, under the assumption that at both sections the kinetic energy correction coefficient is unity, and second, the αs are not unity, and that the following dimensionless velocity profiles are known.

Over the Dam's Crest								
y/Y	0.00	0.07143	0.2143	0.3571	0.5357	0.7143	0.8571	1.0000
v/V	0.00	0.3529	0.8235	1.0353	1.1764	1.2823	1.2235	1.0823
In the Channel at the Dam's Toe								
y/Y	0.00	0.0357	0.0714	0.1429	0.3571	0.7143	1.0000	
v/V	0.00	0.5743	0.8614	0.9763	1.0452	1.0567	1.0452	

Solution

For the first part with the α's = 1, the upstream velocity is determined from $V_1 = Q/A_1 = 48.2/60 = 0.8033$ m/s. Apply the Bernoulli equation,

$$Y_2 + \frac{Q^2}{2g(10Y_2)^2} = Y_1 + \frac{V_1^2}{2g} = 6.033 \text{ m}$$

This is a cubic equation that must be solved by trial, or an iterative technique such as the Newton Method described in Appendix B. The solution is $Y_2 = 0.461$ m. Another solution to this cubic equation is 6 m, but this is obviously not the wanted root since it is the upstream depth. The corresponding velocity is $V_2 = 48.2/4.61 = 10.46$ m/s. For the second part of the problem $q = 4.82 = \int v\,dy$, or $4.82V = \int(v/V)dy$, in which the limits of integration are 0–6. If dy is replaced by the dimensionless depth y/Y, then the upper limit changes to 1. There are many methods that can be used to evaluate this integral including plotting the profile and determining the area by counting squares. We will use a cubic spline as described in Appendix B to interpolate between the given values and Simpson's rule (described in Appendix C) to obtain a numerical integral. A FORTRAN program EPRB1_9.FOR listing to accomplish this task is given below.

```
      EXTERNAL EQUAT
      COMMON /TRA/Y(8),V(8),DQV(8),I1,IP,N
      I1=1
      IP=2
      WRITE(*,*)' Give no. and then this may pairs'
      READ(*,*) N,(Y(I),V(I),I=1,N)
      CALL SPLINESU(N,Y,V,DQV,D,0)
      CALL SIMPR(EQUAT,0.,Y(N),FLOW,1.E-5,30)
      WRITE(*,*) FLOW
      END
      FUNCTION EQUAT(YY)
      COMMON /TRA/Y(8),V(8),DQV(8),I1,IP,N
1     IF(YY.LT.Y(IP).OR.IP.EQ.N) GO TO 2
      I1=IP
      IP=IP+1
      GO TO 1
2     IF(YY.GT.Y(I1).OR.I1.EQ.1) GO TO 3
      IP=I1
      I1=I1-1
      GO TO 2
3     DY=Y(IP)-Y(I1)
      AY=(Y(IP)-YY)/DY
      BY=1.-AY
      EQUAT=AY*V(I1)+BY*V(IP)+((AY*AY-1.)*AY*DQV(I1)+(BY*BY-1.)*
     &BY*DQV(IP))*DY*DY/6.
      RETURN
      END
```

In the program, the subroutine SPLINESU is called to provide the values of the second derivatives at the points. The common block passes the arrays of dimensionless depths y/Y and corresponding dimensionless velocities, v/V, i.e., the values in the above table over the crest of the weir. The result of this numerical integration is 1.000 and (meaning that the above data truly give a proper dimensionless velocity profile), therefore, the average velocity V_1 is the same as above or $V_1 = 0.8033$ m/s. Next, the last two lines of the function subprogram EQUAT can be changed to the following:

```
 EQUAT=(AY*V(I1)+BY*V(IP)+((AY*AY−1.)*AY*DQV(I1)+(BY*BY−1.)*
&BY*DQV(IP))*DY*DY/6.)**3
```

Giving both the above tables of dimensionless values provides the solution for the kinetic energy correction coefficients: $\alpha_1 = 1.2557$, and $\alpha_2 = 1.0555$. Now the implicit equation that must be solved is

$$Y_2 + 1.0555\left\{\frac{4.82^2}{Y_2^2 19.62}\right\} = 6 + 1.2557\left(\frac{0.8033^2}{19.62}\right) = 6.0413$$

The solution is $Y_2 = 0.472$ m. The average velocity $V_2 = 4.82/0.472 = 10.21$ m/s.

EXAMPLE PROBLEM 1.10

A pump increases the pressure from its inlet to its outlet pipe by 52 psi. The inlet and outlet pipes are 10 and 12 in. in diameter, respectively, and the flow rate is Q = 6 cfs. Determine the amount of energy per unit weight that the pump supplies to the water. If the pump has an efficiency of 85%, determine the horsepower required to drive the pump.

Solution

In applying the energy equation to this problem the head of the pump needs to be added to the left side of the Bernoulli equation, or

$$\left(\frac{V^2}{2g} + \frac{p}{\gamma}\right)_1 + h_p = \left(\frac{V^2}{2g} + \frac{p}{\gamma}\right)_2$$

Upon substituting the known values into this equation gives $h_p = 120.97$ ft. To get energy from the head given by Bernoulli's equation multiply by the weight flow rate γQ, and therefore the horsepower is

$$HP = \frac{\gamma Q h_p}{550e} = 96.88\,\text{hp (550 is the conversion for hp from energy in ft-lb/s.)}$$

EXAMPLE PROBLEM 1.11

Assume that the water at the surface of a river flowing around a bend follows the free vortex law. The radius of the inside of the bend is 30 m, and at the outside of the bend the radius is 45 m. If the velocity at the inside of the bend is measured as 0.8 m/s, what is the velocity on the surface at the outside of the bend?

Solution

The free vortex law indicates that Vr = constant. This constant equals 0.8(30) and, therefore, the velocity at the outside of the bend is $V_o = (0.8)(30)/45 = 0.533$ m/s. Since at the bottom of the river the velocity will be smaller than at the surface, there is a net flow outward on the surface and an inflow at the bottom. This movement creates a secondary motion in a river that moves sediment toward the downstream inside of the bend. A pumping station here would bring in river sediments; therefore, such a pumping station is best located on the outside of the bend.

EXAMPLE PROBLEM 1.12

The velocity distribution in a wide open channel can be defined by the following logarithmic function between the channel bottom and a position y = 0.1 ft above the bottom: v = C/Ln(y), and a second-degree polynomial for the rest of the depth. When the depth of flow is 2.6 ft, velocities are measured at three positions with the following results:

Depth, ft	0.1	2.1	2.6
Velocity, fps	2.3	2.45	2.35

Determine the kinetic energy correction coefficient, α.

Solution

To obtain the equation for the velocity distribution over the bottom 0.1 ft substitute 2.3 in for v, and 0.1 for y in the given equation and solve for C. This gives C = 2.3 Ln(0.1) = −5.296. The equation for the rest of the profile is obtained by evaluating a,b, and c in $v = a + by + cy^2$ by Lagrange's interpolation equation described in Appendix B, or solving a, b, and c from the three equations obtained by substituting the y and v from above into this polynomial equation to give

$$v = 2.2694 + 0.317y - 0.110y^2$$

The flow rate q per unit width of channel is found first from Q = ∫v dA with the integration carried out in two parts, or

$$q = -5.296\int_0^{-0.1} \frac{dy}{Ln(y)} + \int_{0.1}^{2.4} (2.2694 + 0.317y - 0.11y^2)dy = 0.1715 + 6.0989 = 6.2705 \text{ cfs/ft.}$$

The first integral was evaluated numerically because of the complexity of a closed-form integration. Next, evaluate $\int v^3 dA$ as shown below.

$$\int v^3 dA = -5.296\int_0^{0.1} \frac{dy}{Ln(y)^3} + \int_{0.1}^{2.6} (2.2694 + 0.317y - 0.11y^2)^3 dy = 0.581 + 36.356 = 36.937$$

To evaluate α substitute into $\alpha = \int v^3 dA/(qV^2) = 36.937/[6.2705(2.4117)^2] = 1.013$.

1.12 MOMENTUM PRINCIPLE IN FLUID FLOW

An equally important skill for solving a fluid-flow problem to that associated with the use of the energy principle is the application of the momentum principle. The momentum principle must generally be utilized to solve problems dealing with external forces acting on the fluid. Its use occurs when dealing with the overall flow picture rather than a detailed examination of individual fluid particles, and their associated internal processes. The momentum principle produces vectors, and therefore is the natural choice when external forces are involved. It can be applied to one-, two- or three-dimensional flows, but when dealing with open-channel hydraulics the common use of the momentum principle will be for one- and two-dimensional flows.

The momentum principle in fluid mechanics is based on Newton's second law of motion, **F** = m**a**, where the force **F** and the acceleration **a** are vector quantities. Remember, to define a vector it is necessary that its direction as well as its magnitude are given, and that a change in direction (with the magnitude constant) is equally important to a magnitude change (with the direction remaining constant). Since acceleration is the time rate of change of velocity, Newton's second law can be written as

$$\mathbf{F} = m\frac{dV}{dt} = \frac{m}{dt}dV$$

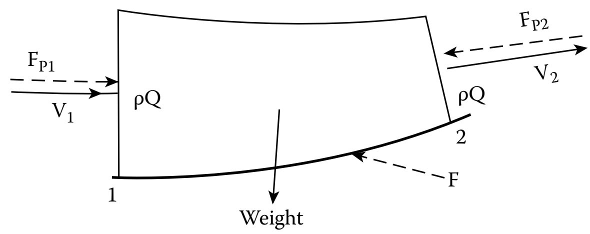
In writing the term after the last equal sign in the above equation the concept is that rather than dealing with a derivative we are dealing with a differential change in the velocity vector d**V** within the differential time dt. In solid mechanics dt is moved to the other side of the equal sign, associated with vector **F** and the product **F**Δt is called the linear impulse, and this is equated to the change in linear momentum that is defined as Δ(m**V**). In fluid mechanics, however, we deal with flow rates, and dm/dt will be denoted as m, the mass flow rate, which equals ρQ. The interpretation of m/dt is the change in mass per time or a mass flow rate ρQ. Therefore, for fluid applications the above equation can be written as

$$\mathbf{R} = \rho Q \Delta V$$

where **R** is the resultant force on the CV. When applied to a control volume, CV, of fluid between the two sections 1 and 2, as shown in the sketch below, it is written as

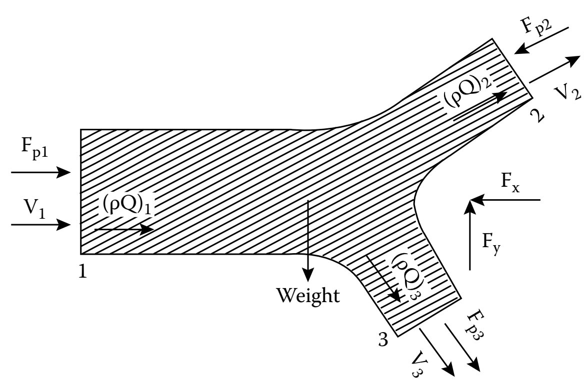
$$\mathbf{R} = \rho Q(V_2 - V_1) \tag{1.20}$$

The forces on the CV include the body forces, i.e., the weight of the CV, as well as external forces that may be applied by adjacent fluid and structures, etc., that act against the CV of fluid.



If the flow divides so that it leaves through sections 2 and 3 as shown below in the sketch, then the momentum equation would need to be written as

$$\mathbf{R} = (\rho QV)_2 + (\rho QV)_3 - (\rho QV)_1$$



The quantity (ρQ**V**) is called the momentum flux. A more generalized expression of the momentum principle states that the resultant external force acting on a CV of fluid equals the difference between the momentum flux vectors leaving the CV minus the momentum flux vectors entering the CV. Expressed as an equation the momentum principle becomes

$$\mathbf{R} = \sum (\rho QV)_{\text{leaving}} - \sum (\rho QV)_{\text{entering}} \tag{1.21}$$

Since Equation 1.21 is a vector equation, for two- and three-dimensional problems it can be written as two or three scalar equations, respectively, in coordinate directions of that space. For a one-dimensional problem the coordinate direction will coincide with the directions of the vectors on both sides of the equal sign in Equation 1.21 and the vectors become effectively scalar quantities. For a two-dimensional problem in Cartesian coordinates Equation 1.21 can be written as the two scalar equations,

$$\sum F_x = (\rho Q V_{x2} - \rho Q V_{x1}) \tag{1.21a}$$

and

$$\sum F_y = (\rho Q V_{y2} - \rho Q V_{y1}) \tag{1.21b}$$

where

the subscripts x and y represent the x and y coordinate directions, respectively

the 1 and 2 subscripts represent entering and leaving the control volume, respectively

The summation of forces in the x- and y-directions include all external and body forces acting on this CV.

The application of the momentum principle to most problems includes the following five steps:

1. Construct a control volume of the fluid. The upstream section(s) of this CV is (are) upstream from the occurrence of interest to a section of the flow where the velocity and other variables of the problem can be defined. Likewise, the downstream section(s) is (are) taken where fluid variables can be defined. Adjacent fluid and structures are removed from this CV.
2. Replace all removed fluid and structural components by the equivalent forces that they apply to the fluid in the CV.
3. Apply Equations 1.21a and 1.21b. In applying these equations it is vital to watch signs. Should the component of force or velocity be in the direction opposite to the positive direction taken for either x or y, then it is negative in these equations. Likewise, any vector not in either the x- or y-direction must be multiplied by the cos or sin of the appropriate angle to get the component in the coordinate direction.
4. Solve for the unknowns. Since Equations 1.21a and 1.21b are two equations, two unknowns can be obtained. For a one-dimensional problem only one unknown may exist, and for a three-dimensional problem three unknowns may exist. These unknowns may be the components of one unknown vector, such as the force.
5. If the unknown(s) solved for in step 4 consists of a vector, note that its sense will be opposite on the structure than on the fluid. The value obtained by the above procedure is the force on the CV of fluid.

In implementing step 2 above, the fluid adjacent to the CV of fluid is replaced by an equivalent force. In applying this to open-channel flows, this adjacent fluid will often have a hydrostatic pressure distribution, and the force will be a hydrostatic force. A hydrostatic, F_s, force equals the cross-sectional area times the pressure at the centroid of this area, or

$$F_s = p_c A = \gamma h_c A = \gamma A h_c \tag{1.22}$$

where h_c is the distance from the free surface to the centroid of the area. The last form given in Equation 1.22 contains the term Ah_c which is the first moment of area about the free surface. For rectangular and trapezoidal cross sections, this first moment of area is easily obtained. For a rectangle it equals the area times the distance from the surface to the center or $bY^2/2$. For a trapezoid it can be obtained by the method of composite areas, e.g., the rectangle and the two triangles and is $Ah_c = bY^2/2 + mY^3/3$. For other cross sections such as an arc of a circle, this first momentum of area must be obtained by integration. See Appendix A and Table A.1 for this first moment of area for a circular cross section.

EXAMPLE PROBLEM 1.13

A trapezoidal channel with a bottom width b_1 = 14 ft and a side slope m_1 = 1.5 divides into a rectangular channel with b_2 = 8 ft whose direction is 30° from the direction of the upstream channel, and a pipe with a 4 ft diameter whose direction is 45° from the direction of the upstream channel. The top of the pipe is below the water surface so it flows full. The depth in the downstream channel is 5.2 ft. The pipe drops in elevation by 110 ft over its length of 1500 ft, and delivers water at a pressure of 40 psi. The wall roughness for the pipe is e = 0.02 ft. Determine the resultant force on this structure for the design flow rate of Q_1 = 450 cfs.

Solution

First applying the Bernoulli equation between sections 1 and 2 and then sections 1 and 3 (or 2 and 3) along with the continuity equation and equations for pipe flow give the following five simultaneous equations to solve for the five unknowns Q_2, Q_3, f, Y_1, and p_t (pressure on top of pipe):

$$H = \frac{Q_1^2}{2gA_1^2} + Y_1 \frac{Q_2^2}{2gA_2^2} = \frac{Q_2^2}{64.4(38.4)^2} + 5.2 \qquad (1)$$

$$H = \frac{Q_3^2}{2gA_3^2} + \frac{pt}{\gamma} + 4.0 \qquad (2)$$

$$Q_2 + Q_3 = 450 \qquad (3)$$

$$h_f = \frac{f(L/D)Q_3^2}{2gA_3^2} = 110 - \frac{40(144)}{\gamma} + H \text{ (the Darcy–Weisbach equation from pipe flow)} \qquad (4)$$

and

$$\frac{1}{\sqrt{f}} = 1.14 - 2\log\left(\frac{e}{D} + \frac{9.35}{\mathrm{Re}\sqrt{f}}\right) \text{(the ColeBrook–White equation from pipe flow)} \qquad (5)$$

The solution gives Q_2 = 322.40 cfs, Q_3 = 127.60 cfs, f = 0.03036, Y_1 = 5.965 ft and p_t = 33.17 psf. (To solve these five equations you need to use the techniques described in Appendix B as the Newton method, or Mathcad or MATLAB on your PC.) From these the forces on the CV become

$$F_1 = \gamma A_1 h_{c1} = \gamma\left(\frac{14Y_1^2}{2} + \frac{1.5Y_1^3}{3}\right) = 22{,}162 \text{ lbs}, \quad F_2 = \gamma A_2 h_{c2} = \gamma 41.6(2.6) = 6{,}749.2 \text{ lbs},$$

$$F_3 = \gamma A_3\left(\frac{p_t}{\gamma} + 2\right) = 1{,}985.1 \text{ lbs}.$$

Applying the momentum equation in the x- and y-directions, respectively, gives the following two equations:

x-direction

$$22{,}162 - 6{,}749.18 \cos 30 - 1{,}985.1 \cos 45 - R_x = 1.94[(322.4)7.75 \cos 30$$

$$+ (127.6)10.15 \cos 45 - 450(3.288)] = 3{,}105.1$$

and solving $R_x = 11{,}808$ lbs

y-direction

$$-6{,}749.2 \sin 30 + 1{,}985.1 \sin 45 + R_y = 1.94[(322.40)7.75 \sin 30$$

$$-(127.6)10.154 \sin 45] = 646.3$$

and solving $R_y = 2{,}617$ lbs

which results in R = 12,095 lbs at an angle of 12.5° from the horizontal with its direction downward to the right on the structure.

1.12.1 Momentum Flux Correction Coefficient, β

When the momentum principle is applied as if a three-dimensional problem is a two-dimensional problem, as in Example Problem 1.13, or as if a two-dimensional problem is a one-dimensional problem, i.e., when an average velocity is used for a section of flow, then $\rho Q\mathbf{V}$ does not represent the exact momentum flux, EMF, passing the section. The EMF can be obtained by multiplying by a correction coefficient, β in a manner similar to the use of α as a correction for the velocity head. By definition this coefficient, β, is

$$\beta = \frac{\text{EMF}}{\rho QV} = \frac{\rho \int v\,dQ}{\rho QV} = \frac{\int v^2 dA}{V^2 A} \tag{1.23}$$

For a natural channel the integral is replaced by a summation and Equation 1.23 becomes

$$\beta = \frac{\text{EMF}}{\rho QV} = \frac{\rho \sum v\Delta Q}{\rho QV} = \frac{\sum v^2 \Delta A}{V^2 A} \tag{1.23a}$$

and Equation 1.20 becomes

$$\mathbf{R} = (\rho\beta QV)_2 - (\rho\beta QV)_1 \tag{1.20}$$

EXAMPLE PROBLEM 1.14

The velocity distribution with depth in a wide rectangular channel is given by v = Ln(1 + 10y) where y is the distance from the channel bottom to the position where the velocity v is given. If the depth of flow Y = 5 ft, determine the momentum correction coefficient β.

Solution

Per unit width $\beta = \int v^2 dy/(qV)$ in which q is the volumetric flow rate per unit width and equal the depth time the average velocity V. To solve for β it is necessary to first solve $q = \int v dy = \int \text{Ln}(1 + 10y)dy$ with limits of 0 to 5. This can be integrated by letting x = 1 + 10y, and dy = dx/10; thus $q = \int \text{Ln}(x)dx/10$ with limits from 1 to 51, or q = [x Ln(x) − x]/10 evaluated between 51 and 1; giving q = 15.0523107. You will find it instructive to have your HP48 calculator obtain this answer by numerically integrating the above equation, and also have it carry out the symbolic integration. Since you will be using computer programs throughout this course, how about gaining some experience with this problem, but utilizing the Simpson's numerical integration described in Appendix B? The main program that calls SIMPR is listed below; the first column obtains q and also prints out the average velocity, and the second column integrates $\int v^2/dy$ between 0 and 5, and also prints out β.

```
EXTERNAL EQUAT
CALL SIMPR(EQUAT,0.,5.,VALUE)
WRITE(*,*) VALUE,VALUE/5.
END
FUNCTION EQUAT(Y)
EQUAT=ALOG(1.+10.*Y)
RETURN
END
```

```
EXTERNAL EQUAT
CALL SIMPR(EQUAT,0.,5.,VALUE)
WRITE(*,*) VALUE,VALUE/45.312708
END
FUNCTION EQUAT(Y)
EQUAT=ALOG(1.+10.*Y)**2
RETURN
END
```

The results are q = 15.05205 cfs/ft, V = 3.0104 fps from the first integration, and $\int v^2 dy = 48.73822$ and $\beta = 1.0756$.

EXAMPLE PROBLEM 1.15

Determine α and β for the natural channel in Example Problem 1.5.

Solution

The numerator in Equation 1.19 can be viewed as a double integral since the differential area dA = dy dx, or this numerator is

$$\iint v^3 dy\,dx = \int_0^{X_f}\left[V_s^3 Y\int_0^1 V'^3 dy'\right]dx = \left\{\int_0^1 v'^3 dy'\right\}\int_0^{X_f}\left(V_s^3 Y\right)dx$$

Note that the quantity within { } after the last equal sign is constant, since the dimensional velocity profile does not change from position to position and, therefore, it needs to be evaluated only once. Thus, the argument of numerical integration with respect to x (across the channel) involves the surface velocity Vs cubed, multiplied by the depth Y at this section, i.e., Vs^3Y, and both Vs and Y can be evaluated using the cubic spline interpolation of the data in the first table in Example Problem 1.5. The same applies for the numerator in Equation 1.23 to evaluate β; with the surface velocity squared rather than cubed. The program EXPR1_15 (in both FORTRAN and C) is given below to provide the numerator for both Equations 1.19 and 1.23. It also provides the area of the cross section, which can be done by evaluating the last integral in the above equation with the exponent of the velocity equal to zero, since $V_s^0 = 1$. Note that this program contains the additional variable NE that is used as the exponent of the surface velocity. When NE = 0, the area of the cross section is evaluated; when NE = 2, the numerator for β is evaluated, and when NE = 3, the numerator for α is evaluated.

Listing for Program EPR1_15.FOR

```
      EXTERNAL VPROF,Dq
      REAL DUM(30)
      COMMON YP(20),VP(20),X(30),Y(30),VS(30),
     &D2VP(20),D2VS(30),D2Y(30),I1,I2,NP,NX,NE
      READ(2,*) NP,(YP(I),VP(I),I=1,NP)
      READ(2,*) NX,(X(I),Y(I),VS(I),I=1,NX)
      CALL SPLINESU(NP,YP,VP,D2VP,DUM,0)
      CALL SPLINESU(NX,X,Y,D2Y,DUM,0)
      CALL SPLINESU(NX,X,VS,D2VS,DUM,0)
      NE=0
      CALL SIMPR(Dq,0.,X(NX),AREA,1.E-6,20)
      WRITE(*,*)' Area=',AREA
      NE=2
5     I1=1
```

```
      I2=2
      CALL SIMPR(VPROF,0.,1.,qPRIM,1.E-6,20)
      WRITE(*,*)' Integral of dimensionless',
     &velocity**',NE,' profile=',qPRIM
      I1=1
      I2=2
      CALL SIMPR(Dq,0.,X(NX),Q,1.E-4,20)
      WRITE(*,*)' Integral with respect to x='
     &,Q*qPRIM
      NE=NE+1
      IF(NE.LT.4) GO TO 5
      END
      FUNCTION VPROF(YY)
      COMMON YP(20),VP(20),X(30),Y(30),VS(30),
     &D2VP(20),D2VS(30),D2Y(30),I1,I2,NP,NX,NE
1     IF(YY.LT.YP(I2) .OR. I2.EQ.NP) GO TO 2
      I1=I2
      I2=I2+1
      GO TO 1
2     IF(YY.GE.YP(I1) .OR. I1.EQ.1) GO TO 3
      I2=I1
      I1=I1-1
      GO TO 2
3     DYP=YP(I2)-YP(I1)
      A=(YP(I2)-YY)/DYP
      B=1.-A
      VPROF=(A*VP(I1)+B*VP(I2)+((A*A-1.)*A*
     &D2VP(I1)+(B*B-1.)*B*D2VP(I2))*DYP**2/6.)**NE
      RETURN
      END
      FUNCTION Dq(XX)
      COMMON YP(20),VP(20),X(30),Y(30),VS(30),
     &D2VP(20),D2VS(30),D2Y(30),I1,I2,NP,NX,NE
1     IF(XX.LT.X(I2) .OR. I2.EQ.NX) GO TO 2
      I1=I2
      I2=I2+1
      GO TO 1
2     IF(XX.GE.X(I1) .OR. I1.EQ.1) GO TO 3
      I2=I1
      I1=I1-1
      GO TO 2
3     DX=X(I2)-X(I1)
      A=(X(I2)-XX)/DX
      B=1.-A
      AA=A*(A*A-1.)*DX**2/6.
      BB=B*(B*B-1.)*DX**2/6.
      DEPTH=A*Y(I1)+B*Y(I2)+AA*D2Y(I1)+BB*D2Y(I2)
      VSURF=A*VS(I1)+B*VS(I2)+AA*D2VS(I1)+
     &BB*D2VS(I2)
      Dq=DEPTH*VSURF**NE
      RETURN
      END
```

Listing for Program EXPR1_15.C

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
```

```
float yp[20],vp[20],x[30],y[30],vs[30],d2vp[20],d2y[30],d2vs[30];
int i1,i2,np,nx,ne;
extern float simpr(float (*equat)(float xx),float xb,float xe,\
  float err,int max);
extern void splinesu(int n,float *x,float *y, float *d2y,float *d,\
  int ity);
float vprof(float yy){float a,b,dyp;
 while((yy>=yp[i2])&&(i2<np-1)){i1=i2;i2++;}
 while((yy<yp[i1])&&(i1>0)){i2=i1;i1–;} dyp=yp[i2]-yp[i1];
 a=(yp[i2]-yy)/dyp;b=1.-a;
 return pow(a*vp[i1]+b*vp[i2]+((a*a-1.)*a*d2vp[i1]+(b*b-1.)\
   *b*d2vp[i2])*dyp*dyp/6.,ne);
} // End of vprof
float qp(float xx){float a,b,dx,aa,bb,depth;
 while((xx>=x[i2])&&(i1<nx-1)){i1=i2;i2++;}
 while((xx<x[i1])&&(i1>0)){i2=i1;i1––;}
 dx=x[i2]-x[i1];a=(x[i2]-xx)/dx;b=1.-a;
 aa=a*(a*a-1.)*dx*dx/6.;bb=b*(b*b-1.)*dx*dx/6.;
 depth=a*y[i1]+b*y[i2]+aa*d2y[i1]+bb*d2y[i2];
 return depth*pow(a*vs[i1]+b*vs[i2]+aa*d2vs[i1]+bb*d2vs[i2],ne);
} // End of qp
void main(void){FILE *fili; char filnam[20];int i;
 float dum[30],qprim,q;
 printf("Give input file name\n"); scanf("%s",filnam);
 if((fili=fopen(filnam, "r"))==NULL){printf("Cannot open file\n");
   exit(0);}
 fscanf(fili, "%d",&np);
 for(i=0;i<np;i++)fscanf(fili, "%f %f",&yp[i],&vp[i]);
 fscanf(fili, "%d",&nx);for(i=0;i<nx;i++)fscanf(fili, "%f %f %f",\
   &x[i],&y[i],&vs[i]);
 splinesu(np,yp,vp,d2vp,dum,0);splinesu(nx,x,y,d2y,dum,0);
 splinesu(nx,x,vs,d2vs,dum,0);
ne=0;printf("Area=%f\n",simpr(qp,0.,x[nx-1],1.e-6,20));ne=2;
L5:i1=0;i2=1; qprim=simpr(vprof,0.,1.,1.e-6,20); i1=0;i2=1;
printf("Integral of dimensionless velocity**%d\
  profile =%f\n",ne,qprim);
printf("Integral with respective to x =%f\n",\
  qprim*simpr(qp,0.,x[nx-1],1.e-4,20));
ne++; if(ne<4) goto L5;
}
```

Using as input the data in file EXPRB1_5.DAT, used in Example Problem 1.5, program EXPR1_15 produces the following as the solution (the units have been added):

AREA = 857.0909 m^2

Average velocity V = Q/A = 342.5057/857.0909 = 0.3991689 m/s
Integral of dimensionless velocity**2 profile = 0.8307201
Integral with respect to x = 153.15
Integral of dimensionless velocity**3 profile = 0.8180358
Integral with respect to x = 71.6953

From which V = Q/A = 342.5057/857.0909 = 0.399169 m/s

$\alpha = \text{I}v^3dA/(AV^3) = 71.6953/(857.0909\text{x}.3991689^3) = 1.3108$
$\beta = \text{I}v^2dA/(AV^2) = 153.15/(857.0909\text{x}.3991689^2) = 1.1189$

PROBLEMS

Terminology

1.1 Define the flows in the following situations according to (1) steady or unsteady; (2) uniform or nonuniform and if nonuniform, whether rapidly or gradually varied; (3) subcritical or supercritical; and (4) turbulent or laminar. If insufficient information is provided for a classification, indicate what other information is needed.

(a) A flow in a river that produces a storm hydrograph at the site of observation.

(b) The flow downstream from a gate supplying water to a prismatic channel from a constant water surface elevation reservoir.

(c) The same for as in (b) except the gate is slowly being closed.

(d) A constant flow rate entering a trapezoidal canal of constant cross section that supplies water to three turnouts (consider: (1) the flow upstream from the first turnout, (2) the flow between the turnouts, and (3) the section downstream from the last turnout). A depth of water is required for the canal to supply the last turnout.

(e) Water flow into a canal from a large lake with the control gate fully open. The canal has a constant cross section and a small bottom slope.

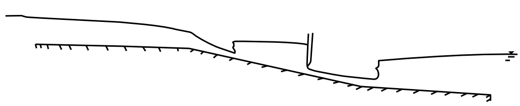
1.2 Water at a temperature of 55°F is flowing at a rate Q = 550 cfs in a trapezoidal channel with a bottom width of b = 12 ft, and a side slope m = 1.5. If the depth of flow is 4.5 ft, compute (a) the Froude number, (b) the Reynolds number, and (c) the speed of a small amplitude gravity wave in this flow. Classify the flow in this channel.

1.3 A cross section of a natural channel has the following transect data:

x (ft)	0	2	4	5	6	8	10	12	14	15	17
y (ft)	0	0.6	1.5	2.5	4.5	5.0	4.3	2.8	1.2	0.3	0

For a flow rate of Q = 280 cfs at a depth of 4.6 ft compute (a) the cross-sectional area, (b) the Froude number, and (c) the Reynolds number if the water temperature is T = 50°F.

1.4 The sketch below shows a long canal system whose bottom slope changes at several points, and is controlled by gates. Indicate the reaches where the flow is subcritical (downstream controlled) and where it is supercritical (under upstream control). Under uniform flow a steep channel will sustain a supercritical flow whereas a mild channel will sustain a subcritical flow.



1.5 Starting with the perfect gas law ρ = p/(RT) and the definition of bulk modulus, prove that a gas undergoing an isothermal compression or expansion has a bulk modulus equal to its absolute pressure.

1.6 The speed of sound propagates through a fluid with a velocity given by $c = \sqrt{E_v/\rho}$. Compare the speed of sound in pure water to that of water that contains 1% free air by volume.

1.7 Compute the pressure variation in the ocean to a depth of 4000 m if the compressibility is not ignored, and then compute it on the basis of an incompressible fluid. In this computation assume both g and E_v are constant. Take the density of ocean water as ρ = 1020 kg/m^3.

1.8 Water is flowing down a steep spillway with slope S_o = 0.30 at a depth of 0.8 m. Determine the pressure at the bottom of this channel.

1.9 A pressure transducer records 4.00 psi of pressure on the bottom of a steep channel with a bottom slope of two to one (horizontal-to-vertical distances). What are the normal and vertical depths of this flow? The bottom slope is constant.

1.10 At the base of a dam spillway the bottom of the channel suddenly changes from having a constant slope of S_o = 0.40 to a circular arc with a radius of 30 m. The normal depth of flow at this point of change is 0.75 m and the velocity V = 10 m/s. Assume the radius of curvature of the streamlines changes linearly from 30 to 20 m through this depth, and compute the pressure distribution through the depth of flow where the radius is 20 m, and also the pressure distribution along the bottom of the channel through the length where the radius of curvature varies. What is the force against a section of fluid per unit width of flow at the section where the circular arc first begins?

1.11 Water is being rotated in a cylindrical drum of 2 m radius and height 5 m at a rotational speed of (a) $\omega = 1200$ rpm and (b) $\omega = 40$ rpm. The axis of rotation is vertical. Compute the pressure distribution in the vertical direction at the following three radii: $r_1 = 0$, $r_2 = 1$ m, and $r_3 = 2$ m. When rotated, water is to the top of the tank.

1.12 Determine the average velocity, V, for a laminar flow that has a velocity profile given by Equation 1.11. How is the average velocity related to the velocity on the surface? How does this result relate to the area under a parabola?

1.13 Solve illustrative Example Problem 1.7 if the flow rate is $Q = 20\,m^3/s$ and Y still equals 3 m in the upstream channel.

Continuity

1.14 Prove whether the following two-dimensional flow fields are (1) continuous (satisfy the continuity equation), and (2) are rotational or irrotational.

(a) $u = U$ (constant) and $v = 0$

(b) $u = 10x/(x^2 + y^2)$ and $v = 10y/(x^2 + y^2)$. For this flow determine the tangential and radial components of velocity.

(c) $u = \sin(x)\cosh(y)$ and $v = \cos(2x)\sinh(y)$

(d) $u = 3x^2 - 2y^2$ and $v = 6xy$

(e) $u = \sinh(y)\cos(x)$ and $v = \cosh(y)\sin(x)$

(f) $u = 5(x^2 - y^2)$ and $v = 10xy$

1.15 The velocity and depth of water flowing in a 2 m diameter pipe are 3 m/s and 1.4 m at one section. At another section the depth equals 0.6 m. What is the velocity at this second section?

1.16 The flow from a trapezoidal channel with b = 10 m and m = 2 goes through a transition section and enters a circular culvert with an 8 m diameter. If the flow rate is $350\,m^3/s$ and the depths in these two channels equal 4 m and 6.4 m respectively, what are the average velocities at the two sections.

1.17 A river whose cross section is defined by the following x y data from the left bank (y is positive downward from the top of the left bank) has been gaged by a current meter giving the velocities shown at the 0.2 and 0.6 depth at five position across the river, when the river stage was 2 ft. Under the assumption that the average of these two velocities gives the average velocity for this incremental position across the river, determine the flow rate in the river.

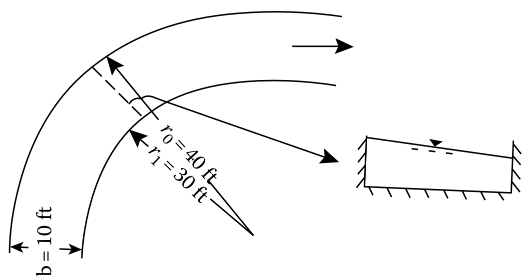
Cross-section data for the river

x (ft)	0	2.0	4.0	6.0	8.0	10.0	12.0
y (ft)	0	0.8	1.1	2.3	2.1	1.0	0.1

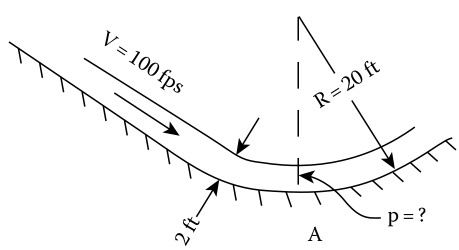
Current meter measurements

Position	**1**	**2**	**3**	**4**	**5**
x (ft)	1.2	3.6	6.0	8.4	10.8
$v_{0.2}$ (fps)	1.1	1.3	1.6	1.4	1.2
$v_{0.8}$ (fps)	1.3	1.5	1.9	1.5	1.3

1.18 Water in a rectangular channel flows around a 90° bend. The channel is 10 ft wide, the inside radius of the bend is $r_i = 30$ ft and the radius of the outside of the bend is $r_o = 40$ ft. A flow rate of $Q = 400$ cfs is in the channel when the depth of water at the inside of the bend is measured at $Y_i = 5$ ft. Make the following assumptions: (a) there is no secondary flow; (b) the angular momentum is constant, i.e., the flow around the bend follows the free vortex law; (c) the velocity does not change with position between the bottom of the channel to the water surface at any give radial distance. Determine the following: (1) The depth at the outside of the bend, (2) The velocity at the inside of the bend, V_i, and (3) the velocity at the outside of the bend. (To accomplish these tasks you should first develop the equation that gives the depth through the bend as a function of the radius r, and the free vortex constant, and then evaluate the free vortex constant by making sure that 400 cfs passes through the bend.)



1.19 Water comes off a spillway with a velocity of V = 100 fps at its toe. The spillway flip bucket at its end consists of a circular arc with a radius R = 20 ft. The depth of flow through the flip bucket is 2 ft. Assume that the velocity distribution through the depth of flow is constant. Derive the equation that gives the pressure distribution in the water on the spillway bucket through a vertical section passing through the center of the circle, i.e., the section vertically above point A on the sketch. What is the pressure at the bottom of the flow at this position?



Energy

1.20 Prove that for an irrotational flow that if the radius of curvature for a particle of fluid remains constant that its velocity is given by v = C/r (C = constant). Therefore, as rϖ0, then vϖ∞.

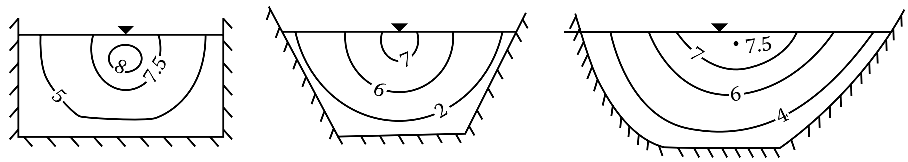
1.21 Obtain the equation y = f(x) that describes the top surface of water flowing over a sharp crested weir in a wide channel under the following assumptions: (a) the top surface remains horizontal for all x < 0 (where x has its origin at the weir crest), (b) there is no friction between this top streamline and the fluid below it, and (c) there is no pressure gradient in the normal direction at the top streamline. The average velocity and depth upstream from the weir are V_o and Y_o, respectively. What will cause the real flow to have a top surface that deviates from this mathematical description.

1.22 A weir at the end of a wide rectangular channel has its crest 1.2 m above the channel bottom, and the depth of water above the weir crest is 0.8 m. Under the assumption that the flow is inviscid, compute the thickness of the falling water at a point 2.5 m below the weir crest.

1.23 An irrotational flow occurs around a 90° circular bend in a rectangular channel. The channel is 10 ft wide, and the inside radius of the bend is 60 ft. If velocity at the inside of the bend is 8 fps, and the velocity does not vary in the vertical direction, determine the flow rate in this channel if the depth at the inside of the bend is 4.5 ft. What is the difference between the water surface elevation at the inside and outside of the bend under this ideal situation? How would the real flow around this bend deviate from this behavior?

1.24 Assume the velocity distribution is parabolic in a shallow depth of flow in a wide channel. Determine the value of the kinetic energy correction coefficient, α.

1.25 Lines of constant velocity are shown in cross sections of a flow in three channels. Determine the kinetic energy correction coefficient for each.



1.26 Lateral outflow at a rate of 2 cfs/ft is taking place over a 20 ft long side weir. There is zero flow downstream from this weir, and the width of the channel is 5 ft. Develop the equation that describes the depth of flow as a function of x across the length of the side weir. How would this solution be complicated if the discharge from the side weir depended upon the depth of water above its crest to the 3/2 power?

1.27 The velocity distribution in a wide rectangular channel is as given in illustrative Problem 1.13, v = Ln(1 + 10y). Determine the kinetic energy correction coefficient, α.

1.28 Assume the velocity distribution with depth in a trapezoidal channel with a bottom width b = 3 m and a side slope m = 1.2 is given by the equation v = 0.5 Ln(1 + 3y) (m/s). The depth of flow Y = 2 m. What is the volumetric flow rate? What is the kinetic energy correction coefficient, α? What is the average energy; per unit weight in this channel flow?

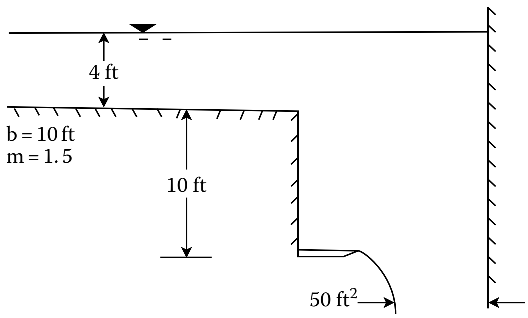
Momentum

1.29 Solve Example Problem 1.13 if the pressure delivered at the end of the pipe is 45 psi instead of 40 psi.

1.30 Compute the resultant force on a spillway bucket that turns a flow rate of q = 300 cfs/ft from its direction down the spillway at a slope of $S_o = 0.4$ to the horizontal direction. The depth of flow at the beginning of this bucket is 5 ft and at the end of the bucket is 4.6 ft.

1.31 If the spillway bucket of the previous problem turned through a total angle of 33° what would the force on the bucket be? How high would the jet of water rise above the bottom of the bucket and what would its velocity be at this highest point?

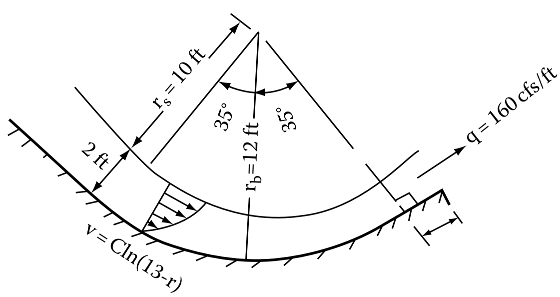
1.32 A structure takes water in the vertical direction from a position 10 ft below the bottom of a channel under atmospheric pressure. The area of this outlet is 50 ft^2. The upsteam channel is of trapezoidal shape with b = 10 ft and m = 1.5, and the depth in this channel just upstream from this structure is 4 feet. Compute the flow rate and the resultant force on this structure. Ignore frictional losses.



1.33 Determine the momentum flux correction coefficient, β for the velocity v = 0.5 Ln(1 + 3y) given for the flow in the trapezoidal channel of Problem 1.28 (b = 3 m, m = 1.2 and Y = 2 m).

1.34 A flip bucket at the toe of a spillway has a central portion consisting of a circle with radius of r_b = 12 ft. After a short smooth transition from the constant slope spillway face, the circular portion of the bucket begins at an angle of 35° to the left of the vertical and it ends at an angle of 35° to the right of the vertical as shown in the sketch. After the end of the circular portion a constant section 2 ft in length ends the bucket. A flow rate of q = 160 cfs/ft is coming down the spillway, and the depth normal to the bottom at the beginning of the circular arc is 2 ft. Assume that the velocity distribution can be defined by the following logarithmic function of the radius r. v = C ln(13 − r).

Do the following: (1) Determine the constant C (in integral tables you will find Iln(x)dx = x ln(x) − x). (2) Write the expression that provides the pressure gradient dp/dr through the circular portion of the spillway bucket. How would you obtain the pressure distribution at the end of the circular portion of the bucket at the plus 35° angle?



1.35 Repeat Example Problems 1.5 and 1.15, except use the following dimensionless velocity profile, rather than the one given in Example Problem 1.5.

Dimensionless depth, y′	0.00	0.03	0.05	0.10	0.15	0.20	0.60	0.80	0.90	1.00
Dimensionless velocity, V′	0.00	0.30	0.55	0.75	0.95	0.98	0.985	1.00	1.01	1.00

1.36 The program EXPRB1_5 that was written to solve Example Problem 1.5 integrated the dimensionless velocity profile to obtain a dimensionless unit flow rate q′. Modify this program so the actual velocity profile at each position x is integrated to get the actual unit flow rate q. (In doing this you can evaluate q for each unit of width of channel, and sum these q's to get the total flow rate Q.) What flow rate, channel area, and average velocity do you get with this process?

1.37 Using the approach used in the previous problem, determine the values of the energy correction coefficient α, and the momentum correction coefficient β for the natural channel in Example Problem 1.5.

2 Energy and Its Dissipation in Open Channels

2.1 INTRODUCTION

In applying the energy principle to the flow of liquids in open channels, it is necessary at the outset to be able to describe the resistance of fluid to motion. This resistance occurs because all real fluids have the property called viscosity, which causes internal shearing stresses to exist within the fluid as a consequence of a velocity gradient as it passes over a solid boundary. Viscosity is defined in Chapter 1. At the solid boundary, the velocity in the fluid must be zero, or agree with the velocity of the boundary, and increase therefrom for movement to take place. Thus it takes work, or energy, to cause motion of fluids relative to the boundaries that contain the fluids. This resistance and its effects are given several names such as "frictional resistance," "viscous shear," "friction factors," "friction losses," "friction energy dissipation," "frictional head loss," etc. Often, friction is omitted. In the case of liquids, the internal processes associated with fluid friction involve the conversion of useful fluid energy, which is originally in the form of a potential or kinetic energy, into a non recoverable energy, i.e., increase in temperature of the liquid. It is appropriate, therefore, to consider it a head loss, or dissipation of energy per unit weight of fluid. The increase in temperature is very small and of minor, if any, significance.

The first part of this chapter deals with how this head loss can be determined practically in computing depths of flow, and velocities that will occur if a given volumetric flow rate is to occur in a channel of a given size. These losses will be restricted to **uniform flows**. Nonuniform flows will be dealt with in Chapter 4. After being able to determine this loss, the second part of this chapter deals with the application of the energy principle to open channel flows. In this part, concepts associated with **specific energy** in open channel flow will be covered, as well as the differences between subcritical and supercritical flows.

2.2 APPROACHES TO FRICTIONAL RESISTANCE

The problem of frictional resistance in open channel flows is complex and depends on sound engineering judgment in selecting appropriate coefficients. The subject is made more complex by the fact that many channels are unlined, and therefore may have a **moveable** bed under some, or all flow conditions, that exist in that channel. The effects of bed movement are not considered in this chapter and are dealt with in articles and volumes dealing with sediment transport, its scour, and deposition and flow in alluvial channels. This chapter is restricted to flow resistance in **fixed bed channels**.

The different means for handling the resistance to motion in channels can be roughly classified as (1) the more fundamentally sound friction factor approach similar to that used for pipe flows when utilizing the Darcy–Weisbach equation and (2) based on tested and widely used empirical equations. In engineering practice, the use of empirical equations dominates in computing frictional losses in open channels. The reason for this is associated with the complexities involved with the friction factor approach. The same occurs in pipe flow computations in which the Hazen–Williams equation is used more extensively than is the Darcy–Weisbach equation. The use of empirical equations should be limited to ranges of situations for which they give good answers. A more fundamentally sound approach generally does not have these restrictions. As computers take over the task of doing the arithmetic, there will likely be a trend in engineering practice to switch to the friction factor approach. The friction factor approach will be described first and it will be followed by a section dealing with the use of Manning's equation.

Before beginning this discussion of fluid frictional losses in open channel flows, it should be pointed out that other complex phenomena may operate in dissipating fluid energy that are not covered by this theory, and using the results from this theory for these channel flows may produce erroneous answers. Such a flow, for example, exists in a steep mountain stream with very large bed elements that extend up to, or above, the water surface in many locations. Data taken from such a river/stream that flows through the Rocky Mountain Hydraulics Laboratory (Jarrett, 1991) indicate that energy dissipated is proportional to the velocity raised to the 8.3 power, whereas frictional theory for turbulent flows has the energy dissipation proportional to the velocity squared. Empirical equations, such as Manning's equation, are not appropriate for such flows. In fact, neither is Manning's equation appropriate to determine flow depths, etc. on steep spillways with slopes 0.2 or greater even though they are made of relatively smooth concrete, because it has not been developed to describe such large velocity flows that have large-scale turbulence associated with them. Neither is Manning's equation suited for extremely low-velocity flows through tall grass, for example.

2.2.1 Friction Factors in Open Channels

In 1768, a French engineer, Antoine Chezy, reasoned that the resistance to flow in an open channel would vary with the wetted perimeter and with the square of the velocity and that the force to balance this resistance would vary with the area of the cross section and with the slope of the channel. Therefore, he proposed that

$$\frac{V^2P}{(AS)} = \frac{V^2}{(R_hS)} = \text{Constant}$$

and would be the same for any similar channel. He used this in designing a canal for the Paris water supply. Years later, in 1897 his manuscript was published, and as his method became known and adopted by other engineers, the square root of the constant became know as the Chezy coefficient, and the equation

$$Q = CA(R_hS)^{1/2} \tag{2.1a}$$

or

$$V = C(R_hS)^{1/2} \tag{2.1b}$$

became known as Chezy's equation. While Chezy took C as a constant for a channel with a fixed wall roughness, we now know that C is a function of the Reynolds number ($R_e = 4VR_h/\nu = 4Q/(\nu P)$; also note as above that the hydraulic radius R_h is the area divided by the wetted perimeter, or $R_h = A/P$) of the flow as well as the relative roughness (e/R_h) of the channel wall, or $C = f(R_e, e/R_h)$. In other words, not only does C depend on the type of channel, but its value also depends on the flow conditions in that channel.

The processes associated with fluid resistance in open channel flows are similar to those that cause head losses in pipeline flows. Therefore, it is possible to get considerable insight into channel resistance by utilizing what is known from experimentation and theory, and has been adopted into practice in pipe fluid friction. The Darcy–Weisbach equation, $h_f = f(L/D)(V^2/2g)$ defines the frictional head loss in a pipe flow, in which f is a friction factor whose magnitude depends upon the relative roughness of the pipe wall, e/D, and the Reynolds number of the flow in the pipe. The Darcy–Weisbach equation is accepted as the fundamentally sound and best method for computing head losses or pressure drops in pipelines due to known flow rates. Therefore, this equation will be compared with Chezy's equation. The Darcy–Weisbach equation indicates that the slope of energy line, or the head loss, h_f, divided by the length, L, of pipe over which this loss occurs equals a friction factor, f, divided by the pipe diameter (which is a representative parameter for the pipe size with dimensions of length), multiplied by the velocity head, $V^2/2g$. If the hydraulic radius R_h multiplied by 4 is used in place of the pipe diameter, then the Darcy–Weisbach equation can be written as

$$S = \frac{h_f}{L} = \frac{fV^2}{4R_h(2g)} = \frac{fQ^2}{8R_h(gA^2)} \tag{2.2}$$

in which the friction factor, f, is a function of Reynolds number and the relative roughness, e/D, of the pipe wall (e is the roughness of the pipe wall and D is the diameter of the pipe). The substitution of $4R_h$ in place of D can be justified by noting that for a pipe, the hydraulic radius that equals the area divided by the perimeter is $R_h = (\pi D^2/4)/(\pi D) = D/4$.

It is worth noting that if f is dimensionless, then the terms separated by equal signs in Equation 2.2 are all dimensionless. Thus, Equation 2.2 can be obtained from dimensional analysis, and this analysis leads to the conclusion that f does depend on the two dimensionless parameters, R_e (Reynolds number) and e/D. The friction factor f will have the same value regardless of whether ES or SI units are used.

A comparison of Equation 2.1a and b with Equation 2.2 gives the following relationship between the Darcy–Weisbach friction factor, f, and Chezy's coefficient, C:

$$C = \sqrt{\frac{8g}{f}} \tag{2.3}$$

For pipe flow, it is known that the friction factor f can be defined well by a Moody diagram that is a plot of f as the ordinate against Reynolds number as the abscissa with different curves for different values of relative roughness e/D. This diagram is defined by the following equations for the following four different types of flow that occur:

1. Laminar flow

$$f = \frac{64}{R_e}$$

2. Hydraulically smooth flow (the wall roughness are well embedded within the laminar sublayer, less than 1/5 its size, and consequently have no influence on the magnitude of f)

$$\frac{1}{\sqrt{f}} = 2\mathrm{Log}_{10}(R_e\sqrt{f}) - 0.8$$

3. Transitional zone (in which both R_e and e/D determine f's magnitude; this is the Colebrook–White equation)

$$\frac{1}{\sqrt{f}} = -2\mathrm{Log}_{10}\left(\frac{e}{3.7D} + \frac{2.52}{R_e\sqrt{f}}\right) = 1.14 - 2\mathrm{Log}_{10}\left(\frac{e}{D} + \frac{9.35}{R_e\sqrt{f}}\right)$$

4. Wholly rough zone (in which R_e no longer effects f's magnitude)

$$\frac{1}{\sqrt{f}} = 1.14 - 2\text{Log}_{10}\left(\frac{e}{D}\right) = 2\text{Log}_{10}\left(\frac{3.7D}{e}\right)$$

Equations that define Chezy's C might be determined by substituting Equation 2.3 into the above four equations. When doing this for laminar flow, the following equation results:

$$C = \sqrt{\frac{gR_e}{8}} \tag{2.4}$$

Equation 2.4 applies for flows in which the Reynolds number is less than about 2100. Experimentation has shown that the value of 8 in Equation 2.4 does not hold constant for all channel, but no fully accepted range of values has been established. Since laminar flows are rare when dealing with water as the fluid, the value of 8 can probably be used, at least for a reasonable first approximation.

Substituting Equation 2.3 into the hydraulically smooth equation gives

$$C = \sqrt{32g}\text{Log}_{10}\left(\frac{gR_e}{0.887C}\right) \tag{2.5}$$

There is some question about whether the value 0.887 should be modified slightly to fit available experimental data. However, since very limited quality experimental data are available for hydraulically smooth flow, the derived value will be used in this book.

For flows that occur in the transitional zone in which both the relative roughness of the channel wall and the Reynolds number have an influence on the magnitude of the frictional resistance, the constants that are obtained from substituting Equation 2.3 into the above transitional equation might be modified slightly to give a better fit of some experimental data. (See ASCE Task Force, 1963.) With these modified coefficients the equation is

$$C = -\sqrt{32g}\text{Log}_{10}\left(\frac{e}{12R_h} + \frac{0.884C}{R_e\sqrt{g}}\right) \tag{2.6}$$

If the original coefficients that come from the Colebrook–White or other experimentally based equation are preferred, then Equation 2.6 can be written in the more general form

$$C = -c\text{Log}_{10}\left(\frac{e}{aR_h} + \frac{bC}{R_e\sqrt{g}}\right) \tag{2.6a}$$

in which the a, b, and c can be given slightly different values. For example, when using the Colebrook–White equation directly a = 12, b = 0.887, and $c = (32\,g)^{1/2}$.

For the wholly rough zone substitution of Equation 2.3 into the wholly rough equation that defines the Moody diagram gives

$$C = -\sqrt{(32g)}\text{Log}_{10}\left(\frac{e}{12R_h}\right) = \sqrt{(32g)}\text{Log}_{10}\left(\frac{12R_h}{e}\right) \tag{2.7}$$

These equations are summarized in Table 2.1, and Equations 2.5 through 2.7 are plotted on Figure 2.1 to give "Chezy C" diagram for turbulent flows. If Equation 2.4 were plotted on a left addition to Figure 2.1, it would result in a straight line with a slope of 1/2, since power equations such as Equation 2.4 plot as straight lines on log–log graph paper with the exponent in the equation giving the slope on this plot.

TABLE 2.1
Summary of Equations That Define Chezy's Coefficient, C

Type of Flow	Equation Giving C	Equation No.	Range of Application
Laminar	$C = (gR_e/8)$	2.4	$R_e < 2100$
Hydraulically smooth	$C = \sqrt{32g}\log_{10}\left(\frac{R_e\sqrt{g}}{0.887C}\right)$	2.5	$R_e < 2100$
Transition	$C = -\sqrt{32g}\log_{10}\left(\frac{e}{12R_h} + \frac{0.884C}{R_e\sqrt{g}}\right)$	2.6	$2100 < R_e < V_e/(C\nu) = 100$
Wholly rough	$C = -\sqrt{32g}\log_{10}\left(\frac{e}{12R_h}\right)$ or $C = \sqrt{32g}\log_{10}\left(\frac{12R_h}{e}\right)$	2.7	$R_e > V_e/(C\nu) = 100$

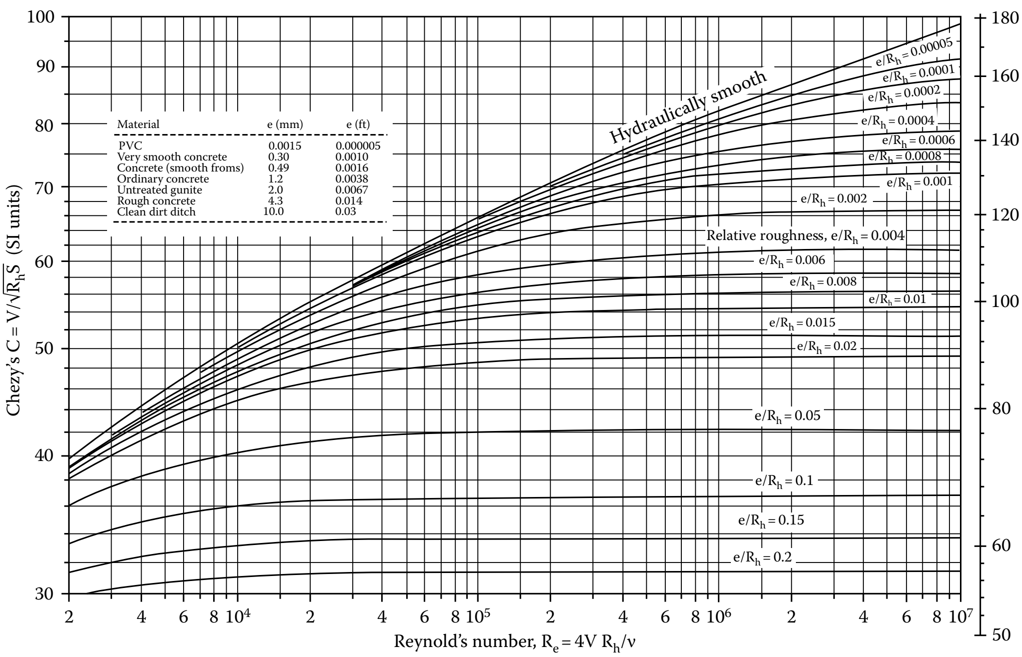


FIGURE 2.1 Diagram for Chezy's C for use in determining the flow rate, velocity and slope of the energy line, or head loss in open channels.

The friction factor f in the Darcy–Weisbach equation has a value that is independent of the units used. However, since C is not dimensionless, but has the dimensions of the square root of gravity, e.g., $L^{1/2}/T$, its value will be different when using SI units than when using ES units. In Figure 2.1, the values for C when using SI units are the left-side ordinate, whereas the right ordinate applies when using ES units. An alternative to having different values of C for different units would be to modify Chezy's equation to include g, or define the Chezy equation as

$$V = C_1\sqrt{gR_hS} \tag{2.1c}$$

However, historical developments have not done this, and therefore you must be sure that the appropriate value of C is used to the system of units that you are using.

Since many open channel flows do fall within the transitional type of flow, some discussion of the characteristics of Equation 2.6 are in order. It should be noted that C occurs on both sides of Equation 2.6 (the same is true of Equation 2.5). It is not possible to rearrange this equation so that C is on one side of the equation all by itself. Equations of this type are referred to as implicit equations, since an explicit solution of them is not possible. Use of general iterative techniques, such as the Newton method described in Appendix B can be used. However, because of the nature of Equations 2.5 and 2.6 they can be solved by a simple feedback iteration, called a Gauss–Seidel iteration, in which the C solved for on the left side of the equal sign is used for C on the right side of the equal sign for the next iteration. The value produced by the explicit Equation 2.7 can be used as the initial starting value for this iterative solution or easier, just start with a reasonable guess. The following few lines of FORTRAN and C code illustrate implementation of this iterative solution for a trapezoidal channel. For other types of cross sections, the two function statements at the top of this listing that define the area, A(B,FM,Y) and the wetted perimeter, P(B,FM,Y) need to be modified.

Program CHEZYC.FOR

```
      A(B,FM,Y)=(B+FM*Y)*Y
      P(B,FM,Y)=B+2*Y*SQRT(FM*FM+1.)
5     WRITE(6,*)' Give:B,FM,Y,G,VISC,E,Q'
      READ(5,*,ERR=30) B,FM,Y,G,VISC,E,Q
      AR=A(B,FM,Y)
      V=Q/AR
      G8=.884/SQRT(G)
      SQG=SQRT(32.*G)
      C1=SQG*ALOG10(12.*AR/P(B,FM,Y)/E)
10    RH=AR/P(B,FM,Y)
      RE=4.*V*RH/VISC
      C=-SQG*ALOG10(E/(12.*RH)+G8*C1/RE)
      IF(ABS(C-C1).LT. 1.E-8) GO TO 20
      C1=C
      GO TO 10
20    WRITE(6,*)' CHEZYS COEF=',C
      GO TO 5
30    STOP
      END
```

Program CHEZYC.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
float a(float b,float m,float y){return (b+m*y)*y;}
float p(float b,float m,float y){return b+2.*y*sqrt(m*m+1.);}
void main(void){float b,m,y,g,visc,e,q,ar,v,g8,rh,re,c,c1,sqg;
 printf("Give: b,m,Y,g,Visc,e,Q\n");
 scanf("%f %f %f %f %f %f %f",&b,&m,&y,&g,&visc,&e,&q);
 ar=a(b,m,y); v=q/ar; g8=.884/sqrt(g);sqg=sqrt(32.*g);
 c=sqg*log10(12.*ar/p(b,m,y)/e);
 do {c1=c; rh=ar/p(b,m,y); re=4.*v*rh/visc;
   c=-sqg*log10(e/(12.*rh)+g8*c1/re);}while(fabs(c-c1)<1.e-8);
 printf("CHEZYS COEF =%f\n",c);}
```

A typical problem involves considerably more than solving one of the equations in Table 2.1 for Chezy's coefficient. The types of problems associated with steady flow can be categorized as follows:

1. The flow rate, or velocity is unknown, and all other variables are known.
2. The depth is unknown and all other variables are known. This type of problem typically asks a question like: With this given channel what will the depth of flow be if the flow rate equals Q.
3. One of the variables associated with the channel size is unknown, but the flow rate is known. This type of problem can be consider a design problem in which the size of channel needed to convey a specified flow rate is wanted. For a trapezoidal channel the unknown may be the bottom width, or the side slope, and for a circular section the diameter is the unknown.
4. All variables are known except the wall roughness.
5. The slope of the channel bottom is unknown, and all other variables are known. Of all problems this is the easiest, since its solution can be obtained most directly, by (a) solving the appropriate equation for Chezy's coefficient, and (b) solving Chezy's equation for S. (When the slope, S, refers to the channel bottom subscript o will be used and when the slope refers to the energy line subscript f will be used e.g., in most subsequent equations S_o or S_f will appear in Chezy's as well as Mannings equation.)

Problems in any of these categories might be view, as a mathematical problem of solving two nonlinear simultaneous equations; Chezy's equation and the appropriate equation from Table 2.1 that defines Chezy's coefficient. Appendix B describes the Newton method for solving systems of nonlinear equations. Alternately software packages, such a Mathcad, TK-Solver, or MATLAB can be used, or math packs for pocket calculators might be used. In addition to the procedure described above for case (5) problems under category (1) can be solved by cycling through the following steps: (a) solving for C (Equation 2.6), (b) using this C compute Q, or V from Equation 2.1a and b, and (c) based on this Q, or V update Reynolds number and repeat steps (a) through (c) until a small enough change occurs between consecutive values of Q or V that you are willing to accept the results. Convergence of this procedure will be rapid because changes in flow rate have a relatively small effect on the value of C. Should the flow be in the wholly rough zone then only one cycle of steps (a) and (b) above completes the solution since Chezy's coefficient is independent of the flow rate.

Typical values for wall roughnesses that are appropriate are given in Table 2.2.

EXAMPLE PROBLEM 2.1

What is the flow rate in a trapezoidal channel with b = 10 ft, m = 1.5, and a bottom slope of S_o = 0.0005 if the depth of flow is measured equal to 5 ft.

Solution

This problem falls in Category (1) above. The solution might begin by solving for C from Equation 2.7, which gives C = 127.49. Next the implicit Equation 2.6 is solved by the Gauss–Seidel iteration based on an assumed Reynolds number, i.e., 1.0E6 (or if one wishes the above C could be used in

TABLE 2.2
Value of Wall Roughness, e, for Different Channel Materials

Material	e (m)	e (ft)
PVC	0.0000015	0.000005
Very smooth concrete	0.00030	0.0010
Concrete (smooth forms)	0.00049	0.0016
Ordinary concrete	0.0012	0.0038
Untreated gunite	0.0020	0.0067
Rough concrete	0.0043	0.014
Clean dirt ditch	0.0100	0.03

Chezy's equation to compute the Reynolds number). The result is C = 125.15. Now using this C the velocity is V = 4.944 fps from Chezy's equation, and the associated Reynolds number equals 5,019,874 (assuming v = 1.23E–5). To get the solution the steps of (a) solving C from Equation 2.6 based on the most recent Reynolds number, (b) solve V from the Chezy equation (Equation 2.1a and b) and updating $R_e = 4R_h/v$. The results are iteration # 2, C = 126.98, V = 5.017 fps, R_e = 5,093,390; iteration # 3, C = 126.99, which is close enough giving a flow rate Q = 439.0 cfs.

EXAMPLE PROBLEM 2.2

A trapezoidal channel with b = 8 ft, m = 1.2, and a bottom slope of 0.0006 is to convey a flow rate of Q = 300 cfs. The wall roughness of the channel is e = 0.004 ft. Determine the depth of flow in this channel.

Solution

An effective way to solve this problem is to use the Newton method (Appendix B) to solve Chezy equation and Equation 2.6 simultaneously for C and Y. For the Newton method these equation can be written as

$$F_1 = C + (32g)^{1/2} \operatorname{Log}_{10}\left(\frac{e}{12R_h} + \frac{0.884C}{\left(R_e\sqrt{g}\right)}\right) = 0 \tag{1}$$

$$F_2 = Q - CA(R_hS)^{1/2} = 0 \tag{2}$$

Using the Newton method the unknown vector, consisting of the two values C and Y, is updated by the following iterative equation:

$$\begin{Bmatrix} C \\ Y \end{Bmatrix}^{(m+1)} = \begin{Bmatrix} C \\ Y \end{Bmatrix}^{(m)} - \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix}$$

in which the vector z is the solution to the linear system of equations:

$$\begin{bmatrix} \dfrac{\partial F_1}{\partial C} & \dfrac{\partial F_1}{\partial Y} \\ \dfrac{\partial F_2}{\partial C} & \dfrac{\partial F_2}{\partial Y} \end{bmatrix} \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix}$$

Instead of actually taking the partial derivatives shown above a numerical approximation can be used. This numerical approximation evaluates the function (i.e., equation) twice with the second evaluation based on decreasing (or increasing) the variable that the derivative is taken with respect to by a small increment and then dividing the two values of the function by this increment.

The following FORTRAN program can be used to obtain this solution. It is designed to solve for C as well as any of the following variables: e, b, m, Y, Q, or S in a trapezoidal channel. Input to this program as well as the values to solve this problem consist of: 1 the number of the unknown variable which is 4 for Y for this problem, the list of known, including a guess for Y, in the following order: e = 0.004, b = 8, m = 1.2, Y = 4(guess), Q = 300, S = 0.0006, g = 32.2, and kinematic viscosity = 1.23E–5. The solution produced by the program is Y = 4.46 ft. The problem can be solved by using Figure 2.1 in connection with Equation 2.1a and b. You should at least verify Chezy's C from Figure 2.1. The value of C from the above procedure is C = 125.

Listing of FORTRAN program to solve problem using the Chezy equation (CH2PR2):

```
      REAL X(6),D(2,2),F(2)
      EQUIVALENCE (E,X(1)),(B,X(2)),(FM,X(3)),(Y,X(4)),(Q,X(5)),
     &(S,X(6))
      A(B,FM,Y)=(B+FM*Y)*Y
      RH(AR,B,FM,Y)=AR/(B+2.*Y*SQRT(FM*FM+1.))
      F1(C,SQG,RH1,RE,G8,E)=C+SQG*ALOG10(E/(12.*RH1)+G8*C/RE)
      F2(Q,C,AR,RH1,S)=Q-C*AR*SQRT(RH1*S)
```

```
1     WRITE(6,*)' Give: No. of UNK,1-e,2-b,3-m,4-Y,5-Q,6-S,g & Vis'
      READ(5,*) IUN,X,G,VISC
      IF(IUN.LT.1 .OR. IUN.GT.6) GO TO 1
      G8=.884/SQRT(G)
      SQG=SQRT(32.*G)
      C=SQG*ALOG10(12.*RH(A(B,FM,Y),B,FM,Y)/E)
20    AR=A(B,FM,Y)
      RH1=RH(AR,B,FM,Y)
      RE=4.*Q*RH1/(VISC*AR)
      F(1)=F1(C,SQG,RH1,RE,G8,E)
      F(2)=F2(Q,C,AR,RH1,S)
      C=.98*C
      D(1,1)=(F(1)-F1(C,SQG,RH1,RE,G8,E))/(.02040816*C)
      D(2,1)=(F(2)-F2(Q,C,AR,RH1,S))/(.02040816*C)
      C=C/.98
      X(IUN)=.95*X(IUN)
      AR=A(B,FM,Y)
      RH1=RH(AR,B,FM,Y)
      RE=4.*Q*RH1/(VISC*AR)
      D(1,2)=(F(1)-F1(C,SQG,RH1,RE,G8,E))/(.05263158*X(IUN))
      D(2,2)=(F(2)-F2(Q,C,AR,RH1,S))/(.05263158*X(IUN))
      X(IUN)=X(IUN)/.95
      FAC=D(2,1)/D(1,1)
      D(2,2)=D(2,2)-FAC*D(1,2)
      F(2)=F(2)-FAC*F(1)
      DIF=F(2)/D(2,2)
      X(IUN)=X(IUN)-DIF
      DIF1=(F(1)-DIF*D(1,2))/D(1,1)
      C=C-DIF1
      IF(ABS(DIF)+ABS(DIF1).GT. .001) GO TO 20
      WRITE(6,100) C,X(IUN),X
100   FORMAT(' C=',F8.2,' Unknown=',F10.3,/,' e=',F10.5,/,
     &' b=',F8.2,/,' m=',F8.2,/,' Y=',F8.3,/,' Q=',F10.2,/,
     &' S=',F8.5)
      WRITE(6,*)' Give 1 to solve another problem; otherwise 0'
      READ(5,*) IUN
      IF(IUN.EQ.1) GO TO 1
      STOP
      END
```

EXAMPLE PROBLEM 2.3

Determine the depth of flow in a circular channel with D = 10 ft, if its bottom slope equals 0.0005 and its wall roughness equals 0.004 ft and it is to convey a flow rate of Q = 150 cfs.

Solution

The above program can be modified by changing the function statement to obtain the area by a function subprogram, and function statement for the hydraulic radius to apply for a circular channel. These statements could consist of

```
      FUNCTION A(D,BETA,Y)
      BETA=ACOS(1.-2.*Y/D)
      A=D*D/4.*(BETA-COS(BETA)*SIN(BETA))
      RETURN
      END
```

and

```
      RH(AR,D,BETA,Y)=AR/(D*BETA)
```

The above listing might be altered with BETA replacing FM, and or the present FORTRAN names used with different meaning. The solution gives Y = 4.613 ft, and Chezy's C = 123.1.

EXAMPLE PROBLEM 2.4

You are to size a trapezoidal channel that is to carry a flow rate of $Q = 50\,m^3/s$. The slope of the channel is 0.0012 and it is to be made of formed concrete. For stability of the channel sides their slopes are to be 1.5, and the depth is not to exceed 2 m. What should the bottom width be?

Solution

The computer program of Example Problem 2.2 will solve this problem. If this program is used the input consists of: 2,.00049,3,1.5,2,50,.0012,9.81,1.14E−6. The solution is b = 4.94 m. If Mathcad is available to you it would be a excellent experience to use it, or some other software package to get the same solution.

2.3 COMBINING THE CHEZY AND THE CHEZY C EQUATIONS

An alternative to solving Chezy's equation and the Chezy C equation for the transitional zone simultaneously for any of the variables is to eliminate C first by solving for it from Chezy's equation, and then substituting this in the equation that applies within the transitional zone where it occurs on both sides of the equal sign. By eliminating C between these two equations there is one equation and one of the variables can be solved as the unknown. The resulting equation is implicit for all variables except e, and so in general must be solved by an iterative technique. The HP48 calculator's SOLVE capability can be used; however, because A and P are functions of the depth and size variables the resulting equation becomes long and complex. However, this approach is readily implemented in a computer program, in TK-Solver, or Mathcad models where separate statements can be used to define A and P.

Solving Chezy's equation for C gives

$$C = \frac{Q\sqrt{P/(AS_0)}}{A} \quad \text{or} \quad C = V\sqrt{\frac{P}{AS_0}}$$

The transitional zone equation that gives C can be written as

$$F(\xi) = C + \sqrt{32g}\,\mathrm{Log}_{10}\left(\frac{eP}{12A} + \frac{0.221\nu CP}{Q\sqrt{g}}\right) = 0$$

If the first of the two equations above, that contains Q, is used that gives C using Q then the following equation results:

$$F(\xi) = \frac{Q\sqrt{P/(AS_0)}}{A} + \sqrt{32g}\,\mathrm{Log}_{10}\left(\frac{eP}{12A} + \frac{0.221\nu P^{\frac{3}{2}}}{A^{\frac{3}{2}}\sqrt{gS_0}}\right) = 0$$

and if the second of the above two equations that gives C using the velocity V then the following equation results:

$$F(\xi) = V\sqrt{\frac{P}{(AS_0)}} + \sqrt{32g}\,\mathrm{Log}_{10}\left(\frac{eP}{12A} + \frac{0.221\nu P^{\frac{3}{2}}}{A^{\frac{3}{2}}\sqrt{gS_0}}\right) = 0$$

After solving either of these two equations depending on whether the flow rate Q is given (or the unknown) or whether the velocity is given (or the unknown), then Chezy's equation is solve for the coefficient C if this value is desired. The program CHEZYCTC.FOR, which is given below, implements such a solution using the Newton method to solve for the selected unknown from the variables: m, b, S, Y, e, Q, or V if the channel is trapezoidal, and D, S, Y, e, Q, or V if the channel is circular. The technique used to accommodate both trapezoidal and circular channels is to not use X(1) (for m) and change V(2) to D (the Character string) if the channel is circular. When IC = 1, for a circular channel, then the equation that gives A and P for a circular channel are used; otherwise those that give these quantities for a trapezoidal channel are used. (See Statements starting with label 55.) The approach is very similar to that used in solving the DW and CW equations. This program does not contain the logic, however to generate a guess for the unknown, that is needed in the Newton method. Rather the user must supply this guess as well as the known values.

The variable and rule sheets from TK-Solver are listed below the program listing that handle first the equation that assumes that the flow rate Q is the flow variable, which is in the combined equation with the other channel property variables, and the second is for the equation that involves the V as the flow variable.

Listing of program CHEZYCTC.FOR for solving the Chezy equation for any of the variables for both a trapezoidal and a circular channel

```
      REAL X(7)
      CHARACTER*19 FMT/'(1X,A2,3H = ,F10.4)'/
      CHARACTER*1 V(7)/'m','b','S','Y','e','Q','V'/
1     WRITE(*,*)' Give 1=ES or 2=SI(or 0/=STOP),',
     &'0=trap or 1=cir. & Visc'
      READ(*,*) II,IC,VISC
      IF(II.LT.1) STOP
      G=32.2
      IF(II.GT.1) G=9.81
      VISC2=.221*VISC/SQRT(G)
      G32=SQRT(32.*G)
      IF(IC.GT.1) THEN
      I2=2
      V(2)='D'
      ELSE
      I2=1
      V(2)='b'
      ENDIF
      WRITE(*,100)(I,V(I),I=I2,7)
100   FORMAT(' Give No. of Unknown',/(I2,' - ',A2))
      READ(*,*) IU
      IF(IU.GT.5) GO TO 10
      WRITE(*,*)' Give 1 if Q will be given or 2',= if V is known'
      READ(*,*) IV
      GO TO 12
10    IV=IU-5
12    WRITE(*,*)' Give values to knowns &=,= GUESS for unknown'
      I3=7
      IF(IV.EQ.1) I3=6
      DO 20 I=I2,I3
```

```
      IF(IV.EQ.2 .AND. I.EQ.6) GO TO 20
      WRITE(*,"(A2,' = ',\)") V(I)
      READ(*,*) X(I)
20    CONTINUE
50    M=0
52    XX=X(IU)
      X(IU)=1.005*X(IU)
      DX=X(IU)-XX
55    IF(IC.EQ.1) THEN
      COSB=1.-2.*X(4)/X(2)
      BETA=ACOS(COSB)
      A=.25*X(2)**2*(BETA-COSB*SIN(BETA))
      P=X(2)*BETA
      ELSE
      A=(X(2)+X(1)*X(4))*X(4)
      P=X(2)+2.*X(4)*SQRT(X(1)**2+1.)
      ENDIF
      ADL=G32*ALOG10(X(5)*P/(12.*A)+VISC2*(P/A)**1.5/SQRT(X(3)))
      IF(IV.EQ.1) THEN
      F=X(6)*SQRT(P/(A*X(3)))/A+ADL
      ELSE
      F=X(7)*SQRT(P/(A*X(3)))+ADL
      ENDIF
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 60
      X(IU)=XX
      F1=F
      GO TO 55
60    DIF=DX*F/(F1-F)
      X(IU)=XX-DIF
      IF(ABS(DIF).GT. .00001 .AND. M.LT.30) GO TO 52
      IF(IV.EQ.1) THEN
      X(7)=X(6)/A
      ELSE
      X(6)=A*X(7)
      ENDIF
      DO 70 I=1,7
      FMT(18:18)='3'
      IF(I.EQ.3 .OR. I.EQ.5) FMT(18:18)='6'
70    WRITE(*,FMT) V(I),X(I)
      WRITE(*,FMT) 'C',X(7)*SQRT(P/(A*X(3)))
      GO TO 1
      END
```

```
═══════════ VARIABLE SHEET ════════════
St Input──── Name─── Output─── Unit────
             A       75
   10        b
   1         m
   5         Y
```

```
            P       24.142136
            Q       478.55564
   .001     S
   32.099844 g32
   .01      e
   .0000141 v
   32.2     g
            C       114.47967
============== RULE SHEET ================
S Rule-------------------------------------------
   A=(b+m*Y)*Y
   P=b+2*Y*sqrt(m^2+1)
   Q*sqrt(P/(A*S))/A+g32*Log(e*P/(12*A)+.221*v/
      sqrt(g*S)*(P/A)^1.5)=0
   C=Q*sqrt(P/(A*S))/A

========== VARIABLE SHEET ============
St Input---- Name--- Output--- Unit-----
            A       75
   10       b
   1        m
   5        Y
            P       24.142136
            V       6.3807419
   .001     S
   32.099844 g32
   .01      e
   .0000141 v
   32.2     g
            C       114.47967
============== RULE SHEET ================
S Rule-------------------------------------------
   A=(b+m*Y)*Y
   P=b+2*Y*sqrt(m^2+1)
   V*sqrt(P/(A*S))+g32*Log(e*P/(12*A)+.221*v/
      sqrt(g*S)*(P/A)^1.5)=0
   C=V*sqrt(P/(A*S))
```

For laminar flow or water to occur in open channels either the depth or the velocity must be very small. For example, assume water at 15.6°C (60°F) so its kinematic viscosity is $v = 1.123 \times 10^{-6}$ m²/s (1.217×10^{-5} ft²/s), and that the largest value of Reynolds number, VR_h/v allowed for laminar flow is 500, and that the channel is very wide so that $R_h = Y$, then the product of the velocity time the depth $VY \le 500(1.123 \times 10^{-6}) = 0.0005615$ for SI units or $VY \le 0.0061$ for ES unit. The tables below show these limiting values. The Froude number F_r in the third column of these tables is defined by $F_r = V/\sqrt{gY}$. Notice that as the depth becomes very small, in the order of 0.01 ft, or 0.003 m, and the velocity consequently larger that the flow may becomes supercritical (F_r greater than 1). To have conditions right to allow a flow to be simultaneously laminar and supercritical are not common. Furthermore, when the depth becomes small enough for Froude numbers to be larger than unity then the surface tension of the water becomes a significant factor. When water flows in very thin sheets on steep surfaces, it tends to form thread like streamlets. Also most channel surfaces, such as gutter, or road way beds, where very small depths of open channel water flows may be found in practice, are not smooth enough and the water will actually be seen to flow in the lower indentations. The sheet flow over watershed surfaces that occurs from rainfall, which does not infiltrate into the soil, tends to erode the smaller particles and by so doing forms a system of mini channel, which will grow in size in time especially if the rainfall is intense. The last columns in these tables give the bottom slope of the channel that would be required for the flow to take place as computed by the Chezy equation. Note that for super critical flows the bottom slopes must also be very large.

Limiting Depths, Velocities and Froude Numbers for Laminar Flow of Water in Wide Open Channels. T = 15.6°C (60°F) $\nu = 1.123 \times 10\,m^2/s$ ($1.217 \times 10\,ft^2/s$)

V (ft/s)	Y (ft)	F_r	S_o	V (m/s)	Y (m)	F_r	S_o
0.020	0.304	0.006	0.000015	0.0060	0.0936	0.0063	0.000008
0.040	0.152	0.018	0.000117	0.0120	0.0468	0.0177	0.000062
0.0625	0.097	0.035	0.000447	0.0200	0.0281	0.0381	0.000288
0.125	0.049	0.100	0.003577	0.0500	0.0112	0.1506	0.004495
0.250	0.024	0.282	0.028619	0.1000	0.0056	0.4261	0.35962
0.500	0.012	0.799	0.228955	0.1500	0.0037	0.7828	0.121372
0.010	0.609	1.072	0.412688	0.0030	0.1872	1.0910	0.235792
0.025	0.243	0.271	0.026412	0.0050	0.1123	0.5071	0.050931
0.050	0.122	0.096	0.003302	0.0100	0.0561	0.1793	0.006366
0.100	0.061	0.034	0.000413	0.0200	0.0281	0.0634	0.000796
0.200	0.030	0.012	0.000052	0.0500	0.0112	0.0160	0.000051
0.300	0.020	0.007	0.000015	0.1000	0.0056	0.0057	0.000006

2.4 EMPIRICAL FORMULA: USE OF MANNING'S EQUATION

Before the turn of the twentieth century, systematic research was underway to define better fluid resistance in open channel flow. It was recognized then that C in the Chezy equation was not constant under all flow conditions in a given channel. Bazin proposed the following formula that found use in the past to better define C:

$$C = \frac{157.6}{1 + m\sqrt{R_h}} \quad \text{(for ES units)}$$

in which m takes on a different value depending on the roughness of the channel wall. In 1868, Gauckler proposed that for flat slopes in open channel flow, C varies as the sixth root of the hydraulic radius. Others came to the same conclusion and the result has now been widely accepted throughout the world and is known in the United States as Manning's formula even though the name Manning is a misnomer and it has been proposed that the formula be called Gauckler–Manning's equation in part to undo the incorrect naming. In Europe the same formula is called the Strickler formula. Because of its wide use, this will be the formula used in this text book as an alternative to Chezy's formula. Manning's equation can be written as

$$V = \frac{C_u}{n} R_h^{2/3} \sqrt{S_o}$$

or

$$Q = \frac{C_u}{n} A R_h^{2/3} \sqrt{S_o} = \frac{C_u}{n} A \frac{A^{2/3}}{P^{2/3}} \sqrt{S_o} = \frac{C_u}{n} \frac{A^{5/3}}{P^{2/3}} \sqrt{S_o} \tag{2.8}$$

in which C_u equals 1 when using SI units, and $C_u = 1.486$ (the cube root of the number of feet per meter) when using ES units, and n is the roughness coefficient of the channel wall. In this equation the following symbols apply: A = the cross section of the flow in ft^2 when using ES units and in m^2 when using SI units; P is the wetted perimeter and is the length of contact between the water and the channel when viewed in a direction normal to the flow direction, and has units of ft in the ES system, and m in the SI system; R_h is the hydraulic radius, which is defined as the area A divided by the wetted perimeter P, with units of ft in ES units and m in SI units; and S_o is the slope of the channel bottom, which for uniform flow equal the slope of both the water surface in the channel as well as the energy line of the flow. The slope of the energy line equal the head loss divided by the length over which this loss occurs. Therefore S_o might be thought of as h_L/L for uniform flow. Since three digits of precision cannot be maintained in the selection of n, it is common to use $C_u = 1.49$ in ES units. Typical values for use in Manning's equation for different material that channel are constructed from are given in Table 2.3. If n value larger than 0.05, the largest value in Table 2.3, is needed to describe a given channels flow, other mechanisms than just fluid friction are likely involved and Manning's equation is probably inappropriate, e.g., a constant value of n will not describe depths, etc. over much of a range of flow rates. This condition exists for very small velocity flows that may be approaching flow through a porous media, or very large velocity flows.

It is worth noting that Manning's formula indicates that the head loss is proportional to the square of the velocity, or flow rate. On the Chezy C diagram this corresponds to the wholly rough zone. When the Reynolds number becomes small with a magnitude of 100,000 or less then according to the Chezy equation the head loss is proportional to the velocity to a power less than 2, but greater than 1. For laminar flow, Equation 2.4 indicates that the head loss is proportional to the velocity to the first power. Thus the use of Manning's equation will essentially duplicate the results obtained from Chezy's formula for very large Reynolds numbers when the flow lies in the wholly rough zone, provided corresponding roughness values are selected. On the other hand, the use of Manning's equation is questionable for low-velocity flows in small smooth channels.

A natural question is: What is the relationship between the values of Manning's n and the equivalent sand roughness e used in connection with Chezy's formula? The answer is that there is no direct

TABLE 2.3
Typical Values for Manning's n

Channel Material	n
Lined channels	
Smooth brass, glass, lucite, PVC	0.010
Cement plaster	0.011
Planed lumber, unpainted steel, trowelled concrete	0.012
Unplaned lumber, smooth asphalt, vitrified clay, brick in cement mortar, cast iron	0.013
Asphalt (rough), untreated gunite	0.016
Rough concrete	0.020
Corrugated metal	0.023
Natural channels	
Clean excavated earth	0.023
Earth (good condition), rock excavation, gravel (straight chan.)	0.025
Earth (straight with some grass)	0.026
Earth (winding, no grass), clean natural beds	0.030
Gravel beds (plus large boulders)	0.040
Earth (winding), weedy streams	0.050

relationship, or simple equation that can give n from e, or e from n. For any given e, the value of n is different depending on the channel type and size, and the flow rate, i.e., Reynolds number of this flow. To get an n that corresponds to an e for any given situation, one must solve both equations. For example, if one wanted to determine what n corresponds to e = 0.004 ft, it is first necessary to decide what channel and flow rate this correspondence is to apply for. To illustrate, assume a flow rate of 400 cfs occurs in a rectangular channel with a bottom width of 10 ft and bottom slope of 0.0005 has a known e = 0.004 ft. First the depth is solved from Chezy's equation. The depth is Y = 8.04 ft. Next Manning's equation is solved for n, giving n = 0.0141. Thus for this channel containing this flow rate the corresponding n = 0.0141 for an e = 0.004 ft. For this e, the value of n will be different for each different flow rate and each channel.

Table 2.4 illustrates this variation of n with several different variables that can be changed in a trapezoidal channel. Since a rectangular channel is a special trapezoidal channel with side slope m = 0, this table includes a couple of rectangular channels. All of the values for n given in this table correspond to a equivalent sand roughness of e = 0.004 ft in the Chezy equation. The first three columns in Table 2.4 are for the rectangular channel used in the above illustration with a bottom width of 10 ft and a bottom slope of $S_o = 0.0005$. The flow rate was varied from 8 to 400 cfs in this channel. The second columns gives the depths that are obtained by solving Chezy's formula, and the third column indicates the value of n that is solved by Manning's equation for this column 2. This third column of n values shows that n increases with Q and Y from 0.0131 to 0.0141, or a 7.6% change over this range of conditions.

The next three columns, i.e., columns 4, 5, and 6 in Table 2.4, were obtained similarly except for a trapezoidal channel with a side slope of 1.5 (with b = 10 ft and $S_o = 0.0005$ as for the rectangular channel). With the flow rate changing from 12 to 600 cfs, the variation in Manning's n is again 7.6%.

The next four groups of three columns each were obtained by holding the flow rate constant, letting the slope of the channel bottom vary, solving Chezy's equation for the depth, and based on this computed depth and other variables solving Manning's equation for n. The third column of each group of 3 shows the variation of n. These variations in n are from 3.9% to 5.1%.

The last three columns in Table 2.4 were obtained for a rectangular channel with a bottom slope of $S_o = 0.0005$ and the depth held constant at 2 ft. The first column of this group that gives the bottom width b was varied from 0.5 to 25.0 ft, and the flow rate in column 2 was obtained by solving Chezy's formula. The last column represents the solution of Manning's equation for this flow rate and channel. Note again a variation in the n value of 5.4%.

Table 2.5 represents the results from similar calculations to those used to get the values in Table 2.4 with the exception that they apply to circular cross sections, and the units are SI instead of ES. Approximately the same variations in n occur. Both of these tables represent common channel sizes, and the value of e = 0.004 ft is a typical value for man-made lined channel. The variation of 3%–8% shown in these tables is within the accuracy of selecting n for a given application. Therefore, we might conclude that for most practical problems it is satisfactory to use Manning's equation. Should the Reynolds number of the flow be less than 100,000, then it is probably best to use Chezy's formula even if it does entail a little more arithmetic. The accuracy of predicting flow rates and/or other variables in open channels cannot be expected to be better than the percentages shown in Tables 2.4 and 2.5.

If one assumes Manning's equation applies only in the wholly rough zone, then equating Chezy's and Manning's equations, $V = C(R_hS_o)^{1/2} = (C_u/n)R_h^{2/3}S_o^{1/2}$, with C defined by Equation 2.7 gives

$$\frac{C_u}{n}R_h^{1/6} = \sqrt{32g}\,\mathrm{Log}_{10}\left(\frac{12R_h}{e}\right)$$

If we take n as dimensionless, then this equation indicates C_u has dimensions of $L^{1/3}/t$. Making the assumption that n is dimensionless allows the same values of n to be used for both ES and SI units. This is what has occurred in practice, i.e., $C_u = 1$ for SI units, and for ES units C_u is the cubic root

TABLE 2.4
Variations of Manning's n with a Fixed Value of e = 0.004 ft ($\nu = 1.217 \times 10^{-5}$ ft²/s) and Other Variables Changed in a Trapezoidal Channel

Variables Held Constant

b = 10 ft, m = 0, S_o = 0.0005			b = 10 ft, m = 1.5, S_o = 0.0005			b = 10 ft, m = 0, Q = 300 cfs			b = 10 ft, m = 1.5, Q = 500 cfs			b = 2 ft, m = 0, Q = 5 cfs			b = 2 ft, m = 1.5, Q = 20 cfs			m = 0, Y = 2 ft, S_o = 0.0005		
Q	**Y**	**n**	**Q**	**Y**	**n**	**S_o**	**Y**	**n**	**S_o**	**Y**	**n**	**S_o**	**Y**	**n**	**S_o**	**Y**	**n**	**b**	**Q**	**n**
8	0.52	0.0131	12	0.64	0.0132	0.0001	12.39	0.0143	0.0001	8.05	0.0145	0.0001	2.77	0.0134	0.0001	2.58	0.0137	0.5	0.937	0.0130
80	2.39	0.0137	120	2.46	0.0137	0.0005	6.39	0.0141	0.0005	5.35	0.0142	0.0005	1.42	0.0132	0.0005	1.76	0.0134	1.0	2.758	0.0131
160	3.95	0.0139	240	3.62	0.0140	0.0010	4.88	0.0140	0.0010	4.46	0.0141	0.0010	1.08	0.0131	0.0010	1.49	0.0133	5.0	26.462	0.0135
240	5.37	0.0140	360	4.50	0.0141	0.0025	3.46	0.0138	0.0025	3.48	0.0139	0.0025	0.765	0.0130	0.0025	1.18	0.0132	10.0	61.932	0.0136
400	8.04	0.0141	600	5.88	0.0142	0.0050	2.69	0.0137	0.0050	2.87	0.0138	0.0050	0.594	0.0129	0.0050	0.988	0.0131	25.0	174.04	0.0137
Variation		7.6%			7.6%			4.4%			5.1%			3.9%			4.6%			5.4%

Note: Q is in cfs and Y is in ft, b is in ft.

TABLE 2.5
Variations of Manning's n with a Fixed Value of e = 0.001219 m ($\nu = 1.31 \times 10^{-6}$ m²/s) and Other Variables Changed in a Circular Channel

Variables Held Constant

D = 6 m, S_o = 0.0005			D = 2 m, S_o = 0.0005			D = 0.33333 m, S_o = 0.0005			D = 6 m, Q = 25 m³/s			D = 1 m, Q = 0.20 m³/s			D = 3 m, Y = 2 m			D = 0.5 m, Y = 0.3 m		
Q	**Y**	**n**	**Q**	**Y**	**n**	**Q**	**Y**	**n**	**S_o**	**Y**	**n**	**S_o**	**Y**	**n**	**S_o**	**Q**	**n**	**S_o**	**Q**	**n**
1.20	0.58	0.0135	0.073	0.204	0.0131	0.0007	0.0359	0.0134	0.0002	3.70	0.0147	0.0001	0.722	0.0136	0.0001	3.23	0.0142	0.0001	0.025	0.0134
6.00	1.29	0.0140	0.438	0.492	0.0134	0.0054	0.0987	0.0131	0.0004	2.97	0.0145	0.0003	0.498	0.0134	0.0005	7.26	0.0141	0.0005	0.056	0.0131
26.4	2.87	0.0145	1.02	0.770	0.0136	0.0122	0.1526	0.0131	0.0010	2.28	0.0144	0.0010	0.354	0.0132	0.0010	10.3	0.0141	0.0010	0.080	0.0131
40.0	3.77	0.0146	1.75	1.05	0.0137	0.0204	0.2087	0.0131	0.0020	1.57	0.0141	0.0020	0.294	0.0131	0.0020	14.6	0.0141	0.0020	0.113	0.0130
60.0	5.38	0.0147	3.43	1.83	0.0138	0.0306	0.3086	0.0131	0.0050	1.24	0.0139	0.0050	0.238	0.0130	0.0050	21.8	0.0140	0.0050	0.179	0.0130
Variation		8.8%			5.3%			2.3%			5.8%			4.6%			1.4%			3.1%

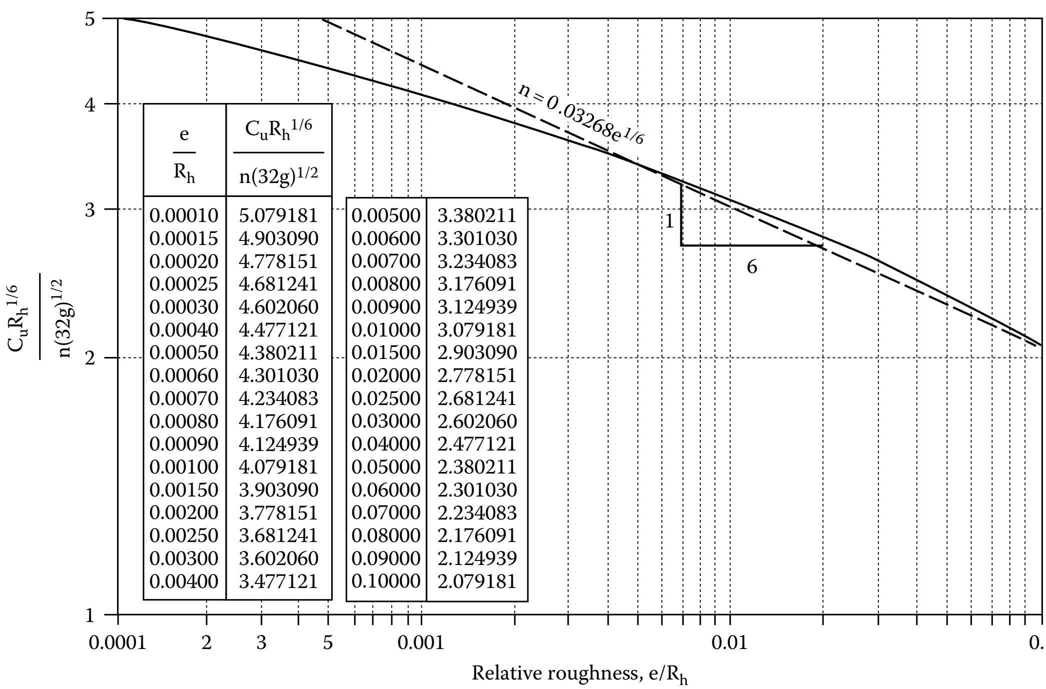
Note: Q is in m³/s and Y is in m, and D is in m.

of the number of feet per meter, or $C_u = (1/.3048)^{1/3} = 1.486$. However, it is generally accepted that n is not dimensionless. Often when working in one system of units (either ES or SI), the quantity on the right of the above equation, which is Chezy's C, as well as C_u are taken as dimensionless, and therefore the literature will often suggest that n has dimensions of $L^{1/6}$. A log–log plot of the above equation is given below. If n is assumed to have dimensions of $L^{1/6}$ then one would expect n to vary as the one-sixth root of the wall roughness size, e, or $n = Ke^{1/6}$ (where K is a constant). Such a relationship is shown on the graph as a dashed line, but any other line parallel to this line could be used depending on what n one selects to correspond to e for a given hydraulic radius. For the given dashed line R_h is taken equal to 3 ft and $C_uR_h^{1/6}/\{n(32g)^{1/2}\} = 2.079$ corresponding to $e/R_h = 0.1$ (Figure 2.2). Thus for e = 0.3 ft, n = 0.0267, or $n = 0.03268e^{1/6}$. This gives n = 0.0182 corresponding to e = 0.03 ft, and n = 0.0124 corresponding to e = 0.003 ft. Note from this graph that the dashed line would fit the curve closer if its slope were flatter; suggesting that the dimensions of n might be closer to $L^{1/7}$.

Let us now focus attention on solving Manning's equation, which can be solved directly if the flow rate Q, the roughness coefficient n, or the slope of the channel bottom S_o is the unknown. To solve for n, its place is interchanged in Equation 2.8 with Q. In solving for S_o Manning's equation becomes

$$S_o = \left\{\frac{nQ}{C_u}\frac{P^{2/3}}{A^{5/3}}\right\}^2 = \left\{\frac{nV}{C_u}R^{2/3}\right\}^2 = \left\{\frac{nQ}{C_uA}\left(\frac{P}{A}\right)^{2/3}\right\}^2 \tag{2.8a}$$

If one of the variables that goes into defining the cross-sectional area A, the wetted perimeter P, i.e., the hydraulic radius R_h is unknown, then Manning's equation becomes implicit in that variable, and must be solved by an iterative method such as the Newton's method, or by trial and error. Consider a trapezoidal section for example. For a trapezoid, the area and wetted perimeters are



e/R_h	$\frac{C_uR_h^{1/6}}{n(32g)^{1/2}}$	e/R_h	$\frac{C_uR_h^{1/6}}{n(32g)^{1/2}}$
0.00010	5.079181	0.00500	3.380211
0.00015	4.903090	0.00600	3.301030
0.00020	4.778151	0.00700	3.234083
0.00025	4.681241	0.00800	3.176091
0.00030	4.602060	0.00900	3.124939
0.00040	4.477121	0.01000	3.079181
0.00050	4.380211	0.01500	2.903090
0.00060	4.301030	0.02000	2.778151
0.00070	4.234083	0.02500	2.681241
0.00080	4.176091	0.03000	2.602060
0.00090	4.124939	0.04000	2.477121
0.00100	4.079181	0.05000	2.380211
0.00150	3.903090	0.06000	2.301030
0.00200	3.778151	0.07000	2.234083
0.00250	3.681241	0.08000	2.176091
0.00300	3.602060	0.09000	2.124939
0.00400	3.477121	0.10000	2.079181

FIGURE 2.2 Relationship between n parameter and relative roughness to establish relation of n to e/R_h.

defined respectively by A = (b + mY)Y, and $P = b + 2Y\sqrt{m^2+1}$, and Manning's equation written as a function of this unknown equal to zero for use in the Newton iterative Equation B.2 becomes

$$F(\xi) = nQ\left[b + 2Y\sqrt{m^2+1}\right]^{2/3} - [(b+mY)Y]^{5/3}C_u\sqrt{S_o} = 0 \tag{2.8b}$$

in which variable ξ represent b, Y, or m depending respectively whether the bottom width, the depth or the side slope is the unknown. The Newton method for solving implicit equations such as Equation 2.8b is discussed in Appendix B, and is illustrated by example problems below.

If the channel is circular, then it is best to introduce the additional angle $\beta = \cos^{-1}(1 - 2Y/D)$ as defined in Appendix A. Thus the depth Y is related to this angle (in radians) by $Y = D(1 - \cos\beta)/2$, and Manning's equation can be written as

$$F(\xi) = nQ(\beta D)^{2/3} - \left[\frac{D^2(\beta - \cos\beta\sin\beta)}{4}\right]^{5/3} C_u\sqrt{S_o} = 0 \tag{2.8c}$$

in which variable ξ now represents either β or D depending respectively whether the depth, or the diameter is unknown. Should the depth be unknown then β is first obtained by solving for it from Equation 2.8c, and thereafter the depth Y is computed by the equation above Equation 2.8c.

EXAMPLE PROBLEM 2.5

A flow rate of Q = 450 cfs is taking place in a trapezoidal channel with the following properties: b = 10 ft, m = 1, and $S_o = 0.0006$. Determine the uniform depth of flow in this channel if the appropriate value for Manning's n is, n = 0.013.

Solution

This problem can be solved using Equation 2.8b and the Newton method. The Newton method can easily be solved using a programmable pocket calculator using the following steps:

1. Assign the variables, Y, b, m, Q, S_o, and n to storage registers, and place the values for these variable in the assigned registers (this includes a guess for the unknown Y in this case).
2. Put the calculator in program mode and program Equation 2.8b into it with the value of the equation F(ξ) = F(Y) being displayed upon completion of the equation.
3. Press the operate button on the calculator, and when it is complete store the value displayed in an unused register.
4. Retrieve the value from the register that hold Y, and increase it by a small amount such as 0.001.
5. Store this increased value in the same register for Y again.
6. Press the operate button again.
7. Recall the last value of the equation, subtract it from the current value and divide this difference by 0.001.
8. Recall Y, subtract the result from step 7, and also subtract 0.001 from it and store Y back in its register.
9. Repeat steps 4 through 8 until convergence has occurred. If you want you can also program these steps into your pocket calculator.

The implementation of these steps for this problem results in the following: Step 1

Register	1(Y)	2(b)	3(m)	4(Q)	5(S_o)	6(n)
Value	5	10	1	450	0.0005	0.013

from step 3 F(Y) = 2.902; from step 6 F(Y) = 2.738, after step 8 register # 1 contains 5.018 for Y. After iteration # 2 Y = 5.018, which represents no change to three digits beyond the decimal point, and the solution is terminated. If you have a calculator with the capability to solve implicit equations with a SOLVE key such as the HP 48xs, then all that is needed is to define the equation and give values to the variables.

EXAMPLE PROBLEM 2.6

A flow rate of Q = 30 cfs is to be carried by a pipe with a bottom slope of $S_o = 0.00028$, and a Manning's roughness coefficient of 0.013. If the depth is not to exceed 3/4 of the diameter, what size pipe should be used?

Solution

Since the depth is not to exceed 3/4 of the diameter, then $\cos\beta = 1 - 3/2 = -1/2$, and the equation that must be solved is

$$F(D) = nQ(\beta D)^{2/3} - \left\{\frac{D^2}{4}(\beta - \sin\beta\cos\beta)\right\}^{5/3} C_u\sqrt{S_o}$$

for the pipe diameter D. Using the Newton method, the solution to this equation is D = 4.49 ft. Taking the next standard pipe size would call for using a pipe with a 54 in. diameter.

EXAMPLE PROBLEM 2.7

A pipe of diameter 2 m and Manning's n = 0.013 has a bottom slope of $S_o = 0.00112$. Generate a table that gives the depths of flow that would be expected in this pipe under uniform flow conditions for flow rates of $Q = 0.5\,m^3/s$ to $Q = 4.5\,m^3/s$ in increments of $0.5\,m^3/s$.

Solution

The solution for each entry in this table requires that Equation 2.8c be solved by an iterative method such as the Newton method. The following BASIC program implements such a solution for the nine different flow rates giving the depth shown to the right of the BASIC listing.

BASIC program listing to solve Example Problem 2.7

```
10 INPUT "Give:n,So,D,Q1,DQ,est. for Y & No ",N,S,D,Q,DQ,Y,NO%
20 FOR I%=1 TO NO%
30 NCT%=0
40 ARG=1-2*Y/D
50 TANA=SQR(1-ARG*ARG)/ABS(ARG)
60 IF ARG>0 THEN BETA=ATN(TANA) ELSE BETA=3.14159265#-ATN
   (TANA)
70 A=.25*D*D*(BETA-ARG*SIN(BETA))
80 P=BETA*D
90 F=Q-A/N*(A/P)^.6666667*SQR(S)
100 IF NCT%>0 THEN GOTO 150
110 F1=F
120 Y=Y-.001
130 NCT%=1
140 GOTO 40
150 DY=.001*F1/(F1-F)
160 Y=Y-DY+.001
170 IF ABS(DY) > .00001 THEN GOTO 30
180 PRINT Q,Y
190 Q=Q+DQ
200 NEXT I
210 END
```

Q(m3/s)	Y(m)
0.5	0.42
1.0	0.60
1.5	0.74
2.0	0.87
2.5	0.99
3.0	1.10
3.5	1.22
4.0	1.33
4.5	1.46

EXAMPLE PROBLEM 2.8

A natural canal that has a bottom slope of 0.002, and a Manning's n = 0.018 has the cross section defined by the following transect data for a long distance. Determine the depth of flow in this canal if the flow rate is Q = 90 cfs.

x (ft)	0	2	4	7	9	11	14	18
y (ft)	0	0.8	1.5	3.0	3.3	2.5	1.2	0.0

Solution

From this data it is possible to interpolate along both sides of the canal to generate the following data giving the top width, the area, and perimeter for equal increments of the depth y.

Depth, Y	Top Width	Area	Perimeter
0.17	1.58	0.13	1.62
0.33	3.01	0.51	3.10
0.50	4.30	1.11	4.44
0.66	5.44	1.92	5.63
0.83	6.44	2.90	6.68
0.99	7.29	4.03	7.59
1.16	7.99	5.29	8.37
1.32	8.55	6.65	9.02
1.49	8.96	8.10	9.61
1.65	9.22	9.62	10.22
1.82	9.41	11.18	10.88
1.98	10.21	12.80	11.75
2.15	11.07	14.56	12.66
2.31	12.03	16.46	13.68
2.48	13.05	18.53	14.75
2.64	14.03	20.77	15.78
2.81	15.01	23.16	16.82
2.97	15.99	25.72	17.86
3.14	16.99	28.44	18.91
3.30	18.00	31.33	19.98

The solution might proceed by selecting a depth, and then by interpolation in the above table determine the corresponding area and perimeter, substitute these into Manning's equation, and compute the flow rate. Based on the difference between the computed flow rate and the wanted value of 90 cfs, adjust the depth and repeat the process. A more formal procedure would be to implement the Newton method in the above table interpolation to obtain a better estimate for the next depth to use. The answer is Y = 2.60 ft, with A = 20.21 ft^2 and P = 15.56 ft. See Appendix B.5 for details related to how areas and wetted perimeters can be defined from x y cross-sectional coordinates, etc.

Solving for the depth, or diameter, in circular sections by the Newton method requires that a good guess be supplied to start the iterative solution process, or else the method will fail. For circular sections, this guess must be much better than for trapezoidal channels. In the small BASIC program given under Problem 2.7 above it is left to the user to provide a satisfactory starting value. However, requiring the user to supply an initial guess is not always desirable. In most cases, an adequate initial guess can be easily generated within a computer program for solving for either the depth or diameter in a circular section, the two variables in Manning's equation for which an explicit solution is not possible.

If the area and wetted perimeter for a circular section are replaced in Manning's equation by functions of the auxiliary angle β that define these quantities, and terms rearranged, then Manning's equation can be written as

$$Q' = \frac{nQ}{C_u D^{8/3}\sqrt{S_o}} = \frac{\left(\beta - \cos\beta\sin\beta\right)^{5/3}}{10.079368\beta^{2/3}} = F(\beta) \tag{2.9}$$

It should be noted that Q' is dimensionless if C_u has units of the cube root of length per time (with n taken as dimensionless). A close approximation of the above dependency of Q' on β is given by the following power equation:

$$Q' = 0.0313\beta^{4.0984} \tag{2.10a}$$

or the inverse of this equation is,

$$\beta = 2.3286(Q')^{0.244} \tag{2.10b}$$

Thus if the depth of flow is the unknown it is possible to compute $Q' = nQ/(C_u D^{8/3}\sqrt{S_o})$, and then from Equation 2.10b obtain a starting value for β. The reasonableness of this approximation is shown by the values in the small table below that gives the approximate β, the actual β and the difference corresponding to several values of Q'. Note that as the depth approaches the diameter, i.e., β approaches π the difference get larger.

Q'	β_{act}	β_{appr}	Difference
1.00E–06	0.0822	0.0800	0.0022
1.00E–05	0.1400	0.1403	–0.0003
1.00E–04	0.2388	0.2461	–0.0072
1.00E–03	0.4098	0.4316	–0.0218
1.00E–02	0.7160	0.7570	–0.0410
0.100	1.3478	1.3277	0.0201
0.200	1.7364	1.5723	0.1641
0.300	2.1842	1.7359	0.4484
0.330	2.4564	1.7767	0.6797
0.335	2.5955	1.7832	0.8123

Manning's equation will need to be solved frequently throughout the chapters of this book. It will be useful for you to develop a computer program, or program your pocket calculator so that it will be possible to readily obtain a solution to any variable that may be unknown in Manning's equation for either a trapezoidal or a circular section. Below is a listing of such a program in TURBO PASCAL, which utilizes the clear screen CLRSCR and the GOTOXY capabilities to give the program a little "user friendliness." You should study over the program carefully. The array element X[IU] contains the unknown. For a trapezoidal channel the correspondence between the X's and the variables are: X[1] = Q, X[2] = n, X[3] = S, X[4] = Y, X[5] = b, and X[6] = m. For a circular section the variables are the same through X[4], but then X[5] = D (the pipe diameter). A good way for you to understand how this program solves Manning's equation completely in either a circular or a trapezoidal section, and allows either ES or SI units to be used is for you to translate the program into FORTRAN, C or BASIC depending upon what you are most familiar with.

Listing of PASCAL program MANNING.PAS that completely solves Manning's equation in both trapezoidal and circular channels

```
Program Manning;
Const NX:array[1..7] of char=('Q','n','S','Y','b','m','D');
  VNAM:array[1..4] of string[12]=('flow rate','coefficient',
    'bottom slope','depth');
```

```
  JW:array[1..4] of integer=(2,4,6,3);
Var X:array[1..7] of real; F1,P,Beta,DF,DX,C,AA:real;
  ITY,I,IU,m:integer;
Function Expn(a,b:real):real;
  Begin if a<0 then Writeln('error in power',a,b)
  else Expn:=Exp(b*Ln(a)) End; {raises a to the power b}
Function A:real; Begin {computes area A & AA and perimeter P}
  If ITY=1 then begin P:=X[5]+2*X[4]*sqrt(sqr(X[6])+1);
    AA:=(X[5]+X[6]*X[4])*X[4];
  end else Begin P:=1-2*X[4]/X[5]; if P=0 then Beta:=Pi/2 else begin
  Beta:=sqrt(1-sqr(P))/abs(P); if P>0 then Beta:=arcTan(Beta) else
  Beta:=Pi-arcTan(Beta) end;AA:=sqr(X[5])/4*(Beta-P*sin(Beta));
    P:=Beta*X[5] End;
  A:=AA End;
Function F:real; Begin {Defines Manning's Equation for Newton Method}
  F:=X[2]*X[1]-C*A*Expn(AA/P,0.666667)*SQRT(X[3]) End;
Var Ch:Char;
Label L1;
BEGIN      {Start of program}
L1:ClrScr; GoToXY(1,10);Writeln('Do you want to use:');
  Writeln('1 - ES units, or'); Writeln('2 - SI units?');
  repeat Readln(Ch); until Ch in ['1','2','E','e','S','s'];
  If Ch in ['1','E','e'] then C:=1.486 else C:=1;
  ClrScr;GoToXY(1,10); Writeln('Is section:');
  Writeln('1 - Trapezoidal, or');Writeln('2 - Circular?');
  repeat Readln(Ch); until Ch in ['1','2','T','t','C','c'];
  if Ch in ['1','t','T'] then ITY:=1 else ITY:=2; ClrScr;
  Writeln('Give no. of unknown');
   For I:=1 to 4 do Writeln(I:2,' - ',NX[I]:1,' (',VNAM[I]);
   if ITY=2 then Writeln(' 5 - D (diameter)') else begin
     Writeln(' 5 - b (bottom width)');
     Writeln(' 6 - m (side slope)') end;
  repeat Readln(IU); until IU in [1..7]; ClrScr;
  Writeln('Give values for knowns');For I:=1 to 4 do if I<>IU then
    begin GoToXY(1,I+1);Write(NX[I]:1,' = '); Readln(X[I]) end;
  If (ITY=2) and (IU<>5) then begin GoToXY(1,6);Write('D = ');
    Readln(X[5]) end;
  If ITY=1 then Begin I:=5; if IU<>5 then begin I:=I+1;
    GoToXY(1,I); Write('b = ');Readln(X[5]) end;
    if IU<>6 then begin I:=I+1; GoToXY(1,I);
     Write('m = ');Readln(X[6]) end; End;
  Case IU of
    {1 thru 3 solve explicit eqs, 4,5 & 6 use Newton Method}
   1:X[1]:=C/X[2]*A*Expn(AA/P,0.6666667)*sqrt(X[3]);
   2:X[2]:=C/X[1]*A*Expn(AA/P,0.6666667)*sqrt(X[3]);
   3:X[3]:=sqr(X[1]*X[2]/(C*A*Expn(AA/P,0.6666667)));
   4,5,6:Begin If ITY=2 Then Begin If IU=4 then begin
     Beta:=2.3286*Expn(X[1]*X[2]/(C*Expn(X[5],
       2.6666667)*sqrt(X[3])),0.244);
      X[4]:=X[5]/2*(1-cos(Beta)) end else X[5]:=2*X[4] End Else
```

```
     case IU of 4:X[4]:=X[5]/2; 5:X[5]:=2*X[4]; 6:X[6]:=1;
       end; m:=0;
     repeat F1:=F; DX:=X[IU]/100; X[IU]:=X[IU]-DX;
      DF:=DX*F1/(F1-F); X[IU]:=X[IU]+DX-DF; m:=m+1;
        until (m>20) or (abs(DF)<0.0001);
     End; End; ClrScr;
  Writeln('Solution to unknown');
    If (ITY=2) and (IU=5) then Write('D = ') else
    Write(NX[IU]:1,' = '); Writeln(X[IU]:11:JW[IU]+1); Writeln;
  Writeln('Variables of Problem:');
   For I:=1 to 4 do Writeln(NX[I]:1,' = ',X[I]:10:JW[I]);
   If ITY=2 then Writeln('D = ',X[5]:10:3) else begin
    Writeln('b = ',X[5]:10:3); Writeln('m = ',X[6]:10:4) end;
  GoToXY(1,24);
    Write('Do You want to solve another problem?(Y or N) ');
   Readln(Ch); If (Ch='Y') or (Ch='y') then GoTo L1;
END.
```

Listing of FORTRAN program MANNING.FOR designed to solve Manning's equation in either trapzoidal or circular channels

```
      LOGICAL REPT
      REAL X(6),n,m
      CHARACTER*1 V(6)
      COMMON X,BETA
      EQUIVALENCE (Q,X(1)),(n,X(2)),(S,X(3)),(Y,X(4)),
     &(b,X(5)),(m,X(6))
      DATA V/'Q','n','S','Y','b','m'/
      WRITE(6,*)' Give 1.49 for ES units or 1. for SI'
     &,= units.'
      READ(5,*) C
      WRITE(6,*)' Give 1 for trap. sec., or 2 for cir.'
     &,= sec.'
      READ(5,*) ITYP
      IF(ITYP.EQ.1) THEN
      No=6
      ELSE
      No=5
      V(5)='D'
      ENDIF
1     WRITE(6,100) (I,V(I),I=1,No)
100   FORMAT(' Give No. of unknown:'/6(I2,'-',A1))
      READ(5,*) IUNK
      DO 10 I=1,No
      IF(I.LT.4.AND.I.EQ.IUNK) GO TO 10
      WRITE(6,101) V(I)
101   FORMAT(3X,A1,' = ',$)
      READ(5,*) X(I)
10    CONTINUE
      NCT=0
20    AA=A(ITYP)
```

```
      GO TO(21,22,23,24,24,24), IUNK
21    Q=C/n*AA*(AA/P(ITYP))**.6666667*SQRT(S)
      GO TO 40
22    N=C/Q*AA*(AA/P(ITYP))**.6666667*SQRT(S)
      GO TO 40
23    S=(n*Q/C*(P(ITYP)/AA)**.6666667/AA)**2
      GO TO 40
24    REPT=.TRUE.
25    F=n*Q-C*AA*(AA/P(ITYP))**.6666667*SQRT(S)
      IF(REPT) THEN
      REPT=.FALSE.
      F1=F
      X(IUNK)=X(IUNK)-.001
      AA=A(ITYP)
      GO TO 25
      ENDIF
      DIF=.001*F1/(F1-F)
      X(IUNK)=X(IUNK)+.001-DIF
      NCT=NCT+1
      IF(NCT.LT.20 .AND. ABS(DIF).GT..00001)
      * GO TO 20
      IF(NCT.EQ.20)WRITE(6,*)' FAILED TO CONVERGE' *,DIF
40    WRITE(6,105)(V(I),X(I),I=1,No)
105   FORMAT(' SOLUTION:',6(A2,' =',F12.6))
      WRITE(6,*)' Give 1 for another prob.; else 0'
      READ(5,*) NCT
      IF (NCT.EQ.1) GO TO 1
      STOP
      END
      FUNCTION A(ITYP)
      COMMON X(6),BETA
      IF(ITYP.EQ.1) THEN
      A=(X(5)+X(6)*X(4))*X(4)
      ELSE
      BETA=ACOS(1.-2.*X(4)/X(5))
      A=.25*X(5)*X(5)*(BETA-.5*SIN(2.*BETA))
      ENDIF
      RETURN
      END
      FUNCTION P(ITYP)
      COMMON X(6),BETA
      IF(ITYP.EQ.1) THEN
      P=X(5)+2.*X(4)*SQRT(X(6)**2+1.)
      ELSE
      P=X(5)*BETA
      ENDIF
      RETURN
      END
```

Listing of C program designed to solve Manning's equation in trapezoidal or circular channels

```
/*  Solves Manning's Equation in Trapezoidal & Circular Channels  */
#include <stdio.h>
#include <math.h>
int type; float beta;
float A(float b,float m,float Y) {
  if (type==1) return((b+m*Y)*Y); else
 {beta=acos(1.-2.*Y/b); return(0.25*b*b*(beta-0.5*sin(2.*beta)));}}
float P(float b,float m,float Y) {
 if(type==1) return(b+2.*Y*sqrt(m*m+1.));
 else return(b*sin(beta));}
main () {
float x[6],cc,AA,F,F1,DIF,dumm; int IUNK,nct,No,II,i; char u,V[7];
  dumm=sqrt(.04); /* drag floating pt lib. into the linker */
  strcpy(V,"QnSYbm\n");
 puts("Give 1.49 for ES units or 1. for SI units.");
   scanf("%f",&cc);
 puts("Give 1 for trapezoidal section, or 2 for circular section.");
   scanf("%d",&type); if (type==1) No=6; else {V[4]='D'; No=5;}
 a2:puts("Give No. of unknown:");
   for (i=0;printf(" %d - %c,",i,V[i]),i<No-1;i++);
 printf("\n"); scanf("%d",&IUNK);
 puts("Give:\n"); for (i=0;i<No;i++) if ((i!=IUNK) | (i>2)) {
  printf(" %c = ",V[i]); scanf("%f",&x[i]);}
 switch (IUNK) {
  case 0: x[0]=cc/x[1]*pow(A(x[4],x[5],x[3]),1.6666667)/pow(P(x[4],\
    x[5],x[3]),0.66667)*sqrt(x[2]); break;
  case 1: x[1]=cc/x[0]*pow(A(x[4],x[5],x[3]),1.6666667)/pow(P(x[4],\
    x[5],x[3]),0.66667)*sqrt(x[2]); break;
  case 2: AA=A(x[4],x[5],x[3]); DIF=x[1]*x[0]/cc*pow(P(x[4],\
    x[5],x[3])/AA,0.666667)/AA; x[2]=DIF*DIF; break;
  default: nct=0; do {II=0; a1:AA=A(x[4],x[5],x[3]);
    F=x[0]*x[1]-cc*AA*pow(AA/P(x[4],x[5],x[3]),0.66667)*sqrt(x[2]);
    if (II==0) {II=1; F1=F; x[IUNK]=x[IUNK]-0.001; goto a1;}
    DIF=0.001*F1/(F1-F); x[IUNK]=x[IUNK]+0.001-DIF;
    } while (abs(DIF)>0.00001 && ++nct<15);}
  for (i=0;i<No;i++) printf(" %c =%f\n",V[i],x[i]);
  puts(" Give 1 to solve another problem; else 0");
  scanf("%d",&II);
  if (II==1) goto a2;
}
```

The Newton method can be use to solve a linear equation. When the Newton method is used in solving a linear equation the solution will be obtained with the first iteration, even though this will generally not be known until the computations are done for the second iteration, and the function of the unknown equals zero. The logic of the above computer programs can, therefore, be simplified by omitting the separate equations that solve for those variable that can be explicitly obtained from Manning's equation. Below are listing of a FORTRAN program and a C program that only use the Newton method; even when solving for Q, n, and S.

Listing of FORTRAN program that uses the Newton method to solve all variables (MANNTC. FOR)

```
      PARAMETER (EX=.6666667,EX1=1.6666667)
      CHARACTER*1 VAR(6)/'Q','n','S','y','b','m'/,UNK
      CHARACTER*33 FMT/"(' Solution for ',A1,' = ',F10.3)"/
      REAL X(6),ns,neq,C/1.486/
C  X(1)=Q, X(2)=n, X(3)=S, X(4)=y, X(5)=b, X(6)=m
      LOGICAL R /.FALSE./
      WRITE(*,*)'Give 0 if ES; 1 if SI units'
      READ(*,*) IUNIT
      IF(IUNIT.EQ.1) C=1.
      WRITE(*,*)'Is the section: 1   trapzoidal, or 2   circular?'
      READ(*,*) ISECT
      IF(ISECT.EQ.2) THEN
      NVAR=5
      VAR(5)='D'
      ELSE
      NVAR=6
      ENDIF
      WRITE(*,*)'Give value to variables'
      DO 10 I=1,NVAR
      WRITE(*,"(2X,A1,' = '\)") VAR(I)
      IF(I.EQ.2 .AND. ISECT.EQ.1) THEN
      WRITE(*,"(' (1st for sides=) ')")
      READ(*,*) ns
      WRITE(*,"(' (now for bottom=) ')")
      ENDIF
10    READ(*,*) X(I)
15    WRITE(*,*)'Give symbol for unknown'
      READ(*,'(A1)') UNK
      IUNK=1
      DO 20 WHILE (VAR(IUNK).NE.UNK.AND.IUNK.LE.NVAR)
20    IUNK=IUNK+1
      IF(IUNK.GT.NVAR) GO TO 15
      DO 30 I=1,20
25    IF(ISECT.EQ.2) THEN
      COSB=1. 2.*X(4)/X(5)
      B=ACOS(COSB)
      A=.25*X(5)*X(5)*(B COSB*SIN(B))
      P=B*X(5)
      F=X(2)*X(1) C*SQRT(X(3))*A*(A/P)**EX
      ELSE
      P=X(5)+2.*X(4)*SQRT(X(6)**2+1.)
      neq=X(5)/P*X(2)+2.*X(4)*SQRT(X(6)**2+1.)/P*ns
      F=neq*X(1)*P**EX C*SQRT(X(3))*((X(5)+X(6)*X(4))*X(4))**EX1
      ENDIF
      IF(R) GO TO 28
      XX=X(IUNK)
      X(IUNK)=1.005*X(IUNK)
      F1=F
```

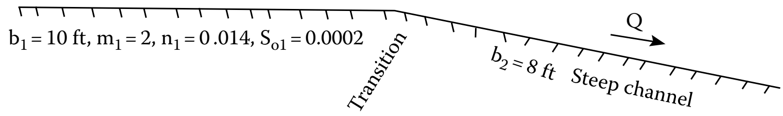
```
      R=.TRUE.
      GO TO 25
28    DIF=F1*(X(IUNK) XX)/(F F1)
      X(IUNK)=XX DIF
      IF(ABS(DIF).LT. .000001) GO TO 40
30    CONTINUE
40    IF(IUNK.EQ.2)FMT(32:32)='4'
      IF(IUNK.EQ.3)FMT(32:32)='7'
      WRITE(*,FMT) VAR(IUNK),X(IUNK)
      END
```

Listing of C program that uses the Newton method to solve all variables (MANNTC.C)

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
main() {int units,sect,i,r,iunk,nvar=5,nct=0;
 float c,x[6],f,f1,xx,dif,cosb,a,b,p,ns,neq,ex=.6666667,\
   ex1=1.6666667;
 char var[7]="QnSybm\0",unk; clrscr();
 printf("Give 0 if ES; 1 if SI units ");scanf("%d",&units);
 if (units) c=1; else c=1.486;
 printf("Is the section: 1 - trapezoidal, or 2 - circular? ");
 scanf("%d",&sect);if(sect==2) {nvar=4;var[4]='D';}
 printf("Give value to variables\n");for(i=0;i<=nvar;i++){
  printf(" %c = ",var[i]); if((i==1)&&(sect==1)){
   printf(" (1st for sides =) ");scanf("%f",&ns);
   printf(" (now for bottom=) ");} scanf("%f",&x[i]);}
 do {printf("Give symbol for unknown ");scanf("%s",&unk);iunk=0;
 while (var[iunk]!=unk && iunk<=nvar) iunk++;} while(iunk>unk);
 do {r=1;
L1:if(nvar==4){cosb=1-2*x[3]/x[4];b=acos(cosb);
  a=.25*x[4]*x[4]*(b-cosb*sin(b)); p=b*x[4];
  f=x[1]*x[0]-c*sqrt(x[2])*a*pow(a/p,ex);} else {
  p=x[4]+2.*x[3]*sqrt(x[5]*x[5]+1.);
    neq=x[4]/p*x[1]+2.*x[3]*sqrt(x[5]*x[5]+1.)/p*ns;
  f=neq*x[0]*pow(p,ex)-c*sqrt(x[2])*pow((x[4]+x[5]*x[3])*x[3],\
    ex1);}
 if(r) {xx=x[iunk];x[iunk]=1.005*x[iunk];f1=f;r=0;goto L1;}
 dif=f1*(x[iunk]-xx)/(f-f1); x[iunk]=xx-dif; nct++;
 } while (nct<20 && fabs(dif)>.000001);
 printf("\nSolution for %c = %f",var[iunk],x[iunk]);}
```

EXAMPLE PROBLEM 2.9

Determine the maximum flow rate that can be accommodated in the channel shown below without causing the depth in the upstream channel to rise above its normal depth. The upstream channel has a bottom width $b_1 = 10\,ft$, a side slope $m_1 = 2$, a Manning's roughness coefficient, $n_1 = 0.014$, and a bottom slope, $S_{o1} = 0.0002$. The downstream channel is steep, i.e., under uniform flow conditions the depth will be less than critical depth and is rectangular in shape with a bottom width $b_2 = 8\,ft$. Investigate the relationship between the slope of bottom of the upstream channel on the flow rates and the depths that are possible in this channel under uniform flow conditions.



Solution

Since the downstream channel is steep, critical flow will occur at the end of the transition to 8 ft wide rectangular channel, the specific energy here is given by

$$E_c = 1.5Y_c = 1.5\left(\frac{q_2^2}{g}\right)^{1/3} = 1.5\left[\frac{(Q/b_2)}{g}\right]^{1/3} = 0.11787Q^{2/3}$$

The specific energy in the larger upstream channel must equal E_c or

$$Y_1 + \frac{(Q/A)_1^2}{(2g)} = E_c = 0.11787Q^{2/83}$$

If the depth upstream is to be uniform, then

$$Q = \frac{1.486}{n}A_1\left(\frac{A}{P_1}\right)_1^{2/3} S_{o1}^{1/2}$$

These three equations allow for the variables E_c, Y_1, and Q to be solved, or if one wishes to eliminate the first equation then the latter two equations will solve for Y_1 and Q. Their solution is

$$Y = 4.146\text{ ft},\quad Q = 218.42\text{ cfs.}$$

In general, a problem of this nature that is to determine what the maximum flow rate is that can occur in a channel with a transition from an upstream mild channel to a downstream steep channel requires the simultaneous solution of (1) the critical flow equation in the downstream channel; (2) the energy equation, which equates the specific energy in the upstream channel to the critical specific energy in the downstream channel; and (3) the uniform flow equation.

To investigate the relationship between the bottom slope of the upstream channel and the maximum uniform flow possible requires that the latter two above equations be solved for different values of S_{o1}. The table below shows the results from several such solutions:

Bottom slope S_{o1}	0.002	0.001	0.0009	0.0008	0.0007	0.0006	0.0005	0.0004
Flow rate Q (cfs)	0.457	6.703	9.466	13.59	19.88	29.66	45.37	71.72
Depth Y_1 (ft)	0.0616	0.376	0.476	0.610	0.792	1.094	1.406	1.922
	0.0003	0.0002	0.00018	0.00014	0.00012	0.0001	0.00008	
	119.4	218.4	251.4	344.1	412.1	504.7	638.6	
	2.734	4.146	4.566	5.662	6.404	7.352	8.627	

It is interesting to note that the flow rate decreases to extremely small values as the bottom slope of the upstream channel becomes larger. Thus it is very easy to "choke" the upstream flow, i.e., cause it to be above normal depth by reducing the size of a downstream channel that will have critical flow in it.

If you don't have a computer available, nor a programmable calculator, then it is possible to use graphical means for solving Manning's equation. To develop such a graphical solution let us define a dimensionless depth Y′ = Y/b for trapezoidal channels. (When defining a dimensionless depth for the specific energy later we will let Y′ = mY/b because this eliminates m from the resulting dimensionless equation.) Then after some algebraic manipulation the following equation can be obtained:

$$\frac{nQ}{C_u b^{\frac{8}{3}} \sqrt{S_0}} = \frac{(Y' + mY'^2)^{\frac{5}{3}}}{\left(1 + 2Y'\sqrt{1+m^2}\right)^{\frac{2}{3}}} = Q'$$

The parameter on the left of the equal sign might be taken as a dimensionless flow rate, denote as Q′. (Note that Q′ is dimensionless if n is taken as dimensionless and C_u is assumed to have dimensions of $L^{1/3}/t$.) This equation shows that the flow rate parameter Q′ is a function of the dimensionless depth Y′ = Y/b, and the side slope m of the trapezoidal channel, and when m = 0 the channel is rectangular. This relationship is given in Figure 2.3 using several different curves for different m values. The main graph is a linear plot, and the insert gives the same graph except using log–log paper. This graph can be used to solve Manning's equation for several different unknowns (Figure 2.3). For example, to find the normal depth Y_o, one would first compute the flow rate parameter Q′ from the known values; then enter this value on the ordinate of the graph, and read the corresponding dimensionless depth Y′ on the abscissa; and finally compute Y_o as the product of Y′ and the bottom width b.

Manning's equation can also be solved graphically for a circular section. To develop such a graphical solution Equation 2.9 can be used to define the relationship between the angle β and

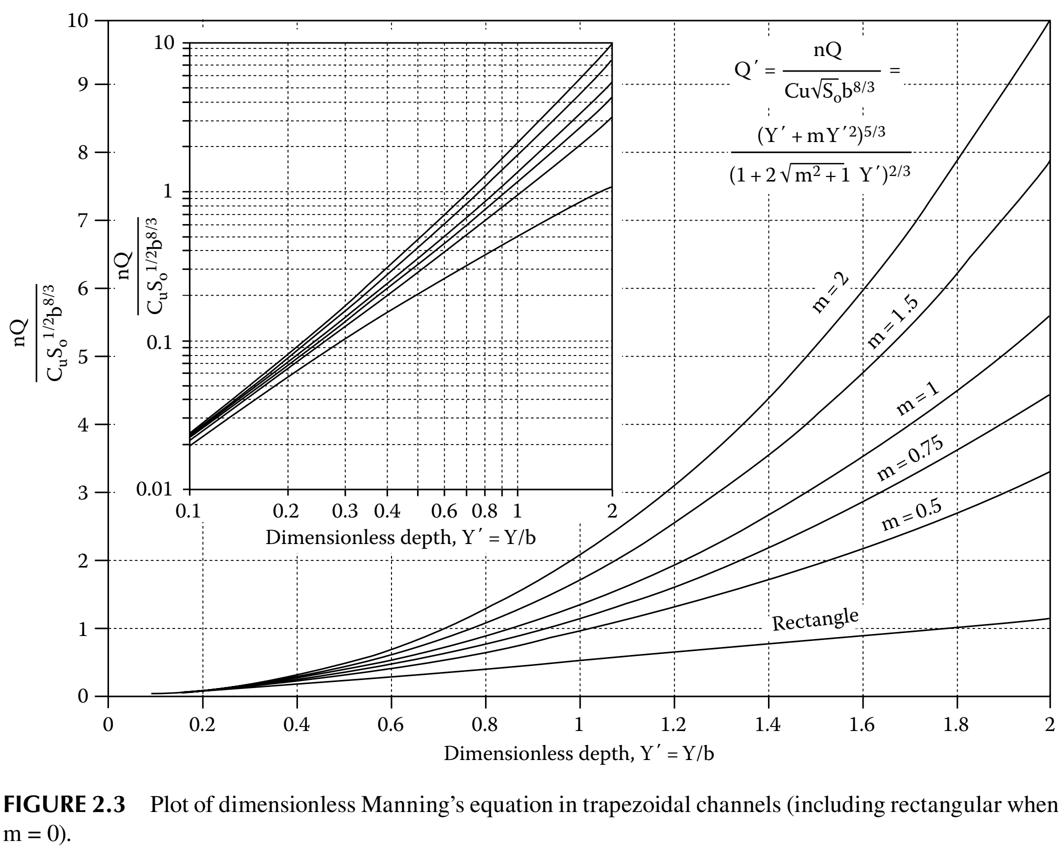


FIGURE 2.3 Plot of dimensionless Manning's equation in trapezoidal channels (including rectangular when m = 0).

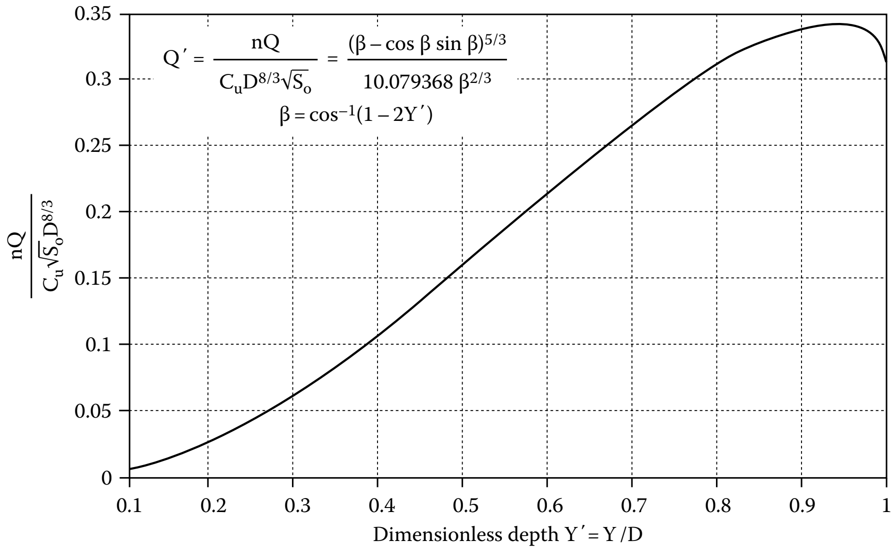


FIGURE 2.4 Plot of dimensionless Manning's equation in circular channels.

the flow rate parameter $Q' = (\beta - \cos\beta \sin\beta)^{5/3}/(10.0793684\beta^{2/3})$, and the angle β in turn can be obtained from the dimensionless depth $Y' = Y/D$ from $\beta = \cos^{-1}(1 - 2Y')$. Such a graphical solution of Manning's equations for circular channels is given in Figure 2.4. Table D.2 provides greater precision than can be obtained from the graph, and the example problems at the end of Table D.2 illustrate how the table can be used.

2.5 CHANNELS WITH VARYING WALL ROUGHNESS, BUT Q = CONSTANT

Uniform flow cannot exist in a channel in which the wall roughness varies in the direction of the channel unless the bottom slope varies in precisely the correct manner so the velocity and depth remain constant with x. This combination of n and S_o would create uniform flow only for one flow rate. However, uniform flow can occur in long channels with constant bottom slopes when the wall roughness varies along the position of the channel cross section. It is not uncommon to have a channel's sides with a different roughness than its bottom. An example is a laboratory flume whose sides are Plexiglas and its bottom is filled with gravel.

A modification to Manning's equation for channels with varying roughness coefficients in different portions of its cross section might be to compute an equivalent Manning's n that weights the individual n values according to the portion of the perimeter to which they apply. For a trapezoidal channel with a different Manning's n along the bottom, n_b, than that for the sides, n_s, the equivalent roughness coefficient would then be

$$n_{eq} = \frac{b}{P} n_b + \frac{2Y\sqrt{1+m^2}}{P} n_s$$

The last two program listings are designed to handle a trapezoidal channel that has a different roughness coefficient along the bottom than the sides of the channel using an equivalent roughness coefficient computed by this equation. The validity of using an equivalent roughness coefficient would need to be verified by field or laboratory measurements for a given channel. The need for verification is that Manning's equation is empirical and therefore it is not possible to use theory alone to derive an "equivalent" Manning's equation for channels with varying roughnesses along the cross section. The above formula will produce an n_{eq} that equals n_b and n_s when these are the same, whereas other methods will not. For example, one might be inclined to associate n with $P^{2/3}$ for which then n applies. Then $nP^{2/3}$ would be replaced by $\sum n_i P_i^{2/3}$. Using this approach for a trapezoidal channel with a different bottom roughness than side roughness would use one of the following Manning's equation:

$$Q = \frac{C_u A^{5/3}\sqrt{S_o}}{\left\{n_b b^{2/3} + \left[2Y\sqrt{m^2+1}\right]^{2/3} n_s\right\}}$$

or

$$Q = C_u A\sqrt{S_o}\,\frac{\left\{(Y/n_b)^{2/3} + \left[mY\Big/\left\{2n_g\sqrt{m^2+1}\right\}\right]^{2/3}\right\}}{\{n_b + n_s\}^{1/3}}$$

or

$$Q = C_u A\sqrt{S_o}\,\frac{\left\{Y/n_b + mY\Big/\left[2n_s\sqrt{m^2+1}\right]\right\}^{2/3}}{\{n_b + n_s\}^{1/3}}$$

The problem with any of these latter formulas is that they will not produce the same results as the original Manning's equation does when $n = n_b = n_s$.

2.6 SPECIFIC ENERGY, SUBCRITICAL AND SUPERCRITICAL FLOWS

When dealing with open channel flow it is convenient to reference the energy per unit weight from the channel bottom. Thus instead of having a horizontal datum from which the energy is referenced, a sloping data is used. The result is call the "specific energy," and it consists of the depth of flow in the channel plus the velocity head, or

$$E = Y + \alpha\frac{V^2}{2g} = Y + \alpha\frac{Q^2}{2gA^2} \tag{2.11}$$

in which α is the kinetic energy correction coefficient defined in Chapter 1. In Equation 2.11 Y is the depth of flow, and if the pressure distribution is hydrostatic, then this depth Y will equal the pressure head p/γ on the bottom of the channel. Therefore, regardless of the position within the channel flow the sum of the two terms in Equation 2.11 represent the distance between the channel bottom and the energy line. For a uniform flow the specific energy will be constant. If the channel is on a steep slope then it is necessary to adjust the depth as described in Chapter 1, that is, if Y represents the vertical distance through the fluid, then it needs to be multiplied by the cosine squared of the angle of the bottom slope. It is common in practice to ignore the kinetic energy correction coefficient α (i.e., assume $\alpha = 1$) and then the specific energy becomes

$$E = Y + \frac{V^2}{2g} = Y + \frac{Q^2}{2gA^2} \tag{2.11a}$$

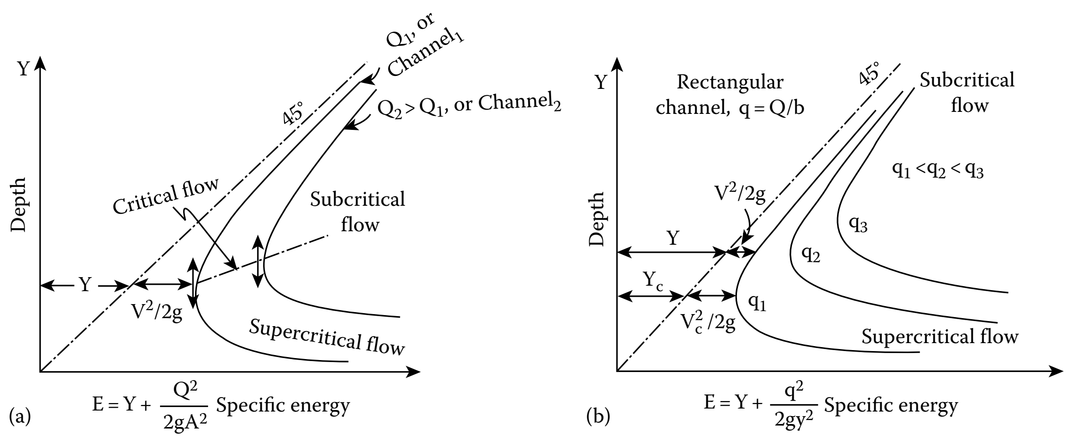


FIGURE 2.5 Sketches of specific energy diagrams in (a) a general channel and (b) in a rectangular channel in which the bottom width changes, but Q = constant.

For a rectangular channel, it is convenient to deal with the flow rate per unit width of channel, q = Q/b. For a rectangular channel the specific energy can be written as follows if α is assumed equal to 1, and the bottom slope of the channel is small enough so that the $\cos\theta = 1$:

$$E = Y + \frac{V^2}{2g} = Y + \frac{q^2}{2gY^2} \tag{2.11b}$$

A plot of the depth Y as the ordinate, and the specific energy E as the abscissa is referred to as a specific energy diagram. In Figure 2.5 two sketches of specific energy diagrams are given. The first applies for any channel, and the second is specific for a rectangular channel. The following should be observed:

1. If the specific energy is held constant (i.e., a vertical lines is drawn on the specific energy diagram) then there are two depths. These depths are called alternative depths. The flow associated with the larger of these two depths produces **subcritical flow**, and the smaller depth is associated with **supercritical flow**.
2. As these two depths merge into a single depth, a minimum value for the specific energy occurs that can exist for any given flow rate. This depth is called **critical depth**, and the flow associated with it is called **critical flow** and will be denoted by Y_c. The specific energy associated with critical depth will be denoted by E_c.
3. As the flow rate in a given channel increases the specific energy curve is shifted to the right and upward, i.e., the critical depth is increased, and both the subcritical and supercritical depths are closer to the critical depth. In dealing with a rectangular channel the flow rate per unit width q increases (with Q constant) when the bottom width of the channel becomes less, i.e., a channel transition reduces the width of the channel.

That there are generally two depths associated with any given value of the specific energy can be understood best by examining Equation 2.11b. By multiplying this equation by Y^2 one notes that Equation 2.11b is a cubic equation. A cubic equation has three roots, generally, and if there are imaginary roots they occur in pairs. The third root of Equation 2.11b gives a negative value for Y, but since a negative depth is physically not possible, this root is ignored. From a mathematical view point, if the specific energy is reduced to a value less than the critical value, E_c, for a given flow rate in a given channel, then the alternative depths become imaginary, or complex roots of Equation

2.11b. For a general channel it is not obvious what type of equation (Equation 2.11b) represents, but two real positive roots of Y exist for values of $E > E_c$ and these are called alternative depths.

A simple explicit equation exists for computing the alternate depths in a **rectangular** channel. This equation can be used to obtain the depth upstream from a gate if the downstream depth is known or the downstream depth if the upstream depth is known, for example. To obtain this equation, equate E_1 to E_2 or

$$Y_1 + \frac{q^2}{2gY_1^2} = Y_2 + \frac{q^2}{2gY_2^2} \tag{2.11c}$$

Let the specific energy computed from the known depth be denoted by E_k and the other depth be given without a subscript. The Equation 2.11c becomes the following cubic equation:

$$Y^3 - E_k Y^2 + \frac{q^2}{2g} = 0 \tag{2.11d}$$

Since one depth Y_k is known this equation can be reduced to a quadratic equation by synthetic division or

$$\begin{array}{cccc|c} 1 & -E_k & 0 & q^2/2g & Y_k \\ & Y_k & Y_k(Y_k - E_k) & & \\ \hline 1 & Y_k - E_k & Y_k(Y_k - E_k) & 0 & \end{array}$$

giving

$$Y^2 + (Y_k - E_k)Y + (Y_k - E_k)Y_k = 0$$

Solving for Y gives

$$Y = \frac{1}{2}\left\{E_k - Y_k + \sqrt{(E_k - Y_k)^2 + 4Y_k(E_k - Y_k)}\right\} \tag{2.11e}$$

Since $E_k - Y_k$ is the velocity head $V_k^2/(2g) = V_h$ associated with the known depth, Y_k, Equation 2.11e can be written as

$$Y = \frac{1}{2}\left(V_h + \sqrt{V_h^2 + 4Y_k V_h}\right) \tag{2.11f}$$

Note that Equations 2.11e and f are valid only for situations in which $E_1 = E_2$ and the flow rate per unit width in the rectangular channel is the same at the two positions 1 and 2.

To find the minimum value of the specific energy, i.e., E_c and the corresponding critical depth that is associated with critical flow, the well known principle of calculus can be employed of setting the first derivative of E with respect to Y equal to zero. This principle will first be applied to Equation 2.11b, and later to 2.11a. Differentiation of Equation 2.11b with respect to Y and equating dE/dY to zero gives (with q held constant for a given specific energy curve)

$$\frac{dE}{dY} = 1 - \frac{q^2}{gY^2} = 0 \quad \text{or} \quad q^2 = gY^2 \quad \text{or} \quad q = \sqrt{gY}$$

which can be rewritten in several different ways as given by the following equations that define critical flow (the subscript c has been added to emphasize that these equations define critical flow conditions):

$$q_c = \sqrt{gY_c^3} \tag{2.12}$$

$$Y_c = \left(\frac{q_c^2}{g}\right)^{\frac{1}{3}} \tag{2.12a}$$

$$\frac{V_c^2}{2g} = \frac{Y_c}{2} \quad \text{(velocity head equal 1/2 the critical depth)} \tag{2.12b}$$

$$Y_c = \left(\frac{2}{3}\right)E_c \tag{2.12c}$$

$$E_c = 1.5Y_c \tag{2.12d}$$

It needs to be noted that Equations 2.12 apply only for rectangular channels.

Since the energy equation for a rectangular channel is a cubic equation (Equation 2.11d) in terms of depth Y, then if the specific energy E, and the flow rate per unit width q are known then the general solution of a cubic equation can be used (see CRC standard Math. Tables for methods to solve cubic equations), to solve it. This solution procedure consists of first substituting x + E/3 for Y in Equation 2.11d to eliminate the squared term. This substitution produces

$$x^3 - \frac{1}{3}E^2x + \frac{q^2}{2g} - \frac{2}{27}E^3 = 0$$

Let $a = -E^2/3$ and $b = q^2/(2g) - 2E^3/27$. Then the three roots for x are solved for next by the following three equations:

$$x_1 = A + B, \quad x_2 = 0.5(i\sqrt{3})(A - B) - 0.5x_1, \quad \text{and} \quad x_3 = 0.5(i\sqrt{3})(B - A) - 0.5x_1$$

in which

$$A = \left\{\left(\frac{b^2}{4} + \frac{a^3}{27}\right)^{1/2} - 0.5b\right\}^{1/3} \quad \text{and} \quad B = \left\{-\left(\frac{b^2}{4} + \frac{a^3}{27}\right)^{1/2} - 0.5b\right\}^{1/3}$$

If $b^2/4 + a^3/27 > 0$. there will be one real root and two conjugate imaginary roots, i.e., the specific energy $E < E_c$ and physically only a meaningless negative solution for Y exists.

If $b^2/4 + a^3/27 = 0$, there will be three real roots of which at least two are equal, i.e., critical flow occurs in which the two real roots gives the critical depth, and the other real root is a meaningless negative solution for Y. If $b^2/4 + a^3/27 > 0$ there will be three real and unequal roots, i.e., the two positive roots are the alternative depths and the third root is a meaningless negative depth Y.

After the solution for x values have been obtained the final step is to obtain the depths Y from

$$Y_1 = x_1 + \frac{1}{3}E; \quad Y_2 = x_2 + \frac{1}{3}E; \quad Y_3 = x_3 + \frac{1}{3}E$$

The program ALTDEP, listed below, implements this method for solving for the alternative depths. Generally to use the program one would first solve the specific energy equation to get E for the known depth, and then select the other depth from the output from the program as the alternative depth. However, the program lets us know what portion of the specific energy diagram the problem as been specified in. If two of the root are imaginary and the program gives the message "$E < E_c$ so negative real root and 2 imaginary roots" we know that physically the energy must be increased for the specified flow rate to be possible. An alternative to using the complex arithmetic involved in the above procedure for solving a general cubic equation is to use an implicit solution method, such as Newton's method, to solve for the depth desired by providing an appropriate guess, or software such as TK-Solver or Mathcad (or an HP calculator or a spreadsheet) that has the capability to solve implicit equations.

Listing of program ALTDEP.FOR

```
      COMPLEX AA,BB,C3,ARG,X1,X2,X3
      C3=.5*CSQRT(CMPLX(-3.,0.))
1     WRITE(*,*)' Give q, E & g or 0/=STOP'
      READ(*,*) Q,E,G
      IF(Q.LT.1.E-8) STOP
      A=-E*E/3.
      B=Q*Q/(2.*G)+2.*A*E/9.
      BH=.5*B
      BS4=B*B/4.
      YC=(Q**2/G)**.33333333
      ARG=CSQRT(CMPLX(BS4,0.)+CMPLX(A,0.)**3/27.)
      WRITE(*,*)' Yc =',YC,' Ec =',1.5*YC
      AARG=BS4+A**3/27.
      IF(AARG.GT.0.) THEN
      WRITE(*,*)' E<Ecsonegative real root and' &,'2',' imaginary
     &roots'
      ELSEIF(AARG.GT.-1.E-5 .AND. ARG.LT.1.E-5) THEN
      WRITE(*,*)' Critical condition'
      ELSE
      WRITE(*,*)'Alternative Depths&negative Y'
      ENDIF
      AA=(ARG-CMPLX(BH,0.))**.33333333
      BB=(-ARG-CMPLX(BH,0.))**.33333333
      X1=AA+BB
      X2=(AA-BB)*C3-.5*X1
      X3=(BB-AA)*C3-.5*X1
      WRITE(*,100) X1+E/3.,X2+E/3.,X3+E/3.
100   FORMAT(3(2F9.3,3X))
      GO TO 1
      END
```

Listing of Program ALTDEP.CPP

```
#include <iostream.h>
#include <stdlib.h>
#include <complex.h>
#include <iomanip.h>
void main(void){float a,b,q,e,g,bh,bs4,yc,aarg;
 complex aa,bb,c3,arg,x1,x2,x3;
 c3=.5*sqrt(3.)*complex(0.,1.);
L1:cout <<"Give q, E & g or 0 0 0 =STOP"<< endl;
 cin >>q>>e>>g; if(q<1.e-5) exit(0);
 a=-e*e/3.; b=q*q/(2.*g)+2.*a*e/9.; bh=.5*b;bs4=b*b/4.;
 yc=pow(q*q/g,.3333333);
 arg=sqrt(complex(bs4,0.)+pow(complex(a,0.),3.)/27.);
 cout <<"Yc ="<<yc<<" Ec ="<<1.5*yc<<"\n"; aarg=bs4+a*a*a/27.;
 if(aarg>0.)
   cout<<"E<Ec so negative real root and 2 imaginary roots"<<endl;
 else if((aarg>-1.e-5)&&(aarg<1.e-5))\
   cout <<"Critical condition"<< endl;
 else cout <<"Alternative Depths & negative Y"<< endl;
 aa=pow(arg-complex(bh,0.),.3333333);
   bb=pow(-arg-complex(bh,0.),.3333333);
 x1=aa+bb;x2=(aa-bb)*c3-.5*x1;x3=(bb-aa)*c3-.5*x1; cout.width(9);
 cout<<setprecision(3)<<x1+e/3.<<" "<<x2+e/3.<<" "<<x3+e/3.<< endl;
 goto L1;}
```

Example use of program

```
Input: 5 2 32.2
Output:
  Yc =   9.190971E-01  Ec =       1.378646
  Alternative depths and negative Y
    1.891     .000        .511     .000       -.402     .000
Input: 5 1.378646 32.2
Output:
  Yc =   9.190971E-01  Ec =       1.378646
  Critical condition
     .920     .000        .919     .000        -.460     .000
Input: 5 1 32.2
Output:
  Yc =   9.190971E-01  Ec =       1.378646
  E<Ec so negative real root and 2 imaginary roots
     .754     .444        .754    -.444        -.507     .000
Input: 0/
```

The above procedure can be simplified by using the arc cosine (and subsequently the cosine). To use this alternate method, first compute the angle θ from

$$\theta = \cos^{-1}\left\{\frac{\left[\left(6.75q^2/g - E^3\right)/27\right]}{(E/3)^3}\right\}$$

The three roots are then obtained from

$$Y_1 = \left(\frac{E}{3}\right)\left\{1 - 2\cos\left(\frac{\theta}{3}\right)\right\} \quad \text{(negative depth)}$$

$$Y_2 = \left(\frac{E}{3}\right)\left\{1 - 2\cos\left(\frac{[\theta + 2\pi]}{3}\right)\right\}$$

$$Y_3 = \left(\frac{E}{3}\right)\left\{1 - 2\cos\left(\frac{[\theta + 4\pi]}{3}\right)\right\}$$

Of course, if $E = E_c$ the two latter depths become equal, or become the critical depth Y_c associated with the given q. If the given E is less than the critical depth, then Y_2 and Y_3 are imaginary and the above procedure will not work because the argument for the arc cosine is not within the allowable range of −1 to +1. Thus if this alternative method is implemented in a computer program, as in ROOTSE listed below, the critical specific energy E_c, as described below, should be computed and if $E < E_c$ the computation of the roots should not be attempted.

Program ROOTSE.FOR

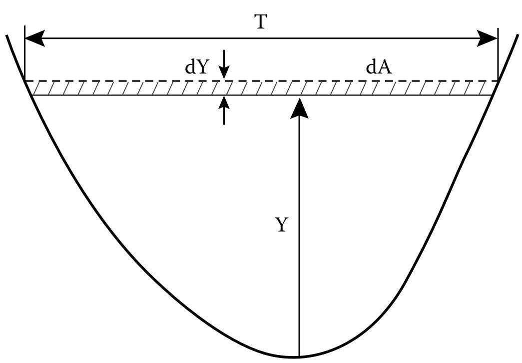
```
      PARAMETER (PI=3.14159265)
      REAL X(3)
      PI2=2.*PI
      PI4=4.*PI
1     WRITE(*,*)' Give: q,E,g'
      READ(*,*) q,E,g
      IF(q.LT.1.E-5) STOP
      Yc=(q*q/g)**.33333333
      Ec=1.5*Yc
      IF(E.LT.Ec) THEN
      WRITE(6,110) q,E,g,Yc,Ec
110   FORMAT(' q=',F8.3,' E=',F8.3,' g=',F8.2,/' Only 1 real
     &root(neg.)',' Yc=',F8.3,' Ec=',F8.3)
      GO TO 1
      ENDIF
      E3=E/3.
      THETA=ACOS(((6.75*q*q/g-E**3)/27.)/E3**3)
      X(1)=E3*(1.-2.*COS(THETA/3.))
      X(2)=E3*(1.-2.*COS((THETA+PI2)/3.))
      X(3)=E3*(1.-2.*COS((THETA+PI4)/3.))
      WRITE(6,100) q,E,g,Yc,Ec,X
100   FORMAT(' q=',F8.3,' E=',F8.3,' g=',F8.2,/,' Yc=',F8.3,'
     &Ec=',F8.3,' Roots are:',/,3F9.3)
      GO TO 1
      END
```

Program ROOTSE.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
void main(void){float pi=3.14159265,q,g,e,yc,ec,theta,e3,y1,y2,y3;
L1:printf("Give: q,E,g\n"); scanf("%f %f %f",&q,&e,&g);
 if(q<1.e-5) exit(0);
 yc=pow(q*q/g,.33333333); ec=1.5*yc;
 if(e<ec) {printf("q=%8.3f E=%8.3f g=%8.2f \nOnly 1 real \
   root(beg.)\n Yc=%8.3f Ec=%8.3f\n",q,e,g,yc,ec); goto L1;}
 e3=e/3.; theta=acos(((6.75*q*q/g-pow(e,3.))/27.)/pow(e3,3.));
 y1=e3*(1.-2.*cos(theta/3.)); y2=e3*(1.-2.*cos((theta+2.*pi)/3.));
 y3=e3*(1.-2.*cos((theta+4.*pi)/3.));
 printf("q=%8.3f       E=%8.3f      g=%8.3f\nYc=%8.3f \
   Ec=%8.3f\nRoots      are:\n%9.3f      %8.3f \
   %8.3f\n",q,e,g,yc,ec,y1,y2,y3);
 goto L1;}
```

For any channel Equation 2.11a is differentiated, and dE/dY is set to zero to find the minimum value of E. This procedure leads to

$$\frac{Q^2}{gA^3}\frac{dA}{dY} = 1$$



From the accompanying sketch it is clear that dA/dY equals the top width T for all channels and therefore the critical flow equation for all channels becomes

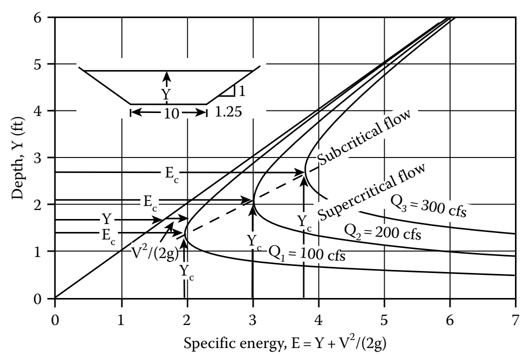
$$\frac{Q_c^2 T_c}{gA_c^3} = F_r^2 = \frac{V^2}{c^2} = 1 \tag{2.13}$$

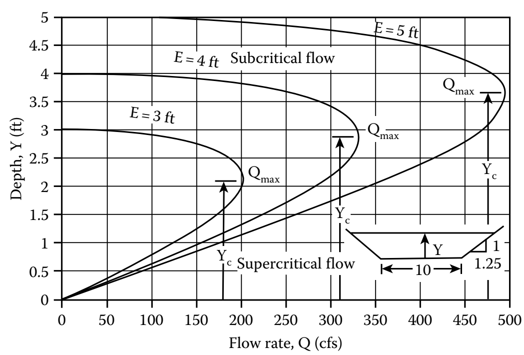
in which F_r is the Froude number which is defined as the ratio of inertia to gravity forces. The speed at which a small amplitude gravity wave travel in an open channel is given by

$$c = \sqrt{\frac{gA^3}{T}} \tag{2.14}$$

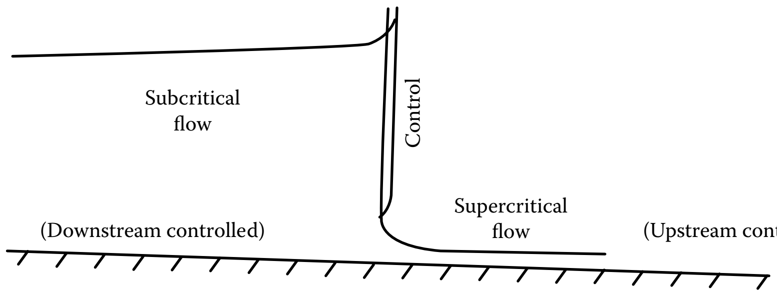
Proof of this equation is given in Chapter 3. From Equations 2.13 and 2.14, we see that the Froude number is also the ratio of the average velocity in the channel to the speed of a small amplitude gravity wave. From Equation 2.13, it can be concluded that critical flow occurs when the average velocity V of a channel flow exactly equals the speed c of a small amplitude gravity wave. For subcritical flows the Froude number F_r will be less than unity, since the denominator of Equation 2.13 increases with increasing depth, and for supercritical flows the Froude number will be larger than unity.

In a subcritical flow, the fact that the velocity in the channel is less than the speed of a small amplitude gravity wave allows these waves to move upstream against the flowing fluid. Thus in a sense the fluid can receive a signal that things are to change downstream, and it adjust gradually. This adjusting to downstream conditions is referred to as downstream control. Subcritical flows are always controlled by downstream conditions, i.e., an obstruction to the flow such as a partly closed gate will cause the depth upstream from the gate to increase, and the velocity to decrease. On the other hand if the flow is supercritical, the velocity in the channel exceeds the speed of a small amplitude gravity wave, and the flow does not receive a signal about downstream conditions. Therefore, if the flow is supercritical it will not change its depth or velocity to meet downstream conditions. For example, should the channel abruptly end, a supercritical flow would continue to the end of the channel at the same depth and velocity as it has upstream there from. Supercritical flows are, therefore, said to be upstream controlled. An example of a control that affects both the upstream flow as well as the downstream flow conditions is a sluice gate in a channel flow. Since subcritical flows are controlled downstream and supercritical flows are controlled by an upstream device we can conclude that the flow upstream from the gate must be subcritical, and the flow downstream from the gate must be supercritical.





Above a specific energy diagram and a depth-discharge diagram for a trapezoidal channel are given. The separate curves on the depth-discharge diagram show the relationship of the flow rate to changes in depth for a constant specific energy. The maximum point on these curves corresponds to critical conditions, i.e., the depth is critical and the Q_{max} is the maximum flow rate that could be obtained into this trapezoidal if it were supplied by a reservoir with a water surface elevation equal to the E for that curve. This assumes there is no entrance loss. You might note that the minimum point on the specific energy diagram corresponds to the maximum point on the depth-discharge diagram. For example, the curve for E = 3 ft on the depth–discharge diagram gives a maximum flow rate Q_{max} = 200 cfs, and the minimum specific energy on the specific energy diagram of the curve on the for Q = 200 cfs is E_c = 3 ft.



To further illustrate the meanings of these diagrams assume a reservoir with a head H = 5 ft supplies a steep trapezoidal channel with b = 10 ft, and m = 1.25, the flow rate that will enter the channel is Q = 494 cfs (the maximum flow rate possible on the depth-discharge diagram of the curve for E = 5 ft), and the depth at the entrance will be $Y_c = 3.62$ ft, the ordinate of the maximum point of this curve. These values are at the extreme, Q_{max} position of the curve for E = 5 ft, but these values could also be obtained by solving the critical flow equation $Q^2T/(gA^3) = 1$ and the specific energy equation $5 = Y + (Q/A)^2/(2g)$ simultaneously. Likewise if the reservoir level were 4 ft above the channel bottom $Q_c = Q_{max} = 331.5$ cfs and $Y_c = 2.87$ ft, or if H = 3 ft, $Q_c = Q_{max} = 201.2$ cfs and $Y_c = 2.12$ ft. It is not possible for the flow from a reservoir into a steep channel to be in the lower (the supercritical) portion of the depth-discharge diagram, because the flow rates would be less than the amount Q_{max} that can be supplied. However, if the channel is not steep, but mild, then the flow into the channel will be reduced by downstream conditions, i.e., the fluid frictional resistance. For example, for the trapezoidal channel for which the depth-discharge diagram is made if the reservoir supplied a specific energy of 5 ft, and the flow rate were 400 cfs, the depth would be 4.5 ft in the channel. If the channel has an n = 0.014, then Manning's equation could be solved to show that it would have to have a bottom slope of $S_o = 0.000701$ for this uniform flow to occur.

The need to determine critical depth, or critical flow conditions, occurs frequently in open channels. The illustrative problems, as well as the problems at the end of this chapter, point out a few situations where computation of critical depth is needed. Therefore, it is worth discussing how the above critical flow Equations 2.12 and 2.13 can be effectively solved. Should the channel be rectangular the solution of critical flow conditions is very easy since Equations 2.12 are all explicit. However, for a nonrectangular channel Equation 2.13 must be solved for the critical depth Y_c, and since this equation is implicit, the solution requires an iterative technique such as the Newton method. Alternatively graphical solution can be utilized as discussed below to get answers with adequate precision for most applications.

To solve Equation 2.13 by the Newton method, it can be rewritten as

$$F = Q_c^2 T_c - gA_c^2 = 0 \tag{2.15}$$

in which T_c and A_c are functions of the critical depth Y_c and the parameters that define the given channel.

EXAMPLE PROBLEM 2.10

Water enters a steep rectangular channel that has a bottom width of 8 ft from a reservoir whose water surface is 5 ft above the channel bottom. Determine the flow rate into the channel, and the depth of flow at the channel entrance. Neglect the entrance loss coefficient. A long distance downstream a gate exists that produces a depth $Y_2 = 1.5$ ft downstream from it. What is the depth immediately upstream from the gate?

Solution

Equations 2.12 apply to this problem because the channel is rectangular. Since the channel is steep, and a steep channel will contain a supercritical flow under uniform flow conditions, and the water in the reservoir (where V = 0) is subcritical, the flow must pass through critical depth at the entrance of the channel. Therefore, $Y_c = (2/3)\ E = (2/3)5 = 3.333$ ft, and the flow rate per unit width $q = (gY_c^3)^{1/2} = (32.3(3.333)^3)^{1/2} = 34.53$ cfs/ft, or Q = bq = 8(34.53) = 276.3 cfs. The second part of the problem requires that the specific energy downstream from the gate be solved, or $E_2 = Y_2 + q^2/(2gY_2^2) = 1.5 + (34.53/1.5)^2/64.4 = 9.729$ ft. Now the alternative depth to 1.5 is sought with E = 9.729 ft. One way is to extract the root 1.5 from the cubic equation, another is to solve the cubic equation with the Newton method starting with a "subcritical" guess for Y_1, or use the program ALTDEP. The solution for the depth upstream from the gate is $Y_1 = 9.525$ ft. A hydraulic jump (discussed in Chapter 3) will occur somewhere upstream from the gate changing the flow from super to subcritical.

EXAMPLE PROBLEM 2.11

Instead of the rectangular channel a steep pipe of 8 ft diameter is used. What is the flow rate and depth of flow at the entrance? A gate downstream causes a depth downstream from it of $Y_2 = 2.8$ ft. What is the depth immediately upstream from the gate?

Solution

Since this is a circular channel, the implicit Equation 2.13 applies. Since there are two unknowns, Y and Q, a second equation must be obtained. This second equation is the specific energy Equation 2.11a. If angle β is used initially as a substitute for Y, the two equations that must be solved simultaneously are

$$F_1 = Q^2(D\sin\beta) - g\left\{\frac{D^2}{4}(\beta - \cos\beta\sin\beta)\right\}^3$$

and

$$F_2 = \frac{E - D}{2(1-\cos\beta)} - \frac{\{1-K_L\}Q^2}{2g}\left\{\frac{D^2}{4}(\beta - \cos\beta\sin\beta)\right\}^2$$

The solution to these equations by the Newton method uses the following iterative equation

$$\begin{Bmatrix} Q \\ \beta \end{Bmatrix}^{(m+1)} = \begin{Bmatrix} Q \\ \beta \end{Bmatrix}^{(m)} - \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix}$$

in which the Z values are obtained by solving the following linear system of two equations:

$$\begin{bmatrix} \dfrac{\partial F_1}{\partial Q} & \dfrac{\partial F_1}{\partial \beta} \\ \dfrac{\partial F_2}{\partial Q} & \dfrac{\partial F_1^2}{\partial \beta} \end{bmatrix} \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix}$$

in which F_1 and F_2 and its derivatives are evaluated using the current values, i.e., those with an m superscript. The solution gives Q = 207.2 cfs, and β = 1.474 rad, and from this angle $Y_c = D(1 - \cos\beta)/2 = 3.615$ ft. This solution assumed $K_L = 0$, e.g., ignores entrance losses. If the entrance loss coefficient is taken as 0.1, then the solution is Q = 197.2 cfs, and Y = 3.523 ft. To solve part two, $E_2 = Y_2 + (Q/A_2)^2/(2g) = 5.512$ ft. Now the energy equation must be solve using an implicit method because extracting a root or using the general solution to a cubic equation is not available. In fact since the variable β relates Y to A, it also needs to be solved. The solution is β = 1.787 rad, and $Y_1 = 4.859$ ft.

EXAMPLE PROBLEM 2.12

Instead of the rectangular channel (with b = 8 ft) of Example Problem 2.10 having a steep slope this problem deals with this channel on a "mild slope" with S_o = 0.00075. The channel has a Manning's n = 0.014, and is long. What are the depth and the flow rate? The entrance loss coefficient is K_e = 0.1. As a second part solve for the flow rate and the uniform depth if the channel is trapezoidal with b = 8 ft, and m = 1.2. (S_o is still 0.00075, n = 0.014 and H = 5 ft). Also solve these problems using Chezy's, rather than, Manning's equation, if e = 0.01 ft. As a fourth part of this problem assume the bottom width b of the trapezoidal channel is sought that will supply a flow rate Q = 400 cfs.

Solution

For these mild channels the two equations that govern are: (1) a uniform flow equation which is Manning's equation for the first part of the problem or

$$Q = \frac{C_u}{n} A R_h^{2/3} S_o^{1/2} \quad \text{or} \quad F_1 = nQP^{2/3} - C_u A^{5/3} S_o^{1/2} = 0$$

and (2) the energy equation

$$H = Y + \frac{1-K_L}{2g}\frac{Q^2}{\lambda^2} \quad \text{or} \quad F_2 = H - Y - \frac{1-K_L}{2g}\frac{Q^2}{\lambda^2} = 0$$

Using the Newton method, the solution to these two equations for the rectangular channel is: Q = 177.72 cfs, and Y = 4.602 ft, and for the trapezoidal channel is: Q = 336.03 cfs, and Y = 4.451 ft.

The program E_UN listed below is designed to solve Manning's and energy equations simultaneously using the Newton method. You should study either the FORTRAN or C versions of these programs to understand how this solution is accomplished. The subroutine FUN (void function in the C program) supplies the values to the two above equations whenever it is called. The main program numerically evaluates the four derivatives in the Jacobian matrix using $\partial F_i/\partial x = \{F_i(x + \Delta x) - F_i(x)\}/\Delta x$, in which Δx is obtained by multiplying the current value of the unknown x by 1.005 and then subtracting its current value or $\Delta x = 1.005x - x = 0.005x$. Carefully study the listing starting with the statement 30 SUM=0. to the statement IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 30 (the do {} while; in the C-program) to see how the Newton method is implement in solving a system of simultaneous equation because this approach will be used repeatedly. The subroutine SOLVEQ is called on to solve the linear system of equation even though a 2 × 2 matrix such as occurs in this program could be solved with a few lines of code. Notice that the program is designed to solve for any two of the first 7 variables 1 = b, 2 = m, 3 = S_o, 4 = n, 5 = Q, 6 = H, 7 = Y, and 8 = K in the array X() and that the array ID() is given a value of 1 to identify these two unknown variables and a value of 0 if the variable is known. Also notice that the program accommodates either a trapezoidal (which includes a rectangular channel with m = 0) and a circular channel by giving ITYPE a 0 or a 1 respectively.

Program E_UN.FOR

```
C Solves Manning's (uniform flow) and Energy simultaneously
C for any 2 unknowns
C  See E_UN1 to solve Q & Y and method that can be used
C  with calculator
      CHARACTER*17 FMT/'(1X,A1,'' ='',F9.3)'/
      CHARACTER*1 CX(8)/'b','m','S','n','Q','H','Y','K'/
      CHARACTER*5 CH(0:1)/'value','guess'/
      READ(*,*) G,FKE,ITYPE
1     DO 10 I=1,7
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 10
      WRITE(*,'(I2,2X,A1)') I,CX(I)
10    ID(I)=0
```

```
2     WRITE(*,*)' Give two numbers for 2 unknown variables'
      READ(*,*) I1,I2
      IF(I1.LT.1.OR.I1.GT.7.OR.I2.LT.1.OR.I2.GT.7) GO TO 2
      ID(I1)=1
      ID(I2)=1
      DO 20 I=1,7
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 20
      WRITE(*,100) CH(ID(I)),CX(I)
100   FORMAT(' Give ',A5,' for ',A1,' = ',\)
      READ(*,*) X(I)
20    CONTINUE
      NCT=0
30    SUM=0.
      CALL FUN(F)
      I1=0
      DO 40 I=1,7
      IF(ID(I).EQ.0) GO TO 40
      XX=X(I)
      I1=I1+1
      X(I)=1.005*X(I)
      CALL FUN(F1)
      DO 35 J=1,2
      IF(ITYPE.EQ.1) THEN
      CX(1)='D'
      CX(2)=' '
      ENDIF
      IF(G.GT.15.) THEN
      Cu=1.486
      ELSE
      Cu=1.
      ENDIF
      X(8)=FKE
      FKE=(1.+FKE)/(2.*G)
      REAL F(2),F1(2),D(2,2)
      INTEGER*2 ID(8),INDX(2)
      COMMON X(8),G,FKE,Cu,ITYPE
      WRITE(*,*)' Give:g,entr. loss C. & 0=TRAP or 1=CIRLCE'
35    D(J,I1)=(F1(J)-F(J))/(X(I)-XX)
      X(I)=XX
40    CONTINUE
      CALL SOLVEQ(2,1,2,D,F,1,DD,INDX)
      I1=0
      DO 50 I=1,7
      IF(ID(I).EQ.0) GO TO 50
      I1=I1+1
      SUM=SUM+ABS(F(I1))
      X(I)=X(I)-F(I1)
50    CONTINUE
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,' SUM=',SUM
      IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 30
      WRITE(*,*)' Solution:'
      DO 60 I=1,8
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 60
      IF(I.EQ.3) THEN
      FMT(16:16)='6'
```

```
      ELSEIF(I.EQ.4) THEN
      FMT(16:16)='3'
      ELSE
      FMT(16:16)='3'
      ENDIF
      WRITE(*,FMT) CX(I),X(I)
60    CONTINUE
      WRITE(*,*)' Give 1 to solve another prob. (0=STOP)'
      READ(*,*) I2
      IF(I2.EQ.1) GO TO 1
      END
      SUBROUTINE FUN(F)
      REAL F(2)
      COMMON X(8),G,FKE,Cu,ITYPE
      IF(ITYPE.EQ.1) THEN
      BETA=ACOS(1.-2.*X(7)/X(1))
      A=.25*X(1)**2*(BETA-COS(BETA)*SIN(BETA))
      P=X(1)*BETA
      ELSE
      A=(X(1)+X(2)*X(7))*X(7)
      P=X(1)+2.*X(7)*SQRT(X(2)**2+1.)
      ENDIF
      F(1)=X(4)*X(5)*P**.6666667-Cu*A**1.6666667*SQRT(X(3))
      F(2)=X(6)-X(7)-FKE*(X(5)/A)**2
      RETURN
      END
```

Program E_UN.C

```
// Solves Manning's (uniform flow) and Energy simultaneously for
// any 2 unknowns
// See E_UN1 to solve Q & Y and method that can be used with
// calculator
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float x[8],g,fke,cu; int itype;
extern void solveq(int n,float **d,float *f,int itype,float *dd,\
  int *indx);
void fun(float *f){float beta,a,p;
 if(itype){beta=acos(1.-2.*x[6]/x[0]);
   a=.25*x[0]*x[0]*(beta-cos(beta)*sin(beta));p=x[0]*beta;}
 else {a=(x[0]+x[1]*x[6])*x[6];
   p=x[0]+2.*x[6]*sqrt(x[1]*x[1]+1.);}
 f[0]=x[3]*x[4]*pow(p,.6666667)-cu*pow(a,1.6666667)*sqrt(x[2]);
 f[1]=x[5]-x[6]-fke*pow(x[4]/a,2.);return;} //End of fun
void main(void){char *fmt=" %c =%9.3f\n",*ch[]={"value","guess"},\
  *cx="bmSnQHYK";
 float f[2],f1[2],xx,sum,**d,*dd; int id[8],indx[2],i,j,i1,i2,nct;
 d=(float**)malloc(2*sizeof(float*));
 for(i=0;i<2;i++)d[i]=(float*)malloc(2*sizeof(float));
 printf("Give: g,entrance loss C. & 0=TRAP or 1=CIRLCE\n");
 scanf("%f %f %d",&g,&fke,&itype);
 if(itype){stpcpy(cx[0],"D");stpcpy(cx[1]," ");} if(g>15.) cu=1.486;
 else cu=1.; x[7]=fke;fke=(1.+fke)/(2.*g);
L1: for(i=0;i<7;i++){if((!itype) || ((itype)&&(i != 1)))\
  printf("%2d %c\n",i+1,cx[i]);id[i]=0;}
```

```
do{printf(" Give two numbers for 2 unknown variables\n");
  scanf("%d %d",&i1,&i2);}while((i1<1)||(i1>7)||(i2<1)||(i2>7));
id[i1-1]=1;id[i2-1]=1;
for(i=0;i<7;i++){if((!itype) || ((itype)&&(i != 1)))
  {printf("Give %s for %c =",ch[id[i]],cx[i]);
scanf("%f",&x[i]);}} nct=0;
do{sum=0.; fun(f); i1=-1;
for(i=0;i<7;i++){if(id[i]){xx=x[i];i1++;x[i]*=1.005;fun(f1);
  for(j=0;j<2;j++) d[j][i1]=(f1[j]-f[j])/(x[i]-xx); x[i]=xx;}}
solveq(2,d,f,1,dd,indx); i1=-1;
for(i=0;i<7;i++){if(id[i]){sum+=fabs(f[++i1]);x[i]-=f[i1];}}
printf("nct= %d SUM=%f\n",nct,sum);}while((++nct<30)&&(sum>1.e-5));
printf("Solution:\n");for(i=0;i<8;i++){
  if((!itype) || ((itype)&&(i != 1))){
  if(i==2)stpcpy(fmt[8],"6"); else stpcpy(fmt[8],"3");
  printf(fmt,cx[i],x[i]);}}
printf("Give 1 to solve another prob. (0=STOP)\n");
scanf("%d",&i2); if(i2) goto L1;}
```

The above listing of the program will solve the fourth part of the problem in which a flow rate of Q = 400 cfs is given and b is wanted along with Y. The difference is for parts 1 and 2 variables 5 and 7 (Q and Y) are given as the unknowns and for part 3 variables 1 and 7 (b and Y) are identified as the unknowns. The solution for part 3 is b = 10.194 ft and Y = 4.416 ft.

If the flow rate Q and depth Y are always the two unknowns, then it is not necessary to solve the two equations simultaneously. Rather Manning's equation is substituted into the energy to eliminate the flow rate Q, or $F = H - Y - (1 + K_e)(C_u/n)^2(A/P)^{4/3}S_o/(2g) = 0$ and solve this equation for the depth Y. Thereafter solve for Q from either of the original equations. This is the approach you would use to solve parts 1 and 2 with a pocket calculator such as an HP with a SOLVE function. The program E_UN1, listed below use this latter approach to solve for Q and Y.

Program E_UN1.FOR

```
C Solves Manning's and Specific Energy for Y & Q by substituting
  Manning's equation
C into Energy and first solving for Y and thereafter Q. (This is
  the procedure
C to use with a pocket calculator) See program E_UN to solve any
  2 variables.
      COMMON B,FM,D,FN,SO,H,C,ITYPE
      WRITE(*,*)' Give: g,(0=trap or 1=cir)'
      READ(*,*) G,ITYPE
      IF(ITYPE.EQ.0) THEN
      WRITE(*,*)' Give: b,m,n,So,H,Ke'
      READ(*,*) B,FM,FN,SO,H,FKE
      ELSE
      WRITE(*,*)' Give: D,n,So,H,Ke'
      READ(*,*) D,FN,SO,H,FKE
      ENDIF
      Y=.8*H
      Cu=1.
      IF(G.GT.15.) Cu=1.486
      C=(Cu/FN)**2*SO*(1.+FKE)/(2.*G)
      NCT=0
10    F=FUN(Y)
      DIF=.05*F/(FUN(Y+.05)-F)
      Y=Y-DIF
      NCT=NCT+1
```

```
      WRITE(*,*) NCT,DIF
      IF(NCT.LT.30 .AND. ABS(DIF).GT. 1.E-5) GO TO 10
      CALL AREAP(Y,A,P)
      Q=CU/FN*A*(A/P)**.6666667*SQRT(SO)
      WRITE(*,100) Y,Q
100   FORMAT(' Solution: Y =',F8.3,' Q =',F10.2)
      END
      SUBROUTINE AREAP(Y,A,P)
      COMMON B,FM,D,FN,SO,H,C,ITYPE
      IF(ITYPE.EQ.1) THEN
      COSB=1.-2.*Y/D
      BETA=ACOS(COSB)
      P=D*BETA
      A=.25*D*D*(BETA-COSB*SIN(BETA))
      ELSE
      P=B+2.*Y*SQRT(FM**2+1.)
      A=(B+FM*Y)*Y
      ENDIF
      RETURN
      END
      FUNCTION FUN(Y)
      COMMON B,FM,D,FN,SO,H,C,ITYPE
      CALL AREAP(Y,A,P)
      FUN=H-Y-C*(A/P)**1.3333333
      RETURN
      END
```

Program E_UN1.C

```
/* Solves Manning's and Specific Energy for Y & Q by
  substituting Manning's Eq.into Energy and first solving for Y
  and thereafter Q. (This is the procedure to use with a pocket
  calculator) See program E_UN to solve any 2 variables. */
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float b,m,d,n,so,h,c; int itype;
void areap(float y,float *a,float *p){float cosb,beta;
 if(itype){cosb=1.-2.*y/d;beta=acos(cosb);
   *p=d*beta;*a=.25*d*d*(beta-cosb*sin(beta));}
 else {*p=b+2.*y*sqrt(m*m+1.); *a=(b+m*y)*y;return;}}
float fun(float y){float *a,*p; areap(y,a,p);
  return(h-y-c*pow((*a)/(*p),1.333333));}
void main(void){float g,dif,ke,y,f,q,cu=1.,*a,*p; int nct=0;
 printf("Give: g(0=trap or 1=cir)\n"); scanf("%f %d",&g,&itype);
 if(itype){printf("Give: D,n,So,H,Ke\n");
   scanf("%f %f %f %f %f",&d,&n,&so,&h,&ke);}
 else {printf("Give: b,m,n,So,H,Ke\n");
    scanf("%f %f %f %f %f %f",&b,&m,&n,&so,&h,&ke);}
 y=.8*h; if(g>15.) cu=1.486; c=pow(cu/n,2.)*so*(1.+ke)/(2.*g);
 do{f=fun(y); dif=.05*f/(fun(y+.05)-f); y-=dif;
 printf("%d %f %f\n",++nct,dif,y);
 }while((nct<30) && (fabs(dif)>1.e-5));
 areap(y,a,p); q=cu/n*(*a)*pow((*a)/(*p),.6666667)*sqrt(so);
 printf("Solution: Y = %8.3f Q = %10.2f\n",y,q);
}
```

To solve the third part of the problem that requests that Chezy's equation be used instead of Manning equation, requires that 3 (rather than 2) equations be solved simultaneously for Q, Y, and C. These equations are as follows:

$$F_1 = Q - CA(R_h S_o)^{1/2} = 0 \quad \text{(Chezy's equation)}$$

$$F_2 = C + (32g)^{1/2} \text{Log}_{10}\left\{\frac{e}{(12R_h)} + \frac{0.884C}{\left(R_e g^{1/3}\right)}\right\} = 0 \quad \text{(Chezy's C equation)}$$

$$F_3 = H - Y - \left\{\frac{(1+K_e)}{(2g)}\right\}\left\{\frac{Q}{A}\right\}^2 = 0 \quad \text{(Energy equation)}$$

Program UENCHEZ is designed to solve these three equations simultaneously. Notice that now the subroutine FUN provides values to 3 F values, and the part of the program that implements the Newton method has the DO loops changed from 2 to 3. The program UENCHEZ1 reduces the Newton solution to the depth by substituting Chezy's equation into the energy equation, as has been done previous when using Manning's equation as the uniform flow equation. Now however it is necessary to use a Gauss–Seidel type iteration to solve C associated with the current value of Q. This technique can be used with a calculator with a SOLVE function.

Program UENCHEZ.FOR

```
CHARACTER*17 FMT/'(1X,A1,'' ='',F9.3)'/
      CHARACTER*1 CX(9)/'b','m','S','e','Q','H','Y','C','K'/
      CHARACTER*5 CH(0:1)/'value','guess'/
      REAL F(3),F1(3),D(3,3)
      INTEGER*2 ID(9),INDX(3)
      COMMON X(9),G,FKE,G32,CG,ITYPE
      WRITE(*,*)' Give:g,k.vis.,entrance loss C.',
     &' & 0=trap,1=cir'
      READ(*,*) G,VIS,FKE,ITYPE
      IF(ITYPE.EQ.1) THEN
      CX(1)='D'
      CX(2)=' '
      ENDIF
      IF(G.GT.15.) THEN
      X(8)=100.
      ELSE
      X(8)=60.
      ENDIF
      X(9)=FKE
      FKE=(1.+FKE)/(2.*G)
      G32=SQRT(32.*G)
      CG=.884*VIS/(4.*SQRT(G))
1     DO 10 I=1,7
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 10
      WRITE(*,'(I2,2X,A1)') I,CX(I)
10    ID(I)=0
2     WRITE(*,*)' Give 2 num. for 2 unknown var.',
      ' in addition to C'
      READ(*,*) I1,I2
      IF(I1.LT.1.OR.I1.GT.8.OR.I2.LT.1.OR.I2.GT.8) GO TO 2
      ID(I1)=1
      ID(I2)=1
```

```
      ID(8)=1
      DO 20 I=1,7
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 20
      WRITE(*,100) CH(ID(I)),CX(I)
100   FORMAT(' Give ',A5,' for ',A1,' = ',\)
      READ(*,*) X(I)
20    CONTINUE
      NCT=0
30    SUM=0.
      CALL FUN(F)
      I1=0
      DO 40 I=1,8
      IF(ID(I).EQ.0) GO TO 40
      XX=X(I)
      I1=I1+1
      X(I)=1.005*X(I)
      CALL FUN(F1)
      DO 35 J=1,3
35    D(J,I1)=(F1(J)-F(J))/(X(I)-XX)
      X(I)=XX
40    CONTINUE
      CALL SOLVEQ(3,1,3,D,F,1,DD,INDX)
      I1=0
      DO 50 I=1,8
      IF(ID(I).EQ.0) GO TO 50
      I1=I1+1
      SUM=SUM+ABS(F(I1))
      X(I)=X(I)-F(I1)
50    CONTINUE
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,' SUM=',SUM
      IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 30
      WRITE(*,*)' Solution:'
      DO 60 I=1,9
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 60
      IF(I.EQ.3 .OR. I.EQ.4) THEN
      FMT(16:16)='6'
      ELSE
      FMT(16:16)='3'
      ENDIF
      WRITE(*,FMT) CX(I),X(I)
60    CONTINUE
      WRITE(*,*)' Give 1 for another pb. (0=STOP)'
      READ(*,*) I2
      IF(I2.EQ.1) GO TO 1
      END
      SUBROUTINE FUN(F)
      REAL F(3)
      COMMON X(9),G,FKE,G32,CG,ITYPE
      IF(ITYPE.EQ.1) THEN
      BETA=ACOS(1.-2.*X(7)/X(1))
      A=.25*X(1)**2*(BETA-COS(BETA)*SIN(BETA))
      P=X(1)*BETA
      ELSE
      A=(X(1)+X(2)*X(7))*X(7)
      P=X(1)+2.*X(7)*SQRT(X(2)**2+1.)
```

```
      ENDIF
      Rh=A/P
      F(1)=X(5)-X(8)*A*SQRT(X(3)*RH)
      F(2)=X(8)+G32*ALOG10(X(4)/(12.*Rh)+CG*X(8)*P/X(5))
      F(3)=X(6)-X(7)-FKE*(X(5)/A)**2
      RETURN
      END
```

Program UENCHEZ.C

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float x[9],g,fke,g32,cg; int itype;
extern void solveq(int n,float **d,float *f,int itype,\
  float *dd,int *indx);
void fun(float *f){float beta,a,p,rh;
 if(itype){beta=acos(1.-2.*x[6]/x[0]);
   a=.25*x[0]*x[0]*(beta-cos(beta)*sin(beta));p=x[0]*beta;}
 else {a=(x[0]+x[1]*x[6])*x[6];p=x[0]+2.*x[6]*sqrt(x[1]*x[1]+1.);}
   rh=a/p;
 f[0]=x[4]-x[7]*a*sqrt(x[2]*rh);
 f[1]=x[7]+g32*log10(x[3]/(12.*rh)+cg*x[7]*p/x[4]);
 f[2]=x[5]-x[6]-fke*pow(x[4]/a,2.);return;} //End of fun
void main(void){char *fmt=" %c =%9.3f\n",*ch[]={"value","guess"},\
  *cx="bmSeQHYCK";
 float f[3],f1[3],xx,sum,**d,*dd,vis;
 int id[9],indx[3],i,j,i1,i2,nct;
 d=(float**)malloc(3*sizeof(float*));
 for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
 printf("Give: g,k.viscosity,entrance loss C. & 0=trap or 1=cir\n");
 scanf("%f %f %f %d",&g,&vis,&fke,&itype);
 if(itype){stpcpy(cx[0],"D");stpcpy(cx[1]," ");}
 if(g<15.) x[7]=100.; else x[7]=60.; x[8]=fke;
 fke=(1.+fke)/(2.*g);g32=sqrt(32.*g);cg=.884*vis/(4.*sqrt(g));
L1: for(i=0;i<7;i++){if((!itype) || ((itype)&&(i != 1)))
  printf("%2d %c\n",i+1,cx[i]);id[i]=0;}
 do{printf(" Give two numbers for 2 unknown variables\n");
   scanf("%d %d",&i1,&i2);
 }while((i1<1)||(i1>7)||(i2<1)||(i2>7)); id[i1-1]=1;id[i2-1]=1;
 for(i=0;i<7;i++){if((!itype) || ((itype)&&(i != 1)))
   {printf("Give %s for %c =",ch[id[i]],cx[i]);
   scanf("%f",&x[i]);}} nct=0;
do{sum=0.; fun(f); i1=-1;for(i=0;i<8;i++){if(id[i]){xx=x[i];i1++;
  x[i]*=1.005;fun(f1);
  for(j=0;j<3;j++) d[j][i1]=(f1[j]-f[j])/(x[i]-xx); x[i]=xx;}}
  solveq(3,d,f,1,dd,indx); i1=-1;
  for(i=0;i<8;i++){if(id[i]){sum+=fabs(f[++i1]);x[i]-=f[i1];}}
  printf("nct= %d SUM=%f\n",nct,sum);
  }while((++nct<30)&&(sum>1.e-5));
printf("Solution:\n");
for(i=0;i<9;i++){if((!itype) || ((itype)&&(i != 1))){
if((i==2)||(i==3))stpcpy(fmt[8],"6"); else stpcpy(fmt[8],"3");
printf(fmt,cx[i],x[i]);}}
printf("Give 1 to solve another prob. (0=STOP)\n");
scanf("%d",&i2); if(i2) goto L1;}
```

Using this program to solve for the flow rate Q and the depth Y gives Q = 164.18 cfs, Y = 4.67 ft, and C = 109.32 for the rectangular channel, and Q = 368.70 cfs, Y = 4.52 ft, and C = 113.45 for the trapezoidal channel. When part 4 is solved for the bottom width b = 11.14 ft, Y = 4.51 ft, and C = 113.77.

Program UENCHEZ1.FOR

```
C Solves Chezys and Specific Energy for Y & Q by substituting
  Chezys Eq.
C into Energy and first solving for Y and thereafter Q.  The
  needed Chezy's
C C is solved using Gauss_Seidel iteration (This is the
  procedure
C to use with a pocket calculator)  See program E_UN to solve
  any 2 variables.
      COMMON B,FM,D,H,CC,FMS,C,ITYPE
      WRITE(*,*)' Give: g,viscosity,(0=trap or 1=cir)'
      READ(*,*) G,VIS,ITYPE
      IF(ITYPE.EQ.0) THEN
      WRITE(*,*)' Give: b,m,e,So,H,Ke'
      READ(*,*) B,FM,e,SO,H,FKE
      FMS=2.*SQRT(FM*FM+1.)
      ELSE
      WRITE(*,*)' Give: D,e,So,H,Ke'
      READ(*,*) D,e,SO,H,FKE
      ENDIF
      G32=-SQRT(32.*G)
      CG=.884*VIS/(4.*SQRT(G))
      CC=SO*(1.+FKE)/(2.*G)
      IF(G.GT.15.) THEN
      C=100.
      ELSE
      C=60.
      ENDIF
      Y=.8*H
      NCT=0
10    F=FUN(Y)
      DIF=.05*F/(FUN(Y+.05)-F)
      Y=Y-DIF
      CALL AREAP(Y,A,P)
      Rh=A/P
      Q=C*A*SQRT(Rh*SO)
      MCT=0
12    C1=C
      C=G32*ALOG10(e/(12.*Rh)+CG*C1*P/Q)
      MCT=MCT+1
      IF(ABS(C-C1).GT. 5.E-6 .AND. MCT.LT.30) GO TO 12
      IF(MCT.EQ.30) WRITE(*,*)' Did not converge',C,C1
      NCT=NCT+1
      WRITE(*,*) NCT,DIF,Y,Q
      IF(NCT.LT.30 .AND. ABS(DIF).GT. 1.E-5) GO TO 10
      WRITE(*,100) Y,Q,C
100   FORMAT(' Solution: Y =',F8.3,' Q =',F10.2,' C=',F9.2)
      END
      SUBROUTINE AREAP(Y,A,P)
      COMMON B,FM,D,H,CC,FMS,C,ITYPE
      IF(ITYPE.EQ.1) THEN
```

```
      COSB=1.-2.*Y/D
      BETA=ACOS(COSB)
      P=D*BETA
      A=.25*D*D*(BETA-COSB*SIN(BETA))
      ELSE
      P=B+FMS*Y
      A=(B+FM*Y)*Y
      ENDIF
      RETURN
      END
      FUNCTION FUN(Y)
      COMMON B,FM,D,H,CC,FMS,C,ITYPE
      CALL AREAP(Y,A,P)
      FUN=H-Y-CC*(A/P)*C*C
      RETURN
      END
```

Program UENCHEZ1.C

```
/* Solves Chezys and Specific Energy for Y & Q by substituting
   Chezys Eq.into Energy and first solving for Y and thereafter
   Q. The needed Chezy's C is solved using Gauss_Seidel
   iteration (This is the procedure to use with a pocket
   calculator) See program E_UN to solve any 2 variables. */
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float b,m,d,h,cc,fms,c; int itype;
void areap(float y,float *a,float *p){float cosb,beta;
   if(itype){cosb=1.-2.*y/d;beta=acos(cosb);*p=d*beta;
      *a=.25*d*d*(beta-cosb*sin(beta));}
   else {*p=b+fms*y; *a=(b+m*y)*y;return;}}
float fun(float y){float *a,*p; areap(y,a,p);
   return(h-y-cc*((*a)/(*p))*c*c);}
void main(void){float g,dif,ke,y,f,q,e,so,vis,g32,cg,rh,c1,*a,*p;
   int mct,nct=0; printf("Give: g,viscosity,(0=trap or 1=cir)\n");
   scanf("%f %f %d",&g,&vis,&itype);
   if(itype){printf("Give: D,e,So,H,Ke\n");
      scanf("%f %f %f %f %f",&d,&e,&so,&h,&ke);}
   else {printf("Give: b,m,e,So,H,Ke\n");
      scanf("%f %f %f %f %f %f",&b,&m,&e,&so,&h,&ke);
      fms=2.*sqrt(m*m+1.);}
   if(g>15.) c=100.; else c=70.; g32=-sqrt(32.*g);
   cg=.884*vis/(4.*sqrt(g)); cc=so*(1.+ke)/(2.*g); y=.8*h;
   do{f=fun(y); dif=.05*f/(fun(y+.05)-f); y-=dif;
      printf("%d %f %f\n",++nct,dif,y);areap(y,a,p); rh=(*a)/(*p);
      q=c*(*a)*sqrt(rh*so); mct=0;
      do{c1=c;c=g32*log10(e/(12.*rh)+cg*c1*(*p)/q);
      }while((fabs(c-c1)>5.e-6) && (++mct<30));
   }while((nct<30) && (fabs(dif)>1.e-5));
   printf("Solution: Y = %8.3f Q = %10.2f C= %9.2f\n",y,q,c);
}
```

EXAMPLE PROBLEM 2.13

Rather than having a trapezoidal channel as in the previous example problem the channel is circular with a diameter D = 10 ft. Its Manning's n = 0.014 and its bottom slope is $S_o = 0.00075$ as in the previous problem. ($K_e = 0.1$) Find the flow rate and depth. As a second part assume this circular channel has a steep slope. Now solve for Q and Y. As a third part the channel is a steep trapezoidal channel with b = 10 ft, and m = 1.2. Now solve for Q and Y. As a fourth part find the size of steep circular channel requires to convey a flow rate Q = 300 cfs. For all part the reservoir head is H = 5 ft and the entrance loss coefficient is $K_e = 0.1$.

Solution

For the first part either the program E_UN or E_UN1, given in the previous problem, can be used to solve the problem identifying the type of channel as circular. The solve gives: Q = 178.58 cfs and Y = 4.549 ft. You should verify these results with you calculator, using TK-Solver and/or Mathcad.

For the second part in which the channel is steep the critical flow equation $F_1 = Q^2T - gA^3 = 0$ replaces the uniform flow equation of the previous example problem. However, when the unknowns are Q and Y, then Q can be solve from this equation and substituted into the energy equation to give the single equation $F = H - Y - (1 + K_e)A/(2T) = 0$ to solve for Y and thereafter solve the critical flow equation for Q. The program E_CR1 uses this later approach to solve for Y and then Q, and the program E_CR solves the critical and energy equations simultaneously. The answers to second part of the problem are: Y = 3.56 ft and Q = 230.20 cfs. For the third part of the problem in which the steep channel has a trapezoidal cross section the answers are: Y = 3.51 ft and Q = 474.37 cfs. For the fourth part of the problem in which the diameter is want that will convey Q = 300 cfs, the two equations must be solved simultaneously for Y and D (or program U-CR must be used.) The answers are Y = 3.60 ft and D = 15.413.

Program E_CR1.FOR

```
C Solves Critical and Specific Energy for Y & Q by substituting
C Critical Eq. into Energy and first solving for Y and
  thereafter Q.
C  This is the procedure to use with a pocket calculator) See
     program E_CR to solve any 2 variables.
      COMMON B,FM,D,H,C,ITYPE
      WRITE(*,*)' Give: g,(0=trap or 1=cir)'
      READ(*,*) G,ITYPE
      IF(ITYPE.EQ.0) THEN
      WRITE(*,*)' Give: b,m,H,Ke'
      READ(*,*) B,FM,H,FKE
      ELSE
      WRITE(*,*)' Give: D,H,Ke'
      READ(*,*) D,H,FKE
      ENDIF
      C=.5*(1.+FKE)
      Y=.7*H
      NCT=0
10    F=FUN(Y)
      DIF=.05*F/(FUN(Y+.05)-F)
      Y=Y-DIF
      NCT=NCT+1
      WRITE(*,*) NCT,DIF,Y
      IF(NCT.LT.30 .AND. ABS(DIF).GT. 1.E-5) GO TO 10
      CALL AREAP(Y,A,T)
      Q=SQRT(G*A**3/T)
      WRITE(*,100) Y,Q
100   FORMAT(' Solution: Y =',F8.3,' Q =',F10.2)
      END
      SUBROUTINE AREAP(Y,A,T)
      COMMON B,FM,D,H,C,ITYPE
      IF(ITYPE.EQ.1) THEN
      COSB=1.-2.*Y/D
```

```
      BETA=ACOS(COSB)
      T=D*SIN(BETA)
      A=.25*D*D*(BETA-COSB*SIN(BETA))
      ELSE
      T=B+2.*Y*FM
      A=(B+FM*Y)*Y
      ENDIF
      RETURN
      END
      FUNCTION FUN(Y)
      COMMON B,FM,D,H,C,ITYPE
      CALL AREAP(Y,A,T)
      FUN=H-Y-A*C/T
      RETURN
      END
```

Program E_CR1.C

```
/* Solves Critical and Specific Energy for Y & Q by substituting
   Critical Eq.into Energy and first solving for Y and thereafter
   Q. (This is the procedure to use with a pocket calculator) See
   program E_CR to solve any 2 variables. */
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float b,m,d,h,c; int itype;
void areap(float y,float *a,float *t){float cosb,beta;
 if(itype){cosb=1.-2.*y/d;beta=acos(cosb);*t=d*sin(beta);
   *a=.25*d*d*(beta-cosb*sin(beta));}
 else {*t=b+2.*y*m; *a=(b+m*y)*y;return;}}
float fun(float y){float *a,*t; areap(y,a,t);
  return(h-y-c*(*a)/(*t));}
void main(void){float g,dif,ke,y,f,q,*a,*t; int nct=0;
 printf("Give: g(0=trap or 1=cir)\n"); scanf("%f %d",&g,&itype);
 if(itype){printf("Give: D,H,Ke\n"); scanf("%f %f %f",&d,&h,&ke);}
 else {printf("Give: b,m,H,Ke\n");
   scanf("%f %f %f %f",&b,&m,&h,&ke);}
 y=.7*h; c=.5*(1.+ke);
 do{f=fun(y); dif=.05*f/(fun(y+.05)-f);
  y-=dif;printf("%d %f %f\n",++nct,dif,y);
 }while((nct<30) && (fabs(dif)>1.e-5));
 areap(y,a,t); q=sqrt(g*pow((*a),3.)/(*t));
 printf("Solution: Y = %8.3f Q = %10.2f\n",y,q);
}
```

Program E_CR.FOR

```
C Solves critical flow and Energy simultaneously
C for any 2 unknowns
      CHARACTER*17 FMT/'(1X,A1,'' ='',F9.3)'/
      CHARACTER*1 CX(6)/'b','m','Q','H','Y','K'/
      CHARACTER*5 CH(0:1)/'value','guess'/
      REAL F(2),F1(2),D(2,2)
      INTEGER*2 ID(6),INDX(2)
      COMMON X(6),G,FKE,ITYPE
      WRITE(*,*)' Give: g,entrance loss C. & 0=TRAP',' or 1=CIRLCE'
      READ(*,*) G,FKE,ITYPE
      IF(ITYPE.EQ.1) THEN
```

```
      CX(1)='D'
      CX(2)=' '
      ENDIF
      X(6)=FKE
      FKE=(1.+FKE)/(2.*G)
1     DO 10 I=1,5
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 10
      WRITE(*,'(I2,2X,A1)') I,CX(I)
10    ID(I)=0
2     WRITE(*,*)' Give two numbers for 2 unknown variables'
      READ(*,*) I1,I2
      IF(I1.LT.1.OR.I1.GT.7.OR.I2.LT.1.OR.I2.GT.7) GO TO 2
      ID(I1)=1
      ID(I2)=1
      DO 20 I=1,5
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 20
      WRITE(*,100) CH(ID(I)),CX(I)
100   FORMAT(' Give ',A5,' for ",A1,' = ',\)
      READ(*,*) X(I)
20    CONTINUE
      NCT=0
30    SUM=0.
      CALL FUN(F)
      I1=0
      DO 40 I=1,5
      IF(ID(I).EQ.0) GO TO 40
      I1=I1+1
      X(I)=1.005*X(I)
      XX=X(I)
      CALL FUN(F1)
      DO 35 J=1,2
35    D(J,I1)=(F1(J)-F(J))/(X(I)-XX)
      X(I)=XX
40    CONTINUE
      CALL SOLVEQ(2,1,2,D,F,1,DD,INDX)
      I1=0
      DO 50 I=1,5
      IF(ID(I).EQ.0) GO TO 50
      I1=I1+1
      SUM=SUM+ABS(F(I1))
      X(I)=X(I)-F(I1)
50    CONTINUE
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,' SUM=',SUM
      IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 30
      WRITE(*,*)' Solution:'
      DO 60 I=1,6
      IF(ITYPE.EQ.1 .AND. I.EQ.2) GO TO 60
      WRITE(*,FMT) CX(I),X(I)
60    CONTINUE
      WRITE(*,*)' Give 1 to solve another prob. (0=STOP)'
      READ(*,*) I2
      IF(I2.EQ.1) GO TO 1
      END
      SUBROUTINE FUN(F)
      REAL F(2)
```

```
      COMMON X(6),G,FKE,ITYPE
      IF(ITYPE.EQ.1) THEN
      BETA=ACOS(1.-2.*X(5)/X(1))
      A=.25*X(1)**2*(BETA-COS(BETA)*SIN(BETA))
      T=X(1)*SIN(BETA)
      ELSE
      A=(X(1)+X(2)*X(5))*X(5)
      T=X(1)+2.*X(2)*X(5)
      ENDIF
      F(1)=X(3)**2*T-G*A**3
      F(2)=X(4)-X(5)-FKE*(X(3)/A)**2
      RETURN
      END
```

Program E_CR.C

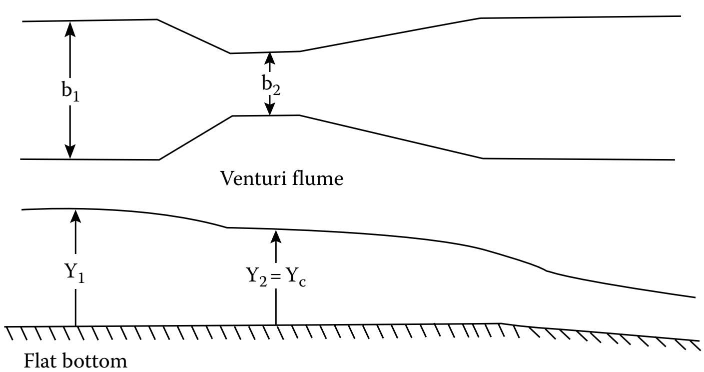
```
// Solves critical flow and Energy simultaneously for any 2
// unknowns
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float x[6],g,fke; int itype;
extern void solveq(int n,float **d,float *f,int itype,\
  float *dd,int *indx);
void fun(float *f){float beta,a,p,t;
 if(itype){beta=acos(1.-2.*x[4]/x[0]);
   a=.25*x[0]*x[0]*(beta-cos(beta)*sin(beta));
   t=x[0]*sin(beta);}
 else {a=(x[0]+x[1]*x[4])*x[4];t=x[0]+2.*x[1]*x[4];}
 f[0]=x[2]*x[2]*t-g*pow(a,3.);
 f[1]=x[3]-x[4]-fke*pow(x[2]/a,2.);return;} //End of fun
void main(void){
  char *fmt=" %c =%9.3f\n",*ch[]={"value","guess"},*cx="bmQHYK";
 float f[2],f1[2],xx,sum,**d,*dd; int id[6],indx[2],i,j,i1,i2,nct;
 d=(float**)malloc(2*sizeof(float*));
 for(i=0;i<2;i++)d[i]=(float*)malloc(2*sizeof(float));
 printf("Give: g,entrance loss C. & 0=TRAP or 1=CIRLCE\n");
 scanf("%f %f %d",&g,&fke,&itype);
 if(itype){stpcpy(cx[0],"D");stpcpy(cx[1]," ");}
 x[5]=fke;fke=(1.+fke)/(2.*g);
L1: for(i=0;i<5;i++){if((!itype) || ((itype)&&(i != 1)))\
  printf("%2d %c\n",i+1,cx[i]);id[i]=0;}
 do{printf(" Give two numbers for 2 unknown variables\n");
   scanf("%d %d",&i1,&i2);
  }while((i1<1)||(i1>7)||(i2<1)||(i2>7)); id[i1-1]=1;id[i2-1]=1;
 for(i=0;i<5;i++){if((!itype) || ((itype)&&(i != 1)))\
   {printf("Give %s for %c =",ch[id[i]],cx[i]);
   scanf("%f",&x[i]);}} nct=0;
 do{sum=0.; fun(f); i1=-1;
  for(i=0;i<5;i++){if(id[i]){xx=x[i];i1++;x[i]*=1.005;fun(f1);
  for(j=0;j<2;j++) d[j][i1]=(f1[j]-f[j])/(x[i]-xx); x[i]=xx;}}
  solveq(2,d,f,1,dd,indx); i1=-1;
  for(i=0;i<5;i++){if(id[i]){sum+=fabs(f[++i1]);x[i]-=f[i1];}}
   printf("nct= %d SUM=%f\n",nct,sum);
   }while((++nct<30)&&(sum>1.e-5));
 printf("Solution:\n");
```

```
for(i=0;i<6;i++){if((!itype) || ((itype)&&(i != 1))){
 if(i==2)stpcpy(fmt[8],"6"); else stpcpy(fmt[8],"3");
 printf(fmt,cx[i],x[i]);}}
printf("Give 1 to solve another prob. (0=STOP)\n");
scanf("%d",&i2); if(i2) goto L1;}
```

2.7 FLUMES

Flow rate measurements in open channels can be made by devices called flumes, which reduce the size of a throat section by reducing its width and/or raising its bottom. If the channel downstream from where the device is installed does not back the water up, so "free flow conditions" exist, then critical flow occurs within the throat length. To help assure that critical flow occurs the section diverges downstream from the throat and the bottom may have a steep slope through the throat or possibly downstream therefrom. Such devices are called "critical flow" meters because their means of measuring the flow is based on causing the depth to be critical in the throat section, or the raised section. If only a hump in the bottom occurs the device is referred to as a broad-crested weir. Chapter 5 gives more information about open channel flow measurement devices and how they are constructed. Using the theory associated with critical flow to determine how the flow rate is determined by a measured depth in such devices is covered here.

First let us deal with the simple case of a flat bottom device with contracting sides in a rectangular channel upstream from the device and diverging walls downstream therefrom to a larger, or steeper sloping downstream channel that can readily convey the water away from the device, so critical flow occurs within the throat as shown in the sketch below.



With the depth critical within the throat the flow rate Q equals unit flow rate q_2 times the throat width b_2. And from Equation 2.12 $q_c = q_2 = (gY_c^3)^{1/2}$, so

$$Q = b_2\sqrt{gY_c^3} = \left(\frac{2}{3}\right)^{1.5}\sqrt{g}b_2E_c^{1.5} \tag{2.12e}$$

(The number 12 is used as a reminder that the equation is restricted to rectangular channels.)

The latter of which comes from Equation 2.12c $Y_c = 2E_c/3$. Because the depth at section 2 is less stable with the larger velocities here, the upstream depth Y_1 is generally measured rather than $Y_2 = Y_c$. So the typical flow equation equates $E_1 = Y_1 + Q^2/(2gb_1^2Y_1^2)$ to $E_c = E_2$, and the flow rate equation becomes the following implicit equation that gives Q as a function of the upstream depth Y_1:

$$Q = \left(\frac{2}{3}\right)^{1.5} \sqrt{g} b_2 \left\{ Y_1 + \frac{Q^Z}{2gb_1^Z Y_1^Z} \right\}^{1.5} \tag{2.12f}$$

Rather than solve this implicit equation practice replaces E_c by Y_1 and introduces a coefficient, C_v, called the velocity coefficient, since it accounts for the upstream velocity head and approaches unity as the upstream velocity head becomes negligible, or

$$Q = C_v \left(\frac{2}{3}\right)^{1.5} \sqrt{g} b_2 Y_1^{1.5} \tag{2.12g}$$

To obtain an equation that gives C_v let's relate Y_1 to the critical depth at section 2, or

$$Y_1 + \frac{Q^2}{(2gb_1^2 Y_1^2)} = 1.5Y_c \tag{2.12h}$$

By dividing by Y_c and defining the dimensionless upstream depth $Y_1' = Y_1/Y_c$ we can, with a little algebraic manipulation, obtain the following cubic equation relating Y_1' to the ratio of the throat to the upstream width b_2/b_1.

$$2Y_1'^3 - 3Y_1'^2 + \left(\frac{b_2}{b_1}\right)^2 = 0 \tag{2.12i}$$

The table below gives the three roots to this equation for several width ratios.

Table Giving Roots of Equation 2.12i for Several Width Ratios

b_2/b_1	0.95	0.90	0.85	0.80	0.75	0.70	0.65	0.60
root 1	1.1708	1.2341	1.2793	1.3149	1.3444	1.3693	1.3908	1.4094
root 2	0.8069	0.7211	0.6531	0.5945	0.5418	0.4933	0.4481	0.4055
root 3	−.4777	−.4551	−.4324	−.4094	−.3862	−.3627	−.3389	−.3149

	0.55	0.50	0.45	0.40	0.35	0.30	0.25
	1.4256	1.4397	1.4520	1.4626	1.4717	1.4794	1.4910
	0.3651	0.3264	0.2892	0.2533	0.2186	0.1850	0.1523
	−.2906	−.2660	−.2411	−.2159	−.1904	−.1644	−.1381

Of the three roots of this equation only the larger of the two positive roots occurs in practice because the negative root is physically impossible, and the smaller of the positive roots is associated with supercritical flow upstream. The largest root, on the other hand, allows us to conclude that for any flow rate in a rectangular flume with a flat bottom there is a fixed upstream depth Y_1 associated with any Q. To find this Y_1 first solve $Y_c = \{q^2/g\}^{1/3} = \{(Q^2/b^2)/g\}^{1/3}$ (Equation 2.12). Next solve Equation 2.12i for Y_1', and finally compute $Y_1 = Y_1'Y_c$. In order for the flume to make its measurement properly (i.e., have critical flow at its throat) the depth of flow (which is often normal depth) at the position where the flume is to be installed must be equal to or less than this value of Y_1. If it is larger than Y_1 then the flow through the flume's throat will be subcritical. For example, assume a venturi flume with a width ratio of 0.8 is to be installed in a 5 m wide rectangular channel with n = 0.013 to measure a maximum flow rate Q = 40 m³/s. What is the maximum bottom slope this channel can have for the flume to work properly? From the above table $Y_1' = 1.3149$ corresponding to $b_2/b_1 = 0.8$, and $b_2 = 0.8(5) = 4$ m, so $q_2 = 40/4 = 10$ m²/s and $Y_c = 2.168$ m. Thus the maximum normal depth is $Y_1 = Y_1'Y_c = 2.851$ m. Solving Manning's equation we get a bottom slope of $S_o = 0.000908$. If the bottom slope were 0.0008 for example then its normal depth would be $Y_{o1} = 2.992$ with $E_{o1} = 3.357$, and setting $E_2 = 3.357$ and solving for $Y_2 = 2.607$. The associated Froude number is 0.758, i.e., the flow is not critical at section 2. Let us assume the flow rate is Q = 20 m³/s. Then the normal depth, with $S_o = 0.000908$ is $Y_{o1} = 1.709$ m and $E_{o1} = 1.988$ m. The critical depth is 1.366 m and $E_c = 2.049$ m. Thus the flume will cause the depth upstream to be larger than Y_{o1} so $E_1 = E_c = 2.049$ m.

Also Equation 2.12i can be used to get an equation for the velocity coefficient C_v. By dividing Equation 2.12g by Equation 2.12e we get $1 = C_v(Y_1/E_1)^{1.5}$, or $Y_1/E_1 = C_v^{-2/3}$, but $Y_1' = Y_1/Y_c = 1.5Y_1/E_1$, so $Y_1' = 1.5C_v^{-2/3}$, which when substituted into Equation 2.12i gives

$$\left(\frac{4}{27}\right)\left(\frac{b_2}{b_1}\right)^2 C_v^2 - C_v^{2/3} + 1 = 0 \qquad (2.12j)$$

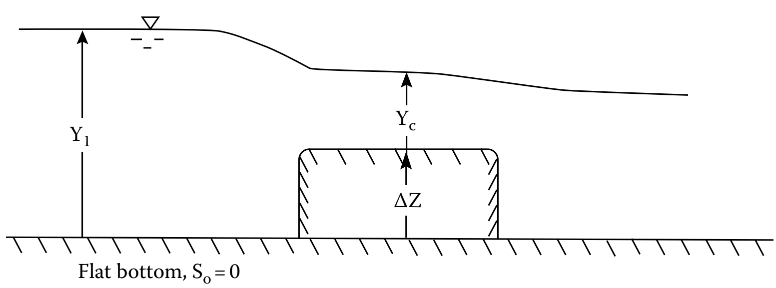
Alas! Another implicit equation, but you can readily let your pocket calculator solve it for you. The table below provides solutions to C_v for several width ratios b_2/b_1.

Table That Provides the Solution of the Velocity Coefficient C_v from Equation 2.12j for Several Width Ratios

b_2/b_1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
C_v	1.002233	1.009064	1.020918	1.038597	1.063487	1.097979	1.146490	1.218388	1.340083	1.835955

Thus you can choose whether you solve Equation 2.12f for Q given an upstream depth Y_1, or solve Equation 2.12j and substitute the C_v obtained therefrom into Equation 2.12 g to solve Q. A final option is to solve the energy equation $E_1 = E_{2c} + \Delta z$ and the critical flow equation $Q^2T^5/(2gA_2^3) = 1$ simultaneously for Y_c and Q. If the channel at either position 1 or 2 is not rectangular, the bottom is not flat, and Y_1 is known then only this later option exists.

A broad crested weir measures the flow rate by having a hump, with a height Δz, in the bottom of a rectangular channel, which is high enough to cause critical flow over it as shown in the sketch below.



Equation 2.12e applies equally to this device as it does to the venturi flume. Now, however, $E_c = E_1 - \Delta z$, and therefore the implicit equation that gives Q is as follows, rather than Equation 2.12f:

$$Q = \left(\frac{2}{3}\right)^{1.5}\sqrt{g}b_2\left\{Y_1 - \Delta_Z + \frac{Q^2}{2gA_1^2}\right\}^{1.5} \quad (2.12k)$$

Following the same procedure used to obtain the dimensionless cubic Equation 2.12i, that gives the dimensionless upstream depth $Y_1' = Y_1/Y_c$ as a function of the width ratio yields

$$2Y_1'^3 - (3+2\Delta_Z)Y_1'^2 + \left(\frac{b_2}{b_1}\right)^2 = 0 \quad (2.12l)$$

Equation 2.12l also assumes that the channel is rectangular upstream from the broad crested weir, but Equation 2.12k applies for a non rectangular upstream channel whose cross-sectional area is A_1. (Likewise Equation 2.12f could be used for situations in which the channel is not rectangular if A_1 replaces b_1Y_1 in it.)

It is not practical, however, to introduce a velocity coefficient into Equation 2.12k and thereafter develop an equation for C_v. The difference between Equations 2.12l and 2.12i is that now Y_1' depends upon two parameters, the width ratio b_2/b_1 and $(3 + 2\Delta z/Y_c)$. Thus the upstream dimensionless depth Y_1' is fixed by b_2/b_1 but this value will be different for each different height of weir divided by the critical depth over the weir $\Delta z/Y_c$, and since Y_c varies with Q therefore Y_1' varies with b_2/b_1, Δz, and Q.

2.8 DELIVERY DIAGRAMS

A delivery diagram is a graph that gives the flow rate Q that will occur in a given channel as a function of the head H of the reservoir that supplies it. Often only one channel is involved and this channel is long so that until critical flow occurs at its entrance the flow will be uniform throughout its length. Under these conditions the data for Q versus H to plot the delivery diagram can be obtained by solving the energy and uniform flow equations simultaneously, as has been done in the previous example problems, for a series of H values until the Froude number becomes unity, and thereafter solve the energy and critical flow equations. For a given channel (its size variables, roughness, and bottom slope do not change with depth and flow rate), the Froude number associated with the flow will increase as the head H of the supply reservoir increases. Thus in obtaining the series of solutions for the delivery diagram it is best to start with the smallest reservoir head H that is anticipated and increase with a specified increment toward the largest H anticipated.

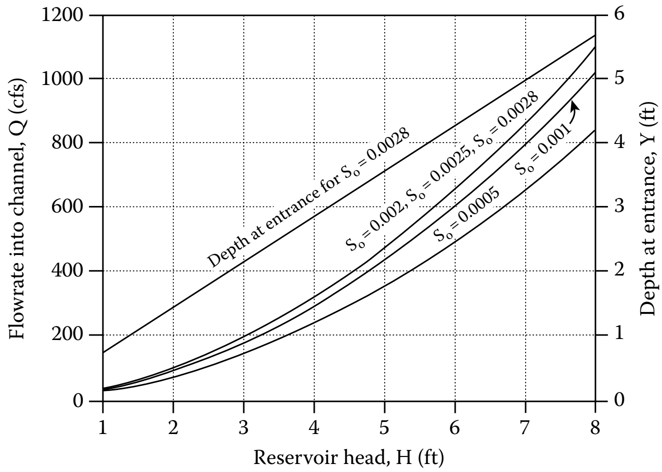
To illustrate how delivery diagrams are created let's use a trapezoidal channel with b = 10 ft, m = 1.2, n = 0.014. Delivery diagrams will be created for this channel for bottom slopes varying from $S_o = 0.0005$ to $S_o = 0.0028$ to demonstrate that as the bottom slope of the channel increases and H increases that critical flow at the entrance can limit the flow rate that the channel will carry. The table below provides the solution for the depth Y and the flow rate Q from the simultaneous solution of the Energy and Manning's equations for the reservoir head H varying from 1 to 8 ft, for five different bottom slopes. The Froude number squared has been added as a third column to each of these solutions. The last two columns in this table contain the critical depth Y_c and the critical flow rate Q_c, i.e., the simultaneous solution of the energy and critical flow equations associated with the same H values. Notice that for bottom slopes of 0.0005, 0.001, and 0.002 that all the Froude numbers are less than one. Thus all these Q versus H data do represent the delivery diagram. However with bottom slopes of 0.0025 and 0.0028 the Froude

Table that Solves the Energy and Mannings Equations Simultaneously for Varying Reservoir Heads H and Bottom Slopes, and also the Energy and Critical Flow Equations for Y_c and Q_c for these Same H's for a Trapezoidal Channel with b = 10 ft, m = 1.2, n = .014 and an Entrance Loss Coefficient K_e = 0.1

	$S_o = 0.0005$			$S_o = 0.001$			$S_o = 0.002$			$S_o = 0.0025$			$S_o = 0.0028$			Critical	
H	Y	Q	F_r^2	Y	Q	F_r^2	Y	Q	F_r^2	Y	Q	F_r^2	Y	Q	F_r^2	Y_c	Q_c
1.0	.929	21.0	0.15	0.868	26.5	0.30	0.771	30.8	0.58	0.731	31.5	0.72	0.709	31.6	0.80	0.661	31.7
1.2	1.112	28.5	0.16	1.037	35.9	0.31	0.918	41.3	0.61	0.869	42.1	0.75	0.843	42.3	0.83	0.797	42.3
1.4	1.295	36.9	0.17	1.207	46.3	0.33	1.065	53.0	0.63	1.007	54.0	0.78	0.976	54.1	0.87	0.934	54.1
1.6	1.479	46.3	0.17	1.376	57.9	0.34	1.212	65.9	0.65	1.145	67.0	0.80	1.109	67.1	0.89	1.071	67.1
1.8	1.662	56.5	0.18	1.545	70.5	0.34	1.358	80.1	0.67	1.282	81.2	0.83	1.241	81.3	0.92	1.210	81.2
2.0	1.845	67.6	0.18	1.714	84.3	0.35	1.505	95.3	0.68	1.420	96.5	0.84	1.374	96.6	0.94	1.349	96.4
2.2	2.028	79.7	0.18	1.883	99.1	0.36	1.651	111.8	0.70	1.557	113.0	0.86	1.506	112.9	0.96	1.489	112.9
2.4	2.212	92.6	0.19	2.052	115.0	0.37	1.798	129.3	0.71	1.694	130.6	0.88	1.638	130.5	0.98	1.630	130.4
2.6	2.395	106.4	0.19	2.221	132.0	0.37	1.944	148.1	0.72	1.831	149.3	0.89	1.770	149.1	1.00	1.771	149.1
2.8	2.578	121.2	0.19	2.391	150.1	0.38	2.091	167.9	0.74	1.969	169.2	0.91	1.903	168.8	1.01	1.913	168.9
3.0	2.762	136.8	0.19	2.560	169.2	0.38	2.237	189.0	0.75	2.106	190.2	0.92	2.035	189.6	1.02	2.056	189.9
3.2	2.945	153.4	0.20	2.729	189.5	0.39	2.384	211.2	0.76	2.244	212.3	0.93	2.168	211.6	1.04	2.199	212.1
3.4	3.129	170.9	0.20	2.899	210.9	0.39	2.531	234.5	0.76	2.381	235.6	0.94	2.300	234.7	1.05	2.343	235.4
3.6	3.312	189.3	0.20	3.068	233.4	0.40	2.677	259.1	0.77	2.519	260.0	0.95	2.433	258.9	1.06	2.488	259.8
3.8	3.496	208.7	0.20	3.238	257.0	0.40	2.824	284.8	0.78	2.656	285.6	0.97	2.565	284.2	1.07	2.632	285.4
4.0	3.679	229.0	0.21	3.407	281.8	0.41	2.971	311.7	0.79	2.794	312.4	0.97	2.698	310.7	1.08	2.778	312.2

4.2	3.863	250.3	0.21	3.577	307.7	0.41	3.118	339.8	0.80	2.932	340.3	0.98	2.831	338.3	1.09	2.924	340.2
4.4	4.046	272.5	0.21	3.746	334.7	0.41	3.265	369.1	0.80	3.070	369.4	0.99	2.964	367.1	1.10	3.070	369.4
4.6	4.230	295.8	0.21	3.916	362.9	0.42	3.412	399.6	0.81	3.208	399.6	1.00	3.097	397.0	1.11	3.216	399.8
4.8	4.413	320.0	0.21	4.086	392.3	0.42	3.560	431.3	0.82	3.346	431.1	1.01	3.230	428.1	1.12	3.363	431.5
5.0	4.597	345.2	0.21	4.255	422.9	0.42	3.707	464.3	0.82	3.484	463.8	1.02	3.363	460.4	1.13	3.511	464.3
5.2	4.781	371.4	0.22	4.425	454.7	0.43	3.854	498.5	0.83	3.622	497.6	1.02	3.496	493.9	1.14	3.658	498.4
5.4	4.964	398.6	0.22	4.594	487.6	0.43	4.001	534.0	0.84	3.760	532.7	1.03	3.629	528.5	1.15	3.806	533.7
5.6	5.148	426.9	0.22	4.764	521.9	0.43	4.149	570.7	0.84	3.898	569.0	1.04	3.762	564.3	1.16	3.955	570.3
5.8	5.331	456.2	0.22	4.934	557.3	0.43	4.296	608.7	0.85	4.036	606.6	1.05	3.895	601.4	1.16	4.103	608.2
6.0	5.515	486.6	0.22	5.103	594.0	0.44	4.443	647.9	0.85	4.174	645.3	1.05	4.028	639.7	1.17	4.252	647.3
6.2	5.698	518.0	0.22	5.273	631.9	0.44	4.590	688.5	0.86	4.312	685.4	1.06	4.161	679.1	1.18	4.401	687.8
6.4	5.882	550.5	0.22	5.442	671.1	0.44	4.738	730.3	0.86	4.450	726.7	1.07	4.295	719.8	1.19	4.550	729.5
6.6	6.065	584.1	0.23	5.612	711.5	0.45	4.885	773.5	0.87	4.589	769.2	1.07	4.428	761.8	1.19	4.700	772.5
6.8	6.249	618.8	0.23	5.781	753.3	0.45	5.032	818.0	0.87	4.727	813.1	1.08	4.561	805.0	1.20	4.850	816.9
7.0	6.432	654.6	0.23	5.951	796.3	0.45	5.179	863.8	0.88	4.865	858.2	1.08	4.694	849.4	1.21	5.000	862.6
7.2	6.615	691.5	0.23	6.120	840.7	0.45	5.326	910.9	0.88	5.003	904.6	1.09	4.827	895.2	1.21	5.150	909.6
7.4	6.799	729.5	0.23	6.290	886.4	0.46	5.474	959.4	0.89	5.141	952.3	1.10	4.961	942.2	1.22	5.301	958.0
7.6	6.982	768.7	0.23	6.459	933.4	0.46	5.621	1009.3	0.89	5.279	1001.4	1.10	5.094	990.4	1.22	5.451	1007.7
7.8	7.166	809.0	0.23	6.629	981.7	0.46	5.768	1060.5	0.90	5.417	1051.7	1.11	5.227	1040.0	1.23	5.602	1058.8
8.0	7.349	850.5	0.23	6.798	1031.4	0.46	5.915	1113.1	0.90	5.555	1103.4	1.11	5.360	1090.8	1.24	5.753	1111.3

numbers becomes larger than unity when H becomes larger than 4.6 and 2.6 ft, respectively. In other words, for these latter two bottom slopes critical conditions at the channel entrance limit the flow rates as the head of the reservoir rises above these values. What occurs is critical depth at the entrance of the channel, and then the depth gradually decreases downstream therefrom toward the normal depth associated with this critical flow rate. Such gradually varied flow profiles will be dealt with in Chapter 4. To plot the delivery diagram the critical depth Y_c, and the critical flow rate Q_c from the last column are substituted in place of the computed Y and Q whenever the Froude number is larger than unity. The delivery diagrams for this channel for the five different bottom slopes are given in the graph below, with the depths plotted using the right ordinate.

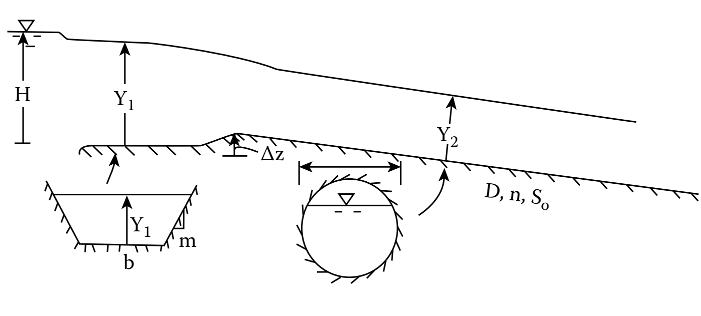


In the design of entrances from reservoirs to channels the size, or shape of the entrance may be different than the channel. If this is the case then solving the flow into the channel will involve the solution of three simultaneous equation for the three unknowns, Q, Y_1, and Y_2, in which Y_1 is the depth in the entrance geometry and Y_2 is the normal depth in the channel downstream therefrom. In other words in place of a single normal depth two depths are added to the list of unknowns. Therefore an additional equation is needed, and this equation comes from writing the energy equation across the transition from the entrance geometry to the channel geometry downstream therefrom. These three equations are

$$F_1 = nQP_2^{2/3} - C_u A_2^{5/3} S_o^{1/2} = 0 \quad \text{(Manning's, or a Uniform F, Equation)}$$

$$F_2 = H - Y_1 - \frac{(1+K_e)(Q/A_1)^2}{(2g)} = 0 \quad \text{(Energy at enterance)}$$

$$F_3 = Y + \frac{(Q/A_1)^2}{(2g)} - Y_2 - \frac{(1+K_L)(Q/A_2)^2}{(2g)} - \Delta z = 0 \quad \text{(Energy across transition)}$$



in which Δz is the rise in the bottom from the entrance to the beginning of the channel, as shown above in the sketch that shows a trapezoidal section at the entrance from the reservoir and a transition to a circular channel downstream therefrom.

Program E_UNTC is designed to solve for any 3 of the 12 variables involved if the entrance from a reservoir to a circular channel is trapezoidal. Generally this program would be used to solve for variables 6 = Q, 8 = Y_1, and 9 = Y_2. For example, if a reservoir with H = 5 ft supplies a circular channel with a diameter D = 12 ft, n = 0.014, and S_o = 0.001 and the entrance consists of a trapezoidal section with b = 10 ft, m = 1.2, and the entrance loss coefficient is K_e = 0.1, and the loss coefficient K_L = 0.1, and there is no change in bottom elevation across the transition from the trapezoidal to the circular sections then this program produces the following: b = 10 ft, m = 1.2, D = 12 ft, S_o = 0.001, n = 0.014, Q = 223.7 cfs (solution), H = 5.0, Y_1 = 4.86 ft (solution), Y_2 = 4.37 ft (solution), K_e = 0.1, K_L = 0.1 and D_z = 0.

E_UNTC.FOR

```
C Solves flow in circular channel that has a trapezoidal entrance
C at the reservoir.
C Equations are: Manning's, Energy at entrance, & Energy between
C 2 shapes.
      CHARACTER*17 FMT/'(1X,A2,'' ='',F9.3)'/
      CHARACTER*2 CX(12)/'b ','m ','D ','So','n ','Q ','H ',
     &'Y1','Y2','Ke','KL','Dz'/
      CHARACTER C1(12)/'2','2','2','5','3','1','2','2','2','3',
     &'3','2'/
      CHARACTER*5 CH(0:1)/'value','guess'/
      REAL F(3),F1(3),D(3,3)
      INTEGER*2 ID(12),INDX(3)
      COMMON X(12),G,G2,Cu
      WRITE(*,*)' Give: g'
      READ(*,*) G
      G2=2.*G
      Cu=1.
      IF(G.GT.15.) Cu=1.486
1     DO 10 I=1,12
      WRITE(*,'(I3,2X,A2)') I,CX(I)
10    ID(I)=0
2     WRITE(*,*)' Give three numbers for 3 unknown','variables'
      READ(*,*) I1,I2,I3
      IF(I1.LT.1.OR.I1.GT.12.OR.I2.LT.1.OR.I2.GT.12
     &.OR.I3.LT.1.OR.I3.GT.12) GO TO 2
      ID(I1)=1
      ID(I2)=1
      ID(I3)=1
      DO 20 I=1,12
      WRITE(*,100) CH(ID(I)),CX(I)
100   FORMAT(' Give ',A5,' for ',A2,' = ',\)
      READ(*,*) X(I)
20    CONTINUE
      NCT=0
30    SUM=0.
      CALL FUN(F)
      I1=0
```

```
      DO 40 I=1,12
      IF(ID(I).EQ.0) GO TO 40
      XX=X(I)
      I1=I1+1
      X(I)=1.005*X(I)
      CALL FUN(F1)
      DO 35 J=1,3
35    D(J,I1)=(F1(J)-F(J))/(X(I)-XX)
      X(I)=XX
40    CONTINUE
      CALL SOLVEQ(3,1,3,D,F,1,DD,INDX)
      I1=0
      DO 50 I=1,12
      IF(ID(I).EQ.0) GO TO 50
      I1=I1+1
      SUM=SUM+ABS(F(I1))
      X(I)=X(I)-F(I1)
50    CONTINUE
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,' SUM=',SUM
      IF(NCT.LT.30 .AND. SUM.GT.3.E-5) GO TO 30
      WRITE(*,*)' Solution:'
      DO 60 I=1,12
      FMT(16:16)=C1(I)
      WRITE(*,FMT) CX(I),X(I)
60    CONTINUE
      WRITE(*,*)' Give 1 to solve another prob.','(0=STOP)'
      READ(*,*) I2
      IF(I2.EQ.1) GO TO 1
      END
      SUBROUTINE FUN(F)
      REAL F(3)
      COMMON X(12),G,G2,Cu
      BETA=ACOS(1.-2.*X(9)/X(3))
      A2=.25*X(3)**2*(BETA-COS(BETA)*SIN(BETA))
      P2=X(3)*BETA
      A1=(X(1)+X(2)*X(8))*X(8)
      FKe=(1.+X(10))/G2
      FKL=(1.+X(11))/G2
      F(1)=X(5)*X(6)*P2**.6666667-Cu*A2**1.6666667*SQRT(X(4))
      F(2)=X(7)-X(8)-FKE*(X(6)/A1)**2
      F(3)=X(8)+(X(6)/A1)**2/G2-X(9)-FKL*(X(6)/A2)**2-X(12)
      RETURN
      END
```

E_UNTC.C

```
// Solves Manning's (uniform flow) and Energy simultaneously for
// any 2 unknowns
// See E_UN1 to solve Q & Y and method that can be used with
// calculator
#include <stdio.h>
```

```
#include <stdlib.h>
#include <math.h>
float x[12],g,g2,cu;
extern void solveq(int n,float **d,float *f,int itype,\
  float *dd,int *indx);
void fun(float *f){float beta,a1,a2,p2,fke,fkl;
 beta=acos(1.-2.*x[8]/x[2]);
 a2=.25*x[2]*x[2]*(beta-cos(beta)*sin(beta));p2=beta*x[2];
 a1=(x[0]+x[1]*x[7])*x[7];fke=(1.+x[9])/g2;fkl=(1.+x[10])/g2;
 f[0]=x[4]*x[5]*pow(p2,.6666667)-cu*pow(a2,1.6666667)*sqrt(x[3]);
 f[1]=x[6]-x[7]-fke*pow(x[5]/a1,2.);
 f[2]=x[7]+pow(x[5]/a1,2.)/g2-x[8]-fkl*pow(x[5]/a2,2.)-x[11];
 return;} //End of fun
void main(void){char *fmt=" %s =%9.3f\n",*ch[]={"value","guess"},\
  *c1="222531222332";
 char *cx[]={"b ","m ","D ","So","n ","Q ","H ","Y1","Y2","Ke","KL","Dz"};
 float f[3],f1[3],xx,sum,**d,*dd;
 int id[12],indx[3],i,j,i1,i2,i3,nct;
 d=(float**)malloc(3*sizeof(float*));
 for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
 printf("Give: g\n");scanf("%f",&g); g2=2.*g; if(g>15.) cu=1.486;
 else cu=1.;
L1: for(i=0;i<12;i++){printf("%2d %s\n",i+1,cx[i]);id[i]=0;}
 do{printf(" Give three numbers for 3 unknown variables\n");
   scanf("%d %d %d",&i1,&i2,&i3);
  }while((i1<1)||(i1>12)||(i2<1)||(i2>12)||(i3<1)||(i3>12));
 id[i1-1]=1;id[i2-1]=1;id[i3-1]=1;
 for(i=0;i<12;i++){printf("Give %s for %s =",ch[id[i]],cx[i]);
 scanf("%f",&x[i]);} nct=0;
 do{sum=0.; fun(f); i1=-1;for(i=0;i<12;i++){if(id[i]){xx=x[i];
   i1++;x[i]*=1.005;fun(f1);
  for(j=0;j<3;j++) d[j][i1]=(f1[j]-f[j])/(x[i]-xx); x[i]=xx;}}
  solveq(3,d,f,1,dd,indx); i1=-1;
  for(i=0;i<12;i++){if(id[i]){sum+=fabs(f[++i1]);x[i]-=f[i1];}}
  printf("nct= %d SUM=%f\n",nct,sum);
  }while((++nct<30)&&(sum>3.e-5));
 printf("Solution:\n");for(i=0;i<12;i++){fmt[8]=c1[i];
   printf(fmt,cx[i],x[i]);}
 printf("Give 1 to solve another prob. (0=STOP)\n");
 scanf("%d",&i2); if(i2) goto L1;}
```

In developing a delivery diagram of flow into a channel with a different geometry at entrance than the channel, critical flow in the channel or its entrance geometry may control. To check which controls the Froude numbers associated with both Y_1 and Y_2 can be computed, and should either of these become larger than unity, then the critical flow equation in this section will replace the uniform flow equation above.

Problems associated with water entering canal systems from reservoirs can become more complex than those illustrated above in which a single channel is taking water from the reservoir. Often several channels may take water from the same inlet works, with the different channels conveying the water away in different directions, at different slopes, and each may have its own controls. Problems may be further complicated by not having uniform flow in some of these channels because controls in them may exist downstream quite some distance from the inlet works creating gradually varied flows in these channels. Problems in which gradually varied flows exist will be discussed in a later Chapter 4. However, if the flow is uniform in the channels, or critical at the entrances in some of the channels, or if the controls in the separate channels are short distances downstream from where the water is taken from the reservoir, then the tools we have covered in this chapter suffice in solving the problems. The explanation and illustrative examples below will help in understanding how to identify what variables need to be solved, and what the governing equations are for a given multiply branched channel system. At this juncture in the course the governing equations come from the following four principles: (1) Conservation of mass, e.g., that the flow out from a junction equals that into the junction. In brief the junction continuity equation applies at each junction. (2) The energy is the same (with local losses accounted for if desired) in upstream and downstream channels branching therefore. Also the head in a reservoir equals the specific energy in the channel it supplies (minus the entrance loss), and the energy line is the same upstream and downstream of gates. (3) For channels that are long after the junction, and do not contain gates, the uniform flow equation (Manning's or Chezy's equation is available). (4) If a situation results in critical flow, the critical flow equation is available. Later, after the tools associated with the momentum principle (Chapter 3) and means for solving gradually varied flows (Chapter 4) are covered, we will allow different depths to exist at the beginning and end of each of the channels in the system, with hydraulic jumps occurring if the conditions allow this.

Based on the current tools there will generally be two unknowns in each channel, the flow rate Q_i and the depth Y_i. Thus if five channels are in the branched system, there will be 10 unknowns and 10 equations need to be written to solve for these 10 variables. The junction continuity equations take on the form

$$F_j = Q_u - \Sigma Q_{di} = 0$$

in which subscript u is the upstream channel number (generally one, but may be more), and subscript d stands for the downstream channels with the extra i indicating there will generally be several of these. The energy equation at the reservoir that supplies the system is of the form

$$F_j = H - Y_1 + \frac{(1+K_e)Q_1^2}{2gA_1^2}$$

The energy equation at the junctions is of the form

$$F_j = Y_u + \frac{Q_u^2}{2gA_u^2} - Y_{di} - (1+K_e)\frac{Q_{di}^2}{2gA_{di}^2} - \Delta z_{ui} = 0$$

The number of these equations is one less than the number of channels at this junction. If desired energy between two of the downstream channels can replace an equation requiring the energy at the upstream channel be the same as in the downstream channel. The energy equations across gates are of the form

$$F_j = Y_u + \frac{Q_u^2}{2gA_u^2} - Y_{di} - (1+K_e)\frac{Q_{di}^2}{2gA_{di}^2} - \Delta z_G = 0$$

in which subscripts u and d denote upstream and downstream of the gate, respectively in channel i.

For any channel that contains uniform flow an equation of the form

$$F_j = n_i Q_i P_i^{2/3} - C_u A_i^{5/3}\sqrt{S_o} = 0$$

is available (or Chezy's equation might be used, alternatively, in which case an additional Chezy's C is added to the list of unknown variables for each channel, and a additional Chezy's C equation is added to the list of equations for each channel.)

One must have insight in deciding if critical flow occurs. If so then the critical flow equation in conjunction with the energy equation can be used to solve for a limiting flow rate. For example, if the downstream channels have greater capacity than the upstream channel can get from the reservoir head H, then the simultaneous solution of the energy equation and the critical flow equation for Y_c and Q_c dictate what the flow rate is. Under these circumstances the specific energy in the upstream channel at the junction, H_j, is unknown and is less than the reservoir head H.

EXAMPLE PROBLEM 2.14

Water is taken from a reservoir by means of a rectangular inlet channel that is 10 m wide. A short distance downstream therefrom the channel divides into a trapezoidal section with $b_2 = 4$ m, $m_2 = 1.5$, $n_2 = 0.015$, and $S_{o2} = 0.0008$, and a pipe with a diameter $D_3 = 3$ m, $n_3 = 0.013$, and $S_{o3} = 0.0014$. The bottom of the pipe is 1.8 m above the bottom of the rectangular channel, and the trapezoidal and rectangular channels have the same bottom elevation. When the water surface elevation in the reservoir is 4.5 m above the bottom of the channel determine the depths and flow rates in all three channel (six unknowns). Assume the entrance loss coefficient equals 0.12.

Solution

To solve for the six unknowns in this problem it is necessary to write six equations, and solve them simultaneously. The equations are

$$F_1 = Q_1 - Q_2 - Q_3 = 0 \quad \text{(continuity)} \tag{1}$$

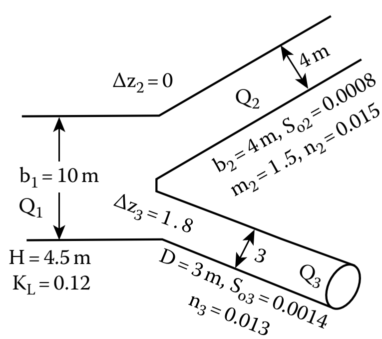
$$F_2 = H - Y_1 - \frac{(1+K_L)(Q_1/A_1)^2}{2g} \quad \text{(energy entrance)} \tag{2}$$

$$F_3 = Y_1 - Y_2 + \frac{\left\{(Q_1/A_1)^2 - (Q_2/A_2)^2\right\}}{2g} = 0 \quad \text{(energy)} \tag{3}$$

$$F_4 = Y_1 - Y_3 - \Delta z_3 + \frac{\left\{(Q_1/A_1)^2 - (Q_3/A_3)^2\right\}}{2g} = 0 \quad \text{(energy)} \tag{4}$$

$$F_5 = Q_2 - \left(\frac{C_u}{n_2}\right) A_2 R_{h2}^{2/3}\sqrt{S_{o2}} = 0 \quad \text{(uniform flow)} \tag{5}$$

$$F_6 = Q_3 - \left(\frac{C_u}{n_3}\right) A_3 R_{h3}^{2/3}\sqrt{S_{o3}} = 0 \quad \text{(uniform flow)} \tag{6}$$



in which H = 4.5 m, and C_u is 1 for SI units. A systematic method such as the Newton method described in Appendix B is needed to solve these equations for the six unknowns: Q_1, Q_2, Q_3, Y_1, Y_2, Y_3. It is relatively easy to provide reasonable guesses to these unknowns since the velocity heads will be small for this problem in comparison to the depths. Therefore one would expect Y_1, Y_2, and Y_3 to be only slightly less than the water surface elevation in the reservoir. The solution provides the following values: Q_1 = 136.433 m³/s, Q_2 = 121.127 m³/s, Q_3 = 15.306 m³/s, Y_1 = 3.741 m, Y_2 = 3.920 m, Y_3 = 2.249 m.

The following TURBO PASCAL computer program implements a solution to this problem with the following input data: 4.5 .12 10 0 0 1.8 4 1.5 .015 .0008 3 .013 .0014

140 3.6 125 3.8 15 1.25 {est. for unknown}

The FORTRAN and C programs are written so the input consist of g, H, K L and then for each channel thereafter: a 0 for trapezoidal and 1 for circular channel, b, m, n, So and Δz for each channel including #1. When giving the data for channel 1 its n, S_o, and Δz can be given as any values since they are not used. The input for these programs is

```
9.81 4 .12 0 10 0 .013 .001 0 0 4 1.5 .015 .0008 0 1 4 0 .013
  .0014 1.8 140 3.6 125 3.8 15 1.25
```

A TK-Solver model and a Mathcad model are also given to solve three channel problems. The given variables in these models solve this problem.

PASCAL program for solving the above problem (THREEC.PAS)

```
Program ThreeCh;
 Const NC=3;NU=6;g=9.81;g2=19.62;C=1.0;
 Var H,KL,P,AA: real;
  b,m,n,So,Y,Q,Z:array[1..NC] of real; EQ:array[1..NU] of real;
  D:array [1..NU] of array [1..NU] of real;
    Cir:array [1..NC] of boolean;
 Function EXPN(a,b:real):real;Begin EXPN:=Exp(b*Ln(a)) End;
 Function A(K:integer):real;
  Begin If Cir[K] then begin P:=1-2*Y[K]/b[K];
    if abs(P)<0.00001 then m[K]:=Pi/2
   else begin AA:=sqrt(1-sqr(P))/abs(P);
   if P<0 then m[K]:=Pi-ArcTan(AA) else m[K]:=ArcTan(AA);
   AA:=sqr(b[K])/4*(m[K]-P*sin(m[K]));P:=m[K]*b[K];
     A:=AA end; end else begin
   P:=b[K]+2*Y[K]*sqrt(sqr(m[K])+1);
     AA:=(b[K]+m[K]*Y[K])*Y[K];A:=AA end;
  End;
 Function F(K:integer):real;
  Begin Case K of
   1: F:=Q[1]-Q[2]-Q[3];
   2: F:=H-Y[1]-KL*sqr(Q[1]/A(1))/g2;
   3,4:F:=Y[1]-Y[K-1]+(sqr(Q[1]/A(1))-
     sqr(Q[K-1]/A(K-1)))/g2-Z[K-1];
   5,6:F:=Q[K-3]-C/n[K-3]*A(K-3)*EXPN(AA/P,0.6666667)
     *sqrt(So[K-3]) end;
  End;
 Var I,J,KI:integer; ADIF,FAC,SUM:real;
   Dif:array [1..NU] of real;
 BEGIN
  Cir[1]:=false;Cir[2]:=false;Cir[3]:=true;
  Writeln('Give:H,KL,b1,m1,Z2,Z3,b2,m2,n2,So2,D3,n3,So3');
  Readln(H,KL,b[1],m[1],Z[2],Z[3],b[2],m[2],n[2],So[2],b[3],
    n[3],So[3]);
  Writeln('Give est. for:Q1,Y1,Q2,Y1,Q3,Y3');
```

```
      Readln(Q[1],Y[1],Q[2],Y[2],Q[3],Y[3]); KL:=1+KL;
     { Defines equations and Jacobian matrix for Newton Solution}
     repeat
      For I:=1 to NU do Begin EQ[I]:=F(I);
       For J:=1 to NU do begin
        If J>NC then Y[J-NC]:=Y[J-NC]-0.001 else Q[J]:=Q[J]-0.001;
        D[I,J]:=(EQ[I]-F(I))/0.001;
        If J>NC then
        Y[J-NC]:=Y[J-NC]+0.001 else Q[J]:=Q[J]+0.001 end; End;
     { Solves linear equations}
      For KI:=1 to NU-1 do Begin
       For I:=KI+1 to NU do if D[I,KI]<>0 then begin
        FAC:=D[I,KI]/D[KI,KI];
        For J:=KI+1 to NU do D[I,J]:=D[I,J]-FAC*D[KI,J];
         EQ[I]:=EQ[I]-FAC*EQ[KI] end; End;
      Dif[NU]:=EQ[NU]/D[NU,NU]; Y[NC]:=Y[NC]-Dif[NU];
        ADIF:=abs(Dif[NU]);
 For I:=NU-1 downto 1 do Begin SUM:=0;
  For J:=I+1 to NU do SUM:=SUM+Dif[J]*D[I,J];
    Dif[I]:=(EQ[I]-SUM)/D[I,I];
   If I>NC then Y[I-NC]:=Y[I-NC]-Dif[I] else Q[I]:=Q[I]-Dif[I];
   ADIF:=ADIF+abs(Dif[I]) End;
until (ADIF < 0.0001);
Write('Unknown flow rates: ');
 For I:=1 to NC-1 do Write('Q(',I:1,')=',Q[I]:10:3,',');
    Writeln('Q(',NC:1,')=',Q[NC]:10:3);
Write('Unknown Depths: ');
 For I:=1 to NC-1 do Write('Y(',I:1,')=',Y[I]:10:3,',');
    Writeln('Y(',NC:1,')=',Y[NC]:10:3);
END
```

FORTRAN program to solve problem (THREECH.FOR)

```
      PARAMETER (NC=3,NU=6)
      INTEGER*2 Cir(NC)
      CHARACTER*49 FMT/'(3(3H Q(,I1,2H)=,F10.3),/,3(3H Y(,I1,2H)=,
     &F10.4))'/
      COMMON g,g2,C,H,FKL,P,b(NC),FM(NC),FN(NC),So(NC),Y(NC),Q(NC),
     &Z(NC),EQ(NU),D(NU,NU),Dif(NU),Cir
      FMT(2:2)=CHAR(48+NC)
      FMT(27:27)=CHAR(48+NC)
      WRITE(*,90) NC
90    FORMAT(' Give g,H,KL & for',I2,' Channels:0=Trap or
     &1=Cir,b,m,n,So & z(or Cir D & 0 for b,m)')
      READ(*,*) g,H,FKL,(Cir(I),b(I),FM(I),FN(I),So(I),z(I),I=1,NC)
      WRITE(*,*)' Give Est. of Q and Y for',NC,' Channels'
      READ(*,*)(Q(I),Y(I),I=1,NC)
      g2=2.*g
      C=1.486
      IF(G.LT.20.) C=1.
      FKL=FKL+1
10    DO 20 I=1,NU
      EQ(I)=F(I)
      DO 20 J=1,NU
      IF(J.GT.NC) THEN
      Y(J-NC)=Y(J-NC)-.001
      ELSE
```

```
      Q(J)=Q(J)-.001
      ENDIF
      D(I,J)=(EQ(I)-F(I))/.001
      IF(J.GT.NC) THEN
      Y(J-NC)=Y(J-NC)+.001
      ELSE
      Q(J)=Q(J)+.001
      ENDIF
20    CONTINUE
      DO 40 KI=1,NU-1
      DO 40 I=KI+1,NU
      IF(ABS(D(I,KI)).LT.1.E-7) GO TO 40
      FAC=D(I,KI)/D(KI,KI)
      DO 30 J=KI+1,NU
30    D(I,J)=D(I,J)-FAC*D(KI,J)
      EQ(I)=EQ(I)-FAC*EQ(KI)
40    CONTINUE
      Dif(NU)=EQ(NU)/D(NU,NU)
      Y(NC)=Y(NC)-Dif(NU)
      ADIF=ABS(Dif(NU))
      DO 60 I=NU-1,1,-1
      SUM=0.
      DO 50 J=I+1,NU
50    SUM=SUM+Dif(J)*D(I,J)
      Dif(I)=(EQ(I)-SUM)/D(I,I)
      IF(I.GT.NC) THEN
      Y(I-NC)=Y(I-NC)-Dif(I)
      ELSE
      Q(I)=Q(I)-Dif(I)
      ENDIF
60    ADIF=ADIF+ABS(Dif(I))
      IF(ADIF.GT. .0001) GO TO 10
      WRITE(6,FMT) (I,Q(I),I=1,NC),(I,Y(I),I=1,NC)
      END
      FUNCTION A(K)
      PARAMETER (NC=3,NU=6)
      INTEGER*2 Cir(NC)
      COMMON g,g2,C,H,FKL,P,b(NC),FM(NC),FN(NC),So(NC),Y(NC),Q(NC),
     &Z(NC),EQ(NU),D(NU,NU),Dif(NU),Cir
      IF(Cir(K).GT.0) THEN
      COSB=1.-2.*Y(K)/b(K)
      FM(K)=ACOS(COSB)
      P=FM(K)*b(K)
      A=.25*b(K)**2*(FM(K)-COSB*SIN(FM(K)))
      ELSE
      P=b(K)+2.*Y(K)*SQRT(FM(K)**2+1.)
      A=(b(K)+FM(K)*Y(K))*Y(K)
      ENDIF
      RETURN
      END
      FUNCTION F(K)
      PARAMETER (NC=3,NU=6)
      INTEGER*2 Cir(NC)
      COMMON g,g2,C,H,FKL,P,b(NC),FM(NC),FN(NC),So(NC),Y(NC),Q(NC),
     &Z(NC),EQ(NU),D(NU,NU),Dif(NU),Cir
      IF(K.LT.2) THEN
```

```
      QQ=Q(2)
      DO 2 J=3,NC
2     QQ=QQ+Q(J)
      F=Q(1)-QQ
      ELSE IF(K.EQ.2) THEN
      F=H-Y(1)-FKL*(Q(1)/A(1))**2/g2
      ELSE IF(K.LT.NC+2) THEN
      F=Y(1)-Y(K-1)+((Q(1)/A(1))**2-(Q(K-1)/A(K-1))**2)/
     &g2-Z(K-1)
      ELSE
      AAA=A(K-NC)
      F=Q(K-NC)-C*AAA*(AAA/P)**.6666667*SQRT(So(K-NC))/FN(K-NC)
      ENDIF
      RETURN
      END
```

Program THREECH.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#define NC 3
#define NU 6
float g,g2,c,h,fkl,p,b[NC],fm[NC],fn[NC],so[NC],y[NC],q[NC],\
  z[NC],eq[NU];int cir[NC];
float a(int k){float cosb;
 if(cir[k]){cosb=1.-2.*y[k]/b[k]; fm[k]=acos(cosb);p=fm[k]*b[k];
   return .25*b[k]*b[k]*(fm[k]-cosb*sin(fm[k]));}
 else {p=b[k]+2.*y[k]*sqrt(fm[k]*fm[k]+1.);
   return (b[k]+fm[k]*y[k])*y[k];}} // End of a
float f(int k){float qq,aaa; int j;
 if(k<1){qq=q[1];for(j=2;j<NC;j++) qq+=q[j];
   return q[0]-qq;} else if(k==1){
   return h-y[0]-fkl*pow(q[0]/a(0),2.)/g2;} else if(k<NC+1) {
   return y[0]-y[k-1]+(pow(q[0]/a(0),2.)-pow(q[k-1]/a(k-1),2.))
     g2-z[k-1];} else {
   aaa=a(k-NC); return q[k-NC]-c*aaa*pow(aaa/p,.6666667)
   *sqrt(so[k-NC])/fn[k-NC];}} // End of f
void main(void){int i,j,ki; float **d,fac,adif,sum,dif[NU];
 d=(float**)malloc(NU*sizeof(float*));
 for(i=0;i<NU;i++)d[i]=(float*)malloc(NU*sizeof(float));
 printf("Give g,H,KL & for %2d Channels:0=Trap or 1=Cir,b,m,n,So\
   & z(or Cir D & 0 for b,m)\n",NC);
 scanf("%f %f %f",&g,&h,&fkl);
 for(i=0;i<NC;i++) scanf("%d %f %f %f %f %f",&cir[i],&b[i],\
   &fm[i],&fn[i],&so[i],&z[i]);
 printf("Give Est. of Q and Y for %2d Channels\n",NC);
 for(i=0;i<NC;i++) scanf("%f %f",&q[i],&y[i]); g2=2.*g;c=1.486;
 if(g<20.) c=1.; fkl+=1.;
 do{for(i=0;i<NU;i++){eq[i]=f(i);
   for(j=0;j<NU;j++){if((j+1)>NC) y[j-NC]-=.001; else q[j]-=.001;
   d[i][j]=(eq[i]-f(i))/.001; if((j+1)>NC) y[j-NC]+=.001;
   else q[j]+=.001;}}
   for(ki=0;ki<NU-1;ki++){for(i=ki+1;i<NU;i++){
   if(fabs(d[i][ki])>1.e-7){fac=d[i][ki]/d[ki][ki];
   for(j=ki+1;j<NU;j++) d[i][j]-=fac*d[ki][j];
   eq[i]-=fac*eq[ki];}}}
```

```
 dif[NU-1]=eq[NU-1]/d[NU-1][NU-1]; y[NC-1]-=dif[NU-1];
     adif=fabs(dif[NU-1]);
 for(i=NU-2;i>-1;i--){sum=0.;
     for(j=i+1;j<NU;j++) sum+=dif[j]*d[i][j];
 dif[i]=(eq[i]-sum)/d[i][i];if(i+1>NC) y[i-NC]-=dif[i];
     else q[i]-=dif[i];
 adif+=fabs(dif[i]);}}while (adif>.0001);
 for(i=0;i<NC;i++) printf("Q(%2d)=%10.3f\n",i+1,q[i]);
 for(i=0;i<NC;i++) printf("Y(%2d)=%10.3f\n",i+1,y[i]);
}
```

TK-solver model

```
============= VARIABLE SHEET =========
St Input----- Name---- Output---- Unit----
              Q1       136.43337
              Q2       121.1267
              Q3       15.306669
   4.5        H
              Y1       3.7405792
              A1       37.405792
   19.62      g2
              Y2       3.9201675
              A2       38.73224
   4          b2
   1.5        m2
   10         b1
              beta     2.0936528
              Y3       2.2490355
   3          D
              A3       5.6841611
   .12        Ke
   1.8        dz
   .0008      So2
   .015       n2
   .013       n3
   .0014      So3
============ RULE SHEET ==========
S Rule----------------------------
  Q1=Q2+Q3
  A1=b1*Y1
  A2=(b2+m2*Y2)*Y2
  cos(beta)=1.-2*Y3/D
  A3=.25*D^2*(beta-cos(beta)*sin(beta))
  H=Y1+(1.+Ke)*(Q1/A1)^2/g2
  Y1+(Q1/A1)^2/g2=Y2+(Q2/A2)^2/g2
  Y1+(Q1/A1)^2/g2=Y3+(Q3/A3)^2/g2+dz
  Q2=(A2/(b2+2*Y2*sqrt(m2^2+1)))^.6666667*A2*sqrt(So2)/n2
  Q3=(A3/(D*beta))^.6666667*A3*sqrt(So3)/n3
```

Mathcad-model (THREEC.MCD)

```
Variables:H = 4.5 Ke = 0.12 b1 = 10 b2 = 4 m2 = 1.5 n2 = 0.015
So2 = 0.0008 D = 3 n3 = 0.013 So3 = 0.0014 dz = 1.8g = 9.81
Q1 = 130 Q2 = 110 Q3 = 12 Y1 = 3.8 Y2 = 3.9 Y3 = 2.25 Beta = 1.8
```

Given

$$Q_1 = Q_2 + Q_3 \quad H = Y1 + (1+Ke)\cdot\frac{Q1^2}{2\cdot g\cdot(b1\cdot Y1)^2} \quad Y1 + \frac{Q1^2}{2\cdot g\cdot(b1\cdot Y1)^2} = Y2 + \frac{Q2^2}{2\cdot g\cdot((b2+m2\cdot Y2)\cdot Y2)^2}$$

$$\cos(\mathrm{Beta})=1-2\cdot\frac{Y3}{D} \quad Y1+\frac{Q1^2}{2\cdot g\cdot(b1\cdot Y1)^2}=dz+Y3+\frac{Q3^2}{2\cdot g\cdot[D\cdot D/4\cdot(\mathrm{Beta}-\cos(\mathrm{Beta})\cdot\sin(\mathrm{Beta}))]^2}$$

$$Q2=\frac{((b2+m2\cdot Y2)\cdot Y2)^{1.666667}}{\left(b2+2\cdot Y2\sqrt{m2\cdot m2+1}\right)^{.6666667}}\cdot\frac{\sqrt{So2}}{n2} \quad Q3=\frac{[D\cdot D/4\cdot(\mathrm{Beta}-\cos(\mathrm{Beta})\cdot\sin(\mathrm{Beta}))]^{1.6666667}}{(D\cdot \mathrm{Beta})^{6666667}}\cdot\frac{\sqrt{So3}}{n3}$$

$$\mathrm{Find}(Q1,Q2,Q3,Y1,Y2,Y3,\mathrm{Beta})=\begin{bmatrix}136.433\\121.127\\15.307\\3.741\\3.92\\2.249\\2.094\end{bmatrix}$$

The PASCAL, FORTRAN, and C programs are written in a general way so that with minor modifications they would solve problems in which more than two channels branch from a single channel. The constants NC (number of channels) and NU (number of unknowns) would need to be increased to 4 and 8, respectively if four channels were involved. The "Case of" statement to 3, 4 and 5, 6 would need to read 3, 4, 5 and 5, 6, 8 respectively, and the values subtracted from K would also need to be changed. Also the input and output statements would need modification to handle a larger problem, but the portion of the program that generates the Jacobian for the Newton method, and the solution by this method would need no change. D is a two-dimensional array for the Jacobian matrix used in the Newton method, and its derivative elements are numerically evaluated by subtracting 0.001 from the value of the variable that the derivative is being taken with respect to, evaluating the equation again, and then dividing the difference between the two values of the equation by 0.001. The program is also written general enough that any of the three channels may be of circular or trapezoidal cross sections, by setting the boolean Cir equal to true for any circular section. When a section is circular b[I] holds the diameter of the section instead of the bottom width, and upon obtaining the solution m[I] will contain the angle β.

It should be noted that if the entrance loss coefficient were zero, with the short length for the upstream channel so that there is no variation of depth from its beginning to end, then it would be possible to solve for the flow rate and depth in each of the downstream branched channels separately because the head at the beginning of each of these would be H then. However, with a nonzero entrance loss coefficient K_e the specific energy becomes dependent upon the flow rate in the upstream main channel, which in turn depends upon the flow rate in the two downstream channels; thus the system of six equations must be solved simultaneously. A good estimate of the solution could be obtained, however by solving each downstream channel separately.

EXAMPLE PROBLEM 2.15

Modify the above program to handle similar problem but in ES units.

Solution

The only modification needed is to change the constants g, g2 and C to 32.2, 64.4, and 1.486, respectively.

EXAMPLE PROBLEM 2.16

Consider the following variation of the previous problem. A channel takes the water from a reservoir and then divides into two channels as in the previous problem, but in this problem gates in the two downstream channels are used to control the flows in each. In channel 2 the gate is set so its bottom is 1.0 m above the channel bottom, and in channel 3 the gate is 1.5 m above the channel bottom. All channels have the same bottom elevation at the Y section. If the water surface elevation in the reservoir is 3.5 m above the channel bottom, and the minor loss coefficient is 0.09 determine the flow rates in the three channels.

Solution

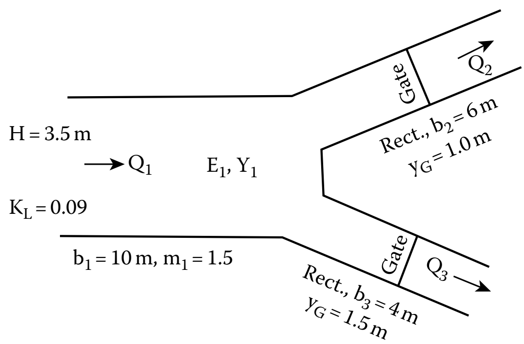
In this problem it is a reasonable assumption that the depths upstream from the gates in all three channels are the same, i.e., Y_1, since the gates will restrict the flow the velocity heads in all channels upstream from the gates will be relatively small. Under this assumption the unknowns are: Q_1, Q_2, Q_3, and Y_1. If one is not willing to live with this assumption then two additional unknowns, Y_2 and Y_3 must be added. Based on the above assumption the four equations whose simultaneous solution provides the solution to the problem are

$$F_1 = Q_1 - Q_2 - Q_3 = 0$$

$$F_2 = H - (1 + K_L)(Q_1^2/(2gA_1^2)) = 0$$

$$F_3 = Y_1 + \frac{(Q_2/A_{2u})^2}{2g} - Y_{2d} - \frac{(Q_2/A_{2d})^2}{2g} = 0$$

$$F_4 = Y_1 + \frac{(Q_3/A_{3u})^2}{2g} - Y_{3d} - \frac{(Q_3/A_{3d})^2}{2g} = 0$$



in which H = 3.5 m, K_L = 0.09, and the added second subscripts u and d stand for upstream and downstream from the gates. The depth downstream from the gate will equal the gate position times its contraction coefficient, C_c = 0.6.

The solutions to these four equations are Y_1 = 3.316 m, Q_1 = 46.348 m³/s, Q_2 = 13.855 m³/s, Q_3 = 32.493 m³/s.

EXAMPLE PROBLEM 2.17

A flow rate of 15 cfs/ft passes under a sluice gate whose tip is placed at the distance of 1 ft above the channel bottom. The contraction coefficient for the gate is C_c = 0.6, and the channel is rectangular. Determine the depth of flow upstream from this gate.

Solution

This problem is relative easy to solve because it deals with a rectangular channel. The depth a very short distance downstream from the gate will equal the height of the gate times the contraction coefficient, or Y_2 = 0.6(1) = 0.6 ft. The problem now consists of finding the alternate depth to this downstream supercritical depth. Substituting this depth into the specific energy Equation 2.11c gives E_2 = 10.305 ft. There is very small energy loss past a gate and therefore $E_1 = E_2$, or writing this out gives the following equation:

$$Y + \frac{q^2}{2gY^2} = 10.305$$

Rearranging results in the cubic equation,

$$Y^3 - 10.305Y^2 + 3.494 = 0 \left(\text{or in general } Y^3 - EY^2 + \frac{q^2}{2g} = 0\right)$$

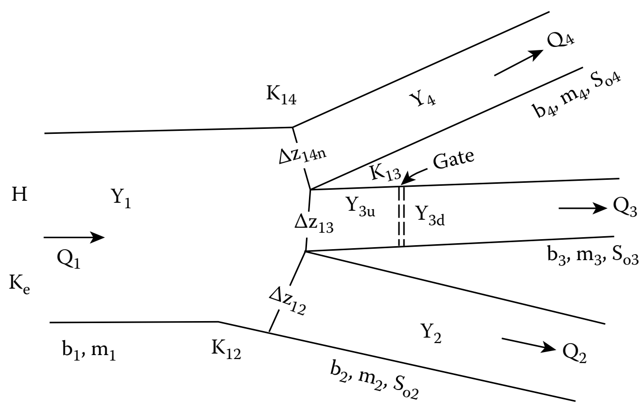
Since the downstream depth $Y_2 = 0.6$ ft is a known root of this cubic equation it can be reduced to a quadratic equation by extracting this root. The use of synthetic division as follows is a convenient means of reducing the equation to a quadratic:

$$\begin{array}{rrrr|l} 1 & -10.305 & 0 & 3.494 & .6 \\ & .6 & -5.823 & -3.49 & \\ \hline 1 & -9.705 & -5.823 & 0 & \end{array}$$

and therefore the quadratic equation is $Y^2 - 9.705Y - 5.823 = 0$, which can be solved by the quadratic formula giving $Y = 10.272$ ft. (Note the negative root obtained from the quadratic equation can be discarded as physically impossible.) This technique works only if $E_1 = E_2$.

EXAMPLE PROBLEM 2.18

Three channels branch from a main channel a short distance downstream from where it receives water from a reservoir as shown in the sketch below. The properties of the channels are as follows: $b_1 = 15$ ft, $m_1 = 2$, $b_2 = 8$ ft, $m_2 = 1$, $n_2 = 0.013$, $S_{o2} = 0.0008$, $b_3 = 5$ ft, $m_3 = 1$, $n_3 = 0.013$, $S_{o3} = 0.0008$, $b_4 = 5$ ft, $m_4 = 1$, $n_4 = 0.013$, $S_{o4} = 0.0008$, $H = 5$ ft. All minor loss coefficients, K's = 0.2, which include the entrance loss coefficient from the reservoir into main channel, and from this main channel into each of the three branch channels. All bottom elevations at the junction are the same. Channel # 3 contains a gate. Determine the flow rates and depth in all of the channels as the gate is closed from a wide open position to a completely closed position in channel # 3. The gates contraction coefficient is $C_c = 0.6$.



Solution

In this problem there are eight unknown, namely the flow rates and the depths in all four channels. These unknowns are: Q_1, Y_1, Q_2, Y_2, Q_3, Y_3, Q_4, Y_4. The eight equations needed to solve these eight unknowns are

$$F_1 = Q_1 - Q_2 - Q_3 - Q_4 = 0$$

$$F_2 = H - Y_1 - \frac{(1+K_e)\{Q_1/A_1\}^2}{(2g)} = 0$$

$$F_3 = Y_1 + \frac{\{Q_1/A_1\}^2}{2g} - Y_2 - \frac{(1+K_{12})\{Q_2/A_2\}^2}{2g} - \Delta z_{12} = 0$$

$$F_4 = Y_1 + \frac{\{Q_1/A_1\}^2}{2g} - Y_{3u} - \frac{(1+K_{13})\{Q_3/A_{3u}\}^2}{2g} - \Delta z_{13} = 0$$

$$F_5 = Y_1 + \frac{\{Q_1/A_1\}^2}{2g} - Y_4 - \frac{(1+K_{14})\{Q_4/A_4\}^2}{2g} - \Delta z_{14} = 0$$

$$F_6 = n_2 Q_2 P_2^{2/3} - C A_2^{5/3} S_{o2}^{1/2} = 0$$

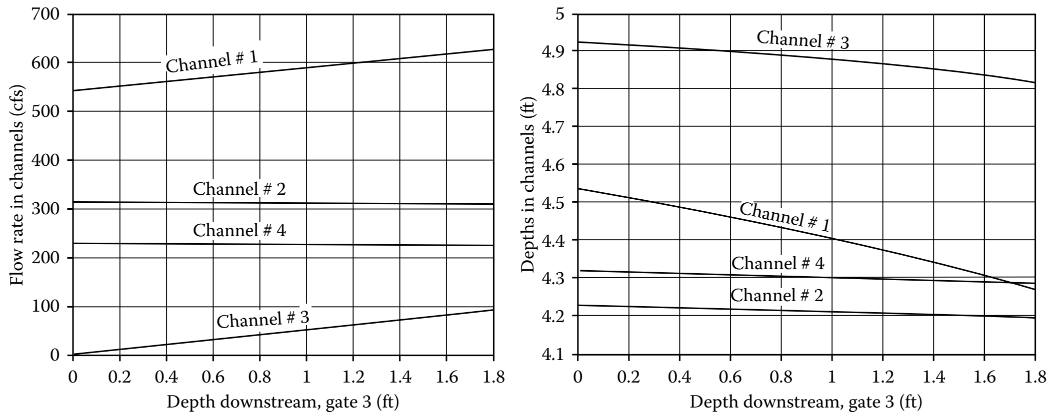
$$F_7 = Y_{3u} + \frac{\{Q_3/A_{3u}\}^2}{2g} - Y_{3d} - \frac{(1+K_g)\{Q_3/A_{3d}\}^2}{2g} = 0$$

$$F_8 = n_4 Q_4 P_4^{2/3} - C A_4^{5/3} S_{04}^{1/2} = 0$$

The listing below is a FORTRAN program that is designed to solve for 2 × N_c unknowns in which N_c is the number of channels at the branch. The unknown might be the width of the channel, if the flow rate is specified, for example. This program has been modified slightly in PRB18.FOR (and PRB18.C) to solve the above problem repeatedly for different depths Y_{d3}, in channel # 3 downstream from the gate to produce the table given below. These depth and flow rates are plotted in the figures below. Notice from these figures (and the table) that as the gate in channel # 3 is closed only a very modest increase in depths and flow rates in channel # 2 and # 3 occur. This is the case since the velocity heads are relatively small in comparison to the depths. For steeper channels the effects would be larger.

Table Giving Depth and Flowrate in All Four Channels in Problem 2.18 as They Change with the Depth Downstream from the Gate in Channel # 3. The Other Channels Do Not Contain Gates

Height	Channel # 1		Channel # 2		Channel # 3		Channel # 4	
Gate (ft)	Q (cfs)	Y (ft)	Q (cfs)	Y (ft)	Q (cfs)	Y (ft)	Q (cfs)	Y (ft)
0.01	545.1	4.535	315.5	4.229	0.5	4.923	229.1	4.318
0.10	549.2	4.526	315.3	4.228	4.9	4.921	229.0	4.317
0.20	553.7	4.514	315.1	4.226	9.9	4.918	228.8	4.315
0.30	558.3	4.503	314.8	4.225	14.9	4.916	228.7	4.314
0.40	563.0	4.491	314.6	4.223	19.9	4.913	228.5	4.312
0.50	567.6	4.478	314.3	4.221	25.0	4.909	228.3	4.310
0.60	572.3	4.466	314.1	4.219	30.2	4.905	228.1	4.308
0.70	577.0	4.452	313.8	4.217	35.3	4.900	227.9	4.306
0.80	581.8	4.438	313.6	4.216	40.5	4.896	227.7	4.304
0.90	586.5	4.424	313.3	4.213	45.8	4.890	227.5	4.302
1.00	591.3	4.409	313.0	4.211	51.1	4.884	227.2	4.300
1.10	596.0	4.394	312.7	4.209	56.3	4.878	227.0	4.298
1.20	600.8	4.378	312.4	4.207	61.7	4.871	226.8	4.295
1.30	605.6	4.362	312.1	4.205	67.0	4.863	226.5	4.293
1.40	610.3	4.345	311.7	4.202	72.3	4.855	226.3	4.290
1.50	615.0	4.327	311.4	4.200	77.7	4.847	226.0	4.288
1.60	619.8	4.309	311.0	4.197	83.0	4.838	225.7	4.285
1.70	624.5	4.290	310.6	4.194	88.4	4.828	225.4	4.282
1.80	629.1	4.270	310.3	0.191	93.8	0.817	25.1	4.80



Program BRANCHCHL.FOR solves general branched channels

```
      REAL X[ALLOCATABLE](:),F[ALLOCATABLE](:),
     &F1[ALLOCATABLE](:),D[ALLOCATABLE](:,:)
     &,DZ[ALLOCATABLE](:),KLOS[ALLOCATABLE](:)
      INTEGER*2 IT[ALLOCATABLE](:),INDX[ALLOCATABLE](:),
     &IV[ALLOCATABLE](:)
      LOGICAL*1 STEEP[ALLOCATABLE](:),KEYB
      CHARACTER*2 CU[ALLOCATABLE](:)
      CHARACTER*6 S(2)/'SQYbmn','SQYDn '/
      CHARACTER*1 CH
      CHARACTER*38  FMT/"(1X,A1,' =',F8.6,/,(1X,A1,' =',F8.3))"/
      COMMON CMAN,G,G2,YGA(5),EYG(5),IYG(5),IGATE(6),NGATE
      WRITE(*,*)' Give: (1) 1=ES, 2=SI;', (2) No. channels;',
     &' (3) No. gates (4) IN-unit;',= (5) OUT-unit'
      KEYB=.FALSE.    ! Trapezoidal    Circular
      READ(*,*)II,NC,NGATE,IN,IOUT!-------------  ---------
      IGATE(NGATE+1)=NC+1 ! H  = x(1)       H  = x(1)
      IF(IN.EQ.0.OR.IN.EQ.5)KEYB=.TRUE.!Q1 = x(2) Q1 = x(2)
      IF(II.EQ.2) THEN  ! Y1 = x(3)       Y1 = x(3)
      G=9.81            ! b1 = x(4)       D1 = x(4)
      CMAN=1.           ! m1 = x(5)       n1 = x(5)
      ELSE              ! n1 = x(6)
      G=32.2            !  So2= x(7)       So2= x(6)
      CMAN=1.486        !  Q2 = x(8)       Q2= x(7)
      ENDIF             !  Y2 = x(9)       Y2 = x(8)
      G2=2.*G           !  b2 = x(10)      D2 = x(9)
      N=2*NC            !  m2 = x(11)      n2 = x(10)
      ALLOCATE(IT(NC),KLOS(NC),DZ(NC),F(N),F1(N),D(N,N),
     &CU(N),STEEP(NC))
      IF(KEYB)WRITE(*,*)' For each channel give 1 = trap.',
     &', or 2 = cir.'
      NV=0
      DO 10 I=1,NC
      IF(KEYB) WRITE(*,"(' Channel #',I2,' = ',\)") I
      READ(IN,*) IT(I)
10    NV=NV-IT(I)+7
      ALLOCATE(X(NV),IV(N),INDX(N))
      II=0
      DO 30 I=1,NC
      STEEP(I)=.FALSE.
```

```
      IF(KEYB) THEN
      WRITE(*,"('Give variables for Channel',I2)") I
      DO 20 J=1,7-IT(I)
      CH=S(IT(I))(J:J)
      IF(I.EQ.1 .AND. J.EQ.1) CH='H'
      WRITE(*,"(5X,A1,' = ',\)") CH
20    READ(*,*) X(J+II)
      WRITE(*,"(' Loss Coef =',\)")
      READ(*,*) KLOS(I)
      IF(I.GT.1) THEN
      WRITE(*,"(' Change in bottom position =',\)")
      READ(*,*) DZ(I)
      ENDIF
      ELSE
      IF(I.EQ.1) THEN
      READ(IN,*)(X(J),J=1,7-IT(I)),KLOS(I)
      ELSE
      READ(IN,*)(X(J+II),J=1,7-IT(I)),KLOS(I),DZ(I)
      IF(X(II+1).GT. .008) STEEP(I)=.TRUE.
      ENDIF
      ENDIF
30    II=II+7-IT(I)
      IF(NGATE.GT.0)READ(IN,*) (IGATE(I),YGA(I),I=1,NGATE)
      IF(KEYB) WRITE(*,*)' Give symbols for',N,
     &' Unknowns, i.e. Q1 Y1 b2 etc.'
      READ(IN,120) (CU(I),I=1,N)
120   FORMAT(26(A2,1X))
      IU=0
      IPOS=0
      II1=1
      DO 40 I=1,N
      II=ICHAR(CU(I)(2:2))-48
      DO 35 J=1,7-IT(II)
      IF(S(IT(II))(J:J).NE.CU(I)(1:1)) GO TO 35
      IF(II.NE.II1) THEN
      IPOS=IPOS+7-IT(II1)
      II1=II
      ENDIF
      IU=IU+1
      IV(IU)=IPOS+J
      GO TO 40
35    CONTINUE
40    CONTINUE
      NCT=0
42    CALL FUNCT(NC,N,NV,IT,STEEP,X,F,DZ,KLOS)
      DO 50 J=1,N
      XX=X(IV(J))
      X(IV(J))=1.005*X(IV(J))
      CALL FUNCT(NC,N,NV,IT,STEEP,X,F1,DZ,KLOS)
      DO 45 I=1,N
45    D(I,J)=(F1(I)-F(I))/(X(IV(J))-XX)
50    X(IV(J))=XX
      CALL SOLVEQ(N,1,N,D,F,1,DD,INDX)
      NCT=NCT+1
      SUM=0.
      DO 60 I=1,N
      X(IV(I))=X(IV(I))-F(I)
```

```
60    SUM=SUM+ABS(F(I))
      WRITE(*,*)' Iteration=',NCT,' SUM=',SUM
      IF(NCT.LT.20 .AND. SUM.GT. .0005) GO TO 42
      DO 65 I=1,NGATE
      NCT=2
      II=IYG(I)
63    FF=EYG(I)-X(II)-(X(II-1)/((X(II+1)+X(II+2)*
     &X(II))*X(II)))**2/G2
      IF(MOD(NCT,2) .NE. 0) GO TO 64
      NCT=NCT+1
      F11=FF
      XX=X(II)
      X(II)=1.005*X(II)
      GO TO 63
64    NCT=NCT+1
      DIF=(X(II)-XX)*F11/(FF-F11)
      X(II)=XX-DIF
      WRITE(*,*)' NCT=',NCT,'  SUM=',SUM
      IF(NCT.LT. 40 .AND. ABS(DIF).GT. .0001) GO TO 63
      IF(NCT.GE.40) WRITE(*,*)' Failed to converge',
     &' with gates',DIF
65    CONTINUE
      II=0
      S(IT(1))(1:1)='H'
      FMT(16:16)='3'
      DO 70 I=1,NC
      WRITE(IOUT,"(' For Channel',I2)") I
      WRITE(IOUT,FMT)(S(IT(I))(J:J),X(J+II),J=1,7-IT(I))
      IF(I.EQ.1) S(IT(1))(1:1)='S'
      FMT(16:16)='6'
70    II=II-IT(I)+7
      END
      SUBROUTINE FUNCT(NC,N,NV,IT,STEEP,X,F,DZ,KLOS)
      REAL X(NV),F(N),DZ(N),KLOS(N)
      INTEGER IT(NC)
      LOGICAL*1 STEEP(NC)
      COMMON CMAN,G,G2,YGA(5),EYG(5),IYG(5),IGATE(6),NGATE
      II=0
      IG=1
      DO 20 I=1,NC
      IF(IT(I).EQ.1) THEN
      A=(X(II+4)+X(II+5)*X(II+3))*X(II+3)
      IF(STEEP(I)) THEN
      T=X(II+4)+2.*X(II+5)*X(II+3)
      ELSE
      P=X(II+4)+2.*SQRT(X(II+5)**2+1.)*X(II+3)
      ENDIF
      ELSE
      ARG=1.-2.*X(II+3)/X(II+4)
      BETA=ACOS(ARG)
      A=.25*X(II+4)**2*(BETA-SIN(BETA)*ARG)
      IF(STEEP(I)) THEN
      T=X(II+4)*SIN(BETA)
      ELSE
      P=BETA*X(II+4)
      ENDIF
      ENDIF
```

```
      IF(I.EQ.1) THEN
      E1=X(3)+(X(2)/A)**2/G2
      F(1)=X(1)-E1-KLOS(1)*(X(2)/A)**2/G2
      F(2)=X(2)
      JJ=7-IT(1)
      DO 10 J=2,NC
      F(2)=F(2)-X(JJ+2)
10    JJ=JJ+7-IT(J)
      ELSE
      IF(I.EQ.IGATE(IG)) THEN
      F(2*I-1)=E1-YGA(IG)-(1.+KLOS(I))*(X(II+2)/((X(II+4)
     &+X(II+5)*YGA(IG))*YGA(IG)))**2/G2-DZ(I)
      EYG(IG)=E1
      IYG(IG)=II+3
      IG=IG+1
      ELSE
      F(2*I-1)=E1-X(II+3)-(1.+KLOS(I))*(X(II+2)/A)**2/G2-DZ(I)
      ENDIF
      IF(STEEP(I)) THEN
      F(2*I)=T*X(II+2)**2-G*A**3
      ELSE
      F(2*I)=X(II+7-IT(I))*X(II+2)-CMAN*A*(A/P)**
     &.66666667*SQRT(X(II+1))
      ENDIF
      ENDIF
20    II=II+7-IT(I)
      RETURN
      END
```

Input data file to solve Example Problem 2.18

```
1
1
1
1
5. 531 3.961 15 2 .013 .2
.0008 206.5 4.589 8 1 .013 .2 0.
.0008 161.8 4.572 5 1 .013 .2 0.
.0008 161.8 4.752 5 1 .013 .2 0.
3 1.701
Q1 Y1 Q2 Y2 Q3 Y3 Q4 Y4

Input from keyboard
1 4 1 2 3
```

Program BRANCHCH.C

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float cman,g,g2,yga[5],eyg[5]; int ngate,iyg[5],igate[6];
extern void solveq(int n,float **a,float *b,int itype,\
   float *dd,int *indx);
void funct(int nc,int *it,int *steep,float *x,float *f,\
   float *dz,float *klos){
 int ii,ig,j,jj,i; float a,t,p,arg,beta,e1;
 ii=0; ig=0;
```

```
 for(i=0;i<nc;i++){
  if(it[i]){arg=1.-2.*x[ii+2]/x[ii+3]; beta=acos(arg);
    a=.25*x[ii+3]*x[ii+3]*(beta-sin(beta)*arg);
    if(steep[i])t=x[ii+3]*sin(beta); else p=beta*x[ii+3];}
  else {a=(x[ii+3]+x[ii+4]*x[ii+2])*x[ii+2];
    if(steep[i]) t=x[ii+3]+2.*x[ii+4]*x[ii+2];
    else p=x[ii+3]+2.*sqrt(x[ii+4]*x[ii+4]+1.)*x[ii+2];}
  if(i){
    if((i+1)==igate[ig]){
      f[2*i]=e1-yga[ig]-(1.+klos[i])*pow(x[ii+1]/((x[ii+3]+\
      x[ii+4]*yga[ig])*yga[ig]),2)/g2-dz[i];
      eyg[ig]=e1;iyg[ig++]=ii+3;}
    else f[2*i]=e1-x[ii+2]-(1.+klos[i])*\
      pow(x[ii+1]/a,2.)/g2-dz[i];
    if(steep[i]) f[2*i+1]=t*x[ii+1]*x[ii+1]-g*pow(a,3.);
    else f[2*i+1]=x[ii+5-it[i]]*x[ii+1]-cman*a*pow(a/p,\
      .6666667)*sqrt(x[ii]);}
  else {e1=x[2]+pow(x[1]/a,2.)/g2;
    f[0]=x[0]-e1-klos[0]*pow(x[1]/a,2.)/g2; f[1]=x[1];
    jj=6-it[0]; for(j=1;j<nc;j++){f[1]-=x[jj+1];jj+=6-it[j];}}
  ii+=6-it[i];}
} // end funct
void main(void){ int *it,*indx,*iv,*steep,keyb=0,screen=0,\
  ii,nc,n,nv=0,in,iout,i,j,iu,ipos,k,nct=0;
 float *x,*f,*f1,**d,*dz,*klos,xx,sum,ff,f11,dif,*dd;
 char ch2[2],*cu1,*cu2,*s[]={"SQYbmn","SQYDn "},ch,\
  *fmt=" %c =%8.3f\n",*fm1=" %c =%8.3f\n";
 FILE *fili,*filo; char filnam[20];
 printf("Give:(1) 1=ES,2=SI;(2) No. of channels;\
  (3) No. gates,(4) IN-unit,(5) OUT-unit\n");
 scanf("%d %d %d %d %d",&ii,&nc,&ngate,&in,&iout);
 igate[ngate]=nc+1;
 if((in==0) || (in==5)) keyb=1;
 else {printf("Give input file name\n"); scanf("%s",filnam);
 if((fili=fopen(filnam,"r"))==NULL){
  printf("\nFile %s cannot be opened\n",filnam);exit(0);}}
 if((iout==0) || (iout==6)) screen=1;
 else {printf("Give output file name\n");scanf("%s",filnam);
 if((filo=fopen(filnam,"w"))==NULL){
  printf("\nFile %s cannot be opened\n",filnam);exit(0);}}
 if(ii==2) {g=9.81;cman=1.;} else {g=32.2;cman=1.486;}
 g2=2.*g;n=2*nc;
 it=(int *)calloc(nc,sizeof(int));
 steep=(int *)calloc(nc,sizeof(int));
 klos=(float *)calloc(nc,sizeof(float));
 dz=(float *)calloc(nc,sizeof(float));
 f=(float *)calloc(n,sizeof(float));
 f1=(float *)calloc(n,sizeof(float));
 cu1=(char *)calloc(n,sizeof(char));
 cu2=(char *)calloc(n,sizeof(char));
 d=(float **)malloc(n*sizeof(float*));
 for(i=0;i<n;i++)d[i]=(float*)malloc(n*sizeof(float));
 if(keyb) printf("For each channel give: 1=Trap., or 2=cir.\n");
 for(i=0;i<nc;i++){
  if(keyb){printf("Channel # %2d =",i+1);scanf("%d",&it[i]);}
  else fscanf(fili,"%d",&it[i]); nv+=7-it[i]--;}
```

```
x=(float *)calloc(nv,sizeof(float));
iv=(int *)calloc(n,sizeof(int));
indx=(int *)calloc(n,sizeof(int));
for(i=0,ii=0;i<nc;i++){steep[i]=0;
 if(keyb){printf("Give variables for Channel %2d\n",i+1);
   for(j=0;j<(6-it[i]);j++){ch=s[it[i]][j];
      if((i==0) && (j==0)) ch='H';
      printf(" %c =",ch);scanf("%f",&x[j+ii]);}
   printf("Loss Coef =");scanf("%f",&klos[i]);
   if(i){printf("Change in bottom position =");
     scanf("%f",&dz[i]);}}
 else {
   if(i==0){
     for(j=0;j<(6-it[i]);j++) fscanf(fili,"%f",&x[j]);
     fscanf(fili,"%f",&klos[i]);}
   else {
     for(j=0;j<(6-it[i]);j++) fscanf(fili,"%f",&x[j+ii]);
     fscanf(fili,"%f %f",&klos[i],&dz[i]);
     if(x[ii]>.008)steep[i]=1;}}
 ii+=6-it[i];} // end for i
if(ngate>0){
 if(keyb) printf("Give pairs of values: channel no\
   and depth downs. from gate(s)\n");
 for(i=0;i<ngate;i++){
   if(keyb) scanf("%d %f",&igate[i],&yga[i]);
   else fscanf(fili,"%d %f",&igate[i],&yga[i]);}}
if(keyb)
 printf("Give symbols for %d Unknowns, i.e. Y1 Q1 b2 etc.\n",n);
for(i=0;i<n;i++){
 if(keyb) scanf("%s",ch2);
 else fscanf(fili,"%s",ch2);
 cu1[i]=ch2[0];cu2[i]=ch2[1];}
for(iu=0,i=0;i<n;i++){
 L30: ii=cu2[i]-49;
 for(j=0;j<(6-it[ii]);j++){
    if(s[it[ii]][j]==cu1[i]){
      ipos=0; for(k=0;k<ii;k++) ipos+=6-it[k];
      iv[iu]=ipos+j; goto L40;}}
 printf("Do not have %c%c as a variable\n",cu1[i],cu2[i]);
 printf("Give correct variable (z1=stop) ");
 scanf("%s",ch2);if(ch2[0]=='z')exit(0);
 cu1[i]=ch2[0];cu2[i]=ch2[1];goto L30; L40:iu++;} // end for i
do{funct(nc,it,steep,x,f,dz,klos);
 for(j=0;j<n;j++){xx=x[iv[j]];x[iv[j]]*=1.005;
    funct(nc,it,steep,x,f1,dz,klos);
    for(i=0;i<n;i++) d[i][j]=(f1[i]-f[i])/(x[iv[j]]-xx);
    x[iv[j]]=xx;}
 solveq(n,d,f,1,dd,indx); nct++; sum=0.;
 for(i=0;i<n;i++){x[iv[i]]-=f[i]; sum+=fabs(f[i]);}
 printf("Iteration=%d sum=%f\n",nct,sum);
 }while ((nct<20) && (sum>.0005));
for(i=0,nct=1;i<ngate;i++){ii=iyg[i]-1;
 L63:ff=eyg[i]-x[ii]-pow(x[ii-1]/((x[ii+1]+x[ii+2]*x[ii])*\
    x[ii]),2.)/g2;nct++;
 if((nct%2)==0){f11=ff;xx=x[ii];x[ii]*=1.005; goto L63;}
 dif=(x[ii]-xx)*f11/(ff-f11); x[ii]=xx-dif;
```

```
  printf("NCT= %d DIF=%f\n",nct,dif);
  if((nct<40) && (fabs(dif)>.0001)) goto L63;
  if(nct==40) printf("Not converge with gates\n");}
 s[it[0]][0]='H';
 for(i=0,ii=0;i<nc;i++){
  if(screen) printf("\nFor Channel %2d\n",i+1);
  else fprintf(filo,"\nFor Channel %2d\n",i+1);
  if(screen) printf(fm1,s[it[i]][0],x[ii]);
  else fprintf(filo,fm1,s[it[i]][0],x[ii]);
  for(j=1;j<6-it[i];j++){
     if(screen) printf(fmt,s[it[i]][j],x[j+ii]);
     else fprintf(filo,fmt,s[it[i]][j],x[j+ii]);}
  if(i==0) s[it[0]][0]='S'; fm1[8]='6';ii+=6-it[i];}
 if(keyb==0) fclose(fili); if(screen==0) fclose(filo);}
```

In solving the system of equations above that describe the flow rates and depths in several channels that branch off from a short main channel, it was assumed that critical depth at the entrance of the main channel does not occur and limit the flow rate into it. If the composite carrying capacities of the channels that branch off from this main channel exceed the capability of the main channel, then critical flow will occur at the entrance. When this occurs a valid solution to the system of equations does not exist. It is generally not easy to determine that a valid solution to the system of equations does not exist, however. An easier means for determining whether critical flow limits the flow rate is to complete the following steps:

1. Solve for the critical depth and flow rate from the reservoir into the main channel by using the energy and critical flow equations. By substituting the critical flow equation into the energy equation gives

$$F = Y + \frac{(1+K_e)A}{2T} - H = 0 \tag{2.16}$$

from which $Y = Y_c$ can be solved. With Y_c solved, the critical flow rate can be solved from

$$Q_c = \sqrt{\frac{gA^3}{T}} \tag{2.17}$$

2. Solve for the critical specific energy E_c from

$$E_c = Y_c + \frac{Q_c^2}{2gA_c^2}$$

If the entrance loss coefficient is zero ($K_e = 0$), then $E_c = H$ and this equation does not need to be solved.

3. Using the specific energy solve for the normal depths and flow rates in all of the branched channels from the two equations,

$$F_1 = E_c - Y_i - \frac{(1+K_{L1-i})Q_i^2}{2gA_i^2} - \Delta z_{1i} = 0 \tag{2.18}$$

$$F_2 = n_i Q_i P_i^{2/3} - C_u A_i^{5/3}\sqrt{S_o} = 0 \tag{2.19}$$

4. Sum the flow rates that the downstream branched channel can carry by $Q_{branches} = \Sigma Q_i$, with i = 2 to n. If $Q_{branches}$ is larger than Q_c then no valid solution to the equations exist, and Q_c limits the flow rate into the channels. With the flow rate limited into the downstream

branched channels they will not flow at normal depths even if no downstream control influences their upstream flow conditions based on the critical specific energy. Rather the limiting flow rate will force specific energy to be less.

The solution to branched channels whose flow is limited by critical depth in the main upstream channel at the reservoir can be based on the assumption that energy is lost in the main channel such that the specific energy at the junction with the branched channels is that needed to simultaneously satisfy all the energy equations at this junction and the uniform flow equations, in addition to the continuity equation. What happens in the real branched channel system is that just downstream from the entrance the flow will become supercritical with a hydraulic jump occurring in the main channel before the junction of the branched channels, or oblique jumps will occur at the junctions where the direction of the flow must change to enter the individual branch channels. In writing these equations the main channel will have a subscript 1, the first branched channel a subscript 2, etc. to the last with a subscript n. There will be 2(n − 1) + 1 equations needed to solve for n − 1 flow rates, n − 1 depths in the downstream channel, and the specific energy or head H_j at the junction. The head H_j will be less than the reservoir head H reduced by the local entrance loss $K_eQ_1^2/(2gA_1^2)$. These equation are

$$Q_2 + Q_3 + \cdots + Q_n - Q_c = 0 \quad \text{(Continuity)} \tag{2.20}$$

$$H_j - Y_2 - \frac{(1+K_{L1-2})Q_2^2}{2gA_2^2} - \Delta z_{1-2} = 0 \quad \text{(Energy)} \tag{2.21a}$$

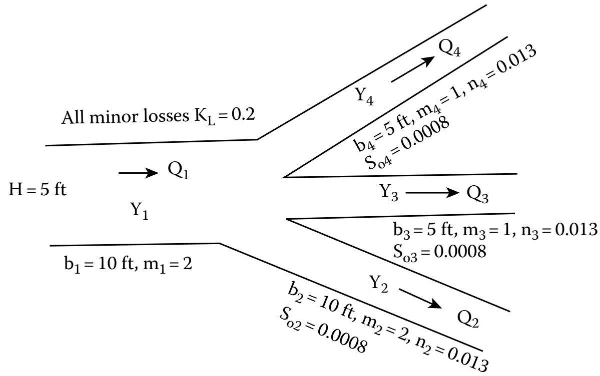
$$\frac{n_2Q_2}{C_u} - A_2\left(\frac{A_2}{P_2}\right)^{2/3}\sqrt{S_{o2}} = 0 \quad \text{(Uniform flow)} \tag{2.22a}$$

$$H_j - Y_3 - \frac{(1+K_{L1-3})Q_3^2}{2gA_2^2} - \Delta z_{1-3} = 0 \quad \text{(Energy)} \tag{2.21b}$$

$$\frac{n_3Q_3}{C_u} - A_3\left(\frac{A_3}{P_3}\right)^{2/3}\sqrt{S_{o3}} = 0 \quad \text{(Uniform flow)} \tag{2.22b}$$

$$H_j - Y_n - \frac{(1+K_{L1-n})Q_n^2}{2gA_n^2} - \Delta z_{1-n} = 0 \quad \text{(Energy)} \tag{2.21c}$$

$$\frac{n_{n3}Q_n}{C_u} - A_n\left(\frac{A_n}{P_n}\right)^{2/3}\sqrt{S_{on}} = 0 \quad \text{(Uniform flow)} \tag{2.22c}$$



EXAMPLE PROBLEM 2.19

A main channel with $b_1 = 10$ ft, and $m_1 = 2$ takes water from a reservoir with a head H = 5 ft above the channel bottom. A short distance downstream from its entrance this channel branches into three channels with the following properties: $b_2 = 8$ ft, $m_2 = 1$, $n_2 = 0.013$, $S_{o2} = 0.0008$, $b_3 = 5$ ft, $m_3 = 1$, $n_3 = 0.013$, $S_{o3} = 0.0008$, $b_4 = 5$ ft, $m_4 = 1$, $n_4 = 0.013$, $S_{o4} = 0.0008$. The bottoms of all channel at the junction are at the same elevation, i.e., all Δz values are zero, and the entrance loss coefficient K_e and loss coefficients between the main and the branched channel K_{L1-i} are all 0.2. Determine the flow rates and depths in the three branched channels.

Solution

Solving the energy and critical flow equations simultaneously at the entrance to the main channel from the reservoir gives: $Q_c = 531.4$ cfs, $Y_c = 3.509$ ft, and because of the loss coefficient $K_e = 0.2$, the specific energy associated with this critical flow is $E_c = 4.74$ ft. Next using a head of $H = E_c = 4.74$ the energy equation and Manning's equation are solve simultaneously for the three branched channels with the results: $Y_{o2} = 4.071$ ft, $Q_2 = 292.5$ cfs, $Y_{o3} = 4.157$ ft, $Q_3 = 211.4$ cfs, $Y_{o4} = 4.157$ ft, $Q_4 = 211.4$ cfs. These flow rates sum to 715.4 cfs, which exceeds the critical flow into the main channel. Therefore the flow will be limited to $Q_c = 531.4$ cfs. To determine the flow rates and depths in the branched channels the seven equations below are solved for the following seven unknowns: H_j, Y_2, Q_2, Y_3, Q_3, Y_4, Q_4 (H_j is the head at the junction).

$$F_1 = Q_2 + Q_3 + Q_4 - 531.4 = 0$$

$$F_2 = H_j - Y_2 - \frac{(1 + K_{L1-2})(Q_2/A_2)^2}{2g} - \Delta z_{1-2} = 0$$

$$F_3 = \frac{n_2 Q_2}{C_u} - A_2\left(\frac{A_2}{P_2}\right)^{2/3}(S_{o2})^{1/2} = 0$$

$$F_4 = H_j - Y_3 - \frac{(1 + K_{L1-3})(Q_3/A_3)^2}{2g} - \Delta z_{1-3} = 0$$

$$F_5 = \frac{n_3 Q_3}{C_u} - A_3\left(\frac{A_3}{P_3}\right)^{2/3}(S_{o3})^{1/2} = 0$$

$$F_6 = H_j - Y_4 - \frac{(1 + K_{L1-4})(Q_4/A_4)^2}{2g} - \Delta z_{1-4} = 0$$

$$F_7 = \frac{n_4 Q_4}{C_u} - A_4\left(\frac{A_4}{P_4}\right)^{2/3}(S_{o4})^{1/2} = 0$$

The solution to these seven equations gives: $H_j = 4.04$ ft, $Q_2 = 219.6$ cfs, $Y_2 = 3.467$ ft, $Q_3 = 155.9$ cfs, $Y_3 = 3.540$ ft, $Q_4 = 155.9$ cfs, $Y_4 = 3.540$ ft. Thus the energy loss in the main channel from its entrance to the branched channel must equal $\Delta H = 4.74 - 4.04$ ft = 0.70 ft-lb/lb. This solution was obtained using TK-Solver. It can also be obtained with the program above by specifying the flow rate in the upstream channel as the unknown, and indicating that S1 is unknown for the upstream main channel. The reason for specifying S1 as unknown, rather than H1, is that the program is written so that for all channels, except the first, variable S_o, the bottom slope, is in the first position of the array x for the variables for that channel. For the first, or main, channel the head H is in this first position, but the logic to note this does not exist in the program. Thus S1 actually is interpreted as H when given for channel # 1. The input to the above program BRANCHCH and its solution are given below. The real occurrence for a situation like this for which critical conditions limit the flow rate is that this limiting flow reduces the energy available at the junction below that of the reservoir.

Input to BRANCHCH for critical flow at entrance

```
1
1
1
1
4.6 531.41 3.882 10 2 .013 .2
0.0008 217.3 4. 8 1 .013 .2 0.
0.0008 157.05 4. 5 1 .013 .2 0.
0.0008 157.05 4. 5 1 .013 .2 0.
S1 Y1 Q2 Y2 Q3 Y3 Q4 Y4
```

Solution for critical flow at entrance

For Channel 1
H = 4.038
Q = 531.410
Y = 3.875
b = 10.000
m = 2.000

For Channel 2
S = .000800
Q = 219.610
Y = 3.467
b = 8.000
m = 1.000
n = .013

For Channel 3
S = .000800
Q = 155.900
Y = 3.540
b = 5.000
m = 1.000
n = .013

For Channel 4
S = .000800
Q = 155.900
Y = 3.540
b = 5.000
m = 1.000
n = .013

In obtaining this solution with program BRANCHCH you will notice that difficulties exist in achieving convergence. As soon as the residual becomes small, it get large the next iteration. This numerical action results because BRANCHCH also attempts to solve $Y_1 = Y_c$ associated with the critical flow rate Q_1, which is specified. The infinite derivative of the depth in the upstream channel causes the Newton method to move off the desired solution as Y_1 approaches Y_{c1}. To solve this problem adequately BRANCHCH should be modified so that Y_1 is not included as an unknown, i.e., solve 7 rather than 8 equations simultaneously.

2.9 GRAPHICAL AIDS TO SOLVING CRITICAL FLOW PROBLEMS

Since the critical flow equation (with the exception for rectangular channels), and the specific energy equation are implicit when solving them for the depth, or channel size graphical methods for solving these problems are widely used in practice. The need for such graphical solutions is rapidly diminishing with the wide spread use of programmable pocket calculators, and PC computers. However, even with these modern tools it is often advantageous to obtain a "quick" graphical solution, in examining proposed alternatives.

To examine how graphs for solving critical flow equations might be developed the critical flow equation will be written specifically for a trapezoidal channel as

$$\frac{Q^2(b+2mY_c)}{g(bY_c+mY_c{}^2)^3}=\frac{Q^2b(1+2mY_c/b)}{gb^6\left\{Y_c/b+m(Y_c/b)^2\right\}^3}=\frac{\left\{Q^2/(gb^5)\right\}(1+2mY_c/b)}{\left\{Y_c/b+m(Y_c/b)^2\right\}^3}=1 \tag{2.23}$$

In the last form of this equation it should be noted that $Q^2/(gb^5)$ is a dimensionless parameter, which can be denoted as Q'. Likewise the ratio Y_c/b is dimensionless and can be denoted by Y'. Thus the last part of the above equation can be written as

$$Q'=\frac{\{Y'+mY'^2\}^3}{1+2mY'} \tag{2.24}$$

or the dimensionless flow rate Q' is a function of the dimensionless depth Y', and the side slope of the trapezoidal channels. This equation is in a form in which it is relative easy to generate the values needed to plot Q' versus Y' for selected values of m. Such a plot is given in Figure 2.6. This figure also includes a similar curve for circular channels.

To use Figure 2.6 to determine the critical depth the value of $Q' = Q^2/(gb^5)$ is computed from the flow rate, and bottom width of the channel. This value is entered on the abscissa of the graph, and projected vertically upward to the curve for the known side slope of the channel. The dimensionless critical depth $Y' = Y_c/b$ is next read from the graph, and from this value the critical depth determined.

It was convenient to leave the side slope m out of the definition of Q' in Equation 2.24, so different curves could be plotted on Figure 2.6. Note, however, if the definition for dimensionless depth $Y' = mY/b$ and $Q' = m^3Q^2/(gb^5)$ for a trapezoidal channel then Equation 2.23 can be written as

$$Q_c'=\frac{Y_c'+Y_c'^2}{1+2Y_c'}\quad\left(\text{with } Y_c'=\frac{mY_c}{b}\text{ and } Q_c'=\frac{m^3Q_c^2}{gb^5}\right) \tag{2.24a}$$

(These latter definitions for dimensionless variables will be used below.)

To obtain the curve for circular channels on Figure 2.6 write the critical flow equation for a circle using the auxiliary angle $\beta = \cos^{-1}(1 - 2Y_c/D)$ as the variable to give

$$\frac{Q^2D\sin\beta}{gD^6(\beta-\sin\beta\cos\beta)^3/64}=1 \tag{2.25}$$

Letting $Q' = Q^2/(gD^5)$ be a dimensionless flow rate, this equation becomes

$$Q'=\frac{(\beta-\sin\beta\cos\beta)^3}{64\sin\beta}$$

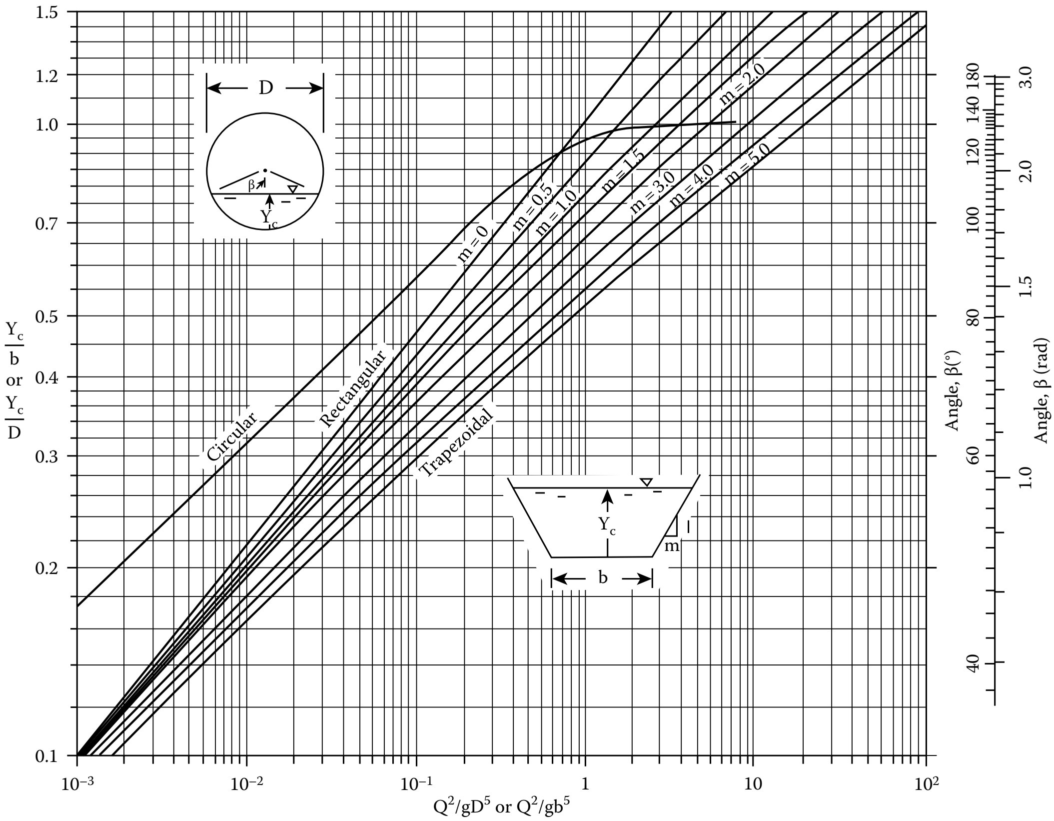


FIGURE 2.6 Relationship of dimensionless depth to dimensionless flow rate parameters under "critical flow" conditions.

This equation is in the form in which it is relatively easy to generate a table of values for Q′ corresponding to the dimensionless β and since $Y' = Y_c/D = (1 - \cos\beta)/2$ this table can be expanded to give values of Q′ versus Y′. Thus a single curve defines critical flow for all circular sections. You should note that the first portion of this curve on the log–log plot of Figure 2.6 is nearly a straight line. Thus Y′ or β can be approximated quite closely by a power function of Q′ in the form a Q'^b. This power function was discussed earlier for providing starting values to solve for critical depth in circular sections by the Newton method. It is also possible to develop dimensionless specific energy diagrams that provide solutions for critical flow conditions as well as other problems associated with the use of the specific energy principle in open channel flow. Figures 2.7 and 2.8 provide such dimensionless plots for trapezoidal and circular channels respectively. If one wants to determine the critical depth by using these figures, one needs only compute the value of Q′ (which you should note is defined differently, as is Y′, than in Figure 2.6 for a trapezoidal channel to include m) and read the ordinate Y′ corresponding to the minimum value for the curve for that Q′. The flow rate can be determined by reading the value from the abscissa corresponding to the minimum value of this curve.

In addition, it is relatively simple to solve for an alternate depth. For example, if the supercritical depth is known and its alternate subcritical depth is desired, one need only compute Q′ and Y′ for the supercritical depth, enter this Y′ on the ordinate, and project horizontally to the curve corresponding to Q′. Next project vertically upward to this same curve, and then horizontally over the ordinate where the subcritical value for Y′ is read.

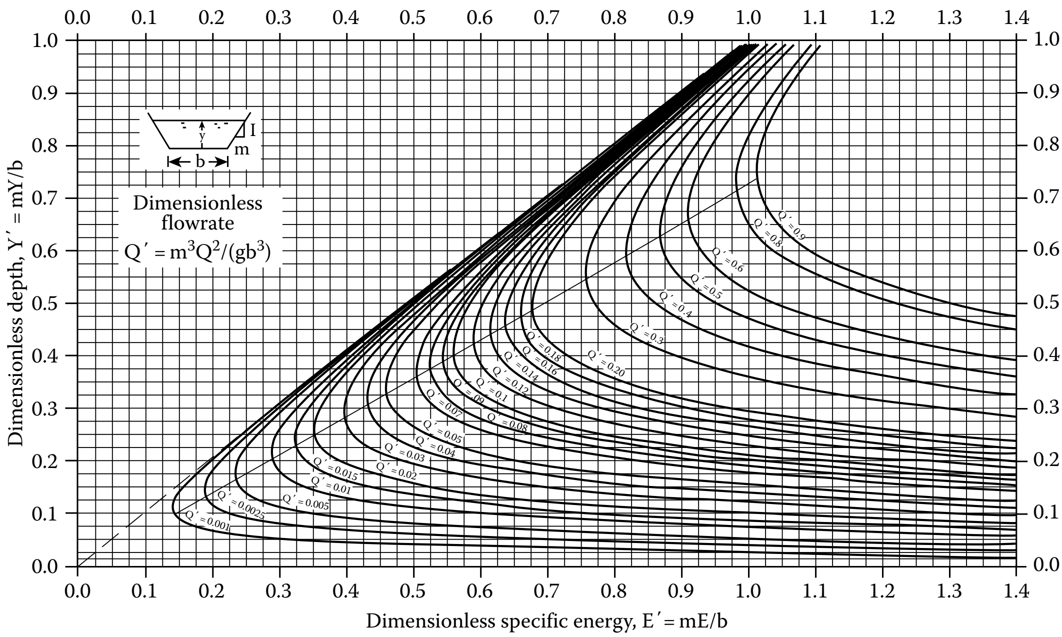


FIGURE 2.7 Dimensionless specific energy diagram trapezoidal channels. (Individual curves apply for $Q' = m^3Q^2/(gb^5)$.)

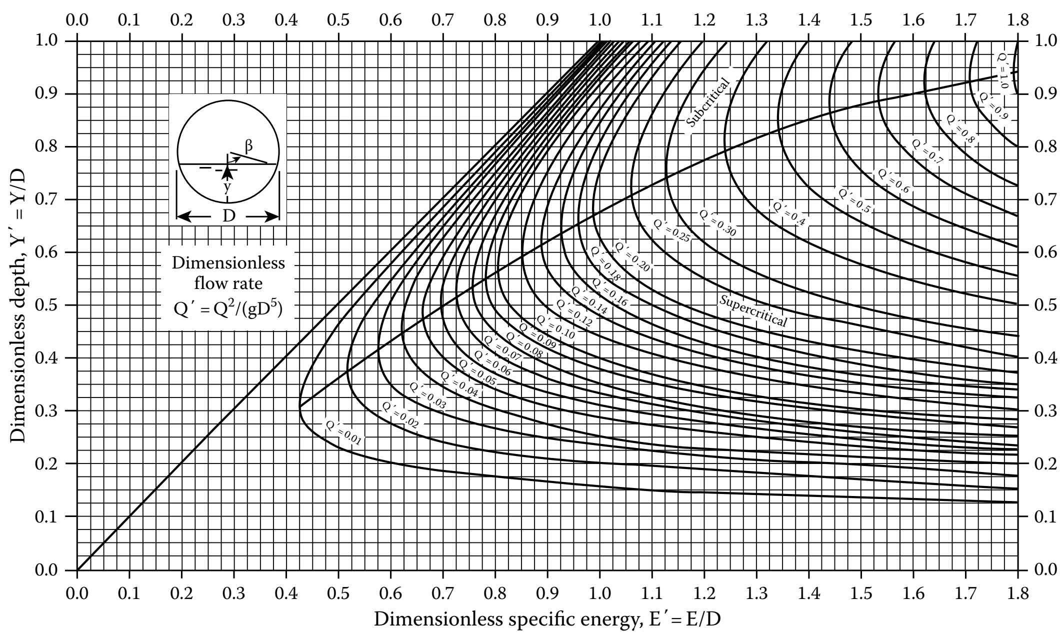


FIGURE 2.8 Dimensionless specific energy diagrams for circular channels. (Individual curves apply for $Q' = Q^2/(gD^5)$.)

To determine how Figure 2.7 is developed write the specific energy equation specifically for a trapezoidal channel as shown below.

$$E = Y + \frac{Q^2}{2g(bY + mY^2)^2} = Y + \frac{Q^2}{2gb^4[(y/b) + m(Y/b)^2]^2} \tag{2.26}$$

If this equation is multiplied by m and divided by b, and the dimensionless specific energy mE/b defined as E′, and mY/b denoted as a dimensionless depth Y′ then the following results:

$$E' = Y' + \frac{m^2Q^2}{2gb^5(mY/b + (mY/b)^2)^2} = Y' + \frac{Q'}{2(Y' + Y'^2)^2} \tag{2.27}$$

in which $Q' = m^3Q^2/(gb^5)$. Thus it is clear that dimensionless specific energy E′ can be defined as a function of dimensionless depth Y′ with dimensionless flow rate Q′ held constant, i.e., for any Q′ the dimensionless specific energy curve can be defined, as has been done in Figure 2.7.

Figure 2.8 for a circular section can be obtained in a similar manner. The difference is that the specific energy equation is written specifically for a circular channel using angle β, as follows:

$$E = \frac{D(1 - \cos\beta)}{2} + \frac{Q^2}{gD^4(\beta - \sin\beta\cos\beta)^2/8} \tag{2.28}$$

by dividing this equation by D, and defining $Q' = Q^2/(gD^5)$, the above equation becomes,

$$E' = \frac{1 - \cos\beta}{2} + \frac{8Q'}{(\beta - \sin\beta\cos\beta)^2} = Y' + \frac{8Q'}{(\beta - \sin\beta\cos\beta)^2} \tag{2.28a}$$

and since $\cos\beta = 1 - 2Y/D = 1 - 2Y'$ it is possible to develop a dimensionless specific energy curve for any dimensionless value of Q′. Figure 2.8 represents such a graph.

EXAMPLE PROBLEM 2.20

A transition occurs from a 2 m diameter pipe to one of 1.5 m diameter. The transition keeps the centerlines of the two pipes lined up. For a flow rate of $Q = 2\,m^3/s$, and a bottom slope of $S_o = 0.00095$, and $n = 0.013$ for the downstream 1.5 m diameter pipe, determine the depth of flow just upstream from the transition. Solve this problem graphically, and then verify the solution.

Solution

The solution must begin by solving a uniform flow equation for the downstream 1.5 m diameter channel. Using Manning's equation the uniform, or normal, depth in the downstream channel is $Y_{o2} = 1.131$ m. Dividing this by the downstream diameter of 1.5 m gives $Y' = 0.754$. Next compute $Q'_2 = 2^2/(9.81(1.5)^5) = 0.054$. Entering the ordinate of Figure 2.8 with 0.754 and finding the intersection with the curve $Q' = 0.054$ gives $E' = 0.82$ on the abscissa. Therefore $E_2 = 1.5(0.82) = 1.23$ m. The specific energy in the larger pipe just upstream from the transition must be $E_1 = E_2 + \Delta z = 1.23 + 0.25 = 1.48$ m, and therefore $E'_1 = 0.74$. Also $Q'_1 = 2^2/(9.81(2)^5) = 0.013$. Entering Figure 2.8 with these latter two values produces , $Y' = 0.725$ or $Y_1 = 1.45$ m.

A numerical solution to the problem consists of solving the following implicit equation for Y_1:

$$E_1 = E_2, \quad \text{or} \quad Y_1 + \frac{Q^2}{2gA_1^2} = Y_2 + \frac{Q^2}{2gA_2^2} + \Delta z$$

in which A_2 can be evaluated from the known depth of 1.131 m, and the area A_1 must be defined in terms of Y_1 and/or the auxiliary angle β. Solving this implicit equation by the Newton method gives: $Y_1 = 1.446$ m.

EXAMPLE PROBLEM 2.21

What is the smallest pipe that could be used downstream in Example Problem 2.20 if the slope were sufficient to carry the flow that could pass through the transition and upstream conditions are not to be changed?

Solution

Critical flow will occur at the end of the transition in the entrance of the smaller downstream pipe. If the depth upstream is to remain the same as in Example Problem 2.20, then the critical specific energy for the downstream channel will be $E_c = E_1 - (D_1 - D_2)/2 = 1.48 - (2 - D_2)/2 = 0.48 + D_2/2$. This equation involves two unknowns; D_2 and Y_2 since the critical specific energy E_c is the sum of the depth and the velocity head. The second needed equation to solve the problem is the critical flow equation applied to the downstream channel, or $Q^2T/(gA^3) = 1$. Solving these two implicit equations simultaneously gives $D_2 = 1.237$ m and $Y_2 = 0.770$ m.

The above dimensionless equations that were used to plot Figures 2.7 and 2.8 for trapezoidal and circular channels, respectively, can be used to develop dimensionless equations that relate the critical dimensionless flow rate, Q'_c to the critical dimensionless depth Y'_c and the critical dimensionless energy E'_c. These equations can be used to solve more readily for critical flow conditions than the critical flow equation. Furthermore, explicit equations can be developed that closely approximate these equations, and these approximate equations can be used for rough answers, or can be used to provide starting guesses for the Newton method to solve the critical flow equation.

First consider the dimensionless specific energy in a trapezoidal channel, i.e., $E' = Y' + Q''/(Y' + Y'^2)^2$ (Note that Q'' is 1/2 the Q' used in Figures 2.7 and 2.8, or $Q'' = Q'/2$). Taking the derivative of E' with respect to Y', and setting this derivative to zero produces the following dimensionless equation for Q'' after a little mathematical rearrangement:

$$Q''_c = 0.5\frac{(Y'_c + Y'^2_c)^3}{1+2Y'_c} = \frac{0.5Y'^3_c(1+Y'_c)^3}{1+2Y'_c} \tag{2.29}$$

the subscript c has been added to Y' and Q'' because setting the derivative to zero produces critical flow conditions. Figure 2.9 is a log–log plot of this equation, and also the above dimensionless critical specific energy equation. Note that the curves on this graph can be approximated by straight lines. The equations for these approximate straight line fits are:

$$Y'_c = 0.925(Q''_c)^{0.284} \tag{2.30}$$

and

$$E'_c = 1.238(Q''_c)^{0.272} \tag{2.31}$$

The same can be accomplished for a circular open channel. Taking the derivative of the equation $E' = Y' + 8Q'/(\beta - \cos\beta\sin\beta)^2$ with respect to Y' and setting this derivative to zero gives the following equation after some algebraic manipulations:

$$Q'_c = \frac{(\beta - \cos\beta\sin\beta)^3}{(64\sin\beta)} \quad \text{(Also given below Equation 2.25)} \tag{2.32}$$

in which β is the angle associated with the dimensionless critical depth $Y'_c = Y_c/D$, or $\beta = \cos^{-1}(1-2Y'_c)$. This equation as well as E'_c are plotted on Figure 2.10. Again note that except as Q_c and Y_c approach unity these relationships are approximated by a straight lines on this log–log graph. The equations for such straight lines fits are

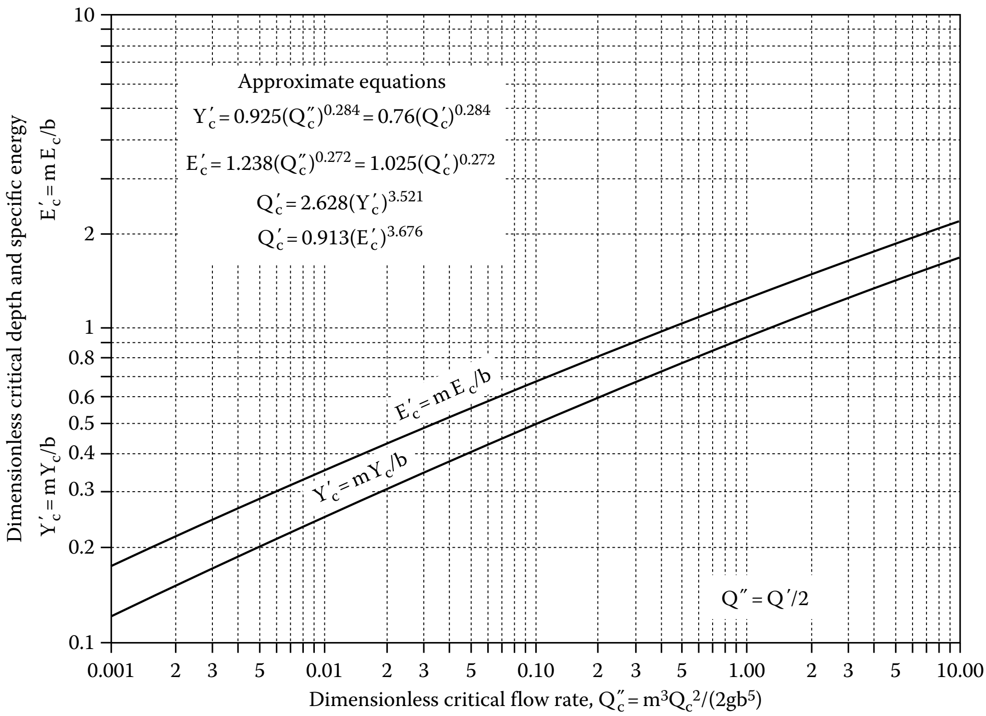


FIGURE 2.9 Critical condition in a trapezoidal channel.

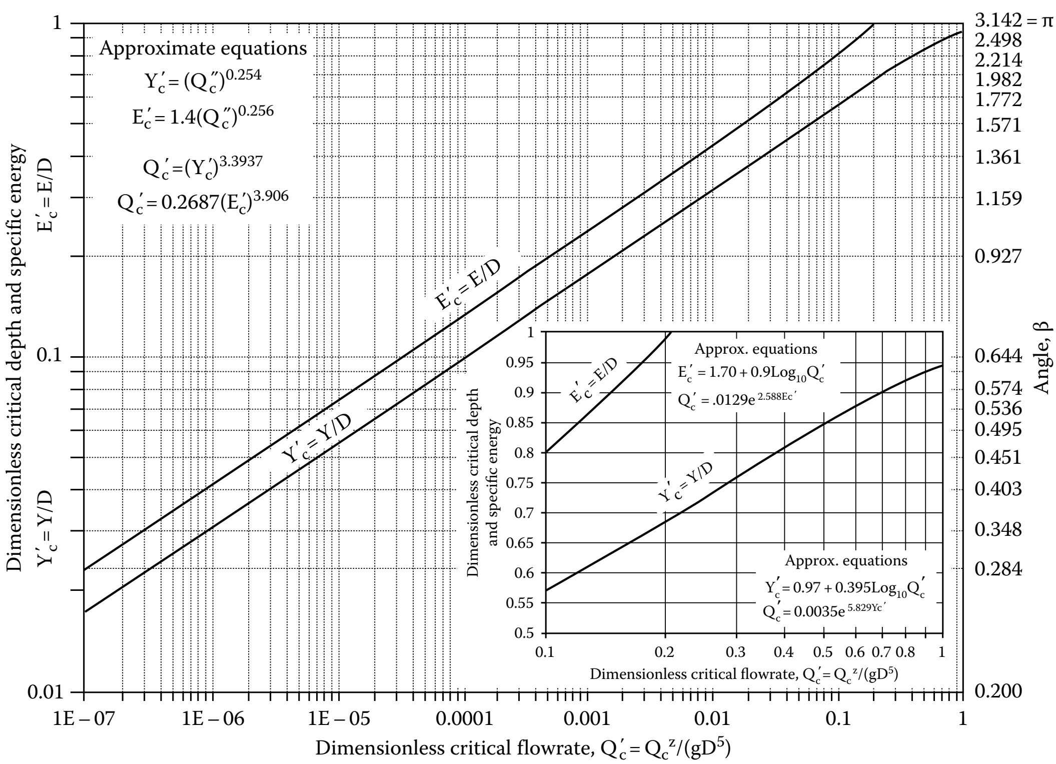


FIGURE 2.10 Critical condition in a circular channel.

$$Y'_c = (Q'_c)^{0.254} \tag{2.33}$$

and

$$E'_c = 1.4(Q'_c)^{0.256} \tag{2.34}$$

The insert in the lower right hand corner of Figure 2.10 shows Y'_c and E'_c plotted against Q'_c on semi-logarithmic paper that starts with a value of $Q'_c = 0.1$. On this insert a straight line plot for these curves are given by the following equations:

$$Y'_c = 0.97 + 0.395 \log_{10}(Q'_c) \quad \text{or} \quad Q'_c = 0.0035e^{5.829Yc'} \tag{2.35}$$

and

$$E'_c = 1.70 + 0.9 \log_{10}(Q'_c) \quad \text{or} \quad Q'_c = 0.0129e^{2.558Ec'} \tag{2.36}$$

If the channel is rectangular then another definition of the dimensionless depth needs to be used since multiplication of Y by m will always produce a zero dimensionless depth. A useful dimensionless depth is defined by dividing the depth by the critical depth Y_c or defining $Y' = Y/Y_c$. If the dimensionless specific energy E' is also defined by dividing E by Y_c, then the dimensionless specific energy equation for a **rectangular** channel becomes,

$$E' = Y' + 0.5/Y'^2 \text{ or the cubic equation } Y'^3 - E'Y'^2 + 0.5 = 0$$

The methods described earlier can be used to solve this resulting cubic equation for the three real roots that exist when E' is greater, or equal, to 1.5. Using the alternate method that uses the arc cosine (and cos) results in defining the angle θ as

$$\theta = \cos^{-1}\left\{\frac{\left[(6.75 - E'^3)/27\right]}{(E'/3)^3}\right\}$$

with the three roots given by

$$Y'_1 = \left(\frac{E'}{3}\right)\left[1 - 2\cos\left(\frac{\theta}{3}\right)\right]$$

$$Y'_2 = \left(\frac{E'}{3}\right)\left[1 - 2\cos\left(\frac{[\theta + 2\pi]}{3}\right)\right]$$

$$Y'_3 = \left(\frac{E'}{3}\right)\left[1 - 2\cos\left(\frac{[\theta + 4\pi]}{3}\right)\right]$$

The program ROOTSED, given below obtains these three roots for any given value for the dimensionless depth Y'. By modifying this program to include a DO loop one could easily generate data to provide a graph of the alternate dimensionless depths as a function of the dimensionless specific energy.

Program ROOTSED.FOR

```
      PARAMETER (PI=3.14159265)
1     WRITE(*,*)' Give: E''=E/Yc must be 1.5 or greater'
      READ(*,*) Ep
      IF(Ep.LT. 1.E-5) STOP
      E3=Ep/3.
      THETA=ACOS(((6.75-Ep**3)/27.)/E3**3)
      Y1=E3*(1.-2.*COS(THETA/3.))
      Y2=E3*(1.-2.*COS((THETA+2.*PI)/3.))
      Y3=E3*(1.-2.*COS((THETA+4.*PI)/3.))
      WRITE(*,100) Y1,Y2,Y3
100   FORMAT(' Y1=',F8.4,' Y2=',F8.4,' Y3=',F8.4)
      GO TO 1
      END
```

For rectangular channels, it was useful to know that under critical flow the depth equals two-thirds of the critical specific energy, or $Y_c = 2E_c/3$. What this equation represents is the flow rate eliminated between the critical flow equation and the specific energy equation. While not as simple, the same can be done for nonrectangular channels by solving for Q from the critical flow equation and substituting for it into the energy equation, giving Equation 2.16, or

$$H_c = Y_c + \frac{(1+K_e)A_c}{(2T_c)} \tag{2.37}$$

For a trapezoidal channel, let the dimensionless critical head be defined as $H'_c = mH_c/b$ (the same definition as for the dimensionless specific energy) and $Y'_c = mY_c/b$, then the dimensionless form of the critical flow equation that relates head to dimensionless critical depth is

$$H'_c = Y_c + (1+K_e)\frac{Y' + Y'^2}{2(1+2Y')} \tag{2.38}$$

This equation could also be obtained by eliminating Q' between the dimensionless critical flow and energy equations for a trapezoidal channel. Figure 2.11 shows this relationship with different curves on it for different values for the entrance loss coefficient, K_e. If $K_e = 0$ the distance between the E'_c and Y'_c curves could be added to Y'_c. Or stated differently the results on Figure 2.11 along the $K_e = 0$ curve can be obtained from Figure 2.9, by entering the ordinate for Y'_c and then move horizontally over to this curve; then move vertically up to the E'_c curve, and finally read the ordinate for E'_c.

For circular channels define the critical dimensionless head as $H'_c = H_c/D$ (and as before $Y'_c = Y_c/D = (1-\cos\beta)/2$, and then the dimensionless version of Equation 2.37 is,

$$H'_c = \frac{1-\cos\beta}{2} + \frac{(1+K_e)(\beta - \cos\beta\,\sin\beta}{8\sin\beta} = Y'_c + \frac{(1+K_e)(\beta - \cos\beta\,\sin\beta}{8\sin\beta} \tag{2.39}$$

Figure 2.12 shows this dimensionless relationship.

Generally problems involving the use of Equations 2.38 or 2.39 have the total H head given. So the first step is to determine the dimensionless H'_c, and from this value find the corresponding value for the critical depth Y'_c. Whether the channel is trapezoidal or circular the implicit Equations 2.38 or 2.39 can be solved for Y'_c (with H'_c known), rather than using the graphs. Once Y'_c is determined then the dimensionless flow rate can be computed from Equation 2.29 (or Equation 2.24) for trapezoidal channels, or Equation 2.32 for circular channels.

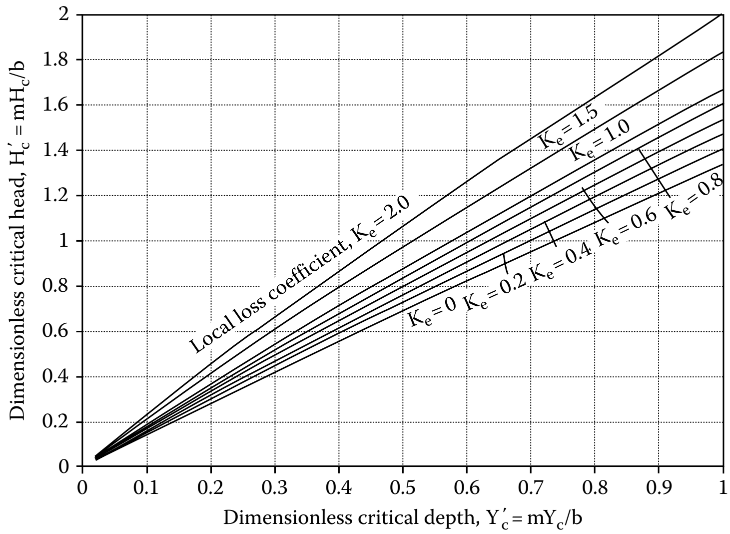


FIGURE 2.11 Dimensionless critical depth at the beginning of a trapezoidal channel feed by a reservoir.

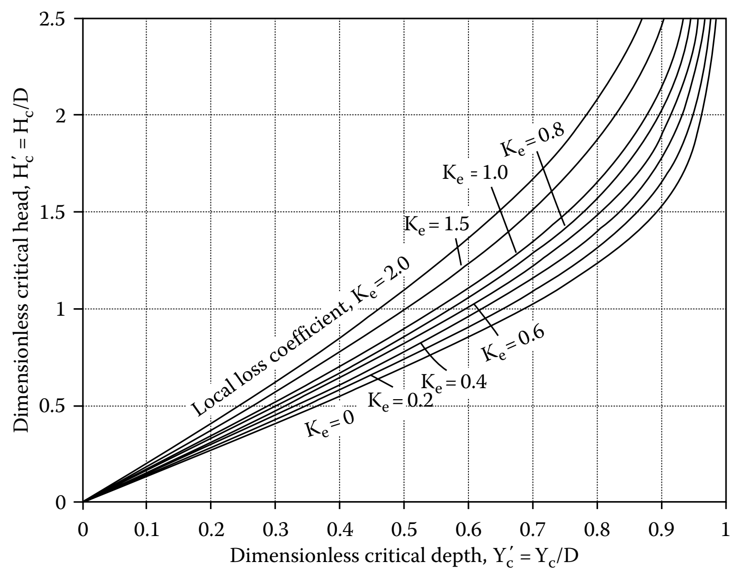


FIGURE 2.12 Dimensionless critical depth at the beginning of a circular channel feed by a reservoir.

Often problems whose solutions are based on the energy principle are design problems that call for sizing the channel that will carry a specified flow rate. The bottom slope is known, or dictated by the general slope of the land over which the channel is to carry the water, and the roughness coefficient is determined by the type of material the channel will be built from. Thus if the channel is trapezoidal in shape the bottom width and/or the side slope are/is the unknown variable(s), or if the channel is circular then it is the diameter that is sought after. Basically such problems are not different than those we have dealt with previously, e.g., they involve the simultaneous solution to a couple of implicit equations. The difference is that for this type of problem Q is known, and the size variable, and the depth of flow are generally the unknown variables. The two equations that need to be solved are either (1) a uniform flow equation, and the energy equation if the flow is subcritical, or (2) the critical flow equation and the energy equation if the situation will create a critical flow section.

EXAMPLE PROBLEM 2.22

It is desired to find the pipe diameter that would be carry a flow rate of 30 m³/s from a reservoir whose water surface elevation will be 4.5 m above the bottom of the channel at its entrance. The slope of the channel is $S_o = 0.001$, and it Manning's roughness coefficient is n = 0.013. The entrance loss coefficient is estimated to equal 0.12.

Solution

Since this is a mild slope the governing equations are the energy equation and Manning's (or Chezy's) equation, or

$$F_1 = H - Y - \frac{(1+K_L)Q^2}{2gA^2} = 0$$

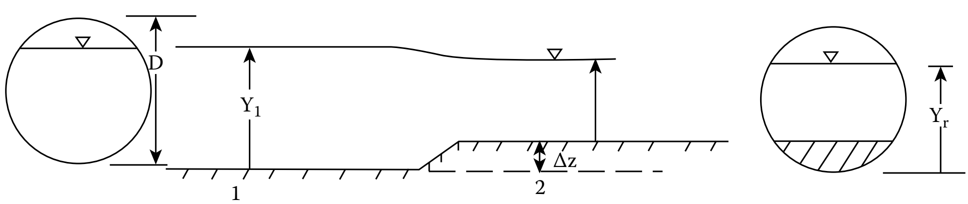
and

$$F_2 = nQP^{2/3} - A^{5/3}\sqrt{S} = 0$$

in which the area A and the perimeter P involve the second unknown D. The solution gives Y = 3.047 m and D = 4.152 m. You should verify these answers, by at least substituting into the above two equations, but better still solve these two equations using the Newton method, or by utilizing an available software package such as TK solver, MATLAB, or Mathcad.

EXAMPLE PROBLEM 2.23

A pipe has its bottom filled at section 2 to a depth of Δz, as shown in the sketch. Using dimensionless variables generate a graph that gives the maximum dimensionless height of hump for given upstream flow conditions. The diameter of the pipe remains constant between section 1 and 2, only the rise in the bottom reduces the size of the pipe at section 2.



Solution

The specific energy and continuity equations are written in dimensionless form below by dividing length variables by the pipe diameter D and area by the diameter squared D².

$$Y_1' + F_{r1}{}^2\left(\frac{Y_{d1}'}{2}\right) = Y_r' + F_{r2}{}^2\left(\frac{Y_{d2}}{2}\right) = Y_r' + \frac{F_{r2}{}^2(A_r' - A_b')}{(2T_r')}$$

in which subscript 1 denotes section 1, subscript r denotes section 2 as if the step did not exist, and subscript b denotes the bottom step. A is area and Y_d is the hydraulic depth, or area divided by the top width ($A' = A/D^2$ and $Y_d' = Y_d/D$). The continuity equation becomes:

$$A_1' F_{r1} Y_{d1}'^{0.5} = \frac{F_{r2}(A_r' - A_b')^{1.5}}{T_r'^{0.5}}$$

Define the angle $\beta = \cos^{-1}(1 - 2Y')$, then $A' = (\beta - \cos\beta \sin\beta)/4$, $T' = \sin\beta$. (These apply to define the dimensionless area at section 1, for the bottom step, or the regular area, sub r at section 2 ignoring the step.) The TK-Solver Variable and Rule Sheets are given below.

```
═══════════════════ VARIABLE SHEET ══════════ (Model PAPER6.TK or PAPER*.TK for other Y1')
St  Input────  Name───  Output───  Unit─────  Comment─────────
    .6         Y1                             Upstream dimensionless depth
    1.7721542  B1                             Upstream angle β
    .49202836  A1                             Upstream Area
    .50217434  Yd1                            Upstream Hydraulic Depth
L   .01        Fr2                            Upstream Froude No.squared (Fr(2) = 1)
LG  .54897512  Yr                             Depth from bot. circle to water at sect. 2
LG  .44221115  Yb                             Depth from bot. circle to rise
LG  1.6689039  Br
LG  .44159578  Ar
LG  1.4549598  Bb
LG  .33503915  Ab
L              r6
L              Fr

═══════════════════ RULE SHEET ═════════════════════════
S Rule──────────────────────────────────────────────────
* A1*sqrt(Yd1*Fr2)=(Ar-Ab)*sqrt((Ar-Ab)/sin(Br))
* Y1+Fr2*Yd1/2=Yr+.5*(Ar-Ab)/sin(Br)
* Br=acos(1-2*Yr)
* Ar=.25*(Br-cos(Br)*sin(Br))
* Bb=acos(1-2*Yb)
* Ab=.25*(Bb-cos(Bb)*sin(Bb))
* r6=Yb/Y1
* Fr=sqrt(Fr2)
```

```
Table Sheet
═══════════════════ TABLE: ══════════════════════════
Title:
Element   Yr────   Yb────   Br────   Ar────   Bb────   Ab────   Fr────   r6────   r7────
1         .54898   .44221   1.6689   .4416    1.455    .33504   .1       .73702   .91496
2         .53184   .38071   1.6345   .42452   1.3299   .27455   .16673   .63452   .88641
3         .5229    .34336   1.6166   .41559   1.2522   .23867   .21354   .57227   .87149
.           .        .        .        .        .        .        .        .        .
24        .5198    .1015    1.6104   .4125    .6485    .04178   .64761   .16917   .86634
25        .52166   .09599   1.6141   .41436   .63      .03849   .66121   .15998   .86944
 .           .        .        .        .        .        .        .        .        .
42        .56063   .02814   1.6923   .45318   .3371    .00624   .86012   .0469    .93438
43        .56321   .02536   1.6975   .45574   .31983   .00534   .8704    .04226   .93868
.           .        .        .        .        .        .        .        .        .
59        .59067   .0036    1.7531   .48287   .12      .00029   .96954   .00599   .98444
60        .59144   .00319   1.7547   .48363   .1131    .00024   .97211   .00532   .98573
61        .59221   .00281   1.7563   .48439   .106     .0002    .97468   .00468   .98702
```

A similar Mathcad model EXPRB2~1.MCD is listed below (with only a top portion of the range variables given by changing Fr2:=0.01,0.03…0.12)

Variables Y1:=6 B1:=1.7721542 A1:=.49202836 Yd1:=.5 Y_r:=.5
Yb:=.4 Br:=1.7 Ar:=.4 Bb:=1.5 Ab:=.3

Given

$$A1\sqrt{(Yd1\cdot Fr2)} = (Ar - Ab)\cdot\sqrt{\frac{Ar - Ab}{\sin(Br)}}$$

$$Y1 + Fr2\cdot\frac{Yd1}{2} = Yr + .5\cdot\frac{Ar - Ab}{\sin(Br)}$$

$$\cos(Br) = 1 - 2\cdot Yr$$

$$Ar = .25\cdot(Br - \cos(Br)\cdot\sin(Br))$$

$$\cos(Bb) = 1 - 2\cdot Yb$$

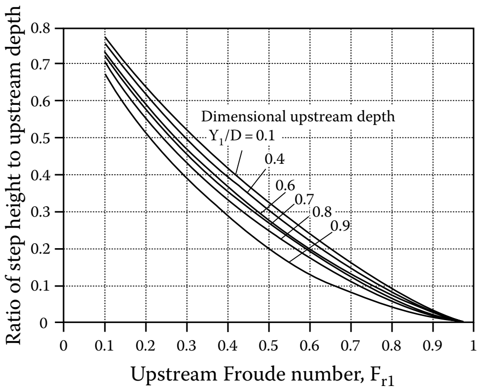
$$Ab = .25\cdot(Bb - \cos(Bb)\cdot\sin(Bb))$$

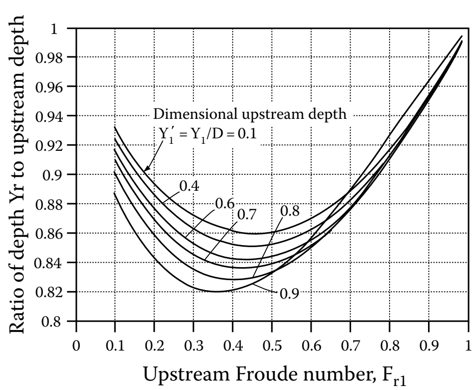
F(Fr2):=Find(Yr,Yb,Br,Ar,Bb,Ab)

Fr2:=0.01,0.03..0.12

$F(Fr2)_0$	$F(Fr2)_1$	$F(Fr2)_2$	$F(Fr2)_3$	$F(Fr2)_4$	$F(Fr2)_5$	$\sqrt{Fr2}$	$\frac{F(Fr2)_1}{Y1}$
0.549	0.442	1.669	0.442	1.455	0.335	0.1	0.737
0.531	0.376	1.632	0.423	1.319	0.27	0.173	0.626
0.521	0.336	1.613	0.414	1.237	0.232	0.224	0.56
0.516	0.307	1.602	0.408	1.174	0.204	0.265	0.511
0.512	0.283	1.594	0.404	1.122	0.183	0.3	0.472
0.509	0.263	1.589	0.402	1.076	0.165	0.332	0.438

The variables listed are used to obtain the curve for an upstream dimensionless depth of 0.6. To get the separate curves a series of solutions were obtained for different dimensionless upstream depths, Y_1'. The dimensionless flow rate can be defined as $Q' = \{Q^2/(gD^5)\}^{1/2}$. Then $Q' = F_{r1}/\left\{A_1'^3/T_1'\right\}^{1/2}$. The graphs below were obtained by plotting the solutions for Y_b/Y1 and Y_r/Y1, respectively as the ordinate, and the Froude number F_{r1} as the abscissa for Y_1' = 0.1, 0.4, 0.6, 0.7, 0.8, and 0.9.





EXAMPLE PROBLEM 2.24

Develop a special algorithm, and implement it in a subroutine, based on the Newton method to solve for the bottom width b and the critical depth Y_c of a trapezoidal channel that will result in critical flow for a given flow rate Q and specific energy E (or head H).

Solution

The two equations needed to solve for these two unknowns Y_c (used as Y in equations below) and b are the critical flow and energy equations, or,

$$F_1 = Q^2T - gA^3 = 0$$

$$F_2 = E - Y - \frac{Q^2}{2gA^2} = 0$$

Notice that the derivative $\partial F_2/\partial Y$ in the Jacobian for the Newton method consists of

$$\frac{\partial F_2}{\partial Y} = -1 + \frac{Q^2T}{gA^3}$$

and since we want the flow to be critical, we will set $\partial F_2/\partial Y = 0$, even though when it is evaluated for guesses for b and Y it may not be zero. Thus the Jacobian becomes

$$[\mathbf{D}] = \begin{bmatrix} \frac{\partial F_1}{\partial Y} & \frac{\partial F_1}{\partial b} \\ \frac{\partial F_2}{\partial Y} & \frac{\partial F_2}{\partial b} \end{bmatrix} = \begin{bmatrix} 2mQ^2 - 3gA^2T & Q^2 - 3gA^2Y \\ 0 & \frac{Q^2Y}{(gA^3)} \end{bmatrix} = \begin{bmatrix} D_{11} & D_{12} \\ 0 & D_{22} \end{bmatrix}$$

Notice that the Jacobian is already in upper triangular form, so the solution vector {**z**} in [**D**]{**z**} = {**F**} can be obtained from back substitution, or $z_2 = F_2/D_{22}$, and $z_1 = (F_1 - D_{12}z_2)/D_{11}$.

The subroutine SOLYB given below implements this special Newton iteration.

SOLYB.FOR

```
      SUBROUTINE SOLYB(G,FM,E,Q,Y,B)
      FM2=2.*FM
      QS=Q*Q
      FMQ=FM2*QS
      QSG=QS/G
      G3=3.*G
      M=0
1     A=(B+FM*Y)*Y
      T=B+FM2*Y
      AS=A*A
      F1=QS*T-G*A*AS
      F2=E-Y-QSG/(2.*AS)
      D11=FMQ-G3*AS*T
      D12=QS-G3*AS*Y
      Z2=F2/(QS*Y/(G*A*AS))
      Z1=(F1-D12*Z2)/D11
      B=B-Z2
      Y=Y-Z1
      M=M+1
      IF(ABS(Z1)+ABS(Z2).GT.1.E-5 .AND. M.LT.20) GO TO 1
      IF(M.EQ.20) WRITE(*,*) ' Didnot converge'
      RETURN
      END
```

A main program that calls on this subroutine can consist of the following. If the input: 32.2 1.5 5 500 3.5 10 is given then the solution is: Y = 3.666, b = 9.216.

```
1      WRITE(*,*)' Give:g,m,E,Q, and guesses for Y & b'
       READ(*,*,ERR=99) G,FM,E,Q,Y,B
       CALL SOLYB(G,FM,E,Q,Y,B)
       WRITE(*,100) Y,B
100    FORMAT(' Solution Y =',F8.3,' b =',F8.3)
       GO TO 1
99     STOP
       END
```

2.10 UPSTREAM DEPTH WHEN CRITICAL CONDITIONS OCCUR AT REDUCED DOWNSTREAM SECTION

This section will expand upon the problems of determining upstream conditions from a reduced section that causes critical flow to exist. The principles have already been covered: First the critical flow equation $Q^2T/(gA^3) = 1$ is solved at the downstream section and using the solved critical depth Y_c, the specific energy is computed $E_2 = E_c = Y_c + (Q/A_c)^2/(2g)$, and finally the energy upstream is equated to the downstream critical specific energy accounting for any change in the channel bottom and/or local loss, or $E_1 = Y_1 + (Q/A_1)^2/(2g) = E_c + \Delta z + K(Q/A_c)^2/(2g)$. In order for a solution to exist for the upstream depth Y_1 the specific energy E_1 must be larger than the upstream critical specific energy E_{c1} ($E_1 > E_{c1}$) associated with this flow rate. This generally means that the upstream channel must have a larger cross-sectional area for this depth Y_1, than the cross-sectional in the downstream channel associated with the critical depth here, but for special geometries this may not be true. Furthermore, of the two real solutions available for Y_1 if $E_1 > E_{c1}$ the one that produces a Froude number less than one, i.e., the subcritical depth, is selected. Otherwise a hydraulic jump would take place, as discussed in Chapter 3.

A common design problem that involves this procedure is to size the upstream channel so that for the design flow rate the upstream channel will flow under uniform flow conditions. The terrain over which the downstream channel occurs may slope significantly so that critical flow will occur at its beginning and thereafter the depth will decrease toward the normal supercritical depth supported by its steep slope. It is since the bottom slope increases that the size of the channel is reduce. The design of transitions from the upstream to downstream channel is covered in Chapter 5, in which the transition's design may result in normal depths in both the upstream and downstream channels for the selected flow rate. In this section we will assume that there is no change in bottom elevation across the transition, and that the local loss caused by the transition can be ignored, or perhaps a more realistic statement is that the energy loss through the transition equals the drop in the channel bottom so $E_1 = E_{c2}$.

The following are some of the possible combinations of upstream and downstream channels: (1) upstream trapezoidal channel to smaller downstream trapezoidal channel, (2) upstream trapezoidal channel to downstream rectangular channel, (3) upstream rectangular channel to smaller downstream rectangular channel, (4) upstream trapezoidal (or rectangular) channel to downstream circular channel, (5) upstream circular channel to downstream trapezoidal (or rectangular) channel, and (6) upstream circular channel to smaller downstream circular channel. The approach to the solution to any of these six combinations follows the procedure illustrated in Example Problem 2.25.

EXAMPLE PROBLEM 2.25

A channel system is to be designed for a flow rate Q = 400 cfs. The slope of the land changes so that the upstream channel will have $S_{o1} = 0.00035$, and the downstream channel $S_{o2} = 0.015$, the channels are of concrete with n = 0.013. The downstream channel will consist of a circular section with a diameter $D_2 = 8$ ft. Determine the bottom width b_1 of an upstream trapezoidal channel with a side slope $m_1 = 0.8$ that will flow under uniform conditions. (The bottom remains horizontal through the transition from the trapezoidal to the circular channels.)

Solution

Step 1: Since the downstream channel is steep, critical flow will occur at its entrance so solve $Q^2T/(gA^3) = 1$ in the 8 ft diameter pipe, for $Y_c = 5.084$ ft.

Step 2: Compute the specific energy associated with this Y_c, or $E_c = E_2 = Y_c + (Q/A_c)^2/(2g) = 7.272$ ft.

Step 3: Equate the upstream energy to that downstream, or $Y_1 + (Q/A_1)^2/(2g) = E_c = 7.272$ ft and solve this equation simultaneously with the uniform flow equation (Manning's equation) $Q = (C_u/n)A_1^{5/3}(S_{o1})^{1/2}/P_1^{2/3}$ for Y_1 and b_1 to give: $Y_{o1} = Y_1 = 6.906$ ft and $b = 6.407$ ft. The Froude number associated with this solution is $F_{r1} = 0.39$. Using $b_1 = 6.407$ ft, another solution for Y_1 is 2.722 ft, but the Froude Number associated therewith is 2.047, obviously not the sought for solution. (The normal depth in the downstream circular channel is $Y_{o2} = 3.308$ ft ($E_{o2} = 9.757$ ft), so the depth will be critical at its beginning and gradually thereafter decrease to this latter uniform depth.)

2.11 DIMENSIONLESS TREATMENT OF UPSTREAM TRAPEZOIDAL CHANNEL TO DOWNSTREAM RECTANGULAR CHANNEL

Generalization of the solution to the problem of finding conditions in an upstream trapezoidal channel when critical flow exists in a downstream rectangular channel can be handled by introducing dimensionless variables. (This is combination (2) mentioned above.) The following three dimensionless variables will be introduced:

$Y_1' = m_1Y_1/b_1$ (the dimensionless depth in the upstream channel)
$b_1' = b_1/b_2$ (the ratio of the upstream bottom width to the downstream bottom width), and
$b' = b_1/Y_c$ (the ratio of the upstream bottom width to the critical depth in the downstream rectangular channel).

Since the downstream channel is rectangular the critical depth here can be computed from $Y_c = \{q_2^2/g\}^{1/3} = \{(Q/b_2)^2/g\}^{1/3}$. We should also note, since the downstream Froude number is unity, that $Q^2/(gY_c^3b_2^2) = 1$ and $E_c = E_2 = 1.5Y_c$. Multiplying the energy equation $E_1 = E_c$ by m_1/Y_c gives,

$$\frac{m_1Y_1}{Y_c} + \frac{m_1Q^2}{2gY_c(b_1Y_1 + m_1Y_1^2)^2} = 1.5m_1$$

or written in terms of the dimensionless variables this energy equation becomes,

$$Y_1'b_1 + \frac{0.5m_1^3/(b_1')^2}{b'^2\left(Y_2' + Y_2'^2\right)^2} = 1.5m_1 \tag{2.40}$$

Equation 2.40 can be written as the following fifth degree polynomial.

$$Y_1'^5 + \left(2 - \frac{1.5m_1}{b'}\right)Y_1'^4 + \left(1 - \frac{3m_1}{b'}\right)Y_1'^3 - \frac{1.5m_1}{b'Y_1'^2} + \frac{0.5m_1^3}{\left(b_1'^2b'^3\right)} = 0 \tag{2.41}$$

By introducing these dimensionless variables in the upstream Froude number squared, $F_{r1}^2 = Q^2(b_1 + 2m_1Y_1)/\{g(b_1Y_1 + m_1Y_1^2)^3\}$ after some manipulation, it can be expressed by the following equation as a function the upstream side slope m_1 and these dimensionless variables:

$$F_{r1}^2 = \frac{m_1^3(1 + 2Y_1')}{b_1'^2b'^3(Y_1' + Y_1'^2)^3} = \frac{(m_1/b')^3(1 + 2Y_1')}{b_1'^2(Y_1' + Y_1'^2)^3} \tag{2.42}$$

Since a fifth degree polynomial has five roots, it is clear that some precautions be used in solving either Equations 2.40 or 2.41 to insure that the sought after subcritical root for Y_1' is obtain.

If m_1 and b_1' are large enough then two positive real positive roots, one the subcritical and the other the supercritical depth, one negative root, and two complex conjugate roots are available. The values of these roots will depend on the ratio $b' = b_1/Y_c$. It will be a worthwhile exercise for you to write a computer program to produce several tables for different m_1 values of solutions Y_1' as a function of b_1' and b'. (See the homework problems at the end of this chapter.) The solutions for dimensionless depth Y_1' from Equations 2.40 or 2.41 for eight different upstream side slopes $m_1 = 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75$ and 2.0 are given in the eight graphs in Figure 2.13. On these

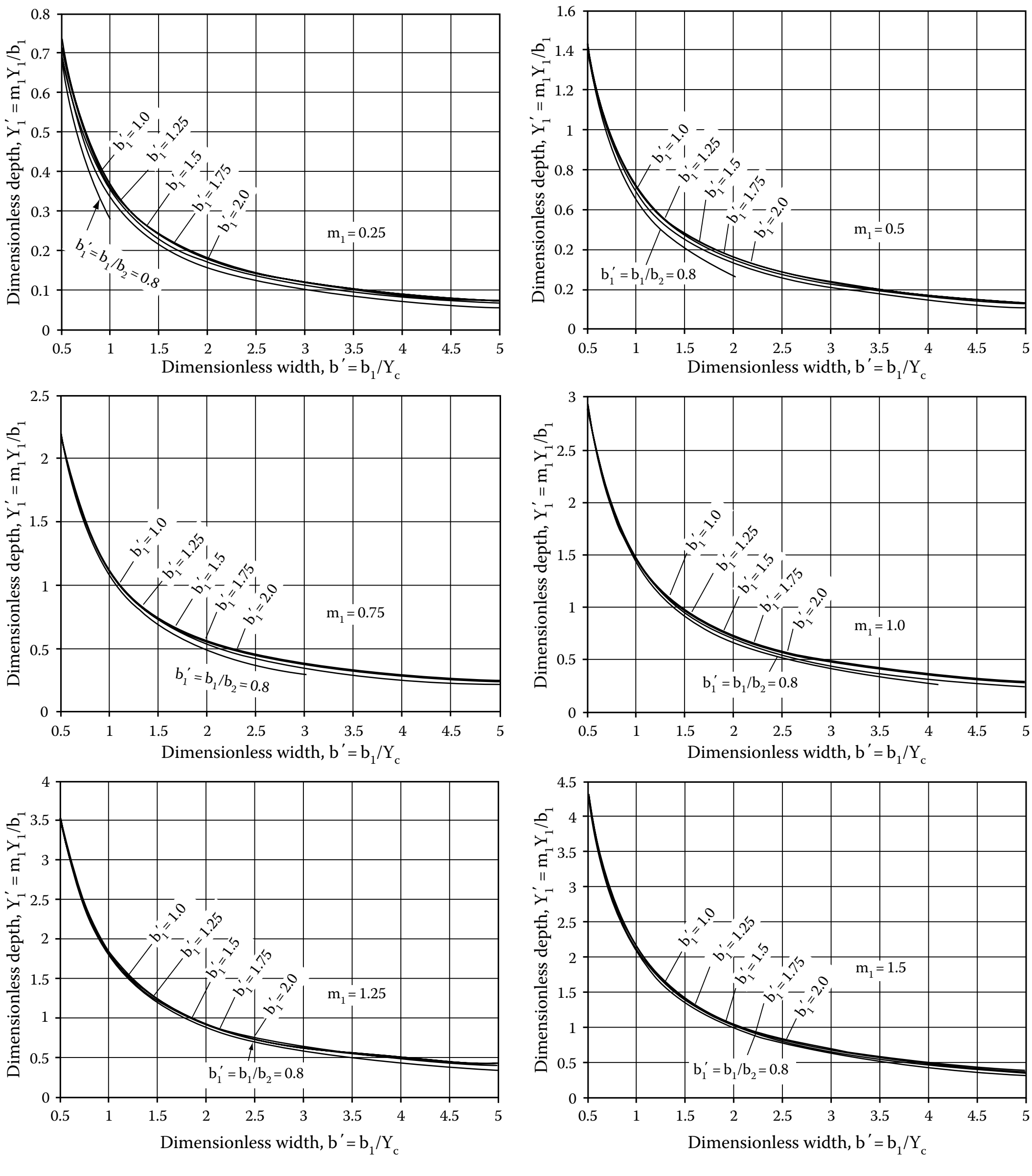


FIGURE 2.13 Eight graphs for different upstream side slopes that provide the solution of Equation 2.40 of dimensionless depth Y_1' as a function of b' and b_1' for upstream trapezoidal channels to downstream rectangular channels with critical flow.

(continued)

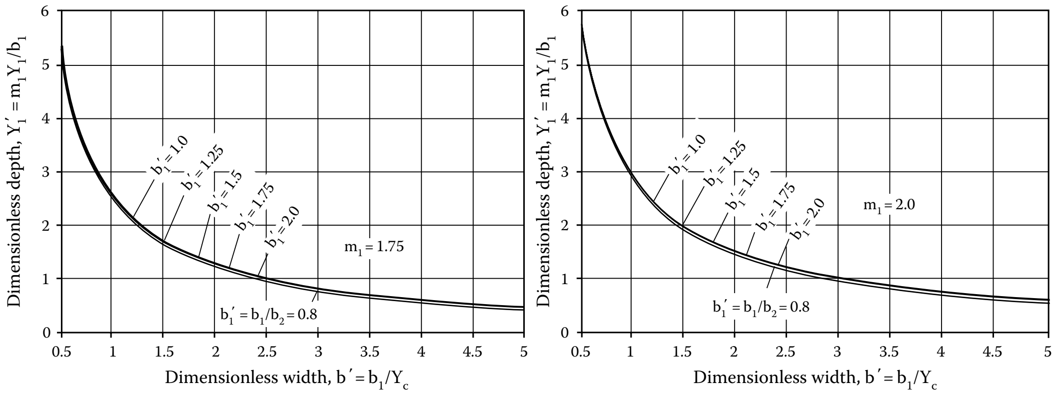


FIGURE 2.13 (continued)

graphs the ratio of upstream bottom width b_1 to the downstream critical depth $b' = b_1/Y_c$ are the abscissas, and Y_1' are the ordinates, with different lines for the ratio $b_1' = b_1/b_2$. The bottom curve on each graph is for $b_1' = 0.8$, the next curve up for $b_1' = 1.0$, etc. with the top curve for $b_1' = 2.0$. The lower curves end when a real root to Equation 2.41 is not available, i.e., the downstream critical specific energy is not larger than the upstream critical specific energy because b_1 is too small. (Conditions of no solution will occur only for $b_1' < 1$.) In other words the upstream flow has also become critical, or $F_{r1} = 1$ as well as $F_{r2} = 1$. The dimensionless depth Y_1' and ratio b', associated with any b_1' can be solve from Equations 2.40 (or 2.41) and the equation obtained by setting the Froude number, Equation 2.42, to unity. A TK-Solver model providing such solutions for several different values of b_1' less than 1 and for six side slopes m_1 is given below. (The "Solve List" capability of TK-Solver was used separately for each of the m_1 as given and these separate solutions were combined into the single table given below.) The dimensionless upstream depth Y_1' does not change with m_1 as can be observed from the separate graphs on Figure 2.13, or by examining the nature of the equations, but of course the actual depth Y_1 will change. Notice how b' becomes larger with both m_1 and b_1'.

TK-Solver model to solve for $b' = b_1/Y_c$ and $Y_1' = m_1Y_1/b_1$ for different values of $b_1' = b_1/b_2$. (Solves dimensionless $F_{r1}^2 = 1$ and $E_1 = E_{c2}$ simultaneously.)

```
══════════════ VARIABLE SHEET ══════════
St Input──── Name─── Output─── Unit─────
LG .25707709 Y1p
   .25       m
LG 1.0308088 bp
L  .5        b1p
══════════════ RULE SHEET ════════════
S Rule────────────────────────────────
  (m/bp)^3/b1p^2*(1+2*Y1p)/(Y1p+Y1p^2)^3=1
  Y1p*bp+.5*m^3/(b1p*bp*(Y1p+Y1p^2))^2=1.5*m
```

		$b' = b_1/Y_c$					
b_1'	Y_1'	$m_1 = 0.2$	$m_1 = 0.25$	$m_1 = 0.30$	$m_1 = 0.40$	$m_1 = 0.50$	$m_1 = 0.75$
0.500	1.05510900	0.213722	0.267152	0.320582	0.427443	0.534304	0.801456
0.550	0.86045400	0.259824	0.324780	0.389736	0.519648	0.649560	0.974341
0.600	0.69857550	0.317100	0.396375	0.475650	0.634200	0.792750	1.189124
0.650	0.56197330	0.390313	0.487891	0.585470	0.780626	0.975783	1.463674

(continued)

		$b' = b_1/Y_c$					
b_1'	Y_1'	$m_1 = 0.2$	$m_1 = 0.25$	$m_1 = 0.30$	$m_1 = 0.40$	$m_1 = 0.50$	$m_1 = 0.75$
0.700	0.44529950	0.487402	0.609253	0.731104	0.974805	1.218506	1.827759
0.750	0.34464220	0.622656	0.778320	0.933984	1.245312	1.556640	2.334960
0.800	0.25707710	0.824647	1.030809	1.236971	1.649294	2.061618	3.092426
0.850	0.18037750	1.160043	1.450054	1.740065	2.320086	2.900108	4.350162
0.900	0.11282030	1.828846	2.286057	2.743268	3.657691	4.572114	6.858171
0.920	0.08804428	2.329720	2.912150	3.494580	4.659440	5.824300	8.736450
0.930	0.07609940	2.687304	3.359131	4.030957	5.374609	6.718261	10.077390
0.940	0.06443842	3.163939	3.954925	4.745909	6.327879	7.909849	11.864770
0.950	0.05305317	3.831054	4.788817	5.746581	7.662107	9.577634	14.366450
0.960	0.04193582	4.831504	6.039380	7.247256	9.663008	12.078760	18.118140
0.970	0.03107887	6.498624	8.123280	9.747935	12.997250	16.246560	24.369840
0.971	0.03000722	6.728554	8.410693	10.092830	13.457110	16.821390	25.232080
0.972	0.02893809	6.974905	8.718632	10.462360	13.949810	17.437260	26.155900
0.973	0.02787148	7.239501	9.049376	10.859250	14.479000	18.098750	27.148130
0.974	0.02680738	7.524447	9.405559	11.286670	15.048890	18.811120	28.216670
0.975	0.02574579	7.832185	9.790231	11.748280	15.664370	19.580460	29.370690

EXAMPLE PROBLEM 2.26

An upstream trapezoidal channel is to have a bottom width $b_1 = 9$ ft and a side slope $m_1 = 0.5$, and transitions to a rectangular channel with a bottom width $b_2 = 10$ ft at the beginning of a steep slope. What flow rate will result in critical depth in both the trapezoidal and rectangular channels? What are the depths, Y_1 and Y_2?

Solution

From the above table on the line corresponding to $b_1' = 9/10 = 0.9$ and the column for $m_1 = 0.50$ we read $b' = 4.572114$ and therefore $Y_{c2} = b_1/b' = 9/4.572114 = 1.968$ ft ($E_{c2} = 1.5Y_{c2} = 2.952$ ft). The flow rate per unit width in the rectangular channel is $q_2 = (gY_{c2}^3)^{1/2} = (32.2 \times 1.968^3)^{0.5} = 15.672$ cfs/ft, so $Q = 10(15.672) = 156.72$ cfs. The depth in the upstream channel will be $Y_1 = b_1 Y_1'/m_1 = 9(0.11282030)/0.5 = 2.031$ ft ($E_1 = E_{1c} = 2.953$ ft), i.e., just 0.067 ft greater than the downstream depth, but the specific energies are equal. If Manning's n = 0.013, this critical flow will occur in the upstream channel if $S_{o1} = 0.002641$.

EXAMPLE PROBLEM 2.27

An upstream trapezoidal channel with $b_1 = 15$ ft and $m_1 = 1$, changes to a rectangular channel with $b_2 = 10$ ft at the head of a steep slope. For a flow rate of Q = 400 cfs what are the depths in the trapezoidal and rectangular channels and the Froude number of the upstream flow?

Solution

$b_1' = 15/10 = 1.5$, and from the critical flow equation in the rectangular channel $Y_{c2} = (40^2/32.2)^{1/3} = 3.676$ ft so $b' = b_1/Y_{c2} = 15/3.676 = 4.080$. Next solve Equation 2.40 (or 2.41), or read the graph in Figure 2.13, or use the table you generated for the home problem for $Y_1' = 0.35333$ (the graph will produce at most 2 digits of accuracy), so $Y_1 = 15(0.35333)/1 = 5.300$ ft, and from Equation 2.41 $F_{r1} = 0.316$.

EXAMPLE PROBLEM 2.28

An upstream trapezoidal channel is to have a side slope $m_1 = 1.5$, and a bottom width of 1.5 times that of a downstream rectangular channel at the head of a steep slope. If the upstream channel has n = 0.013, and a bottom slope $S_{o1} = 0.0003$, what should the bottom widths of the two channels be for a design flow rate of $Q = 15\,m^3/s$ if uniform flow is to take place in the upstream channel. Solve the problem using dimensionless variables, and verify this solution by solving the problem using dimensioned variables. What occurs when the bottom slope of the upstream channel increases, i.e., say to $S_{o1} = 0.0005$?

Solution

A good approach involves solving Manning's equation with Y_1 replaced by b_1Y_1'/m_1 and Equation 2.40 with $b' = b_1/Y_c = b_1 \Big/ \left\{\left[\left(b_1'Q/b_1\right)^2/g\right]^{1/3}\right\} = b_1^{1/3}/3.723$ simultaneously for Y_1' and b_1. The solution is $Y_1' = 0.0525$ and $b_1 = 20.092$ m. The upstream depth $Y_1 = b_1Y_1'/m_1 = 0.704$ m. A general TK-Solver model (TRARECS1.TK) designed to solve problems of this type is given below.

```
============ VARIABLE SHEET ============
St Input──── Name─── Output─── Unit───
             Y1p     .05254506
             b1      20.092804
             Y1      .70385174
             bp      39.887384
   1.5       m
   1.5       b1p
   15        Q
   1         Cu
   .0003     So1
   .013      n
   9.81      g
============== RULE SHEET ============
S Rule────────────────────────────────
  Y1p=m*Y1/b1
  Q=Cu*So1^.5*((b1+m*Y1)*Y1)^1.666667/(n*(b1+2.*(m*m+1)^.
    5*Y1)^.6666667)
  bp=b1/(((b1p*Q/b1)^2/g)^.3333333)
  Y1p*bp+.5*m^3/(b1p*bp*(Y1p+Y1p^2))^2=1.5*m
```

To solve the problem using dimensioned variables requires that Y_1, E_c, and b_1 be solved using: (1) Manning's equation in the upstream channel, (2) The critical specific energy equation $E_c = 1.5\left\{\left(b_1'Q/b_1\right)^2/g\right\}^{1/3}$, and (3) the energy equation $Y_1 + (Q/A_1)^2/(2g) = E_c$. The solution is the same as above. As the upstream bottom slope increases the width of channel increases rapidly to be unrealistically wide, i.e., the above solution with $b_1 = 20.1$ m is a very wide channel already. A better solution would be to use a smaller ratio b_1', i.e., let $b_1' = 1.2$ or smaller. If $b_1' = 1.2$ and $S_{o1} = 0.0005$, then the above model gives: $0.979Y_1' = 0.0979$, $Y_1 = 0.805$ m, $b_1 = 12.333$ m, and $b' = 20.52$.

2.11.1 Upstream Channel Also Rectangular

The above dimensionless variables are not valid for a rectangular channel since the dimensionless depth is obtained by multiplying the actual depth by the side slope m, and this is zero. For a rectangular channel let

$$Y_1' = Y_1/b_1 \quad \text{with } b_1' = b_1/b_2 \quad \text{and} \quad b' = b_1/Y_{c2} \text{ as before}$$

With this redefined Y_1' the dimensionless specific energy between the upstream rectangular channel, and the downstream rectangular channel where critical flow occurs is,

$$b'Y_1' + \frac{1}{2b'^2b_1'^2Y_1'^2} = 1.5 \tag{2.43}$$

or written as a third degree polynomial

$$Y_1'^3 - \left(\frac{1.5}{b'}\right)Y_1'^2 + \frac{0.5}{(b'^3 b_1'^2)} = 0 \tag{2.44}$$

and the Froude number squared,

$$F_{r1}^2 = \frac{1}{(b_1'^2 b'^3 Y_1'^3)} = \frac{3}{(b' Y_1')} - 2 \tag{2.45}$$

When both upstream and downstream channels are rectangular, and the depth is to be critical in the downstream channel it is clear that the width ratio b_1' must be larger than unity for a real solution to be available from Equation 2.44 (or Equation 2.43). When this condition is met then there will be at least one positive real root. Figure 2.14 provides graphical solutions (one using linear paper and the other using log–log paper) of this larger positive dimensionless depth Y_1' as a function of the ratio $b' = b_1/Y_c$ for several width ratios b_1' starting with 1.05 and ending with 2.5. The lowest curve applies for $b_1' = 1.05$, and the upper curve for $b_1' = 2.5$, but note again as was the case when the upstream channel is trapezoidal, that the spread between the curves is small and gets smaller with increasing values of b_1' and b'. You will find it useful to generate a table of values that would allow Figure 2.14 to be plotted to be used in solving problems. (See one of the homework problems at the end of this chapter.)

Because whenever Y_1' or b = occur in Equation 2.43 they form a product $b'Y_1' = c$, which must be a constant for any given ratio $b_1' = b_1/b_2$. This fact is apparent also from the log–log plot on Figure 2.14 with straight lines with a slope of −1. Straight lines on a log–log plot conform to the equation $y = cx^b$ in which b is the slope and c is the intercept. ($b'Y_1'$ is not only constant for the subcritical dimensional depth Y_1', but also is constant for the supercritical dimensionless depth Y_1' and the nonphysical negative root Y_1'.) Therefore a much simpler means for evaluating Y_1' for any b_1' is to find $c = b'Y_1'$ (the constant, which is given in the table below on the row of Y_1') and then solve for $Y_1' = c/b'$.

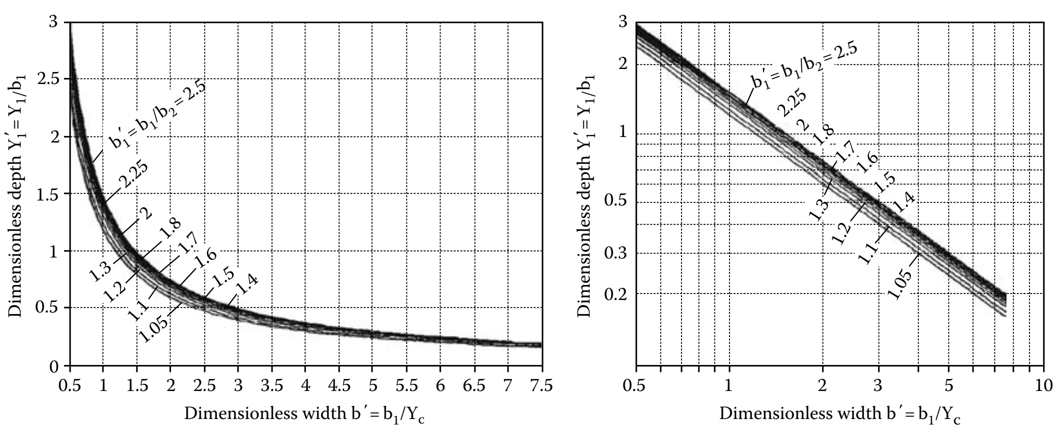


FIGURE 2.14 Solution of Equation 2.43 of dimensionless depth Y' as a function of b' and b_1' for rectangular channels both upstream and downstream and critical flow in the downstream channel.

The second means for evaluating the Froude number given by Equation 2.45 shows that F_{r1} (or F_{r1}^2), does not change with the ratio $b_1' = b_1/b_2$. Thus the upstream Froude number for any upstream rectangular channel with a width b_1 that reduces to a rectangular channel with width b_2 in which the flow is critical can be computed by finding this constant $c = b'Y_1'$ and then using $F_{r1} = \{3/c - 2\}^{1/2}$. The table below gives the subcritical solution Y_1' corresponding to $b' = 1$ for different ratios b_1', so this Y_1' equals this constant c, and the corresponding upstream Froude numbers.

Table of Subcritical Roots Y_1' of the Third Degree Polynomial for a Rectangular-to-Rectangular Channel Reduction that Produces Y_c at b_2, Corresponding to $b' = 1$

b_1'	1.05	1.10	1.20	1.30	1.40	1.50	1.60	1.70	1.80	2.00	2.25	2.50
Y_1'	1.1670	1.2243	1.2920	1.3337	1.3626	1.3840	1.4004	1.4134	1.4239	1.4397	1.4532	1.4626
F_{r1}	0.755	0.671	0.567	0.494	0.449	0.409	0.377	0.350	0.327	0.289	0.254	0.226

EXAMPLE PROBLEM 2.29

A flow rate Q = 400 cfs occurs in a upstream rectangular channel with a width of $b_1 = 15$ ft, which reduces to a rectangular channel with a width $b_2 = 10$ ft at the head of a steep slope. Determine the upstream depth and Froude number. If n = 0.013 what bottom slope of this upstream channel will result in uniform flow? Solve for the same unknowns when Q = 200 cfs.

Solution

The critical depth in the smaller channel is $Y_c = \{(Q/b_2)^2/g\}^{1/3} = \{40^2/32.2\}^{1/3} = 3.676$ ft, so $b' = b_1/Y_c = 15/3.676 = 4.08$. With $b_1' = 1.5$ the larger positive solution to Equation 2.43 (or 2.44), which may be read to two digits from Figure 2.14 but can be solved easiest by dividing c by b', is $Y_1' = 1.385/4.08 = 0.3392$, so $Y_1 = 5.088$ ft. From Equation 2.45 $F_{r1} = 0.4095$. Solving Manning's equation gives $S_{o1} = 0.000479$. For Q = 200 cfs, $Y_c = 2.314$ ft, $b' = 6.477$, $Y_1' = 1.384/6.477 = 0.2137$, so $Y_1 = 3.205$ ft, and $F_{r1} = 0.4095$ (the same since b_1' is the same), and $S_{o1} = 0.000449$.

EXAMPLE PROBLEM 2.30

What width b_2 of steep downstream rectangular channel will result in uniform flow in a $b_1 = 4$ m wide upstream rectangular channel with n = 0.013 and a bottom slope $S_{o1} = 0.0005$ if the flow rate is $Q = 10\,m^3/s$.

Solution

Depth ratio b' can be given as the following function of b_1': $b' = 4/Y_c = 4/\{[(10/4)b_1']^2/g\}^{1/3} = 4.6486086/b_1'^{2/3}$. With this substituted into Equation 2.43, it can be solved simultaneously with Manning's equation for b_1' and Y_1'. The solution is $b_1' = 1.52135$ and $Y_1' = 0.3949$ ($b' = 3.514$), so $b_2 = 4/1.52135 = 2.629$ m, and $Y_1 = 4(.3949) = 1.580$ m. The Froude number $F_{r1} = \left\{3/\left(b'Y_1'\right) - 2\right\}^{0.5} = \left\{3/\left(3.514 \times 0.3949\right) - 2\right\}^{0.5} = 0.402$. Critical depth in the downstream channel can now be computed from $Y_c = \{(10/2.629)^2/9.81\}^{1/3} = 1.138$ m, or $Y_c = b_1/b' = 4/3.514 = 1.138$ m. The Froude number can also be computed from $F_{r1} = \{(10/4)^2/(9.81 \times 1.58^3\}^{0.5} = 0.402$. A TK-Solver model for the above solution is,

```
═══════════════ VARIABLE SHEET ═══════════
St Input──── Name─── Output─── Unit─────
             bp      3.5142794
             b1p     1.5213531
             Y1p     .39491513
             Y1      1.5796605
   10        Q
   .0005     So
   .013      n
```

```
============== RULE SHEET ===========
S Rule————————————————————————————
  bp=4.6486086/b1p^.6666667
  bp*Y1p+.5/(bp*b1p*Y1p)^2=1.5
  Y1=4*Y1p
  Q=(4*Y1)^1.666667*So^.5/(n*(4+2*Y1)^.666667)
```

2.12 HYDRAULICALLY MOST EFFICIENT SECTION

A major consideration in the design of a channel should be to keep total costs to a minimum. The costs associated with a channel lining are directly proportional to its wetted perimeter. The fluid frictional losses are directly related to the perimeter also. Thus a key factor in designing a channel is to maximize its conveyance capabilities while minimizing its perimeter. The hydraulically most efficient section is defined as a channel with the least wetted perimeter for a given conveyance capability. Therefore, the costs of a channel will generally be near minimum if it is designed as the "hydraulically most efficient section. See the Section 5.9 for designing least cost trapezoidal channels accounting for lining and excavation costs.

If Manning's equation is assumed to properly account for the frictional losses then to determine the relationship between the geometric variables that will result in the least wetted perimeter the flow rate, Q, the roughness coefficient, n and the bottom slope, S_o are held constant. Combining these variables in a single constant c results in

$$\frac{nQ}{(C_u S^{1/2})P^{2/3}} = A^{5/3} \quad \text{or} \quad cP^{2/3} = A^{5/3}$$

which can be written with c redefined as

$$A = cP^{2/5} \text{ (in which c is redefined as } c = c^{3/5}) \tag{2.46}$$

First, consider a circular section because it is generally known that a circle is the shape that has the largest possible area for a given perimeter. For a circle the area is given by $A = D^2/4(\beta - \sin\beta\cos\beta)$ and Equation 2.46 can be written as

$$\frac{D^2}{4}(\beta - \sin\beta\cos\beta) = cP^{2/5}$$

Differentiating this equation with respect to β and setting $\partial P/\partial\beta = 0$ gives,

$$1 + \sin^2\beta - \cos^2\beta = 0$$

replacing $\cos^2\beta$ with $1 - \sin^2\beta$ gives $\sin\beta = 0$, or $\beta = 0$. Clearly this is a maximum. Replacing $\sin^2\beta$ with $1 - \cos^2\beta$ gives,

$$1 + 1 - \cos^2\beta - \cos^2\beta = 2 - 2\cos^2\beta = 0 \quad \text{or} \quad \cos\beta = 1 \quad \text{or} \quad \beta = \frac{\pi}{2}$$

Thus a circular section is hydraulically most efficient when flowing half full.

Don't confuse this section with the depth in a circular section that will maximize the flow rate it can carry. For a given diameter, wall roughness, and bottom slope the latter depth equals $Y = 0.938D$, or $\beta = 2.6391$ rad (151.21°), as shown in Figure A.2. To get this depth, the conveyance $K = A^{5/3}/P^{2/3}$ is differentiated with respect to Y and the set to zero. After some algebra $5 \sin\beta = 1 - \cos\beta \sin\beta/\beta$ results. Its solution is $\beta = 2.6390536$ radians, and $Y = D(1 - \cos\beta)/2 = 0.93818122D$.

Next, consider a trapezoidal channel. For a trapezoidal channel its cross-sectional area is given by $A = (b + mY)Y$, and its perimeter is given by $P = b + 2Y(m^2 + 1)^{1/2}$. Solving b from the last equation and substituting this for b in the equation for the area and finally using this area in Equation 2.46 gives the following equation that we wish to find the extreme value of:

$$(P - 2Y(m^2 + 1)^{1/2})Y + mY^2 = cP^{2/5} \tag{2.47}$$

Taking the partial derivative with respect to m, with Y held constant, results in,

$$Y\left(\frac{\partial P}{\partial m}\right) - \frac{2Y^2 m}{(m^2 + 1)^{1/2}} + Y^2 = \left(\frac{2}{5}\right)cP^{-3/5}\left(\frac{\partial P}{\partial m}\right)$$

setting $\partial P/\partial m = 0$ results in

$$\frac{2m}{(m^2 + 1)^{1/2}} = 1 \quad \text{or} \quad 4m^2 = m^2 + 1 \quad \text{or} \quad m = \frac{1}{\sqrt{3}} = \frac{3}{\sqrt{3}} \tag{2.48}$$

A side slope of $m = \sqrt{3}/3$ result in an angle of 30° from the vertical or an angle of 120° from the horizontal.

Next, differentiate Equation 2.47 partially with respect to Y, with m held constant, gives,

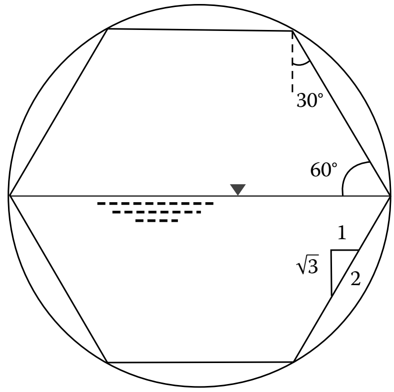
$$Y\left(\frac{\partial P}{\partial Y}\right) + P - 4Y(m^2 + 1)^{1/2} + 2mY = \left(\frac{2}{5}\right)cP^{-3/5}\left(\frac{\partial P}{\partial Y}\right)$$

Setting $\partial P/\partial Y = 0$ results in,

$P - 4(2m)Y + 2mY = 0$ since $(m^2 + 1)^{1/2} = 2m$ from above, or

$$P = 6mY = 6Y\left(\frac{\sqrt{3}}{3}\right) = 2\left(\sqrt{3}\right)Y = 3\left(\frac{2}{\sqrt{3}}\right)Y = 3b \tag{2.49}$$

since the hypotenuse of a right triangle with sides 1 and $m = \sqrt{3}/3$ is $2/\sqrt{3}$. Thus with $P = 3b$ and $Y = (\sqrt{3}/2)b$, it is apparent that the hydraulically most efficient section is 1/2 of a hexagon. This result is not surprising since a hexagon is inscribed within a circle. Substitution of $Y = (\sqrt{3}/2)b$ into the equation for the area of a trapezoidal channel gives $A = (bY + mY^2) = (\sqrt{3}/2)b^2 + (1/\sqrt{3})(3/4)b^2 = \sqrt{3}(1/2 + 1/4)b^2 = 1.299038b^2$.



The area for this section for a trapezoidal channel is

$A = \left(\frac{2}{\sqrt{3}}\right)Y^2 + \frac{1}{\sqrt{3}Y^2} = \left(\frac{3}{\sqrt{3}}\right)Y^2 = \sqrt{3}Y^2$ and the wetted perimeter by,

$P = 3\left(\frac{2}{\sqrt{3}}\right)Y = 2\sqrt{3}Y$ and the hydraulic radius by,

$$R_h = \frac{\left(\sqrt{3}Y^2\right)}{\left(2\sqrt{3}Y\right)}\left(\frac{Y}{2}\right) \qquad (2.50)$$

The last equation when applied to a rectangular section indicates that the channel is operating hydraulically most efficient when the depth of flow is 1/2 the width of the channel.

2.12.1 Nondimensional Variables for This Section

It is interesting to note the values of dimensionless variables coming out of the hydraulically most efficient cross section, or one-half a hexagon, described in the previous section. The dimensionless depth $Y' = mY/b = (1/\sqrt{3})[(\sqrt{3}/2)b]/b = 1/2$ or a constant for all sizes. Defining the dimensionless area $A' = mA/b^2$, it also is a constant or,

$$A' = \frac{mA}{b^2} = Y' + Y'^2 = 0.5 + 0.5^2 = 0.75 = \frac{3}{4}, \text{ or substituting } A = \sqrt{3}Y^2 \quad \text{and} \quad A' = \frac{\sqrt{3}}{\sqrt{3}}\frac{3}{4} = \frac{3}{4}$$

As is noted above P = 3b, thus if the dimensionless wetted perimeter is define by $P' = P/b = 3$. The hydraulic radius $R_h = A/P = [(b^2/m)A']/(bP') = (b/m)(A'/P')$, or letting

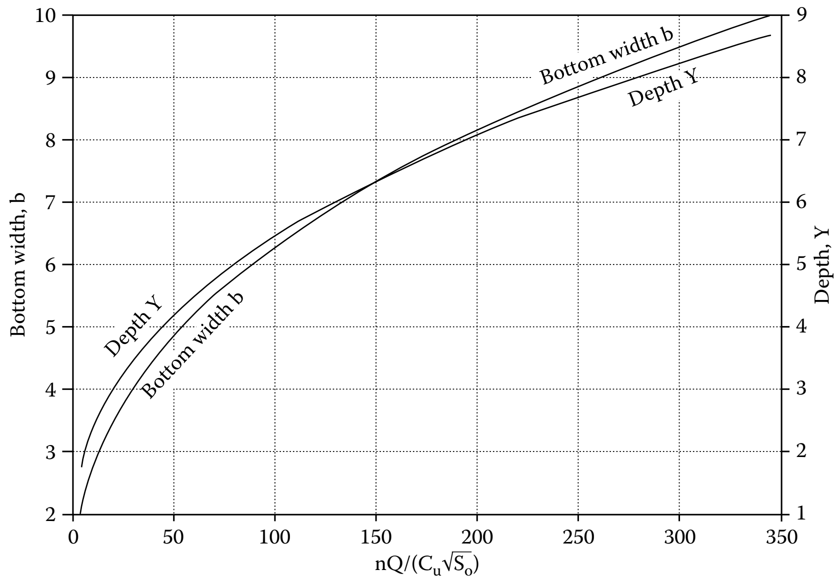
$$R'_h = m\left(\frac{R_h}{b}\right) = \frac{A'}{P'} = \left(\frac{3}{4}\right)\left(\frac{1}{3}\right) = 0.25 = \frac{1}{4}$$

The same result can be obtained from Equation 2.50, namely $R_h = Y/2 = (bY'/m)/2$, or $R'_h = mR_h/b = Y'/2 = 1/4$.

Since Manning's equation is not dimensionally homogeneous (unless dimensions are assigned to C_u and/or n), it makes little sense to introduce a dimensionless flow rate Q into it. However, by substituting the above into Manning's equation the following relationships between Q and C_u, n, and S_o, and b, or Y can be obtained for the hydraulically most efficient trapezoidal channel:

$$Q = 0.7435137\left(\frac{C_u}{n}\right)\sqrt{S_o}\,b^{8/3} = 1.0911236\left(\frac{C_u}{n}\right)\sqrt{S_o}\,Y^{8/3}$$

The graph below provides this relationship of $nQ/(C_u\sqrt{S_o})$ and the bottom width b and the depth Y.



The relationship of the optimal bottom width, and the optimal depth with Manning parameter $nQ/(C_u\sqrt{S_o})$ trapezoidal channels

EXAMPLE PROBLEM 2.31

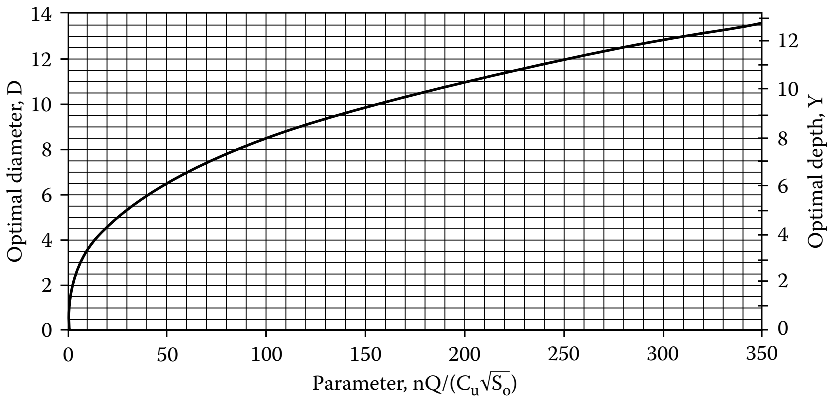
A flow rate of Q = 500 cfs is to be conveyed in a channel of maximum hydraulic efficiency with n = 0.014 and S_o =0.0008. What width of trapezoidal channel b, and what diameter of circular channel D should be used? If the cost of channel per foot of length is a linear function of its perimeter, i.e., \$ = 300P, compare the costs of these two channels. Also determine the perimeter, and cost of a circular section in which the conveyance is a maximum.

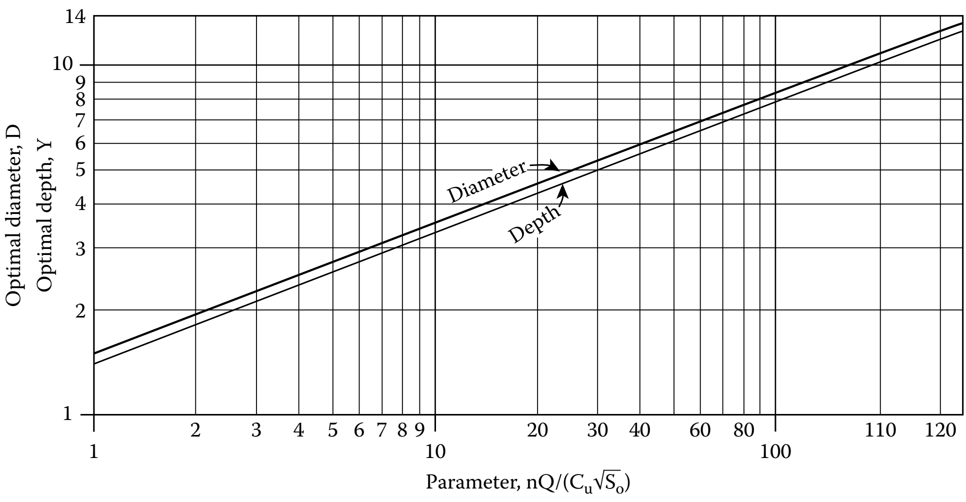
For a circular section, make a graph similar to the one above that shows how the optimal diameter D (and/or the optimal depth Y) varies with the parameter $nQ/(C_u\sqrt{S_o})$.

Solution

When P = 3b and A = $1.299038b^2$ are substituted into Manning's equation and b solved for the following equation is found: $b = \left\{Q/(0.7435137C_u\sqrt{S_o}/n)\right\}^{3/8}$ = $\{500/(0.7435137(1.486)(0.0008)^{1/2}/0.014)\}^{3/8}$ = 7.609 ft for the trapezoidal channel. For the hydraulically most efficient circular section, i.e., one-half a circle, $\beta = \pi/2$, and from Manning's equation $500 = 1.486(0.0008)^{1/2}/0.014[D^2/4(\pi/2)]^{5/3}/(D\pi/2)^{2/3}$. Solving for D = 13.672 ft. For the circular channel with the maximum conveyance β = 2.6391 rad $500 = 1.486(0.0008)^{1/2}/0.014[D^2/4(\beta - \cos\beta\sin\beta)]^{5/3}/[D\beta]^{2/3} = 1.0065746D^{8/3}$, or D = 10.2576 ft. For the trapezoidal section P = 3b = 3(7.609) = 22.828 ft, and for the circular sections P = $(\pi/2)$13.672 = 21.475 ft, and P = βD = 2.6391(10.2576) = 27.071 ft when K is a maximum, or the costs are: trapezoidal section \$/ft = \$6848.40, and circular section \$/ft = \$6442.57/ft and \$/ft = \$8121.30 when K is a maximum, or \$1678.83 more than for the one-half circle.

For the last part of this problem that require a graph to be produced that shows how the optimal diameter varies with $nQ/(C_u\sqrt{S_o})$, we start with $nQ/(C_u\sqrt{S_o}) = A^{5/3}/P^{2/3}$, and note that for the optimal $\beta = 2.6390536$ rad, the area which is $A = (D^2/4)(\beta - \sin\beta\cos\beta)$ that the optimal area is $A = 0.7652888D^2$, and the optimal wetted perimeter is $P = \beta D = 2.6390536D$ or the optimal $nQ/(C_u\sqrt{S_o}) = 0.335282D^{8/3}$ or $D = 1.506507[nQ/(C_u\sqrt{S_o})]^{3/8}$. Sine the optimal depth $Y = 0.93818122D$, we get $Y = 1.41338\,[nQ/(C_u\sqrt{S_o})]^{3/8}$. The graphs below provide plots of these latter two equations; the first graph uses linear axes, whereas the second graph uses logarithmic axes. Because of the nature of the equations, they plot as straight lines on the log–log graph.





Relationship the optimal diameter D, and optimal depth Y to the parameter $nQ/(C_u\sqrt{S_o})$ for a circular channel. The first graph uses linear axes, and the second graph uses logarithmic axes.

PROBLEMS

2.1 Determine what the uniform flow rate will be in a circular channel with a diameter of 2 m, if the bottom slope is 0.0004, and its wall roughness is 0.0003 m, if the depth of flow is 1.3 m. The kinematic viscosity of the water flowing in this pipe is $v = 1.519 \times 10^{-6}$ m²/s.

2.2 What depth of flow would be expected in a 5 ft diameter channel that has a bottom slope of 0.004, if the wall roughness is 0.0005 ft and 200 cfs is flowing in this channel under uniform conditions. ($v = 1.41 \times 10^{-5}$ ft²/s)

2.3 What is the depth of flow in Problem 2.2 if the flow rate is 100 cfs?

2.4 Determine the depth of flow that would occur in a trapezoidal channel with b = 10 ft, m = 1.5, S_o = 0.0008 if it is carrying 400 cfs. The wall roughness for this channel is e = 0.001 ft. What is Chezy's C? What is the corresponding value of Manning's n? ($\nu = 1.41 \times 10^{-5}$ ft²/s)

2.5 What size pipe should be used to carry 6 m³/s down a mild slope of 0.0015 if the pipe is not to flow more than 3/4 full (e = 0.0015 m and temperature = 20°C).

2.6 A 6 ft circular conduit has a bottom slope S_o = 0.001, and a wall roughness of e = 0.004 ft. What will the depth be for a flow rate of 80 cfs (temperature = 50°F). Using this depth, compute the corresponding value for Manning's coefficient.

2.7 Use Chezy's formula to fill in the blank in the table below that applies for uniform flow in channels of trapezoidal shapes ($\nu = 1.41 \times 10^{-5}$ ft²/s).

Flow Rate Q (cfs)	Rough. e (ft)	Bot. Slope S_o	Bot. Width b (ft)	Side Slope m	Depth Y (ft)
—	0.005	0.0006	6.0	1.6	4.2
324.5	—	0.0008	8.0	0.0	5.0
550.0	0.003	—	10.0	1.3	4.8
400.0	0.004	0.0015	—	1.5	4.0
450.0	0.004	0.0009	7.0	—	4.1
600.0	0.005	0.00075	10.0	1.2	—

2.8 The table below applies for uniform flow in circular channels. Fill in the mission blanks ($\nu = 1.317 \times 10^{-6}$ m²/s).

Flow Rate Q (m³/s)	Rough. e (m)	Bot. Slope S_o	Diameter D (m)	Depth Y (m)
—	0.005	0.00080	3.0	2.3
20.0	—	0.00068	5.0	2.52
25.0	0.003	—	6.0	2.9
30.0	0.0025	0.00040	—	3.0
60.0	0.0045	0.00045	8.0	—

2.9 Assume the ratio of Y/D (depth to diameter) is the same as in Problem 2.6 but the conduit is of 1 ft diameter. What flow rate is occurring? What is Manning's n for this flow? Why is n not the same as in Problem 2.6?

2.10 A flow rate of 40 m³/s is flowing in a trapezoidal channel under uniform conditions at a depth of Y = 2 m. The channel has a bottom width of b = 3 m, and a side slope of m = 1.5. Its wall roughness e = 0.002 m. What is the bottom slope of this channel? ($\nu = 1.519 \times 10^{-6}$ m²/s).

2.11 Water is to enter a circular channel from a reservoir so that the depth in the channel will be 3.5 ft. The channel will have a bottom slope of S_o = 0.0009, and a wall roughness of e = 0.002 ft. What size pipe must be used to carry a flow rate of 120 cfs?

2.12 Use the following example of an open channel to show that the selection of an inappropriate n causes a much larger variation in the computed flow rate, than does a corresponding percent error in the selection of e. D = 3 m, Y = 2 m, S_o = 0.0008. Assume the correct value of n = 0.013, but you use n = 0.015 (a 15% error), and that the correct e = 0.00057 m, but you use e = 0.00066 m (also a 15% error) ($\nu = 1.519 \times 10^{-6}$ m²/s).

2.13 Determine the appropriate bottom width that should be used in the design of an open channel with a trapezoidal cross section and a side slope of m = 1.6, if the depth of flow is to be 5.8 m when the flow rate is 550 m³/s. The slope of the channel bottom is S_o = 0.00113, and its wall roughness is e = 0.0015 m.

2.14 Water enters a trapezoidal canal from a reservoir at the uniform depth of 5.2 ft. The canal is to convey 450 cfs over a long distance at a slope of 0.0005, and stability considerations dictate that the side slope should be 2 to 1. If the wall roughness e = 0.012 ft, determine what the bottom width of the canal should be.

2.15 The following data defines the cross section of an earthen canal, in which x represents the distance in feet from the left bank when looking downstream, and y represents the corresponding vertical distance from this bank elevation to the elevation of the canal bottom.

x (ft)	0	3	5	7	10	12	14	18	21	25	27	29
y (ft)	0	1.4	2.3	2.8	3.4	3.5	3.3	2.8	1.5	0.9	0.2	0.0

Estimate the flow rate in the canal if its bottom slope is 0.001, and the depth of flow is 3.2 ft.

2.16 Use Manning's formula to fill in the blank in the table below that applies for uniform flow in channels of trapezoidal shapes.

Flow Rate Q (cfs)	Coeff. n	Bot. Slope S_o	Bot. Width b (ft)	Side Slope m	Depth Y (ft)
—	0.013	0.0006	6.0	1.6	4.2
324.5	—	0.0008	8.0	0.0	5.0
550.0	0.014	—	10.0	1.3	4.8
400.0	0.013	0.0015	—	1.5	4.0
450.0	0.013	0.0009	7.0	—	4.1
600.0	0.013	0.00075	10.0	1.2	—

2.17 Fill in the missing blank in the table below using Manning's formula. These are trapezoidal channel, and the flow in them is uniform.

Flow Rate Q (m^3/s)	Coeff. n	Bot. Slope S_o	Bot. Width b (m)	Side Slope m	Depth Y (m)
—	.014	0.0015	3.0	1.0	2.3
50.0	—	0.0012	3.0	1.5	2.2
60.0	.013	—	2.5	2.0	2.5
100.0	.015	0.0008	—	2.0	3.0
80.0	.014	0.00075	6.0	—	2.9
20.0	.0135	0.00068	2.0	1.5	—

2.18 For each entry in the table for Problem 2.17 compute the value of Chezy's C coefficient, and from this C determine what the wall roughness e is.

2.19 Write a computer program that is designed to solve Manning's formula completely for a trapezoidal section. It should display a menu of variables, allow the user to select the unknown, then request values for the knowns, and finally provide the solution to the unknown.

2.20 The table below applies for uniform flow in circular channels. Fill in the mission blanks.

Flow Rate Q (cfs)	Coeff. n	Bot. Slope S_o	Diameter D (ft)	Depth Y (ft)
—	.0142	0.00055	10.0	6.8
300.0	—	0.00065	10.0	7.2
250.0	.013	—	5.0	3.2
200.0	.013	0.03000	—	2.5
100.0	.013	0.00100	8.0	—

2.21 The table below applies for uniform flow in circular channels. Fill in the mission blanks.

Flow Rate $Q(m^3/s)$	Coeff. n	Bot. Slope S_o	Diameter D (m)	Depth Y (m)
—	.015	0.00080	3.0	2.3
20.0	—	0.00068	5.0	2.52
25.0	.013	—	6.0	2.9
30.0	.0125	0.00040	—	3.0
60.0	.0145	0.00045	8.0	—

2.22 Compute the normal depth in a trapezoidal channel with b = 10 ft, m = 1.5, S_o = 0.0008 if Q = 400 cfs is flowing in the channel and it has a bottom roughness coefficient for Manning's equation of n_b = 0.05, and a side roughness coefficient n_s = 0.013.

2.23 A rectangular laboratory flume has Plexiglas sides (n_s = 0.010) and gravel on its bottom (n_b = 0.035). The flume is 3 ft wide has a bottom slope of S_o = 0.001. If the depth of flow is 1.5 ft estimate the flow rate in the flume.

2.24 A 8 ft wide rectangular channel made of concrete (n = 0.013) is filled to a depth of 1 ft with gravel (n = 0.04). The slope of the channel is S_o = 0.0015. For a flow rate of Q = 90 cfs predict the height of water in the channel with and without the gravel.

2.25 Determine the alternate depth in a rectangular channel to a depth Y_2 = 0.2 m if the flow rate is q = 3 m^2/s.

2.26 Determine the alternate depth to 1.5 ft in a trapezoidal channel with b = 5 ft, and m = 1.2 if the flow rate is Q = 150 cfs.

2.27 Determine the depth that will occur upstream of gate in a trapezoidal channel with b = 6 ft, and m = 1.2 if the flow rate is Q = 200 cfs, and the channel bottom rises in elevation by 1.5 ft across the gate, and the depth of flow downstream of the gate is Y_2 = 2.0 ft.

2.28 Determine the alternate depth to 4 m in a trapezoidal channel with b = 3 m, and m = 1.5 if the flow rate is Q = 30 m^3/s.

2.29 The depth of flow downstream from a sluice gate in a rectangular channel is 1.2 ft. If the flow rate per unit width under the gate is q = 20 cfs/ft determine the depth upstream of the gate.

2.30 The depths upstream and downstream from a sluice gate in a rectangular channel are measured to be 5.2 and 1.1 ft, respectively. If the channel is 15 ft wide, determine the flow rate passing under the gate.

2.31 A specific energy diagram has the specific energy E along the abscissa and the depth Y along the ordinate with different curves for constant values of flow rate Q. Generate tables of values, and then plot these to make a depth–discharge diagram for constant specific energies, i.e., graph Q on the abscissa, Y on the ordinate and different curves apply for constant values of the specific. Make this graph specific for a trapezoidal channel with b = 10 ft, m = 1.25, and use three curves for E = 3 ft, E = 4 ft, and E = 5 ft. Prove that the maximum flow rate for any constant E corresponds to the critical depth Y_c.

2.32 The depth upstream from a sluice gate in a rectangular channel 4.1 m. The gate is set at a distance 0.5 m above the bottom of the channel, and the contraction coefficient for the gate is 0.58. Determine the flow rate passing the gate per unit width.

2.33 A contraction reduces a rectangular channel from b = 4 m to b = 3 m. The downstream channel has a bottom slope S_{o2} = 0.0015, and n_2 = 0.013. For a flow rate Q = 10 m^3/s determine the upstream depth and the change in water surface elevation. The bottom elevation of the channel does not change through the contraction.

2.34 In Problem 2.33, the bottom rises 0.2 m. Now determine the upstream depth and change in water surface elevation.

2.35 In Problem 2.33 the bottom fall 0.2 m. Now determine the upstream depth and the change in water surface elevation.

2.36 How much must the bottom rise or fall in Problem 2.33 so that critical flow occurs at the smallest section of the contraction, but the water level upstream is the same as in Problem 2.33?

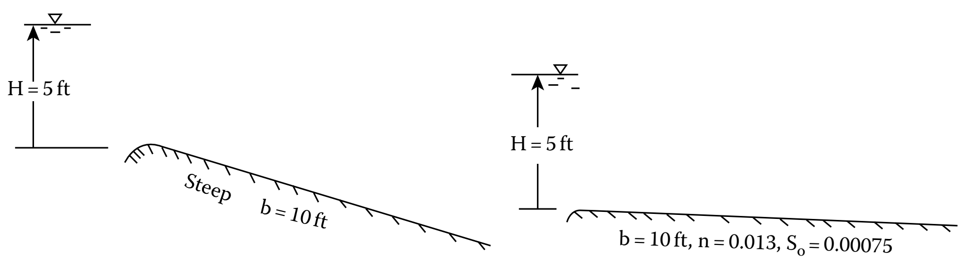
2.37 A transition from a trapezoidal channel with $b_1 = 10$ ft, $m_1 = 1.5$, and $S_{o1} = 0.0001$ to a circular channel with $D_2 = 8$ ft, $n_2 = 0.012$, and $S_{o2} = 0.058$. What are the depths at the upstream and downstream ends of this transition if the flow rate $Q = 450$ cfs?

2.38 Water from a reservoir with a water surface elevation 5 m above the bottom of a 10 m wide and steep rectangular channel enters it unrestricted. Determine the flow rate.

2.39 A gate exists in Problem 2.38 with a contraction coefficient of 0.8 to control the flow rate. If the gate is 0.3 m above the channel bottom what is the flow rate?

2.40 A transition occurs between an upstream trapezoidal channel with a bottom width $b_1 = 3$ m, and a side slope $m_1 = 1.5$ to a rectangular section with a bottom width $b_2 = 2.5$ m. Simultaneously with the reduction of width the bottom drops by $\Delta z = 0.2$ m from the trapezoidal to the rectangular sections. If the bottom slope of the trapezoidal channel is $S_{o1} = 0.0006$ and has a Manning's roughness coefficient $n = 0.013$, determine the maximum flow rate that can pass through the transition without increasing the depth upstream therefrom above the uniform depth. What is the maximum flow rate if the bottom remains horizontal across the transition?

2.41 A steep and a mild rectangular channel with a bottom width of $b = 10$ ft take water from a reservoir whose water surface is 5 ft above the channel bottom. The mild channel has a bottom slope $S_o = 0.00075$, and a Manning's roughness coefficient $n = 0.013$. Determine the flow rate in both of these channels. Assume the entrance loss coefficient is 0.



2.42 A transition from a trapezoidal channel with $b_1 = 4$ m, $m_1 = 1$ to a rectangular channel with $b_2 = 3$ m occurs. The bottom of the channel remains at the same elevation, and both channels have a Manning's roughness coefficient, $n_1 = n_2 = 0.014$. The upstream channel has a bottom slope $S_{o1} = 0.0005$, and the downstream channel has a bottom slope $S_{o2} = 0.009$. If a flow rate of $Q = 24$ m^3/s is occurring what are the depths at the beginning of the transition and at its end?

What change in elevation of the bottom of the channel would be necessary so the upstream depth remains at uniform depth?

What should the width of the downstream rectangular channel be if uniform flow to exist throughout the upstream channel if the bottom elevation does not change?

2.43 Generate the data needed to plot a dimensionless specific energy graph that has the subcritical dimensionless depth $Y_1' = mY_1/b$ as the abscissa and the supercritical depth $Y_2' = mY_2/b$ as the ordinate that applies for trapezoidal channels, and create this graph. The graph should have curves for $Q' = m^3Q^2/(gb^5)$ the same as Figure 2.4.

2.44 Repeat the graph of the previous problem except make it apply for circular channels.

2.45 Develop a relationship for the diameter D in a circular channel with the bottom width b and the side slope m in a trapezoidal channel so the dimensionless flow rates Q' are the same in these two sections. Assume that a circular section exists downstream from a gate with a diameter that satisfied the above relationship, and that a trapezoidal section with b and m exists upstream from the gate. Develop the data and plot three graphs that apply for $m = 0.5$, $m = 1.0$, and $m = 1.5$ that provide the relationship between the downstream supercritical depth in the circular section to the upstream subcritical depth in the trapezoidal section, assuming that the above relationship of D to b and m applies so the dimensionless depths are the same in the two sections.

2.46 Develop a graphical solution that provides the depth downstream from a gate if the depth upstream is known, or provides the depth upstream from the gate if the depth downstream therefrom is known. Have this graphical solution apply for flow rates per unit width of rectangular channel from q = 1 to 10 m^2/s. Note that if a log–log plot is used that the curves for the different unit flow rates are not nearly as curved as when the plot uses a linear horizontal and vertical axis. The graphical solution can be developed rather easily using a spreadsheet since the equations that provide the alternative depths are explicit for a rectangular channel. Also write a computer program that provides a table of these solutions for different flow rates, q. (You should also generate the above tables using CHANNEL.)

2.47 Repeat the previous problem for an upstream trapezoidal channel with a bottom width b_1 = 3 m and a side slope m = 1.5 and a rectangular channel at the position of the gate with a bottom width b_2 = 2.5 m. Use flow rates of Q = 3 to 25 m^3/s.

2.48 A pipe with a diameter D = 6 m and on a steep slope takes water from a reservoir with a water surface elevation 5 m above the bottom of the pipe. What flow rate might be expected to enter the pipe?

2.49 A trapezoidal channel with b = 7 ft, and m = 1.5 and a bottom slope of S_o = 0.08 gets water directly from a reservoir with its water surface 4.5 ft above the channel bottom. What flow rate would you predict will enter this channel?

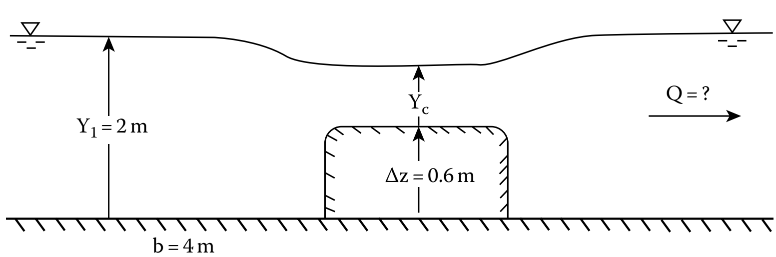
2.50 A transition takes a trapezoidal channel with b = 10 ft, and m = 1.5 to a rectangular section with b = 8 ft. Through the transition the bottom of the channel rises 0.2 ft. The slope of the very long upstream trapezoidal channel is S_o = 0.00015, and its n = 0.014. For a flow rate of Q = 330 cfs determine the following: (a) the depth upstream from the transition, (b) the depth immediately downstream from the transition, (c) the change in water surface through the transition, and (d) the Froude numbers associated with both the upstream flow and that immediate downstream from the transition. If the depth in the rectangular channel is not to change downstream from the transition what should its slope be? Its n = 0.014 also. If its slope is less than this amount what will happen?

2.51 If the water surface is not to change through a transition from a trapezoidal to a rectangular channel with the dimensions in the previous problem how much must the bottom rise or fall as a tabular function of the upstream depth? Plot this function.

2.52 What is the maximum rise that can take place in the bottom elevation through the transition in Problem 2.50 for the upstream flow to be possible?

2.53 Everything is the same as in Problem 2.50 except that the bottom rises by 0.8 ft through the transition. What is the depth upstream?

2.54 A broad crested weir consists of a rounded rectangular hump placed in the bottom of a channel that is sufficiently high to cause critical flow over its top, and is a common flow measurement device. If the depth Y_1 upstream from a broad crested weir, that is 0.6 m high, is 2 m, and the rectangular channel containing the weir is b = 4 m wide, what is the flow rate Q? What is the depth over the crest of the weir?



2.55 For the flow over the broad crested weir of the previous problem plot the function F(q) on the ordinate, and q on the abscissa of a graph and show that two solutions exist, and demonstrate that one of these is associated with an upstream supercritical flow, and the other subcritical. However, since physically the supercritical condition cannot exist that the only physically viable solution is the larger of the two roots. (Note that if the upstream flow were supercritical that a hydraulic jump would form upstream so that the flow is subcritical in front of the weir so that the flow depth decreases as the velocity head increases to pass the reduced flow section.)

2.56 Water enters a trapezoidal channel with $b = 8$ ft, $m = 2.0$, and a bottom slope of $S_o = 0.000115$ from a reservoir whose water surface is 8 ft above the channel bottom. If the entrance loss coefficient is $K_L = 0.04$ determine the flow rate and depth of flow in this channel. The channel is very long, and $n = 0.013$.

2.57 Gates control flow in a trapezoidal channel with $b = 6$ m, $m = 1.5$, $n = 0.014$, and $S_o = 0.00025$. There are three rectangular gates each 1.5 m wide with contraction coefficients of $C_c = 0.8$. Two gates are set 0.5 m above the channel bottom and the third gate is set 0.2 m above the channel bottom. The depth upstream from the gates is 4.2 m. Determine the flow rate in the channel. How much is the upstream depth above or below the uniform (normal) depth in this channel?

2.58 The land slopes at a rate of 0.0016 ft per foot of length and you are to design a trapezoidal channel that will carry 400 cfs of water when the depth of water in the reservoir is 5 ft above its bottom. The channel is to have a side slope of $m = 1.4$, and a Manning's $n = 0.013$. What should its bottom width be? What is the Froude number associated with this flow?

2.59 Develop the stage discharge curve for the channel you designed in the previous problem with the reservoir water surface elevation varying from 0 to 5.5 ft above the channel bottom.

2.60 Determine the size of pipe that should be used to convey a flow rate of $Q = 16\,m^3/s$ from a reservoir whose water surface elevation is 3 m above the pipe's bottom. The pipe is to have a bottom slope of $S_o = 0.00035$, and a Manning's $n = 0.014$. The entrance loss coefficient $K_L = 0.1$. What will the depth of flow be in this channel? What is the Froude number associated with this flow.

2.61 A 3 m diameter pipe with a steep bottom slope is taking water from a reservoir with a depth of 2.8 m above the pipes bottom. What flow rate would you estimate is entering the pipe, and at what depth?

2.62 You are to size a pipe that will take $30\,m^3/s$ from a reservoir whose water surface elevation is 3 m above the pipe's bottom, and the depth at its entrance in the pipe is not to exceed 2 m. Determine the size of pipe to use. After a very short distance a transition occurs to a trapezoidal section with $b = 6$ m, and $m = 1.2$. What is the depth in this trapezoidal section? What must the bottom slope of this trapezoidal channel be if this depth is not to change, and its Manning's roughness coefficient $n = 0.013$?

2.63 Water is to be taken from a reservoir with $H = 6$ ft using a trapezoidal channel with $b = 8$ m and $m = 1.5$. After a very short distance the section changes to that of a pipe with a diameter $D = 6$ m. The entrance loss coefficient including the transition is $K_L = 0.15$. What will the flow rate be if the roughness of the pipe wall is $n = 0.013$ and its slope is $S_o = 0.0008$? What will the depth be at the entrance of the pipe? What will the depth be at the entrance of the trapezoidal channel?

2.64 Repeat the previous problem except the pipe is made of corrugated steel with a roughness of $n = 0.035$.

2.65 Water enters a trapezoidal channel from a reservoirs whose bottom has a Manning's roughness coefficient of $n_b = 0.035$, and whose sides have a Manning's roughness coefficient $n_s = 0.012$. The channels bottom width is $b = 5$ m, and its side slope $m = 1.5$ and its bottom slope is $S_o = 0.00085$. If the depth of water in the reservoir is 3.2 m above the channel bottom and the entrance loss coefficient is $K_e = 0.08$ determine the discharge into the channel.

(Solve this problem by using an equivalent n_{eq} based on weighting according to the fraction of the perimeter each roughness applies for.)

2.66 Repeat the previous problem except assume $n_b = 0.012$ and $n_s = 0.035$. Also solve Problem 2.65 assuming that $nP^{2/3}$ in Manning's equation should be replace by $(nP^{2/3})_b$ plus $(nP^{2/3})_s$. Why is the depth obtained by an equivalent Manning's n different?

2.67 Develop the delivery diagrams for a trapezoidal channel with b = 3 m, m = 1.4, n = 0.013 for bottom slopes varying from $S_o = 0.001$ to $S_o = 0.003$. The entrance loss coefficient is $K_e = 0.12$ and the reservoir head varies from H = 0.2 m to H = 2.2 m.

2.68 A 10 ft diameter pipe that has a Manning's n = 0.012 takes water from a reservoir whose head varies between H = 1 ft and H = 10 ft. The entrance loss coefficient is $K_e = 0.12$. Obtain delivery diagrams for this long channel for bottom slopes S_o = 0.001, 0.002, 0.0025, 0.0028, and 0.003.

2.69 For the 12 ft diameter pipe with n = 0.012, that has a trapezoidal entrance section through which water from the reservoir passes before entering the pipe, and that was solved in the text with Program E_UNTC, develop the delivery diagrams for H varying from 1 to 8 ft (b = 10′, m = 1.2, $K_e = 0.1$, $K_L = 0.1$).

2.70 Program E_UNTC assumes uniform flow occurs in the downstream circular channel. Modify this program so critical depth may occur in either the upstream entrance (trapezoidal section), or at the beginning of the circular channel. Use this program to solve the problem solved in the text except that the channel is steep.

2.71 In the text the program E_UNTC is designed to solve flow rate into a circular channel that has a trapezoidal entrance where it receives water from a reservoir. Write a computer program, or develop a computer model, to solve problems in which a trapezoidal channel has a circular section at its entrance to the supply reservoir. Use this program (or model) to solve for Q, Y_1 and Y_2 from H = 3 m, D = 5 m, $K_e = 0.1$, b = 3 m, m = 1.4, n = 0.013, $S_o = 0.001$, $K_L = 0.1$ and $\Delta z = 0.8$ m.

2.72 For the channel with the transitional entrance of the previous problem develop the delivery diagrams with H varying from 0.25 to 5 m for bottom slopes varying from $S_o = 0.0002$ to $S_o = 0.0015$.

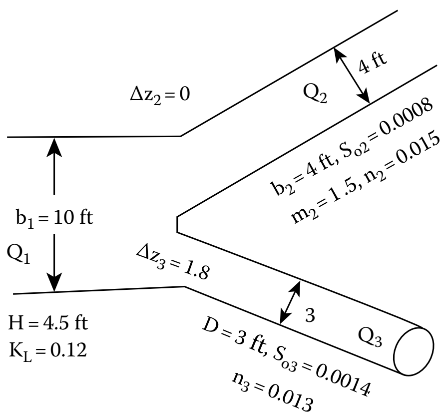
2.73 In developing the delivery diagrams in the previous problem you should have computed supercritical depths at the beginning of the trapezoidal section when critical flow occurs in the circular entrance. To have smooth flow from this position on, when this condition of critical flow occurs, the channel must have a bottom slope sufficient to maintain uniform flow at this depth, or a steep slope. Compute these slopes and compare them with S_c needed to separate a mild from s steep channel if the trapezoidal channel receives water directly from the reservoir.

2.74 For the trapezoidal channel with a circular entrance of Problem 2.71 compute the change in bottom elevation Δz needed so that critical flow occurs in both the circular section and at the beginning of the trapezoidal channel when the reservoir head is H = 2.5 m.

2.75 Develop an iterative solution for either the subcritical, or supercritical depth corresponding to a given value of the specific energy E in a trapezoidal channel based on being able to solve the cubic equation for the three real roots if the channel is rectangular. The development of this iterative solution can be based on the follow two observations. (1) When the channel is rectangular, the root Y_2 (or the root obtained by adding 2π to θ to make the argument for the cosine) is the subcritical depth, and root Y_3 (obtained by adding 4π to θ to make the argument of the cosine) is the supercritical depth. (2) A mean flow rate per unit width q in a trapezoidal channel can be define by $q = 2Q/(T + b) = Q/(mY + b) = Q/(A/Y)$. Use this iterative approach to solve both the sub- and supercritical depth in a 10 ft wide trapezoidal channel with m = 1 if E = 5 ft, and the flow rate is Q = 400 cfs.

2.76 Modify the iterative solution method developed to solve for the depths associated with a given specific energy E in a trapezoidal channel of the previous problem to find the subcritical depth, or the supercritical depth in a circular channel with a known diameter, and a specified value of the specific energy E.

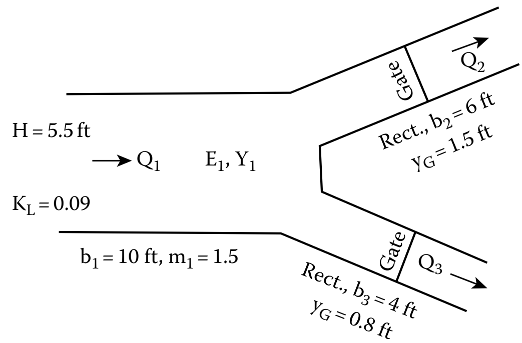
2.77 Water is taken from a reservoir by means of a rectangular inlet channel that is 10 ft wide. A short distance downstream therefrom the channel divides in a trapezoidal section with $b_2 = 4$ ft, $m_2 = 1.5$, $n_2 = 0.015$, and $S_{o2} = 0.0008$, and a pipe with a diameter $D_3 = 3$ ft, $n_3 = 0.013$, and $S_{o3} = 0.0014$. The bottom of the pipe is 1.8 ft above the bottom of the rectangular channel, and the trapezoidal and rectangular channel have the same bottom elevation. When the water surface elevation in the reservoir is 4.5 ft above the bottom of the channel determine the depths and flow rates in all three channel (six unknowns). Assume the entrance loss coefficient equals 0.12.



2.78 Modify the program THREECH to accommodate any number of channels branching from the upstream main channel, i.e., allow the number of channels to be 4, 5, etc. Also make the following changes: (1) In place of the FUNCTION F make this a subroutine (a void function in C) that supplies all of the equations each time it is called, and (2) Rather than have a built in linear algebra solver when implementing the Newton method, call on a linear algebra subroutine such as SOLVEQ. Use this modified program to solve the following problem. A reservoir with a head H = 5 ft, and an entrance loss coefficient, $K_e = 0.1$, supplies a main trapezoidal channel with $b_1 = 10$ ft and m = 1.5. A short distance downstream this channel branches into three identical long trapezoidal channels with b = 4, m = 1, n = 0.013 and $S_o = 0.0005$. Solve for the flow rates and depths in the four channels.

2.79 For the four channel system of the previous problem develop the delivery diagram for reservoir heads varying from H = 1 ft to H = 8 ft in increments of $\Delta H = 0.25$ ft.

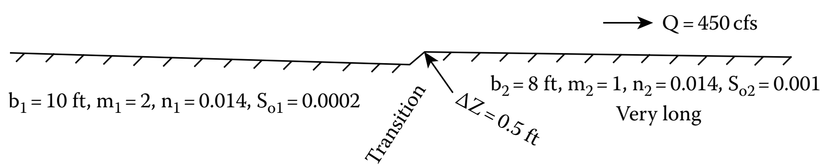
2.80 A trapezoidal channel with $b_1 = 10$ ft, and $m_1 = 1.5$ takes water from a reservoir with a water surface elevation 5.5 ft above its bottom. As short distance downstream therefrom it divides into two rectangular channels with the following properties: $b_2 = 6$ ft, and $n_2 = 0.014$, and $b_3 = 4$ ft, and $n_3 = 0.013$. A short distance downstream in the two rectangular channels there are gates to control the flow rate. The gates are set at $y_{G2} = 1.5$ ft and $y_{G3} = 0.8$ ft above the bottom of the channel, respectively. The bottom of all three channels is at the same elevation, and the contraction coefficients for the gates are both 0.6. If the entrance loss coefficient equals $K_L = 0.09$. Determine the depths in all three channels upstream from the gates, and the flow rates in each of the three channels.



2.81 Write a computer program, or use an available software program capable of solving system of nonlinear simultaneous equations to develop the depths and discharges that will occur in all three channels of the previous problem with the water surface elevation in the reservoir at 5.5 ft, but with different gate setting in channel two varying from y_{G2} = 0 ft to wide open. This second rectangular channel has a bottom slope of S_{o2} = 0.0005, and extends downstream for a very long distance.

2.82 Assume the gate in channel 2 of Problem 2.80 is wide open and that this channel has a steep bottom slope. Also its bottom is 1.5 ft above the bottom of the other two channels. What will the depths and flow rates be in all three channels now.

2.83 Two long channels are joined by a smooth transition. The upstream channel has the following properties: Its bottom width is b_1 = 10 ft; its side slope is m_1 = 2; its Manning's roughness coefficient is n_1 = 0.014, and its bottom slope is, S_{o1} = 0.0002. The downstream channel has the following properties: b_2 = 8 ft, m_2 = 1.0, n_2 = 0.014, and S_{o2} = 0.001. The bottom rises by Δz = 0.5 ft through the transition. The design flow rate is Q = 450 cfs. Determine the depths both immediately upstream and downstream from the transition.



2.84 Write a computer program capable of solving any of the variables that may be unknown through a transition between two trapezoidal channels. The variables of the problem are: Y_1, b_1, m_1, Δz, Y_2, b_2, m_2, and Q.

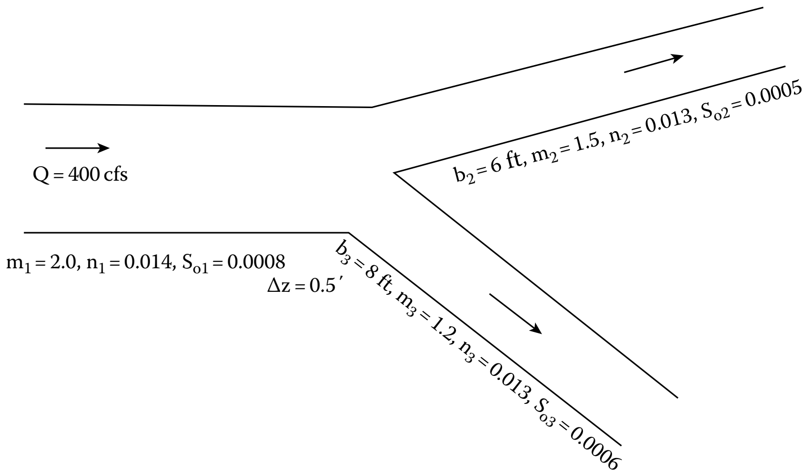
2.85 Write a computer program capable of solving any of the variables that may be unknown through a transition between two circular channels. The variables of the problem are: Y_1, D_1, Δz, Y_2, D_2, and Q.

2.86 Write a computer program to solve the problem of water entering a trapezoidal channel from a reservoir at uniform flow. This program should be able to solve any of the variables in the following list in addition to determine the flow rate Q: H, b, m, K_L, S_o, or n. (H is the head of water in reservoir above the channel bottom.)

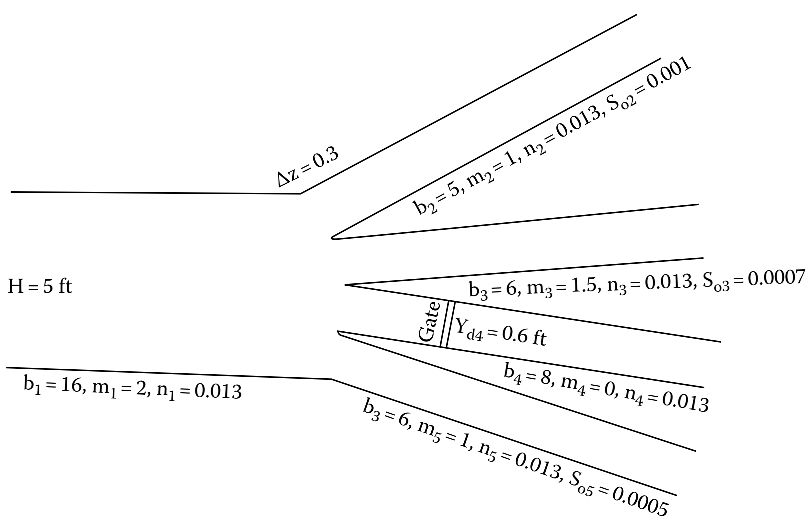
2.87 Write a computer program to solve the problem of water entering a circular channel from a reservoir at uniform flow. This program should be able to solve any of the variables in the following list in addition to determine the flow rate Q: H, D, K_L, S_o, or n.

2.88 A trapezoidal channel is to be designed to carry a flow rate Q = 400 cfs. The single channel divides into two trapezoidal channels with the first having a bottom width b_2 = 6 ft, a side slope m_2 = 1.5, a Manning's n_2 = 0.013, and a bottom slope S_{o2} = 0.0005, and the second having a bottom width b_3 = 8 ft, a side slope m_3 = 1.2, a Manning's n_3 = 0.013 and a bottom slope $S_{o3} = 0.0006$. The bottom of the channel with a bottom width of 8 ft is 0.5 ft above the other divided channel. The upstream single channel is to have a bottom slope of $S_{o1} = 0.0008$, $m_1 = 2.0$, and a Manning's $n_1 = 0.014$. Determine the depths and flow rates in the downstream divided channels, and the bottom width and depth in the upstream channel so that uniform flow will occur in it when the above design flow rate occurs.

Solution: $Q_2 = 202.61$ cfs, $Q_3 = 197.39$ cfs, $Y_2 = 3.87$ ft, $Y_3 = 3.28$ ft, $b_1 = 12.13$ ft, and $Y_1 = 3.69$ ft.



2.89 Four channels branch from the upstream main channel as shown in the sketch below. The sizes, etc. of the channel are as shown on the sketch, including the proposed width of 16 ft for the upstream main channel. A gate exists a short distance downstream in channel 4, with a setting that produces a depth downstream from it of 0.6 ft. Also notice that the bottom of channel 2 is 0.3 ft above the bottoms of the other channels. Is it possible to have this upstream channel 16 ft and not restrict the flow that the other channels can carry under the conditions given? Solve for the flow rates and depths in all of the channels if the upstream trapezoidal channel has a bottom width $b_1 = 20$ ft, and a side slope $m_1 = 2$.



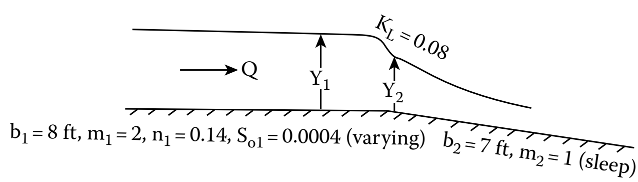
2.90 Determine the minimum width b_1 that the upstream channel of the previous channel system must have so that critical flow will not occur at its entrance, thus limiting the flow rate that the downstream channels can carry. You might use one of the models you used to solve the previous problem in which you attempt to successively reduct the width from 20 ft until it fails to produce a solution to get an idea how the upstream depth decrease with decreasing b_1 until critical depth occurs. Why are the flow rates and depths of the channels downstream from the junction not affected by changing widths of the upstream channel until critical conditions occur?

2.91 Fix the upstream width of channel #1 in the system of the previous two problems to $b_1 = 19$ ft, and obtain a series of solutions in which the reservoir water surface H increases from 5 ft until a solution no longer exists. What caused this situation with rising reservoir heads?

2.92 Enlarge the downstream steep rectangular channel in illustrative Example Problem 2.9 from $b_2 = 8$ ft to $b_2 = 10$ ft, and repeat this problem including solving values of Q and Y for a number of different bottom slopes for the upstream channel.

2.93 A trapezoidal channel with $b_1 = 8$ ft, and $m_1 = 1.5$ is laid on a bottom slope of $S_{o1} = 0.0005$, and has a Manning's roughness coefficient, $n_1 = 0.013$. This channel conveys the water into a smooth transition to a circular channel with a diameter $D_2 = 18.2$ ft. The circular channel has a steep slope. Determine the maximum flow rate that can exist, and not increase the depth in the upstream channel above its normal depth, through the transition if the bottom elevation does not change between the trapezoidal and circular channels. Determine what the relationship is between the maximum flow rate possible and the slope of the upstream channel by solving the problem for several different values of S_{o1}.

2.94 Determine the maximum flow rate that can be accommodated in the channel shown below without causing the depth in the upstream channel to rise above its normal depth. The upstream channel has a bottom width $b_1 = 8$ ft, a side slope $m_1 = 2$, a Manning's roughness coefficient, $n_1 = 0.014$, and a bottom slope, $S_{o1} = 0.00025$. The downstream channel is steep, i.e., under uniform flow conditions the depth will be less than critical depth, has a bottom width $b_2 = 7$ ft, and a side slope $m_2 = 1$. Solve the problem with the upstream slope changing and plot the maximum flow rate, and the depths upstream and at the head of the steep channel versus S_{o1}. Can you explain these trends? (Increase S_o to about 0.001.)



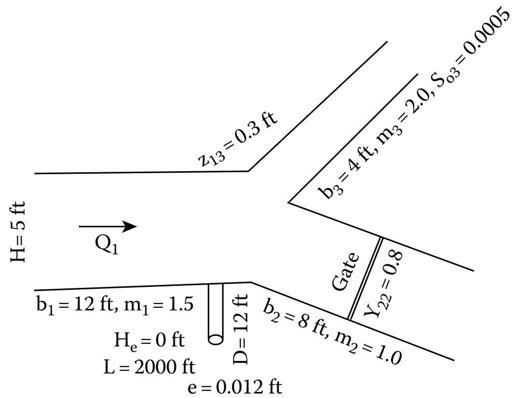
2.95 A transition between an upstream mild channel and a downstream steep channel is smooth with no change in the bottom position. The upstream channel has $b_1 = 12$ ft, $m_1 = 2$, $n_1 = 0.014$, and a bottom slope $S_{o1} = 0.0002$. The downstream steep channel has a bottom width $b_2 = 8$ ft and a side slope $m_2 = 1$. Determine the maximum flow rate possible in this channel if the flow in the upstream channel is to be uniform. What is the depth of this upstream uniform flow, and what is the depth at the beginning of the steep channel?

2.96 In the previous problem obtain the solutions for the same unknowns for the downstream channel having a side slope of $m_2 = 0.5$ and $m_2 = 0$, respectively. The upstream mild channel has the same size, slope, etc. as in the previous problem and the bottom width of the steep downstream channel is $b_2 = 8$ ft.

2.97 Solve Problem 2.95 for several width of downstream rectangular channel with bottom widths varying from $b_2 = 6$ ft to $b_2 = 12$ ft ($m_2 = 0$). Note from these solution how significant the choking effect is as the size of the downstream steep channel is reduced in size.

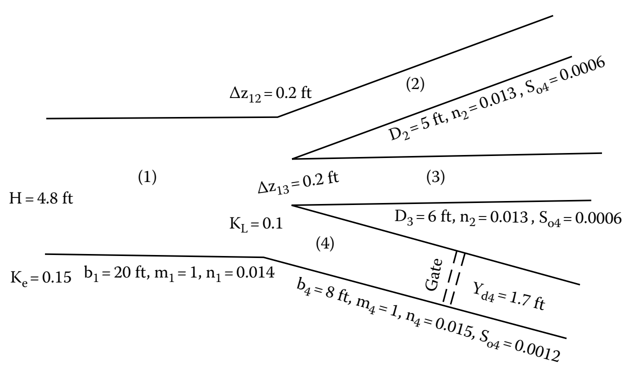
2.98 Water is taken from a reservoir with a water surface elevation 5 ft above a 12 ft wide, trapezoidal channel with a side slope $m_1 = 1.5$. A short distance downstream from the channel entrance it divides into two trapezoidal channels with b_2 = 8 ft, m_2 = 1.0, and b_3 = 4 ft, and m_3 = 2. The second channel has a gate in it a short distance downstream from the branch that causes the depth of flow downstream from it to be at Y_{22} = 0.8 ft. The third channel is long, has a Manning's n_3 = 0.015, and a bottom slope S_{o3} = 0.0005. The bottom rises by Z_{13} = 0.3 ft between the upstream channel and the third channel. Just upstream from the junction a 12 in. diameter pipe takes water from near the bottom of the channel. This pipe is 2000 ft long, has an equivalent sand roughness e = 0.012 in., and delivers water at it end with a head H_e = 0 ft. Determine the flow rates in all channels and the pipe, as well as the depths in the channels.

Solution: Q_1 = 367.6 cfs, Q_2 = 114.9 cfs, Q_3 = 250.6 cfs, Q_4 = 2.11 cfs, Y_1 = 4.683, Y_2 = 4.885, Y_3 = 4.639.



2.99 Example Problem 2.18 obtains a series of solution for a four channel branched system in which the height of the gate in channel 3 varies from 1.7 ft to being closed. Solve this same channel system except the upstream channel # 1 has a width of 12 ft rather than 15 ft. The gate's contraction coefficient is C_c = 0.6. Obtain this series of solutions with the gate's position starting at 0.1 ft.

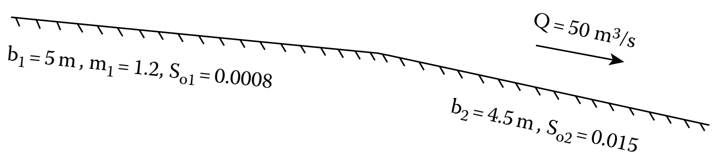
2.100 A reservoir with a water surface elevation 4.8 ft above the bottom of a trapezoidal channel with b_1 = 20 ft, m_1 = 1, n_1 = 0.014, supplies water to a branched channel system as shown, in which this upstream channel divides into three channels; channels 2 and 3 consisting of pipes with D_2 = 5 ft, n_2 = 0.013, S_{o2} = 0.0006, and D_3 = 6 ft, n_2 = 0.013, S_{o3} = 0.0005. Channel 4 is trapezoidal with b_4 = 8 ft, m_4 = 1, n_4 = 0.015. This channel contains a gate that produces a depth of 1.7 ft downstream from it. The entrance loss coefficient is K_e = 0.15, and the loss coefficients to the 3 branched channels are all 0.1. The bottoms of the pipes for channels 2 and 3 are $\Delta z_{12} = \Delta z_{13}$ = 0.2 ft above the bottom of channel 1. Solve for the flow rates and depths in these four channels.



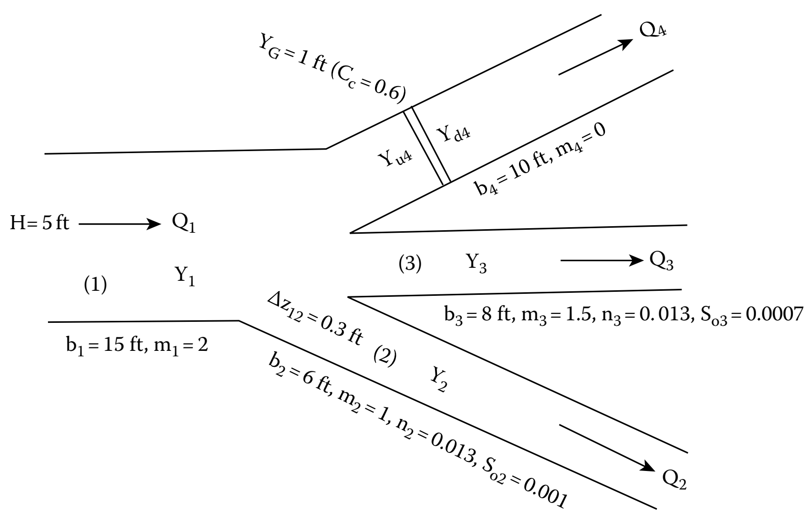
2.101 Example Problem 2.19 gave the answers that were obtained by solving the problem using, but did not actually provide the sheets of this solution. Using TK-Solver, Mathcad, or other software capable of solving a system of equations verify the given answers. Obtain these answers: (1) by using the seven equations given in this example problem (you may add areas and wetted perimeters to the list of unknown variables if you want) but specify the value of $Q_1 = 531.4$, and (2) Also include Q_1 as an unknown and include the additional equations needed.

2.102 Water enters a rectangular channel with a 20 ft bottom width from a reservoir whose water surface is 9 ft above the channel bottom. The channel has a Manning's roughness coefficient n = 0.014. (a) If the slope of the channel bottom is $S_o = 0.055$ what is the flow rate into the channel? (b) If the slope of the channel bottom is $S_o = 0.0005$ what is the flow rate into the channel? (c) A vertical gate is placed a short distance downstream from the channel entrance with its tip 2 ft above the channel bottom. The gates contraction coefficient is $C_c = 0.6$. Now what is the flow rate?

2.103 A trapezoidal channel with $b_1 = 5$ m, $m_1 = 1.2$, and $S_{o1} = 0.0008$ changes to a rectangular channel with a bottom width of $b_2 = 4.5$ m and a bottom slope $S_{o2} = 0.015$. Both channels have n = 0.014 and the bottom elevation remains constant through the transition. For a flow rate $Q = 50\,m^3/s$ what will the depths be immediately upstream and at the end of the transition? What is the change in water surface through the transition? As a (b) part to this problem, what will the depths be if the bottom drops by 1 m through the transition, i.e., $\Delta z = -1$.



2.104 Give the equations that need to be solved to determine the flow rate in each of the four channels shown in the sketch below. The upstream main channel is short and is supplied by a reservoir with a water surface elevation of H = 5 ft. The bottom elevation of channel 2 is 0.3 ft above the bottom of the main channel and the other channel's bottoms are at the same elevation. Channel 4 contains a sluice gate with a contraction coefficient, $C_c = 0.6$, and its gate's position above the channel bottom is $Y_G = 1.0$ ft. Ignore all minor losses



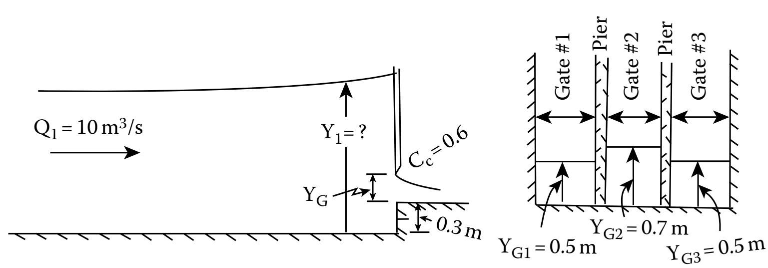
2.105 Write a computer program, or use a software package such as Mathcad, or TK-SOLVER to obtain the solution to Example Problem 2.19.

2.106 Solve the same problem as given as Example Problem 2.19 with the exception that the bottom slopes of the three channels downstream from the branch are: $S_{o2} = S_{o3} = S_{o4} = 0.0005$.

2.107 Solve the same problem as given as Example Problem 2.19 with the exception that the bottom slopes of the 3 channels downstream from the branch are: $S_{o2} = S_{o3} = S_{o4} = 0.0003$. What are the flow rates and depths in the four channels when $S_{o2} = S_{o3} = S_{o4} = 0.0002$.

2.108 The program BRANCHCH.FOR will solve problems in which the critical depth at the entrance of the channel from the reservoir restricts the flow rate. However, to obtain such a solution it is necessary that you previously determine what this restrictive flow rate is. Modify the program so that if it fail to converge, or if you specify that critical flow governs that it will solve the problem without the necessity of giving what this restricting flow rate is.

2.109 At the location of sluice gates in a trapezoidal channel with b = 5 m and m = 1.5, n = 0.013 and $S_o = 0.0005$, the bottom of the channel rises by 0.3 m and the cross-section changes to a rectangle that contains 3 gates each 1 m wide as shown in the sketch below. The two outside gates are set 0.5 m above the channel bottom and the center gate is set 0.7 m above the channel bottom. All three gates have contraction coefficients equal to $C_c = 0.6$, and a pier 0.5 m wide exists between the gates to support them. If the flow rate in the upstream channel is $Q_1 = 10\,m^3/s$, write the system of equations, and then solve them, that give the upstream depth, Y_1, and the flow rates past each of the gates.

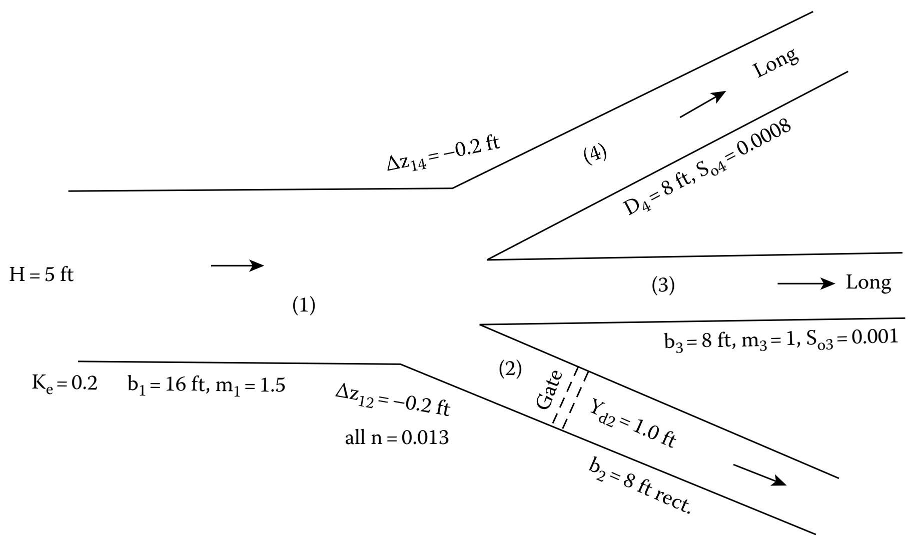


2.110 Write a computer program that is capable of completely solving the critical flow equation for a trapezoidal channel and that uses the dimensionless form of this equation in the computations. By completely solving the equation is meant that any of the variables Y_c, E_c, Q_c, b, or m may be unknown with the other variables known. Also accomplish this with a software package such as TK-Solver or Mathcad.

2.111 Write a computer program that is capable of completely solving the critical flow equation for a circular channel and that uses the dimensionless form of this equation in the computations. By completely solving the equation is meant that nay of the variables Y_c, E_c, Q_c, or D may be unknown with the other variables known. Also accomplish this with a software package such as TK-Solver or Mathcad.

2.112 Write a computer program that provides a table of values relating the depth upstream from a sluice gate to the depth downstream therefrom. Write this program so it only needs to evaluate explicit equations. Execute the program for a flow rate $Q = 40\,m^3/s$ in a 4 m wide channel with the downstream depth varying from $Y_2 = 0.2$ m to $Y_2 = 1.8$ m in increments of 0.05 m and plot the results. Note if this were done for several flow rates, q, the graph could be used to solve the specific energy across the gate. Verify the results from your program with CHANNEL. Also use a spreadsheet, TK-Solver, Mathcad or similar software to solve the problem, and plot the results.

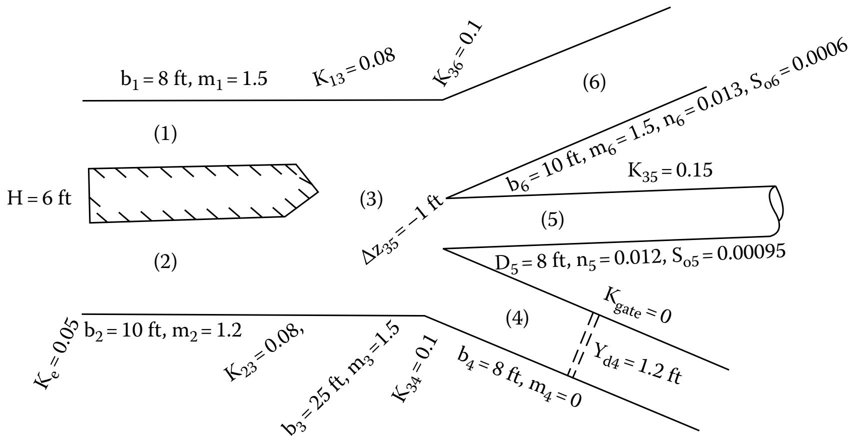
2.113 An upstream channel receives water from a reservoir with a head of H = 5 ft, and shortly downstream therefrom divides into three branch channel, with the following sizes: Channel (2) is rectangular with a bottom width of 8 ft, and it contains a gate a short distance downstream from its beginning that cause the depth of flow downstream therefrom to be Y_{d2} = 1.0 ft; Channel (3) is trapezoidal with b_3 = 8 ft, a side slope m_3 = 1, and a bottom slope of S_{o3} = 0.001; Channel (4) is circular with a diameter D_4 = 8 ft, and has a bottom slope S_{o4} = 0.0008. The bottom drops by Δz_{12} = 0.2 ft between channels (1) and (2) and also by Δz_{14} = −0.2 ft between channels (1) and (4). All Manning's roughness coefficients are n = 0.013, and the entrance loss to the reservoir is K_e = 0.2. If the upstream main channel has a side slope m_1 = 1.5, and a bottom width b_1 = 16 ft, what are the flow rates in the four channels and the depths? If the width of the upstream channel is reduced to b_1 = 14 ft what are the flow rates and the depths? What is the critical flow rate, i.e., the maximum flow rate, for channel (1)? What will occur if the width of channel (1) is reduced to 14 ft?



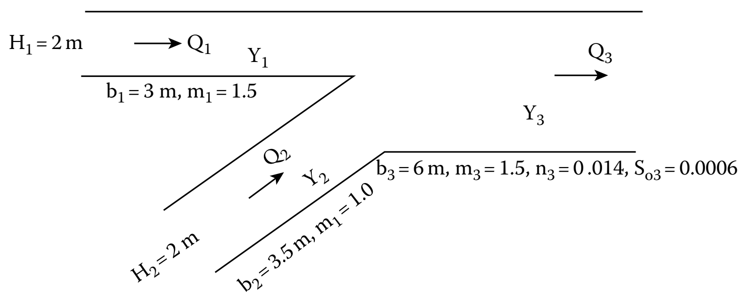
2.114 The plan view of the inlet portion of a channel system is shown in the sketch below. It consist of two parallel channels (1) and (2) that receive water from a reservoir with a water surface elevation H = 6 ft above the bottom of these channels. Channels (1) has a bottom width b_1 = 8 ft, and a side slope m_1 = 1.5, and channel (2) has b_2 = 10 ft, and m_2 = 1.2. A short distance downstream these two channels join to form channel (3), but shortly thereafter branch into three separate channels denoted by (4), (5), and (6) on the sketch. A gate controls the flow in channel (4) and is set so it produces a depth Y_{d4} = 1.2 ft immediately downstream from the gate. Channel (4) is rectangular with b_4 = 8 ft. Channel (5) is a pipe with a diameter D_5 = 8 ft, and its bottom is 1 ft below the other channels. Channel (5) has a bottom slope S_{o5} = 0.00095, and a Manning's n_5 = 0.012. Channel (6) has b_6 = 10 ft, m_6 = 1.5, S_{o6} = 0.0006, and n_6 = 0.013. The minor loss coefficients are K_e = 0.05 (entrance from reservoirs to both channels (1) and (2)), K_{13} = K_{23} =0.08, K_{34} = K_{36} = 0.1, and K_{35} = 0.15, in which the double subscripts denote from which channel to what channel the flow occurs.

Do the following: (a) Determine the variables that are unknown and list them, (b) Write out the system of equations that will provide the solution to these unknowns, (c) Solve this system of equations.

If instead of dropping 1 ft to the bottom of channel (5) its bottom were 1 ft above the other channels what would the flow rates and depths in the six channels be?

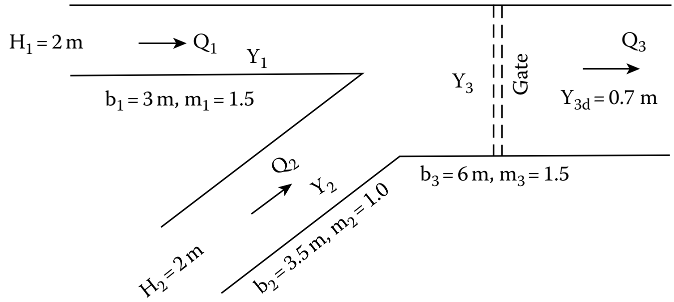


2.115 The sketch below shows two channels that take water from separate reservoirs, and combine this flow into a single channel a short distance downstream. Using the sizes of trapezoidal channels shown on the sketch below solve for the flow rates and depth in the three channels. (The loss coefficient at both entrances, as well as between the channels is K = 0.05.)

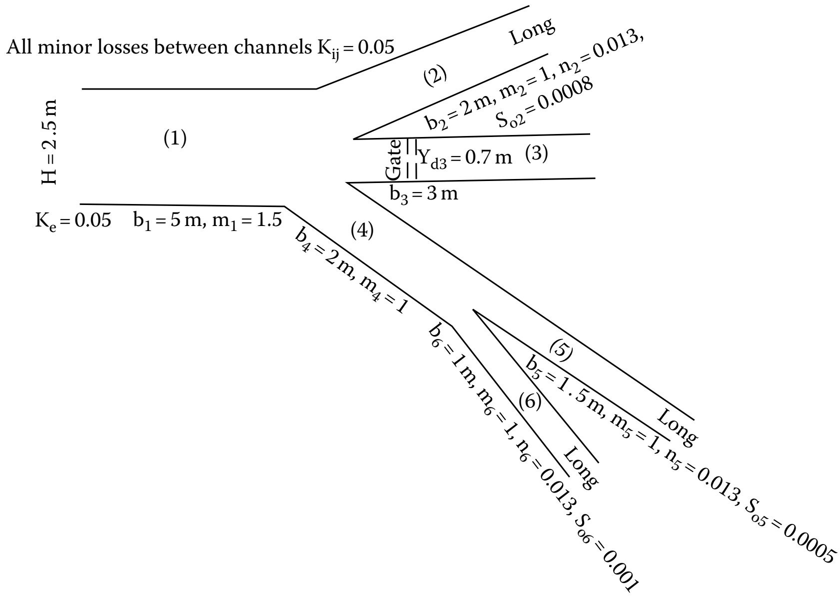


2.116 The same channel system as in the previous problem, except that the depth of the water surface in reservoir 2 is H_2 = 2.04 m, or 0.04 m above that in reservoir 1. How much has this effected the flow. Why do you think the flow from the reservoir into channel 2 approaches critical conditions so rapidly as H_2 increases? What is the limiting value of H_2 for critical conditions to occur. What would happen if this water surface rose to 2.1 m?

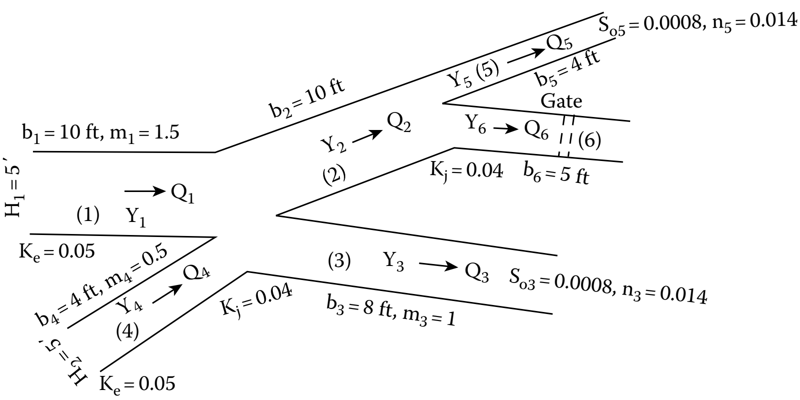
2.117 The same channel system as in Problem 2.115, except that a gate is placed in channel 3 that causes a depth of Y_{3d} downstream from it equal to 0.7 m, solve for the depths and flow rates in all channels.



2.118 A system of six channels are involved in branches as shown in the sketch below. Assume all minor loss coefficients between an upstream channel and its branch downstream, as well as the entrance loss coefficient equal 0.05, i.e., $K_e = K_{ij} = 0.05$. Channel (3), which contains a gate that produces a depth of $Y_{d3} = 0.7$ m downstream therefrom, is rectangular with a bottom width of $b_3 = 3$ m. The sizes and properties of the other channels, which are trapezoidal, are: $b_1 = 5$ m, $m_1 = 1.5$, $b_2 = 2$ m, $m_2 = 1$, $n_2 = 0.013$, $S_{o2} = 0.0008$, $b_4 = 2$ m, $m_4 = 1$, $b_5 = 1.5$ m, $m_5 = 1$, $n_5 = 0.013$, $S_{o5} = 0.0005$, and $b_6 = 1$ m, $m_6 = 1$, $n_6 = 0.013$, $S_{o6} = 0.001$. The bottoms of all channels are at the same elevation at the junctions. Write out the system of equations whose solution will provide the depths and flow rates in all six channels, and obtain a solution to these equations.

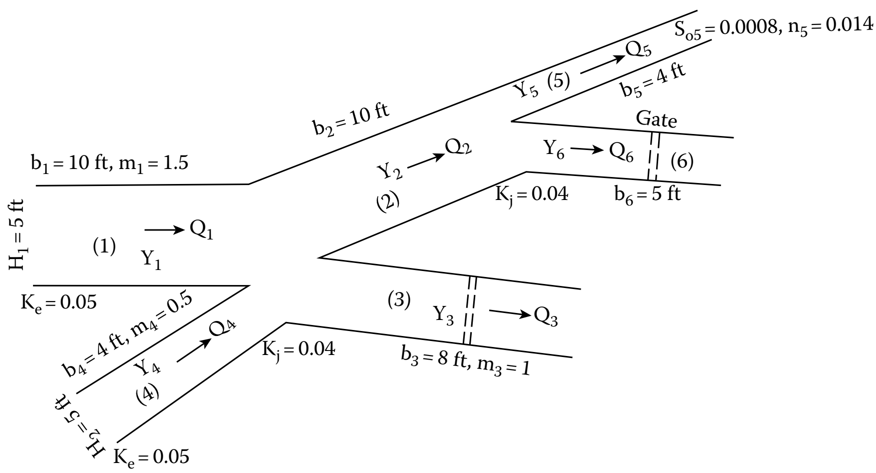


2.119 Solve for the flow rates and depths in the various channels shown in the sketch below if the head of water from both of the supply reservoirs equals 5 ft, i.e., $H_1 = 5$ ft and $H_2 = 5$ ft. The flow in channel 6 is controlled by a vertical gate, so that the depth downstream from it is $Y_{d6} = 0.6$ ft, and Channels 3 and 4 are very long both with bottom slopes of $S_{o3} = S_{o4} = 0.0008$, and Manning's roughness coefficients $n_3 = n_4 = 0.014$ so that uniform flow exist in these channels. The entrance loss coefficients are $K_e = 0.05$, and the loss coefficients from each upstream to downstream channel is $K_j = 0.04$. There is no loss across the gate.

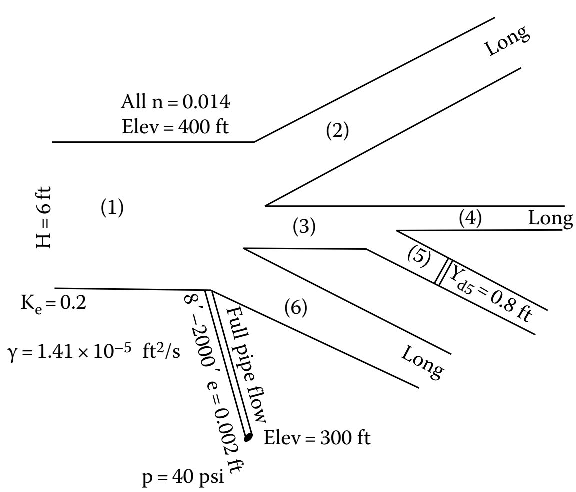


2.120 In the previous problem solve for the flow rates and depths if the height of the water surface of the reservoir that supplies channel 4 increases a small amount to $H_2 = 5.05$ ft. Solve for the flow rates and depths if this reservoir water surface fall a small amount to 4.98 ft. How do you explain why the depth in channel 4 decreases as the reservoir water surface elevation rises, and vice versus? What is the minimum H_2 possible to have the flow in channel 4 not reverse itself and flow into reservoir 2? What is the maximum head H_2 of reservoir 2 to have the condition in the other channels as specified in the previous problem? What is the minimum H_2 possible to have the other conditions as specified in the previous problem with the flow reversing itself and flowing into reservoir 2 from channel 4?

2.121 The same channel system as in Problem 2.119 except that a gate is placed in channel 3 that causes a depth of Y_{3d} downstream from it equal to 0.7 m. Solve for the depths and flow rates in all channels.



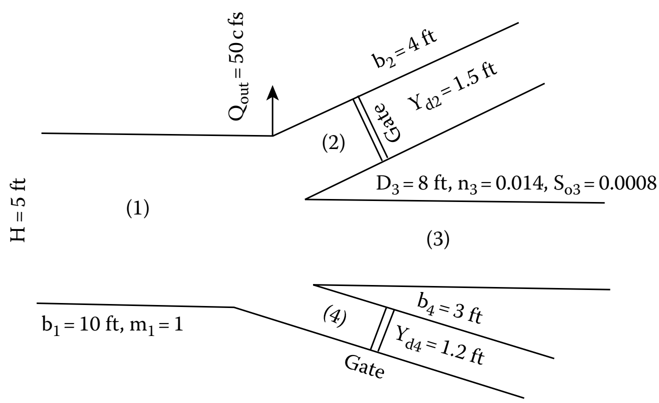
2.122 Water is taken from a reservoir whose water surface is 6 ft above a large channel bottom that branches into several channels as shown in the plan view of the system sketched below. The properties of the channels are given in the table below. The distances between branches are small enough that losses can be ignored, except that the entrance loss from the reservoir which has a loss coefficient $K_e = 0.2$. A gate exists downstream in channel 5 that produces a depth of $Y_d = 0.8$ ft downstream from it. An 8 in. diameter pipe also takes water from channel 1 near where it branches and delivers the water at 40 psi at its end where the elevation is 300 ft. The pipe is 2000 ft long and has an equivalent sand roughness for use in the Darcy–Weisbach, Colebrook–White equations of e = 0.002 in. The elevation of the bottoms of all the channels where they branch is 400 ft.



Channel	b (ft)	m	S_o
1	20	1.8	0.001
2	6	1.0	0.001
3	8	0.0	0.0005
4	3	0.0	0.0005
5	3	0.0	0.0007
6	5	1.0	0.0007

2.123 In the previous six channel pipe system, it is desired that channel 4 delivers 100 cfs. What size should it be? In addition the pipe size is to be increased to 2 ft diameter (same e, etc.) and the gate in channel 5 is raised so the depth downstream from it is 1.1 ft. Now what are the flow rates and depths in the channels (and the width need for channel 4)?

2.124 A trapezoidal channel with $b_1 = 10$ ft and $m_1 = 1$ receives water from a reservoirs whose water surface is $H = 5$ ft above the channel bottom. A short distance downstream from the entrance the channel branches into channels 2, 3, and 4. Channels 2 and 4 have gates a short distance downstream that cause downstream depths of $Y_{d2} = 1.5$ ft and $Y_{d4} = 1.2$ ft, respectively. These channels are rectangular with $b_2 = 4$ ft, and $b_4 = 3$ ft, respectively. Channel 3 consists of a pipe with a diameter $D_3 = 8$ ft, a Manning's $n_3 = 0.014$, and a bottom slope $S_{o3} = 0.0008$. There is a diversion at the junction of the channels of $Q_{out} = 50$ cfs. At the branch all channel have the same elevation. Write out the equations whose solution will provide the flow rates and depths in all channels.



2.125 For the branched channel system of the previous problem make up a table that gives the discharges and depths in the four channels as a function of the gate setting Y_{G2} in channel 2. The contraction coefficient for this gate is $C_c = 0.6$, and the channel downstream from the gate is rectangular with $b_2 = 4$ ft, $n_2 = 0.014$, and $S_{o2} = 0.0009$.

2.126 For the branched channel system of Problem 2.124 make up a table that gives the discharges and depths in the four channels as a function of the gate setting Y_{G4} in channel 4. The contraction coefficient for this gate is $C_c = 0.6$, and the channel downstream from the gate is rectangular with $b_4 = 3$ ft, $n_4 = 0.014$, and $S_{o4} = 0.0012$.

2.127 Repeat Problem 2.124 with all four channel with the same sizes and gate settings, etc. but the bottom of channel 3 (the pipe) at its beginning is 1 ft below the bottom of the other channels at the junction position, i.e., $\Delta z_{13} = -1.0$ ft.

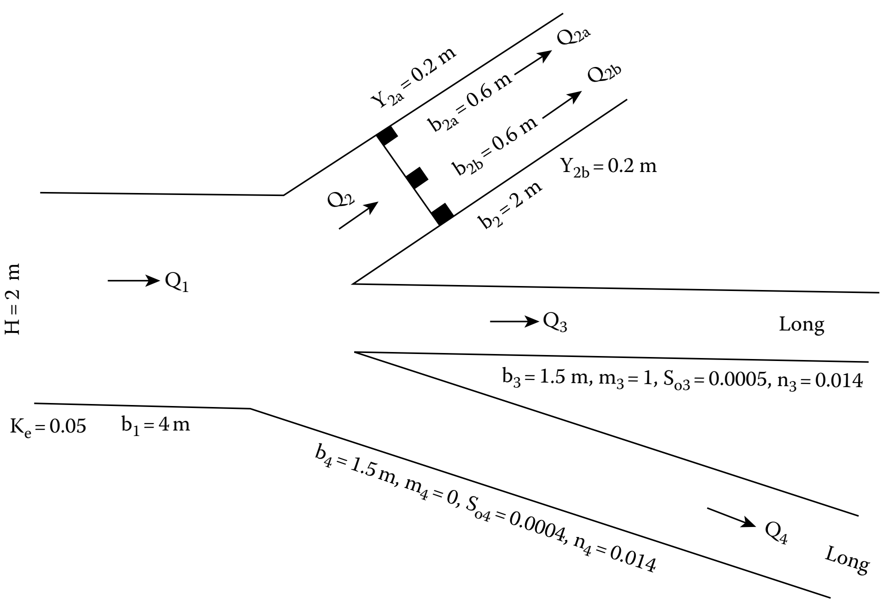
2.128 In the branched channel system of the previous problems, what is the maximum flow rate that can be obtained through channel 3 if its diameter were to be increased? What are the flow rates in the other channels when this condition occurs and the gates are both completely open? What is the minimum diameter that the pipe can have for this maximum flow rate to occur? Solve the flow rates and depths for both $\Delta z_{13} = 0$ and $\Delta z_{13} = -1.0$ ft.

2.129 Modify the model used to obtain the dimensionless graph of Example Problem 2.23 so that it can solve the specific energy between section 1 and 2 (with the filled bottom) of a pipe for other variables without the flow being critical as section 2.

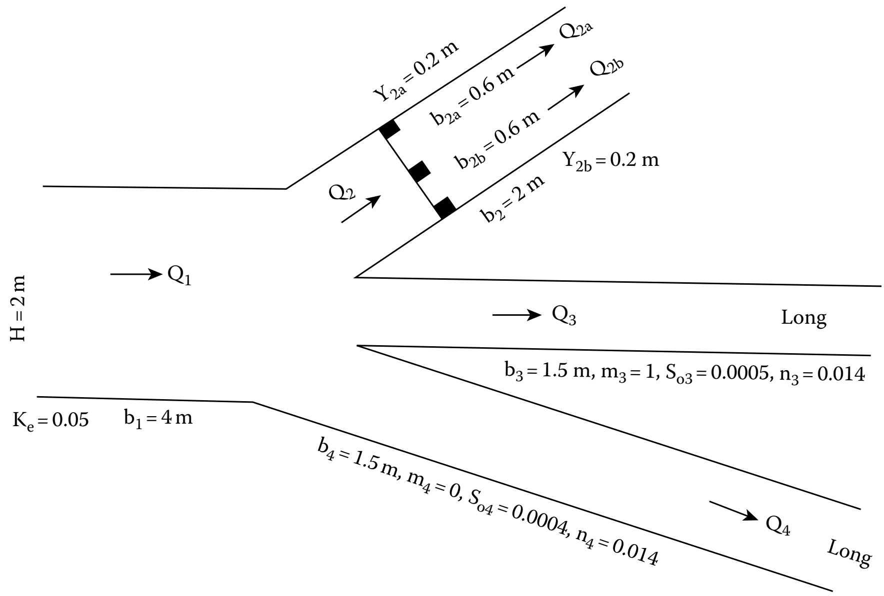
2.130 Develop a model that is capable of solving problems involving the specific energy between section 1 and 2 in a pipe such as in Example Problem 2.23 in which the bottom of the pipe is filled to a depth Δz, but do this so that the actual, rather than the dimensionless, variables are solved.

2.131 Make a graph similar to that given in Example Problem 2.23 except relate the dimensionless flow rate to the Froude number.

2.132 A complex branching channel system takes water from a reservoir with a water surface elevation of H = 2 m above the channel bottom as shown. The sizes of the channels are shown on the sketch below. All channels are rectangular except number 3, which is trapezoidal. Channel 2 contains two gates each 0.6 m wide that both have contraction coefficients $C_c = 0.6$. The first gate is set so its tip is 0.2 m above the channel bottom and the other gate is set 0.15 m above the channel bottom. Channels 3 and 4 are long with Manning's roughness coefficients, and bottom slope as shown on the sketch. Identify what unknowns you would solve for and then give the equations that need to be solved. Then solve these equations.



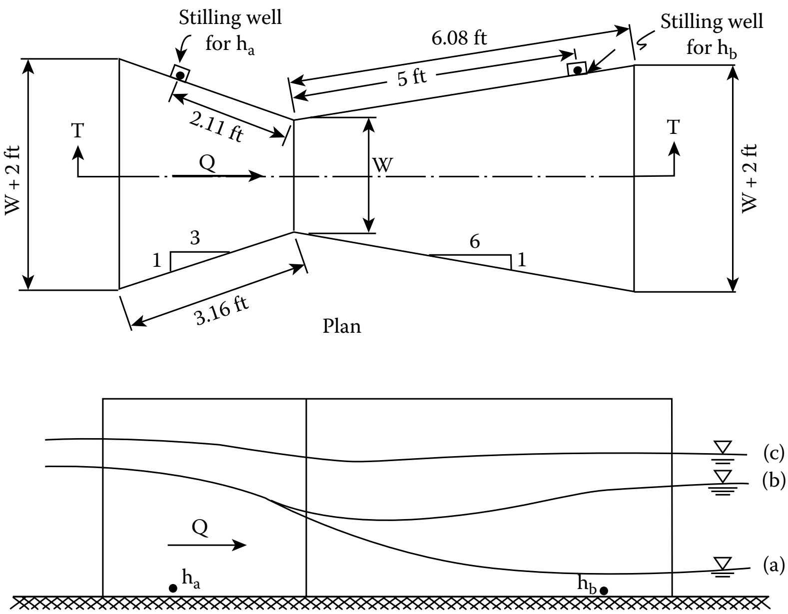
2.133 The same channel exists as in the previous problem except now channel 4 contains a pier over a length of it that divides the flow into two parts Q_5 and Q_6. Set up and solve the equations that now govern the problem. (Note: Under the assumptions that we do not need to solve the gradually varied flow in the channel upstream and near the junctions, and that the pier's length is short so that the energy at it upstream and downstream ends are the same we end up with one less equation than unknowns unless some condition is established across the pier. Since under our assumptions one would expect the flow in the two channels through the pier length to be close to the ratio of the channel widths, this might be used as the condition.)



2.134 Parshall and Cutthroat flumes that are widely used to measure flow rates in open channels and ditch make use of the energy principle, and if free flow conditions exist critical depth occurs close to the throat. A Cutthroat flume always has its throat width, W, 2 ft less than its upstream and downstream width, and its bottom is flat. The upstream stilling well is 2.11 ft upstream from the throat whereas the upstream convergence sides are 3.16 ft long. (The length of the flume downstream from the throat is 3 ft, and flume beginning is 2 ft upstream from the throat, so the diverging portion is 1 on 6 and the upstream converging portion of the flume converges 1 on 3.) The laboratory calibration equation for Cutthroat flumes is Q (cfs) = $Ch_a^{1.56}$ (h_a in ft) in which $C = 3.50\,W^{1.025}$ (with the width W in ft). Use this free flow equation for a Cutthroat flume to locate the position where critical flow will occur in a 6 ft wide Cutthroat flume for a range of flow rates. Use one dimensional hydraulics in this determination, and assume there is no energy loss between the upstream stilling well position where depth h_a is measured and the critical flow section.

2.135 Repeat the previous problem except for a 2 ft wide Cutthroat flume.

2.136 A 6 ft wide Cutthroat flume is installed in a trapezoidal channel with a bottom width of 12 ft, and a side slope of m = 2. The slope of the downstream channel bottom is $S_o = 0.00287$. When the flume was first installed the downstream channel consisted of a large sand grain materials that had a Manning's roughness coefficient n = 0.018. However because this material was transported downstream the downstream channel was lined to a depth of 6 in. around it entire cross section with riprap rock with a Manning's n = 0.035. Assuming the downstream channel is very long so it will flow at uniform depth, and that the local loss from the downstream stilling well into the wider trapezoidal channel equals the difference in velocity heads between these two section, or that the normal depth in the downstream channel equals the depth in stilling well h_b determine the effect of the riprap in changing the flow from free flow to submerged flow through the Cutthroat flume. The transition submergence ratio, $S_t = h_b/h_a$ equals 0.88 for a 6 ft wide Cutthroat flume.



2.137 Example Problem 2.23 examined how the ratio of step height in the bottom of a circular channel divided by the upstream depth that produced critical flow varied as a function of the Froude number. Derive the dimensionless equations, whose solution provided this relationship.

2.138 A circular channel with a diameter D = 3 m, with a Manning's n = 0.014 is laid on a slope of $S_o = 0.0008$. If the flow rate being carried by this channel is $Q = 10\,m^3/s$, what is the maximum height of hump that can fill the bottom of the channel (but the diameter at this section is still the same, i.e., $D_2 = 3\,m$) if the upstream depth is not to be effected. How does this height compare with a reduction in radius that produces critical flow?

2.139 A flow rate Q = 400 cfs is taking place in a circular channel with a diameter D = 15 ft. The bottom slope of the channel is $S_o = 0.0005$, and its Manning's roughness is n = 0.013. Concrete has been placed in the bottom of this channel to a depth of 1.5 ft at a section, so that its bottom is flat here. The filled in portion of the channel is 1200 ft long and then it discharges into a reservoir whose water surface is 7.0 ft above the channel bottom. Determine the depths immediately upstream and downstream from where the filled in bottom occurs, and the GVF profiles upstream from this position and to the downstream reservoir.

2.140 Write a program, or a computer model, that will generate tables of values to plot the graphs in Figure 2.13, i.e., make tables for different side slopes m_1 that solve Equation 2.41 (or Equation 2.40). It will be instructive for you to use the Laguerre method described in Chapter 3 and extract all roots from this fifth degree polynomial, at least for smaller values of m_1.

2.141 An upstream trapezoidal channel has a bottom width $b_1 = 3\,m$, and a side slope $m_1 = 0.75$, and transitions to a rectangular channel with $b_2 = 3.5\,m$. What flow rate will result in critical depths in both the trapezoidal and rectangular channels? What are these depths and their corresponding specific energies? (Solve the problem using dimensionless variables.)

2.142 An upstream trapezoidal channel with $m_1 = 1.5$ and $b_1 = 5\,m$ smoothly changes to a rectangular channel with $b_2 = 5\,m$ wide, and critical flow occurs here. If the flow rate is $Q = 12\,m^3/s$, what are the depths upstream and downstream from the transition? What is the upstream

Froude number? If $n = 0.014$, what bottom slope of the upstream channel will result in uniform flow? (Solve this problem using both dimensionless and dimensioned variables.)

2.143 An upstream trapezoidal channel has $m_1 = 1.5$, $n = 0.013$, and $S_{o1} = 0.00025$, and a bottom width ratio to a downstream rectangular channel of $b_1' = b_1/b_2 = 1.5$. For a flow rate $Q = 500$ cfs, solve for b_1 and b_2 so critical depth occurs in the rectangular channel, and uniform flow occurs upstream.

2.144 Write a program, or a computer model, that will generate a table of values to plot Figure 2.14, i.e., make a table that solves Equation 2.44 (or Equation 2.43). It will be instructive for you to use the Laguerre method described in Chapter 3 and extract the three roots from this third degree polynomial.

2.145 A flow rate $Q = 11.5\,m^3/s$ occurs in a rectangular channel that reduces its width by 2/3 through a transition. The upstream channel width is $b_1 = 5\,m$. Determine the upstream depth and Froude number. If $n = 0.014$ what bottom slope of upstream channel will result in uniform flow?

2.146 What width b_2 of a steep downstream rectangular channel will result in uniform flow upstream if the upstream rectangular channel's width is $b_1 = 12\,ft$, $n = 0.015$ and $S_{o1} = 0.0005$ for a flow rate $Q = 350$ cfs.

2.147 Determine the value of c in the equation $Y_1' = c/b'$ that defines the upstream dimensionless depth $Y_1' = Y_1/b_1$ as a function of the width ratio $b' = b_1/Y_{c2}$ for a transition from an upstream rectangular channel with $b_1 = 6\,ft$ and a downstream rectangular channel with $b_2 = 4.8\,ft$. What is the upstream Froude number?

2.148 What ratio of upstream to downstream widths b_1/b_2 of rectangular channels should be used if the upstream width b_1 should equal twice the critical depth in the downstream channel, and the depth Y_1 is to be 10% larger than the critical depth. For a flow rate $Q = 20\,m^3/s$ and the upstream width $b_1 = 5\,m$, what are these depths? What is the upstream Froude number?

2.149 Write a computer program, or develop a computer model, that solves Equation 2.43 for any of the variables: b_1', b', or Y_1' given the other two and use this model to solve Problems 2.145 and 2.148.

2.150 The eight graphs in Figure 2.13 use linear graph paper to display the relationship of the dimensionless variables, Y_1', b_1', and b' for different values of m_1. Plot these relationships on eight log-log graphs and obtain an approximate equation of the form $Y_1' = a(b')^b$ for each b_1' and m_1. From these equations can you suggest one equation that may be used to provide a "guess" to start the Newton solution to Equation 2.40.

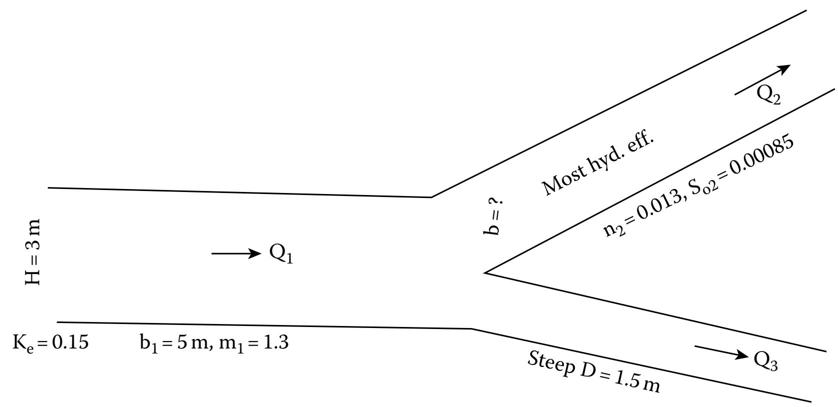
2.151 Write a computer program, or develop a computer model, that solves Equation 2.40 (or Equation 2.41) for any of the variables: b_1', b', or Y_1', and use this program (model) to solve Problem 2.142.

2.152 An upstream trapezoidal channel with a bottom slope of $S_{o1} = 0.0002$, a side slope $m_1 = 1$, and $n = 0.013$ has a smooth transition to a steep downstream circular channel with $D = 10\,ft$, (a) If the upstream channel has a width $b_1 = 10\,ft$, what flow rate will result in uniform flow upstream and what will this uniform depth be, and what will the critical depth be at the beginning of the circular channel? Repeat this solution for larger downstream pipe diameters up to 12 ft, and examine how the flow rates and depths change. (b) Determine how the upstream bottom width must vary for uniform flow to occur in the upstream channel for specified flow rates varying from 400 to 800 cfs in increments of 50 cfs. (Also determine the upstream uniform depth Y_1, and the critical depth Y_{c2} associated with each of these flow rates.) (c) Determine how the upstream bottom slope S_{o1} necessary for uniform flow varies as the flow rates vary from 400 to 800 cfs.

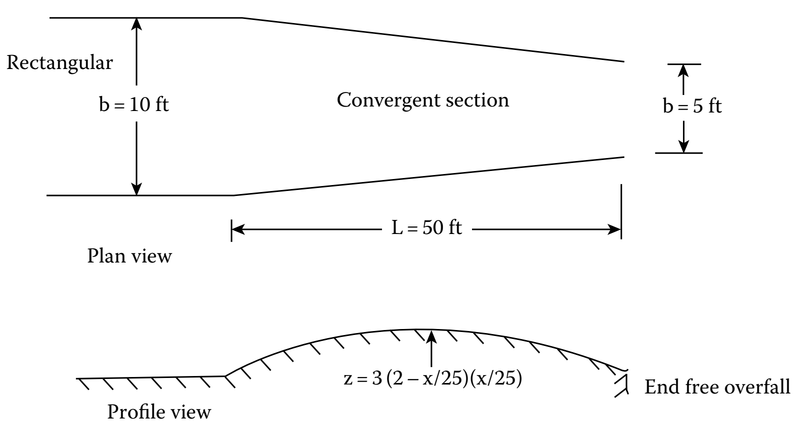
2.153 It has been determined that the side slope of a trapezoidal channel must be $m = 1.5$. What should the bottom width, b, be to have the hydraulically most efficient section, if the channel is to be designed for a flow rate $Q = 500$ cfs, its Manning's roughness coefficient is $n = 0.015$, and it has a bottom slope $S_o = 001$?

2.154 The most efficient trapezoidal sections is to be used to convey 500 cfs of water from a reservoir whose head H = 5 ft above the channel bottom. The channel will have a bottom slope $S_o = 0.0008$ and its Manning's n = 0.013. Find the bottom width of channel to use. The entrance loss coefficient is $K_e = 0.15$.

2.155 Determine the hydraulically most efficient trapezoidal channel # 2 in the three channel system shown below. This # 2 channel has a Manning's n = 0.013 and a bottom slope of $S_{o2} = 0.00085$. Channel # 1 is trapezoidal also with a bottom width $b_1 = 5$ m and a side slope $m_1 = 1.3$, and receives its water from a reservoir with a head H = 3 m. The entrance loss coefficient is $K_e = 0.15$. Channel # 3 is a pipe with a diameter D = 1.5 m and is steep. Its bottom is 1.2 m above the level of the other two channels.



2.156 If the elevation of the bottom of a channel changes simultaneously with a contraction of its size, critical flow may not occur at the channel's throat, or at its end, if it ends in a free overfall. The position where this control (or critical depth) occurs will be where the sum of the critical specific energy E_c, and the elevation of the bottom z is a maximum, or where $H_c = E_c + z$ is maximum. A rectangular channel that is b = 10 ft wide and is carrying a flow rate Q = 250 cfs end in a free overfall. However, before the termination of the channel the width of the channel reduces to 5 ft at its end over a 50 ft length, while simultaneously the bottom of the channel rises and then fall again according to the second degree polynomial $z = 3(2 - x/25)(x/25)$ as shown in the sketch below. Find the position x_c where critical depth will occur and this critical depth. To help understand why critical depth occurs where H is maximum, make up a table that gives z, E, Y, and F_r^2 as a function of the position x.



2.157 For the channels of the previous problem investigate how the position, and magnitude of the critical depth varies with the flow rate, using values of Q from 50 to 400 cfs.

2.158 Rather than have the bottom of the channels in the previous two problems rise according to a second degree polynomial the bottom rises 3 ft linearly over the first 10 ft of the transition, and then falls 3 ft over the last 40 ft of the transition. Now where does critical depth occur?

2.159 A trapezoidal channel, which is b = 5 m wide with a side slope m = 1, except near its end carries a flow rate Q = 25 m^3/s. Before ending in a free overfall at its end the channel has a 25 m long transition in which the bottom width reduces to 2.5 m, and the side slope m to 0, both linearly. Simultaneously the bottom of the channel rises according to the second degree polynomial, $z = (x/12.5)(2 - x/12.5)$. Find the position where critical depth occurs, and then make up a table as in Problem 2.156 that gives the Froude number as a function of x.

2.160 Figure A.2 provides a plot of several dimensionless variables as functions of the dimensionless depth $Y' = Y/D$ for circular channels. Prove that the maximum flow rate Q will occur in a given circular channel (D, S_o and n fixed) according to Manning's equation when $\beta = 2.6391$ rad (a corresponding dimensionless depth $Y' = Y/D = 0.93818$. Also prove that the dimensionless hydraulic radius $R_h' = R_h/D^2$ occurs when $\beta = 2.2468$ rad, and the maximum dimensionless conveyance $K' = A' = (A'/P') = 0.5015$, i.e., verify some of the values given on Figure A.2. What value of A′ is associated with the maximum Q? Give a physical explanation why the maximum Q is associated with a larger β than the maximum hydraulic radius.

2.161 As in the previous problem determine the angle β associated with the maximum flow rate Q that will occur in a circular channel based on Chezy's Equation, rather than Manning's equation, assuming that Chezy's C is constant. Compute A′, A_d, R_h', and K′ associated with this condition. Give a physical explanation why this β is larger than the β of the previous problem.

2.162 In the previous problem you were to determine angle β associated with the maximum Q in a circular channel based on Chezy's C being constant. Determine this angle for several values of the relative roughness e/R_h if the flow is wholly rough.

REFERENCES

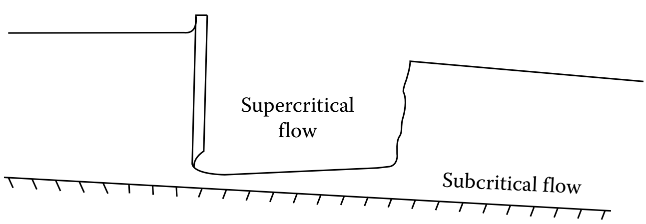
ASCE Task Force. 1963. Friction factors in open channels. *Journal of the Hydraulic Division, ASCE*, 89(2), 97–143.

Jarrett, R. D. 1991. Hydrologic and hydraulic research in mountain rivers, *Water Resources Bulletin*, American Water Resources Association, 26(3), 419–429.

3 The Momentum Principle Applied to Open Channel Flows

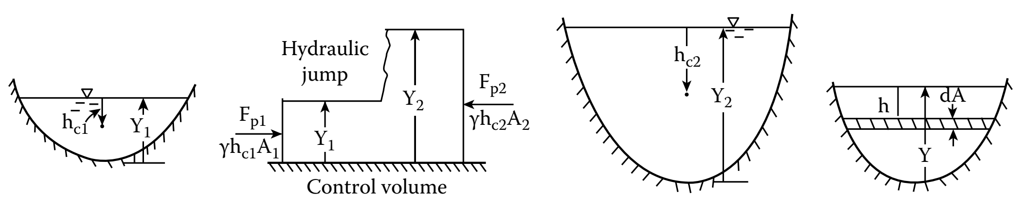
3.1 THE MOMENTUM FUNCTION

Use of the momentum principle is needed when forces control the direction or conditions associated with fluid motions, or when it is not possible to define what is happening to the fluid on a small element basis but a large picture of a mass of fluid within a control volume is possible. If a vector quantity, such as force or velocity with both magnitude and direction, is the unknown, or is one of the important known variables of the problem, then it will most likely be necessary to use the momentum principle in solving the problem. In order to introduce and develop the momentum function for use in open channel hydraulics, an interesting phenomena, the hydraulic jump will be analyzed. A hydraulic jump abruptly takes the flow in an open channel from supercritical flow to subcritical flow, so that through a hydraulic jump the depth of flow rather abruptly increases. Downstream from a hydraulic jump there will be a control, which may be a flatter channel, a gate, or dam, etc. that requires the flow to be at a subcritical depth. Upstream from the hydraulic jump something will cause the flow to be supercritical, such as a gate, or steep channel. The supercritical flow rushes down the channel with a velocity in excess of the speed of small amplitude gravity waves, and consequently receives no signal from the downstream flow. Since it must change to subcritical flow at some position because a downstream control dictates this, the change occurs in the form of a hydraulic jump. The sketch below illustrates these conditions resulting in a hydraulic jump. A hydraulic jump actually takes place over a finite length of several feet, but since sketches herein have an enlarged vertical to the horizontal scale the hydraulic jump is shown as a near vertical line.



To apply the momentum principle to the hydraulic jump, the steps outlined in Chapter 1 will be followed:

Step # 1: A control volume of the hydraulic jump will be created by moving a short distance upstream and downstream from it and removing the fluids.



Step # 2: These columns of removed fluid will be replaced by the hydrostatic forces that they apply to the control volume fluid. A hydrostatic pressure force equals the pressure at the centroid of the area times the area, and the pressure at the centroid equals the specific weight γ times the distance from the water surface to the centroid, h_c. Therefore, on the upstream side this force is, $F_{p1} = \gamma h_{c1} A_1$ and on the downstream side this force is, $F_{p2} = \gamma h_{c2} A_2$, in which the subscripts denote upstream and downstream respectively. The quantity $h_c A$ is also the first moment of area about the fluid surface. Methods for its evaluation for common cross sections are given in Table A.1, but in general it is given by

$$h_c A = \int h dA \tag{3.1}$$

in which h is the distance from the fluid surface down to the differential element dA of the cross-sectional area. Only momentum fluxes in the direction of flow will be considered. Neither the normal nor shear force from the bottom of the channel will be included. Nor will the weight of the fluid in the control volume. In steep channels, the component of the fluid weight in the direction of flow will be a significant force and possibly also the shear force. However, since the weight will depend upon the size of the control volume, its inclusion into the analysis must be based on experimental data. Our intent here is to develop the momentum principle for use in open channel flow analysis, and inclusion of the component of weight, and shear forces in the flow direction complicate the problem so that basic principles cannot be developed. Should a hydraulic jump occur in a steep channel, then the results that follow will need modification.

Step # 3: The application of the momentum equation in the direction of flow results in

$$\gamma h_{c1} A_1 - \gamma h_{c2} A_2 = \rho Q(V_2 - V_1)$$

Note that every term in the above equation has dimensions of force, since γ is dimensionally force per length cubed, or (F/L^3), and this is multiplied by (L^3) in $h_c A$, and the momentum flux terms have dimensions of $(Ft^2/L^4)(L^3/t)(L/t) = (F)$. Since $\rho = \gamma/g$, the above equation can be divided by the specific weight, which will result in each term having the dimensions of L^3. Upon rearranging terms with the same subscript on the same side of the equal sign the following equation results:

$$h_{c1} A_1 + \frac{Q^2}{(gA_1)} = h_{c2} A_2 + \frac{Q^2}{(gA_2)} \tag{3.2}$$

The sum of the two term $Ah_c + Q_2/(gA)$ is called the momentum function, and will be given the symbol M. Thus an alternative way of expressing Equation 3.2 is that the momentum function on the two ends of the hydraulic jump are equal, or

$$M_1 = M_2 \tag{3.3}$$

in which the M's are defined by

$$M = Ah_c + \frac{Q^2}{(gA)} = Ah_c + \frac{V^2 A}{g} \tag{3.4}$$

Table A.1 gives the result from integrating $\int hdA$ for trapezoidal and circular shapes. For a trapezoidal channel,

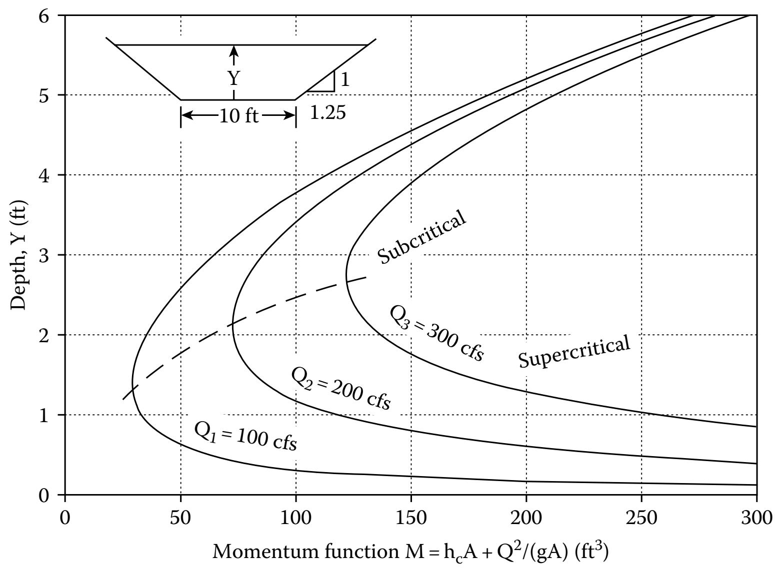
$$Ah_c = \frac{bY^2}{2} + \frac{mY^3}{3} \quad \text{(for a trapezoid)}$$

and for a circular channel one form in which the results can be given is

$$Ah_c = \frac{D}{2}\left\{\frac{D^2}{6}\sin^2\beta - A\cos\beta\right\} = \frac{D^3}{24}(3\sin\beta - 3\beta\cos\beta - \sin^3\beta) \quad \text{(for a circle)}$$

3.2 CHARACTERISTICS OF THE MOMENTUM FUNCTION

A plot of $M = Ah_c + Q^2/(gA)$ as the abscissa, and the depth Y as the ordinate is called a momentum diagram (or some books may refer to it as a thrust diagram) just as E versus Y is referred to as a specific energy diagram. A sketch of a momentum diagram is shown below. From an examination of the two terms that sum to give M, it is clear that if a fixed channel is to convey a specified flow rate Q, then as the velocity in that channel approaches zero the term Ah_c becomes very large, and the term $Q^2/(gA)$ becomes very small since A becomes large; thus M becomes large. On the other hand, as the depth approaches zero (the velocity becomes large), the area must become very small and the term $Q^2/(gA)$ becomes very large, even if Ah_c is small. Therefore, a momentum function diagram will have a minimum value for M at some intermediate depth, much as is the case with a specific energy diagram.



To find this minimum value the principle of calculus of setting the first derivative to zero can be used, that is dM/dY will be equated to zero. In order to differentiate Ah_c it is necessary to utilize Leibniz's rule since Ah_c is defined by the above integral. Leibniz's rule is

$$\frac{d}{d\alpha}\int_{\phi_1(\alpha)}^{\phi_2(\alpha)} F(x,\alpha)dx = \int_{\phi_1(\alpha)}^{\phi_2(\alpha)} \frac{\partial F_1}{\partial \alpha}dx + F(\phi_1,\alpha)\frac{d\phi_1}{d\alpha} - F(\phi_2,\alpha)\frac{d\phi_2}{d\alpha}$$

Applying this rule to the derivative $d(Ah_c)/dY$ gives

$$\frac{d(Ah_c)}{d\alpha} = \int_Y^0 \frac{\partial h}{\partial Y}dA + (0)1 - Y(0) = A$$

and therefore

$$\frac{dM}{dY} = A - \frac{Q^2}{gA^2}\frac{\partial A}{\partial Y} = A - \frac{Q^2 T}{gA^2} = 0$$

or

$$\frac{Q^2 T}{gA^3} = 1 \quad \text{or} \quad F_r^2 = 1 \tag{3.5}$$

This result is also the "critical flow" equation that indicates that the average velocity of the flow equals the speed of propagation of a small amplitude gravity wave. Therefore the minimum value for the momentum function M exactly coincides with the minimum value for the specific energy for this same flow rate in the same channel. Thus a hydraulic jump will always take the flow from a supercritical condition to a subcritical condition. Furthermore, because of the shape of the momentum function the height of the hydraulic jump will be greater as the depth upstream is further below critical depth. Should the depth upstream from a jump be only slightly below Y_c then the downstream depth will only be slightly above Y_c. This is referred to as a **mild hydraulic jump**, and in observing such occurrences in a channel one would note only a waviness in the water surface. A **strong hydraulic jump** on the other hand will have a highly supercritical flow upstream (e.g., F_{r1} is much larger than unity), a slow moving flow downstream, with a large change in the water surface elevation. Such a strong hydraulic jump will occur, for example, on the apron of a dam spillway where the water after flowing down the steep spill way of the dam has a very high velocity. Strong hydraulic jumps dissipate a large fraction of the total energy per unit weight that the upstream fluid possesses.

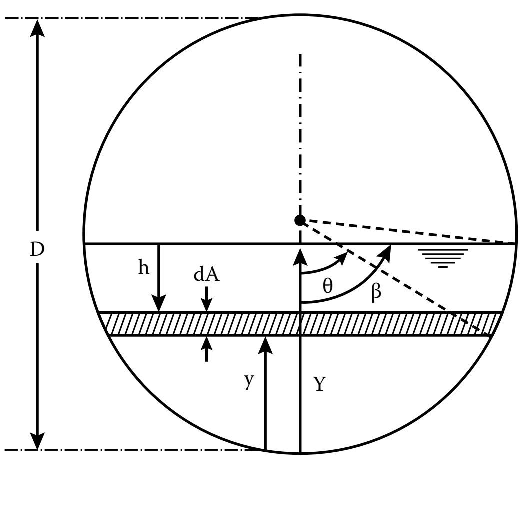
The two depths across a hydraulic jump are referred to as **conjugate depths**. Thus the momentum function M is constant for conjugate depths, one of which is associated with supercritical flow and the other is associated with subcritical flow. Conjugate depths are not equal to alternate depth for which the specific energy is constant for a given flow rate.

EXAMPLE PROBLEM 3.1

Find the first moment of area about the water surface of a flow in a 6 ft diameter pipe that is 2 ft deep. Determine this value for Ah_c by the following four procedures: (1) use your calculator to numerically integrate the appropriate function, (2) use the Simpson's rule SIMPR described in Appendix B, (3) and (4) use the formula given in Appendix A.

Solution

First solve for $\beta = \cos^{-1}(1 - Y/R) = \cos^{-1}(1 - 2/3) = 1.23096$ rad, and the area $A = R^2(\beta - \cos\beta \sin\beta) = 8.25021$ ft^2. The first moment of area is shown in the sketch below to be



$$Ah_c = \int_0^Y h dA \quad \text{with} \quad \cos\theta = 1 - \frac{y}{R}$$

$$h = Y - R(1-\cos\theta), \quad \text{and} \quad dA = 2R\sin\theta dy \quad \text{so}$$

$$dy = R\sin\theta d\theta$$

$$Ah_c = \int_0^\beta [Y - R(1-\cos\theta)][2R\sin\theta](R\sin\theta d\theta)$$

$$Ah_c = 2R^3 \int_0^{1.23096} [Y/R - 1 + \cos\theta]\sin^2 d\theta$$

$$Ah_c = 54 \int_0^{1.23096} \left(\cos\theta - \frac{1}{3}\right)\sin^2\theta d\theta$$

1. Integrating with the HP48G calculator produces $Ah_c = 6.8347\,ft^3$
2. The computer program that calls on SIMPR can consist of the following:

```
PARAMETER (NMAX=21,A=0,B=1.23096,ERR=1.E-5)
EXTERNAL EQUAT
CALL SIMPR(EQUAT,A,B,VALUE,ERR,NMAX)
WRITE(*,*) 54.*VALUE
END
FUNCTION EQUAT(T)
EQUAT=(COS(T)-.33333333)*SIN(T)**2
RETURN
END
```

   As a solution it prints out 6.83474 for the solution.
3. Using the equation $Ah_c = D^3/24\{3\sin\beta - 3\beta\cos\beta - \sin^3\beta\} = 6.8347\,ft^3$
4. Using the equation $Ah_c = 0.5D\{D^2/6\sin^3\beta - A\cos\beta\} = 6.8347\,ft^3$

EXAMPLE PROBLEM 3.2

A hydraulic jump occurs in a circular channel with a diameter D = 4 m. The flow rate Q = 22 m^3/s and the depth upstream from the jump is 0.8 m. What is the depth downstream from the hydraulic jump, and how much energy per unit weight of fluid is dissipated by the jump?

Solution

From the known conditions upstream from the jump the momentum function can be calculated here, or $M = (D/2)\{(D^2/6)\sin^3\beta - A\cos\beta\} + Q^2/(gA)$ in which $A = (D^2/4)(\beta - \cos\beta\sin\beta)$ and $\beta = \cos^{-1}(1 - 2Y/D)$. Substituting the known values into this equation produces, $M_1 = 28.16\,m^3$. Across the hydraulic jump $M_2 = M_1$, and therefore the above implicit equation (Equation 3.2) needs to be solved in which M = 28.16 and D = 4. Using the Newton method, the solution is 2.871 rad, from which $Y_2 = 3.927$ m. The Froude number associated with the upstream flow is $F_{r1} = 5.25$, and that associated with the downstream flow conditions is $F_{r2} = 0.16$. The specific energy upstream is $E_1 = 8.51$ m and that associated with the downstream conditions is $E_2 = 4.08$ m. The difference represents the head loss through the hydraulic jump, or the energy per unit weight of fluid dissipated by the jump. The head loss equals 4.43 m, or more energy is dissipated than remains in the downstream flow. The most difficult part of this solution is solving the implicit momentum equation for the depth of flow.

3.3 RECTANGULAR CHANNELS AND MOMENTUM FUNCTION PER UNIT WIDTH

Simplification occurs if the channel is rectangular in shape. For a rectangular channel it is useful to deal with the momentum function per unit width of channel. This unit momentum function will be denoted by lower case m, or m = M/b much the same as q = Q/b. For a rectangle $Ah_c = bY^2/2$, and dividing Equation 3.4 by b gives

$$m = \frac{M}{b} = \frac{Y^2}{2} + \frac{q^2}{(gY)} = \frac{Y^2}{2} + \frac{V^2 Y}{g} \tag{3.6}$$

Therefore a hydraulic jump in a rectangular channel is defined by

$$\frac{Y_1^2}{2} + \frac{q^2}{(gY_1)} = \frac{Y_2^2}{2} + q^2(gY_2) \tag{3.7}$$

It can be noted that when Equation 3.7 is multiplied by either Y_1 or by Y_2 that a cubic equation results just as a cubic equation defined the alternative depths when dealing with specific energy in a rectangular channel. To investigate further the characteristic of Equation 3.7 collect the terms with Y squared on one side of the equal sign, and the terms contain q on the other side so that Equation 3.7 becomes

$$\frac{Y_1^2}{2} - \frac{Y_2^2}{2} = \frac{(Y_1 + Y_2)(Y_1 - Y_2)}{2} = \frac{q^2}{g}\left\{\frac{1}{Y_2} - \frac{1}{Y_1}\right\} = \frac{q^2(Y_1 - Y_2)}{gY_1Y_2} \tag{3.8}$$

Upon dividing this equation by $(Y_1 - Y_2)$ and then multiplying it by Y_1 the following quadratic equation results:

$$Y_1^2 + Y_2 Y_1 - \frac{2q^2}{(gY_2)} = 0$$

It should also be noted that the above cubic equation has been reduced to a quadratic equation by the division of $(Y_1 - Y_2)$. Using the quadratic formula to solve this equation and then dividing the result by Y_2 gives the following useful equation to computer the depth upstream of a hydraulic jump if the downstream depth is known:

$$\frac{Y_1}{Y_2} = \frac{-1 + \sqrt{1 + 8q^2/(gY_2^3)}}{2} = \frac{-1 + \sqrt{1 + 8F_{r2}^2}}{2} \tag{3.9}$$

The possible minus in front of the square root is ignored since it produces a physically impossible negative depth.

If upon dividing Equation 3.8 by $(Y_1 - Y_2)$ the result had been multiplied by Y_2 instead of Y_1, then a quadratic equation in Y_2 will result. After using the quadratic formula the following equation results which is identical to Equation 3.9 with the subscripts reversed:

$$\frac{Y_2}{Y_1} = \frac{-1 + \sqrt{1 + 8q^2/(gY_1^3)}}{2} = \frac{-1 + \sqrt{1 + 8F_{r1}^2}}{2} \tag{3.10}$$

Thus regardless of whether the depth is known upstream, or downstream the other conjugate depth in a rectangular channel can be obtained by use of the explicit Equations 3.9 or 3.10.

Another useful form for the hydraulic jump equation, (or the equation resulting from equating m_1 to m_2, i.e., $m_1 = m_2$ can be obtained from Equation 3.8 by solving it for the Froude number squared, or

$$F_{r1}^2 = \frac{q^2}{gY_1^3} = \frac{1}{2}\frac{Y_2}{Y_1}\left(\frac{Y_2}{Y_1}+1\right) = 0.5Y'(Y'+1) \tag{3.11}$$

or

$$F_{r2}^2 = \frac{q^2}{gY_2^3} = \frac{1}{2}\frac{Y_1}{Y_2}\left(\frac{Y_1}{Y_2}+1\right) = \frac{Y'+1}{2Y'^2} \tag{3.12}$$

in which the latter parts of Equations 3.11 and 3.12 contain the dimensionless depth $Y' = Y_2/Y_1$.

The momentum function per unit width for critical flow m_c can be evaluated either from the critical depth, Y_c or the flow per unit width q_c. For critical flow $q_c^2/g = Y_c^3$ or $q_c^2/(gY_c) = Y_c^2$. Substituting this in Equation 3.6 gives

$$m_c = \frac{Y_c^2}{2} + Y_c^2 = 1.5Y_c^2 = 1.5\left(\frac{q_c^2}{g}\right)^{2/3}$$

It is interesting that the constant is also 1.5 = 3/2 in the relationship between critical specific energy and critical depth for a rectangular channel ($E_c = 1.5Y_c$); the difference is now Y_c is squared, as needed, so the dimensions are the same on both sides of the equation since the dimensions of m are L^2. In the case of E_c two-thirds of its value is the depth (potential energy/unit weight) and one-third the velocity head $V_c^2/(2g)$, whereas two-thirds of m_c comes from the momentum flux divided by the specific weight, or $\rho qV/\gamma = q_c^2/(gY_c) = Y_c^2$ and one-third from the hydrostatic force divided by γ.

Writing Equation 3.6 as a cubic equation gives

$$F(Y) = Y^3 - 2mY + \frac{2q^2}{g} = 0$$

Any general cubic equation $x^3 + px^2 + qx + r = 0$ can be put in the form $y^3 + ay + b = 0$ by substituting $y = p/3$ for x (see CRC Standard Math. Tables). Our momentum function per unit width cubic equation is already in this form. The three roots are

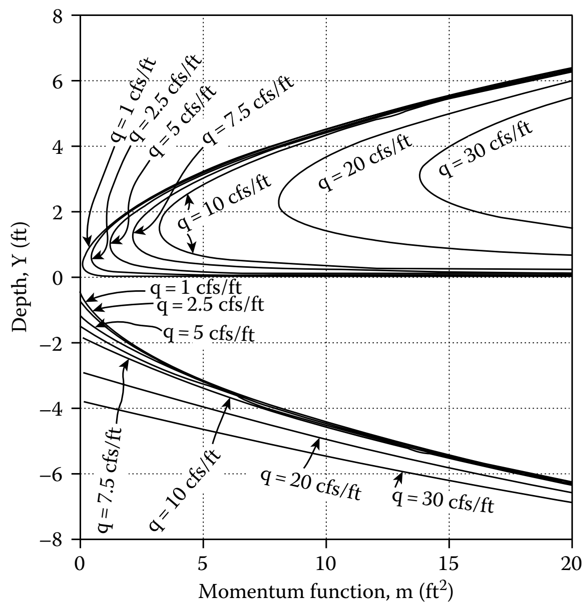
$$y_1 = A + B, \quad y_2 = 0.5(i\sqrt{3})(A - B) - 0.5y_1, \quad \text{and} \quad y_3 = 0.5(i\sqrt{3})(B - A) - 0.5y_1$$

in which

$$A = \left\{\left(\frac{b^2}{4} + \frac{a^3}{27}\right)^{1/2} - 0.5b\right\}^{1/3} \quad \text{and} \quad B = \left\{-\left(\frac{b^2}{4} + \frac{a^3}{27}\right)^{1/2} - 0.5b\right\}^{1/3}$$

If $b^2/4 + a^3/27 > 0$ there will be one real root and two conjugate imaginary roots.
If $b^2/4 + a^3/27 = 0$ there will be three real roots of which at least two are equal.
If $b^2/4 + a^3/27 < 0$ there will be three real and unequal roots.

For the cubic momentum equation $a = -2m$ and $b = 2q^2/g$. The first condition occurs for values of $m < m_c$ and the one real root gives a negative depth (which of course has no physical meaning); the second condition represents critical flow conditions; and the third condition where three real unequal roots occur is when $m > m_c$ and there are two positive real roots that are the two conjugate depths, and the third root gives a negative depth. The graph below is a plot obtained by solving the cubic momentum equation for $m \geq m_c$ for q = 1, 2.5, 5.0, 7.5 and 10.0 cfs/ft. One can perform the above complex arithmetic to find the conjugate depths, but it is much easier to use Equations 3.9 or 3.10.



If one only knows the momentum function per unit width for a rectangular channel, and the unit flow rate q, and not one of the conjugate depths so Equations 3.9 or 3.10 cannot be used, it is slightly easier to use the solution for the above cubic F(Y) that involves the arc cosine (and cosine), as was used in solving the alternate depths for a given specific energy in the previous chapter. To use this alternative method for solving the cubic equation for its three roots first compute the angle θ from the first root Y_1 is the negative depth, the second root Y_2 is $\theta = \cos[(q^2/g)/(2m/3)^{1.5}]$ and thereafter solve for the three roots from the subcritical depth and the third root Y_3 is the supercritical depth.

$$Y_1 = -2\left(\frac{2m}{3}\right)^{1/2}\cos\left(\frac{\theta}{3}\right),\quad Y_2 = -2\left(\frac{2m}{3}\right)^{1/2}\cos\left\{\frac{(\theta+2\pi)}{3}\right\},\quad Y_3 = -2\left(\frac{2m}{3}\right)^{1/2}\cos\left\{\frac{(\theta+4\pi)}{3}\right\}$$

The first root Y_1 is the negative depth, the second root Y_2 is the subcritical depth and the third root Y_3 is the supercritical depth. The program ROOTSM, listed below is designed to provide these three depths given the unit flow rate q, the unit momentum function m, and the acceleration of gravity.

Note that it computes critical depth and the associated unit momentum function, and tells you if you give an m less than m_c.

Program ROOTSM.FOR

```
      PARAMETER (PI=3.14159265)
1     WRITE(*,*)' Give: q,m (momentum/width),g'
      READ(*,*) q,F,g
      IF(q.LT.1.E-4) STOP
      qg=q*q/g
      Yc=qg**.3333333
      Fc=.5*Yc**2+qg/Yc
      IF(F.LT.Fc) THEN
      WRITE(*,100) Fc,F,Yc
100   FORMAT(' No real roots exist since m<mc m(crit)=',F8.3,
     &' m=',F8.3,/,' Yc=',F8.3)
      GO TO 1
      ENDIF
      FF=2.*sqrt(.66666667*F)
      THETA=ACOS(qg/(.6666667*F)**1.5)
      Y1=-FF*COS(THETA/3.)
      Y2=-FF*COS((THETA+2.*PI)/3.)
      Y3=-FF*COS((THETA+4.*PI)/3.)
      WRITE(*,*) Y1,Y2,Y3
      GO TO 1
      END
```

An explicit equation giving the head loss across a hydraulic jump in a rectangular channel can be obtained as a function of the upstream and downstream depths, Y_1 and Y_2. The head loss caused by a hydraulic jump is given by

$$h_L = E_1 - E_2 = Y_1 - Y_2 + \frac{q^2}{2g}\left(\frac{1}{Y_1^2} - \frac{1}{Y_2^2}\right) \tag{3.13}$$

in which the expression after the second equal sign applies only for rectangular channels. From Equation 3.12

$$\frac{q^2}{(2g)} = \left(\frac{1}{4}\right)(Y_1Y_2)(Y_1 + Y_2)$$

which when substituted into Equation 3.13 gives

$$h_L = Y_1 - Y_2 + \frac{(Y_1 + Y_2)(Y_2^2 - Y_1^2)}{4Y_1Y_2}$$

Upon multiplying $(Y_1 - Y_2)$ by $4Y_1Y_2$ to give it the same denominator, the numerator becomes the cube of $(Y_1 - Y_2)$ and the head loss is given by

$$h_L = \frac{(Y_2 - Y_1)^3}{4Y_1Y_2} \tag{3.14}$$

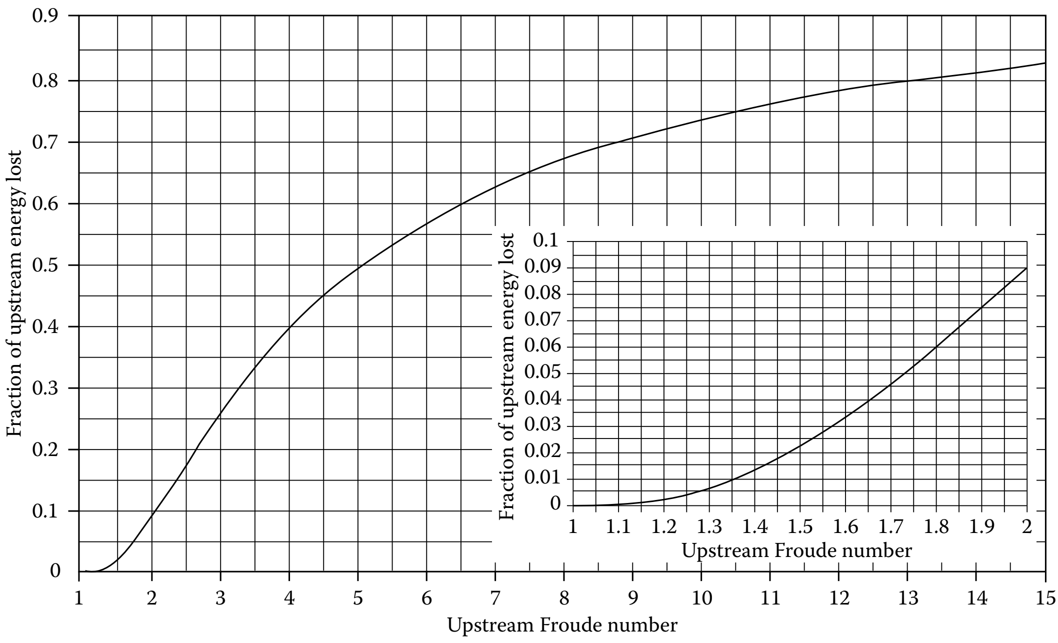
By combining Equation 10, which gives the ratio of the conjugate depths Y_2/Y_1 as a function of the upstream Froude number, the head loss across a hydraulic jump (Equation 3.14) can be expressed by the following equation as a function of the upstream Froude number F_{r1} and the upstream depth Y_1:

$$h_L = \frac{Y_1\left\{\sqrt{1+8F_{r1}^2}-3\right\}^3}{16\left\{\sqrt{1+8F_{r1}^2}-1\right\}} \tag{3.14a}$$

and as a fraction of the upstream specific energy $E_1 = Y_1 + q^2/(2gY_1^2) = Y_1(1+F_{r1}^2/2)$ by the following equation that shows that h_L/E_1 is only a function of the upstream Froude number, F_{r1}.

$$\frac{h_L}{E_1} = \frac{\left\{\left(1+8F_{r1}^2\right)^{1/2}-3\right\}^3}{8\left(2+F_{r1}^2\right)\left\{\left(1+8F_{r1}^2\right)^{1/2}-1\right\}} \tag{3.14b}$$

Note from the plot of h_L/E_1 versus F_{r1} that the fraction of total upstream energy lost through a hydraulic jump in a rectangular channel is relatively small for upstream Froude numbers only modestly larger than unity, and increases rapidly for large upstream Froude numbers.



If Equation 3.14 is written using the dimensionless depth $Y' = Y_2/Y_1$ then the following dimensionless equation results:

$$h'_L = \frac{h_L}{Y_1} = \frac{(Y'-1)^3}{4Y'} \tag{3.14c}$$

A plot of h'_L versus Y' below shows how the dimensionless headloss in a rectangular channel rapidly increases from 0, when $Y' = 1$ to 18.225, when $Y' = 10$. When $Y' = 100$, $h'_L = 2425.75$.

Another dimensionless relationship giving the headloss in a rectangular channel can be obtained by defining the following dimensionless alternative depths by dividing by the critical depth $Y_c : Y'_{1c} = Y_1/Y_c$ and $Y'_{2c} = Y_2/Y_c$. Then Equation 3.14 becomes

$$h'_{Lc} = \frac{h_L}{Y_c} = \frac{(Y'_{2c} - Y'_{1c})^3}{4Y'_{1c}Y'_{2c}} \tag{3.14d}$$

If $Y_c = (q^2/g)^{1/3}$ is substituted in the middle part of this equation and both numerator and denominator of the result divided by Y_1, then $h_L/Y_c = (h_L/Y_1)/F_{r1}^{2/3} = h'_L/F_{r1}^{2/3}$ and Equation 3.14d becomes

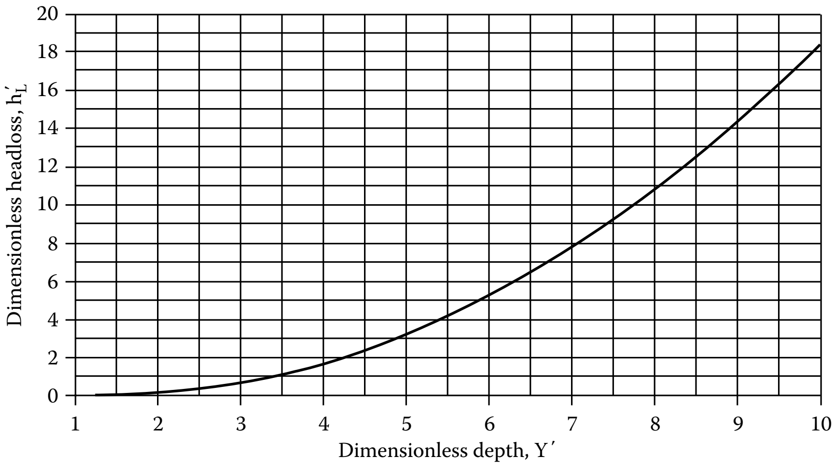
$$h'_L = \frac{(Y'_{2c} - Y'_{1c})^3}{4Y'_{1c}Y'_{2c}} F_{r1}^{2/3} \tag{3.14e}$$

Since Equation 3.10 gives the dimensionless depth $Y' = 0.5\{(1+8F_{r1}^2)^{1/2} - 1\}$, and $Y'_{1c} = Y_1/Y_c = (Y_1/Y_2)/\left[q^2/(gY_2^3)\right]^{1/3} = 1/(Y'F_{r2}^{2/3})$ and $Y'_{2c} = Y_2/Y_c = (Y_2/Y_1)/\left[q^2/(gY_1^3)\right]^{1/3} = Y'/F_{r1}^{2/3}$ and

$$F_{r2}^2 = \frac{q^2}{gY_2^3} = \frac{q^2/(gY_1^3)}{Y'^3} = \frac{F_{r1}^2}{Y'^3}$$

it is possible to express all of these dimensionless variables as a function of the upstream Froude Number, F_{r1}. The table below provides these dimensionless variables, and the accompanying figure plots them.

F_{r1}	y'	F_{r2}	y'_{1c}	y'_{2c}	h'_L
1.0	1.000	1.000	1.000	1.0000	0.0000
1.5	1.6794	0.6892	0.7631	1.2817	0.0467
2.0	2.3723	0.5474	0.6300	1.4944	0.2723
2.5	3.0707	0.4646	0.5429	1.6670	0.7229
3.0	3.7720	0.4095	0.4807	1.8134	1.4117
3.5	4.4749	0.3697	0.4338	1.9412	2.3442
4.0	5.1789	0.3394	0.3969	2.0553	3.5228
4.5	5.8836	0.3153	0.3669	2.1586	4.9489
5.0	6.5887	0.2956	0.3420	2.2533	6.6233
5.5	7.2942	0.2792	0.3209	2.3410	8.5465
6.0	8.0000	0.2652	0.3029	2.4228	10.7188
6.5	8.7060	0.2530	0.2871	2.4996	13.1403
7.0	9.4121	0.2424	0.2733	2.5721	15.8113
7.5	10.1184	0.2330	0.2610	2.6408	18.7319
8.0	10.8248	0.2246	0.2500	2.7062	21.9022
8.5	11.5312	0.2171	0.2401	2.7686	25.3221
9.0	12.2377	0.2102	0.2311	2.8284	28.9918
9.5	12.9443	0.2040	0.2229	2.8858	32.9114
10.0	13.6510	0.1983	0.2154	2.9410	37.0807



Graph showing how the dimensionless head loss $h'_L = h_L / Y_1$ varies with the dimensionless depth $Y' = Y_2/Y_1$.

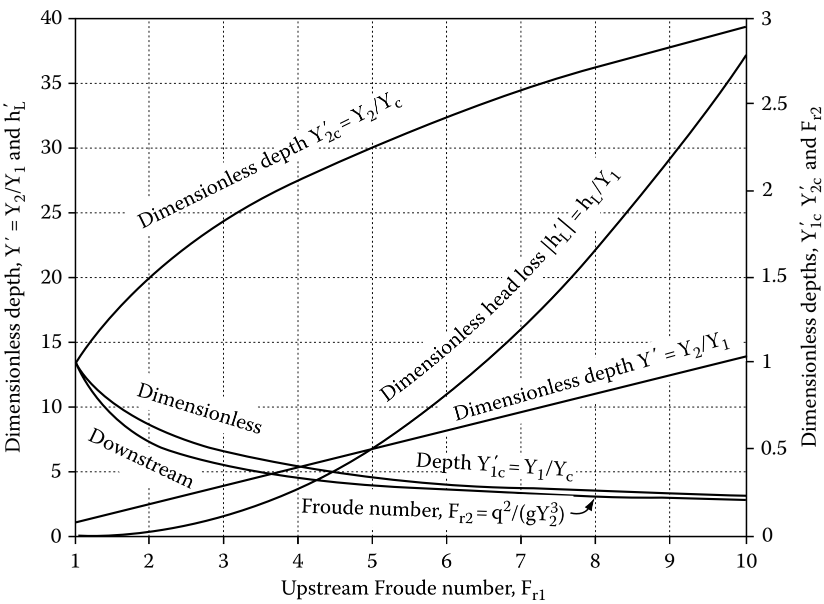


Figure relationship of dimensionless variables to upstream Froude number

3.4 POLYNOMIAL FORM FOR MOMENTUM FUNCTION

A natural question to raise is whether it is possible to simplify the hydraulic jump equation to an explicit form for other cross sections besides a rectangle. The answer is no, but it is useful to go through similar attempts for a trapezoidal section because doing so provides insights into the characteristics of the hydraulic jump equation. To simplify the notation let $r = Y_2/Y_1$ and t = (area of the rectangular part)/(area of the triangular parts) = $b/(mY_1)$ (which you might note is the reciprocal of the dimensionless depth Y′ used in defining Figure 2.4, the dimensionless specific energy diagram). After some algebraic manipulation of the momentum equation $M_1 = M_2$ for a trapezoidal section the following fifth degree polynomial can be produced.

$$r^5 + 2.5tr^4 + 1.5t^2r^3 - (1.5t + 3F^2(t+1) + 1)r^2 - (1.5t^2 + t + 3F^2t(t+1))r + 3F^2(t+1)^2 = 0 \quad (3.15)$$

in which $F^2 = V_1^2/(gY_1)$. Note F is not the Froude number for a trapezoidal channel. This fifth degree polynomial can be reduced to the following fourth degree polynomial by extracting the root r = 1:

$$r^4 + (2.5t+1)r^3 + (1.5t+1)(t+1)r^2 + (0.5t^2 + (t - 3F^2)(t+1))r - 3F^2(t+1)^2 = 0 \quad (3.16)$$

These equations are also reversible, i.e., r could be defined as Y_1/Y_2 and the subscript for t and F could be 2, and identically the same equations would result. For a triangular section, (i.e., b = 0, t = 0) the above fourth degree polynomial reduces to:

$$r^4 + r^3 + r^2 - 3F^2(r+1) = 0 \quad (3.17)$$

In general a fourth degree polynomial will have four roots, or solutions. Should there be imaginary roots these must occur as complex conjugate pairs. Since two real roots exist that have the same value for the momentum function in a trapezoidal channel, e.g., the conjugate depths, we can conclude that two complex, or imaginary roots exist, or two additional real but negative roots exist. The root for the known real depth was eliminated by reducing the fifth degree polynomial to a fourth degree polynomial.

In solving problems involving finding a conjugate depth, the only advantage in solving Equation 3.16 over solving Equation 3.3 is that much is known about finding roots of polynomials. However, Equation 3.16 is restricted to trapezoidal channels and Equation 3.3 is general for all cross sections. Later in this chapter general methods for finding roots of a polynomial will be used to solve a dimensionless form of the momentum function equation.

EXAMPLE PROBLEM 3.3

The depth upstream from a hydraulic jump in a rectangular channel is 0.6 m. If the channel has a bottom width b = 5 m and contains a flow rate of Q = 20 m³/s, what depth of flow would be expected downstream from the hydraulic jump? How much head loss occurs through the jump, and how much power in horsepower is dissipated?

Solution

The solution for the conjugate depth after the hydraulic jump can be computed directly from Equation 3.10 or $Y_2 = 0.6\left[-1 + \sqrt{1 + 8q^2/(gY^3)}\right]\Big/2 = 2.05$ m. The head loss $h_L = E_1 - E_2 = 2.87 - 2.24 = 0.63$ m. The power dissipated = $\gamma Q h_L$ = 9.8(20)(0.63) = 123.48 kW or hp = 123.48/0.746 = 165.5 hp.

Should an external force exist on a control volume of fluid, then Equation 3.3 can be modified to allow for this by noting that in the development of the momentum function M, it is equal to force divided by the specific weight γ of the fluid or $M(L^3) = F/\gamma(F/(F/L^3))$ and therefore Equation 3.3 might be generalized to

$$M_1 = M_2 + \frac{F}{\gamma}$$

and likewise if a rectangular channel is involved the momentum equation per unit width becomes,

$$m_1 = m_2 + \frac{f}{\gamma}$$

in which f = F/b.

EXAMPLE PROBLEM 3.4

A sluice gate is positioned 0.6 ft above the channel bottom. Its contraction coefficient is $C_c = 0.6$. If a flow rate of $q = 10$ cfs/ft is passing the gate determine the force on the gate. How does this actual force compare with the force computed assuming that the pressure on the gate were hydrostatic?

Solution

The solution of the energy equation $E_1 = E_2$ (with $Y_2 = (0.6)(0.6) = 0.36$ ft) gives $Y_1 = 12.33$ ft. Therefore the force per unit width is obtained from $f = \gamma(m_1 - m_2) = \gamma\{Y_1^2/2 - Y_2^2/2 + q^2/g(1/Y_1 - 1/Y_2)\} = 62.4(75.95 - 8.38) = 4216.4$ lb/ft. If the force is computed from a hydrostatic pressure distribution its magnitude equals $\gamma(E - y_G)^2/2 = 62.4(12.34 - 0.6)^2/2 = 4300.4$ lb/ft, or 2% larger. The reason for the larger hydrostatic force is that the actual pressure distribution goes to zero at the tip of the gate.

EXAMPLE PROBLEM 3.5

Baffles exist at the toe of a dam to stabilize the hydraulic jump on the apron of the dam. The total cross section of these baffles normal to the direction of flow is 20 ft^2, and they exist in a trapezoidal channel with $b = 10$ ft, and $m = 1.2$. This channel extends downstream from the dam for a long distance at a slope of $S_o = 0.0012$, and its roughness coefficient is $n = 0.014$. If the drag coefficient for the baffles has been determined equal to 0.75 based on the downstream dynamic pressure, determine what depth of the upstream flow should be if the flow rate is $Q = 500$ cfs.

Solution

Uniform flow exists downstream. Therefore solving Manning's formula gives the depth downstream from the hydraulic jump as $Y_2 = 4.425$ ft and $M_2 = 247.17$ ft^3. The velocity downstream is $V_2 = 7.38$ fps, and therefore the drag force against the baffles can be obtained from Drag $= C_D A_b(\rho V^2/2) = 803.05$ lb. The upstream depth can now be solved from the momentum equation $M_1 = M_2 - \text{Drag}/\gamma = 260.04$ ft^3. Solution of this equation by the Newton method gives $Y_1 = 1.924$ ft. Without the baffles the upstream depth must be larger for the downstream depth to be possible, i.e., the conjugate depth to 4.425 ft, which is 2.973 ft.

3.5 DIMENSIONLESS MOMENTUM FUNCTIONS

Dimensionless momentum functions have utility similar to dimensionless-specific energy curves. Dimensionless momentum functions will be developed for both trapezoidal and circular sections in the follow few paragraphs. For a trapezoidal section the momentum function is

$$M = \frac{1}{2}bY^2 + \frac{1}{3}mY^3 + \frac{Q^2}{g(bY + mY^2)}$$

This equation can be rearranged as

$$M = \frac{b^3}{m^2}\left(\frac{m^2Y^2}{2b^2} + \frac{m^3Y^3}{3b^3}\right) + \frac{Q^2/g}{b^2/m(mY/b + m^2Y^2/b^2)}$$

If the following dimensionless parameters are defined: $M' = m^2M/b^3$, $Y' = mY/b$ and $Q' = m^3Q^2/(gb^5)$, then the above equation can be written as the following dimensionless momentum function:

$$M' = \frac{1}{2}Y'^2 + \frac{1}{3}Y'^3 + \frac{Q'}{Y' + Y'^2} \tag{3.18}$$

Dimensionless momentum function curves for a number of values of Q' are plotted on Figure 3.1. The two dimensionless depths associated with a constant value of M' on this graphs for a given curve Q' are the dimensionless conjugate depths. Dimensionless critical depth Y'_c is also shown

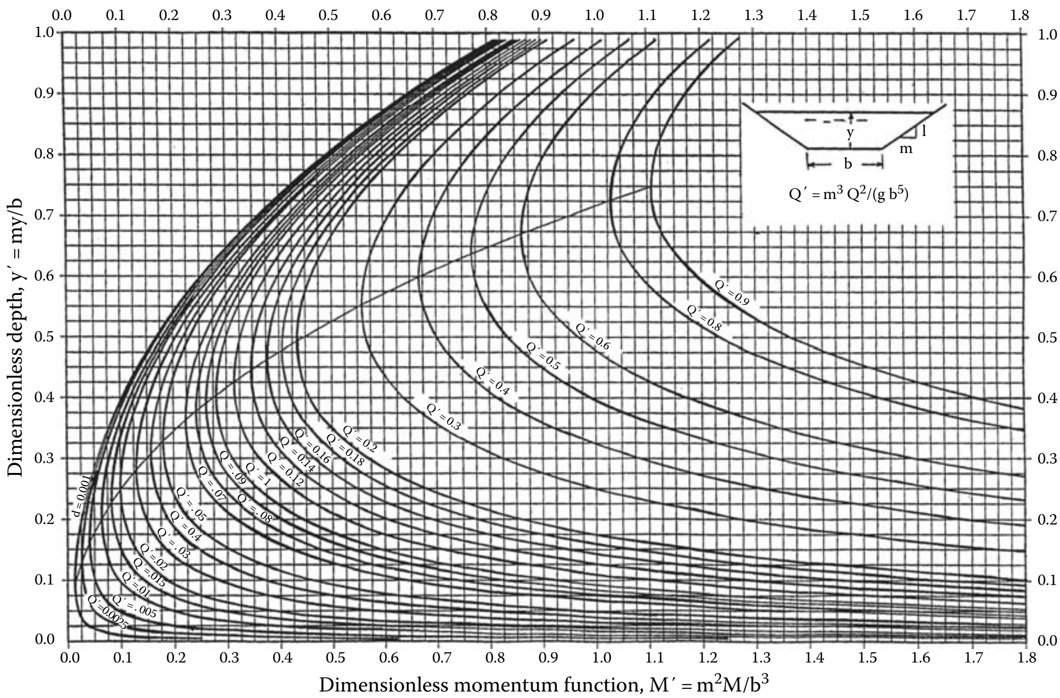


FIGURE 3.1 Dimensionless momentum function diagrams for trapezoidal sections. (Individual curves apply for $Q' = m^3Q^2/(gb^5)$.)

in the graph. The graph can be used to solve problems associated with the momentum function. In addition to having the dimensionless critical depth read as the minimum point on the curves on Figure 3.1, the critical flow equation $Q^2T/(gA^3) = 1$ can be written using the same dimensionless variables. For a trapezoidal section the critical flow equation becomes $Q^2(b + 2mY) = Q^2b(1 + 2Y') = (b^2/m)^3g(mbY/b^2 + m^2Y^2/b^2)^3$, or letting $Q' = m^3Q^2/(gb^5)$, the dimensionless critical flow equation for a trapezoidal section becomes

$$Q' = \frac{(Y' + Y'^2)}{(1 + 2Y')} \tag{3.18a}$$

and this equation gives Y' corresponding to the minimum M' for any Q' on Figure 3.1, or the position on the Y' ordinate where critical flow occurs for any Q' curve.

A similar graph for a circular channel is given in Figure 3.2. Dimensionless momentum functions for a circular channel can be obtained by defining the dimensionless depth in a circle as $Y' = Y/D$. Then with auxiliary angle defined from $\beta = \cos^{-1}(1 - 2Y')$ the momentum function is

$$M = \frac{1}{2}\left(\frac{1}{6}D^2\sin^3\beta - \frac{1}{4}D^2(\beta\cos\beta - \sin\beta\cos^2\beta)\right) + \frac{Q^2/g}{\frac{1}{4}D^2(\beta - \cos\beta\sin\beta)}$$

If M' is defined as $M' = M/D^3$ and $Q' = Q^2/(gD^5)$, then the following dimensionless momentum function occurs:

$$M' = \frac{1}{2}\left(\frac{1}{6}\sin^3\beta + \frac{1}{4}\cos\beta(\sin\beta\cos\beta - \beta)\right) + \frac{4Q'}{\beta - \sin\beta\cos\beta} \tag{3.19}$$

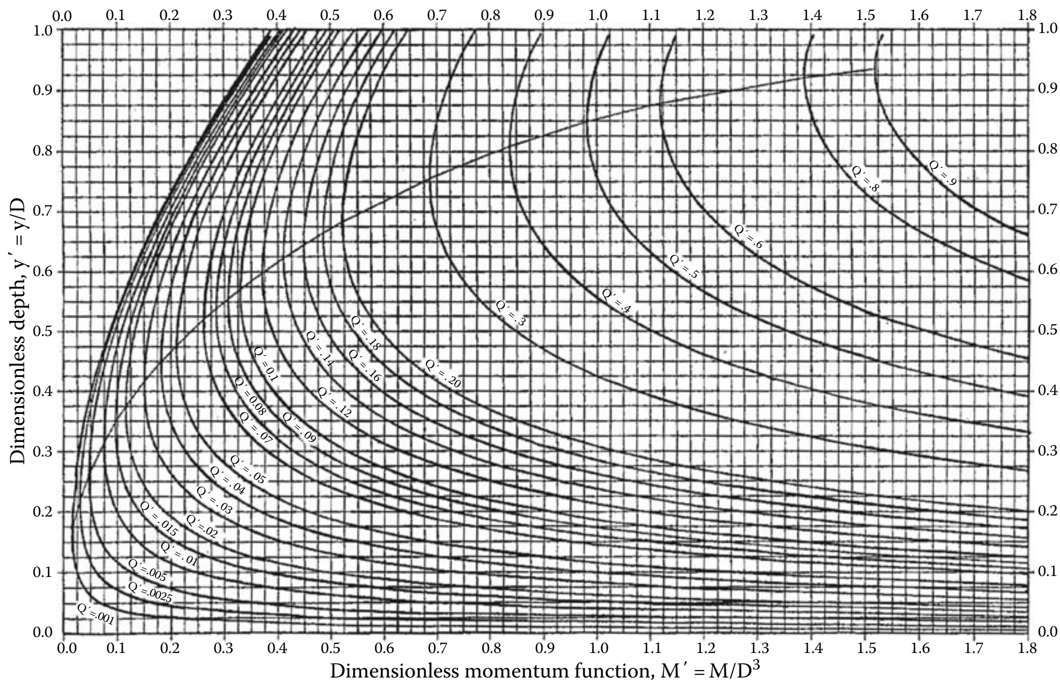


FIGURE 3.2 Dimensionless momentum function diagrams for circular sections. (Individual curves apply for $Q' = Q^2/(gD^5)$.)

You should note that Y′ and M′ are defined differently for a trapezoidal section than for a circular section. The dimensionless critical flow equation for a circular section is

$$Q' = \frac{(\beta - \cos\beta\sin\beta)^3}{64\sin\beta} \tag{3.19a}$$

and this equation gives the ordinate Y′ of the minimum position on any Q′ curve on Figure 3.2.

There are often advantages in solving a dimensionless equation since much of the variation due to the size of the variables no longer exists. If a hydraulic jump occurs in a trapezoidal channel then the dimensionless momentum function M_1' as defined by Equation 3.18 equals this value evaluated downstream from the hydraulic jump, or

$$\frac{1}{2}Y_1'^2 + \frac{1}{3}Y_1'^3 + \frac{Q'}{Y_1' + Y_1'^2} = \frac{1}{2}Y_2'^2 + \frac{1}{3}Y_2'^3 + \frac{Q'}{Y_2' + Y_2'^2}$$

If this equation is manipulated into the form of a polynomial in Y_1', it is possible to eventually produce the following equation that assumes that Y_2' is known:

$$Y_1'^4 + (2.5 + Y_2')Y_1'^3 + (Y_2'^2 + 2.5Y_2' + 1.5)Y_1'^2 + \left\{Y_2'^2 + 1.5Y_2' - \frac{3Q'}{(Y_2'^2 + Y_2')}\right\}Y_1' - \frac{3Q'}{Y_2'} = 0$$

This equation is a fourth degree polynomial, and therefore would be expected to possibly have four real roots. In the process of obtaining Equation 3.20 the value $(Y_1' - Y_2')$ was divided out, and therefore the known root downstream from the hydraulic jump is no longer a solution to Equation 3.20. It turns out that two of the possible roots are generally complex conjugate roots, but may become real negative roots, and one of the remaining roots always gives a negative value for Y_1', and since negative roots are generally complex conjugate roots, but may become negative roots, and one of the remaining roots always gives a negative value for Y_1', and since negative roots are physically not possible, it is the remaining real root with a positive value for Y_1' that is sought. The trick is how do you find this root best? A host of iterative methods might be used, including the Newton method, which has been used extensively to this point. It is informative to utilize a method capable of finding all roots including the complex roots. Laguerre's method is well suited for this purpose. However, since Laguerre's method is quite involved, it is not described herein. For a description of this method for finding roots of a polynomial consult a book that deals with numerical methods. The following FORTRAN program is designed to extract all four roots from Equation 3.20, print them out, and then compute the depth which is conjugate to the given depth. Appendix B describes the arguments of Subroutine LAGU. The results from the program are applicable only for trapezoidal channels.

Listing of FORTRAN program LAGU.FOR that uses Laguerre's method of finding all roots of dimensionless momentum function $M_1' = M_2'$.

```
      PARAMETER (ND=4,EPS=1.E-6)
      REAL m
      COMPLEX C(ND+1),ROOTS(ND),AD(ND+1),Z1,Z2,Z3
      EPS1=2.*EPS*EPS
10    WRITE(6,*)' Give:b,m,Q,g & Y or one of conjugate depths'
      READ(5,*) b,m,Q,G,Y
      QP=(Q/b)**2*(m/b)**3/G
      YP=m*Y/b
      C(5)=CMPLX(1.,0.)
      C(4)=CMPLX(2.5+YP,0.)
      YP1=YP*(YP+1.)
      C(3)=CMPLX(YP1+1.5*(YP+1.),0.)
      C(2)=CMPLX(YP1+.5*YP-3.*QP/YP1,0.)
      C(1)=CMPLX(-3.*QP/YP,0.)
      DO 20 J=1,ND+1
20    AD(J)=C(J)
      DO 30 J=ND,1,-1
      Z1=CMPLX(0.,0.)
      CALL LAGU(AD,J,Z1,EPS)
      IF(ABS(AIMAG(Z1)).LE.EPS1*ABS(REAL(Z1)))
     & Z1=CMPLX(REAL(Z1),0.)
      ROOTS(J)=Z1
      Z2=AD(J+1)
      DO 30 JJ=J,1,-1
      Z3=AD(JJ)
      AD(JJ)=Z2
30    Z2=Z1*Z2+Z3
      DO 50 J=2,ND
      Z1=ROOTS(J)
      DO 40 I=J-1,1,-1
      IF(REAL(ROOTS(I)).LE.REAL(Z1)) GO TO 50
40    ROOTS(I+1)=ROOTS(I)
      I=0
```

```
50    ROOTS(I+1)=Z1
      DO 60 I=1,ND
      IF(AIMAG(ROOTS(I)).NE. 0.) GO TO 60
      IF(REAL(ROOTS(I)).LT. 0.) GO TO 60
      YP=REAL(ROOTS(I))
60    WRITE(6,70) ROOTS(I)
70    FORMAT(2F12.4)
      WRITE(6,80) Y,YP*b/m
80    FORMAT(' Conjugate depth to',F10.3,' equals ',F10.3,//)
      WRITE(6,*)' Give 1 if to solve another problem,',
     &' otherwise 0'
      READ(5,*) I
      IF(I.EQ.1) GO TO 10
      STOP
      END
      SUBROUTINE LAGU(C,ND,Z1,EPS)
      COMPLEX C(ND+1),Z1,DX,ZO,Z2,Z3,Z4,Z5,DZ,SS,Z6,Z7,Z8,
     &ZERO,XX,FF
      ZERO=CMPLX(0.,0.)
      DO 20 ITER=1,50
      Z2=C(ND+1)
      Z3=ZERO
      Z4=ZERO
      DO 10 J=ND,1,-1
      Z4=Z1*Z4+Z3
      Z3=Z1*Z3+Z2
10    Z2=Z1*Z2+C(J)
      IF(CABS(Z2).LE.1.E-8) THEN
      DX=ZERO
      ELSE IF(CABS(Z3).LE.1.E-8.AND.CABS(Z4).LE.1.E-8) THEN
     &DX=CMPLX(CABS(Z2/C(ND+1))**(1./FLOAT(ND)),0.)
      ELSE
      Z5=Z3/Z2
      Z8=Z5*Z5
      DZ=Z8-2.*Z4/Z2
      XX=(ND-1)*(ND*DZ-Z8)
      YY=ABS(REAL(XX))
      ZZ=ABS(AIMAG(XX))
      IF(YY.LT.1.E-12.AND. ZZ.LT.1.E-12) THEN
      SS=ZERO
      ELSE IF (YY.GE.ZZ) THEN
      FF=(1./YY)*XX
      SS=SQRT(YY)*CSQRT(FF)
      ELSE
      FF=(1./ZZ)*XX
      SS=SQRT(ZZ)*CSQRT(FF)
      ENDIF
      Z6=Z5+SS
      Z7=Z5-SS
      IF(CABS(Z6).LT.CABS(Z7)) Z6=Z7
      DX=FLOAT(ND)/Z6
```

```
      ENDIF
      ZO=Z1-DX
      IF(Z1.EQ.ZO) RETURN
      Z1=ZO
      IF(CABS(DX).LE.EPS*CABS(Z1)) RETURN
20    CONTINUE
      WRITE(6,*)' FAILED TO CONVERGE'
      RETURN
      END
```

Listing of program LAGU.CPP

```
#include <complex.h>
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
void lagu(complex *c,int nd,complex *z1,float eps){
 complex dx,zo,z2,z3,z4,z5,dz,ss,z6,z7,z8,zero=complex(0.,0.),xx,ff;
   int iter,j;float yy,zz;
   for(iter=1;iter<=50;iter++){z2=c[nd]; z3=zero;z4=zero;
  for(j=nd;j>0;j--){z4=(*z1)*z4+z3;z3=(*z1)*z3+z2;
   z2=(*z1)*z2+c[j-1];}
  if(abs(z2)<=1.e-20) dz=zero;
  else if((abs(z3)<=1.e-8)&&(abs(z4)<=1.e-8))\
    dx=complex(pow(abs(z2/c[nd]),1./(float) nd),0.);
  else {z5=z3/z2;z8=z5*z5;dz=z8-2.*z4/z2;
    xx=(float)(nd-1)*((float)nd*dz-z8); yy=fabs(real(xx));
    zz=fabs(imag(xx)); if((yy<1.e-12)&&(zz<1.e-12)) ss=zero;
  else if(yy>=zz){
  ff=(1./yy)*xx;ss=sqrt(yy)*sqrt(ff);}
  else {ff=(1./zz)*xx;ss=sqrt(zz)*sqrt(ff);}
 z6=z5+ss;z7=z5-ss;if(abs(z6)<abs(z7))z6=z7;dx=(float) nd/z6;}
 zo=(*z1)-dx; if((*z1)==zo) return; *z1=zo;
   if(abs(dx)<=eps*abs(*z1)) return;}
 printf("FAILED TO CONVERGE\n");} // end of lagu
void main(void){float eps=1.e-4,eps1,b,m,q,g,y,qp,yp,yp1;
 int nd=4,j,jj,i,ii;
 complex c[5],roots[4],ad[5],z1,z2,z3,*zz1;
 eps1=2.*eps*eps;
L10: printf("Give:b,m,Q,g & Y or one of conjugate depths\n");
 scanf("%f %f %f %f %f",&b,&m,&q,&g,&y);
 qp=pow(q/b,2.)*pow(m/b,3.)/g;yp=m*y/b;
 c[4]=complex(1.,0.); c[3]=complex(2.5+yp,0.); yp1=yp*(yp+1.);
 c[2]=complex(yp1+1.5*(yp+1.),0.);
 c[1]=complex(yp1+.5*yp-3.*qp/yp1,0.); c[0]=complex(-3.*qp/yp,0.);
 for(j=0;j<=nd;j++) ad[j]=c[j];
 for(j=nd;j>0;j--){z1=complex(0.,0.);*zz1=z1;
 lagu(ad,j,zz1,eps); z1=(*zz1);
  if(fabs(imag(z1))<=eps1*fabs(real(z1))) z1=complex(real(z1),0.);
 roots[j-1]=z1; z2=ad[j];for(jj=j;jj>0;jj--){z3=ad[jj-1];
   ad[jj-1]=z2;z2=z1*z2+z3;}}
```

```
 for(j=2;j<=nd;j++){z1=roots[j−1]; for(i=j−1;i>0;i−−){ii=i−1;
   if(real(roots[ii])<=real(z1)) goto L50;
   roots[i]=roots[ii];}ii=−1;
L50:roots[ii+1]=z1;}
 for(i=1;i<=nd;i++){ii=i−1;if(imag(roots[ii])!=0.) goto L60;
   if(real(roots[ii])<0.) goto L60;yp=real(roots[ii]);
L60:printf("%12.4f %11.4f\n",roots[ii]);}
 printf("Conjugate depth to %10.3f equals %10.3f \n\n",y,yp*b/m);
 printf("Give 1 if to solve another problem, otherwise 0\n");
 scanf("%d",&i);
 if(i>0) goto L10;}
```

Equation 3.20 is written so that Y_1' is unknown, and Y_2' is known. By developing the equation slightly differently, it is possible to obtain an equation identical to Equation 3.20 except that the 1 and 2 subscripts are interchanged. In other words Equation 3.20 is valid if we consider Y_2' the depth upstream from the hydraulic jump and Y_1' the depth downstream from the hydraulic jump. Thus while one of the purposes associated with examining dimensionless momentum functions was to obtain graphical solutions for problems controlled by the momentum principle, we see that their usefulness extends also into solving these problems numerically. The Newton method might be used in solving Equation 3.20 in place of Laguerre's method. It is generally easier to supply a good enough starting value to a dimensionless equation, than to the original equation.

An alternative form of the dimensionless momentum equation to Equation 3.20 that contains both real positive conjugate roots plus the roots that are physically not possible is to use Equation 3.18 directly so that $M' = m^2M/b^3$ is evaluated using the known depth. Upon multiplying Equation 3.18 through by the denominator under Q' the following fifth degree polynomial equation results:

$$\frac{Y'^5}{3} + \left(\frac{5}{6}\right)Y'^4 + \frac{Y'^3}{2} - M'Y'^2 - M'Y' + Q' = 0 \tag{3.18a}$$

The subscripts are left off from the dimensionless variables, but Q' and M' will be evaluated using the known dimensionless depth, and the roots of the equation for Y' will include this value as one of the possibilities. To extract the roots from this equation using the above program that implements the Laguerre's method the degree of polynomial needs to be increased by 1 by changing ND to 5 and defining the coefficient with the following statements rather than those in the listed program.

Fortran

```
      QP=(Q/b)**2*(m/b)**3/G
      YP=m*Y/b
      MP=(.5+YP/3.)*YP**2+QP/(YP*(1.+YP))
      C(6)=CMPLX(.3333333,0.)
      C(5)=CMPLX(.8333333,0.)
      C(4)=CMPLX(.5,0.)
      C(3)=CMPLX(−MP,0.)
      C(2)=CMPLX(−MP,0.)
      C(1)=CMPLX(QP,0.)
```

C

```
      qp=pow(q/b,2.)*pow(m/b,3.)/g;
      yp=m*y/b;
      mp=(.5+yp/3.)*yp*yp+qp/(yp*(1.+yp));
      c[5]=complex(.3333333,0.);
```

```
c[4]=complex(.8333333,0.);
c[3]=complex(.5,0.);
c[2]=complex(-mp,0.);c[1]=complex(-mp,0.);
c[0]=complex(qp,0.);
```

The other change desirable is to bring the WRITE statement within the DO 60 loop so that both real positive roots are written out.

The same procedure and program can be used to obtain all roots from the dimensionless specific energy Equation 2.27. Rewriting this equation as a fifth degree polynomial results in

$$Y'^5 + (2-E')Y'^4 + (1-2E')Y'^3 - E'Y'^2 + \frac{Q'}{2} = 0$$

and the statements that define the polynomial coefficients in the program to find the roots of this equation can be modified to the following (LAGUE.FOR):

```
QP=(Q/b)**2*(m/b)**3/g
YP=m*Y/b
EP=.5*QP/(YP+YP**2)**2+YP
C(6)=CMPLX(1.,0.)
C(5)=CMPLX(2.-EP,0.)
C(4)=CMPLX(1.-2.*EP,0.)
C(3)=CMPLX(-EP,0.)
C(2)=CMPLX(0.,0.)
C(1)=CMPLX(.5*QP,0.)
```

To handle any of these problems, the main program can be modified to read in the degree of the polynomial and its coefficients as given below.

Listing of program LAGU5.FOR designed to extract roots of any polynomial

```
      PARAMETER (EPS=1.E-6)
      REAL CO(10)
      COMPLEX C(11),ROOTS(10),AD(11),Z1,Z2,Z3
      EPS1=2.*EPS*EPS
10    WRITE(6,*)' Give: ND, C0,C1,C2,..CND'
      READ(5,*) ND,(CO(I),I=1,ND+1)
      DO 15 I=1,ND+1
15    C(I)=CMPLX(CO(ND+2-I),0.)
      DO 20 J=1,ND+1
20    AD(J)=C(J)
      DO 30 J=ND,1,-1
      Z1=CMPLX(0.,0.)
      CALL LAGU(AD,J,Z1,EPS)
      IF(ABS(AIMAG(Z1)).LE.EPS1*ABS(REAL(Z1)))Z1=CMPLX(REAL(Z1),0.)
      ROOTS(J)=Z1
      Z2=AD(J+1)
      DO 30 JJ=J,1,-1
      Z3=AD(JJ)
      AD(JJ)=Z2
30    Z2=Z1*Z2+Z3
      DO 50 J=2,ND
```

```
      Z1=ROOTS(J)
      DO 40 I=J-1,1,-1
      IF(REAL(ROOTS(I)).LE.REAL(Z1)) GOTO 50
40    ROOTS(I+1)=ROOTS(I)
      I=0
50    ROOTS(I+1)=Z1
      DO 60 I=1,ND
      IF(AIMAG(ROOTS(I)).NE. 0.) GO TO 60
      IF(REAL(ROOTS(I)).LT. 0.) GO TO 60
      YP1=REAL(ROOTS(I))
      WRITE(6,65) YP1
65    FORMAT(' A positive real root is',F10.3)
60    WRITE(6,70) ROOTS(I)
70    FORMAT(2F14.8)
      END
```

EXAMPLE PROBLEM 3.6

Using the dimensionless momentum Equation 3.20 find the depth conjugate to 6 ft in a trapezoidal channel with a bottom width b = 10 ft, and a side slope m = 1.5 for a flow rate Q = 450 cfs.

Solution

With $Y_2 = 6$ ft, $Y_2' = 0.9$, and Equation 3.20 becomes,

$$Y_1'^4 + 3.4Y_1'^3 + 4.00Y_1'^2 + 1.788Y_1' - 0.707 = 0$$

The roots for this equation, as determined by the above program are: root # 1 (–1.261,0.) real but negative, root # 2 (–1.186,–1.004) complex, root # 3 (–1.186,1.004) complex, and root # 4 (0.2324,0.) real and positive or the root that is sought.

It is informative to solve this problem for different flow rates and depths downstream from the hydraulic jump. The small table below shows a few such solutions. Note that only for the problem in which the flow rate was reduced to 50 cfs, and the downstream depth $Y_2 = 3$ ft did the two complex conjugate roots become two additional real negative roots.

Table giving four solution to $M_1' = M_2'$ in trapezoidal with b = 10 ft, and m = 1.5

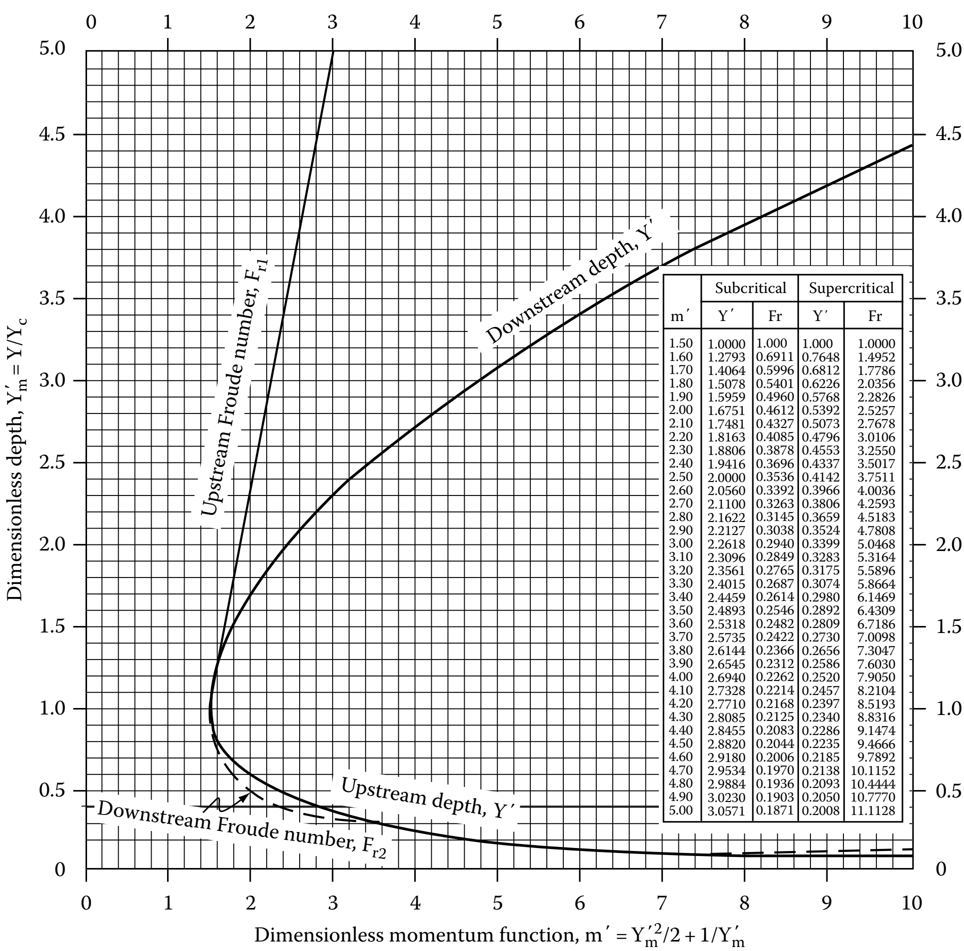
Flow Rate		Root # 1		Root # 2		Root # 3		Root # 4		Conjugate Depth
Q (cfs)	Y_2 (ft)	Real	Imaginary	Real	Imaginary	Real	Imaginary	Real	Imaginary	Y_1 (ft)
600	6	–1.356	0.000	–1.198	–1.094	–1.198	1.094	0.3523	0.000	2.349
450	6	–1.261	0.000	–1.185	–1.004	–1.185	1.004	0.2324	0.000	1.550
100	3	–1.284	0.000	–0.867	–0.232	–0.867	0.232	0.0676	0.000	0.450
50	3	–1.258	0.000	–0.873	0.000	–0.838	0.000	0.0190	0.000	0.127

It will be instructive for you to obtain these same answers by using the modification of the program that solve the fifth degree polynomial.

For a rectangular section the above definition of Y′, when dealing with dimensionless momentum functions, cannot be used because m = 0. However by redefining the dimensionless depth as the depth of flow divided by the critical depth for rectangular channels some interesting characteristics of this special dimensionless momentum function will be seen. If Equation 3.6 is divided by the critical depth squared and this quantity denoted as a dimensionless momentum function per unit width, then the following results:

$$m' = \frac{m}{Y_c^2} = \frac{Y^2}{2Y_c^2} + \frac{q^2}{(Y/Y_c)gY_c^3} = \frac{Y_m'^2}{2} + \frac{1}{Y_m'} \tag{3.21}$$

in which the definition for Y_c comes from setting the Froude number squared for a rectangular channel to unity, or $F_r^2 = q^2/(gY_c^3) = 1$ and the dimensionless depth is defined as $Y_m' = Y/Y_c$. The subscript m used to define this dimensionless depth denotes that it applies for the momentum function per unit width. Below Y_E' will be used for a similar dimensionless depth but applicable to specific energy per unit width. Since m′ is a function of the single variable Y_m' Equation 3.21 applies for all rectangular channels and any flow rate in any of these channels. In other words momentum functions for rectangular channels can be reduced to a single dimensionless function per unit width. Figure 3.3 gives this dimensionless momentum function, and it can be used to graphically solve any problem involving momentum in a rectangular channel. Since the Froude number squared $F_r^2 = q^2/(gY^3) = (Y_c/Y)^3$ for a rectangular channel, it can be noted that either the upstream supercritical, or downstream subcritical Froude number squared can be obtained by taking the reciprocal of the appropriate dimensionless depth cubed, or the Froude number, $F_r = 1/(Y_m')^{1.5}$.



m′	Subcritical		Supercritical	
	Y′	Fr	Y′	Fr
1.50	1.0000	1.000	1.000	1.0000
1.60	1.2793	0.6911	0.7648	1.4952
1.70	1.4064	0.5996	0.6812	1.7786
1.80	1.5078	0.5401	0.6226	2.0356
1.90	1.5959	0.4960	0.5768	2.2826
2.00	1.6751	0.4612	0.5392	2.5257
2.10	1.7481	0.4327	0.5073	2.7678
2.20	1.8163	0.4085	0.4796	3.0106
2.30	1.8806	0.3878	0.4553	3.2550
2.40	1.9416	0.3696	0.4337	3.5017
2.50	2.0000	0.3536	0.4142	3.7511
2.60	2.0560	0.3392	0.3966	4.0036
2.70	2.1100	0.3263	0.3806	4.2593
2.80	2.1622	0.3145	0.3659	4.5183
2.90	2.2127	0.3038	0.3524	4.7808
3.00	2.2618	0.2940	0.3399	5.0468
3.10	2.3096	0.2849	0.3283	5.3164
3.20	2.3561	0.2765	0.3175	5.5896
3.30	2.4015	0.2687	0.3074	5.8664
3.40	2.4459	0.2614	0.2980	6.1469
3.50	2.4893	0.2546	0.2892	6.4309
3.60	2.5318	0.2482	0.2809	6.7186
3.70	2.5735	0.2422	0.2730	7.0098
3.80	2.6144	0.2366	0.2656	7.3047
3.90	2.6545	0.2312	0.2586	7.6030
4.00	2.6940	0.2262	0.2520	7.9050
4.10	2.7328	0.2214	0.2457	8.2104
4.20	2.7710	0.2168	0.2397	8.5193
4.30	2.8085	0.2125	0.2340	8.8316
4.40	2.8455	0.2083	0.2286	9.1474
4.50	2.8820	0.2044	0.2235	9.4666
4.60	2.9180	0.2006	0.2185	9.7892
4.70	2.9534	0.1970	0.2138	10.1152
4.80	2.9884	0.1936	0.2093	10.4444
4.90	3.0230	0.1903	0.2050	10.7770
5.00	3.0571	0.1871	0.2008	11.1128

FIGURE 3.3 Dimensionless momentum diagram for rectangular sections.

Cubic Equation 3.21 can readily be solved for its three roots. First solve θ from

$$\theta = \cos^{-1}\left\{\left(\frac{1.5}{m'}\right)^{1.5}\right\}$$

Then the three roots are given by

$$Y'_{m1} = -1.632993m'^{1/2}\cos\left[\frac{\theta}{3}\right] \quad \text{negative root}$$

$$Y'_{m2} = -1.632993m'^{1/2}\cos\left[\frac{(\theta+2\pi)}{3}\right] \quad \text{subcritical root}$$

$$Y'_{m1} = -1.632993m'^{1/2}\cos\left[\frac{(\theta+4\pi)}{3}\right] \quad \text{supercritical root}$$

The minimum value for m' is 1.5 corresponding to critical conditions when $m' = m'_c$.

Since a single dimensionless momentum function diagram can be developed for rectangular channels, let us attempt to accomplish the same for the specific energy by dividing E, applicable per unit width, by Y_c. The result of this division is

$$\frac{E}{Y_c} = \frac{Y}{Y_c} + \frac{q^2}{2gY^2Y_c} = \frac{Y}{Y_c} + \frac{Y_c^3}{2Y^2Y_c}$$

by defining $E' = E/Y_c$, and $Y'_E = Y/Y_c$, this equation becomes the following dimensionless specific energy for a rectangular channel.

$$E' = Y'_E + \frac{1}{(2Y'^2_E)} \tag{3.22}$$

It needs to be noted that if $Y'_E = 1/Y'_m$ is substituted into Equation 3.22, then Equation 3.21 results. Thus Figure 3.3 can be used to solve specific energy problems in rectangular channels as well as momentum problems. To use Figure 3.3 for specific energy problem the ordinate becomes $1/Y'_E$ and the abscissa is interpreted as E'.

EXAMPLE PROBLEM 3.7

A hydraulic jump occurs immediately downstream from a sluice gate in a rectangular channel. If the depth upstream in this channel is 4.5 ft, and the flow rate per unit width is q = 5 cfs/ft, then what is the depth downstream from the hydraulic jump, and how much energy is dissipated?

Solution

This problem can be solved using the appropriate equations, or by utilizing Figure 3.3. If equations are used, then the energy equation $E_1 = E_2$ is applied across the gate first to determine the depth downstream from the gate. Solution gives $Y_2 = 0.30$ ft. Next the momentum equation $m_1 = m_2$ is applied across the hydraulic jump to find the depth downstream. Therefore, this depth $Y_3 = 2.13$ ft from this last solution. The energy loss due to the hydraulic jump is $h_L = E_1 - E_3 = 4.52 - 2.22 = 2.30$ ft-lb/lb. Using Figure 3.3 requires that the critical depth be computed first from $Y_c = (q^2/g)^{1/3} = 0.919$ ft. Therefore $Y_1' = 4.5/0.919 = 4.896$, or the Y_E' for use with the figure equals 1/4.896 = 0.204. Entering with this value gives $E' = 5.0$ and therefore $1/Y_{E2}' = 3.06$, which is the value for Y_{m2}' on the ordinate of Figure 3.3. Thus $Y_{E2}' = 1/3.06 = 0.327$, and $Y_2 = 0.327(0.919) = 0.300$ ft. Also $Y_{m2}' = 0.327$(or $m_m' = (0.327)^2/2 + 1/0.327 = 3.112$). Then from Figure 3.3 $Y_{m3}' = 2.32$ and therefore $Y_3 = 2.13$ ft.

An alternative to the dimensionless relationships for a hydraulic jump (i.e., $M_1' = M_2'$) is to express the ratio of depths $r = Y_2'/Y_1'$ as a function of the upstream Froude number, F_{r1} (or $r = Y_1'/Y_2'$ as a function of F_{r2}). For a trapezoidal channel the area is $A = bY + mY^2 = (b^2/m)/(Y' + Y'^2)$ and the Froude number squared is $F_r^2 = Q^2T/(gA^3) = m^3Q^2(1+2Y')/\{gb^5(Y'+Y'^2)^3\} = Q'(1+2Y')/(Y'+Y'^2)^3$ in which, as before, $Q' = m^3Q^2/(gb^5)$. Thus Q' is related to the Froude number by $Q' = \{(Y'+Y'^2)^3/(1+2Y')\}F_r^2$. The dimensionless momentum function M' as given by Equation 3.18 can be written across a hydraulic jump in a trapezoidal channel, (i.e., $M_1' = M_2'$) as

$$\frac{Y_1'^2}{2} + \frac{Y_1'^3}{3} + \left\{\frac{(Y_1' + Y_1'^2)^2}{(1+2Y_1')}\right\}F_{r1}^2 = \frac{Y_2'^2}{2} + \frac{Y_2'^3}{3} + \frac{Q'}{(Y_2' + Y_2'^2)}$$

But since Q' on the right side of the equation can also be expressed in terms of the upstream dimensionless depth and the upstream Froude number, the last term in this equation can be written as $(Y_1' + Y_1'^2)^3/\{(Y_2' + Y_2')(1+2Y_2')\}F_{r1}$. Upon dividing the above equation by Y_1^2 and defining the ratio $r = Y_2'/Y_1'$, and rearranging terms the following cubic equation results:

$$\frac{Y_1'(r^3-1)}{3} + \frac{(r^2-1)}{2} + \left\{\frac{\left[(1+Y_1')^3/(r(1+rY_1')) - (1+Y_1')^2\right]}{(1+2Y_1')}\right\}F_{r1}^2 = 0$$

One root of this equation is r = 1 and generally one of the two remaining roots will be positive (the one being sought), and the other is negative (a physical impossible root). Note from this equation that the ratio r of downstream to upstream depth in a trapezoidal channel is a function of the upstream Froude number F_{r1} and the dimensionless upstream depth $Y_1' = mY_1/b$. This relationship is shown in Figure 3.4, and also given in Table 3.1.

If the ratio r is defined as upstream depth divided by the downstream depth, $r = Y_1/Y_2$, and the same procedure followed except that the downstream Froude number squared, F_{r2}^2, is retained, then an identical equation with the subscripts reversed will be obtained. This equation is

$$\frac{Y_2'(r^3-1)}{3} + \frac{(r^2-1)}{2} + \left\{\frac{\left[(1+Y_2')^3/(r(1+rY_2')) - (1+Y_2')^2\right]}{(1+2Y_2')}\right\}F_{r2}^2 = 0$$

For a rectangular channel the arithmetic is simpler. Equating the dimensionless momentum function per unit width $m_1' = m_2'$, dividing by $Y_1'^2$, and rearranging terms gives

$$r^3 - (2F_{r1}^2 + 1)r + 2F_{r1}^2 = 0$$

and upon extracting the known root r = 1 reduces this equation to the quadratic equation,

$$r^2 + r - 2F_{r1}^2 = 0$$

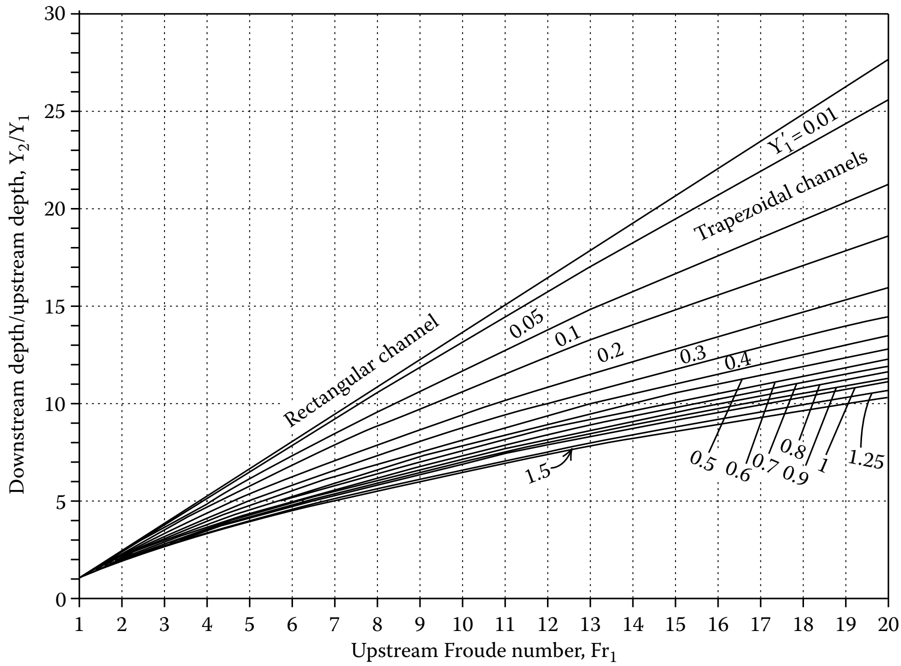


FIGURE 3.4 Relationship between ratio of depths $r = Y_2/Y_1$ with the upstream Froude number $F_{r1} = q/(gY_1^2)^{1/2} = V_1/(gY_1)^{1/2}$ for rectangular channels and with the upstream Froude number $F_{r1} = \{Q^2T_1/(gA_1^3)\}^{1/2}$ and the dimensionless downstream depth $Y_1' = mY_1/b$ for trapezoidal channels.

Applying the quadratic formula to this equation produces Equation 3.10, as would be expected, since $r = Y_2'/Y_1' = Y_2/Y_1$. Figure 3.4 and Table 3.1 contains the result for a rectangular channel, as well as for trapezoidal channels with different values for the dimensionless upstream depth Y_1'.

Figure 3.5 and Table 3.2 are similar to Figure 3.4 and Table 3.1, except the ratio is now for the upstream depth divided by the downstream depth, or $r = Y_1/Y_2 = Y_1'/Y_2'$, and the downstream Froude number F_{r2} is given in the first column.

3.6 CELERITY OF SMALL AMPLITUDE GRAVITY WAVES

An application of the momentum principle is the determination of the speed of movement, or celerity, c, of a small amplitude gravity wave on the surface of a body of water. Should one throw a stone in a still pool of water, the entry of the stone through the water surface causes the depth to increase by a very small amount, and an increase in depth propogates radially outward with a celerity c. Should a stick be held in a stream of flowing water its existence in the flow causes the depth to rise, and this effect is propogated outward also. Should the velocity of flow be larger than c, e.g., the flow is supercritical flow, then the effect would not proprogate upstream but be seen as a wedge-shaped wave downstream from the stick. For larger velocities the angle of the wedge becomes smaller. The Froude number is the ratio of the average velocity of the flow in a channel divided by the celerity of a small amplitude gravity wave. If the amplitude, or height of the wave, is more than a very small fraction of the depth, then the speed of this wave is not defined by the variable c.

Consider a small amplitude gravity wave as shown in the sketch below in which the increase in depth at the wave is dY, and the depth in this channel, which we will assume is not flowing, to

TABLE 3.1
Ratio of Downstream to Upstream Depth, Y_2/Y_1, across a Hydraulic Jump in Rectangular and Trapezoidal Channels

		Dimensionless Upstream Depth $Y_1' = mY_1/b$													
F_r	Rect.	0.01	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	1.25	1.5
1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
2	2.37	2.36	2.32	2.27	2.20	2.14	2.09	2.05	2.02	1.99	1.97	1.95	1.94	1.90	1.88
3	3.77	3.74	3.61	3.49	3.29	3.15	3.05	2.96	2.90	2.84	2.80	2.76	2.73	2.66	2.61
4	5.18	5.11	4.87	4.63	4.30	4.08	3.91	3.78	3.69	3.60	3.54	3.48	3.43	3.34	3.27
5	6.59	6.47	6.08	5.73	5.25	4.94	4.71	4.54	4.41	4.30	4.22	4.14	4.08	3.96	3.87
6	8.00	7.82	7.26	6.78	6.15	5.75	5.46	5.25	5.09	4.96	4.85	4.76	4.68	4.53	4.42
7	9.41	9.16	8.41	7.78	7.00	6.51	6.18	5.93	5.73	5.58	5.45	5.35	5.26	5.08	4.95
8	10.82	10.50	9.53	8.76	7.82	7.25	6.86	6.57	6.35	6.17	6.02	5.90	5.80	5.60	5.45
9	12.24	11.82	10.63	9.70	8.61	7.96	7.51	7.19	6.93	6.73	6.57	6.44	6.32	6.10	5.94
10	13.65	13.13	11.70	10.62	9.37	8.64	8.14	7.78	7.50	7.28	7.10	6.95	6.82	6.58	6.40
11	15.06	14.43	12.74	11.51	10.11	9.30	8.75	8.35	8.05	7.81	7.61	7.45	7.31	7.04	6.85
12	16.48	15.72	13.77	12.38	10.84	9.94	9.34	8.91	8.58	8.32	8.11	7.93	7.78	7.49	7.28
13	17.89	17.00	14.78	13.23	11.54	10.57	9.92	9.45	9.10	8.82	8.59	8.40	8.24	7.93	7.70
14	19.31	18.28	15.77	14.07	12.22	11.18	10.48	9.98	9.60	9.30	9.06	8.85	8.68	8.35	8.11
15	20.72	19.54	16.74	14.88	12.89	11.77	11.03	10.50	10.09	9.77	9.52	9.30	9.12	8.77	8.52
16	22.13	20.80	17.70	15.68	13.55	12.36	11.57	11.00	10.58	10.24	9.96	9.74	9.54	9.18	8.91
17	23.55	22.04	18.64	16.47	14.19	12.93	12.10	11.50	11.05	10.69	10.40	10.16	9.96	9.57	9.29
18	24.96	23.28	19.57	17.24	14.82	13.49	12.61	11.98	11.51	11.13	10.83	10.58	10.37	9.96	9.67
19	26.37	24.51	20.49	18.00	15.44	14.04	13.12	12.46	11.96	11.57	11.25	10.99	10.77	10.35	10.04
20	27.79	25.74	21.39	18.75	16.05	14.58	13.62	12.93	12.41	12.00	11.67	11.40	11.17	10.72	10.40

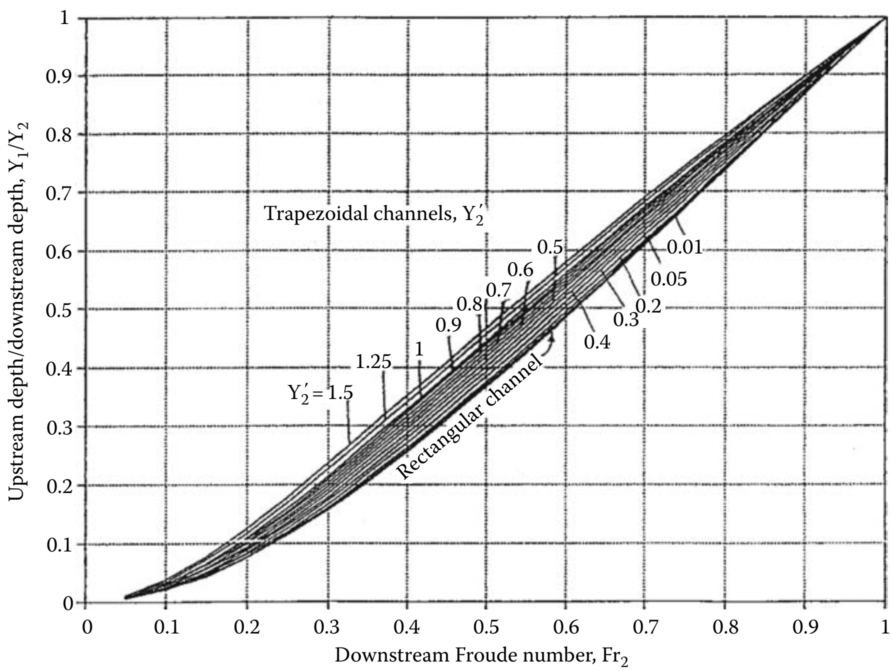
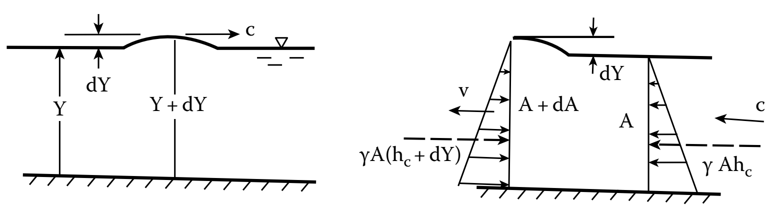


FIGURE 3.5 Relationship between ratio of depths $r = Y_1/Y_2$ with the downstream Froude number $F_{r2} = q/(gY_2^3)^{1/2} = V_2/(gY_2)^{1/2}$ for rectangular channels and with the downstream Froude number $F_{r2} = \{Q^2T_2/(gA_2^3)\}^{1/2}$ and the dimensionless downstream depth $Y_2' = mY_2/b$ for trapezoidal channels.

simplify the analysis but will not affect the magnitude determine for c, is Y. The channel is prismatic and has a cross-sectional area, A. To analyze this wave consider an observer moving with the celerity of the wave so that from his point of view a steady state flow exists. A stationary observer would see unsteady flow in the channel as the wave passed.



To the moving observer the continuity equation is,

$$v(A + dA) = cA$$

in which v is the average velocity this moving observer sees at the section with the incrementally larger depth Y + dY and dA is the incremental increase in this area due to the rise, dY, of the liquid surface. Rewriting this equation so that v appears by itself on one side of the equal sign gives

$$v = c\frac{A}{A + dA} \quad \text{(continuity equation as seen by a moving observer)}$$

TABLE 3.2
Ratio of Upstream to Downstream Depth, Y_1/Y_2, across a Hydraulic Jump in Rectangular and Trapezoidal Channels

		Dimensionless Downstream Depth $Y_2' = mY_2/b$													
F_{r2}	Rect.	0.01	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	1.25	1.5
.05	.0050	.0050	.0051	.0052	.0054	.0057	.0060	.0063	.0066	.0069	.0072	.007	.0079	.0087	.0096
.10	.0196	.0197	.0200	.0204	.0213	.0223	.0235	.0246	.0258	.0270	.0282	.029	.0306	.0336	.0365
.15	.0431	.0433	.0439	.0448	.0467	.0489	.0512	.0536	.0559	.0583	.0607	.0631	.0654	.0711	.0764
.20	.0745	.0747	.0757	.0771	.0804	.0839	.0875	.0913	.0950	.0987	.1023	.1058	.1092	.1174	.1249
.25	.1124	.1127	.1142	.1162	.1208	.1257	.1307	.1358	.1408	.1456	.1503	.1549	.1593	.1694	.1785
.30	.1557	.1562	.1581	.1608	.1667	.1729	.1792	.1854	.1915	.1973	.2029	.2082	.2133	.2249	.2349
.35	.2036	.2041	.2065	.2097	.2168	.2241	.2315	.2387	.2456	.2522	.2584	.2642	.2697	.2821	.2927
.40	.2550	.2556	.2584	.2621	.2702	.2785	.2867	.2945	.3020	.3090	.3156	.3218	.3275	.3402	.3509
.45	.3093	.3100	.3131	.3173	.3261	.3351	.3438	.3521	.3599	.3671	.3739	.3801	.3859	.3985	.4090
.50	.3660	.3668	.3701	.3746	.3840	.3933	.4023	.4108	.4186	.4259	.4326	.4387	.4444	.4566	.4667
.55	.4247	.4255	.4290	.4336	.4432	.4527	.4617	.4701	.4778	.4849	.4914	.4973	.5026	.5142	.5237
.60	.4849	.4857	.4893	.4940	.5036	.5129	.5217	.5298	.5372	.5439	.5499	.5555	.5605	.5713	.5799
.65	.5464	.5473	.5508	.5553	.5647	.5736	.5819	.5895	.5964	.6026	.6082	.6132	.6178	.6275	.6353
.70	.6091	.6099	.6132	.6176	.6263	.6346	.6423	.6492	.6554	.6609	.6659	.6704	.6745	.6831	.6899
.75	.6726	.6733	.6764	.6804	.6883	.6958	.7025	.7086	.7140	.7188	.7232	.7270	.7305	.7378	.7436
.80	.7369	.7376	.7403	.7437	.7506	.7569	.7626	.7677	.7722	.7762	.7798	.7830	.7858	.7918	.7965
.85	.8019	.8025	.8047	.8075	.8129	.8180	.8225	.8264	.8300	.8330	.8358	.8382	.8404	.8450	.8485
.90	.8675	.8679	.8695	.8715	.8753	.8789	.8820	.8848	.8872	.8893	.8912	.8928	.8943	.8974	.8998
.95	.9335	.9337	.9346	.9357	.9377	.9396	.9412	.9426	.9439	.9449	.9459	.9467	.9475	.9490	.9503
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Since this observer sees a steady-state flow, the momentum at both sections must be equal or $M_1 = M_2$, or,

$$A(h_c + dY) + \frac{v^2(A + dA)}{g} = Ah_c + \frac{c^2 A}{g} \quad \text{(momentum equation as seen by a moving observer)}$$

In the above equation, it should be noted that since the water surface has risen an amount dY at the wave that a first-order approximation (i.e., ignoring products of two differential amounts) is obtained by adding dY to h_c. If v, as defined by the continuity equation, is substituted into this momentum equation, the result rearranged, the second order term dY/dA deleted, and dA/dY replaced by T (the top width) the following equation is obtained:

$$c = \sqrt{\frac{gA}{T}} = \sqrt{gY_h} \tag{3.23}$$

in which $Y_h = A/T$ is called the hydraulic depth. For a rectangular channel the hydraulic depth is the same as the depth Y, but for other channels this is not the case. For a rectangular channel the celerity of a small amplitude gravity wave is given by

$$c = \sqrt{gY} \tag{3.24}$$

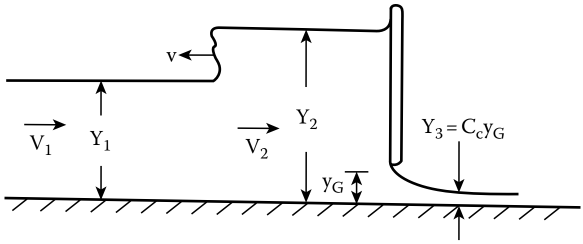
3.7 CONSTANT HEIGHT WAVES

General unsteady flow problems in channels is the subject of Chapter 6 (and the numerical solution to such problems in Chapter 7), but certain special situations occur that allow an unsteady problem to be solved by having the observer move such that the flow is steady to him. For these types of problems, the momentum principle often provides one of the equations needed for a solution. A class of these problems involves the rapid operation of gates in a channel that contains uniform, or near uniform flow, prior to the change in the gate opening. For example, assume that a gate is suddenly closed down further. Since this incremental closure will reduce the flow rate past the gate, a surge will form upstream from the gate across which the prior depth changes to the new depth established by energy for the new gate setting. If the upstream flow is uniform, then the speed, v, at which this wave moves will be constant, and to an observer moving with this wave a steady-state flow will exit. In fact, he will see a hydraulic jump in front of him, with an incoming velocity equal to the sum of the previous velocity and his velocity in the opposite direction. This combined velocity produces a supercritical flow in his eyes. Downstream he sees a subcritical flow. In other words from his view point the momentum equation $M_1 = M_2$ is valid; however, in place of the upstream flow rate Q_1 the equation would contain $(V_1 + v)A_1$ and the downstream flow rate $Q_2 = (V_2 + v)A_2$, in which V_1 is the original velocity upstream from the surge, and V_2 is the new velocity in the channel downstream from the surge. Making these substitutions the momentum equation across this moving surge becomes

$$(Ah_c)_1 + \frac{(v + V_1)^2 A_1}{g} = (Ah_c)_2 + \frac{(v + V_2)^2 A_2}{g} \tag{3.25}$$

Likewise the continuity equation to this moving observer is

$$(V_1 + v)A_1 = (V_2 + v)A_2 \tag{3.26}$$

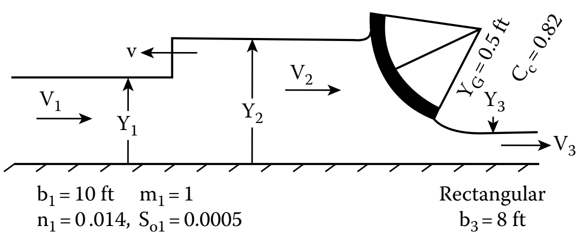


The unknown variables of a problem of this type are v, Y_2, and V_2. The third equation that is required comes from energy at the gate from the viewpoint of a stationary observer.

As the gate closes rapidly, it will cause a reduction in depth downstream from the gate. If an observer moves with this resulting surge again a steady-state flow will be seen if the original downstream depth is constant. The above Equations 3.25 and 3.26 are valid from his view point if the + signs within the ( ) are changed to − signs. The reason for the sign change is that the observer is now moving in the same direction as the flow.

EXAMPLE PROBLEM 3.8

A gate in a channel is suddenly closed from an entirely open position to a distance of 0.5 ft from the channel bottom. Its contraction coefficient is $C_c = 0.82$. Upstream from the gate the channel is trapezoidal with $b_1 = 10$ ft, $m_1 = 1$, $n_1 = 0.014$, and $S_{o1} = 0.0005$. Downstream from the gate the width of the channel is 8 ft, and it is rectangular here. Prior to the gate closure uniform flow existed throughout the trapezoidal channel. Determine the flow rate past the gate after it is partly closed, the depth on the upstream side of the gate and the speed at which the surge moves upstream if prior to closure the flow rate was Q = 250 cfs.



Solution

The assumption made in solving this problem is that to an observer moving with the speed of the surge a steady state flow occurs, and he sees a hydraulic jump in front of him. First, uniform flow is solved in the trapezoidal channel with the results: $V_1 = 4.504$ fps, and $Y_1 = 3.973$ ft. Now the unknowns to this problem are: v, Y_2, and V_2. The three available equations are:

From moving observer

$$F_1 = (V_1 + v)A_1 - (V_2 + v)A_2 = 0 \quad \text{(continuity)}$$

$$F_2 = h_{c1}A_1 - h_{c2}A_2 + \left(\frac{Q^2}{g}\right)\left(\frac{1}{A_1} - \frac{1}{A_2}\right) = 0 \quad \text{(momentum)}$$

Stationary observer at gate

$$F_3 = Y_2 - Y_3 + \frac{V_2^2}{(2g)} - \frac{(V_2A_2)^2}{(2gA_3^2)} = 0 \quad \text{(energy)}$$

The solution to these three equation produces the following results: $Y_2 = 5.202'$, $V_2 = 0.729$ fps, v = 8.101 fps.

The listing of the following PASCAL and FORTRAN programs illustrate how problems of this type can be solved utilizing the Newton method.

PASCAL computer program to solve wave problems

```
Program Gate_Wave;
Var Y3,vw,b,m,A3S,A1:Real;
  Y,V: array [1..2] of real; EQ:array [1..3] of real;
  D: array[1..3] of array[1..3] of real;
Const g=32.2;g2=64.4;
Function A(K:integer):real;
Begin A:=(b+m*Y[K])*Y[K] End;
Function MOM(K:integer):real;
Begin
  MOM:=(b/2+m*Y[K]/3)*sqr(Y[K])+sqr((V[K]+vw))*A(K)/g; End;
Function F(K:integer): real;
Begin Case K of
  1:F:=(V[1]+vw)*A1-(V[2]+vw)*A(2);
  2:F:=MOM(1)-MOM(2);
  3:F:=Y[2]-Y3+(sqr(V[2])-sqr(V[2]*A(2))/A3S)/g2 end; End;
Var I,J,N:integer; FAC,Cc,yG,b3,m3,D1,D2,D3:real;
BEGIN
 Writeln('Give: b,m,V1,Y1,yG,Cc,b3,m3 & est. for vw,Y2,V2');
 Readln(b,m,V[1],Y[1],yG,Cc,b3,m3,vw,Y[2],V[2]);
 Y3:=Cc*yG; A3S:=sqr((b3+m3*Y3)*Y3); A1:=A(1);
 repeat
  For I:=1 to 3 do Begin EQ[I]:=F(I);
   For J:=1 to 3 do begin
   Case J of 1:vw:=vw-0.001; 2:Y[2]:=Y[2]-0.001;
     3:V[2]:=V[2]-0.001;end;
   D[I,J]:=(EQ[I]-F(I))/0.001;
   Case J of 1:vw:=vw+0.001; 2:Y[2]:=Y[2]+0.001;
     3:V[2]:=V[2]+0.001;end;
  end; End;
{Solves equations}
 For N:=1 to 2 do Begin
  For I:=N+1 to 3 do begin FAC:=D[I,N]/D[N,N];
   For J:=N+1 to 3 do D[I,J]:=D[I,J]-FAC*D[N,J];
     EQ[I]:=EQ[I]-FAC*EQ[N];end;
 End;
 D1:=EQ[3]/D[3,3]; V[2]:=V[2]-D1;
 D2:=(EQ[2]-D1*D[2,3])/D[2,2]; Y[2]:=Y[2]-D2;
 D3:=(EQ[1]-D2*D[1,2]-D1*D[1,3])/D[1,1]; vw:=vw-D3;
 until (abs(D1)+abs(D2)+abs(D3))<0.0001;
 Writeln('Wave speed=',vw:10:3,' Depth=',Y[2]:10:3,
   ' Velocity=',V[2]:10:3);
END.
```

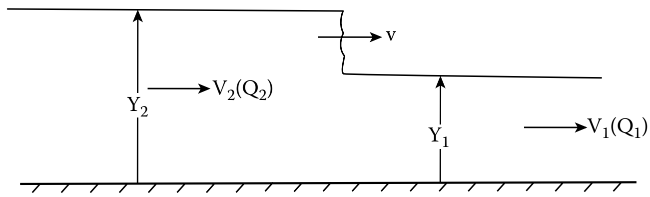
FORTRAN computer program for wave prob., GWAVE.FOR

```
      LOGICAL LDV
      COMMON b,Fm,Y(2),V(2),EQ(3),D(3,3),vw,A1,Y3,A3S,g2,g,FMOM1
      g=32.2
      g2=64.4
      WRITE(6,*)' Give: b,m,V1,Y1,yG,Cc,b3,m3, & est. vw,Y2,V2'
      READ(5,*) b,Fm,V(1),Y(1),yG,Cc,b3,Fm3,vw,Y(2),V(2)
      Y3=Cc*yG
      A3S=((b3+Fm3*Y3)*Y3)**2
      A1=A(1)
```

```
      FMOM1=FOM(1)
10    DO 30 I=1,3
      EQ(I)=F(I)
      DO 30 J=1,3
      DV=-.001
      LDV=.FALSE.
21    IF(J-2) 22,23,24
22    vw=vw+DV
      GO TO 25
23    Y(2)=Y(2)+DV
      GO TO 25
24    V(2)=V(2)+DV
25    IF(LDV) GO TO 30
      D(I,J)=(EQ(I)-F(I))/.001
      DV=.001
      LDV=.TRUE.
      GO TO 21
30    CONTINUE
      DO 50 N=1,2
      DO 50 I=N+1,3
      FAC=D(I,N)/D(N,N)
      DO 40 J=N+1,3
40    D(I,J)=D(I,J)-FAC*D(N,J)
50    EQ(I)=EQ(I)-FAC*EQ(N)
      D1=EQ(3)/D(3,3)
      V(2)=V(2)-D1
      D2=(EQ(2)-D1*D(2,3))/D(2,2)
      Y(2)=Y(2)-D2
      D3=(EQ(1)-D2*D(1,2)-D1*D(1,3))/D(1,1)
      vw=vw-D3
      IF(ABS(D1)+ABS(D2)+ABS(D3).GT..0001) GO TO 10
      WRITE(6,100) vw,Y(2),V(2)
100   FORMAT(' Wave speed=',F10.3,' Depth=',F10.3,
     &' Velocity=',F10.3)
      STOP
      END
      FUNCTION F(K)
      COMMON b,Fm,Y(2),V(2),EQ(3),D(3,3),vw,A1,Y3,A3S,g2,g,FMOM1
      GO TO (1,2,3),K
      1 F=(V(1)+vw)*A1-(V(2)+vw)*A(2)
      RETURN
2     F=FMOM1-FOM(2)
      RETURN
3     F=Y(2)-Y3+(V(2)**2-(V(2)*A(2))**2/A3S)/g2
      RETURN
      END
      FUNCTION A(K)
      COMMON b,Fm,Y(2),V(2),EQ(3),D(3,3),vw,A1,Y3,A3S,g2,g,FMOM1
      A=(b+Fm*Y(K))*Y(K)
      RETURN
      END
      FUNCTION FOM(K)
      COMMON b,Fm,Y(2),V(2),EQ(3),D(3,3),vw,A1,Y3,A3S,g2,g,FMOM1
      FOM=(.5*b+Fm*Y(K)/3.)*Y(K)**2+(V(K)+vw)**2*A(K)/g
      RETURN
      END
```

The previous application creates a constant depth upstream moving wave by instantly closing a gate some incremental amount. This will be referred to as a downstream controlled wave (DCW), because it is created by a downstream control. By instantly increasing the flow rate, or depth, at the upstream end of a channel a constant height wave is created that moves downstream. This will be referred to as an upstream-controlled wave (UCW), because it is created by an upstream control. In both DCW's and UCW's the depths, velocities and flow rates, both upstream and downstream from the waves are constant so that an observer moving with the velocity v of the wave sees a steady-state hydraulic jump. We will now examine an UCW, and thereafter manipulate the resulting equations into various other forms. The equivalent DCW equations will also be given. You should verify these equations.

Assume uniform flow exists in a channel with depth Y_1, and velocity V_1 (or flow rate Q_1) constant, when suddenly at its upstream end the velocity (or flow rate) is suddenly increased to V_2 (or Q_2). Then a constant height wave will result that will move down the channel with a velocity v, and the depth upstream from this wave will be constant at Y_2 as shown on the sketch below.



From the viewpoint of an observer traveling downstream with the wave velocity v, the continuity equation is

$$(v - V_1)A_1 = (v - V_2)A_2 \quad \text{(Cont. Equation)} \tag{3.27}$$

and equating the momentum function upstream to that downstream ($M_1 = M_2$) gives

$$A_1h_{c1} + \frac{(v - V_1)^2 A_1}{g} = A_2h_{c2} + \frac{(v - V_2)^2 A_2}{g} \quad \text{(Mom. Equation)} \tag{3.28}$$

Notice in both Equations 3.27 and 3.28 that the velocity seen by the moving observer is v–V, rather than v + V as in the DCW. Equations 3.27 and 3.28 allow for two variables to be solved. Typically Y_1 and $V_1(Q_1)$ are known, so if a new velocity $V_2(Q_2)$ is given then the depth Y_2 and v are solved. However, another variable might be specified, such as v, and then V_2 might be solved. Depending upon the type of upstream control and what is known, a third equation might be added to Equations 3.27 and 3.28, as was done with a gate having its opening changed in the DCW. For an UCW the gate's opening would be increased, however, rather than decreased.

By examining Equations 3.27 and 3.28 further we can learn more about constant height waves. First let us eliminate V_2 by substituting Equation 3.27 into Equation 3.28 to give

$$g(A_2h_{c2} - A_1h_{c1}) = (v - V_1)^2 A_1\left(1 - \frac{A_1}{A_2}\right) = (v - V_1)^2\left(\frac{A_1}{A_2}\right)(A_2 - A_1)$$

or UCW

$$v = V_1 + \sqrt{\frac{gA_2(A_2h_{c2} - A_1h_{c1})}{A_1(A_2 - A_1)}} = V_1 + \sqrt{\frac{g(h_{c2}A_2/A_1 - h_{c1})}{1 - A_1/A_2}} \tag{3.29}$$

DCW

$$v = \sqrt{\frac{gA_2(A_2h_{c2} - A_1h_{c1})}{A_1(A_2 - A_1)}} - V_1 = \sqrt{\frac{g(h_{c2}A_2/A_1 - h_{c1})}{1 - A_1/A_2}} - V_1 \tag{3.29a}$$

Of course if one also knows what V_2 is by having solved it then the wave velocity can be solved from only the continuity equation, or

UCW

$$v = \frac{V_2A_2 - V_1A_1}{A_2 - A_1} = \frac{Q_2 - Q_1}{A_2 - A_1} \tag{3.30}$$

DCW

$$v = \frac{V_1A_1 - V_2A_2}{A_2 - A_1} = \frac{Q_1 - Q_2}{A_2 - A_1} \tag{3.30a}$$

For a rectangular channel Equations 3.27 and 3.28 simplify to

UCW

$$(v - V_1)Y_1 = (v - V_2)Y_2 \quad \text{(Rect. Cont. Equation)} \tag{3.31}$$

DCW

$$(v + V_1)Y_1 = (v + V_2)Y_2 \quad \text{(Rect. Cont. Equation)} \tag{3.31a}$$

and equating the momentum per unit width upstream to that downstream ($m_1 = m_2$ by a moving observer) gives

UCW

$$\frac{Y_1^2}{2} + \frac{(v - V_1)^2 Y_1}{g} = \frac{Y_2^2}{2} + \frac{(v - V_2)^2 Y_2}{g} \quad \text{(Rect. Mom. Equation)} \tag{3.32}$$

DCW

$$\frac{Y_1^2}{2} + \frac{(v + V_1)^2 Y_1}{g} = \frac{Y_2^2}{2} + \frac{(v + V_2)^2 Y_2}{g} \quad \text{(Rect. Mom. Equation)} \tag{3.32a}$$

Equation 3.29 simplifies to the following equations for a rectangular channel:

UCW

$$v = V_1 + \sqrt{\tfrac{1}{2}gY_1}\sqrt{\frac{Y_2^2/Y_1^2 - 1}{1 - Y_1/Y_2}} = V_1 + c_1\sqrt{\frac{Y_2^2/Y_1^2 - 1}{2 - 2Y_1/Y_2}} \tag{3.33}$$

or manipulated to another form

$$v = V_1 + (gY_1/2)^{1/2}\left(\frac{Y_2^2}{Y_1^2} + \frac{Y_2}{Y_1}\right)^{1/2} = V_1 + \frac{c_1}{\sqrt{2}}\left(\frac{Y_2^2}{Y_1^2} + \frac{Y_2}{Y_1}\right)^{1/2} = V_1 + \left(\frac{gY_2}{2Y_1}(Y_2 + Y_1)\right)^{1/2} \tag{3.33a}$$

DCW

$$v = \sqrt{\tfrac{1}{2} g Y_1} \sqrt{\frac{Y_2^2/Y_1^2 - 1}{1 - Y_1/Y_2}} - V_1 = c_1 \sqrt{\frac{Y_2^2/Y_1^2 - 1}{2 - 2Y_1/Y_2}} - V_1 \tag{3.33b}$$

or manipulated to another form

$$v = \left(\frac{gY_1}{2}\right)^{1/2} \left(\frac{Y_2^2}{Y_1^2} + \frac{Y_2}{Y_1}\right)^{1/2} - V_1 = \frac{c_1}{\sqrt{2}} \left(\frac{Y_2^2}{Y_1^2} + \frac{Y_2}{Y_1}\right)^{1/2} - V_1 = \left(\frac{gY_2}{2Y_1}\left(Y_2 + Y_1\right)\right)^{1/2} - V_1 \tag{3.33c}$$

or just using the Continuity Equation,
UCW

$$v = \frac{V_2 Y_2 - V_1 Y_1}{Y_2 - Y_1} = \frac{q_2 - q_1}{Y_2 - Y_1} \tag{3.34}$$

DCW

$$v = \frac{V_1 Y_1 - V_2 Y_2}{Y_2 - Y_1} = \frac{q_1 - q_2}{Y_2 - Y_1} \tag{3.34a}$$

As the height of the wave becomes very small the constant height wave becomes a small amplitude wave, and the equations should reflect this. For a small amplitude wave, Y_2 approaches the depth Y_1, or $Y_2/Y_1 \rightarrow 1$. Let the velocity $V_1 = 0$ and note that if $Y_2/Y_1 = 1$ in Equations 3.33a and 3.33c that $v = (gY_1)^{1/2} = (gY)^{1/2} = c$, e.g., indicates that the wave velocity equals the celerity of a small amplitude gravity wave as given by Equation 3.24 for a rectangular channel. Notice also that if the wave is moving upstream due to downstream control that the wave velocity is $v = c - V_1$, or the speed with which a small change in depth moves upstream equals the celerity of a small amplitude gravity wave minus the channel velocity that carries it downstream. Likewise a small disturbance in depth will move downstream with a velocity $v = c + V_1$ according to Equation 3.33a as $Y_2/Y_1 = 1$. The same results can be obtained from Equations 3.33 and 3.33b, but since they produce the indeterminate form 0/0 L'Hospital's rule must be applied.

We might define the celerity of a standing wave as the speed it moves in a channel minus the velocity in the channel, e.g., the speed it moves as seen by an observer moving with the velocity in the channel. Thus for a UCW this celerity is $c = v + V_1$ and for a DCW this celerity is $c = v + V_1$. From Equations 29 see this celerity is given by

$$v = \sqrt{\frac{gA_2(A_2 h_{c2} - A_1 h_{c1})}{A_1(A_2 - A_1)}} = \sqrt{\frac{g(h_{c2} A_2/A_1 - h_{c1})}{1 - A_1/A_2}} \tag{3.35}$$

and for a rectangular channel from Equations 3.33,

$$v = \left(\frac{gY_1}{2}\right)^{1/2} \left(\frac{Y_2^2}{Y_1^2} + \frac{Y_2}{Y_1}\right)^{1/2} = \frac{c_1}{\sqrt{2}} \left(\frac{Y_2^2}{Y_1^2} + \frac{Y_2}{Y_1}\right)^{1/2} = \left(\frac{gY_2}{2Y_1}\left(Y_2 + Y_1\right)\right)^{1/2} \tag{3.35a}$$

For problems in which a third equation is not available so that only the continuity and momentum equations are used to solve for v and Y_2, it is convenient to eliminate v from these two equations and solve Y_2 from one implicit equation. Thereafter v can be obtained from Equation 3.30 (or 3.30a), or if the channel is rectangular, from Equation 3.34 (or 3.34a). Substituting v from the continuity equation into the momentum equation gives

UCW

$$\frac{g(h_{c2}A_2/A_1 - h_{c1})}{1 - A_1/A_2} = \left(\frac{V_2A_2 - V_1A_1}{A_2 - A_1} - V_1\right)^2 = \left(\frac{Q_2 - Q_1}{A_2 - A_1} - V_1\right)^2 \tag{3.36}$$

DCW

$$\frac{g(h_{c2}A_2/A_1 - h_{c1})}{1 - A_1/A_2} = \left(\frac{V_2A_2 - V_1A_1}{A_2 - A_1} + V_1\right)^2 = \left(\frac{Q_2 - Q_1}{A_2 - A_1} + V_1\right)^2 \tag{3.36a}$$

For a rectangular channel Equations 3.36 simplify to

UCW

$$\frac{gY_2}{2Y_1}(Y_2 + Y_1) = \left(\frac{V_2Y_2 - V_1Y_1}{Y_2 - Y_1} - V_1\right)^2 = \left(\frac{q_2 - q_1}{Y_2 - Y_1} - V_1\right)^2 \tag{3.37}$$

An alternative to this equation is

$$\frac{g(Y_2 - Y_1)}{2Y_1Y_2}(Y_2^2 - Y_1^2) = (V_1 - V_2)^2 \tag{3.37a}$$

DCW

$$\frac{gY_2}{2Y_1}(Y_2 + Y_1) = \left(\frac{V_2Y_2 - V_1Y_1}{Y_2 - Y_1} + V_1\right)^2 = \left(\frac{q_2 - q_1}{Y_2 - Y_1} + V_1\right)^2 \tag{3.37b}$$

With the same alternative Equation 3.37a.

EXAMPLE PROBLEM 3.9

Obtain a series of solutions that provide the depth and velocity and wave speed that will occur in a rectangular channel if the flow rate is instantly increased at its beginning from uniform conditions of $q = 2\,m^2/s$ and $Y_1 = Y_o = 2\,m$.

Solution

$V_1 = q/Y_o = 2/2 = 1\,m/s$. For each solution in this series Equation 3.37 is solved first for Y_2. With Y_2 known, Equation 3.34 is used to solve the wave speed v. $V_2 = q/Y_2$ in which q is the new specified flow rate per unit width. A TK-Solver model is given below to solve this problem. In this model Equation 3.37 (using Y22 for Y_2 and V22 for V_2) and Equation 3.37a are solved to verify that identical results are produced by these two equations. A Fortran and C program that solves this problem follows the TK-Solver model. This program uses the Newton method with a numerical evaluation of the derivative of Equation 3.37a.

```
TK-Solver Model
================VARIABLE SHEET=================
St  Input----    Name----  Output----  Unit----
    1            V1
LG  1.3793999    V2
    9.81         g
    2            Y1
LG  2.1748587    Y2
LG  2.1748587    Y22
L   3            q2
LG  1.3793999    V22
LG  6            v
================RULE SHEET=================
S Rule---------------------------------------
  (V1-V2)∧2=g*(Y1-Y2)*(Y1∧2-Y2∧2)/(2.*Y1*Y2)
  ((V22*Y22-V1*Y1)/(Y22-Y1)-V1)∧2=.5*g*Y22/Y1*(Y22+Y1)
  V2=q2/Y2
  V22=q2/Y22
  v=(V2*Y2-V1*Y1)/(Y2-Y1)
=====================TABLE: wave==================
Title:
Element  q2-   V2--------   V22-------   Y2--------   Y22-------   v---------
1        2.01  1.00407586   10.6695871   2.00184077   .188385922   5.4325044
2        2.2   1.0803217    1.0803217    2.03643044   2.03643044   5.48991413
3        2.4   1.15825367   1.27939865   2.07208496   1.87588129   5.54900801
.        .     .            .            .            .            .
38       9.4   3.09385488   3.09385488   3.03828084   3.03828084   7.12716607
39       9.6   3.13633146   3.13633146   3.06090097   3.06090097   7.16372235
40       9.8   3.17837366   3.17837366   3.08333791   3.08333791   7.19996959
41       10    3.21999374   3.21999374   3.1055961    3.1055961    7.23591551
```

Fortran Program WAVE.FOR

```
      WRITE(*,*) 'Give: Y1,V1,g,q2,N'
      READ(*,*) Y1,V1,G,q2,N
      q1=V1*Y1
      dq=(q2-q1)/FLOAT(N-1)
      Y2=Y1+.05
      v=SQRT(G*Y2)+V1
      DO 10 I=1,N
      q=q1+dq*float(I-1)
      IF(I.EQ.1) q=q+.01
      M=0
1     F=(V1-q/Y2)**2-g*(Y1-Y2)*(Y1**2-Y2**2)/(2.*Y1*Y2)
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 2
      F1=F
      Y22=Y2
      Y2=1.005*Y2
      GO TO 1
2     DIF=(Y2-Y22)*F1/(F-F1)
      Y2=Y22-DIF
```

```
      IF(ABS(DIF).GT.1.E-5.AND.M.LT.30)GO TO 1
      IF(M.EQ.30)WRITE(*,*)'Failed to conv.',I,DIF
      WRITE(3,100) q,q/Y2,Y2,(q-q1)/(Y2-Y1)
10    Y2=1.05*Y2
100   FORMAT(F5.2,3F8.3)
      END
```

C-Program WAVE.C

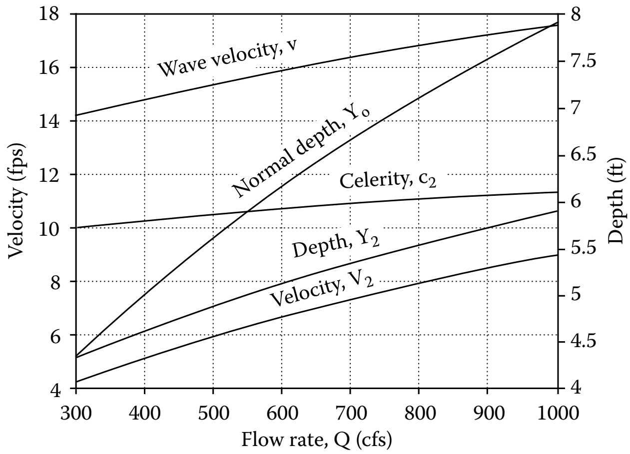
```
#include <math.h>
#include <stdlib.h>
#include <stdio.h>
void main(void){int n,i,m;
FILE *filo; char fnam[20];
float y1,v1,v,g,q2,q1,dq,y2,q,f,f1,dif,hc1,hc2a,y22,a;
printf("Give: Y1,V1,g,Q2,N\n");
scanf("%f %f %f %f %d",&y1,&v1,&g,&q2,&n);
printf("Give output filename\n");scanf("%s",fnam);
if((filo=fopen(fnam,"w"))==NULL)
{printf("Cannot open output file\n");exit(0);}
q1=v1*y1;dq=(q2-q1)/(float)(n-1); y2=y1+.05;
 for (i=1;i<n;i++){q=q1+dq*(float)i; m=0;
   L1:f=pow(v1-q/y2,2.)-g*(y1-y2)*(y1*y1-y2*y2)/(2.*y1*y2);
 if((++m)%2){f1=f;y22=y2;y2*=1.005;goto L1;}
   dif=(y2-y22)*f1/(f-f1);y2=y22-dif;
 if((fabs(dif)>1.e-5)&&(m<20)) goto L1;
 if(m==30) printf("Failed to converge %d %f/n",i+1,dif);
  fprintf(filo,"%5.2f %7.3f %7.3f %7.3f\n",q,q/y2,y2,\
   (q-q1)/(y2-y1));
 y2*=1.05;}}
```

Input Data: 2 1 9.81 10 41

q	V_2	Y_2	v
2.01	1.004	2.002	5.306
2.20	1.080	2.036	5.490
2.40	1.158	2.072	5.549
2.60	1.234	2.107	5.607
.	.	.	.
9.40	3.094	3.038	7.127
9.60	3.136	3.061	7.164
9.80	3.178	3.083	7.200
10.00	3.220	3.106	7.236

EXAMPLE PROBLEM 3.10

A uniform flow of Q = 300 cfs exists in a trapezoidal channel with b = 10 ft, m = 1.5, n = 0.014, and S_o = 0.0004. Compare the velocity of the constant height waves that will occur from increasing the flow rate in increments of 50 to 1000 cfs with the celerities of small amplitude gravity waves in the channel upstream from the wave.



Solution

First the initial condition is obtained by solving Manning's equation giving $Y_1 = 4.327$ ft and $V_1 = 4.2043662$ fps (the accuracy seven digits is used so Q =300 cfs). For each new increment of Q Equation 3.36 needs to be solved for the new depth Y_2, i.e., Equation 3.36 replaces Equation 3.33c used in the previous problem. Thereafter the wave velocity v is solved using Equation 3.30, and the celerity is computed from $c_2 = (gA_2/T_2)^{1/2}$. The Fortran program WAVETR.FOR given below solves the problem for 15 increments, and the solution table is given thereafter. Because of the nature of Equation 3.36 for depth Y_2 close to Y_1 (or Q_2 close to 300) guesses very close to the sought after solution must be provided, or else a solution for Y_2 less than Y_1, "the supercritical root," or another root will be obtained. For flow rates close to 300 cfs the following table provides the desired roots for Y_2.

Q(cfs)	301	302	303	304	305	310	315	320	330	340	350
Y_2 (ft)	4.3301	4.3331	4.3362	4.3392	4.3422	4.3574	4.3725	4.3875	4.4172	4.4464	4.4755

Notice from the solution table (and the graph) that even with the increasing area of this trapezoidal channel with m = 1.5 with depth Y, that the normal depth Y_o given in the last column of the solution table is greater than Y_2. Therefore, in practice a constant height wave would not actually occur because the depth upstream of the wave would be gradually varied.

Program WAVETR.FOR

```
      WRITE(*,*) 'Give: Y1,V1,g,Q2,N,b,m,n,So'
      READ(*,*) Y1,V1,G,Q2,N,B,FM,FN,So
      CU=1.486
      IF(G.LT.20) CU=1.
      So=SQRT(So)
      FMS=2.*SQRT(FM*FM+1.)
      A1=(B+FM*Y1)*Y1
      Q1=V1*A1
      dQ=(Q2-Q1)/FLOAT(N-1)
      Y2=4.48
      Yo=Y2
      FM2=2.*FM
      FM3=FM/3.
      B2=B/2.
      T=B+FM2*Y1
      C1=SQRT(G*A1/T)
      v=C1-V1
      WRITE(3,110) C1,v,V1,Q1
```

```
110   FORMAT(' Celerity c1=',F8.3,' Initial Wave Vel=',F8.3,'
     & V1=',F8.3,' Q1=',F8.3)
      HC1=(B2+FM3*Y1)*Y1**2/A1
      WRITE(3,100) Q1,V1,Y1,C1+V1,C1,Y1
      DO 10 I=2,N
      Q=Q1+dQ*float(I-1)
      M=0
1     A=(B+FM*Y2)*Y2
      HC2A=(B2+FM3*Y2)*Y2**2
      F=g*(HC2A/A1-HC1)/(1.-A1/A)-((Q-Q1)/(A-A1)-V1)**2
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 2
      F1=F
      Y22=Y2
      Y2=1.005*Y2
      GO TO 1
2     DIF=(Y2-Y22)*F1/(F-F1)
      Y2=Y22-DIF
      IF(ABS(DIF).GT. 1.E-5 .AND. M.LT.30) GO TO 1
      IF(M.EQ.30) WRITE(*,*)' Failed to converge',I,DIF
      A=(B+FM*Y2)*Y2
      C2=SQRT(g*A/(B+FM2*Y2))
      M=0
3     Ao=(B+FM*Yo)*Yo
      F=FN*Q-CU*Ao*(Ao/(B+FMS*Yo))**.66666667*So
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 4
      F1=F
      Yoo=Yo
      Yo=1.005*Yo
      GO TO 3
4     DIF=(Yo-Yoo)*F1/(F-F1)
      Yo=Yoo-DIF
      IF(ABS(DIF).GT. 1.E-5 .AND. M.LT.30) GO TO 1
      IF(M.EQ.30) WRITE(*,*) ' Manning failed',I,DIF
      WRITE(3,100) Q,Q/A,Y2,(Q-Q1)/(A-A1),C2,Yo
10    Y2=1.05*Y2
100   FORMAT(F8.2,5F8.3)
      END
```

Program WAVETR.C

```
#include <math.h>
#include <stdlib.h>
#include <stdio.h>
void main(void){int n,i,m;
FILE *filo; char fnam[20];
float y1,v1,g,q2,b,fm,fn,so,a1,q1,dq,y2,yo,yoo,fms,fm2,fm3,b2,t,\
  c1,q,f,f1,dif,hc1,hc2a,y22,a,c2,ao,cu=1.486;
printf("Give: Y1,V1,g,Q2,N,b,m,n,So\n");
scanf("%f %f %f %f %d %f %f %f %f",&y1,&v1,&g,&q2,&n,&b,&fm,&fn,\
  &so);
printf("Give output filename\n");scanf("%s",fnam);
if((filo=fopen(fnam,"w"))==NULL){
  fprint("Cannot open output file\n");exit(0);}
if(g<20.) cu=1; so=sqrt(so); fms=2.*sqrt(fm*fm+1.);
  a1=(b+fm*y1)*y1;   q1=v1*a1;dq=(q2-q1)/(float)(n-1);
```

```
y2=4.48;yo=y2;fm2=2.*fm;fm2=fm/3.;t=b+fm2*y1;c1=sqrt(g*a1/t);
fprintf(filo,"Celerity c1= %8.3f Initial Wave Vel= %8.3f\
  V1= %8.3f Q1 = %8.3f\n",q1,v1,y1,c1+v1,c1,y1);
hc1=(b2+fm3*y1)*y1*y1/a1;fprintf(filo,"%8.2f   %7.3f   %7.3f\
  %7.3f   %7.3f %7.3f/n",q1,v1,y1,c1+v1,c1,y1);
for (i=1;i<n;i++){q=q1+dq*(float)i; m=0; L1:a=(b+fm*y2)*y2;
 hc2a=(b2+fm3*y2)*y2*y2;
f=g*(hc2a/a1-hc1)/(1.-a1/a)-pow((q-q1)/(a-a1)-v1,2.);
if((++m)%2){f1=f;y22=y2;y2*=1.005;goto L1;}
dif=(y2-y22)*f1/(f-f1);y2=y22-dif;
 if((fabs(dif)>1.e-5)&&(m<20)) goto L1;
if(m==30) printf("Failed to converge %d %f/n",i+1,dif);
a=(b+fm*y2)*y2; c2=sqrt(g*a/(b+fm2*y2));m=0;
L3: ao=(b+fm*yo)*yo;
 f=fn*q-cu*ao*pow(ao/(b+fms*yo),.66666667)*so;
if((++m)%2){f1=f;yoo=yo;yo*=1.005;goto L3;} dif=(yo-yoo)*f1/(f-f1);
 yo=yoo-dif;
if((fabs(dif)>1.e-5)&&(m<30)) goto L3;
 if(m==30)printf("Manning failed %d %f\n",i,dif);
fprintf(filo,"%8.2f %7.3f %7.3f %7.3f %7.3f %7.3f\n",q,q/a,y2,\
 (q-q1)/(a-a1),c2,yo);
y2*=1.05;}}
```

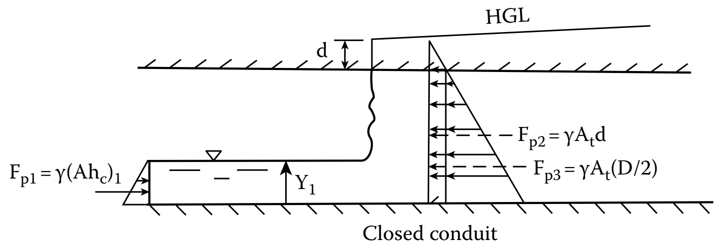
Input data: 4.327 4.2043662 32.2 1000 15 10 1.5 .014 .0004
Output table:
Celerity c1= 9.999 Initial Wave Vel= 5.795 V1= 4.204 Q1= 300.000

Q	V_2	Y_2	v	c_2	Y_o
300.00	4.204	4.327	14.203	9.999	4.327
350.00	4.679	4.475	14.514	10.140	4.688
400.00	5.121	4.615	14.810	10.270	5.021
450.00	5.535	4.748	15.091	10.391	5.332
500.00	5.926	4.874	15.360	10.504	5.624
550.00	6.296	4.994	15.618	10.611	5.900
600.00	6.647	5.110	15.867	10.712	6.162
650.00	6.983	5.220	16.106	10.808	6.412
700.00	7.304	5.327	16.338	10.898	6.651
750.00	7.612	5.430	16.563	10.985	6.880
800.00	7.909	5.529	16.782	11.068	7.101
850.00	8.195	5.625	16.994	11.147	7.314
900.00	8.472	5.719	17.200	11.224	7.520
950.00	8.739	5.809	17.401	11.297	7.720
1000.00	8.998	5.897	17.598	11.368	7.913

3.8 OPEN CHANNEL TO PIPE FLOW

If the flow is under supercritical conditions at a depth such that its conjugate depth is above the top of the conduit, then a modified hydraulic jump will change the flow from open channel to closed conduit flow. The momentum principle allows this situation to be analyzed. Following the usual steps needed in utilizing the momentum principle, the first step produces a control volume such as that above. The force upstream on this control volume is the same as that upstream from a hydraulic jump, namely the hydrostatic force from the removed upstream fluid, e.g., $\gamma A h_{c1}$. Downstream in the closed conduit section the force from the removed fluid can best be represented by two forces;

that due to the head of water above the top of the conduit, and that due to a hydrostatic pressure distribution of liquid depth at the top of the conduit. These two downstream forces are: $F_{p21} = \gamma A_t d$ and $F_{p22} = \gamma A_t(D/2)$, respectively. $A_t = \pi D^2/4$ for a pipe conduit, or the entire cross-sectional area of a noncircular conduit, and (D/2) is the distance from the top of the conduit to its centroid, and d is the depth water would stand in a standpipe attached to the top of the conduit just downstream from where the conduit flows full. The pressure on the top of the conduit is $p = \gamma d$.



Applying the equation that the summation of forces on the control volume of fluid must equal the momentum flux leaving the control volume minus the momentum flux entering the control volume in the x direction gives

$$\gamma A_1 h_{c1} - \gamma A_t d - \gamma \frac{D}{2} A_t = \frac{\gamma Q^2}{g}\left(\frac{1}{A_t} - \frac{1}{A_1}\right)$$

or solving for the head d on the top of the conduit at the downstream section

$$d = \frac{\left\{A_1 h_{c1} - \frac{1}{2} D A_t + \frac{Q^2}{g}\left(\frac{1}{A_1} - \frac{1}{A_t}\right)\right\}}{A_t} \tag{3.38}$$

An alternate form of Equation 3.38 involves writing it in terms of the momentum functions, or

$$d = \frac{(M_1 - M_t)}{A_t} = \frac{4(M_1 - M_t)}{(\pi D^2)} \tag{3.28a}$$

in which M_t is the momentum function based on the pipe flowing full and is given by $M_t = A_t(D/2) + Q^2/(gA_t)$.

EXAMPLE PROBLEM 3.11

A 12-in. diameter PVC sewer line suddenly changes grade from $S_{o1} = 0.0496$ to $S_{o2} = 0.0005$. The flow rate is Q = 1.5 cfs. Downstream from the change in grade at a distance of 5060 ft the pipe discharges into a pond with a water surface elevation at the top of the pipe. Determine where the flow changes from open channel to pipe flow.

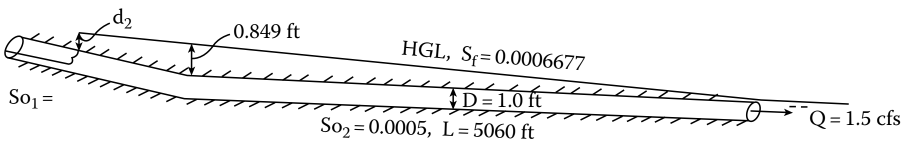
Solution

The normal depth for Q = 1.5 cfs equals $Y_{o1} = 0.230$ ft from Manning's equation. A computation of the slope S_f of the HGL for the downstream pipe flowing full gives, $S_f = [nQP^{2/3}/(1.49A^{5/3})]^2 = 0.0006677$ (using n = 0.008 for PVC pipe). Since this value is greater than the slope of the downstream pipe, it will flow full from its end to the position where the modified hydraulic jump takes place. Substituting into Equation 3.38 gives,

$$d = \left(0.013 - \frac{\pi 1}{4 2} + \frac{1.5^2}{32.2} - \frac{1}{0.136}\frac{\pi}{4}\right)\Big/\frac{\pi}{4} = 0.070$$

From the slope of the downstream HGL it is 0.849 ft above the top of the pipe at the position of the break in grade. Therefore the position where the modified jump will occur upstream from the break in grade is

$$L = \frac{0.849}{(0.0174 - 0.0006677)} = 15.9 \text{ ft}$$



EXAMPLE PROBLEM 3.12

Water is flowing down a steep pipe with a diameter, D_1 = 3.5 ft, a Manning's roughness coefficient, n_1 = 0.012, and a bottom slope of S_{o1} = 0.15. At the end of the pipe the water is directed into a trapezoidal channel with b_2 = 2 ft, m_2 = 0.8, n_2 = 0.013, and S_{o2} = 0.000006. It is observed that 3 ft from the end of the pipe the flow changes from open channel to pipe flow. For the pipe flow the equivalent sand roughness is e = 0.002 in. Determine what the flow rate Q is.

Solution

There are actually six unknowns in this problem. Q, S_f (the slope of the energy line in the pipe flow), d, Y_1, Y_2, and f. The equations needed to solve this problem are

$$Q = \frac{1.486}{n_1 A_1}\left(\frac{A_1}{P_1}\right)^{2/3} S_{o1}^{1/2} \quad \text{(Manning's equation: upstream channel)} \tag{1}$$

$$Q = \frac{1.486}{n_2 A_2}\left(\frac{A_2}{P_2}\right)^{2/3} S_{o2}^{1/2} \quad \text{(Manning's equation: downstream channel)} \tag{2}$$

$$d = \frac{\left[M_1 - A_t(D/2) - Q^2/(gA_t)\right]}{A_t} \quad \text{(Momentum)} \tag{3}$$

$$(S_{o1} - S_f)x = Y_2 - D - d \quad \text{(From geometry of slopes)} \tag{4}$$

$$S_f = \frac{fQ^2}{(2g(\pi/4)^2 D^5)} \quad \text{(Darcy–Weisbach for pipe flow)} \tag{5}$$

$$\frac{1}{\sqrt{f}} = 1.14 - 2\text{Log}\left(\frac{e}{D} + \frac{7.343473\nu D}{Q/\sqrt{f}}\right) \quad \text{(Colebrook–White)} \tag{6}$$

These equations are used in the TK-Solver variable and rule sheets as shown below to obtain the solution. Thereafter a similar Mathcad model is given. Notice in these models that separate equations have been used to define β, the areas, the wetted perimeters and the upstream momentum function so that 6 (and if one wants to include A_t in this list 7) additional variables, β_1, A_1, P_1, A_2, P_2, M_1, and A_t, have been added to the list of unknowns, 12 (or 13) equations are being solved simultaneously. The flow rate is Q = 141.2 cfs. Notice in this model, since separate equations have been used for areas, wetted perimeters, the upstream momentum function, and the total area, that 7 additional variables have been added in the list of unknown variables, so 13 equations are being solved for 13 unknowns.

```
=====================VARIABLE SHEET=====================
St   Input---     Name---   Output---    Unit---   Comment---
                  Q         141.28483
     1.486        C
     .012         n1
                  A1        3.5758985
                  P1        4.7823839
     .15          So1
     .013         n2
                  A2        177.62387
                  P2        37.096815
     .000006      So2
                  beta      1.3663954
                  Y1        1.394784
     3.5          D
     2            b2
     .8           m2
                  Y2        13.703004
                  M1        175.44462
                  d         9.788304
                  At        9.6211275
     32.2         g
                  Sf        .01176663
     3            x
                  f         .01229892
     .00033333    e
     .00001217    v
=======================RULE SHEET=======================
S Rule-------------------------------------------------
  Q=C/n1*A1*(A1/P1)^.66666667*sqrt(So1)
  Q=C/n2*A2* (A2/P2)^.66666667*sqrt(So2)
  cos (beta)=1.-2*Y1/D
  A1=D*D/4* (beta-cos (beta)*sin(beta))
  P1=beta*D
  A2=(b2+m2*Y2)*Y2
  P2=b2+2*Y2*sqrt (m2*m2+1)
  M1=.5*D*(D*D/6.*sin(beta)^3-A1*cos (beta))+Q*Q/g/A1
  d=(M1-.5*D*At-Q*Q/(g*At))/At
  At=.25*pi()*D*D
   (So1-Sf)*x=Y2-D-d
  Sf=f*Q*Q/(1.23370055*g*D^5)
1/sqrt(f)=1.14-2*log(e/D+7.3434728*v*D/abs(Q)/sqrt(f))
```

Mathcad model to solve Problem 3.12.

Variables Cu: = 1.486 nl: = .012 n2: = .013 Sol: = .15 So2: = .000006 D: = 3.5 b2: = 2 m2: = .8 g: = 32.2 x: = 3 e: = .0003333333 v: = .00001217 Q: = 130 A1: = 3.5 P1: = 4.5 A2: = 180 P2: = 35 β: = 1.5 Y1: = 1.4 Y2: = 13.7 Ml: = 170 d: = 9.8 At: = 9.6 Sf: = .0117 f: = .013

Given

$$Q = \frac{Cu}{nl} \cdot A1 \cdot \left(\frac{A1}{P1}\right)^{.6666667} \cdot \sqrt{So1}\ Q = \frac{Cu}{n2} \cdot A2 \cdot \left(\frac{A2}{P2}\right)^{.6666667} \cdot \sqrt{So2}\ \cos(\beta) = 1. - 2 \cdot \frac{Y1}{D}$$

$$Al = 25 \cdot D^2 \cdot (\beta - \cos(\beta) \cdot \sin(\beta))$$

$$P1 = \beta \cdot D\ A2 = (b2 + m2 \cdot Y2) \cdot Y2 \quad P2 = b2 + 2 \cdot Y2 \cdot \sqrt{m2 \cdot m2 + 1}$$

$$M1 = 5 \cdot D \cdot \left(\frac{D^2}{6} \cdot \sin(\beta)^3 - A1 \cdot \cos(\beta)\right) + \frac{Q^2}{g \cdot A1}$$

$$At = .25 \cdot \pi \cdot D^2\ d = \frac{M1 - D \cdot \frac{At}{2} - \frac{Q^2}{g \cdot At}}{At}\ (Sol - Sf) \cdot x = Y2 - D - d\ Sf = \frac{f \cdot Q^2}{1.2337005 \cdot g \cdot D^5}$$

$$\frac{1}{\sqrt{f}} = 1.14 - 2 \cdot \log\left(\frac{e}{D} + 7.3434728 \cdot v \frac{D}{Q \cdot \sqrt{f}}\right)$$

Find (Q, Sf, d, Y1, Y2, f, β, A1, P1, A2, P2, M1, At) =

	0
0	141.2848
1	0.0118
2	9.7883
3	1.3948
4	13.703
5	0.0123
6	1.3664
7	3.5759
8	4.7824
9	177.6239
10	37.0968
11	175.4446
12	9.6211

The logic needed to program a solution of simultaneous nonlinear equations has been discussed earlier. A FORTRAN and a C program to solve this problem are given below. In these programs the Colebrook–White equation is solved by a Gausel iteration based on the current values of Q and thus only five simultaneous equations are solved for the variables Q, S_f, d, Y_1, and Y_2, with the array X containing these unknowns in this order. The Jacobian matrix for the Newton method is evaluated numerically by calling on the subroutine (function) FUN with each variable increment as well as without this increment, as described previously.

It does not take big changes in the variable that describe this problem in order for a solution not to exist, i.e., the jump is washed out from the pipe.

Program PRB3_12.FOR

```
      REAL F(5),F1(5),DJ(5,5)
      INTEGER*2 INDX(5)
      COMMON G,CU,SO1,SO11,SO2,D,B2,FM2,ED,VIST,RF,DS4,FMS,AT,DG5,
     &XX,X(5)
```

```
      WRITE(*,*)' Give: g,n1,n2,So1,So2,D,b2,m2,e,X'
      READ(*,*) G,FN1,FN2,SO11,SO2,D,B2,FM2,E,XX
      WRITE(*,*)' Provide guesses for: Q,Sf,d,Y1,Y2'
      READ(*,*) X
      RF=8.
      CU=1.486
      VIST=1.217E-5
      IF(G.GT.20.) GO TO 1
      CU=1.
      VIST=1.31E-6
1     ED=E/D
      VIST=7.3434728*VIST*D
      DS4=.25*D*D
      FMS=2.*SQRT(FM2*FM2+1.)
      SO1=CU*SQRT(SO11)/FN1
      SO2=CU*SQRT(SO2)/FN2
      AT=.78537816*D**2
      DG5=1.23370055*G*D**5
      M=0
10    SUM=0.
      CALL FUN(F)
      DO 20 J=1,5
      XT=X(J)
      X(J)=1.01*X(J)
      CALL FUN(F1)
      DO 18 I=1,5
18    DJ(I,J)=(F1(I)-F(I))/(X(J)-XT)
20    X(J)=XT
      CALL SOLVEQ(5,1,5,DJ,F,1,DD,INDX)
      DO 30 I=1,5
      X(I)=X(I)-F(I)
30    SUM=SUM+ABS(F(I))
      M=M+1
      IF(SUM.GT. 1.E-4 .AND. M.LT.20) GO TO 10
      WRITE(*,100) X
100   FORMAT(' Solution:',/' Q =',F8.1,/' Sf =',F8.6,/
     &' d =',F8.3,/' Y1=',F8.3,/' Y2 =',F8.3)
      END
      SUBROUTINE FUN(F)
      REAL F(5)
      COMMON G,CU,SO1,SO11,SO2,D,B2,FM2,ED,VIST,RF,DS4,FMS,AT,
     &DG5,XX,X(5)
      COSB=1.-2.*X(4)/D
      BETA=ACOS(COSB)
      A1=DS4*(BETA-COSB*SIN(BETA))
      F(1)=X(1)-SO1*A1*(A1/(D*BETA))**.66666667
      A2=(B2+FM2*X(5))*X(5)
      F(2)=X(1)-SO2*A2*(A2/(B2+FMS*X(5)))**.66666667
      F(3)=X(3)-(.5*D*(D*D/6.*SIN(BETA)**3-A1*COSB-AT)+
     &X(1)**2/G*(1./A1-1./AT))/AT
1     RF1=RF
      RF=1.14-2.*ALOG10(ED+VIST*RF1/X(1))
      IF(ABS(RF-RF1).GT. 1.E-6) GO TO 1
      SF=(X(1)/RF)**2/DG5
```

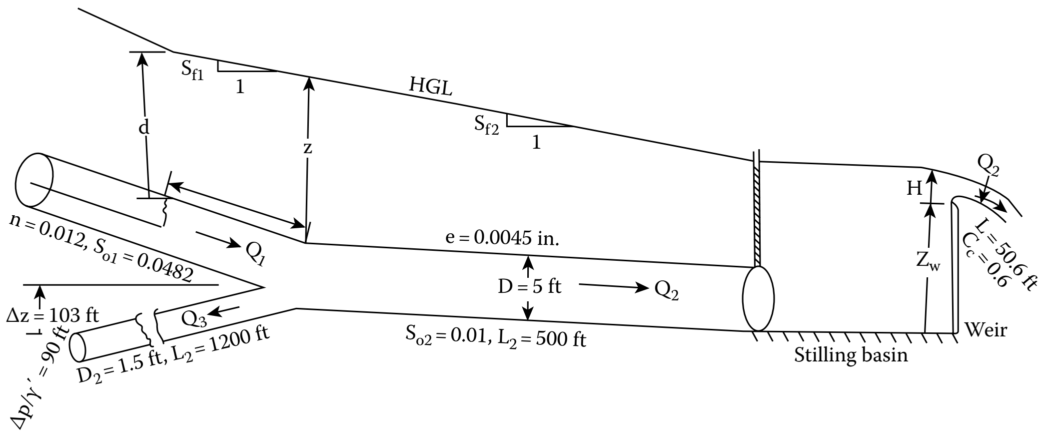
```
      F(4)=(SO11-SF)*XX-X(5)+D+X(3)
      F(5)=X(2)-SF
      RETURN
      END
```

Program PRB3_12.C

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float g,cu,so1,so11,so2,d,b2,m2,ed,vist,rf,ds4,fms,at,dg5,xx,x[5];
void fun(float *f){float cosb,sinb,beta,a1,a2,rf1,sf;
 cosb=1.-2.*x[3]/d; sinb=sqrt(1.-cosb*cosb); beta=atan(sinb/cosb);
   if(beta<0.) beta+=3.14159265;
 a1=ds4*(beta-cosb*sinb);
   f[0]=x[0]-so1*a1*pow(a1/(d*beta),.6666667);
 a2=(b2+m2*x[4])*x[4];
   f[1]=x[0]=so2*a2*pow(a2/(b2+fms*x[4]),.6666667);
 f[2]=x[2]-(.5*d*(d*d/6.*pow(sinb,3.)-a1*cosb-at)+x[0]\
   *x[0]/g*(1./a1–1./at))/at;
 do{rf1=rf; rf=1.14–2.*log10(ed+vist*rf1/x[0])
 } while(fabs(rf-rf1)<1.e-6);
 sf=pow(x[0]/rf,2.)/dg5; f[3]=(so11-sf)*xx-x[4]+d+x[2];
   f[4]=x[1]-sf;} // end fun
void solveq(int n,float **a,float *b,int itype,float *dd,\
  int *indx);
void main(void){float e,n1,n2,xt,sum,f[5],f1[5],*dd,**dj;
  int i,j,m,indx[5];
dj=(float**)malloc(5*sizeof(float*));
  for(i=0;i<5;i++)dj[i]=(float*)malloc(5*sizeof(float));
 printf(" Give: g,n1,n2,So1,So2,D,b2,m2,e,x\n");
 scanf("%f %f %f %f %f %f %f %f %f %f",&g,&n1,&n2,&so11,&so2,&d,\
   &b2,&m2,&e,&xx);
 printf(" Provide guesses for: Q,Sf,d,Y1,Y2\n");
 scanf("%f %f %f %f %f",&x[0],&x[1],&x[2],&x[3],&x[4]);
   rf=8.,cu=1.486;vist=1.217e-5;
 if(g<20.) {cu=1.;vist=1.31e-6;} ed=e/d; vist*=7.3434728*d;
   ds4=.25*d*d;fms=2.*sqrt(m2*m2+1.);
 so1=cu*sqrt(so11)/n1; so2=cu*sqrt(so2)/n2; at=.78537816*d*d;
   dg5=1.23370055*g*pow(d,5.); m=0;
 do{sum=0.; fun(f); for(j=0;j<5;j++){xt=x[j]; fun(f1);
   for(i=0;i<5;i++) dj[i][j]=(f1[i]-f[i])/(x[j]-xt);
   x[j]=xt;} solveq(5,dj,f,1,dd,indx);
   for(i=0;i<5;i++){x[i]-=f[i]; sum+=fabs(f[i]);}
 } while ((sum<1.e-4) && (++m<20));
printf("Solution:\n Q =%8.1f\n Sf =%8.6f\n d =%8.3f\n\
  Y1 =%8.3f\n Y2 =%8.3f\n", x[0],x[1],x[2],x[3],x[4]);}
```

EXAMPLE PROBLEM 3.13

A steep open channel pipe with D = 5 ft, S_{o1} = 0.0482, and n = 0.012 has a change in its bottom slope to S_{o2} = 0.01 at a point where a 1.5 ft diameter pipe takes water out from the bottom of the larger pipe as shown in the sketch. At a distance 500 ft downstream from the break-in-grade the pipe discharges into a stilling basin that discharges the water over a sharp crested weir that is 50.6 ft long. The crest of the weir is Z_w = 4 ft above the bottom of the pipe, and its discharge coefficient is C_d = 0.6. The 1.5 ft diameter pipe is 1200 ft long and discharges at an elevation 103 ft below the bottom of the larger pipe. It is to supply water with a pressure head $p/\gamma = 90$ ft. For use in the Darcy–Weisbach equation use an equivalent sand roughness e = 0.0045 in. for both pipes. If the flow rate coming down the 5 ft diameter pipe is $Q_1 = 600$ cfs, determine the discharge in the smaller pipe Q_o, the flow rate entering the spillway Q_2, and the position x where a modified hydraulic jump will occur.



Solution

First from $Q_1 = 600$ cfs and the slope and n of the upstream pipe the normal depth can be solved from Manning's equation as $Y_{o1} = 3.961$ ft. Next from the Darcy–Weisbach and Colebrook–White equations, the slope of the EL (or HGL) after the modified jump to the break in grade can be solved as $S_{f1} = 0.0336853$. There are seven unknowns: Q_2, Q_p, x, d, S_{f2}, H, and z. The seven available equations are:

$$Q_2 = C_d\left(\frac{2}{3}\right)(2g)^{1/2} L_w H^{1.5} \tag{1}$$

$$z + D + S_{o2}L_2 = Z_w + H + S_{f2}L_2 \tag{2}$$

$$d + S_{o1}x = z + S_{f1}x \tag{3}$$

$$d = \frac{(M_1 - M_t)}{A_t} \tag{4}$$

$$Q_2 = Q_1 - Q_p \tag{5}$$

$$z + D + \Delta\text{elev} - \frac{p}{\gamma} = f_p\left(\frac{L_p}{D_p}\right)\frac{(Q_p/A_p)^2}{(2g)} \tag{6}$$

$$S_{f2} = \left(\frac{f}{D}\right)\frac{(Q_2/A_t)^2}{(2g)} \tag{7}$$

Plus 2 Colebrook–White Equations for f and f_p.

A TK-Solver model to solve this problem is given below.

```
========== VARIABLE SHEET ==========
St Input----    Name---    Output---
                Q2         577.62052
                Qp         22.379478
                x          532.04304
                H          2.3298401
                z          11.87257
                Sf2        .03108546
                d          4.1501248
                f          .01156612
                fp         .01499386
5               D
.01             So2
.6              Cd
32.2            g
50.6            Lw
500             L2
4               Zw
.0482           So1
.0336853        Sf1
                M1         699.97395
                beta       2.1954298
                A1         16.68629
3.962           Y1
                At         19.634954
                Mt         618.48644
600             Q1
103             Delev
90              phead
1200            Lp
1.5             Dp
.000375         e
.0000141        vis
============ RULE SHEET ============
SRule------------------------------


* Q2=Cd*(2/3)*sqrt(2.*g)*Lw*H^1.5
* z+D+So2*L2=Zw+ H+Sf2*L2
* d+D+So1*x=D+z+Sf1*x
* cos(beta)=1.-2*Y1/D
* A1=D^2/4* (beta-cos(beta)*sin(beta))
* M1=.5*D* (D^2/6*sin (beta)^3-A1*cos (beta))+Q1*Q1/(g*A1)
* At=pi()/4*D^2
* Mt=.5*D*At+Q1^2/(g*At)
* d=(M1-Mt)/At
* Q2=Q1-Qp
* z+Delev+D-phead=fp*Lp/Dp*(Qp/(pi()/4*Dp^2))^2/(2.*g)
* Sf2=f/D*(Q2/At)^2/(2.*g)
* 1/sqrt(f)=1.14-2*log(e/D+7.34347283*vis*D/(Q2*sqrt(f)))
* 1/sqrt(fp)=1.14-2*log(e/Dp+7.34347283*vis*Dp/(Qp*sqrt(fp)))
```

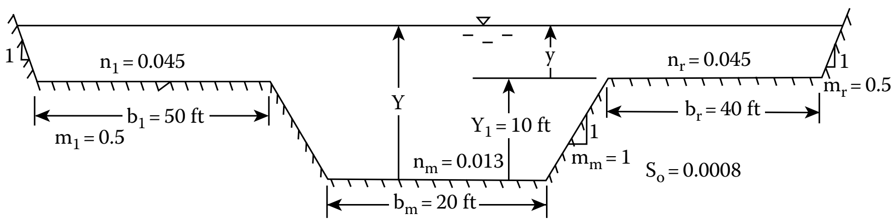
3.9 MULTIPLE ROUGHNESS COEFFICIENT FOR CHANNEL SECTION—COMPOUND SECTIONS

Under flood flow conditions, the depth of water may exceed the height of the main river bank and then the flood plain areas on one or both sides of the main river channel become part of the section that conveys water. The side flood plains generally cause significantly larger resistance to flow than the main river channel. Simulation of such flows in natural streams and rivers will be dealt with in Chapter 6. A channel with several values for the roughness coefficients for different portions of the channel is commonly referred to as a compound channel. The flow in such channels can be visualized as different channel flows that have become combined because a wall between them is missing. Such compound channel flows are different than those discussed in Chapter 2, in which an equivalent Manning's roughness coefficient was used that was obtained by weighting the separate n′ values according to the fraction of the wetted perimeter to which each applied. The approach in Chapter 2 is applicable for a concrete channel with gravel deposits in its bottom, for example. The approach described in this section for compound channels is applicable for situations in which a main channel is overtopped and the areas adjacent to its sides convey the spilled water in the direction of the main channel flow.

We will now examine several interesting hydraulic properties of compound sections. Let us consider an example in which the main channel has a bottom width of b = 9 m, and a side slope of m = 0.5, and when the depth reaches 5 m then the water flows out of the main channel onto both sides with expanded widths of b_r = 15 m and b_l = 15 m, as shown in the sketch. The main channel has a Manning's roughness of n_m = 0.018 and both overflow sides have roughness values of n_r = 0.055 and n_l = 0.055. The bottom slope of all portions of the channel are the same, with S_o = 0.0008.

For this channel, its geometric properties need to be defined by equations that apply when the depth of flow is less and greater than the capacity of the main channel as follows:

For Y ≤ 5 m



Sketch of channel section with several roughness coefficients

$$A = (b + mY)Y, \quad P = b + 2Y(m^2 + 1)^{1/2}, \quad T = b + 2mY$$

(the usual equations for a trapezoidal channel)

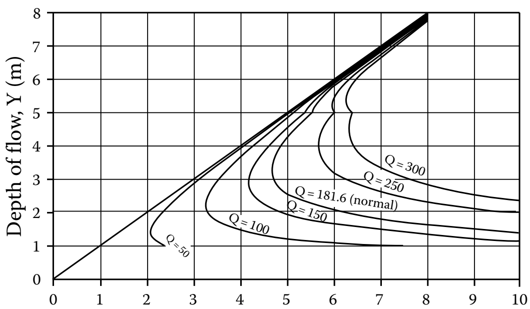
For Y > 5 m

$$A = 57.5 + (b_r + b_l + 14.0)(Y - 5), \quad P = 20.18 + b_r + b_l + 2(Y - 5), \quad T = 14.0 + b_r + b_l$$

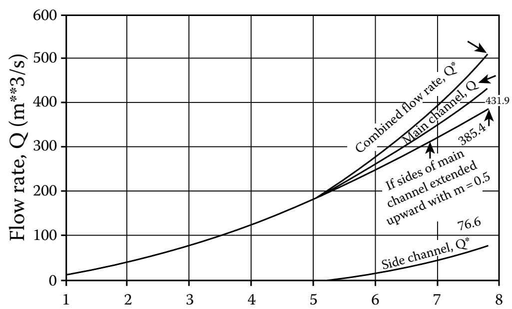
in which 57.5, 20.18, and 14.0 are the area, perimeter, and top width for the main channel when full, respectively. These equations will need to be modified according to the geometry of the main channel and overflow channels for other compound sections. If uniform flow were to occur at the top of the main channel then the flow rate would be Q_o = 181.6 m^3/s, and the Froude number would be

$F_r = \{Q^2T/(gA^3)\}^{1/2} = 0.50$. The specific energy diagram for this normal flow rate is shown as one of the curves on the specific energy diagram below. Note this diagram looks similar to such diagrams in regular sections with the exception that the curve breaks toward the 45° line, especially for larger flow rates in which velocities are larger, as the flow leaves the main channel into the side channels because the rapid expansion in width at this depth reduces the average velocity. In reality, the flow in the main channel will not be reduced, but rather the smaller velocities in the side channels reduce the average velocity.

A graph below shows how Manning's equation defines the relationship between the flow rate and the depth. (The channel bottom slope would need to change if these were to occur, of course.) For this compound channel, when the depth Y = 7.8 m, the main channel contains 431.9 m^3/s whereas the side channel contributes 76.6 m^3/s or 15% of the total flow even though the area of the side channels is 44.3% of the total. These computations are based on having the main channel continue upward with a side slope of 0.5, but not adding any to the wetted perimeter above the 5 m depth since the flow in the side channels will exist here rather than a channel wall. This graphs also shows the flow rate–depth relationship if the main channel actually did have side walls that extended upward to the 7.8 ft depth. Such an upward extending real walled main channel would carry only 385.4 m^3/s instead of 431.9 m^3/s because of the added wall resistance.



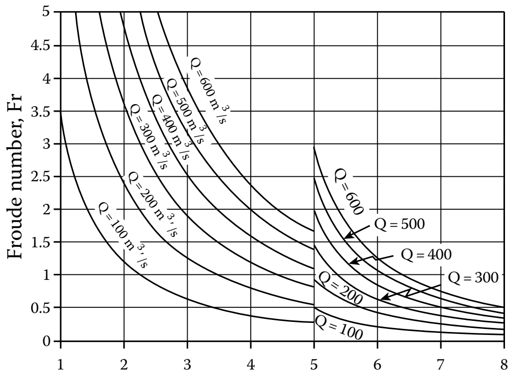
Specific energy diagram



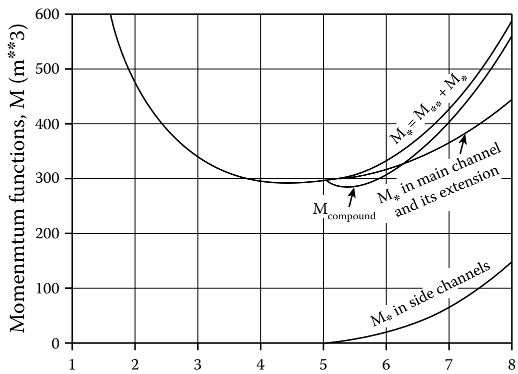
Flow rates computed based on Manning's equation

In a compound channel, it is possible that critical flow may occur for three different depths. These three critical flow conditions will occur for a given flow rate if critical depth occurs for that flow rate at a depth modestly less than the top of the main channel. When this occurs the Froude number becomes less than 1 as the depth rises to the top of the main channel. As the water rises above the main channel into the side channels, the top width of the compound channel immediately increases thus rapidly increasing the Froude number so it becomes unity again. Further increases in depth reduce the Froude number until it equals unity for a third depth. The graph below shows that for a flow rate of Q = 300 m^3/s critical flows occur at Y = 4.436 m, 5.0 + m, and 5.37 m. Again these results are based on computations that assume one-dimensional hydraulics. In actuality, especially for compound channels of the size used in this example, the main channel portion will be supercritical flow at certain depths while the flow in the side channels will still be subcritical.

In the table below, the assumption is made that the velocities are different in the main channel than the side channels and the total flow rate is Q = 300 m^3/s. To determine what portion of this total flow rate occurs in the main and side portions of the channel, the assumption is used that the slopes of the energy lines are the same for both. Letting K be the conveyance, $K = A(A/P)^{2/3}/n$, gives that $Q_m/K_m = Q_s/K_s$, or $Q_m = 300K_m/(K_m + K_s)$. In determining the areas it is assumed that the main portion of the channel extends upward with a side slope of 0.5.



Depth of flow in compound channel, Y (m)
Variation of Froude number, F_r with depth for several flow rate



Depth of flow, Y (m)
Variations of individual and compound momentum functions with depth for a flow rate Q = 300 m³/s

Computations based on assuming the flow in the main portion of the channel is separated from that in the side portions of the channel. Values are based on a total flow rate Q = 300 m³/s.

Depth	Main Channel					
Y (m)	Q_m (m³/s)	V_m (m/s)	A_m (m²)	P_m (m)	T_m (m)	F_{rm}
4.436	300.000	6.029	49.763	18.919	13.436	1.000
4.60	300.000	5.771	51.980	19.286	13.600	.943
4.80	300.000	5.482	54.720	19.733	13.800	.879
5.00	300.000	5.217	57.500	20.180	14.000	.822
5.10	299.471	5.084	58.905	20.404	14.100	.794
5.20	298.383	4.947	60.320	20.628	14.200	.766
5.30	296.942	4.809	61.745	20.851	14.300	.739
5.40	295.251	4.673	63.180	21.075	14.400	.712
5.50	293.377	4.540	64.625	21.298	14.500	.687
5.60	291.371	4.409	66.080	21.522	14.600	.662

Depth	Side Channels						Compound
Y (m)	Q_s (m³/s)	V_s (m/s)	A_s (m²)	P_s (m)	T_s (m)	F_r	Froude No.
4.436							1.000
4.60							.943
4.80							.879
5.00							.822
5.10	.529	.177	2.995	29.976	29.900	.178	1.305
5.20	1.617	.270	5.980	29.953	29.800	.193	1.177
5.30	3.058	.341	8.955	29.929	29.700	.199	1.069
5.40	4.749	.398	11.920	29.906	29.600	.200	.976
5.50	6.623	.445	14.875	29.882	29.500	.200	.896
5.60	8.629	.484	17.820	29.858	29.400	.199	.827

The first depth in this table is for critical flow. Note under the assumption that the flow is actually two different flows rather than flow in a single channel that the Froude number in the main channel continues to decrease, and of course the Froude number for the side channel flows is less than one. The last column in this table gives the Froude number computed as if this were a single compound channel. Note that F_r for the compound channel is larger than one whereas if one divides the flow into separate channels then both F_r' values are less then unity. Therefore, we might conclude that only for smaller sized cross sections are the computations of single compound channel cross section applicable for real occurrences.

Values of momentum function in the main channel (extended upward with m = 0.5), the side channel and the compound channel are plotted in the above graph. The values of the momentum functions in the side channels are small compared to that in the main channel. There are two reasons for this: first, the depth of flow is small in comparison to that in the main channel, and therefore the term Ah_c is small, and second, the flow rate in the side channels is small for a given area and therefore the term $Q^2/(gA^3)$ is also small. The computations of the momentum function for the compound channel becomes smaller for depths just larger than 5 m as shown. All these facts indicate that caution is called for when using one-dimensional hydraulic equations for compound channels.

PROBLEMS

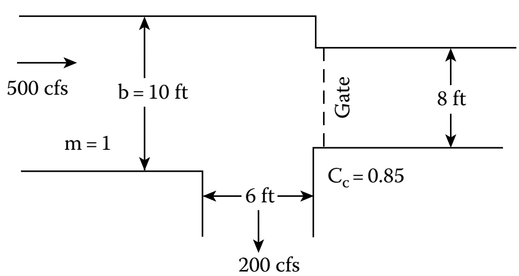
3.1 A hydraulic jump occurs in a rectangular channel with a bottom width of b = 8 m. For a flow rate Q = 80 m^3/s the depth upstream from the jump is 0.3 m. What is the depth downstream from the jump?

3.2 A flow rate of Q = 70 m^3/s exists in a trapezoidal channel with b = 3 m, and a side slope m = 1.5. If the depth downstream from the jump is measured as Y_2 = 6 m, what is the depth upstream from the jump?

3.3 A gate in a rectangular channel with b = 10 ft causes the depth of water downstream from it to be 2 ft. At the gate section the bottom of the channel rises abruptly by 2.0 ft. For a flow rate of Q = 450 cfs determine what the force is on the gate. The minor loss coefficient K_L = 0.2.

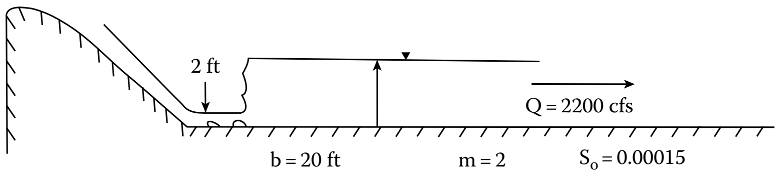
3.4 A hydraulic jump occurs at the end of a transition between a trapezoidal channel with b_1 = 8 ft, and m_1 = 1.5 and a rectangular channel with b_2 = 7 ft. For a flow rate of Q = 400 cfs, the depth in the downstream channel is Y_2 = 8 ft. What is the force against the transition? (Assume head loss across the hydraulic jump is given by the formula for a rectangular channel.)

3.5 A trapezoidal channel with a bottom width of 10 ft, and a side slope of 1.0 has a diversion from its side in a 6 ft wide rectangular channel as shown in the sketch. Upstream the flow rate is 500 cfs, and 200 cfs leave from the diversion. At a gate, the main channel changes to a rectangular channel with a bottom width of 8 ft. The gate is set 3 ft above the channel bottom and has a contraction coefficient of C_c = 0.85.

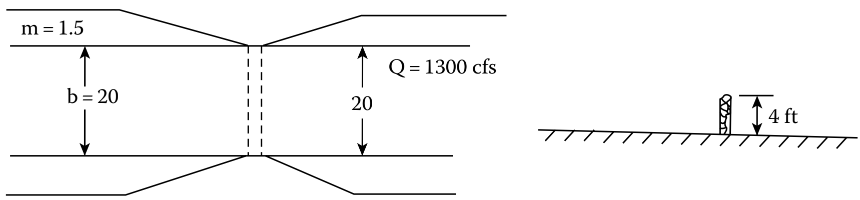


Determine the depth just upstream from the gate, the depth upstream in the main channel where the flow rate is 500 cfs, and the force on the gate and diversion structure, and the combined force on both.

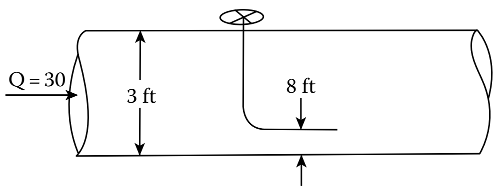
3.6 A flow rate of Q = 2220 cfs comes down the spillway of a dam, and at the toe of the dam on its apron the depth is 2 ft. The channel here has a bottom width b = 20 ft, and side slope m = 2. Baffles are to be placed in the apron to keep the hydraulic jump on the apron. If the downstream channel has a bottom slope S_o = 0.00015, and a roughness coefficient n = 0.013 determine what force should exist on these baffles to keep the jump immediately downstream from the dam. How much cross-sectional area should these baffles have if their drag coefficient based on the velocity upstream from the jump is C_D = 0.73?



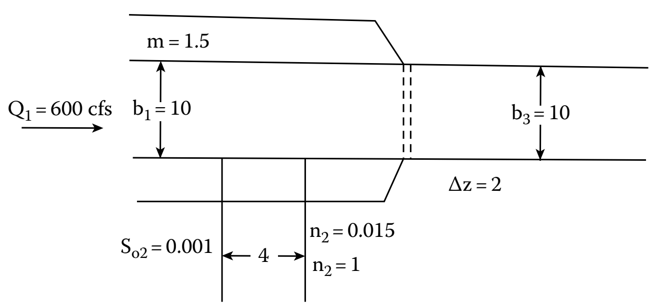
3.7 A trapezoidal channel with a bottom width of b = 20 ft, and a side slope m = 1.5 contains stop logs in a rectangular contraction of this channel to a width of 20 ft, and downstream the flow depth is considerably less than the upstream depth. The stop logs extend upward from the bottom of the channel a distance of 4 ft. If the flow rate in the channel is Q = 1300 cfs, determine the force against the stop logs and its contracting structure. Now assume that the downstream depth is greater, creating a depth above the stop logs of 5.5 ft and determine this force. Also what is the upstream depth Y_1 for this latter situation?



3.8 A gate valve is used to control the discharge from a 3 ft diameter pipe. Open channel flow exists downstream from the valve. If the valve causes a depth of Y_2 = 0.8 ft downstream of the valve, and the flow rate is Q = 30 cfs, determine the force against the gate valve. Ignore minor losses.



3.9 Stop logs in the bottom of a trapezoidal channel with b_1 = 10 ft, and m = 1.5 are used to divert water into a side channel with a bottom width of b_2 = 4 ft, and a side slope of m_2 = 1. The bottom slope of this channel is S_{o2} = 0.001, and its Manning's n_2 = 0.015. This channel is very long, and its bottom is 2 ft above the bottom of the main channel. The section of the main channel where the stop logs exist contracts to a rectangular section with a bottom width b_3 = 10 ft. The stop logs are 3 ft above the channel bottom and cause the flow to be critical over their top. If the flow rate coming into the main channel is Q_1 = 600 cfs, determine (a) the flow rates going by the stop logs and into the side channel, and (b) the force against the stop logs, and their hold structure. The side channel runs at 90° from the main channel.



3.10 Derive Equation 3.15.

3.11 Extract the root r = 1 from Equation 3.15 and prove that Equation 3.16 results. Explain what the difference is between Equations 3.15 and 3.16.

3.12 Starting from the dimensionless depth Y' for a trapezoidal channel, equate M_1' to M_2' and derive Equation 20.

3.13 Solve for the upstream depth in Example Problem 3.5 if the baffles did not exist.

3.14 If the upstream depth is $Y_1 = 1.924$ ft as determined in Example Problem 3.5 with the baffles in place, what would the downstream depth be without the baffles in place?

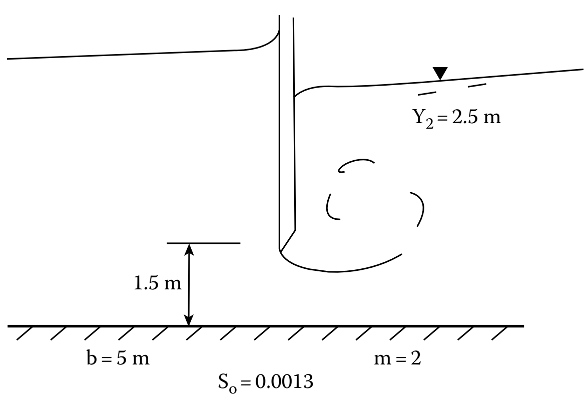
3.15 Type up the FORTRAN program, or utilize a similar program that will extract roots from a polynomial, including the complex roots, and investigate what range of flow rates, and/or depths upstream from a hydraulic jump will result in three real but negative depths, in addition to the real positive root being solved for instead of two complex roots. The channel you should do this investigation for has a bottom width of b = 4 m and a side slope m = 2.

3.16 Using the dimensionless momentum function find all roots of Equation 3.20 that are associated with a depth of 2.2 ft, and a trapezoidal channel with a bottom width of b = 20 ft, and a side slope m = 2 for a flow rate Q = 1500 cfs. Select the correct root, and determine the conjugate depth that would exit downstream of a hydraulic jump if the 2.2 ft were upstream from the hydraulic jump. Verify the result using Figure 3.1.

3.17 Determine the roots of the dimensionless momentum function associated with a flow rate of (a) 2000 cfs, and (b) 2500 cfs, if the channel is the same size as in the previous problem and $Y_1 = 2.2$ ft as in the previous problem.

3.18 Modify the FORTRAN program used to extract the roots from the dimensionless momentum function from Equation 3.20 so that it will extract the roots from Equations 3.15 and 3.16. Then to verify that your programs work for these two equations solve the problem to get the conjugate depth to $Y_1 = 2$ ft, and a trapezoidal channel with b = 10 ft and m = 1, with a flow rate Q = 400 cfs.

3.19 A gate discharges flow into a trapezoidal channel with a bottom width of b = 5 m, and a side slope m = 2, under submerged conditions. The channel has a bottom slope $S_o = 0.0013$, and a Manning's n = 0.015. The gate is set 1.5 m above the channel bottom. If the depth downstream from the gate is measured as $Y_2 = 2.5$ m, determine the following: (a) the flow rate, Q, (b) the depth upstream assuming that the velocity head under the gate is dissipated, and (c) the force against the gate. How does this force compare with the force computed if the pressure distribution were hydrostatic over the entire gate area wetted by the water?



3.20 Using dimensionless values obtained by dividing by the critical depth (or critical depth squared for m′) generate a table that contains the following columns for across a hydraulic jump: (1) the dimensional depth upstream, Y_u'. (2) The Froude number corresponding to

(1), F_{ru}. (3) The value of the dimensionless momentum function m′. (4) The corresponding value of the dimensionless specific energy E'_u. (5) The dimensionless depth downstream, Y'_d. (6) The Froude number corresponding to (5), F_{rd}. (7) The dimensionless specific energy downstream, E'_d. (8) The loss of dimensionless specific energy across the hydraulic jump. (9) Using E'_d from (7) use the solution of a cubic equation that produces the conjugate dimensionless depths of Figure 3.3 to obtained the downstream depth Y_d, i.e., duplicate the values in column 5, except use E'_d in place of m′ for the abscissa, and the ordinate is $1/Y'_e$ rather than Y'_m. Start this table with Y'_u equal to 1 in column 1 and decrement Y'_u by 0.05 and end with $Y'_u = 0.1$ in the final row of column 1 of the table.

3.21 As in the previous problem use dimensionless values divided by the critical depth to generate a table that contains the following columns across a *vertical gate*, rather than a hydraulic jump. (1) the dimensionless depth, Y'_u. (2) The Froude number corresponding to (1), F_{ru}. (3) The value of the dimensionless specific energy E'_u. (4) The corresponding value of the dimensionless momentum function m'_u. (5) The dimensionless depth downstream, Y'_d. (6) The Froude number corresponding to (5), F_{rd}. (7) The dimensionless downstream specific energy computed from (5), just to verify that $E'_d = E'_u$. (8) The dimensionless momentum function downstream, m'_d. (9) The dimensionless force per unit width on the gate, i.e., the difference in dimensionless momentum across the gate. (10) Using m'_d from (8) use the solution of a cubic equation that produces the alternate dimensionless depths of Figure 3.3 to obtained the downstream depth Y_d, i.e., duplicate the values in column 5, using m'_d for the abscissa. (11) Provide the alternate depth (subcritical depth) to that given in column 10. Start this table with Y'_u equal to 1 in column 1 and increment Y'_u by 0.05 and end with $Y'_u = 2.5$ in the final row of column 1 of the table.

3.22 The dimensionless cubic equations for energy and momentum for a rectangular channel have the reciprocal properties that when $1/Y'_e$ is substituted into the energy equation the momentum equation is produced, and if $1/Y'_m$ is substituted into the momentum equation the energy equation is produced. In this process m′ is treated as E′ and vice versa. The dimensionless values of depth and energy are obtained by dividing by the critical depth, and m′ by dividing by the critical depth squared. Verify these reciprocal properties by generating two tables of values. In the first column of Table 3.1 let m′ vary from 1.5 to 2.5 in increments of 0.05, as the next three columns provide the three roots obtained from solving the cubic momentum equation, and as the final three columns provide the three dimensionless depths associated with the first column of m′ values but obtained from solving the three roots of the energy equation. Note in this table that columns 5 through 7 contain the same values for the dimensionless momentum depths as columns 2 through 4. In the first column of Table 3.2 let E′ vary from 1.5 to 2.5 in increments of 0.05, as the next three columns provide the three dimensionless depths associated with the first column of E′ obtained from solving the three roots of the cubic energy equation, and as the final three columns provide the three dimensionless values associated with the first column of E′ values but obtained from solving the cubic momentum equation. Note in this second table that columns 5 through 7 contain the same values for the dimensionless energy depths as columns 2 through 4.

3.23 An iterative method for solving either the subcritical or supercritical depth associated with a given value for the momentum function in a trapezoidal channel is to use the cubic equation that applies for a rectangular channel, i.e., $F(Y) = Y^3 - 2cm_oY + 2cq^2/g = 0$, in which m_o is the average momentum function per unit width, or M/b_{av}, and q is the average flow rate per unit width or $q = Q/b_{av}$, in which $b_{av} = A/Y = b + mY$. Do the following: (1) Write the momentum function equation for a trapezoidal channel in the form of the above cubic equation, and in this process define how c is computed in this equation (note that as the side slope of the trapezoidal channel goes to 0 that c becomes 1). (2) Write a program that will solve for either the subcritical or supercritical depth associated with a specified value for the momentum function M. (3) Verify that your program works to solve the subcritical and supercritical depths if $Q = 400$ cfs, $M = 300\,ft^3$, $b = 10$ ft, and $m = 1$.

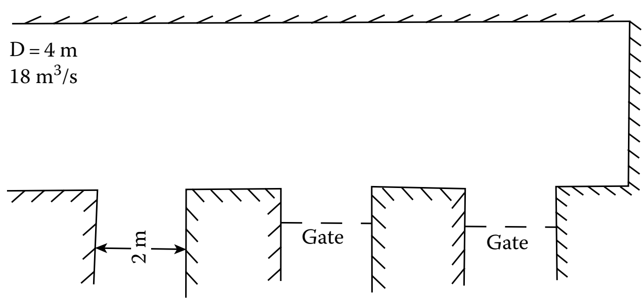
3.24 As requested in the previous problem develop an iterative solution for the subcritical or supercritical depth associated with a given value for the momentum function M, except have it apply for a circular section, rather than a trapezoidal channel. In other words obtain iterative solutions to the cubic equation for a rectangular channel, so that each subsequent iteration causes this solution to give the appropriate depth for a circular channel. Test you solution method by solving for both the sub- and supercritical depth associated with a flow rate of Q = 200 cfs in a D = 10 ft diameter pipe if the momentum function M = 100 ft^3.

3.25 The dimensionless momentum equation (Equation 3.21) for a rectangular channel is obtained by dividing the depth by the critical depth Y_c rather than by the bottom width b and multiplying by m as was done to obtain the dimensionless equation (Equations 3.15 or 3.16) for a trapezoidal channel. Obtain a dimensionless equation for a trapezoidal channel by dividing Y by Y_c and define the dimensionless momentum function by dividing by the critical depth cubed, or $M' = M/Y_c^3$. Note this dimensionless equation contains the side slope m and the bottom with b, as well as Y_c and therefore fails to satisfy one of the purposes for nondimensionalizing an equation, namely to remove channel sizes from the equation. Therefore, this equation lacks the practical applications of Equations 3.15 or 3.16. Prove that with the side slope m = 0 in this equation that it reduces to the momentum equation for a rectangular channel. Also develop a solution to this fifth degree polynomial equation that extracts all five of its roots. What are the roots of this equation if Q = 400 cfs, b = 10 ft, m = 1 and the momentum function M = 300 ft^3?

3.26 Modify Equation 3.19, which gives the dimensionless momentum function M′ for a circular channel so that rather than involving the dimensionless flow rate $Q' = Q^2/(gD^5)$ it contains the angle associated with critical depth, i.e., $\beta_c = \cos^{-1}(1-2Y_c')$. Use this equation to generate a series of tables for different values of β_c (which of course has a critical depth associated with it since $Y_c' = 0.5(1-\cos\beta_c)$) that provide the dimensionless depths Y′ = Y/D (and associated values of angle β) and the associated values of the dimensionless momentum functions $M' = M/D^3$. If the data from these separate tables were plotted, a graph similar to Figure 3.2 would be produced, the difference being that rather than the separate curves being associated with a Q′, they would be associated with a given critical value of angle β_c or a dimensionless critical depth Y_c'.

3.27 A wave-making device in a rectangular channel of 2 ft width consists of a vertical plate that can be moved against the still water in the channel by a mechanical drive mechanism. If the plate moves at a speed of 2 fps and the depth of water in the channel is 6 ft, determine the height of the wave and the speed of its movement. What force is required to drive the plate forward?

3.28 At the end of a circular channel with a diameter D = 8 m, that carries a discharge of Q = 18 m^3/s, there are three identical rectangular side channels that take an equal amount of water from the circular channel at right angles from the direction of its flow. The gate in the first channel is wide open. The side channels are 2 m wide, and have a bottom slope of 0.001, and a Manning's n = 0.013. The bottoms of the side channels are 1 m above the bottom of the pipe. What are the depths in the main channel between each of the side channels, and the depth upstream from the first side channel and downstream from the last side channel. What force is needed to cause the diversion into each side channel? Explain how this force is developed. (Ignore minor losses.)



3.29 If the depth of the water in the Problem 3.27 is 4 ft, determine the speed at which the plate should move to create a wave with a speed of 6 fps.

3.30 A smooth hump of 0.5 ft exists in the bottom of a 10 ft wide rectangular channel. If the upstream depth is 5 ft and the flow rate is Q = 100 cfs, determine the force on the hump. Develop an equation that gives this force in general.

3.31 Flow upstream from a gate in a rectangular channel with b = 15 ft is at a depth of 4 ft and a velocity of 3 fps. Suddenly the gate is completely closed. What is the depth at the gate and the speed of the surge?

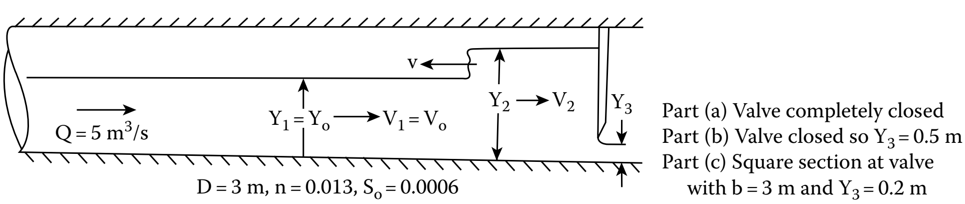
3.32 Flow in a very wide channel with $S_o = 0.0013$, and n = 0.012 is suddenly decreased from q = 12 cfs/ft to q = 3 cfs/ft. What is the new depth of flow and the velocity of the wave?

3.33 A gate is used to control the flow rate in a channel. Upstream from the gate the channel has a bottom width b = 10 ft, and a side slope m = 1.5. At the gate a smooth transition changes the section to rectangular with b = 10 ft. The gate has been set 4 ft above the channel bottom for a long time, and the flow rate under this setting has been measured to be Q = 400 cfs. Suddenly the gate is lowered to a height 3 ft above the channel bottom. Under the assumption that the depth upstream of the gate is constant compute (a) the new depth upstream from the gate, (b) the speed at which the surge will move upstream, (c) the flow rate past the gate, (d) the force on the gate prior to being closed to the new position, and (e) the force on the gate after being closed to the new position. The contraction coefficient for the gate is $C_c = 0.6$. Ignore minor losses.

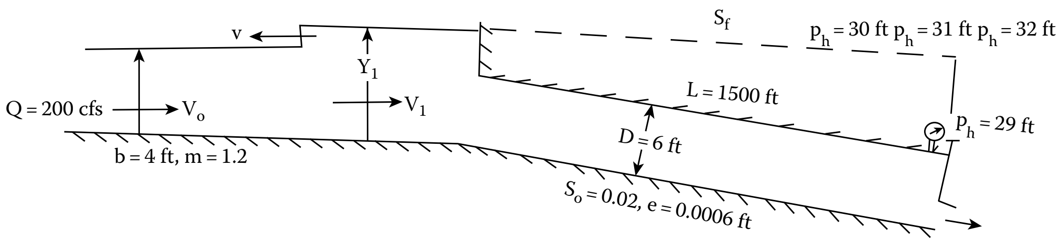
3.34 Initial conditions are as in the previous problem, but the gate is suddenly closed to a position 1 foot above the channel bottom. Compute the same quantities asked for in the previous problem.

3.35 Initial conditions are as in Problem 3.33, but the gate is suddenly completely closed. Compute the same quantities asked for in Problem 3.33.

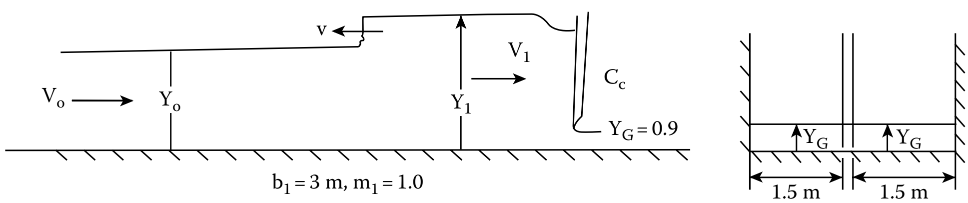
3.36 A pipe with a diameter D = 3 m is laid on a slope of $S_o = 0.0006$, and has a Manning's n = 0.013. If the flow rate in the pipe is Q = 5 m³/s, find the wave speed v and the depth Y_2 and velocity V_2 in the pipe upstream from a valve at its end if (a) the valve is instantly closed completely, (b) the valve is instantly closed to a position so $Y_3 = 0.5$ m above the bottom of the pipe, and (c) just before the valve a smooth transition changes the pipe to a square section with a width equal to D = 3 m, and the valve is closed to a position so $Y_3 = 0.2$ m above the bottom of the pipe.



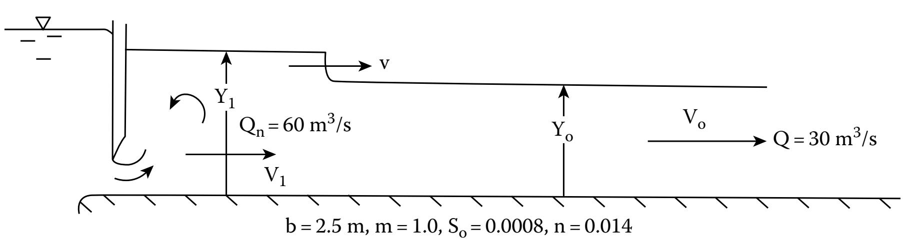
3.37 A channel with a bottom width of b = 4 ft, and a side slope m = 1.2 discharges into a pipe with a diameter D = 6 ft. The pipe has a bottom slope $S_o = 0.02$, and an equivalent sand roughness for use in the Darcy–Weisbach equation of e = 0.0006 ft. At a distance of 1500 ft a valve controls the flow and creates a pressure head of 29 ft on the top of the pipe. The loss coefficient is $K_L = 0.5$ between the channel and the pipe. If the flow rate is Q = 200 cfs determine what the depth is in the channel upstream of the pipe. Under the assumption that the depth you determine is constant throughout the channel determine: (a) the flow rate Q_n, (b) the new depth Y_1, and (c) the velocity v of a surge if the valve is adjusted so as to create a pressure head of $p_h = 30$ ft. Repeat the solution for these variables if $p_h = 31$ and 32 ft.



3.38 Two gates exist at the end of a trapezoidal channel with a bottom width $b_l = 3$ m, and a side slope $m_l = 1.0$. Each gate is 1.5 m wide, has a contraction coefficient $C_c = 0.6$, and initially both gates are open with their tips $Y_{G1} = Y_{G2} = 0.9$ m above the bottom of the channel. Suddenly one of the gates is completely closed. Determine how much the depth will increase upstream from the gates, and the speed of the surge that will result. The flow coming into the channel is $Q = 10\,m^3/s$.



3.39 A gate is releasing a flow rate of $Q = 30\,m^3/s$ into a trapezoidal channel with $b = 2.5$ m and $m = 1.0$, and $S_o = 0.0008$ from a reservoir and the channel downstream causes the gate to be submerged so the depth downstream of the gate equals the normal depth in the downstream channel. Suddenly the gate is raised so that the flow rate is doubled, e.g., is now $Q = 60\,m^3/s$. If the gate remains submerged what is the new depth downstream of the gate, and what is the speed of the surge that will travel downstream in the channel.



3.40 Derive Equation 3.33a from Equation 3.33, and thus show that the wave velocity v equals the sum of the upstream velocity V_1, plus the celerity c_2 times the square root of the average depth $(Y_1 + Y_2)/2$, i.e., v increases more rapidly than c_2 does with Y_2.

3.41 A flow rate per unit width of $q = 20$ cfs/ft occurs in a 10 ft wide rectangular channel with $n = 0.013$ and a bottom slope $S_o = 0.0005$ under uniform conditions. Suddenly a gate is closed completely at the downstream end of the channel. Determine the depth upstream from the gate and the velocity of the wave. Solve this problem by (a) simultaneously solving the continuity and momentum equations from the viewpoint of a moving observer, and (b) first solving only one implicit equation, Equation 3.37 (or 3.37a).

3.42 Resolve the previous problem with the gate shut only part way so that flow past the gate is one-half the original flow rate, or $q_2 = 10$ cfs/ft.

3.43 Solve Example Problem 3.9 using two simultaneous equations rather than one implicit equation followed by the explicit continuity equation to solve v.

3.44 Solve Example Problem 3.10 using two simultaneous equations rather than one implicit equation followed by the explicit continuity equation to solve v.

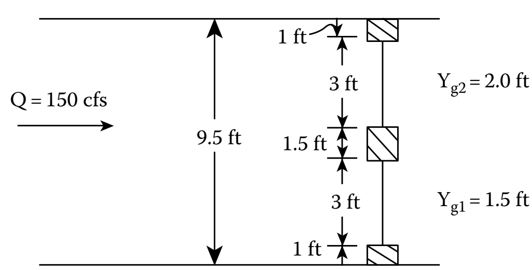
3.45 A gate at the upstream end of a trapezoidal channel with b = 3 m, m = 1, n = 0.014, and S_o = 0.0003, is initially set with its tip 0.5 m above the bottom of the channel. The flow behind the gate is submerged, and the gate causes a headloss equal to 1.1 times the velocity head in the jet coming from under the gate, where this velocity is computed by dividing the flow rate by the area corresponding to the gate's height times its contraction coefficient, which is C_c = 0.60. The water that flows into the channel comes from a constant head reservoir whose water surface elevation is 5 m above the bottom of the channel. Obtain a series of solutions that provides the submerged depth Y_2 downstream from the gate, the constant height wave speed v, and the velocity V_2 that will occur if the gate's height is suddenly increased from the initial setting of 0.5 to 0.75, 1.00, 1.25, 1.50, etc. m.

3.46 Flow in a steep pipe of 3 ft diameter is controlled downstream by a valve, so that at the valve the pressure at the top of the pipe is 40 psi. The slope of the pipe is S_o = 0.0855, its Manning's n = 0.013, and it contains a flow rate of 32.0 cfs. Determine the position upstream from the valve where the flow changes from open channel to closed conduit flow.

3.47 A square box conduit with a width and height of 2 ft contains a flow rate of 60 cfs. The roughness coefficient for this conduit is n = 0.014. The bottom slope changes from S_{o1} = 0.102 to S_{o2} = 0.00085. Determine where the flow becomes closed conduit flow if the channel ends with water at its top 1000 ft downstream from the change in grade.

3.48 A smooth transition takes place from a 4 m diameter pipe to a 0.5 m diameter pipe. The larger pipe has a bottom slope of S_{o1} = 0.015, and the smaller pipe has a slope S_{o2} = 0.250, and a length of 1000 m. At its end the water flows into a tank with a water surface elevation 10 m above the ground surface. (a) Determine where the flow changes from open channel to pipe flow if the flow rate Q = 0.3 m^3/s. (n = 0.013 for both pipes and for use in the Darcy–Weisbach equation for pipe flow use an equivalence roughness e = 0.001m.) (b) What flow rate will just cause the smaller diameter downstream pipe to flow full to its beginning? (c) If the flow rate is Q = 0.325 m^3/s where will the flow change from open channel to pipe flow?

3.49 Two vertical gates, each 3 ft wide are in a rectangular channel that is 9.5 ft wide. The pier between the gates is 1.5 ft wide. When a flow rate of Q = 150 cfs is occurring, the depth immediately downstream from gate #1 is 1.5 ft, and from gate #2 is 2.0 ft. Determine the following: (a) The amount of flow passing each gate. (b) The force against gate #1. (c) The force against gate #2. (d) The force against the middle pier. (State clearly any assumptions used to obtain this force.)



3.50 Write a program, model or spreadsheet to generate data so that a graph can be constructed that provides the supercritical dimensionless depth $Y_1' = mY_1/b$ on the ordinate as a function of the subcritical depth $Y_2' = mY_2/b$ on the abscissa for a trapezoidal channel, and then construct this graph. Have a curve on this graph for the following values of the dimensionless flow rate $Q' = m^3Q_2/(gb^5)$: 0.01,0.02,0.03,0.04,0.05,0.075,0.10,0.15,0.2,0.3,0.4,0.5,0.6,0.8 and 1.

3.51 Repeat the previous problem except make the graph for a circular channel with $Y_1' = Y_1/D$, $Y_2' = Y_2/D$ and $Q' = Q^2/(gD^5)$.

3.52 A gate in the main channel is used to control the amount of flow that leaves the main channel, and is conveyed away by a side channel that runs at an angle 90° from the main channel. The main channel is trapezoidal in shape with b = 5 m, and m = 1.8. The side channel is circular with D = 3 m, and has a bottom slope $S_{o2} = 0.00125$, and a Manning's n = 0.014 and is very long. Its bottom is 2 m above the bottom of the main channel, If the main channel is carrying a flow rate of Q = 100 m³/s, and it is desirable to divert a flow rate 10 m³/s in the side channel determine how far above the bottom of the channel the gate should be set. Its contraction coefficient is $C_c = 0.56$. What is the force against the gate? What is the force against the side channel wall that is required to divert the water?

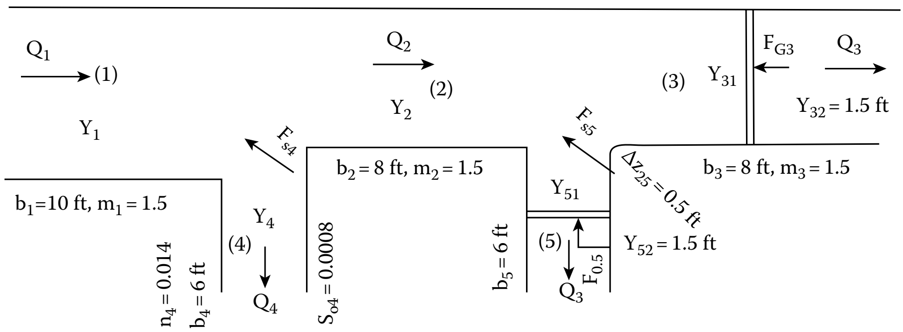
3.53 A sewer pipe of 12 in. diameter and a roughness coefficient e = 0.005 in. for use in the Darcy–Weisbach equation discharges into a sewerage treatment pond whose water surface elevation is 2 ft above the top of the pipe. The pipe has a constant slope of $S_o = 0.025$ for a long distance. For use in Manning's formula assume n = 0.013. Determine the position where the flow changes from open channel to pipe flow for flow rates of: (a) Q = 3 cfs, (b) Q = 4 cfs, and (c) Q = 5 cfs. What is the major factor that causes the flow to move upstream with increasing flow rates?

3.54 For Example Problem 3.12, determine how the flow rate Q, the upstream depth Y_1, and the downstream depth Y_2, etc. vary as the position x changes where the flow changes from open channel to closed pipe flow. Change the position x from 0 ft to just beyond 20 ft where a solution of the governing system of equations is no longer possible. (*Note*: The solution fails at about 21.875 ft.)

3.55 Compare the headlosses that occur in a rectangular and circular channel across a hydraulic jump (or if the jump causes pipe flow in the circular channel), if the flow rate is Q = 400 cfs, and the diameter of the pipe is 8 ft. For this comparison let the rectangular channel have a width so that it has the same area as the circle when the depth is 8 ft in the rectangle. Have the upstream depth vary from 5 to 2 ft. Also note when the jump hits the top of the circular channel.

3.56 Repeat the previous problem but increase the flow rate from Q = 400 cfs to Q = 800 cfs, and have the upstream depths vary from 7.8 to 2.4 ft.

3.57 At a certain position in a trapezoidal channel it has two side channels that take water in directions 90° from the direction of the main channel. The first such side channel is rectangular with a bottom width of $b_4 = 6$ ft, $n_4 = 0.014$, and $S_{o4} = 0.0008$. This channel is very long. The second side channel is also rectangular, is controlled by a gate that produces a depth of $Y_{52} = 1.5$ ft downstream from the gate. Upstream the main channel has a bottom width, $b_1 = 10$ ft, a side slope $m_1 = 1.5$, and after the first branch channel the width reduces to 8 ft so that here $b_2 = 8$ ft, $m_2 = 1.5$. Downstream from the second side channel the main channel is controlled by a gate that produces a depth downstream from the gate of $Y_{32} = 1.5$ ft. The second side channel has its bottom raised by $\Delta z_{25} = 0.5$ ft. The flow rate in the upstream main channel is $Q_1 = 500$ cfs. Ignoring all minor loss coefficient determine the following four forces: The force on the channel structure between the upstream main channel and the first side channel, F_{s4}, the force on the channel structure upstream and downstream from the second side channel, F_{s5}, the force on the gate in channel 3, F_{G3} and the force on the gate in channel 5, F_{G5}. Before obtaining these forces you will find it necessary to compute the flow rates Q_2, Q_3, Q_4, Q_5, and the depths Y_1, Y_2, Y_{31}, Y_4, and Y_{51}.



Solution: $F_{s4} = 6324.2$ lb to the right and upward at 63.13° from the horizontal, $F_{s5} = 4848.1$ lb to the right and upward at 78.19° from the horizontal, $F_{G3} = 4071$ lb, and $F_{G5} = 722.0$ lb.

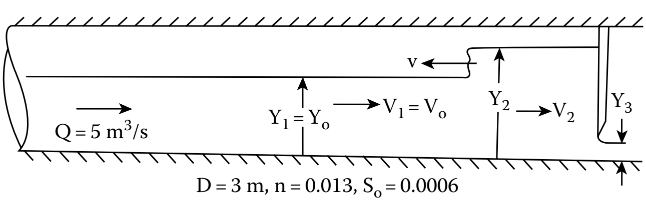
3.58 If the gate is raised in channel # 3 of the previous problem so that it produces a depth of 1.7 ft downstream from it, what will the forces, and flow rates become in this problem if the upstream flow rate Q_1 is maintained at 500 cfs. What occurs if the gate in channel 3 is raised further so that it attempts to produce a depth of 2.0 ft downstream?

3.59 If the gate is raised in channel # 5 of Problem 3.57 so that it produces a depth of 2.0 ft downstream from it, what will the forces, and flow rates become in this problem if the upstream flow rate Q_1 is maintained at 500 cfs.

3.60 For the same channel configuration as given in Problem 3.57 the depth of flow immediately upstream from the gate in channel # 3 is measured to equal 5.40 ft. Determine the same forces asked for in Problem 3.57 as well as all the flow rates and depth, including the flow rate Q^1 in the main upstream channel.

3.61 Obtain a series of solutions to Example Problem 3.12 in which the bottom slope S_{o1} of the upstream channel varies.

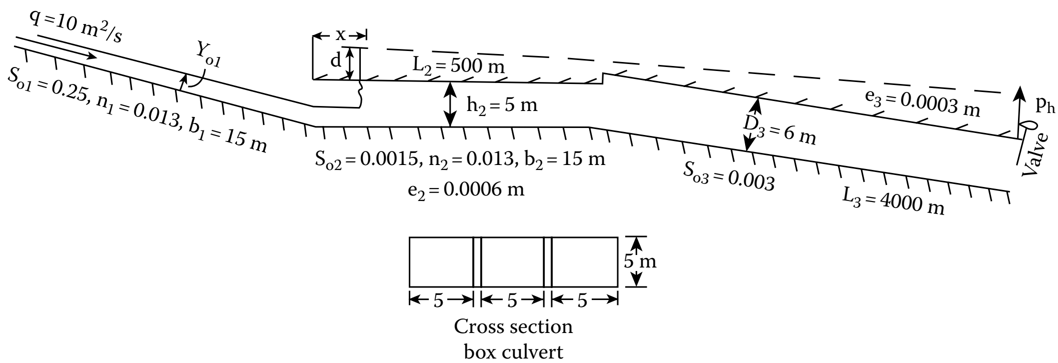
3.62 A $D_1 = 12$ ft diameter pipe which is laid on a slope of $S_{o1} = 0.20$ (with $n_1 = 0.013$ and $e_1 = 0.0005$ ft) connects into a pipe with a diameter $D_2 = 8$ ft, which is laid on as lope of $S_{o2} = 0.04$ (with $e_2 = 0.0005$ ft). This second pipe is $L_2 = 500$ ft long and discharges into a reservoir with a water surface elevation $H_2 = 13$ ft above the bottom of the pipe. The local loss coefficient $K_L = 0.5$ for the transition between the two pipes. If the flow rate is $Q = 1800$ cfs, determine the position x upstream from the junction of the two pipes where a modified hydraulic jump will occur.



3.63 Solve for the flow rate Q in the previous problem if the modified hydraulic jump occurs at x = 50 ft upstream from the position where the pipe diameters change.

3.64 Solve the previous problem, except that the upstream pipe has a diameter $D_1 = 10$ ft. Can you explain why reducing the pipe diameter allows a larger flow rate?

3.65 A flow rate per unit width of $q = 10$ m^2/s is coming down a steep rectangular spillway that is 15 m wide, and has a bottom slope of $S_{o1} = 0.25$, and $n_1 = 0.013$. A box culvert occurs at the end of the spillway with a width of 15 m, and a height of 5 m, as shown in the sketch, with two vertical partitions spaced at a distance of 5 m. This box culvert is 500 m long, when it changes into a 6 m diameter pipe.



The box culvert has a bottom slope $S_{o2} = 0.0015$, (with $n_2 = 0.013$ and $e_2 = 0.0006$ m), and the pipe has a bottom slope $S_{o3} = 0.003$ (with $e_3 = 0.0004$ m). The pipe is 4000 m long where its flow is controlled by a valve. If the pressure head created by the valve on the top of the pipe is $p_h = 2$ m, where will a modified hydraulic jump occur within the box culvert? (Ignore that a gradually varied flow will actually increase the depth in the first portion of the box culvert where the flow is supercritical upstream from the jump, e.g., assume up to the jump in the box culvert the depth remains at Y_{o1}.) The box culvert has a bottom slope $S_{o2} = 0.0015$, (with $n_2 = 0.013$ and $e_2 = 0.0006$ m), and the pipe has a bottom slope $S_{o3} = 0.003$ (with $e_3 = 0.0004$ m). The pipe is 4000 m long where its flow is controlled by a valve. If the pressure head created by the valve on the top of the pipe is $p_h = 2$ m, where will a modified hydraulic jump occur within the box culvert? (Ignore that a gradually varied flow will actually increase the depth in the first portion of the box culvert where the flow is supercritical upstream from the jump, e.g., assume up to the jump in the box culvert the depth remains at Y_{o1}.)

3.66 Solve Example Problem 3.13 except that the smaller diameter pipe has a 2.0 ft diameter (rather than 1.5 ft), and is 1000 ft long (rather than 1200 ft). It delivers the same pressure head of 90 ft at its end.

3.67 Solve Example Problem 3.13 except in place of the stilling basin and overflow weir there is a long trapezoidal channel with b = 8 ft, m = 1, n = 0.014, and $S_o = 0.0007$ into which the 5 ft diameter pipe discharges.

3.68 For a compound channel verify that the flow rate in the main channel is given by,

$$Q_m = \frac{Q_t K_m}{(K_m + K_s)}$$

in which K are conveyances defined from Manning's formula as $K = A(A/P)^{2/3}/n$ and subscript m and s apply for the main channel and side channels, respectively. Q_t is the total flow rate. The assumption in the verification of the equation is that the slope of the energy lines S_f of the main and side channels are the same.

3.69 If a flow rate of $Q = 300\,m^3/s$ occurred in the compound channel used as an example in the text ($b_m = 9$ m, $b_r = 15$ m, $b_l = 15$ m, $m_m = 0.5$, $n_m = 0.018$, and $n_r = n_l = 0.055$) what will the flow rate be in the main channel and in the side channels under the assumption that the side of the main channel portion of the flow extends upward with the side slope of 0.5 for a depth of Y = 5.4 m. (In other words verify the results provided in the table given in the text.)

3.70 Repeat the previous problem but assume that the main channel portion of the flow extends vertically upward from where its banks end.

3.71 For the compound channel in the text ($b_m = 9$ m, $b_r = 15$ m, $b_l = 15$ m, $m_m = 0.5$, $n_m = 0.018$, and $n_r = n_l = 0.055$) for a flow rate of $Q = 300\,m^3/s$ and a downstream depth of 5.6 m determine the depth upstream and the upstream Froude number that would result in a hydraulic jump. Carry out these computations: (a) assuming one-dimensional hydraulic equations are valid, and the channel downstream from the hydraulic jump can be handled as a compound channel, and (b) if the hydraulic jump occurred only in the main channel portion.

3.72 Determine the normal and critical depths in a compound channel with the following measurements if the flow rate in this channel is $Q = 400$ cfs, $b_m = 4$ ft, $m_m = 1.2$, $b_r = 2$ ft, $m_r = 0.5$, $b_l = 3$ ft, $m_l = 0.5$, $n_m = 0.013$, $n_r = n_l = 0.022$, $S_o = 0.00085$ and the height of the main channel is 4 ft.

3.73 Determine the normal and critical depths in a compound channel with the following measurements if the flow rate in this channel is $Q = 3400$ cfs, $b_m = 15$ ft, $m_m = 1.0$, $n_m = 0.020$, $b_r = 35$ ft, $m_r = 0.5$, $b_l = 40$ ft, $m_l = 0.5$, $n_m = 0.015$, $n_r = n_l = 0.042$, $S_o = 0.00085$, and the height of the main channel banks before the water flow into the side channels is 10 ft. Based on both the assumptions that the sides of the main channel are vertically above the top bank, and extend upward with a side slope of 1, determine what portion of the total flow rate will be within the main channel extended to the water surface and what portion will be in the side channels. Compute the kinetic energy correction coefficient under the assumption that the velocities in the three separate portions of the channel are constant. Compute the momentum functions under normal depth for: (a) the compound section, (b) the main channel flow, and (c) the side channel flow.

3.74 Write a computer program that will solve for any desired four variables associated with the flow in a compound channel. Assume the compound channel consists of a main channel and right and left side channels, all of which are trapezoidal in shape. The four equations available to solve the four unknowns are: (1) the continuity equation that requires that the total flow Q equal the sum of the flows in the main channel; (2) the right side channel and (3) the left side channel, or $F_1 = Q - Q_m - Q_r - Q_l = 0$; and (4) Manning's equations for the three component channels.

Solution: The listing for a FORTRAN program that does this is given below: This program is designed to first prompt for the input of all variables. For those variables that are later selected as the unknowns these given values will be used as the initial values to start the Newton iterative solution. The names used to prompt for the variables are as follows: Q = total flow rate; Q_m = flow rate in main channel; Q_r = flow rate in right side channel; Q_l = flow rate in left side channel; y = depth above top of main channel; b_m = bottom width; m_m = side slope of main channel; n_m = Manning's n for main channel; b_r = bottom width of right side channel; m_r = side slope of right side channel; n_r = Manning's n for right side channel; b_l = bottom width of left side channel; m_l = side slope of right side channel; n_l = Manning's n for right side channel; S_o = bottom slope of channels; Y_1 = depth to top of main channel, i.e., the depth is the sum of y and Y_1; g = acceleration of gravity to determine with SI or ES units are to be used; and Iv = parameter to determine whether the main channel is to extend upward to the surface vertically from its top width, or whether the main channel is to extend upward with the same side slope to the water surface. If vertically upward Iv = 1, otherwise IV = 0.

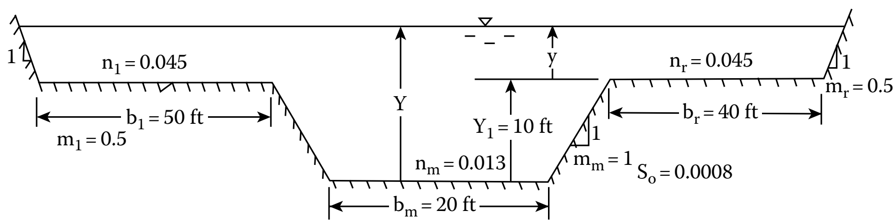
Listing of program, MANNCO1.FOR

```
      INTEGER*2 INDX(4),IUNK(4)
      REAL D(4,4),X(18),F(4),F1(4),mm,mr,ml,nm,nr,nl
      CHARACTER*2 CH(18)/'Q ','Qm','Qr','Ql','y','bm','mm','nm',
     &'br','mr','nr','bl','ml', 'nl','So','Y1','g ','Iv'/
      COMMON X,CC,A1,P1,Ivert
      EQUIVALENCE (Q,X(1)),(Qm,X(2)),(Qr,X(3)),(Ql,X(4)), (Y,X(5)),
     &(bm,X(6)),(mm,X(7)),(nm,X(8)),(br,X(9)), (mr,X(10)),
     &(nr,X(11)),(bl,X(12)),(ml,X(13)), (nl,X(14)),(So,X(15)),
     &(Y1,X(16))
      WRITE(*,*)' Give value to each variable'
      DO 5 I=1,18
      WRITE(*,110) I,CH(I)
110   FORMAT(I3,2X,A2,' = ',\)
```

```
      IF(I.EQ.18) WRITE(*,"(' 1 for vertically up ',\)")
5     READ(*,*) X(I)
      Ivert=X(18)+.1
      WRITE(*,*)' Give number of 4 unknowns'
      DO 6 I=1,16
6     WRITE(*,120) I,CH(I),X(I)
120   FORMAT(I3,2X,A2,F10.3)
      READ(*,*) IUNK
      IF(X(17).LT.30.) THEN
      CC=1.
      ELSE
      CC=1.486
      ENDIF
      Qm=.75*Q
      Qr=.125*Q
      Ql=Qr
      A1=(bm+mm*Y1)*Y1
      P1=bm+2.*Y1*SQRT(1.+mm*mm)
      NCT=0
10    CALL FUNCT(F)
      DO 20 I=1,4
      XX=X(IUNK(I))
      X(IUNK(I))=1.005*X(IUNK(I))
      CALL FUNCT(F1)
      DO 15 J=1,4
15    D(J,I)=(F1(J)-F(J))/(X(IUNK(I))-XX)
20    X(IUNK(I))=XX
      CALL SOLVEQ(4,1,4,D,F,1,DD,INDX)
      DIF=0.
      DO 30 I=1,4
      X(IUNK(I))=X(IUNK(I))-F(I)
30    DIF=DIF+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,' DIF=',DIF
      IF(NCT.LT.20 .AND. DIF.GT. .0004) GO TO 10
      WRITE(*,100) (IUNK(I),I=1,4),(I,CH(I),X(I),I=1,16),Y1+X(5)
100   FORMAT(' Solution to:'4I3,/,1X,24('-'),/, 16(2X,I2,1X,A2,'=',
     &F12.5,/),' 17 Depth =',F10.2)
      END
      SUBROUTINE FUNCT(F)
      REAL F(4),X(18),mm,mr,ml,nm,nr,nl
      COMMON X,CC,A1,P1,Ivert
      EQUIVALENCE (Q,X(1)),(Qm,X(2)),(Qr,X(3)),(Ql,X(4)),(Y,X(5)),
     &(bm,X(6)),(mm,X(7)),(nm,X(8)),(br,X(9)), (mr,X(10)),
     &(nr,X(11)),(bl,X(12)),(ml,X(13)), (nl,X(14)),(So,X(15)),
     &(Y1,X(16))
      IF(Ivert.EQ.1) THEN
      Am=A1+(bm+2.*mm*Y1)*Y
      Ar=.5*(2.*br+mr*Y)*Y
      Al=.5*(2.*bl+ml*Y)*Y
      ELSE
      Am=A1+(bm+2.*mm*Y1+mm*Y)*Y
      Ar=.5*(2.*br+(mr-mm)*Y)*Y
      Al=.5*(2.*bl+(ml-mm)*Y)*Y
      ENDIF
      Pr=br+Y*SQRT(1.+mr*mr)
```

```
Pl=bl+Y*SQRT(1.+ml*ml)
F(1)=Q-Qm-Qr-Ql
F(2)=nm*Qm-Am*(Am/P1)**.666666667*CC*SQRT(So)
F(3)=nr*Qr-Ar*(Ar/Pr)**.666666667*CC*SQRT(So)
F(4)=nl*Ql-Al*(Al/Pl)**.666666667*CC*SQRT(So)
RETURN
END
```

3.75 For the compound channel shown in the sketch below with a main channel bottom width $b_m = 20$ ft, and side slope $m_m = 1$, and a Manning's $n_m = 0.013$, with a right side channel with $b_r = 40$ ft, $m_r = 0.5$, and $n_r = 0.045$, a left side channel with $b_l = 50$ ft, $m_l = 0.5$, and $n_l = 0.045$, and a bottom slope $S_o = 0.0008$, determine the flow rates in all component channels for depths varying from 0.5 ft above the top of the main channel to a depth of 10 ft above the main channel, in increments of 0.5 ft. The height of the main channel is $Y_1 = 10$ ft. Solve for these flow rates: (a) assuming that the main channel extends vertically upward from its outer banks to the water surface, and (b) that it continues with the side slope of $m_m = 1$ to the water surface. Plot the 4 flow rates against the depth of flow in the compound channel.



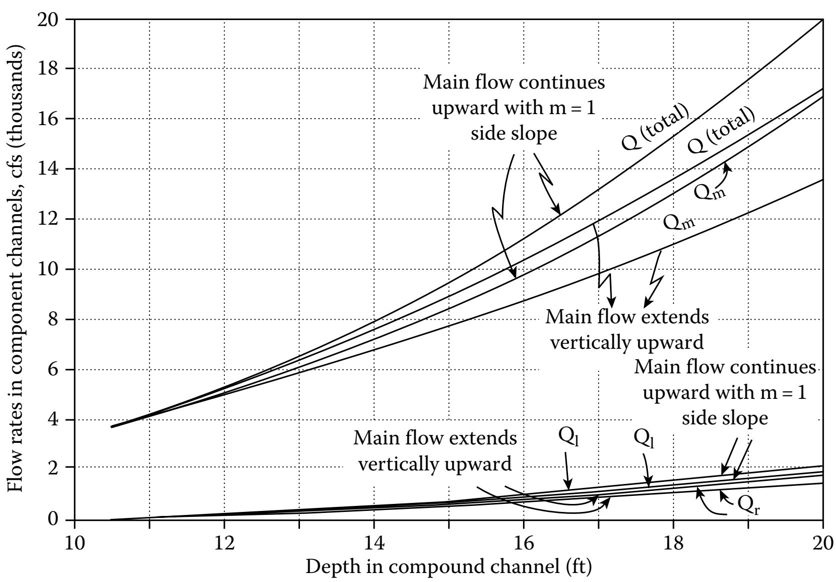
Solution: Solving Manning's equation for the main channel with a depth of 10 ft gives Q = 3278.1 cfs. The solution obtained by using a program such as that in the previous problem are given in the tables below.

Main Channel Consist of Area Vertically Above

y (ft)	Y (ft)	Q (cfs)	Q_m (cfs)	Q_r (cfs)	Q_l (cfs)
0.5	10.5	3676.72	3650.34	11.72	14.66
1.0	11.0	4121.92	4038.45	37.06	46.40
1.5	11.5	4605.58	4442.09	72.57	90.92
2.0	12.0	5124.16	4860.97	116.78	146.41
2.5	12.5	5675.32	5294.81	168.78	211.74
3.0	13.0	6257.34	5743.35	227.90	286.09
3.5	13.5	6868.87	6206.37	293.65	368.84
4.0	14.0	7508.76	6683.64	365.62	459.50
4.5	14.5	8176.07	7174.95	443.48	557.64
5.0	15.0	8869.95	7680.10	526.94	662.92
5.5	15.5	9589.68	8198.90	615.76	775.02
6.0	16.0	10334.59	8731.18	709.72	893.69
6.5	16.5	11104.09	9276.77	808.64	1018.68
7.0	17.0	11897.64	9835.50	912.36	1149.78
7.5	17.5	12714.76	10407.24	1020.71	1286.81
8.0	18.0	13554.98	10991.82	1133.57	1429.59
8.5	18.5	14417.89	11589.11	1250.82	1577.97
9.0	19.0	15303.10	12198.97	1372.33	1731.80
9.5	19.5	16210.26	12821.28	1498.02	1890.95
10.0	20.0	17139.01	13455.92	1627.79	2055.30

Main Channel Continues with m_m = 1

y (ft)	Y (ft)	Q (cfs)	Q_m (cfs)	Q_r (cfs)	Q_l (cfs)
0.5	10.5	3681.23	3655.09	11.60	14.54
1.0	11.0	4140.20	4058.27	36.30	45.63
1.5	11.5	4647.47	4488.46	70.34	88.68
2.0	12.0	5200.17	4946.55	112.01	141.61
2.5	12.5	5796.71	5433.41	160.21	203.10
3.0	13.0	6436.16	5949.93	214.09	272.14
3.5	13.5	7117.96	6497.02	273.00	347.94
4.0	14.0	7841.82	7075.57	336.38	429.86
4.5	14.5	8607.62	7686.50	403.78	517.34
5.0	15.0	9415.41	8330.71	474.80	609.90
5.5	15.5	10265.33	9009.14	549.07	707.12
6.0	16.0	11157.62	9722.71	626.29	808.62
6.5	16.5	12092.59	10472.36	706.18	914.05
7.0	17.0	13070.63	11259.03	788.48	1023.12
7.5	17.5	14092.17	12083.67	872.96	1135.54
8.0	18.0	15157.70	12947.23	959.41	1251.05
8.5	18.5	16267.74	13850.67	1047.65	1369.42
9.0	19.0	17422.86	14794.96	1137.48	1490.42
9.5	19.5	18623.66	15781.06	1228.75	1613.86
10.0	20.0	19870.79	16809.96	1321.30	1739.53

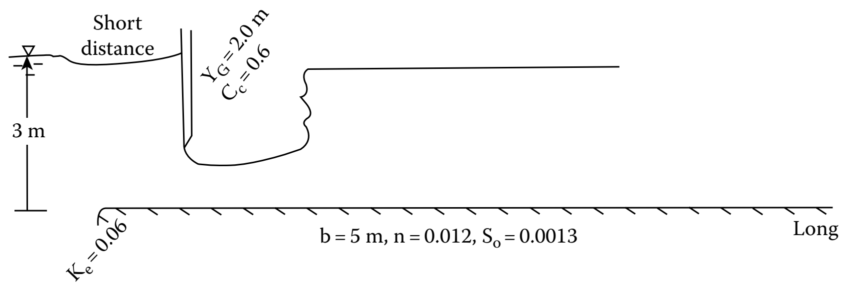


3.76 Repeat the previous problem except the side channels have ½ their widths, i.e., b_r = 20 ft, and b_l = 25 ft.

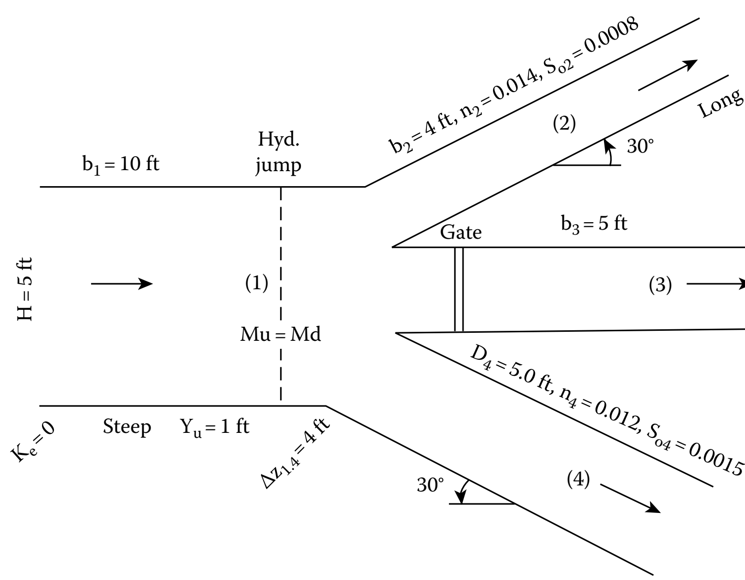
3.77 Same as Problem 3.75 except solve for the flow rates in the main channel, and the two side channels, and the total depth of flow if the total flow rate is Q = 6000 cfs. The properties of the channels are as given in Problem 3.75.

3.78 Repeat the previous problem except for side channels having ½ the widths, i.e., b_r = 20 ft and b_l = 25 ft.

3.79 A gate controls the flow of water into a long 5 m wide rectangular channel that has a Manning's n = 0.012, and a bottom slope S_o = 0.0013, as shown. The gate is a short distance downstream from the reservoir that supplies the flow with a head H = 3 m. If the gate is 2.0 m above the bottom of the channel, and its contraction coefficient is $C_c = 0.6$, and the entrance loss coefficient is $K_e = 0.06$ determine: (1) the flow rate Q, (2) the depths upstream and downstream from the hydraulic jump, (3) the force on the gate, (4) the force on the gate if the pressure distribution on it were hydrostatic, and (5) a channel roughness that would cause the gate to be submerged.



3.80 A steep 10 ft wide rectangular channel receives water from a reservoir with a water surface elevation that is 5 ft above the channel's bottom. This channel is steep, and it is observed that at the end of this channel where it branches into three channels that the depth upstream from the hydraulic jump that occurs here is $Y_u = 1$ ft. The downstream channels consist of: Channel # 2 is rectangular with a bottom width $b_2 = 4$ ft, a roughness coefficient $n_2 = 0.014$, and a bottom slope $S_{o2} = 0.0008$; Channel # 3 is rectangular with a bottom width $b_3 = 5$ ft and it contains a gate a short distance downstream from the junction; Channel # 4 is circular with a diameter D = 5.0 ft, has a roughness coefficient $n_4 = 0.012$, a bottom slope $S_{o4} = 0.0015$, and its beginning is $\Delta z_{14} = 4$ ft above the bottom of the other channels at the junction. Channels # 2 and # 4 each branch off at an angle of 30° from Channel # 1, and Channel # 3 is in the same direction as Channel # 1. Determine the flow rates in all of the channels, and the depth downstream from the gate in Channel # 3. Also determine the force the fluid applies against the junction structure (including the gate). (Ignore all local losses.)



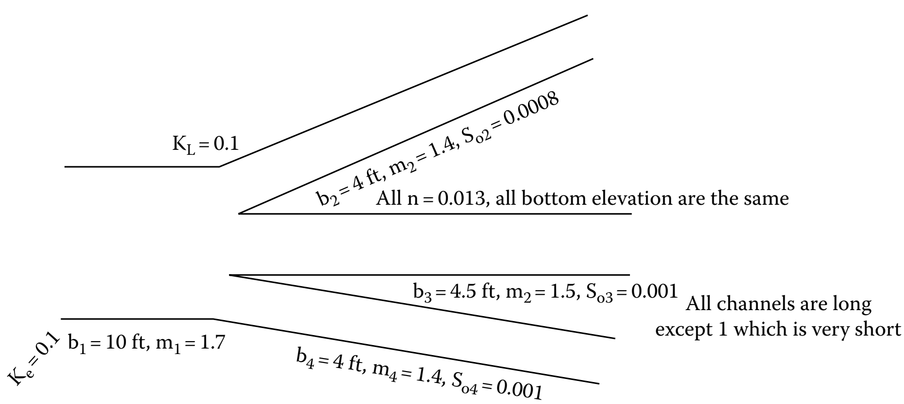
3.81 A trapezoidal channel with b = 3 m, m = 1.4, n = 0.013, and S_o = 0.0008 has a circular section with a diameter D = 5 m at its entrance where it receives water from a reservoir with a head H = 4 m. The entrance loss coefficient is K_e = 0.1. Determine the flow rate Q into the channel, and if critical flow controls at its circular entrance section, determine how much headloss a hydraulic jump dissipates in the transition between the circular and trapezoidal sections, if this occurs.

As a second part to this problem locate the position of the hydraulic jump based on the following assumptions: (1) The specific energy in the transition upstream from the jump is constant and equals E_c at the entrance to the reservoir, (2) The specific energy in the transition downstream from the jump is constant and equals E_o, the specific energy associated with the normal depth Y_o in the trapezoidal channel, (3) The shape of the transition is such that the area and the first moment of area Ah_c vary linearly from the circular to the trapezoidal shapes, i.e., $A = (1 - x)A_{cir} + xA_{trap}$ in which x is the fraction of the distance across the transition, and (4) To account for the force on the enlarging control volume for the jump add 4.2 times the depth immediately downstream of the jump, Y_d, to the downstream side of the momentum equation, or $(Ah_c)_u + Q^2/(gA_u) = (Ah_c)_d + Q^2/(gA_d) + 4.2Y_d$.

Where will the hydraulic jump occur if the area and first moment of area vary as the square root of the fraction of the distance across the transition, i.e., $A = (1 - \sqrt{x})A_{cir} + \sqrt{x}A_{trap}$.

3.82 In Problem 2.72 you were to develop the delivery diagrams for several bottom slopes for a trapezoidal channel with b = 3 m, m = 1.4, and n = 0.013 with a circular section with D = 5 m at its entrance to a reservoir, with K_e = 0.1 (the channel of the previous problem), for several different bottom slopes S_o. Take the delivery diagram for S_o = 0.0008 (with H varying from 0.25 to 5 m), and for each of these H versus Q entries compute: (1) whether a hydraulic jump will occur in the transition from the circular to the trapezoidal sections and if a jump occurs what headloss it must create for uniform flow to exit in the downstream trapezoidal channel, and (2) the slope the downstream trapezoidal channel must have so the flow will remain supercritical through the transition and into the trapezoidal channel.

3.83 The reservoir head H that supplies the 4 channel system shown below varies from H = 0.5 ft to H = 10 ft. Do the following for a series of H′ values: (1) Solve for the flow rates and depths in all four channels (i.e., develop the delivery diagrams). If critical flow occurs at the entrance, i.e., in Channel 1, assume that the specific energy of the three downstream channels are the same. (2) For those H′ values for which critical flow occurs in Channel 1 determine the energy loss between the upstream Channel 1 and the downstream channels. (3) For the H′ values involved in (2) assume that the supercritical flow exists into the beginning of each of the three downstream channels, and compute this depth under the assumption that the specific energy within the supercritical region is constant. (4) Compute the force that the branching structure must apply to the fluid within the region of the hydraulic jump for each of the downstream branched channels.



PROBLEMS TO SOLVE USING PROGRAM CHANNEL

Now that you developed your skills in solving the implicit equations that are associated with Open Channel flow, you will be provided computer program, CHANNEL that will solve these problems for you. In using CHANNEL it is the basic principle from open channel flow, such as UNIFORM Flow, ENERGY, MOMENTUM, CRITICAL, that identifies the type of problem you want solved. The first time you use CHANNEL ask for **Help**, study the document produced, and then solve the following 10 problems using channel. You should verify enough answers to be confident that the answers are correct. For the following 10 problems write down the answer(s) or have the answer screen from CHANNEL printed by using the Print Screen key on your PC. (The program CHANNEL will either be provided to you by your instructor, or it will be on the diskette that was distributed with this book. Ask your instructor about the details in how to execute program CHANNEL if he provides it for your use.)

1. Find the uniform depth of flow that will exist in a pipe with an 8 ft diameter if its bottom slope is 0.001, its Manning's roughness coefficient is 0.012, and it contains a flow rate of 100 cfs.
2. A transition from a pipe with a diameter of 2 m to one with a diameter of 2.5 m occurs. If the depth upstream is 1.0 m, and the flow rate is 1.5 m^3/s, what is the depth downstream and what is the change in the water surface elevation? The transition loss coefficient equals 0.1.
3. A transition takes a trapezoidal channel with $b_1 = 10$ ft, and a side slope $m_1 = 1.2$ to a rectangular section with $b_2 = 8$ ft, and the bottom rises 0.3 ft. What is the depth downstream and the change in the water surface elevation if the flow rate is Q = 250 cfs, and the upstream depth is $Y_1 = 5$ m. The transition loss coefficient equal 0.1.
4. Water enters a steep trapezoidal channel with b = 10 ft and m = 1.5 from a reservoir with a water surface elevation of 4 ft above the channel bottom. What is the flow rate, and the depth of flow at the entrance ($K_e = 0.12$).
5. Water enters a mild channel with the size of Problem 3.4. The slope of the channel bottom is $S_o = 0.0005$, and Manning's n = 0.013. ($K_e = 0.12$). Find the flow rate and the depth of flow.
6. A gate valve exists in an 8 ft diameter pipe that is flowing as an open channel with Q = 90 cfs. Downstream from the valve the depth of flow is $Y_2 = 2$ ft. Determine the force of the water against the valve.
7. Water is entering a long pipe with a bottom slope of $S_o = 0.0006$, n = 0.012, a diameter D = 6 ft from a reservoir whose water surface is H = 5 ft above the bottom of the pipe. The entrance loss coefficient $K_e = 0.2$. Determine the flow rate and the depth in this pipe. Solve the same problem except the pipe is laid on a steep slope.
8. Determine the bottom width that a trapezoidal channel should have if it is to carry a flow rate of Q = 300 cfs, is to have a bottom slope $S_o = 0.0009$, a roughness coefficient n = 0.013, and a side slope m = 1.5, and if the water surface of the reservoir that supplies the channel is H = 5 ft above its bottom ($K_e = 0.02$). Repeat this problem except that the channel has a bottom slope that is steep.
9. Generate the stage discharge relationship that gives the depth and flow rate that a reservoir will supply a trapezoidal channel if b = 10 ft, m = 1.5, $S_o = 0.001$, and n = 0.015 for depths of water in the reservoir from H = 2 ft to H = 8 ft in increments of 0.5 ft ($K_e = 0.2$).
10. Generate a table of values that gives the bottom width of a trapezoidal channel that will carry flow rates ranging from 200 to 1000 cfs in increments of 100 cfs if the following hydraulic properties of the channel are to be maintained: the side slope m = 1.3, the roughness coefficient n = 0.013, and the slope of the channel bottom $S_o = 0.00085$. For all of these cases the water is supplied by a reservoir whose water surface is H = 5 ft above the channel bottom, and the entrance loss coefficient is $K_e = 0.09$.

Whenever you need to solve a problem that CHANNEL is capable of solving during the rest of this course you should feel free to use CHANNEL, but remember you are still responsible for knowing whether the results are correct.

11. Use Chezy's equation to determine the flow rate that a trapezoidal channel with a bottom width b = 10 ft, a side slope m = 1.5, a bottom slope S_o = 0.001, and a equivalent sand roughness of e = 0.012 ft, if it is at a depth of Y = 5 ft under uniform flow conditions. (Also what is Chezy's C and the Reynolds number of the flow?)
12. Determine the depth of flow needed to convey a flow rate Q = 400 cfs in a trapezoidal channel with b = 8 ft, m = 1.3, S_o = 0.0005 if its equivalent wall roughness is e = 0.01 ft.
13. Determine the depth Y in a circular channel with a diameter D = 12 ft that will occur when 300 cfs is flowing. The channel has a bottom slope S_o = 0.0006, and an equivalent wall roughness e = 0.014 ft. (Also what is Chezy's C and the Reynolds number of the flow?)
14. What size circular channel with e = 0.009 ft is needed to carry a flow rate of Q = 20 m^3/s if the bottom slope S_o = 0.00085, and the equivalent wall roughness is e = 0.011 ft.
15. The cross section of a natural channel is defined by the transect data in the table below. The wall roughness is defined by Manning's n = 0.015, and the bottom slope is S_o = 0.00085. If the depth of flow is Y = 5 ft, what is the flow rate under uniform conditions?

Pt	x (ft)	y (ft)
1	0	8.0
2	5	6.5
3	8	5.0
4	10	3.2
5	13	2.8
6	17	3.1
7	22	5.3
8	25	7.0
9	30	8.0

16. For the natural channel defined in the previous problem determine the depth of water under uniform flow conditions if the flow rate is Q = 350 cfs. Determine this depth using both linear and quadratic interpolation of the cross-sectional data.
17. Using the cross-sectional data of the previous problem determine the depth if rather than using Manning's equation, it is modified so that the exponent e_1 of the hydraulic radius is 0.715, and the exponent e_2 of the slope of the channel bottom is 0.35. For this problem use a bottom slope S_o = 0.0018 and a Manning's n = 0.045. (Use C_u = 1.486.)
18. Use e_1, e_2, S_o and n as in the previous problem, the difference is that the channel has a trapezoidal cross section with b = 10 ft, and m = 1. Determine the flow rate Q if the depth of flow is Y = 5 ft.
19. Using the geometry solving capability of channel to solve the variables left blank in the table below if the other variables have the values given for a trapezoidal channel.

Depth (ft)	Bottom Width (ft)	Side Slope	Perimeter (ft)	Top Width (ft)	Area (ft^2)	1st Mom. of A (ft^3)
5	10	1.2				
	8	1.5			70	
4		1.0			50	

4 Nonuniform Flows

4.1 TYPES OF NONUNIFORM FLOWS

In Chapter 1, it was noted that nonuniform flows in open channels are subdivided into gradually varied and rapidly varied on the basis of whether normal accelerations of fluid moving along the curved streamlines can be ignored or not. Situations, such as the flow directly underneath a gate, and over the crest of a weir or a dam are examples of rapidly varied flows. For such situations, the stream lines curve rapidly and normal accelerations have a significant influence on the flow pattern, i.e., the flow needs to be handled as a two- or three-dimensional problem. When the water depth and the velocity change over long distances as water impounds behind a gate, or the depth gradually decreases in a mild channel upstream from a break in grade to a steeper channel, then normal accelerations are insignificant, and the flow can be considered one-dimensional. In this chapter, only gradually varied flows will be dealt with. Furthermore, only steady-state flows are considered in this chapter.

If the flow rate Q increases or decreases in the direction of the channel, then this special type of gradually varied flow is referred to as a **spatially varied flow**. The adjective "spatially" denotes that the flow rate changes in space, e.g., in the direction of the x coordinate. Alternative terminology calls for situations in which the flow rate changes along the channel, i.e., **lateral inflow** or **lateral outflow**. An example of lateral inflow is the gutter flow along a roadway during a rain storm. The runoff from the road crest feeds the gutter flow so that it increases in the flow direction until the gutter passes over a storm drain grate at which time a lateral outflow occurs from the gutter. A side weir, or overflow spillway along a canal side is a lateral outflow.

In the previous three chapters, the equations used to describe the problems were algebraic. When dealing with gradually varied flows, the governing equation is an ordinary differential equation (an ODE). Since this ODE can be solved in the closed form for only a limited number of simplified cases, much of the material in this chapter deals with numerical methods for obtaining approximate solutions to this governing equation. Rather than dealing with the numerical methods per se, the emphasis will be on the use of standard algorithms implemented into computer programs. The widespread use of computers in recent years has made some of the hand techniques that engineers have used in the past to solve gradually varied flow problems obsolete.

The general gradually varied flow equations will be developed first. The general case will include spatially varied flows in nonprismatic channels. After the general equations have been developed, solutions will be obtained for the simpler cases for gradually varied flows with no lateral inflow or outflow in a prismatic channel. For these simpler cases, gradually varied flows are classified, and this classification system will be described. After dealing with the simpler cases, more complex situations will be handled.

4.2 ORDINARY DIFFERENTIAL EQUATION FOR GRADUALLY VARIED FLOW

In developing differential equations for a gradually varied flow, the case of lateral outflow and lateral inflow will be different. Lateral outflow will be dealt with first.

4.2.1 Bulk Lateral Outflow

In dealing with a gradually varied flow in which there may be a lateral outflow from the channel, it is assumed that all the liquid at a given position x in the channel has the same energy per unit weight. Therefore, each unit weight of liquid that leaves the main channel flow carries with it the same energy as each unit weight of liquid that remains in the channel. The energy principle is the appropriate principle to utilize in describing this situation. From any selected horizontal data, the total head H will be constant at any position along the channel, or

$$H = z + Y + \frac{Q^2}{2gA^2} \tag{4.1}$$

For a nonuniform flow, this total head will vary from position to position, and this variation is defined as the derivative of H with respect to x, or

$$\frac{dH}{dx} = \frac{dz}{dx} + \frac{dY}{dx} + \frac{Q}{gA^2}\frac{dQ}{dx} - \frac{Q^2}{gA^3}\frac{dA}{dx}$$

If the channel is nonprismatic, then the cross section A is both a function of the position x and the depth Y at this position, i.e., A = f(x, Y), and therefore in general, by the chain rule of calculus,

$$\frac{dA}{dx} = \left.\frac{\partial A}{\partial x}\right|_Y + \left.\frac{\partial A}{\partial Y}\right|_Y \frac{dY}{dx}$$

in which the subscripts emphasize which variables are being held constant when the derivative is taken. The partial derivative $\partial A/\partial Y|_x$ equals the top width T. Should the channel be prismatic, e.g., the cross-sectional area is only a function of the depth Y, then $\partial A/\partial x = 0$. The term dQ/dx is the lateral outflow per unit length along the channel, and will have a negative magnitude. It will be defined as $q_o^* = -dQ/dx$. The derivative dH/dx is the negative slope of the energy line, and this slope will be denoted by S_f, or $S_f = -dH/dx$. Also, the derivative dz/dx is the negative slope of the channel bottom, or $S_o = -dz/dx$. Substitution of these results into the above equation produces the following:

$$-S_f = -S_o + \frac{dY}{dx}\left(1 - \frac{Q^2T}{gA^3}\right) - \frac{Q^2}{gA^3}\left.\frac{\partial A}{\partial x}\right|_Y - \frac{Q}{gA^2}q_o^* \tag{4.2}$$

The quantity of interest is the change in depth with respect to position, so Equation 4.2 will be solved for dY/dx, and F_r^2 (Froude number squared) will be substituted in place of $Q^2T/(gA^3)$ giving

$$\frac{dY}{dx} = \frac{S_o - S_f + \frac{Q^2}{gA^3}\left.\frac{\partial A}{\partial x}\right|_Y + \frac{Qq_o^*}{gA^2}}{1 - F_r^2} \tag{4.3}$$

Equation 4.3 is the general equation that describes a gradually varied flow in a channel that may have bulk lateral outflow q_o^*. If the channel is prismatic, and no lateral outflow occurs, then this equation reduces to

$$\frac{dY}{dx} = \frac{S_o - S_f}{1 - F_r^2} \tag{4.4}$$

Before examining the lateral inflow case, it is well to examine the term $\partial A/\partial x|_Y$ for a nonprismatic channel. An example of a nonprismatic channel is a trapezoidal channel going through a transition from b_1 and m_1 at section 1 to b_2 and m_2 at section 2. For this transitional part of the trapezoidal channel, the area is defined by

$$A = b(x)Y + m(x)Y^2 = (b(x) + m(x)Y)Y$$

taking the partial derivative with respect to x gives the following:

$$\frac{\partial A}{\partial x} = Y\frac{db}{dx} + Y^2\frac{dm}{dx}$$

The full derivatives of b and m are used above because these variables depend only on x, and therefore full and partial derivatives are identical. In other words, for a trapezoidal channel, the change in area with respect to x equals the depth times the change in the bottom width with respect to x plus the depth squared times the change in the side slope with respect to x.

For a circular section, the diameter may change through a transition in which case, the area $A = D^2(\beta - \cos\beta\sin\beta)/4$ is differentiated with respect to x to give

$$\frac{\partial A}{\partial x} = \frac{1}{2}D\frac{\partial D}{\partial x}(\beta - \sin\beta\cos\beta) + \frac{1}{4}D^2\frac{\partial \beta}{\partial x}(1 - \cos^2\beta + \sin^2\beta)$$

in which the partial derivative of β with respect to x is obtained by differentiation of $\cos\beta = 1 - 2Y/D$ to give

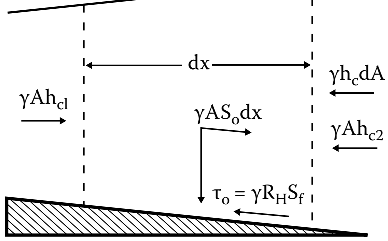
$$\frac{\partial \beta}{\partial x} = -\frac{2Y}{D^2\sin\beta}\frac{dD}{dx}$$

or the substitution of this result into the above equation gives the following for $\partial A/\partial x$:

$$\frac{\partial A}{\partial x} = \left\{\frac{1}{2}D(\beta - \cos\beta\sin\beta) - Y\sin\beta\right\}\frac{dD}{dx}$$

4.2.2 Lateral Inflow

In the case of a lateral inflow, it is necessary to use the momentum principle rather than the energy principle because the amount of energy dissipated as the incoming flow impacts with the main channel flow is unknown. The sketch below shows a control volume of a small length of channel Δx, which can be reduced to a differential length dx, with the forces shown on it.



Since these forces were not included in the development of the momentum function M, and the momentum function was obtained by dividing the summation of force equation by the specific weight γ of the fluid, a summation of forces in the x direction produces the following:

$$\gamma \frac{dM}{dx} dx = \gamma A S_o dx - \gamma R_h S_f P dx + \rho q^* U_q dx - \gamma h_c dA$$

in which

U_q is the component of velocity of the inflowing liquid in the direction of the main channel flow

V is the average velocity of the main channel flow

Differentiation of the momentum function $M = Ah_c + Q^2/(gA)$ gives

$$\frac{dM}{dx} = \frac{d}{dx}(Ah_c) + \frac{2Q}{gA}\frac{dQ}{dx} - \frac{Q^2}{gA^2}\frac{dA}{dx}$$

in which $dA/dx = T(dY/dx) + \partial A/\partial x|_Y$ and by Leibniz's rule $d(Ah_c)/dx = A(dY/dx)$. Upon substitution of these and $q^* = dQ/dx$ into the above equation, and solving dY/dx gives the following spatially varied flow equation for a lateral inflow:

$$\frac{dY}{dx} = \frac{S_o - S_f + \frac{Q^2}{gA^3}\frac{\partial A}{\partial x} - \frac{2Qq^*}{gA^2} + \frac{q^* U_q}{gA} - \frac{h_c}{A}\frac{\partial A}{\partial x}\Big|_Y}{1 - F_r^2} \tag{4.5}$$

4.2.3 Generalization of Gradually Varied Flow Equations

There is considerable similarity between Equations 4.3 and 4.5. The sign in front of the term containing Qq^* is different and a 2 now multiplies this term, but $q^* = -q_o^*$. Equation 4.5 contains a couple of additional terms in the numerator; one that accounts for the momentum flux for the incoming liquid $q^*U_q/(gA)$, and the other, $(h_c/A)(\partial A/\partial x)$ that accounts for the hydrostatic force of the fluid against the expanding or the contracting cross section for a nonprismatic channel.

In addition, there is one additional case that has not been considered, that does occur in channel flow, that of **seepage outflow or inflow**. If some of the fluid from a channel is lost from seepage into (or out from) the soil that forms the canal, then this fluid will leave from the boundary layer where it has a zero velocity. Therefore, fluid lost by seepage will have an energy per unit weight less by the velocity head than the fluid that remains in the channel. The following equation will accommodate all these cases:

$$\frac{dY}{dx} = \frac{S_o - S_f + \frac{Q^2}{gA^3}\frac{\partial A}{\partial x} - \frac{Qq^*}{gA^2} - F_q}{1 - F_r^2} \tag{4.6}$$

in which

$F_q = 0$ for bulk lateral outflow

$F_q = \frac{Vq^*}{2gA} = \frac{Qq^*}{2gA^2}$ for seepage flow

$F_q = \frac{(V - U_q)q^*}{gA} + \frac{h_c}{A}\frac{\partial A}{\partial x}\Big|_Y$ for bulk lateral inflow in which U_q is the velocity component of the inflow in the direction of the main channel flow, and q* represents the lateral inflow or outflow per unit length of main channel with q* negative for lateral outflow, and positive for lateral inflow.

For a prismatic channel, the terms containing $\partial A/\partial x|_Y$ are zero, and Equation 4.6 reduces to

$$\frac{dY}{dx} = \frac{S_o - S_f - \dfrac{Qq^*}{gA^2} - F_q}{1 - F_r^2} \tag{4.7}$$

If there is no lateral inflow or outflow, then note that Equation 4.7 reduces to Equation 4.4. It is Equation 4.4 that we will begin with, and after gaining experience in solving it, more general problems will be discussed in which the channel may be nonprismatic, and lateral inflow or outflow may also occur.

4.3 GRADUALLY VARIED FLOW IN PRISMATIC CHANNELS WITHOUT LATERAL INFLOW OR OUTFLOW

4.3.1 Classification of Gradually Varied Profiles

For the simplest case of gradually varied flow in a prismatic channel in which the flow rate does not change with the position along the channel, a classification of the flow profiles has been adopted and is understood by hydraulic engineers. This classification consists of an upper case letter that denotes whether the slope of the channel bottom, under uniform flow, will produce subcritical, critical, or supercritical flow, and a subscript to this letter that denotes the relationship of the actual depth to the two reference depths, the normal depth Y_o, and the critical depth Y_c. The letter designation is as follows:

M is used if under uniform flow, the slope of the channel is such that the flow will be subcritical. The M stands for **mild** channel.

S is used if under uniform flow, the slope of the channel is such that the flow will be supercritical. The S stands for **steep** channel.

C is used if under uniform flow, the slope of the channel will produce a critical flow. The C stands for **critical**.

H is used if the slope of the channel bottom is zero, i.e., the channel is **horizontal**.

A is used if the slope of the channel bottom is negative, i.e., the bottom of the channel increases in elevation in the direction of the flow. The A stands for **adverse**.

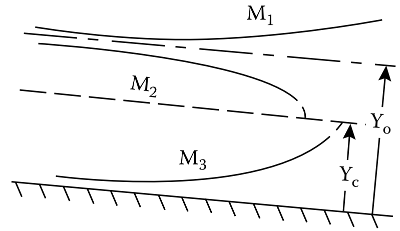
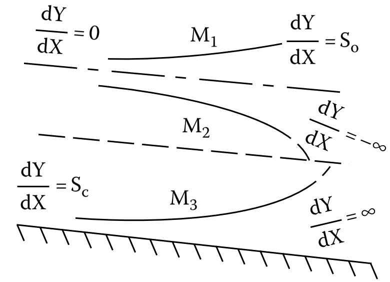
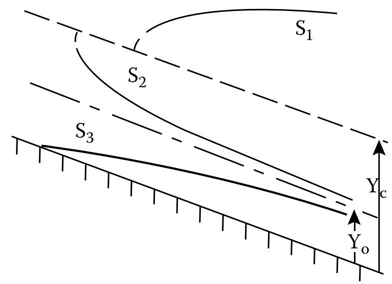
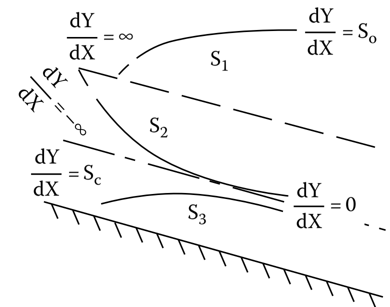
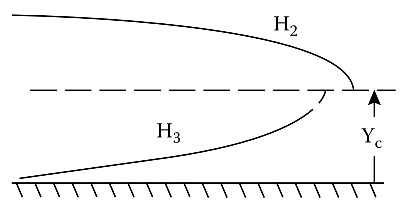
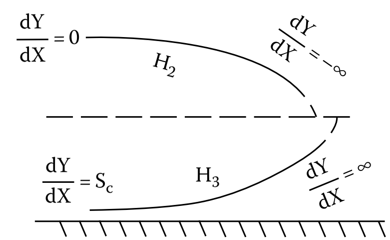
The subscript to this letter will be

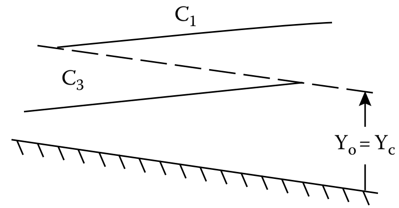
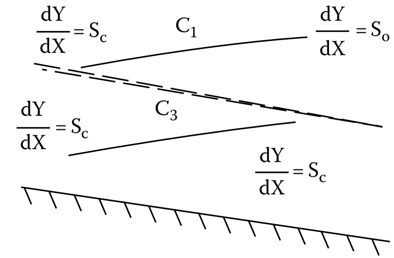
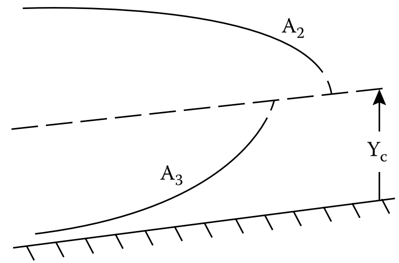
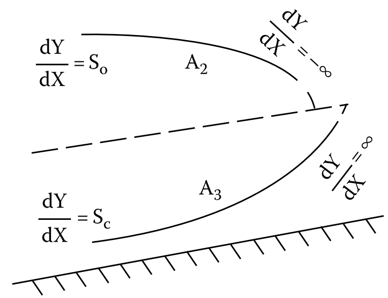
1. If the actual depth is above both the reference depths
2. If the actual depth is between the two reference depths
3. If the actual depth is below both the reference depths

Horizontal and adverse channels cannot have normal depths in them, i.e., Y_o approaches infinity. There are only two gradually varied profiles possible in these channel, H_2 and H_3, and A_2 and A_3, respectively. Since, in a critical channel, Y_o and Y_c coincide, the only two possible profiles in this very special channel are C_1 and C_3.

Table 4.1 shows the possible gradually varied profiles, which will be referred to as the GVF profiles,hereinafter. For example, if a channel is mild, i.e., for the given flow rate this channel would contain a subcritical flow if under uniform conditions, and a dam backs the water up so the actual depth is above the normal (or uniform) depth, then this GVF profile is referred to as an M_1 profile,

TABLE 4.1
Gradually Varied Profiles in Prismatic Channels

Slope of Channel B	Profile Designation	Sign Associated with Equation 4.4	Sketch of Profiles and Reference Lines	Magnitude of dY/dx at Ends of Profile
Mild	M_1	$\frac{dY}{dx} = \frac{+}{+} = +$		
	M_2	$\frac{dY}{dx} = \frac{-}{+} = -$		
	M_3	$\frac{dY}{dx} = \frac{-}{-} = +$		
Steep	S_1	$\frac{dY}{dx} = \frac{+}{+} = +$		
	S_2	$\frac{dY}{dx} = \frac{+}{-} = -$		
	S_3	$\frac{dY}{dx} = \frac{-}{-} = +$		
Horizontal	H_2	$\frac{dY}{dx} = \frac{-}{+} = -$		
	H_3	$\frac{dY}{dx} = \frac{-}{-} = +$		

Critical	C_1	$\frac{dY}{dx} = \frac{+}{+} = +$		
	C_3	$\frac{dY}{dx} = \frac{-}{-} = +$		
Adverse (or negative slope)	A_2	$\frac{dY}{dx} = \frac{-}{+} = -$		
	A_3	$\frac{dY}{dx} = \frac{-}{-} = +$		

or back water curve. If on the other hand, a gate in this mild channel causes a supercritical flow downstream from it, then the GVF profile is denoted by M_3.

It will be advantageous to examine the shape of the GVF profiles by examining the signs associated with both the numerator and the denominator of Equation 4.4, which is repeated below:

$$\frac{dY}{dx} = \frac{S_o - S_f}{1 - F_r^2} \tag{4.4}$$

The numerator consists of the difference between the slope of the channel bottom and the slope of the energy line, $S_o - S_f$, and will be positive if the actual depth Y, is above the normal depth Y_o. The reason is that for greater than normal depth, the velocity will be less than the uniform velocity, and consequently less frictional head loss will occur, and therefore S_f is smaller then S_o. On the other hand, the numerator will be negative whenever the actual depth Y is less than Y_o, based on this same reasoning. The denominator will be positive if the actual depth Y is greater than the critical depth Y_c, and will be negative whenever the actual depth is less than the critical depth. Reasons for this is that, since the denominator consists of $1 - F_r^2$, it will have a negative sign if the Froude number is greater than 1, and positive if the Froude number is less than 1, and the Froude number is greater than unity for supercritical flows, and less than unity for subcritical flows. If the numerator and the denominator have the same sign, dY/dx is positive. A positive value for dY/dx means that the depth will increase in the downstream direction. On the other hand, if the numerator and the denominator of the last part of Equation 4.4 are of opposite signs, then dY/dx is negative. A negative value for dY/dx indicates that the depth decreases in the downstream direction.

Table 4.1 indicates the sign of dY/dx in its third column for the various GVF profiles. As the following GVF profiles approach the critical depth, the slopes of the water surfaces approach infinity: M_2, M_3, H_2, H_3, S_1, S_2, A_2, and A_3. For these profiles, Equation 4.4 indicates that dY/dx becomes extremely large as the critical depth is approached because the denominator of Equation 4.4, $1 - F_r^2$ approaches zero, and the numerator remains finite. In an actual channel flow, the change in depth becomes more rapid as the critical depth is approached, but dY/dx does not approach infinity. The results from Equation 4.4 violate the assumption that was used in its development, namely that the flow be gradually varied and normal acceleration components can be ignored. Therefore, the ODEs for gradually varied flows, Equations 4.2 through 4.7 do not apply near the critical depth. In solving the GVF profile, it is therefore necessary to start with a depth slightly below the critical depth, or slightly above the critical depth. Another problem exists for the ends of the GVF profiles as they approach the normal depth Y_o, for here the numerator $S_o - S_f$ approaches zero, and therefore dY/dx approaches zero, indicating that no change occurs in the depth with a given change in x. No such change is equivalent to saying that the GVF profile has an infinite length in approaching the uniform depth. From a practical viewpoint, however, gradually the varied flow becomes a uniform flow when the depth is within about 1% of the normal depth.

To assist in avoiding numerical difficulties in solving Equation 4.4, it is well to examine, and thereafter be aware of, the extreme or the limiting values that dY/dx can have at the ends of the GVF profiles. These limiting values are shown in the last column of Table 4.1. In other words, the range of changes in depth is with respect to the position dY/dx, for each profile is restricted to values between its extreme ends. One might question the need to be concerned with the magnitude of a derivative like dY/dx, when it is the depth that is ultimately sought. The answer is that if we are not aware of when dY/dx becomes zero, or infinite, we may expect the impossible from the numerical solution, and by being aware, we can avoid problems.

Whenever the GVF profile approaches the critical depth, the denominator of Equation 4.4 becomes zero, since F_r^2 approaches 1 and depending upon whether $S_f > S_o$ or $S_f < S_o$, the magnitude of dY/dx will approach negative, or positive infinity, respectively. GVF profiles whose ending derivative values are negative infinity are M_2, H_2, and A_2. Those with dY/dx equal to positive infinity at their end are M_3, H_3, and A_3. The S_1 profile has a positive infinite dY/dx value at its upstream end, and the S_2 a negative infinite value at its upstream end. For all these profiles, where dY/dx becomes infinite, the curvature of the streamlines becomes too large for the one-dimensional hydraulics to be valid as critical depth is approached, and therefore Equation 4.4 does not describe the actual GVF profile. When solving GVF profiles, judgement must be applied in starting or ending the numerical computations about 5% above or below the critical depth.

For GVF profiles that approach the uniform depth, the numerator of Equation 4.4 approaches zero, since S_f approaches S_o. Such profiles approach normal depth asymptotically, and according to the mathematics, they are infinitely long. In practice, when the depth is within 1% or 2% of the normal depth, the GVF profile ends and the flow is uniform thereafter. The upstream end of the M_1 and the M_2 profiles and the downstream ends of the S_2 and the S_3 profiles are in this category where dY/dx approaches 0. The downstream end of the M_1 and the S_1 GVF profiles have dY/dx approach the slope of the channel bottom, because as the depth becomes large S_f approaches 0 and F_r^2 approaches 0, and thus dY/dx approaches $S_o/1$. The same idea applies to the upstream end of the H_2 and the A_2 GVF profiles. In the case of an H_2 GVF profile, S_o approaches 0, so its upstream dY/dx becomes 0, and for the adverse slope, S_o is negative, so $dY/dx = -|S_o|$.

To determine the values that dY/dx approach at the upstream limit of the M_3, the S_3, the H_3, the A_3, and the C_3 GVF profiles, let us introduce the critical slope. By definition, the critical slope will produce a uniform flow at the critical depth. Therefore, the critical slope is the bottom slope computed from the uniform flow equation (either Manning's or Chezy's equation) when the critical flow equation is satisfied, or

$$S_c = \left(\frac{nQP^{2/3}}{C_u A^{5/3}}\right)^2 = \frac{n^2Q^2P^{4/3}}{C_u A^{10/3}} \quad \text{and} \quad \frac{Q^2T}{gA^3} = 1, \quad \text{or} \quad Q^2 = \frac{gA^3}{T}$$

C_u is used for 1.486 or 1 here to prevent confusion with Chezy's C. Substituting Q from the critical flow equation into Manning's equation gives

$$S_c = \frac{n^2gP^{4/3}}{C^uTA^{1/3}} = \frac{S_f}{F_r^2} \text{ or from Chezy's equation} \quad S_c = \frac{gP}{TC^2} = \frac{S_f}{F_r^2}$$

The result of S_f/F_r^2 is obtained by regrouping the terms as shown below when using the Manning equation, and thereafter by multiplying the numerator and the denominator by $Q^2/A^{10/3}$.

$$S_c = \frac{\left(n^2P^{4/3}/C_u^2\right)\left(Q^2/A^{10/3}\right)}{\left(TA^{1/3}/g\right)\left(Q^2/A^{10/3}\right)} = \frac{n^2Q^2P^{4/3}/\left(C_u^2A^{10/3}\right)}{\left(Q^2T/gA^3\right)} = \frac{S_f}{F_r^2}$$

or using Chezy's equation,

$$S_c = \frac{(P/C^2)(Q^2/A^3)}{(T/g)(Q^2/A^3)} = \frac{Q^2P/(C^2A^3)}{Q^2T/(gA^3)} = \frac{S_f}{F_r^2}$$

Thus, we see that the critical slope is also obtained by the equation that gives the slope of the energy line divided by the Froude number and by setting the Froude number at unity. Consider what happens as the depth becomes very small. The slope of the channel bottom becomes very small in comparison to S_f, and therefore the numerator of Equation 4.4 approaches $-S_f$. Likewise, 1 is small in comparison to F_r^2, and so the denominator of Equation 4.4 approaches $-F_r^2$ and dY/dx approaches $S_f/F_r^2 = S_c$. In other words, profiles whose upstream depths can approach zero (M_3, S_3, H_3, A_3, and C_3), have these end values of dY/dx approach the critical slope S_c. The remaining question is what values for dY/dx exist at the ends of the C_1 and the C_3 profiles as these approach the coincident normal and the critical depths. Here, dY/dx takes on the indeterminate form 0/0, which requires that L'Hôpital's rule of taking the derivative of the numerator and the denominator be followed. An alternative approach that eliminates the mathematics associated with taking these derivatives, etc., is to substitute $S_f = S_c F_r^2$ in Equation 4.4 and to note $S_o = S_c$ for a critical channel, as shown below:

$$\frac{dY}{dx} = \frac{S_o - S_c F_r^2}{1 - F_r^2} = \frac{S_c(1 - F_r^2)}{1 - F_r^2} = S_c \quad \text{by dividing out } (1 - F_r^2)$$

Thus, a C_3 GVF profile has the same value of dY/dx at its two extreme ends.

To avoid numerical problems associated with getting too close to the critical depth and the normal depth, we will define the GVF profiles as the water surface to within 1% of the normal depth, and within 5%–10% of the critical depth.

4.3.2 Sketching GVF Profiles in Prismatic Channels

The first step in solving gradually varied flow problems in open channels is to sketch in and identify the type of profile that will occur. It is not possible to define an exact step-by-step procedure by which this can be done, but the following are usually important considerations or properties of the flow to be identified: (As you read this it will be helpful to look at the sketches in Figure 4.1.)

1. Identification of the "control points" of the flow. A gate in the channel controls both the flow upstream from it, as well as downstream from it. A gate is therefore, always a good point from which to sketch in GVF profiles, both upstream and downstream from it. The upstream flow will be subcritical, and its depth will be determined by the gate setting. If the channel has a bottom slope that will support a uniform supercritical depth for this flow rate, then the upstream GVF profile will be an S_1. If the uniform depth is subcritical, this upstream GVF profile will be an M_1. In a mild channel, the GVF profile downstream from the gates will be of the M_3 type, and if the channel is steep, then the downstream GVF profile will be of the S_3 type. An exception occurs if the flow downstream from the gate is submerged. A break in grade in an upstream mild channel to a steep slope is also a control point. At the break in grade, the depth will be critical with the flow downstream therefrom supercritical unless the water is "backed-up" by another control further downstream. A reservoir at the head of a steep channel is a control point, and the depth here will be critical. For a mild channel, this is not the case.
2. A hydraulic jump will terminate an M_3 GVF profile in a mild channel, and may be one end of an M_1 or an M_2 GVF profile on its subcritical side, or if the channel is long, the subcritical side of a hydraulic jump may be at normal depth. If the hydraulic jump occurs in a steep channel, then there will either be no GVF profile upstream from it, or the GVF profile will be an S_2 (or S_3) profile on its way to approaching the normal depth in this steep channel. Hydraulic jumps are therefore, other key points used in deciding how to sketch in the GVF profile. Often, however, the position of a hydraulic jump must be determined as part of the solution to the GVF problem. If it is not known whether the hydraulic jump exists upstream or downstream from a break in grade, then both possibilities should be shown on the sketch, so that later computations can determine which of these is correct.
3. Downstream reservoirs will create an M_2 or an M_1 profile depending upon whether the elevation of the water surface in the reservoir is below or above the normal depth (for this flow rate) in the channel, respectively. When a steep channel terminates in a downstream reservoir, there are two possible situations that can occur: (a) If the reservoir elevation level is above the critical depth, then an S_1 GVF profile will exist in the upstream channel, and a hydraulic jump will exist upstream from it. This hydraulic jump will be caused by the high water level in the reservoir. (b) If the reservoir water surface is below the critical depth in

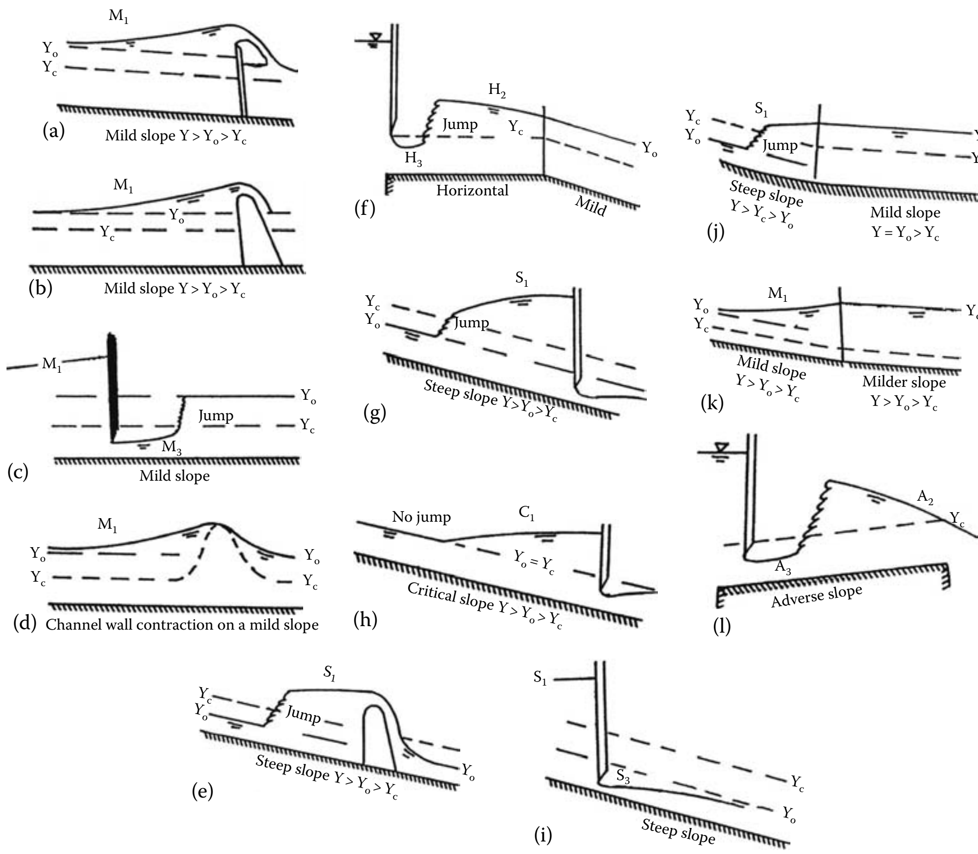


FIGURE 4.1 Gradually varied profiles caused by a single control. (a) The weir causes flow above the normal depth in a mild channel and this is an M_1 GVF. (b) The small dam causes the same M_1 GVF upstream as the weir. (c) The flow upstream from a gate is always subcritical and if flow emerges as "free flow" from the gate it is supercritical downstream. Thus upstream an M_1 GVF occurs, and an M_3 GVF since the channel is mild. (d) If the contraction is severe enough, it will cause critical flow at this reduced section with an M_1 GVF upstream. If the contraction is small and the channel enlarges again downstream, or is steeper, then the upstream flow may remain at normal depth. (e) In a steep channel, a dam will cause a jump to take the flow from supercritical to subcritical conditions, and an S_1 GVF will exist downstream there from. Upstream of the jump the depth will be normal. Thus the S_1 GVF will start at the depth conjugate to Y_0 and end at a depth with a specific energy that is equal to the critical specific energy at the crest of the dam plus the height of the dam, or $E = E_c + \Delta z$. (f) The depth in the mild downstream channel will be normal so an H_2 GVF will occur downstream from the hydraulic jump and an H_3 GVF upstream there from. If the horizontal channel is short, and the water has a very large velocity as it passes under the gate the hydraulic jump may be pushed into the mild channel in which case the H_3 GVF will exist over the entire length of the horizontal channel and continue as an M_3 GVF in the downstream channel until the jump occurs. (g) A gate in a steep channel will force the flow to be subcritical upstream from it so an S_1 GVF will occur that has its beginning at the depth conjugate to the normal depth in the steep channel. Downstream an S_3 GVF will occur. (h) In a critical channel, a gate will cause an C_1 GVF upstream that starts at critical depth so no jump occurs, and an C_3 GVF will occur downstream. (i) No jump occurs downstream from a gate in a steep channel unless something downstream there from forces the flow to be subcritical. (j) A steep channel abruptly changes to a mild channel will either cause an S_1 GVF upstream from the break in grade as shown if the momentum function associated with the downstream normal depth is greater than that for the upstream normal flow, $M_{o1} < M_{o2}$. If $M_{o1} > M_{o2}$ then the jump will occur downstream with an M_3 GVF starting at the break in grade and ending with the depth conjugate to Y_{o2}. (k) A reduction in the bottom slope in a mild channel will cause an M_1 GVF upstream from the break in grade. (l) Adverse or horizontal channels have profiles much like those in mild channels. The difference is they are designated with an A or H letter.

this steep channel, then the water depth in the channel will be unaffected by the reservoir, i.e., no GVF profile will exist unless it is in the lower portion of the varied flow from an upstream cause. If the reservoir water surface level is above normal but below critical, then a surge just off the end of the channel will form in the reservoir, but this surge (or standing wave) will be unable to move into the channel because the channel velocity of the channel flow is larger than the speed of its movement.

4. A break in grade, e.g., a change in the bottom slope, will always cause a length of gradually varied flow to occur and often GVF profiles will exist upstream and downstream therefrom. If the change in grade is from a mild to a steep channel, then an M_2 profile occurs upstream, and an S_2 profile occurs downstream with a critical depth at the break in grade. If the break in grade is from a mild to a milder slope, then an M_1 profile will occur upstream, and if the downstream channel is long enough, then the depth in the downstream milder channel will be uniform. Likewise, a change from a milder to a steeper, but still mild slope will cause a portion of an M_2 profile in the upstream channel. If both the channel upstream and downstream from a break in grade are steep, then the two cases are: (a) to a steeper slope an S_2 GVF profile will exist in the downstream channel, and (b) to a less steep slope, an S_3 will occur in the downstream channel. However, no GVF profiles exist upstream from the grade change.

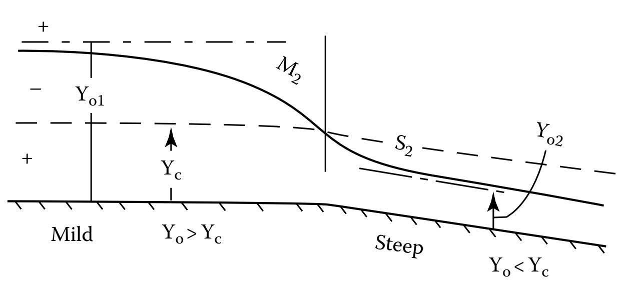
The various possibilities described above are illustrated in the sketches shown in Figure 4.1, which depict situations in which a GVF profile(s) occurs as a result of a single control. It is assumed in these situations that the channel is very long both upstream and downstream from the control. Figure 4.2 depicts situations where this is not the case and the effects of several controls can cause a different type profile to begin where another type ends.

EXAMPLE PROBLEM 4.1

Sketch in the GVF profiles for the four situations shown. You should note that two reference lines are given, one for the normal depth and the other for the critical depth. In a mild channel, the normal depth line is above the critical depth line and in a steep channel the reverse is true, the normal depth is below the critical depth line. Therefore, if the type of channel is not given and reference lines are, you can determine whether the channel is mild or steep, etc.

(1) A break in grade occurs from a mild to a steep channel, but the channel's size does not change.

Solution to (1): Since the flow will be supercritical downstream from the break in grade and the subcritical upstream, the flow must pass through the critical depth at the break in grade. Thus upstream, the depth will be between the normal and critical depths, or $Y_o > Y > Y_c$ and dYdx will be negative, because of a negative numerator and the GVF will be an M_2, that passes through the critical depth at the break in grade and asymptotically approaches the normal depth at its other end. Downstream of the break in grade, an S_2 GVF will occur. The channel is steep and the depth is still between the two reference depths. dY/dx is also negative here since the denominator is negative. The beginning of the S_2 is at the critical depth and at its other end, it asymptotically approaches the normal depth in the downstream steep channel.



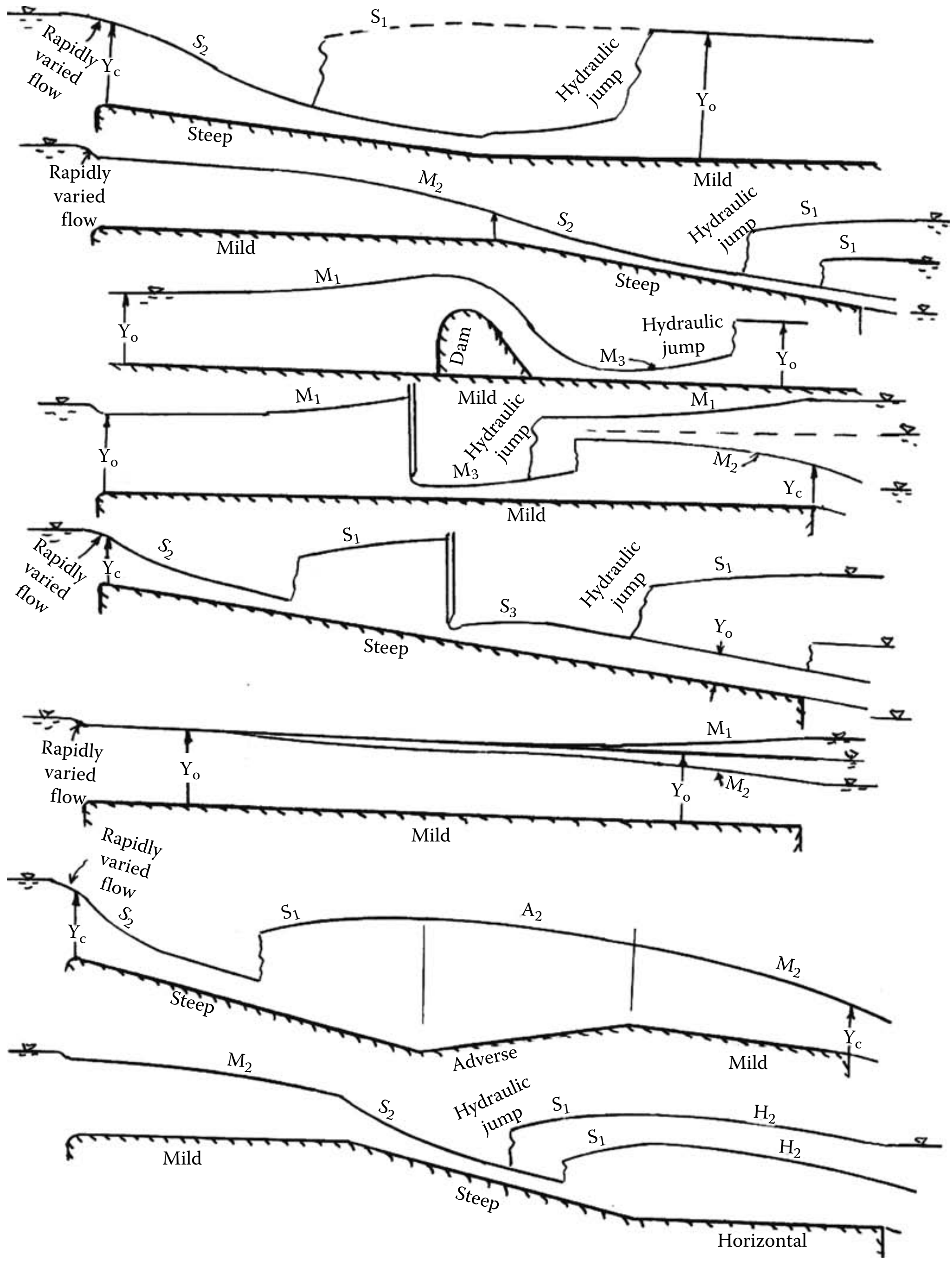
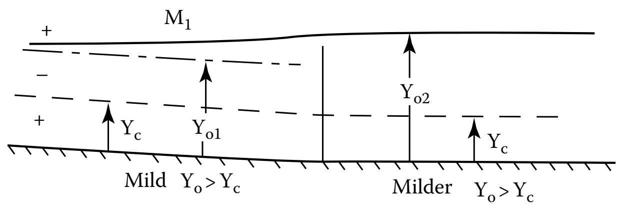


FIGURE 4.2 Examples of GVF profiles.

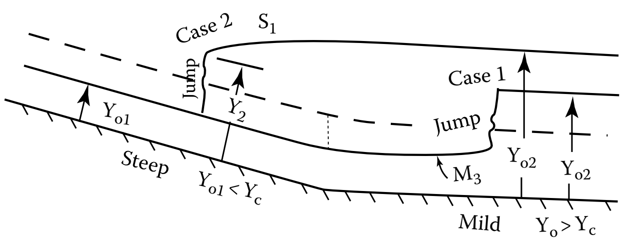
(2) A mild channel has its bottom slope suddenly reduced, but the channel size remains the same.

Solution to (2): Assuming that the downstream channel is very long, fluid friction will control and a uniform flow will exist in it. Therefore, at the break in grade, the depth will equal the normal depth for the downstream channel, or $Y = Y_{o2}$. Since this depth is above the normal depth in the upstream channel, and this is also a mild channel, an M_1 GVF will occur as shown below.



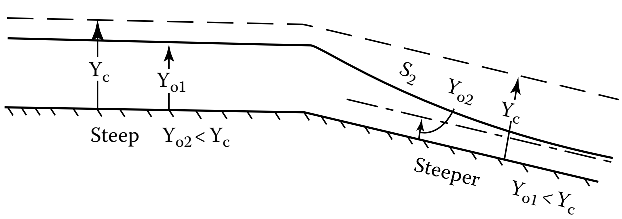
(3) A steep channel suddenly changes to a mild channel but the channel size remains constant. Both the upstream and the downstream channels are very long.

Solution to (3): Upstream from the break in grade, the flow will be supercritical, and therefore cannot be controlled by downstream conditions. However, the flow will be controlled by downstream conditions, and in this case, by fluid friction and will be at normal depth. Thus, the flow must change from the supercritical to the subcritical, and this can occur only by means of an abrupt hydraulic jump. The jump may either occur upstream or downstream from the break in grade depending respectively upon whether the momentum function associated with the downstream normal depth is greater or less than that for the upstream normal depth. A jump will occur upstream if $M_{o1} < M_{o2}$ and downstream if $M_{o1} > M_{o2}$. If the jump occurs upstream, then an S_1 GVF will start at a depth that is conjugate to Y_{o1} and end at the break in grade with Y_{o2}. If the jump occurs downstream, then an M_3 GVF will occur that starts at Y_{o1} and ends at a depth conjugate to Y_{o2}.



(4) A steep channel suddenly has its bottom slope increased but its size remains constant.

Solution to (4): Being supercritical, the flow in the upstream channel cannot gradually react to the steeper downstream slope, so it will remain at normal depth to the break in grade. At this position, the normal depth will be below the actual depth, and an S_2 GVF will exist that gradually becomes less approaching the downstream normal depth asymptotically.



4.3.3 Alternative Forms of the ODE That Describe GVF Profiles

In Equation 4.4, the depth Y is considered the dependent variable (or unknown) and x is considered the independent variable. Alternative forms of ODEs for describing GVF profiles can be developed. The specific energy E may be introduced as the dependent variable in place of the depth Y. To accomplish this change of dependent variables take the differential of the specific energy equation, $E = Y + Q^2/(2gA^2)$, to give

$$dE = dY + \frac{Q^2T}{gA^3}dY = dY + F_r^2 dY$$

Substituting this result into Equation 4.4 gives the following:

$$\frac{dE}{dx} = S_o - S_f \tag{4.8}$$

In Equation 4.8, the specific energy E is considered the dependent variable and x the independent variable. Since both E and S_f depend on the depth, Y must be used as an auxiliary variable. While it may not be apparent, Equation 4.8 allows for a prismatic as well as a nonprismatic channel. That is, Equation 4.8 can be used to solve the same problems as Equation 4.6 can (with the term $\partial A/\partial x|_Y$ included) provided q* is zero. This can be shown by noting that $dA = \partial A/\partial x|_Y + T\,dY$ for a nonprismatic channel.

Another alternative form of Equation 4.4, that still considers Y the dependent variable, can be obtained by solving S_f from Manning's equation, (or from Chezy's equation, as another possibility) giving $S_f = n^2Q^2P^{4/3}/(C_u^2A^{10/3})$. Multiplying and dividing this by gT allows the Froude number squared to be isolated as follows:

$$S_f = \frac{Q^2T}{gA^3}\frac{gn^2P^{4/3}}{C_uTA^{10/3}} = F_r^2\frac{gn^2P^{4/3}}{C_uTA^{10/3}}$$

A critical slope S_c is defined by letting the Froude number equal unity or

$$S_c = \frac{gn^2P^{4/3}}{C_u^2TA^{10/3}}$$

so $S_f = S_cF_r^2$ and the gradually varied ODE becomes

$$\frac{dY}{dx} = \frac{S_o - S_cF_r^2}{1 - F_r^2} \tag{4.9}$$

In solving Equation 4.9 versus solving Equation 4.4, the need to repeatedly compute S_f may appear to have been replaced by solving S_c once. However, this is not true because S_c is not constant, but varies with the depth as can be seen in the equation above Equation 4.9. Thus, the computation of S_f is replaced with the computation for S_c.

A fundamental theorem of calculus is that when dealing with full derivatives of continuous variables, dx/dy = 1/(dy/dx). Therefore, to interchange the role of the variables, all that is needed is to take the reciprocal of both sides of the expressions that define an ODE. Thus, if the desire is to have an ODE that considers x the dependent variable (i.e. the unknown), and the specific energy the independent variable, then the reciprocal of both sides of Equation 4.8 can be taken to give

$$\frac{dx}{dE} = \frac{1}{S_o - S_f} \tag{4.10a}$$

If an ODE is desired that considers Y the independent variable, and x the dependent variable, then the reciprocal of both sides of Equation 4.4 can be taken to give

$$\frac{dx}{dY} = \frac{1 - F_r^2}{S_o - S_f} \tag{4.10b}$$

It is well to examine the characteristics of these alternative forms of ODEs that describe GVF profiles in a prismatic channel with no lateral inflow or lateral outflow. First note that both the Froude number F_r, and the slope of the energy line S_f, are functions of the depth Y and not directly functions of the position along the channel x. Only indirectly are these variables related to x because for GVF profiles, the depth Y changes with x. The Froude number is defined by its definition or $F_r^2 = Q^2T/(gA^3)$, and since both T and A are functions of Y once the shape of the channel is defined, it is a function of Y. The slope of the energy line S_f is defined by a uniform flow equation, either Chezy's equation or Manning's equation, because these equations define the frictional loss due to fluid motion. The difference between using them to define S_f and to solve the slope of the channel bottom (with all other variables known) in a uniform flow is that S_f will have a different value depending upon the depth of flow at a point, and therefore will indirectly vary with x. If Manning's equation is used, S_f is given by

$$S_f = \left(\frac{nQP^{2/3}}{C_uA^{5/3}}\right)^2 \tag{4.11}$$

in which $C_u = 1.486$ when using ES units, and $C_u = 1.00$ when using SI units. If Chezy's equation is used, then S_f is given by

$$S_f = \frac{Q^2}{A^2R_hC^2(e/R_h, R_e)} \tag{4.12}$$

in which C is Chezy's coefficient that is a function of the relative roughness of the channel wall e/R_h and the Reynolds number R_e as described in Chapter 2.

Since the quantities on the right side of the equal sign for both Equation 4.10a and b depend only upon the independent variables Y, they can be solved by multiplying the equation by the differential of this dependent variable and by performing a numerical integration. In other words, Equation 4.10a can be written as

$$L = x_2 - x_1 = \int dx = \int_{E_1}^{E_2} \frac{1}{S_o - S_f} dE \tag{4.10c}$$

and Equation 4.10b can be written as

$$L = x_2 - x_1 = \int dx = \int_{Y_1}^{Y_2} \frac{1 - F_r^2}{S_o - S_f} dY \tag{4.10d}$$

Any valid numerical integration method may be used for carrying out the integration of Equation 4.10c and/or d including the trapezoidal rule, or Simpson's rule. In order to solve Equation 4.10b and c, it is necessary that the depth Y be known at both ends of the GVF profile. In other words, in order to solve the GVF equation with Y the independent variable, its beginning and ending value must be known, and a starting value of the dependent variable must be given. This requirement can be stated as a general rule. "To solve an ODE, it is necessary that the two end values of the independent variable be given, and that one value, the starting value, for the dependent variable be given." Often, this requirement dictates which form of the GVF equation must be solved; one that considers x independent, or one that considers Y independent. Reasons why what is known dictate which form of the equation must be solved is illustrated in some of the example problems that follow.

If you have a calculator that does numerical integrations such as an HP48X, or HP48G, then it can readily solve problems in which Y is considered the independent variable and x the dependent variable, i.e., solve Equation 4.10d. The ultimate way that you would want to write Equation 4.10d is to use the variables, i.e., evaluate the Froude number $F_r^2 = Q^2T/(gA^3)$, and the slope of the energy line $S_f = [(nQ/C)/A(P/A)^{2/3}]^2$ in terms of the variable of the channel and the depth. For a trapezoidal channel, the variables are b, m, n, S_o, and Y, and for a circular channel they are D, n, S_o, and Y. To solve any problem, the variables are given the appropriate values for that problem. Before writing these general equations and storing them in your calculator, you may wish to have the calculator solve a simple problem by carrying out the numerical integration for that problem. For this purpose, consider a flow rate Q = 200 cfs in a rectangular channel with b = 10 ft, n = 0.015, and $S_o = 0.0005$. A gate creates a depth of Y = 6 ft. The solution of Manning's equation indicates that the normal depth is 4.93 ft. Therefore, the gate causes an M_1-GVF backwater curve with a depth change from 5 to 6 ft. After substituting the known values into Equation 4.10d, the length of this profile is given by

$$L = \int_5^6 \frac{1 - 12.42236/Y^3}{0.0005 - \left(\dfrac{0.040757}{Y^2}\right)\left\{\dfrac{1+0.2Y}{Y}\right\}^{4/3}} dY$$

Your calculator should give a solution L = 10,379 ft. Now, develop two numerical integration equations that use the variable names, one for a trapezoidal channel, and one for a circular channel, and use the first one to verify the above result from the numerical integration for this specific problem. Doing such numerical integrations with the calculator are a good means for solving problems in which you know the depths at both ends of the GVF profile, i.e., problems in which Y is the independent variable, but not for solving problems in which Y is the dependent variable. Even if you have a calculator that integrates Equation 4.4, you should consider several other alternatives that are presented below.

For use-in-hand computations of the length of GVF profiles, Equation 4.10a is often written as

$$\Delta x = \frac{\Delta E}{S_o - S_f} \tag{4.10e}$$

and the solution is obtained by carrying out a trapezoidal rule-type numerical integration using a table (see the following problem) in which the depth is incremented in the first column, and the subsequent columns contain the numerator and the denominator of Equation 4.10e. The last column performs the division that computes Δx. The summation of this column of Δx gives the length of the GVF profile corresponding to the first and the last depth in the table.

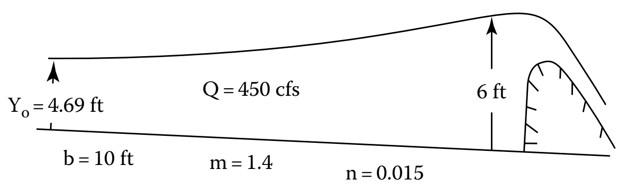
EXAMPLE PROBLEM 4.2

A small dam in a trapezoidal channel with b = 10 ft and m = 1.4 cause the depth to increase to 6 ft. If the bottom slope of this channel is $S_o = 0.0008$, and it has a Manning's n = 0.015 determine how far upstream from the dam the water depth is increased if the flow rate is Q = 450 cfs.

Solution

Solving Manning's equation for the depth indicates that the normal depth in this channel is $Y_o = 4.69$ ft. The GVF in this channel is of the M_1 type, and takes the depth from 1% above normal depth or 1.01(4.69) = 4.74 (4.75 ft will be used) to 6 ft just upstream of the dam. Since the depths are known at both ends of this GVF profile, one of the equation that considers Y the independent variable, can be solved. The table below gives a solution to Equation 4.10e based on using a form of the trapezoidal rule to carry out the numerical integration as done in the FORTRAN, PASCAL, or C programs listed below. There is also a listing below of a program that calls on the more precise Simpson's rule SIMPR algorithm described in Appendix B. You should also solve this problem using your HP48 calculator, or another calculator that has the capability of doing numerical integrations. Engineers have justified using crude trapezoidal rule-type integrations in the past, but with current tools available, these methods should only be used when solution results do not need to be very accurate.

Y	E	delE	S_f	$S_o - S_f$	del x	x
6.000	6.258		0.000307			0.0
		−0.216		0.000465	−464.3	
5.750	6.042		0.000363			−464.0
		−0.210		0.000403	−521.6	
5.500	5.832		0.000432			−986.0
		−0.203		0.000325	−623.2	
5.250	5.629		0.000518			−1609.0
		−0.194		0.000229	−847.5	
5.000	5.435		0.000625			−2457.0
		−0.182		0.000107	−1713.1	
4.750	5.253		0.000762			−4170.0



This solution indicates that this M_1 GVF profile has a length of 4170 ft upstream from the dam, based on five steps and the linear approximation used in the numerical integration. A more precise numerical integration gives this length as 4417 ft or a difference of 6%. The computer program listings below will carry out the solution as given above.

FORTRAN Listing of GVFXY1.FOR to Solve This Problem

```
      WRITE(6,*)' Give:Q,b,m,n,So,Y1,Y2,N,g'
      READ(5,*)Q,b,Fm,Fn,So,Y1,Y2,N,g
      C=1.
      If(g.GT.30.) C=1.486
      Q2G=Q*Q/(2.*g)
      QnC=Fn*Q/C
      A=(b+Fm*Y1)*Y1
      E1=Y1+Q2G/(A*A)
      X=0.
      SF1=(QnC*((b+2.*Y1*SQRT(Fm*Fm+1.))/A)**.6666667/A)**2
      WRITE(3,100)Y1,E1,SF1,X
100   FORMAT(5X,'Y',7X,'E','   delE',6X,'Sf','  So-Sf',
     &1'del x',7X,'x',/,1X,56('-'),/2F8.3,8X,F8.6,16X,F9.0)
      DY=(Y2-Y1)/N
      DO 10 I=1,N
      Y=Y1+DY*FLOAT(I)
      A=(b+Fm*Y)*Y
      E=Y+Q2G/(A*A)
      SF=(QnC*((b+2.*Y*SQRT(Fm*Fm+1.))/A)**.6666667/A)**2
      SFOAV=So-.5*(SF+SF1)
```

```
      DELX=(E-E1)/SFOAV
      X=X+DELX
      WRITE(3,'(16X,F8.3,8X,F8.6,F8.1)') E-E1,SFOAV,DELX
      WRITE(3,'(2F8.3,8X,F8.6,16X,F9.0)') Y,E,SF,X
      E1=E
10    SF1=SF
      STOP
      END
```

PASCAL Listing of Program GVFXY1.PAS to Solve This Problem

```
Program GVFXY1;
Function Expn(a,b:real):real;Begin if a<0 then
  Writeln('error in power',a,b)
  else Expn:=Exp(b*Ln(a)) End;
Var Q,b,m,n,So,Y1,Y2,g,Q2G,C,QnC,E,E1,SF,SF1,A,DY,Y,SFOAV,X,
  DELX:real;
I,No:integer;
BEGIN
 Writeln('Give:Q,b,m,n,So,Y1,Y2,N,g');
 Readln(Q,b,m,n,So,Y1,Y2,No,g); C:=1;
 If g>30 then C:=1.486; Q2G:=Q*Q/(2*g); QnC:=n*Q/C;
 A:=(b+m*Y2)*Y2; E1:=Y2+Q2G/sqr(A); X:=0;
 SF1:=sqr(QnC*Expn((b+2*Y2*sqrt(sqr(m)+1))/A,0.666667)/A);
 Writeln('   Y    E   delE   Sf  So-Sf  del x    x');
 Write('-----------------------------------------------------');
 Writeln('------');
 Writeln(Y2:8:3,E1:8:3,'    ',SF1:8:6,'        ',X:9:0);
 DY:=(Y1-Y2)/No;
 For I:= 1 to No do Begin
  Y:=Y2+DY*I;
  A:=(b+m*Y)*Y;
  E:=Y+Q2G/sqr(A);
  SF:=sqr(QnC*Expn((b+2*Y*sqrt(sqr(m)+1))/A,0.666667)/A);
  SFOAV:=So-0.5*(SF+SF1);
  DELX:=(E-E1)/SFOAV;
  X:=X+DELX;
  Writeln ('        ',E1-E:8:3,'    ',SFOAV:8:6,DELX:8:1);
  Writeln (Y:8:3,E1:8:3,'    ',SF:8:6,'        ',X:9:0);
  E1:=E;
  SF1:=SF End;
END.
```

C Listing of Program GVFXY1.C to Solve This Problem

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#define sqr(x) x*x
void main(void){ int i,no;
   float q,b,m,n,so,y1,y2,g,c=1,q2g,qnc,a,e1,sf1,sf,dy,y,e,\
      sfoav, delx,x=0;
   printf("Give: Q,b,m,n,So,Y1,Y2,N,g\n");
   scanf("%f %f %f %f %f %f %f %d %f",\
      &q,&b,&m,&n,&so,&y1,&y2,&no,&g);
   if(g>30.) c=1.486; q2g=q*q/(2.*g); qnc=n*q/c;
   a=(b+m*y1)*y1; e1=y1+q2g/(a*a);
```

```
   sf1=sqr(qnc*pow((b+2.*y1*sqrt(m*m+1.))/a, .6666667)/a);
   printf("      Y        E       delE       Sf    So-Sf del \
      x         x\n");
   for(i=1;i<57;i++) printf("-");printf("\n");
   printf("%8.3f %7.3f      %8.6f          %9.0f\n",\
      y1,e1,sf1,x); dy=(y2-y1)/(float)no;
   for(i=1;i<=no;i++){
      y=y1+dy*(float)i; a=(b+m*y)*y; e=y+q2g/(a*a);
      sf=sqr(qnc*pow((b+2.*y*sqrt(m*m+1.))/a,.6666667)/a);
      sfoav=so-.5*(sf+sf1); delx=(e-e1)/sfoav;
      x+=delx;
      printf("           %8.3f          %8.6f %7.1f\n",\
         e-e1,sfoav,delx);
      printf("%8.3f %7.3f        %8.6f       %9.0f\n" \,
         y,e,sf,x); e1=e; sf1=sf; }}
```

Listing of EPRB4_2.FOR That Calls on SIMPR for a Numerical Solution

```
      EXTERNAL EQUAT
      CALL SIMPR(EQUAT,6.,4.75,X,1.E-6,21)
      WRITE(*,*) X
      END
      FUNCTION EQUAT(Y)
      A=(10.+1.4*Y)*Y
      P=10.+3.440935*Y
      EQUAT=(1.-6288.82*(10.+2.8*Y)/A**3)/(.0008-20.633359
     &*((P/A)**.6666667/A)**2)
      RETURN
      END
      Returns as the solution −4417.068
```

Listing of EPRB4_2.C

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
extern float simpr(float (*equat)(float xx),float xb,\
   float xe, float err,int max);
float equat(float y){float a,p;
   a=(10.+1.4*y)*y; p=10.+3.440935*y;
   return (1.-6288.82*(10.+2.8*y)/pow(a,3.))/\
      (.0008-20.633359*pow(pow(p/a,.66666667)/a,2.));}
void main(void){
   printf("Length of GVF profile =%f\n",
      simpr(equat,6.,4.75,\1.e-6,21));}
```

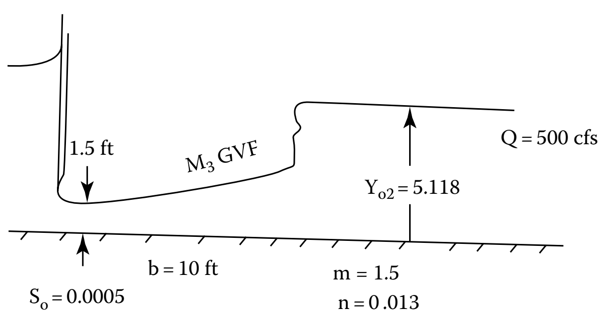
EXAMPLE PROBLEM 4.3

A gate exists in a channel with b = 10 ft and m = 1.5 that causes a depth of Y_1 = 1.5 ft immediately downstream from it. This channel has a bottom slope of S_o = 0.0005, and its n = 0.013 and it is carrying a flow rate Q = 500 cfs. Determine the location of the hydraulic jump.

Solution

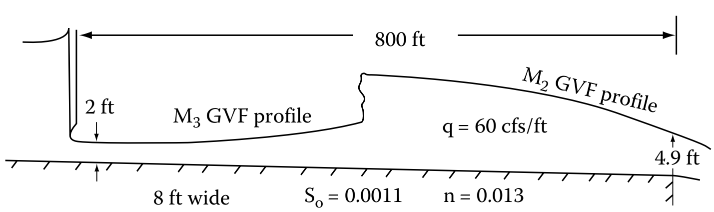
First, Manning's equation is used to solve the normal depth in the channel downstream from the hydraulic jump. This normal depth is Y_{o2} = 5.118. The depth just upstream of the hydraulic jump will be the conjugate depth to this. Therefore, next the momentum equation $M_1 = M_2$ must be solved to find the depth upstream from the jump. For this solution, the momentum function downstream of the jump M_2 is based on the normal depth of 5.118 ft. This solution gives the depth upstream of the jump as 2.297 ft. An M_3 GVF profile will take the depth from 1.5 ft immediately behind the gate to 2.297. The table below gives this solution. Note that since the depths were known at both ends of this GVF profile, it was most appropriate to consider x the dependent variable in this problem also. The solution indicates that the jump will occur 302 ft downstream from the gate. A more precise numerical integration gives this distance as 305 ft. If you are familiar with a spreadsheet (Lotus 123, Quattro, or Excel) you should solve this problem and Problem # 1 using it.

Y	E	delE	S_f	$S_o - S_f$	del x	x
1.500	12.997		0.044810			0.0
		−2.300		−0.037658	61.1	
1.659	10.698		0.031506			61.0
		−1.635		−0.026671	61.3	
1.819	9.062		0.022836			122.0
		−1.185		−0.019406	61.0	
1.978	7.878		0.016975			183.0
		−0.869		−0.014433	60.2	
2.138	7.009		0.012891			244.0
		−0.642		−0.010930	58.7	
2.297	6.367		0.009970			302.0



Example Problem 4.4

A sluice gate causes a depth of 2 ft in an 8 ft wide rectangular channel immediately downstream from it. The channel has a bottom slope of $S_o = 0.0011$, a Manning's $n = 0.013$, and 800 ft downstream from the gate the channel terminates in a free overfall. If the flow rate in this channel per unit width is $q = 60$ cfs/ft, determine the location of the hydraulic jump.



Solution

Since the distance between the sluice gate and the end of the channel is known, and a hydraulic jump, if it occurs, must be somewhere within this distance, it is more convenient to consider x the independent variable, and Y the dependent variable. Thus, Equation 4.4 will be selected. Because this channel is relative short, it is unlikely that a normal depth will exist downstream from the hydraulic jump. If a hydraulic jump does occur, then the depth at the end (free overfall) of the channel will be critical, and an M_2 GVF profile will occur between the jump to this critical depth ($Y_c = (q^2/g)^{1/3} = 4.82$ ft), and an M_3 GVF profile will occur upstream from the hydraulic jump. The hydraulic jump will, therefore, likely occur between these two GVF profiles, at a point where they match the conjugate depths. The three unknowns are as follows: (1) the distance from the gate to the hydraulic jump, (2) the depth upstream from the hydraulic jump, and (3) the depth downstream from the hydraulic jump. The three equations needed to solve these three unknowns are (a) the momentum equation across the jump $M_1 = M_2$, (b) the GVF equation for the M_3 profile, and (c) the GVF equation for the M_2 profile. Methods for solving systems of algebraic and ordinary differential equations will be discussed later. However, for this problem, an easy approach will be to solve both the M_2 and the M_3 GVF profiles past where it is anticipated that the hydraulic jump will occur, and then find the x position where depths from these two solutions are the two conjugate depths across the hydraulic jump. The tables below shows these two GVF solutions and also the values of the momentum functions M, associated with these depths. Note, the solution for the M_2 GVF profile starts at the end of the channel with a depth slightly above Y_c, since $Y = Y_c$ would result in a division by zero in Equation 4.4, e.g., an infinite value for dY/dx, and proceeds upstream. The position where the depths satisfy the conjugate depth equation is where the values of the momentum functions Ms (the last column in the solution tables) and x's are both equal in the two solution tables. This position is x = 354 ft, where an upstream depth of Y_1 = 3.459 ft exists and a downstream depth of Y_2 = 6.494 ft exists. Depending upon the size of the Δx interval used in the solution, it may be necessary to interpolate the distance x between a couple of table entries. The above solutions to the GVF profiles were obtained using the differential equation solvers, ODESOL RUKUST and/or DVERK described in Appendix C. More will be said about using these solvers later. Below are two FORTRAN listings that call these solvers. Note, these programs consist of a main program that defines the problem and appropriately call on the solver, and a subroutine that defines the derivative dY/dx for the solver, so it knows what the differential equation to be solved is. You should read Appendix C now as you go through these program listings and understand how they solve the ODE for a gradually varied flow. The next section discusses what these solver do.

M_3 GVF–Profile

x (ft)	Y (ft)	M (ft³)	
0.000	2.000	463.2	
2.000	2.007	461.7	
4.000	2.014	460.3	
.	.	.	
352.000	3.448	306.9	←
354.000	3.459	306.4	
356.000	3.470	305.9	
358.000	3.480	305.4	
.	.	.	
400.000	3.720	295.8	

M_2 GVF–Profile

x (ft)	Y (ft)	M (ft³)	
800.000	4.900	278.6	
798.000	4.982	278.8	
.	.	.	
358.000	6.488	306.2	
356.000	6.491	306.3	
354.000	6.494	306.4	←
352.000	6.497	306.5	
350.000	6.500	306.6	

FORTRAN Listing of Program EPRB4_2.FOR That Call ODESOL

```
      REAL Y(1),DY(1),XP(1),YP(1,1),WK1(1,13)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRANS/B,FM,FN,SO,Q2,FNQ,FMS
      WRITE(6,*) 'GIVE IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND,g'
      READ(5,*) IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND,G
      H1=-.01
      HMIN=1.E-5
      Y(1)=YB
      FMS=2.*SQRT(FM*FM+1.)
      C=1.
      IF(G.GT.30.) C=1.486
      FNQ=FN*Q/C
```

```
      Q2=Q*Q/G
      X=XBEG
      A=(B+FM*Y(1))*Y(1)
      FMON=B*Y(1)**2/2.+FM*Y(1)**3/3.+Q2/A
      WRITE(IOUT,100) X,Y,FMON
2     XZ=X+DELX
      CALL ODESOL(Y,DY,1,X,XZ,TOL,H1,HMIN,1,XP,YP,WK1,DYX)
      X=XZ
      A=(B+FM*Y(1))*Y(1)
      FMON=B*Y(1)**2/2.+FM*Y(1)**3/3.+Q2/A
      WRITE(IOUT,100) X,Y,FMON
100   FORMAT(6X,2F10.3,F12.2)
      IF(DELX .LT. 0.) GO TO 8
      IF(X .LT. XEND) GO TO 2
      GO TO 1
8     IF(X .GT. XEND) GO TO 2
1     IF(IOUT.NE.6) CLOSE(UNIT=IOUT)
      STOP
      END
      SUBROUTINE DYX(X,Y,DY)
      REAL Y(1),DY(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRANS/B,FM,FN,SO,Q2,FNQ,FMS
20    A=(B+FM*Y(1))*Y(1)
      T=B+2.*FM*Y(1)
      P=B+FMS*Y(1)
      SF=(FNQ*(P/A)**.66666667/A)**2
      FR2=Q2*T/A**3
40    DY(1)=(SO-SF)/(1.-FR2)
      RETURN
      END
```

Input data for above M3 GVF profile

```
3 .0001 2 2 480 .013 .0011 8 0 0 400 32.2
```

FORTRAN Listing of Program EPRB4D.FOR That Call DVERK

```
      REAL Y(1),CC(24),WK(2,9)
      EXTERNAL DYX
      COMMON /TRANS/B,FM,FN,SO,Q2,FNQ,FMS
      WRITE(6,* ) 'GIVE IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND,g'
      READ(5,*) IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND,G
      IER=0
      IND=1
      Y(1)=YB
      FMS=2.*SQRT(FM*FM+1.)
      C=1.
      IF(G.GT.30.) C=1.486
      FNQ=FN*Q/C
      Q2=Q*Q/G
      X=XBEG
      A=(B+FM*Y(1))*Y(1)
      FMON=B*Y(1)**2/2.+FM*Y(1)**3/3.+Q2/A
      WRITE(IOUT,100) X,Y,FMON
2     XZ=X+DELX
      CALL DVERK(1,DYX,X,Y,XZ,TOL,IND,CC,2,WK)
      IF(IER.NE.0) THEN
```

```
      WRITE(6,*) IER,' ERROR TERMINATION'
      STOP
      ENDIF
      A=(B+FM*Y(1))*Y(1)
      FMON=B*Y(1)**2/2.+FM*Y(1)**3/3.+Q2/A
      WRITE(IOUT,100) X,Y,FMON
100   FORMAT(6X,2F10.3,F12.2)
      IF(DELX .LT. 0.) GO TO 8
      IF(X .LT. XEND) GO TO 2
      GO TO 1
8     IF(X .GT. XEND) GO TO 2
1     IF(IOUT.NE.6) CLOSE(UNIT=IOUT)
      STOP
      END
      SUBROUTINE DYX(N,X,Y,DY)
      REAL Y(N),DY(N)
      COMMON /TRANS/B,FM,FN,SO,Q2,FNQ,FMS
20    A=(B+FM*Y(1))*Y(1)
      T=B+2.*FM*Y(1)
      P=B+FMS*Y(1)
      SF=(FNQ*(P/A)**.66666667/A)**2
      FR2=Q2*T/A**3
40    DY(1)=(SO-SF)/(1.-FR2)
      RETURN
      END
```

Input data for above M2 GVF profile

```
3 .0001 −2 4.9 480 .013 .0011 8 0 800 350 32.2
```

FORTRAN Listing of Program That Solves Equation 4.9 rather than Equation 4.4

```
      REAL Y(1),DY(1),XP(1),YP(1,1),WK1(1,13)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRANS/B,FM,FN,SO,Q2,FMS,SCN
      WRITE(6,*) 'GIVE IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND,g'
      READ(5,*) IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND,G
      H1=-.01
      HMIN=1.E-5
      Y(1)=YB
      FMS=2.*SQRT(FM*FM+1.)
      C=1.
      IF(G.GT.30.) C=1.486
      Q2=Q*Q/G
      SCN=G*(FN/C)**2
      X=XBEG
      A=(B+FM*Y(1))*Y(1)
      FMON=B*Y(1)**2/2.+FM*Y(1)**3/3.+Q2/A
      WRITE(IOUT,100) X,Y,FMON
2     XZ=X+DELX
      CALL ODESOL(Y,DY,1,X,XZ,TOL,H1,HMIN,1,XP,YP,WK1,DYX)
      X=XZ
      A=(B+FM*Y(1))*Y(1)
      FMON=B*Y(1)**2/2.+FM*Y(1)**3/3.+Q2/A
      WRITE(IOUT,100) X,Y,FMON
100   FORMAT(6X,2F10.3,F12.2)
      IF(DELX .LT. 0.) GO TO 8
```

```
      IF(X .LT. XEND) GO TO 2
      GO TO 10
8     IF(X .GT. XEND) GO TO 2
10    IF(IOUT.NE.6) CLOSE(UNIT=IOUT)
      STOP
      END
      SUBROUTINE DYX(X,Y,DY)
      REAL Y(1),DY(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRANS/B,FM,FN,SO,Q2,FMS,SCN
      A=(B+FM*Y(1))*Y(1)
      T=B+2.*FM*Y(1)
      FR2=Q2*T/A**3
      SC=SCN*(B+2.*Y(1)*SQRT(FM*FM+1.))**1.3333333/T/A**.3333333
      DY(1)=(SO-SC*FR2)/(1.-FR2)
      RETURN
      END
```

Listing of Program That Call on the Runge–Kutta Method Described in Appendix C

```
      REAL YY(1),YTT(1)
      COMMON B,FM,FM2,TWOM,FNQ,QG2,SO
      EQUIVALENCE (Y,YY(1))
      WRITE(6,*)'Give:IOUT,Q,b,m,So,n,Xbeg',Xend,DX,Ybeg,g,xstart'
      READ(5,*) IOUT,Q,B,FM,SO,FN,XBEG,XEND,DX,YBEG,G,DXS
      C=1.
      IF(G.GT.30.) C=1.486
      N=ABS(XEND-XBEG)/ABS(DX)+.5
      QG2=Q*Q/G
      TWOM=2.*FM
      FNQ=FN*Q/C
      FM2=2.*SQRT(FM*FM+1.)
      Y=YBEG
      WRITE(IOUT,90) XBEG,Y,(.5*B+FM/3.*Y)*Y*Y+QG2/((B+FM*Y)*Y)
90    FORMAT(F10.1,F10.3,F10.1)
      X1=XBEG
      DO 10 I=1,N
      X=X1+DX
      CALL RUKUST(1,DXS,X1,X,1.E-5,YY,YTT)
      WRITE(IOUT,100)X,Y,(.5*B+FM/3.*Y)*Y*Y+QG2/((B+FM*Y)*Y)
10    X1=X
      END
      SUBROUTINE SLOPE(X,Y,DYX)
      REAL Y(1),DYX(1)
      COMMON B,FM,FM2,TWOM,FNQ,QG2,SO
      A=(B+FM*Y(1))*Y(1)
      SF=(FNQ*((B+FM2*Y(1))/A)**.66666667/A)**2
      DYX(1)=(SO-SF)/(1.-QG2*(B+TWOM*Y(1))/A**3)
      RETURN
      END
```

Listing of Program GVFDYXK.C (a C Program) That Calls on the Runge–Kutta Method

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float b,m,m2,twom,fnq,qg2,so;
void rukust(int n,float *dxs,float xbeg,float xend,\
   float err,float *y,float *ytt);
```

```
void slope (float x,float *y, float *dy){ float a,sf;
   a=(b+m*y[0])*y[0]; sf=pow(fnq*pow((b+m2*y[0])/a,.66666667)/a,2.);
   dy[0]=(so-sf)/(1.-qg2*(b+twom*y[0])/pow(a,3.));
   return;} //end slope
void main(void){ int i,no;
   float q,n,xbeg,xend,dx,ybeg,g,xstart,*dxs,x,x1,c=1.,y[1],ytt[1];
   FILE *fil; char filnam[20],fmt[]="%10.1f %9.3f %9.1f\n";
   printf("Give name of output file\n");scanf("%s",filnam);
   if((fil=fopen(filnam,"w"))==NULL){
      printf("File cannot be opened\n"); exit(0);}
   printf("Give:Q,b,m,So,n,xbeg,xend,DX,Ybeg,g,xstart\n");
   scanf("%f %f %f %f %f %f %f %f %f %f %f",
      &q,&b,&m,&so,&n,&xbeg,&xend,&dx,&ybeg,&g,&xstart);
   if(g>30.) c=1.486; no=fabs(xend-xbeg)/fabs(dx)+.5;
   qg2=q*q/g; twom=2.*m;fnq=n*q/c; *dxs=xstart; m2=2.*sqrt(m*m+1.);
   y[0]=ybeg; x1=xbeg;
   fprintf(fil,fmt,x1,y[0],(.5*b+m/3.*y[0])*y[0]*y[0]+qg2/
      ((b+m*y[0])*y[0]));
   for(i=0;i<no;i++){x=x1+dx; rukust(1,dxs,x1,x,1.e-5,y,ytt);
      fprintf(fil,fmt,x,y[0],(.5*b+m/3.*y[0])*y[0]*y[0]+qg2/\
         ((b+m*y[0])*y[0]));
      x1=x;}
   fclose(fil);}
```

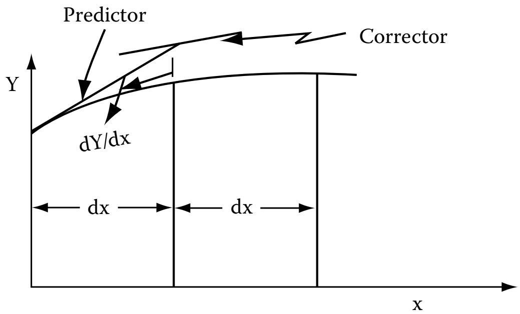
4.4 NUMERICAL METHODS FOR SOLVING ODEs

In obtaining the solution to Example Problem 4.4, it was assumed that a solution to Equation 4.4 was readily available in the form of a tabular list of depths Y as a function of the positions x. This tabular solution was obtained by utilizing an available numerical differential equation solver, such as described in Appendix C. Problems involving ODEs are common in all scientific and engineering fields, and therefore much is known about obtaining closed forms and numerical solutions to them. Solving an ODE involves more than just a numerical integration because the derivative(s) of the dependent variable(s) are functions of both the independent and the dependent variables, whereas if the equation can be separated so that each side of the equation involves only one variable, then both sides of the equation can be integrated, if not in the closed form then numerically. It is because the right side of Equation 4.4 is only a function of Y was it possible to reverse the role of the variables, and solve Example Problems 4.2 and 4.3 by a numerical integration. In the more general ODE for gradually varied flow, Equation 4.6, the derivative dY/dx, is a function of both x and Y. The varied flow equation is special in that it is a first order ODE, i.e., only involves a first derivative.

The emphasis in this book is on the utilization of algorithms that have been developed for numerically solving problems governed by ODEs. However, it is desirable to give a brief overview of the general techniques employed for their solution. Readers interested in more detail should consult the appropriate portions of one of the many available books dealing with the subject of numerical analysis. The starting of the numerical solutions to ODEs generally utilizes the **Runge–Kutta** or the **Euler** method. After the solution has been obtained over a few intervals of Δx, it can be continued using **predictor–corrector** methods. **Runge–Kutta** type methods can start and continue the solution over an interval of as many Δx's as desired. The fourth-order Runge–Kutta method is commonly used by engineers and scientists and is described in Appendix C. You should read this appendix in conjunction with the following pages.

The Euler method is illustrated in the sketch below. The dependent variable Y_o is known at the starting point x_o (referred to as an initial value problem), and therefore the derivative dY/dx can be evaluated at x_o. By multiplying dY/dx by the interval Δx, an estimate of Y_1 at $x_o + \Delta x = x_1$ can be obtained by

$$Y_1 = Y_o + \Delta x\left(\frac{dY}{dx}\right)_0$$



Now that an estimate of Y_1 is available, a better estimate can be obtained for the derivative at x_1 by using the average of the derivatives at x_o and x_1 or

$$Y_1^{(1)} = Y_o + \frac{1}{2}\Delta x\left[\left(\frac{dY}{dx}\right)_0 + \left(\frac{dY}{dx}\right)_1\right]$$

since $Y_1^{(1)}$ is an improvement over Y_1, the above equation can be repeatedly applied giving $Y_1^{(2)}$, $Y_1^{(3)}$. . $Y_1^{(i)}$ always using the most recent Y_1 to evaluate $(dY/dx)_1$. This second equation is called Euler's corrector. In practice, Euler's corrector is generally only applied once since it is more numerically efficient to use a smaller Δx than to apply the corrector more than once. This process could be applied repeatedly for the entire range of x for which the dependent variable Y is sought. However, once Y's are known corresponding to several x's at a spacing of Δx, then higher order polynomials can be used to both predict ahead and correct this prediction. These are referred to as predictor–corrector methods.

Milne's method is one such means of continuing the solution. The predictor equation of Milne's method is

$$Y_{i+1} = Y_{i-3} + \frac{4}{3}\Delta x\left[2\left(\frac{dY}{dx}\right)_{i-2} - \left(\frac{dY}{dx}\right)_{i-1} + 2\left(\frac{dY}{dx}\right)_i\right]$$

in which the i subscripts refer to the ends of the different intervals for which Y has been obtained, with subscript i at the end of the last such interval, and this equation extrapolates to the end of the next interval, i.e., to i + 1. The corrector for Milne's method is

$$Y_{i+1} = Y_{i-1} + \frac{1}{3}\Delta x\left[\left(\frac{dY}{dx}\right)_{i-1} + 4\left(\frac{dY}{dx}\right)_i + \left(\frac{dY}{dx}\right)_{i+1}\right]$$

The Hamming method is quite widely used and provides more mathematical stability in the numerical solution than the Milne method, as it contains a predictors, a modifier, and a corrector, and a final value equation as given below:

Predictor

$$Y_{I+1}^{(0)} = Y_{i-3} + \frac{4}{3}\Delta x\left[2\left(\frac{dY}{dx}\right)_i - \left(\frac{dY}{dx}\right)_{i-1} + 2\left(\frac{dY}{dx}\right)_{i-2}\right]$$

Modifier

$$Y_{i+1}^{(1)} = Y_{i+1}^{(0)} - \frac{112}{121}\left(Y_i^{(0)} - Y_i^{(j)}\right)$$

Corrector

$$Y_{i+1}^{(j)} = \frac{1}{8}\left\{9Y_i - Y_{i-2} + 3\Delta x\left[\left(\frac{dY}{dx}\right)_{i+1} + 2\left(\frac{dY}{dx}\right)_i - \left(\frac{dY}{dx}\right)_{i-1}\right]\right\}$$

Final value

$$Y_{i+1} = Y_i + \frac{9}{121}\left(Y_{i+1}^{(0)} - Y_{i+1}^{n}\right)$$

Since such predictor–corrector methods rely upon starting the solution with a method, such as Euler's method, and the overall accuracy of the solution cannot be greater than that at its start, it can be argued that continuing a solution with sophisticated methods is not justified. Runge–Kutta methods are such a compromise. Consider using a predictor like the above Euler equation to predict Y at the midpoint of the interval, or at $\Delta x/2$. Then use the value of x and Y at this midpoint to compute the derivative over the interval Δx. Written in equation form, this becomes

$$Y_{i+1} = Y_i + \Delta x\left(\frac{dY}{dx}\right)_{i+\frac{1}{2}}$$

in which the evaluation of the derivative is accomplished as shown below:

$$\left(\frac{dY}{dx}\right)_{i+\frac{1}{2}} = f\left(x_i + \frac{1}{2}\Delta x, Y_i + \frac{1}{2}\Delta Y\right); \quad \Delta Y = \Delta x\left(\frac{dY}{dx}\right)_i$$

in which f( ) represents the function on the right side of the equal sign, e.g., it is a simpler way of writing dY/dx especially when the arguments for its evaluation are given. Because of symmetry, the first-order term is canceled, and the method becomes a second-order method versus the Euler predictor being only a first-order method. By a judicious evaluation of dY/dx, i.e., f( ), it is possible to increase the order of the approximation even further. Such a widely used evaluation gives the following **fourth-order Runge–Kutta** method (for more details see Appendix C):

$$\Delta Y_1 = \Delta x f(x_i, Y_i); \quad \Delta Y_2 = \Delta x f\left(x_i + \frac{1}{2}\Delta x, Y_i + \frac{1}{2}\Delta Y_1\right)$$

$$\Delta Y_3 = \Delta x f\left(x_i + \frac{1}{2}\Delta x, Y_i + \frac{1}{2}\Delta Y_2\right); \quad \Delta Y_4 = \Delta x f(x_i + \Delta x, Y_i + \Delta Y_3)$$

and evaluating Y at i + 1 with

$$Y_{i+1} = Y_i + \frac{1}{6}(\Delta Y_1 + \Delta Y_4) + \frac{1}{3}(\Delta Y_2 + \Delta Y_3) + O(\Delta x^5)$$

in which the quantity $O(\Delta x^5)$ indicates that fifth-order terms are being ignored. The following programs are designed to implement this fourth-order Runge–Kutta method. They use a constant interval Δx. For an accuracy of the solution, the interval should be responsive to the properties of the function being solved, etc., the computer code should implement means for sizing this interval for the accuracy requested. Such a code is given in Appendix C in the subroutine ODESOL, DVERK, and RUKUST, which are described in Appendix C.

FORTRAN Listing of Program RUKUY4.FOR to Solve ODE Equation (Equation 4.4)

```
      COMMON B,FM,FN,SO,Q,Q2G,FM2,FNQ,FMS
      WRITE(6,*)'Give:IOUT,Q,b,m,So,n,Xbeg,Xend,DX,Ybeg,g'
      READ(5,*) IOUT,Q,B,FM,SO,FN,XBEG,XEND,DX,YBEG,G
      Q2G=Q*Q/G
      FNQ=FN*Q
      IF(G.GT.30.) FNQ=FNQ/1.486
      FMS=2.*SQRT(FM*FM+1.)
      FM2=2.*FM
      N=ABS(XEND-XBEG)/ABS(DX)
      Y=YBEG
      WRITE(IOUT,100)XBEG,Y
100   FORMAT(2F10.3)
      DO 10 I=1,N
      CALL RUKU4(X,DX,Y)
      X=XBEG+DX*FLOAT(I)
10    WRITE(IOUT,100) X,Y
      IF(ABS(X-XEND).LT. .001) STOP
      CALL RUKU4(X,XEND-X,Y)
      WRITE(IOUT,100) XEND,Y
      STOP
      END
      SUBROUTINE RUKU4(X,DX,Y)
      XH=X+.5*DX
      DY1=DX*SLOPE(X,Y)
      DY2=DX*SLOPE(XH,Y+DY1/2.)
      DY3=DX*SLOPE(XH,Y+DY2/2.)
      DY4=DX*SLOPE(XH,Y+DY3)
      Y=Y+(DY1+DY4)/6.+(DY2+DY3)/3.
      RETURN
      END
      FUNCTION SLOPE(X,Y)
      COMMON B,FM,FN,SO,Q,Q2G,FM2,FNQ,FMS
      A=(B+FM*Y)*Y
      A2=A*A
      FR2=Q2G*(B+FM2*Y)/(A*A2)
      SF=(FNQ*((B+Y*FMS)/A)**.66666667/A)**2
      SLOPE=(SO-SF)/(1.-FR2)
      RETURN
      END
```

Listing of Program RUKUY4.C

```
#include <conio.h>
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#define sqr(x) x*x
float q,b,m,so,n,xbeg,xend,dx,ybeg,g,q2g,fm2,fnq,fms,x,y;
float slope(float xx,float yy){
   float a,a2,fr2,sf;
   a=(b+m*yy)*yy; a2=a*a; fr2=q2g*(b+fm2*yy)/(a*a2);
   sf=sqr(fnq*pow((b+yy*fms)/a,0.66666667)/a);
   return (so-sf)/(1.-fr2);}
void ruku4(void){
   float xh,dy1,dy2,dy3,dy4;
   xh=x+0.5*dx;
   dy1=dx*slope(xh,y);
   dy2=dx*slope(xh,y+dy1/2.);
   dy3=dx*slope(xh,y+dy2/2.);
   dy4=dx*slope(x+dx,y+dy3);
   y=y+(dy1+dy4)/6.+(dy2+dy3)/3.;}
void main(){ int i,no;
   cprintf("Give:Q,b,m,So,n,Xbeg,Xend,DX,Ybeg,g\r\n");
   scanf("%f %f %f %f %f %f %f %f %f %f",
   &q,&b,&m,&so,&n,&xbeg,&xend,&dx,&ybeg,&g);
   q2g=q*q/g; fnq=n*q; if (g > 30) fnq=fnq/1.486;
   fms=2*sqrt(m*m+1); fm2=2.*m; x=xbeg;
   no=fabs(xend-xbeg)/fabs(dx); y=ybeg;
   cprintf("rn%10.3f %10.3f\r\n",xbeg,y); x=xbeg;
   for(i=1;i<=no;i++){ ruku4(); x+=dx;
      cprintf("%10.3f %10.3f\r\n",x,y);}}
```

The above program contain a function subprogram SLOPE that can be modified to solve other ODEs. When Equation 4.4 is being solved, it would not be necessary to pass X as an argument of SLOPE, since the dY/dx does not depend upon x.

EXAMPLE PROBLEM 4.5

Modify one of the above programs to solve problems, such as Example Problems 4.2 and 4.3 in which Y was considered the independent variable and x the dependent variable.

Solution

If one is willing to interpret the input prompt so that Xbeg and Xend are the beginning and the ending values for the depth Y, and Ybeg is the beginning value of x, then the only changes needed is to rewrite SLOPE so its arguments are reversed, i.e., `SLOPE(Y,X)` and the statement that computes `SLOPE` be changed to `SLOPE=(1.-FR2)/(SO-SF)`. The PASCAL program has been modified below (including changes in the prompt). Thus, it is possible to use a differential equation solver to perform a numerical integration with a modest amount of additional computer effort.

```
Program RuKuX4;
Var I,No,IOUT:integer;
Q,b,m,So,n,Xbeg,Xend,DX,Ybeg,g,Q2G,FM2,FNQ,FMS,X,Y:real;
Function Expn(a,b:real):real;Begin if a<0 then Writeln
('error in power',a,b)
  else Expn:=Exp(b*Ln(a)) End;
Function SLOPE(yy,xx:real):real;
 Var A,A2,FR2,SF:real;
 Begin
    A:=(b+m*yy)*yy;
    A2:=A*A;
    FR2:=Q2G*(B+FM2*yy)/(A*A2);
    SF:=sqr(FNQ*Expn((b+yy*FMS)/A,0.66666667)/A);
    SLOPE:=(1.-FR2)/(So-SF);
 End;
Procedure RUKU4;
 Var XH,DY1,DY2,DY3,DY4:real;
 Begin
   XH:=X+0.5*DX;
   DY1:=DX*SLOPE(XH,X);
   DY2:=DX*SLOPE(XH,X+DY1/2);
   DY3:=DX*SLOPE(XH,X+DY2/2);
   DY4:=DX*SLOPE(XH,X+DY3/2);
   Y:=Y+(DY1+DY4)/6+(DY2+DY3)/3
 End;
BEGIN
  Writeln('Give:Q,b,m,So,n,Ybeg,Yend,DY,Xbeg,g');
  Readln(Q,b,m,So,n,Xbeg,Xend,DX,Ybeg,g);
  Q2G:=Q*Q/g; FNQ:=n*Q; If g > 30 then FNQ:=FNQ/1.486;
  FMS:=2*sqrt(m*m+1); FM2:=2*m; X:=Xbeg;
  No:=Trunc(abs(Xend-Xbeg)/abs(DX)); Y:=Ybeg;
  Writeln(Xbeg:10:3,Y:10:3); X:=Xbeg;
  For I:=1 to No do Begin RUKU4;
   X:=X+DX; Writeln(X:10:3,Y:10:3) End;
   If abs(X-Xend) > 0.001 Then begin
    DX:=Xend-X; RUKU4; Writeln(Xend:10:3,Y:10:3) end;
END.
```

EXAMPLE PROBLEM 4.6

Water flows at a rate Q = 20 m^3/s in a 2.5 m wide, with m = 1 trapezoidal channel. The upstream channel is mild, and changes to a steep channel with a bottom slope of S_o = 0.012 and n = 0.014. At a distance L = 100 m from the break in grade, the channel discharges into a reservoir with a water surface elevation H_2 = 3.0 m above the channel bottom. Locate the position of the hydraulic jump. Also, solve the problem using Chezy's equation (e = 0.0012 m and ν = 1.003 × 10^{-6} m^2/s).

Solution

First computing critical conditions gives Y_c = 1.516 m, with E_c = 2.066 m and M_c = 10.73 m^3. Therefore, the problem consists of solving: (1) the S_2-GVF profile downstream from the break in grade, (2) the S_1-GVF profile upstream from the reservoir, and (3) determining the position where the momentum functions for these two GVF profiles are equal. Portions of these two solutions are shown below. The jump occurs at a position x = 22 m downstream from the break in grade.

S_2 GVF with
Q = 20.00, S_o = .012000, n = .0140,
B = 2.5, m = 1.00.

x	Y	E	M	
0.000	1.500	2.066	10.7	
2.000	1.399	2.085	10.8	
4.000	1.357	2.101	10.9	
.				
18.00	1.225	2.205	11.4	
20.00	1.213	2.218	11.5	
22.00	1.203	2.230	11.5	←
24.00	1.194	2.243	11.6	
26.00	1.185	2.255	11.6	

S_1 GVF with
Q = 20.00, S_o = .012000, n = .0140,
B = 2.5, m = 1.00.

x	Y	E	M	
100.0	3.000	3.075	22.7	
98.00	2.974	3.051	22.3	
.				
28.00	1.980	2.239	12.1	
26.00	1.944	2.217	11.9	
24.00	1.906	2.195	11.7	
22.00	1.866	2.173	11.5	←
20.00	1.823	2.151	11.3	
18.00	1.777	2.130	11.2	

The above solutions were obtained using the programs that called on ODESOL as were used to solve Example Problem 4.3. Essential identical solutions will be obtained using the following program that calls on the Runge–Kutta fourth-order method described in Appendix C, if the following input is used:

```
3 20 2.5 1 .012 .014 0 100 2 1.5 9.81 5 & 3 20 2.5 1 .012 .014
  100 12 2 3. 9.81 5
```

Program EPRB4_6.FOR

```
      COMMON B,FM,FN,SO,Q,Q2G,FNQ,FMS,G,C
      WRITE(6,*)'Give:IOUT,Q,b,m,So,n,Xbeg,Xend,DX,Ybeg,g,Nsetp'
      READ(5,*) IOUT,Q,B,FM,SO,FN,XBEG,XEND,DX,YB,G,NSTEP
      C=1.
      IF(G.GT.30.) C=1.486
      FNQ=FN*Q/C
      Q2G=Q*Q/G
      FMS=2.*SQRT(FM*FM+1.)
      N=ABS(XEND-XBEG)/ABS(DX)*FLOAT(NSTEP)
      DX=DX/FLOAT(NSTEP)
      Y=YB
      FMOM=(.5*B+FM/3.*Y)*Y*Y+Q2G/((B+FM*Y)*Y)
      WRITE(IOUT,100)XBEG,Y,FMOM
100   FORMAT(F10.1,F10.3,F10.2)
      DO 10 I=1,N
      CALL RUKU4(X,DX,Y)
      X=XBEG+DX*FLOAT(I)
      FMOM=(.5*B+FM/3.*Y)*Y*Y+Q2G/((B+FM*Y)*Y)
10    IF(MOD(I,NSTEP).EQ.0) WRITE(IOUT,100)X,Y,FMOM
      END
      SUBROUTINE RUKU4(X,DX,Y)
      XH=X+.5*DX
      DY1=DX*SLOPE(X,Y)
      DY2=DX*SLOPE(XH,Y+DY1/2.)
      DY3=DX*SLOPE(XH,Y+DY2/2.)
      DY4=DX*SLOPE(X+DX,Y+DY3)
      Y=Y+(DY1+DY4)/6.+(DY2+DY3)/3.
      RETURN
      END
      FUNCTION SLOPE(X,Y)
```

```
      COMMON B,FM,FN,SO,Q,Q2G,FNQ,FMS,G,C
      A=(B+FM*Y)*Y
      SF=(FNQ*((B+FMS*Y)/A)**.66666667/A)**2
      SLOPE=(SO-SF)/(1.-Q2G*(B+2.*FM*Y)/A**3)
      RETURN
      END
```

In this program, that uses the fixed-step Runge–Kutta method, the number of intermediate computation steps consists of the value given to NSETP, which duplicates the above solutions when equal to 5. If NSETP = 1, then the solution of the S_2 GVF profile is as follows:

x	Y	M
0.0	1.500	10.73
2.0	1.301	11.10
4.0	1.281	11.17

The difference of Y = 1.301 m to Y = 1.399 m is significant over the first 2 m increments used in the computations. The reason is that dY/dx is infinite at the critical depth and starting at a depth of 1.5 m is too close to the critical depth for a fixed-step size algorithm to provide accurate answers unless the step is a fraction of 2 m.

A better approach is to use what is called an "Adaptive Step Size" ODE-Solver. ODESOL, DVERK, and RUKUST fall in this category, i.e., they find a step size that is consistent with the error condition specified. For example, if RUKUST is used, then there is no need for the NSETP (number of intermediate steps) to be greater than 1 to achieve accuracy. Program EPRB4_6 is modified below to call on RUKUST.

Program EPRB4_6A.FOR

```
      COMMON B,FM,FN,SO,Q,Q2G,FNQ,FMS
      REAL Y(1),YTT(1)
      WRITE(*,*)' Give:IOUT,Q,b,m,So,n,Xbeg,Xend,DX,Ybeg,g'
      READ(*,*) IOUT,Q,B,FM,SO,FN,XBEG,XEND,DX,YB,G
      C=1
      IF(G.GT.30.) C=1.486
      FNQ=FN*Q/C
      Q2G=Q*Q/G
      FMS=2.*SQRT(FM*FM+1.)
      N=ABS(XEND-XBEG)/ABS(DX)
      Y(1)=YB
      FMOM=(.5*B+FM/3.*YB)*YB*YB+Q2G/((B+FM*YB)*YB)
      WRITE(IOUT,100) XBEG,Y,FMOM
100   FORMAT(F10.1,F10.3,F10.2)
      DXS=.1
      X1=XBEG
      DO 10 I=1,N
      X2=X1+DX
      CALL RUKUST(1,DXS,X1,X2,1.E-5,Y,YTT)
      FMOM=(.5*B+FM/3.*Y(1))*Y(1)**2+Q2G/((B+FM*Y(1))*Y(1))
      WRITE(IOUT,100) X2,Y,FMOM
```

```
10    X1=X2
      END
      SUBROUTINE SLOPE(X,Y,DYX)
      REAL Y(1),DYX(1)
      COMMON B,FM,FN,SO,Q,Q2G,FNQ,FMS
      A=(B+FM*Y(1))*Y(1)
      SF=(FNQ*((B+FMS*Y(1))/A)**.66666667/A)**2
      DYX(1)=(SO-SF)/(1.-Q2G*(B+2.*FM*Y(1))/A**3)
      RETURN
      END
```

The input needed to solve Example Problem 4.2 is

```
3 450 10 1.4 .0008 .015 6 4.75 −.25 0 32.2
```

with Solution = −4508.8 ft

Listing of Program RUKUSTS.C

```
#include <conio.h>
#include <math.h>
#include <stdlib.h>
#include <stdio.h>
extern void rukust(int neq,float *dxs,float xbeg,float xend,\
   float error,float *y, float *ytt);
float b,m,n,so,q,c=1,q2g,fnq,fms;
   void slope(float x,float *y,float *dy){ float a,fr2,sf;
   a=(b+m*y[0])*y[0]; fr2=q2g*(b+2.*m*y[0])/(a*a*a);
   sf=pow(fnq*pow((b+fms*y[0])/a,.66666667)/a,2.);
   dy[0]=(so-sf)/(1.-fr2); /* End of function slope */
void main(void){float g,xbeg,xend,dx,ybeg,x,*y,*ytt,*dxs;
   int i,nm; char fnam[20];
   FILE *filo; printf("Give output file\n"); scanf("%s",fnam);
   if((filo=fopen(fnam,"w"))==NULL){
      printf("Can not open output file %s",fnam);exit(0);}
   cprintf("Give: Q,b,m,So,n,Xbeg,Xend,DX,Ybeg,gr\n");
   scanf("%f %f %f %f %f %f %f %f %f %f",
      &q,&b,&m,&so,&n,&xbeg,&xend,\&dx,&ybeg,&g);
   y=(float *)calloc(1,sizeof(float));
   ytt=(float *)calloc(1,sizeof(float));
   dxs=(float *)calloc(1,sizeof(float));
   if(g>30.) c=1.486; q2g=q*q/g; fnq=n*q/c;fms=2.*sqrt(m*m+1.);
   nm=fabs(xend-xbeg)/fabs(dx);
   y[0]=ybeg;x=xbeg;
   fprintf(filo,"%10.1f %10.3f\n",xbeg,ybeg);
   cprintf("%10.1f %10.3f\r\n",xbeg,ybeg);*dxs=.1*dx;
   for(i=1;i<=nm;i++){rukust(1,dxs,x,x+dx,1.e-4,y,ytt); x+=dx;
      fprintf(filo,"%10.1f %10.3f\n",x,*y);
      cprintf("%10.1f %10.3f\r\n",x,*y);}}
```

The Program EPRB4_6B.FOR is a modification of the above Program EPRB4_6A.FOR that is designed to solve the problem using Chezy's equation, rather than Manning's equation. Notice in the input that n is replaced by e and ν.

Listing of Program EPRB4_6B.FOR

```
      COMMON B,FM,e,SO,Q,Q2G,FMS,SQG,QV4,SG,C
      REAL Y(1),YTT(1)
      WRITE(*,*)' Give:IOUT,Q,b,m,So,Chezy e,VISC,Xbeg,
     &Xend,DX,Ybeg,g'
      READ(*,*) IOUT,Q,B,FM,SO,e,VISC,XBEG,XEND,DX,YB,G
      e=e/12.
      FMS=2.*SQRT(FM*FM+1.)
      SG=SQRT(G)
      SQG=-SQRT(32.*G)
      QV4=4.*Q/VISC
      C=SQG*ALOG10(e/(((B+FM*YB)*YB)/(B+FMS*YB)))
      Q2G=Q*Q/G
      N=ABS(XEND-XBEG)/ABS(DX)
      Y(1)=YB
      FMOM=(.5*B+FM/3.*YB)*YB*YB+Q2G/((B+FM*YB)*YB)
      WRITE(IOUT,100) XBEG,Y,FMOM
100   FORMAT(F10.1,F10.3,F10.2)
      DXS=.1
      X1=XBEG
      DO 10 I=1,N
      X2=X1+DX
      CALL RUKUST(1,DXS,X1,X2,1.E-5,Y,YTT)
      FMOM=(.5*B+FM/3.*Y(1))*Y(1)**2+Q2G/((B+FM*Y(1))*Y(1))
      WRITE(IOUT,100) X2,Y,FMOM
10    X1=X2
      END
      SUBROUTINE SLOPE(X,Y,DYX)
      REAL Y(1),DYX(1)
      COMMON B,FM,e,SO,Q,Q2G,FMS,SQG,QV4,SG,C
      A=(B+FM*Y(1))*Y(1)
      P=B+FMS*Y(1)
      Rh=A/P
1     C1=C
      C=SQG*ALOG10(E/Rh+.884*C/(SG*QV4/P))
      IF(ABS(C-C1).GT. 1.E-5) GO TO 1
      SF=(Q/(C*A))**2/Rh
      DYX(1)=(SO-SF)/(1.-Q2G*(B+2.*FM*Y(1))/A**3)
      RETURN
      END
```

Inputs to solve upstream and downstream GVFs:

```
3 20 2.5 1 .012 .0012 1.003e−6 0 100 1.5 9.81 and 3 20 2.5 1 .012
  .0012 1.003e−6 100 12 3 9.81
```

Solution outputs:

x (m)	Y (m)	M (m^3)	
.0	1.500	10.73	
2.0	1.399	10.83	
.	.	.	
20.0	1.213	11.49	
22.0	1.202	11.55	←
24.0	1.193	11.60	
26.0	1.184	11.65	
.	.	.	
98.0	1.040	12.80	
100.0	1.039	12.82	
100.0	3.000	22.72	
98.0	2.974	22.33	
.	.	.	
26.0	1.946	11.90	
24.0	1.908	11.72	
22.0	1.869	11.54	←
20.0	1.827	11.36	
.	.	.	
12.0	1.571	10.7	

Mathcad: This problem is solved below using Mathcad's fourth-order fixed-step size Runge–Kutta numerical differential equation solver. Note, it also does not provide a close solution near the critical depth. In other words, in order to obtain a better solution using a fixed-step size algorithm smaller increments of x are required to be used for the computations of the S_2 GVF profile. You should modify the Mathcad solution to accomplish this.

EPRB4_6.MCD

$$m := 1 \quad So := .012 \quad Q := 20 \quad g := 9.81 \quad n := .014 \quad Cu := 1. \quad Y_0 := 1.5 \quad FQN := n \cdot \frac{Q}{Cu}$$

$$P(Y) := b + 2 \cdot Y \cdot \sqrt{m^2 + 1}$$

$$b := 2.5 \quad A(Y) := (b + m \cdot Y) \cdot Y \quad D(x, Y) := \frac{So - \left[FQN \cdot \left[\frac{\left(\frac{P(Y_o)}{A(Y_o)}\right)^{.6666667}}{A(Y_o)}\right]\right]^2}{1 - Q^2 \cdot \frac{b + 2 \cdot m \cdot Y_o}{g \cdot A\langle Y_o \rangle^3}} \quad i := 0..50$$

$$z := \mathrm{rkfixed}(Y, 0, 100, 50, D)$$

$$m1_i := \left[\cdot 5 \cdot b + \frac{m}{3} \cdot (z^{\langle 1 \rangle})_i\right] \cdot \left[(z^{\langle 1 \rangle})_i\right]^2 + \frac{Q^2}{g \cdot A\left[(z^{\langle 1 \rangle})_i\right]}$$

Z =

	0	1
0	0	1.5
1	2	1.301
2	4	1.281
3	6	1.264
4	8	1.249
5	10	1.235
6	12	1.223
7	14	1.212
8	16	1.202
9	18	1.192
10	20	1.184
11	22	1.176
12	24	1.168
13	26	1.161
14	28	1.154

m1 =

	0
0	10.733
1	11.097
2	11.171
3	11.241
4	11.308
5	11.372
6	11.433
7	11.492
8	11.549
9	11.603
10	11.655
11	11.706
12	11.755
13	11.802
14	11.847

$Y_0 := 3 \quad z1 := \mathrm{kfixed}(Y, 100, 0, 50, D) \quad j := 0..50$

$$m2_j := \left[.5 \cdot b + \frac{m}{3} \cdot \left(z1^{<1>}\right)_j\right] \cdot \left[\left(z1^{<1>}\right)_j\right]^2 + \frac{Q^2}{g \cdot A\left[\left(z1^{<1>}\right)_j\right]}$$

z1 =

	0	1
0	100	3
1	98	2.974
2	96	2.949
3	94	2.923
4	92	2.897
5	90	2.871
6	88	2.845
7	86	2.819
8	84	2.793
9	82	2.766
10	80	2.74
11	78	2.714
12	76	2.687
13	74	2.66
14	72	2.633

m2 =

	0
0	22.721
1	22.333
2	21.951
3	21.574
4	21.202
5	20.835
6	20.474
7	20.118
8	19.767
9	19.422
10	19.082
11	18.747
12	18.417
13	18.092
14	17.773

z1 =

	0	1
36	28	1.98
37	26	1.944
38	24	1.906
39	22	1.866
40	20	1.823
41	18	1.777
42	16	1.725
43	14	1.662
44	12	1.556
45	10	1.536
46	8	1.39
47	6	1.464
48	4	1.361
49	2	1.405
50	0	1.566

m2 =

	0
36	12.085
37	11.891
38	11.704
39	11.523
40	11.349
41	11.183
42	11.025
43	10.877
44	10.743
45	10.734
46	10.85
47	10.751
48	10.916
49	10.824
50	10.749

TK-Solver: This problem has also been solved below, using the TK-Solver. Its variable, rule, function, and output table sheets are shown.

Solution for the S2 GVF profile

```
==================== VARIABLE SHEET =====================
St Input──── Name─── Output─── Unit───── Comment─────────
   1.5       y0
L  'yyy      Y
   .014      n
   2.5       b
   1         m
   1         C
   20        Q
   .012      So
   9.81      g
```

```
======================== RULE SHEET ============================
S Rule----------------------------------------------------------
  place(Y,1) = y0
  call RK4_se('DYDx,Y,'x)
```

The call RK4_se is a library in the TK-Solver package (that uses a fixed-step size) and it requires that the user supply another function that defines the ODE that is being solved, which is given as FUNCTION DYDx below.

```
================ PROCEDURE FUNCTION: RK4_se (DIFFEQ\Runge-Kutta)
Comment:             Classical Fourth-Order Runge-Kutta method,
                     single eq-n
Parameter Variables:
Input Variables:     EQ,y,x
Output Variables:
S Statement-----------------------------------------------------------
  ; Notation: EQ  name of a function with the 1st-order equation
                  y'=f(x,y)
  ;           x   independent variable (list)
  ;           y   dependent variable, y=F(x), (list)
  ;           K   Runge-Kutta coefficients (list)

  ; Description: This procedure represents an implementation of a
  ; classical 4th-order Runge-Kutta procedure for numerical
  ; integration of a single ordinary differential equation
  ; y'=f(x,y). Given a function name passed as a symbolic
  ; value of the Input Variable EQ, list of values of the
  ; independent variable x, and an initial condition as the value of
  ; the 1st element of the list y, the procedure generates the
  ; solution in the rest of the list y.
  xi:= x[1]
  yi:= y[1]
  for i=2 to length(x)
     ye:= yi
     h:= (x[i]-xi)/2
     for j=1 to 3
        'K[j]:= apply(EQ,xi,ye)
        if mod(j,2) then xi:= xi + h
        if j=3 then h:= 2*h
        ye:= yi + h*'K[j]
     next j
     'K[4]:= apply(EQ,xi,ye)
     yi:= yi + dot('K,1,2,2,1)*h/6
     y[i]:= yi
  next i
  call delete('K)

==================== PROCEDURE FUNCTION: DYDx ===========
Comment:              Given differential equation
Parameter Variables:  n,b,m,C,Q,So,g
Input Variables:      x,y
Output Variables:     y'
```

```
S Statement————————————————————————————————————————————————
  A:=(b+m*y)*y
  NUM:= So-(n*(b+2*sqrt(m*m+1)*y)^.66666667*Q/(C*A^1.666667))^2
  DEN:=1-Q*Q*(b+2*m*y)/(g*A^3)
  y':=NUM/DEN
```

To solve the problem, the following list sheet was created and the x sheet filled in with values starting with 0 and running through 100 in increments of 2 m.

```
==================== LIST SHEET ========================
Name————— Elements—— Unit————— Comment——————————————————
x         51                   independent variable
yyy       51                   dependent variable
Y         1                    names of dependent variables,
                               set of solutions
```

In order to obtain the values of the momentum function, the rules that solve the ODE were taken out, and the additional rule that computes the momentum function added, as shown below,

```
==================== RULE SHEET ========================
S Rule——————————————————————————————————————————————————
C place(Y,1) = y0
C call RK4_se('DYDx,Y,'x)
  M=(.5*b+m/3*YY)*YY^2+Q^2/(g*(b+m*YY)*YY)
```

and the solution for the depth copied into the list variable YY before pressing L10 for a list solve.

```
==================== VARIABLE SHEET ====================
St Input———— Name——— Output——— Unit————— Comment————————
   1.5       y0
L  'yyy      Y
   .014      n
   2.5       b
   1         m
   1         C
   20        Q
   .012      So
   9.81      g
L  3         YY
LG 10        M
```

```
==================== TABLE: Solution ===================
Title:         y  =  f(x)
Element x————————— y————————— M————————— ——————— ———————
1       0          1.5        10.7332866
2       2          1.30055004 11.0968582
3       4          1.28091077 11.1707535
 .       .            .          .
10      18         1.19248127 11.6029712
11      20         1.18370612 11.6553916
12      22         1.17552185 11.7059391
```

```
13       24          1.16786187 11.7547211 ⟵ Jump
14       26          1.16067065 11.8018339
 .        .              .          .
```

Following the same procedure, except using a value of y0 = 3, and filling in the list table for x starting with 100 and moving back toward 0 results in the following solution for the S_1 gradually varied profile beyond the hydraulic jump. The momentum functions are equal at a position x between 22 and 24 m according to these solutions, which agree closely with the above solution at 22 m, despite the fact that the fixed step algorithm does not provide a good solution close to critical, but since the depth varies rapidly over a small distance here, the computed position of the jump, from a practical viewpoint, is not so very different.

```
═══════════════════ TABLE: Solution ═══════════════════
Title:          y  =  f(x)
Element x───────── y───────── M─────────
1       100        3          22.7211951
2       98         2.9743262  22.3333936
3       96         2.94859016 21.9509297
.        .             .          .
37      28         1.98014744 12.0854955
38      26         1.94377088 11.8913888
39      24         1.905841   11.7037361 ⟵ Jump
40      22         1.86593079 11.5228252
41      20         1.82338687 11.3490387
```

In summary, one should note that an ODE solver can be used to solve dx/dY (or dx/dE) in which x is the dependent variable, as well as dY/dx, in which Y is the dependent variable. However, a numerical integrator can only be used to solve the first form of the ODE in which x is the dependent variable. A numerical integrator requires that the derivatives be only a function of the independent variable. An ODE solver allows the derivative to be a function of both the dependent and independent variables, or just one of these.

4.5 CANAL SYSTEMS

Most of the previous considerations have dealt with an individual channel. As the slope of the land changes, channels are built with changing bottom slopes. To keep the costs near a minimum, the size of the channel will be reduced where the bottom slope increases, and vice versa. With a subcritical flow in such systems, it is easy for a transition from a larger to a smaller channel to cause an M_1 GVF profile even if the normal depth in the smaller channel is less than the normal depth in the larger channel, unless the position of the bottom is changed. This choking effect is very likely to occur especially if the smaller channel is steep, and the critical flow occurs at its beginning.

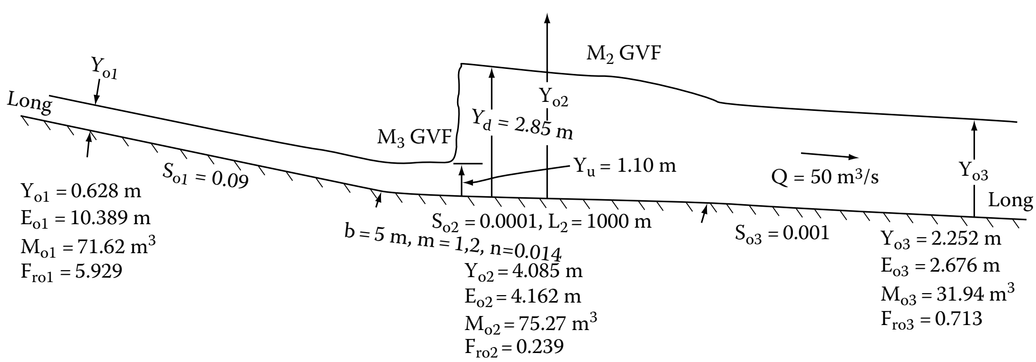
Gates are used to control the flow rates and the water depths in canal systems. The effects of these gates must extend to the supply reservoirs of the system if they are to control the flow rates, as well as the depths. To determine if a gate does control the flow into the channel, and how much it will reduce the flow rate depending upon its setting requires that the GVF profile be solved from the gate up to the supply reservoir through whatever the canal system between the two consists of. The actual solution to such problems involving downstream gate control are governed by the ODE for the GVF, plus the algebraic equations that define energy, critical flow, and possibly momentum, depending upon what is involved in the given situation. The only means available for solving problems defined by simultaneous ODEs and algebraic equations is by trial-and-error or iterative methods such as the Newton method. Thus, canal systems with control gates need to be solved by (1) guessing the flow rate into the system, (2) solving the algebraic equations involved in the problem, (3) solving the GVF profiles and (4) upon completing these solutions, verifying whether the guessed flow rate satisfies all conditions. If not, adjust the assumed flow rate and repeat the above steps until all conditions are satisfied. In the next section, use of the Newton method will be discussed to automate the solution.

EXAMPLE PROBLEM 4.7

Determine the depth, etc., through a channel system consisting of an upstream steep channel followed by a mild channel, and this channel followed by another mild channel but with a larger slope than the middle channel. Channel # 1 and Channel # 3 are very long and the middle channel has a length of $L_2 = 1000\,m$. The flow rate is $Q = 50\,m^3/s$, and all channels have a Manning's $n = 0.014$. Consider (a) the case where all three channels are trapezoidal and have the same size, i.e., $b = 5\,m$, and a side slope $m = 1.2$, and (b) the case where the channels have different sizes, i.e., $b_1 = 5\,m$ (rectangular), $b_2 = 5\,m$, $m_2 = 1.5$, and $b_3 = 5\,m$, $m_3 = 1.2$. The middle and the downstream channel have bottom slopes of $S_{o2} = 0.0001$ and $S_{o3} = 0.001$. Consider two different slopes for the upstream channel, $S_{o1} = 0.09$ and $S_{o1} = 0.01$.

Solution

The solutions are shown on the sketches below. First, the normal depths are computed for all channels for all cases. Since the downstream long channel will cause the depth at its beginning to always be at its normal depth, an M_2 GVF will exist in the downstream portion of the middle channel. Therefore, the next step is to solve this M_2 GVF profile. Should the momentum function M_2 associated with the depth Y_2 thus computed at the beginning of the middle channel be less than M_{o1}, then the hydraulic jump will occur in the middle channel, and its location is determined where the momentum functions from the M_3 and M_2 GVFs are equal. Should $M_2 > M_{o1}$, then an S_1 GVF will exist in Channel # 1. When $S_{o1} = 0.09$, the jump occurs downstream for both the constant size channel and the one with transitions. When $S_{o1} = 0.01$, the jump occurs upstream in the steep channel since $M_2 = 40.2 > M_{o1} = 37.6\,m^3$. However, when there are transitions in the channel, the specific energy $E_2 = 2.935\,m$ associated with $Y_2 = 2.774\,m$ is less than the critical specific energy in the upstream rectangular channel. Thus, even though $M_2 = 42.81 > M_{o1} = 40.03\,m^3$, the jump cannot occur upstream from the transition, and therefore must be within the transition.



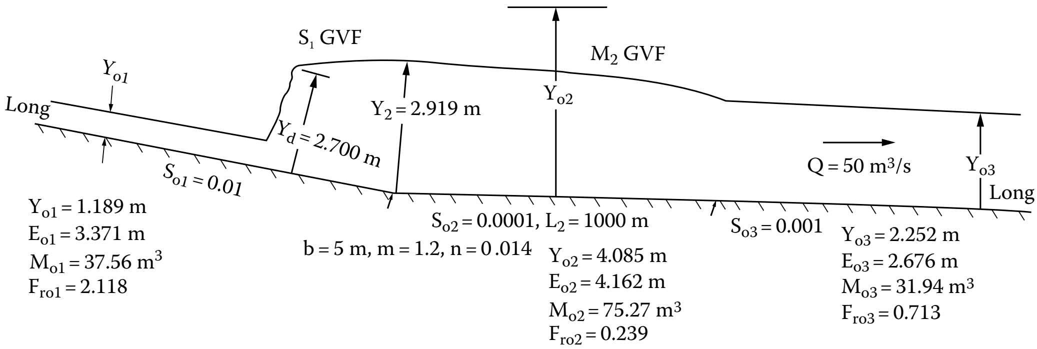
Solve ODE for $M_2 -> Y_2 = 2.919\,m$, $M_2 = 41.51\,m^3$
Jump occurs in middle channel since $M_{o1} > M_2$
Maching $M_u = M_d$ from solutions of ODE for M_3 and M_2 GVF
gives $x = 180\,m$, with $Y_u = 1.10\,m$ and $Y_d = 2.85\,m$ ($M = 40.2\,m^3$)

M_3—GVF solution

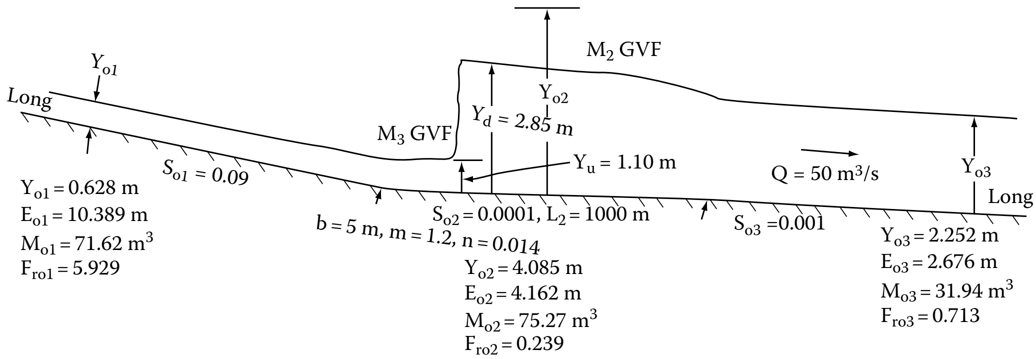
x (m)	Y (m)	E (m)	M (m³)	
.0	.628	10.388	71.615	
20.0	.680	8.817	65.683	
.	.	.	.	
140.0	.991	4.379	44.397	
160.0	1.044	4.036	42.228	
180.0	1.097	3.749	40.299	⟵Jump
200.0	1.152	3.507	38.578	
220.0	1.209	3.304	37.039	

M_2—GVF solution

x (m)	Y (m)	E (m)	M (m³)	
1000.0	2.252	2.675	31.939	
980.0	2.287	2.693	32.245	
.	.	.	.	
220.0	2.836	3.060	39.927	
200.0	2.844	3.067	40.078	
180.0	2.852	3.073	40.227	⟵Jump
160.0	2.860	3.079	40.376	
140.0	2.868	3.085	40.522	



Now equate $M_d = M_{o1} = 37.56$ and solve $Y_d = 2.700$ m
Solve $dx/dY = (1-F_r^2)/(S_o - S_f)$ between Y_d and $Y_2 = 2.919$ m
to get x = −17.6 m



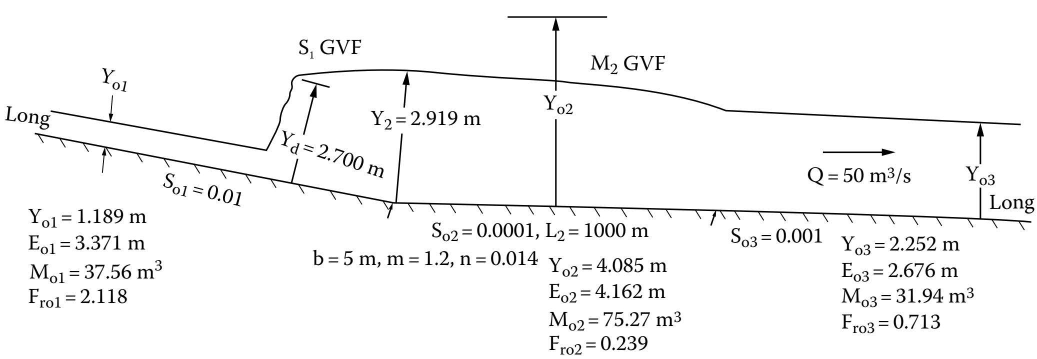
Solve ODE for $M_2 \rightarrow Y_2 = 2.919$ m, $M_2 = 41.51$ m^3
Jump occurs in middle channel since $M_{o1} > M_2$
Matching $M_u = M_d$ from solutions of ODE for M_3 and M_2 GVF
gives x = 180 m, with $Y_u = 1.10$ m and $Y_d = 2.85$ m ($M = 40.2$ m^3)

M_3—GVF solution

x (m)	Y (m)	E (m)	M (m³)	
.0	.628	10.388	71.615	
20.0	.680	8.817	65.683	
.	.	.	.	
140.0	.991	4.379	44.397	
160.0	1.044	4.036	42.228	
180.0	1.097	3.749	40.299	←Jump
200.0	1.152	3.507	38.578	
220.0	1.209	3.304	37.039	

M_2—GVF solution

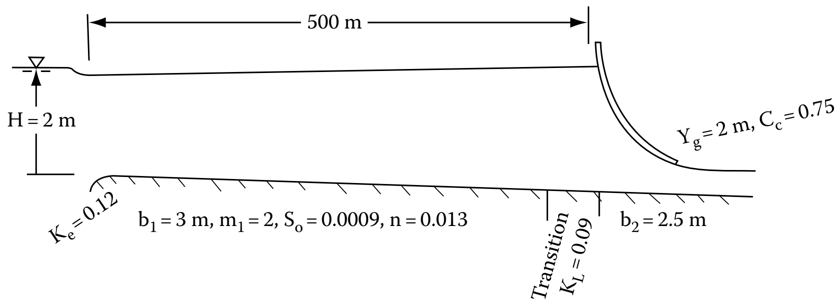
x (m)	Y (m)	E (m)	M (m³)	
1000.0	2.252	2.675	31.939	
980.0	2.287	2.693	32.245	
.	.	.	.	
220.0	2.836	3.060	39.927	
200.0	2.844	3.067	40.076	
180.0	2.852	3.073	40.227	←Jump
160.0	2.860	3.079	40.376	
140.0	2.868	3.085	40.522	



Now equate $M_d = M_{o1} = 37.56$ and solve $Y_d = 2.700$ m
Solve $dx/dy = (1 - F_r^2)/(S_o - S_f)$ between Y_d and $Y_2 = 2.919$ m
to get x = −17.6 m

EXAMPLE PROBLEM 4.8

A gate exists 500 m downstream from a supply reservoir. The channel between the reservoir and the gate has a bottom slope $S_o = 0.0009$, a bottom width $b = 3$ m, a Manning's $n = 0.013$, and a side slope $m = 2$. A short transition with a minor loss coefficient of $K_L = 0.09$ changes the channel to a rectangular shape at the gate with a bottom width of $b = 2.5$ m. The water elevation in the upstream reservoir is 2 m above the bottom of the channel, and the entrance loss coefficient is $K_e = 0.12$. The contraction coefficient for the gate is constant with its setting above the channel bottom and equals $C_c = 0.75$. Determine the discharge through the canal system as a function of the gate setting under the assumption that free flow occurs downstream of the gate, i.e., obtain the discharge for several gate settings allowing this information to be plotted to give a discharge curve of Q versus Y_G.



Solution

The solution might begin by solving Manning's equation simultaneously with the energy equation at the upstream reservoir to determine what the uniform depth and discharge would be. Using this flow rate, the critical depth in the reduced channel at the gate can be solved to provide some guidance about what gate setting to start the requested series of solutions with. These solutions provide $Y_o = 1.688$ m, $Q_o = 25.06\,m^3/s$, and $Y_{c2} = 2.168$ m, with the corresponding specific energy at the end of the trapezoidal channel $E_2 = 3.25$ m. Note that this specific energy is well above the specific energy supplied by the reservoir; therefore an M_1 GVF profile will exist between the reservoir and the gate. The solution will be obtained by writing a FORTRAN program that calls on the ODE solver ODESOL described in Appendix C. This program, whose listing is shown below, is designed to determine the depth of flow downstream from the gate by multiplying the gate setting by the contraction coefficient, and then computing the specific energy upstream from the gate. It reads in the necessary parameters to define the problem, as well as the gate setting for which solutions are to be obtained. The specific energy establishes the depths immediately upstream from the gate, as well as at the end of the trapezoidal channel before the transition reduces its size by using the losses coefficient of 0.09. The program then obtains the solution of the GVF profile to the reservoir and displays a prompt indicating what the depth and the specific energy are at the entrance of the channel with a request that a new guessed flow rate be supplied. If the previous given flow rate is correct, the user supplies a negative flow rate value, the absolute value of which will be used for the next gate setting.

This program solves the ODE $dE/dx = S_o - S_f$, and as such gives an example of an alternative solution methodology. The equation is simpler than the one giving dY/dx, but involves more arithmetic on the part of the computer because S_f is a function of the depth Y, requiring that the specific energy equation $E = Y + Q^2/(2gA^2)$ be solved for Y every time the ODE solver requests that dE/dx be evaluated. For natural channels, there is some advantage in using the ODE in the form of dE/dx, since it automatically takes care of changes in the cross-sectional area without the need for defining a term that involves $\partial A/\partial x$. More information related to natural channels is included in Chapter 5. (You should solve this problem with one of the previously given programs that solves $dY/dx = (S_o - S_f)/(1 - F_r^2)$.)

The gate setting flow rate relationship obtained from this solution process is given in the table below. It should be noted from this solution, that with the gate set at 2 m above the channel bottom, which is only modestly below the critical depth in the rectangular channel here, the gate has reduced the flow rate about 10 m^3/s over what would occur in the trapezoidal channel under uniform flow conditions. You should duplicate this solution for the experience gained in guessing appropriate flow rates.

Solution to Illustrative Problem 45

Gate (m)	Flow Rate (m^3/s)	Depth at Beg. (m)
2.0	14.93	1.93
1.8	14.57	1.93
1.6	13.90	1.94
1.4	12.95	1.95
1.2	11.73	1.96
1.0	10.28	1.97
0.8	8.61	1.98
0.6	6.73	1.99
0.4	4.66	1.99
0.2	2.41	1.99

FORTRAN Listing of Program EPRB4_8.FOR to Assist in Solving Illustrate Example Problem 4.8

```
      CHARACTER*2 UNIT
      LOGICAL SWITCH
      REAL E(1),EPRIME(1),XP(1),YP(1,1),WK1(1,13),YG(20),KE1,KE2
      EXTERNAL DEX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRANS/QN,B1,FM1,B,FM,FN,Y,Q2G,SO,X1,DB,DFM,XBEG,SWITCH
      DATA SWITCH/.FALSE./
      WRITE(6,*)=GIVE: UNIT,IOUT,TOL,DELX,YB,Q,FN,SO,B1,FM1,''B2,
     &FM2,XBEG,XEND,H,CV,KE1,KE2'
      READ(5,*)UNIT,IOUT,TOL,DELX,YB,Q,FN,SO,B1,FM1,B2,FM2,XBEG,
     &XEND,H,CV,KE1,KE2
      WRITE(6,*)' Give N & gate positions'
      READ(5,*) NSET,(YG(I),I=1,NSET)
      H1=-.1
      IF(DELX.GT.0.) H1=ABS(H1)
      CC=1.
      G2=19.62
      IF(UNIT .EQ. 'SI') GO TO 10
      CC=1.486
      G2=64.4
10    DO 70 I=1,NSET
      YY=CV*YG(I)
      Y=YB
12    QN=(FN*Q/CC)**2
      Q2G=Q*Q/G2
      E(1)=YY+(1.+KE2)*Q2G/(((B2+FM2*YY)*YY)**2)
      B=B1
      FM=FM1
      X=XBEG
      WRITE(IOUT,15) YG(I),Q,SO,FN,B1,FM1,B2,FM2
15    FORMAT(/,' SOLUTION TO GRADUALLY VARIED=, = FLOW WITH''
     &Ygate='F8.2,/,' Q=',F10.2,' SO=',F10.6,n=',F7.4,/,'B1=',
     &F8.1,' m1=',F8.2,/, B2=',F8.1,' m2=',F8.2,/1X,30('-'),/,
     &3X, DIST.   DEPTH   E',/1X,30('-'))
      CALL YSOL(E,Y)
```

```
      WRITE(IOUT,40) X,Y,E
20    XZ=X+DELX
      CALL ODESOL(E,EPRIME,1,X,XZ,TOL,H1,HMIN,1,XP,YP,WK1,DEX)
      X=XZ
      WRITE(IOUT,40) X,Y,E
40    FORMAT(1X,3F10.3)
      IF(X.GT.XEND) GO TO 20
      EE=Y+(1.+KE1)*Q2G/(((B+FM*Y)*Y)**2)
      WRITE(*,80) Y,H,EE,Q
80    FORMAT(' Y,H,E,Q',4F12.3,/,' Give new Q (neg O.K.)')
      READ(5,*) QQ
      IF(QQ.GT.0.) THEN
      Q=QQ
      GO TO 12
      ENDIF
      WRITE(4,100) YG(I),Q,Y
70    Q=ABS(QQ)
100   FORMAT(3F12.4)
      STOP
      END
      SUBROUTINE DEX(X,E,EPRIME)
      LOGICAL SWITCH
      REAL E(1),EPRIME(1)
      COMMON /TRANS/QN,B1,FM1,B,FM,FN,Y,Q2G,SO,X1,DB,DFM,XBEG,SWITCH
      CALL YSOL(E,X)
      P=B+2.*SQRT(FM*FM+1.)*Y
      A=(B+FM*Y)*Y
      SF=QN*((P/A)**.6666667/A)**2
      EPRIME(1)=SO-SF
      RETURN
      END
      SUBROUTINE YSOL(E,X)
      REAL E(1)
      LOGICAL SWITCH
      COMMON /TRANS/QN,B1,FM1,B,FM,FN,Y,Q2G,SO,X1,DB,DFM,
     &XBEG,SWITCH
      IF(Y.LT.E(1).AND.Y.GT..6*E(1)) GO TO 10
      Y=.9*E(1)
10    NCT=0
20    NT=0
30    IF(SWITCH) GO TO 40
      XX=1.-(X-X1)/(XBEG-X1)
      B=B1+DB*XX
      FM=FM1+DFM*XX
40    A=(B+FM*Y)*Y
      F=E(1)-Y-Q2G/A**2
      IF(NT.GT.0) GO TO 50
      F1=F
      Y=Y-.001
      NT=1
      GO TO 30
50    Y=Y+.001
      DIF=.001*F1/(F1-F)
      NCT=NCT+1
      Y=Y-DIF
      IF(NCT.LT.15.AND.ABS(DIF).GT. .00001) GO TO 20
```

```
      IF(NCT.EQ.15) WRITE(6,*)' DID NOT CONVERGE'
      RETURN
      END
```

Input to start solution:

```
'SI'  3  .000001  −50  2.1  16  .013  .0009  3  2  2.5  0  500
  0  2  .75 .12 .009 10 2 1.8 1.6 1.4 1.2 1. .8 .6 .4 .2
```

and two file names, one for the data giving the individual profiles, and one for the final solution results given in the previous table. The input not shown above consists of a new guess for the correct flow rate Q, so that the two values of specific energy displayed by the program on the monitor (the second and third values) are equal (or nearly so.) When they are equal, the next flow rate given is preceded by a minus sign and is the first guess for the flow rate that will be used in solving the problem with the next gate position read into the array YG(I). The C program listed below requires the same input, but calls for both file names before the input for the gate settings, whereas the FORTRAN program utilizes the MS-FORTRAN extension of standard FORTRAN 77 to promote a file name when a write first occurs using a logical unit that has not been attached to a file that has been open with the OPEN statement.

Listing of Program EPRB4_8.C Designed to Solve Example Problem 4.8:

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
int swith=0; float qn,b1,fm1,b,fm,fn,y,q2g,so,x1,db,dfm,xbeg;
extern void rukust(int neq,float *dxs,float xbeg,float xend,\
   float error,float *y,float *ytt);
void ysol(float *e, float x){int nct,nt; float xx,a,f,f1,dif;
   if((y>e[0]) || (y<.6*e[0])) y=.9*e[0]; nct=0;
L20: nt=0;
L30: if(!swith){xx=1.-(x-x1)/(xbeg-x1);b=b1+db*xx;fm=fm1+dfm*xx;}
   a=(b+fm*y)*y;f=e[0]-y-q2g/(a*a);
   if(nt==0){f1=f;y-=.001,nt=1; goto L30;}
   y+=.001; dif=.001*f1/(f1-f);nct++; y-=dif;
   if((fabs(dif)>.00001)&&(nct<15)) goto L20;
   if(nct==15) printf("DID NOT CONVERGE\n");} //End of ysol
void slope(float x,float *e,float *eprime){float p,a,sf;
   ysol(e,x); p=b+2.*sqrt(fm*fm+1.)*y; a=(b+fm*y)*y;
   sf=qn*pow(pow(p/a,.6666667)/a,2.);
   eprime[0]=so-sf;} // End of slope
void main(void){char unit[2],fname[20]; FILE *out,*ou1;
   int i,j,iout,nset;
   float tol,delx,yb,q,b2,fm2,xend,h,cv,ke1,ke2,yg[20],x,xz,ee,\
      qq,*h1,cc,g2,yy,e[1],ew[1];
   printf("GIVE:UNIT,IOUT,TOL,DELX,YB,Q,FN,SO,B1,FM1,B2,FM2,\
      XBEG, XEND,H,CV,KE1,KE2\n");
   scanf("%s %d %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f",\
      &unit,&iout,&tol,&delx,&yb,&q,&fn,&so,&b1,&fm1,&b2,&fm2,\
      &xbeg,&xend,&h,&cv,&ke1,&ke2);
   if(iout!=6){printf("Give output file name\n");
      scanf("%s",fname);out=fopen(fname,"wt");}
   printf("Give file for summary sol.\n");
```

```
   scanf("%s",fname);ou1=fopen(fname,"wt");
   printf("Give: N & gate positions\n");
   scanf("%d",&nset);
   for(i=0;i<nset;i++)scanf("%f",&yg[i]); *h1=-.1;
   if(delx>0.) *h1=fabs(*h1); cc=1.; g2=19.62;
   if((unit=="es")||(unit=="ES")){cc=1.486;g2=64.4;}
   for(i=0;i<nset;i++){yy=cv*yg[i]; y=yb;
L12:qn=pow(fn*q/cc,2.); q2g=q*q/g2;
      e[0]=yy+(1.+ke2)*q2g/pow((b2+fm2*yy)*yy,2.);
      b=b1; fm=fm1;x=xbeg;
      if(iout==6){
         printf("nSOLUTION TO GRADUALLY VARIED FLOW WITH Yg=\
         %8.2f\nQ=%10.2f, So=%10.3f, n=%7.4f\n B1=%8.1f,m1=%8.2f\
         \n B2=%8.1f, m2=%8.2f\n",yg[i],q,so,fn,b1,fm1,b2,fm2);
         for(j=0;j<31;j++)printf("-");printf("\n");
         printf("     DIST.     DEPTH     E\n");
         for(j=0;j<31;j++)printf("-"); printf("\n");}
   else {fprintf(out,"nSOLUTION TO GRADUALLY VARIED FLOW WITH \
      Yg=%8.2f\nQ=%10.2f, So=%10.3f, n=%7.4f\n B1=%8.1f,\
      m1=%8.2f\n\B2=%8.1f, m2=%8.2f\n",yg[i],q,so,fn,b1,fm1,b2,fm2);
   for(j=0;j<31;j++)fprintf(out,"-");fprintf(out,"\n");
   fprintf(out,"     DIST.     DEPTH     E\n");
   for(j=0;j<31;j++)fprintf(out,"-");fprintf(out,"\n");}
ysol(e,x);
if(iout==6)printf(" %10.3f %9.3f %9.3f\n",x,y,e[0]);
else fprintf(out," %10.3f %9.3f %9.3f\n",x,y,e[0]);
do{ xz=x+delx; rukust(1,h1,x,xz,tol,e,ew);
   x=xz; if(iout==6) printf(" %10.3f %9.3f %9.3f\n",x,y,e[0]);
   else fprintf(out," %10.3f %9.3f %9.3f\n",x,y,e[0]);
} while(x>xend);
ee=y+(1.+ke1)*q2g/pow((b+fm*y)*y,2.);
printf("Y,H,E,Q\n%12.3f %11.3f %11.3f %11.3f\n",y,h,ee,q);
printf("Give new Q (neg. O.K.)\n"); scanf("%f",&qq);
if(qq>0.){q=qq;goto L12;}
fprintf(ou1,"%12.4f %11.3f %11.3f\n",yg[i],q,y); q=fabs(qq);}
// end of for(i
fclose(ou1);if(iout!=6)fclose(out);}
```

Note that the C program calls on the ODE solver RUKUST, whereas the FORTRAN program calls on the ODESOL. It will be a worthwhile exercise for you to modify the program in the language you are most familiar with to call on the other solver.

4.6 SIMULTANEOUS SOLUTION OF ALGEBRAIC AND ORDINARY DIFFERENTIAL EQUATIONS

We will now consider methods that will allow algebraic and ordinary differential equations to be solved simultaneously. Implementation of these methods in computer code will remove the burden of repeatedly trying values of flow rate, as in Example Problem 4.8, until all equations are satisfied. Instead, the solution can be turned over entirely to the computer. It will be necessary to use some type of iterative process. The Newton method is a good candidate for obtaining this iterative solution since it converges rapidly. As a first step in solving equations by the Newton method, all equations are written as functions of the unknowns equal to zero. Each ODE might be considered a function of unknowns. When the correct combination of these unknowns are used, then each ODE will produce, as its solution, the correct value of its dependent variable. This variable is one of the variables in the algebraic equations. For example, if the variable that the ODE is solving is the depth Y_1 at the beginning of the channel, then let Y_1 be the variable used in the algebraic equations, and let Y_{ode1} be the solution of the upstream depth produced by the solution of the ODE. This solution will be based on a downstream starting depth Y_2 and the other variables that define the GVF problem. Then the equation associated with the ODE for use in the Newton method can be defined as $F_i = Y_1 - Y_{ode1} = 0$. This equation will not be satisfied, e.g., equal zero until the solution of the ODE equals the depth Y_1, the value of which is used in the other equations. In other words, as is the case with an algebraic equation, $F_i = Y_1 - Y_{ode1}$ will not be zero, until the correct solution vector of unknowns is used in evaluating it.

If you have difficulties in comprehending how $F_i = Y_1 - Y_{ode1} = 0$, that comes from an ODE can be included with algebraic equations to form a system of simultaneous equations, consider a simple ODE (with an appropriate boundary, or initial condition) for which a closed-form solution can be obtained. This closed-form solution is an algebraic equation that can be written as a function of the unknown variables equal to zero, i.e., $F(x_1, x_2, \ldots. x_n) = 0$, in which the x's are the unknown variables. Since in general, only numerical solutions and not closed-form solutions to ODEs are available,, and since our ODE has a beginning depth that is known at one end of the channel, and the solution produces the depth at the other end of the channel, a convenient way of expressing a function that must equal zero is to subtract the numerical solution of the ODE from the depth that is being sought, as one of the unknown variables.

The implementation of the Newton method in solving combined algebraic equations and ODEs can be illustrated by taking a general example, such as Example Problem 4.8, in which the water is supplied from a reservoir with a known head H at the upstream end of the channel, and the flow is controlled by a gate at the downstream end. The equations that define this problem are

$$F_1 = H - Y_1 - (1 + K_e)\frac{Q^2}{2gA_1^2} = 0 \quad \text{(Energy applied between reservoir and channel)}$$

$$F_2 = Y_3 + \frac{Q^2}{2gA_3^2} - Y_d + \frac{Q^2}{2gA_d^2} = 0 \quad \text{(Energy across downstream gate)}$$

$$F_3 = Y_2 + \frac{Q^2}{2gA_2^2} - Y_3 - (1 + K_L)\frac{Q^2}{2gA_3^2} = 0 \quad \text{(Energy across transition upstream from gate)}$$

$$F_4 = Y_1 - Y_{ode1} = 0 \quad \text{(From solving an ODE problem from the beginning of the transition to the beginning of a channel)}$$

In these equations, the subscripts of the areas, A_i, correspond to the subscripts of the depths. Subscript d denotes immediately downstream from the gate; subscript 3 denotes immediately upstream from the gate and at the end of the transition; subscript 2 denotes the beginning of the transition; and subscript 1 denotes the beginning of the channel. Y_d is known to be equal to the height of the gate above the channel bottom times the contraction coefficient, or $Y_d = C_cY_G$, and therefore the unknowns are: the flow rate Q, the depth at the beginning of the channel Y_1, the depth at the beginning of the transition Y_2, and the depth at the end of the transition Y_3. The solution of the above four simultaneous equations provides the solution to these four unknowns.

The implementation of the Newton method in solving these equations involves the iterative equation:

$$\{\mathbf{Y}\}^{(m+1)} = \{\mathbf{Y}\}^{(m)} - \{\mathbf{z}\}$$

in which {z} is the unknown vector in the linear system of equations, [D]{z} = {F} in which [D] is the Jacobian matrix and {F} is the equation vector represented by the above four equations, e.g., its elements are obtained by evaluating these equations with the current values of the unknown vector $\{Y\}^{(m)}$. ($\{Y\}^T = [Q, Y_1, Y_2, Y_3]$). The Jacobian D consists of the following derivative matrix:

$$D = \begin{bmatrix} \frac{\partial F_1}{\partial Q} & \frac{\partial F_1}{\partial Y_1} & \frac{\partial F_1}{\partial Y_2} & \frac{\partial F_1}{\partial Y_3} \\ \frac{\partial F_2}{\partial Q} & \frac{\partial F_2}{\partial Y_1} & \frac{\partial F_2}{\partial Y_2} & \frac{\partial F_2}{\partial Y_3} \\ \frac{\partial F_3}{\partial Q} & \frac{\partial F_3}{\partial Y_1} & \frac{\partial F_3}{\partial Y_1} & \frac{\partial F_3}{\partial Y_3} \\ \frac{\partial F_4}{\partial Q} & \frac{\partial F_4}{\partial Y_1} & \frac{\partial F_4}{\partial Y_1} & \frac{\partial F_4}{\partial Y_3} \end{bmatrix}$$

It is not possible to algebraically determine the derivatives of the fourth equation F_4, but these can be evaluated numerically. For example,

$$\frac{\partial F_4}{\partial Q} \approx \frac{1}{\Delta Q}\left\{F_4(Q + \Delta Q, Y_1, Y_2, Y_3) - F_4(Q, Y_1, Y_2, Y_3)\right\}$$

Likewise, derivatives with respect to Y_1, Y_2, and Y_3 are determined by evaluating the equation twice, once with their values incremented and once without their values incremented, taking the difference between these two values of the equation and then dividing this difference by the increment. If x_i denotes the unknowns, then the general equation to evaluate any element (with subscript j, i.e., in column j and row i) in the Jacobian is

$$\frac{\partial F_i}{\partial x_j} = \frac{F_i(x_1, x_2, \ldots, x_j + \Delta x_j, x_{j+1}, \ldots, x_n) - F_i(x_1, x_2, \ldots, x_j, x_{j+1}, \ldots, x_n)}{\Delta x_j}$$

The evaluation of each element of the last row of the above Jacobian D entails solving the ODE that describes the GVF profile twice, once with the flow rate, or the depths incremented, and once without this increment.

Consider as another example, the problem involving an M_1-GVF profile upstream from a control due to a critical depth that exists at a distance x = L downstream from the channel's entrance. The depth at this control is the beginning value of the dependent variable Y_2, and the solution Y_{GVF} at the entrance where x = 0 is the solution produced upon solving the M_1. Then the function F_i might be defined by

$$F_i = Y_1 - Y_{GVF} = 0$$

in which Y_1 is the current value of the depth at the entrance of the channel. To carry this example further, assume that the control consists of a short smooth transition to a steep rectangular channel. Then at the beginning of the rectangular channel, the flow will be critical and the critical depth will be defined by $Y_c = ^3/(Q/b)^2/g$, and if the minor loss due to the transition is defined by a loss coefficient K_L that multiples the downstream velocity head, then the specific energy at the end of the upstream channel is given by $E_2 = (1.5 + K_L/2)Y_c$. At the channel entrance, the energy equation applies. Thus, the following three equations are available:

$$F_1 = Y_2 + \frac{Q^2}{2gA_2^2} - \left(1.5 + \frac{1}{2}K_L\right)\sqrt[3]{(Q/b)^2/g} = 0$$

$$F_2 = H - Y_1 - (1 + K_e)\frac{Q^2}{2gA_1^2} = 0$$

$$F_3 = Y_1 - Y_{GVF} = 0$$

The unknowns in these three equations are: Q, Y_1 (depth at channel entrance), and Y_2 (depth at the end of the upstream channel)

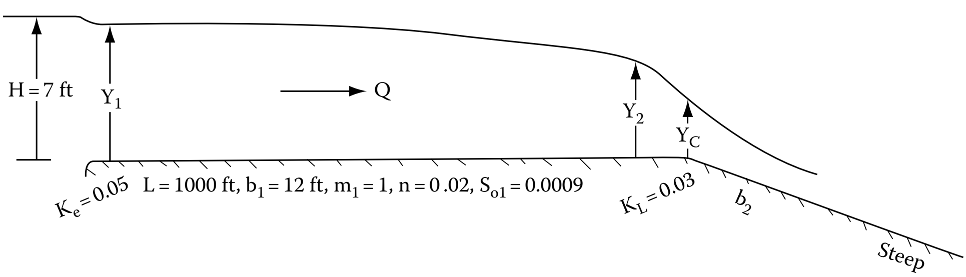
Programming a computer to solve the combination of algebraic and ODE equations that govern Example Problem 4.8 is not difficult. While only derivatives of the equations involving the ODEs are evaluated numerically, the other equations might also be evaluated numerically. Means for doing this are implemented in the computer programs that solve the following problems. With the FUNCTION `FUN(II)` designed to provide the evaluation of equation II (see following listing of `FUN`) this code consists of

```
1      DO 10 I=1,N
       F(I)=FUN(I)
       DO 10 J=1,N
       DX=.005*X(J)
       X(J)=X(J)+DX
       D(I,J)=(FUN(I)-F(I))/DX
10     X(J)=X(J)-DX
```

in which D hold the elements of the N by N Jacobian matrix and F is the equation vector.

4.7 FLOW INTO A MILD CHANNEL WITH A DOWNSTREAM CONTROL

As a relatively simple example of solving combined algebraic and ODEs simultaneously, consider the flow into a reservoir feed channel that has a GVF in it due to a downstream effect. Such a situation results if a break in grade to a steep channel occurs a relatively short distance L downstream from the reservoir, as shown in the sketch below. Let us consider the case where the channel from the reservoir to the break in grade is trapezoidal, and the downstream channel is rectangular with a different width b_2 than the upstream channel.



There are three variables that are to be solved: (1) the flow rate Q, (2) the depth at the upstream end of the channel Y_1, and (3) the depth at the downstream end of the upstream channel Y_2. One might also consider the critical depth Y_c at the beginning of the downstream rectangular channel an unknown, but since this channel is rectangular, the critical depth will be eliminated by using the explicit equations for the critical flow in rectangular channels. Thus, three equations are needed. Two of these are algebraic, the energy equation at the reservoir and the energy equation across the two channels, and the third is the ODE that describes the GVF in the channel. These equations are

$$F_1 = H - Y_1 - (1+K_e)\frac{Q^2}{2gA_1^2} = 0$$

$$F_2 = Y_2 + \frac{Q^2}{2gA_2^2} - \left(1.5 + \frac{K_L}{2}\right)\left\{\frac{(Q/b)^2}{g}\right\}^{\frac{1}{3}} = 0$$

$$F_3 = Y_1 - Y_{1,ode}(Y_2)$$

The notation used in the third equation has the following meanings: $Y_{1,ode}(Y_2)$ is the depth at the beginning of the channel obtained by solving the ODE for a GVF starting with the depth Y_2 at its downstream end. The depth thus obtained should match the depth Y_1 that occurs in the other equations. The second equation equates the specific energy at the end of the trapezoidal channel to the critical specific energy in the rectangular channel, plus the local loss here. The local loss equals $h_L = K_L V_c^2/(2g)$. Noting that $V_c^2/(2g) = Y_c/2$ and $Y_c = \{q^2/g\}^{1/3} = \{(Q/b\}^2/g\}^{1/3}$ and that $E_c = 1.5Y_c$, provides the last term in F_2.

The Newton method, as described above, can solve these three equations simultaneously. Program SOLGVF is designed to accomplish such solutions.

Program SOLGVF.FOR

```
C Solves problem of flow into a mild channel from a reservoir that
C has a steep rectangular channel at a position L downstream
C from the reservoir.
      REAL F(3),D(3,3),X(3),KL2,KE1,KL,KE
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,FM1,B2,H,G,G2,KL2,KE1,FL,TOL,FN,SO,CC,QN,Q2G,X
      EQUIVALENCE (Q,X(1)),(Y1,X(2)),(Y2,X(3))
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,FN,SO,B1,FM1,B2,H,L,g,KL,KE'
      READ(*,*) IOUT,TOL,ERR,FN,SO,B1,FM1,B2,H,FL,G,KL,KE
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      KL2=.5*KL+1.5
      KE1=1.+KE
      WRITE(*,*)' GIVE guess for: Q,Y1,Y2'
      READ(*,*) X
      NCT=0
1     DO 10 I=1,3
      F(I)=FUN(I)
      DO 10 J=1,3
      DX=.005*X(J)
      X(J)=X(J)+DX
```

```
      D(I,J)=(FUN(I)-F(I))/DX
10    X(J)=X(J)-DX
      FAC=D(3,1)/D(1,1)
      D(3,2)=D(3,2)-FAC*D(1,2)
      D(3,3)=D(3,3)-FAC*D(1,3)
      F(3)=F(3)-FAC*F(1)
      FAC=D(2,1)/D(1,1)
      D(2,2)=D(2,2)-FAC*D(1,2)
      D(2,3)=D(2,3)-FAC*D(1,3)
      F(2)=F(2)-FAC*F(1)
      FAC=D(3,2)/D(2,2)
      D(3,3)=D(3,3)-FAC*D(2,3)
      F(3)=F(3)-FAC*F(2)
      DIF1=F(3)/D(3,3)
      Y2=Y2-DIF1
      DIF=(F(2)-DIF1*D(2,3))/D(2,2)
      Y1=Y1-DIF
      SUM=ABS(DIF1)+ABS(DIF)
      DIF=(F(1)-D(1,2)*DIF-D(1,3)*DIF1)/D(1,1)
      SUM=SUM+ABS(DIF)
      Q=Q-DIF
      NCT=NCT+1
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 1
      WRITE(IOUT,100) X
100   FORMAT(' Q =',F10.2,' Y1 =',F10.2,' Y2 =',F10.2)
      END
      FUNCTION FUN(II)
      EXTERNAL DYX
      REAL X(3),W(1,13),KL2,KE1,Y(1),DY(1),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,FM1,B2,H,G,G2,KL2,KE1,FL,TOL,FN,SO,CC,QN,
     &Q2G,X
      GO TO (1,2,3),II
1     Q2G=X(1)*X(1)/G2
      FUN=X(3)+Q2G/((B1+FM1*X(3))*X(3))**2-KL2*((X(1)/B2)**2/G)
     &**.333333333
      RETURN
2     FUN=H-X(2)-KE1*Q2G/((B1+FM1*X(2))*X(2))**2
      RETURN
3     Y(1)=X(3)
      H1=-.05
      HMIN=.001
      XX=FL
      XZ=0.
      QN=(FN*X(1)/CC)**2
      Q2G=X(1)*X(1)/G
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      FUN=X(2)-Y(1)
      RETURN
      END
```

```
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1),KL2,KE1,X(3)
      COMMON/TRAS/B1,FM1,B2,H,G,G2,KL2,KE1,FL,TOL,FN,SO,CC,QN,
     &Q2G,X
      EQUIVALENCE (Q,X(1)),(Y1,X(2)),(Y2,X(3))
      P=B1+2.*SQRT(FM1*FM1+1.)*Y(1)
      A=(B1+FM1*Y(1))*Y(1)
      SF=QN*((P/A)**.666666667/A)**2
      T=B1+2.*FM1*Y(1)
      DY(1)=(SO-SF)/(1.-Q2G*T/A**3)
      RETURN
      END
```

SOLGVF.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
float b1,fm1,fm12,fm1s,b2,h,g,g2,kl2,ke1,fl,tol,hmin=.001,fn,so, cc,\
   fnq,q2g,x[3];
#include "odesol.h"
void slope(float x,float *y,float *dyx){float a,sf;
   a=(b1+fm1*y[0])*y[0];sf=fnq*pow(pow((b1+fm1s*y[0])/a,.6666667)/a,2.);
   dyx[0]=(so-sf)/(1.-q2g*(b1+fm12*y[0])/(a*a*a));} // end slope
float fun(int ii){float h1,y[1]; h1=-.05;
   q2g=x[0]*x[0]/g2;
   if(ii==1) return (x[2]+q2g/pow((b1+fm1*x[2])*x[2],2.)-\
      kl2*pow(pow(x[0]/b2,2.)/g,.3333333));
   else if(ii==2) return(h-x[1]-ke1*q2g/pow((b1+fm1*x[1])*x[1],2.));
   else { y[0]=1.005*x[2]; fnq=pow(fn*x[0]/cc,2.); q2g=x[0]*x[0]/g;
      odesolc(y,fl,0.,tol,h1,hmin); return (x[1]-y[0]);} } // End fun
void main(void){int nct,i,j; float f[3],dx,fac,dif1,dif,err,sum,**d;
   d=(float**)malloc(3*sizeof(float*));
   for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
   printf("Give: TOL,ERR,n,So,b1,m1,b2,H,L,g,KL,Ke\n");
   scanf("%f %f %f %f %f %f %f %f %f %f %f %f",\
   &tol,&err,&fn,&so,&b1,&fm1,&b2,&h,&fl,&g,&kl2,&ke1);
   if(g>20.) cc=1.486; else cc=1.; g2=2.*g;kl2=.5*kl2+1.5;ke1+=1.;
   fm12=2.*fm1;fm1s=sqrt(fm1*fm1+1.);
   printf("Give guess for: Q,Y1,Y2\n");
   scanf("%f %f %f",&x[0],&x[1],&x[2]);nct=0;
   do{
      for(i=1;i<4;i++){f[i-1]=fun(i); for(j=0;j<3;j++){
           dx=.005*x[j]; x[j]+=dx;d[i-1][j]=(fun(i)-f[i-1])/dx;
           x[j]-=dx;}}
      fac=d[2][0]/d[0][0];d[2][1]-=fac*d[0][1];d[2][2]-=fac*d[0][2];
      f[2]-=fac*f[0]; fac=d[1][0]/d[0][0];d[1][1]-=fac*d[0][1];
      d[1][2]-=fac*d[0][2]; f[1]-=fac*f[0];
      fac=d[2][1]/d[1][1];d[2][2]-=fac*d[1][2];f[2]-=fac*f[1];
      dif1=f[2]/d[2][2]; x[2]-=dif1;dif=(f[1]-dif1*d[1][2])/d[1][1];
```

```
    x[1]-=dif;sum=fabs(dif1)+fabs(dif);
    dif=(f[0]-d[0][1]*dif-d[0][2]*dif1)/d[0][0];sum+=fabs(dif);
    x[0]-=dif;
    printf(" NCT=%d SUM=%f ",++nct,sum);
    for(i=0;i<3;i++)printf(" %f",x[i]);printf("\n");
  }while((nct<30) && (sum>err));
  printf("Q = %10.2f Y1 = %10.2f Y2 = %10.2f\n",x[0],x[1],x[2]);}
```

This program contains the following: (1) a main program that defines the problem and implements the Newton solution, (2) a function subprogram, FUN, that defines any of the three equations (which you should write out) when called upon to do this, and when called on to provide the value of the equation that involves the GVF profile, it calls on the ODE solver appropriately, and (3) a subroutine that defines the derivative dY/dx. To obtain values for the equations that are stored in the FORTRAN array F(3), the main program calls the function subprogram FUN with an argument I = 1, 2, and 3 for the different equation numbers. When I = 1 or I = 2, then FUN evaluates the equation with a single FORTRAN line. However, when I = 3, the ODE solver is called upon to supply the solution to Y_{GVF} at the entrance of the channel. The difference between the current value Y_1, which is being adjusted by the Newton method, and this value defines the third equation and `FUN=X1(2)-Y(1)`. Note that I becomes II, the first argument in `FUNCTION FUN` and the different equations are selected with the computed `GO TO(1,2,3),II` statement. (The FORTRAN variables are: `X1(1)=Q, X1(2) =` Y_1, and `X1(3)=`Y_2, as can be seen by the equivalence statement.) After the main program fills the elements for the equation vector, `F(I)`, and the Jacobian matrix `D(I,J)`, it solves the resulting linear system of equations, adjusts the unknown vector, `X(I)` which consists of: Q, Y_1, and Y_2, and repeats another Newton iteration if the convergence error ERR is not satisfied. The error parameter TOL applies for the ODE solver. Since part of this solution involves the numerical solution of the GVF profile, the error for the Newton solution ERR should not be too small. ERR = 0.1 is probably a good value to use. The subroutine DYX supplies dY/dx to ODESOL or DVERK whenever requested.

A more refined solution might be obtained by eliminating the loss coefficient, and in its place solving the GVF profile through the transition using the term that involves $\partial A/\partial x$ for the nonprismatic channel effect. To implement this approach, a small amount of additional logic is needed in the subroutine DYX that properly evaluates the change in area with respect to x while in the transition and adds this term in defining dY/dx, and sets $\partial A/\partial x = 0$ when upstream from the transition. The first equation, which now consists of the energy equation between the beginning of the transition and its end, is replaced by just the critical flow equation applied at the downstream end of the transition. The main program reads in the data for the problem that is to be solved, and implements the Newton method in obtaining the solution. To obtain the equation vector and the Jacobian matrix, it calls on the function subroutine FUN to evaluate the equation number corresponding to its argument. When the third equation is to be evaluated, then the function subprogram FUN calls on the ODE solver ODESOL, which in turn call on the subroutine DYX to evaluate the derivative dY/dx = (So − Sf)/ $(1 - F_r^2)$. After obtaining the equation vector and the Jacobian matrix, this main program uses the Gaussian elimination method to solve this linear system of equations. However, these statements could be replaced by a call to a subroutine such as SOLVEQ that solves linear systems of equations. You should study this listing over carefully to fully understand how the solution is accomplished.

EXAMPLE PROBLEM 4.9

Using program SOLGVF obtain a series of solutions for the flow rate into an upstream trapezoidal channel with $b_1 = 5$ m, $m_1 = 1.2$, n = 0.015, and $S_{o1} = 0.00075$. This channel is supplied by a reservoir whose water surface is 3.5 m above the channel bottom. The entrance loss coefficient is $K_e = 0.05$. At a distance 300 m downstream from the channel's beginning, there is a transition to a steep rectangular channel with a bottom width b_2. The loss coefficient for the transition is

$K_L = 0.025$. Solve the problem with the bottom width of this downstream channel varying from $b_2 = 3$ m to $b_2 = 8$ m.

Solution

If the flow were uniform, then the solution of the energy equation at the entrance and Manning's equation give: $Q_o = 73.743\,m^3/s$, $Y_o = 3.10$ m, $E_o = 3.48$ m, and $M_o = 56.41\,m^3$. Solving the critical flow equation simultaneously with the energy equation at the entrance gives: $Q = 86.26\,m^3/s$, $Y_c = 2.54$ m, $E_c = 3.45$ m, and $M_c = 59.79\,m^3$. This latter flow rate represents the maximum that can be obtained for the given reservoir head, and would occur only if the length of the upstream channel approached zero, and the downstream rectangular channel were wide enough. The solutions to the other cases for varying b_2 are given below. The program failed to produce a solution when the bottom width of the downstream channel was given as $b_2 = 8.0$ m, since the downstream Froude number approaches unity. The following should be observed: (1) With the smaller bottom widths, the flow is considerably smaller than if the flow in the trapezoidal channel were uniform. It is not until the bottom width of the downstream channel gets wider than 6.5 m, that the flow rate exceeds this amount. (2) The GVF profile in the upstream trapezoidal channel is an M_1 for the bottom width of the downstream channel less than 6.5 m. This can be observed by the fact

	Input to SOLGVF													Solution		
No.	IOUT	TOL	ERR	FN	SO	B1	FM1	B2	H	L	g	KL	KE	Q	Y1	Y2
1	6	1.e−6	0.0005	0.015	0.00075	5	1.2	3.0	3.5	300	9.81	0.025	0.05	35.81	3.43	3.63
2	6	1.e−6	0.0005	0.015	0.00075	5	1.2	3.5	3.5	300	9.81	0.025	0.05	41.54	3.40	3.60
3	6	1.e−6	0.0005	0.015	0.00075	5	1.2	4.0	3.5	300	9.81	0.025	0.05	47.15	3.37	3.56
4	6	1.e−6	0.0005	0.015	0.00075	5	1.2	4.5	3.5	300	9.81	0.025	0.05	52.61	3.34	3.50
5	6	1.e−6	0.0005	0.015	0.00075	5	1.2	5.0	3.5	300	9.81	0.025	0.05	57.88	3.29	3.44
6	6	1.e−6	0.0005	0.015	0.00075	5	1.2	5.5	3.5	300	9.81	0.025	0.05	62.91	3.25	3.37
7	6	1.e−6	0.0005	0.015	0.00075	5	1.2	6.0	3.5	300	9.81	0.025	0.05	67.62	3.19	3.28
8	6	1.e−6	0.0005	0.015	0.00075	5	1.2	6.5	3.5	300	9.81	0.025	0.05	71.92	3.13	3.17
9	6	1.e−6	0.0005	0.015	0.00075	5	1.2	7.0	3.5	300	9.81	0.025	0.05	75.68	3.07	3.03
10	6	1.e−6	0.0005	0.015	0.00075	5	1.2	7.5	3.5	300	9.81	0.025	0.05	78.66	3.00	2.82
11	6	1.e−6	0.0005	0.015	0.00075	5	1.2	7.8	3.5	300	9.81	0.025	0.05	79.93	2.97	2.61

that Y_2 is larger than Y_1. Thus, for a width less than the 7.0 entry in the solution table, the change to the steep downstream rectangular channel reduces the flow from the reservoir. (3) As the widths of the downstream channel increase, the depths throughout the channel decrease.

EXAMPLE PROBLEM 4.10

Resolve Example Problem 4.8 by writing a program that solves the ODE and the algebraic equations simultaneously.

Solution

For this problem, one could write three algebraic equations plus one ODE that describes the GVF from the gate up to the reservoir. The three algebraic equation could consist of (1) energy from the reservoir to the beginning of the channel, (2) energy across the gate, and (3) energy across the transition upstream from the gate. However, (2) and (3) can be combined into one equation that equates the specific energy immediately upstream from the transition to the specific energy downstream from the gate. Thereafter, the energy across the transition (or across the gate) can be solved to get the depth immediately upstream from the gate. The three simultaneous equations needed to solve the three unknowns, Q, Y_1, and Y_2 are

$$F_1 = H - Y_1 - (1 + K_e)Q^2/(2gA_1^2) = 0$$

$$F_2 = Y_2 + Q^2/(2gA_2^2) - Y_d - (1 + K_L)Q^2/(2gA_d^2) = 0$$

$$F_3 = Y_1 - Y_{1,\text{ode}}(Y_2) = 0 \quad \text{ODE is } dY/dx = (S_o - S_f)/(1 - F_r^2)$$

in which Y_1 and Y_2 are the depths at the beginning and the end of the trapezoidal channel, respectively, and Y_d is the depth downstream of the gate and is given by $Y_d = C_cY_G$. An alternative is to replace F_2 with

$$F_2 = Y_2 + Q^2/(2gA_2^2) - Y_u - (1 + K_{L1})Q^2/(2gA_u^2) = 0 \quad \text{and}$$

$$F_3 = Y_u + Q^2/(2gA_u^2) - Y_d - (1 + K_{L2})Q^2/(2gA_d^2) = 0$$

and the previous F_3 becomes F_4, in which now K_{L1} is the loss coefficient for the transition, and K_{L2} is the loss coefficient for the gate, whereas earlier F_2 K_L was the combined loss coefficient for the transition and the gate. A third alternative is to use the above F_3 that equates the specific energy immediately upstream of the gate E_u to that downstream of the gate E_d, and to solve the ODE starting at the end of the transition (immediately upstream of the gate) and to include the nonprismatic term in the ODE while solving the GVF through the transition.

The program SOLGATE solves this problem using the first of these approaches. In this program, the array X, with dimensions of 3, contains the three unknowns with X(1) = Q, X(2) = Y_1, and X(3) = Y_2. In the program, the subroutine FUN defines the three equations, i.e., supplies values to F(1) = F_1, F(2) = F_2, and F(3) = F_3, and the Main program implements the Newton method in solving the equations by numerically evaluating the nine elements of the Jacobian matrix. It calls on the ODE solver RUKUST, whose use is described in Appendix C, to solve the GVF in the trapezoidal channel. In order to find the depth Y_u in the rectangular channel upstream of the gate, but at the end of the transition, the function subprogram YGATE uses the Newton method to solve the specific energy equation across the gate after Q is solved. Likewise, to provide the normal depth Y_o corresponding to this flow rate, the subprogram YNORM uses the Newton method to solve Manning's equation. The subroutine OPEN (qopen in the C-program) solves the energy equation at the entrance and Manning's equation simultaneously to provide Y_o and Q_o, as if there were no gate in the channel and it were long so that a uniform flow would occur in the upstream trapezoidal channel.

The input to solve the problem with the 10 given gate settings used in Example Problem 4.8 is

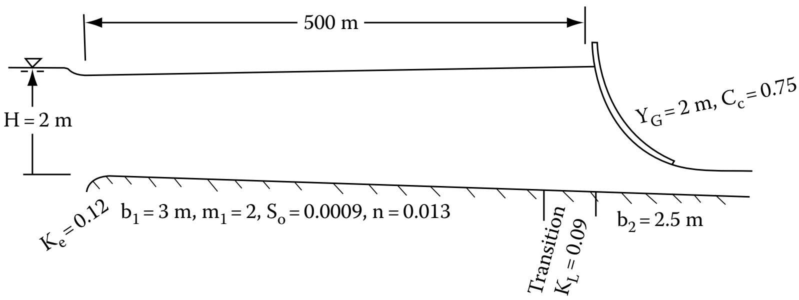
```
3 2 .013 .0009 2 .12 .09 9.81 500 2.5 .0000001 .0005 .75 2 -.2 10
15 1.9 2.2
```

The solution provided is

```
H = 2.0, Ke = .120, b1 = 3.0, m1 = 2.00, n = .0130,
So = .000900, b2 = 2.5, KL = .090, L = 500.0
```

No-gate Uniform Flow: Y_o = 1.688, Q_o = 25.17

Y_G	Y_d	Q	Y_1	Y_2	Y_u	Y_o	Y_c
2.00	1.50	14.76	1.929	2.325	1.830	1.299	1.526
1.80	1.35	14.43	1.933	2.332	1.890	1.284	1.503
1.60	1.20	13.79	1.939	2.344	1.977	1.255	1.458
1.40	1.05	12.87	1.948	2.359	2.069	1.212	1.392
1.20	0.90	11.68	1.957	2.377	2.158	1.154	1.306
1.00	0.75	10.25	1.968	2.395	2.239	1.080	1.197
0.80	0.60	8.59	1.978	2.412	2.310	0.986	1.064
0.60	0.45	6.72	1.987	2.427	2.368	0.868	0.903
0.40	0.30	4.66	1.994	2.439	2.412	0.714	0.707
0.20	0.15	2.41	1.998	2.447	2.440	0.498	0.456



Listing of program SOLGATE.FOR

```
      PARAMETER (N=3)
      INTEGER*2 ITYP(N)
      REAL F(N),F1(N),D(N,N),KL1,KE1
      COMMON /TRAS/B1,FM1,B2,H,G2,KL1,KE1,FL,TOL,FN,SO,CU,QN,Q2G,
     &Yd,Ad,FMS,FM2,X(3),DXS
      WRITE(*,*)' Give:b1,m1,n,So,H,Ke,KL,g,L,b2,TOL,ERR,Cc,Yg1,
     &dYg,N'
      READ(*,*) B1,FM1,FN,SO,H,KE1,KL1,G,FL,B2,TOL,ERR,CC,
     &YG1,DYG,NGATE
      CU=1.486
      DXS=1.
      FM2=2.*FM1
      FMS=2.*SQRT(FM1*FM1+1.)
      IF(G.LT. 30.) CU=1.
      G2=2.*G
      KL1=1.+KL1
      KE1=1.+KE1
      WRITE(*,*)' Give guess for: Q,Y1 & Y2'
      READ(*,*) X
      Yo=.98*X(2)
      CALL OPEN(Yo,Qo)
      WRITE(3,110) H,KE1-1.,B1,FM1,FN,SO,B2,KL1-1.,FL,Yo,Qo
110   FORMAT(' H =',F7.1,', Ke =',F6.3,', b1 =',F8.1,', m1 =',
     &F7.2,', n =',F8.4,',',/,' So =',F9.6,', b2 =',F8.1,',
     &KL =',F6.3,', L =',F8.1,/,' No gate-Uniform Flow: Yo =',
     &F8.3,', Qo =',F8.2,/1X,61('-'),/,4X,'YG',4X,'Yd',7X,'Q',
     &6X,'Y1',6X,'Y2',6X,'Yu',6X,'Yo',6X,'Yc',/,1X,61('-'))
      DO 50 KK=1,NGATE
      YG=YG1+DYG*FLOAT(KK-1)
      Yd=CC*YG
      Ad=(B2*Yd)**2
      NCT=0
1     SUM=0.
      CALL FUN(F1)
      DO 10 J=1,N
      XX=X(J)
      X(J)=1.005*X(J)
      CALL FUN(F)
      DO 5 I=1,N
5     D(I,J)=(F(I)-F1(I))/(X(J)-XX)
10    X(J)=XX
      CALL SOLVEQ(N,1,N,D,F1,1,DET,ITYP)
      DO 15 I=1,N
      X(I)=X(I)-F1(I)
```

```
15     SUM=SUM+ABS(F1(I))
       NCT=NCT+1
       IF(SUM.GT.ERR .AND. NCT.LT.20) GO TO 1
       IF(NCT.EQ.20) WRITE(*,*)' Did not Converge',SUM
50     WRITE(3,100) YG,CC*YG,X,YGATE(X(1)),YNORM(X(1)),
      &((X(1)/B2)**2/G)**.33333333
100    FORMAT(2F7.2,F8.2,5F8.3)
       END
       SUBROUTINE FUN(F)
       REAL F(3),KL1,KE1,Y(1),YW(1)
       COMMON /TRAS/B1,FM1,B2,H,G2,KL1,KE1,FL,TOL,FN,SO,CU,
      &QN,Q2G,Yd,Ad,FMS,FM2,X(3),DXS
       QN=(FN*X(1)/CU)**2
       Q2G=X(1)**2/G2
       F(1)=H-X(2)-KE1*Q2G/((B1+FM1*X(2))*X(2))**2
       F(2)=X(3)+Q2G/((B1+FM1*X(3))*X(3))**2-Yd-KL1*Q2G/Ad
       Y(1)=X(3)
       CALL RUKUST(1,DXS,FL,0.,TOL,Y,YW)
       F(3)=X(2)-Y(1)
       RETURN
       END
       SUBROUTINE SLOPE(XX,Y,DY)
       REAL Y(1),DY(1),KL1,KE1
       COMMON /TRAS/B1,FM1,B2,H,G2,KL1,KE1,FL,TOL,FN,SO,CU,
      &QN,Q2G,Yd,Ad,FMS,FM2,X(3),DXS
       AREA=(B1+FM1*Y(1))*Y(1)
       SF=QN*(B1+FMS*Y(1))**1.33333333/AREA**3.3333333
       DY(1)=(SO-SF)/(1.-.5*Q2G*(B1+FM2*Y(1))/AREA**3)
       RETURN
       END
       FUNCTION YNORM(Q)
       COMMON /TRAS/B1,FM1,B2,H,G2,FKL1,FKE1,FL,TOL,FN,SO,CU,
      &QN,Q2G,Yd,Ad,FMS,FM2,X(3),DXS
       QQ=CU*SQRT(SO)/FN
       Y=X(2)
       M=0
1      F=Q-QQ*((B1+FM1*Y)*Y)**1.6666667/(B1+FMS*Y)**.666667
       Y1=1.005*Y
       DIF=(Y1-Y)*F/(Q-QQ*((B1+FM1*Y1)*Y1)**1.6666667/
      &(B1+FMS*Y1)**.666667-F)
       Y=Y-DIF
       M=M+1
       IF(ABS(DIF).GT. .000001 .AND. M.LT.20) GO TO 1
       IF(M.EQ.20) WRITE(*,*) ' YNORM failed to converge'
       YNORM=Y
       RETURN
       END
       FUNCTION YGATE(Q)
       COMMON /TRAS/B1,FM1,B2,H,G2,FKL1,FKE1,FL,TOL,FN,SO,CU,
      &QN,Q2G,Yd,Ad,FMS,FM2,X(3),DXS
       M=0
       Y=.9*X(3)
       QQ=Q**2/G2
       ED=FKL1*QQ/Ad+Yd
1      F=Y+QQ/(B2*Y)**2-ED
       Y1=1.005*Y
       DIF=(Y1-Y)*F/(Y1+QQ/(B2*Y1)**2-ED-F)
```

```
      Y=Y-DIF
      M=M+1
      IF(ABS(DIF).GT. .000001 .AND. M.LT.20) GO TO 1
      IF(M.EQ.20) WRITE(*,*) ' YGATE failed to converge'
      YGATE=Y
      RETURN
      END
      SUBROUTINE OPEN(Yo,Qo)
      COMMON /TRAS/B1,FM1,B2,H,G2,FKL1,FKE1,FL,TOL,FN,SO,CU,
     &QN,Q2G,Yd,Ad,FMS,FM2,X(3),DXS
      M=0
      QQ=Cu*SQRT(SO)/FN
 1    AREA=(B1+FM1*Yo)*Yo
      Qo=QQ*AREA**1.66666667/(B1+FMS*Yo)**.66666667
      F=H-Yo-FKE1*(Qo/AREA)**2/G2
      Y1=1.005*Yo
      A=(B1+FM1*Y1)*Y1
      DIF=(Y1-Yo)*F/(H-Y1-FKE1*(QQ*A**1.6666667/(B1+FMS*Y1)
     &**.6666667/A)**2/G2-F)
      Yo=Yo-DIF
      M=M+1
      IF(ABS(DIF).GT. .000001 .AND. M.LT.20) GO TO 1
      IF(M.EQ.20) WRITE(*,*) ' OPEN failed to converge'
      Qo=QQ*AREA**1.66666667/(B1+FMS*Yo)**.66666667
      RETURN
      END
```

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float b1,m1,b2,h,g2,kl1,ke1,l,tol,n,so,cu,qn,q2g,yd,ad,fms,\
   fm2,dxs[1],x[3];
extern void solveq(int n,float **a,float *b,int itype,float *dd,\
   int *indx);
extern void rukust(int neq,float *dxs,float xbeg,float xend,\
   float error,float *y,float *ytt);
void fun(float *f){ float y[1],yw[1];
   qn=pow(n*x[0]/cu,2.); q2g=x[0]*x[0]/g2;
   f[0]=h-x[1]-ke1*q2g/pow((b1+m1*x[1])*x[1],2.);
   f[1]=x[2]+q2g/pow((b1+m1*x[2])*x[2],2.)-yd-kl1*q2g/ad;
   y[0]=x[2];rukust(1,dxs,l,0.,tol,y,yw); f[2]=x[1]-y[0]; } //end fun
void slope(float xx,float *y, float *dy){ float area,sf;
   area=(b1+m1*y[0])*y[0];
   sf=qn*pow(b1+fms*y[0],1.33333333)/pow(area,3.3333333);
   dy[0]=(so-sf)/(1.-.5*q2g*(b1+fm2*y[0])/pow(area,3.)); }// end slope
float ynorm(float q) {float qq,y,f,y1,dif; int m;
   qq=cu*sqrt(so)/n; y=x[1]; m=0;
   do {f=q-qq*pow((b1+m1*y)*y,1.66666667)/pow(b1+fms*y,.66666667);
      y1=1.005*y;
      dif=(y1-y)*f/(q-qq*pow((b1+m1*y1)*y1,1.6666667)/\
         pow(b1+fms*y1,.66666667)-f);
      y-=dif;
   }while ((fabs(dif)>.000001) && (++m<20));
   if(m==20) printf("YNORM failed to converge\n");
```

```
   return y;} // end ynorm
float ygate(float q){float y,qq,ed,f,y1,dif; int m;
   m=0; y=.9*x[2]; qq=q*q/g2; ed=kl1*qq/ad+yd;
   do {f=y+qq/pow(b2*y,2.)-ed; y1=1.005*y;
      dif=(y1-y)*f/(y1+qq/pow(b2*y1,2.)-ed-f);
      y-=dif;
   } while ((fabs(dif)>.000001) && (++m<20));
   if(m==20) printf("YGATE failed to converge\n");
   return y;} // end ygate
void qopen(float *yo, float *qo){
   float qq,area,f,y1,a,dif,yyo,qqo; int m;
   m=0; yyo=*yo; qq=cu*sqrt(so)/n;
   do {area=(b1+m1*yyo)*yyo;
      qqo=qq*pow(area,1.6666667)/pow(b1+fms*yyo,.6666667);
      f=h-yyo-ke1*pow(qqo/area,2.)/g2; y1=1.005*yyo;a=(b1+m1*y1)*y1;
      dif=(y1-yyo)*f/(h-y1-ke1*pow(qq*pow(a,1.6666667)/\
         pow(b1+fms*y1,.6666667)/a,2.)/g2-f);
      yyo-=dif;
   }while ((fabs(dif)>.000001) && (++m<20));
   if(m==20) printf("OPEN failed to converge\n");
   qqo=qq*pow(area,1.6666667)/pow(b1+fms*yyo,.66666667);
   *yo=yyo;*qo=qqo; } // end qopen
void main(void){
   float yo[1],qo[1],*det,sum,xx,err,cc,yg1,dyg,g,yg, f1[3],f[3],**d;
   int nn,kk,nct,j,i,indx[3];FILE *fil;
   d=(float**)malloc(3*sizeof(float*));
   for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
   dxs[0]=-1.; cu=1.486;
   printf("Give: b1,m1,n,So,H,Ke,KL,g,L,b2,TOL,ERR,Cc,Yg1,dYg,nn\n");
   scanf("%f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %d",\
     &b1,&m1,&n,&so,&h,&ke1,&kl1,&g,&l,&b2,&tol,&err,&cc,&yg1,&dyg,&nn);
   fm2=2.*m1; fms=2.*sqrt(m1*m1+1.); if(g<30.) cu=1;
   g2=2.*g; kl1+=1.;ke1+=1.;
   printf("Give guess for Q,Y1 & Y2\n");
   scanf("%f %f %f",&x[0],&x[1],&x[2]);
   yo[0]=.98*x[1]; qopen(yo,qo);
   if((fil=fopen("SOLGATE.OUT","w"))==NULL){
      printf("Can not open SOLGATE.OUT\n");exit(0);}
   fprintf(fil, H=%7.1f, Ke=%6.3f, b1=%8.1f, m1=%7.2f, n=%8.4f, n\
      So=%9.6f, b2=%8.1f, KL=%6.3f, L=%8.0f\n",\
      h,ke1-1.,b1,m1,n,so,b2,kl1-1.,l);
   fprintf(fil," No gate-Uniform Flow:Yo=%8.3f, Qo=%8.2f\n",yo[0],qo[0]);
   for(i=0;i<61;i++)fprintf(fil,"-"); fprintf(fil,"\n");
   fprintf(fil,"     YG     Yd     Q     Y1     Y2     Yu\
      Yo     Yc\n");
   for(i=0;i<61;i++)fprintf(fil,"-"); fprintf(fil,"\n");
   for(kk=0;kk<nn;kk++){
      yg=yg1+dyg*(float)kk; yd=cc*yg;ad=pow(b2*yd,2.); nct=0;
      do {sum=0.; fun(f1);
         for(j=0;j<3;j++){xx=x[j];x[j]*=1.005; fun(f);
            for(i=0;i<3;i++) d[i][j]=(f[i]-f1[i])/(x[j]-xx); x[j]=xx;}
```

```
    solveq(3,d,f1,1,det,indx);
    for(i=0;i<3;i++){x[i]-=f1[i];sum+=fabs(f1[i]);}
 } while((sum>err) && (++nct<20));
 if(nct==20)printf("Did not converge %f\n",sum);
 fprintf(fil,"%7.2f %6.2f %7.2f %7.3f %7.3f %7.3f %7.3f %7.3f\n",\
    yg,yd,x[0],x[1],x[2],ygate(x[0]),ynorm(x[0]),
    pow(pow(x[0]/b2,2.)/g,.333333333));} fclose(fil);}
```

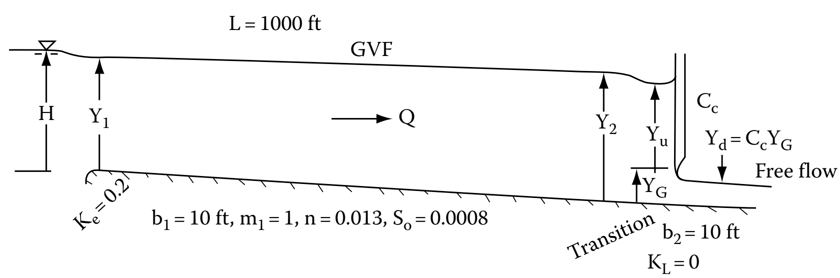
4.8 DIFFERENT MODES OF GATE OPERATION

The operation of gates can be classified into underflow or overflow, free flowing downstream or submerged downstream. The types of gates we have used to illustrate the principles of open channel flow are, underflow gates. The flow under a gate is also often referred to as orifice flow, since the equation describing the discharge past the gate is of the form of an orifice equation that gives the velocity or discharge, proportional to the square root of the head. Overflow gates control the depth and the flow rate by having a plate, etc., rise from the channel bottom. For such overflow gates, the flow rate (discharge) is proportional to the head raised to the 1.5 power. The flow past a gate is classified as either free flowing or submerged depending upon whether the downstream effects the flow past the gate, or not. For example, if an underflow gate is submerged, then the depth immediately downstream from the gate is above the tip of the gate, whereas if a free flow occurs, the depth immediately downstream from the gate is below the tip of the gate. Submerged gate flows will be dealt with in Chapter 5.

Often, the structure that holds the gate in place results in a constriction in the channel size, and therefore even though the tip of the gate is above the water depth, the gate site will affect the flow. Such a flow can be classified as a nonorifice flow. The effect is similar to that of a flume placed in the channel. Flumes and weirs are also dealt with in Chapter 5. Because the depth "contracts" below the tip of the gate if free flow occurs to $Y_d = C_cY_G$, it is possible to have gate settings above the normal depth associated with the flow rate that is occurring in a channel. As the settings of gates are changed, an unsteady flow is initiated in which the volume of water stored in the channel upstream of the gate is either increased or decreased, until another steady-state condition is eventually established. Changes in gate setting may result in the flow changing from free to submerged, or vice versus, or from having the gate's tip above to below the water surface, e.g., from nonorifice to orifice flow, or vice versa, etc. If a vertical gate is lowered into a channel flow, the results can be quite different, than if it is gradually raised above the water surface. The various possibilities can only be adequately dealt with by solving the unsteady flow equations, which are the subject of Chapters 6 and 7.

EXAMPLE PROBLEM 4.11

Obtain a series of steady-state solutions to the flow into a channel from a reservoir past a gate in which the gate is raised until it rises above the water surface. For these solutions, hold the upstream reservoir head constant with H = 5 ft (and an entrance loss coefficient $K_e = 0.2$). The gate is 1000 ft downstream from the reservoir. Upstream of the gate, the channel is trapezoidal with $b_1 = 10$ ft, $m_1 = 1$, $n = 0.013$, and $S_o = 0.0008$. The gate is vertical and is $b_2 = 10$ ft wide, and has an assumed constant contraction coefficient $C_c = 0.6$.



Solution

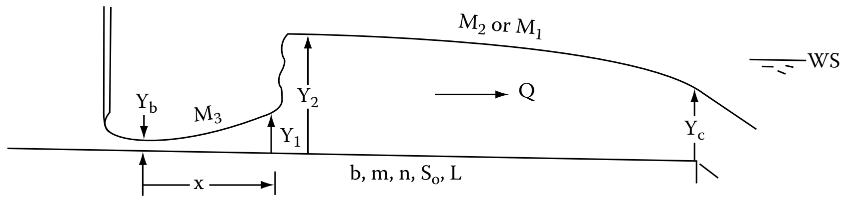
The program developed in the previous problem can be used to solve this problem. The last two columns in the solution table below, that provides the volume and the change in volume of water in the channel, between the reservoir and the gate, have been added to provide an idea about what volumes are involved during the unsteady change from one steady state to the new steady-state condition. Also, the Froude numbers immediately upstream and downstream of the gate have been added. The input is: 10 1 .013 .0008 5 .2 0 32.2 1000 10 .0000001 .0005 .6 3 .2 16, and 300 4.7 5.3 as a second line. The table provides the solution, and the graph below the table plots the flow rate on the right ordinate, and several of the depths in the table on the left ordinate are against the gate height as the abscissa. Notice that when the gate's tip is between 5.6 and 5.8 ft above the channel bottom, the solution of the equations shows $Y_u = Y_d$. Both are critical depths. When this occurs, the flow rate is between 356.41 and 356.31 cfs, which is slightly less than what would occur under uniform conditions if the trapezoidal channel were very long, i.e., $Q_o = 384.29$ (with $Y_o = 4.250$ ft). In other words, the tip of the gate is actually 5.80 − 3.485 = 1.315 ft above the water surface immediately upstream of the gate. Of course, this is nonsense; but illustrates that judgement must be used in solving equations. Actually, the gate will rise above the water surface between gate settings 4.2 and 4.4 ft provided the gate is raised extremely slowly so transient effects are insignificant. However, if the gate is raised rapidly, it could be much higher before clearing the water surface. Notice the last solutions in the table with $Y_G = 5.8$ and 6.0 ft produce identical depths of $Y_u = Y_d = 3.480$ and 3.600 ft, e.g., no longer are alternate depths being solved across the gate.

$H = 5.0$, $K_e = 0.200$, $b_1 = 10.0$, $m_1 = 1.00$, $n = 0.0130$, $S_o = 0.000800$, $b_2 = 10.0$, $K_L = 0.000$, $L = 1000$.
No gate–Uniform Flow: $Y_o = 4.250$, $Q_o = 384.29$.

Y_G	Y_d	F_{rd}	Q	Y_1	Y_2	Y_u	F_{ru}	Y_o	Y_c	Volume	Del Vol.
3.00	1.80	2.02	276.58	4.702	5.284	4.989	0.44	3.534	2.875	67755	0
3.20	1.92	1.91	288.57	4.669	5.223	4.886	0.47	3.620	2.957	67297	−457
3.40	2.04	1.81	299.58	4.637	5.161	4.778	0.51	3.697	3.032	66842	−454
3.60	2.16	1.72	309.59	4.605	5.099	4.667	0.54	3.766	3.099	66395	−445
3.80	2.28	1.63	318.60	4.574	5.037	4.551	0.58	3.827	3.159	65964	−430
4.00	2.40	1.55	326.59	4.545	4.977	4.432	0.62	3.881	3.212	65555	−408
4.20	2.52	1.47	333.58	4.518	4.920	4.311	0.66	3.927	3.257	65174	−380
4.40	2.64	1.40	339.58	4.493	4.867	4.188	0.70	3.966	3.296	64828	−346
4.60	2.76	1.32	344.61	4.472	4.819	4.065	0.74	3.999	3.329	64521	−306
4.80	2.88	1.26	348.71	4.453	4.778	3.940	0.79	4.026	3.355	64259	−261
5.00	3.00	1.19	351.91	4.438	4.743	3.816	0.83	4.046	3.375	64045	−213
5.20	3.12	1.13	354.23	4.427	4.717	3.693	0.88	4.061	3.390	63882	−162
5.40	3.24	1.07	355.73	4.419	4.700	3.570	0.93	4.071	3.400	63771	−110
5.60	3.36	1.02	356.41	4.416	4.692	3.449	0.98	4.075	3.404	63713	−57
5.80	3.48	.97	356.31	4.416	4.693	3.480	0.97	4.074	3.404	63722	10
6.00	3.60	.92	355.44	4.421	4.704	3.600	0.92	4.069	3.398	63793	70

4.9 HYDRAULIC JUMP DOWNSTREAM FROM A GATE IN A FINITE LENGTH CHANNEL

Now, let us consider the problem of finding the position of a hydraulic jump downstream from a gate where the channel terminates at some relatively short distance downstream therefrom either as a free overfall or by discharging into a reservoir with a known depth. With the length of the channel not too long, there will be a GVF downstream from the hydraulic jump, as well as the M_3 GVF upstream therefrom. Thus, the downstream depth Y_2 is unknown. If a free overfall occurs at the end of the channel, this will be an M_2 GVF. When the water level in the reservoir rises above the normal depth, then the downstream GVF is an M_1. In this problem, the flow rate and the depth immediately downstream from the gate are known, and the unknowns are as follows: (1) the distance x where the jump occurs downstream from the gate, (2) the depth Y_1 immediately upstream from the jump, and (3) the depth Y_2 immediately downstream from the jump. The depth at the end of the channel will be considered to be known either from the water surface in the reservoir if this is given larger than the critical depth, or equal to the critical depth otherwise. Since the flow rate is known, this critical depth can be calculated when necessary.



For this problem, there are three equations: the momentum equation across the hydraulic jump, and two ODE equations for the two GVF profiles upstream and downstream from the hydraulic jump. If the channel is trapezoidal, then these three equations are

$$F_1 = \left(\frac{b}{2} + \frac{m}{3} Y_1\right) Y_1^2 + \frac{Q^2}{gA_1} - \left(\frac{b}{2} + \frac{m}{3} Y_2\right) Y_2^2 + \frac{Q^2}{gA_2} = 0$$

$$F_2 = Y_1 - Y_{1.ode}(Y_2) = 0 \quad \text{Solving the } M_3\text{-GVF}$$

$$F_3 = Y_2 - Y_{1.ode}(Y_3) = 0 \quad \text{Solving the GVF from the end of the channel up to the downstream side of the jump}$$

There are two approaches to solving this problem. The first is a computer implementation of the methodology used in Example Problem 4.4 by hand, i.e., solving the two gradually varied flow profiles, but rather than printing out the solution of depths Y and M's corresponding to the various positions x, these are stored in memory, and the position where the two momentum fuctions are equal is determined. In other words, the position of the hydraulic jump is where the x's and the M's are simultaneously equal from the two GVF solutions. The second approach is to use the Newton method to solve the above three equations simultaneously.

First, consider a computer program GVFJMP that uses the first approach. The program will need to solve two GVF profiles; the M_3 upstream of the hydraulic jump, and the one from the end of the channel, at least up to the hydraulic jump. Both these solutions should extend beyond where the jump actually occurs so that the solutions can be examined to determine the actual position. In addition to solving Y, at each increment values of the momentum functions need to be computed and stored in arrays. After these solutions are obtained, an interpolation algorithm needs to determine where the two values of x and the two values of the momentum function are simultaneously equal. Program GVFJMP.FOR is designed for this purpose.

After reading in the problem specification, this program solves the critical depth. Should the downstream reservoir water surface level WS be less than this Y_c, then it sets WS = $1.03Y_c$. Thus, the user does not need to computer Y_c, but rather just give a small value for WS to solve the case of a free overfall. If WS is given larger than Y_c, then the case of the channel discharging into a reservoir is accommodated. The ODE solver RUKUST, described in the appendix is used to solve the two GVF profiles. The program variables associated with the upstream GVF contain a U, i.e., XU, YU, and MU that are the upstream position, the corresponding depths and the momentum functions, and those associated with the downstream GVF contain D, i.e., XD, YD, MD, etc. Starting with the statement labeled 14, the interpolation of the two tables (i.e., arrays of values) begins to find the location where the two values of the two momentum functions are equal. The approach taken in this algorithm is to start at the upstream end of the channel, and find the index ID of the downstream position XD corresponding with XU(IU). Thereafter, the values of the two momentum functions are compared. If MU(IU) is larger than MD(ID), then the next solution entry is checked in the same manner, etc. After two entries in both the upstream solution table and the downstream solution table have been made, $M_d = a_d + b_d x$, in which the b's are evaluated from $(M_1 - M_2)/(x_1 - x_2)$ and the a's from $M_1 - bx_1$. Then setting the two M's equal provides the following interpolation equation for x:

$$x = \frac{(a_u - a_d)}{(b_d - b_u)}$$

The second approach uses the Newton method to solve the one algebraic and two ODEs, simultaneously. Most differential equation solvers, such as ODESOL, DVERK, and RUKUST are designed to solve systems of simultaneous equations but only over the same interval of the independent variable. Since the upstream and downstream GVF profiles need to be solved in different directions to solve the problem of locating the hydraulic jump between a gate and the downstream end of the channel, it will be necessary to call on the solver two different times. The upstream solution will be from 0 to x (the position of the jump), and the downstream GVF solution will be from L (the length of the channel) up to x. If the solver itself does not find an appropriately small interval to get the solution started near the critical depth, it may also be necessary to call on this solver within a DO loop to solve the downstream GVF. Fixed step Runge–Kutta methods fall in to this category. More sophisticated solvers, such as ODESOL and DVERK, however, need to be called on only once for the entire interval. Program GVFJMP2.FOR implements this second approach to solve the position of the hydraulic jump.

Program GVFJMP.FOR

```
      LOGICAL DOWNS
      REAL YY(1),YTT(1)
      REAL XU[ALLOCATABLE](:),YU[ALLOCATABLE](:),MU[ALLOCATABLE](:),
     &XD[ALLOCATABLE](:),YD[ALLOCATABLE](:),MD[ALLOCATABLE](:)
      COMMON B,FM,FM2,TWOM,FNQ,QG2,SO
      EQUIVALENCE (Y,YY(1))
      WRITE(6,*)'Give:IOU,Q,b,m,So,n,L,WS,Yb,dX1,dX2,g'
      READ(5,*) IOUT,Q,B,FM,SO,FN,FL,WS,YB,DX1,DX2,G
      IF(FM.LT.1.E-5) THEN
      YC=((Q/B)**2/G)**.333333
      GO TO 2
      ENDIF
      QP=FM**3*Q**2/(G*B**5)
      YC=.925*(.5*qp)**.284
      NCT=0
1     F=(YC+YC**2)**3/(1.+2.*YC)-QP
      YC1=1.01*YC
      X1=(YC1-YC)*F/((YC1+YC1**2)**3/(1.+2.*YC1)-QP-F)
      YC=YC-X1
      NCT=NCT+1
      IF(ABS(X1).GT.1.E-5 .AND. NCT.LT.20) GO TO 1
      YC=B*YC/FM
2     WRITE(IOUT,*)' Critical Depth =',YC
      IF(WS.LT.YC) THEN
      WRITE(IOUT,*)' Depth at end of channel is critical'
      WS=1.03*YC
```

```
      ENDIF
      YC1=.95*YC
      DXS=.05
      C=1.
      IF(G.GT.30.) C=1.486
      NUE=FL/ABS(DX1)+.5
      NDE=FL/ABS(DX2)+.5
      ALLOCATE(XU(NUE),YU(NUE),MU(NUE),XD(NDE),YD(NUE),MD(NDE))
      QG2=Q*Q/G
      TWOM=2.*FM
      FNQ=FN*Q/C
      FM2=2.*SQRT(FM*FM+1.)
      Y=WS
      A=(B+FM*Y)*Y
      X1=FL
      DX=-ABS(DX2)
      N=NDE
      DOWNS=.TRUE.
      XD(1)=FL
      YD(1)=Y
      MD(1)=(.5*B+FM/3.*Y)*Y*Y+QG2/A
      WRITE(IOUT,100) X1,Y,MD(1)
100   FORMAT(F10.2,F10.3,F10.1)
5     DO 10 I=2,N
      X=X1+DX
      CALL RUKUST(1,DXS,X1,X,1.E-5,YY,YTT)
      A=(B+FM*Y)*Y
      IF(DOWNS) THEN
      XD(I)=X
      YD(I)=Y
      MD(I)=(.5*B+FM/3.*Y)*Y*Y+QG2/A
      WRITE(IOUT,100) X,Y,MD(I)
      ELSE
      XU(I)=X
      YU(I)=Y
      MU(I)=(.5*B+FM/3.*Y)*Y*Y+QG2/A
      WRITE(IOUT,100) X,Y,MU(I)
      IF(Y.GT.YC1) THEN
      NUE=I
      GO TO 14
      ENDIF
      ENDIF
10    X1=X
      IF(DOWNS) THEN
      N=NUE
      Y=YB
      DX=ABS(DX1)
      DOWNS=.FALSE.
      XD(1)=0.
      YD(1)=Y
      A=(B+FM*Y)*Y
```

```
      MD(1)=(.5*B+FM/3.*Y)*Y*Y+QG2/A
      X1=0.
      WRITE(IOUT,100) X1,Y,MD(1)
      GO TO 5
      ENDIF
14    ID=NDE-4
      IDM=NDE-2
      IU=2
15    IU1=IU-1
      DO 20 WHILE (XD(ID).GT.XU(IU).AND.ID.LT.IDM)
20    ID=ID+1
      ID1=ID+1
      DO 30 WHILE (XD(ID1).LT.XU(IU1).AND.ID1.GT.1)
30    ID1=ID1-1
      ID=ID1-1
      IF(MU(IU).LT.MD(ID1) .OR. IU.EQ.NUE) GO TO 50
      IU=IU+1
      GO TO 15
50    BU=(MU(IU)-MU(IU1))/(XU(IU)-XU(IU1))
      WRITE(*,*) IU,IU1,ID,ID1
      AU=MU(IU)-BU*XU(IU)
      BD=(MD(ID)-MD(ID1))/(XD(ID)-XD(ID1))
      AD=MD(ID)-BD*XD(ID)
      X=(AU-AD)/(BD-BU)
      F=YU(IU1)+(X-XU(IU1))/(XU(IU)-XU(IU1))*
     &*(YD(IU)-YD(IU1))
      QP=YD(ID)+(X-XD(ID))/(XU(ID1)-XU(ID))*(YD(ID1)-YD(ID))
      WRITE(*,110) X,F,QP
      WRITE(IOUT,110) X,F,QP
110   FORMAT(' Position x=',F10.2,' Upst Depth Y1 =',F10.3,'
     &Downstr. Depth Y2 =',F10.3)
      DEALLOCATE(XU,YU,MU,XD,YD,MD)
      END
      SUBROUTINE SLOPE(X,Y,DYX)
      REAL Y(1),DYX(1)
      COMMON B,FM,FM2,TWOM,FNQ,QG2,SO
      A=(B+FM*Y(1))*Y(1)
      SF=(FNQ*((B+FM2*Y(1))/A)**.66666667/A)**2
      DYX(1)=(SO-SF)/(1.-QG2*(B+TWOM*Y(1))/A**3)
      RETURN
      END
```

Program GVFJMP.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#define sqr(x) x*x
float b,fm,fm2,twom,fnq,qg2,so;
extern void rukust(int neq,float *dxs,float xbeg,float xend,\
   float error,float *y,float *ytt);
```

```
void slope(float x,float *y,float *dyx){float a,sf;
   a=(b+fm*y[0])*y[0];sf=pow(fnq*pow((b+fm2*y[0])/a,.6666667)/a,2.);
   dyx[0]=(so-sf)/(1.-qg2*(b+twom*y[0])/(a*a*a));} // end slope
void main(void){int downs,nct,i,id,id1,idm,iu,iu1,nde,nue,n;
   char fmt[]="%10.2f %9.3f %9.1f\n";
   float q,fn,fl,ws,yb,dx1,c,*dxs,dx2,g,yc,qp,f,yc1,a,x,x1,dx,bu,au,bd,ad;
   float y[1],ytt[1],*xu,*yu,*mu,*xd,*yd,*md;
   FILE *fil;char fnam[20];
   printf("Give:Q,b,m,So,n,L,WS,Yb,dX1,dX2,g\n");
   scanf("%f %f %f %f %f %f %f %f %f %f %f",\
      &q,&b,&fm,&so,&fn,&fl,&ws,&yb,&dx1,&dx2,&g);
   if(fm<1.e-5) yc=pow(sqr(q/b)/g,.333333);
   else {qp=pow(fm,3.)*sqr(q)/(g*pow(b,5.)); yc=.925*pow(.5*qp,.284);nct=0;
   do{ f=pow(yc+sqr(yc),3.)/(1.+2.*yc)-qp; yc1=1.01*yc;
      x1=(yc1-yc)*f/(pow(yc1+sqr(yc1),3.)/(1.+2.*yc1)-qp-f);yc-=x1;
   }while((++nct<20) &&(fabs(x1)>1.e-5)); yc=b*yc/fm;}
   printf("Give name for output file\n");scanf("%s",fnam);
   if((fil=fopen(fnam,"w"))==NULL){
      printf("Failed to open file\n"); exit(0);}
   fprintf(fil,"Critical Depth =%f\n",yc);
   if(ws<yc){
      fprintf(fil,"Depth at end of channel is critical\n"); ws=1.03*yc;}
   yc1=.95*yc;*dxs=.05; c=1.; if(g>20.) c=1.486; nue=fl/fabs(dx1)+.5;
   nde=fl/fabs(dx2)+.5;
   xu=(float *)calloc(nue,sizeof(float));
   yu=(float *)calloc(nue,sizeof(float));
   mu=(float *)calloc(nue,sizeof(float));
   xd=(float *)calloc(nde,sizeof(float));
   yd=(float *)calloc(nue,sizeof(float));
   md=(float *)calloc(nde,sizeof(float));
   qg2=q*q/g;twom=2.*fm;fnq=fn*q/c;fm2=2.*sqrt(fm*fm+1.); y[0]=ws;
   a=(b+fm*y[0])*y[0]; x1=fl; dx=-fabs(dx2); n=nde;downs=1;xd[0]=fl;
   yd[0]=y[0];md[0]=(.5*b+fm/3.*y[0])*y[0]*y[0]+qg2/a;
   fprintf(fil,fmt,x1,y[0],md[0]);
L5: for(i=1;i<n;i++){x=x1+dx; rukust(1,dxs,x1,x,1.e-5,y,ytt);
      a=(b+fm*y[0])*y[0];
      if(downs){xd[i]=x;yd[i]=y[0];
         md[i]=(.5*b+fm/3.*y[0])*y[0]*y[0]+qg2/a;
         fprintf(fil,fmt,x,yd[i],md[i]);}
      else { xu[i]=x;yu[i]=y[0];mu[i]=(.5*b+fm/3.*y[0])*y[0]*y[0]+qg2/a;
         fprintf(fil,fmt,x,yu[i],mu[i]);
         if(y[0]>yc1){nue=i; goto L14;}}
      x1=x;} // end for
   if(downs){n=nue;y[0]=yb;dx=fabs(dx1);downs=0;xd[0]=0.;yd[0]=y[0];
     a=(b+fm*y[0])*y[0]; md[0]=(.5*b+fm/3.*y[0])*y[0]*y[0]+qg2/a; x1=0.;
      fprintf(fil,fmt,x1,y[0],md[0]); goto L5;}
L14:id=nde-5;idm=nde-3;iu=1;
L15:iu1=iu-1;
   while((xd[id]>xu[iu])&&(id<idm))id++;id1=id+1;
   while((xd[id1]<xu[iu1])&&(id1>0))id1–;id=id1-1;
```

```
   if((mu[iu]<md[id1]) || (iu==(nue-1))) goto L50;
   iu++;goto L15;
L50: bu=(mu[iu]-mu[iu1])/(xu[iu]-xu[iu1]);
   printf("%d %d %d %d\n",iu,iu1,id,id1);
   au=mu[iu]-bu*xu[iu];bd=(md[id]-md[id1])/(xd[id]-xd[id1]);
   ad=md[id]-bd*xd[id];x=(au-ad)/(bd-bu);
   f=yu[iu1]+(x-xu[iu1])/(xu[iu]-xu[iu1])*(yu[iu]-yu[iu1]);
   qp=yd[id]+(x-xd[id])/(xd[id1]-xd[id])*(yd[id1]-yd[id]);
   printf("Position x=%10.2f Upst Depth Y1=%10.3f Downstr Depth\
      Y2=%10.3f\n",x,f,qp);
   fprintf(fil,"Position x=%10.2f Upst Depth Y1=%10.3f Downstr Depth\
      Y2=%10.3f\n",x,f,qp);
   free(xu);free(yu);free(mu);free(xd);free(yd);free(md);}
```

EXAMPLE PROBLEM 4.12

Solve Example Problem 4.4 using program GVFJMP.

Solution

The input needed is, 3 480 8 0 .0011 .013 800 4.9 2 20 –20 32.2 and the solution is, x = 353.7, Y_1 = 3.459, and Y_2 = 6.497.

Program GVFJMP2.FOR

```
      INTEGER*2 INDX(3)
      REAL F(3),D(3,3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON/TRAS/ B,FM,FM2,TWOM,FNQ,QG2,SO,C,G,FL,YB,WS,X(3)
      WRITE(6,*)'Give:IOUT,Q,b,m,So,n,L,WS,Yb,g,',,'Est x,Y1&Y2'
      READ(5,*) IOUT,Q,B,FM,SO,FN,FL,WS,YB,G,X
      IF(FM.LT.1.E-5) THEN
      YC=((Q/B)**2/G)**.3333333   GO TO 2
      ENDIF
      QP=FM**3*Q**2/(G*B**5)
      YC=.925*(.5*qp)**.284
      YC1=1.01*YC
      NCT=0
1     FF=(YC+YC**2)**3/(1.+2.*YC)-QP
      X1=(YC1-YC)*FF/((YC1+YC1**2)**3/(1.+2.*YC1)-QP-FF)
      YC=YC-X1
      NCT=NCT+1
      IF(ABS(X1).GT.1.E-5 .AND. NCT.LT.20) GO TO 1
      YC=B*YC/FM
2     WRITE(IOUT,*)' Critical Depth =',YC
      IF(WS.LT.YC) THEN
      WRITE(IOUT,*)' Depth at end of channel',' is critical'
      WS=1.03*YC
      ENDIF
      C=1.
      IF(G.GT.30.) C=1.486
      QG2=Q*Q/G
      TWOM=2.*FM
```

```
      FNQ=FN*Q/C
      FM2=2.*SQRT(FM*FM+1.)
      NCT=0
3     DO 10 I=1,3
      F(I)=FUN(I)
      DO 10 J=1,3
      DX=.005*X(J)
      X(J)=X(J)+DX
      D(I,J)=(FUN(I)-F(I))/DX
10    X(J)=X(J)-DX
      CALL SOLVEQ(3,1,3,D,F,1,DD,INDX)
      SUM=0.
      DO 12 I=1,3
      SUM=SUM+ABS(F(I))
12    X(I)=X(I)-F(I)
      NCT=NCT+1
      WRITE(*,*) NCT,SUM,X
      IF(NCT.LT.30 .AND. SUM.GT. 2.E-3) GO TO 3
      WRITE(IOUT,100) X
100   FORMAT(' x =',F10.2,' Y1 =',F10.2,' Y2 =',F10.2)
      END
      FUNCTION FUN(II)
      EXTERNAL DYX
      REAL Y(1),W(1,13),DY(1),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON/TRAS/
      B,FM,FM2,TWOM,FNQ,QG2,SO,C,G,FL,YB,WS,X(3)
      HMIN=.0001
      H1=1.
      GO TO (1,2,3),II
1     FUN=(.5*B+FM/3.*X(2))*X(2)**2+QG2/((B+FM*X(2))*X(2))-
     &(.5*B+FM/3.*X(3))*X(3)**2-QG2/((B+FM*X(3))*X(3))
      RETURN
2     Y(1)=YB
      CALL ODESOL(Y,DY,1,0.,X(1),1.E-6,H1,HMIN,1,XP,YP,W,DYX)
      FUN=X(2)-Y(1)
      RETURN
3     Y(1)=WS
      H1=-1.
      CALL ODESOL(Y,DY,1,FL,X(1),1.E-6,H1,HMIN,1,XP,YP,W,DYX)
      FUN=X(3)-Y(1)
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1)
      COMMON/TRAS/ B,FM,FM2,TWOM,FNQ,QG2,SO,C,G,FL,YB,WS,X(3)
      YY=ABS(Y(1))
      A=(B+FM*YY)*YY
      SF=(FNQ*((B+FM2*YY)/A)**.66666667/A)**2
      DY(1)=(SO-SF)/(1.-QG2*(B+TWOM*YY)/A**3)
      RETURN
      END
```

EXAMPLE PROBLEM 4.13

A flow rate Q = 400 cfs passes under a gate with a depth Y_b = 1.5 ft downstream from the gate. The channel downstream from the gate is trapezoidal with b = 10 ft, m = 1, n = 0.014, and a bottom slope S_o = 0.001. At a length 1000 ft downstream from the gate, the channel ends in a free overfall. Locate the position of the hydraulic jump using program GVFJMP2, i.e., solving the momentum equation and the two GVF equations simultaneously.

Solution

The input to program GVFJMP2 consists of 6 400 10 1 .001 .014 1000 0 1.5 32.2 300 2.5 4.1

The solution is: x = 319.0 ft, Y_1 = 2.493 ft, and Y_2 = 4.106 ft.

EXAMPLE PROBLEM 4.14

Solve Example Problem 4.8 writing a computer program that implements the Newton method in simultaneously solving the three algebraic and one ODE equations that govern the problem.

Solution

The listing of a FORTRAN program that solves Example Problem 4.8 iteratively by means of the Newton method follows. The basic difference between it and SOLGVF is that it solves four simultaneous equations rather than three. The equations are (1) energy at the entrance, (2) energy across the gate, (3) energy across the transition, and (4) the GVF from the downstream gate to the reservoir.

Program EPRB4_14.FOR

```
      PARAMETER (N=4)
      REAL F(N),D(N,N),X(N),KL1,KE1,KL,KE
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,FM1,B2,H,G,G2,KL1,KE1,FL,TOL,FN,SO,CC,QN,
     &Q2G,X,Y4,A4,QSG
      EQUIVALENCE (Q,X(1)),(Y1,X(2)),(Y2,X(3)),(Y3,X(4))
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,n,So,b1,m1,b2,H,L,YG,g,KL,
     &KE,Cc,Ns,YG2'
      READ(*,*) IOUT,TOL,ERR,FN,SO,B1,FM1,B2,H,FL,YG,G,KL,KE,
     &CCO,NS,YG2
      WRITE(IOUT,200)
200   FORMAT(' Gate Flow rate   Depths',' (meters)',/,'  (m)
     &(m**3/s) at Beg.C.',' at End C. at End T.')
      DYG=(YG-YG2)/FLOAT(NS-1)
      Y4=CCO*YG
      A4=(Y4*B2)**2
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      KL1=1.+KL
      KE1=1.+KE
      WRITE(*,*)' GIVE guess for: Q,Y1,Y2,Y3'
      READ(*,*) X
50    NCT=0
1     DO 10 I=1,N
      F(I)=FUN(I)
      DO 10 J=1,N
      DX=.005*X(J)
      X(J)=X(J)+DX
      D(I,J)=(FUN(I)-F(I))/DX
```

```
10    X(J)=X(J)-DX
C  Solves system of equations using Gaussian
C Elimination
      DO 12 J=1,N-1
      DO 12 I=J+1,N
      FAC=D(I,J)/D(J,J)
      F(I)=F(I)-FAC*F(J)
      DO 12 K=J+1,N
12    D(I,K)=D(I,K)-FAC*D(J,K)
      F(N)=F(N)/D(N,N)
      X(N)=X(N)-F(N)
      SUM=ABS(F(N))
      DO 16 I=N-1,1,-1
      FAC=0.
      DO 14 J=I+1,N
14    FAC=FAC+D(I,J)*F(J)
      F(I)=(F(I)-FAC)/D(I,I)
      X(I)=X(I)-F(I)
16    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,110) NCT,SUM,X
110   FORMAT(' NCT=',I2,F12.2,4F10.4)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 1
      WRITE(IOUT,100) YG,X
100   FORMAT(F6.2,4F10.3)
      YG=YG-DYG
      IF(YG.LT.YG2-.01) STOP
      Y4=CCO*YG
      A4=(Y4*B2)**2
      GO TO 50
      END
      FUNCTION FUN(II)
      EXTERNAL DYX
      REAL X(4),W(1,13),KL1,KE1,Y(1),DY(1),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,FM1,B2,H,G,G2,KL1,KE1,FL,TOL,FN,SO,CC,QN,
     &Q2G,X,Y4,A4,QSG
      H1=-.05
      HMIN=.001
      Q2G=X(1)*X(1)/G2
      GO TO (1,2,3,4),II
1     FUN=H-X(2)-KE1*Q2G/((B1+FM1*X(2))*X(2))**2
      RETURN
2     FUN=X(3)+Q2G/((B1+FM1*X(3))*X(3))**2-X(4)-KL1*Q2G/
     &(B2*X(4))**2
      RETURN
3     FUN=X(4)+Q2G/(B2*X(4))**2-Y4-Q2G/A4
      RETURN
4     Y(1)=X(3)
      XX=FL
      XZ=0.
      QN=(FN*X(1)/CC)**2
      QSG=X(1)*X(1)/G
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      FUN=X(2)-Y(1)
      RETURN
```

```
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1),KL1,KE1,X(4)
      COMMON/TRAS/B1,FM1,B2,H,G,G2,KL1,KE1,FL,TOL,FN,SO,CC,QN,
     &Q2G,X,Y4,A4,QSG
      EQUIVALENCE (Q,X(1)),(Y1,X(2)),(Y2,X(3)),(Y3,X(4))
      P=B1+2.*SQRT(FM1*FM1+1.)*Y(1)
      A=(B1+FM1*Y(1))*Y(1)
      SF=QN*((P/A)**.666666667/A)**2
      T=B1+2.*FM1*Y(1)
      DY(1)=(SO-SF)/(1.-QSG*T/A**3)
      RETURN
      END
```

Input data to solve the above problem consists of

```
3 .00001 .001 .013 .0009 3 2 2.5 2 500 2 9.81 .09 .12 .75 10 .2
```

with guess for the unknown as: 15 2 2.35 1.6

Solution to problem:

Gate (m)	Flow Rate (m**3/s)	Depths (m)		
		at Beg. C.	at End C.	at End T.
2.00	14.993	1.926	2.345	1.586
1.80	14.795	1.929	2.348	1.740
1.60	14.228	1.935	2.357	1.879
1.40	13.332	1.943	2.370	2.004
1.20	12.137	1.954	2.385	2.114
1.00	10.671	1.965	2.401	2.211
0.80	8.955	1.976	2.416	2.292
0.60	7.012	1.985	2.430	2.358
0.40	4.861	1.993	2.440	2.408
0.20	2.518	1.998	2.447	2.43

The following observations will help you understand this program: there are two subroutines; one, DYX, that defines the derivative dY/dx for the solver ODESOL as usual, and the other a function subprogram, FUN, that evaluates any of the four equations when asked to do so. When asked to evaluate F_4, this function calls on the solver ODESOL to provide the solution over the entire length of the GVF profile. The entire length can be used in a single call to ODESOL since we are not interested in having the depths corresponding to the x's over the length. If desired, a table of x and Y values could be obtained after the flow rate is determined. The main program defines the problem by reading in appropriate variables, defines the Jacobian matrix D (which is stored in the array D(N,N)), and the equation vector (which is stored in the array F(N)). Thereafter, it solves this linear system of equations and implements the Newton method. Finally, it decreases the gate height and solves the new problem.

The program listed below uses the same logic as the above program, but is programmed using Borland's C-Language.

C-Language program EPRB4_14.C designed to solve Example Problem 4.14

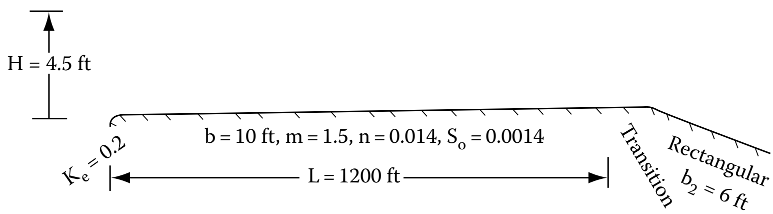
```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "odesol.h"
#define sqr(a) (a*a)
```

```
#define N 4
float f[N],d[N][N],x[N],kl1,ke1,kl,ke,b1,fm1,b2,h,g=9.81,g2=19.62,fn,\
   so,cc=1.,qn,q2g,y4,a4,qsg,tol,err,fl;
float fun(int ii,float dx,int j){
   float h1=-.05,hmin=.001,x1[N]; int i;float *y;
   y=(float *)calloc(nv,sizeof(float));
   for(i=0;i<4;i++) x1[i]=x[i];
   if(j > -1) x1[j]+=dx;q2g=sqr(x1[0])/g2;
   switch (ii){
   case 0:return(h-x1[1]-kl1*q2g/sqr((b1+fm1*x1[1])*x1[1]));
   case 1:return(x1[2]+q2g/sqr((b1+fm1*x1[2])*x1[2])-x1[3]-kl1*q2g/\
      sqr(b2*x1[3]));
   case 2:return(x1[3]+q2g/sqr(b2*x1[3])-y4-q2g/a4);
   case 3: y[0]=x1[2];
   qn=sqr(fn*x1[0]/cc); qsg=sqr(x1[0])/g;
   odesolc(y,fl,0.,tol,h1,hmin,nstor); return(x1[1]-y[0]);}} /* end of fun*/
void slope(float x,float *y,float *dydx) { float p,a,sf,t;
   p=b1+2.*sqrt(fm1*fm1+1.)*y[0]; a=(b1+fm1*y[0])*y[0];
   sf=sqr(qn*pow(p/a,0.666666667)/a); t=b1+2.*fm1*y[0];
   dydx[0]=(so-sf)/(1.-qsg*t/(a*sqr(a))); return;} /* end of slope */
void main(void) { int iout,ns,step,nct,k,i,j;
   float yg,yg2,dyg,cco,dx=0.,fac,sum; char filena[12];
   FILE *fil;
   cprintf("GIVE:IOUT,TOL,ERR,n,So,b1,m1,b2,H,L,YG,g,KL,KE,Cc,Ns,YG2r\n");
   scanf("%d %f %f %f %f %f %f %f %f %f %f %f %f %f %f %d %f",&iout,\
      &tol,&err,&fn,&so,&b1,&fm1,&b2,&h,&fl,&yg,&g,&kl,&ke,&cco,&ns,&yg2);
   if(iout !=6){
      cprintf("Give filename for solution data\r\n");scanf("%s",filena);
   fil=fopen(filena,"w");
   fprintf(fil," Gate Flow rate Depths (meters)\n");
   fprintf(fil," (m) (m**3/s) at Beg.C. at End C. at End T.\n");}
   dyg=(yg-yg2)/(ns-1); if(g>30.) cc=1.486;
   g2=2.*g; kl1=1.+kl; ke1=1.+ke;
   cprintf("Give guess for:Q,Y1,Y2,Y3r\n");
   scanf("%f %f %f %f",&x[0],&x[1],&x[2],&x[3]);
   for(step=1;step<=ns;step++) {
      y4=cco*yg;a4=sqr(y4*b2); nct=0;
      do {
         for(i=0;i<N;i++){
            f[i]=fun(i,dx,-1);
            for(j=0;j<N;j++){dx=.005*x[j];
               d[i][j]=(fun(i,dx,j)-f[i])/dx;}}
/* Solves system of equations using Gaussian Elimination */
      for(j=0;j<N-1;j++) {
         for(i=j+1;i<N;i++){fac=d[i][j]/d[j][j]; f[i]=f[i]-fac*f[j];
           for(k=j+1;k<N;k++) d[i][k]=d[i][k]-fac*d[j][k];} } j=N-1;
      f[j]=f[j]/d[j][j]; x[j]=x[j]-f[j]; sum=fabs(f[j]);
      for(i=N-2;i>=0;i–){fac=0.;for(j=i+1;j<N;j++) fac=fac+d[i]\[j]*f[j];
         f[i]=(f[i]-fac)/d[i][i]; x[i]=x[i]-f[i];sum+=fabs(f[i]);}
      nct++;
      cprintf("%3d %11.2f %9.4f %9.4f %9.4f %9.4f\r\n",\
         nct,sum,x[0],x[1],x[2],x[3]);
      } while (nct<30 && sum>err);
      if(iout!=6) fprintf(fil,"%6.2f %9.3f %9.3f %9.3f \
         %9.3f\n",yg,x[0],x[1],x[2],x[3]);
      else cprintf("%6.2f %9.3f %9.3f %9.3f \
```

```
      %9.3f\r\n",yg,x[0],x[1],x[2],x[3]);
    yg-=dyg; } /* end of step */
if(iout!=6) fclose(fil);}
```

EXAMPLE PROBLEM 4.15

A trapezoidal channel that is 1200 ft long receives its water from a reservoir with a water surface elevation H = 4.5 ft above its bottom. The bottom slope of the channel is $S_o = 0.0014$, n = 0.014 b = 10 ft, and m = 1.5. The entrance loss coefficient is $K_e = 0.2$. At the downstream end of this channel there is a rectangular channel with a bottom width of $b_2 = 6$ ft, and this downstream rectangular channel is steep. Assume that the loss coefficient for this downstream transition is $K_L = 0.12$.



Solution

This problem can be solved using program SOLGVF. It is solved using the program SOLGVFDF, whose listing is provided below. This program illustrates the following alternative techniques to those used in SOLGVF: (1) it calls on DVERK (the FORTRAN program, and the C program calls on rukust), rather than ODESOL as the ODE-solver; (2) it calls on the linear algebra equation solver SOLVEQ, rather than having the Gaussian elimination method built into the code: and (3) it calls on the subroutine FUN(F), in which the argument F is an array containing the three equation values upon return, rather than calling on FUN(II) once to evaluate each equation, and thereafter again with each unknown incremented to evaluate the elements of the Jacobian as is done in SOLGVF. Program SOLGVFDF calls subroutine FUN(F) four times; the first time to obtain the values of the equation vector F, and then three times thereafter with each of the unknowns incremented by multiplying their current value by 1.005. Note this alternative means that the Jacobian matrix evaluation requires two arrays (F and FF) to hold the values of the equations and evaluates the rows within the inner loop with the columns as the outer loop, and results in N + 1 calls to the subroutine FUN(F) that evaluates the equations, rather than N + N*N calls to the function FUN(II), in which N is the number of equations. Study the nested DO 12 loop to understand how the elements of the Jacobian are evaluated.

The input data to solve this problem consists of (The same input works for SOLGVF.)

```
6 1.e-5 .0001 .014 .0014 10 1.5 6 4.5 1200 32.2 .12 .2
250 4 6
```

and the solution is Q = 250.2 cfs, $Y_1 = 4.26$ ft, $Y_2 = 5.83$ ft.

Listing of Program SOLGVFDF.FOR

```
C Generates Jacobian by columns by calling subroutine FUN that
  supplies all eqs.
C Also calls on linear algebraic equation solver SOLVEQ
      INTEGER*2 INDX(3)
      REAL F(3),FF(3),D(3,3),X(3),KL2,KE1,KL,KE
      COMMON B1,FM1,B2,H,G,G2,KL2,KE1,FL,TOL,FN,SO,CC,QN,Q2G,X
     WRITE(*,*)' GIVE:IOUT,TOL,ERR,FN,SO,B1,FM1,''B2,H,L,g,KL,KE'
      READ(*,*) IOUT,TOL,ERR,FN,SO,B1,FM1,B2,H,FL,G,KL,KE
      IF(G.GT.30.) THEN
```

```
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      KL2=.5*KL+1.5
      KE1=1.+KE
      WRITE(*,*)' GIVE guess for: Q,Y1,Y2'
      READ(*,*) X
      NCT=0
1     CALL FUN(F)
      DO 12 J=1,3
      XX=X(J)
      X(J)=1.005*X(J)
      CALL FUN(FF)
      DO 10 I=1,3
10    D(I,J)=(FF(I)-F(I))/(X(J)-XX)
12    X(J)=XX
      CALL SOLVEQ(3,1,3,D,F,1,DET,INDX)
      SUM=0.
      DO 14 I=1,3
      X(I)=X(I)-F(I)
14    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 1
      WRITE(IOUT,100) X
100   FORMAT(' Q =',F10.2,' Y1 =',F10.2,' Y2 =',F10.2)
      END
      SUBROUTINE FUN(F)
      EXTERNAL DYX
      REAL X(3),C(24),W(1,9),KL2,KE1,Y(1),F(3)
      COMMON B1,FM1,B2,H,G,G2,KL2,KE1,FL,TOL,FN,SO,CC,QN,Q2G,X
      Q2G=X(1)*X(1)/G2
      F(1)=X(3)+Q2G/((B1+FM1*X(3))*X(3))**2-KL2*((X(1)/
     &B2)**2/G)**.333333333
      F(2)=H-X(2)-KE1*Q2G/((B1+FM1*X(2))*X(2))**2
      Y(1)=X(3)
      XX=FL
      XZ=0.
      IND=1
      QN=(FN*X(1)/CC)**2
      Q2G=X(1)*X(1)/G
      CALL DVERK(1,DYX,XX,Y,XZ,TOL,IND,C,1,W)
      F(3)=X(2)-Y(1)
      RETURN
      END
      SUBROUTINE DYX(N,XX,Y,YPRIME)
      REAL Y(N),YPRIME(N),KL2,KE1,X(3)
      COMMON B1,FM1,B2,H,G,G2,KL2,KE1,FL,TOL,FN,SO,CC,QN,Q2G,X
      P=B1+2.*SQRT(FM1*FM1+1.)*Y(1)
      A=(B1+FM1*Y(1))*Y(1)
      SF=QN*((P/A)**.666666667/A)**2
      T=B1+2.*FM1*Y(1)
      YPRIME(1)=(SO-SF)/(1.-Q2G*T/A**3)
      RETURN
      END
```

Listing of Program SOLGVFKF.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
float b1,fm1,fm12,fm1s,b2,h,g,g2,kl2,ke1,fl,tol,fn,so,cc,\
   fnq,q2g,x[3];
extern void rukust(int neq,float *dxs,float xbeg,float xend,\
   float error,float *y,float *ytt);
extern void solveq(int n,float **a,float *b,int itype,\
   float *dd,int *indx);
void slope(float x,float *y,float *dyx){float a,sf;
   a=(b1+fm1*y[0])*y[0];
   sf=pow(fnq*pow((b1+fm1s*y[0])/a,.6666667)/a,2.);
   dyx[0]=(so-sf)/(1.-q2g*(b1+fm12*y[0])/(a*a*a));} // end slope
void fun(float *f){float xx,xz,y[1],ytt[1],*dxs;
   q2g=x[0]*x[0]/g2;
   f[0]=x[2]+q2g/pow((b1+fm1*x[2])*x[2],2.)-\
      kl2*pow(pow(x[0]/b2,2.)/g,.3333333);
   f[1]=h-x[1]-ke1*q2g/pow((b1+fm1*x[1])*x[1],2.);
   y[0]=x[2]; fnq=pow(fn*x[0]/cc,2.);q2g=2.*q2g;*dxs=.02;
   rukust\(1,dxs,fl,0.,tol,y,ytt);f[2]=x[1]-y[0]; } // End fun
void main(void){int nct,i,j,indx[3];
   float xx,err,sum,f[3],ff[3],*dxs,*det,**d;
   d=(float**)malloc(3*sizeof(float*));
   for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
   printf("Give: TOL,ERR,n,So,b1,m1,b2,H,L,g,KL,Ke\n");
   scanf("%f %f %f %f %f %f %f %f %f %f %f %f",\
      &tol,&err,&fn,&so,\&b1,&fm1,&b2,&h,&fl,&g,&kl2,&ke1);
   if(g>20.) cc=1.486; else cc=1.; g2=2.*g;kl2=.5*kl2+1.5;ke1+=1.;
   fm12=2.*fm1;fm1s=sqrt(fm1*fm1+1.);
   printf("Give guess for: Q,Y1,Y2\n");
   scanf("%f %f %f",&x[0],&x[1],&x[2]); nct=0;
   do{ fun(f); for(j=0;j<3;j++){xx=x[j]; x[j]+=1.005; fun(ff);
         for(i=0;i<3;i++) d[i][j]=(ff[i]-f[i])/(x[j]-xx); x[j]=xx;}
      solveq(3,d,f,1,det,indx); sum=0.;
      for(i=0;i<3;i++){x[i]-=f[i]; sum+=fabs(f[i]);}
   } while((++nct<30) && (sum>err));
   printf("Q = %10.2f, Y1 =%10.2f, Y2 =%10.2f\n",x[0],x[1],x[2]);}
```

A simultaneous solution of the energy equation and Manning' equation at the entrance of the channel produces the following normal flow rate and depth: $Q_o = 385.8$ cfs and $Y_o = 3.56$ ft. The input data to these programs consists of

```
6 .000001 .001 .014 .0014 10 1.5 6 4.5 1200 32.2 .12 .2
250 4 6
```

and the solution produced consists of

$$Q = 250.16 \text{ cfs}, \quad Y_1 = 4.26 \text{ ft}, \quad \text{and} \quad Y_2 = 5.83 \text{ ft}.$$

Software packages such as Mathcad, TK-Solver, etc., can also be utilized to handle problems involving the simultaneous solution of algebraic and ODEs. For example, the above problem of flow from a reservoir into a channel that changes to a steep channel after a relative short distance is solved by TK-Solver with the following "Rule" and "Variable" sheets. The library function RK4_se that comes with this package is utilized to solve the GVF equation. This library requires that a function be added that defines the ODE that is to be solved. This function is DYDx and consists of a statement that evaluates dY/dx in Equation 4.4 (see listing below). The list of x's defines where the solution for Y is to be obtained at. The "rules" basically include four equations: (1) the critical flow at the beginning of the steep channel, (2) the energy across the transition, (3) the GVF ODE equation starting with the depth Y_2 at the end of the upstream mild channel to the entrance where it produces Y_1, and (4) the specific energy at the entrance of the channel. This TK-Solver model can be operated in two ways. The first way is to give a guess for the flow rate and to give the variable Y2 the status of G (for guess), and then press the F9 key to obtain the solution that includes H. The H from this solution is then compared with the water surface elevation and the flow rate is adjusted until the two compare. The second, and most advantageous method, is to let the TK-Solver's iterative solver do the entire job, by giving H a value, and indicating that the variables, Y1, Y2, and Q are guesses as shown below.

TK-Solver model to solve from into channel from short upstream reservoir to break to a steep grade.

Before solution

```
VARIABLE SHEET
St     Input       Name
L      'yyy        Y
       .014        n
       10          b
       1.5         m
       1.486       C
       .0014       So
       32.2        g
       .2          Ke
       6           b2
       4.5         H
       G 200       Q
       G 6         Y2
       G 4.3       Y1
                   Yc
                   Ec
```

After solution

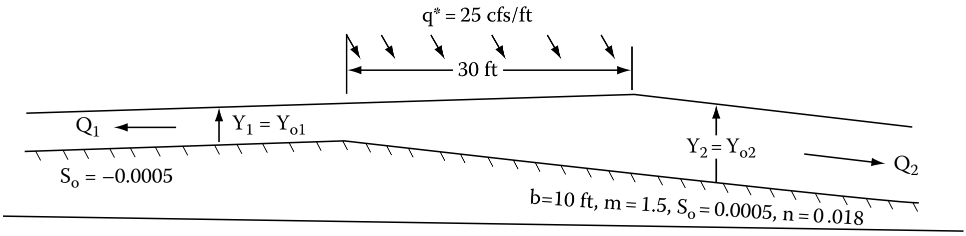
```
VARIABLE SHEET
St    Input     Name      Output
L     'yyy      Y
      .014      n
      10        b
      1.5       m
      1.486     C
      .0014     So
      32.2      g
      .2        Ke
      6         b2
      4.5       H
                Q         265.73
                Y2        5.81
                Y1        4.22
                Yc        3.93
                Ec        5.90
```

```
==== RULE SHEET========
S Rule─────────────────
Yc=((Q/b2)^2/g)^.3333333
Ec=1.5*Yc
Y2+(Q/((b+m*Y2)*Y2))^2/(2*g)=Ec
  place(Y,1)=Y2
call RK4_sen('DYDx,Y,'x)
Y1=ELT(Y,21)
H=Y1+(1+Ke)*(Q/((b+m*Y1)*Y1))^2/(2*g)

==== FUNCTION SHEET=====
Name────────── Type───── Arguments── Comment──
DYDx            Procedure 2;1        Given differential equation
RK4_sen Procedure 3;0
Classical 4th-Order Runge-Kutta

============ PROCEDURE FUNCTION: DYDx =========
Comment:                 Given differential equation
Parameter Variables: n,b,m,C,Q,So,g
Input Variables:     x,y
Output Variables:    y'
S Statement──────────────────────────────────────────────────────
 y':=(So-(n*(b+2*sqrt(m*m+1)*y)^.66666667*Q/(C*((b+m*y)*y)
   ^1.666667))^2)/(1-Q*Q/g/((b+m*y)*y)^3)

============ PROCEDURE FUNCTION: RK4_sen ========
Comment:                 Classical Fourth-Order Runge-Kutta method,
                         single eq-n
Parameter Variables: b,m,g,Q
Input Variables:     EQ,y,x
Output Variables:
S Statement──────────────────────────────────────────────────────
 ; Notation: EQ  name of a function with the 1st-order equation
                 y'=f(x,y)
 ;           x   independent variable (list)
 ;           y   dependent variable, y=F(x), (list)
 ;           K   Runge-Kutta coefficients (list)
 ; Description: This procedure represents an implementation of a
 ; classical 4th-order Runge-Kutta procedure for numerical
 ; integration of a single ordinary differential equation
 ; y'=f(x,y). Given a function name passed as a symbolic value
 ; of the Input Variable EQ, list of values of the independent
 ; variable x, and an initial condition as the value of the 1st
 ; element of the list y, the procedure generates the solution
 ; in the rest of the list y.
 xi:= x[1]
 yi:= y[1]
 for i=2 to length(x)
    ye:= yi
    h:= (x[i]-xi)/2
    for j=1 to 3
       'K[j]:= apply(EQ,xi,ye)
       if mod(j,2) then xi:= xi + h
       if j=3 then h:= 2*h
       ye:= yi + h*'K[j]
    next j
```

```
    'K[4]:= apply(EQ,xi,ye)
    yi:= yi + dot('K,1,2,2,1)*h/6
    y[i]:= yi
 next i
 call delete('K)

════ LIST: x ════════
independent variable
Element—— Value——————
1         1200
2         1140
3         1080
4         1020
5         960
6         900
7         840
8         780
9         720
10        660
11        600
12        540
13        480
14        420
15        360
16        300
17        240
18        180
19        120
20        60
21        0

═════ TABLE: Solution ═════
Title:    y = f(x)
Element    x        y
1        1200     5.809
2        1140     5.727
3        1080     5.646
4        1020     5.564
5         960     5.483
6         900     5.402
7         840     5.321
8         780     5.240
9         720     5.160
10        660     5.079
11        600     5.000
12        540     4.920
13        480     4.841
14        420     4.762
15        360     4.683
16        300     4.605
17        240     4.528
18        180     4.451
19        120     4.374
20         60     4.299
21          0     4.224
```

EXAMPLE PROBLEM 4.16

A channel receives its water over a 30 ft length. The flow is free to move in both directions from this inflow section. The inflow is constant at a rate $q^* = 25$ cfs/ft, has an incoming velocity of $U = 8$ fps, and is directed at an angle of 45° downward and to the right as shown. From this inflow section, the channel slopes downward in both directions with $S_o = 0.0005$. The channel has a bottom width of $b = 10$ ft, a side slope of $m = 1.5$, and a Manning's $n = 0.018$ in both directions. Solve the flow rates and depths in both directions.



Solution

In this problem, there are four unknowns: Q_1 (flow rate to the left), Q_2 (flow rate to the right), and the depths Y_1 and Y_2 in the channels flowing to the left and right respectively. The four equations needed to be solved for these four unknowns consist of

$$F_1 = Q_1 + Q_2 - Lq^* = 0$$

$$F_2 = Q_1 - C_u/n\ A_1 R_{h1}^{2/3}/S_o = 0$$

$$F_3 = Q_2 - C_u/n\ A_2 R_{h2}^{2/3}/S_o = 0$$

$F_4 = Y_{o2} - Y_{gvf} = 0$ (in which the GVF solution will begin at the beginning of the inflow section with $Y = Y_{o1}$.)

The solution to the problem gives $Y_1 = 4.60$ ft, $Y_2 = 5.76$ ft, $Q_1 = 293.5$ cfs, and $Q_2 = 456.5$ cfs. The program that obtains this solution is listed below. It calls on a linear algebra solver SOLVEQ that returns the solution in the vector F. Notice that the subroutine DYX that supplies dY/dx to the ODE-Solver shows that the flow rate (variable QQ) is negative upstream from where the flow divides, and if $(S_f)^{1/2}$ is solved from Manning's equation with Q negative, a negative value is produced, but when a negative is squared, a positive value results. Therefore, S_f is computed from $S_f = (S_f)^{1/2}\left|S_f^{1/2}\right|$, so it is negative in the numerator of the ODE.

FORTRAN listing SOLINF.FOR to solve above four equations.

```
      REAL F(4),D(4,4),X(4)
      INTEGER*2 INDX(4)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B,FM,FMS,FM2,SS,QS,G,Uqx,FL,TOL,FN,SO,CC,X
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,FN,SO,B,FM,L,g,qs,Uq,Angle'
      READ(*,*) IOUT,TOL,ERR,FN,SO,B,FM,FL,G,QS,Uq,Angle
      Uqx=Uq*COS(.017455329*Angle)
      FM2=2.*FM
      FMS=2.*SQRT(FM*FM+1.)
      SS=SQRT(SO)
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      WRITE(*,*)' GIVE guess for: Q1,Q2,Y1,Y2'
```

```
      READ(*,*) X
      NCT=0
1     DO 10 I=1,4
      F(I)=FUN(I)
      DO 10 J=1,4
      DX=.005*X(J)
      X(J)=X(J)+DX
      D(I,J)=(FUN(I)-F(I))/DX
10    X(J)=X(J)-DX
      CALL SOLVEQ(4,1,4,D,F,1,DD,INDX)
      WRITE(6,222) F
      NCT=NCT+1
      DIF=0.
      DO 20 I=1,4
      X(I)=X(I)-F(I)
20    DIF=DIF+ABS(F(I))
      IF(NCT.LT.30 .AND. DIF.GT. ERR) GO TO 1
      WRITE(IOUT,100) X
100   FORMAT(' Q1 =',F10.2,' Q2 =',F10.2,' Y1 =',F10.2,' Y2 =',
     &F10.2)
      END
      FUNCTION FUN(II)
      EXTERNAL DYX
      REAL X(4),W(2,13),Y(1),DY(1),XP(2),YP(2,2)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B,FM,FMS,FM2,SS,QS,G,Uqx,FL,TOL,FN,SO,CC,X
      GO TO (1,2,3,4),II
1     FUN=X(1)+X(2)-FL*QS
      RETURN
2     FUN=FN*X(1)*(B+FMS*X(3))**.666667-CC*((B+FM*X(3))*
     &X(3))**1.666667*SS
      RETURN
3     FUN=FN*X(2)*(B+FMS*X(4))**.666667-CC*((B+FM*X(4))*
     &X(4))**1.666667*SS
      RETURN
4     Y(1)=X(3)
      H1=.05
      HMIN=.001
      XX=0.
      XZ=FL
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      FUN=X(4)-Y(1)
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1),X(4)
      COMMON /TRAS/B,FM,FMS,FM2,SS,QS,G,Uqx,FL,TOL,FN,SO,CC,X
      YY=ABS(Y(1))
      P=B+FMS*YY
      A=(B+FM*YY)*YY
      A2=A*A*G
      QQ=QS*XX-X(1)
      VD=QQ/A-Uqx
      Q2=QQ*QQ
      SF=FN*ABS(QQ)/CC*(P/A)**.666666667/A
      SF=SF*ABS(SF)
```

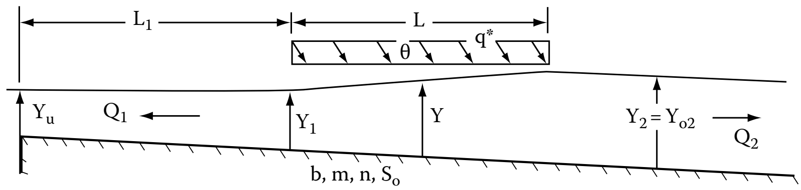
```
      DY(1)=(SO-SF-QQ*QS/A2-VD*QS/(G*A))/(1.-Q2*(B+FM2*YY)/(A*A2))
      RETURN
      END
```

C listing SOLINF.C to solve the above four equations

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
float b,fm,fms,fm2,ss,qs,g,uqx,fl,tol,fn,so,cc,x[4];
extern void solveq(int n,float **d,float *f,int itype,\
   float *dd, int *indx);
extern rukust(int n,float *dxs,float xb,float xe,float err,\
   float*y,float *ytt);
void slope(float xx,float *y,float *dy){
   float yy,p,a,a2,qq,vd,q2,sf;
   yy=fabs(*y);p=b+fms*yy;a=(b+fm*yy)*yy;a2=a*a*g;
   qq=qs*xx-x[0];
   vd=qq/a-uqx;q2=qq*qq;sf=fn*fabs(qq)/cc*pow(p/a,.6666667)/a;
   sf*=fabs(sf);
   *dy=(so-sf-qq*qs/a2-vd*qs/(g*a))/(1.-q2*(b+fm2*yy)/(a*a2));}
   // End of slope
float fun(int ii){float dxs[1],y[1],ytt[1],xx,xz,arg;
   if(ii==0) arg=x[0]+x[1]-fl*qs;
   else if(ii==1) arg=fn*x[0]*pow(b+fms*x[2],.666667)-\
      cc*ss*pow((b+fm*x[2])*x[2],1.666667);
   else if(ii==2) arg=fn*x[1]*pow(b+fms*x[3],.666667)-\
      cc*ss*pow((b+\fm*x[3])*x[3],1.666667);
   else{y[0]=x[2];dxs[0]=.05;xx=0.,xz=fl;
      rukust(1,dxs,xx,xz,tol,y,ytt);arg=x[3]-y[0];}
   return arg;} // End fun
void main(void){int i,j,nct,indx[4];
   float dx,dif,uq,angle,err,f[4],dd[1],**d;char fname[20];
   FILE *filo;
   d=(float**)malloc(4*sizeof(float*));
   for(i=0;i<4;i++)d[i]=(float*)malloc(4*sizeof(float));
   printf("Give name of output file\n");scanf("%s",fname);
   if((filo=fopen(fname,"w"))==NULL){
      printf("Cannot open file"); exit(0);}
   printf("GIVE:TOL,ERR,FN,SO,B,FM,L,g,qs,Uq,Angle\n");
   scanf("%f %f %f %f %f %f %f %f %f %f %f",\
      &tol,&err,&fn,&so,&b,\&fm,&fl,&g,&qs,&uq,&angle);
   uqx=uq*cos(.017455329*angle); fm2=2.*fm;
   fms=2.*sqrt(fm*fm+1.); ss=sqrt(so);
   if(g>30.) cc=1.486; else cc=1.;
   printf("Give guess for: Q1,Q2,Y1,Y2\n");
   for(i=0;i<4;i++)scanf("%f",&x[i]);nct=0;
   do{for(i=0;i<4;i++){ f[i]=fun(i); for(j=0;j<4;j++){
         dx=.005*x[j];x[j]+=dx; d[i][j]=(fun(i)-f[i])/dx;
         x[j]-=dx;}}
      solveq(4,d,f,1,dd,indx); dif=0.;
      for(i=0;i<4;i++){
         x[i]-=f[i];dif+=fabs(f[i]);}
      printf("nct=%d, dif=%f\n",++nct,dif);
   }while((nct<30) && (dif>err));
   fprintf(filo," Q1 =%10.2f, Q2 =%10.2f, Y1 =%10.3f,
   Y2=%10.3f\n",\
      x[0],x[1],x[2],x[3]);
   fclose(filo);}
```

In solving the above problem, it was assumed that the slope of the channel upstream from the beginning of the lateral inflow was equal, but in the opposite direction. If the channel has a constant slope in the same direction, then a uniform flow cannot exit upstream, and eventually the depth will become zero for a wedge of water upstream from the inflow that is stationary under a steady-state flow. For this situation, all the lateral inflow will contribute to the downstream flow rate $Q_2 = q^*L$. To solve this case, since Q_2 is known, first solve the downstream uniform depth Y_{o2}, and as a second step solve the GVF (with the lateral inflow term included) starting at the downstream end with Y_{o2} for Y_1. For this example problem, $Q_2 = 750$ cfs, $Y_{o2} = 7.366$ ft, and $Y_1 = 6.855$ ft.

Another case is that the slope S_o continues upstream for an additional distance L_1 as shown in the sketch below. At the upstream end, there will be a boundary condition that establishes the depth Y_u. The three possible conditions are (1) the channel ends in a free overfall, (2) the channel discharges into a reservoir with a known water surface elevation WS, or (3) there is a channel with a bottom slope in the opposite direction and it contains a uniform flow. Notice that we have now added one more unknown Y_u so that Q_1, Q_2, Y_1, $Y_2 = Y_{o2}$, and Y_u are five unknowns. The five equations needed for case (1) are



$$F_1 = Q_1 + Q_2 - Lq^* = 0$$

$$F_2 = Q_1^2 T_u - gA_u^3 = 0 \quad \text{(the critical flow equation)}$$

$$F_3 = Q_2 - C_u/n\ A_2 R_{h2}^{2/3}/S_o = 0 \quad \text{(Manning's equation downstream)}$$

$$F_4 = Y_1 - Y_{1ode}(Y_u) = 0 \quad \text{(ODE solution over } L_1 \text{ starting with } Y_u\text{)}$$

$$F_5 = Y_{o2} - Y_{2ode}(Y_1) = 0 \quad \text{(ODE sol. over L starting with } Y_1 \text{ and including lateral inflow terms)}$$

If the second boundary condition with a known reservoir water surface WS applies, then $Y_u = WS$, and again only four variables are unknown. An alternative for case (2) is to replace the second equation with $F_2 = Y_u - WS = 0$ and keep Y_u as unknown. For the third case, the second equation would be replaced by Manning's equation writing the function of Y_u equal to zero. For example, $F_2 = nQP_u^{2/3} - C_u A_u^{5/3} S_{o2}^{1/2} = 0$. The program SOLINF2, whose listing is given below, is designed to solve problems with an upstream channel with a length L_1 upstream from the lateral inflow, and any of the three above mentioned upstream boundary conditions.

Listing of Program SOLINF2.FOR

```
      EXTERNAL DYX
      REAL F(5),D(5,5),X(5)
      INTEGER*2 INDX(5)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON/TRAS/B,FM,FMS,FM2,SS,QS,G,Uqx,FL,FL1,TOL,FN,SO,
     &CC,X,So2,WS,IBC,INFLOW
```

```
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,FN,SO,B,FM,L,L1,g,qs,Uq,Angle,
     &BC(1=Yc,2=WS or 3=Channel)'
      READ(*,*) IOUT,TOL,ERR,FN,SO,B,FM,FL,FL1,G,QS,Uq,Angle,IBC
      IF(IBC.EQ.2) THEN
      WRITE(*,*)' Give upstream ws-elevation'
      READ(*,*) WS
      ENDIF
      IF(IBC.EQ.3) THEN
      WRITE(*,*)' Give slope of upstream channel'
      READ(*,*) So2
      So2=SQRT(So2)
      ENDIF
      Uqx=Uq*COS(.017455329*Angle)
      FM2=2.*FM
      FMS=2.*SQRT(FM*FM+1.)
      SS=SQRT(SO)
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      WRITE(*,*)' GIVE guess for: Q1,Q2,Y1,Y2,Yu'
      READ(*,*) X
      NCT=0
1     DO 10 I=1,5
      F(I)=FUN(I,DX,0)
      DO 10 J=1,5
      DX=.005*X(J)
10    D(I,J)=(FUN(I,DX,J)-F(I))/DX
      CALL SOLVEQ(5,1,5,D,F,1,DD,INDX)
      NCT=NCT+1
      DIF=0.
      DO 20 I=1,5
      X(I)=X(I)-F(I)
20    DIF=DIF+ABS(F(I))
      WRITE(*,200) NCT,DIF,X
200   FORMAT(' NCT=',I2,' DIF=',E12.5,/,2F10.2,3F10.3)
      IF(NCT.LT.30 .AND. DIF.GT. ERR) GO TO 1
      WRITE(IOUT,100) X
100   FORMAT(' Q1 =',F10.2,' Q2 =',F10.2,' Y1 =',F10.2,' Y2 =',
     &F10.2,' Yu =',F10.2)
      END
      FUNCTION FUN(II,DX,J)
      EXTERNAL DYX
      REAL X(5),W(2,13),Y(1),DY(1),XP(2),YP(2,2)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON/TRAS/B,FM,FMS,FM2,SS,QS,G,Uqx,FL,FL1,TOL,FN,SO,CC,X,
     &So2,WS,IBC,INFLOW
      IF(J.GT.0) X(J)=X(J)+DX
      GO TO (1,2,3,4,5),II
```

```
1      FUN=X(1)+X(2)-FL*QS
       GO TO 10
2      IF(IBC.EQ.1) THEN
       FUN=X(1)**2*(B+FM2*X(5))-G*((B+FM*X(5))*X(5))**3
       ELSE IF(IBC.EQ.2) THEN
       FUN=WS-X(5)
       ELSE
       FUN=FN*X(1)*(B+FMS*X(5))**.666667-CC*((B+FM*X(5))*X(5))
      &**1.666667*So2
       ENDIF
       GO TO 10
3      FUN=FN*X(2)*(B+FMS*X(4))**.666667-CC*((B+FM*X(4))*X(4))
      &**1.666667*SS
       GO TO 10
4      Y(1)=X(5)
       IF(IBC.EQ.1) Y(1)=1.1*Y(1)
       INFLOW=0
       H1=.01
       HMIN=.00001
       XX=0.
       XZ=FL1
       CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
       FUN=X(3)-Y(1)
       GO TO 10
5      Y(1)=X(3)
       INFLOW=1
       H1=.01
       XZ=FL
       CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
       FUN=X(4)-Y(1)
10     IF(J.GT.0) X(J)=X(J)-DX
       RETURN
       END
       SUBROUTINE DYX(XX,Y,DY)
       REAL Y(1),DY(1),X(5)
       COMMON/TRAS/B,FM,FMS,FM2,SS,QS,G,Uqx,FL,FL1,TOL,FN,SO,
      &CC,X,So2,WS,IBC,INFLOW
       YY=ABS(Y(1))
       P=B+FMS*YY
       A=(B+FM*YY)*YY
       A2=A*A*G
       IF(INFLOW.EQ.1) THEN
       QSS=QS
       QQ=QS*XX-X(1)
       VD=QQ/A-Uqx
       ELSE
       QQ=-X(1)
       VD=0.
       QSS=0.
       ENDIF
```

```
      Q2=QQ*QQ
      SF=FN*QQ/CC*(P/A)**.666666667/A
      SF=SF*ABS(SF)
      DY(1)=(SO-SF-QSS*(QQ/A2+VD/(G*A)))/(1.-Q2*(B+FM2*YY)/(A*A2))
      RETURN
      END
```

Let us solve the previous problem with an upstream channel with a length L_1 = 500 ft using the three boundary conditions. The input to solve case (1) for a free overfall is

```
6 1.e−5 .001 .018 .0005 10 1.5 30 500 32.2 25 8 45 1
300 450 4.6 5.7 2.7
```

with the output for the solution as

Q1 = 308.91 Q2 = 441.09 Y1 = 4.33 Y2 = 5.66 Yu = 2.69

For case (2), let us specify a reservoir water surface elevation 4 ft above the channel bottom. The input now becomes

```
6 1.e−5 .001 .018 .0005 10 1.5 30 500 32.2 25 8 45 2
4
300 450 4.6 5.7 4.
```

with the output:

Q1 = 289.44 Q2 = 460.56 Y1 = 4.67 Y2 = 5.79 Yu = 4.00

For case (3), let us give the channel upstream a bottom slope S_{o2} = 0.001; then the input to INFSOL2 is

```
6 1.e−5 .001 .018 .0005 10 1.5 30 500 32.2 25 8 45 3
.001
300 450 4.6 5.7 4.4
```

with the output:

Q1 = 294.51 Q2 = 455.49 Y1 = 4.59 Y2 = 5.76 Yu = 3.84

A couple of comments are in order. First, it was assumed that the long channel upstream from Y_u had the same size as the channel downstream, therefore only its bottom slope was in the opposite direction. Should this channel be of a different size, or there be a difference in the bottom elevation, then Y_u would need to be replaced by two depths Y_{u1} and Y_{u2} in these two different channels. Thus, there would be six unknown variables. The additional sixth equation would be the energy equation across the transition between the two channels, or $F_6 = Y_{u2} + Q_1^2/(gA_{u2}^2) - Y_{u1} - (1 + K_L) Q_1^2/(gA_{u1}^2) - \Delta z = 0$.

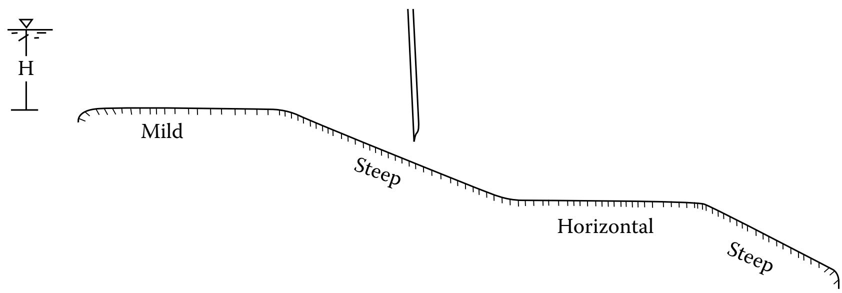
The second comment is that if the GVF is solved from the beginning of the channel where the depth is Y_u to the end of the lateral inflow where the depth is $Y_2 = Y_{o2}$, then Y_1 could be eliminated as an unknown. The solution to the ODE would need to turn to the lateral inflow terms when the solution is with the length L, but have these terms zero when the solution is within L_1. If this alternative is used, then equations F_4 and F_5 would be replaced with the single equation $F_4 = Y_{o2} - Y_{2ode}(Y_u) = 0$.

4.10 NONPRISMATIC CHANNELS

Transitions between channels of different sizes can alter the type of GVF profile that exists in a channel. The GVF sketches that were previously shown in this chapter assumed that the channel is of constant size and shape throughout its entire length. Such channels are called prismatic channels. If the slope of the land changes, it would not be economical to use the same large channel for steep slopes, that are needed for very mild slopes. The size of the channel will in practice, therefore, be changed to reflect the changing land slopes. When a channel changes from one size to another, or from one shape to another, it is referred to as a nonprismatic channel. The following examples illustrate how effects due to nonprismatic channels can alter the type of GVF that may exist in different portions of the channel system.

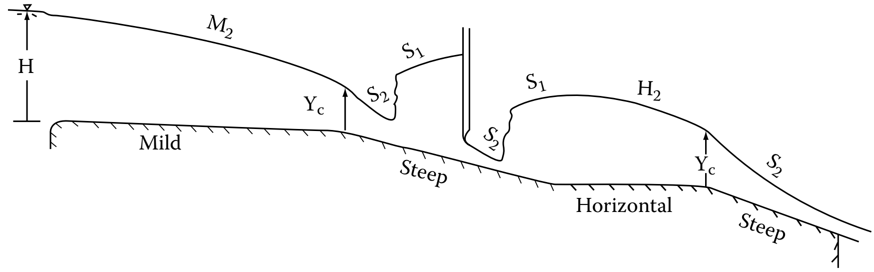
EXAMPLE PROBLEM 4.17

Sketch in the GVF profile in the channel shown below assuming it is prismatic throughout its entire length.



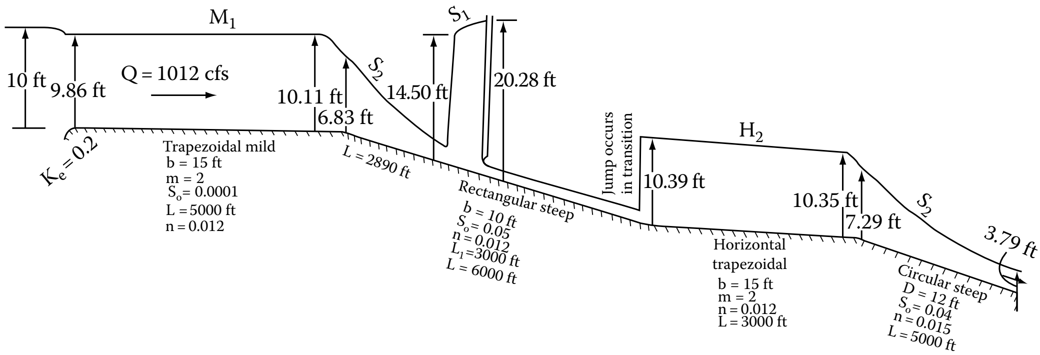
Solution

Since this channel is prismatic, i.e., neither its shape nor size changes throughout its entire length, the GVF profiles are as given below.



EXAMPLE PROBLEM 4.18

The channel in the previous problem consists of (1) a 5000 ft long trapezoidal channel, followed by; (2) a 6000 ft long rectangular channel that contains a sluice gate, which is followed by; (3) a horizontal trapezoidal channel with a 3000 ft length; and (4) a 5000 ft long steep circular channel as shown in the sketch below. Determine the flow rate entering this channel if the upstream reservoir level is 10 ft above the channel bottom. Also, locate any hydraulic jumps and determine the water surface profiles throughout the channel. Smooth transitions occur between the different shapes of the channel, and the bottom elevation does not change through the transitions. The gate with a contraction coefficient of $C_c = 0.6$ has its tip at $Y_G = 5$ ft above the channel bottom so the depth downstream from the gate is 3 ft.



Solution

The transitions can alter the types of GVF profiles considerably from the previous problem. The question, "What is the flow rate?" raises several other questions, such as (a) Does the critical flow section between the upstream mild trapezoidal channel and the steep rectangular channel "choke" the flow causing an M_1 GVF profile to exist in the upstream channel rather than an M_2 GVF profile? (b) Does the sluice gate increase the depth at the head of the rectangular channel above the critical depth? (c) If the S_1 GVF profile upstream from the gate does not extend up to the break in grade, where will the hydraulic jump occur? (d) Where will the hydraulic jump downstream from the gate occur? (e) What will the depth be at the end of the horizontal channel before the transition changes the cross section to that of a circle? Unfortunately, the answers to all of these questions depend upon knowing the flow rate, and the flow rate will depend upon whether the M_2 or M_1 GVF profile, which ever it is, extends to the reservoir or not.

To arrive at the solution, begin by assuming that a normal depth does exist in the upper portion of the first trapezoidal channel. If so, a simultaneous solution of Manning's equation and the energy equation will give the flow rate and the normal depth. These equations are

$$F_1 = \frac{nQP^{2/3} - 1.486A^{5/3}}{S_o} = 0 \tag{1}$$

and

$$F_2 = H - Y_o - (1 + K_L)Q^2(2gA^2) = 0 \tag{2}$$

The solution gives Q = 1320.3 cfs and Y_o = 9.707 ft. With this flow rate, the critical depth at the head of the next rectangular channel can be calculated from $Y_c = \sqrt[3]{q^2/g} = 8.150$ ft (q = 132.03 cfs/ft), and $E_c = 1.5\ Y_c = 12.225$ ft. If the losses through this transition are ignored, then the depth in the trapezoidal channel at the beginning of the transition can be obtained from

$$Y_2 + \frac{Q^2}{\left(2gA_2^2\right)} = 12.225$$

giving Y_2 = 12.105 ft. Since this depth is above the normal depth Y_o = 9.697 ft, the transition has "choked" the flow causing an M_1 GVF profile to exist rather than an M_2 as sketched in for the previous problem. Starting from this downstream depth in the trapezoidal channel, the M_1 GVF profile will be solved. Since the depth may not be normal at the entrance of the channel, it will be more convenient to use an ODE for this computation that assumes Y is the dependent variable and x is the independent variable. The FORTRAN listing below utilizes ODESOL described in Appendix C to obtain this solution. The input data to this program consists of

```
3 .001 −500 12.105 1320.3 .0001 .012 15 2 5000 0
```

FORTRAN listing that utilizes ODESOL (EPRB4_18.FOR)

```
      REAL Y(1),DY(1),XP(1),YP(1,1),WK1(1,13)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE,B,FM,FN,SO,Q2,FNQ
      WRITE(6,* )'GIVE IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND'
1     READ(5,*) IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND
      H1=-.01
      Y(1)=YB
      FNQ=FN*Q/1.49
      Q2=Q*Q/32.2
      X=XBEG
      WRITE(IOUT,100) X,Y
2     XZ=X+DELX
      CALL ODESOL(Y,DY,1,X,XZ,TOL,H1,HMIN,1,XP,YP,WK1,DYX)
      X=XZ
      WRITE(IOUT,100) X,Y
100   FORMAT(6X,2F10.3)
      IF(DELX .LT. 0.) GO TO 8
      IF(X .LT. XEND) GO TO 2
      GO TO 1
8     IF(X .GT. XEND) GO TO 2
      STOP
      END
      SUBROUTINE DYX(X,Y,DY)
      REAL Y(1),DY(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE,B,FM,FN,SO,Q2,FNQ
20    A=(B+FM*Y(1))*Y(1)
      T=B+2.*FM*Y(1)
      P=B+2.*SQRT(FM*FM+1.)*Y(1)
      SF=(FNQ*(P/A)**.66666667/A)**2
      A3=A**3
      FR2=Q2*T/A3
40    DY(1)=(SO-SF)/(1.-FR2)
      RETURN
      END
```

```
Solution M1 GVF profile
     x(ft)     Y(ft)
    -------    -------
     5000      12.105
     4500      12.073
     4000      12.042
     3500      12.011
     3000      11.980
     2500      11.949
     2000      11.918
     1500      11.888
     1000      11.858
      500      11.828
        0      11.798
```

The depth of 11.798 at the beginning of the trapezoidal channel is well above the depth computed based on uniform flow conditions, and is even above the water surface elevation in the reservoir. Therefore, the flow rate must be reduced and the entire process, given above, repeated. This flow rate must be selected by trial. After several trials, the flow rate Q = 1012 cfs is used. Based on this flow rate, the critical depth at the end of the first transition is $Y_c = 6.826$ ft, $E_c = 10.239$ ft, and $Y_2 = 10.114$ ft. The solution to the M_1 GVF profile gives

x (ft)	Y (ft)
5000.000	10.114
4500.000	10.088
4000.000	10.061
3500.000	10.035
3000.000	10.010
2500.000	9.985
2000.000	9.959
1500.000	9.935
1000.000	9.910
500.000	9.886
.000	9.862

If the upstream depth of $Y_1 = 9.862$ ft is substituted into the energy equation

$$H = Y_1 + Q^2/(2gA^2) = 10.02 \text{ ft}$$

which is close enough to the 10 ft of head on the reservoir. The flow rate is now known and the rest of the problem can be solved. This solution is given below, but you should verify the details, because getting them right will ensure that you understand what needs to be done. (You should also solve this problem using the Newton method as has been done in the last Example Problems 4.6 and 4.7.)

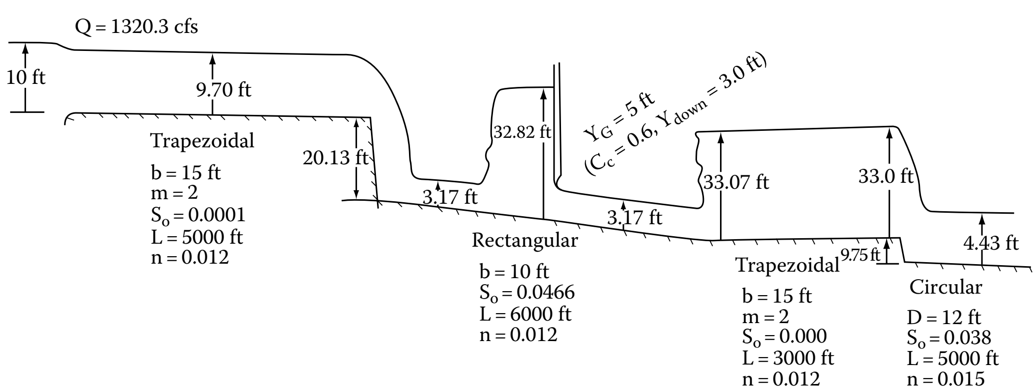
Having a critical section at the end of a transition, as in Example Problem 4.18, that reduces the flow rate in the channel from 1320.3 to 1012 cfs, does not constitute a very effective design. Elevation changes through the transitions are needed to prevent such occurrences. This will often require some earth work to change the slope of the channel instead of having the bottom of the channel on the same grade as the ground. However, since ground elevations seldom change abruptly, this will be needed anyway. The subject of the design of transitions is covered in Chapter 5.

EXAMPLE PROBLEM 4.19

Redesign the channel in the previous problem by changing the bottom elevation through the transition, so that a uniform depth occurs throughout the entire length of the upstream trapezoidal channel. Also change the bottom elevation through the transition just upstream from the circular section.

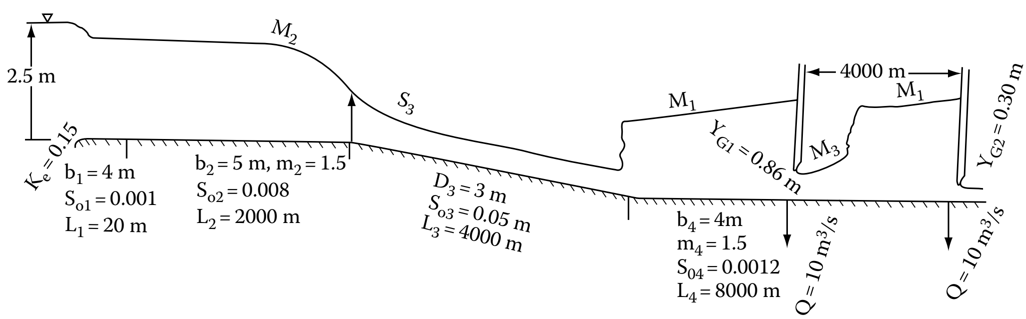
Solution

The sketch below shows this redesigned channel.



EXAMPLE PROBLEM 4.20

In the sketch below, a channel system is shown that receives water from a reservoir with a water surface elevation 2.5 m above the channel bottom. At its beginning, the channel is rectangular with $b_1 = 4$ m, $S_{o1} = 0.001$, and a length $L_1 = 20$ m, then there is a short transition to a trapezoidal channel with $b_2 = 5$ m, $m_2 = 1.5$, $S_{o2} = 0.0008$, and $L_2 = 2000$ m, then again a short transition to a circular section with $D_3 = 3$ m, $S_{o3} = 0.05$, and $L_3 = 4000$ m, and finally a third transition to a trapezoidal channel with $b_4 = 4$ m, $m_4 = 1.5$, $S_{o4} = 0.0012$, and $L_4 = 8000$ m. Manning's n = 0.013 for all channels. The last trapezoidal channel contains two gates, one at its end, and the other 4000 m upstream therefrom. Immediately in front of each gate, a diversion of 10 m^3/s takes place and each gate has a contraction coefficient of $C_d = 0.58$. The gates are set at $Y_{G1} = 0.86$ m and $Y_{G2} = 0.30$ m, respectively, above the channel bottoms. Determine the flow rate into this system and compute the water surface profiles throughout the system's length. (The entrance loss coefficient is $K_e = 0.15$.)



Solution

Note that channel 3, which is the pipe, is steep since $S_{o3} = 0.05$, and therefore a critical depth is expected at its beginning. The solution will be accomplished without the aid of a computer program that solves a system of algebraic and ordinary differential equations, simultaneously. Thus, the solution must begin by estimating what the flow rate will be. Limiting values for Q are likely between (a) that would occur if the upstream rectangular channel were very long, and (b) if the channel began with the second trapezoidal channel, and it were very long. The simultaneous solution of the energy equation and Manning's equation for these two cases gives 21.13 and 44.69 m^3/s, respectively, with corresponding normal depths of $Y_{o1} = 2.138$ and $Y_{o2} = 2.085$ m, respectively. The flow rate might be outside of these bounds if the transition to the circular section caused an M_1 GVF profile that extended to the reservoir, or if an M_2 GVF profile from this position extended to the reservoir. The latter seems unlikely. Start by assuming Q = 25 m^3/s, then the critical depth and the specific energy at the beginning of the circular section are $Y_{c3} = 2.192$ and $E_{c3} = 3.23$ m, respectively. Equating E_2 at the downstream end of the 5 m wide trapezoidal channel to 3.23 m and solving the depth gives $Y_{2end} = 3.198$ m (the normal depth in this trapezoidal channel for a flow rate of 25 m^3/s is $Y_{o2} = 1.521$ m). Next, solving the GVF profile for the beginning of this trapezoidal channel gives $Y_{2beg} = 1.801$ with a corresponding specific energy $E_{2beg} = 1.967$. However, if we check the specific energy in the upstream rectangular channel associated with a critical flow of Q = 25 m^3/s, we find that $E_{c1} = 2.378$ m ($Y_{c1} = 1.585$ m), ($E_{c1} + K_eV_{c1}^2/(2g) = 2.507$ m), which is larger than E_{1beg}, and therefore the backwater curve in the trapezoidal channel will have no effect on the flow rate since $E_{c1} = 2.378$ m is greater than $E_{2beg} = 1.967$ m. The upper limiting flow rate rather than being 44.69 m^3/s, as determined above, will be that obtained by the simultaneous solution of the energy, and the critical flow equation for channel 1, or 24.82 m^3/s (with $Y_{c1} = 1.587$ m). The actual flow rate will be slightly less than this because the frictional losses in the upstream 20 m long channel will reduce this. After a few trial flow rates, shown below, followed by solving the M_2 GVF profile to the reservoir, it is found that Q = 24.8 m^3/s (e.g., the upstream 20 m long channel frictional loss has had an insignificant effect in reducing the flow rate).

Trial	Q (m^3/s)	Y_c (m)	Y_{1beg} (m)	E (m)
1	23	1.499	1.684	2.361
2	24	1.542	1.731	2.435
3	24.3	1.555	1.746	2.456
4	24.6	1.568	1.762	2.478
5	24.8	1.577	1.771	2.594

The normal depths associated with this flow rate of Q = 24.8 m^3/s in each of the channels are Y_{o2} = 1.515 m, Y_{o3} = 1.016 m, Y_{o4} = 1.496 m, respectively, and the normal depths associated with the reduced flow rates of 14.8 m^3/s downstream from the first gate is 1.132 m.

The difference between the critical specific energy in channel 1 (E_{c1} = 2.37 m) and that associated with the normal depth in channel 2 (E_{o2} = 1.77 m) will be dissipated in the expansion. The solution to the two M_1 GVF profiles in front of the two gates are shown below, and because of the small differences in the normal depths between channels 3 and 4, the hydraulic jump that takes the supercritical flow from channel 3 to the subcritical flow in channel 4 will take place in the enlarging transition.

M_1 GVF profile upstream from gate # 2

x (m)	Y (m)	E (m)	M (m^3)
8000.000	2.271	2.310	17.50
7600.000	1.809	1.885	11.34
7200.000	1.391	1.547	7.85
6800.00	1.159	1.411	6.82
6400.00	1.134	1.401	6.76
6000.00	1.133	1.401	6.75
5600.00	1.132	1.400	6.75

M_1 GVF profile upstream from gate # 1

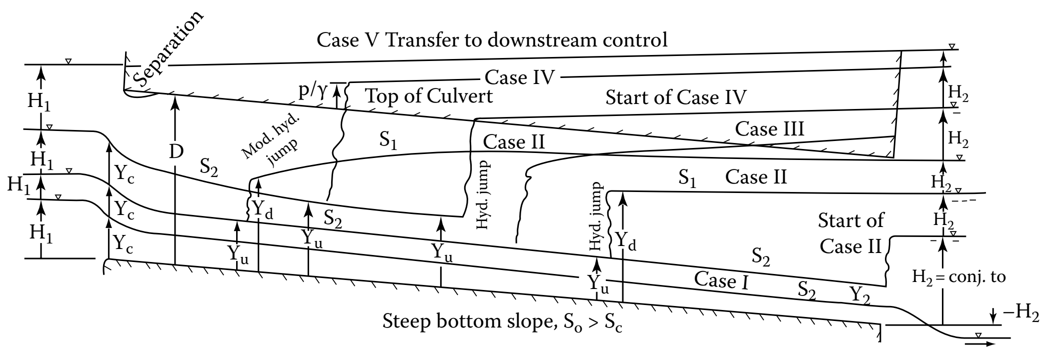
x (m)	Y (m)	E (m)	M (m^3)
4000.000	2.394	2.489	21.77
3600.000	1.957	2.127	16.02
3200.000	1.615	1.906	13.37
2800.000	1.505	1.858	12.89
2400.000	1.497	1.856	12.87
2000.000	1.496	1.855	12.86

4.11 CULVERTS

An application that involves applying the principles covered thus far, including solutions to GVFs, is the prediction of flow rates and depths throughout culverts. The term culvert is used for circular, oval, rectangular, square or other shaped conduits that pass beneath highways, railroads, or other embankments. The flow through a culvert can be controlled by upstream or downstream conditions. If the culvert has a steep bottom slope, its flow rate (and entrance depth) may be controlled upstream by having a critical flow at its entrance. But if the GVF caused by the downstream depth extends to its entrance, it will not have a critical flow at it entrance even if the bottom slope is steep. It is easier for a downstream control to exist in a short steep culvert than in longer culverts. If the bottom slope is mild, then the flow through the culvert will always be downstream controlled. Generally, the velocities upstream and downstream from culverts are small. Therefore, we will assume reservoirs exist upstream and downstream with water surface elevations H_1 and H_2, respectively, above the culverts invert (bottom). Since the velocity downstream of the culvert is small, the velocity head in the culvert will be assumed to be dissipated as it enters the downstream reservoir.

4.11.1 Solutions When Upstream Control Exists

The sketch below depicts the flow patterns that may occur when an upstream control exists. For **Case I**, GVFs exist throughout the culvert's length. If H_2 is less than the



downstream depth Y_2 at the end of the S_2 GVF profile, then the flow ignores it because it is supercritical. As the downstream depth rises above Y_2, a roller will appear at the end of the culvert, but in the reservoir. This roller will not penetrate the culvert until the depth H_2 becomes larger than the conjugate depth to depth Y_2. With downstream depths above the conjugate depth, a hydraulic jump will move into the culvert, and this represents the start of **Case II** shown in the above sketch. The higher H_2 the further this jump will move up the culvert. When a hydraulic jump occurs within the culvert but does not close the top of the culvert, the profiles are denoted as Case II on the above sketch. Since the slope of the channel bottom S_o is larger than S_f, and the flow is subcritical, the depth will increase beyond the jump in the downstream direction along an S_1 GVF profile. Therefore, the hydraulic jump will not be able to close the top of the culvert unless the depth H_2 is larger than the height of the culvert. **Cases III and IV** on the sketch show these possibilities, in which the downstream portion of the flow in the culvert is a closed conduit flow (also commonly called pipe flow even if the culvert is not circular), and the upstream portion contains a supercritical open channel flow. Case III is distinguished from IV in that the conjugate depth to the depth Y_u at the end of the S_2 GVF is less than D, rather than equal to or above D. Thus, for Case III there is an S_2 GVF starting at the entrance, a hydraulic jump, an S_1 GVF that ends at the top of the culvert, and then a pipe flow to the culvert's end. In **Case IV** a modified jump occurs in which the top of the culvert downstream therefrom has a pressure head on it equal to $p/\gamma = d = (M_1 - M_t)/A_t$, the equation derived in Chapter 3 in the section "Open Channel to Pipe Flow." Finally, as the downstream depth H_2 rises, it will eventually push the modified hydraulic jump up to the entrance. When this occurs, control is transferred downstream.

The sketch does not show all the possibilities, and the cases shown are not all possible in a given culvert. For example, if the bottom slope of the culvert is large so that the Froude number associated with the flow at its end is large, then the subcritical conjugate depth to this depth may be larger than the diameter of the culvert; in which case there can be no Case II because the roller will be kept in the downstream reservoir for H_2 above the top of the culvert. In order for Case IV to occur, the momentum function associated with the supercritical flow must be larger than the momentum function for a full conduit flow with no pressure head on the top, or $M_t = A_t(D/2) + Q^2/(gA_t)$. The position where the momentum function of the supercritical flow equals M_t represents the "Start of Case IV." For a given size culvert and a specified flow rate, or a specified upstream reservoir head H_1, Case IV will generally be downstream of what is shown as "Start of Case IV" on the sketch as the bottom slope S_o is increased more than needed for this limiting case. In other words, the cases shown on the sketch do not necessarily occur in the sequence shown as H_2 is increased, but rather depend upon all the variables involved in describing the flow through a culvert with a critical flow at its entrance. In other words, you need to visualize the bottom slope, Manning's n, etc., being varied, as well as H_1 and H_2 to understand the different cases shown on the sketch.

For Cases I, II, and III, the solution consists of first simultaneously solving the energy and the critical flow equations. For Case I, the gradually varied S_2 GVF profile is solved to the end of the channel. For Case II, the governing equations are

$$F_1 = H_1 - Y_c - \frac{(1+K_g)Q^2}{2gA^2} = 0 \text{ Variables}$$

$$F_2 = Q^2T - gA^3 = 0 \quad Q = \text{Flow rate,} \quad Y_c = \text{Critical depth at entrance}$$

$$F_3 = Y_u - Y_{uode} = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f}{1 - F_r^2} \quad \text{for } S_2 \text{ GVF} \quad Y_u = \text{Depth upstr. H. J.,} \; Y_d = \text{Depth D. H. J.}$$

$$F_4 = (Ah_c)_u + \frac{Q^2}{gA_u} - (Ah_c)_d - \frac{Q^2}{gA_d} = 0 \quad x = \text{Position of H. J.}$$

$$F_5 = Y_d - Y_{dode} = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f}{1 - F_r^2} \quad \text{for } S_1 \text{ GVF}$$

The unknowns are Q, Y_c, Y_u, Y_d, and x, but since a critical flow at the entrance governs, equations F_1 and F_2 are first solved simultaneously (or solved using Equation 2.16) for Q and Y_c, as for Case I, and thereafter equations F_3, F_4, and F_5 are solved simultaneously for the depths upstream and downstream from the jump and the position of the jump, or Y_u, Y_d, and x.

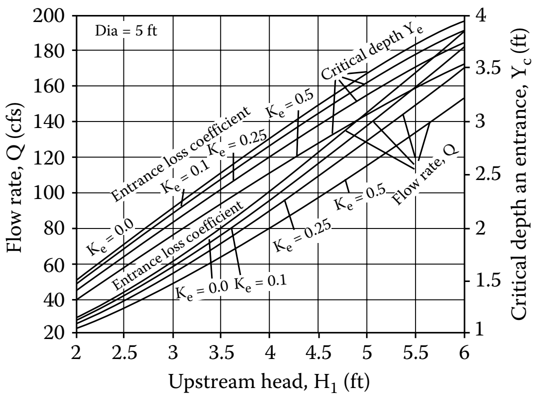
When the jump hits the top of the culvert as shown in Case IV (and beyond), then Y_d is no longer an unknown, but the pressure head $d = p/\gamma$ on the top of the culvert becomes an unknown. Also, in place of solving the downstream GVF, the slope S_f of the energy line (and slope of the HGL) must be solved. On might use the Darcy–Weisbach equation (and the Colebrook–White equation) for this purpose. However, since the Reynolds number is generally very large for most culvert flows, and f from the Darcy–Weisbach equation is in the wholly rough zone on a Moody diagram, Manning's equation can be used to solve the slope of the energy line for these pipe flows. The Darcy–Weisbach equation gives the slope of the energy line by $S_f = h_f/L = fQ_2^2/(2gA_t^2D)$, or if Manning's equation is used, then $S_f = [nQ/C_uA_t(D/4)^{2/3}]^2$, in which A_t is the full area of the culvert. This slope is utilized to find where the HGL intersects with the pressure head $d = p/\gamma$ on the top of the culvert. Thus, the last two equations above are replaced by

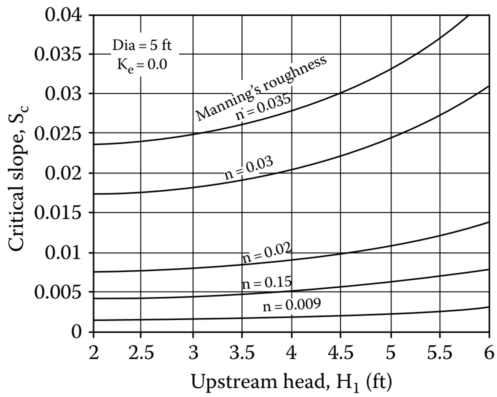
$$F_4 = (Ah_c)_1 + \frac{Q^2}{gA_1} - \left(\frac{1}{2}D\right)A_t - \frac{Q^2}{gA_t}$$

$$F_5 = D + d + (S_0 - S_f)(L - x) - H_2 = 0$$

and these with F_3 solve Y_u, p/γ and x. The assumption in the last equation F_5 is that the velocity head in the culvert is lost as the flow enters a much slower moving flow outside the exit of the culvert.

The determination of the case that may exist under given conditions is often more complex than might appear from examining the above sketch, especially for culverts that gradually close in on their tops, such as circular culverts. This difficulty is compounded by having the Froude number approach zero as the depth in the culvert approaches its diameter. Thus, for example, solutions exist for Y_c and Q based on solving the upstream energy and the critical flow equations for upstream depths H_1 far larger than the diameter D of circular culverts. However, in practice, when H_1 is greater than about 1.2D, the flow into the culvert becomes governed by an orifice-type equation, rather than the critical flow equation. Another complication is that as the upstream depth rises and causes larger flow rates into the culvert, the value of the critical bottom slope S_c increases. How the flow rate and the critical depths, and also how the critical slope of the culvert vary with H_1 for a 5 ft diameter culvert are shown in the figures below. Since S_c increases with H_1, a culvert that has a steep slope for smaller flow rates may become a mild culvert for larger flow rates. When this occurs, the control is also passed from upstream to the downstream water level as the upstream head H_1 increases.





For example, take a 5 ft diameter culvert with a bottom slope of $S_o = 0.02$, $n = 0.03$, and $K_e = 0$. If the upstream head $H_1 = 2$ ft, then the flow is upstream controlled (unless H_2 is above D) since the critical bottom slope is $S_c = 0.0175$, and therefore at the beginning of the culvert, the depth will be $Y_c = 1.47$ ft and $Q = 28.1$ cfs. However, as H_1 rises to the top of the culvert, or $H_1 = 5$ ft, then the critical bottom slope is $S_c = 0.0244$, and the downstream depth controls because the culvert has a mild slope. (You should solve the appropriate equations to verify these values.)

In solving conditions in culverts under upstream control, either the upstream head H_1 or the flow rate Q will be specified along with the downstream head H_2. Whether H_1 or Q is specified, the upstream energy equation F_1 and the critical flow equation F_2 are to be solved. If Q is specified, first solve the critical depth from the critical flow equation F_2, and thereafter the energy equation F_1 can be solved explicitly for H_1. If H_1 is given, it is best to eliminate Q from F_1 and F_2 by substituting Q from F_2 into F_1 and solving the resulting equation. Using the dimensionless depth $Y' = Y_c/D$ and $H' = H_1/D$, this equation is $F = 0.5(1 - \cos\beta) + (1 + K_e)\{\beta - \cos\beta \sin\beta\}/(8 \sin\beta) - H' = 0$, in which $\beta = \cos^{-1}(1 - 2Y')$. After solving β, Q is obtained from $Q = [gD^5(\beta - \cos\beta \sin\beta)^3/(64 \sin\beta)]^{1/2}$.

For Cases II, III, or IV, if the momentum function associated with the S_1 is larger than that associated with the S_2 GVF for all positions up to the entrance, then no longer will a critical depth occur at the entrance, and the flow is no longer controlled by upstream conditions. Rather, the effects of the depth H_2 downstream for the culvert are felt to its entrance and the downstream control occurs. These conditions are not shown on the above sketch, but are handled in the next sections. The computer program CULVERTU.FOR listed below is designed to solve culvert flow problems when upstream control exists. The approach taken in this program is to first solve critical flow conditions at the entrance. It allows for the flow rate Q, or the upstream reservoir water surface elevation H_1 to be given. In the input, this is done by giving a 0 to the one that is unknown. Then starting just below the critical depth, the S_2-GVF profile is solved throughout the length of the culvert. If the given depth H_2 at the downstream end of the culvert is less than this, then Case I occurs, i.e., the flow is supercritical throughout the entire culvert. The conjugate depth to this downstream depth is next computed. Should H_2 be less, then this conjugate depth Case I still occurs with a roller occurring just outside the end of the culvert. The program contains logic to distinguish between Cases II, III, and IV and also determines whether a hydraulic jump (or a modified jump for Case IV) can occur, and if not, prints out a message that downstream control exists. It divides the culvert into 30 stations at which the depths, the areas and the momentum functions are computed. In computing the S_1 GVF profile at these stations, the program terminates when a computed depth becomes less than $(1.5 + S_o)Y_c$, but if the depth before the next station reaches critical depth, then ODESOL may terminate because it cannot achieve the specified tolerance, and stops with the message that the minimum step size has been reached. To make CULVERTU handle all possible situations without failure, considerably more logic needs to be added to the program.

Listing of program CULVERTU.FOR designed to solve problem in which critical flow occurs at the culvert's entrance

```
C Program to solve Flow in Culvert with upstream control.
      PARAMETER (N=3,NP=31)
      EXTERNAL DYX
      CHARACTER*3 CASE
      REAL FMU(NP),FMD(NP),YU(NP),YD(NP)
      REAL W(1,13),Y(1),DY(1),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/D,DH,DS6,D25,So,G,QN,Q2G,H1,H2,A,YC,IOUT
      WRITE(*,*)' GIVE:IOUT,TOL,D,Q,H1,H2,Ke,L,n,So,g'
      READ(*,*) IOUT,TOL,D,Q,H1,H2,FKE,FL,FN,So,G
      IF(Q.EQ.0. .OR. H1.EQ.0.) GO TO 1
      WRITE(*,*)' Both Q and H1 cannot be given. Which is unknown?'
      WRITE(*,*)' Give 1 = Q, or 2 = H1'
      READ(*,*) M
      IF(M.EQ.1) THEN
      Q=0.
      ELSE
      H1=0.
      ENDIF
1     IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      DH=.5*D
      DS6=D*D/6.
      D25=.25*D*D
      AT=.78539816*D*D
      FKE1=(FKE+1.)/(2.*G)
      FKE=(FKE+1.)/8.
      M=0
      IF(Q.EQ. 0.) THEN
      IQ=0
      H1D=H1/D
      Q=.2687*H1D**3.906
      YC=Q**.254
      COSB=1.-2.*YC
      IF(COSB.LT. -1.) COSB=-.98
      BETA=ACOS(COSB)
2     F=.5*(1.-COS(BETA))+FKE*(BETA-COS(BETA)*SIN(BETA))/
     &SIN(BETA)-H1D
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 3
      F1=F
```

```
      BET=BETA
      BETA=1.01*BETA
      GO TO 2
3     DIF=(BETA-BET)*F1/(F-F1)
      BETA=BET-DIF
      IF(ABS(DIF).GT. 1.E-6 .AND. M.LT.30) GO TO 2
      IF(M.GE.30) WRITE(*,*)' Did not comverge. DIF=',DIF,BETA
      Q=(BETA-COS(BETA)*SIN(BETA))**3/(64.*SIN(BETA))
      Q=SQRT(G*Q*D**5)
      YC=DH*(1.-COS(BETA))
      Q2G=Q*Q/G
      ELSE
      IQ=1
      Q2G=Q*Q/G
      BETA=1.5
      Q2GD=D*Q2G
4     F=Q2GD*SIN(BETA)-(D25*(BETA-COS(BETA)*SIN(BETA)))**3
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 5
      F1=F
      BET=BETA
      BETA=1.01*BETA
      GO TO 4
5     DIF=(BETA-BET)*F1/(F-F1)
      BETA=BET-DIF
      IF(ABS(DIF).GT.1.E-6 .AND. M.LT.30) GO TO 4
      YC=DH*(1.-COS(BETA))
      H1=YC+4.*FKE*Q2G/(D25*(BETA-COS(BETA)*SIN(BETA)))**2
      ENDIF
      WRITE(IOUT,100) Q,YC,H1,H2,D,FL,FN,So
100   FORMAT(' Q=',F8.2,', Yc=',F8.3,', H1=',F8.3,', H2=',F8.3,/
     &' D=',F7.1,', L=',F8.0, ', n=',F8.4,', So=',F8.6)
      FMT=DH*AT+Q2G/AT
      YCP=(1.15+SO)*YC
      HMIN=.000001
      DELX=FL/FLOAT(NP-1)
C Solves S2-GVF from entrance to end of culvert
      XX=0.
      XZ=DELX
      H11=.05
      Y(1)=.95*YC
      YU(1)=Y(1)
      FMU(1)=FMOM(Y(1))
      WRITE(IOUT,115) 'S2',0.,YU(1),A,FMU(1)
115   FORMAT(/,A3,' - GVF Profile',/,4F10.3)
118   FORMAT(4F10.3)
      DO 10 I=2,NP
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H11,HMIN,1,XP,YP,W,DYX)
      YU(I)=Y(1)
      FMU(I)=FMOM(Y(1))
      WRITE(IOUT,118) XZ,YU(I),A,FMU(I)
```

```
      XX=XZ
10    XZ=XZ+DELX
      Y2=2.*YC-YU(NP)
      IF(Y2.GT.D) Y2=.9*D
      FMOM1=FMOM(YU(NP))
C Solves Conjugate Depth at end of culvert
      M=0
12    F=FMOM1-FMOM(Y2)
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 13
      F1=F
      Y22=Y2
      Y2=1.01*Y2
      GO TO 12
13    IF(ABS(F-F1).LT.1.E-20) THEN
      DIF=0.
      ELSE
      DIF=F1*(Y2-Y22)/(F-F1)
      ENDIF
      Y2=Y22-DIF
      IF(ABS(DIF).GT.1.E-6 .AND. M.LT.30) GO TO 12
      IF(M.GE.30) WRITE(*,*)' Did not converge for conj. Y',DIF,Y2
      IF(H2.GT.Y2) THEN
      IF(Y2.GT.D) WRITE(IOUT,106) Y2
106   FORMAT(' Conjugate depth at end is above top',' of culvert',
     &F8.2)
      GO TO 15
      ENDIF
      WRITE(IOUT,110) H1,H2,Q,YC,YU(NP),Y2
110   FORMAT(' Case I occurs with S2-GVF thru entire length'
     &' of culvert',/,' H1 =',F8.2,' H2 =',F8.2,' Q =',F8.2,
     &'Yc =',F8.2,' Depth at end =',F8.2,/,' Conjugate
     &depth =',F8.2)
      IF(H2.LE.Y2) WRITE(IOUT,120) Y2-YU(NP)
120   FORMAT(' Roller exists at end of culvert with',' height =',
     &F8.2)
      STOP
15    IF(H2.GT.D) GO TO 40
C    Case II - Hyd. Jump in culvert
      XX=FL
      H11=-1.
      XZ=XX-DELX
      IB=1
      Y(1)=H2
      YD(NP)=H2
      FMD(NP)=FMOM(Y(1))
      WRITE(IOUT,115)'S1',FL,YD(NP),A,FMD(NP)
      IE=NP
      CASE='II '
C Solves S1-GVF profile
19    DO 20 I=IE-1,1,-1
```

```
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H11,HMIN,1,XP,YP,W,DYX)
      YD(I)=Y(1)
      FMD(I)=FMOM(Y(1))
      WRITE(IOUT,118) XZ,YD(I),A,FMD(I)
      IF(Y(1).LT.YCP) THEN
      IB=I
      GO TO 22
      ENDIF
      XX=XZ
20    XZ=XZ-DELX
22    IF(IB.EQ.1 .AND. FMD(1).GT.FMU(1)) THEN
      WRITE(IOUT,160) H1,H2,Q
160   FORMAT(' Effects extend to entrance so downstream', control
     &occurs.',/,' Change specifications.',/,' H1=',F7.2,' H2=',
     &F7.2,' Q=',F8.2)
      STOP
      ENDIF
      DO 25 I=IB,NP
      IF(FMU(I).LT.FMD(I)) GO TO 26
25    CONTINUE
26    IP=I-1
      FAC=(FMU(IP)-FMD(IP))/(FMD(IP+1)-FMD(IP)-FMU(IP+1)+FMU(IP))
      XX=DELX*(FLOAT(IP-1)+FAC)
      WRITE(IOUT,130) CASE,Q,YC,YU(IP)+FAC*(YU(IP+1)-YU(IP)),
     &YD(IP)+FAC*(YD(IP+1)-YD(IP)),XX
130   FORMAT(' Case ',A3,' Jump occurs within culvert:',
     &' Q =',F8.2,' Yc =',F7.2,/,' Depth upst. jump=',F6.2,
     &' Depth downst. jump =',F6.2,' Jump position x=',F7.1)
      STOP
30    XX=(H2-D)/(SO-SF)
      WRITE(IOUT,150) XX
150   FORMAT(' Case III - Culvert full at downstr. end',
     &' for a distance of',F8.2)
      IE=NP-XX/DELX-1.9
32    Y(1)=.98*D
      YD(IE)=D
      FMD(IE)=FMOM(Y(1))
      XX=DELX*FLOAT(IE-1)
      XZ=XX-DELX
      H11=-1.
      IB=1
      WRITE(IOUT,115)'S1',XX,Y(1),A,FMD(IE)
      CASE='III'
      GO TO 19
40    SF=(FN*Q/(CC*AT*(.25*D)**.6666667))**2
      XX=(H2-D)/(SO-SF)
      IF(XX.GE.FL) THEN
      WRITE(IOUT,155) (XX-FL)*(SO-SF)+FKE1*(Q/AT)**2
155   FORMAT(' Case V - Downstream H2 causes depth',' at
     &entrance of',F9.2,/,' Flow is downstream controlled.')
      STOP
```

```
      ENDIF
      DO 41 I=NP,1,-1
      IF(FMU(I).LT.FMT) GO TO 42
41    CONTINUE
      WRITE(IOUT,160) H1,H2,Q
      STOP
42    IE=I
      IF(IE.EQ.NP) GO TO 30
      X24L=DELX*(FLOAT(IE-1)+(FMT-FMU(IE))/(FMU(IE+1)-FMU(IE)))
      H24L=D+(SO-SF)*(FL-X24L)
      WRITE(IOUT,180) X24L,FL-X24L,H24L
180   FORMAT(' Start of Case IV at x =',F8.2,' (from end=',F8.2,')
     &Downstr. H2 required=',F8.2)
      DO 50 I=IE-1,NP
      PHEAD=H2-D-(SO-SF)*(FL-DELX*FLOAT(I-1))
C     WRITE(IOUT,*) I,PHEAD,PHEAD*AT+FMT,FMU(I)
      IF(PHEAD.LT.0.) GO TO 50
      IF(PHEAD*AT+FMT.GT.FMU(I)) GO TO 50
      WRITE(IOUT,140) DELX*FLOAT(I-1),PHEAD,FMU(I)
140   FORMAT(' Case IV, H. Jump causes pipe',' flow downstream',
     &'from x=',F8.2,/,' The pressure head on top of culvert=',
     &F8.2,' Mom=',F10.1)
      STOP
50    CONTINUE
      WRITE(IOUT,170)
170   FORMAT(' Did not find location where Mu=Mt+AtPh')
      GO TO 32
      END
      FUNCTION FMOM(Y)
      COMMON /TRAS/D,DH,DS6,D25,So,G,QN,Q2G,H1,H2,A,YC,IOUT
      COSB=1.-Y/DH
      IF(COSB.GT.1.) COSB=1.
      IF(COSB.LT.-1.) COSB=-1.
      BETA=ACOS(COSB)
      SINB=SIN(BETA)
      A=D25*(BETA-COSB*SINB)
      FMOM=DH*(DS6*SINB**3-A*COSB)+Q2G/A
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1)
      COMMON /TRAS/D,DH,DS6,D25,So,G,QN,Q2G,H1,H2,A,YC,IOUT
      YY=Y(1)
      IF(YY.LT. 0.05) YY=.05
      COSB=1.-YY/DH
      IF(COSB.LT. -1.) COSB=-.995
      BETA=ACOS(COSB)
      SINB=SIN(BETA)
      P=D*BETA
      A=D25*(BETA-COSB*SINB)
      SoSf=So-QN*((P/A)**.666666667/A)**2
```

```
FRM1=1.-Q2G*D*SINB/A**3
DY(1)=SoSf/FRM1
RETURN
END
```

EXAMPLE PROBLEM 4.21

A circular culvert with a diameter D = 5 ft is 80 ft long and has a Manning's roughness coefficient n = 0.013. Its bottom slope is $S_o = 0.01$ and its entrance loss coefficient is $K_e = 0.2$. If the upstream water surface elevation is $H_1 = 4.5$ ft, determine the discharge through the culvert and the water surface profile through it if the downstream water surface elevations are (a) $H_2 = 2$ ft, (b) $H_2 = 3.5$ ft, (c) $H_2 = 4.0$ ft, (d) $H_2 = 4.4$ ft, (e) $H_2 = 4.5$ ft, and (f) $H_2 = 5.5$ ft.

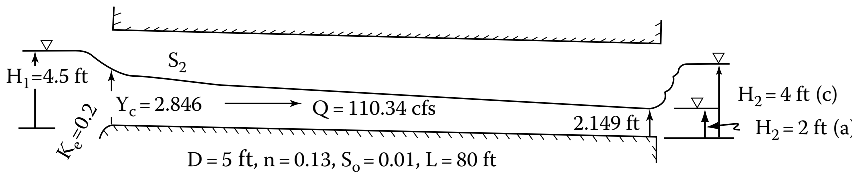
Solution

The input to CULVERTU to solve the (a) part of this problem is 3 .00001 5 0 4.5 2 .2 80 .013 .01 32.2. The solution shows that the S_2 GVF exists through the entire 80 ft length of culvert with the depth at its end equal to 2.436 ft, or slightly above the downstream reservoir water surface elevation. The flow rate is Q = 110.3 cfs, and is the solution of the critical flow and the energy equation at the entrance ($Y_c = 3.00$ ft). The output from CULVERTU consists of the following (with part of the GVF omitted):

```
Q = 110.34, Yc= 2.996, H1= 4.500, H2 = 2.000
D= 5.0, L = 80.0, n = 0.0130, So = 0.010000
```

S_2—GVF Profile

```
  .000   2.846   11.544   46.868
 2.667   2.794   11.283   47.025
 5.333   2.755   11.092   47.172
   .       .        .        .
77.333   2.441    9.524   49.546
80.000   2.436    9.500   49.602
```



Case I occurs with an S_2—GVF through the entire length of the culvert

H_1 = 4.50 H_2 = 2.00 Q = 110.34 Yc = 3.00 Depth at end = 2.44

Conjugate depth = 3.65

A roller exists at the end of culvert with height = 1.22

(b) The input for part (b) is the same as that above with the value of $H_2 = 3.5$ (rather than 2). The solution is identical since the specified value of H_2 is less than the conjugate depth of 4.10 ft to the downstream depth of 2.436 ft. (c) The input for part (c) is again identical to that above with the exception that H_2 is given a valve of 4.0. The solution shows that a hydraulic jump occurs at a position 45.8 ft downstream from the entrance with the depth upstream from the jump $Y_u = 2.52$ ft and the downstream depth $Y_d = 3.54$ ft. (d) For this part, H_2 is given as 4.4. Now, the solution shows the jump at x = 5.3 ft, or close to the entrance of the culvert. The output from CULVERTU consists of

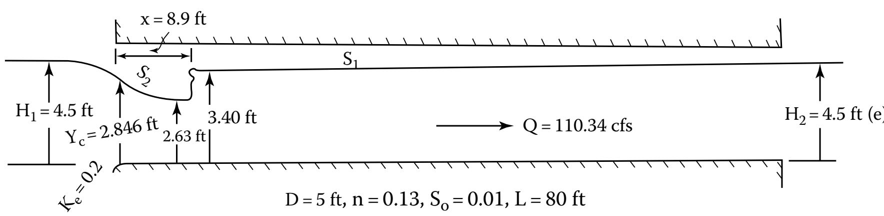
```
Q= 110.34, Yc=  2.996, H1=  4.500, H2=  4.400
D=   5.0, L=    80.0, n=  0.0130, So= 0.010000
```

S_2—GVF Profile

```
   .000  2.846 11.544 46.868
  2.667  2.794 11.283 47.025
  5.333  2.755 11.092 47.172
  8.000  2.725 10.939 47.310
 10.667  2.699 10.810 47.441
    .      .      .      .
 80.000  2.436  9.500 49.602
```

S_1—GVF Profile

```
 80.000  4.400 18.300 58.290
    .      .      .      .
 16.000  3.637 15.301 49.468
 13.333  3.598 15.126 49.157
 10.667  3.557 14.942 48.853
  8.000  3.515 14.750 48.556
  5.333  3.471 14.548 48.266
```



A Case II jump occurs within the culvert: Q = 110.34 Y_c = 3.00
Depth upst. jump= 2.76 Depth downst. jump = 3.39 Jump position x= 5.3

Notice, the flow rate and the depth at the culvert's entrance has not changed for these four values of H_2.

(e) The solution for the (e) part with the downstream water surface elevation H_2 = 4.5 ft, i.e., the same as H_1, follows. It shows that the S_1 GVF flow extends to the entrance causing a depth of Y_1 = 3.542 ft at the entrance, or a value greater than the critical depth Y_c = 2.996 ft; therefore the flow is downstream controlled, and will be handled by the procedures described in the next section.

(f) When H_2 = 5.5 ft, then the downstream depth is above the top of the culvert. The solution given below shows that the downstream portion of the culvert has a pipe flow in it a distance of 60.94 ft, and that the S_1 GVF profile that begins at this position extends to the entrance, thus the flow is downstream controlled, and the problem falls into the category handled in the next sections. That downstream control exists should be obvious, in fact the flow would be reversed with this downstream head since the downstream water surface is above the upstream water surface by 0.2 ft, i.e., a level water surface occurs when H_2 = 5.3 ft. Before a critical flow is possible at the entrance, the slope downstream H_2 must be small enough to allow for the frictional loss with the culvert flowing full, and the entrance loss, or $H_{2max} = 5.3 - S_fL - K_eV^2/(2g) = 5.3 - .0017948(80) - .2(.4904) = 5.06$ ft. The output from CULVERTU consists of

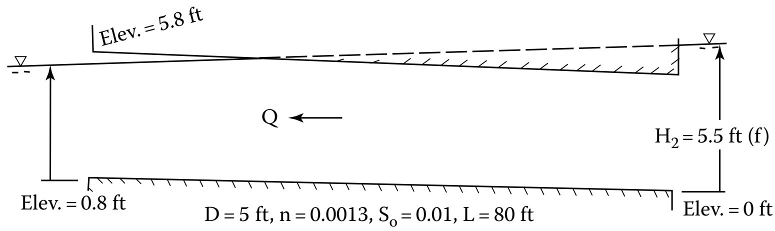
```
Q= 110.34, Yc=  2.996, H1=  4.500, H2=  5.500
D=   5.0, L=   80.0, n=  0.0130, So= 0.010000
```

S_2—GVF Profile

```
   .000   2.846  11.544  46.868
    .       .       .       .
 77.333   2.441   9.524  49.546
 80.000   2.436   9.500  49.602
```

Case III—Culvert full at downstr. end for a distance of 60.94
S_1—GVF Profile

```
13.333  4.900 19.541 66.475
10.667  4.876 19.505 66.038
 8.000  4.851 19.466 65.601
 5.333  4.827 19.422 65.164
 2.667  4.802 19.375 64.728
 0.000  4.777 19.324 64.293
```



Effects extend to entrance so downstream control occurs.

Change specifications.
$H_1 = 4.50$ $H_2 = 5.50$ $Q = 110.34$

EXAMPLE PROBLEM 4.22

Repeat the previous problem, however rather than specifying the upstream head H_1, specify the flow rate to be Q = 100 cfs. Also increase the length of the culvert to 150 ft.

Solution

The input to CULVERTU will now give a 0 to H_1 and 100 to Q so it consists of 3 .00001 5 100 0. 2 .2 .013 .01 32.2. The results are very similar to those of the previous problem with Case I occurring for parts (a) and (b). Now $H_1 = 4.245$ ft, $Y_c = 2.85$ ft, and the downstream depth is equal to 2.223 ft for parts (a) and (b). For part (c), in which the water surface elevations $H_2 = 4.0$ ft, Case II occurs with the hydraulic jump at position x = 71.0 ft. With $H_2 = 5.5$ ft for part (f), the culvert is full to x = 58.7 ft, and the momentum function associated with the S_1 GVF that starts at this position is always greater than that associated with the supercritical flow along the S_2 GVF program that begins just below Y_c at the culvert's entrance, and therefore again downstream control exists, and critical flow will not take place at the entrance.

EXAMPLE PROBLEM 4.23

Increase the bottom slope of Example Problem 4.21 to $S_o = 0.018$ and repeat the five parts of that problem.

Solution

The result for parts (a) and (b) show that Case I occurs as in Example Problem 4.21, with the downstream depth now decreased to 2.175 ft, and the depth conjugate to this equal to 4.05 ft. (c) Now with this greater bottom slope, when $H_2 = 4.0$ ft, the S_2 extends through the 80 ft length of the culvert as in parts (a) and (b) since the conjugate depth is 4.05 ft. For part (d) in which $H_2 = 4.4$ ft, Case II occurs with the hydraulic jump occurring at position x = 59.4 ft, with the depths upstream and downstream therefrom as $Y_u = 2.24$ ft and $Y_d = 3.95$ ft, respectively. (e) For this part in which $H_2 = H_1 = 4.5$ ft, a jump occurs at x = 3.91 ft, but the flow rate is still Q = 110.34 cfs as in the previous parts, and in Example Problem 4.21. For part (f) with $H_2 = 5.5$ ft, the culvert flows full for a distance of 30.8 ft at its lower end, but the effect of the S_1 GVF that begins at this position extends to the culvert's entrance causing the flow to be downstream controlled as in Example Problem 4.21.

EXAMPLE PROBLEM 4.24

A 5 ft diameter culvert that is 150 ft long and a Manning's n = 0.013 has a bottom slope $S_o = 0.06$. If the upstream water depth is $H_1 = 4.5$ ft and the downstream depth is $H_2 = 7.0$ ft, determine the flow rate and the profile through the culvert.

Solution

The flow rate will be the same as in Example Problem 4.21 because D and H_1 are the same if critical flow conditions govern. The input to CULVERTU consists of 3 .00001 5 0 4.5 7 .2 150 .013 .06 32.2 and the solution consists of

```
Q= 110.34, Yc=  2.996, H1=  4.500, H2=  7.000
D=  5.0, L=  150.0, n=  0.0130, So= 0.060000
```

S_2—GVF Profile

.000	2.846	11.544	46.868
5.000	2.435	9.495	49.614
10.000	2.269	8.663	51.922
15.000	2.157	8.109	53.972
20.000	2.073	7.692	55.828
25.000	2.006	7.362	57.530
30.000	1.950	7.089	59.101
35.000	1.903	6.859	60.559
40.000	1.862	6.662	61.917
45.000	1.826	6.490	63.188
50.000	1.795	6.338	64.378
55.000	1.767	6.204	65.496
60.000	1.741	6.083	66.548
65.000	1.719	5.975	67.539
70.000	1.698	5.877	68.473
75.000	1.679	5.787	69.356
80.000	1.662	5.706	70.189
85.000	1.646	5.631	70.978
90.000	1.631	5.563	71.724
95.000	1.618	5.499	72.430
100.000	1.605	5.441	73.100
105.000	1.594	5.387	73.734
110.000	1.583	5.337	74.336
115.000	1.573	5.290	74.907
120.000	1.563	5.247	75.449
125.000	1.555	5.206	75.963
130.000	1.547	5.169	76.451
135.000	1.539	5.133	76.915
140.000	1.532	5.100	77.356
145.000	1.525	5.069	77.774
150.000	1.519	5.040	78.172

Conjugate depth at end is above top of culvert 5.70

Start of Case IV at x = 69.30 (from end = 80.70) Downstr. H2 required = 9.70

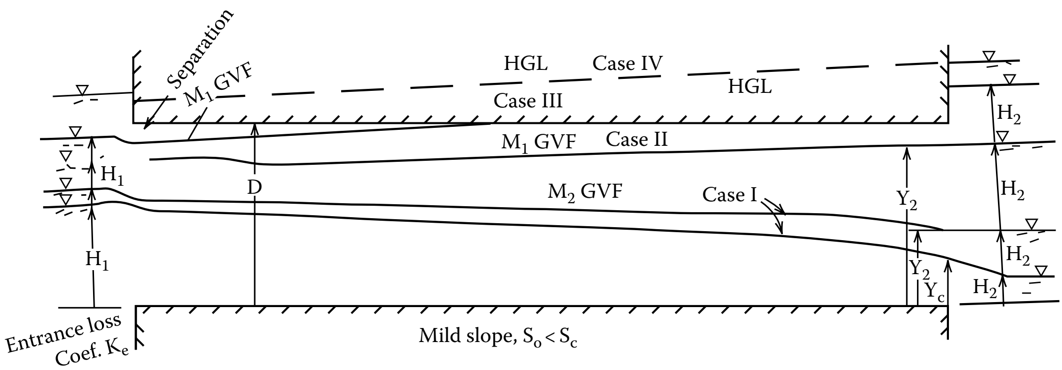
Case IV, H. Jump causes pipe flow downstream from x = 120.00

The pressure head on top of culvert = 0.25 Mom = 75.4

Notice from this solution that it takes a downstream water depth of 5.70 ft, which is the downstream conjugate depth, before the downstream water can cause a pipe flow. Case IV occurs in this problem with a modified hydraulic jump taking place at 120 ft, or 30 ft from the culvert's end. As this modified jump occurs, it creates a pressure head of 0.25 ft on the top of the culvert.

4.11.2 Solutions When Downstream Control Exists

When the culvert's flow is controlled by downstream conditions, the possible flow profiles are not as numerous as when an upstream control exists. A GVF will begin at the downstream end of the culvert with a depth equal to the downstream water depth, or $Y_2 = H_2$. There are four possible conditions as shown on the sketch below. (1) If depth Y_2 is above the normal depth



associated with the flow rate that occurs, and less than the height of the culvert, then an M_1 GVF exits, that will likely reach to the culvert's entrance, unless the culvert is very long and reduces the flow from that obtained by solving the upstream energy equation simultaneously with Manning's equation, i.e., the normal depth Y_o and Q_o. This possibility is shown as **Case II** on the sketch. (2) If H_2 is below the normal depth, then an M_2 GVF will exist in the culvert that will likely cause a flow rate larger than Q_o and a depth at the entrance less than Y_o. This GVF will have a beginning depth at the exit of the culvert equal to H_2 if $H_2 < Y_o$ and $H_2 > Y_c$. If H_2 is less than Y_c, then the starting depth for the M_2 GVF is Y_c (i.e., 5% above this to prevent numerical difficulties). This is shown as **Case I** on the sketch. (3) If H_2 is larger than D, then a pipe flow will exit, as least over the downstream portion of the culvert. This is shown as **Case III**. If $(S_o - S_f)L + D < H_2$ then the entire culvert will flow full, in which S_f is the slope of the energy line (or HGL) computed for the pipe flow for the flow rate that is occurring. This is shown as **Case IV**.

In solving problems with downstream control, one might wish to specify the flow rate and the downstream depth. With Q given, the solution proceeds by solving the GVF starting at the downstream depth H_2, and upon obtaining Y_1 at the beginning of the culvert, solve the upstream head needed, or $H_1 = Y_1 + (1 + K_e)Q^2/(2gA^2)$. If both the upstream head H_1 and the downstream head H_2 are given, then the flow rate must be determined. The flow rate Q, as well as the upstream depth Y_1, are obtained by simultaneously solving the upstream energy equation and the GVF through the entire length of the culvert.

$$F_1 = H_1 - Y_1 - \frac{(1+K_e)Q^2}{2gA^2} = 0$$

$$F_2 = Y_{1ode}(H_2) - Y_1 = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f}{1 - F_r^2}$$

If the downstream head H_2 is equal to or less than the critical depth, then Y_c becomes another unknown, and the starting value F_2 for the GVF equation is Y_c rather than H_2, as shown above. If this occurs, the critical flow equation must be added to the above two equations to give the following third equation:

$$F_3 = Q^2T/g - A^3 = 0$$

to solve Q, Y_1, and Y_c.

When H_2 is larger than D, then the slope of the HGL, S_f, needs to be computed from either the Darcy–Weisbach equation or Manning's equation, and from this the position, x determines where the flow changes from the open channel to the pipe flow, i.e., from $x = (H_2 - D)/(S_o - S_f)$, will be. If Q is specified, then this position can be determined once, and thereafter the M_1 GVF is solved at the beginning of the culvert. If H_1 is specified, this x becomes a third unknown that must be solved simultaneously with Q and Y_1, since S_f depends upon Q. Programs CULVERTD.FOR and CULVERD1.FOR are designed to obtain these solutions; CULVERTD if both H_1 and H_2 are given, and CULVERD1 if H_2 and Q are given. Program CULVERTD allows you to provide guesses for the uniform depth and critical depth, and then for the actual flow rate that will occur and the corresponding depth at the entrance of the culvert, or to use default initial values for the Newton solution. The default values are generally good enough if the uniform flow is subcritical, i.e., the slope of the culvert is mild, but not good because the values will start the Newton method looking for the subcritical root. Therefore, if the culvert has a steep slope it is best to supply guesses for the above variables. To supply such guesses, the logical unit for the output should be given a negative value.

The input to both programs should be identified easily from the variables since they correspond to variables used in the equations and in the previous programs. After solving the problem, CULVERTD repeats the last GVF profile but uses an increment DX = Δx on which to write out the profile position and depth. These programs are designed to use Manning's equation to compute S_f if e = 0, should H_2 be greater than D. To use the Darcy–Weisbach equation to compute S_f gives e (the value of the equivalent sand roughness) in basic units (ft for ES units and m for SI units.) as the diameter is also given in. After reading the input data, CULVERTD first solves the upstream energy and Manning's equations simultaneously. This is done by substituting Q from Manning's equation into the energy equation to get

$$F = H_1 - Y_1 - (1+K_e)\frac{C_u^2 S_o}{2gn^2}\left(\frac{A}{P}\right)^{4/3} = 0$$

and after obtaining Y_1 (which is also the normal depth Y_o) from this equation using the Newton method, it solves Manning's equation for Q_o. This solution starts 2 lines above label 1 and ends with the WRITE with FORMAT 112. These values for Q_o and Y_{o1} provide guidance in initializing the Newton solution of the GVF and the entrance energy equations. Next, based on Q_o, the critical depth equation $F = Q^2T/g - A^3 = 0$ is solved by calling on the subroutine CRIT. If H_2 is less than Y_c, then CULVERTD makes the decision to solve the three equations above, simultaneously, rather than just the first two. The subroutine FUN is designed to provide the equations evaluated for the current iterative values of the unknowns in implementing the Newton solution in the main program. Array X contains the unknowns as follows: Q = X(1), Y_1 = X(2), and Y_c = X(3) (should $H_2 > D$ then x = X(3)).

Program CULVERD1 is simpler because H_2 and Q are specified, and therefore a simultaneous solution of F_1 and F_2 (and possibly F_3) is not needed. It first solves the GVF equation F_2 starting at the downstream end, and with the upstream depth Y_1 now known, solves the energy equation F_1 for the upstream head H_1. It also calls on the subroutine CRIT to obtain Y_c, so the proper decision can be made whether to give H_2 or Y_c as the beginning depth at the downstream end in solving the GVF equation. Should H_2 be greater than D, then the position x, where the pipe flow begins, is first solved and thereafter the M_1 GVF is solved upstream therefrom to the culvert's beginning, giving Y_1, from which H_1 is computed.

Program CULVERTD.FOR

```
C Program to solve flow in culverts with downstream control
      PARAMETER (N=3)
      LOGICAL BCDWN,GUESS/.FALSE./
      EXTERNAL DYX
      INTEGER*2 INDX(N)
      REAL FF(N),FF1(N),DJ(N,N)
      COMMON /ODE/ W(1,13),YODE(1),DY(1),XP(1),YP(1,1),H11
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/D,DH,D25,So,FL,G,TOL,QN,X(3),Q2G,H1,H2,FN,
     &FKE,eD,VIS,E,FL1,Sf,IOUT,NEQ,BCDWN,SSF,CC
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,D,H1,H2,Ke,L,n,So,g,DX,e'
      READ(*,*) IOUT,TOL,ERR,D,H1,H2,FKE,FL,FN,So,G,DXI,e
      WRITE(IOUT,203) D,H1,H2,FKE,FL,FN,SO,e
203   FORMAT(' D=',F7.2,' H1=',F7.2,' H2=',F7.2,' Ke=',F7.3,
     &' L=',F8.1,/,' n=',F7.4,' So=',F9.6,' e=',F8.5)
      SSF=7.5
      IF(IOUT.LT.0) THEN
      GUESS=.TRUE.
      IOUT=IABS(IOUT)
      ENDIF
      IF(H2.GT.H1+So*FL) THEN
      WRITE(*,111) H2,H1,H2-So*FL-H1
111   FORMAT(' Downstream head',F8.2,' has a water surface',
     & elev. above upstream head',F8.2,/' This difference of',F8.3,'
     &Will cause reverse flow.')
      STOP
      ENDIF
      X(3)=0.
      FL1=FL
      BCDWN=.FALSE.
      IF(G.LT.30.) THEN
      CC=1.
      VIS=1.31E-6
      ELSE
      CC=1.486
      VIS=1.317E-5
      ENDIF
      DH=.5*D
      D25=.25*D*D
      eD=e/D
      AT=.78539816*D*D
      FKE=(FKE+1.)/2.
      CMA=FKE*So*(CC/FN)**2/G
      IF(GUESS) THEN
      WRITE(*,*)' Give est. for uniform & critical depths'
      READ(*,*) Y,YB
      ELSE
      Y=.92*H1
      ENDIF
      M=0
```

```
      COSB=1.-Y/DH
      BETA=ACOS(COSB)
1     F=H1-DH*(1.-COS(BETA))-CMA*(D25*(BETA-SIN(BETA)*
     &COS(BETA))/(D*BETA))**1.3333333
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 2
      F1=F
      BET=BETA
      BETA=1.01*BETA
      GO TO 1
2     DIF=(BETA-BET)*F1/(F-F1)
      BETA=BET-DIF
      IF(ABS(DIF).GT. 1.E-5 .AND. M.LT.30) GO TO 1
      Y=DH*(1.-COS(BETA))
      A=D25*(BETA-COS(BETA)*SIN(BETA))
      Q=CC*A/FN*SQRT(So)*(A/(D*BETA))**.6666667
      QN=(FN*Q/CC)**2
      WRITE(IOUT,112) Q,Y,A
112   FORMAT(' Uniform Q =',F9.2,' Y =',F8.3,' Area =',F8.2)
      Q2G=Q*Q/G
      IF(.NOT.GUESS) YB=.6*Y
      CALL CRIT(YB,YC)
      WRITE(IOUT,113) YC
113   FORMAT(' Corresponding Critical Depth, Yc =',F8.3)
      IF(GUESS) THEN
      WRITE(*,*)' Give guess of Q and Y at entrance'
      READ(*,*) X(1),X(2)
      Q=X(1)
      ELSE IF(Y+So*FL.GT.H2) THEN
      X(1)=Q
      X(2)=Y
      ELSE
      X(1)=.6*Q
      X(2)=1.01*Y
      ENDIF
      Q2G=Q*Q/G
      NEQ=2
      IF(H2.GT.YC) GO TO 8
      NEQ=3
      X(3)=YC
      BCDWN=.TRUE.
8     M=0
10    CALL FUN(FF)
      IF(FL1.LT.FL) NEQ=3
      DO 30 J=1,NEQ
      DX=.005*X(J)
      X(J)=X(J)+DX
      CALL FUN(FF1)
      DO 20 I=1,NEQ
20    DJ(I,J)=(FF1(I)-FF(I))/DX
30    X(J)=X(J)-DX
```

```
      CALL SOLVEQ(NEQ,1,N,DJ,FF,1,DD,INDX)
      SUM=0.
      DO 40 I=1,NEQ
      IF(FF(I).GT. 0.8*X(I)) FF(I)=.5*X(I)
      X(I)=X(I)-FF(I)
40    SUM=SUM+ABS(FF(I))
      M=M+1
      IF(M.LT.40 .AND. SUM.GT.ERR) GO TO 10
      IF(FL1.LT.FL) THEN
      FL1=X(3)+.025*D*(So-Sf)
      WRITE(IOUT,114) Sf,FL1
114   FORMAT(' Slope HGL for submerged culvert=',E10.4,
     &' Intersection with top at x=',F8.2)
      ENDIF
      WRITE(IOUT,100) X(1),X(2)
100   FORMAT(' Flow rate =',F10.2,', Upstream Depth =',F10.3)
      IF(BCDWN) WRITE(IOUT,101) X(3)
101   FORMAT(' Critical Depth at Downstream End =',F10.3)
      WRITE(IOUT,115)
115   FORMAT(' Solution to GVF',/,'   x     Y',/,1X,19('-'))
      XX=FL1
      IF(BCDWN) THEN
      YODE(1)=1.05*X(3)
      ELSE
      IF(FL1.LT.FL) THEN
      YODE(1)=.975*D
      ELSE
      YODE(1)=H2
      ENDIF
      ENDIF
      WRITE(IOUT,120) FL1,YODE
      DO 50 I=1,IFIX(FL1/DXI+.95)
      X2=XX-DXI
      IF(X2.LT.0.) X2=0.
      CALL ODESOL(YODE,DY,1,XX,X2,TOL,H11,.000001,1,XP,YP,W,DYX)
      WRITE(IOUT,120) X2,YODE
120   FORMAT(F10.2,F10.3)
50    XX=X2
      END
      SUBROUTINE CRIT(YB,YC)
      LOGICAL BCDWN
      COMMON /TRAS/D,DH,D25,So,FL,G,TOL,QN,X(3),Q2G,H1,H2,FN,FKE,
     &eD,VIS,E,FL1,Sf,IOUT,NEQ,BCDWN,SSF,CC
      M=0
      COSB=1.-YB/DH
      BETA=ACOS(COSB)
1     F=Q2G*D*SIN(BETA)-(D25*(BETA-COS(BETA)*SIN(BETA)))**3
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 2
      F1=F
      BET=BETA
```

```
      BETA=1.01*BETA
      GO TO 1
2     DIF=(BETA-BET)*F1/(F-F1)
      BETA=BET-DIF
      IF(ABS(DIF).GT.1.E-5 .AND. M.LT.30) GO TO 1
      YC=DH*(1.-COS(BETA))
      RETURN
      END
      SUBROUTINE FUN(F)
      LOGICAL BCDWN
      EXTERNAL DYX
      REAL F(3)
      COMMON /ODE/ W(1,13),Y(1),DY(1),XP(1),YP(1,1),H11
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/D,DH,D25,So,FL,G,TOL,QN,X(3),Q2G,H1,H2,FN,FKE,
     &eD,VIS,E,FL1,Sf,IOUT,NEQ,BCDWN,SSF,CC
      H11=-1.
      Q2G=X(1)**2/G
      QN=(FN*X(1)/CC)**2
      COSB=1.-X(2)/DH
      IF(COSB.LT. -1.) COSB=-.98
      BETA=ACOS(COSB)
      A=D25*(BETA-COSB*SIN(BETA))
      F(1)=H1-X(2)-FKE*Q2G/A**2
      IF(BCDWN) THEN
      Y(1)=1.05*X(3)
      H11=-.05
      ELSE
      IF(H2.LT.D) THEN
      Y(1)=H2
      ELSE
      IF(e.LT.1.E-12) THEN
      Sf=10.293591*QN/D**5.3333333
      ELSE
3     SSF1=SSF
      SSF=1.14-2.*ALOG(eD+7.3434728*VIS*D/(X(1)*SSF1))
      IF(ABS(SSF-SSF1).GT.1.E-6) GO TO 3
      Sf=Q2G/(2.*(SSF*.78539816*D*D)**2)
      ENDIF
      IF(X(3).EQ.0.) THEN
      FL1=FL-(H2-D)/(So-Sf)
      X(3)=FL1-.025*D*(So-Sf)
      ELSE
      F(3)=FL-(H2-.975*D)/(So-Sf)-X(3)
      ENDIF
      Y(1)=.975*D
      IF(X(3).LT.0. .OR. (Sf-So).GT.0.) THEN
      F(2)=H1-H2-FL*(Sf-So)
      WRITE(IOUT,112) H1,H1+1.6211389*FKE*Q2G/D**4
112   FORMAT(' Submerged to entrance, HGL at beg. =',F8.2,
     &' H1=',F8.2)
```

```
      RETURN
      ENDIF
      ENDIF
      ENDIF
      CALL ODESOL(Y,DY,1,FL1,0.,TOL,H11,.00001,1,XP,YP,W,DYX)
      F(2)=X(2)-Y(1)
      IF(NEQ.LT.3 .OR.FL1.LT.FL) RETURN
      BETA=ACOS(1.-X(3)/DH)
      A=D25*(BETA-COS(BETA)*SIN(BETA))
      F(3)=Q2G*D*SIN(BETA)-A**3
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      LOGICAL BCDWN
      REAL Y(1),DY(1)
      COMMON /TRAS/D,DH,D25,So,FL,G,TOL,QN,X(3),Q2G,H1,H2,FN,FKE,
     &eD,VIS,E,FL1,Sf,IOUT,NEQ,BCDWN,SSF,CC
      YY=Y(1)
      IF(YY.LT. 0.05) YY=.05
      COSB=1.-YY/DH
      IF(COSB.LT. -1.) COSB=-.995
      BETA=ACOS(COSB)
      SINB=SIN(BETA)
      P=D*BETA
      A=D25*(BETA-COSB*SINB)
      SoSf=So-QN*((P/A)**.666666667/A)**2
      FRM1=1.-Q2G*D*SINB/A**3
      DY(1)=SoSf/FRM1
      RETURN
      END
```

Program CULVERD1.FOR

```
C Program to solve flow in culverts with downstream
C     control-Q (given)
      EXTERNAL DYX
      REAL Y(1),DY(1),W(1,13),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/D,DH,D25,So,Q2G,QN
      WRITE(*,*)' GIVE:IOUT,TOL,D,Q,H2,Ke,L,n,So,g,DX,e'
      READ(*,*) IOUT,TOL,D,Q,H2,FKE,FL,FN,So,G,DX,e
C If e=0, then Mannings equation used to compute Sf for pipe flow.
      FL1=FL
      IF(G.LT.30.) THEN
      CC=1.
      VIS=1.31E-6
      ELSE
      CC=1.486
      VIS=1.317E-5
      ENDIF
      DH=.5*D
      eD=e/D
```

```
      SSF=8.
      D25=.25*D*D
      QN=(FN*Q/CC)**2
      Q2G=Q*Q/G
      CALL CRIT(D*(Q2G/D**5)**.254,YC)
      WRITE(IOUT,110) YC
110   FORMAT(' Critical Depth =',F8.3)
      HMIN=.00001
      H11=1.
      IF(H2.GT.D) THEN
      IF(e.LT.1.E-12) THEN
      Sf=10.293591*QN/D**5.3333333
      ELSE
1     SSF1=SSF
      SSF=1.14-2.*ALOG(eD+7.3434728*VIS*D*SSF1/Q)
      IF(ABS(SSF-SSF1).GT.1.E-6) GO TO 1
      Sf=Q2G/(2.*(SSF*.78539816*D*D)**2)
      ENDIF
      FL1=FL-(H2-D)/(So-Sf)
      WRITE(IOUT,111) Sf,FL1
111   FORMAT(' Slope HGL for submerged culvert='/,E10.4,
     &' Intersection with top at x=',F8.2)
      FL1=FL1-.025*D*(So-Sf)
      IF(FL1.LT.0. .OR. (Sf-So).GT.0.) THEN
      H1=H2+FL*(Sf-So)
      WRITE(IOUT,112) H1,H1+.81056947*(FKE+1.)*Q2G/D**4
112   FORMAT(' Submerged to entrance, HGL at beg. =',F8.2,
     &' H1=',F8.2)
      STOP
      ENDIF
      Y(1)=.975*D
      ELSE IF(H2.GT.1.05*YC) THEN
      Y(1)=H2
      ELSE
      Y(1)=1.05*YC
      WRITE(IOUT,113) YC,H2
113   FORMAT(' Depth at downstream end is critical',F8.2,
     &' rather than H2=',F8.2)
      ENDIF
      WRITE(IOUT,114)
114   FORMAT(' GVF Profile',/,'   x     Y',/1X,18('-'))
      XX=FL1
      WRITE(IOUT,115) XX,Y
      DO 10 I=1,IFIX(FL1/DX+.9)
      X2=XX-DX
      IF(X2.LT. 0.) X2=0.
      CALL ODESOL(Y,DY,1,XX,X2,TOL,H11,HMIN,1,XP,YP,W,DYX)
      WRITE(IOUT,115) X2,Y
115   FORMAT(2F9.3)
10    XX=X2
      COSB=1.-Y(1)/DH
```

```
      BETA=ACOS(COSB)
      A=D25*(BETA-COSB*SIN(BETA))
      H1=Y(1)+(FKE+1.)*Q2G/(2.*A*A)
      WRITE(IOUT,116) Y(1),H1,A
116   FORMAT(' Upstream depth in Culvert =',F8.2,' Upstream
     &head H1 =',F8.2, ' Area =',F8.2)
      END
      SUBROUTINE CRIT(YB,YC)
      COMMON /TRAS/D,DH,D25,So,Q2G,QN
      M=0
      COSB=1.-YB/DH
      BETA=ACOS(COSB)
1     F=Q2G*D*SIN(BETA)-(D25*(BETA-COS(BETA)*SIN(BETA)))**3
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 2
      F1=F
      BET=BETA
      BETA=1.01*BETA
      GO TO 1
2     DIF=(BETA-BET)*F1/(F-F1)
      BETA=BET-DIF
      IF(ABS(DIF).GT.1.E-5 .AND. M.LT.30) GO TO 1
      YC=DH*(1.-COS(BETA))
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1)
      COMMON /TRAS/D,DH,D25,So,Q2G,QN
      YY=Y(1)
      IF(YY.LT. 0.05) YY=.05
      COSB=1.-YY/DH
      IF(COSB.LT. -1.) COSB=-.995
      BETA=ACOS(COSB)
      SINB=SIN(BETA)
      P=D*BETA
      A=D25*(BETA-COSB*SINB)
      SoSf=So-QN*((P/A)**.666666667/A)**2
      FRM1=1.-Q2G*D*SINB/A**3
      DY(1)=SoSf/FRM1
      RETURN
      END
```

EXAMPLE PROBLEM 4.25

A 5 ft circular culvert is 100 ft long, has a bottom slope $S_o = 0.001$ and a Manning's n = 0.013. If the downstream depth $H_2 = 4.5$ ft, and the culvert is to convey a flow rate Q = 50 cfs, what must the upstream head H_1 be? (The entrance loss coefficient is $K_e = 0.1$)

Solution

The input to program CULVERD1 to solve this problem consists of .000001 5 50 4.5 .1 100 .013 .001 32.2. The solution indicates that the upstream depth at the beginning of the culvert is $Y_1 = 4.43$ ft and the upstream head $H_1 = 4.56$ ft. It is interesting to note that if the 50 cfs were flowing at a uniform depth in this culvert, the depth would be $Y_o = 2.81$ ft (with $E_o = 3.11$ ft and $F_r = 0.51$). If the depth were uniform at 4.5 ft (i.e., the downstream depth), then the flow rate would be $Q_o = 87.8$ cfs (with $E_o = 4.85$ ft, and $F_r = 0.33$). Thus, even though the required upstream head is greater than the downstream head, the flow rate is considerably less than would take place if the depth were the same as that at the downstream end. The entrance loss has some influence on this result, but the major effect comes from the fact that the depth through the culvert is increasing along an M_1 GVF profile from 4.43 to 4.5 ft. Notice that if the upstream depth were the same, i.e., $H_1 = 4.43$ ft, and that if the downstream depth rose to 4.53 ft, the flow in the culvert would cease, and with a larger H_2, the flow would reverse.

EXAMPLE PROBLEM 4.26

A 5 ft circular culvert is 100 ft long, has a bottom slope $S_o = 0.003$, and a Manning's n = 0.013. If the upstream head is $H_1 = 4.0$ ft, and the downstream head is $H_2 = 4.2$ ft, what is the flow rate and the depth at the entrance of the culvert, if $K_e = 0.1$?

Solution

The input to program CULVERTD to solve this problem consists of 6 .000001 .0001 5 4 4.2 .1 100 .013 .003 32.2 5 0. The solution gives a flow rate Q = 37.6 cfs, with the upstream depth in the culvert $Y_1 = 3.91$ ft. It is interesting to note that if $H_1 = 4.0$ ft and a uniform flow were occurring in the channel, the flow rate would be $Q_o = 94.4$ cfs, with $Y_o = 2.97$ ft and $F_{ro} = 0.867$. If the uniform depth in the culvert were $Y_o = 4.0$ ft, then the flow rate would be $Q_o = 139.44$ cfs, with $E_o = 5.07$ ft. Or if the uniform depth equaled the downstream head or $Y_o = 4.2$ ft, then $Q_o = 145.66$ cfs, with $E_o = 5.26$ ft and $F_r = 0.665$.

EXAMPLE PROBLEM 4.27

If the downstream head H_2 of the previous problem is less than the critical depth, what is the flow rate? Solve this problem for a bottom slope of $S_o = 0.001$ and also $S_o = 0.003$.

Solution

The input to CULVERTD is 6 .0000001 .0005 5 4 2 .1 100 .013 .001 32.2 5 0 (for $S_o = 0.003$, the third from the last value is changed to .003). The first solution gives a flow rate Q = 89.9 cfs, an upstream depth $Y_1 = 3.235$ ft, and a critical depth Y_c at the end of the channel of 2.692 ft. When $S_o = 0.003$, then Q = 94.5 cfs, $Y_1 = 2.957$ ft, and $Y_c = 2.763$ ft. Note that the increase in bottom slope has a relatively small effect on increasing the flow rate. Also the M_2 that occurs in these problems has a rather small effect on increasing the flow rate beyond normal for the steeper of these two the bottom slopes. If $S_o = 0.001$, then $Q_o = 71.9$ cfs, and if $S_o = 0.003$, then $Q_o = 94.4$ cfs.

EXAMPLE PROBLEM 4.28

A 5 ft diameter culvert that is 300 ft long has a bottom slope $S_o = 0.003$, and a Manning's n = 0.013, and $K_e = 0.1$. If the downstream depth is 5.05 ft, and the upstream depth is 4.8 ft, determine the flow rate and the GVF profile in the culvert.

Solution

The input to CULVERTD is 3 .0000001 .05 5 4.8 5.05 .1 300 .013 .003 32.2 10 0. The top of the culvert is submerged from x = 192.9 ft to the end and the flow rate is 96.3 cfs, which is less by 29.7 cfs than if a uniform flow were to occur ($Q_o = 126.0$ cfs). The upstream depth is $Y_1 = 4.31$ ft.

4.12 GVF PROFILES IN NONPRISMATIC CHANNELS

The process of obtaining a GVF profile in a nonprismatic channel is almost identically to the procedure defined previously for prismatic channels when considering Y, the dependent variable. The essential difference is that the derivative dY/dx must be defined by including the term involving $\partial A/\partial x$ that exists in the numerator of Equation 4.6. It should be observed that if the channel contracts in the downstream direction then $\partial A/\partial x$ is negative, and that the effect of this term is to make the numerator of Equation 4.6 smaller. Therefore, for subcritical flows, a contracting cross section will cause a water surface profile below what it would be otherwise. Likewise, for subcritical flows, an enlarging cross section in the direction of the flow will cause the water surface to rise above what it would do under similar conditions, but in a prismatic channel. Reasons for this behavior can be rationalized by considering that the water surface elevation is the velocity head below the energy line. On the other hand, if the flow is supercritical, then the flow in a nonprismatic channel causes the opposite effects. For a supercritical flow, a contracting cross section in the direction of flow will cause the water surface to rise higher than it will in a prismatic channel, and if the cross section expands, the water surface elevation will be lower than in a prismatic channel. To enlarge an expansion will cause the flow to be subcritical with the formation of a hydraulic jump upstream therefrom, however. Enlargements for supercritical flows will be dealt with in Chapter 5.

If one desires to solve the alternative form of the ODE that involves the specific energy E as the dependent variable, or the form that considers x the dependent variable and E the independent variable, i.e., Equations 4.8 or 4.10a, then it is not necessary to include an extra term for the effects of an enlarging or contracting cross section, because these equations already include the effects of a nonprismatic channel. However, if x is considered the independent variable, then the use of Equation 4.8 requires that the implicit energy equation $E = Y + Q^2/(2gA^2)$ be solved repeatedly for depth Y, as a needed auxiliary variable, because the area, the wetted perimeter, and the top width are all defined as functions of Y rather than E. (See the last of Appendix C for another example solved in a nonprismatic channel.)

EXAMPLE PROBLEM 4.29

A trapezoidal channel reduces in size from $b_1 = 15$ ft and $m_1 = 2$ to $b_2 = 8$ ft and $m_2 = 1.5$ over a 200 ft length. The bottom slope across this reduction is $S_{o1} = 0.0013$, and Manning's $n = 0.015$. Downstream, the channel is very long and has a bottom slope $S_{o2} = 0.00099$. For a flow rate of $Q = 500$ cfs, determine the water surface profile across the contracting section of the channel.

Solution

The solution begins by noting that the downstream channel controls the depth at the end of the contracting section equal to its normal depth. A solution of Manning's equation for the downstream channel indicates its normal depth is 5 ft. Therefore, the GVF profile must begin at the downstream end of the contracting channel. Using the subroutine ODESOL described in Appendix C, the following general program was used to solve this problem. Note, this program also allows a lateral inflow or an outflow to take place. In other words, it solves Equation 4.6. The input required by this program to solve the problem consists of

6 .001 –10 5 500 .015 .0013 15 –.035 2 –.0025 200 0 32.2

Note that BO and FMO are the FORTRAN variables representing the bottom width and the side slope at the upstream end of the nonprismatic channel, and the DB and the DM are the FORTRAN variables that represent the derivatives of db/dx and dm/dx, respectively. These derivatives are assumed constant across the length of the channel, and are negative for this problem even though the computations proceed from the downstream end toward the upstream end.

Program EPR4_29.FOR to solve Example Problem 4.29 (EPR4_29K.FOR call on RUKUSTF as ODE solver) {EPR4_29.c}

```
      REAL Y(1),DY(1),XP(1),YP(1,1),WK1(1,13)
      EXTERNAL DYX
      CHARACTER*1 ANS
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE,BO,FMO,FN,SO,
     &Q2,FNQ,DB,DM,QS,*UQ,QG,ICASE,XBEG,QBEG,C,G
      WRITE(6,* )'GIVE IOUT,TOL,DELX,YB,Q,FN,SO,BO,DB,
     &FMO,DM,XBEG,XEND,g'
1     READ(5,*)IOUT,TOL,DELX,YB,Q,FN,SO,BO,DB,FMO,DM,XBEG,XEND,G
      C=1.
```

```
      IF(G.GT.30.) C=1.486
      QBEG=Q
      WRITE(6,*)'IS THERE LATERAL OUTFLOW(INFLOW)? Y/ N'
      READ(5,200) ANS
200   FORMAT(A1)
      IF(ANS.EQ.'Y'.OR. ANS.EQ.'y') THEN
      WRITE(6,*)'ICASE,QS,UQ: ICASE=1-bulk',' outflow,2-seep.,
     &3-inflow'
      READ(5,*) ICASE,QS,UQ
      Q=QBEG+XBEG*QS
      UQ=UQ/G
      ELSE
      ICASE=0
      QS=0.
      UQ=0.
      ENDIF
      H1=-.01
      Y(1)=YB
      FNQ=FN*Q/C
      QG=Q/G
      Q2=Q*Q/G
      X=XBEG
      B=BO+DB*X
      FM=FMO+DM*X
      A=(B+FM*Y(1))*Y(1)
      E=Y(1)+Q2/(2.*A*A)
      FUNM=Y(1)*Y(1)*(B/2.+FM*Y(1)/3.)+Q2/A
      WRITE(IOUT,100) X,Y,E,FUNM
2     XZ=X+DELX
      CALL ODESOL(Y,DY,1,X,XZ,TOL,H1,HMIN,1,XP,YP,WK1,DYX)
      X=XZ
      B=BO+DB*X
      FM=FMO+DM*X
      A=(B+FM*Y(1))*Y(1)
      IF(ICASE.GT.0) THEN
      Q=QBEG+X*QS
      QG=Q/G
      Q2=Q*Q/G
      FNQ=FN*Q/C
      ENDIF
      E=Y(1)+Q2/(2.*A*A)
      FUNM=Y(1)*Y(1)*(B/2.+FM*Y(1)/3.)+Q2/A
      WRITE(IOUT,100) X,Y,E,FUNM
100   FORMAT(6X,3F10.3,F12.2)
      IF(DELX .LT. 0.) GO TO 8
      IF(X .LT. XEND) GO TO 2
      GO TO 1
8     IF(X .GT. XEND) GO TO 2
      STOP
      END
      SUBROUTINE DYX(X,Y,DY)
      REAL Y(1),DY(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE,BO,FMO,FN,SO,Q2,FNQ,DB,
     &DM,QS,UQ,QG,IC,XBEG,QBEG,C,G
      B=BO+DB*X
      FM=FMO+DM*X
```

```
20     A=(B+FM*Y(1))*Y(1)
       IF(IC.GT.0) THEN
       Q=QBEG+X*QS
       QG=Q/G
       Q2=Q*Q/G
       FNQ=FN*Q/C
       ENDIF
       T=B+2.*FM*Y(1)
       P=B+2.*SQRT(FM*FM+1.)*Y(1)
       SF=(FNQ*(P/A)**.66666667/A)**2
       A3=A**3
       FR2=Q2*T/A3
       DA=Y(1)*(DB+Y(1)*DM)
       IF(DA.NE. .0) THEN
       TA=Q2*DA/A3
       ELSE
       TA=0.
       ENDIF
       IF(IC.EQ.0) THEN
       TQ=0.
       ELSE
       GO TO (31,32,33),IC
31     FQ=0.
       GO TO 34
32     FQ=QG*QS/A**2
       GO TO 34
33     HC=(Y(1)*Y(1)*(B/2.+FM*Y(1)/3.))/A
       FQ=((QG/A-UQ)*QS+HC*DA)/A
34     TQ=QG*QS/(A*A)+FQ
       ENDIF
40     DY(1)=(SO-SF+TA-TQ)/(1.-FR2)
       RETURN
       END
```

The solution to this problem is given below:

x (ft)	Y (ft)	E (ft)	M (ft^3)
200.000	5.000	5.646	262.68
190.000	5.053	5.643	267.87
180.000	5.095	5.638	273.09
170.000	5.130	5.633	278.31
160.000	5.158	5.627	283.53
150.000	5.182	5.620	288.75
140.000	5.202	5.613	293.95
130.000	5.218	5.605	299.14
120.000	5.232	5.597	304.30
110.000	5.243	5.589	309.44
100.000	5.252	5.581	314.55
90.000	5.259	5.572	319.63
80.000	5.265	5.563	324.67
70.000	5.269	5.554	329.68
60.000	5.272	5.544	334.65

(*continued*)

(continued)

x (ft)	Y (ft)	E (ft)	M (ft^3)
50.000	5.274	5.535	339.57
40.000	5.275	5.525	344.46
30.000	5.275	5.515	349.31
20.000	5.274	5.505	354.11
10.000	5.272	5.495	358.87
.000	5.270	5.484	363.58

The effects of the converging channel can be seen by comparing the above solution with the two solutions given below for prismatic channels, first with b = 15 ft and m = 2, and second with b = 8 ft and m = 1.5. For both of these latter solutions, the downstream depth of 5 ft is above the normal depth and so the depth increases from the beginning to the 200 ft length of the channel, whereas when the channel contracts as in the above solution, the depth decreases, i.e., since $\partial A/\partial x$ is negative, the term $(Q^2/A^3)\ \partial A/\partial x$ tends to make the numerator of the ODE negative, whereas the numerator is positive for the prismatic channels since $S_o > S_f$.

	b = 15 ft, m = 2, Y_o = 3.47 ft			b = 8 ft, m = 1.5, Y_o = 4.67 ft		
x	Y	E	M	Y	E	M
200.0	5.000	5.248	332.95	5.000	5.646	262.68
190.0	4.989	5.239	331.73	4.995	5.643	262.44
180.0	4.977	5.229	330.52	4.990	5.640	262.21
170.0	4.966	5.219	329.33	4.985	5.637	261.98
160.0	4.955	5.210	328.13	4.980	5.634	261.75
150.0	4.944	5.200	326.95	4.976	5.631	261.52
140.0	4.932	5.191	325.77	4.971	5.628	261.30
130.0	4.921	5.181	324.60	4.966	5.626	261.08
120.0	4.910	5.171	323.44	4.961	5.623	260.87
110.0	4.899	5.162	322.29	4.957	5.620	260.65
100.0	4.888	5.152	321.14	4.952	5.617	260.45
90.0	4.877	5.143	320.00	4.948	5.615	260.24
80.0	4.866	5.134	318.87	4.943	5.612	260.04
70.0	4.855	5.124	317.74	4.939	5.609	259.84
60.0	4.843	5.115	316.62	4.935	5.607	259.64
50.0	4.832	5.106	315.51	4.930	5.604	259.45
40.0	4.821	5.096	314.41	4.926	5.602	259.26
30.0	4.810	5.087	313.32	4.922	5.599	259.07
20.0	4.799	5.078	312.23	4.918	5.597	258.89
10.0	4.789	5.069	311.15	4.914	5.594	258.71
0.0	4.778	5.060	310.07	4.910	5.592	258.53

The same problem can be solved using Equation 4.8. The FORTRAN listing below uses the ISML subroutine DVERK, described in Appendix C to obtain this solution. The input to this program consists of

```
'ES' 6 .001 −10 −10 5 500 .015 .0013 8 1.5 15 2 200 0 0
```

This program illustrates a slightly different form of input that might be used. In this case, B1 and FM1 are the bottom width and the side slope at the section where the computations begins, and B2 and FM2 are these values where the computations end, i.e., that correspond to XEND. Furthermore, the program is arranged to allow for a prismatic channel attached to a nonprismatic channel. The FORTRAN variables DELX1 and DELX2 are the intervals for these two portions of the total channel, respectively, at which the results are to be computed and printed. The FORTRAN variable X1 provides the x distance between the connected prismatic and non-prismatic channels. You should also note that this program contains a subroutine YSOL that solves the implicit specific energy equation $E = Y + Q^2/(2gA^2)$.

Program EPR4_29A.FOR to solve Example Problem 4.29 using Equation 4.8 (EPR4_29AK.FOR call on RUKUSTF as ODE solver and EPR4_29O.FOR call on ODESOL as the solver, also EPR29A.C is on the diskette)

```
      CHARACTER*2 UNIT
      LOGICAL SWITCH
      REAL E(1),C(24),W(2,9)
      EXTERNAL DEX
      COMMON QN,B1,FM1,B,FM,FN,Y,Q2G,SO,X1,,XBEG,SWITCH
      DATA NN,IND,NW/1,1,2/
      WRITE(6,*)' GIVE: UNIT,IOUT,TOL,DELX1,DELX2,N,SO,B1,FM1,B2,
     &FM2,XBEG,XEND,X1'
      READ(5,*)UNIT,IOUT,TOL,DELX1,DELX2,YB,Q,FN,FM1,B2,FM2,XBEG,
     &XEND,X1
      SWITCH=.FALSE.
      CC=1.
      G2=19.62
      IF(UNIT .EQ. 'SI') GO TO 10
      CC=1.486
      G2=64.4
10    Y=YB
      QN=(FN*Q/CC)**2
      Q2G=Q*Q/G2
      Q2=2.*Q2G
      DB=B2-B1
      DFM=FM2-FM1
      B=B1
      FM=FM1
      DELX=DELX1
      X=XBEG
      A=(B+FM*YB)*YB
      E(1)=Y+Q2G/A**2
      WRITE(IOUT,15) Q,SO,FN,B1,FM1,B2,FM2
15    FORMAT(' SOLUTION TO GRADUALLY VARIED FLOW WITH',/,' Q=',
     &F10.2, ' SO=',F10.6,' n=',F7.4,/,' B1=',F8.1,' m1=',F8.2,/,
     &' B2=',F8.1, ' m2=',F8.2,/1X,40('-'),/,' x Y EM',/,1X,40('-'))
      FUNM=Y**2*(.5*B+FM*Y/3.)+Q2/A
      WRITE(IOUT,40) X,Y,E,FUNM
20    IF(X.GT.X1) GO TO 30
      IF(SWITCH) GO TO 30
      SWITCH=.TRUE.
      DELX=DELX2
      B=B2
      FM=FM2
30    XZ=X+DELX
      CALL DVERK(NN,DEX,X,E,XZ,TOL,IND,C,NW,W)
      IF(IND.LT.1) GO TO 50
      CALL YSOL(E,X)
      FUNM=Y**2*(.5*B+FM*Y/3.)+Q2/((B+FM*Y)*Y)
      WRITE(IOUT,40) X,Y,E,FUNM
40    FORMAT(3F10.3,F10.2)
      IF(X.GT.XEND) GO TO 20
```

```
      STOP
50    WRITE(6,*)' ERROR TERMINATION',IND
      END
      SUBROUTINE DEX(N,X,E,EPRIME)
      LOGICAL SWITCH
      REAL E(N),EPRIME(N)
      COMMON QN,B1,FM1,B,FM,FN,Y,Q2G,SO,X1,DB,DFM,XBEG,SWITCH
      CALL YSOL(E,X)
      P=B+2.*SQRT(FM*FM+1.)*Y
      A=(B+FM*Y)*Y
      SF=QN*((P/A)**.6666667/A)**2
      EPRIME(1)=SO-SF
      RETURN
      END
      SUBROUTINE YSOL(E,X)
      REAL E(1)
      LOGICAL SWITCH
      COMMON QN,B1,FM1,B,FM,FN,Y,Q2G,SO,X1,DB,DFM,XBEG,SWITCH
      IF(Y.LT.E(1).AND.Y.GT..6*E(1)) GO TO 10
      Y=.9*E(1)
10    NCT=0
20    NT=0
30    IF(SWITCH) GO TO 40
      XX=1.-(X-X1)/(XBEG-X1)
      B=B1+DB*XX
      FM=FM1+DFM*XX
40    A=(B+FM*Y)*Y
      F=E(1)-Y-Q2G/A**2
      IF(NT.GT.0) GO TO 50
      F1=F
      Y=Y-.001
      NT=1
      GO TO 30
50    Y=Y+.001
      DIF=.001*F1/(F1-F)
      NCT=NCT+1
      Y=Y-DIF
      IF(NCT.LT.15.AND.ABS(DIF).GT. .000001) GO TO 20
      IF(NCT.EQ.15) WRITE(6,*)' DID NOT CONVERGE'
      RETURN
      END
```

4.13 GVF PROFILES IN BRANCHED CHANNEL SYSTEMS

Flow in branched channel systems is uniform only under very special conditions. These conditions must create uniform depths in all channels that join at a junction, and simultaneously satisfy the energy equation between all branches of the channel at this junction, as well as the junction continuity equation. In other words, if the main channel is designated by subscript 1, and subscript n denotes the last channel of the branch, then the following system of simultaneous equations must be satisfied:

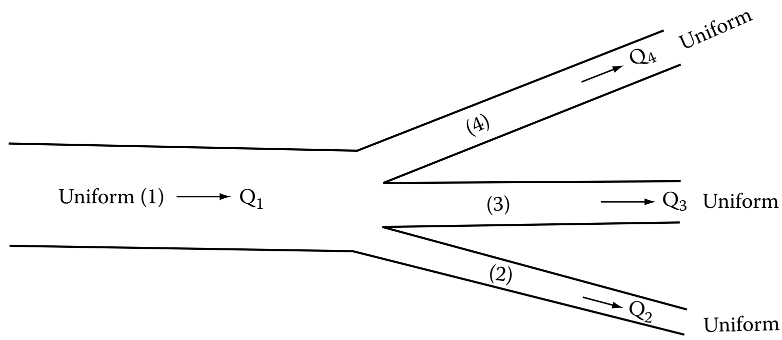
$$Q_i = \frac{C_u}{n_i} \frac{A_i^{5/3}}{P_i^{2/3}} S_o^{1/2} \quad \text{(for i = 1 to n, or n equations)}$$

$$E_1 = E_i + \Delta E_i + \Delta z_i \quad \text{(for i = 2 to n, or n - 1 equations)}$$

$$Q_1 = \sum_{2}^{n} Q_i \quad \text{(1 equation)}$$

in which ΔE_i is the energy loss between channel 1 and channel i and is generally determined by multiplying the velocity head of channel i by a minor loss coefficient K_{Li}, and Δz_i is the difference in the bottom elevation between channel i and channel 1 at the junction, or $\Delta z_i = z_i - z_1$. Thus, there are 2n equations available.

If no constraints existed, a branched system of channels could be designed for a given flow rate Q_1 so that a uniform depth occurred in all channels, since more than 2n variables are available. For example, the slope of the channel bottoms S_{oi} and a size variable, such as the bottom widths b_i could be selected as the unknowns.



Main channel branching into three channels with the flow from the main channel divided into the three channels.

To illustrate how such a design might be accomplished, consider a main channel that branches into three channels such that n = 4. All channels are trapezoidal. The table below lists the variables. For this case, there are 30 variables, 4 Q's + 4 Y's + 4 b's + 4 m's + 4 S_o's + 4 n's + 3 K_L's + 3 Δz's = 30. The number of equations available are eight (four Manning's equations + three energy equations + one junction continuity equation = eight).

Table giving solution to uniform flow problems for four cases of a junction of four channels; allowing for 8 variables to be considered unknowns.

Case No.	Channel No.	Q (m^3/s)	Y (m)	b (m)	m	S_o	n	K_L	Δz (m)
1	1	90.	2.731	4.00	1.5	.0015	0.013		
	2	20.	3.404	1.629	0.5	.0004	0.013	0.08	0
	3	25.	3.322	0.185	1.0	.0006	0.013	0.08	0
	4	45.	3.116	0.325	1.5	.0008	0.013	0.08	0
2	1	90.	2.588	4.707	1.5	.0015	0.013		
	2	13.721	3.404	1.0	0.5	.0004	0.013	0.08	0
	3	29.447	3.142	1.0	1.0	.0006	0.013	0.08	0
	4	46.832	2.955	1.0	1.5	.0008	0.013	0.08	0
3	1	47.589	1.983	4.0	1.5	.0015	0.013		
	2	7.672	2.482	1.0	0.5	.0004	0.013	0.08	0
	3	15.570	2.380	1.0	1.0	.0006	0.013	0.08	0
	4	24.347	2.250	1.0	1.5	.0008	0.013	0.08	0
4	1	50.	2.034	4.0	1.5	.0015	0.013		
	2	10.	2.3	0.566	0.5	.00178	0.013	0.08	0
	3	15.	2.3	0.273	1.0	.00120	0.013	0.08	0
	4	25.	2.3	0.839	1.5	.000836	0.013	0.08	0

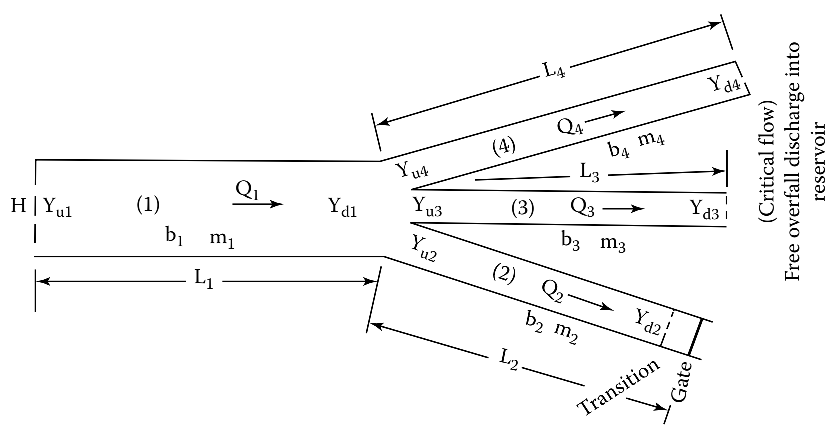
In other words, 22 variables must be given values as knowns. In the table above, several case have been solved for different combinations of known and unknown variables. Variables that have been solved are underlined. The others have been specified.

For cases # 1 and # 4, the continuity equation is actually satisfied by the specified flow rates, and therefore for these cases only seven equations are available. For case # 1 the solution can proceed by (1) solving Manning's equation for channel 1 for the depth, (2) from this depth compute the specific energy at the junction, (3) as pairs of two equations solve the energy and Manning's equation simultaneously for the depth and the bottom width for Channels 2, 3, and 4. For case # 4 Manning's equation is solved for the depth in channel # 1 and from this, the energy is determined at the junction as for case # 1. Thereafter, the energy equation and Manning's equation are solved as pairs for the bottom width b and the bottom slope of the channel.

For case # 2, a flow rate of $90\,m^3/s$ has been specified in Channel 1, and both its depth and the bottom width are unknown. The flow rate and the depth are unknown for the remaining three channels. For case # 3, the flow rates and depths of flow are the unknowns for all channels. For these two latter cases, all eight of the above equations are available and must be solved simultaneously.

In practice, solutions to problems of branched channels are not as simple as illustrated above because channels are generally not sufficiently long for a uniform flow to occur. Furthermore, if the channels were all very long, and if a branching system of channels were designed as above so that a uniform flow occurred, a nonuniform flow would exist for all flow rates different from the designed flow rate. For channels that are not extremely long, the effects from GVFs caused by downstream gates, breaks in grade, transitions, etc., will extend to the junction and will alter the depths and flow rates here. The solution procedure to obtain these depths, flow rates, etc., must now be altered to include the numerical solution to these GVFs. In other words, the solution procedure requires that a combined system of ODEs and algebraic equations be solved, simultaneously. The general procedures for solving systems of ODEs and algebraic equations simultaneously have been described earlier. In the following discussion, these procedures will be defined specifically for a branched system of channels in which controls may exist in each of the downstream channels that either increase or decrease, the depth in it from the normal depth. Furthermore, it will be assumed that the main channel that flows into the branch is supplied by a reservoir with a known water surface elevation. All channels have a known length.

To define variables in such a branched system of channels, a double subscript notation will be used for depths; the second of which denotes the channel number, as before, and the first indicates the depth at the upstream end if this is a u, and at the downstream end if this is a d. Since flow rates will be the same at both the upstream and the downstream ends of each channel in the branched system, the Q's will only have a single subscript to denote the channel number.



Sketch that contains four channels, but which might contain n channels.

In writing the governing equations, n will be used in place of 4 so that more or fewer channels might be involved. On this sketch, the three most common types of control are shown: (1) a gate, (2) a free overfall, and (3) the channel discharging into a reservoir with a known water surface elevation. For the present, it will be assumed that a subcritical flow exists in all channels, so the possibility of a hydraulic jump occurring is eliminated. The equations that govern the problem are

$$F_1 = H_u - Y_{u1} - (1+K_e)\frac{Q_1^2}{2gA_{u1}^2} = 0 \quad \text{(1 equation)}$$

$$F_{ener,j} = Y_{d1} + \frac{Q_1^2}{2gA_{d1}^2} - Y_{ui} - (1+K_{Li})\frac{Q_j^2}{2gA_{ui}^2} - \Delta z_i = 0 \quad \text{(n − 1 equations) (i and j = 2,…, n)}$$

$$F_{GVF} = Y_{iu} - Y_{i\,ode} = 0 \quad \text{(n equations)}$$

$$F_{cont} = Q_i - \sum_2^n Q_i = 0 \quad \text{(1 equation)}$$

$$F_{ener,end} = Y_{Gi} + \frac{Q_1^2}{2gA_{Gi}^2} - Y_{i2} - (1+K_{Li})\frac{Q_i^2}{2gA_{di}^2} = 0 \quad \text{(n − 1 equations), (i = 2,…, n)}$$

or

$$F_{crit,end} = Q_i^2 T_{di} - gA_{di}^3 = 0 \quad \text{(n − 1 equations), (i = 2,…, n)}$$

or

$$F_{reser,end} = Y_{di} - H_{di} = 0 \quad \text{(n − 1 equations), (i = 2,…, n)}$$

There are now 3n equations available. The equations coming from solving the GVF profiles replace Manning's equation that was available for the problem that assumed uniform flow in all channels, and n additional equations are added; F_1 comes from the energy at the upstream end of the main supply channel, and n − 1 equations come from the controls at the downstream end of each of the branches. For these latter equations, the downstream condition dictates whether the equation is $F_{ener,end}$, $F_{crit,end}$, or $F_{reser,end}$. The first of these is the energy equation written across the gate. The depths downstream from the gate are denoted by Y_{di}, and these are assumed known being equal, generally, to the height of the gate times its contraction coefficient or $Y_{Gi} = C_{ci}Y_{gate,i}$. The second type of equations $F_{crit,end}$, assume the flow is critical at the end of the channel. In the case of $F_{reser,end}$, this equation indicates that the downstream depth is known in that branched channel and equals the depth of the water in the reservoir above the channel bottom. (The velocity head is dissipated as the flow enters the reservoir.) In all other cases, an additional unknown variable is added since now each channel has two depths, an upstream depth Y_{1i} and a downstream depth Y_{2i}.

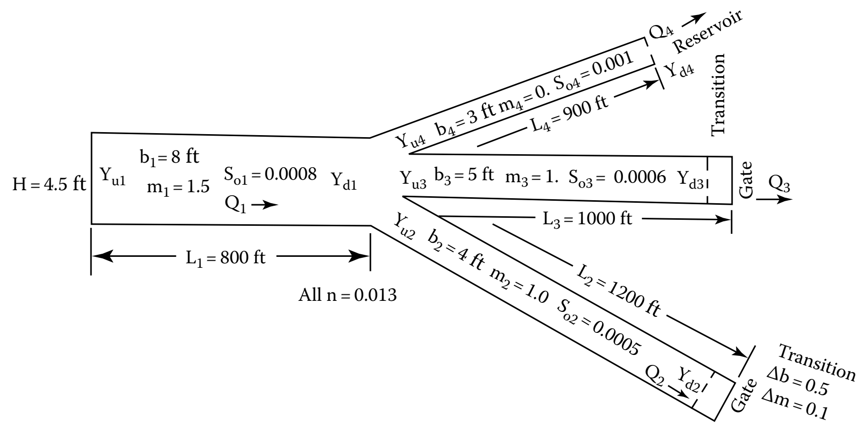
It is assumed that the GVF solutions in the above equations will include the term involving $\partial A/\partial x$ for nonprismatic channels as the computation proceeds through any transitions that may exist. An alternative to solving the GVF through transitions is to use another algebraic energy equation written from the beginning to the end of the transition, with an appropriate minor loss coefficient.

If the latter alternative were taken, then an additional unknown also would be added to each channel containing a transition; the depth at the end of the transition, as well as that at its beginning.

If the size, the roughness coefficient, the bottom slope, and the length of each channel is assumed to be known, then the above 3n equations allow for the following 3n variables to be solved: (Q_i, Y_{ui}, and Y_{di}) for i = 1 to n. For a combined system of algebraic equations and ODEs, the best method for solving the system is to utilize the Newton method described earlier. The computer program used to solve the following example problem illustrates how this method is implemented. This program has been written to allow any number of channels to join at the junction, and also accommodates any of the three types of downstream controls given in the above sketch. It also allows for any of the channels to be circular, as well as trapezoidal. If circular, however, transitions are not allowed at their downstream ends.

EXAMPLE PROBLEM 4.30

A trapezoidal channel with b_1 = 8 ft, m_1 = 1.5, S_{o1} = 0.0008, and L_1 = 800 ft. divides into three channels with the following: b_2 = 4 ft, m_2 = 1, S_{o2} = 0.0005, L_2 = 1200 ft, b_3 = 5 ft, m_2 = 1, S_{o3} = 0.0006, L_3 = 1000 ft, b_4 = 3 ft, m_2 = 0, S_{o2} = 0.001, L_2 = 900 ft. All channels have a Manning's n = 0.013. The length of the transition is 15 ft. At the downstream ends of channels #2 and #3, there are gates that cause the depth of flow downstream from the gate to be 1.5 ft. In the case of channel #2, the channel changes to a rectangular section with a 3.5 ft width at the gate. Channel #4 discharges into a reservoir whose water surface elevation is 4.5 ft above the downstream end of this channel. The reservoir that supplies channel #1 has a water surface elevation that is H = 4.5 ft above the channel bottom. Assume all minor loss coefficients to be K_L = 0.08. Determine the flow rates in the four channels, and the depths at both their upstream and downstream ends.



Solution

There are 12 unknowns in this problem; the flow rates in the four channels, the depth at the upstream end, and the depth at the downstream end of the four channels. The 12 equations needed to solve the problem are as follows: one energy equation between the reservoir and the beginning of channel #1, three energy equations between the four channels at their junction, one continuity equation at the junction, four equations that come from solving the ODE that described the GVF in the four channels, two equations across the gates at the downstream ends of channels # 2 and #3, and finally the equation that indicates that the depth in channel #4 is the depth of the reservoir into which it discharges. (Actually, since this depth is known, there are only 11 unknowns and 11 equations available.)

Below is a FORTRAN listing that solves this problem. In this program, NO is the number of channels and the array X contains the unknowns. The order of the unknowns in X are as follows:

Q_1, Q_2, Q_3, Q_4, Y_{u1}, Y_{u2}, Y_{u3}, Y_{u4}, Y_{d1}, Y_{d2}, Y_{d3}, Y_{d4} = 4.5 ft. The program uses ITYP(I) to distinguish whether a channel is trapezoidal or circular (ITYP(I)=1 for trapezoidal and ITYP(I)=2 for circular). The array ICTL determines the type of downstream boundary condition; if ICTL(I)=1, then a gate exists with a water depth downstream from the gate given by YG(I), if ICTL(I)=2, then the depth is critical and YG(I) must be zero, and if ICTL(I)=3, then the channel discharges into a reservoir whose water surface elevation is given by YG(I). The variables TOL,ERR,H and G in the second read statement are: the accuracy parameter in solving the ODE, the error criteria for the Newton iteration, the head in the upstream reservoir that supplies channel #1, and the acceleration of gravity. The order of the guesses used for the NEWTON method are read in the third read statement and consist of (Q_i, Y_{ui}, & Y_{di} for i = 1 … 4). The function subprogram FUN defines the equations, it in turn uses the function subprogram AR to give the area (the wetted perimeter, and the top width when needed). The logical variable NTRAN (not transition) is used by AR. If NTRAN is true, then the bottom width and the side slope of the channel given in B(I) and FM(I) are used. When NTRAN is false, then AR expects BB and FMM to be defined for the transitional part of the channel. The arrays DB(I) and DFM(I) are read in to define the total change (negative for a contraction) across the transition, and the array LT(I) contains the length of the transition at the end of each channel.

FORTRAN listing of program to solve branched channel with downstream controls (SOLBRA. FOR)

```
      PARAMETER (N=12,M=36)
      LOGICAL*2 IPERM,NTRAN
      REAL F(M),D(M,M),FN(N),SO(N),L(N),LT(N),B(N),FM(N),DZ(N),
     &KL(N),DB(N),DFM(N),YG(N),X(M)
      INTEGER*2 INDX(M),ITYP(N),ICTL(N)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NO,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,DZ,KL,
     &DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QN,Q2G,BB,FMM,IPERM,
     &NTRAN,JI
      DATA IN,IOUT/2,3/
      IPERM=.FALSE.
      NTRAN=.TRUE.
C ICTL = 1,2 OR 3 for types of downstream controls,
C gate=1,critical=2, reservoir=3.
C YG is depth behind gate for gate; = 0 for critical;
C res. depth if C reservoir. DB and DFM are changes (+ or -)
C across transition of b and m.
C IN UNKNOWN VECTOR X Q COMES 1ST; UPSTREAM DEPTH NEXT &
C DOWNSTREAM DEPTH LAST,
C i.e. Q(I)=X(I);YU(I)=X(I+NO); YD(I)=X(I+NO2)
      READ(IN,*) NO,(ITYP(I),B(I),FM(I),FN(I),SO(I),L(I),LT(I),
     &DZ(I),DB(I),DFM(I),KL(I),YG(I),ICTL(I),I=1,NO)
      NO2=2*NO
      NEQS=3*NO
      READ(2,*) TOL,ERR,H,G
      READ(2,*) (X(I),X(I+NO),X(I+NO2),I=1,NO) ! GUESSES
      IF(G.GT. 30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      DO 10 I=1,NO
10    KL(I)=(KL(I)+1.)/G2
      NCT=0
```

```
20    DO 30 I=1,NEQS
      F(I)=FUN(I)
      DO 30 J=1,NEQS
      DX=.005*X(J)
      X(J)=X(J)+DX
      D(I,J)=(FUN(I)-F(I))/DX
30    X(J)=X(J)-DX
      CALL SOLVEQ(NEQS,1,M,D,F,1,DD,INDX)
      DIF=0.
      DO 40 I=1,NEQS
      X(I)=X(I)-F(I)
40    DIF=DIF+ABS(F(I))
      NCT=NCT+1
      IF(NCT.LT.30 .AND. DIF.GT.ERR) GO TO 20
      WRITE(IOUT,100) NO
100   FORMAT(' Solution to',I3,' Channel at Junction',/,1X,79('-'),
     &/,' No Ty b m n So',/,' L dz db dm Yu Yd Q',/,1X,79('-'))
      DO 50 I=1,NO
50    WRITE(IOUT,110)I,ICTL(I),B(I),FM(I),FN(I),SO(I),L(I),DZ(I),
     &DB(I),DFM(I),X(I+NO),X(I+NO2),X(I)
110   FORMAT(I3,I3,F7.2,F5.2,F6.3,F8.6,F7.0,3F6.2,2F7.3,F8.2)
      END
      FUNCTION FUN(II,DX,J)
      PARAMETER (N=12,M=36)
      EXTERNAL DYX
      LOGICAL*2 IPERM,NTRAN
      REAL FN(N),SO(N),L(N),LT(N),B(N),FM(N),DZ(N),KL(N),DB(N),
     &DFM(N),YG(N),X(M),Y(1),DY(1),W(1,13),XP(1),YP(1,1)
      INTEGER*2 ITYP(N),ICTL(N)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NO,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,DZ,KL,
     &DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QN,Q2G,BB,FMM,IPERM,
     &NTRAN,JI
      H1=-.05
      HMIN=.00001
      IF(II.EQ.1) THEN
      FUN=H-X(NO+1)-KL(1)*(X(1)/AR(1,X(NO+1)))**2
      ELSE IF(II.GT.1 .AND. II.LE.NO) THEN
      FUN=X(NO2+1)+(X(1)/AR(1,X(NO2+1)))**2/G2-X(NO+II)-KL(II)*
     &(X(II)/AR(II,X(NO+II)))**2+DZ(II)
      ELSE IF(II.GT.NO .AND. II.LE.NO2) THEN
      JI=II-NO
      XX=L(JI)+LT(JI)-1.E-5
      XZ=0.
      IPERM=.TRUE.
      Y(1)=X(NO2+JI)
      IF(ICTL(JI).EQ.2) Y(1)=1.1*Y(1)
      QN=(FN(JI)*X(JI)/CC)**2
      Q2G=X(JI)**2/G
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      FUN=X(NO+JI)-Y(1)
      ELSE IF(II.EQ.NO2+1) THEN
      FUN=X(1)
      DO 20 I=2,NO
20    FUN=FUN-X(I)
      ELSE
```

```
 JI=II-NO2
 NTRAN=.FALSE.
 BB=B(JI)+DB(JI)
 FMM=FM(JI)+DFM(JI)
 IF(ICTL(JI).EQ.1) THEN
 FUN=X(NO2+JI)+(X(JI)/AR(JI,X(NO2+JI)))**2/G2-YG(JI)-
&KL(JI)*(X(JI)/AR(JI,YG(JI)))**2
 NTRAN=.FALSE.
 ELSE IF(ICTL(JI).EQ.2) THEN
 AA=AR(JI,X(NO2+JI))
 IPERM=.FALSE.
 FUN=X(JI)**2*TOPW-G*AA**3
 ELSE
 FUN=X(NO2+JI)-YG(JI)
 ENDIF
 NTRAN=.TRUE.
 ENDIF
 RETURN
 END
 FUNCTION AR(I,YY)
 PARAMETER (N=12,M=36)
 LOGICAL*2 IPERM,NTRAN
 REAL FN(N),SO(N),L(N),LT(N),B(N),FM(N),DZ(N),KL(N),DB(N),
&DFM(N),YG(N),X(M)
 INTEGER*2 ITYP(N),ICTL(N)
 COMMON /TRAS/NO,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,DZ,KL,
&DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QN,Q2G,BB,FMM,IPERM,
&NTRAN,JI
 IF(ITYP(I).EQ.1) THEN
 IF(NTRAN) THEN
 BB=B(I)
 FMM=FM(I)
 ENDIF
 AR=(BB+FMM*YY)*YY
 IF(IPERM) THEN
 PERM=BB+2.*YY*SQRT(1.+FMM*FMM)
 TOPW=BB+2.*FMM*YY
 ENDIF
 ELSE
 COSB=1.-2.*YY/B(I)
 BETA=ACOS(COSB)
 AR=.25*B(I)**2*(BETA-SIN(BETA)*COSB)
 ELSE
 COSB=1.-2.*YY/B(I)
 BETA=ACOS(COSB)
 AR=.25*B(I)**2*(BETA-SIN(BETA)*COSB)
 IF(IPERM) THEN
 PERM=BETA*B(I)
 TOPW=B(I)*SIN(BETA)
 ENDIF
 ENDIF
 RETURN
 END
 SUBROUTINE DYX(XX,Y,DY)
 PARAMETER (N=12,M=36)
 LOGICAL*2 IPERM,NTRAN
```

```
 REAL FN(N),SO(N),L(N),LT(N),B(N),FM(N),DZ(N),KL(N),DB(N),
&DFM(N),YG(N),X(M),Y(1),DY(1)
 INTEGER*2 ITYP(N),ICTL(N)
 COMMON /TRAS/NO,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,DZ,KL,
&DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QN,Q2G,BB,FMM,IPERM,
&NTRAN,JI
 IF(XX.LE.L(JI)) THEN
 NTRAN=.TRUE.
 DAX=0.
 ELSE
 NTRAN=.FALSE.
 FLEN=(XX-L(JI))/LT(JI)
 BB=B(JI)+FLEN*DB(JI)
 FMM=FM(JI)+FLEN*DFM(JI)
 DAX=X(JI)**2/G*Y(1)*(DB(JI)+Y(1)*DFM(JI))/LT(JI)
 ENDIF
 AA=AR(JI,Y(1))
 A3=AA**3
 SF=QN*(ABS(PERM/AA)**.666666667/AA)**2
 DY(1)=(SO(JI)-SF+DAX/A3)/(1.-Q2G*TOPW/A3)
 RETURN
 END
```

The input data needed to solve the above problem with this program consist of the following:

```
4
1 8 1.5  .013 .0008  800.  0. 0. 0.   0. .08 0. 1
1 4 1.   .013 .0005 1200. 15. 0. -.5 -1. .08 1.5 1
1 5 1.   .013 .0006 1000. 15. 0. 0.   0. .08 1.5 1
1 3 0.   .013 .0006  900.  0. 0. 0.   0. .08 4.5 3
.0000001 .1 4.5 32.2
300. 4.2 4.4 100. 4.4 4.6 150. 4.4 3. 50. 4.4 5.
```

The solution to this problem is given in the following table written by the above computer program:

Solution to four channels at a junction

No.	T_y	b	m	n	S_o	L	d_z	d_b	d_m	Y_u	Y_d	Q
1	1	8.00	1.50	0.013	0.000800	800.0	0.00	0.00	0.00	4.149	4.562	269.93
2	1	4.00	1.00	0.013	0.000500	1200.0	0.00	–0.50	–1.00	4.748	5.007	79.28
3	1	5.00	1.00	0.013	0.000600	1000.0	0.00	0.00	0.00	4.632	5.074	144.79
4	3	3.00	0.00	0.013	0.000600	900.0	0.00	0.00	0.00	4.626	4.500	45.85

EXAMPLE PROBLEM 4.31

The problem is identical to the four channel problem above with the exception that at the downstream end of channel # 3, there is a short transition to a steep rectangular channel with a 4 ft bottom width.

Solution

The following is the input to this program for the above computer program:

```
4
1 8 1.5 .013 .0008 800. 0. 0. 0. 0. .08 0. 1
1 4 1. .013 .0005 1200. 15. 0. -.5 -1. .08 1.5 1
```

```
1 5 1. .013 .001 1000. 15. 0. -1. -1. .08 0 2
1 3 0. .013 .001 900. 0. 0. 0. 0. .08 4.5 3
.0000001 .1 4.5 32.2
280. 3.9 3.9 80. 4.4 4.9 150. 3.55 4.2 50. 4.2 4.5
```

and the solution consist of

Solution to four channels at a junction

No.	T_y	b	m	n	S_o	L	d_z	d_b	d_m	Y_u	Y_d	Q
1	1	8.00	1.50	.013	.000800	800.	.00	.00	.00	4.084	4.424	287.34
2	1	4.00	1.00	.013	.000500	1200.	.00	−.50	−1.00	4.667	4.919	78.41
3	2	5.00	1.00	.013	.001000	1000.	.00	−1.00	−1.00	4.505	3.628	156.88
4	3	3.00	.00	.013	.001000	900.	.00	.00	.00	4.478	4.500	52.05

In solving problems such as those above, it is important to keep in mind numerical accuracies and their impacts on the solution process. In the sketch showing a branched system of channels, the critical depth occurs at the downstream end of channel #3, yet a solution of the GVF ODE results in a division of zero at the critical depth. To accommodate this singular condition in the computer program, you will note from the listing that the starting depth for the downstream end of any channel with a critical depth specified, is multiplied by 1.1 (If(ICTL(I).EQ.2) Y(1)=1.1*Y(1)). Thus, the GVF computation begins at 10% above the critical depth. However, since the Newton method attempts to cause all equations (including the critical depth equation) to produce zero, this expedient could easily fail if "good initial guesses" are not supplied. Also, it is unrealistic to expect final solution accuracies of flow rates to two or three digits beyond the decimal point. Therefore, the ERR variable, which determines when the Newton method will stop its iteration, is the input at a much larger value (0.1) than the TOL variable that is an index to the accuracy of each individual GVF solution.

EXAMPLE PROBLEM 4.32

Many of the elements in the Jacobian matrix used by the Newton method are zero when solving problems dealing with channels at junctions. In the above program, the derivatives are evaluated numerically, and therefore unnecessary computations take place for these zero elements. Modify the above program so that these unnecessary computations are eliminated.

Solution

The solution can consist of defining a logical two-dimensional array NZERO(M,M), elements of which will be true only if nonzero Jacobian matrix elements occur. The following statements are added to the main program before the Newton iteration begins:

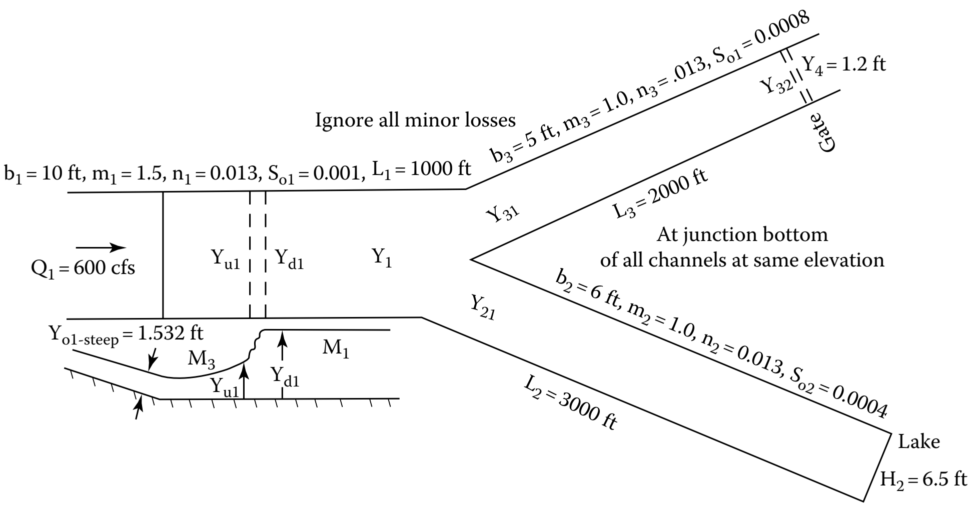
```
      DATA IN,IOUT/2,3/,NZERO/1296*.FALSE./
      NZERO(1,1)=.TRUE.
      NZERO(1,NO+1)=.TRUE.
      DO 2 I=2,NO
      NZERO(I,1)=.TRUE.
      NZERO(I,I)=.TRUE.
      NZERO(I,I+NO)=.TRUE.
2     NZERO(I,NO2+1)=.TRUE.
      DO 4 I=NO+1,NO2
      NZERO(I,I-NO)=.TRUE.
      NZERO(I,I)=.TRUE.
4     NZERO(I,I+NO)=.TRUE.
      DO 6 I=1,NO
6     NZERO(NO2+1,I)=.TRUE.
      DO 8 I=NO2+2,NEQS
      NZERO(I,I-NO2)=.TRUE.
8     NZERO(I,I)=.TRUE.
```

and the DO loops that evaluate the equations and Jacobian elements are modified as follows:

```
      DO 30 J=1,NEQS
      IF(NZERO(I,J)) THEN
      DX=.005*X(J)
      D(I,J)=(FUN(I,DX,J)-F(I))/DX
      ELSE
      D(I,J)=0.
      ENDIF
30    CONTINUE
```

EXAMPLE PROBLEM 4.33

A flow rate of $Q_1 = 600$ cfs occurs in a channel system as shown in the sketch below. The upstream channel is trapezoidal with $b_1 = 10$ ft, $m_1 = 1.5$, $n_1 = 0.013$. Upstream from a break in grade, the bottom slope is $S_{o11} = 0.06$, and downstream therefrom, the bottom slope is $S_{o12} = 0.001$. At a distance $L_1 = 1000$ ft downstream from the break in grade, the channel branches into two trapezoidal channels with $b_2 = 6$ ft, $m_2 = 1.0$, $n_2 = 0.013$, $S_{o2} = 0.0004$, and $b_3 = 5$ ft, $m_3 = 1.0$, $n_3 = 0.013$, $S_{o3} = 0.0008$, respectively. The first branch channel discharges into a lake at a distance $L_2 = 3000$ ft from the branch position with a water surface elevation that is $H_2 = 6.5$ ft above the channel bottom. The second branch channel has its flow controlled by a gate that is $L_2 = 2000$ ft downstream from the branch and produces a depth $Y_4 = 1.2$ ft immediate downstream from the gate. Determine the flow rates in the two branch channels, the depths in these channels, and the position of the hydraulic jump.



Solution

There are nine unknowns for this problem consisting of the following: Q_2, Q_3, Y_1, Y_{21}, Y_{31}, Y_{32}, Y_{u1}, Y_{d1}, and x. The nine equations needed to solve the nine unknowns consist of four GVF computations, and five algebraic equations. The first GVF equation defines the M_3-profile upstream from the hydraulic jump and the second, the M_1 (or M_2) GVF downstream from the jump. The computation for the M_3-profile will start at position x = 0, with the normal depth in the upstream steep channel, which consists of $Y_{o1\text{-steep}} = 1.532$ ft and proceeds downstream to the position x (unknown) where the hydraulic jump will occur, and the second such GVF profile will begin at the end of the upstream channel and proceed upstream to the position x. The other two GVF profiles are in the two branch channels, and both of these computations will proceed upstream. In the case of channel 2, the downstream boundary condition for the depth is the lake level of

6.5 ft, and in the case of channel 3, the downstream depth is the unknown Y_{32}, which is the alternative depth to the depth Y_4 across the gate. The nine equations are

$F_1 = Y_{u1} - Y_{GVF}(Y_{o1\text{-steep}}) = 0$ (M_3-GVF upstream from the hydraulic jump)

$F_2 = Y_{d1} - Y_{GVF}(Y_1) = 0$ (M_1(or M_2)-GVF downstream from the hydraulic jump)

$F_3 = Y_{21} - Y_{GVF}(H_2) = 0$ (M_1(or M_2)-GVF from the lake up to the branch in the channels)

$F_4 = Y_{31} - Y_{GVF}(Y_{32}) = 0$ (M_1-GVF from the gate up to the branch in the channels)

$F_5 = Q_2 + Q_3 - 600 = 0$ (Junction continuity)

$F_6 = Y_1 + (Q_1/A_1)^2/(2g) - Y_{21} - (Q_2/A_{21})^2/(2g) = 0$ (Energy at the junction between Channels # 1 and # 2)

$F_7 = Y_1 + (Q_1/A_1)^2/(2g) - Y_{31} - (Q_3/A_{31})^2/(2g) = 0$ (Energy at the junction between Channels # 1 and # 3)

$F_8 = Y_{32} + (Q_3/A_{32})^2/(2g) - Y_4 - (Q_3/A_4)^2/(2g) = 0$ (Energy across the gate)

$F_9 = b_1(Y_{u1}^2 - Y_{d1}^2)/2 + m_1(Y_{u1}^3 - Y_{d1}^3)/3 + Q_1^2/g\{1/A_{u1} - 1/A_{d1}\} = 0$ (Momentum across the hydraulic jump)

The program listed below provides the following solution: Q_2 = 437.98 cfs, Q_3 = 1.62.02 cfs, Y_1 = 6.734 ft, Y_{21} = 6.610 ft, Y_{31} = 6.981 ft, Y_{32} = 8.533 ft, Y_{u1} = 2.270 ft, Y_{d1} = 6.141 ft, and x = 298.7 ft.

Listing of program SOLJMP2.FOR designed to solve this example problem

```
      PARAMETER (N=9)
      EXTERNAL DYX
      CHARACTER*3 UNK(N)/'Q2 ','Q3 ','Y1 ','Y21','Y31','Y32',
     &'Y1u','Y1d','x '/
      REAL F(N),F1(N),D(N,N),X(N),B(3),FM(3),So(3),FN(3),L(3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B,FM,FN,So,L,G,G2,TOL,CC,QN,X,Y4,A4,QSG,
     &H2,BB,FMM,SSo,Q1,YO
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,3(b,m,n,So,L),Q1,Y4,H2,g,Yo'
      READ(*,*) IOUT,TOL,ERR,(B(I),FM(I),FN(I),So(I),L(I),I=1,3),
     &Q1,Y4,H2,G,YO
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      A4=G2*((B(3)+FM(3)*Y4)*Y4)**2
      WRITE(*,*)' GIVE guess for:'
      DO 2 I=1,N
      WRITE(*,3) UNK(I)
```

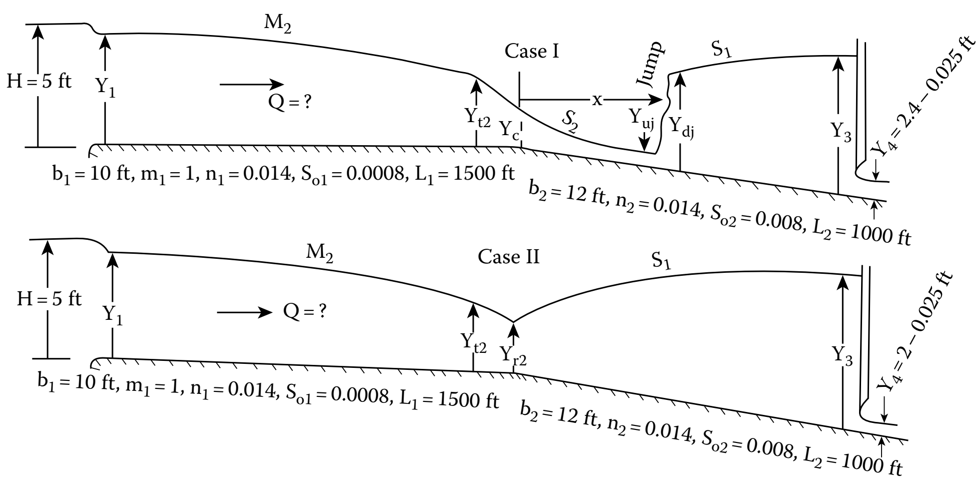
```
2     READ(*,*) X(I)
3     FORMAT(1X,A3,' =',\)
50    NCT=0
1     CALL FUN(F)
      DO 10 J=1,N
      DX=.005*X(J)
      X(J)=X(J)+DX
      CALL FUN(F1)
      DO 5 I=1,N
5     D(I,J)=(F1(I)-F(I))/DX
10    X(J)=X(J)-DX
      CALL SOLVEQ(N,1,N,D,F,1,DD,INDX)
      SUM=0.
      DO 15 I=1,N
      X(I)=X(I)-F(I)
15    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,110) NCT,SUM,X
110   FORMAT(' NCT=',I2,' SUM=',E12.5,3F10.4,/6F10.4)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 1
      WRITE(IOUT,*)' Solution:'
      DO 20 I=1,N
20    WRITE(IOUT,100) UNK(I),X(I)
100   FORMAT(1X,A3,' =',F10.3)
      END
      SUBROUTINE FUN(F)
      EXTERNAL DYX
      REAL X(9),F(9),W(1,13),Y(1),DY(1),XP(1),YP(1,1),B(3),FM(3),
     &So(3),FN(3),L(3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B,FM,FN,So,L,G,G2,TOL,CC,QN,X,Y4,A4,QSG,H2,
     &BB,FMM,SSo,Q1,YO
      H1=-.05
      HMIN=.00001
      DO 10 I=1,4
      IF(I.EQ.1) THEN
      XX=0.
      XZ=X(9)
      BB=B(1)
      FMM=FM(1)
      SSo=So(1)
      YY=X(7)
      QN=(FN(1)*Q1/CC)**2
      QSG=Q1*Q1/G
      Y(1)=YO
      ELSE IF(I.EQ.2) THEN
      XX=L(1)
      XZ=X(9)
      YY=X(8)
      Y(1)=X(3)
      ELSE IF(I.EQ.3) THEN
      XX=L(2)
      XZ=0.
      BB=B(2)
      FMM=FM(2)
      SSo=So(2)
```

```
      YY=X(4)
      QN=(FN(2)*X(1)/CC)**2
      QSG=X(1)**2/G
      Y(1)=H2
      ELSE
      XX=L(3)
      XZ=0.
      BB=B(3)
      FMM=FM(3)
      SSo=So(3)
      YY=X(5)
      QN=(FN(3)*X(2)/CC)**2
      QSG=X(2)**2/G
      Y(1)=X(6)
      ENDIF
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
10    F(I)=YY-Y(1)
      F(5)=X(1)+X(2)-Q1
      E1=X(3)+(Q1/((B(1)+FM(1)*X(3))*X(3)))**2/G2
      F(6)=E1-X(4)-(X(1)/((B(2)+FM(2)*X(4))*X(4)))**2/G2
      F(7)=E1-X(5)-(X(2)/((B(3)+FM(3)*X(5))*X(5)))**2/G2
      F(8)=X(6)+(X(2)/((B(3)+FM(3)*X(6))*X(6)))**2/G2-Y4-X(2)**2/A4
      F(9)=.5*B(1)*(X(7)**2-X(8)**2)+FM(1)*(X(7)**3-X(8)**3)/3.+
     &Q1**2/G*(1./((B(1)+FM(1)*X(7))*X(7))-1./((B(1)+FM(1)*X(8))*
     &X(8)))
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1),X(9),B(3),FM(3),So(3),FN(3),L(3)
      COMMON /TRAS/B,FM,FN,So,L,G,G2,TOL,CC,QN,X,Y4,A4, QSG,H2,BB,
     &FMM,SSo,Q1,YO
      YY=Y(1)
      IF(YY.LT. 0.05) YY=.1
      P=BB+2.*SQRT(FMM*FMM+1.)*YY
      A=(BB+FMM*YY)*YY
      SF=QN*((P/A)**.666666667/A)**2
      DY(1)=(SSo-SF)/(1.-QSG*(BB+2.*FMM*YY)/A**3)
      RETURN
      END
```

The order of the unknowns in the array X(N) is defined by the list of names in the CHARACTER*3 UNK(N) array. There is a slight difference in the logic used in this program over that given in previous programs. The FUNCTION subprogram has been changed into a subroutine FUN that returns all the equation values whenever it is called. Thus, to defined the known equation vector for the Newton iteration, this subroutine needs only be called once. The Jacobian matrix is a defined column at a time within the DO 10 J=1,N loop. In this loop, the unknown is first incremented, then the subroutine FUN called again with the argument F1 array. Thereafter, all the partial derivatives in the J'th column are evaluated by dividing the difference between the elements of F1 and F by the increment in the unknown. To solve problems like this problem, it is generally necessary that some preliminary computations occur that provide reasonable guesses to the unknowns.

EXAMPLE PROBLEM 4.34

A channel consists of an upstream trapezoidal section with a length L_1 = 1500 ft and a bottom width b_1 = 10 ft, m_1 = 1, n_1 = 0.014, and S_{o1} = 0.0008, at which point there



is a smooth transition to a rectangular channel with b_2 = 12 ft, n_2 = 0.014, and S_{o2} = 0.008. A gate exists at L_2 = 1000 ft downstream from the break in grade. Determine the flow rates in the channel, the depth profiles, etc., that will occur with gate settings that vary and produce depths downstream from the gate from 2.4 to .025 ft.

Solution

There are two possible flow situations that can occur as shown in the two sketches. First, for higher gate settings, the flow will be critical at the end of the transition to the rectangular channel and a jump will occur from the S_2-GVF profile to the S_1-GVF profile. As the gate's height is reduced, the jump moves upstream farther but the flow rate will be controlled by a critical flow at the break in the grade until the gate closes sufficiently to cause the hydraulic jump to move up to the break in grade, at which time the jump disappears. To solve the problem, as long as the hydraulic jump occurs, the following three equations determine the following three unknowns in the upstream portion of the channel: Q, Y_1, Y_2 (depth in trapezoidal channel at its end, or the beginning of the transition.)

$$F_1 = H - Y_1 - Q^2/(2gA_1^2) = 0 \quad (1)$$

$$F_2 = Y_2 + Q^2/(2gA_2^2) - 1.5\left[(Q/b_2)^2/g\right]^{1/3} = 0 \quad (2)$$

$$F_3 = Y_1 - Y_{GVF}(Y_2) = 0 \quad (3)$$

The solution to these three equations produces Q = 391.84 cfs, Y_1 = 4.41 ft, Y_2 = 4.11 ft (and Y_{cr} = 3.21 ft) (solved using program SOLGVF).

As long as a hydraulic jump occurs, the following four equations can be solved to obtain the following four variables in the downstream portion of the channel: Y_{uj} (depth upstream of jump), Y_{dj} (depth downstream of jump), Y_3 (depth upstream from gate), x (position of jump).

$$F_1 = Y_{uj} - Y_{GVF}(Y_c) = 0 \quad (1)$$

$$F_2 = Y_{dj} - Y_{GVF}(Y_3) = 0 \quad (2)$$

$$F_3 = (Ah_c)_{uj} + Q^2/(gA_{uj}) - (Ah_c)_{dj} - Q^2/(gA_{dj}) = 0 \quad (3)$$

$$F_4 = Y_3 + Q^2/\left(2gA_3^2\right) - Y_4 - Q^2/\left(2gA_4^2\right) = 0 \quad (4)$$

With the flow rate established by the critical flow at the break in grade equal to Q = 391.84 cfs, the first part of the table below provides solutions for depth Y_4 downstream of the gate from 2.4 to

1.3 ft. (Obtain with program PRB4_12J.) Thereafter, the jump gets too close to the critical depth and the program fails to converge properly.

For depth downstream from the gate equal to 1.2 and smaller, the problem changes to the situation in which an M_2-GVF profile exists upstream from the break in grade and this connects into an S_1-GVF profile downstream from the break in grade. For this condition, the following five unknowns occur: Q, Y_1, Y_{t2} (depth at the beginning of the transition in a trapezoidal channel), Y_{r2} (depth at the end of the transition in a rectangular channel), and Y_3. The equations that govern are

$$F_1 = Y_1 - Y_{GVF}(Y_{t2}) = 0 \quad (1)$$

$$F_2 = Y_{r2} - Y_{GVF}(Y_3) = 0 \quad (2)$$

$$F_3 = Y_{t2} + \left[Q/A_{t2}\right]^2/(2g) - Y_{r2} - \left[Q/A_{r2}\right]^2/(2g) = 0 \quad (3)$$

$$F_4 = Y_3 + \left[Q/A_3\right]^2/(2g) - Y_4 - [Q/A_4^2]/(2g) = 0 \quad (4)$$

$$F_5 = H - Y_1 - \left[Q/A_1\right]^2/(2g) = 0 \quad (5)$$

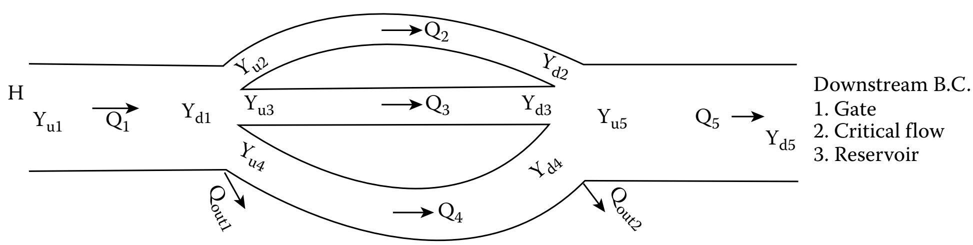
The solution to these equations is shown in the latter portion of the table below. (Solution obtained by program PRB4_12.)

Depths				Posit. J.	
Y_4 (ft)	**Y_{uj} (ft)**	**Y_{dj} (ft)**	**Y_3 (ft)**	**x (ft)**	
2.40	2.401	4.187	4.431	978.0	
2.30	2.401	4.187	4.671	954.8	
2.20	2.401	4.186	4.943	927.1	
2.10	2.401	4.186	5.255	894.0	
2.00	2.402	4.186	5.614	854.4	
1.90	2.397	4.193	6.031	807.8	
1.80	2.403	4.184	6.521	750.2	
1.70	2.401	4.187	7.101	681.8	
1.60	2.405	4.181	7.795	597.9	
1.50	2.410	4.174	8.636	495.0	
1.40	2.423	4.155	9.670	366.5	
1.30	2.479	4.076	10.959	200.8	
1.25	3.01	3.42	11.726	7.0	
	Q (cfs)	**Y_1**	**Y_{2t}**	**Y_{2t}**	**Y_3**
1.20	388.521	4.425	4.311	3.800	12.398
1.15	378.193	4.469	4.678	4.324	12.717
1.10	365.954	4.516	4.934	4.649	12.949
1.00	338.390	4.607	5.284	5.078	13.278
0.90	308.447	4.689	5.523	5.368	13.509
0.80	277.008	4.758	5.699	5.583	13.685
0.70	244.499	4.818	5.835	5.749	13.822
0.60	211.158	4.868	5.942	5.880	13.931
0.50	177.141	4.909	6.026	5.984	14.017
0.40	142.555	4.942	6.091	6.064	14.085
0.30	107.485	4.968	6.139	6.124	14.136
0.20	71.998	4.986	6.173	6.167	14.172
0.10	36.152	4.996	6.193	6.192	14.193
0.05	18.111	4.999	6.198	6.198	14.198
0.025	9.064	5.000	6.200	6.199	14.200

4.14 GVF PROFILES IN PARALLEL CHANNELS

In the previous section, methods for solving flow rates and depths under gradually varied flow conditions in several channels that branched at a common junction were described. Especially in natural channels, such as rivers and streams, when a main channel divides into two or more channels these two or more channels will recombine downstream into a single channel again. Upon combining into a single channel at a downstream junction, a system of parallel channels is created. If the division of a main channel is into two channels that join downstream again, then the resulting flow is commonly referred to as an "island flow" because it leaves an island between the two parallel channels. In this section, methods will be described for solving the flow rates and the depths throughout a system of parallel channels.

In formulating the problem of flow in parallel channels, the following will be assumed: (1) The division of a single main channel will occur at a common junction into p parallel channels. p is the number of parallel channels that the main channel divides into, and therefore p will be 2 or larger. The number of channels involved will be $n = p + 2$. (2) All of the channels that divide at the upstream junction join at a common downstream junction, thus forming a single main channel again. (3) A reservoir with a known water surface elevation H supplies the upstream main channel, and the downstream main channel will also generally contain a gradually varied flow that is controlled by (a) a gate, (b) a break in grade (after a transition) so that the critical flow occurs here, or (c) the flow will be in a downstream body of water (reservoir) with a known water surface elevation H_d. Such a parallel system of channels is shown in the sketch below, in which the division is into three parallel channels, i.e., $p = 3$. However, the equations and solution methods will be more general for $p = 2, 3$, to any number.



A parallel channel system (five channels are shown with three parallel, but this can be generalized to n channels).

While only parallel channels are described that have common upstream and downstream junctions, e.g., two junctions exist in the system, the methods can be readily extended to parallel systems of channels in which intermediate junctions of two or more channels may occur that join further downstream with other channels, such that the number of junctions can expanded into any number more than two. In this parallel system of channels, an outflow of Q_{out1} will be allowed at the upstream junction of the system, and another outflow Q_{out2} will be allowed at the downstream junction of the system. If either Q_{out1} or Q_{out2} is negative, then inflow into that junction will take place. Thus, the flow in the last channel of the system will equal the flow in the first channel minus the junction outflows, or $Q_n = Q_1 - Q_{out1} - Q_{out2}$.

It will be assumed that the geometry of all channels, their roughness coefficients, and bottom slopes are known. The flow rates and depths through the channel system are unknown. Thus, in a system involving n channels, there will be n unknown flow rates, n unknown upstream depths, and n unknown downstream depths, or a total of 3n unknowns. However, since there is no need of including the equation above that gives the flow rate Q_n in the last channel after the parallel channel has recombined, Q_n will not be considered an unknown. Furthermore, the minor loss from one channel flowing into another channel with a different velocity is typically taken equal to the difference between the velocity heads in these two channels, especially if the velocity in the upstream channel is larger than that in the downstream channel. Therefore, in applying the specific energy between each of the branch channels, and the channel they join at the second junction, the depths here will be taken as equal, or $Y_{d2} = Y_{u5}$, $Y_{d3} = Y_{u5}$, $Y_{d4} = Y_{u5}$, etc. (n = 5 for our five channels example), in which the Y_{di}'s are the downstream depths in the parallel channels, and Y_{un} is the upstream depth in the last channel of the system. You might ask, "Why not also assume all depths at the upstream junction are equal, i.e., let $Y_{u2} = Y_{u3} = Y_{u4} = \cdots = Y_{d1}$?" The answer is that when dealing with a subcritical flow, as is the case for the problems we are solving, the control comes from downstream. Therefore, at the downstream junction, there will generally be smaller differences in velocity heads than at the upstream junction. If there is a large difference in the velocity head between a channel at the downstream junction, then the upstream channel will have the larger velocity head. As mentioned above, the minor or local loss resulting from this difference will be about equal to this difference in velocity heads; thus the depths will be approximately equal. Therefore, making the depths equal at the downstream junction provides somewhat of an implicit means for obtaining the loss coefficients, and by equating depths, we remove the burden from ourselves of estimating what the minor loss coefficients are because they do not appear in the equations being solved.

Again, rather than including these simple equations in the system of equations being solved, the number of unknowns will be reduced by p. By using the above simple equations to reduce the number of unknowns, a parallel system involving n channels will have 2n + 1 unknowns. These unknowns are: (Q_i, i = 1 to n − 1), (Y_{ui}, i = 1 to n), Y_{d1}, and Y_{dn}. For the five channel system shown in the above sketch, the following 11 variables are unknown: Q_1, Q_2, Q_3, Q_4, Y_{u1}, Y_{u2}, Y_{u3}, Y_{u4}, Y_{u5}, Y_{d1}, and Y_{d5}.

To solve these 2n + 1 unknowns, it will be necessary to have 2n + 1 independent equations. These equations are

$$F_1 = H - Y_{u1} - (1+K_e)\frac{Q_1^2}{2gA_{u1}^2} = 0 \quad \text{(Energy at entrance, 1 equation)}$$

$$F_i = Y_{d1} + \frac{Q_1^2}{2gA_{d1}^2} - Y_{ui} - (1+K_{Li})\frac{Q_i^2}{2gA_{ui}^2} = 0 \quad (\text{for } i = 2 \text{ to } n-1)\ \text{(Energy equations at upstream junction, } P = n-2 \text{ equations)}$$

$$F_j = Y_{ui} - Y_{ode}(Y_{di}) = 0 \quad (\text{for } i = 1 \text{ to } n,\ j = i+n-1)$$

(ODE equations from solving GVFs from end to beginning of each channel, n equations)
(the Y_{di}'s in parenthesis denote the starting depth for the GVF solutions)

$$F_{2n} = Q_1 - Q_{out1} - \sum Q_i = 0 \quad \text{(Upstream junction continuity, 1 equation)}$$

$$F_{2n+1} = Y_{dn} + \frac{Q_n^2}{2gA_{n2}^2} - Y_{dn} - \frac{Q_n^2}{2gA_G^2} = 0 \quad \text{(for gate downstream)}$$

or

$$F_{2n+1} = Q_n^2 T_{end} - gA_{end}^3 = 0 \quad \text{(Critical flow at downstream end)}$$

or

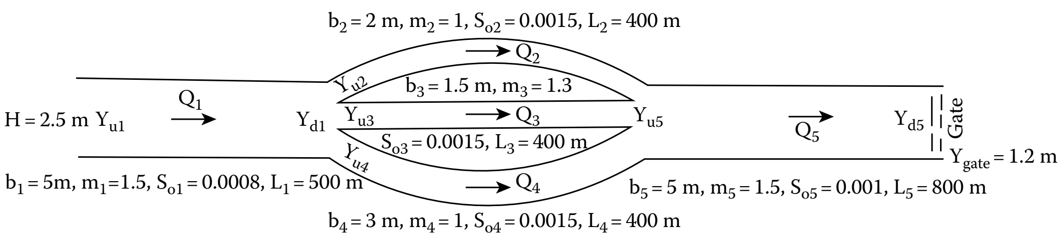
$$F_{2n+1} = Y_{dn} - Y_{reser} = 0 \quad \text{(Flow into reservoir)}$$

In solving the ODE equations, the assumption is that if the gradually varied flow passes through a transition, then the term involving $\partial A/\partial x$ for nonprismatic channels will be included. The above equations total $2n + 1$ in number, the same number as the unknown variables. In using the above equations, it is important to remember the simple equations that were eliminated from the system of equations. Thus, in place of Q_n its equivalent $Q_1 - Q_{out1} - Q_{out2}$ is substituted, and in place of the downstream depths y_{di}'s, i = 2 to n − 1 in the parallel channels, the upstream depth Y_{un} in the last channel is substituted. The last equation F_{2n+1} will vary depending upon the condition at the downstream end of the last channel. The three equations given above as possibilities for this boundary condition include (1) a gate just downstream from a transition at the end of this last channel, (2) a break in grade to a steep channel just downstream from a transition at the end of this last channel, and (3) the last channel discharges into a reservoir, so that after considering that the velocity head in the channel will be dissipated in flowing into the reservoir, the channel depth will be maintained equal to the known depth in the reservoir. Other equations can be written in place of these for other downstream boundary conditions.

The solution to the system of equations defined above can proceed using the Newton method, as has been utilized previously, to solve ODEs in combination with algebraic equations. The implementation of such a solution in a computer program can be accomplished by modifying the code used to solve the branched system of channels. First, the number of equations will need to be changed from 3n to 2n + 1. Thus, the dimensions of some arrays will be different also. Second, the subroutine that evaluates the equations needs to be modified so that the correct statements are accessed depending upon the equation number that is being passed as an argument in its call. Third, proper consideration must be given to the fact that some simple algebraic equations have been eliminated from the system being solved, and appropriate substitutions must be made. The computer program, whose listing is given for the follow example problem, contains these modifications.

EXAMPLE PROBLEM 4.35

A main channel with $b_1 = 5$ m, $m_1 = 1.5$, $S_{o1} = 0.0008$, and a length $L_1 = 500$ m divides into three parallel channels with the following: $b_2 = 2$ m, $m_2 = 1.0$, $S_{o2} = 0.0015$, $L_2 = 400$ m, $b_3 = 1.5$ m, $m_3 = 1.3$, $S_{o3} = 0.0015$, $L_3 = 400$m, $b_4 = 3.0$ m, $m_4 = 1.0$, $S_{o4} = 0.0015$, and $L_4 = 400$ m. These channel join again into a fifth channel with $b_5 = 5.0$ m, $m_5 = 1.5$, $S_{o5} = 0.001$, $L_5 = 800$m. All channel have a Manning's roughness coefficient, n = 0.014. The upstream channel is supplied by a reservoir with a water surface elevation of 2.5 m above the bottom of the channel, and the last downstream channel has a gate at its downstream end after a 5 m long transition to a rectangular section with a 5 m width that produces a depth of 1.2 m downstream from it. All minor loss coefficients equal 0.08.



Solution

To solve this problem 2n + 1 = 11 equations will be solved simultaneously. Of these equations, five will be ODEs and six will be algebraic. Using the computer code listed below, the problem can be solved using the following input:

```
1 5
1 5.0 1.5 .014 .0008 500 0 .08 0
1 2.0 1. .014 .0015 400 0 .08 5
1 1.5 1.3 .014 .0015 400 0 .08 5
1 3.0 1.0 .014 .0015 400 0 .08 5
1 5.0 1.5 .014 .001 800 0 .08 0
5 0. -1.5 1.2 1
.000001 .001 2.5 9.81
4 1 2 3 4 0.
35. 10. 12.5 12.5 2.18 2.3 2.3 2.3 2.5 2.18 2.736
```

The first line of this input consists of NJ = 1, which is the number of junctions minus one since the continuity equation is not written at the downstream junction under the assumption that the downstream depths at all channels that connect into it have depths equal to its upstream depth, and NO = 5, which is the number of channels. The next five lines of input contain the information about each of the five channels as follows: (1) an integer ITYP that is 1 for a trapezoidal channel and 2 for a circular channel, (2) the bottom width B for a trapezoidal channel or the diameter for a circular channel, (3) the side slope FM for a trapezoidal channel and 0 (ignored) for a circular channel, (4) Manning's roughness coefficient FN for this channel, (5) the bottom slope SO for this channel, (6) the length L of this channel, (7) the change in bottom elevation DZ of this channel and the channel that connects at its downstream end, (8) the minor loss-coefficient KL at the entrance of this channel, and (9) the channel number downstream NODOWN whose upstream depth equals the downstream depth in this channel. If the downstream depth of this channel is not equal to the upstream depth of the channel it connects to, then NODOWN must be given as zero. For this example, there are no intermediate channels that further branch, and therefore only the first and the last channel have NODOWN for them assigned zero. The next line contains (1) LT = length of transition at the end of the last channel, i.e., before a gate, etc., that might control as the downstream boundary condition, (2) the change in the bottom width through this transition, (3) the change in the side slope (for a Trap. Ch.) through this transition, (4) the depth of the control, i.e., the depth downstream from the gate, and the ICTL that defines the type of boundary condition to be used at the downstream end of the last channel as described in the comments in the program. The next lines of input, consisting of NJ in number contain, the number of channels at this junction followed by their numbers (the numbers are established by the order of the channel property lines above), and the point discharge QOUT at this junction. Since the last junction is not included in the continuity and the energy equation, no point outflow can be specified at the junction with the last channel. The last line of input consists of the guesses for the unknowns that are to be solved. The order of the unknowns are the flow rates in the first NO-1 channels (the flow rate in the last channel is not an unknown, but assumed equal to $Q_1 - \Sigma QOUT$), followed by the **upstream** depth in the NO channels, and these are followed by the downstream depths in the first channel, any intermediate channel with NODOWN(I)=0, and the final channel's downstream depth, which is controlled by the downstream boundary condition.

The solution produced is

Solution to the five channels at a junction

No.	b	m	n	S_o	L	d_z	Y_u	Y_d	Q	Y_o
1	5.00	1.50	0.014	.000800	500.0	0.00	2.210	2.370	42.18	2.092
2	2.00	1.00	0.014	.001500	400.0	0.00	2.522	3.105	12.72	1.487
3	1.50	1.30	0.014	.001500	400.0	0.00	2.520	3.105	13.62	1.568
4	3.00	1.00	0.014	.001500	400.0	0.00	2.519	3.105	15.84	1.406
5	5.00	1.50	0.014	.001000	800.0	0.00	3.105	3.648	42.18	1.973

The normal depths have been added as a final column to the computer output. Note that the gate at the downstream end of channel # 5 has increased depths throughout the system of the channel.

If a flow were to enter a very long channel with the dimensions of channel # 1 from the reservoir with H = 2.5 m, the normal depth and the flow rate would be $Y_o = 2.152$ m and $Q_o = 44.12\,m^3/s$, respectively. (Program SOLPAR will also solve Prob.)

FORTRAN listing of program to solve parallel channel systems, SOLPARG.FOR.
(This program uses FUNCTION AR and SUBROUTINE DYX that are identical to those given in the list of program SOLBRA with the exception that their declarations, and COMMON statements are as given in this listing below.)

```
      PARAMETER (N2=4,N=12,M=25)
      LOGICAL*2 IPERM,NTRAN
      REAL F(M),F1(M),D(M,M),FN(N),SO(N),L(N),B(N),FM(N),
     &DZ(N),KL(N),X(M),QOUT(N2),LT
      INTEGER*2 INDX(M),ITYP(N),ICTL
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NJ,NO,NOM,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,DZ,
     &KL,QL,DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QOUT,QOUTS,QN,Q2G,
     &BB,FMM,IPERM,NTRAN,JI,NI(N2+1),JN(M),NODOWN(N)
      DATA IN,IOUT/2,3/
      IPERM=.FALSE.
      NTRAN=.TRUE.
C NJ=No. of Jun. minus 1 (jun. to last channel not counted),
c     NO=No. channels
C ITYP = 1 for trapezoidal channels, otherwise channel
C    is circular.
C ICTL = 1,2 OR 3 for types of downstream controls,
C    gate=1,critical=2,
C NODOWN is the downstr. chan. whose upstr. depth equals this
C  channel's downst. depth. If NODOWN=0 then downstr. depth
C  for this channel is unknown.
C reser=3. YG is depth behind gate for gate; = 0 for
C  critical;res. depth if reservoir.  DB and DFM are
C  changes (+ or -) across transition of b and m.
C IN UNKNOWN VECTOR X (NO-1) Q's COMES 1ST; (NO) UPSTREAM
C  DEPTHS NEXT & NJ DOWNSTREAM DEPTHS (for 1st upstr.
C   channel at each jun.+last ch.)
C i.e. Q(I)=X(I) for I=1 to NO-1;YU(I)=X(I+NO-1) for I=1
C  to NO;
C YD(1)=X(NEQS-1 & YD(2)=X(NEQS). Note other YD's = YU(NO).
      READ(IN,*) NJ,NO,(ITYP(I),B(I),FM(I),FN(I),SO(I),L(I),
     &DZ(I),KL(I),NODOWN(I),I=1,NO),LT,DB,DFM,YG,ICTL
      NOM=NO-1
      NEQS=2*NO+NJ
      NO2=NEQS-NJ-1
      READ(IN,*) TOL,ERR,H,G
      NI(1)=0
      II=1
      QOUTS=0.
      DO 2 J=1,NJ
      READ(IN,*) JI,(JN(I),I=II,II+JI-1),QOUT(J)
      QOUTS=QOUTS+QOUT(J)
```

```
      II=II+JI
2     NI(J+1)=II-1
      READ(IN,*)(X(I),I=1,NEQS) ! unks:(NO-1 Q'S, Yu,Yd
      IF(G.GT. 30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      QL=X(1)-QOUTS
      DO 10 I=1,NO
10    KL(I)=(KL(I)+1.)/G2
      NCT=0
20    CALL FUN(F)
      DO 30 J=1,NEQS
      DX=.015*X(J)
      IF(J.EQ.1) QL=QL+DX
      X(J)=X(J)+DX
      CALL FUN(F1)
      DO 25 I=1,NEQS
25    D(I,J)=(F1(I)-F(I))/DX
      X(J)=X(J)-DX
      IF(J.EQ.1) QL=QL-DX
30    CONTINUE
      CALL SOLVEQ(NEQS,1,M,D,F,1,DD,INDX)
      DIF=0.
      DO 40 I=1,NEQS
      X(I)=X(I)-F(I)
40    DIF=DIF+ABS(F(I))
      QL=X(1)-QOUTS
      NCT=NCT+1
      WRITE(*,101) NCT,DIF,(X(I),I=1,NEQS)
101   FORMAT(' NCT =',I3,' SUM =',E12.5,/(8F10.3))
      IF(NCT.LT.30 .AND. DIF.GT.ERR) GO TO 20
      WRITE(IOUT,100) NO
100   FORMAT(' Solution to',I3,' Channel at Junction',/,1X,65('-'),/,
     &' No    b   m    n    So    L   dz    Yu',' Yd    Q',/,1X,65('-'))
      DO 50 I=1,NO
      IF(I.EQ.NO) THEN
      YD=X(NEQS)
      QQ=QL
      ELSE IF(I.EQ.1) THEN
      III=NO2+1
      YD=X(III)
      QQ=X(1)
      ELSE
      IF(NODOWN(I).EQ.0) THEN
      III=III+1
      YD=X(III)
      ELSE
```

```
      YD=X(NOM+NODOWN(I))
      ENDIF
      QQ=X(I)
      ENDIF
50    WRITE(IOUT,110) I,B(I),FM(I),FN(I),SO(I),L(I),DZ(I),
     &X(I+NOM),YD,QQ
110   FORMAT(I3,F7.2,F5.2,F6.3,F8.6,F7.0,F6.2,2F7.3,F8.2)
      END
      SUBROUTINE FUN(F)
      PARAMETER (N2=4,N=12,M=25)
      EXTERNAL DYX
      LOGICAL*2 IPERM,NTRAN
      REAL F(M),FN(N),SO(N),L(N),B(N),FM(N),DZ(N),KL(N),X(M),Y(1),
     &DY(1),W(1,13),XP(1),YP(1,1),LT,QOUT(N2)
      INTEGER*2 ITYP(N),ICTL
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NJ,NO,NOM,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,DZ,
     &KL,QL,DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QOUT,QOUTS,QN,Q2G,
     &BB,FMM,IPERM,NTRAN,JI,NI(N2+1),JN(M),NODOWN(N)
      H1=-.05
      HMIN=.0000001
      IPERM=.FALSE.
      II=1
      F(II)=H-X(NO)-KL(1)*(X(1)/AR(1,X(NO)))**2
      DO 10 I=1,NJ
      I1=NI(I)+1
      I2=NI(I+1)
      II=II+1
      JI=JN(I1)
      II1=II
      F(II1)=X(JI)-QOUT(I)
      EN=X(NO2+I)+(X(JI)/AR(JI,X(NO2+I)))**2/G2
      DO 10 J=I1+1,I2
      JI=JN(J)
      II=II+1
      F(II)=EN-X(NOM+JI)-KL(JI)*(X(JI)/AR(JI,X(NOM+JI)))**2-DZ(JI)
10    F(II1)=F(II1)-X(JI)
      DO 20 I=1,NO
      JI=I
      II=II+1
      XX=L(I)
      XZ=0.
      IPERM=.TRUE.
      IF(I.EQ.1) THEN
      QQ=X(1)
      III=NO2+1
      Y(1)=X(III)
      IF(ICTL.EQ.2) Y(1)=1.2*Y(1)
      ELSE IF(I.EQ.NO) THEN
      XX=XX+LT-1.E-5
```

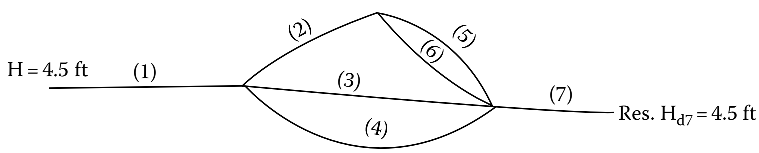
```
      QQ=QL
      Y(1)=X(NEQS)
      ELSE
      IF(NODOWN(I).EQ.0) THEN
      III=III+1
      Y(1)=X(III)
      ELSE
      Y(1)=X(NOM+NODOWN(I))
      ENDIF
      QQ=X(I)
      ENDIF
      QN=(FN(I)*QQ/CC)**2
      Q2G=QQ**2/G
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
20    F(II)=X(NOM+I)-Y(1)
      II=II+1
      NTRAN=.FALSE.
      BB=B(NO)+DB
      FMM=FM(NO)+DFM
      IF(ICTL.EQ.1) THEN
      F(II)=X(NEQS)+(QL/AR(NO,X(NEQS)))**2/G2-YG-KL(NO)*
     &(QL/AR(NO,YG))**2
      NTRAN=.TRUE.
      ELSE IF(ICTL.EQ.2) THEN
      AA=AR(NO,X(NEQS))
      F(II)=QL**2*TOPW-G*AA**3
      ELSE
      F(II)=X(NEQS)-YG
      ENDIF
      NTRAN=.TRUE.
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      PARAMETER (N2=4,N=12,M=25)
      LOGICAL*2 IPERM,NTRAN
      REAL FN(N),SO(N),L(N),B(N),FM(N),DZ(N),KL(N),X(M),Y(1),DY(1),
     &LT,QOUT(N2)
      INTEGER*2 ITYP(N),ICTL
      COMMON /TRAS/NJ,NO,NOM,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,DZ,
     &KL,QL,DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QOUT,QOUTS,QN,Q2G,
     &BB,FMM,IPERM,NTRAN,JI,NI(N2+1),JN(M),NODOWN(N)
      IF(XX.LE.L(JI)) THEN
      NTRAN=.TRUE.
      DAX=0.
      ELSE
      NTRAN=.FALSE.
      FLEN=(XX-L(JI))/LT
      BB=B(JI)+FLEN*DB
      FMM=FM(JI)+FLEN*DFM
      DAX=Q2G*Y(1)*(DB+Y(1)*DFM)/LT
```

```
ENDIF
AA=AR(JI,Y(1))
A3=AA**3
SF=QN*(ABS(PERM/AA)**.666666667/AA)**2
DY(1)=(SO(JI)-SF+DAX/A3)/(1.-Q2G*TOPW/A3)
RETURN
END
```

EXAMPLE PROBLEM 4.36

An upstream channel divides into three parallel channels, and at a short distance after the first junction, the second channel divides further into two parallel channels, and all channels combine thereafter into a single channel, as shown in the sketch below. Solve the depths and the flow rates in this parallel system of channels if their properties are as given in the table below. The upstream reservoir has a water surface elevation 5 ft above the channel bottom. Channel 7 discharges into a reservoir whose water surface elevation is 4.5 ft above the channel bottom.

No.	B (ft)	m	n	S_o	L (ft)
1	14	1.5	0.014	0.0006	1000
2	8	1.0	0.014	0.001	400
3	6	1.0	0.014	0.0008	1500
4	6	0	0.014	0.00085	1800
5	5	1.0	0.014	0.0006	1200
6	4	1.0	0.014	0.0006	1200
7	12	0	0.014	0.001	1000



Solution

Again, if the assumption is used that the depths of the channel flowing into a downstream channel equals the upstream depth of that channel, then $Y_{5d} = Y_{6d} = Y_{3d} = Y_{4d} = Y_{7u}$, and $Q_7 = Q_1$. Therefore the unknowns are Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, Y_{u1}, Y_{u2}, Y_{u3}, Y_{u4}, Y_{u5}, Y_{u6}, Y_{u7}, Y_{d1}, Y_{d2}, Y_{d7} (in which the subscript u denotes the upstream, d denotes the downstream and the number of the channel.) To solve these 16 unknowns, 16 equations are needed. These equations are

$$F_1 = H - Y_{u1} - (Q_1/A_{u1})^2/(2g) = 0$$

$$E_{d1} = Y_{d1} + (Q_1/A_{d1})^2/(2g)$$

$$F_2 = E_{d1} - Y_{u2} - (Q_2/A_{u2})^2/(2g) = 0$$

$$F_3 = E_{d1} - Y_{u3} - (Q_3/A_{u3})^2/(2g) = 0$$

$$F_4 = E_{d1} - Y_{u4} - (Q_4/A_{u4})^2/(2g) = 0$$

$$E_{d2} = Y_{d2} + (Q_2/A_{d2})^2/(2g)$$

$$F_5 = E_{d2} - Y_{u5} - (Q_5/A_{u5})^2/(2g) = 0$$

$$F_6 = E_{d2} - Y_{u6} - (Q_6/A_{u6})^2/(2g) = 0$$

$$F_7 = Q_1 - Q_2 - Q_3 - Q_4 = 0$$

$$F_8 = Q_2 - Q_5 - Q_6 = 0$$

$$F_9 = Y_{u1} - Y_{1,ode}(Y_{d1}) = 0$$

$$F_{10} = Y_{u2} - Y_{2,ode}(Y_{d2}) = 0$$

$$F_{8+i} = Y_{ui} - Y_{i,ode}(Y_{u7}) = 0 \quad (i = 3, \ldots 6)$$

$$F_{15} = Y_{u7} - Y_{7,ode}(Y_{d7}) = 0$$

$$F_{16} = Y_{d7} - H_{d7} = Y_{d7} - 4.5 = 0$$

The input to program SOLPARG to solve this problem consists of

```
2 7
1 14 1.5 .014 .0006 1000 0. 0.  0
1 8 1    .014 .001 400  0.  0. 0
1 6 1    .014 .0008 1500 0. 0. 7
1 6 0    .014 .00085 1800 0. 0. 7
1 5 1    .014 .0006 1200  0. 0. 7
1 4 1    .014 .0006 1200  0. 0. 7
1 12. 0. .014 .001  1000. 0. 0. 0
0. 0. 0. 4.5 3
.000001 .001 4.5 32.2
4 1 2 3 4 0.
3 2 5 6 0.
400 150 150 100 75 75 4.3 4.4 4.4 4.4 4.42 4.42 4.43 4.39 4.41 4.5
```

and the solution therefrom is

Solution to the seven channels at a junction

No.	b	m	n	S_o	L	d_z	Y_u	Y_d	Q
1	14.00	1.50	0.014	0.000600	1000.0	0.00	4.031	3.863	443.97
2	8.00	1.00	0.014	0.001000	400.0	0.00	4.119	4.407	207.26
3	6.00	1.00	0.014	0.000800	1500.0	0.00	4.192	5.109	151.38
4	6.00	0.00	0.014	0.000850	1800.0	0.00	4.209	5.109	85.34
5	5.00	1.00	0.014	0.000600	1200.0	0.00	4.529	5.109	110.41
6	4.00	1.00	0.014	0.000600	1200.0	0.00	4.533	5.109	96.85
7	12.00	0.00	0.014	0.001000	1000.0	0.00	5.109	4.500	443.97

4.15 SOLUTIONS TO SPATIALLY VARIED FLOWS

Problems associated with spatially varied flows are relatively common. Channel systems generally have turnouts periodically along their lengths. The common design is that downstream from these larger turnouts, the channel has its size reduced appropriately for the smaller downstream flow rate to reduce the cost. To prevent the downstream channel from being overtopped should outflow from the turnouts be shut off, side weirs, or spillways are built into the system so that either water is wasted, or stored in its reservoirs off the side of the channel. Water leaving such a side weir creates a spatially varied outflow length. Likewise, if the storage from the side reservoir reenters the channel over some length rather than at a single point, it constitutes a spatially varied inflow problem. Often, water entering a channel from a lake or reservoir may enter the channel from an overflow structure that runs parallel to the channel for some distance. Such structures create spatially varied inflow channel problems. Water leaving the gutter alongside a roadway into a storm grate creates a spatially varied outflow problem in the gutter, and possibly a lateral inflow problem into the storm drain if it operates as an open channel. Rainfall flowing over a road crest or parking lot to a side gutter represents a spatially varied inflow problem to the gutter flow. The collection channel that receives water from the spillway at the Hoover dam is an example of a spatially varied inflow problem in which the channel downstream from this lateral inflow is steep. If these flows are steady state, then they can be handled by numerically solving the general ODE for a gradually varied flow, Equation 4.6.

4.15.1 Outflow from Side Weirs

From the viewpoint of numerically solving spatially varied flow problems, the major difference to that described previously is to include the terms in the numerator of Equation 4.6 that contains the lateral inflow/outflow q*. However, often before a spatially varied flow problem can be solved, considerable work is required to set up the appropriate boundary conditions for the problem governed by the GVF equation. To begin this discussion, consider a length of channel that contains a side weir such as shown in the sketch below. To simplify, conceptionalize what may occur. It will be assumed that the channel is prismatic across the side weir, and therefore the differential equation that describes how the depth varies across the side weir is (Equation 4.3 without the term containing $\partial A/\partial x$.)

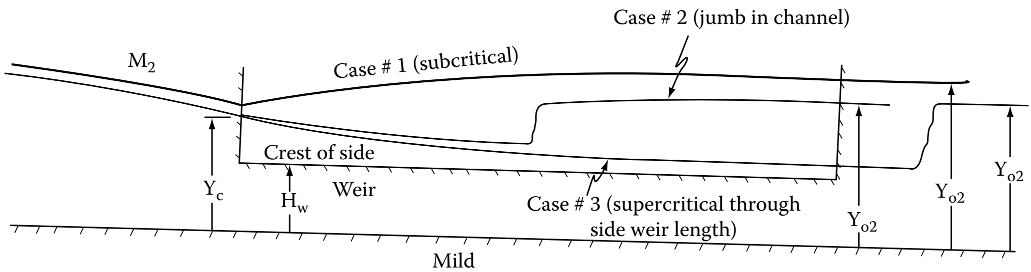
$$\frac{dY}{dx} = \frac{S_0 - S_f + Qq_0^*/(gA^2)}{1 - F_r^2} \tag{4.13}$$

in which q_o^* is a positive magnitude for the lateral outflow, and for a weir along the side of a channel, the equation will depend upon the depth of water above the weir crest. The equation giving this lateral outflow is of the form

$$q_0^* = \frac{2}{3}C_d\sqrt{2g}(Y - H_w)^{1.5} \tag{4.14}$$

If the flow is subcritical, e.g., the denominator of Equation 4.13 is positive, then the effect of the lateral outflow will be to increase the depth across the side weir above what it would be without the outflow because the term that contains q_o^* is positive. If the flow is supercritical, the lateral outflow from the side weir will cause the depth to be less than otherwise. If the downstream channel is mild, it will establish the depth at the downstream end of the side weir, and depending upon the amount of water leaving and other conditions, the depth at the upstream end of the side weir may approach the critical depth, or be well above the critical depth. If it is well above critical depth, then the upper water surface profile shown in the sketch below will exist across the entire side weir, e.g., the depth will increase and the flow will be subcritical.

Should the depth become critical at the beginning of the side weir, then one of the other profiles shown on the sketch might occur. Both of these possibilities assume that over the first small portion of the side weir, the flow becomes supercritical such that the denominator of Equation 4.13 is negative. In real flows



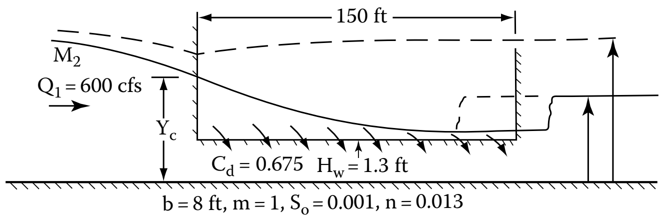
the critical depth actually occurs about five depths upstream from the free overall, so that in practice, the supercritical depth may occur at or before the beginning of the lateral outflow section, even though from a theoretical viewpoint this will not occur. For this case, the depth decreased across the side weir. However, the downstream conditions may not permit a supercritical flow to occur across the entire length of the side weir, in which case a hydraulic jump within the side weir length will take the depth from its supercritical value to its conjugate depth value above the critical depth. Since the flow rate varies from point to point across the weir portion of the channel, and conjugate depths depend upon flow rates, the position of the hydraulic jump depends upon the solution to the entire problem. The unknowns of the problem are (1) Y_{o2}—the depth in the channel downstream from the side weir (which will be the normal depth unless the downstream portion of the channel is short, or this flow is affected by a downstream control), (2) Y_{1w}—the depth upstream from the jump within the weir length, (3) Y_{2w}—the depth immediately downstream from the jump, (4) x—the distance from the beginning of the weir to the hydraulic jump. The four equations needed to solve these unknowns are: (1) Manning's or Chezy's uniform flow equation applied to the downstream channel, and based on the flow rate in this downstream channel, (2) the hydraulic jump equation $M_1 = M_2$, and (3) the ODE equation (Equation 4.13) applied to the length upstream from the jump starting with the critical flow at the beginning of the weir, and (4) the ODE equation (Equation 4.13) applied to the length downstream from the jump starting with the uniform depth at the end of the weir and solving the problem upstream. While it is possible to solve the above system of ODEs and algebraic equations, simultaneously using the Newton method and the trial-and-error procedure is instructive because it gives greater insight into the host of possibilities that exist, and therefore will be used in the following discussion.

Should the hydraulic jump occur downstream of the side weir, then x is no longer unknown, and the problem becomes a little simpler. For this case Equation 4.13 can be solved across the entire length of the side weir. The flow rate in the channel downstream from the weir is now known from this solution, and the location of the hydraulic jump can proceed without concerns for a varying flow rate with x. On the other hand, we will not know if the critical depth will occur at the upstream end of the side weir until the discharge from the side weir is known, and if the upstream depth is substantially above critical, then the depth will increase across the entire length as shown in the above sketch as case # 1.

Obviously, to effectively obtain such a solution, it is most desirable that a computer solution of Equation 4.13 be readily obtained. Unless one is able to do an excellent job in guessing the case of the flow, and other appropriate trial values, a relative large number of solutions will be required until eventually the correct solution is obtained.

EXAMPLE PROBLEM 4.37

A channel with a bottom width of b = 8 ft, a side slope m = 1, a bottom slope $S_o = 0.001$, and a roughness coefficient n = 0.013 contains a side weir with a crest 1.3 ft above the channel bottom. The side weir is 150 ft long and has a discharge coefficient equal to $C_d = 0.675$ (the value is larger than for a sharp-crested weir because the weir is rounded). If the flow rate in the channel upstream from the side weir is $Q_1 = 600$ cfs, determine the water surface profiles in the channel and the amount of water that is discharged from the side of the channel.

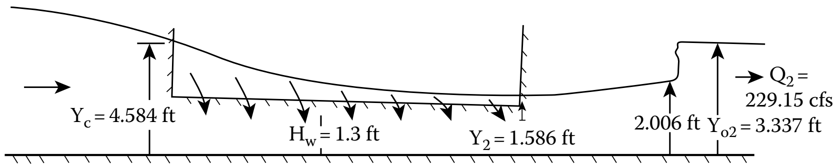


Solution

Since the weir crest is relatively low, we might begin the analysis of this problem by assuming that the critical flow occurs at the entrance to the side weir, and that at least a part of the way through the weir length of the channel the flow is supercritical. Therefore, the critical depth in this channel associated with 600 cfs is computed and found to be $Y_{c1} = 4.584$ ft. The water surface profile computations at the beginning of the weir is begun at a depth of 4.45 ft, a small amount below the critical depth. The solution to this profile, which was obtained by the program listed below, is provided in the first table below. From this solution, we might guess that if a hydraulic jump occurs within the length of the channel containing the side weir, that it will be near the end. Therefore, assume that a flow rate of 220 cfs remains in the downstream channel. From a solution of Manning's equation the normal depth in this channel associated with a flow rate of 220 cfs is 3.262 ft. Therefore, starting at the downstream end of the weir, the GVF profile is solved upstream giving the results in the second table that follows.

x (ft)	Y (ft)	Q (cfs)	q_o^* (cfs/ft)	M (ft³)
.000	4.450	600.000	20.189	310.382
10.000	2.797	465.995	6.612	261.925
20.000	2.431	411.215	4.344	235.517
30.000	2.208	373.874	3.124	215.697
40.000	2.056	346.388	2.373	200.041
50.000	1.946	325.141	1.876	187.213
60.000	1.864	308.111	1.530	176.396
70.000	1.801	294.060	1.280	167.064
80.000	1.751	282.183	1.095	158.859
90.000	1.712	271.936	0.954	151.531
100.000	1.680	262.935	0.846	144.902
110.000	1.654	254.907	0.760	138.841
120.000	1.632	247.649	0.692	133.249
130.000	1.614	241.009	0.636	128.053
140.000	1.599	234.872	0.591	123.195
150.000	1.586	229.150	0.553	188.629

x (ft)	Y (ft)	Q (cfs)	q_o^* (cfs/ft)	M (ft³)
150.000	3.262	220.000	9.924	95.048
149.750	3.239	222.459	9.750	95.508
149.500	3.215	224.874	9.567	95.970
149.250	3.189	227.242	9.375	96.432
149.000	3.162	229.560	9.173	96.894
148.750	3.132	231.826	8.958	97.357
148.500	3.101	234.037	8.728	97.818
148.250	3.067	236.188	8.479	98.277
148.000	3.028	238.273	8.205	98.734
147.750	2.985	240.286	7.896	99.186
147.500	2.932	242.214	7.529	99.631
147.250	2.861	244.035	7.041	100.066
147.000	2.656	245.628	5.701	100.666



By trial, it is necessary to match from the above two solutions: (1) the momentum functions given in the last columns, (2) the position x in the first columns, and (3) the flow rate Q in the third columns. From the above tables, it is clear that the hydraulic jump will occur beyond the end of the weir. The normal depth in this channel for a flow rate of Q = 229.15 cfs, that remaining at the end of the weir from the first table, is $Y_{o2} = 3.337$ ft. The depth conjugate to this depth is 2.006 ft that is above the depth of 1.586 ft at the end of the weir. Therefore, an M_3 GVF profile will exist downstream from the weir. A solution of the ODEs for the GVF that uses x as the dependent variable, and Y varying from 1.586 to 2.006 as the independent variable gives a length of 142.4 ft. The flow rate leaving the side weir is 370.85 cfs, or the difference between the first and the last values of Q from the first table.

Listing of FORTRAN program for obtaining the above solution EPRB4_35.FOR

```
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV
      COMMON /TRAN/ CMA,CDG,HW,G,XO,B,FM,FN,SO,Q,SFMQ,QS1
      WRITE(6,* )'GIVE IOUT,TOL,DELX,YB,QB,FN,SO,B,FM,XBEG,',
     &'XEND,Hw,Cd,g'
      READ(5,*) IOUT,TOL,DELX,YB,QB,FN,SO,B,FM,XBEG,
     &XEND,HW,CD,G
      CDG=.6666667*CD*SQRT(2.*G)
      CMA=1.
      IF(G.GT.30.) CMA=1.486
      REAL Y(1),XP(1),YP(1,1),WK1(1,13)
      Y(1)=YB
      H1=.01
      Q=QB
      XO=XBEG
      ADELX5=.5*DELX
      SFMQ=2.*SQRT(FM*FM+1.)
      X=XBEG
      QS1=CDG*(Y(1)-HW)**1.5
      SM=(.5*B+FM*Y(1)/3.)*Y(1)*Y(1)+Q**2/(G*(B+FM*Y(1))*Y(1))
      WRITE(IOUT,100) X,Y,QB,QS1,SM
2     XZ=X+DELX
      CALL ODESOL(Y,YPRIME,1,X,XZ,TOL,H1,0.,1,XP,YP,WK1,DYX)
      QS2=CDG*(Y(1)-HW)**1.5
      XO=XZ
      Q=Q-ADELX5*(QS1+QS2)
      QS1=QS2
      X=XZ
      SM=(.5*B+FM*Y(1)/3.)*Y(1)*Y(1)+Q**2/(G*(B+FM*Y(1))*Y(1))
      WRITE(IOUT,100) X,Y,Q,QS1,SM
100   FORMAT(6X,5F10.3)
      IF(DELX .LT. 0.) GO TO 8
      IF(X .LT. XEND) GO TO 2
      GO TO 99
8     IF(X .GT. XEND) GO TO 2
99    STOP
```

```
END
SUBROUTINE DYX(X,Y,YPRIME)
REAL Y(1),YPRIME(1)
COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV
COMMON /TRAN/ CMA,CDG,HW,G,XO,B,FM,FN,SO,QO,SFMQ,QS1
A=(B+FM*Y(1))*Y(1)
T=B+2.*FM*Y(1)
P=B+SFMQ*Y(1)
QS=CDG*(Y(1)-HW)**1.5
Q=QO-.5*(X-XO)*(QS1+QS)
SF=(FN*Q*(P/A)**.66666667/(CMA*A))**2
A2=A*A*G
FR2=Q*Q*T/(A*A2)
YPRIME(1)=(SO-SF+Q*.5*(QS+QS1)/A2)/(1.-FR2)
RETURN
END
```

The input data for the two solutions given above consist of

```
6 .001 10 4.45 600 .013 .001 8 1 0 150 1.3 .675 32.2
```

and

```
6 .001 −.25 3.262 220 .013 .001 8 1 150 147 1.3 .675 32.2, respectively.
```

From the above analysis, we can conclude that the assumption of the critical depth at the entrance is correct. The analysis shows that well over half of the flow discharges from the side weir. If the depth were above the critical depth at the entrance to the weir, then the depth would increase through the entire length of the side weir, and before the end of the weir the entire 600 cfs would have discharged from the main channel.

EXAMPLE PROBLEM 4.38

The length of the side weir in the previous problem is increased from 150 to 200 ft. Now will the hydraulic jump occur within the side weir length?

Solution

The input to the above program will be the same except that 150 is changed to 200. At the end of the side weir, Y = 1.544 ft, Q = 204.79 cfs, $q_o^* = 0.435$ cfs/ft, and M = 99.160 ft^3. The normal depth in the downstream channel with Q = 204.79 cfs is $Y_{o2} = 3.134$ ft, and the momentum function associated with this flow rate is $M_{o2} = 86.87$ ft^3. Thus, since $M > M_{o2}$, the jump will still occur downstream from the side weir and the solution shows this position is 99.1 ft. This position is obtained by solving the depth conjugate to 3.134 which is 1.865 ft and then by solving the M_3 GVF profile. In other words, the increase of 50 ft of the side weir length has moved the position of the jump up by a distance $\Delta x = 142.4 - 99.1 = 43.3$ ft. With this extended length of side weir, the amount of water leaving from it is 395.21 cfs or 395.21 − 370.85 = 24.36 cfs more than that left from the 150 ft length of weir. The reasons why lengthening the side weir does not have a larger effect in moving the position of the hydraulic jump is that the depth over the side weir rapidly decreases so that the majority of the lateral outflow occurs over the first portion of its length. In other words, the water surface profile gradually approaches the height of the weir and as this occurs, the amount of lateral outflow approaches zero.

4.15.2 Closed-Form Solution to Side Weir Outflow

A closed-form solution to Equation 4.13 can be obtained, as given in Bakhmeteff's book (1932), and expanded upon by de Marchi (1934), and more fully developed by Chow (1959), by making the following assumptions: (1) $S_o = S_f$, which is equivalent to assuming that the specific energy E is constant; (2) the channel is rectangular, and its width b is constant; and (3) the discharge coefficient is constant. From the definition of the specific energy $Q = bY\{2g(E - Y)\}^{1/2}$. Substituting this for Q in Equations 4.13 and4.14 for q_o^* into Equation 4.13 with S_o and S_f eliminated from its numerator results in

$$\frac{dY}{dx} = \frac{4}{3} C_d \frac{\sqrt{(E-Y)(Y-H_w)^3}}{b(3Y-2E)} \tag{4.15}$$

Note that when the flow is critical, the denominator of this ODE is also zero, as one would expect. The solution to this ODE is the following equation, which you can demonstrate by taking the differentials of both sides:

$$\frac{2}{3}\frac{C_d x}{b} = \frac{(2E-3H_w)}{(E-H_w)}\sqrt{\frac{E-Y}{Y-H_w}} - 3\sin^{-1}\left(\frac{E-Y}{Y-H_w}\right)^{1/2} + K \tag{4.16}$$

If the channel is mild downstream from the side weir, then the constant of integration K is evaluated by substituting the downstream normal depth Y_{o2} for Y and the length of the weir L for x, and evaluating the constant E from the downstream normal depth $E = Y_{o2} + (Q_2/A_{o2})^2/(2g)$. If the flow at the beginning of the side weir is critical, or supercritical, then this upstream depth Y_1 controls, and K is evaluated by replacing Y with Y_1 and x with 0, and E with $Y_1 + (Q_1/A_1)^2/(2g)$.

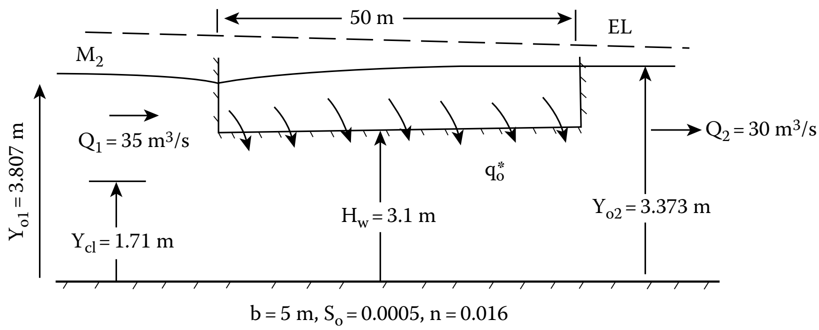
EXAMPLE PROBLEM 4.39

A rectangular channel with a bottom width b = 5 m, a bottom slope of $S_o = 0.0005$, and a roughness coefficient n = 0.016 has a 50 m long side weir whose crest is 3.1 m above the channel bottom. The discharge coefficient for this side weir is $C_d = 0.60$. Determine the GVF profiles associated with the side weir and the discharge from it if the flow in the upstream channel is 35 m³/s. Compare the result from the closed form Equation 4.15 with the numerical solution.

Solution

Based on Manning's equation, the normal depth in the channel upstream from the side weir is 3.807 m. The critical depth associated with Q = 35 m³/s is Y_{c1} = 1.71 m. With the crest at 3.1 m, it is quite clear that the flow will remain subcritical throughout the length of channel containing the weir. To obtain the solution, the flow rate remaining in the channel must be guessed (30 m³/s will be the start of such a guess), the normal depth associated with this flow rate computed (Y_{o2} = 3.373 m for 30 m³/s), and the GVF profile solved starting with the depth at the end of the weir. If the solution shows a flow rate of 35 m³/s in the main channel at beginning of weir, then the guessed flow rate is correct. Otherwise, the guess must be adjusted and the entire procedure repeated. The solution based on a downstream flow rate Q = 30 m³/s is given below.

x (m)	Y (m)	Q (m³/s)	q_o^*	M (m³)
50.000	3.373	30.000	0.253	33.883
45.000	3.358	31.213	0.233	34.110
40,000	3.344	32.329	0.214	34.331
35.000	3.331	33.355	0.197	34.547
30.000	3.318	34.299	0.181	37.756
25.000	3.307	35.167	0.166	34.959
20.000	3.296	35.966	0.153	35.154
15.000	3.285	36.702	0.141	35.344
10.000	3.276	37.383	0.131	35.527
5.000	3.267	38.013	0.121	35.704
0.000	3.259	38.597	0.113	35.875



The flow rate $Q_1 = 38.597\,m^3/s$ from this solution at the beginning of the weir is too large. Next, a flow rate of $Q_2 = 29.4\,m^3/s$ was guessed as the amount remaining in the channel. This guess resulted in a flow rate of $35.9\,m^3/s$ at the beginning of the side weir, and is close enough that it will be accepted. The following table provides this solution:

x (m)	Y (m)	Q (m^3/s)	q_o^*	M (m^3)
50.000	3.321	29.400	0.184	32.879
45.000	3.310	30.287	0.171	33.043
40.000	3.300	31.110	0.518	33.203
35.000	3.290	31.873	0.147	33.359
30.000	3.281	32.582	0.137	33.511
25.000	3.273	33.241	0.127	33.658
20.000	3.265	33.855	0.118	33.801
15.000	3.257	34.426	0.110	33.940
10.000	3.250	34.959	0.103	34.075
5.000	3.244	35.458	0.097	34.206
0.000	3.238	35.926	0.091	34.334

The input to the program listed under Example Problem 4.37 to obtain this solution is

3 .001 −5 3.322 29.4 .016 .0005 5 0 50 0 3.1 23 .6 9.81

Since this solution gives $Q_1 = 35\,m^3/s$, it is the correct solution, and an M_2 GVF profile exists upstream from the beginning of the weir that changes the depth from 3.769 m (1% below normal depth) to 3.203 m. The solution of this GVF profile is L = 520 m.

A better approach than solving this problem by trial is to simultaneously solve Q_2, Y_{o2}, and Y_1 using the three available equations: (1) Manning's equation for uniform flow downstream from the side weir; (2) the GVF ODE through the side weir length, including a numerical integration to evaluate the outflow $Q_{out} = \int q_o^* dx$ and (3) the continuity equation $Q_1 = Q_2 + Q_{out}$. This approach is implemented in program SOLWEIM (discussed later), and using a 1 m increment to evaluate $\int q_o^* dx$ and solve the ODE, the solution is $Q_2 = 28.80\,m^3/s$, $Y_1 = 3.194$ m, and $Y_{o2} = 3.273$ m.

To use Equation 4.16 to solve this problem, we will use the downstream flow rate $Q_2 = 29.42\,m^3/s$ obtained from the numerical solution. If this were not available. then a trial-and-error solution of the equation would be necessary since it is the normal downstream depth that controls. $E = 3.322 + (29.42/16.61)^2/19.62 = 3.482$ m. Substituting x = L = 50, b = 5, $C_d = 0.6$, and $H_w = 3.1$ into Equation 4.16 and evaluating K = 10.948803, the table below gives the depth Y and the outflow for different positions along the side weir by solving Equation 4.16 for Y.

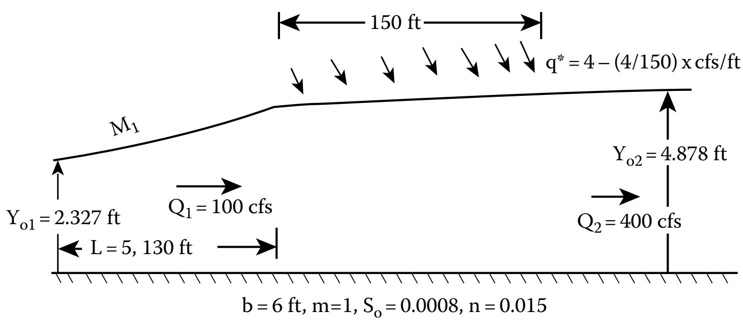
Solution of closed form equation

x (m)	Y (m)	q_o^* (m³/s)	ΔQ (m³/s)	Q (m³/s)
50	3.322	0.1980		29.42
45	3.321	0.1837	0.954	30.37
40	3.310	0.1703	0.885	31.26
35	3.299	0.1576	0.820	32.08
30	3.289	0.1457	0.758	32.84
25	3.279	0.1346	0.701	33.54
20	3.270	0.1242	0.647	34.19
15	3.261	0.1167	0.602	34.79
10	3.253	0.1058	0.556	35.34
5	3.245	0.0976	0.509	35.85
0	3.237	0.0901	0.469	36.32

Notice that this closed form solution gives slightly larger depths, and as a consequence the flow rate at the beginning of the side weir is larger than 35 m³/s. If it were being used solely to solve this problem, then the downstream flow rate should be guessed smaller and the solution repeated until $Q_1 = 35\,m^3/s$.

EXAMPLE PROBLEM 4.40

Lateral inflow into a main channel varies linearly over a 150 ft length of this channel from 4 cfs/ft at its beginning to 0 cfs/ft at its end. The channel contains a flow rate of $Q_1 = 100$ cfs upstream from this inflow and has the following properties: b = 6 ft, m = 1, $S_o = 0.0008$, and n = 0.015. Determine the GVF profile and the depth of water at the beginning of the inflow length.



Solution

Integration of the lateral inflow gives 300 cfs of total inflow. Therefore, the channel downstream from the spatially varied flow will carry a flow rate $Q_2 = 400$ cfs. The normal depth associated with this flow rate is $Y_{o2} = 4.878$ ft, and this depth will be the depth at the end of the spatially varied flow length. The solution can proceed in a relatively straight forward manner by numerically solving Equation 4.7 from the downstream end to the beginning of the inflow length. In this solution, it is necessary to properly determine both the total flow rate Q and the lateral inflow q* for every x as the solution progresses. With the origin for x at the beginning of the spatially varied flow length, the flow rate at any point will be given by $Q(x) = Q_1 + x(4 + q^*(x))/2$, and the lateral inflow at this point $q^*(x) = 4 - 4x/150$. The solution to this GVF profile is given below (using program EP4_38):

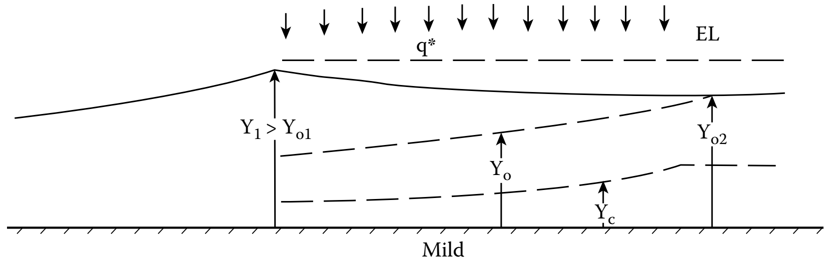
x (ft)	Y (ft)	E (ft)	M (ft³)	q*	Q (cfs)
150.0	4.878	5.760	203.7	0.0000	400.00
140.0	4.917	5.773	204.1	0.2667	398.67
130.0	4.995	5.797	204.5	0.5333	394.67
120.0	5.097	5.828	204.7	0.8000	388.00
110.0	5.213	5.865	204.9	1.0667	378.67
100.0	5.335	5.906	205.1	1.3333	366.67
90.0	5.458	5.950	205.1	1.6000	352.00
80.0	5.578	5.995	205.0	1.8667	334.67
70.0	5.692	6.039	204.9	2.1333	314.67
60.0	5.800	6.083	204.7	2.4000	292.00
50.0	5.900	6.124	204.4	2.6667	266.67
40.0	5.991	6.163	204.0	2.9333	238.67
30.0	6.072	6.197	203.6	3.2000	208.00
20.0	6.142	6.227	203.1	3.4667	174.67
10.0	6.199	6.251	202.6	3.7333	138.67
0.0	6.242	6.268	202.0	4.0000	100.00

Since the normal depth for the channel upstream from this inflow length is $Y_{o1} = 2.327$ ft, i.e., for that associated with a flow rate of 100 cfs, there will be an M_1 GVF profile upstream that takes the depth from 2.35 (1 % above normal) to 6.24 ft. The length of this upstream back water curve can best be solved by considering Y as the independent variable. With this solution of a simple spatially varied inflow problem as background, various possibilities are examined in the next section.

4.16 SPATIALLY VARIED INFLOWS

When dealing with spatially varied GVF profiles, the effect of the inflow terms involving q* is opposite to that for the lateral outflow. In a prismatic channel, these terms for lateral inflow add a negative quantity to the numerator of Equation 4.7. These negative terms cause the GVF profile in a prismatic channel to decrease in the downstream direction for subcritical flows, and increase in the downstream direction for supercritical flow, provided that S_o is not larger than the sum of these terms and S_f. Even if the influence of the terms containing q* are small in comparison to S_o minus S_f, a length of the channel with lateral inflow will have a smaller sloping negative water surface profile than an equivalent situation without the lateral inflow. In conceptionalizing why this is the case, it can be reasoned that because of the increasing flow rate in the downstream direction, the velocity must be larger, and this larger velocity forces the water surface in the channel flow to be further below the energy line than would be if there were no lateral inflow occurring. In addition, because of the increasing flow rate in the downstream direction, the energy line will curve downward because its slope increases in the downstream direction. On the other hand, under supercritical flow conditions, the lateral inflow terms will tend to cause a rising water surface profile in the downstream direction.

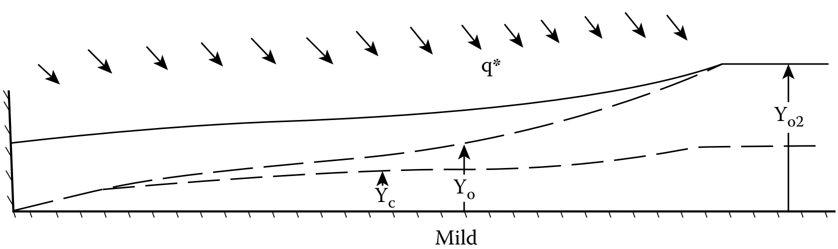
Simultaneous with the effects described above, an increase in the channel flow, from lateral inflow, will increase the normal depth of flow in the channel downstream from the inflow length of the channel. Thus, in a mild channel, a lateral inflow over a short length of a channel will cause an M_1 GVF profile to exist in the channel upstream from where the inflow begins, as shown in the sketch below, since under subcritical flow conditions downstream control exists.



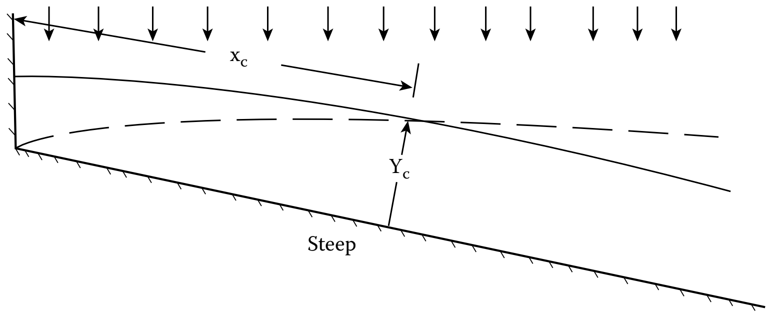
An exception to the effects can occur if the velocity component of the incoming flow in the direction of the main flow U_q is sizably larger in magnitude than the main channel velocity V. Should U_q be greater than 2V, then the term Fq (which becomes positive under these conditions) will contribute a larger magnitude in the numerator of Equation 4.7 than the negative magnitude of $Qq^*/(gA^2) = Vq^*/(gA)$, and then a reversal of the effects described above will occur. Thus, spatially varied inflows can be very complex.

A common but special spatially varied inflow is when the beginning of the flow is at this position in the channel, e.g., there is no upstream flow. It does not matter if the channel physically begins at the beginning of the lateral inflow or whether there is nonmoving water in a channel upstream from this position, the depth of water at the beginning of the inflow will be the same. Thus, without loss of generality, it is possible to consider only the case of a channel beginning at the point where the inflow begins.

Consider first the case in which a long mild channel exists downstream from the spatially varied flow. Since the depth, etc., will be "downstream controlled" the actual depth will be well above the critical depth throughout the lateral inflow length, as shown in the sketch below, particularly since the critical depth will be zero at the beginning of the channel, where Q = 0. Therefore, the solution to problems in this category are obtained by beginning at the downstream end of the channel at a depth equal to the normal depth associated with a flow rate equal to the integral of the lateral inflow over the length of the inflow section or $Q = \int q^* dx$. If q* is constant, then this flow rate is given by $Q = q^*L$, where L is the length over which the lateral inflow takes place.



Next, consider the case in which a steep channel exists downstream from the inflow. For this situation, the flow will pass through a critical depth somewhere between the beginning of the lateral inflow and the end of the lateral inflow length. However, this position of critical flow may be at the very end. At the beginning of the channel, regardless of how steep its bottom may be, the flow will be subcritical because it will have a finite depth, and the critical depth will be zero where the flow rate is zero. It can be reasoned that if the flow changes from a subcritical to a supercritical flow within the length of channel containing the lateral inflow, and not at the very end of this length, then the depth will change continuously,



i.e., the assumption of negligible normal accelerations, upon which the ODEs of gradually varied flows is based, is valid. Therefore, since the denominator, $1 - F_r^2$ of Equation 4.7 becomes zero and dY/dx must be finite, at critical flow, this equation must be of the indeterminate form 0/0. If the channel is prismatic, then equating the numerator of Equation 4.7 to zero gives the following equation for a prismatic channel:

$$S_o - S_f - \frac{2Qq^*}{gA^2} = 0 \tag{4.17a}$$

or

$$S_o - S_f - \frac{2X_c q^{*2}}{gA^2} = 0 \tag{4.17b}$$

This equation also assumes that the component of velocity U_q of the lateral inflow in the direction of the main channel flow is zero. If U_q is not zero, then this equation needs to be modified accordingly. This equation applies only at the location x_c where the flow passes through a critical depth. Upstream from this position, the flow will be subcritical, and downstream from x_c the flow will be supercritical. Since the critical depth varies through the spatially varied inflow length as Q varies, there are three unknowns, i.e., S_f, x_c, and Y_c. The three equations that allow for the solution of these three unknowns are Equation 4.17, the critical flow equation $Q^2T/(gA^3) = 1$, and a uniform flow equation. If the length x_c computed from the solution of these three simultaneous equations is larger than the lateral inflow length, then the critical depth will occur at the end of the inflow, since the flow in the downstream steep channel must be supercritical.

If the solution of the above three simultaneous equations is accomplished by hand, one may be inclined to stop when the equations are close to being solved on the basis that the so-called known variables, such as q*, are not very precisely known. Poor accuracy in solving the equations may well lead to numerical problems associated with the solution of the ODE for a gradually varied flow, especially if this solution is accomplished by an ODE solver on a computer. Should the position used for x_c be downstream from where the critical depth actually does occur, and Y is taken too close to Y_c, then the solution of Equation 4.7 moving upstream, for example, may be working with $1 - F_r^2$ negative instead of positive, and the solution will proceed along an incorrect branch of the GVF function.

EXAMPLE PROBLEM 4.41

Lateral inflow feeds water into the beginning of a prismatic channel with a bottom width of b = 5 ft, a side slope of m = 1.5, a roughness coefficient of n = 0.014, and a bottom slope of $S_o = 0.0300$. The inflow is constant with $q^* = 1$ cfs/ft over a length of 200 ft and $U_q = 0$. Determine whether the flow passes through a critical depth within the inflow length or at its end, and then solve the spatially varied flow profiles.

Solution

Since S_f is defined by a uniform flow equation, e.g., Manning's equation, the location of the critical flow section involves the simultaneous solution of Equation 4.17, the critical flow equation, and Manning's equation, for the three variables S_f, x_c, and Y_c. The Newton method as described earlier, and in Appendix B, can be utilized for this purpose. The solution gives: $S_f = 0.002912$, $x_c = 105.6$ ft, and $Y_c = 1.960$ ft. Since x_c is less than the 200 ft length of lateral inflow, the flow will be subcritical upstream from this position and supercritical downstream therefrom. The solution of the GVF profile will therefore need to begin just a small distance upstream, and a small distance downstream from $x_c = 105.6$ ft at depths slightly above and below $Y_c = 1.96$ ft. These solutions are given below and are obtained by numerically solving Equation 4.7.

Solution from critical point $x_c = 105.6$ ft upflow to the beginning of the channel.

x (ft)	Y (ft)	E (ft)	M (ft³)	F_r
105.0	2.000	2.669	35.40	0.959
100.0	1.927	2.599	33.29	0.976
95.0	1.894	2.529	31.24	0.956
90.0	1.861	2.459	29.23	0.934
85.0	1.827	2.388	27.26	0.912
80.0	1.793	2.316	25.33	0.888
75.0	1.756	2.242	23.45	0.862
70.0	1.719	2.167	21.61	0.836
65.0	1.680	2.091	19.81	0.808
60.0	1.640	2.013	18.07	0.778
55.0	1.597	1.934	16.37	0.747
50.0	1.553	1.853	14.72	0.713
45.0	1.506	1.769	13.13	0.677
40.0	1.457	1.684	11.60	0.637
35.0	1.404	1.595	10.13	0.594
30.0	1.347	1.504	8.72	0.546
25.0	1.286	1.408	7.38	0.493
20.0	1.219	1.309	6.11	0.432
15.0	1.444	1.204	4.93	0.360
10.0	1.059	1.091	3.84	0.273
5.0	0.958	0.969	2.86	0.161
0.0	0.830	0.830	2.01	0.000

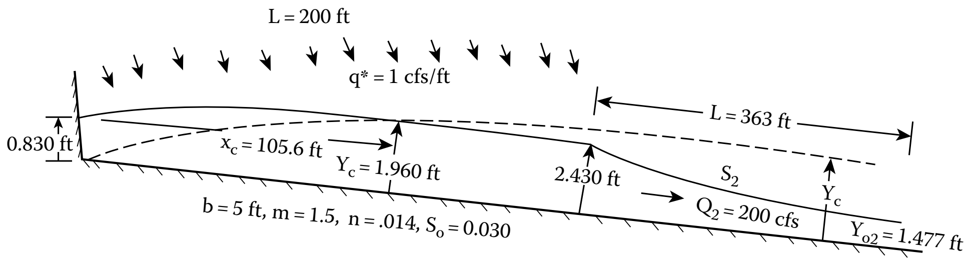
Solution downstream from critical section $x_c = 105.6$ ft.

x (ft)	Y (ft)	E (ft)	M (ft³)	F_r
110.0	1.900	2.745	37.65	1.101
115.0	1.985	2.804	39.70	1.065
120.0	2.036	2.868	41.86	1.062
125.0	2.069	2.932	44.07	1.074
130.0	2.098	2.996	46.33	1.089
135.0	2.125	3.060	48.62	1.105
140.0	2.152	3.123	50.94	1.121
145.0	2.177	3.185	53.29	1.136
150.0	2.202	3.247	55.68	1.152
155.0	2.227	3.308	58.09	1.166
160.0	2.251	3.369	60.53	1.180
165.0	2.275	3.429	63.00	1.194
170.0	2.298	3.489	65.50	1.208

(*continued*)

(continued)

x (ft)	Y (ft)	E (ft)	M (ft³)	F_r
175.0	2.321	3.548	68.03	1.221
180.0	2.344	3.607	70.59	1.234
185.0	2.366	3.665	73.17	1.247
190.0	2.388	3.723	75.77	1.259
195.0	2.409	3.780	78.41	1.271
200.0	2.430	3.837	81.07	1.283



From this solution, it should be noted that a supercritical flow exists at the end of the spatially varied flow length. To complete the GVF profiles associated with this problem, the S_2 GVF profile should be solved. At its beginning, the depth is 2.430 ft, and it continues until the depth is within 1% of the normal depth that is Y_{o2} = 1.477 ft. Since depths are known at both end of this S_2 GVF profile, it will be most convenient to consider Y the independent variable and x the dependent variable. This length is L = 363 ft. Note that the depth at the beginning of the channel where the flow rate is zero equals 0.83 ft.

EXAMPLE PROBLEM 4.42

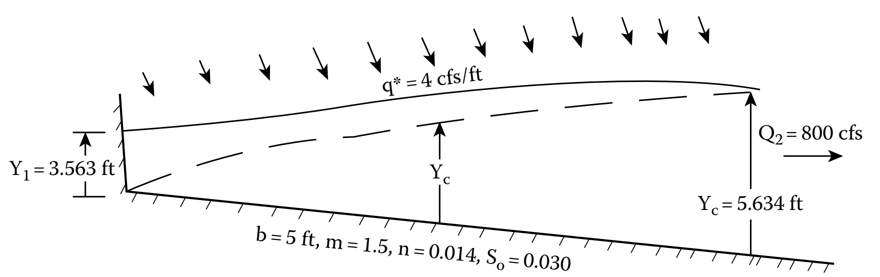
The rate of lateral inflow to the channel in the previous problem is q* = 2 cfs/ft. What are the water surface profiles?

Solution

The simultaneous solution of Equation 4.15, the critical flow equation, and Manning's equation now gives the following: S_f = 0.002519, x_c = 179.5 ft, and Y_c = 3.777 ft. The solution will proceed as in the previous problem.

EXAMPLE PROBLEM 4.43

The rate of the lateral inflow to the channel in Example Problem 4.41 is q* = 4 cfs/ft. What are the GVF profiles now?



Solution

The simultaneous solution to the same three equations now gives: S_f = 0.002192, x_c = 290.5 ft, and Y_c = 6.742 ft. Since this is beyond the length of the spatially varied flow, a critical flow will occur at 200 ft. Upstream from this position the flow will be subcritical, and downstream from here the flow will be supercritical, e.g., an S_2 GVF profile. Critical depth associated with Q = 800 cfs in this channel is $Y_c = 5.634$ ft. The GVF profile computation for the spatially varied flow will therefore begin slightly above this depth and proceed upstream. The solution is given below.

Solution from critical depth at the end of the lateral inflow length.

x (ft)	Y (ft)	E (ft)	M (ft³)	F_r
200.0	5.700	7.366	431.16	0.976
190.0	5.870	7.236	408.63	0.873
180.0	5.859	7.092	385.66	0.830
170.0	5.814	6.942	362.77	0.797
160.0	5.752	6.787	340.14	0.767
150.0	5.680	6.628	317.86	0.738
140.0	5.600	6.465	296.00	0.709
130.0	5.514	6.298	274.58	0.680
120.0	5.422	6.127	253.66	0.649
110.0	5.324	5.953	233.26	0.618
100.0	5.220	5.774	213.42	0.585
90.0	5.109	5.590	194.16	0.550
80.0	4.992	5.401	175.52	0.512
70.0	4.867	5.207	157.54	0.472
60.0	4.733	5.006	140.25	0.428
50.0	4.588	4.797	123.70	0.379
40.0	4.430	4.579	107.94	0.325
30.0	4.255	4.351	93.04	0.264
20.0	4.060	4.109	79.07	0.193
10.0	3.834	3.849	66.15	0.108
0.0	3.565	3.565	54.42	0.000

4.16.1 Algebraic Solution

A closed form solution giving the depth Y as a function of position x along a lateral inflow can also be obtained under the following assumptions:

1. The channel is a constant width rectangular channel.
2. $S_o - S_f = 0$, i.e., these terms are omitted from the numerator of the ODE.
3. No flow exists upstream from the start of the inflow.
4. The inflow is constant, i.e., q^* = constant, i.e., $Q = xq^*$.

Based on these assumptions, the ODE becomes

$$\frac{dY}{dx} = \frac{2xq^{*2}Y}{x^2q^{*2} - gb^2Y^3} \quad \text{or} \quad \frac{dx}{dY} = \frac{(xq^*)^2 - gb^3Y^3}{2xq^{*2}Y}$$

Multiplying the inverted form of this equation by 2x and dividing by Y gives

$$\frac{dx^2}{YdY} - \frac{x^2}{Y^2} = \frac{d(x^2/Y)}{dY} = -\frac{gb^2Y}{q^{*2}}$$

Integration of both sides leads to

$$\frac{x^2}{Y} = -\frac{g(bY)^2}{2q^{*2}} + K = -0.5g(by/q^*)^2 + K \tag{4.17a}$$

If the flow is subcritical, it will be controlled by the depth Y_2 downstream from the lateral inflow, and this will be the normal depth Y_o if the channel is long, based on the flow rate $Q_2 = Lq^*$, or

$$K = L^2/Y_2 + 0.5g(bY_2/q^*)^2$$

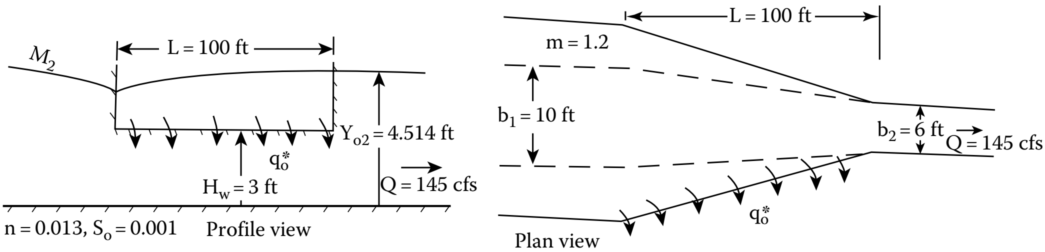
but if this equation is used upstream and downstream from the position x_c where the critical depth occurs in a steep channel, then K is evaluated by substituting the value x_c for x and Y_c for Y. When using the equation in steep channels caution must be exercised to obtain the subcritical root for positions upstream from x_c and the supercritical root downstream therefrom. The theory does not do a good job in duplicating the numerical solution results for a steep channel because the value of S_f is quite different from S_o in these channels. Homework Problem 4.172 illustrates this point. This problem examines the lateral inflow of $q^* = 1.5\,m^2/s$ into a 3 m wide rectangular channel for three different cases: (a) the channel is mild with a slope of $S_o = 0.0005$ and a lateral inflow length of 10 m. For this case, the flow downstream will be normal, which is $Y_{o2} = 2.924$ m (for $Q_2 = 15\,m^3/s$) and K = 201.94633, and the theoretical and numerical results are essentially identical. (b) For this second case, the bottom slope is very steep, $S_o = 0.25$ with a length of inflow of L = 20 m. The critical position computes as $x_c = 13.66$ m (with $Y_c = 1.682$ m here), and therefore both the theoretical and numerical solutions need to be in two parts, upstream and downstream from this position (K = 166.4665 for the theoretical solution). Now, the numerical solution gives a depth of 0.498 m at the beginning of the channel and a depth 1.929 m at its end. The numerical solution gives Y = 2.0 at x = 0 and Y = 1.5 m at x = L = 20 m, i.e., there is only agreement near the critical condition because that is where the solutions begin. (c) The last and third case has a slope $S_o = 0.15$ with L = 20 m, and this slope produces a critical position beyond the length of the inflow, and therefore the critical depth of $Y_c = 2.168$ m will occur at the end of the lateral inflow. Now K = 276.72 for the theoretical solution, and at x = 0, it gives a depth of 2.5 m, i.e., the depth decreases continuously over the entire inflow length from 2.5 to 2.168 m, whereas the numerical solution gives a depth of 1.303 m at the beginning of the channel, i.e., an increasing depth over the lateral inflow length, i.e., the difference between $S_o - S_f$ even for this case is sufficient to overcome the negative influence of the lateral inflow term in the numerator of the ODE. Thus, we might conclude that the exact solution should only be used if the channel slope is mild.

4.17 SPATIALLY VARIED FLOW IN NONPRISMATIC CHANNELS

From the viewpoint of numerically solving spatially varied problems in nonprismatic channels, the extension of what has just been discussed is to include the term(s) in the numerator of Equation 4.6 that accounts for the change in the cross-sectional area with respect to the position, i.e., terms involving $\partial/\partial x$. From a conceptional viewpoint, the problem is much more difficult because reduction in the size of the channel can more than offset the effects of the lateral outflow, for example, because the signs associated with these two terms in the numerator of Equation 4.6 are opposite. Thus, a solution of the ODE is required before it is possible to decide what the shape of the GVF profile will be even if a selected situation has been assumed.

EXAMPLE PROBLEM 4.44

A spatially varied flow occurs over a 100 ft long section of a side weir with a discharge coefficient of $C_d = 0.45$, and whose crest is 3 ft above the channel bottom. Downstream from the side weir, the channel is rectangular with a bottom width of 6 ft, and a bottom slope $S_{o2} = 0.001$, and a Manning's n = 0.013. The channel upstream from the weir is trapezoidal with b = 10 ft and m = 1.2 and has the same bottom slope and roughness coefficient as the downstream channel. If the flow rate downstream from the weir has been measured as $Q_2 = 145.0$ cfs, determine the water surface profiles across the side weir length of channel, and the flow rate in the channel upstream from the side weir.



Solution

Since the flow rate downstream from the weir is known, the normal depth in the downstream channel will control, and the solution to Equation 4.6 will begin here and proceed upstream. The starting depth is $Y_{o2} = 4.514$ ft obtained from a solution of Manning's equation. The solution to this spatially varied GVF profile, using the following input to program EPRB4_42, is given below:

3 1.e−5 −2 4.514 145 .013 .001 10 1.2 6 0 100 0 3 .45 32.2 5

x (ft)	Y (ft)	Q (cfs)	q cfs/ft	M (ft³)	Fr²
100.0	4.514	145.00	4.4849	85.24	0.197
90.0	4.368	186.48	3.8504	100.09	0.291
80.0	4.268	222.76	3.4362	114.29	0.366
70.0	4.203	255.73	3.1780	127.97	0.423
60.0	4.163	286.66	3.0213	141.28	0.464
50.0	4.139	316.37	2.9272	154.35	0.493
40.0	4.124	345.33	2.8708	167.26	0.515
30.0	4.116	373.86	2.8368	180.09	0.531
20.0	4.110	402.11	2.8161	192.87	0.544
10.0	4.107	430.21	2.8036	205.61	0.554
0.0	4.105	458.20	2.7961	218.34	0.563

The program listed under problem 35 needs slight modification to include the term involving $\partial A/\partial x$, and is given below.

FORTRAN listing EPRB4_44.FOR to solve above problem

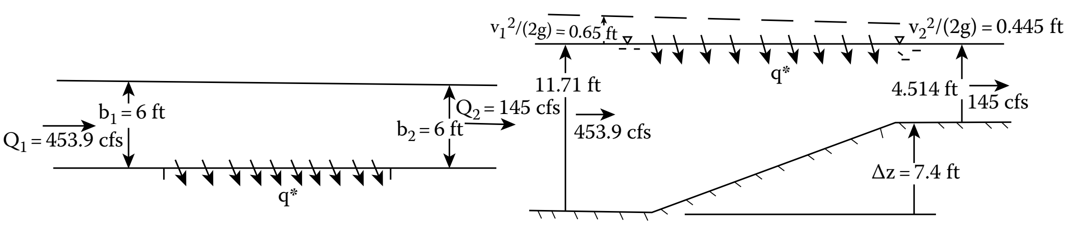
```
      REAL Y(1),XP(1),YP(1,1),WK1(1,13)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV
      COMMON /TRAN/CMA,CDG,HW,G,XO,BO,FMO,FN, *SO,Q,QS1,DB,DM
```

```
       WRITE(6,* )'GIVE IOUT,TOL,DELX,YB,QB,FN,SO,BO,FMO
     &*,B2,FM2,XBEG,XEND,Hw,Cd,g,MPRT'
       READ(5,*)IOUT,TOL,DELX,YB,QB,FN,SO,BO,
     &*FMO,B2,FM2,XBEG,XEND,HW,CD,G,MPRT
       CDG=.666667*CD*SQRT(2.*G)
       CMA=1.
       IF(G.GT.30.) CMA=1.486
       DB=(B2-BO)/ABS(XEND-XBEG)
       DM=(FM2-FMO)/ABS(XEND-XBEG)
       Y(1)=YB
       H1=.01
       Q=QB
       XO=XBEG
       ADELX5=.5*DELX
       X=XBEG
       B=BO+DB*X
       FM=FMO+DM*X
       IF(Y(1).GT.HW) THEN
       QS1=CDG*(Y(1)-HW)**1.5
       ELSE
       QS1=0.
       ENDIF
       AREA=(B+FM*Y(1))*Y(1)
       SM=(.5*B+FM*Y(1)/3.)*Y(1)*Y(1)+Q**2/(G*AREA)
      WRITE(IOUT,100) X,Y,QB,QS1,SM,Q**2*(B+2.*FM*Y(1))/(G*AREA**3)
       MP=0
2      XZ=X+DELX
       CALL ODESOL(Y,YPRIME,1,X,XZ,TOL,H1,0.,1,XP,YP,WK1,DYX)
       IF(Y(1).GT.HW) THEN
       QS2=CDG*(Y(1)-HW)**1.5
       ELSE
       QS2=0.
       ENDIF
       XO=XZ
       Q=Q-ADELX5*(QS1+QS2)
       QS1=QS2
       X=XZ
       B=BO+DB*X
       FM=FMO+DM*X
       AREA=(B+FM*Y(1))*Y(1)
       SM=(.5*B+FM*Y(1)/3.)*Y(1)*Y(1)+Q**2/(G*AREA)
       MP=MP+1
       IF(MOD(MP,MPRT).EQ.0) WRITE(IOUT,100) X,Y,Q,QS1,SM,
     &Q**2*(B+2.*FM*Y(1))/(G*AREA**3)
100    FORMAT(1X,F8.1,F10.3,F10.2,F10.4,F10.2,F10.3)
       IF(DELX .LT. 0.) GO TO 8
       IF(X .LT. XEND) GO TO 2
       GO TO 99
8      IF(X .GT. XEND) GO TO 2
99     STOP
       END
       SUBROUTINE DYX(X,Y,YPRIME)
       REAL Y(1),YPRIME(1)
       COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV
       COMMON /TRAN/CMA,CDG,HW,G,XO,BO,FMO,FN,
     &*SO,QO,QS1,DB,DM
```

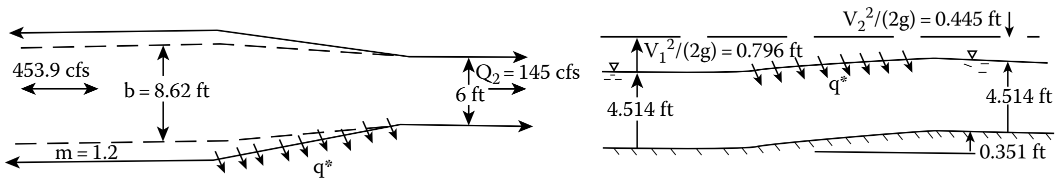
```
      B=BO+DB*X
      FM=FMO+DM*X
      A=(B+FM*Y(1))*Y(1)
      T=B+2.*FM*Y(1)
      P=B+2.*SQRT(FM*FM+1.)*Y(1)
      IF(Y(1).GT.HW) THEN
      QS=CDG*(Y(1)-HW)**1.5
      ELSE
      QS=0.
      ENDIF
      Q=QO-.5*(X-XO)*(QS1+QS)
      SF=(FN*Q*(P/A)**.66666667/(CMA*A))**2
      A2=A*A*G
      FR2=Q*Q*T/(A*A2)
      YPRIME(1)=(SO-SF+Q*.5*(QS+QS1)/A2+Q*Q/(A*A2)*Y(1)*
     &(DB+Y(1)*DM))/(1.-FR2)
      RETURN
      END
```

This solution indicates that a flow rate of 453.9 cfs exists in the upstream channel and the depth at the beginning of the side weir is 4.10 ft. The normal depth in the upstream channel associated with a flow rate of $Q_1 = 453.9$ cfs is 4.24 ft. Therefore, an M_2 GVF profile occurs in the channel upstream from the weir taking the depth from 4.24 to 4.10 ft.

In this problem, the effect of the contracting section has been too little to keep the water surface completely level; it has risen from 4.10 to 4.514 or 0.41 ft across the side weir length. However, if the channel were of the same rectangular shape with a 6 ft width throughout, and the same amount of discharge were to occur from the side weir, then the bottom of the channel would need to rise by an amount of 7.40 ft across the side weir if normal depths were to exist in the upstream



channel also. This amount is computed by first noting that the specific energy associated with the downstream normal depth is $E_{o2} = 4.96$ ft. Next, based on Manning's equation the normal depth in a 6 ft wide rectangular channel with a bottom slope of 0.001, n = 0.013, and a flow rate Q = 453.9 cfs is 11.71 ft, with a specific energy $E_{o1} = 12.36$ ft. The difference between E_{o1} and E_{o2} represents the change in the bottom elevation. In other words, an effect of the contraction is to keep the depth more nearly constant across the side weir while keeping the bottom level. Another associated effect is to keep the discharge more nearly constant across the side weir. If the normal depths were equal in the channels both upstream and downstream from the side weir, then the water surface level would be essentially constant over the entire weir length thus causing a nearly constant discharge over its entire length. The design of a side weir can be based on this criteria. The amount of discharge from the side weir is determined from the design criteria. The size of the upstream channel is determined so that its normal depth equals that of the downstream channel. Using this design procedure for this problem would size the upstream channel so that its normal depth is also 4.514 ft (the normal depth in the 6 ft rectangular channel for a flow rate of $Q_2 = 145$ cfs). If the side slope m were still to change linearly from 1.2 to 0 at the end of the side weir, and the upstream flow rate were to be 453.9 cfs, then the width of the upstream channel should be b = 8.62 ft instead of 10 ft. However, the velocity heads upstream and downstream from the side weir are different, ($V_2^2/(2g) = 0.445$ ft and $V_1^2/(2g) = 0.796$ ft), and therefore the channel bottom would need to rise 0.351 ft above the frictional slope.



For situations in which the lateral inflow feeds the main channel flow in steep channels, the position where the critical flow occurs can be found by setting the numerator of Equation 4.6 to zero, as described previously. If the channel is nonprismatic, however, additional variables are introduced. If both the bottom width b and the side slope m for a trapezoidal channel change through the inflow length, then these are two additional unknowns so that five variables are unknown: S_f, x_c, Y_c, b, and m. But two additional equations are available also; the equation that gives b as a function of x, and the equation that gives m as a function of x. If the variation of b and m are linear functions of x, then these five equations are

$$b = b_o + x\left(\frac{db}{dx}\right) \quad (1)$$

$$m = m_o + x\left(\frac{dm}{dx}\right) \quad (2)$$

$$Q^2T - gA^3 = 0 \quad (3)$$

$$nQP^{2/3} - C_uA^{5/2}\sqrt{S_o} = 0 \quad (4)$$

and

$$S_o - S_f + \{Q^2/(gA^2)\}\{(\partial A/\partial x)|_Y\} - Qq^*/(gA^2) - F_q = 0 \quad (5)$$

in which

$$F_q = 0 \quad \text{for bulk lateral outflow}$$

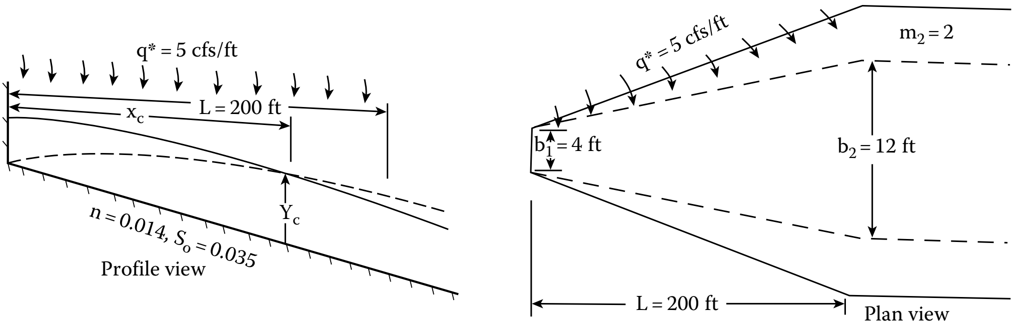
$$F_q = (Vq^*)/(2gA) = (Qq^*)/(2gA^2) \quad \text{for seepage outflow, and}$$

$$F_q = (V - U_q)q^*/(gA) + (h_c/A)[(\partial A/\partial x)\,|_Y \quad \text{for inflow}$$

Once the critical position x_c has been properly located, then problems involving lateral inflow in nonprismatic channels can be solved numerically in much the same manner as in prismatic channels. The major difference is that the term(s) involving $\partial A/\partial x$ in the numerator of Equation 4.6 must be included in evaluating dY/dx. Also, shapes of the GVF profiles are more difficult to predict, and therefore it is advisable to check carefully that the correct branch of the GVF-function is being followed in obtaining the numerical solution.

EXAMPLE PROBLEM 4.45

Over a 200 ft length of linearly expanding channel, an inflow of $q^* = 5$ cfs/ft occurs. The component of velocity of this inflow in the direction of the main channel flow is 1 fps. At its beginning, the channel is rectangular with $b_1 = 4$ ft. At the end of the 200 ft inflow length, the channel has a bottom width $b_2 = 12$ ft, and a side slope $m_2 = 2$. Downstream from here the channel is very long with this as a constant size. Throughout, the channel has a bottom slope of $S_o = 0.035$ (steep), and a Manning's n = 0.014. Determine the spatially varied flow profiles throughout the channel.



Solution

Since the channel is steep, the first step requires that it be determined whether a critical flow exist within the lateral inflow length by solving the five equations immediately above to determine x_c. The solution is: $S_f = 0.00223$, $x_c = 155.44$ ft, $Y_c = 4.472$ ft, b = 10.22 ft, and m = 1.554. The following listing of a FORTRAN program was used to solve these equations. This program is designed to solve any system of nonlinear equations by the Newton method. The Function Subprogram FUN needs to be modified to define the given system of equations. To solve other systems of equations, this subprogram needs to be rewritten so that the equations being solved are evaluated. The statements in this subprogram labeled 1, 2, 3, etc., are the equations being solved. The main program calls upon a standard linear algebra program SOLVEQ to give the solution vector given the Jacobian matrix D and the known, or the equation vector F. For this problem, the definition of the unknown and known variables are given after the program listing.

FORTRAN program EQUNOM.FOR that uses the Newton method to solve system of equations

```
      REAL F(10),D(10,10),X(10),KN(20)
      INTEGER*2 INDX(10)
50    WRITE(6,*)' HOW MAY EQS'
      READ(5,*) N
      IF(N .EQ. 0) STOP
      WRITE(6,*)' HOW MANY KNOWNS?'
      READ(5,*) NK
      WRITE(6,*)' GIVE ME THESE KNOWN VALUES.'
      READ(5,*) (KN(I),I=1,NK)
      WRITE(6,*) ' PROVIDE',N,' ESTIMATES OF UNKNOWNS'
```

```
      READ(5,*)(X(I),I=1,N)
      NCT=0
1     DO 10 I=1,N
      F(I)=FUN(I,X,DX,0,N,KN)
      DO 10 J=1,N
      DX=.005*X(J)
10    D(I,J)=(FUN(I,X,DX,J,N,KN)-F(I))/DX
C UPON ENTRY F=KNOWN VECTOR, UPON RETURN F=SOLUTION
      CALL SOLVEQ(N,1,10,D,F,1,DD,INDX)
      NCT=NCT+1
      DIF=0.
      DO 20 I=1,N
      DIF=DIF+ABS(F(I))
20    X(I)=X(I)-F(I)
      IF(NCT .LT. 20 .AND. DIF .GT. .0001) GO TO 1
      IF(NCT .EQ. 20) WRITE(6,*)' DID NOT CONVERGE',DIF
      WRITE(6,100) (I,X(I),I=1,N)
100   FORMAT(' SOLUTION',/10('-'),/(5(',X(',I2,')=',
      *F10.4)))
      GO TO 50
      END
      FUNCTION FUN(I,X1,DX,J,N,KN)  ! KN(1)= bo
      REAL X(10),X1(10),KN(20)      ! KN(2)=db/dx
      DO 10 K=1,N                   ! KN(3)=mo
10    X(K)=X1(K)                    ! KN94)=dm/dx
      IF(J .GT. 0) X(J)=X(J)+DX     ! kn(5)=n
      IF(I.LT.4) THEN               ! KN(6)=So
      A=(X(1)+X(2)*X(3))*X(3)       ! KN(7)=q*
      Q=X(4)*KN(7)                  ! KN(8)=Uq
      ENDIF                         ! KN(9)=C
      GO TO (1,2,3,4,5), I          ! KN(10)=g
1     FUN=KN(5)*Q*(X(1)+2.*X(3)*SQRT(X(2)**2+1.))**
     &*.666667-KN(9)*A**1.666667*SQRT(ABS(X(5)))
      GO TO 50
2     DA=X(3)*(KN(2)+KN(4)*X(3))
      AG=KN(10)*A
      A2G=AG*A
      FUN=KN(6)-X(5)+Q*Q/(A*A2G)*DA-Q*KN(7)/A2G-(Q/A-
     &*KN(8))*KN(7)/AG-DA*(.5*X(1)+X(2)*X(3)/3.)*
     &*X(3)**2/(A*A)
      GO TO 50
3     FUN=Q*Q*(X(1)+2.*X(3)*X(2))/(KN(10)*A**3)-1.
      GO TO 50                     ! X(1)=b
4     FUN=X(1)-KN(1)-X(4)*KN(2)    ! X(2)=m
      GO TO 50                     ! X(3)=Yc
5     FUN=X(2)-KN(3)-X(4)*KN(4)    ! X(4)=xc
50    RETURN                       ! X(5)=Sf
      END
3     FUN=Q*Q*(X(1)+2.*X(3)*X(2))/(KN(10)*A**3)-1.
      GO TO 50
4     FUN=X(1)-KN(1)-X(4)*KN(2)
      GO TO 50
5     FUN=X(2)-KN(3)-X(4)*KN(4)
50    RETURN
      END
```

Knowns	(value)	Unknowns (guess)
KN(1) - b_o	(4)	X(1) - b (10)
KN(2) - db/dx	(.04)	X(2) - m (1.5)
KN(3) - m_o	(0)	X(3) - Y_c (4.5)
KN(4) - dm/dx	(.01)	X(4) - x_c (150)
KN(5) - n	(.014)	X(5) - S_f (.0022)
KN(6) - S_o	(.035)	
KN(7) - q*	(5)	
KN(8) - U_q	(1)	
KN(9) - C	(1.486)	
KN(10) - g	(32.2)	

The solution obtained for these five unknowns consists of: b = X(1) = 10.2175 ft, m = X(2) = 1.5544, Y_c = X(3) = 4.4716 ft, x_c = X(4) = 155.4364 ft, S_f = X(5) = 0.0022.

A TK-Solver model for solving these five simultaneous equations follows:

```
=============== VARIABLE SHEET =============
St Input----- Name---- Output---- Unit------
              b        10.217456
              x        155.43639
              m        1.5543639
              Y        4.471585
              Sf       .00223028
  4           bo
  .04         dbx
  0           mo
  .01         dmx
              A        76.767842
              T        24.118397
              P        26.746716
              Q        777.18197
  5           qs
  32.2        g
  .014        n
  1.486       Cu
  .035        So
              dAx      .37881413
  1           Uq
```

```
 ============ RULE SHEET =============
S Rule------------------------------------
* b=bo+x*dbx
* m=mo+x*dmx
* A=(b+m*Y)*Y
* T=b+2.*m*Y
* P=b+2.*Y*sqrt(m^2+1)
* Q=x*qs
* Q^2*T-g*A^3=0
* Sf=(n*Q*(P/A)^.66666667/(Cu*A))^2
* dAx=Y*dbx+Y^2*dmx
* So-Sf+Q^2/(g*A^3)*dAx-Q*qs/(g*A^2)-(Q/A-Uq)*qs/
  (g*A)-(.5*b*Y^2+m*Y^3/3.)*dAx/A^2=0
```

The solution obtained upstream and downstream from this critical section are given below from solving Equation 4.6 numerically.

x (ft)	**Y (ft)**	**Q (cfs)**	F_x
150.0	4.500	750.00	0.979
140.0	4.570	700.00	0.934
130.0	4.641	650.00	0.889
120.0	4.717	600.00	0.842
110.0	4.800	550.00	0.793
100.0	4.890	500.00	0.742
90.0	4.989	450.00	0.689
80.0	5.098	400.00	0.634
70.0	5.220	350.00	0.575
60.0	5.358	300.00	0.512
50.0	5.514	250.00	0.44
40.0	5.697	200.00	0.374
30.0	5.912	150.00	0.295
20.0	6.176	100.00	0.200
10.0	6.512	50.00	0.112
0.0	6.969	0.00	0.000
x (ft)	**Y (ft)**	**Q (cfs)**	F_x
160.0	4.440	800.00	1.038
165.0	4.402	825.00	1.045
170.0	4.382	850.00	1.062
175.0	4.357	875.00	1.081
180.0	4.332	900.00	1.100
185.0	4.307	925.00	1.119
190.0	4.282	950.00	1.138
195.0	4.258	975.00	1.156
200.0	4.235	1000.00	1.175

A Mathcad model to solve the position of the critical flow for this example problem follows if you prefer this software over TK-Solver, both of which allow you to "list solve" to create a table of values if you wish to determine the effects that a given variable has on the solution, as is asked for with homework problems.

Mathcad model to solve the critical position for this example problem (EQUNOM.MCD)

$$bo := 4 \quad mo := 0 \quad dbx := .04 \quad dmx := .01 \quad g := 32.2 \quad n := .014 \quad Cu := 1.486 \quad So := .035 \quad qs := 5 \quad Uq := 1$$

$$b := 10 \quad m := 1.5 \quad x := 15C \quad Y := 4.4 \quad Sf := .002 \quad Q := 777 \quad A := 76 \quad T := 24 \quad P := 28 \quad dAx := .3$$

Given $\quad A = (b + m \cdot Y) \cdot Y \quad T = b + 2 \cdot m \cdot Y \quad P = b + 2 \cdot Y \cdot \sqrt{m^2 + 1} \quad dAx = (dbx + dmx \cdot Y) \cdot Y$

$b = bo + x \cdot dbx \quad m = mo + x \cdot dmx \quad Q^2 \cdot T - g \cdot A^3 = 0 \quad Q = x \cdot qs$

Find (b, m, x, Y, Sf, Q, A, T, P, dAx) =

	0
0	10.217
1	1.554
2	155.436
3	4.472
4	2.23·10−3
5	777.182
6	76.768
7	24.118
8	26.747
9	0.379

From these solutions, it should be noted that the greatest depth in the spatially varied flow length occurs at the beginning of the channel where $Y_1 = 6.97$ ft, and that the depth at the end thereof is $Y_2 = 4.24$ ft that is below the critical depth ($Y_{c2} = 4.62$ ft). The S_2 GVF profile that exists downstream from this end takes the depth from 4.24 ft to $Y_{o2} = 2.20$ ft, and its length equals 707 ft from a numerical integration of Equation 4.10.

EXAMPLE PROBLEM 4.46

Modify the previous program that solves the critical position if the lateral inflow channel is prismatic and apply this modified program to solve the following problem: At the beginning of a channel the lateral inflow occurs uniformly over a length of L = 200 ft into a trapezoidal prismatic channel with a bottom width of b = 20 ft, and a side slope m = 1. The slope of the channel's bottom is $S_o = 0.05$, and it has a Manning's n = 0.014. Downstream from the inflow length the channel is long and steep. If the incoming flow rate is q* = 5 cfs/ft, determine the spatially varied flow profiles within the inflow length of the channel.

Solution

For this problem, there are but three unknowns, Y_c, x_c, and S_f. Furthermore, since the channel is prismatic, db/dx and dm/dx are no longer knowns. Thus, the equations that need solution are the previous equations 3, 4, and 5, namely, the critical flow equation, Manning's equation, and the equation that is obtained by setting the numerator of the GVF equation to zero. For use in modifying the previous FORTRAN listing, the following are noted:

Knowns	(Value)
KN (1) – b	(20)
KN (2) – m	(1)
KN (3) – n	(0.014)
KN (4) – S_o	(0.05)
KN (5) – q*	(5)
KN (6) – U_q	(0)
KN (7) – C	(1.486)
KN (8) – g	(32.2)

Unknowns	(Guess)
X (1) – Y_c	(2.5)
X (2) – X_c	(100)
X (3) – S_f	(1.002)

Modified program EPRB4_46.FOR of prismatic channels

```
      REAL F(10),D(10,10),X(10),KN(20),Z(10)
50    WRITE(6,*)' HOW MAY EQS'
      READ(5,*) N
      IF(N .EQ. 0) STOP
      WRITE(6,*)' HOW MANY KNOWNS?'
```

```
      READ(5,*) NK
      WRITE(6,*)' GIVE ME THESE KNOWN VALUES.'
      READ(5,*) (KN(I),I=1,NK)
      WRITE(6,*) ' PROVIDE',N,' ESTIMATES OF UNKNOWNS'
      READ(5,*)(X(I),I=1,N)
      NCT=0
1     DO 10 I=1,N
      F(I)=FUN(I,X,DX,0,N,KN)
      DO 10 J=1,N
      DX=.005*X(J)
10    D(I,J)=(FUN(I,X,DX,J,N,KN)-F(I))/DX
      WRITE(6,304)(F(I),I=1,N)
304   FORMAT(5F12.4)
      NCT=NCT+1
      DO 20 K=1,N-1
      DO 20 I=N,K+1,-1
      IF(ABS(D(K,I)).LT.1.E-8) GO TO 20
      FAC=D(I,K)/D(K,K)
      F(I)=F(I)-FAC*F(K)
      DO 18 J=K+1,N
18    D(I,J)=D(I,J)-FAC*D(K,J)
20    CONTINUE
      I=N
      Z(I)=F(I)/D(I,I)
      DIF=ABS(Z(I))
      X(N)=X(N)-Z(I)
25    I1=I-1
      SUM=0.
      DO 30 J=I,N
30    SUM=SUM+Z(J)*D(I1,J)
      Z(I1)=(F(I1)-SUM)/D(I1,I1)
      I=I1
      X(I1)=X(I1)-Z(I1)
      DIF=DIF+ABS(Z(I1))
      IF(I .GT. 1) GO TO 25
      IF(NCT .LT. 20 .AND. DIF .GT. .0001) GO TO 1
      IF(NCT .EQ. 20) WRITE(6,*)' DID NOT CONVERGE',DIF
      WRITE(6,100) (I,X(I),I=1,N)
100   FORMAT(' SOLUTION',/(5(' ,X(',I2,')=',F10.4)))
      GO TO 50
      END
      FUNCTION FUN(I,X1,DX,J,N,KN)
      REAL X(10),X1(10),KN(20)
      DO 10 K=1,N
10    X(K)=X1(K)
      IF(J .GT. 0) X(J)=X(J)+DX
      A=(KN(1)+KN(2)*X(1))*X(1)
      Q=X(2)*KN(5)
      GO TO (1,2,3), I
1     FUN=KN(3)*Q*(KN(1)+2.*X(1)*SQRT(KN(2)**2+1.))**.666667-
     &KN(7)*A**1.666667*SQRT(ABS(X(3)))
      GO TO 50
2     AG=KN(8)*A
      A2G=AG*A
      FUN=KN(4)-X(3)-Q*KN(5)/A2G-(Q/A-KN(6))*KN(5)/AG
```

```
      GO TO 50
3     FUN=Q*Q*(KN(1)+2.*KN(2)*X(1))/(KN(8)*A**3)-1.
50    RETURN
      END
```

This program has its own solution of the linear system of equations built in. The solution yields X(1) = Y_c = 2.454 ft, X(2) = x_c = 93.03 ft, and S_f = 0.0024. The gradually varied solutions are given below.

Solution upstream from the critical position

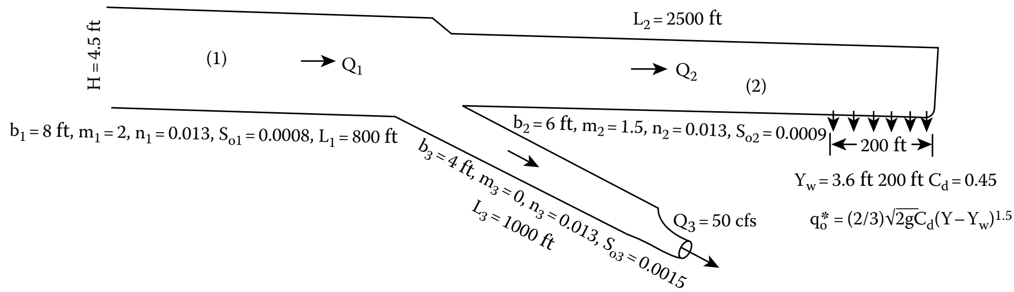
x	Y	Q	E	M	F_r
90.0	2.600	450.0	3.511	180.48	0.78
80.0	2.395	400.0	3.259	154.58	0.80
70.0	2.257	350.0	3.011	130.51	0.74
60.0	2.125	300.0	2.757	107.80	0.65
50.0	1.984	250.0	2.494	86.47	0.56
40.0	1.829	200.0	2.219	66.62	0.46
30.0	1.656	150.0	1.928	48.43	0.35
20.0	1.455	100.0	1.614	32.1	0.23
10.0	1.205	50.0	1.265	18.15	0.10
0.0	0.841	0.0	0.841	7.27	0.00

Solution downstream from the critical position

x	Y	Q	E	M	F_r
95.0	2.200	475.0	3.669	195.42	1.47
100.0	2.401	500.0	3.743	206.62	1.24
110.0	2.693	550.0	3.951	232.76	1.05
120.0	2.791	600.0	4.173	260.90	1.11
130.0	2.884	650.0	4.390	289.98	1.18
140.0	2.974	700.0	4.604	319.94	1.24
150.0	3.060	750.0	4.814	350.76	1.30
160.0	3.143	800.0	5.021	382.39	1.36
170.0	3.224	850.0	5.225	414.80	1.41
180.0	3.302	900.0	5.427	447.98	1.47
190.0	3.378	950.0	5.625	481.89	1.52
200.0	3.452	1000.0	5.822	516.51	1.58

EXAMPLE PROBLEM 4.47

A channel system consists of a main upstream channel with b_1 = 8 ft, m_1 = 2, n_1 = 0.013, S_{o1} = 0.0008, and L_1 = 800 ft, a larger branch with b_2 = 6 ft, m_2 = 1.5, n_2 = 0.013, S_{o2} = 0.0009, and L_2 = 2500 ft, and a smaller branch with b_3 = 4 ft, m_3 = 0, n_3 = 0.013, S_{o3} = 0.0015, and L_3 = 800 ft, as shown on the sketch below. The flow rate at the end of the smaller branch is controlled by automatic valves at an amount equal to Q_3 = 50 cfs. The larger branch has a side weir with a crest height of Y_w = 3.6 ft above the channel bottom over its last 200 ft of length from which the water leaves the channel. The main channel receives its water supply from a reservoir whose water surface elevation is H = 4.5 ft above the bottom of the channel. Determine the depths and the flow rates throughout this channel system. Ignore all minor losses and assume that the discharge from the side weir has a discharge coefficient of 0.45.



Solution

There are a number of possibilities that may influence the depths, etc., in this problem that may result in flows being different than near normal, such as (1) the side weir at the end of channel 2 may cause an M_1 GVF profile in this channel, and the effect may be felt through the junction of the three channels to the reservoir; (2) the side weir may have greater discharge capacity than the system can deliver; in which event the depth just upstream from the side weir will be critical; (3) the critical depth may occur in channel 1 at the junction of the three channels and control the flow.

As reference values to start guessing how to proceed with this problem, the following are computed: (a) based on Q_3 = 50 cfs in channel 3, Y_{o3} = 2.60 ft and E_{o3} = 2.96 ft; (b) based on the reservoir head H = 4.5 ft, the uniform flow conditions in channel 1 are Q_{o1} = 369.3 cfs and Y_{o1} = 3.96 ft ; and (c) taking the difference in flow rates for channel 2 (Q_2 = 319.3 cfs), its normal conditions are: Y_{o2} = 4.19 ft and E_{o2} = 4.79 ft. Since the length of the side weir (200 ft) is quite long, the possibility (2) mentioned above of the critical flow in channel 2 at the beginning of the later outflow is possible. Let us investigate this first. The limiting critical flow rates for different depths at this position are

Y_c (ft)	3.6	3.7	3.8
Q_c (cfs)	363.99	383.38	403.88

The program EPRB4_44, developed in Example Problem 4.44, can be used to obtain solutions across the lateral outflow length of channel 2. Starting with a selected depth at the end of this channel gives the flow rates and the depths at the beginning of the lateral outflow in the first table below. These solutions used Δx = 2 ft, but printed out every 10th value. Notice the flow rate associated with Y_{2e} = 4.888 ft equals the critical depth in channel 2 associated with Y_c = 3.6 ft, a condition that would occur if the lateral outflow caused the water level to be at the weir height at its beginning. This solution is provided in the second table below.

Y_{2e}	Q_{2b}	Y_{2b}
5.05	398.66	3.785
5.00	388.37	3.778
4.90	354.72	3.798
4.85	354.72	3.820
4.88	361.95	3.807
4.888	363.84	3.803
4.75	382.72	3.861
4.70	314.71	3.878
4.65	300.06	3.893
4.60	284.83	3.904

x	Y	Q	q_o^*	M	F_r
200.0	4.888	0.00	3.5192	130.07	0.000
180.0	4.852	69.16	3.3736	130.06	0.011
160.0	4.783	134.06	3.0969	132.19	0.045
140.0	4.683	192.33	2.7148	136.00	0.101
120.0	4.559	242.18	2.2614	140.86	0.179
100.0	4.417	282.58	1.7773	146.08	0.275
80.0	4.265	313.35	1.3061	151.04	0.388
60.0	4.115	335.19	0.8897	155.30	0.509
40.0	3.979	349.54	0.5618	158.67	0.630
20.0	3.871	358.38	0.3403	161.23	0.736
0.0	3.803	363.84	0.2207	163.25	0.811

This solution gives a depth of 3.803 ft at the beginning of the lateral outflow. Thus, it appears that if the critical flow controls just before the lateral outflow, the depth might be about 3.7 ft, with a flow rate of about 380 cfs, and a hydraulic jump will occur near the beginning of the lateral outflow section. The two solutions below solve the GVFs in Channels 2 and 3, if this were to be the case.

GVF-Solution in channel 1
$Q = 380$ cfs, $b = 6$ ft, $m = 1.5$, $n = 0.013$,
$S_o = 0.0009$

x	Y	E	M	F_r
2300.0	3.750	4.930	171.42	0.93
2200.0	4.085	4.999	174.67	0.67
2000.0	4.276	5.072	178.44	0.56
1800.0	4.370	5.115	180.75	0.52
1600.0	4.426	5.143	182.30	0.49
1400.0	4.464	5.162	183.38	0.48
1200.0	4.489	5.176	184.15	0.47
1000.0	4.508	5.185	184.71	0.46
800.0	4.521	5.192	185.12	0.45
600.0	4.530	5.198	185.42	0.45
400.0	4.537	5.201	185.64	0.45
200.0	4.542	5.204	185.81	0.45
0.0	4.546	5.206	185.93	0.44

GVF-Solution in channel 2
$Q = 430$ cfs, $b = 8$ ft, $m = 2$, $n = 0.013$,
$S_o = 0.0008$

x	Y	E	M	F_r
800.0	4.805	5.206	234.17	0.26
700.0	4.763	5.175	231.60	0.27
600.0	4.724	5.147	229.21	0.28
500.0	4.686	5.119	226.99	0.28
400.0	4.651	5.094	224.94	0.29
300.0	4.617	5.071	223.05	0.30
200.0	4.585	5.049	221.31	0.31
100.0	4.556	5.028	219.72	0.32
0.0	4.528	5.010	218.27	0.33

To get the starting depth of 4.805 ft for the solution in channel 1, the specific energy at the upstream end of channel 2 of 5.206 ft was used to solve $Y_{12} = 4.805$ for $Q = 430$ cfs. Notice that the solution for channel 1 gives a depth at its entrance of 4.528 ft, with a specific energy of 5.010 ft, which is above the reservoir's water surface elevation of 4.5 ft, thus obviously the flow rate will be less than 430 cfs, and the assumption that the critical flow will occur just upstream from the lateral outflow is not correct. Thus, the solution will be obtained by guessing the depth at the end of the lateral outflow in channel 2; then solve the ODE over the downstream 200 length of lateral outflow; next solve the ODE in channel 2 from 2300 ft to its beginning; followed by solving the specific energy equation at the end of channel 1 to find its end depth Y_{12}; followed by solving the GVF through it and checking whether the specific energy at its beginning equals H = 4.5 ft. When the match is obtained, then the GVF through channel 3 can be solved. The solutions below start with Y_{2e} = 4.665 ft, which from the above table gives a flow rate of Q_2 = 300.06 cfs.

Solution over the lateral outflow

x	Y	Q	Q*	M	Fr
200.0	4.650	0.00	2.5903	115.14	0.000
190.0	4.638	25.70	2.5466	114.77	0.002
180.0	4.621	50.87	2.4831	114.73	0.007
170.0	4.598	75.30	2.4013	115.01	0.017
160.0	4.571	98.83	2.3028	115.59	0.029
150.0	4.539	121.31	2.1900	116.42	0.046
140.0	4.503	142.59	2.0650	117.47	0.065
130.0	4.463	162.58	1.9304	118.70	0.088
120.0	4.420	181.18	1.7888	120.06	0.113
110.0	4.375	198.34	1.6430	121.53	0.141
100.0	4.328	214.03	1.4956	123.05	0.171
90.0	4.280	228.25	1.3492	124.58	0.203
80.0	4.231	241.03	1.2062	126.10	0.237
70.0	4.182	252.40	1.0690	127.58	0.271
60.0	4.134	262.43	0.9394	129.00	0.307
50.0	4.087	271.22	0.8191	130.33	0.342
40.0	4.043	278.85	0.7091	131.58	0.377
30.0	4.000	285.44	0.6102	132.73	0.412
20.0	3.961	291.09	0.5227	133.78	0.445
10.0	3.925	295.93	0.4466	134.75	0.476
0.0	3.893	300.06	0.3814	135.63	0.505

Solution through channel 2
Q = 300.06 cfs, b = 6 ft, m = 1.5,
n = 0.013, S_o = 0.0009

x	Y	E	M	Fr
2300	3.893	4.551	135.63	0.50
2200	3.923	4.566	136.33	0.49
2000	3.966	4.588	137.38	0.47
1800	3.994	4.604	138.10	0.46
1600	4.013	4.614	138.61	0.45
1400	4.027	4.622	138.97	0.44
1200	4.036	4.627	139.23	0.44
1000	4.043	4.631	139.41	0.44
800	4.048	4.634	139.55	0.43
600	4.052	4.636	139.65	0.43
400	4.054	4.637	139.72	0.43
200	4.056	4.638	139.77	0.43
0	4.057	4.639	139.80	0.43

Solution through channel 1
$Q = 350.06$ cfs, $b = 8$ ft, $m = 2$,
$n = 0.013$, $S_o = 0.0008$

x	Y	E	M	F_r
800	4.253	4.639	177.85	0.28
700	4.217	4.613	176.05	0.28
600	4.184	4.589	174.40	0.29
500	4.152	4.567	172.90	0.30
400	4.123	4.547	171.53	0.31
300	4.096	4.528	170.29	0.32
200	4.071	4.511	169.17	0.33
100	4.048	4.496	168.15	0.33
0	4.026	4.482	167.24	0.34

Solution through channel 3
$Q = 50$ cfs, $b = 4$ ft, $m = 0$, $n = 0.013$,
$S_o = 0.0015$

x	Y	E	M	F_r
0	4.520	4.639	45.16	0.05
200	4.761	4.868	49.40	0.04
400	5.008	5.105	54.03	0.04
600	5.261	5.349	59.05	0.03
800	5.519	5.599	64.44	0.03
1000	5.782	5.854	70.22	0.03

Notice that the above solution through channel 1 produces a specific energy of 4.482 ft, which is essentially the water surface elevation of the reservoir, so these tables represent the solution to the problem.

As homework/project you might consider writing a computer program to handle problems such as this one that solves the appropriate system of algebraic equations and ODEs simultaneously.

4.18 TILE DRAINAGE

An important application of lateral inflow along the full length of a pipe is the drainage of fields, or foundations with tile drains. The details of their spacing and the depth involve the properties of the porous media (soil), the depth to which the water table is to be maintained below the land surface, and the amount of the incoming flow. For these details, consult a book dealing with Drainage Engineering. In this section, we will be concerned only with the open channel flow in the drain itself. Nowadays, practically all of these drains are circular, and so the solution methodologies discussed will be for a pipe, but the principles apply for other sections also.

The operation of the drainage tiles can result in a variety of gradually varied open channel flow situations. If the slope of the drain is large then much, if not most, of the flow within the pipe will be supercritical. If the slope is small, then the subcritical flow with exist throughout the drain, and if the slope is moderately steep, then the subcritical flow may exist in the upper portion of the pipe and the supercritical flow in the lower portion thereof. These various possibilities are just a special application of the gradually varied flows as discussed earlier, in which a constant diameter circular section defines the channel. If an upper portion of the drain receives a lateral inflow, and the last portion receives no lateral inflow but is used to convey this water to some other location, then only that portion with the lateral inflow may be a gradually varied flow. For example, if the downstream pipe with no lateral inflow has a constant bottom slope and is long, then the depth of the uniform flow in the downstream pipe will be the depth at the end of the portion of the pipe into which the seepage flow occurs. For problems in which this uniform depth Y_o provides the downstream boundary condition for the spatially varied flow, the depth will generally be less than Y_o at the beginning of the tile. This occurrence is the result of the slope of the energy line S_f becoming zero at its beginning as Q approaches zero, despite the fact that the lateral inflow term's contribution to the numerator of the GVF ODE is negative, which would suggest an increase in depth toward the beginning of the tile. Only if the bottom slope of the portion of the tile receiving the lateral inflow is smaller than the bottom slope of the portion downstream therefrom without a lateral inflow, will the depth of flow increase throughout the entire upstream portion of the pipe. (See problems at the end of this chapter of examples.)

In the remainder of this section, we will deal with the determination of tile drain lengths that will result in the pipe just being full at its beginning. For such design problems, at the end of each tile lateral one of either two common conditions may apply. (1) If the lateral discharges into another tile, then a solution of the flow in this main tile provides the depth of flow at its downstream end, and this depth will be above the critical depth. (2) The lateral discharges free from its end so that the critical flow occurs here. The slope of the tile will be assumed to be small enough so that the subcritical flow occurs throughout its length. Only steady-state flows will be considered, and for these, the amount of lateral inflow per unit length q* is known. One design criteria that may be used is to determine the length of tile with a specified diameter D, bottom slope S_o, and Manning's n so that it is full at its beginning. With these assumptions, the length is the unknown to be solved. The equations that govern are: (1) the ODE for such seepage lateral inflow,

$$\frac{dY}{dx} = \frac{S_o - S_f - 1.5Qq^*/(gA^2)}{1 - F_r^2}$$

and (2) the downstream boundary condition, which we will take here as the critical flow at the end of the pipe, or

$$Q^2T - gA^3 = 0$$

in which the flow rate Q at any position x along the pipe is given by Q = xq* if q* is constant. If q* is not constant, then $Q = \int q^*(x)dx$.

One might approach the problem in which the critical flow occurs at the tile's end as one in which two unknowns L and Y_c are to be solved simultaneously by the above two equations. However, a better approach is to take the current length, use this to determine the flow rate at the end of the channel, and use this flow rate to determine the critical depth. Then start the GVF solution 10% above this critical depth and continue the GVF solution until the depth equals 0.93D, i.e., just before the water depth reaches the top of the pipe. If the depth at the downstream end of the tile drain is specified, then the only unknown is the length of the drain since the flow rate at the end of the pipe is Q = Lq* (providing q* is constant.)

The program TDRAIN given below is designed to solve the length of the tile drain, with a specified diameter, wall roughness, and the bottom slope is required so that it is full (or Y = 0.93D) at its beginning. The input data to this program consists of (1) DI = the pipe diameter in basis units; (2) FN = Manning's roughness coefficient; (3) SO = the bottom slope; (4) q* = the lateral inflow per unit length of drain; (5) g = acceleration of gravity; (6) TOL = accuracy requirement for ODESOLF to satisfy; (7) icrit = 1 if the downstream boundary condition is critical flow, or icrit = 0 if depth at downstream end is given; (8) L = a guess for length; and (9) YC = a guess for the critical depth, or if icrit = 0, then this is the value specified for the depth at the downstream end of the tile drain. The essential steps accomplished by the program are as follows: (a) determine the flow rate at the end of the drain by Q = q*L, using the current value for the pipe length, (b) solve the critical flow equation (this step is omitted if the downstream depth is specified), (c) solve the above ODE from the end x = L to the beginning of the pipe x = 0, and define the function that is to be driven to zero as $F(L) = (Y_u)_{ode} - 0.93D = 0$, (d) implement the Newton method in improving the length L by $L^{(m+1)} = L^{(m)} - F(L)/(dF/dL) = L^{(m)} - \Delta L F_1/(F_2 - F_1)$, in which F_1 is the value of the function evaluated using $L^{(m)}$, and F_2 is its value evaluated using $L^{(m)} + \Delta L$.

Program TDRAIN.FOR

```
      REAL Y(1),DY(1),XP(1),YP(1,1),W(1,13)
      LOGICAL NFS
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ DI,G,FL,TOL,FN,SO,QS,YC,DLIM,DI2,CU,ICRIT
      WRITE(*,*)' Give: Dia,n,So,q*,g,TOL,icrit,',' (guess for) L,
     &Yc(Yend)'
      READ(*,*) DI,FN,SO,QS,G,TOL,ICRIT,FL,YC
      DLIM=.93*DI
      DI2=.25*DI**2
      CU=1.486
      IF(G.LT.30.) CU=1.
      NCT=0
10    F1=FUN(FL)
      DIF=F1/(FUN(FL+1.)-F1)
      NCT=NCT+1
      FL=FL-DIF
      WRITE(*,*)' NCT=',NCT,DIF,FL
      IF(ABS(DIF).GT. .05 .AND. NCT.LT.30) GO TO 10
      WRITE(3,100) FL,QS*FL
100   FORMAT(' Length =',F9.1,' Flow rate =',F10.5)
      DX=FL/20.
      H11=.05
      HMIN=1.E-5
      XX=FL
      NFS=.TRUE.
      WRITE(3,109)
109   FORMAT(/,'   X    Depth')
      WRITE(3,110) FL,YC
      Y(1)=1.01*YC
      DO 20 I=1,20
      XZ=FL-DX*FLOAT(I)
      CALL ODESOLF(Y,DY,1,XX,XZ,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      WRITE(3,110) XZ,Y
110   FORMAT(F8.1,F10.3)
      NFS=.FALSE.
20    XX=XZ
      END
      FUNCTION FUN(FLL)
      EXTERNAL DYX
      LOGICAL NFS
      REAL Y(1),DY(1),XP(1),YP(1,1),W(1,13)
```

```
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ DI,G,FL,TOL,FN,SO,QS,YC,DLIM,DI2,CU,ICRIT
      HMIN=1.E-5
      Q=QS*FLL
      IF(ICRIT.EQ.0) THEN
      Y(1)=YC
      ELSE
C  Solves Critical depth
      NCT=0
1     IF(YC.GT.DI) YC=.95*DI
      COSB=1.-2.*YC/DI
      BETA=ACOS(COSB)
      SINB=SIN(BETA)
      A=DI2*(BETA-SINB*COSB)
      F2=Q*Q*DI*SINB-G*A**3
      NCT=NCT+1
      IF(MOD(NCT,2).EQ.0) GO TO 2
      YC1=YC
      YC=1.05*YC
      F1=F2
      GO TO 1
2     DIF=(YC-YC1)*F1/(F2-F1)
      YC=YC1-DIF
      IF(ABS(DIF).GT. .00001 .AND. NCT.LT.40) GO TO 1
      IF(NCT.GE.40) WRITE(*,*)' Failed with critical',YC,DIF
      Y(1)=1.1*YC
      ENDIF
      H11=.05
      NFS=.TRUE.
      CALL ODESOLF(Y,DY,1,FLL,0.,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      FUN=DLIM-Y(1)
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ DI,G,FL,TOL,FN,SO,QS,YC,DLIM,DI2,CU,ICRIT
      Q=QS*XX
      YY=Y(1)
      IF(YY.LT. .001) YY=.001
      IF(YY.GT.DI) YY=DI-1.E-4
      COSB=1.-2.*YY/DI
      BETA=ACOS(COSB)
      SINB=SIN(BETA)
      P=DI*BETA
      A=DI2*(BETA-COSB*SINB)
      SF=(FN*Q/CU*(P/A)**.66666667/A)**2
      FR2=Q*Q/G*DI*SINB/A**3
      DY(1)=(SO-SF-1.5*Q*QS/(G*A*A))/(1.-FR2)
      RETURN
      END
```

EXAMPLE PROBLEM 4.48

Determine the length of a 0.15 m diameter tile drain that will receive a constant lateral inflow of $q^* = 0.00009339\,m^2/s$ if at its end there is a free outfall. The bottom slope of the drain is to be zero, and its Manning's n = 0.016.

Solution

First estimate the length of drain needed, e.g., about 60 m. Next determine the critical flow associated with $Q = 60q^*$, e.g., Y_c is about 0.057 m. The input to program TDRAIN to solve this problem is

```
.15 .016 0. .00009339 9.81 .0001 60 .057
```

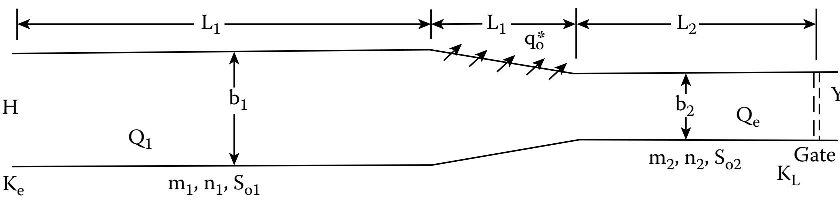
The solution provided is
Length = 56.1 Flowrate = .00524

x	Depth
56.1	0.066
53.3	0.096
50.5	0.106
47.7	0.112
44.9	0.118
42.1	0.122
39.3	0.125
36.5	0.128
33.7	0.130
30.9	0.132
28.1	0.134
25.2	0.135
22.4	0.137
19.6	0.137
16.8	0.138
14.0	0.139
11.2	0.139
8.4	0.139
5.6	0.139
2.8	0.140
0.0	0.140

4.19 DOWNSTREAM CONTROLS IN NONPRISMATIC CHANNELS

In practice, channels are not infinitely long upstream and downstream from nonprismatic sections, or where the lateral inflow or the outflow occurs. Therefore, the depths downstream from these special sections are not normal under subcritical flow conditions, but are rather effected by controls in the channel downstream therefrom. An example of such a situation is a side weir at some mid position in a channel that contracts in size over the length of this outflow section, and at a modest distance downstream therefrom contains a gate that controls the flow rate. Instead of a gate, the channel may flow into a reservoir with a constant (known) water surface elevation, or may end in a free overall. In all cases, if the distance between the side weir and the end of the channel is not extremely large, then the depth will not be normal at the end of the side weir.

The methods described earlier for solving algebraic and the ODEs for GVF profiles simultaneously are effective means for obtaining solutions to problems of this type. The exact details involved in the solution will vary depending upon the downstream and the upstream conditions. The general method of solution can be illustrated, however, by considering a channel being supplied water by a reservoir at its upstream end. At some length L_1 downstream from the channel's beginning there is a reduction in the channel size over a length L_t where water is being taken out of the channel. For our example, it will be assumed that this outflow takes place uniformly over the length of the transition (i.e., nonprismatic portion of the channel). Further downstream at a distance L_2 from the outflow, a gate exists in the channel, and there is a short transition to a rectangular section just before the gate. The sketch below illustrates this problem. To simplify the problem, assume that the amount of outflow q_o^* is known.



The following equations describe this problem:

The energy equation at the channel's beginning

$$F_1 = H - Y_1 - (1 + K_e)Q_1^2/(2gA_1^2) = 0 \tag{4.18}$$

The energy equation at the channel's end across the downstream gate

$$F_2 = Y_2 + Q_2^2/(2gA_2^2) - Y_3 - (1 + K_L)Q_2^2/(2gA_3^2) = 0 \tag{4.19}$$

GVF Solution starting just upstream from the gate with Y2 and terminating just downstream from the reservoir with Y1

$$F_3 = Y_1 - Y_{GVF} = 0 \tag{4.20}$$

In solving the GVF, it will be necessary that the solution for a prismatic channel, without a lateral inflow or outflow be from the gate until it has advanced to the position of the outflow section, and while carrying out the computations across this nonprismatic channel, both the lateral outflow term and the nonprismatic channel term be included in the ODE. After passing through this mid section, the solution of the GVF equation will again only include $S_o - S_f$ in the numerator. In other words, the computer program that implements the solution must have the logic in it so that the appropriate additional terms are included in the numerator of the ODE only while x is within the lateral outflow portion of the channel. The following FORTRAN program is designed to solve problems of the above type.

FORTRAN program SOLSID.FOR to solve Equations 4.18 through 4.20.

```
      PARAMETER (N=3)
      EXTERNAL DYX
      REAL F(N),D(N,N),X(N),KL1,KE1,KL,KE,X1(N),W(2,13),XP(2),
     &YP(2,2),Y(1)
      CHARACTER*20 FNAME
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,FM1,B2,FM2,B3,H,G,G2,KL1,KE1,FL,FL1,FLT,TOL,
     &FN1,FN2,SO1,SOT,SO2,CC,Uqx,X,X1,Y3,DB,DM,A3,Q2G,QE,QSS,FLE,
     &IQT,IBC
```

```
      EQUIVALENCE (Q,X(1)),(Y1,X(2)),(Y2,X(3))
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,FN1,SO1,B1,FM1,FL1,FN2,SO2,B2,
     &FM2,FL2,B3,FLT,SOT,H,QSS,G,KL,KE,Y3,Uq,ANGLE,IQT,IBC'
      READ(*,*)IOUT,TOL,ERR,FN1,SO1,B1,FM1,FL1,FN2,SO2,B2,FM2,FL2,
     &B3,FLT,SOT,H,QSS,G,KL,KE,Y3,Uq,ANGLE,IQT,IBC
C IQT=1 outflow;IQT=2 seepage;IQT=3 bulk inflow
C IBC=1 downstr. gate;IBC=2 critical;IBC=3 downstr. Reser.
      IF(IOUT.NE.6 .OR. IOUT.NE.0) THEN
      WRITE(*,*)' Give filename for output'
      READ(*,144) FNAME
144   FORMAT(A20)
      OPEN(IOUT,FILE=FNAME,STATUS='NEW')
      ENDIF
      DB=(B2-B1)/FLT
      DM=(FM2-FM1)/FLT
      Uqx=Uq*COS(.17455329*ANGLE)
      FLE=FL1+FLT
      FL=FLE+FL2
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      A3=(Y3*B3)**2*G2
      KL1=1.+KL
      KE1=1.+KE
      WRITE(*,*)' GIVE guess for: Q,Y1,Y2'
      READ(*,*) X
50    NCT=0
1     DO 10 I=1,N
      F(I)=FUN(I,DX,0)
      DO 10 J=1,N
      DX=.005*X(J)
10    D(I,J)=(FUN(I,DX,J)-F(I))/DX
      WRITE(*,505) X
505   FORMAT(' Unknowns',5F12.5)
      WRITE(*,510) F
510   FORMAT(' Equations ',5F12.5)
C  Solves system of equations using Gaussian Elimination
      DO 12 J=1,N-1
      DO 12 I=J+1,N
      FAC=D(I,J)/D(J,J)
      F(I)=F(I)-FAC*F(J)
      DO 12 K=J+1,N
12    D(I,K)=D(I,K)-FAC*D(J,K)
      F(N)=F(N)/D(N,N)
      X(N)=X(N)-F(N)
      SUM=ABS(F(N))
      DO 16 I=N-1,1,-1
      FAC=0.
```

```
      DO 14 J=I+1,N
14    FAC=FAC+D(I,J)*F(J)
      F(I)=(F(I)-FAC)/D(I,I)
      X(I)=X(I)-F(I)
16    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,110) NCT,SUM,X,QE
110   FORMAT(' NCT=',I2,F12.2,4F10.4)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 1
      QE=X(1)+QSS*FLT
      WRITE(*,100) X,QE
      WRITE(IOUT,100) X,QE
100   FORMAT(' Q =',F10.2,' Y1 =',F10.2,' Y2 =',F10.2, ' QE =',F10.2)
      WRITE(*,*)' Give Del x for sol. of x Y values'
      READ(*,*) DELX
      IF(DELX.LT. .01) GO TO 150
      H1=-.01
      HMIN=.00001
      XX=FL
      Y(1)=X(3)
      WRITE(IOUT,120) XX,Y(1)
120   FORMAT(2F10.4)
      WRITE(*,120) XX,Y(1)
18    IF(XX.LE.FLE+.1 .AND. XX.GT.FL1) THEN
      XZ=XX-.1*DELX
      ELSE
      XZ=XX-DELX
      ENDIF
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      WRITE(IOUT,120) XZ,Y(1)
      WRITE(*,120) XZ,Y(1)
      XX=XZ
      IF(XX.GT. .01) GO TO 18
150   CLOSE(IOUT)
      END
      FUNCTION FUN(II,DX,J)
      EXTERNAL DYX
      REAL X(3),X1(3),W(1,13),KL1,KE1,Y(1),DY(1),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,FM1,B2,FM2,B3,H,G,G2,KL1,KE1,FL,FL1,FLT,TOL,
     &FN1,FN2,SO1,SOT,SO2,CC,Uqx,X,X1,Y3,DB,DM,A3,Q2G,QE,QSS,FLE,
     &IQT,IBC
      H1=-.05
      HMIN=.001
      DO 10 I=1,3
10    X(I)=X1(I)
      IF(J.GT.0) X(J)=X(J)+DX
      Q2G=X(1)*X(1)/G2
      QE=X(1)+FLT*QSS
      GO TO (1,2,3),II
```

```
1      FUN=H-X(2)-KE1*Q2G/((B1+FM1*X(2))*X(2))**2
       RETURN
2      Y(1)=X(3)
       CALL ODESOL(Y,DY,1,FL,FLE,TOL,H1,HMIN,1,XP,YP,W,DYX)
       H1=-.05
       DXX=(FLE-FL1)/10.
       XX=FLE
       DO 5 I=1,10
       XZ=XX-DXX
       CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
5      XX=XZ
       H1=-.05
       CALL ODESOL(Y,DY,1,FL1,0.,TOL,H1,HMIN,1,XP,YP,W,DYX)
       FUN=X1(2)-Y(1)
       RETURN
3      IF(IBC.EQ.1) THEN
       FUN=X(3)+(QE/((B2+FM2*X(3))*X(3)))**2/G2-Y3-KL1*QE*QE/A3
       ELSE IF(IBC.EQ.2) THEN
       FUN=X(3)+(QE/((B2+FM2*X(3))*X(3)))**2/G2-(1.+KL1/2.)*
      &((QE/B3)**2/G)**.333333333
       ELSE
       FUN=X(3)-Y3
       ENDIF
       RETURN
       END
       SUBROUTINE DYX(XX,Y,DY)
       REAL Y(1),DY(1),KL1,KE1,X(3),X1(3)
       COMMON /TRAS/B1,FM1,B2,FM2,B3,H,G,G2,KL1,KE1,FL,FL1,FLT,TOL,
      &FN1,FN2,SO1,SOT,SO2,CC,Uqx,X,X1,Y3,DB,DM,A3,Q2G,QE,QSS,FLE,
      &IQT,IBC
       YY=ABS(Y(1))
       IF(XX.GT.FLE) THEN
       SO=SO2
       FN=FN2
       EXTERM=0.
       QQ=QE
       BB=B2
       FM=FM2
       ELSE IF(XX.GT.FL1) THEN
       SO=SOT
       FN=.5*(FN1+FN2)
       QQ=X(1)+QSS*(XX-FL1)
       EXTERM=(DB+DM*YY)*YY
       BB=B1+DB*(XX-FL1)
       FM=FM1+DM*(XX-FL1)
       ELSE
       EXTERM=0.
       QQ=X(1)
       SO=SO1
       FN=FN1
```

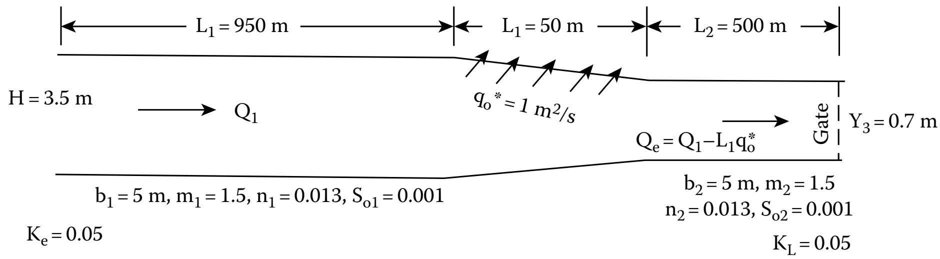
```
BB=B1
FM=FM1
ENDIF
P=BB+2.*SQRT(FM*FM+1.)*YY
A=(BB+FM*YY)*YY
A2=A*A*G
IF(ABS(EXTERM).GT.0.) THEN
IF(IQT.EQ.3) THEN
HADX=EXTERM*(.5*BB+FM*YY/3.)*(YY/A)**2
ENDIF
EXTERM=EXTERM*QQ*QQ/(A*A2)-QQ*QSS/A2
IF(IQT.EQ.2) EXTERM=EXTERM-QQ*QSS/(2.*A*A2)
IF(IQT.EQ.3) EXTERM=EXTERM-(QQ/A-Uqx)*QSS/(G*A)-HADX
ENDIF
SF=((FN*QQ/CC)*(P/A)**.666666667/A)**2
DY(1)=(SO-SF+EXTERM)/(1.-QQ/A*QQ/A2*(BB+2.*FM*YY))
RETURN
END
```

This program assumes that the transition just before the downstream gate can be handled by using a minor loss coefficient. Thus, the depth Y3 that is given as input is the depth downstream from the gate, if a gate is specified at the downstream end. The gate is assumed to be in a rectangular section that is B3 wide. Immediately upstream from the gate, the channel has a bottom width B2 and a side slope FM2. Upstream from the lateral outflow section, the channel has a bottom width B1, a side slope FM1 and the difference between these channel size parameters with a 1 and a 2 subscript determines the magnitude of the nonprismatic term $\partial A/\partial x$. The flow rate in the channel downstream from the outflow is QE and is obtained by subtracting the outflow, which is specified by SSQ (q_o^*), times the length of the section FLT, over which the outflow takes place. The length of this outflow section FLT needs to be some multiple (include 1) of the length between stations.

The program will handle problems that have an inflow, as well as an outflow. For an inflow, the input variable Uq represents the velocity of the inflow and the angle of this inflow from the direction of the channel flow is given by ANGLE in degrees. The input variable IQT must be 1, 2, or 3 for a bulk lateral outflow, an inflow, or an outflow from seepage, or a bulk lateral outflow, respectively. The input variable IBC determines the type of downstream boundary condition that will be used. If IBC = 1, then a gate is assumed to exist at the downstream end of a rectangular channel with a width B3 and depth Y3 as described above. If IBC = 2, then the critical flow occurs at the downstream end. The channel is assumed to be rectangular at this critical flow section with a width B3. For this downstream boundary condition, the value given to Y3 is ignored. If IBC = 3, then it is assumed that the channel discharges into a reservoir with a water surface elevation Y3 above the channel bottom. The channel, as it enters the reservoir, is taken as rectangular with a width B3.

EXAMPLE PROBLEM 4.49

A trapezoidal channel with an upstream bottom width $b_1 = 5$ m, a side slope $m_1 = 1.5$, and bottom slope $S_{o1} = 0.001$ is 950 m long before a lateral outflow section occurs that is 50 m long that reduces the channel to a size $b_2 = 4$ m, $m_2 = 1$. The bottom slope remains at $S_{o2} = S_{o1} = 0.001$. Manning's n is also constant throughout the channel and is n = 0.013. At a distance 500 m downstream from the outflow section, a gate exists in a rectangular section with $b_3 = 4$ m. The gate causes the depth downstream from it to be $Y_3 = 0.7$ m. The upstream reservoir has a water surface elevation 3.5 m above the channel bottom. Assume that both the entrance minor loss coefficient and the loss coefficient to the gate equal 0.05. Determine the flow rate into the channel if the outflow q_o^* over the side weir equals 1 m²/s over its 50 m length.



Solution

This problem can be solved using the above program. The input data for such a solution consists of

3 .00001 .05 .013 .001 5 1.5 950 .013 .001 4 1. 500. 4. 50. .001 3.5 -1. 9.81 .05 .05 .7 0 90. 1

SOLSID.OUT

```
70. 3.25 3.4
50.
```

and the solution provided by the program consists of

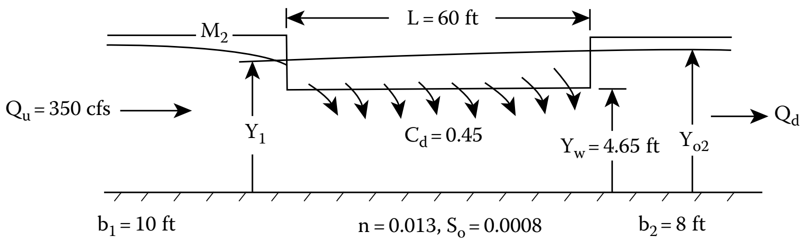
$Q = 74.28, Y_1 = 3.20, Y_2 = 4.71$, QE = 24.28.	
1500.0000	4.7060
1450.0000	4.6564
1400.0000	4.6068
1350.0000	4.5572
1300.0000	4.5076
1250.0000	4.4580
1200.0000	4.4085
1150.0000	4.3590
1100.0000	4.3095
1050.0000	4.2601
1000.0000	4.2097
995.0000	4.1957
990.0000	4.1810
985.0000	4.1657
980.0000	4.1499
975.0000	4.1338
970.0000	4.1173
965.0000	4.1006
960.0000	4.0837
955.0000	4.0666
950.0000	4.0497
900.0000	4.0024
850.0000	3.9553
800.0000	3.9084
750.0000	3.8616
700.0000	3.8151
650.0000	3.7688
600.0000	3.7228
550.0000	3.6770
500.0000	3.6315

(continued)

450.0000	3.5863
400.0000	3.5414
350.0000	3.4969
300.0000	3.4528
250.0000	3.4091
200.0000	3.3659
150.0000	3.3232
100.0000	3.2811
50.0000	3.2395
0.0000	3.1986

EXAMPLE PROBLEM 4.50

A rectangular channel contains a 60 ft long side weir whose height is w = 4.65 ft above the channel bottom which has n = 0.013 and $S_o = 0.0008$. At the beginning of the weir, the width of the channel is $b_1 = 10$ ft, and at its end the width is $b_2 = 8$ ft. If the flow rate upstream of the side weir is $Q_u = 350$ cfs, determine the flow rate in the downstream channel Q_d, the depth at the beginning of the side weir Y_1, and the depth at the end of the side weir Y_{o2} if the channel is very long. The discharge coefficient for the side weir is $C_d = 0.45$.



Solution

This problem is governed by the following three equations:

$$F_1 = nP_d^{2/3}Q_d - C_u A_d^{5/3}(S_{o2})^{1/2} = 0 \quad \text{(Manning's equation)}$$

$$F_2 = Y_1 - Y_{GVF}(Y_{o2}) = 0 \quad \text{(GVF starting at the end of the side weir)}$$

$$F_3 = Q_u - Q_d - q_o^* dx = 0 \quad \text{(Continuity including acc. outflow)}$$

The program listing is designed to solve problems of this type. It is designed to accommodate trapezoidal channels as well as rectangular channels.

Listing of Program SOLWEIM.FOR (outflow-mild channel)

```
C Solves problem of lateral outflow from a side weir in a mild
C channel. The flow is assumed to be subcritical at beginning of
C side weir and downstream channel is mild and contains normal
C depth. Unknows: Qd (downstream flow rate), Y1 (depth at beginning
C of side weir), & Yod (normal depth downstream channel.
      LOGICAL DONE
      REAL F(3),D(3,3),X(3),FFF(3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
```

```
      COMMON /TRAS/B1,FM1,B2,FM2,G,G2,Qu,FL,TOL,FN,SO,CC,FMSQ,CS,
     &CDG,W,SQSTAR,QSTAR1,DB,DM,XX,X,FFF,IPRNT,DONE,IOUT
      EQUIVALENCE (Qd,X(1)),(Y1,X(2)),(Yo2,X(3))
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,Qu,FN,SO,B1,FM1,','B2,FM2,L,
     &W,Cd,g,PRNT'
      READ(*,*) IOUT,TOL,ERR,Qu,FN,SO,B1,FM1,B2,FM2,FL,W,Cd,G,IPRNT
      DONE=.FALSE.
      FMSQ=2.*SQRT(FM2*FM2+1.)
      DB=(B1-B2)/FL
      DM=(FM1-FM2)/FL
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      CS=CC*SQRT(SO)
      G2=2.*G
      CDG=Cd*SQRT(G2)
      WRITE(*,*)' GIVE guess for: Qd,Y1,Yo2'
      READ(*,*) X
      NCT=0
1     CALL FUN
      DO 8 I=1,3
8     F(I)=FFF(I)
      DO 10 J=1,3
      DX=.005*X(J)
      X(J)=X(J)+DX
      CALL FUN
      DO 9 I=1,3
9     D(I,J)=(FFF(I)-F(I))/DX
10    X(J)=X(J)-DX
      FAC=D(3,1)/D(1,1)
      D(3,2)=D(3,2)-FAC*D(1,2)
      D(3,3)=D(3,3)-FAC*D(1,3)
      F(3)=F(3)-FAC*F(1)
      FAC=D(2,1)/D(1,1)
      D(2,2)=D(2,2)-FAC*D(1,2)
      D(2,3)=D(2,3)-FAC*D(1,3)
      F(2)=F(2)-FAC*F(1)
      FAC=D(3,2)/D(2,2)
      D(3,3)=D(3,3)-FAC*D(2,3)
      F(3)=F(3)-FAC*F(2)
      DIF1=F(3)/D(3,3)
      Yo2=Yo2-DIF1
      DIF=(F(2)-DIF1*D(2,3))/D(2,2)
      Y1=Y1-DIF
      SUM=ABS(DIF1)+ABS(DIF)
      DIF=(F(1)-D(1,2)*DIF-D(1,3)*DIF1)/D(1,1)
      SUM=SUM+ABS(DIF)
      Qd=Qd-DIF
      NCT=NCT+1
      WRITE(*,110) NCT,SUM,X
110   FORMAT(' NCT=',I3,' SUM=',E12.5,3F10.3)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 1
      DONE=.TRUE.
      CALL FUN
```

```
      WRITE(IOUT,100) X
100   FORMAT(' Qd =',F10.2,' Y1 =',F10.3,' Yo2 =',F10.3)
      END
      SUBROUTINE FUN
      LOGICAL DONE
      EXTERNAL DYX
      REAL X(3),WW(1,13),Y(1),DY(1),XP(1),YP(1,1),F(3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,FM1,B2,FM2,G,G2,Qu,FL,TOL,FN,SO,CC,FMSQ,CS,
     &CDG,W,SQSTAR,QSTAR1,DB,DM,XX,X,F,IPRNT,DONE,IOUT
      H1=-.05
      HMIN=.001
      F(1)=FN*X(1)*(B2+FMSQ*X(3))**.666666667-CS*((B2+FM2*X(3))
     &*X(3))**1.6666667
      Y(1)=X(3)
      XX=FL
      IF(DONE) THEN
      QSTARA=0.
      NPRT=0
      XX1=XX
      WRITE(IOUT,100)
100   FORMAT('   x    Y  acc. q* av. q*','  X1  X2',/,1X,57('-'))
      ENDIF
      SQSTAR=0.
      IF(Y(1).LT.W) THEN
      QSTAR1=0.
      ELSE
      QSTAR1=CDG*(Y(1)-W)**1.5
      ENDIF
4     XZ=XX-1.
      IF(XZ.LT.0.) XZ=0.
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(Y(1).LT.W) THEN
      QSTAR2=0.
      ELSE
      QSTAR2=CDG*(Y(1)-W)**1.5
      ENDIF
      SQSTAR=SQSTAR+(XX-XZ)*(QSTAR1+QSTAR2)/2.
      QSTAR1=QSTAR2
      IF(DONE) THEN
      QSTARA=QSTARA+(QSTAR1+QSTAR2)/2.
      NPRT=NPRT+1
      IF(MOD(NPRT,IPRNT).EQ.0 .OR. XZ.LT. .001) THEN
      WRITE(IOUT,110) XZ,Y(1),SQSTAR,QSTARA/FLOAT(IPRNT),XX1,XZ
110   FORMAT(F8.2,F8.3,2F8.3,' betw.',F8.2,'',F8.2)
      QSTARA=0.
      XX1=XZ
      ENDIF
      ENDIF
      XX=XZ
      IF(XZ.GT. .001) GO TO 4
      F(2)=X(2)-Y(1)
      F(3)=Qu-X(1)-SQSTAR
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
```

```
      REAL Y(1),DY(1),X(3),F(3)
      LOGICAL DONE
      COMMON /TRAS/B1,FM1,B2,FM2,G,G2,Qu,FL,TOL,FN,SO,CC,FMSQ,CS,
     &CDG,W,SQSTAR,QSTAR1,DB,DM,XXB,X,F,IPRNT,DONE,IOUT
      YY=Y(1)
      IF(YY.LT. .01) YY=.01
      B=B1+DB*XX
      FM=FM1+DM*XX
      P=B+2.*SQRT(FM*FM+1.)*YY
      A=(B+FM*YY)*YY
      IF(YY.LT.W) THEN
      QSTAR=0.
      ELSE
      QSTAR=CDG*(YY-W)**1.5
      ENDIF
      QQ=X(1)+SQSTAR+ABS(XXB-XX)*(QSTAR+QSTAR1)/2.
      SF=(FN*QQ/CC*(P/A)**.666666667/A)**2
      T=B+2.*FM*YY
      A3=A**3
      QQS=QQ*QQ
      DY(1)=(SO-SF+QQ*QSTAR/(G*A*A)+QQS/(G*A3)*(DB+YY*DM)*YY)/
     &(1.-QQS*T/G/A3)
      RETURN
      END
```

Input data to solve this problem.

```
3 .0000001 .001 350 .013 .0008 10 0 8 0 60 4.65
.45 32.2 3
```

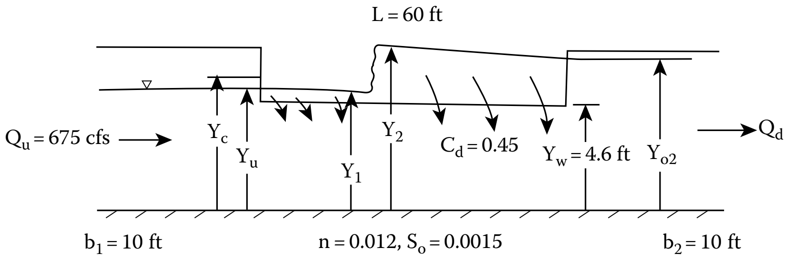
Solution to this problem.

$Q_d = 249.34$, $Y_1 = 4.938$, $Y_{o2} = 5.506$.

x	Y	acc. q*	av. Q*	X_1	X_2
57.00	5.483	8.409	2.784	Betw.	60.00 and 57.00
54.00	5.460	16.474	2.669	Betw.	57.00 and 54.00
51.00	5.435	24.190	2.552	Betw.	54.00 and 51.00
48.00	5.410	31.551	2.434	Betw.	51.00 and 48.00
45.00	5.384	38.551	2.313	Betw.	48.00 and 45.00
42.00	5.358	45.188	2.192	Betw.	45.00 and 42.00
39.00	5.331	51.458	2.070	Betw.	42.00 and 39.00
36.00	5.303	57.361	1.947	Betw.	39.00 and 36.00
33.00	5.275	62.897	1.825	Betw.	36.00 and 33.00
30.00	5.246	68.066	1.703	Betw.	33.00 and 30.00
27.00	5.217	72.870	1.581	Betw.	30.00 and 27.00
24.00	5.187	77.314	1.461	Betw.	27.00 and 24.00
21.00	5.157	81.402	1.343	Betw.	24.00 and 21.00
18.00	5.127	85.139	1.226	Betw.	21.00 and 18.00
15.00	5.096	88.532	1.112	Betw.	18.00 and 15.00
12.00	5.065	91.590	1.001	Betw.	15.00 and 12.00
9.00	5.033	94.321	0.892	Betw.	12.00 and 9.00
6.00	5.002	96.735	0.787	Betw.	9.00 and 6.00
3.00	4.970	98.843	0.686	Betw.	6.00 and 3.00
0.00	4.938	100.658	0.589	Betw.	6.00 and 0.00

EXAMPLE PROBLEM 4.51

A flow rate of $Q_u = 675$ cfs occurs in a 10 ft wide rectangular channel upstream from a 60 ft long side weir with a discharge coefficient of $C_d = 0.45$. The channel retains the same width throughout the side weir, and has a Manning's roughness coefficient n = 0.012, and a bottom slope $S_o = 0.0015$. The weir's crest is 4.6 ft above the channel bottom. Assume that the critical depth, or slightly below Y_c, occurs at the beginning of the side weir, and determine the following: (1) the discharge in the downstream channel Q_d, (2) the depth upstream from the hydraulic jump Y_1, (3) the depth downstream from the hydraulic jump Y_2, (4) the normal depth in the downstream channel Y_{o2} (assume the channel is very long and its flow is not effected by any downstream control), and (5) the position where the hydraulic jump occurs, x.



Solution

The five equations needed for these five unknowns consist of

$$F_1 = nP_d^{2/3}Q_d - C_uA_d^{5/3}(S_{o2})^{1/2} = 0 \quad \text{(Manning's equation)}$$

$$F_2 = Y_1 - Y_{GVF}(Y_c) = 0 \quad \text{(GVF starting at the beginning of the side weir)}$$

$$F_3 = Y_2 - Y_{GVF}(Y_{o2}) = 0 \quad \text{(GVF starting at the end of the side weir)}$$

$$F_4 = hc_1A_1 + Q^2/(gA_1) - hc_2A_2 - Q^2/(gA_2) = 0 \quad \text{(Momentum equation)}$$

$$F_5 = Q_u - Q_d - \int q_o^* dx - \int q_o^* dx = 0 \quad \text{(Continuity including acc. outflow)}$$

to jump after jump

The program SOLWEIS is designed to solve problems of this type. It is designed to accommodate trapezoidal channels, as well as rectangular channels and allows for the channel to contract or enlarge across the side weir length.

Listing of SOLWEIS.FOR to solve outflow with jump in side weir length—Y1 < Yc

```
C Solves problem of lateral outflow from a side weir in a mild
C channel. The flow is assumed to be subcritical at beginning of
C side weir and downstream channel is mild and contains normal
C depth. Unknows: Qd (downstream flow rate), Y1 (depth at beginning
C of side weir), & Yod (normal depth downstream channel. X(1) = Qd;
C X(2) = Y1 (upstream jump); X(3) = Y2 (downstream jump), X(4) =Yo2
C (depth downstream channel); X(5) = x (postion of jump)
      LOGICAL DONE,UPST
      INTEGER*2 INDX(5)
      REAL F(5),D(5,5),X(5),FFF(5)
```

```
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,FM1,B2,FM2,G,G2,Qu,Y1,FL,TOL,FN,SO,CC,FMSQ,
     &CS,CDG,W,SQSTAR,QSTAR1,DB,DM,XX,X,FFF,IPRNT,DONE,UPST,IOUT
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,Qu,Y1,FN,SO,B1,FM1,B2,FM2,L,W,
     &Cd,g,PRNT,PRNTM'
      READ(*,*) IOUT,TOL,ERR,Qu,Y1,FN,SO,B1,FM1,B2,FM2,FL,W,Cd,G,
     &IPRNT,IPRMOR
      DONE=.FALSE.
      FMSQ=2.*SQRT(FM2*FM2+1.)
      DB=(B1-B2)/FL
      DM=(FM1-FM2)/FL
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      CS=CC*SQRT(SO)
      G2=2.*G
      CDG=Cd*SQRT(G2)
      WRITE(*,*)' GIVE guess for: Qd,Y1,Y2,Yo2,x'
      READ(*,*) X
      NCT=0
      IF(IPRMOR.GT.1) DONE=.TRUE.
1     CALL FUN
      DO 8 I=1,5
8     F(I)=FFF(I)
      DO 10 J=1,5
      DX=.005*X(J)
      X(J)=X(J)+DX
      CALL FUN
      DO 9 I=1,5
9     D(I,J)=(FFF(I)-F(I))/DX
10    X(J)=X(J)-DX
      IF(IPRMOR.GT.0) THEN
      DO 11 I=1,5
11    WRITE(*,333)(D(I,J),J=1,5),F(I)
333   FORMAT(6F10.3)
      ENDIF
      CALL SOLVEQ(5,1,5,D,F,1,DD,INDX)
      IF(IPRMOR.GT.0) WRITE(*,334) (F(I),I=1,5)
334   FORMAT(' sol',6F10.3)
      NCT=NCT+1
      DIF=0.
      DO 20 I=1,5
      X(I)=X(I)-F(I)
20    DIF=DIF+ABS(F(I))
      WRITE(*,110) NCT,DIF,X
110   FORMAT(' NCT=',I3,' DIF=',E12.5,/,5F10.3)
      IF(NCT.LT.30.AND.DIF.GT.ERR)GOTO 1
      DONE=.TRUE.
```

```
      CALL FUN
      WRITE(IOUT,100) X
100   FORMAT(' Qd =',F10.2,' Y1 =',F10.3,' Y2 =', F10.3,' Yo2 =',
     &F10.3,' x =',F10.2)
      END
      SUBROUTINE FUN
      LOGICAL DONE,UPST
      EXTERNAL DYX
      REAL X(5),WW(1,13),Y(1),DY(1),XP(1),YP(1,1),F(5)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON/TRAS/B1,FM1,B2,FM2,G,G2,Qu,Y1,FL,TOL,FN,SO,CC,FMSQ,CS,
     &CDG,W,SQSTAR,QSTAR1,DB,DM,XX,X,F,IPRNT,DONE,UPST,IOUT
      F(1)=FN*X(1)*(B2+FMSQ*X(3))**.666666667-CS*((B2+FM2*X(3))*
     &X(3))**1.6666667
      IF(X(5).LT.0.) THEN
      SQSTAU=0.
      GO TO 3
      ENDIF
      H1=.05
      HMIN=.00001
      Y(1)=Y1
      XX=0.
      UPST=.FALSE.
      IF(DONE) THEN
      QSTARA=0.
      NPRT=0
      XX1=XX
      WRITE(IOUT,100)
100   FORMAT('   x    Y  acc. q* av.''  q*     X1    X2    Q',/,
     &1X,67('-'))
      ENDIF
      SQSTAR=0.
      IF(Y(1).LT.W) THEN
      QSTAR1=0.
      ELSE
      QSTAR1=CDG*(Y(1)-W)**1.5
      ENDIF
2     XZ=XX+1.
      IF(XZ.GT.X(5)) XZ=X(5)
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(Y(1).LT.W) THEN
      QSTAR2=0.
      ELSE
      QSTAR2=CDG*(Y(1)-W)**1.5
      ENDIF
      SQSTAR=SQSTAR+(XZ-XX)*(QSTAR1+QSTAR2)/2.
      QSTAR1=QSTAR2
      IF(DONE) THEN
      QSTARA=QSTARA+(QSTAR1+QSTAR2)/2.
```

```
      NPRT=NPRT+1
      IF(MOD(NPRT,IPRNT).EQ.0.OR.ABS(XZ-X(5)).LT. .001) THEN
      QQ=Qu-SQSTAR
      WRITE(IOUT,110) XZ,Y(1),SQSTAR,QSTARA/FLOAT(IPRNT),XX1,XZ,QQ
110   FORMAT(F8.2,F8.3,2F8.3,' betw.',F8.2,'',F8.2,F10.2)
      QSTARA=0.
      XX1=XZ
      ENDIF
      ENDIF
      XX=XZ
      IF(XZ.LT. X(5)-.001) GO TO 2
      SQSTAU=SQSTAR
      F(2)=X(2)-Y(1)
3     H1=-.05
      HMIN=.00001
      Y(1)=X(4)
      IF(X(5).GT.FL) GO TO 8
      XX=FL
      UPST=.TRUE.
      IF(DONE) THEN
      QSTARA=0.
      NPRT=0
      XX1=XX
      WRITE(IOUT,100)
      ENDIF
      SQSTAR=0.
      IF(Y(1).LT.W) THEN
      QSTAR1=0.
      ELSE
      QSTAR1=CDG*(Y(1)-W)**1.5
      ENDIF
4     XZ=XX-1.
      IF(XZ.LT.X(5)) XZ=X(5)
      IF(XZ.LT.0.) XZ=0.
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(Y(1).LT.W) THEN
      QSTAR2=0.
      ELSE
      QSTAR2=CDG*(Y(1)-W)**1.5
      ENDIF
      SQSTAR=SQSTAR+(XX-XZ)*(QSTAR1+QSTAR2)/2.
      QSTAR1=QSTAR2
      IF(DONE) THEN
      QSTARA=QSTARA+(QSTAR1+QSTAR2)/2.
      NPRT=NPRT+1
      IF(MOD(NPRT,IPRNT).EQ.0.OR.ABS(XZ-X(5)).LT. .001) THEN
      QQ=X(1)+SQSTAR
      WRITE(IOUT,110) XZ,Y(1),SQSTAR,QSTARA/FLOAT(IPRNT),XX1,XZ,QQ
      QSTARA=0.
```

```
      XX1=XZ
      ENDIF
      ENDIF
      XX=XZ
      IF(XZ.GT. X(5)+.001 .AND. XZ.GT. 0.) GO TO 4
      SQSTAD=SQSTAR
8     F(3)=X(3)-Y(1)
      B=B1+DB*X(5)
      FM=FM1+DM*X(5)
      FMM1=(B/2.+X(2)*FM/3.)*X(2)**2+(Qu-SQSTAU)**2/G/
     &((B+X(2)*FM)*X(2))
      FMM2=(B/2.+X(3)*FM/3.)*X(3)**2+(Qu-SQSTAU)**2/G/
     &((B+X(3)*FM)*X(3))
      F(4)=FMM1-FMM2
      F(5)=Qu-X(1)-SQSTAU-SQSTAD
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1),X(5),F(5)
      LOGICAL DONE,UPST
      COMMON /TRAS/B1,FM1,B2,FM2,G,G2,Qu,Y1,FL,TOL,FN,SO,CC,FMSQ,
     &CS,CDG,W,SQSTAR,QSTAR1,DB,DM,XXB,X,F,IPRNT,DONE,UPST,IOUT
      YY=Y(1)
      IF(YY.LT. .01) YY=.01
      B=B1+DB*XX
      FM=FM1+DM*XX
      P=B+2.*SQRT(FM*FM+1.)*YY
      A=(B+FM*YY)*YY
      IF(YY.LT.W) THEN
      QSTAR=0.
      ELSE
      QSTAR=CDG*(YY-W)**1.5
      ENDIF
      IF(UPST) THEN
      QQ=X(1)+SQSTAR+ABS(XXB-XX)*(QSTAR+QSTAR1)/2.
      ELSE
      QQ=Qu-SQSTAR-ABS(XX-XXB)*(QSTAR+QSTAR1)/2.
      ENDIF
      SF=(FN*QQ/CC*(P/A)**.666666667/A)**2
      T=B+2.*FM*YY
      A3=A**3
      QQS=QQ*QQ
      FR1=1.-QQS*T/G/A3
      IF(ABS(FR1) .LT. .01) THEN
      DY(1)=0.
      ELSE
      DY(1)=(SO-SF+QQ*QSTAR/(G*A*A)+QQS/(G*A3)*(DB+YY*DM)*YY)/FR1
      ENDIF
      RETURN
      END
```

The critical depth $Y_c = 5.21$ ft is associated with Q = 675 cfs. Use 5.1 as the starting depth. Input data to solve this problem.

```
3 .000001 .001 675 5.1 .012 .0015 10 0 10 0 60
4.6 .45 32.2 3 0
450 4.8 5.5 7 50
```

Solution to this problem

$Q_d = 599.99$, $Y_1 = 4.826$, $Y_2 = 5.448$, $Y_{o2} = 6.312$, x = 43.6.

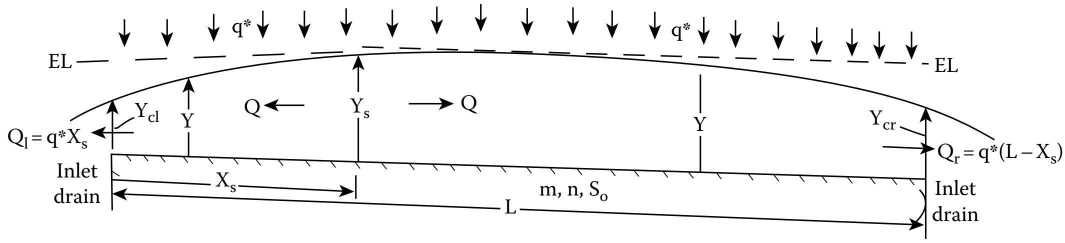
x	Y	acc. q*	av. q*		X_1 X_2	Q
3.00	4.988	2.072	0.646	Betw.	0.00 and 3.00	672.93
6.00	4.939	3.643	0.506	Betw.	3.00 and 6.00	671.36
9.00	4.909	4.972	0.433	Betw.	6.00 and 9.00	670.03
12.00	4.889	6.150	0.386	Betw.	9.00 and 12.00	668.85
15.00	4.874	7.225	0.354	Betw.	12.00 and 15.00	667.77
18.00	4.863	8.227	0.330	Betw.	15.00 and 18.00	666.77
21.00	4.854	9.174	0.313	Betw.	18.00 and 21.00	665.83
24.00	4.847	10.079	0.300	Betw.	21.00 and 24.00	664.92
27.00	4.842	10.952	0.289	Betw.	24.00 and 27.00	664.05
30.00	4.838	11.800	0.281	Betw.	27.00 and 30.00	663.20
33.00	4.834	12.627	0.275	Betw.	30.00 and 33.00	662.37
36.00	4.831	13.439	0.270	Betw.	33.00 and 36.00	661.56
39.00	4.829	14.237	0.265	Betw.	36.00 and 39.00	660.76
42.00	4.827	15.024	0.262	Betw.	39.00 and 42.00	659.98
43.69	4.826	15.464	0.173	Betw.	42.00 and 43.69	659.54
57.00	6.173	15.219	4.966	Betw.	60.00 and 57.00	615.21
54.00	6.029	28.511	4.324	Betw.	57.00 and 54.00	628.51
51.00	5.877	39.886	3.685	Betw.	54.00 and 51.00	639.88
48.00	5.715	49.348	3.048	Betw.	51.00 and 48.00	649.34
45.00	5.537	56.881	2.402	Betw.	48.00 and 45.00	656.88
43.69	5.448	59.542	1.278	Betw.	45.00 and 43.69	659.54

4.20 GUTTER FLOW AND OUTFLOW THROUGH GRATES

Two additional applications, that are often connected, are the open channel flows resulting from the accumulations of the lateral inflow into gutters from rainfall on roadways, and the outflow from the gutters through grates (racks) into storm drains. Under some circumstances, these two applications must be handled as a single problem because each affects the equations of the other. Many applications allow the gutter flow to be solved first, and thereafter the flow through the grates can be solved as a separate problem. Therefore, we will start by considering them as separate problems. Later, situations will be handled in which the two processes interact so that they must be solved together. Flow through grates in the bottom of channels is not limited to taking gutter flow into storm drains, as evidenced by previous applications.

4.20.1 Gutter Flow

Assume that a length L of gutter with a mild bottom slope S_o is supplied by a lateral inflow q* over its entire length, and that at both ends of this length of gutter there are drains with sufficient capacity to accept all of the flow. Under these assumptions, the depths will be critical at both ends of the gutter, and the flow in between will be a subcritical lateral inflow, as shown on the sketch below.



The position X_s that separates the flow moving in the left direction versus that moving in the right direction is unknown. Also, the depth Y_s at this position, as well as the two critical depths at the two ends of the gutter, are unknown. Thus, there are four unknowns. We will assume that the lateral inflow q* is known, and for now let us take it as constant, and that the gutter's bottom slope S_o, its Manning's n, and geometric properties are also known. Therefore, four equations are needed to solve the four unknown variables: Y_{cl}, Y_{cr}, Y_s, and X_s. The flow rates at the left and the right ends of the gutter are given by, $Q_l = X_s q^*$ and $Q_r = (L - X_s)q^*$, in which the flow rate to the left Q_l moves up against the adverse bottom slope S_o. There are two means of handling this flow; one is to use a positive x direction in the direction of the flow, and then S_o is negative. The other is to use x as positive in the direction of the gutter slope, i.e., from left to right, and to consider the flow in the left portion of the gutter negative, i.e., Q, is negative. We will use the latter approach. The four equations needed to be solved for the four unknown variables are two critical flow equations, and two ODEs, as given below.

$$F_1 = [X_s q^*]^2 T_l - gA_l^3 = 0 \tag{4.21a}$$

$$F_2 = [(L - X_s)q^*]^2 T_r - gA_r^3 = 0 \tag{4.21b}$$

$$F_3 = Y_s - Y_{sode}(Y_{c1}) = 0 \quad (\text{with } x = 0 \text{ to } x = X_s) \tag{4.21c}$$

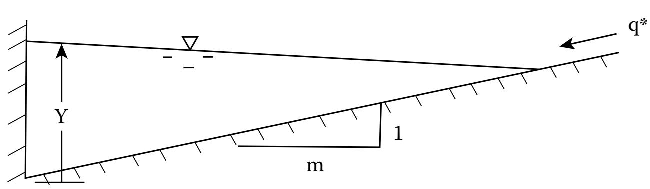
$$F_4 = Y_s - Y_{sode}(Y_{cr}) = 0 \quad (\text{with } x = L \text{ to } x = X_c) \tag{4.21d}$$

The two ODE equations will get the Y_{sode} from the solution of

$$dY/dx = \{(S_o - S_f - 2Qq^*/(gA^2))\}/\left(1 - F_r^2\right) \tag{4.22}$$

in which the solution for F_3 will be from x = 0 to x = X_s, and the solution for F_4 will be from x = L to x = X_s. Also, the solution on the left side of X_s (e.g., F_3) will take Q as negative as mentioned above. When doing this, the bottom slope S_o will be positive in the ODE; but rather than computing S_f as $S_f = [nQ(P/A)^{2/3}/(C_uA)]^2$, it will be computed as, $S_f = |nQ(P/A)^{2/3}/(C_uA)|[nQ(P/A)^{2/3}/(C_uA)]$; thus giving a negative value to S_f when Q is negative. Also, note that since Q is negative on the left side of X_s, the term $2Qq^*/(gA^2)$ in the numerator of the ODE adds to its positive amount causing the depth Y to increase in the positive x direction. When solving the ODE on the right of X_s, both S_f and $2Qq^*/(gA^2)$ in the numerator of the ODE are positive and the negative sign in front of these terms will cause the depth Y to decrease in the positive x direction.

Often, gutters have a cross section that consists of one-half of a triangle as shown in the sketch.



For such triangular gutters, the area, the perimeter, and the top width are given by $A = 0.5mY^2$, $P = Y + Y(1 + m^2)^{1/2} = Y\{1 + (1 + m^2)^{1/2}\}$, and $T = mY$, respectively. (Notice these are different from the special trapezoidal section with b = 0.) For triangular gutters, the first two of the above four equations become (the critical flow equations):

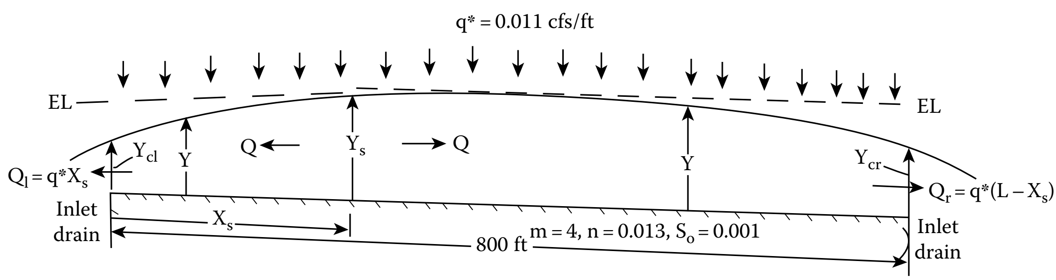
$$F_l = mY_{cl}Q_l^2 - 0.125gm^3Y_{cl}^6 = 0 \quad \text{or} \quad Y_{cl} = [8Q_l^2/(gm^2)]^2 \tag{4.23a}$$

$$F_2 = mY_{cr}Q_r^2 - 0.125gm^3Y_{cr}^6 = 0 \quad \text{or} \quad Y_{cr} = [8Q_r^2/(gm^2)]^2 \tag{4.23b}$$

and if one were using the Newton method ($Y_c^{(m+1)} = Y_c^{(m)} - F/dF/dY$) to solve either of these single equations for the critical depth, the correction F/dF/dY is given by $F/dF/dY = Y\{Q^2 - 0.125gm^2Y^5\}/\{Q^2 - 0.75gm^2Y^5\}$.

EXAMPLE PROBLEM 4.52

A triangular gutter with a side slope m = 4, n = 0.013, and a bottom slope $S_o = 0.001$ is L = 800 ft long between bottom drains. Assume that the drains readily accept the flow coming into them, and solve flow rates Q_l and Q_r entering the left and right drains, respectively, as well as the position and the depth X_s and Y_s where the flow separates if the lateral inflow is constant and equal to q* = 0.011 cfs/ft of length.



Solution

The program GUTTER, whose listing is given below, is designed to solve this problem. In studying the listing of this program you will note that it implements the Newton method in solving the above four equations. The unknowns are contained in the array X according to Y_{cl} = X(1), Y_{cr} = X(2), Y_s = X(3), and X_s = X(4). You will note that the program asks the user to supply a guess to only one of these unknowns, X_s. It then notes that $Q_l = X_sq^*$ and $Q_r = (L - X_s)q^*$, and solves the critical flow equations for guesses for Y_{cl} and Y_{cr}, and then adds these two values to get a guess for Y_s.

Program GUTTER.FOR (also see diskette for GUTTER.C)

```
C Solves the problem of side flow into a triangular gutter.
      INTEGER*2 INDX(4)
      LOGICAL DONE
      REAL F(4),D(4,4),FFF(4)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(4),SO,FM,FMS,QS,CC,FN,HM,G,G2,G5,FL,DX,GM2,TOL,
     &DONE,IOUT
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,n,So,m,L,g,DX,q*'
      READ(*,*) IOUT,TOL,ERR,FN,SO,FM,FL,G,DX,QS
      DONE=.FALSE.
      HM=.5*FM
```

```
      FMS=1.+SQRT(FM*FM+1.)
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      G5=.125*G
      GM2=G5*FM*FM
      WRITE(*,*)' GIVE guess for Position Xs'
      READ(*,*) X(4)
      X(1)=((QS*X(4))**2/GM2)**.2
      X(2)=((QS*(FL-X(4)))**2/GM2)**.2
      X(3)=X(1)+X(2)
      WRITE(*,*)' Guesses for x',X
      GM2=FM*GM2
      NCT=0
8     CALL FUN(F)
      WRITE(*,*) NCT,F
      DO 10 J=1,4
      DXX=.005*X(J)
      X(J)=X(J)+DXX
      CALL FUN(FFF)
      DO 9 I=1,4
9     D(I,J)=(FFF(I)-F(I))/DXX
10    X(J)=X(J)-DXX
      CALL SOLVEQ(4,1,4,D,F,1,DD,INDX)
      NCT=NCT+1
      SUM=0.
      DO 20 I=1,4
      X(I)=X(I)-F(I)
20    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,X
110   FORMAT(' NCT=',I3,' SUM=',E12.6,/,4F10.3)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 8
      WRITE(3,100) X
100   FORMAT(' Yc1=',F8.3,', Yc2=',F8.3,', Depth at Q=0',F8.3,'Xs=', F8.1)
      DONE=.TRUE.
      CALL FUN(F)
      END
      SUBROUTINE FUN(F)
      LOGICAL DONE
      EXTERNAL DYX
      REAL WW(1,13),Y(1),DY(1),XP(1),YP(1,1),F(4)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(4),SO,FM,FMS,QS,CC,FN,HM,G,G2,G5,FL,DX,GM2,TOL,
     &DONE,IOUT
      F(1)=FM*X(1)*(QS*X(4))**2-GM2*X(1)**6
      F(2)=FM*X(2)*(QS*(FL-X(4)))**2-GM2*X(2)**6
      HMIN=.001
      IF(DONE) THEN
      A=HM*X(1)**2
      Q=QS*X(4)
      WRITE(IOUT,100) 0.,X(1),Q,X(1)+(Q/A)**2/G2,A,Q/(FM*SQRT(G5*
     &X(1)**5))
```

```
      ENDIF
100   FORMAT('  x  Y  Q  E  A  Fr',/,1X,57('-'),/,F10.1,5F10.3)
      H1=.1
      XX=0.
      Y(1)=1.1*X(1)
10    XZ=XX+DX
      IF(XZ.GT.X(4)) XZ=X(4)
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS*(X(4)-XZ)
      WRITE(IOUT,110) XZ,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT(G5*
     &Y(1)**5))
110   FORMAT(F10.1,5F10.3)
      ENDIF
      XX=XZ
      IF(XZ.LT. X(4)) GO TO 10
      F(3)=X(3)-Y(1)
      H1=-.1
      IF(DONE) THEN
      A=HM*X(2)**2
      Q=QS*(FL-X(4))
      WRITE(IOUT,100) FL,X(2),Q,X(2)+(Q/A)**2/G2,A,Q/(FM* SQRT
     &(G5*X(2)**5))
      ENDIF
      XX=FL
      Y(1)=1.1*X(2)
20    XZ=XX-DX
      IF(XZ.LT.X(4)) XZ=X(4)
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS*(XZ-X(4))
      WRITE(IOUT,110) XZ,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM* SQRT
     &(G5*Y(1)**5))
      ENDIF
      XX=XZ
      IF(XZ.GT. X(4)) GO TO 20
      F(4)=X(3)-Y(1)
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      LOGICAL DONE
      REAL Y(1),DY(1)
      COMMON /TRAS/X(4),SO,FM,FMS,QS,CC,FN,HM,G,G2,G5,FL,DX,GM2,TOL,
     &DONE,IOUT
      A=HM*Y(1)**2
      Q=QS*(XX-X(4))
      SF=FN*Q/CC*(FMS*Y(1)/A)**.66666667/A
      SF=SF*ABS(SF)
      FR2=Q*Q*FM*Y(1)/(G*A**3)
      DY(1)=(SO-SF-2.*Q*QS/(G*A**2))/(1.-FR2)
      RETURN
      END
```

Program GUTTER.C

```
// Solves the problem of side flow into a triangular gutter
#include <stdlib.h>
#include <math.h>
#include <stdio.h>
float x[4],so,fm,fms,qs,cc,fn,hm,g,g2,g8,fl,dx,gm2,tol;
int done;
FILE *filo;
extern void solveq(int n,float **a,float *b,int itype,\
   float *dd,int *indx);
extern void rukust(int ne,float *dxs,float xbeg,float xend,\
   float err,float *y,float *ytt);
void slope(float xx,float *yp,float *dy){
   float a,q,sf,fr2,yy;
   yy=*yp; a=hm*yy*yy; q=qs*(xx-x[3]);
   sf=fn*q/cc*pow(fms*yy/a,.66666667)/a;
   sf*=fabs(sf);fr2=q*q*fm*yy/(g*pow(a,3.));
   *dy=(so-sf-2.*q*qs/(g*a*a))/(1.-fr2);
   return; } // End slope
void fun(float *f){float y[1],ytt[1],dxs[1],a,q,xx,xz; int i;
   f[0]=fm*x[0]*pow(qs*x[3],2.)-gm2*pow(x[0],6.);
   f[1]=fm*x[1]*pow(qs*(fl-x[3]),2.)-gm2*pow(x[1],6.);
   if(done){a=hm*x[0]*x[0];q=qs*x[3];
   fprintf(filo,"  x     Y     Q     E     \
      A     Fr\n");
   for(i=0;i<57;i++) fprintf(filo,"-");fprintf(filo,"\n");
   fprintf(filo,"%10.1f %9.3f %9.3f %9.3f %9.3f %9.3f\n",0.,\
     x[0],q,x[0]+pow(q/a,2.)/g2,a,q/(fm*sqrt(g8*pow(x[0],5.))));}
   dxs[0]=.1;xx=0.;y[0]=1.1*x[0];
L10: xz=xx+dx; if(xz>x[3]) xz=x[3];
   rukust(1,dxs,xx,xz,tol,y,ytt);
   if(done){a=hm*y[0]*y[0];q=qs*(x[3]-xz);
   fprintf(filo,"%10.1f %9.3f %9.3f %9.3f %9.3f %9.3f\n",xz,\
     y[0],q,y[0]+pow(q/a,2.)/g2,a,q/(fm*sqrt(g8*pow(y[0],5.))));}
   xx=xz; if(xz<(x[3]-1.e-6)) goto L10; f[2]=x[2]-y[0];
   dxs[0]=-.1;
   if(done){a=hm*x[1]*x[1]; q=qs*(fl-x[3]);
   fprintf(filo,"%10.1f %9.3f %9.3f %9.3f %9.3f %9.3f\n",fl,\
     x[1],q,x[1]+pow(q/a,2.)/g2,a,q/(fm*sqrt(g8*pow(x[1],5.))));}
   xx=fl; y[0]=1.1*x[1];
L20:xz=xx-dx;if(xz<x[3]) xz=x[3];
   rukust(1,dxs,xx,xz,tol,y,ytt);
   if(done){a=hm*y[0]*y[0];q=qs*(xz-x[3]);
   fprintf(filo,"%10.1f %9.3f %9.3f %9.3f %9.3f %9.3f\n",xz,\
     y[0],q,y[0]+pow(q/a,2.)/g2,a,q/(fm*sqrt(g8*pow(y[0],5.))));}
   xx=xz; if(xz>(x[3]+1.e-6)) goto L20; f[3]=x[2]-y[0];
   return; } // End of fun
void main(void){
   char fname[20]; int i,j,nct,indx[4]; float err,sum,dxx,dd[1];
   float f[4],**d,fff[4];
   d=(float**)malloc(4*sizeof(float*));
   for(i=0;i<4;i++)d[i]=(float*)malloc(4*sizeof(float));
   printf("Give name of output file\n");scanf("%s",fname);
   if((filo=fopen(fname,"w"))==NULL){
      printf("Cannot open file");exit(0);}
   printf("GIVE:TOL,ERR,n,So,m,L,g,DX,q*\n");
   scanf("%f %f %f %f %f %f %f %f %f",\
```

```
      &tol,&err,&fn,&so,&fm,&fl,&g,&dx,&qs);
   done=0;hm=.5*fm;fms=1.+sqrt(fm*fm+1.);
   if(g>20.) cc=1.486; else cc=1.;
   g2=2.*g;g8=.125*g;gm2=g8*fm*fm;
   printf("GIVE guess for Position Xs\n"); scanf("%f",&x[3]);
   x[0]=pow(pow(qs*x[3],2.)/gm2,0.2);
   x[1]=pow(pow(qs*(fl-x[3]),2.)/gm2,0.2);
   x[2]=x[0]+x[1];
   printf("Guesses for x %f %f %f %f\n",x[0],x[1],x[2],x[3]);
   gm2*=fm; nct=0;
   do{ fun(f);
      printf("nct=%d, %f %f %f %f\n",nct,f[0],f[1],f[2],f[3]);
      for(j=0;j<4;j++){dxx=.005*x[j];x[j]+=dxx; fun(fff);
         for(i=0;i<4;i++)d[i][j]=(fff[i]-f[i])/dxx; x[j]-=dxx;}
      solveq(4,d,f,1,dd,indx); nct++;sum=0.;
      for(i=0;i<4;i++){x[i]-=f[i];sum+=fabs(f[i]);}
      printf("nct= %d, sum= %fn %10.3f %10.3f %10.3f %10.3f\n",\
         nct,sum,x[0],x[1],x[2],x[3]);
   }while((nct<30) && (sum>err));
   fprintf(filo," Yc1= %8.3f, Yc2= %8.3f, Depth at Q=0 %8.3f,\
      Xs= %8.1f\n",x[0],x[1],x[2],x[3]);
   done=1; fun(f); fclose(filo); exit(0);}
```

Input: 3 1.e–5 .0001 .013 .001 4 800 32.2 20 .011
200

Solution:

$Y_{o1} = 0.554$, $Y_{C2} = 0.945$, Depth at Q = 0.964, $X_S = 166.8$.

x	Y	Q	E	A	F_r
0.0	0.554	1.835	0.693	0.614	1.000
20.0	0.734	1.615	0.769	1.078	0.435
40.0	0.790	1.395	0.810	1.249	0.313
.	.	.	.	.	.
160.0	0.958	0.075	0.958	1.834	0.010
166.8	0.964	0.000	0.964	1.860	0.000
800.0	0.945	6.965	1.181	1.786	1.000
780.0	1.117	6.745	1.230	2.495	0.638
760.0	1.154	6.525	1.247	2.662	0.569
.	.	.	.	.	.
200.0	0.996	0.365	0.997	1.986	0.046
180.0	0.977	0.145	0.978	1.911	0.019
166.8	0.964	0.000	0.964	1.860	0.000

EXAMPLE PROBLEM 4.53

The gutter in the previous problem has a horizontal bottom rather than a slope $S_o = 0.001$. Now, solve from the flows, etc.

Solution

Now, the position that separates the flow toward the left and right sides will be in the middle of the gutter, L/2 = 400 ft, and it is possible to solve the two critical flow equations at the ends of the gutter separately as single equations, but since $Q_l = Q_r$, the critical depths $Y_{cl} = Y_{cr}$. Therefore, the depth Y_s at the middle of the gutter can be obtained by solving the ODE, either starting from the left or the right side, i.e., a system of simultaneous equations does not need to be solved. The program GUTTERHZ solves this problem.

Program GUTTERHZ.FOR

```
      REAL YY(1),YTT(1)
      COMMON FM,FMS,QS,C,FN,HM,G,QE,FL
      EQUIVALENCE (Y,YY(1))
      WRITE(6,*)'Give:IOUT,m,n,L,DX,g,Qstar'
      READ(5,*) IOUT,FM,FN,FL,DX,G,QS
      DXS=1.
      FL2=FL/2.
      IF(G.GT.30.) THEN
      C=1.486
      YC=3.
      ELSE
      C=1.
      YC=.82
      ENDIF
      DX=ABS(DX)
      N=FL2/DX+.5
      HM=.5*FM
      FMS=1.+SQRT(FM*FM+1.)
      QE=QS*FL2
      G8=.125*G
      G2=2.*G
      YC=((QE/FM)**2/G8)**.2
      Y=1.05*YC
      WRITE(IOUT,99) FM,FL,FN,QS
99    FORMAT(' m =',F8.3,' L =',F10.1,' n =',F8.4,' q* =',F10.6,/,
     &'    x     Y     Q    E    A    Fr')
      A=HM*YC*YC
      WRITE(IOUT,100)FL,YC,QE,YC+(QE/A)**2/G2,A,QE/(FM*SQRT
     &(G8*YC**5))
100   FORMAT(F10.1,5F10.3)
      X1=FL
      DO 10 I=1,N
      X=X1-DX
      CALL RUKUST(1,DXS,X1,X,1.E-4,YY,YTT)
      Q=QE-(FL-X)*QS
      A=HM*Y**2
      WRITE(IOUT,100)X,Y,Q,Y+(Q/A)**2/G2,A,Q/(FM*SQRT(G8*YC**5))
10    X1=X
      END
      SUBROUTINE SLOPE(X,Y,DYX)
      REAL Y(1),DYX(1)
      COMMON FM,FMS,QS,C,FN,HM,G,QE,FL
      A=HM*Y(1)**2
      Q=QE-(FL-X)*QS
      SF=(FN*Q/C*((FMS*Y(1))/A)**.66666667/A)**2
      DYX(1)=(SF+2.*Q*QS/(G*A**2))/(Q*Q*(FM*Y(1))/(G*A**3)-1.)
      RETURN
      END
```

Input: 3 4 .013 800 20 32.2 .011

Solution:

m = 4.000, L = 800.0, n = 0.0130, q* = 0.011000.

x	Y	Q	E	A	Fr
800.0	0.786	4.400	0.983	1.237	1.000
780.0	0.970	4.180	1.047	1.883	0.950
.	.	.	.	.	.
440.0	1.189	0.440	1.189	2.827	0.100
420.0	1.190	0.220	1.190	2.830	0.050
400.0	1.190	0.000	1.190	2.831	0.000

You might use program GUTTER to solve this same problem.

Since for a triangular gutter, an explicit equation gives the critical depth from a known flow rate, it is possible to eliminate Equations 4.21a and b from the system of simultaneous equations to solve the problem. Thus, program GUTTER could be modified to solve only two equations simultaneously rather than four. (See a homework problem.)

4.20.2 Flow into Grates at Bottom of Channel

The flow from the bottom of a gutter through grates is a problem in which the lateral outflow depends upon the depth of the flow at any position. The orifice formula is generally used to define this lateral outflow,

$$q_o^* = C_d\sqrt{2g}(A_o/L)Y^{0.5} = C_d\sqrt{2g}(fb)Y^{0.5} \tag{4.24}$$

in which A_o/L is the area of the opening per unit length, and the second form of the above equation is taken equal to the fraction f, of the bottom that is open times the width b, of the grate. Notice from this formula, that the lateral outflow is proportional to the square root of the depth at any position over the grate; whereas the lateral discharge from a side weir is proportional to the head over the crest of the weir raised to the 1.5 power. (Also the 2/3 is not present in this formula that occurs in the weir formula.) The discharge coefficient C_d will depend upon the type of grates. For bar type grates that run parallel to the direction of the flow in the gutter, the value of C_d will be larger than if these bars are normal to the direction of the flow. We will assume that the storm drain that receives the flow from the grates has the capacity to carry off the flow. Obviously, if this is not true, then the lateral outflow will involve the hydraulics of the storm drain system of pipes, etc. The ODE that describes the change in depth across a outflow grate is

$$\frac{dY}{dx} = \frac{S_o - S_f + Qq_o^*/(gA^2)}{1 - F_r^2} \tag{4.25}$$

in which the flow rate Q will need to be obtained by numerically integrating the lateral outflow over the length through which it flows, i.e.,

$$Q = Q_o - \int q_o^* dx \tag{4.26}$$

in which Q_o is the flow rate at the beginning of the grate. An easy, not very precise, but generally adequate means for carrying out the numerical integration is to use the trapezoidal rule, or

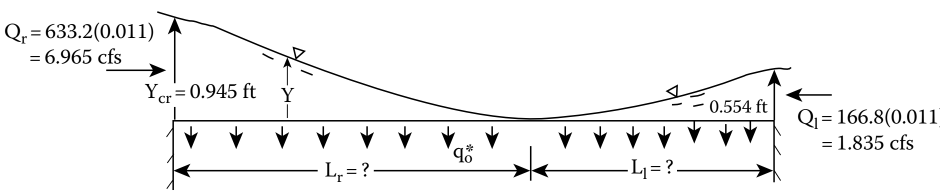
$$\int q_o^* dx = (x_{i+1} - x_i)[(q_o^*)_i + (q_o^*)_{i+1}]/2 = \Delta x (q_o^*)_{av} \tag{4.27}$$

in which the subscript i denotes the past position and the subscript i + 1 denotes the current position. In computing the slope of the energy line S_f, we will assume Manning's equation is valid and the triangular cross section will be used; but without experimental data, the value to use for n is questionable.

If the flow in the gutter at the both ends of the grate is critical, as assumed in the gutter flow described above, then the solution to the ODE will begin with a depth just slightly below critical depth, resulting in a negative denominator for the ODE, and this causes the depth to generally decrease in the x direction.

EXAMPLE PROBLEM 4.54

Determine the length of a grate needed in the Example Problem 4.52, so that all the accumulated flow in the gutter between grates is discharged into the grates. The grate's discharge coefficient is $C_d = 0.45$, and one-half of its bottom area is opened.



Solution

The length of the grate will be determined by finding both the length needed to discharge the gutter flow from its right end, and its left end, and then these two lengths are added. The program GRATE listed below is designed to solve such a length of grate if the flow is in one-half of a triangular section. The assumption is that the grate is b units wide at the bottom of this gutter, and the discharge coefficient accounts for the transitional effects of going from one-half of a triangular section to a section that has a flat bottom b units wide. The assumption also is that the discharge coefficient is constant, and not a function of the depth, and/or the position along the grate. The program terminates the solution of the ODE if either the depth Y over the grate becomes negative, or if the flow rate becomes negative.

Listing of Program GRATE.FOR

```
C Solve flow from triangular grate, B is width of grate, but
C area is based on having a triangular gutter.
      REAL YY(1),YTT(1)
      COMMON FM,FMS,C,FN,HM,G,Q,QS1,CD,X,SO
      EQUIVALENCE (Y,YY(1))
      WRITE(6,*)'Give:IOUT,b,m,n,So,DX,g,Qb,Cd,Fac A'
      READ(5,*) IOUT,B,FM,FN,SO,DX,G,QB,CD,FA
      DXS=1.
      IF(G.GT.30.) THEN
      C=1.486
      ELSE
      C=1.
      ENDIF
      HM=.5*FM
      FMS=1.+SQRT(FM*FM+1.)
      G8=.125*G
      G2=2.*G
      CD=CD*SQRT(G2)*FA*B
```

```
      YC=((QB/FM)**2/G8)**.2
      WRITE(IOUT,98) YC,QB
98    FORMAT(' Critical Depth at beg. of grate =',F8.3,', Qb =',F8.2)
      Y=.95*YC
      WRITE(IOUT,99) FM,FN,So
99    FORMAT(> b==,F8.1,' m =',F8.3,' n =',F8.4,' So =',F10.6,/,
     &'    x     Y     Q    E    q*    Fr')
      A=HM*YC*YC
      WRITE(IOUT,100) 0.,YC,QB,YC+(QB/A)**2/G2,A,QB/(FM*SQRT
     &(G8*YC**5))
100   FORMAT(F10.2,5F10.3)
      QS1=CD*SQRT(YC)
      X1=0.
      Q=QB
10    X=X1+DX
      CALL RUKUST(1,DXS,X1,X,1.E-4,YY,YTT)
      QS2=CD*SQRT(Y)
      QSA=.5*(QS1+QS2)
      Q=Q-QSA*DX
      A=HM*Y**2
      WRITE(IOUT,100)X,Y,Q,Y+(Q/A)**2/G2,QSA,Q/(FM*SQRT
     &(G8*YC**5))
      QS1=QS2
      X1=X
      IF(Y.GT. .1 .AND. Q.GT.0.) GO TO 10
      END
      SUBROUTINE SLOPE(XX,Y,DYX)
      REAL Y(1),DYX(1)
      COMMON FM,FMS,C,FN,HM,G,Q,QS1,CD,X,SO
      A=HM*Y(1)**2
      QS=CD*SQRT(Y(1))
      QQ=Q-.5*(QS1+QS)*(XX-X)
      IF(QQ.LT.0. .OR. Y(1).LT.0.) THEN
      DYX(1)=0.
      ELSE
      SF=(FN*QQ/C*((Y(1)+FMS*Y(1))/A)**.66666667/A)**2
      DYX(1)=(SO-SF+QQ*QS/(G*A**2))/(1.-QQ*QQ*(FM*Y(1))/(G*A**3))
      ENDIF
      RETURN
      END
```

GRATE.C

```
// Solve flow from triangular grate, b is width of grate,
// but area is based on having a triangular gutter
#include <stdlib.h>
#include <math.h>
#include <stdio.h>
float fm,fms,qs,c,fn,hm,g,g2,q,qs1,cd,x,so;
FILE *filo;
extern void rukust(int ne,float *dxs,float xbeg,float xend,\
   float err,float *y,float *ytt);
void slope(float xx,float *yp,float *dy){
   float a,qs,qq,sf,yy;
   yy=*yp; a=hm*yy*yy; qs=cd*sqrt(yy); qq=q-.5*(qs1+qs)*(xx-x);
   sf=fn*q/c*pow(fms*yy/a,.66666667)/a; sf*=fabs(sf);
   *dy=(so-sf+qq*qs/(g*a*a))/(1.-qq*qq*fm*yy/(g*pow(a,3.)));
```

```
   return; } // End slope
void main(void){
   char fname[20];int i,j,nct;
   float dxs[1],y[1],ytt[1],dif,dx,gm,g8,gm2,qe2,yc5,yc,b,\
      fa,qb,a,x1,qsa,qs2;
   printf("Give name of output file\n");scanf("%s",fname);
   if((filo=fopen(fname,"w"))==NULL){
      printf("Cannot open file");exit(0);}
   printf("GIVE:b,m,n,So,DX,g,Qb & guess Yc,Cd,Fac A\n");
   scanf("%f %f %f %f %f %f %f %f %f %f",\
      &b,&fm,&fn,&so,&dx,&g,&qb,&yc,&cd,&fa);dxs[0]=1.;
   hm=.5*fm;fms=1.+sqrt(fm*fm+1.);
   if(g>20.) c=1.486; else c=1.;
   g2=2.*g;g8=.125*g;gm2=g8*fm*fm;
   gm=6.*gm2;qe2=qb*qb;cd=cd*sqrt(g2)*fa*b;nct=0;
   do{ yc5=pow(yc,5.); dif=(yc*(qe2-gm2*yc5))/(qe2-gm*yc5);
      yc-=dif;}while((++nct<30)&&(fabs(dif)>1.e-5));
   if(nct==30)printf("Critical depth failed. DIF=%f\n",dif);
   fprintf(filo,"Critical Depth at beg. of grate =%8.3f,\
     Qb =%8.2f\n",yc,qb);y[0]=.95*yc;
   fprintf(filo,"b =%8.1f m =%8.3f, n =%8.4f, So =%10.6f\n",
      fm,fn,so);
   fprintf(filo," x      Y      Q      E      q*      Fr\n");
      a=hm*yc*yc;
   fprintf(filo,"%10.1f %9.3f %9.3f %9.3f %9.3f %9.3f\n",\
      0.,yc,qb,yc+pow(qb/a,2.)/g2,a,qb/(fm*sqrt(g8*pow(yc,5.))));
   qs1=cd*sqrt(yc); x1=0.;q=qb;
   do{x=x1+dx; rukust(1,dxs,x1,x,1.e-4,y,ytt);
      qs2=cd*sqrt(y[0]);qsa=.5*(qs1+qs2);
      q-=qsa*dx;a=hm*y[0]*y[0];
      fprintf(filo,"%10.1f, %9.3f %9.3f %9.3f %9.3f %9.3f\n",\
         x,y[0],q,y[0]+pow(q/a,2.)/g2,qsa,\
         q/(fm*sqrt(g8*pow(y[0],5.))));
      qs1=qs2; x1=x; }while((y[0]>.01) && (q>0.));
   fclose(filo);}
```

Input for flow from right end of the gutter:

```
3 1 4 .012 .001 .5 32.2 6.965 .45 .5
```

The critical depth at the beginning of the grate = 0.945, Qb = 6.97

m = 4.000, n = 0.0130, S_o = 0.001000.

x	Y	Q	E	q*	F_r
0.0	0.945	6.965	1.181	1.786	1.000
0.5	0.804	6.122	1.152	1.687	0.879
1.0	0.731	5.331	1.118	1.581	0.765
1.5	0.666	4.577	1.079	1.509	0.657
2.0	0.607	3.857	1.033	1.440	0.554
2.5	0.549	3.170	0.978	1.372	0.455
3.0	0.493	2.519	0.911	1.303	0.362
3.5	0.436	1.904	0.826	1.230	0.273
4.0	0.377	1.329	0.716	1.151	0.191
4.5	0.316	0.798	0.565	1.062	0.115
5.0	0.248	0.320	0.353	0.956	0.046
5.5	0.164	−0.087	0.205	0.815	−.013

Input for flow from left end of the gutter:

```
3 1 4 .012 -.001 .25 32.2 1.840 .554 .45 .5
```

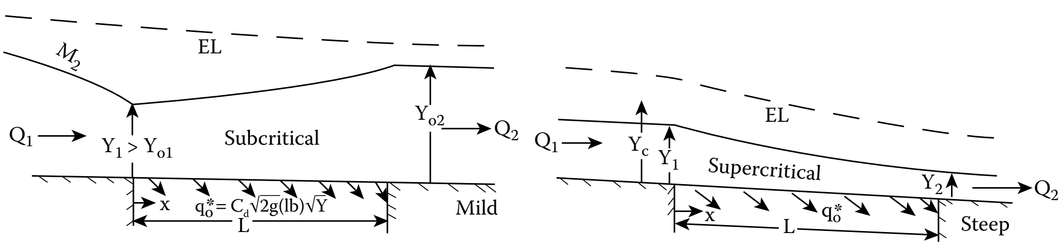
Solution:
Critical depth at the beginning of the grate = 0.555, Qb = 1.84

m = 4.000, n = 0.0130, S_o = −0.001000.

x	Y	Q	E	q*	F_r
0.00	0.555	1.840	0.694	1.490	1.000
0.25	0.454	1.485	0.655	1.419	1.331
0.50	0.395	1.159	0.609	1.303	1.473
.75	0.341	0.856	0.551	1.213	1.571
1.00	0.288	0.576	0.475	1.121	1.607
1.25	0.235	0.320	0.366	1.022	1.498
1.50	0.176	0.094	0.212	0.094	0.905
1.75	0.101	−.090	0.400	0.738	−3.432

Thus, the length of the grate needed is 5.5 + 1.8 = 7.3 ft.

Let us examine the case in which the grate is relatively short, so only a portion of the flow is taken from the grate (i.e., the channel bottom). Then two of the possible situations are shown below; the first contains a subcritical flow through the entire grate length, and upstream and downstream therefrom, the flows are also subcritical with the downstream depth as the control; the second contains a supercritical flow throughout the grate, and the upstream depth Y_1 is the control. If the channel downstream is not steep in the second case, then a supercritical flow over the first portion of the grate will result in a hydraulic jump either within the grate length, or in the downstream channel.



If the channel is mild and the channel is long downstream from the grate, then a normal depth Y_{o2}, based on the flow rate Q_2 remaining in the channel, will exist at the end of the grate for the first case, and since the term $Qq_o^*/(gA^2)$ contributes to the positiveness of the numerator of the ODE and its denominator is positive since $F_r^2 < 1$, the depth increases across the grate, as shown in the sketch. An M_2-GVF will occur upstream from the grate. The solution to this subcritical flow situation is to simultaneously satisfy Manning's equation in the downstream channel and to solve the ODE through the grate to give the variation in depth across the grate, as well as to evaluate $\int q_o^* dx$ so the continuity equation $Q_1 = Q_2 + \int q_o^* dx$ can be satisfied.

If the channel is steep (the second case) so that the upstream flow is supercritical, then the denominator of the ODE is negative and the depth will decrease across the grate, as shown, in the supercritical sketch above. Also, the upstream depth will not be affected by the outflow from the grate so the solution can proceed by solving the ODE starting with depth Y_1 at the upstream end of the grate and let this solution provide the outflow $\int q_o^* dx$, as well as the depth Y_2 at the end of the grate. The solution to the given problem is thus straight forward, and can be accomplished by a general solution of the ODE with the spatially varied flow term included, such as GVFALL.

The details for solving the problem in which the subcritical flow occurs throughout the entire grate length will now be discussed. There are three unknowns involved, Y_1, $Y_2 = Y_{o2}$, and Q_2. The three available equations are: (1) Manning's equation in the downstream channel, (2) the ODE across the grate, and (3) the continuity equation, or

$$F_1 = nQ_2P^{2/3} - C_uA^{5/3}S_o^{1/2} = 0 \tag{4.28a}$$

$$F_2 = Y_1 - Y_{1ode}(Y_{o2}) = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f + Qq_o^*/(gA^2)}{1 - F_r^2} \tag{4.28b}$$

Solved from x = L to x = 0, including the numerical integration to evaluate the total lateral outflow,

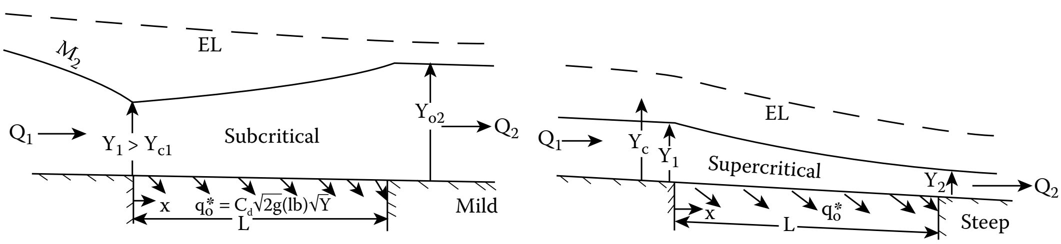
$$F_3 = Q_1 - Q_2 - \int q_o^*\, dx = 0 \tag{4.28c}$$

An alternative is to solve only Y_1 and Q_2 simultaneously, and based on any current value for Q_2, solve Manning's equation (i.e., F_1) for the downstream normal depth Y_{o2} needed in equation F_2. The program GRATMILD is designed to obtain solutions for this case of mild flow through the entire length of the grate. It assumes that the coefficient of discharge is constant but allows the size of the channel over the grate to vary so that the nonprismatic term is included in the numerator of the ODE, and it uses the alternative mentioned above of only solving the two simultaneous equations F_2 and F_3. More will be said about the use of this program later in the next example problem, but first a little theoretical discussion.

Listing of program GRATMILD.FOR

```
C This program is designed to solve GVF over grate if subcritical
  flow occurs over entire length.
      CHARACTER PROMPT(3)*50/'IOUT,Q1,L,b,n,So,DX,g,Cd,fac A,guess
     &for Y1 & Q2','IOUT,Q1,L,b,m,n,So,DX,g,Cd,fac A,guess for
     &Y1 & Q2','IOUT,Q1,L,m,n,So,DX,g,Cd,fac A,guess for Y1 & Q2'/
      CHARACTER PRM(3)*5/'b','b & m','m'/
      LOGICAL DONE
      COMMON B1,FM1,DB,DFM,Q1,FL,FN,SO,G,G2,Q,QS1,CD,DX,DXH,YO,
     &X1,X(2),SOO,AR,TOP,B,FM,DONE,IOUT,IPRISM,ITYPE
      REAL F(2),F1(2),D(2,2)
      WRITE(*,*)' Give: 1=Prismatic C., or 0=not; and Type 1=rect,
     &2=trap, 3=1/2 triag.'
      READ(*,*) IPRISM,ITYPE
      WRITE(*,*)' Give:',PROMPT(ITYPE)
      IF(ITYPE.EQ.1) THEN
      READ(*,*) IOUT,Q1,FL,B1,FN,SO,DX,G,CD,FA,X
      ELSEIF(ITYPE.EQ.2) THEN
      READ(*,*) IOUT,Q1,FL,B1,FM1,FN,SO,DX,G,CD,FA,X
      ELSE
      READ(*,*) IOUT,Q1,FL,FM1,FN,SO,DX,G,CD,FA,X
      WRITE(*,*)' Give width of grate'
      READ(*,*) B1
      B=0.
      ENDIF
```

```
      IF(IPRISM.EQ.1) GO TO 2
      WRITE(*,*)' Give change in ',PRM(ITYPE)
      IF(ITYPE.EQ.1) THEN
      READ(*,*) DB
      DB=DB/FL
      ELSEIF(ITYPE.EQ.2) THEN
      READ(*,*) DB,DFM
      DB=DB/FL
      DFM=DFM/FL
      ELSE
      READ(*,*) DFM
      DFM=DFM/FL
      ENDIF
```




```
2     DX=ABS(DX)
      DXH=.5*DX
      DONE=.FALSE.
      IF(G.GT.30.) THEN
      C=1.486
      ELSE
      C=1.
      ENDIF
      G2=2.*G
      CD=CD*SQRT(G2)*FA*B1
      SOO=SQRT(SO)
      FN=FN/C
      YO=1.2*X(1)
      NCT=0
10    CALL FUN(F)
      DO 14 J=1,2
      DXX=.005*X(J)
      X(J)=X(J)+DXX
      CALL FUN(F1)
      DO 12 I=1,2
12    D(I,J)=(F1(I)-F(I))/DXX
14    X(J)=X(J)-DXX
      FAC=D(2,1)/D(1,1)
      D(2,2)=D(2,2)-FAC*D(1,2)
      F(2)=F(2)-FAC*F(1)
      Z2=F(2)/D(2,2)
      X(2)=X(2)-Z2
      Z1=(F(1)-Z2*D(1,2))/D(1,1)
      X(1)=X(1)-Z1
```

```
      NCT=NCT+1
      WRITE(*,99) NCT,Z1,Z2,X
99    FORMAT(' NCT,Z1,Z2,X=',I3,4E12.5)
      IF(ABS(Z1)+ABS(Z2).GT.1.E-5 .AND. NCT.LT.30) GO TO 10
      DONE=.TRUE.
      WRITE(IOUT,100) X,Q1-X(2)
100   FORMAT(' Depth at Beg.=',F8.3,', Remain. Q=',F9.2,', Outflow=',
     &F8.3,/,'   x    Y    P    A    E    Q   q*',/,1X,55('-'))
      CALL FUN(F)
      END
      SUBROUTINE FUN(F)
      REAL F(2),Y(1),YTT(1)
      LOGICAL DONE
      COMMON B1,FM1,DB,DFM,Q1,FL,FN,SO,G,G2,Q,QS1,CD,DX,DXH,YO,
     &X1,X(2),SOO,AR,TOP,B,FM,DONE,IOUT,IPRISM,ITYPE
      NCT=0
1     FF=FN*X(2)*P(FL,YO)**.666667-SOO*AR**1.6666667
      NCT=NCT+1
      IF(MOD(NCT,2).NE.0) THEN
      FF1=FF
      YOO=YO
      YO=1.01*YO
      GO TO 1
      ENDIF
      DIF=(YO-YOO)*FF1/(FF-FF1)
      YO=YOO-DIF
      IF(ABS(DIF).GT.1.E-5 .AND. NCT.LT.30) GO TO 1
      DXS=-.1
      Y(1)=YO
      X1=FL
      Q=X(2)
      QS1=CD*SQRT(Y(1))
      IF(DONE) WRITE(IOUT,100) X1,Y,P(X1,Y(1)),AR,Y(1)+(X(2)/AR)**2/
     &G2,Q,QS1
100   FORMAT(F8.2,F8.3,2F8.2,F8.3,F8.2,F8.3)
10    X2=X1-DX
      IF(X2.LT.0.) X2=0.
      CALL RUKUST(1,DXS,X1,X2,1.E-4,Y,YTT)
      QS2=CD*SQRT(Y(1))
      Q=Q+DXH*(QS1+QS2)
      IF(DONE) WRITE(IOUT,100)X2,Y,P(X2,Y(1)),AR,Y(1)+(Q/AR)**2/
     &G2,Q,QS2S1=QS2
      X1=X2
      IF(X2.GT.0.) GO TO 10
      F(1)=X(1)-Y(1)
      F(2)=Q1-Q
      RETURN
      END
      SUBROUTINE SLOPE(XX,Y,DY)
      REAL Y(1),DY(1)
      LOGICAL DONE
```

```
      COMMON B1,FM1,DB,DFM,Q1,FL,FN,SO,G,G2,Q,QS1,CD,DX,DXH,YO,
     &X1,X(2),SOO,AR,TOP,B,FM,DONE,IOUT,IPRISM,ITYPE
      PP=P(XX,Y(1))
      QS=CD*SQRT(Y(1))
      QQ=Q+.5*(X1-XX)*(QS+QS1)
      SF=(FN*QQ*(PP/AR)**.666667/AR)**2
      IF(IPRISM.EQ.1) THEN
      DAF=0.
      ELSE
      DAF=(QQ/AR)**2/(G*AR)*(DB+DFM*Y(1))*Y(1)
      ENDIF
      DY(1)=(SO-SF+QQ*QS/(G*AR**2)+DAF)/(1.-QQ*QQ*TOP/(G*AR**3))
      RETURN
      END
      FUNCTION P(XX,Y)
      LOGICAL DONE
      COMMON B1,FM1,DB,DFM,Q1,FL,FN,SO,G,G2,Q,QS1,CD,DX,DXH,YO,
     &X1,X(2),SOO,AR,TOP,B,FM,DONE,IOUT,IPRISM,ITYPE
      IF(ITYPE.EQ.1) THEN
      B=B1
      IF(IPRISM.EQ.0) B=B+XX*DB
      P=B+2.*Y
      AR=B*Y
      TOP=B
      ELSEIF(ITYPE.EQ.2) THEN
      B=B1
      FM=FM1
      IF(IPRISM.EQ.1) GO TO 2
      B=B+XX*DB
      FM=FM+XX*DFM
2     P=B+2.*Y*SQRT(FM*FM+1.)
      AR=(B+FM*Y)*Y
      TOP=B+2.*FM*Y
      ELSE
      FM=FM1
      IF(IPRISM.EQ.0) FM=FM+XX*DFM
      P=Y*(1.+SQRT(FM*FM+1.))
      AR=.5*FM*Y*Y
      TOP=FM*Y
      ENDIF
      RETURN
      END
```

GRATMILD.C

```
// This program is designed to solve GVF over grate if
// subcritical flow occurs over entire length
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
float b1,fm1,db,dfm,q1,fl,fn,so,g,g2,q,qs1,cd,dx,dxh,yo,\
   x1,x[2],soo,ar,top,b,fm;
```

```
int done,iprism,itype; FILE *filo;
extern void rukust(int ne,float *dxs,float xbeg,float xend,\
   float err,float *y,float *ytt);
float p(float xx,float y){
   if(itype==1){b=b1;
      if(iprism==0) b+=xx*db;ar=b*y;top=b;return(b+2.*y);}
   else if(itype==2){b=b1;fm=fm1;
      if(iprism!=1){b+=xx*db;fm+=xx*dfm;}\
      ar=(b+fm*y)*y;top=b+2.*fm*y;
      return(b+2.*y*sqrt(fm*fm+1.));}
   else{fm=fm1;
      if(iprism==0)fm+=xx*dfm;ar=.5*fm*y*y;top=fm*y;
      return(y*(1.+sqrt(fm*fm+1.)));}}//End p
void slope(float xx,float *yp,float *dy){
   float pp,qs,qq,sf,daf,y;
   y=*yp; pp=p(xx,y);qs=cd*sqrt(y);qq=q+.5*(x1-xx)*(qs+qs1);
   sf=pow(fn*qq*pow(pp/ar,.6666667)/ar,2.);
   if(iprism==1) daf=0.;
   else daf=pow(qq/ar,2.)/(g*ar)*(db+dfm*y)*y;
   *dy=(so-sf+qq*qs/(g*ar*ar)+daf)/(1.-qq*qq*top/(g*ar*ar*ar));}
   // End slope
void fun(float *f){
   int nct;float ff,ff1,yoo,dif,dxs[1],x2,qs2,y[1],ytt[1];
   nct=0;
L1: ff=fn*x[1]*pow(p(fl,yo),.6666667)-soo*pow(ar,1.6666667);
   nct++;
   if(nct%2){ff1=ff;yoo=yo;yo*=1.01;goto L1;}
   dif=(yo-yoo)*ff1/(ff-ff1);
   yo=yoo-dif; if((fabs(dif)>1.e-5) && (nct<30)) goto L1;
   dxs[0]=-.1; y[0]=yo;x1=fl;q=x[1];qs1=cd*sqrt(y[0]);
   if(done)fprintf(filo,"%8.2f %7.3f %7.2f %7.2f %7.3f %7.2f \
      %7.3f\n",x1,y[0],p(x1,y[0]),ar,y[0]+\
      pow(x[1]/ar,2.)/g2,q,qs1);
L10: x2=x1-dx; if(x2<0.) x2=0.;rukust(1,dxs,x1,x2,1.e-4,y,ytt);
   qs2=cd*sqrt(y[0]);q+=dxh*(qs1+qs2);
   if(done)fprintf(filo,"%8.2f %7.3f %7.2f %7.2f %7.3f %7.2f
      %7.3f\n",x2,y[0],p(x2,y[0]),ar,y[0]+\
      pow(q/ar,2.)/g2,q,qs2);
   qs1=qs2; x1=x2; if(x2>0.) goto L10;
   f[0]=x[0]-y[0]; f[1]=q1-q; } // End fun
void main(void){
   int nct=0,i,j,itypm; float fa,fac,z1,z2,c,dxx,f[2],f1[2],**d;
   char *prompt[3]={
      "Q1,L,b,n,So,DX,g,Cd,fac A,guess for Y1 & Q2",\
      "Q1,L,b,m,n,So,DX,g,Cd,fac A,guess for Y1 & Q2",\
      "Q1,L,m,n,So,DX,g,Cd,fac A,guess for Y1 & Q2"};
   char *prm[3]={"b","b & m","m"}; char fname[20];
   printf("Give name of output file\n");scanf("%s",fname);
   if((filo=fopen(fname,"w"))==NULL){
      printf("Cannot open file");exit(0);}
   d=(float**)malloc(2*sizeof(float*));
```

```
for(i=0;i<2;i++)d[i]=(float*)malloc(2*sizeof(float));
printf("Give: 1=prismatic C., or 0=not; and type 1=rect,\
   2=trap,3=1/2 triag.\n");
scanf("%d %d",&iprism,&itype);itypm=itype-1;
printf("Give:%s\n",prompt[itypm]);
if(itype==1)scanf("%f %f %f %f %f %f %f %f %f %f %f",\
   &q1,&fl,&b1,&fn,&so,&dx,&g,&cd,&fa,&x[0],&x[1]);
else if(itype==2)
   scanf("%f %f %f %f %f %f %f %f %f %f %f %f",\
      &q1,&fl,&b1,&fm1,&fn,&so,&dx,&g,&cd,&fa,&x[0],&x[1]);
else {
   scanf("%f %f %f %f %f %f %f %f %f %f %f",\
      &q1,&fl,&fm1,&fn,&so,&dx,&g,&cd,&fa,&x[0],&x[1]);
   printf("Give width of grate\n"); scanf("%f",&b1); b=0;}
if(iprism!=1){printf("Give change in %s\n",prm[itypm]);
   if(itype==1){scanf("%f",&db);db/=fl;}
   else if(itype==2){scanf("%f %f",&db,&dfm);db/=fl;dfm/=fl;}
   else {scanf("%f",&dfm);dfm/=fl;}}
dx=fabs(dx);dxh=.5*dx;done=0;
if(g>30.) c=1.486; else c=1.; g2=2.*g; cd*=sqrt(g2)*fa*b1;
soo=sqrt(so);fn/=c;yo=1.2*x[0];nct=0;
do{fun(f);
   for(j=0;j<2;j++){
      dxx=.005*x[j];x[j]+=dxx;fun(f1);
      for(i=0;i<2;i++)d[i][j]=(f1[i]-f[i])/dxx;x[j]-=dxx;}
   fac=d[1][0]/d[0][0];d[1][1]-=fac*d[0][1];f[1]-=fac*f[0];
   z2=f[1]/d[1][1];x[1]-=z2;
   z1=(f[0]-z2*d[0][1])/d[0][0];x[0]-=z1;
   printf("NCT,Z1,Z2,X= %d %11.5e %11.5e %11.5e %11.5e\n",\
   ++nct,z1,z2,x[0],x[1]);
}while(((fabs(z1)+fabs(z2))>1.e-5) && (nct<30));
done=1;
fprintf(filo,"Depth at Beg.=%8.3f, Remain. Q=%9.2f,\
   Outflow= %8.3f\n",x[0],x[1],q1-x[1]);
fprintf(filo," x Y P A E Q q*\n");
for(i=0;i<55;i++)fprintf(filo,"-");fprintf(filo,"\n");
fun(f); fclose(filo);}
```

A closed-form solution to the orifice outflow problem in a rectangular channel is possible if the following assumptions are made: (1) the energy line is parallel to the bottom, i.e., $S_f = S_o$, (in other words, the specific energy E is constant), (2) the lateral outflow is proportional to the square root of the specific energy E according to $q_o^* = C_e\sqrt{2g}(bf)\sqrt{E}$ (note, this makes q_o^* constant), and (3) that this special discharge coefficient C_e is constant. The ODE then becomes

$$\frac{dY}{dx} = \frac{Qq_o^*/(gA^2)}{1-Q^2b/(gA^3)} \tag{4.29a}$$

by substituting Q from the definition of the specific energy ($Q = bY[2g(E-Y)]^{.5}$ into this equation and noting $q_o^* = -dQ/dx$, it becomes

$$\frac{dY}{dx} = \frac{2fC_e[E(E-Y)]^{0.5}}{3Y-2E} \quad \text{Note, for the critical flow, the denominator becomes zero.} \tag{4.29b}$$

and the solution is

$$x = K - \frac{Y}{fC_e}\left\{1 - \frac{Y}{E}\right\}^{0.5} \tag{4.30}$$

(You can verify this solution by taking the differentials of both sides and substituting into the ODE.) in which the constant of the integration K is evaluated by substituting the appropriate boundary condition into the equation. If the flow is subcritical throughout the grate, then the downstream normal depth Y_{o2} is used for Y and the length of the grate L for x. This substitution gives

$$K = L + Y_{o2}/(fC_e)\{1 - Y_{o2}/E\}^{0.5} \tag{4.31}$$

and E can be determined from $E = Y_{o2}+(Q/A_{o2})^2/(2g)$. If the flow is supercritical, then the upstream depth Y_1 is substituted for Y and x = 0, giving

$$K = Y_1/(fC_e)\{1 - Y_1/E\}^{0.5} \tag{4.32}$$

and $E = Y_1+(Q/A_1)^2/(2g)$.

While the assumptions made to obtain Equation 4.30 make the results therefrom only approximate since it is a closed-form solution, it can be used to determine what range of fC_e are valid for a given length L, or what maximum length L a grate can have for a given fC_e before the critical depth occurs at the beginning of the grate for a known depth Y_{o2} (and E) at the end of the grate. To make this analysis, the more general Equation 4.30 will be divided by E making the terms dimensionless. At the upstream end Y/E equals 2/3 for the critical flow, and from Equation 4.30 the following dimensionless length parameter results

$$fC_e(L/E) = 0.3849018 - (Y_{o2}/E)\{1 - Y_{o2}/E\}^{0.5} \tag{4.33}$$

and the fraction of flow remaining in the channel is given by

$$q_2/q_1 = \{6.75(1 - Y_{o2}/E)(Y_{o2}/E)^2\}^{0.5} = 2.59808(Y_{o2}/E)\{1 - Y_{o2}/E\}^{0.5} \tag{4.34}$$

The results from evaluating these quantities for several values of Y_{o2}/E are given in the table below.

Y_{o2}/E	fC_e (L/E)	q_2/q_1
0.670	0.1609E–04	0.99996
0.700	0.1496E–02	0.99612
0.750	0.9902E–02	0.97428
0.800	0.2713E–01	0.92952
0.850	0.05570	0.85530
0.900	0.10030	0.73943
0.950	0.17248	0.55190
0.960	0.19290	0.49883
0.970	0.21689	0.43650
0.975	0.23074	0.40052
0.980	0.24631	0.36007
0.985	0.26426	0.31342
0.990	0.28590	0.25721
0.995	0.31454	0.18279
0.996	0.32191	0.16366
0.997	0.33029	0.14188
0.998	0.34027	0.11596
0.999	0.35331	0.08208
1.000	0.38490	0.00000

Thus, for example, if the specific energy E = 4, the downstream depth Y_{o2} equals 0.9 times the specific energy, or 3.6 (or the downstream velocity head equals 0.1 times E, or 0.4), and C_e = 0.4 and f = 0.5, then since $fC_e(L/E) = 0.1003$ (the table value with $Y_{o2}/E = 0.9$) the maximum length the grate can have without having the critical flow at its beginning is $L = E(0.1003)/(fC_e) = 4(0.1003)/0.2 = 2.006$, and the downstream channel would contain $q_2 = 0.73943q_1$ (or $Q_2 = 0.73943Q_1$). With no flow remaining after the grate, $fC_e(L/E) = 0.3849$. Thus, for a grate that is 5 ft long, and if E = 4 ft, then the product of the fraction of the area open times the discharge coefficient fC_e can have a maximum value $fC_e = 0.3849(4/5) = 0.308$ in order to have the case of the subcritical flow through the grate length.

An analysis for the case of the supercritical flow of lengths needed to discharge the entire flow can be obtained by substituting Y = 0 into Equation 4.30. If Equation 4.30 is first made nondimensional by dividing by E, then this dimensionless length parameter is given by, $fC_e(L/E) = (Y_1/E)\{1 - Y_1/E\}^{0.5}$, and the upstream flow rate per unit width can be obtained from $q^2/(gY_1^3) = F_{r1}^2 = 2(E/Y_1 - 1)$. The following table provides values of both of these parameters with Y_1/E starting with the critical depth.

Y_1/E	fC_e (L/E)	F^2_{r1}
2/3	0.38490	1.00000
0.66	0.38484	1.03030
0.65	0.38455	1.07692
0.60	0.37947	1.33333
0.55	0.36895	1.63636
0.50	0.35355	2.00000
0.45	0.33373	2.44444
0.40	0.30984	3.00000
0.35	0.28218	3.71429
0.30	0.25100	4.66667
0.25	0.21651	6.00000
0.20	0.17889	8.00000
0.15	0.13829	11.33333
0.10	0.09487	18.00000

EXAMPLE PROBLEM 4.55

A bottom rack (grate) that is 4 ft wide and 2 ft long has bars that are parallel to the direction of the flow that covers one-half of the bottom area of the rectangular channel, with n = 0.013 and S_o = 0.0005. If the flow rate upstream from the rack is Q_1 = 60 cfs, and the discharge coefficient is C_d = 0.4, what flow rate Q_2 remains in the channel after the rack, and at what depth? What is the depth at the beginning of the rack? How do the results based on the simplified theory for which a closed form solution is available compare with the numerical results?

Solution

The input to program GRATMILD to solve this problem consists of 1 1

3 60 2 4 .013 .0005 .25 32.2 .4 .5 4.3 43.5

and the output consists of

Depth at Beg. = 2.919, Remain. Q = 37.49, Outflow = 22.513.

x	Y	P	A	E	Q	q*	Y_{theo} (C_e=0.3844)
2.00	3.194	10.39	12.78	3.328	37.49	11.474	3.194
1.75	3.171	10.34	12.68	3.328	40.35	11.431	3.171
1.50	3.145	10.29	12.58	3.328	40.20	11.385	3.145
1.25	3.116	10.23	12.46	3.328	46.04	11.333	3.117
1.00	3.084	10.17	12.34	3.328	48.87	11.275	3.085
0.75	3.049	10.10	12.20	3.328	51.68	11.211	3.050
0.50	3.011	10.02	12.04	3.328	54.47	11.139	3.011
0.25	2.967	9.93	11.87	3.328	57.25	11.059	2.966
0.00	2.919	9.84	11.67	3.329	60.00	10.968	2.916

To obtain the theoretical solution one must decide what value of E to use. The average is $(3.328 + 3.329)/2 = 3.3285$. Then if q_o^* is taken as $22.513/2 = 11.2566$ cfs/ft, the value of C_e can be computed as 0.3844 and $K = 5.340553$. The last column in the above table gives the Y's obtained from Equation 4.30. Notice, these values are essentially identical with those obtained from the numerical solution. One would anticipate these results since the specific energy is essentially constant, and the outflow q_o^* does not vary much. Assuming that the same n and S_o apply for the channel upstream from the rack, then the normal depth here is $Y_{o1} = 4.686$ ft. Therefore, the M_2-GVF upstream from the rack reduces the depth from 4.686 to 2.196 ft, or 1.767 ft and its length is 10,280 ft to 1% of the normal depth.

If the grate is followed by another gutter with a lateral inflow, or something else so that the normal depth does not exist downstream, then we may need to solve the problem with the subcritical flow over the grate, but be able to specify what the flow rate is that leaves the end of the grate, as well as what flow rate enters. In other words, we wish to know the depths at the beginning and at the end of the grate that will cause a specified amount of outflow over the grate. For this problem, two equations are available; the solution of the ODE across the grate and the continuity equation, namely, Equations 4.28b and c. The only difference is that the depth Y_2 will be used in place of the normal depth Y_{o2} in the first equation (Equation 4.28b). These two equations can be used to solve the depths at the beginning and at the end of the grate, Y_1 and Y_2, respectively, with the flow rates entering and leaving the grate Q_{in} and Q_{out}, respectively, specified. Program GRATE2E is designed to solve this problem. This program uses X(1) for the depth Y_2 and X(2) for the depth Y_1. The subroutine DYXG defines dY/dx for the ODE solver ODESOL in solving the profile over the length of the grate, i.e., determining $Y_{1ode}(Y_2)$ in Equation 4.28b by starting with the depth Y_2. The Newton method is again used to solve Equations 4.28b and c simultaneously.

Listing of program GRATE2E.FOR (also see diskette for GRATE2E.C)

```
C Solves the problem of outflow from grates in its bottom of a
C triangular channel. The case being solved is when Q = Qout at
C the end of the grate and a specified  flow rate Qin enters at
C its beginning.
C Y2=X(1), Y1=X(2)
      INTEGER*2 INDX(2)
      EXTERNAL DYX
      LOGICAL DONE
      REAL F(2),D(2,2),FFF(2)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(2),SO,FM,FMS,Qin,Qout,CC,FN,HM,G,G2,G8,DXG,TOL,
     &FLG,DXGH,CD,SQSTAR,QS1,XX,DONE,IOUT
      WRITE(*,*)'Give:IO,TOL,ERR,n,So,m,Lgrate,g,DXG,Qin,Qout,b,
     &frac,Cd'
      READ(*,*) IOUT,TOL,ERR,FN,SO,FM,FLG,G,DXG,Qin,Qout,B,FRAC,CD
      IOU1=IOUT+1
      DONE=.FALSE.
      DXGH=DXG/2.
      HM=.5*FM
      FMS=1.+SQRT(FM*FM+1.)
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
```

```
      CD=CD*SQRT(G2)*FRAC*B
      G8=.125*G
      WRITE(*,*)' GIVE guess for:Y2 & Y1'
      READ(*,*) X
7     NCT=0
8     CALL FUN(F)
      WRITE(*,*) NCT,F
      DO 10 J=1,2
      DXX=.005*X(J)
      X(J)=X(J)+DXX
      CALL FUN(FFF)
      DO 9 I=1,2
9     D(I,J)=(FFF(I)-F(I))/DXX
10    X(J)=X(J)-DXX
      CALL SOLVEQ(2,1,2,D,F,1,DD,INDX)
      NCT=NCT+1
      SUM=0.
      DO 20 I=1,2
      X(I)=X(I)-F(I)
20    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,X
110   FORMAT(' NCT=',I3,' SUM=',E12.6,/,6F10.3)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 8
      WRITE(*,*)' Give:1=write profiles, 2=just Y''s, 3=both'
      READ(*,*) III
      Frr=SQRT(Qin**2*FM*X(2)/(G*(HM*X(2)**2)**3))
      Frl=SQRT(Qout**2*FM*X(1)/(G*(HM*X(1)**2)**3))
      IF(III.GT.1) THEN
      WRITE(IOU1,151) X,Frr,Frl
151   FORMAT(4F7.3)
      IF(III.EQ.2) GO TO 45
      ENDIF
      WRITE(IOUT,100) X,Frr,Frl
100   FORMAT(' Y2=',F7.3,', Y1=',F7.3,', (Fr)u=',F7.3,', (Fr)d=',F7.3)
      DONE=.TRUE.
      CALL FUN(F)
45    WRITE(*,*)' Give 1 to solve another problem or 0 = STOP'
      READ(*,*) III
      IF(III.EQ.0) STOP
      WRITE(*,*)' Give new Qin & Qout'
      READ(*,*) Qin,Qout
      GO TO 7
      END
      SUBROUTINE FUN(F)
      LOGICAL DONE
      EXTERNAL DYXG
      REAL WW(1,13),Y(1),DY(1),XP(1),YP(1,1),F(2)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(2),SO,FM,FMS,Qin,Qout,CC,FN,HM,G,G2,G8,DXG,TOL,
     &FLG,DXGH,CD,SQSTAR,QS1,XX,DONE,IOUT
      HMIN=1.E-6
```

```
      H1=.005
      XX=0.
      SQSTAR=0.
      QS1=CD*SQRT(X(2))
      Y(1)=X(2)
      IF(DONE) THEN
      A=HM*X(2)**2
      WRITE(IOUT,120) 0.,X(2),Qin,QS1,X(2)+(Qin/A)**2/G2,A,Qin/(FM*
     &SQRT(G8*X(2)**5))
120   FORMAT(/,'  x  Y  Q  q*  E  A  Fr',/,1X,69('-'),/,F10.1,
     &6F10.3)
      ENDIF
30    XZ=XX+DXG
      IF(XZ.GT.FLG) XZ=FLG
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYXG)
      QS2=CD*SQRT(Y(1))
      SQSTAR=SQSTAR+DXGH*(QS1+QS2)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=Qin-SQSTAR
      WRITE(IOUT,130) XZ,Y,Q,QS2,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
130   FORMAT(F10.3,6F10.3)
      ENDIF
      QS1=QS2
      XX=XZ
      IF(XZ.LT.FLG) GO TO 30
      F(1)=X(1)-Y(1)
      F(2)=Qin-Qout-SQSTAR
      RETURN
      END
      SUBROUTINE DYXG(XP,Y,DY)
      LOGICAL DONE
      REAL Y(1),DY(1)
      COMMON /TRAS/X(2),SO,FM,FMS,Qin,Qout,CC,FN,HM,G,G2,G8,DXG,
     &TOL,FLG,DXGH,CD,SQSTAR,QS1,XX,DONE,IOUT
      YY=Y(1)
      QS2=CD*SQRT(YY)
      A=HM*YY**2
      Q=Qin-SQSTAR-.5*(XP-XX)*(QS1+QS2)
      SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
      SF=SF*ABS(SF)
      FR2=Q*Q*FM*YY/(G*A**3)
      DY(1)=(SO-SF+Q*QS2/(G*A**2))/(1.-FR2)
      RETURN
      END
```

EXAMPLE PROBLEM 4.56

Obtain a series of solutions to determine how the depths at the beginning and at the end of a 1 ft long grate with one-half of a bottom width of 4 ft open, and a discharge coefficient of $C_d = 0.45$. Assume this grate length exists in a triangular gutter with a side slope of 4, a Manning's n = 0.013, and a bottom slope $S_o = 0.0002$. For this series of solutions, specify the flow past the end of the grate ($Q_{out} = 0$) and start with a flow $Q_{in} = 8.8$ cfs at its beginning and obtain solutions both increasing and decreasing from this value.

Solution

The input to program GRATE2E consists of the following two lines followed by a series of new values for Q_{in} and Q_{out}.

```
3 1.e-5 .001 .013 .0002 4 1 32.2 .05 8.8 0. 4 .5 .45
1.45 1.4
```

The solution giving the profile across the grate for $Q_{in} = 8.8$ cfs consists of the following, and the tables given thereafter give the depths and the Froude numbers associated with them for the series of solutions in which Q_{in} has been increased and then decreased from 8.8 cfs.

$Y_2 = 1.507$, $Y_1 = 1.436$, $(Fr)_u = 0.444$, $(Fr)_d = 0.000$.

x	Y	Q	q*	E	A	F_r
0.0	1.436	8.800	8.654	1.507	4.123	0.444
0.050	1.444	8.367	8.679	1.507	4.171	0.416
0.100	1.451	7.932	8.701	1.507	4.214	0.389
0.150	1.458	7.497	8.722	1.507	4.253	0.364
0.200	1.464	7.060	8.740	1.506	4.289	0.339
0.250	1.470	6.623	8.757	1.506	4.322	0.315
0.300	1.475	6.184	8.772	1.506	4.352	0.292
0.350	1.480	5.745	8.786	1.506	4.379	0.269
0.400	1.484	5.306	8.798	1.506	4.404	0.246
0.450	1.488	4.866	8.809	1.506	4.427	0.225
0.500	1.491	4.425	8.819	1.506	4.447	0.203
0.550	1.494	3.984	8.828	1.506	4.465	0.182
0.600	1.497	3.542	8.836	1.506	4.481	0.161
0.650	1.499	3.100	8.843	1.506	4.495	0.140
0.700	1.501	2.658	8.849	1.506	4.507	0.120
0.750	1.503	2.215	8.854	1.506	4.517	0.100
0.800	1.504	1.772	8.858	1.507	4.525	0.080
0.850	1.505	1.330	8.861	1.507	4.531	0.060
0.900	1.506	0.886	8.863	1.507	4.536	0.040
0.950	1.506	0.443	8.864	1.507	4.538	0.020
1.000	1.507	0.000	8.865	1.507	4.539	0.000

Q_{in}	Q_{out}	Y_2	Y_1	F_{r1}	F_{r2}
8.8000	0.0000	1.5065	1.4358	0.4439	0.0000
9.0000	0.0000	1.5718	1.5116	0.3992	0.0000
9.2000	0.0000	1.6391	1.5873	0.3612	0.0000
9.4000	0.0000	1.7083	1.6635	0.3282	0.0000
9.6000	0.0000	1.7794	1.7403	0.2994	0.0000
9.8000	0.0000	1.8523	1.8180	0.2740	0.0000
10.0000	0.0000	1.9269	1.8968	0.2515	0.0000
10.2000	0.0000	2.0032	1.9766	0.2314	0.0000
10.4000	0.0000	2.0812	2.0577	0.2134	0.0000
10.6000	0.0000	2.1609	2.1400	0.1972	0.0000
10.8000	0.0000	2.2422	2.2235	0.1826	0.0000
11.0000	0.0000	2.3252	2.3085	0.1693	0.0000
11.2000	0.0000	2.4097	2.3947	0.1573	0.0000
11.4000	0.0000	2.4959	2.4824	0.1463	0.0000

(*continued*)

(continued)

Q_{in}	Q_{out}	Y_2	Y_1	F_{r1}	F_{r2}
11.6000	0.0000	2.5836	2.5715	0.1363	0.0000
11.8000	0.0000	2.6729	2.6620	0.1272	0.0000
12.0000	0.0000	2.7638	2.7539	0.1188	0.0000
12.5000	0.0000	2.9980	2.9902	0.1007	0.0000
13.0000	0.0000	3.2419	3.2357	0.0860	0.0000
13.5000	0.0000	3.4955	3.4905	0.0739	0.0000
14.0000	0.0000	3.7588	3.7547	0.0639	0.0000
14.5000	0.0000	4.0317	4.0284	0.0555	0.0000
15.0000	0.0000	4.3143	4.3115.	0.0484	0.0000
15.5000	0.0000	4.6065	4.6042	0.0425	0.0000
16.0000	0.0000	4.9083	4.9064	0.0374	0.0000
17.0000	0.0000	5.5407	5.5393	0.0293	0.0000
18.0000	0.0000	6.2116	6.2105	0.0233	0.0000
19.0000	0.0000	6.9208	6.9200	0.0188	0.0000
20.0000	0.0000	7.6684	7.6677	0.0153	0.0000

Q_{in}	Q_{out}	Y_2	Y_1	F_{r1}	F_{r2}
8.8000	0.0000	1.5065	1.4358	0.4439	0.0000
8.6000	0.0000	1.4434	1.3595	0.4973	0.0000
8.4000	0.0000	1.3827	1.2812	0.5634	0.0000
8.2000	0.0000	1.3247	1.1984	0.6500	0.0000
8.0000	0.0000	1.2701	1.1020	0.7819	0.0000
8.0000	0.0000	1.2701	1.1020	0.7819	0.0000
7.9200	0.0000	1.2497	1.0492	0.8753	0.0000
7.9000	0.0000	1.2449	1.0304	0.9135	0.0000
7.8900	0.0000	1.2425	1.0180	0.9404	0.0000

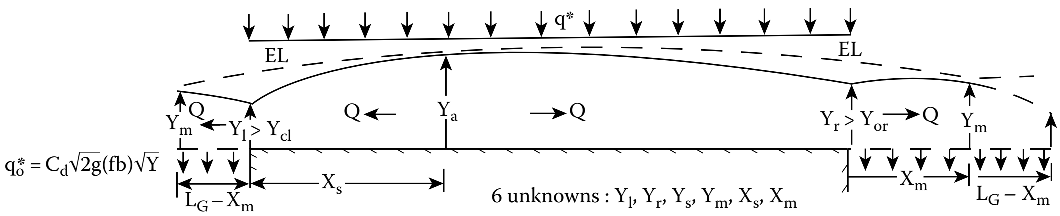
Notice as Q_{in} increases above 8.8 cfs that both Y_2 and Y_1 increase, and the Froude number F_{r1} at the upstream end of the grate decreases. However, as the incoming flow rate decreases, the Froude number at the beginning of the grate increases and when $Q_{in} = 7.89$ cfs, is close to unity ($F_{r1} = 0.9404$). The program fails to obtain a solution for $Q_{in} = 7.88$ cfs because the upstream depth becomes too close to the critical depth. This means, that for smaller incoming flow rates, the flow at the beginning of the grate goes supercritical.

You might wish to study the effects on depths, etc., for situations in which there is flow in the gutter downstream from the grate, $Q_{out} > 0$, but the same total outflow through the grate as in the above tables.

4.20.3 Combined Problem: Gutter Inflow and Grate Outflow

If the length of the grate (or lateral outflow length) is not longer than required for all the gutter flow at the beginning of the grate to enter so the depth feathers down to zero, then it is necessary to combine the problems of lateral inflow along the length of the gutter with the lateral outflow along the length of the grate. In other words, the length of the grate (the outflow length) is short enough in comparison to the amount of flow so that a depth of water will exist over the entire outflow section. We will first consider the case where the flow throughout both the lateral inflow length as well as the lateral outflow length are subcritical. This means that the depth of water over the grate or the outflow length is sufficient that it causes the depths in the gutter both upstream and downstream from the grate to be above the critical depth. For this case, the depths throughout the lateral outflow length of the grate are subcritical, and this outflow is not separated from the lateral inflow by a control caused by the depth being critical at either the upstream or the downstream end of the grate (or gutter).

Assume that there are a series of grates equally spaced along a gutter that receives a constant lateral inflow q*, so that the problem can be defined as depicted in the sketch below in which the length being considered can start in the gutter at the end of the grate, proceed through the gutter, and then finally though the grate. If one wishes to find the point X_m in the grate where the flow rate Q goes to zero, then the control section can begin here and then proceed to the gutter, then through the gutter length, and finally through the grate to the same point in the outflow length where the flow rate is zero.



Assume the following variables are known: (1) The variables that describe the cross section, and these are constant, as well as the bottom slope S_o, and this is constant across the gutter as well as the grate, and Manning's n. (2) The length L of the gutter. (3) The length L_G of the grate. (4) The lateral inflow q* (constant). The unknown variables are (i) The depth on the left side Y_l at the position between the lateral inflow and the lateral outflow sections. This depth we are assuming will be larger than the critical depth associated with the flow at this position $Q_l = X_s q^*$. (ii) The depth on the right side Y_r between the lateral inflow and the lateral outflow sections. This depth will also be larger than the critical depth associated with the flow rate $Q_r = (L - X_s)q^*$ at this point. (iii) The depth Y_s in the gutter's lateral inflow length where the flow rate (and the velocity) are zero, i.e., the flow separates from moving toward the left to toward the right. (iv) The position X_s where the flow in the gutter separates from moving toward the left to toward the right. In addition to these four variables one might add the following: (v) The depth Y_m within the grate outflow length where the flow separates in moving upstream to downstream, i.e., where Q = 0 within the grate length. (vi) The position X_m where this Q = 0 occurs. These latter two variables shown on the sketch can be determined by the solution of the spatially varied flow throughout the entire grate length. (See a homework problem to solve the additional two variables, i.e., considering all of the above six variables as unknowns.) In the description that follows, we will consider the first four variables as unknown.

To solve these four unknown variables, four equations are required. These four equations are

$$F_1 = Y_s - Y_{sode}(Y_l) = 0 \quad \text{with ODE solved from } x = 0 \text{ to } x = X_s \tag{4.35a}$$

$$F_2 = Y_s - Y_{sode}(Y_r) = 0 \quad \text{with ODE solved from } x = L \text{ to } x = X_s \tag{4.35b}$$

$$F_3 = Y_l - Y_{lode}(Y_r) = 0 \quad \text{with ODE solve from } x'\ 0 \text{ to } x'\ L_G \tag{4.35c}$$

$$F_4 = Lq^* - \int_0^{L_G} q_o^*\,dx = 0 \tag{4.35d}$$

Note that the first two equations are identical to Equations 4.21c and d used to solve the gutter flow in which we assumed the critical depth at both ends. The difference is that the two critical flow equations used previously (Equations 4.21a and b) are now replaced by a third ODE through the grate outflow length, and a continuity equation that numerically evaluates the outflow through the grate length and equates this total outflow to the total inflow though the gutter length Lq*.

Before dealing with the simultaneous solutions of these four equations, it is well to conceptionalize what must occur for this case to apply. We have already noted that because the lateral inflow term $2Qq^*/(gA^2)$ adds to the negativeness of the numerator of the ODE on the right side of X_s, and to the positiveness of the numerator of the ODE on the left side of X_s, that the depth Y_s will be larger than Y_l on the left side of the gutter and Y_r on the right side of the gutter. The ODE that applies through the grate length is $dY/dx = \{S_o - S_f + Qq_o^*/(gA^2)\}/(1 - F_r^2)$. Where Q is positive within the grate length, the term $Qq_o^*/(gA^2)$ in the numerator of the ODE adds to its positiveness, thus making Y_m larger than Y_r, generally. Likewise, where Q is negative from the position $L_G - X_m$ in the grate to the position L_G, this term adds to the negativeness of the numerator of the ODE that tends to make Y_m larger than Y_l. With a slope S_o greater than zero, one would expect X_m to be larger than $L_G/2$, and therefore the length $L_G - X_m$ will generally be quite small in consideration that the length of the grate will generally be much smaller than the length of the gutter. Therefore, generally there will not be as much difference between Y_l and Y_r (and Y_m) as there will be if the depths are critical at both ends of the gutter. Now Y_l may be larger than Y_r, and one would also expect the position X_s in the gutter that separates the positive from the negative Q's to be small even for relatively small slopes S_o. In fact, the effects of the increased depth, since larger depths are required for the accumulated inflow to exit through the grates, can easily result in no length of negative Q through the gutter, and this will also result in no length of negative Q through the grate.

The techniques described previously will be used to solve the above four equations simultaneously, and while F_3 is being solved, the lateral outflow will be determined by numerically integrating $\int q_o^* dx$ using, for example,the trapezoidal rule.

Program GUTTER4T is designed to solve problems that are of this case in which the subcritical flow occurs throughout both the gutter and the grate and there is a negative flow through some length of these regions. As with the previous case, the gutter is assumed to be triangular in shape in this program. In computing the slope of the energy line S_f in solving the ODE through the grate length, this same triangular shape is used in the program with the same n as for the gutter. However, in computing the outflow q_o^*, a bottom width of b (which is given in the input) is used as the bottom width of the channel. (In all likelihood, n will be different for the grate, but without experimental data the best value to use is unknown.)

The variables read by the first line are essentially as defined previously, with DX giving the Δx that will be used in writing the solution of the ODEs within the gutter length, DXG giving the Δx that will be used in writing the solution within the grate length, as well as the Δx used by the trapezoidal rule in the numerical integration, b is the width used in the orifice outflow equation, frac is the fraction of this width open, and C_d is the discharge coefficient. The subroutine DYX supplies values for dY/dx to the ODE solver ODESOL within the gutter length, and the subroutine DYXG supplies dY/dx within the grate length in solving the ODE. You will notice in this program, that subroutine FUN is called one last time after the Newton method has converged after the logical variable DONE is set to .TRUE.. This last call results in the profiles across both the gutter and grate lengths being written to the output file.

Program GUTTER4T.FOR

```
C Solves the problem of side flow into a triangular gutter, and
C the outflow from grates in its bottom.
C Yl=X(1), Yr=X(2), Ys=X(3), Xs=X(4)
      INTEGER*2 INDX(4)
      LOGICAL DONE
      REAL F(4),D(4,4),FFF(4)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(4),SO,FM,FMS,QS,CC,FN,HM,G,G2,G8,FL,DX,DXG,TOL,
```

```
     &FLG,DXGH,CD,SQSTAR,QS1,XX,DONE,IOUT
      WRITE(*,*)' Give:IO,TOL,ERR,n,So,m,L,Lgrate,g,DX,DXG,q*,b,fra
     &c,Cd'
      READ(*,*) IOUT,TOL,ERR,FN,SO,FM,FL,FLG,G,DX,DXG,QS,B,FRAC,CD
      DONE=.FALSE.
      DXGH=DXG/2.
      HM=.5*FM
      FMS=1.+SQRT(FM*FM+1.)
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      CD=CD*SQRT(G2)*FRAC*B
      G8=.125*G
      WRITE(*,*)' GIVE guess for:Yl,Yr,Ys & Xs'
      READ(*,*) X
      NCT=0
8     CALL FUN(F)
      WRITE(*,*) NCT,F
      DO 10 J=1,4
      DXX=.005*X(J)
      X(J)=X(J)+DXX
      CALL FUN(FFF)
      DO 9 I=1,4
9     D(I,J)=(FFF(I)-F(I))/DXX
10    X(J)=X(J)-DXX
      CALL SOLVEQ(4,1,4,D,F,1,DD,INDX)
      NCT=NCT+1
      SUM=0.
      DO 20 I=1,4
      X(I)=X(I)-F(I)
20    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,X
110   FORMAT(' NCT=',I3,' SUM=',E12.6,/,6F10.3)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 8
      WRITE(IOUT,100) X
100   FORMAT(' Yl=',F7.3,', Yr=',F7.3,', Ys=',F7.3,', Xs=',F8.2)
      DONE=.TRUE.
      CALL FUN(F)
      END
      SUBROUTINE FUN(F)
      LOGICAL DONE
      EXTERNAL DYX,DYXG
      REAL WW(1,13),Y(1),DY(1),XP(1),YP(1,1),F(4)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(4),SO,FM,FMS,QS,CC,FN,HM,G,G2,G8,FL,DX,DXG,TOL,
     &FLG,DXGH,CD,SQSTAR,QS1,XX,DONE,IOUT
      HMIN=1.E-6
```

```
      IF(DONE) THEN
      A=HM*X(1)**2
      Q=QS*X(4)
      WRITE(IOUT,100) 0.,X(1),Q,X(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*X(1)**5))
      ENDIF
100   FORMAT(/,'  x  Y  Q  E  A  Fr',/,1X,59('-'),/,F10.1,5F10.3)
      H1=.1
      XX=0.
      Y(1)=X(1)
10    XZ=XX+DX
      IF(XZ.GT.X(4)) XZ=X(4)
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS*(X(4)-XZ)
      WRITE(IOUT,110) XZ,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
110   FORMAT(F10.1,5F10.3)
      ENDIF
      XX=XZ
      IF(XZ.LT. X(4)) GO TO 10
      F(1)=X(3)-Y(1)
      H1=-.1
      IF(DONE) THEN
      A=HM*X(2)**2
      Q=QS*(FL-X(4))
      WRITE(IOUT,100) FL,X(2),Q,X(2)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*X(2)**5))
      ENDIF
      XX=FL
      Y(1)=X(2)
20    XZ=XX-DX
      IF(XZ.LT.X(4)) XZ=X(4)
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS*(XZ-X(4))
      WRITE(IOUT,110) XZ,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
      ENDIF
      XX=XZ
      IF(XZ.GT. X(4)) GO TO 20
      F(2)=X(3)-Y(1)
      H1=.005
      XX=0.
      SQSTAR=0.
      QS1=CD*SQRT(X(2))
      IF(DONE) THEN
      A=HM*X(2)**2
```

```
      Q=QS*(FL-X(4))
      WRITE(IOUT,120) 0.,X(2),Q,QS1,X(2)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*X(2)**5))
120   FORMAT(/,'  x  Y  Q  q*  E  A  Fr',/,1X,69('-'),/,F10.1,6F10.3)
      ENDIF
      Y(1)=X(2)
30    XZ=XX+DXG
      IF(XZ.GT.FLG) XZ=FLG
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYXG)
      QS2=CD*SQRT(Y(1))
      SQSTAR=SQSTAR+DXGH*(QS1+QS2)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS*(FL-X(4))-SQSTAR
      WRITE(IOUT,130) XZ,Y,Q,QS2,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
130   FORMAT(F10.3,6F10.3)
      ENDIF
      QS1=QS2
      XX=XZ
      IF(XZ.LT.FLG) GO TO 30
      F(3)=X(1)-Y(1)
      F(4)=FL*QS-SQSTAR
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      LOGICAL DONE
      REAL Y(1),DY(1)
      COMMON /TRAS/X(4),SO,FM,FMS,QS,CC,FN,HM,G,G2,G8,FL,DX,DXG,TOL,
     &FLG,DXGH,CD,SQSTAR,QS1,XDUM,DONE,IOUT
      YY=ABS(Y(1))
      A=HM*YY**2
      Q=QS*(XX-X(4))
      SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
      SF=SF*ABS(SF)
      FR2=Q*Q*FM*YY/(G*A**3)
      DY(1)=(SO-SF-2.*Q*QS/(G*A**2))/(1.-FR2)
      RETURN
      END
      SUBROUTINE DYXG(XP,Y,DY)
      LOGICAL DONE
      REAL Y(1),DY(1)
      COMMON /TRAS/X(4),SO,FM,FMS,QS,CC,FN,HM,G,G2,G8,FL,DX,DXG,TOL,
     &FLG,DXGH,CD,SQSTAR,QS1,XX,DONE,IOUT
      YY=Y(1)
      QS2=CD*SQRT(YY)
      A=HM*YY**2
      Q=QS*(FL-X(4))-SQSTAR-.5*(XP-XX)*(QS1+QS2)
      SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
      SF=SF*ABS(SF)
```

```
FR2=Q*Q*FM*YY/(G*A**3)
DY(1)=(SO-SF+Q*QS2/(G*A**2))/(1.-FR2)
RETURN
END
```

EXAMPLE PROBLEM 4.57

Investigate the effects on the depths Y_1, Y_r, and Y_s as well as the position X_s in the triangular gutter of Example Problem 4.52 with changes in the bottom slope. This channel has a side slope m = 4 and n = 0.013, and the gutter is L = 800 ft. The grate is 1 ft long and the lateral inflow over the gutter length is q* = 0.011 cfs/ft. Start by assuming that the gutter and the grate are horizontal. Then solve the problem with S_o = 0.00005 and continue to increase S_o until there is no length of negative flow within the gutter.

Solution

The input provided to Program GUTTER4T for six different solutions with different values for S_o are given below.

```
Input #1:  3 1.E−5 .001 .013 0.      4 800 1 32.2 10 .05 .011 4
.5 .45
1.5 1.5 1.6 400
Input #2:  3 1.E−5 .001 .013 0.00005 4 800 1 32.2 10 .05 .011 4
.5 .45
1.6 1.4 1.62 300
Input #3:  3 1.E−5 .001 .013 0.0001  4 800 1 32.2 10 .05 .011 4
.5 .45
1.5 1.5 1.5 300
Input #4:  3 1.E−5 .001 .013 0.0002  4 800 1 32.2 10 .05 .011 4
.5 .45
1.5 1.5 1.55 200
Input #5:  3 1.E−5 .001 .013 0.00025 4 800 1 32.2 10 .05 .011 4
.5 .45
1.5 1.45 1.52 30
Input #6:  3 1.E−5 .001 .013 0.00026 4 800 1 32.2 10 .05 .011 4
.5 .45
1.51 1.44 1.51 2
```

The table below summarizes the solutions of the four unknown variables.

No.	S_o	Y_1 (ft)	Y_r (ft)	Y_s (ft)	X_s (ft)	
1	0.0	1.474	1.474	1.525	400.00	
2	0.00005	1.481	1.466	1.525	311.29	
3	0.0001	1.488	1.459	1.524	225.90	<– – Not Valid since $X_s < 0$
4	0.0002	1.501	1.444	1.516	72.24	
5	0.00025	1.506	1.436	1.507	4.63	
6	0.00026	1.507	1.435	1.505	−8.19	

Notice that when $S_o = 0$, the flow divides in the middle of the gutter at X_s = 400 ft, with one-half of the inflow moving upstream and entering the grate at its left end, but that as S_o increases, even though still a very mild slope, X_s rapidly decreases to zero. Of course, solution # 6 is not valid in which S_o = 0.00026, because to satisfy the above four equations X_s is required to be negative. The solutions for these six different bottom slopes are given below with the majority of the output tables deleted.

Solution # 1 ($S_o = 0.0$)

$Y_l = 1.474$, $Y_r = 1.474$, $Y_s = 1.525$, $X_s = 400.0$.

x	Y	Q	E	A	F_r	
0.0	1.474	4.400	1.490	4.345	0.208	
10.0	1.477	4.290	1.492	4.364	0.202	
.	.	.	.	.	.	
400.0	1.525	.000	1.525	4.650	0.000	
800.0	1.474	4.400	1.490	4.345	0.208	
790.0	1.474	4.290	1.492	4.364	0.202	
.	.	.	.	.	.	
420.0	1.525	0.220	1.525	4.649	0.010	
410.0	1.525	0.110	1.525	4.650	0.005	
400.0	1.525	0.000	1.525	4.650	0.000	
x	**Y**	**Q**	**q***	**E**	**A**	**F_r**
0.0	1.474	4.400	8.768	1.490	4.345	0.208
0.050	1.477	3.961	8.778	1.490	4.363	0.186
0.100	1.480	3.522	8.786	1.490	4.379	0.165
.	.	.	.	.	.	.
0.450	1.490	0.441	8.815	1.490	4.438	0.020
0.500	1.490	0.000	8.816	1.490	4.439	0.000
0.550	1.490	−0.441	8.815	1.490	4.438	−0.020
.	.	.	.	.	.	.
0.950	1.477	−3.961	8.778	1.490	4.363	−0.186
1.000	1.474	−4.400	8.768	1.490	4.345	−0.208

Solution # 2 ($S_o = 0.00005$)

$Y_l = 1.481$, $Y_r = 1.466$, $Y_s = 1.525$, $X_s = 311.29$.

x	Y	Q	E	A	F_r	
0.0	1.481	3.424	1.491	4.388	0.160	
10.0	1.484	3.314	1.493	4.404	0.154	
.	.	.	.	.	.	
310.0	1.525	0.014	1.525	4.649	0.001	
311.3	1.525	0.000	1.525	4.649	0.000	
800.0	1.466	5.376	1.491	4.300	0.257	
790.0	1.471	5.266	1.494	4.325	0.250	
320.0	1.525	0.096	1.525	4.652	0.004	
311.3	1.525	0.000	1.525	4.649	0.000	
x	**Y**	**Q**	**q***	**E**	**A**	**F_r**
0.0	1.466	5.376	8.746	1.491	4.300	0.257
0.050	1.470	4.938	8.758	1.491	4.324	0.235
.	.	.	.	.	.	.
0.950	1.484	−2.985	8.797	1.491	4.402	−0.139
1.000	1.481	−3.424	8.790	1.491	4.388	−0.160

Solution # 3 & # 4

.	.	.	.	.	.	.

Solution # 5 (S_o = 0.00025)

Y_1 = 1.506, Y_r = 1.436, Y_s = 1.507, X_s = 4.63.

x	Y	Q	E	A	F_r
0.0	1.506	0.051	1.506	4.537	0.002
4.6	1.507	0.000	1.507	4.544	0.000
.	.	.	.	.	.
800.0	1.436	8.749	1.506	4.126	0.441
790.0	1.446	8.639	1.513	4.184	0.428
.	.	.	.	.	.
10.0	1.509	0.059	1.509	4.552	0.003
4.6	1.507	0.000	1.507	4.544	0.000

x	Y	Q	q*	E	A	F_r
0.0	1.436	8.749	8.656	1.506	4.126	0.441
0.050	1.444	8.316	8.680	1.506	4.173	0.413
0·	.	.	.	.	.	.
0.950	1.506	0.392	8.864	1.506	4.536	0.018
1.000	1.506	−0.051	8.864	1.506	4.537	−0.002

EXAMPLE PROBLEM 4.58

For the same gutter–grate as in the previous problem investigate the effects on Y_1, Y_r, Y_s, and X_s as the length of the grate is changed. For this series of solutions use a bottom slope S_o = 0.0002.

Solution

Program GUTTER4T has been modified to ask for a ΔL_G after completing a solution, and then solving the problem for a new grate length $L_G = (L_G)_{old} + \Delta L_G$. In the tables below, the results of these solutions are provided in which the length of the grate was increased and then decreased.

Solution of ODE in grate length using a Δx = 0.05 ft

L_G (ft)	Y_1 (ft)	Y_r (ft)	Y_s (ft)	X_s (ft)	Q_x (cfs)	$(F_r)_r$	
1.0000000	1.501	1.444	1.516	72.24	8.005	0.3983	
1.0500000	1.358	1.298	1.404	174.82	6.877	0.4465	
1.1000000	1.233	1.166	1.316	243.86	6.117	0.5190	
1.1500000	1.120	1.039	1.251	287.86	5.633	0.6375	
1.1500001	1.031	0.865	1.211	311.43	5.374	0.9624	<– – failed to converge

Solution of ODE in grate length using a Δx = 0.05 ft

L_G (ft)	Y_1 (ft)	Y_r (ft)	Y_s (ft)	X_s (ft)	Q_x (cfs)	$(F_r)_r$	
1.0000000	1.501	1.444	1.516	72.24	8.005	0.3983	
0.9900000	1.501	1.444	1.516	72.05	8.007	0.3984	
0.9800000	1.501	1.444	1.516	71.88	8.009	0.3985	
0.9700000	1.501	1.444	1.516	71.73	8.011	0.3987	
0.9600000	1.501	1.444	1.516	71.61	8.012	0.3987	
0.9550000	1.501	1.444	1.516	71.56	8.013	0.3987	
0.9500000	1.663	1.610	1.650	−69.06	9.560	0.3620	Invalid

Solution of ODE in grate length using a $\Delta x = 0.01$ ft

L_G (ft)	Y_1 (ft)	Y_r (ft)	Y_s (ft)	X_s (ft)	Q_x (cfs)	$(F_r)_r$	
1.0000000	1.471	1.413	1.492	95.50	7.749	0.4068	
0.9900000	1.501	1.444	1.516	72.07	8.007	0.3984	
0.9800000	1.531	1.475	1.541	47.09	8.282	0.3905	
0.9700000	1.563	1.508	1.567	20.55	8.574	0.3829	
0.9600000	1.595	1.541	1.594	−7.57	8.883	0.3756	Invalid

When the length of the grate is increased, the Froude number associated with the right side of the gutter repeatedly bounces around unity when L_G becomes a very small amount larger than 1.15 ft. It appears that the combined effects of the larger inflow into the grate and the solutions of the ODEs cause this critical condition, in which the flow attempts to become supercritical at the beginning of the grate. On the other hand, when the length of the grate is decreased less than 0.97 ft, the position X_s wants to become negative, i.e., the no negative Q situation. It is interesting to note that there are significant differences in the Y's obtained in the second and third tables above. These differences are due to using a different interval $\Delta x = 0.05$ ft versus $\Delta x = 0.01$ ft in numerically evaluating the integral of the outflow from the grate. These results indicate that there is a very narrow range of grate lengths for this case to apply. Example Problem 4.57 indicated that the bottom slopes must be very mild, also.

EXAMPLE PROBLEM 4.59

For the same gutter–grate as in the previous problem, investigate the effects on Y_l, Y_r, Y_s, and X_s as the lateral inflow over the gutter is changed. For this series of solutions, use a bottom slope $S_o = 0.0002$ and a length of grate $L_G = 1$ ft.

Solution

Program GUTTER4T has been modified so that it requests a new lateral inflow for the next solution after it completes a solution for the past value of q*. The following two tables provide the results from these two series of solutions, first in which q* has been increased in small steps from 0.011 cfs/ft, and the second in which q* has been decreased from 0.011 cfs/ft. When q* becomes larger than 0.011387 cfs/ft, the position X_s moves too close to the beginning of the gutter and the solution fails, and when q* becomes smaller than 0.0088 cfs, the Froude number at the downstream end of the gutter $(F_r)_r$ becomes too close to unity and the solution fails. Notice again, there is a very small range of q* for this case to occur.

q* (cfs/ft)	Y_1 (ft)	Y_r (ft)	Y_s (ft)	X_s (ft)	Q_r (cfs)	$(F_r)_r$
0.011000	1.501	1.444	1.516	72.24	8.005	0.3983
0.011200	1.556	1.500	1.563	36.29	8.554	0.3869
0.011300	1.584	1.528	1.587	17.58	8.841	0.3816
0.011350	1.598	1.543	1.600	8.06	8.989	0.3790
0.011350	1.601	1.545	1.602	6.14	9.018	0.3785
0.011370	1.604	1.548	1.604	4.22	9.048	0.3780
0.011375	1.605	1.550	1.606	3.25	9.063	0.3778
0.011376	1.605	1.550	1.606	3.06	9.066	0.3777
0.011377	1.606	1.550	1.606	2.87	9.069	0.3777
0.011378	1.606	1.551	1.606	2.68	9.072	0.3776
0.011378	1.606	1.551	1.607	2.58	9.073	0.3776
0.011379	1.606	1.551	1.607	2.48	9.075	0.3776
0.011379	1.606	1.551	1.607	2.38	9.076	0.3775
0.011380	1.606	1.551	1.607	2.29	9.078	0.3775
0.011381	1.607	1.551	1.607	2.09	9.081	0.3775
0.011382	1.607	1.551	1.607	1.90	9.084	0.3774

(*continued*)

(continued)

q* (cfs/ft)	Y_1 (ft)	Y_r (ft)	Y_s (ft)	X_s (ft)	Q_r (cfs)	$(F_r)_r$
0.011383	1.607	1.552	1.608	1.71	9.087	0.3774
0.011384	1.608	1.552	1.608	1.52	9.090	0.3773
0.011385	1.608	1.552	1.608	1.32	9.093	0.3773
0.011386	1.608	1.553	1.608	1.13	9.096	0.3772
0.011387	1.608	1.553	1.609	0.94	9.099	0.3772

q* (cfs/ft)	Y_1 (ft)	Y_r (ft)	Y_s (ft)	X_s (ft)	Q_r (cfs)	$(F_r)_r$
0.011000	1.501	1.444	1.516	72.24	8.005	0.3983
0.010800	1.446	1.388	1.470	106.07	7.494	0.4111
0.010600	1.392	1.334	1.425	137.60	7.021	0.4257
0.010400	1.339	1.280	1.382	166.66	6.587	0.4427
0.010200	1.287	1.227	1.340	193.16	6.190	0.4628
0.10000	1.235	1.174	1.301	217.04	5.830	0.4867
0.009000	0.988	0.896	1.139	299.38	4.506	0.7396
0.008900	0.965	0.860	1.126	304.65	4.409	0.8012
0.008800	0.942	0.812	1.115	309.40	4.317	0.9043
0.008700	Failed to Converge; F_r too close to 1.					

4.20.4 Lateral Inflow over Grate Length

In handling the combined subcritical flow problem of lateral inflow over a length L of gutter, and outflow over a length L_G of grate, it has been assumed that no lateral inflow occurs within the grate's length. This may not be true for some applications, even though in most cases, L_G is small enough in comparison to L that the inflow can be ignored. If this lateral inflow needs to be accounted for, one might be inclined to subtract it from the computed lateral outflow $\{(q_o^*)_{mod} = (q_o^*)_{comp} - q^*\}$ and use this modified outflow in solving the ODE over the grate's length. This approach would not be fundamentally correct, however, since the inflow term in the ODE has a 2 that multiplies it besides being of opposite sign, i.e., the lateral inflow term in the numerator of the ODE is $-2q_o^*Q/(gA^2)$ and the outflow term is $q^*Q/(gA^2)$. This difference is due to the fact that the energy per unit weight of the lateral outflow is the same as that of the fluid that continues in the channel, whereas the inflow has no kinetic energy per unit weight in the direction of the main channel; thus the energy principle can be used to derive the ODE for the outflow, but the momentum principle must be used for the inflow case.

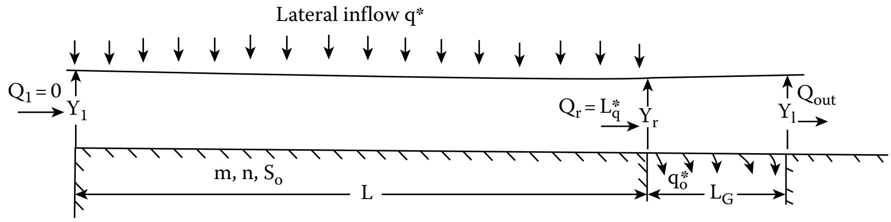
Thus, to account for the lateral inflow over the length of the grate, the ODE must include both the lateral inflow and the outflow terms, or

$$\frac{dY}{dx} = \frac{S_o - S_f - 2q^*Q/(gA^2) + q_o^*Q/(gA^2)}{1 - F_r^2}$$

See a homework problem for the implementation of the inflow over the grate's length.

4.20.5 No Negative Flow Rates

When the combination of variables for a gutter–grate problem result in a no reverse flow at the beginning of the gutter, then the mathematical problem simplifies because the position X_s that separates the positive from the negative flows is known; $X_s = 0$ and Y_s does not exist. However, unless other variables are just the right magnitude, there will be a flow in the channel at the end of the grate, i.e., the grate will not discharge all of the flow that accumulates over the inflow length of the gutter. While there are other ways of posing the problem, let us consider that the unknowns are: (1) the depth Y_l on the left side of the gutter, which we will also take as the depth at the end of the grate; (2) the depth Y_r on the right side of the gutter, which is also the depth at the beginning of the grate; and (3) the flow rate Q_{out} that leaves the end of the grate to add to the flow in the next series of gutter grates, as shown in the sketch below.



The three equations needed to solve these three unknowns are (1) the ODE across the gutter, (2) the ODE across grate, and (3) the continuity equation that indicates that the accumulated inflow through the gutter minus the flow rate Q_{out} that leaves the end of the grate minus the outflow from the grate is equal to zero. These three equations are

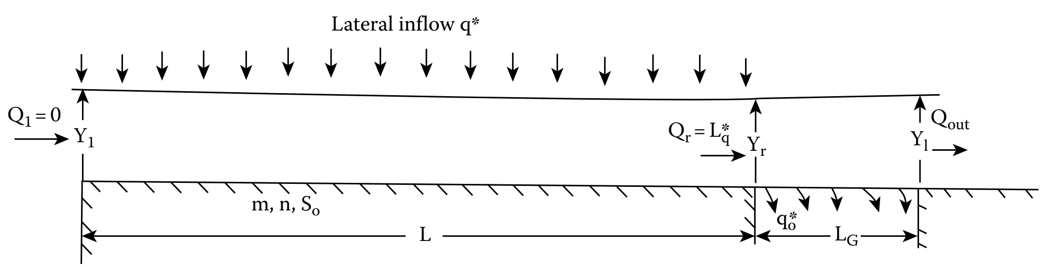
$$F_1 = Y_r - Y_{rode}(Y_l) = 0 \quad \text{with the ODE solved over } x = 0 \text{ to } x = L \tag{4.36a}$$

$$F_2 = Y_l - Y_{lode}(Y_r) = 0 \quad \text{with the ODE solved over } x' = 0 \text{ to } x' = L_G \tag{4.36b}$$

$$F_3 = q^*L - \int_0^{L_G} q_o^*\,dx' - Q_{out} = 0 \tag{4.36c}$$

The integral in F_3 is numerically evaluated while solving the ODE for F_2.

Program GUTTER3I, listed below, is designed to solve problems in triangular gutters that fall within the category of this case in which there is no reverse flow over the first part of the gutter, i.e., $Q_l = Q_{in} = 0$. This program is written so that an inflow Q_{in} at the left side, or at the beginning of the gutter, is allowed. By giving Q_{in} a value of zero, the case is handled in which there is no reverse flow, or $X_s = 0$. The first line of input is the same as in the previous program GUTTER4T. The second line of input provides guesses for the three variables being solved by the Newton method Y_l, Y_r, and Q_{out}. For problems that fall within this category, as well as those in the previous category of combined gutter–grate flow with negative as well as positive flows, it is necessary to provide very good guesses for the unknowns, or the converge to the solution will fail.



Program GUTTER3I.FOR (Also see GUTTER3I.C on diskette)

```
C Solves the problem of side flow into a triangular gutter, and
C the outflow from grates in its bottom. The case being solved is
C when Q =Qin at beginning of the gutter, i.e. all the lateral
C inflow moves down the gutter channel, with no reverse flow.
C However not all of the accumulated lateral inflow plus Qin is
C require to discharge through the grate; some amount Qout may
C flow beyond into the next gutter so that Yl (at begin. of
C gutter) equals Yl (at end of grate.)
C Yl=X(1), Yr=X(2) Qout=X(3)
      INTEGER*2 INDX(3)
      LOGICAL DONE
      REAL F(3),D(3,3),FFF(3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(3),SO,FM,FMS,QS,QT,CC,FN,HM,G,G2,G8,FL,DX,
     &DXG,TOL,&FLG,DXGH,CD,SQSTAR,QS1,Qin,XX,DONE,IOUT
      WRITE(*,*)' Give:IO,TOL,ERR,n,So,m,L,Lgrate,g,DX,DXG,Qin,q*,
     &b,frac,Cd'
      READ(*,*) IOUT,TOL,ERR,FN,SO,FM,FL,FLG,G,DX,DXG,Qin,QS,B,
     &FRAC,CD
      QT=QS*FL
      DONE=.FALSE.
      DXGH=DXG/2.
      HM=.5*FM
      FMS=1.+SQRT(FM*FM+1.)
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      CD=CD*SQRT(G2)*FRAC*B
      G8=.125*G
      WRITE(*,*)' GIVE guess for:Yl, Yr &',' Qout'
      READ(*,*) X
      NCT=0
8     CALL FUN(F)
      WRITE(*,*) NCT,F
      DO 10 J=1,3
      DXX=.005*X(J)
      X(J)=X(J)+DXX
      CALL FUN(FFF)
      DO 9 I=1,3
9     D(I,J)=(FFF(I)-F(I))/DXX
10    X(J)=X(J)-DXX
      CALL SOLVEQ(3,1,3,D,F,1,DD,INDX)
      NCT=NCT+1
      SUM=0.
      DO 20 I=1,3
```

```
      X(I)=X(I)-F(I)
20    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,X
110   FORMAT(' NCT=',I3,' SUM=',E12.6,/,6F10.3)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 8
      WRITE(IOUT,100) X
100   FORMAT(' Yl=',F7.3,', Yr=',F7.3,', Qout=',F8.3)
      DONE=.TRUE.
      CALL FUN(F)
      END
      SUBROUTINE FUN(F)
      LOGICAL DONE
      EXTERNAL DYX,DYXG
      REAL WW(1,13),Y(1),DY(1),XP(1),YP(1,1),F(3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(3),SO,FM,FMS,QS,QT,CC,FN,HM,G,G2,G8,FL,DX,
     &DXG,TOL,FLG,DXGH,CD,SQSTAR,QS1,Qin,XX,DONE,IOUT
      HMIN=1.E-6
      IF(DONE) THEN
      A=HM*X(1)**2
      WRITE(IOUT,100) 0.,X(1),Qin,X(1)+(Qin/A)**2/G2,A,Qin/(FM*SQRT
     &(G8*X(1)**5))
      ENDIF
100   FORMAT(/,'  x   Y   Q   E   A   Fr',/,1X,59('-'),/,F10.1,5F10.3)
      H1=.1
      XX=0.
      Y(1)=X(1)
10    XZ=XX+DX
      IF(XZ.GT.FL) XZ=FL
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS*XZ+Qin
      WRITE(IOUT,110) XZ,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
110   FORMAT(F10.1,5F10.3)
      ENDIF
      XX=XZ
      IF(XZ.LT.FL) GO TO 10
      F(1)=X(2)-Y(1)
      H1=.005
      XX=0.
      SQSTAR=0.
      QS1=CD*SQRT(X(2))
      IF(DONE) THEN
      A=HM*X(2)**2
      Q=QT+Qin
      WRITE(IOUT,120) 0.,X(2),Q,QS1,X(2)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*X(2)**5))
```

```
120   FORMAT(/,'  x  Y  Q  q*  E  A  Fr',/,1X,69('-'),/,F10.1,6F10.3)
      ENDIF
30    XZ=XX+DXG
      IF(XZ.GT.FLG) XZ=FLG
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYXG)
      QS2=CD*SQRT(Y(1))
      SQSTAR=SQSTAR+DXGH*(QS1+QS2)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QT-SQSTAR+Qin
      WRITE(IOUT,130) XZ,Y,Q,QS2,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
130   FORMAT(F10.3,6F10.3)
      ENDIF
      QS1=QS2
      XX=XZ
      IF(XZ.LT.FLG) GO TO 30
      F(2)=X(1)-Y(1)
      F(3)=QT-X(3)-SQSTAR+Qin
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      LOGICAL DONE
      REAL Y(1),DY(1)
      COMMON /TRAS/X(3),SO,FM,FMS,QS,QT,CC,FN,HM,G,G2,G8,FL,DX,
     &DXG,TOL,FLG,DXGH,CD,SQSTAR,QS1,Qin,XDUM,DONE,IOUT
      YY=ABS(Y(1))
      A=HM*YY**2
      Q=QS*XX+Qin
      SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
      SF=SF*ABS(SF)
      FR2=Q*Q*FM*YY/(G*A**3)
      DY(1)=(SO-SF-2.*Q*QS/(G*A**2))/(1.-FR2)
      RETURN
      END
      SUBROUTINE DYXG(XP,Y,DY)
      LOGICAL DONE
      REAL Y(1),DY(1)
      COMMON /TRAS/X(3),SO,FM,FMS,QS,QT,CC,FN,HM,G,G2,G8,FL,DX,
     &DXG,TOL,FLG,DXGH,CD,SQSTAR,QS1,Qin,XX,DONE,IOUT
      YY=Y(1)
      QS2=CD*SQRT(YY)
      A=HM*YY**2
      Q=QT-SQSTAR-.5*(XP-XX)*(QS1+QS2)+Qin
      SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
      SF=SF*ABS(SF)
      FR2=Q*Q*FM*YY/(G*A**3)
      DY(1)=(SO-SF+Q*QS2/(G*A**2))/(1.-FR2)
      RETURN
      END
```

EXAMPLE PROBLEM 4.60

From the channel of the previous example problems, solve the depths when the bottom slope is $S_o = 0.00026$. ($m = 4$, $n = 0.013$, $L = 800\,ft$, $q^* = 0.011\,cfs/ft$, $L_G = 1\,ft$, $b = 4\,ft$, $f = 0.5$, and $C_d = 0.45$). After this, obtain a series of solutions in which the channel's bottom slope is increased in small increments to investigate how the depths Y_1 and Y_r, and the outflow Q_{out} vary with S_o.

Solution

The input to program GUTTER3T to solve this problem consists of

3 1.e−5 .001 .013 .00026 4 800 1 32.2 10 .05 0 .011 4 .5 .45
1.5 1.4 .2

The output, with many of the lines deleted, consists of

$Y_1 = 1.498$, $Y_r = 1.425$, $Q_{out} = .027$

x	Y	Q	E	A	F_r	
0.0	1.498	.000	1.498	4.490	0.000	
10.0	1.501	.110	1.501	4.505	0.005	
.	.	.	.	.	.	
780.0	1.446	8.580	1.511	4.181	0.425	
790.0	1.436	8.690	1.505	4.125	0.438	
800.0	1.425	8.800	1.498	4.064	0.452	
x	**Y**	**Q**	**q***	**E**	**A**	**F_r**
0.0	1.425	8.800	8.623	1.498	4.064	0.452
0.050	1.434	8.368	8.649	1.498	4.112	0.424
.	.	.	.	.	.	
0.950	1.498	0.469	8.840	1.498	4.489	0.021
1.000	1.498	0.027	8.841	1.498	4.490	0.001

With program GUTTER3T, modified slightly so that it requests if another solution with a different bottom slope is desired, the series of solutions given in the following table have been obtained. Note that as the bottom slope increases, that (1) the flow rate Q_{out} past the grate increases, (2) the depths both at the left and the right sides of the gutter increases, and (3) this results in increases of the Froude numbers. For this example problem when $S_o = 0.0005$, the Froude number $(F_r)_r$ on the right side of the gutter (which is at the upstream end of the grate), equals 0.735. When solutions are sought for substantially larger values of S_o, the iterative process fails because critical depths occur. This example problem, as well as the previous ones, indicate that if grates are spaces at a given interval along a gutter, and they are to discharge most, if not all, the lateral inflow that has accumulated in the gutter, that only for very flat gutters will the flow be subcritical across the grates. A more likely situation is that the depth will be critical at the position where the grate begins, especially if the bottom slope is very large.

Solution fails with $S_o = 0.00046$.

S_o	Y_1	Y_r	Q_{out}	$\int q_o^* dx$	$(F_r)_r$	$(F_r)_1$
0.0002600	1.498	1.425	0.027	8.773	0.451	0.399
0.0003000	1.451	1.364	0.181	8.619	0.494	0.432
0.0003500	1.400	1.293	0.355	8.445	0.554	0.473
0.0003750	1.377	1.257	0.435	8.365	0.588	0.492

(*continued*)

(continued)

S_o	Y_1	Y_r	Q_{out}	$\int q_o^* dx$	$(F_r)_r$	$(F_r)_1$
0.0004000	1.356	1.222	0.513	8.287	0.626	0.512
0.0004100	1.348	1.207	0.544	8.256	0.643	0.520
0.0004200	1.340	1.191	0.574	8.226	0.662	0.527
0.0004300	1.332	1.175	0.604	8.196	0.682	0.535
0.0004400	1.325	1.158	0.634	8.166	0.705	0.543
0.0004500	1.317	1.140	0.665	8.135	0.731	0.551

EXAMPLE PROBLEM 4.61

Repeat the series of solutions requested in the previous example problem but vary the lateral inflow q* in the gutter, first decreasing it by small increments from q* = 0.011 cfs/ft, and then increasing it. For this series of solutions use a bottom slope S_o = 0.00026 (m = 4, n = 0.013, L = 800 ft, L_G = 1 ft, b = 4 ft, f = 0.5 and C_d = 0.45).

Solution

Using a version of GUTTER3T, modified to obtain a series of solutions with different values of q*, the two tables given below were obtained. Note that when q* is decreased, the Froude number at the beginning of the grate increases and soon a critical flow occurs. When q* is increased, then the flow rate at the end of the grate Q_{out} becomes negative indicating that the previous case occurs in which Q is negative in the first portion of the gutter.

Solution failed with q* = 0.0084.

q*	Y_1	Y_r	Q_{out}	$\int q_o^* dx$	$(F_r)_r$	$(F_r)_1$
0.0110000	1.498	1.425	0.027	8.773	0.451	0.399
0.0010500	1.463	1.381	0.140	8.660	0.482	0.423
0.0100000	1.427	1.332	0.261	8.539	0.520	0.451
0.0095000	1.390	1.277	0.392	8.408	0.569	0.482
0.0090000	1.350	1.210	0.538	8.262	0.639	0.518
0.0089000	1.341	1.194	0.569	8.231	0.658	0.526
0.0088000	1.333	1.177	0.602	8.198	0.680	0.535
0.0087000	1.324	1.157	0.637	8.163	0.707	0.544
0.0086000	1.315	1.133	0.675	8.125	0.740	0.553
0.0085000	1.305	1.101	0.717	8.083	0.791	0.564
0.0110000	1.498	1.425	0.027	8.773	0.451	0.399
0.0120000	1.565	1.506	−0.181	8.981	0.402	0.358
0.0130000	1.627	1.579	−0.370	9.170	0.365	0.325
0.0140000	1.686	1.645	−0.545	9.345	0.335	0.297
0.0150000	1.743	1.708	−0.707	9.507	0.311	0.273
0.0160000	1.797	1.767	−0.859	9.659	0.290	0.253
0.0170000	1.850	1.823	−1.002	9.802	0.272	0.236
0.0180000	1.900	1.876	−1.138	9.938	0.257	0.220
0.0190000	1.949	1.927	−1.268	10.068	0.243	0.207
0.0200000	1.996	1.977	−1.391	10.191	0.231	0.195

EXAMPLE PROBLEM 4.62

Solve the previous gutter–grate system for the situation that will eventually develop a constant gutter–grate flow if 1 ft long grates are spaced after a long series of gutters each 800 ft long. Over this entire length of gutters assume that the lateral inflow is q* = 0.011 cfs/ft. As in previous example problems n = 0.013, S_o = 0.00026, f = 0.5, and C_d = 0.45. Resolve the problem but increase the bottom slope to S_o = 0.0006, and explain why the changes in $Q_{out} = Q_{in}$ increases.

Solution

This problem could be solved using program GUTTER3I by trial using different values for Q_{in} until it equals the computed Q_{out}. The program may be modified to automatically do this trial by assigning Q_{in} equal to the last computed Q_{out} and repeating a new solution until the change in Q_{out} between such consecutive solutions is within a tolerance requirement. The best approach is to modify program GUTTER3I (GUTTER3S) by removing Q_{in} from the READ and COMMON statements, and by replacing the Q_{in} in all arithmetic statements with X(3), which is the third unknown Q_{out}, being solved by the Newton method. Using this latter approach produces the output given below for S_o = 0.00026.

x	Y	Q	E	A	F_r	
0.0	1.507	0.090	1.507	4.544	0.004	
10.0	1.510	0.200	1.510	4.559	0.009	
20.0	1.512	0.310	1.512	4.574	0.014	
30.0	1.515	0.420	1.515	4.589	0.019	
.	.	.	.	.	.	
780.0	1.455	8.670	1.520	4.233	0.423	
790.0	1.445	8.780	1.514	4.178	0.436	
800.0	1.435	8.890	1.507	4.118	0.449	
x	**Y**	**Q**	**q***	**E**	**A**	**F_r**
0.0	1.435	8.890	8.651	1.507	4.118	0.449
0.050	1.443	8.457	8.677	1.507	4.166	0.421
0.100	1.451	8.023	8.699	1.507	4.210	0.394
0.950	1.507	0.533	8.866	1.507	4.542	0.024
1.000	1.507	0.090	8.867	1.507	4.544	0.004

The following table provides the values for Y_l, Y_r, and Q_{out} as S_o is gradually increased to 0.0006. Notice there is a small increase in the depth at the left side of the gutter Y_l, and a slight decrease in the depth at the right side of the gutter, or the beginning of the grate. The amount of flow Q_{out} passing on to the next series of gutters and grates, however, increases significantly. This is necessary since there must be an increase in frictional loss in the gutter associated with an increasing Y_l and a decreasing Y_r. The total outflow from the grate remains constant and equal to q*L = 0.011(800) = 8.8 cfs, and therefore an average depth over the grate must be maintained; if Y_r increases, Y_l must decrease.

Y_l = 1.543, Y_r = 1.370, Q_{out} = 4.139.

S_o	Y_l	Y_r	Q_{out}
0.000260	1.507	1.435	0.090
0.000300	1.511	1.429	0.632
0.000350	1.517	1.421	1.271
0.000400	1.522	1.413	1.878
0.000450	1.527	1.404	2.463
0.000460	1.528	1.402	2.577
0.000470	1.529	1.401	2.691
0.000480	1.530	1.399	2.805
0.000490	1.531	1.397	2.918
0.000500	1.532	1.395	3.031

(*continued*)

(continued)

S_o	Y_l	Y_r	Q_{out}
0.000510	1.533	1.392	3.143
0.000520	1.534	1.390	3.255
0.000530	1.535	1.388	3.366
0.000540	1.536	1.386	3.477
0.000550	1.537	1.383	3.588
0.000560	1.538	1.381	3.698
0.000570	1.539	1.378	3.809
0.000580	1.540	1.376	3.919
0.000590	1.541	1.373	4.029
0.000600	1.543	1.370	4.139

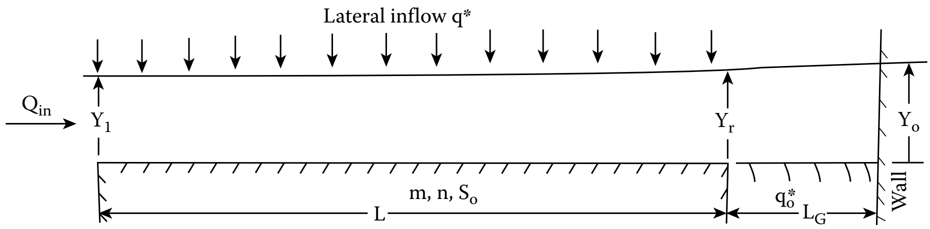
x	Y	Q	E	A	F_r	
0.0	1.543	4.139	1.554	4.759	0.175	
10.0	1.546	4.249	1.559	4.782	0.178	
20.0	1.550	4.359	1.563	4.804	0.182	
30.0	1.553	4.469	1.567	4.826	0.185	
.	.	.	.	.	.	
780.0	1.437	12.719	1.584	4.128	0.641	
790.0	1.408	12.829	1.571	3.966	0.679	
800.0	1.370	12.939	1.555	3.754	0.734	
x	**Y**	**Q**	**q***	**E**	**A**	**F_r**
0.0	1.370	12.939	8.453	1.554	3.752	0.734
0.050	1.393	12.514	8.524	1.554	3.881	0.681
0.100	1.412	12.087	8.581	1.554	3.985	0.636
.	.	.	.	.	.	.
0.900	1.537	5.035	8.953	1.554	4.723	0.214
0.950	1.540	4.587	8.962	1.554	4.742	0.194
1.000	1.543	4.138	8.971	1.554	4.760	0.174

4.20.6 Last Grate at End of Gutter

At the end of the gutter–grate system let us assume that a wall exists so that no flow passes the last grate. If the lateral inflow is larger than can be discharged by the preceding grates, then the depth will be increased over this last grate so that the sum of the inflow Q_{in} at the beginning of the last gutter plus the accumulated lateral inflow over its length will discharge through this grate. For this last gutter–grate system, the same three equations, Equations 4.36a through c are available as previously, with F_2 in Equation 4.36b modified so that rather than forcing the depth from the solution of the ODE to be the same as the depth at the beginning of the gutter Y_l, it equals the depth at the end of the grate Y_e, or Equation 4.36b becomes

$$F_2 = Y_e - Y_{eode}(Y_r) = 0 \quad \text{with ODE solved over } x' = 0 \text{ to } x' = L_G$$

Now, the outflow $Q_{out} = 0$ is known, and it will be replaced by the depth Y_e at the end of the grate as an unknown. The three unknowns for this end grate are: Y_l, the depth at the left side or at the beginning of the gutter, Y_r, the depth at the right side or at the end of the gutter, which will also be the depth at the beginning of the grate, and Y_e the depth at the end of the grate.



Program GUTGRAT1, whose listing is given below, is designed to solve problems in which no flow passes the grate, and a flow rate Q_{in} occurs in the channel at the beginning of the gutter, as well as the lateral inflow into the gutter must be discharged through the grate. The input to this program is identical to GUTTER3I with the exception that the second line now gives an estimate of Y_e rather than an estimate of Q_{out}. The other difference is that the above equation for F_2 (F(2) = X(3) − Y(1)) replaces Equation 4.36b (F(2) = X(1) − Y(1)).

Program GUTGRAT1.FOR (also see GUTGRAT1.C on diskette)

```
C Solves the problem of side flow into a triangular gutter, and
C the outflow from a grate at its end. A flow rate Q =Qin at the
C beginning of the gutter can also occur and all the lateral
C inflow must exit from the grate.
C Yl=X(1), Yr=X(2) Ye=X(3)
      INTEGER*2 INDX(3)
      LOGICAL DONE
      REAL F(3),D(3,3),FFF(3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(3),SO,FM,FMS,QS,QT,CC,FN,HM,G,G2,G8,FL,DX,
     &DXG,TOL,FLG,DXGH,CD,SQSTAR,QS1,Qin,XX,DONE,IOUT
      WRITE(*,*)' Give:IO,TOL,ERR,n,So,m,L,Lgrate,g,DX,DXG,Qin,q*,
     &b,frac,Cd'
      READ(*,*) IOUT,TOL,ERR,FN,SO,FM,FL,FLG,G,DX,DXG,Qin,QS,B,FRAC,CD
      IOU1=IOUT+1
      QT=QS*FL+Qin
      DONE=.FALSE.
      DXGH=DXG/2.
      HM=.5*FM
      FMS=1.+SQRT(FM*FM+1.)
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      CD=CD*SQRT(G2)*FRAC*B
      G8=.125*G
      WRITE(*,*)' GIVE guess for:Yl, Yr & Ye'
      READ(*,*) X
7     NCT=0
```

```
8      CALL FUN(F)
       WRITE(*,*) NCT,F
       DO 10 J=1,3
       DXX=.005*X(J)
       X(J)=X(J)+DXX
       CALL FUN(FFF)
       DO 9 I=1,3
9      D(I,J)=(FFF(I)-F(I))/DXX
10     X(J)=X(J)-DXX
       CALL SOLVEQ(3,1,3,D,F,1,DD,INDX)
       NCT=NCT+1
       SUM=0.
       DO 20 I=1,3
       X(I)=X(I)-F(I)
20     SUM=SUM+ABS(F(I))
       WRITE(*,110) NCT,SUM,X
110    FORMAT(' NCT=',I3,' SUM=',E12.6,/,6F10.3)
       IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 8
       WRITE(*,*)' Give:1=write profiles, 2=just Y''s, 3=both'
       READ(*,*) III
       Frr=SQRT(QT**2*FM*X(2)/(G*(HM*X(2)**2)**3))
       Frl=SQRT(Qin**2*FM*X(1)/(G*(HM*X(1)**2)**3))
       IF(III.GT.1) THEN
       WRITE(IOU1,151) X,Frl,Frr,Qin,QS
151    FORMAT(5F7.3,F8.3,F7.3)
       IF(III.EQ.2) GO TO 45
       ENDIF
       WRITE(IOUT,100) Qin,QS,X,Frl,Frr
100    FORMAT(' Qin=',F8.4,', q*=',F8.4,/' Yl=',F7.3,', Yr=',F7.3,',
     &Ye=',F7.3,', (Fr)l=',F7.3,', (Fr)r=',F7.3)
       DONE=.TRUE.
       CALL FUN(F)
       DONE=.FALSE.
45     WRITE(*,*)' Give 1 to solve another problem or 0 = STOP'
       READ(*,*) III
       IF(III.EQ.0) STOP
       WRITE(*,*)' Give new Qin & q*'
       READ(*,*) Qin,QS
       QT=QS*FL+Qin
       GO TO 7
       END
       SUBROUTINE FUN(F)
       LOGICAL DONE
       EXTERNAL DYX,DYXG
       REAL WW(1,13),Y(1),DY(1),XP(1),YP(1,1),F(3)
       COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
       COMMON /TRAS/X(3),SO,FM,FMS,QS,QT,CC,FN,HM,G,G2,G8,FL,DX,DXG,
     &TOL,FLG,DXGH,CD,SQSTAR,QS1,Qin,XX,DONE,IOUT
```

```
      HMIN=1.E-6
      IF(DONE) THEN
      A=HM*X(1)**2
      WRITE(IOUT,100) 0.,X(1),Qin,X(1)+(Qin/A)**2/G2,A,Qin/(FM*SQRT
     &(G8* X(1)**5))
      ENDIF
100   FORMAT(/,'  x   Y   Q   E   A   Fr',/,1X,59('-'),/,F10.1,5F10.3)
      H1=.1
      XX=0.
      Y(1)=X(1)
10    XZ=XX+DX
      IF(XZ.GT.FL) XZ=FL
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS*XZ+Qin
      WRITE(IOUT,110) XZ,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
110   FORMAT(F10.1,5F10.3)
      ENDIF
      XX=XZ
      IF(XZ.LT.FL) GO TO 10
      F(1)=X(2)-Y(1)
      H1=.005
      XX=0.
      SQSTAR=0.
      QS1=CD*SQRT(X(2))
      IF(DONE) THEN
      A=HM*X(2)**2
      WRITE(IOUT,120) 0.,X(2),QT,QS1,X(2)+(QT/A)**2/G2,A,QT/(FM*SQRT
     &(G8*X(2)**5))
120   FORMAT(/,'  x   Y   Q   q*   E   A   Fr',/,1X,69('-'),/,
     &F10.1,6F10.3)
      ENDIF
30    XZ=XX+DXG
      IF(XZ.GT.FLG) XZ=FLG
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYXG)
      QS2=CD*SQRT(Y(1))
      SQSTAR=SQSTAR+DXGH*(QS1+QS2)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QT-SQSTAR
      WRITE(IOUT,130) XZ,Y,Q,QS2,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
130   FORMAT(F10.3,6F10.3)
      ENDIF
      QS1=QS2
      XX=XZ
```

```
  IF(XZ.LT.FLG) GO TO 30
  F(2)=X(3)-Y(1)
  F(3)=QT-SQSTAR
  RETURN
  END
  SUBROUTINE DYX(XX,Y,DY)
  LOGICAL DONE
  REAL Y(1),DY(1)
  COMMON /TRAS/X(3),SO,FM,FMS,QS,QT,CC,FN,HM,G,G2,G8,FL,DX,DXG,
 &TOL,FLG,DXGH,CD,SQSTAR,QS1,Qin,XDUM,DONE,IOUT
  YY=ABS(Y(1))
  A=HM*YY**2
  Q=QS*XX+Qin
  SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
  SF=SF*ABS(SF)
  FR2=Q*Q*FM*YY/(G*A**3)
  DY(1)=(SO-SF-2.*Q*QS/(G*A**2))/(1.-FR2)
  RETURN
  END
  SUBROUTINE DYXG(XP,Y,DY)
  LOGICAL DONE
  REAL Y(1),DY(1)
  COMMON /TRAS/X(3),SO,FM,FMS,QS,QT,CC,FN,HM,G,G2,G8,FL,DX,DXG,
 &TOL,FLG,DXGH,CD,SQSTAR,QS1,Qin,XX,DONE,IOUT
  YY=Y(1)
  QS2=CD*SQRT(YY)
  A=HM*YY**2
  Q=QT-SQSTAR-.5*(XP-XX)*(QS1+QS2)
  SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
  SF=SF*ABS(SF)
  FR2=Q*Q*FM*YY/(G*A**3)
  DY(1)=(SO-SF+Q*QS2/(G*A**2))/(1.-FR2)
  RETURN
  END
```

EXAMPLE PROBLEM 4.63

Assume that the gutter–grate system used previously with L = 800 ft, L_G = 1 ft, m = 4, n = 0.013, S_o = 0.00026, f = 0.5, and C_d = 0.45 ends and all the flow must exit through this last grate. Obtain two series of solutions; the first in which the inflow to the gutter Q_{in} increases and the lateral inflow q* = 0.011 cfs/ft remains constant, and the second in which Q_{in} = 2 cfs remains constant and the latter inflow q* increases from .01 cfs/ft.

Solution

The input to program GUTGRAT1 consists of the following two lines for both parts followed by the different values of Q_{in} and q*:

```
3 1.e−5 .001 .013 .00026 4 800 1 32.2 10 .05 .011 4 .5 .45
1.5 1.5 1.55
```

The output (with most of the lines giving the profiles over the gutter and the grate deleted) consists of

Q_{in} = 0.0000, q* = 0.0110 Y_l = 1.502, Y_r = 1.436, Y_e = 1.507, $(F_r)_l$ = 0.000, $(F_r)_r$ = 0.444

x	Y	Q	E	A	F_r	
0.0	1.502	0.000	1.502	4.513	0.000	
10.0	1.505	0.110	1.505	4.529	0.005	
20.0	1.507	0.220	1.507	4.544	6.010	
.	.	.	.	.	.	
780.0	1.455	8.580	1.519	4.235	0.419	
790.0	1.446	8.690	1.513	4.181	0.431	
800.0	1.436	8.800	1.507	4.123	0.444	
x	**Y**	**Q**	**q***	**E**	**A**	**F_r**
0.0	1.436	8.800	8.654	1.507	4.123	0.444
0.050	1.444	8.367	8.679	1.507	4.170	0.416
.	.	.	.	.	.	.
0.950	1.506	0.443	8.865	1.507	4.539	0.020
1.000	1.507	0.000	8.865	1.507	4.539	0.000

Q_{in} = 10.0000, q* = 0.0110, Y_l = 6.568, Y_r = 6.775, Y_e = 6.776, $(F_r)_l$ = 0.011, $(F_r)_r$ = 0.020

x	Y	Q	E	A	F_r	
0.0	6.568	10.000	6.569	86.287	0.011	
10.0	6.571	10.110	6.571	86.355	0.011	
.	.	.	.	.	.	
780.0	6.770	18.580	6.770	91.661	0.019	
790.0	6.772	18.690	6.773	91.731	0.020	
800.0	6.775	18.800	6.776	91.801	0.020	
x	**Y**	**Q**	**q***	**E**	**A**	**F_r**
0.0	6.775	18.800	18.799	6.776	91.801	0.020
0.050	6.775	17.860	18.799	6.776	91.803	0.019
.	.	.	.	.	.	.
0.950	6.776	0.940	18.800	6.776	91.825	0.001
1.000	6.776	0.000	18.800	6.776	91.826	0.000

Depths etc. when in flow to gutter Q_{in} is increased

Y_l	Y_r	Y_e	$(F_r)_l$	$(F_r)_r$	Q_{in}	q*
1.502	1.436	1.507	0.000	0.444	.000	0.011
1.617	1.625	1.674	0.019	0.344	.500	0.011
1.754	1.818	1.852	0.031	0.274	1.000	0.011
1.913	2.017	2.042	0.037	0.222	1.500	0.011
2.091	2.224	2.242	0.039	0.183	2.000	0.011
2.286	2.438	2.453	0.039	0.152	2.500	0.011
2.495	2.662	2.673	0.038	0.127	3.000	0.011
2.717	2.895	2.903	0.036	0.108	3.500	0.011
2.951	3.136	3.143	0.033	0.092	4.000	0.011
3.197	3.387	3.393	0.031	0.078	4.500	0.011

(*continued*)

(continued)

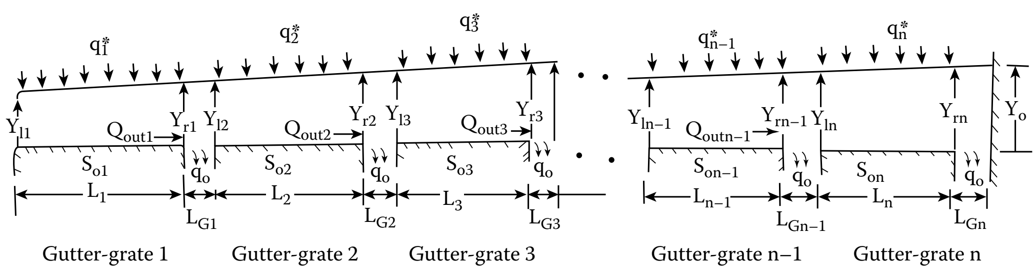
Y_l	Y_r	Y_e	$(F_r)_l$	$(F_r)_r$	Q_{in}	q*
3.453	3.648	3.652	0.028	0.068	5.000	0.011
3.720	3.918	3.921	0.026	0.059	5.500	0.011
3.997	4.197	4.200	0.023	0.051	6.000	0.011
4.284	4.486	4.488	0.021	0.045	6.500	0.011
4.581	4.784	4.786	0.019	0.039	7.000	0.011
4.888	5.092	5.094	0.018	0.035	7.500	0.011
5.205	5.410	5.411	0.016	0.031	8.000	0.011
5.531	5.737	5.738	0.015	0.027	8.500	0.011
5.867	6.073	6.074	0.013	0.024	9.000	0.011
6.213	6.419	6.420	0.012	0.022	9.500	0.011
6.568	6.775	6.776	0.011	0.020	10.000	0.011

Depths etc. when lateral inflow to gutter q*. is increased

Y_l	Y_r	Y_e	$(F_r)_l$	$(F_r)_r$	Q_{in}	q*
1.821	1.897	1.927	0.056	0.251	2.000	0.010
2.091	2.224	2.242	0.039	0.183	2.000	0.011
2.408	2.571	2.584	0.028	0.136	2.000	0.012
2.762	2.942	2.950	0.020	0.104	2.000	0.013
3.146	3.336	3.342	0.014	0.081	2.000	0.014
3.558	3.755	3.759	0.010	0.064	2.000	0.015
3.997	4.197	4.200	0.008	0.051	2.000	0.016
4.461	4.664	4.666	0.006	0.041	2.000	0.017
4.951	5.155	5.157	0.005	0.034	2.000	0.018
5.465	5.670	5.672	0.004	0.028	2.000	0.019
6.005	6.210	6.212	0.003	0.023	2.000	0.020

4.20.7 Subcritical Flow through n Gutter–Grates

As a final application, we will deal with the problem in which the lateral inflow into gutter 1 does not all discharge into grate 1 with its outflow Q_{out1} flowing into gutter 2, etc. This process of having the excess flow passed into the next gutter–grate continues to the last, which will be designated nth, gutter–grate, where the flow is terminated with a wall so that $Q_{outn} = 0$, as shown in the sketch below.



To describe the variables involved in these n gutter–grates, an extra subscript will be added to denote the number of the gutter–grate. Thus, the depth on the left side of gutter 1 will be identified by Y_{l1} and the depth on its right side, which is also the depth at the beginning of grate 1, is y_{r1}. The depth at the end of grate 1 will be the same depth as the beginning of gutter 2 and is Y_{l2}, etc. The depth at the end of the n (and final grate) is Y_e. To accommodate gutters and grates of different lengths and bottom slopes, these will also be given a second subscript as shown in the above sketch. A grate does not exist upstream from gutter 1, and therefore we will assume no reverse flow occurs in this gutter, i.e., the flow at its beginning is zero.

The unknown variables for this system of n gutter–grates are Y_{l1}, Y_{r1}, Q_{out1}, Y_{l2}, Y_{r2}, Q_{out2}, …, Y_{li}, Y_{ri}, Q_{outi}, …, Y_{ln}, Y_{rn}, Y_e. In other words, the number of unknowns equals 3n. Therefore, 3n simultaneous equations are needed, three from each gutter–grate. For each of these gutter–grates, there are two ODEs available and one continuity equation, as has been used in the previous applications in which the flow is subcritical throughout the length of both the gutter and the grate. This system of 3n equations consists of

$$F_1 = Y_{r1} - Y_{r1ode}(Y_{l1}) = 0 \quad \text{with ODE solved over } x = 0 \text{ to } x = L_1$$

$$F_2 = Y_{l2} - Y_{l2ode}(Y_{r1}) = 0 \quad \text{with ODE solved over } x' = 0 \text{ to } x' = L_{G1}$$

$$F_3 = q_1^* L_1 - \int q_o^* dx - Q_{out1} = 0$$

$$F_4 = Y_{r2} - Y_{r2ode}(Y_{l2}) = 0 \quad \text{with ODE solved over } x = 0 \text{ to } x = L_2$$

$$F_5 = Y_{l3} - Y_{l3ode}(Y_{r2}) = 0 \quad \text{with ODE solved over } x' = 0 \text{ to } x' = L_{G2}$$

$$F_6 = q_2^* L_2 - \int q_o^* dx - Q_{out2} + Q_{out1} = 0$$

$$\vdots$$

$$F_{3i-2} = Y_{ri} - Y_{riode}(Y_{li}) = 0 \quad \text{with ODE solved over } x = 0 \text{ to } x = L_i$$

$$F_{3i-1} = Y_{l(i+1)} - Y_{l(i+1)ode}(Y_{r1}) = 0 \quad \text{with ODE solved over } x' = 0 \text{ to } x' = L_{Gi}$$

$$F_{3i} = q_1^* L_i - \int q_o^* dx - Q_{outi} + Q_{outi-1} = 0$$

$$\vdots$$

$$F_{3n-2} = Y_{ln} - Y_{rnode}(Y_{ln}) = 0 \quad \text{with ODE solved over } x = 0 \text{ to } x = L_n$$

$$F_{3n-1} = Y_e - Y_{lnode}(Y_{rn}) = 0 \quad \text{with ODE solved over } x' = 0 \text{ to } x' = L_{Gn}$$

$$F_{3n} = q_n^* L_n - \int q_o^* dx' + Q_{out(n-1)} = 0$$

The Jacobian matrix for use in the Newton method from these equations forms a special banded matrix with one nonzero element in the 1st column, i.e., $D_{1,1} \neq 0$, with three nonzero elements in the 2nd column, i.e., $D_{1,2} = 1$, $D_{2,2} \neq 0$, and $D_{3,2} \neq 0$. In the third, sixth, or in general 3i, columns that are associated with the variables Q_{outi} there are four nonzero elements. In these columns, the diagonal elements will equal $D_{3i,3i} = -1.0$, and nonzero elements will be in the next four rows below this diagonal position caused by Q_{outi-1} on the three equations for the channel-grate i. In general, columns

3i–2 (those associated with Y_l) will have $D_{3i-4,3i-2} = 1$ and $D_{3i-2,3i-2} \neq 0$ (i.e., two elements in these columns will be nonzero). In general, columns 3i – 1 (those associated with Y_r) will have $D_{3i-2,3i-1} = 1$ and $D_{3i-1,3i-1}$ (the diagonal element), and $D_{3i,3i-1}$ (the element just below the diagonal) $\neq 0$. Since for the last channel Y_e replaces Q_{outn} as the unknown variable, the last or (3n)th column of the Jacobian matrix will contain only one nonzero element in the second from the last row, i.e., $D_{3n-1,3n} = 1$.

Program GUTGRTN, whose listing is given below, is designed to solve the depth and the flow rates Q_{outi} passing from one gutter–grate to the next gutter–grate. The c-version of this program calls on the ODE-solver rukust, and takes advantage of the sparseness of the Jacobian matrix by using a one-dimensional array for it, and has its own linear algebra solver, as asked that you do in a homework problem. This program requires that all gutter–grates be triangular with the same side slope m and the same Manning's n. However, each gutter and its following grate can have different bottom slopes S_o, different lateral inflows into the gutter, and different lengths of gutter and grate. Also, each grate can have a different bottom width b, a fraction of area open f, and discharge coefficient C_d. Because of the amount of input data, the program reads this data from a file. The first line consists of NG = Number of gutter–grates, IOUT = logical output unit number, TOL = error criteria to be used in solving ODEs, ERR = error criteria for the Newton method, FN = Manning's n, FM = side slope of triangular gutter–grate, and G = acceleration of gravity. For each of the NG gutter–grates there are two lines of input: the first contains SOO(I) = bottom slope, FL(I) = length of gutter, FLG(I) = length of grate, QS(I) = lateral inflow q* into the gutter, DX(I) = Δx to be used in printing out the profile across the gutter, DXG(I) = Δx to be used in numerically integrating and printing out the profile across the grate, B(I) = bottom width of grate, FRAC(I) = fraction of area at the bottom of the grate that is open, and CD(I) = discharge coefficient. The second line of this input for each gutter–grate provides the estimates for the unknown variables Y_l, Y_r, and Q_{out}, with Q_{out} replaced by Y_e for the last gutter–grate.

The subroutine FUN that supplies values to the 3n equations is designed so that if its last argument K is zero, then it supplies values to all 3n equations in the array F. If this last argument is not zero, but equal to the equation number, then the returned values of the equation(s) for the selected gutter–grate come into array FF, and these values are used with the appropriate logic in the main program to provide the values to the nonzero elements in the Jacobian matrix. You should follow this logic through carefully to determine how the main program interacts with the subroutine FUN to provide values to the nonzero elements of the Jacobian. The program calls on the same linear algebra solver SOLVEQ, used previously, to provide the solution to the linear system of equations, but a much more efficient solution could be developed by utilizing the special sparseness of the Jacobian matrix (see a homework problem.)

Program GUTGRTN.FOR

```
      PARAMETER (MG=6,ME=3*MG)
      REAL F[ALLOCATABLE](:),D[ALLOCATABLE](:,:),FF(3)
      INTEGER*2 INDX[ALLOCATABLE](:)
      LOGICAL DONE
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(ME),DX(MG),SOO(MG),SO,FM,FMS,QS(MG),QT(MG),CC,
     &FN,HM,G,G2,G8,FL(MG),FLG(MG),DXG(MG),CD(MG),TOL,QS1,XX,HMIN,
     &SQSTAR,DONE,IOUT,NG,J
      READ(2,*) NG,IOUT,TOL,ERR,FN,FM,G
      IF(NG.LE.MG) GO TO 1
      WRITE(*,*)' Program dimensioned to only',MG,' channels'
      STOP
1     NE=3*NG
      ALLOCATE(F(NE),D(NE,NE),INDX(NE))
      G2=2.*G
```

```
      G8=.128*G
      FMS=1.+SQRT(FM*FM+1.)
      HM=.5*FM
      DONE=.FALSE.
      CC=1.486
      IF(G.LT.20.) CC=1.
      DO 10 I=1,NG
      READ(2,*) SOO(I),FL(I),FLG(I),QS(I),DX(I),DXG(I),B,FRAC,CD(I)
      CD(I)=CD(I)*SQRT(G2)*FRAC*B
      QT(I)=FL(I)*QS(I)
      II=3*I
10    READ(2,*) X(II-2),X(II-1),X(II)
      NCT=0
20    CALL FUN(NE,F,FF,0)
      DO 22 I=1,NE
      DO 22 J=1,NE
22    D(I,J)=0.
C Determines Jacobian for first channel-grate
      DXX=.005*X(1)
      X(1)=X(1)+DXX
      CALL FUN(NE,F,FF,1)
      D(1,1)=(FF(1)-F(1))/DXX
      X(1)=X(1)-DXX
      D(1,2)=1.
      DXX=.005*X(2)
      X(2)=X(2)+DXX
      CALL FUN(NE,F,FF,2)
      D(2,2)=(FF(2)-F(2))/DXX
      D(3,2)=(FF(3)-F(3))/DXX
      X(2)=X(2)-DXX
C Determines Jacobian for rest of channel-grates
      DO 30 JJ=2,NG
Changes Qo(i-1)
      II=3*JJ-3
      D(II,II)=-1.
      DXX=.005*X(II)
      X(II)=X(II)+DXX
      CALL FUN(NE,F,FF,II+3)
      D(II+1,II)=(FF(1)-F(II+1))/DXX
      D(II+2,II)=(FF(2)-F(II+2))/DXX
      D(II+3,II)=(FF(3)-F(II+3))/DXX
      X(II)=X(II)-DXX
Changes Yl(i) - diagonal element
      II=II+1
      DXX=.005*X(II)
      X(II)=X(II)+DXX
      CALL FUN(NE,F,FF,II)
      D(II,II)=(FF(1)-F(II))/DXX
      X(II)=X(II)-DXX
      D(II-2,II)=1.
      D(II,II+1)=1.
```

```
Changes Yr(i)
      II=II+1
      DXX=.005*X(II)
      X(II)=X(II)+DXX
      CALL FUN(NE,F,FF,II)
      D(II-1,II)=1.
      D(II,II)=(FF(2)-F(II))/DXX
      D(II+1,II)=(FF(3)-F(II+1))/DXX
      X(II)=X(II)-DXX
      IF(JJ.EQ.NG) D(II,II+1)=1.
30    CONTINUE
      CALL SOLVEQ(NE,1,NE,D,F,1,DD,INDX)
      NCT=NCT+1
      SUM=0.
      DO 40 I=1,NE
      X(I)=X(I)-F(I)
40    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,(X(I),I=1,NE)
110   FORMAT(' NCT=',I3,' SUM=',E12.6,/,(8F10.4))
      IF(NCT.LT.30 .AND. SUM.GT.ERR) GO TO 20
      WRITE(IOUT,120)
120   FORMAT(' No  Yl   Yr  Qout (Fr)l (Fr)r')
      DO 50 JJ=1,NG
      II=3*JJ
      IF(JJ.EQ.1) THEN
      Frr=SQRT(QT(JJ)**2*FM*X(2)/(G*(HM*X(2)**2)**3))
      Frl=0.
      ELSE
      Frr=SQRT((QT(JJ)+X(II-3))**2*FM*X(II-1)/(G*(HM*X(II-1)**2)**3))
      Frl=SQRT(X(II-2)**2*FM*X(II-2)/(G*(HM*X(II-2)**2)**3))
      ENDIF
50    WRITE(IOUT,151) JJ,X(II-2),X(II-1),X(II),Frl,Frr
151   FORMAT(I3,5F7.3)
      DONE=.TRUE.
      CALL FUN(NE,F,FF,0)
      END
      SUBROUTINE FUN(NE,F,FF,K)
      PARAMETER (MG=6,ME=3*MG)
      LOGICAL DONE
      EXTERNAL DYX,DYXG
      REAL WW(1,13),Y(1),DY(1),XP(1),YP(1,1),F(NE),FF(3)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/X(ME),DX(MG),SOO(MG),SO,FM,FMS,QS(MG),QT(MG),CC,
     &FN,HM,G,G2,G8,FL(MG),FLG(MG),DXG(MG),CD(MG),TOL,QS1,XX,HMIN,
     &SQSTAR,DONE,IOUT,NG,J
      IF(DONE) WRITE(IOUT,100)
100   FORMAT(/,'  x  Y  Q  E  A  & Fr  q*',/,1X,69('-'))
      IF(K.EQ.0) THEN
      J1=1
      J2=NG
      IODE=0
```

```
      ELSE
      J1=(K-1)/3+1
      J2=J1
      IODE=MOD(K,3)
      ENDIF
      DO 60 J=J1,J2
      I=3*J
      SO=SOO(J)
      DXGH=.5*DXG(J)
      IF(IODE.EQ.2) GO TO 20
      H1=.1
      XX=0.
      Y(1)=X(I-2)
      IF(DONE) THEN
      WRITE(IOUT,111) J
111   FORMAT(' Gutter-Grate #',I3)
      A=HM*Y(1)**2
      Q=0.
      IF(J.GT.1) Q=Q+X(I-3)
      WRITE(IOUT,110) XX,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
      ENDIF
10    XZ=XX+DX(J)
      IF(XZ.GT.FL(J)) XZ=FL(J)
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYX)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS(J)*XZ
      IF(J.GT.1) Q=Q+X(I-3)
      WRITE(IOUT,110) XZ,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5))
110   FORMAT(F10.1,5F10.3)
      ENDIF
      XX=XZ
      IF(XZ.LT.FL(J)) GO TO 10
      IF(K.EQ.0) THEN
      F(I-2)=X(I-1)-Y(1)
      ELSE
      FF(1)=X(I-1)-Y(1)
      ENDIF
      IF(IODE.EQ.1) RETURN
20    H1=.002
      XX=0.
      Y(1)=X(I-1)
      SQSTAR=0.
      QS1=CD(J)*SQRT(Y(1))
      IF(DONE) THEN
      A=HM*X(2)**2
      Q=QS(J)*FL(J)
      IF(J.GT.1) Q=Q+X(I-3)
      WRITE(IOUT,130) XX,Y(1),Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5)),QS1
      ENDIF
```

```
30    XZ=XX+DXG(J)
      IF(XZ.GT.FLG(J)) XZ=FLG(J)
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,WW,DYXG)
      QS2=CD(J)*SQRT(Y(1))
      SQSTAR=SQSTAR+DXGH*(QS1+QS2)
      IF(DONE) THEN
      A=HM*Y(1)**2
      Q=QS(J)*FL(J)-SQSTAR
      IF(J.GT.1) Q=Q+X(I-3)
      WRITE(IOUT,130) XZ,Y,Q,Y(1)+(Q/A)**2/G2,A,Q/(FM*SQRT
     &(G8*Y(1)**5)),QS2
130   FORMAT(F10.3,6F10.3)
      ENDIF
      QS1=QS2
      XX=XZ
      IF(XZ.LT.FLG(J)) GO TO 30
      Q=QS(J)*FL(J)
      IF(J.GT.1) Q=Q+X(I-3)
      IF(J.LT.NG) THEN
      IF(K.EQ.0) THEN
      F(I-1)=X(I+1)-Y(1)
      F(I)=Q-X(I)-SQSTAR
      ELSE
      FF(2)=X(I+1)-Y(1)
      FF(3)=Q-X(I)-SQSTAR
      ENDIF
      ELSE
      IF(K.EQ.0) THEN
      F(I-1)=X(I)-Y(1)
      F(I)=Q-SQSTAR
      ELSE
      FF(2)=X(I)-Y(1)
      FF(3)=Q-SQSTAR
      ENDIF
      ENDIF
60    CONTINUE
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      PARAMETER (MG=6,ME=3*MG)
      LOGICAL DONE
      REAL Y(1),DY(1)
      COMMON /TRAS/X(ME),DX(MG),SOO(MG),SO,FM,FMS,QS(MG),QT(MG),CC,
     &FN,HM,G,G2,G8,FL(MG),FLG(MG),DXG(MG),CD(MG),TOL,QS1,ZZ,HMIN,
     &SQSTAR,DONE,IOUT,NG,J
      YY=ABS(Y(1))
      A=HM*YY**2
      Q=QS(J)*XX
      IF(J.GT.1) Q=Q+X(3*(J-1))
      SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
      SF=SF*ABS(SF)
      FR2=Q*Q*FM*YY/(G*A**3)
```

```
      DY(1)=(SO-SF-2.*Q*QS(J)/(G*A**2))/(1.-FR2)
      RETURN
      END
      SUBROUTINE DYXG(XP,Y,DY)
      PARAMETER (MG=6,ME=3*MG)
      LOGICAL DONE
      REAL Y(1),DY(1)
      COMMON /TRAS/X(ME),DX(MG),SOO(MG),SO,FM,FMS,QS(MG),QT(MG),CC,
     &FN,HM,G,G2,G8,FL(MG),FLG(MG),DXG(MG),CD(MG),TOL,QS1,XX,HMIN,
     &SQSTAR,DONE,IOUT,NG,J
      YY=Y(1)
      QS2=CD(J)*SQRT(YY)
      A=HM*YY**2
      Q=QS(J)*FL(J)-SQSTAR-.5*(XP-XX)*(QS1+QS2)
      IF(J.GT.1) Q=Q+X(3*(J-1))
      SF=FN*Q/CC*(FMS*YY/A)**.66666667/A
      SF=SF*ABS(SF)
      FR2=Q*Q*FM*YY/(G*A**3)
      DY(1)=(SO-SF+Q*QS2/(G*A**2))/(1.-FR2)
      RETURN
      END
```

GUTGRTBA.C (The BA at the end of the name indicates it uses its own band solver for the sparse Jacobian matrix)

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
const mg=6,me=18;
float x[18],dx[6],soo[6],so,fm,fms,qs[6],qt[6],cc,fn,hm,g,g2,\
   g8,fl[6],flg[6];
float dxg[6],cd[6],tol,qs1,xx,sqstar;
int done,ng,jm,kgrate; FILE *fili,*filo;
extern void rukust(int ne,float *dxs,float xbeg,float xend,\
   float err,float *y,float *ytt);
void slope(float xp,float *yp,float *dy);
void fun(float *f,float *ff,int k){
   int j1,j2,iode,j,i; float dxgh,*h1,y[1],ytt[1],xz,q,a,qs2,flm;
   if(done){fprintf(filo," x     Y     Q     E     A\
      Fr     q*\n");
      for(i=0;i<69;i++)fprintf(filo,"-"); fprintf(filo,"\n");}
   if(k==0){j1=1;j2=ng;iode=0;}
   else {j1=(k-1)/3+1;j2=j1;iode=k%3;}
   for(j=j1;j<=j2;j++){jm=j-1;i=3*j;so=soo[jm];
   dxgh=.5*dxg[jm];
   if(iode==2) goto L20;
   *h1=.1; xx=0.; y[0]=x[i-3]; kgrate=0;
   if(done){fprintf(filo,"Gutter-Grate # %3d\n",j);
      a=hm*y[0]*y[0];q=0.;if(j>1)q+=x[i-4];
      fprintf(filo,"%10.1f %9.3f %9.3f %9.3f %9.3f %9.3f\n",xx,\
         y[0],q,y[0]+(q/a)*(q/a),a,q/(fm*sqrt(g8*pow(y[0],5.))));}
```

```
   flm=fl[jm]-1.e-8;
L10: xz=xx+dx[jm];
   if(xz>fl[jm]) xz=fl[jm]; rukust(1,h1,xx,xz,tol,y,ytt);
   if(done){a=hm*y[0]*y[0]; q=qs[jm]*xz; if(j>1) q+=x[i-4];
   fprintf(filo,"%10.1f %9.3f %9.3f %9.3f %9.3f %9.3f\n",xz,\
      y[0],q,y[0]+(q/a)*(q/a),a,q/(fm*sqrt(g8*pow(y[0],5.))));}
   xx=xz; if(xz<flm) goto L10;
   if(k==0) f[i-3]=x[i-2]-y[0]; else ff[0]=x[i-2]-y[0];
   if(iode==1) return;
L20:*h1=.001; xx=0.;kgrate=1;y[0]=x[i-2];sqstar=0.;
   qs1=cd[jm]*sqrt(y[0]);flm=flg[jm]-1.e-8;
   if(done){a=hm*x[1]*x[1];q=qs[jm]*fl[jm]; if(j>1)q+=x[i-4];
      fprintf(filo,"%10.3f %9.3f %9.3f %9.3f %9.3f %9.3f %9.3f\
         \n",xx,y[0],q,y[0]+(q/a)*(q/a),a,q/\
         (fm*sqrt(g8*pow(y[0],5.))),qs1);}
L30:xz=xx+dxg[jm]; if(xz>flg[jm]) xz=flg[jm];
   rukust(1,h1,xx,xz,tol,y,ytt); qs2=cd[jm]*sqrt(y[0]);
   sqstar+=dxgh*(qs1+qs2);
   if(done){a=hm*y[0]*y[0];q=qs[jm]*fl[jm]-sqstar;
      if(j>1)q+=x[i-4];
      fprintf(filo,"%10.3f %9.3f %9.3f %9.3f %9.3f %9.3f %9.3f\
         \n",xz,y[0],q,y[0]+(q/a)*(q/a),a,q/\
         (fm*sqrt(g8*pow(y[0],5.))),qs2);}
      qs1=qs2; xx=xz; if(xz<flm) goto L30;
      q=qs[jm]*fl[jm]; if(j>1) q+=x[i-4];
      if(j<ng){
         if(k==0){f[i-2]=x[i]-y[0];
            f[i-1]=q-x[i-1]-sqstar;}
         else {ff[1]=x[i]-y[0]; ff[2]=q-x[i-1]-sqstar;}}
      else {
         if(k==0){f[i-2]=x[i-1]-y[0];
            f[i-1]=q-sqstar;}
         else {ff[1]=x[i-1]-y[0];ff[2]=q-sqstar;} } }
   return;} //End fun
void slope(float xp,float *yp,float *dy){
   float yy,a,q,sf,fr2,qs2;
   yy=*yp; a=hm*yy*yy;
   if(kgrate) goto L10;
   q=qs[jm]*xp; if(jm>0) q+=x[3*jm-1];
   sf=fn*q/cc*pow(fms*yy/a,.6666667)/a;sf*=fabs(sf);
   fr2=q*q*fm*yy/(g*pow(a,3.));
   *dy=(so-sf-2.*q*qs[jm]/(g*a*a))/(1.-fr2);return;
L10: qs2=cd[jm]*sqrt(yy);
   q=qs[jm]*fl[jm]-sqstar-.5*(xp-xx)*(qs1+qs2);
   if(jm>0) q+=x[3*jm-1];
   sf=fn*q/cc*pow(fms*yy/a,.6666667)/a;sf*=fabs(sf);
   fr2=q*q*fm*yy/(g*pow(a,3.));
   *dy=(so-sf+q*qs2/(g*a*a))/(1.-fr2);return;
   } // End slope
void solbands(int n, float *f, float *d){
   int n1,n3,i,j,i9,ij,i3;float fac;
```

```
   n1=n-1;
// eliminates d[8],d[17],..d[9i-1] .. d[9(n-1)-1]
   for(i=1;i<n;i++){i9=9*i-1;fac=d[i9]/d[i9-1];
      d[i9+5]=d[i9+5]-\fac*d[i9+4];
      if(i<n1)d[i9+10]=-fac;f[3*i+2]-=fac*f[3*i+1];}
   d[9*n-4]=-fac;
// eliminates d[7],d[16]..d[9i-2]..d[9(n-1)-2] &
// adds d[11],d[20]..d[9(i-1)+2]
   for(i=1;i<n;i++){i9=9*i-1;fac=d[i9-1]/d[i9-2];
   d[i9+3]=-fac*d[i9+2];d[i9+4]-=fac;f[3*i+1]-=fac*f[3*i];}
// eliminates element just below diagonal &
// adds d[9],d[18]..d[9(i-1)]
   for(i=1;i<n;i++){i9=9*(i-1)-1;ij=3*(i-1);
      for(j=1;j<4;j++){fac=d[i9+2*j+1]/d[i9+2*j];
         if(j==1) d[i9+4]-=fac;
         else if(j==2) d[i9+10]-=fac;
         else d[i9+11]-=fac*d[i9+10];
         f[ij+j]-=fac*f[ij+j-1];}}
// eliminates elements for last gutter-grate
   i9=9*n1-1;fac=d[i9+3]/d[i9+2];d[i9+4]-=fac; n3=3*n-1;
   f[n3-1]-=fac*f[n3-2]; fac=d[i9+5]/d[i9+4]; d[i9+6]-=fac;
   f[n3]-=fac*f[n3-1];
// back substitution
   f[n3]/=d[i9+6];f[n3-1]=(f[n3-1]-f[n3])/d[i9+4];
   f[n3-2]=(f[n3-2]-f[n3-1])/d[i9+2];
   for(i=n1;i>0;i–){i9=9*(i-1)-1;i3=3*i-1;
      f[i3]=(f[i3]-d[i9+10]*f[i3+1])/d[i9+6];
      f[i3-1]=(f[i3-1]-f[i3+1])/d[i9+4];
      f[i3-2]=(f[i3-2]-f[i3-1])/d[i9+2];}
   return; } // End of solbands
void main(void){ char fname[20];int ne,i,j,ii,jj,ij,nct;
   float dxx,b,frac,err,*f,ff[3],*d,frr,frl,sum;
   printf("Give name of input file\n"); scanf("%s",fname);
   if((fili=fopen(fname,"r"))==NULL){
      printf("Cannot open file");exit(0);}
   printf("Give name of output file\n");scanf("%s",fname);
   if((filo=fopen(fname,"w"))==NULL){
      printf("Cannot open file");exit(0);}
   fscanf(fili,"%d %f %f %f %f %f",&ng,&tol,&err,&fn,&fm,&g);
   ne=3*ng;
   if(ng>mg){
   printf("Program dimensioned to only %d channels\n",mg);
   exit(0);}
   f=(float *)calloc(ne,sizeof(float));
   d=(float *)calloc((9*ng-3),sizeof(float));
   g2=2.*g; g8=.128*g;fms=1.+sqrt(fm*fm+1.);
   hm=.5*fm;done=0;cc=1.486;if(g<20.)cc=1.;
   for(i=0;i<ng;i++){
      fscanf(fili,"%f %f %f %f %f %f %f %f %f",&soo[i],&fl[i],\
         &flg[i],&qs[i],&dx[i],&dxg[i],&b,&frac,&cd[i]);
      cd[i]*=sqrt(g2)*frac*b;qt[i]=fl[i]*qs[i]; ii=3*(i+1)-1;
```

```
      fscanf(fili,"%f %f %f",&x[ii-2],&x[ii-1],&x[ii]);}
   nct=0;
   do{ fun(f,ff,0); for(i=0;i<9*ng-3;i++) d[i]=0.;
// Determines Jacobian for first channel-grate
      dxx=.005*x[0]; x[0]+=dxx; fun(f,ff,1);
      d[1]=(ff[0]-f[0])/dxx; x[0]-=dxx; dxx=.005*x[1];x[1]+=dxx;
      fun(f,ff,2);d[3]=(ff[1]-f[1])/dxx;
      d[4]=(ff[2]-f[2])/dxx;x[1]-=dxx;
// Determines Jacobian for rest of channel-grates
      for(jj=2;jj<=ng;jj++){/* Changes Qo(i-1)*/
         ii=3*jj-4;ij=9*(jj-1)-1; d[ij-3]=-1.;dxx=.005*x[ii];
         x[ii]+=dxx; fun(f,ff,ii+4);d[ij-2]=(ff[0]-f[ii+1])/dxx;
         d[ij-1]=(ff[1]-f[ii+2])/dxx;d[ij]=(ff[2]-f[ii+3])/dxx;
         x[ii]-=dxx;
// Changes Yl(i) - diagonal element
         ii++;dxx=.005*x[ii];x[ii]+=dxx;fun(f,ff,ii+1);
         d[ij+2]=(ff[0]-f[ii])/dxx;x[ii]-=dxx;
// Changes Yr(i)
         ii++;dxx=.005*x[ii];x[ii]+=dxx;fun(f,ff,ii+1);
         d[ij+4]=(ff[1]-f[ii])/dxx;d[ij+5]=(ff[2]-f[ii+1])/dxx;
         x[ii]-=dxx; }
      solbands(ng,f,d); nct++; sum=0.;
      for(i=0;i<ne;i++){x[i]-=f[i]; sum+=fabs(f[i]);}
      printf("NCT= %d SUM= %f ",nct,sum);
      for(i=0;i<ne;i++)printf("%9.4f ",x[i]);printf("\n");
   }while((nct<30) && (sum>err));
   fprintf(filo,"No Yl Yr Qout (Fr)l (Fr)r\n");
   for(jj=1;jj<=ng;jj++){ ii=3*jj;
      if(jj==1){
         frr=sqrt(qt[jj-1]*qt[jj-1]*fm*x[1]/\
            (g*pow(hm*x[1]*x[1],3.)));frl=0.;}
      else {
         frr=sqrt(pow(qt[jj-1]+x[ii-4],2.)*fm*x[ii-2]/\
           (g*pow(hm*x[ii-2]*x[ii-2],3.)));
         frl=sqrt(x[ii-4]*x[ii-4]*fm*x[ii-3]/\
            (g*pow(hm*x[ii-3]*x[ii-3],3.)));}
      fprintf(filo,"%3d %6.3f %6.3f %6.3f %6.3f %6.3f\n",\
         jj,x[ii-3],x[ii-2],x[ii-1],frl,frr);}
   done=1; fun(f,ff,0); fclose(fili); fclose(filo);}
   // End of main
```

EXAMPLE PROBLEM 4.64

Three triangular gutter–grates with $S_o = 0.0003$, $m = 4$, and $n = 0.013$, each with a gutter length of $L = 800$ ft and length of the following grate $L_G = 1$ ft follow each other. The last grate has a wall at its end that forces all the flow into it, and the lateral inflow into all the three gutters is the same and constant as $q^* = 0.011$ cfs. The width of each grate is $b = 4$ ft, and 0.48 of its bottom area is opened, and the discharge coefficients are $C_d = 0.45$. Determine the depths throughout these three gutter–grate systems, and the flow rates passing the first and the second grates into the gutters downstream from them.

Solution

To help you follow the details of nonzero elements in the Jacobian matrix as described above, the Jacobian for this problem consists of (as the solution is obtained):

Jacobian matrix resulting from three gutter–grates in series

	Column Number								
	1	2	3	4	5	6	7	8	9
	Derivative with respect to given variable								
Eq No	Y_{l1}	Y_{r1}	Q_{out1}	Y_{l2}	Y_{r2}	Q_{out2}	Y_{l3}	Y_{r3}	Y_e
1	-2.207	1.0	-.2384E-03	0	0	0	0	0	0
2	-1.867	-.3179E-04	-.2384E-03	1.0	0	0	0	0	0
3	-5.519	0	-1.0	0	0	0	0	0	0
4	0	0	.09680	-1.913	1.0	0	0	0	0
5	0	0	.07319	-1.676	0	0	1.0	0	0
6	0	0	1.225	-4.784	0	-1.0	0	0	0
7	0	0	0	0	0	.05817	-1.535	1.0	0
8	0	0	0	0	0	.04560	-1.410	0	1.0
9	0	0	0	0	0	1.133E	-3.791	0	0

The input to program GUTGRTN to solve this problem consist of (file GUTGRTF.DAT)

```
3 3 1.E-5 .001 .013 4 32.2
.0003 800 1 .011 10 .05 4 .48 .45
1.5 1.5 .2                          <-- guesses for Yl, Yr, and Qout
.0003 800 1 .011 10 .05 4 .48 .45
1.55 1.6 .4
.0003 800 1 .011 10 .05 4 .48 .45
1.65 1.7 1.8                        <-- guesses for Yl, Yr, and Ye
```

The solution provided by GUTGRTN (with most of the lines giving the water depth profiles deleted) is given below.

No	Y_l	Y_r	Q_{out}	$(F_r)_l$	$(F_r)_r$	
1	1.493	1.481	.243	.000	.411	
2	1.543	1.562	.280	.065	.370	
3	1.615	1.688	1.728	.061	.306	(Notice the value under Qout is Y_e)

x	Y	Q	E	A	F_r	q*
Gutter-Grate # 1						
.0	1.493	.000	1.493	4.457	.000	
10.0	1.496	.110	1.496	4.475	.005	
.	.	.	.	.	.	
790.0	1.488	8.690	1.548	4.429	.396	
800.0	1.481	8.800	1.543	4.384	.406	
.000	1.481	8.800	1.543	4.384	.406	8.437
.050	1.487	8.378	1.543	4.425	.382	8.456
.	.	.	.	.	.	.
.900	1.542	1.105	1.543	4.758	.046	8.611
.950	1.543	.674	1.543	4.761	.028	8.612
1.000	1.543	.243	1.543	4.763	.010	8.613
Gutter-Grate # 2						
.0	1.543	.243	1.543	4.763	.010	
10.0	1.546	.390	1.546	4.781	.016	
.	.	.	.	.	.	
790.0	1.567	8.970	1.619	4.910	.359	
800.0	1.562	9.080	1.615	4.877	.367	
.000	1.562	9.080	1.628	4.384	.367	8.665
.050	1.567	8.610	1.615	4.913	.345	8.680
.	.	.	.	.	.	.
.950	1.615	.720	1.615	5.215	.027	8.811
1.000	1.615	.280	1.615	5.217	.010	8.812
Gutter-Grate # 3						
.0	1.615	.280	1.615	5.217	.010	
10.0	1.618	1.838	1.620	5.236	.068	
.	.	.	.	.	.	
790.0	1.691	10.418	1.742	5.716	.345	
800.0	1.688	10.528	1.741	5.700	.350	
.000	1.688	10.528	1.778	4.384	.350	9.009
.050	1.692	8.629	1.728	5.728	.285	9.020
.100	1.696	8.178	1.728	5.754	.269	9.030
.	.	.	.	.	.	.
.950	1.728	.456	1.728	5.969	.014	9.113
1.000	1.728	.000	1.728	5.970	.000	9.114

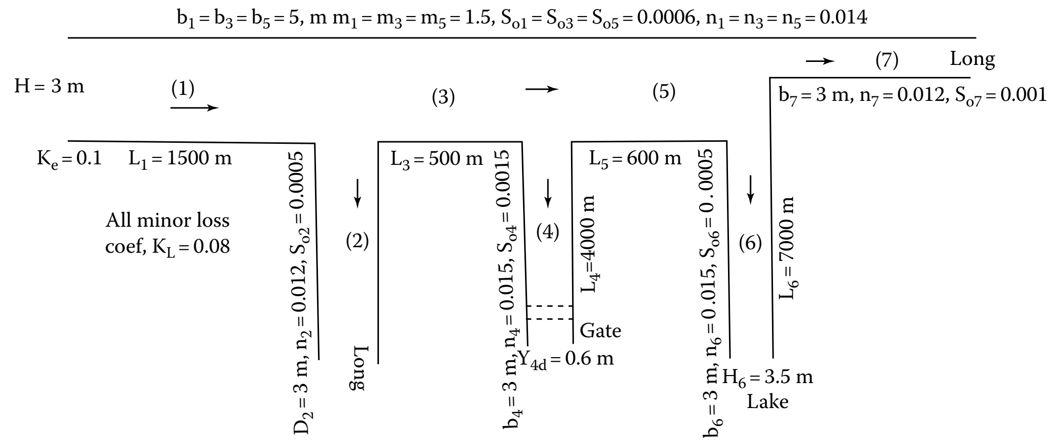
4.21 MULTIPLE BRANCHED CHANNEL SYSTEMS

In setting up and solving the equations that describe depths and flow rates in branched and parallel channel systems, there are some basic principles that should be apparent from the previous problems. These principles are

1. At each junction, i.e., where a channel connects into one or more additional channels, there is generally an additional independent junction continuity equation. Thus, if there are J junctions, then J junction continuity equations exist. The exception to this rule occurs when the flow rate from a single channel branches into several channels and further downstream again combines into a single channel, in which case the number of junction continuity equations is $J - 1$, because the single upstream and downstream channels carry the same flow rate. The parallel systems just considered fall in the category of this exception. If only two channels join, there is no continuity equation here unless this is a point where a concentrated outflow, or inflow, occurs because the flow rate in the two channels are equal, so one less unknown exists. If one wishes, however, an extra equation could be used, i.e., $Q_i - Q_{i+1} = 0$, where i and i + 1 are the two channels at the junction.
2. At each junction, there are $N_i - 1$ independent energy equations, in which N_i is the number of channels that joint at that junction. For example, when the junction consists of two different size channels, i.e., two channels join, then one energy equation $E_1 = E_2 + \Delta z_{12}$ is available; when three channels join at a junction then two energy equations are available, i.e., $E_i = E_{i+1} + \Delta z_{i,i+1}$ and $E_i = E_{i+2} + \Delta z_{i,i+2}$, etc. If only two channels of the same size join at this point and there is no concentrated outflow, then the energy equation does not exist, but since the downstream depth in the upstream channel equals the upstream depth in the downstream channel one less unknown exists. (Actually the energy equation is what requires these depths to be equal, and therefore the energy equation could be written if both depths at the junction are considered unknown.)
3. For each separate channel in the system, there is either an equation that consists of the ODE that solves the gradually varied flow in the channel, or the channel is very long so that the flow will be uniform, then the GVF equation is replaced by an algebraic equation, namely, the uniform flow equation (Manning's equation).
4. An additional equation is available from each boundary condition at those ends of each channel that are not connected to a junction if the depth at this end is not specified, and also the flow in the channel is not uniform. For long channels in which a uniform flow occurs, one can think of the boundary condition as specifying the depth at the nonjunction end of the channel equal to the depth at the junction end of the channel. Such downstream boundary conditions might consist of (a) a gate in which the energy equation can be written across the gate, $E_{ui} = E_{di}$ (subscript u stands for upstream and d for downstream and ui will be the downstream end of a channel, i.e., $E_{ui} = E_{2i}$.); (b) a break in grade to a steep channel, in which the critical flow equation $Q^2T - gA_3 = 0$ applies; (c) the channel discharges into a reservoir, in which case the depth equals the reservoir height above the channel bottom since the velocity head is dissipated in discharging into the reservoir. In case the reservoir supplies flow to this end of the channel, then the boundary condition equation is the energy equation $H - Y - (1+K_e)Q^2/(2g) = 0$.
5. Should a hydraulic jump occur in a channel, then two additional equations become available, the momentum equation $M_{ui} = M_{di}$, in which M is the momentum function and the subscripts u and d represent positions immediately upstream and downstream of the jump, respectively, and i denotes the channel, and two GVF equations, one upstream and the other downstream from the hydraulic jump that replace the single GVF equation (or uniform flow equation) for a channel without a jump. Thus, three equations are available in place of just one equation. Three additional variables are introduced, the depth Y_{1j} immediately upstream from the jump, Y_{2j} immediately downstream from the jump, and the position x where the jump occurs. Since only two new equations can be written for each channel containing a hydraulic jump, it is necessary that the depth at the beginning of the GVF profile upstream from the hydraulic jump be known, and generally also the flow rate.

This is another way of saying that when a supercritical flow occurs, there is a control somewhere upstream, and this constitutes the beginning of the problem consisting of a system of channels. Therefore, a hydraulic jump will only be permitted to occur in the first channel of a channel system. Thus, a gate that causes a critical flow downstream will actually create a new channel system problem.

These general principles are illustrated by the seven channel system below in which there are three junction or positions where the channel branches. Therefore from (1) above, there are three junction



continuity equations. At each of the three junctions, there are two energy equations for six such equations. There is either a GVF equation or Manning's equation available for each of the seven separate channels. In this channel system, Manning's equation will apply to channels (2) and (7) since they are mild and very long. There are two boundary condition equations; one between the supply reservoir and channel (1) and the second from the energy across the gate in channel (4). Thus, there are a total of 3 + 6 + 7 + 2 = 18 equations that allow for the following 18 variables to be solved: Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, Q_7, Y_{11}, Y_{21}, Y_{31}, Y_{41}, Y_{51}, Y_{61}, Y_{71}, Y_{12}, Y_{22}, Y_{42}, Y_{52}, in which the first subscript for the depths Y represent the channel number and the second subscript if 1 denotes upstream and 2 denotes downstream. (Note, the 2nd subscript in some previous problems denoted the channel number.)

These 18 equations are

$$F_1 = Q_1 - Q_2 - Q_3 = 0 \quad \text{Continuity at J.\#1}$$

$$F_2 = Y_{12} + \frac{Q_1^2}{2gA_{12}^2} - Y_{21} - (1+K_L)\frac{Q_2^2}{2gA_{21}^2} - \Delta z_{12} = 0 \qquad \text{Energy at J. \#1}$$

$$F_3 = Y_{12} + \frac{Q_1^2}{2gA_{12}^2} - Y_{31} - (1+K_L)\frac{Q_3^2}{2gA_{31}^2} - \Delta z_{13} = 0 \qquad \text{Energy at J. \#1}$$

$$F_4 = Q_3 - Q_4 - Q_5 = 0 \quad \text{Continuity at J.\#2}$$

$$F_5 = Y_{32} + \frac{Q_3^2}{2gA_{32}^2} - Y_{41} - (1+K_L)\frac{Q_4^2}{2gA_{41}^2} - \Delta z_{34} = 0 \qquad \text{Energy at J. \#2}$$

$$F_6 = Y_{32} + \frac{Q_3^2}{2gA_{32}^2} - Y_{51} - (1+K_L)\frac{Q_5^2}{2gA_{51}^2} - \Delta z_{35} = 0 \quad \text{Energy at J. \# 2}$$

$$F_7 = Q_5 - Q_6 - Q_7 = 0 \quad \text{Continuity at J.\#3}$$

$$F_8 = Y_{52} + \frac{Q_5^2}{2gA_{52}^2} - Y_{61} - (1+K_L)\frac{Q_6^2}{2gA_{61}^2} - \Delta z_{56} = 0 \quad \text{Energy at J. \#3}$$

$$F_9 = Y_{52} + \frac{Q_5^2}{2gA_{52}^2} - Y_{71} - (1+K_L)\frac{Q_7^2}{2gA_{71}^2} - \Delta z_{56} = 0 \quad \text{Energy at J. \#3}$$

$$F_{10} = Y_{11} - Y_{GVF}(Y_{12}) = 0 \quad \text{GVF in Ch. \# 1}$$

$$F_{11} = n_2 Q_2 P_2^{2/3} - C_u A_2^{4/3} S_{o2}^{1/2} = 0 \quad \text{Manning's equation in Ch. \#2}$$

$$F_{12} = Y_{31} - Y_{GVF}(Y_{32}) = 0 \quad \text{GVF in Ch. \#3}$$

$$F_{13} = Y_{41} - Y_{GVF}(Y_{42}) = 0 \quad \text{GVF in Ch. \#4}$$

$$F_{14} = Y_{51} - Y_{GVF}(Y_{52}) = 0 \quad \text{GVF in Ch. \#5}$$

$$F_{15} = Y_{61} - Y_{GVF}(Y_{62}) = 0 \quad \text{GVF in Ch. \#6}$$

$$F_{16} = n_7 Q_7 P_7^{2/3} - C_u A_7^{4/3} S_{o7}^{1/2} = 0 \quad \text{Manning's equation in Ch. \#7}$$

$$F_{17} = H - Y_{11} - (1+K_e)\frac{Q_1^2}{2gA_{11}^2} = 0 \quad \text{Energy bet. res. and Ch. \#1}$$

$$F_{18} = Y_{42} + \frac{Q_4^2}{2gA_{42}^2} - Y_{4d} - \frac{Q_4^2}{2gA_{d4}^2} = 0 \quad \text{Energy across gate}$$

These principles are incorporated into the FORTRAN program SOLGBR.FOR below.

Listing of program SOLGBR.FOR designed to solve branching channel systems

```
      PARAMETER (N=12,M=39)
      LOGICAL*1 TRUX(M)
      LOGICAL*1 IPERM,NTRAN,UPSRES,JUMP,JUMP1
      REAL F(M),F1(M),D(M,M),KL(N),L(N),LT(N)
      INTEGER*2 INDX(M)
      CHARACTER*8 CHDOWN(6)/'Branch 1',' Branch',' Uniform',
     &'Gate','Critical','Reservo.'/
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NJ,NO,NO2,NEQS,ITYP(N),ICTL(N),JN(M),
     &NI(N/2),FN(N),SO(N),L,LT,B(N),FM(N),DZ(N),KL,DB(N),
```

```
     &DFM(N),YG(N),QOUT(N),X(M),H,G2,G,CC,TOL,PERM,TOPW,YO,
     &QN,Q2G,BB,FMM,NO3,IJ,IPERM,NTRAN,UPSRES,JUMP
      DATA IN,IOUT/2,3/
      IPERM=.FALSE.
      NTRAN=.TRUE.
      UPSRES=.TRUE.
C ICTL = -2,-1,0,1,2 OR 3 for types of downstream controls,
C -2=If downstr. branch connects to single ch. of same
C  size(Yi=Yi+1) and QOUT=0,
C i.e. Ei=Ei+1 is not availablae.
C -1=Downstream end of channel is branch.
C 0=uniform, 1=gate, 2=critical, 3=reser.
C YG is depth behind gate for gate; = 0 for critical;
C  =res. depth if reservoir.
C DB and DFM are changes (+ or -) across transition
C  of b and m.
C IN UNKNOWN VECTOR X, Q's COME 1ST; UPSTREAM DEPTHS
C  NEXT & then DOWNS. DEPTHS
C AND IF JUMP IN CH. # 1 THEN Yj1,Yj2 & x, as last
C  three unknowns.
C i.e. Q(I)=X(I);YU(I)=X(I+NO);YG(I)=X(I+NO2);[Yj1,Yj2,x]
      READ(IN,*) NJ,NO,(ITYP(I),B(I),FM(I),FN(I),SO(I),L(I),
     &LT(I),DZ(I),DB(I),DFM(I),KL(I),YG(I),ICTL(I),I=1,NO)
      NO2=2*NO
      NO3=3*NO
      YO=0.
      JUMP=.FALSE.
      DO 1 I=1,NO3+3
1     TRUX(I)=.TRUE.
C If Q1 is specified give 0 for H, and if jump occurs
C  then give ups. supercr. depth for H but precede
C  with a minus sign.
      READ(IN,*) TOL,ERR,H,G
      IUPS=1
      IF(H.LT. 1.E-5) THEN
      UPSRES=.FALSE.
      IUPS=2
      TRUX(1)=.FALSE.
      IF(H.LT.-1.E-5) THEN
      YO=ABS(H)
      TRUX(NO+1)=.FALSE.
      JUMP=.TRUE.
      ENDIF
      ENDIF
      IF(G.GT. 30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
```

```
      G2=2.*G
      NI(1)=0
      II=1
      DO 2 J=1,NJ
C In giving channel nos. that join at junction the first listed
C will have its flow rate positive, and all other will have their
C flow rate subtracted therefrom unless preceded by a minus, i.e.
C the energy eqs. are from downstream of first to upstream of
C other numbers or E(1)downst=E(I)upst, unless preceded by a
C minus, in which case E(1)downstr=E(I)downstr.
      READ(IN,*) JI,(JN(I),I=II,II+JI-1),QOUT(J)
      II=II+JI
2     NI(J+1)=II-1
      DO 4 I=1,NO
      IF(ICTL(I).EQ.3) X(I+NO2)=YG(I)
4     CONTINUE
      DO 6 I=1,NO
      KL(I)=(KL(I)+1.)/G2
      IF(ICTL(I).EQ.3 .OR. ICTL(I).EQ.0) THEN
      READ(IN,*) X(I),X(I+NO)
      IF(ICTL(I).EQ.0) X(I+NO2)=X(I+NO)
      TRUX(I+NO2)=.FALSE.
      ELSE
      READ(IN,*) X(I),X(I+NO),X(I+NO2)
      ENDIF
6     CONTINUE
      IF(YO.GT.0.) THEN
C Guess for 1-depth upst. jump, 2-downst. jump,
C  & 3-position of jump, x
      READ(IN,*) X(NO3+1),X(NO3+2),X(NO3+3)
      NO3=NO3+3
      X(NO+1)=YO
      ENDIF
      NCT=0
      JUMP1=JUMP
15    JUMP=JUMP1
      CALL FUN(F)
      J=0
      DO 20 II=IUPS,NO3
      IF(.NOT.TRUX(II)) GO TO 20
      J=J+1
      DX=.005*X(II)
      X(II)=X(II)+DX
      JUMP=JUMP1
      CALL FUN(F1)
      DO 10 I=1,NEQS
10    D(I,J)=(F1(I)-F(I))/DX
      X(II)=X(II)-DX
20    CONTINUE
```

```
      CALL SOLVEQ(NEQS,1,M,D,F,1,DD,INDX)
      DIF=0.
      J=0
      DO 30 I=IUPS,NO3
      IF(TRUX(I)) THEN
      J=J+1
      X(I)=X(I)-F(J)
      DIF=DIF+ABS(F(J))
      ELSE
      IF(I.EQ.NO+1 .AND. IUPS.EQ.2) GO TO 30
      IF(ICTL(I-NO2).EQ.0) X(I)=X(I-NO)
      IF(ICTL(I-NO2).EQ.-2) X(I)=X(I-NO+1)
      ENDIF
30    CONTINUE
      NCT=NCT+1
      WRITE(*,200) NCT,DIF,(X(I),I=1,NO3)
200   FORMAT(' NCT=',I3,' SUM=',E12.4,/,(10F8.2))
      IF(NCT.LT.30 .AND. DIF.GT.ERR) GO TO 15
      WRITE(IOUT,100) NO
100   FORMAT(' Solution to',I3,' Channel Problem',/,1X,90('-'),/,
     &' No Ty   b  m   n   So   L  dz',' db  dm  Yu  Yd  Q
     &Downstream',/,1X,90('-'))
      DO 50 I=1,NO
50    WRITE(IOUT,110) I,ICTL(I),B(I),FM(I),FN(I),SO(I),L(I),DZ(I),
     &DB(I),DFM(I),X(I+NO),X(I+NO2),X(I),CHDOWN(ICTL(I)+3)
110   FORMAT(I3,I3,F7.2,F5.2,F6.3,F8.6,F7.0,3F6.2,2F7.3,F8.2,1X,A8)
      IF(JUMP1) WRITE(IOUT,120) X(NO3-2),X(NO3-1),X(NO3)
120   FORMAT(' Hydraulic jump in Channel 1 with conjugate'' depths:',
     &2F10.3,/,' At position, x =',F10.1)
      END
      SUBROUTINE FUN(F)
      PARAMETER (N=12,M=39)
      EXTERNAL DYX
      LOGICAL*1 IPERM,NTRAN,UPSRES,JUMP,JMP
      REAL KL(N),L(N),LT(N),F(M),Y(1),DY(1),W(1,13),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NJ,NO,NO2,NEQS,ITYP(N),ICTL(N),JN(M),NI(N/2),
     &FN(N),SO(N),L,LT,B(N),FM(N),DZ(N),KL,DB(N),DFM(N),YG(N),
     &QOUT(N),X(M),H,G2,G,CC,TOL,PERM,TOPW,YO,QN,Q2G,BB,FMM,NO3,IJ,
     &IPERM,NTRAN,UPSRES,JUMP
      II=1
      IF(UPSRES) THEN
      BB=B(1)
      FMM=FM(1)
      F(II)=H-X(NO+1)-KL(1)*(X(1)/AR(1,X(NO+1)))**2
      II=II+1
      ENDIF
C Energy & Continuity Eqs.
      IPERM=.FALSE.
```

```
      DO 10 I=1,NJ
      I1=NI(I)+1
      I2=NI(I+1)
      JI=IABS(JN(I1))
      II1=II
      II=II+1
      F(II1)=X(JI)-QOUT(I)
      BB=B(JI)+DB(JI)
      FMM=FM(JI)+DFM(JI)
      EN=X(NO2+JI)+(X(JI)/AR(JI,X(NO2+JI)))**2/G2
      DO 10 J=I1+1,I2
      JI1=JN(J)
      JI=IABS(JI1)
      BB=B(JI)
      FMM=FM(JI)
      IIIJ=NO+JI
      IF(JI1.LT.0) IIIJ=IIIJ+NO
      F(II)=EN-X(IIIJ)-KL(JI)*(X(JI)/AR(JI,X(IIIJ)))**2-DZ(JI)
      II=II+1
10    F(II1)=F(II1)-X(JI)*FLOAT(JI1/JI)
C Uniform flow or GVF eqs
      IPERM=.TRUE.
      DO 20 I=1,NO
      IJ=I
      BB=B(I)
      FMM=FM(I)
      IF(ICTL(I).EQ.0) THEN
      F(II)=CC*AR(I,X(NO+I))**1.6666667*SQRT(SO(I))-FN(I)*
      &X(I)*PERM**.6666667
      ELSE
      IF(L(I)+LT(I).LT.1.E-5) GO TO 20
      JMP=.FALSE.
      IF(JUMP) THEN
      JMP=.TRUE.
      XX=0.
      XZ=X(NO3)
      Y(1)=YO
      ELSE
      XX=L(I)+LT(I)-1.E-5
      XZ=0.
      Y(1)=X(NO2+I)
      ENDIF
      IF(ICTL(I).EQ.2) Y(1)=1.1*Y(1)
      QN=(FN(I)*X(I)/CC)**2
      Q2G=X(I)**2/G
12    H1=-.05
      HMIN=.00001
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      IF(JMP) THEN
```

```
      F(II)=X(NO3-2)-Y(1)
      II=II+1
      YY1=X(NO3-2)
      YY2=X(NO3-1)
      BB=B(I)
      FMM=FM(I)
      F(II)=YY1*YY1*(.5*B(I)+YY1*FM(I)/3.)-YY2*YY2*(.5*B(I)
     &+YY2*FM(I)/3.)+X(I)**2/G*(1./AR(I,YY1)-1./AR(I,YY2))
      II=II+1
      XX=L(I)+LT(I)
      XZ=X(NO3)
      Y(1)=X(NO2+1)
      JMP=.FALSE.
      GO TO 12
      ENDIF
      IF(JUMP) THEN
      JUMP=.FALSE.
      F(II)=X(NO3-1)-Y(1)
      ELSE
      F(II)=X(NO+I)-Y(1)
      ENDIF
      ENDIF
      II=II+1
20    CONTINUE
C Downstream eqs
      DO 30 I=1,NO
      IF(ICTL(I).LE.0 .OR. ICTL(I).EQ.3) GO TO 30
      BB=B(I)
      FMM=FM(I)
      IF(ICTL(I).EQ.2) THEN
      IPERM=.TRUE.
      F(II)=X(I)**2*TOPW-G*AR(I,X(NO2+I))**3
      ELSE
      IPERM=.FALSE.
      F(II)=X(NO2+I)+(X(I)/AR(I,X(NO2+I)))**2/G2-YG(I)-(X(I)/
     &AR(I,YG(I)))**2/G2
      ENDIF
      II=II+1
30    CONTINUE
      NEQS=II-1
      RETURN
      END
      FUNCTION AR(I,YY)
      PARAMETER (N=12,M=39)
      LOGICAL*1 IPERM,NTRAN,UPSRES,JUMP
      REAL KL(N),L(N),LT(N)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NJ,NO,NO2,NEQS,ITYP(N),ICTL(N),JN(M),
     &NI(N/2),FN(N),SO(N),L,LT,B(N),FM(N),DZ(N),KL,DB(N),
```

```
&DFM(N),YG(N),QOUT(N),X(M),H,G2,G,CC,TOL,PERM,TOPW,YO,
&QN,Q2G,BB,FMM,NO3,IJ,IPERM,NTRAN,UPSRES,JUMP
 IF(ITYP(I).EQ.1) THEN
 AR=(BB+FMM*YY)*YY
 IF(IPERM) THEN
 PERM=BB+2.*YY*SQRT(1.+FMM*FMM)
 TOPW=BB+2.*FMM*YY
 ENDIF
 ELSE
 COSB=1.-2.*YY/BB
 BETA=ACOS(COSB)
 AR=.25*BB**2*(BETA-SIN(BETA)*COSB)
 IF(IPERM) THEN
 PERM=BETA*BB
 TOPW=BB*SIN(BETA)
 ENDIF
 ENDIF
 RETURN
 END
 SUBROUTINE DYX(XX,Y,DY)
 PARAMETER (N=12,M=39)
 LOGICAL*1 IPERM,NTRAN,UPSRES,JUMP
 REAL KL(N),L(N),LT(N),Y(1),DY(1)
 COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
 COMMON /TRAS/NJ,NO,NO2,NEQS,ITYP(N),ICTL(N),JN(M),
&NI(N/2),FN(N),SO(N),L,LT,B(N),FM(N),DZ(N),KL,DB(N),
&DFM(N),YG(N),QOUT(N),X(M),H,G2,G,CC,TOL,PERM,TOPW,YO,
&QN,Q2G,BB,FMM,NO3,IJ,IPERM,NTRAN,UPSRES,JUMP
 IF(XX.LE.L(IJ)) THEN
 NTRAN=.TRUE.
 DAX=0.
 ELSE
 NTRAN=.FALSE.
 FLEN=(XX-L(IJ))/LT(IJ)
 BB=B(IJ)+FLEN*DB(IJ)
 FMM=FM(IJ)+FLEN*DFM(IJ)
 DAX=X(IJ)**2/G*Y(1)*(DB(IJ)+Y(1)*DFM(IJ))/LT(IJ)
 ENDIF
 AA=AR(IJ,Y(1))
 A3=AA**3
 SF=QN*(ABS(PERM/AA)**.666666667/AA)**2
 DY(1)=(SO(IJ)-SF+DAX/A3)/(1.-Q2G*TOPW/A3)
 RETURN
 END
```

The potential unknowns in the program are contained in the array X(M), with the flow rates Q_i, stored first for the NO = number of channels; next the upstream depths Y_{i1}, and finally the downstream depths, Y_{i2}. Thus, for the above system Q_1, Q_2, …, Q_7 are in X(1), X(2), …, X(7), the upstream depths Y_{11}, Y_{21}, …, Y_{71} are in X(8), X(9), …, X(14), and the downstream depths Y_{12}, Y_{22}, …, Y_{72}

are in X(15), X(16), …, X(21). If a hydraulic jump occurs in channel 1, then the last three values of the array X are for (1) the depth immediately upstream of the jump Y_{1j}, (2) the depth immediately downstream from the jump Y_{2j}, and (3) the position of the jump x. If ITYP(I) is given a 1, then the channel is specified to be trapezoidal, if it is a 2 (or otherwise), the channel is specified as circular. The integer array ICTL(I) for each channel allows the downstream boundary condition for that channel to be communicated. If ICTL = −2, then two channels of the same size connect without any concentrated outflow here, and the downstream BC for this channel equates its downstream depth Y_{i2} to the upstream depth of the channel it connect with, $Y_{i+1,1}$. ICTL = −1 communicates that the downstream end of this channel branches into other channels, and therefore no downstream BC exists for this channel, but its specific energy is equated to the specific energies of the branched downstream channels. ICTL = 0 indicates a long channel that contains a uniform flow; ICTL = 1 indicates a gate at its downstream end, ICTL = 2 indicates that the critical depth exists at the downstream end; and ICTL = 3 indicates that the channel discharges into a reservoir.

The input data for this program consists of the following:

1. NJ = Number of junction, or branch points.
2. NO = Number of channels.
3. For each of the NO channels:
   (a) Its type, i.e. a 1 for trapezoidal or a 2 for circular (ITYP)
   (b) Its bottom width b, if trapezoidal or diameter D, if circular (B)
   (c) Its side slope m if trapezoidal, otherwise 0 (FM)
   (d) Its Manning's n (FN)
   (e) Its bottom slope So (SO)
   (f) Its length (L)
   (g) The length of the transition at its downstream end (LT)
   (h) The difference in elevation between this & the preceding channel (DZ)
   (i) The change in bottom width across the transition (DB)
   (j) The change in side slope across the transition (DFM)
   (k) The minor loss coefficient at the entrance of this channel (KL)
   (l) The downstream known depth, i.e. depth downstream from gate if ICTL=1, or reservoir water surface elevation if ICTL=3 (YG).
4. (a) The accuracy of the GVF-solution TOL, (b) the error criteria ERR for the Newton iteration, (c) the head H of the reservoir supplying the upstream channel, and (d) the acceleration of gravity G. If H = 0, then the flow rate in channel 1 is specified, and this flow rate Q_1 = X(1) is provided below as (6), i.e., the given value is now not an estimate. If H is given a negative value, then a hydraulic jump is specified to occur in channel 1 and the absolute value of H is the known depth at the beginning of channel 1. The flow rate given in (6) below is taken as the specified flow rate for channel 1.
5. For each junction the following:
   (a) The number of the channel at this junction JI
   (b) A list of these channel numbers with the upstream channel listed first JN
   (c) The concentrated outflow at this junction QOUT (inflow is negative) QOUT
6. For each channel, an estimate of (a) the flow rate, (b) the upstream depth Y_{i1}, and(c) the downstream depth if this is not specified as a reservoir or uniform depth.

In addition, if a hydraulic jump is specified in channel 1, then three more guess values (on the last line) for (a) the depth immediately upstream of the jump Y_{1j}, (b) the depth immediately downstream of the jump Y_{2j} i.e., the conjugate depth to Y_{1j}, and (c) the position x where the jump occurs must be given.

The input data for SOLGBR to define the problem involving the seven channels above could consist of the following:

```
3 7                                         ! No. of Junction, & No. of Channels Channel layout data:
1 5 1.5 .014 .0006 1500  0 0  0 0 .1  3. -1  ! (1) ITYP=1 trap. channel, ITYP=2 circular
2 3 0   .012 .0005 10000 0 .2 0 0 .08 3.  0  ! channel, (2) B=bottom width, or diameter, (3) FM=side
1 5 1.5 .014 .0006  500  0  0 0 0 .08 3. -1  ! slope, (4) FN=Mannings n, (5) SO=bottom slope,
1 3 0   .015 .0015 4000  0 .2 0 0 .08 .6  0  ! (6) L=length, (7) LT=length of transition at end of L,
1 5 1.5 .014 .0006  600  0  0 0 0 .08 3. -1  ! (8) DZ=Δz rise in bottom, (9) DB=change in bottom
1 3 0   .015 .0005 7000  0  0 0 0 .08 3.1 3  ! width across transition, (10) DFM=change in side slope
1 1.8 0 .014 .001 10000  0  0 0 0 .08 3.  0  ! across transition, (11) KL=minor loss coef.,
.0001 .001 3. 9.81                           ! (12) YG=downstr. depth, (13) ICTL=type downstr.
3 1 2 3 0.    ! TOL=for GVF solver,ERR=Newton accuracy, H=upstr. res. head,G=accel. of gravity.
3 3 4 5 0.    ! Channel connectivity: (1) No. of channels at junctions, (2) list of these channel
3 5 6 7 0.    ! numbers with upstream given first, (3) Concentrated outflow at junction
46.6 2.85 2.9 ! Guesses for unknowns. If ICTY=-2,0 or 3 (=-2 connects to single channel, =0 uniform flow,
11.7 2.7      ! 3=res. downstr.) then two values: (1) Flow rate, Qi, (2) upstr. depth, Yi1.
34.9 2.9 2.9  ! Otherwise three values: (1) Flow rate, Qi, (2) upstr. depth, Yi1, (3) downstream depth.
12.0 2.7
22.9 2.9 2.9
13.4 2.91
9.5 2.9
```

The resulting output and solution consist of:
Solution to seven channel problem

```
---------------------------------------------
No  Ty     b     m     n        So        L    dz
---------------------------------------------
 1  -1  5.00  1.50  .014  .000600   1500.  .00
 2   0  3.00   .00  .012  .000500  10000.  .20
 3  -1  5.00  1.50  .014  .000600    500.  .00
 4   0  3.00   .00  .015  .001500   4000.  .20
 5  -1  5.00  1.50  .014  .000600    600.  .00
 6   3  3.00   .00  .015  .000500   7000.  .00
 7   0  1.80   .00  .014  .001000  10000.  .00
----------------------------------------
db    dm     Yu     Yd      Q  Downstream
----------------------------------------
.00  .00  2.675  2.769  58.03      Branch
.00  .00  2.671  2.671  11.53     Uniform
.00  .00  2.870  3.073  46.50      Branch
.00  .00  2.653  2.653  19.97     Uniform
.00  .00  3.159  3.498  26.52      Branch
.00  .00  3.395  3.100  15.70    Reservo.
.00  .00  3.348  3.348  10.83     Uniform
```

If a hydraulic jump is specified to occur in channel 1 with the flow rate in this channel given as 50 m^3/s, the input could consist of the following: (Note the differences from above with no H.) The jump consists of: (1) giving a negative value to H (which is now the specified depth at the beginning of the upstream channel # 1, Y_{1o}), and (2) giving an additional line at the end of guesses that provides Y_{1j} = depth upstream of jump, i.e., depth at the end of M_3-GVF, Y_{2j} = depth after the H.jump and x = position of H.jump, and the specified flow rate in channel 1 is 50.0 m^3/s; the first line containing the guesses for the unknowns.

```
3 7 ! No. of Junction, & No. of Channels
1 5 1.5 .014 .0006  1500 0  0 0 0 .1  3. -1 ! Channel layout data: (1) ITYP=1 trap. channel, ITYP=2
2 3 0   .012 .0005 10000 0 .2 0 0 .08 3.  0 ! circular channel, (2) B=bottom width, or diameter,
1 5 1.5 .014 .0006   500 0  0 0 0 .08 3. -1 ! (3) FM=side slope, (4) FN=Mannings n, (5) SO=bottom
1 3 0   .015 .0015  4000 0 .2 0 0 .08 .6  0 ! slope, (6) L=length, (7) LT=length of transition at end
1 5 1.5 .014 .0006   600 0  0 0 0 .08 3. -1 ! of L, (8) DZ=Δz rise in bottom, (9) DB=change in bottom
1 3 0   .015 .0005  7000 0  0 0 0 .08 3.5 3 ! width across transition, (10) DFM=change in side slope
1 1.8 0 .014 .001  10000 0  0 0 0 .08 3.  0 ! across transition, (11) KL=minor loss coef.,
.0001 .001 −0.85 9.81      ! (12) YG=downstr. depth, (13) ICTL=type downstr.
3 1 2 3 0.     ! TOL=for GVF solver,ERR=Newton accuracy, H=upstr. depth(known),G=accel. of gravity.
3 3 4 5 0.     ! Channel connectivity: (1) No. of channels at junctions,
3 5 6 7 0.     ! (2) list of these channel numbers with upstream given first, (3) Concentrated outflow
50.0 2.85 2.9 ! at junction Guesses for unknowns. First value for Q1 now becomes the specified flow rate
12.  2.7       ! If ICTY=−2,0 or 3 (=−2 connects to single channel, =0 uniform flow,
38.0 2.9 2.9  ! 3=res. downstr.) then two values: (1) Flow rate, Qi, (2) upstr. depth, Yi1.
14.0 2.7 6.5  ! Otherwise three values: (1) Flow rate, Qi, (2) upstr. depth, Yi1, (3) downstream depth.
24.0 2.9 2.9
13.5 2.91
10.5 2.9
.94  3. 50.   ! depth before H. Jump Y1j, depth after H. Jump Y2j, and position of H. Jump, x.
```

and the solution is contained in the following table:

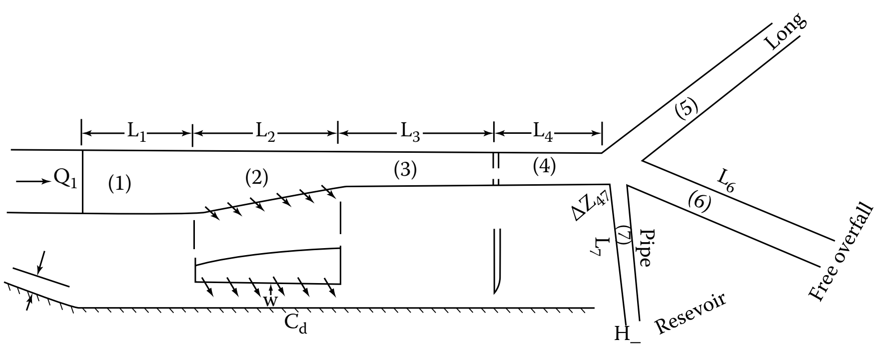
Solution to the seven channel problem

No	Ty	b	m	n	So	L	dz
1	-1	4.00	1.50	.014	.000600	1500.	.00
2	0	3.00	.00	.012	.000500	10000.	.20
3	-1	4.00	1.50	.014	.000600	500.	.00
4	1	3.00	.00	.015	.001500	4000.	.20
5	-1	4.00	1.50	.014	.000600	600.	.00
6	3	3.00	.00	.015	.000500	7000.	.00
7	0	1.80	.00	.012	.001000	10000.	.00

db	dm	Yu	Yd	Q	Downstream
.00	.00	.850	2.257	50.00	Branch
.00	.00	2.347	2.347	10.39	Uniform
.00	.00	2.480	2.568	39.61	Branch
.00	.00	2.242	4.705	16.29	Gate
.00	.00	2.702	3.020	23.33	Branch
.00	.00	2.944	3.500	12.91	Reservo.
.00	.00	2.832	2.832	10.42	Uniform

```
Hydraulic jump in Channel 1 with conjugate depths: 1.410 2.617
At position, x = 216.7
```

Let us consider an additional channel system, shown below, to clarify and to reinforce the general principles for solving such problems. If one does not recognize that the gate between channels 3 and 4 divides this system into two separate problems, a larger system of simultaneous equations will be solved than necessary. The equations for the two separate problems that solve this channel system are



Problem # 1: Seven unknowns: Q_3, Y_{1j1}, Y_{1j2}, x_1, Y_{12}, Y_{31}, Y_{32}

$$F_1 = Q_1 - \int q^* dx - Q_3 = 0 \quad \text{(Continuity)}$$

$$F_2 = Y_{32} + \frac{Q_3^2}{2gA_{32}^2} - Y_{41} - \frac{Q_3^2}{2gA_{41}^2} = 0 \;\text{ in which } Y_{41} = C_c Y_G \quad \text{(Energy)}$$

$$F_3 = \frac{b_1Y_{1j}^2}{2} + \frac{m_{1j}Y_{1j}^3}{3} + \frac{Q_1^2}{gA_{1j}} - \frac{b_1Y_{2j}^2}{2} - \frac{m_{1j}Y_{2j}^3}{3} - \frac{Q_2^2}{gA_{2j}} = 0 \quad \text{(Momentum)}$$

$$F_4 = Y_{1j} - Y_{GVF}(Y_{1o}) = 0 \quad \text{(GVF profile, upstream jump)}$$

$$F_5 = Y_{2j} - Y_{GVF}(Y_{12}) = 0 \quad \text{(GVF profile, downstream jump)}$$

$$F_6 = Y_{12} - Y_{GVF}(Y_{31}) = 0 \quad \text{(ODE solution that includes outflow and nonprismatic ch.)}$$

$$F_7 = Y_{31} - Y_{GVF}(Y_{32}) = 0 \quad \text{(GVF profile in channel 3)}$$

Problem # 2: 11 unknowns: Q_5, Q_6, Q_7, Y_{4j1}, Y_{4j2}, x_4, Y_{42}, Y_{51}, Y_{61}, Y_{62}, Y_{71}

$$F_1 = Q_3 - Q_5 - Q_6 - Q_7 = 0 \quad \text{(Continuity)}$$

$$F_2 = Y_{42} + \frac{Q_3^2}{2gA_{42}^2} - Y_{51} - \frac{Q_5^2}{2gA_{51}^2} = 0 \quad \text{(Energy)}$$

$$F_3 = Y_{42} + \frac{Q_3^2}{2gA_{42}^2} - Y_{61} - \frac{Q_6^2}{2gA_{61}^2} = 0 \quad \text{(Energy)}$$

$$F_4 = Y_{42} + \frac{Q_3^2}{2gA_{42}^2} - Y_{71} - \frac{Q_7^2}{2gA_{71}^2} - \Delta z_{47} = 0 \quad \text{(Energy)}$$

$$F_5 = \frac{b_4Y_{4j1}^2}{2} + \frac{m_4Y_{4j1}^3}{3} + \frac{Q_3^2}{gA_{3j1}} - \frac{b_4Y_{4j2}^2}{2} - \frac{m_4Y_{4j2}^3}{3} - \frac{Q_3^2}{gA_{4j2}} = 0 \quad \text{(Momentum)}$$

$$F_6 = Y_{4j1} - Y_{GVF}(C_cY_G) = 0 \quad \text{(GVF profile, upstream jump)}$$

$$F_7 = Y_{4j2} - Y_{GVF}(Y_{42}) = 0 \quad \text{(GVF profile, downstream jump)}$$

$$F_8 = n_5Q_5P_5^{2/3} - C_uA_5^{5/3}S_o^{1/2} = 0 \quad \text{(Manning's equation)}$$

$$F_9 = Y_{61} - Y_{GVF}(Y_{62}) = 0 \quad \text{(GVF profile in channel 6)}$$

$$F_{10} = Q_6^2T_6 - gA_6^3 = 0 \quad \text{(Critical Flow Equation, end channel 6)}$$

$$F_{11} = Y_{71} - Y_{GVF}(H_7) = 0 \quad \text{(GVF profile in channel 7)}$$

Observe the following: (1) The first continuity equation is not obtained from a junction but rather across the lateral outflow channel 2 and the latter outflow integral must be obtained as part of the GVF solution across this channel. (2) Since no concentrated outflow occurs at the junction between channels 1 and 2, and 2 and 3, continuity equations are not written for these junctions; rather $Q_2 = Q_1$ (known) and Q_2 (end) = Q_3, which in a sense are these junction continuity equations. (The computer program SOLGBRO.FOR listed below, however, adds these two equations to the system of equations.) (3) Since at the above three junctions the channel sizes are the same, and no concentrated outflow occurs, the specific energy equations are not used, but rather $Y_{12} = Y_{21}$ and $Y_{22} = Y_{31}$. (However, in the computer program below these specific energy equations are used. It handles the outflow by placing $\int q^*dx$ in QOUT at junction 1, so the first equation becomes $Q_1 - Q_2 - QOUT(1) = 0$.)

The previous computer program SOLGBA.FOR will not solve Problem # 1 because it contains a lateral outflow, and the program does not include this capability. The program listed below, SOLGBRO.FOR allows a lateral outflow from, or inflow to, selected channels in the system. The needed statement to accommodate these flows have been added to SOLGBR.FOR. If a lateral outflow/inflow occurs, ICTL is given as –3 for the channel whose end is the position where such a spatially varied flow begins. In the equations listed for Problem # 1 to the seven channel system above, the continuity equation at junction #1 is listed first as equation F_1. It is important in solving this system of equations, however, to note that the outflow $\int q^*dx$ is actually determined by equation F_6, and therefore in implementing the solution in a computer program, F_1 must be evaluated after F_6. Therefore, SOLGBRO.FOR moves the statements in SUBROUTINE FUN that evaluate the continuity and the energy equation at the junctions after those that evaluate the GVF equations.

Program SOLGBRO.FOR designed to solve branched channel systems with a section of lateral outflow or inflow

```
      PARAMETER (N=12,M=39,N2=8)
      LOGICAL*1 TRUX(M)
      LOGICAL*1 IPERM,NTRAN,UPSRES,JUMP,JUMP1,LSTA,INF
      REAL F(M),F1(M),D(M,M),KL(N),L(N),LT(N),SQOUT(3)
      INTEGER*2 INDX(M)
      CHARACTER*8 CHDOWN(0:6)/' Spatial','Branch 1',' Branch',
     &' Uniform','  Gate','Critical','Reservo.'/
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/FN(N),SO(N),L,LT,B(N),FM(N),DZ(N),KL,DB(N),
     &DFM(N),YG(N),QOUT(N),X(M),WSTA(3),CD(3),H,G2,G,CC,TOL,
     &PERM,TOPW,YO,QN,Q2G,BB,FMM,SQSTAR,XO,QSTA1,ITYP(N),
     &ICTL(N),JN(M),NI(N2),ISTA(4),NODEU(3),NJ,NO,NO2,NO3,NEQS,
     &NSTA,JSTA,IJ,IPERM,NTRAN,UPSRES,JUMP,LSTA,INF
      DATA IN,IOUT/2,3/
      IPERM=.FALSE.
      NTRAN=.TRUE.
      UPSRES=.TRUE.
C ICTL = -3,-2,-1,0,1,2 OR 3 for types of downstream controls,
C -3=If downstr. end of channel is start of lateral
C  outflow/inflow.
C -2=If downstr. branch connects to single ch. of same
C  size(Yi=Yi+1) and QOUT=0,
C i.e. Ei=Ei+1 is not availablae.
C -1=Downstream end of channel is branch.
C 0=uniform, 1=gate, 2=critical, 3=reser.
C YG is depth behind gate for gate; = 0 for critical;=res.
C  depth if reservoir.
C DB and DFM are changes (+ or -) across transition of
C  b and m.
C IN UNKNOWN VECTOR X, Q's COME 1ST; UPSTREAM DEPTHS NEXT &
```

```
C  then DOWNS. DEPTHS
C AND IF JUMP IN CH. # 1 THEN Yj1,Yj2 & x, are last three
C  unknowns.
C i.e. Q(I)=X(I);YU(I)=X(I+NO); YG(I)=X(I+NO2); [Yj1,Yj2,x]
      READ(IN,*) NJ,NO,(ITYP(I),B(I),FM(I),FN(I),SO(I),L(I),
     &LT(I),DZ(I),DB(I),DFM(I),KL(I),YG(I),ICTL(I),I=1,NO)
      NO2=2*NO
      NO3=3*NO
      YO=0.
      JUMP=.FALSE.
      INF=.FALSE.
      DO 1 I=1,NO3+3
1     TRUX(I)=.TRUE.
C If Q1 is specified give 0 for H, and if jump occurs then
C   give ups. supercr.
C  depth for H but precede with a minus sign.
C If CD is negative then inflow case & magnitude of CD is
C  lateral inflow q* &
C WSTA=comp. vel. Uq, otherwise WSTA is height of weir.
C If lateralinfow/outflow X(I)=flow rate at end of this
C  section.
      READ(IN,*) TOL,ERR,H,G,NSTA,(ISTA(I),WSTA(I),CD(I),I=1,NSTA)
      ISTA(NSTA+1)=100
      IUPS=1
      IF(H.LT. 1.E-5) THEN
      UPSRES=.FALSE.
      IUPS=2
      TRUX(1)=.FALSE.
      IF(H.LT.-1.E-5) THEN
      YO=ABS(H)
      TRUX(NO+1)=.FALSE.
      JUMP=.TRUE.
      ENDIF
      ENDIF
      IF(G.GT. 30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      NI(1)=0
      II=1
      NCT=0
      DO 2 J=1,NJ
C In giving channel nos. that join at junction the first listed
C will have its flow rate positive, and all other will have their
C flow rate subtracted therefrom unless preceded by a minus, i.e.
C the energy eqs. are from downstream of first to upstream of
C other numbers or E(1)downst=E(I)upst, unless preceded by a
C minus, in which case E(1)downstr=E(I)downstr.
      READ(IN,*) JI,(JN(I),I=II,II+JI-1),QOUT(J)
```

```
      IF(NSTA.GT.NCT) THEN
      IF(ICTL(JN(II)).EQ.-3) THEN
      NCT=NCT+1
      NODEU(NCT)=J
      ENDIF
      ENDIF
      II=II+JI
2     NI(J+1)=II-1
      DO 3 I=1,NSTA
      IF(CD(I).GT.0.) CD(I)=.666667*CD(I)*SQRT(G2)
3     CONTINUE
      DO 4 I=1,NO
      IF(ICTL(I).EQ.3) X(I+NO2)=YG(I)
4     CONTINUE
      DO 6 I=1,NO
      KL(I)=(KL(I)+1.)/G2
      IF(ICTL(I).EQ.3 .OR. ICTL(I).EQ.0) THEN
      READ(IN,*) X(I),X(I+NO)
      IF(ICTL(I).EQ.0) X(I+NO2)=X(I+NO)
      TRUX(I+NO2)=.FALSE.
      ELSE
      READ(IN,*) X(I),X(I+NO),X(I+NO2)
      ENDIF
6     CONTINUE
      IF(YO.GT.0.) THEN
C Guess for 1-depth upst. jump, 2-downst.jump, & 3-position
C  of jump, x
      READ(IN,*) X(NO3+1),X(NO3+2),X(NO3+3)
      NO3=NO3+3
      X(NO+1)=YO
      ENDIF
      NCT=0
      JUMP1=JUMP
15    JUMP=JUMP1
      LSTA=.FALSE.
      CALL FUN(F)
      DO 16 I=1,NSTA
16    SQOUT(I)=QOUT(NODEU(I))
      J=0
      DO 20 II=IUPS,NO3
      IF(.NOT.TRUX(II)) GO TO 20
      J=J+1
      DX=.005*X(II)
      X(II)=X(II)+DX
      JUMP=JUMP1
      CALL FUN(F1)
      DO 17 I=1,NEQS
17    D(I,J)=(F1(I)-F(I))/DX
      X(II)=X(II)-DX
      DO 18 I=1,NSTA
18    QOUT(NODEU(I))=SQOUT(I)
```

```
20    CONTINUE
      CALL SOLVEQ(NEQS,1,M,D,F,1,DD,INDX)
      DIF=0.
      J=0
      DO 30 I=IUPS,NO3
      IF(TRUX(I)) THEN
      J=J+1
      X(I)=X(I)-F(J)
      DIF=DIF+ABS(F(J))
      ELSE
      IF(I.EQ.NO+1 .AND. IUPS.EQ.2) GO TO 30
      IF(ICTL(I-NO2).EQ.0) X(I)=X(I-NO)
      IF(ICTL(I-NO2).LT.-1) X(I)=X(I-NO+1)
      ENDIF
30    CONTINUE
      NCT=NCT+1
      WRITE(*,200) NCT,DIF,(X(I),I=1,NO3)
200   FORMAT(' NCT=',I3,' SUM=',E12.4,/,(10F8.2))
      IF(NCT.LT.30 .AND. DIF.GT.ERR) GO TO 15
      WRITE(IOUT,100) NO
100   FORMAT(' Solution to',I3,' Channel Problem',/,1X,90('-'),/,' No
     &Ty  b  m  n  So',' L  dz  db  dm  Yu  Yd',' Q Downstream',/,
     &1X,90('-'))
      DO 50 I=1,NO
50    WRITE(IOUT,110) I,ICTL(I),B(I),FM(I),FN(I),SO(I),L(I),DZ(I),
     &DB(I),DFM(I),X(I+NO),X(I+NO2),X(I),CHDOWN(ICTL(I)+3)
110   FORMAT(I3,I3,F7.2,F5.2,F6.3,F8.6,F7.0,3F6.2,2F7.3,F8.2,1X,A8)
      IF(JUMP1) WRITE(IOUT,120) X(NO3-2),X(NO3-1),X(NO3)
120   FORMAT(' Hydraulic jump in Channel 1 with conjugate depths:',
     &2F10.3,/,' At position, x =',F10.1)
      END
      SUBROUTINE FUN(F)
      PARAMETER (N=12,M=39,N2=8)
      EXTERNAL DYX
      LOGICAL*1 IPERM,NTRAN,UPSRES,JUMP,JMP,LSTA,INF
      REAL KL(N),L(N),LT(N),F(M),Y(1),DY(1),W(1,13),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/FN(N),SO(N),L,LT,B(N),FM(N),DZ(N),KL,DB(N),
     &DFM(N),YG(N),QOUT(N),X(M),WSTA(3),CD(3),H,G2,G,CC,TOL,
     &PERM,TOPW,YO,QN,Q2G,BB,FMM,SQSTAR,XO,QSTA1,ITYP(N),
     &ICTL(N),JN(M),NI(N2),ISTA(4),NODEU(3),NJ,NO,NO2,NO3,NEQS,
     &NSTA,JSTA,IJ,IPERM,NTRAN,UPSRES,JUMP,LSTA,INF
      JSTA=0
      II=1
      IF(UPSRES) THEN
      BB=B(1)
      FMM=FM(1)
      F(II)=H-X(NO+1)-KL(1)*(X(1)/AR(1,X(NO+1)))**2
      II=II+1
      ENDIF
C Uniform flow or GVF eqs
```

```
      IPERM=.TRUE.
      DO 20 I=1,NO
      IJ=I
      BB=B(I)
      FMM=FM(I)
      IF(ICTL(I).EQ.0) THEN
      F(II)=CC*AR(I,X(NO+I))**1.6666667*SQRT(SO(I))-FN(I)*
     &X(I)*PERM**.6666667
      ELSE
      IF(L(I)+LT(I).LT.1.E-5) GO TO 20
      JMP=.FALSE.
      IF(JUMP) THEN
      JMP=.TRUE.
      XX=0.
      XZ=X(NO3)
      Y(1)=YO
      GO TO 12
      ELSE
      XX=L(I)+LT(I)-1.E-5
      IF(ISTA(JSTA+1).GT.I) GO TO 10
      JSTA=JSTA+1
      LSTA=.TRUE.
      IF(CD(JSTA).LT.0.) THEN
      INF=.TRUE.
      SQSTAR=XX*CD(JSTA)
      ELSE
      INF=.FALSE.
      SQSTAR=0.
      IF(X(NO2+I).LT.WSTA(JSTA)) THEN
      QSTA1=0.
      ELSE
      QSTA1=CD(JSTA)*(X(NO2+I)-WSTA(JSTA))**1.5
      ENDIF
      ENDIF
      XO=XX
      XZ=XX-1.
      GO TO 11
10    XZ=0.
11    Y(1)=X(NO2+I)
      ENDIF
12    IF(ICTL(I).EQ.2) Y(1)=1.15*Y(1)
      QN=(FN(I)*X(I)/CC)**2
      Q2G=X(I)**2/G
13    H1=-.05
      HMIN=.00001
14    CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      IF(XZ.LT. .001 .OR.JUMP) GO TO 16
      IF(INF) GO TO 15
      IF(Y(1).LT. WSTA(JSTA)) THEN
      QSTA2=0.
      ELSE
```

```
      QSTA2=CD(JSTA)*(Y(1)-WSTA(JSTA))**1.5
      ENDIF
      SQSTAR=SQSTAR+.5*(QSTA1+QSTA2)
      QSTA1=QSTA2
15    XX=XZ
      XO=XZ
      XZ=XZ-1.
      IF(XZ.GT. .001) THEN
      GO TO 14
      ELSE
      QOUT(NODEU(JSTA))=SQSTAR
      ENDIF
16    IF(JMP) THEN
      F(II)=X(NO3-2)-Y(1)
      II=II+1
      YY1=X(NO3-2)
      YY2=X(NO3-1)
      BB=B(I)
      FMM=FM(I)
      F(II)=YY1*YY1*(.5*B(I)+YY1*FM(I)/3.)-YY2*YY2*(.5*B(I)+
     &YY2*FM(I)/3.)+X(I)**2/G*(1./AR(I,YY1)-1./AR(I,YY2))
      II=II+1
      XX=L(I)+LT(I)
      XZ=X(NO3)
      Y(1)=X(NO2+1)
      JMP=.FALSE.
      GO TO 13
      ENDIF
      IF(JUMP) THEN
      JUMP=.FALSE.
      F(II)=X(NO3-1)-Y(1)
      ELSE
      F(II)=X(NO+I)-Y(1)
      ENDIF
      ENDIF
      LSTA=.FALSE.
      II=II+1
20    CONTINUE
C Energy & Continuity Eqs.
      IPERM=.FALSE.
      DO 28 I=1,NJ
      I1=NI(I)+1
      I2=NI(I+1)
      JI=JN(I1)
      II1=II
      II=II+1
      F(II1)=X(JI)-QOUT(I)
      BB=B(JI)+DB(JI)
      FMM=FM(JI)+DFM(JI)
      EN=X(NO2+JI)+(X(JI)/AR(JI,X(NO2+JI)))**2/G2
      DO 28 J=I1+1,I2
```

```
      JI=JN(J)
      QQ=X(JI)
      DO 25 KI=1,NSTA
      IF(NODEU(KI).EQ.I) QQ=X(JN(I1))
25    CONTINUE
      BB=B(JI)
      FMM=FM(JI)
      IIIJ=NO+JI
      IF(JI1.LT.0) IIIJ=IIIJ+NO
      F(II)=EN-X(IIIJ)-KL(JI)*(QQ/AR(JI,X(IIIJ)))**2-DZ(JI)
      II=II+1
28    F(II1)=F(II1)-X(JI)
C Downstream eqs
      DO 30 I=1,NO
      IF(ICTL(I).LE.0 .OR. ICTL(I).EQ.3) GO TO 30
      BB=B(I)+DB(I)
      FMM=FM(I)+DFM(I)
      IF(ICTL(I).EQ.2) THEN
      IPERM=.TRUE.
      AA=AR(I,X(NO2+I))
      F(II)=X(I)**2*TOPW-G*AA**3
      ELSE
      IPERM=.FALSE.
      F(II)=X(NO2+I)+(X(I)/AR(I,X(NO2+I)))**2/G2-YG(I)-
     &(X(I)/AR(I,YG(I)))**2/G2
      ENDIF
      II=II+1
30    CONTINUE
      NEQS=II-1
      RETURN
      END
      FUNCTION AR(I,YY)
      PARAMETER (N=12,M=39,N2=8)
      LOGICAL*1 IPERM,NTRAN,UPSRES,JUMP,LSTA,INF
      REAL KL(N),L(N),LT(N)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/FN(N),SO(N),L,LT,B(N),FM(N),DZ(N),KL,DB(N),
     &DFM(N),YG(N),QOUT(N),X(M),WSTA(3),CD(3),H,G2,G,CC,TOL,
     &PERM,TOPW,YO,QN,Q2G,BB,FMM,SQSTAR,XO,QSTA1,ITYP(N),
     &ICTL(N),JN(M),NI(N2),ISTA(4),NODEU(3),NJ,NO,NO2,NO3,NEQS,
     &NSTA,JSTA,IJ,IPERM,NTRAN,UPSRES,JUMP,LSTA,INF
      IF(ITYP(I).EQ.1) THEN
      AR=(BB+FMM*YY)*YY
      IF(IPERM) THEN
      PERM=BB+2.*YY*SQRT(1.+FMM*FMM)
      TOPW=BB+2.*FMM*YY
      ENDIF
      ELSE
      COSB=1.-2.*YY/BB
      BETA=ACOS(COSB)
      AR=.25*BB**2*(BETA-SIN(BETA)*COSB)
```

```
      IF(IPERM) THEN
      PERM=BETA*BB
      TOPW=BB*SIN(BETA)
      ENDIF
      ENDIF
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      PARAMETER (N=12,M=39,N2=8)
      LOGICAL*1 IPERM,NTRAN,UPSRES,JUMP,LSTA,INF
      REAL KL(N),L(N),LT(N),Y(1),DY(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/FN(N),SO(N),L,LT,B(N),FM(N),DZ(N),KL,DB(N),
     &DFM(N),YG(N),QOUT(N),X(M),WSTA(3),CD(3),H,G2,G,CC,TOL,
     &PERM,TOPW,YO,QN,Q2G,BB,FMM,SQSTAR,XO,QSTA1,ITYP(N),
     &ICTL(N),JN(M),NI(N2),ISTA(4),NODEU(3),NJ,NO,NO2,NO3,NEQS,
     &NSTA,JSTA,IJ,IPERM,NTRAN,UPSRES,JUMP,LSTA,INF
      IF(XX.LE.L(IJ)) THEN
      NTRAN=.TRUE.
      DAX=0.
      ELSE
      NTRAN=.FALSE.
      FLEN=(XX-L(IJ))/LT(IJ)
      BB=B(IJ)+FLEN*DB(IJ)
      FMM=FM(IJ)+FLEN*DFM(IJ)
      DAX=Y(1)*(DB(IJ)+Y(1)*DFM(IJ))/LT(IJ)
      ENDIF
      AA=AR(IJ,Y(1))
      IF(LSTA) THEN
      IF(INF) THEN
      QQ=X(IJ)+(L(IJ)+LT(IJ)-XX)*CD(JSTA)
      FQ=(QQ/AA-WSTA(JSTA))*CD(JSTA)/(G*AA)-(.5*BB+FMM*
     &Y(1)/3.)*(Y(1)/AA)**2*DAX+QQ*CD(JSTA)/(G*AA**2)
      ELSE
      IF(Y(1).LT.WSTA(JSTA)) THEN
      QSTA=0.
      ELSE
      QSTA=CD(JSTA)*(Y(1)-WSTA(JSTA))**1.5
      ENDIF
      QQST=.5*(QSTA+QSTA1)
      QQ=X(IJ)+SQSTAR+ABS(XO-XX)*QQST
      FQ=QQ*QQST/(G*AA**2)
      ENDIF
      ELSE
      FQ=0.
      ENDIF
      A3=AA**3
      SF=QN*(ABS(PERM/AA)**.666666667/AA)**2
      DY(1)=(SO(IJ)-SF+DAX/A3+FQ)/(1.-Q2G*TOPW/A3)
      RETURN
      END
```

In addition to adding the –3 to the list of values that ICTL can have, the input from the second READ statement has the following added variables:

NSTA = No. of channels that have lateral outflow/inflow along their lengths, and then for each of these
ISTA = the channel number
WSTA = the height of the side weir above the channel bottom, if a lateral outflow occurs, or = the component of velocity U_q in the direction of the channel flow if a lateral inflow occurs.
CD = the discharge coefficient from the side weir if a lateral outflow occurs, or equals minus the lateral inflow per unit length q* if a lateral inflow occurs. If CD is negative, then SOLGBRO.FOR distinguishes that this is a section of a lateral inflow rather than an outflow.

To solve Problem #1 of the above seven channel system, the input to SOLGBRO is given below. (This input data contains the bottom widths, the side slopes, Manning's n's, bottom slopes, lengths, etc. that were used in solving this problem.)

Input to solve Problem #1 of the seven channel system

```
2 3
1 4 2.5 .014 .0006 1000 0. 0. 0. 0. 0. 3. -3
1 4 2.5 .014 .0006 0. 50. 0. -1. -1.5 0. 3. -2
1 3 1 .014 .0006 1500 0. 0. 0. 0. 0. 1.5 1
.000001 .001 -1.4 9.81 1 2 2.4 .45
2 1 2 39.
2 2 3 0.
89.2 3.1 3.1
50.2 3.1 3.1
50.2 3.1 3.2
1.812 3 200.
```

The solution is provided in the following table.

Solution to Problem #1 of the seven channel system

Solution to the three channel problem

```
---------------------------------------------
No  Ty     b     m     n       So      L   dz
---------------------------------------------
 1  -3  4.00  2.50  .014  .000600  1000.  .00
 2  -2  4.00  2.50  .014  .000600     0.  .00
 3   1  3.00  1.00  .014  .000600  1500.  .00

-------------------------------------------
 db     dm     Yu     Yd      Q  Downstream
-------------------------------------------
.00    .00  1.400  3.038  89.20     Spatial
.00  -1.50  3.038  3.291  45.79    Branch 1
.00    .00  3.291  3.666  45.79        Gate

Hydraulic jump in Channel 1 with conjugate depths: 1.781 3.042
At position, x = 157.9
```

Taking the flow rate past the gate, i.e., $Q_3 = Q_4 = 50.2\,m^3/s$ from the above solution, Problem #2 for the above seven channel system can be defined and solved. The input for either SOLGBR or SOLGBRO is given below, as well as the solution obtained therefrom. (Again the channel's geometries, etc. are given in the input data.)

Input data to SOLGBR or SOLGBRO to solve Problem #2 of above seven channel system

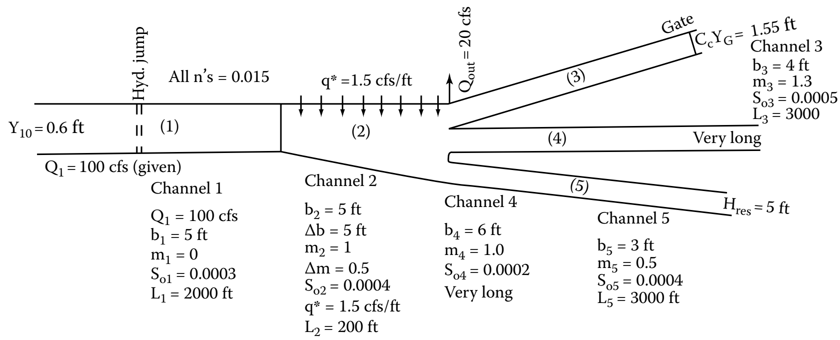
```
1 4
1 3 1 .014 .0004 1200 0 0 0 0 0 3.0 -1
1 2. .5 .014 .00038 10000 0 0 0 0 0 3.09 0
1 2. .5 .014 .0005 2000 0 0 0 0 0 1.9 2
2 3 0 .014 .0005 1500 0 .5 0 0 0 2.0 3
.0000001 .05 -1.5 9.81 0 0 0 0
4 1 2 3 4 0.
45.79 2.9 2.9
18.7 3.09
17.35 2.9 1.9
9.74 2.4
1.8 2.9 50
```

Solution to SOLGBR or SOLGBRO to solve Problem #2 of the above seven channel system

Solution to the four channel problem

```
----------------------------------------------
No  Ty     b     m     n       So        L   dz
----------------------------------------------
 1  -1  3.00  1.00  .014  .000400   1200.  .00
 2   0  2.00   .50  .014  .000380  10000.  .00
 3   2  2.00   .50  .014  .000500   2000.  .00
 4   3  3.00   .00  .014  .000500   1500.  .50

--------------------------------------
db   dm    Yu    Yd     Q   Downstream
--------------------------------------
.00 .00 1.500 2.576 45.79       Branch
.00 .00 2.968 2.968 16.24      Uniform
.00 .00 2.910 1.797 19.10     Critical
.00 .00 2.448 2.000 10.45     Reservo.

Hydraulic jump in Channel 1 with conjugate depths: 1.458 3.204
At position, x = -14.3
```

As an example of a problem with a lateral inflow, consider the system of the five channels below. Upstream from the beginning channel #1, the channel is steep and contains a flow rate $Q_1 = 100$ cfs at a supercritical depth of $Y_{1o} = 0.6$ ft. Downstream therefrom at a distance of 2000 ft a lateral inflow of $q^* = 1.5$ cfs/ft of length occurs over the next 200 ft. This incoming flow has a velocity $U_q = 10$ fps in the direction of the channel flow. Over this inflow section, identified as channel #2, the bottom width increases from 5 to 10 ft, and the side slope of the channel increases from 1.0 to 1.5. At the second junction, a pipeline extracts 20 cfs from the channel system.



The input data to SOLGBRO and the solution is provided below.

Input data to the five channel system

```
2 5
1 5 0 .015 .0003 2000 0 0 0 0 0 5 -3
1 5 1 .015 .0004 0 200 0 5 .5 0 5 -2
1 4 1.3 .015 .0005 3000 0 0 0 0 0 1.55 1
1 6 1 .015 .0002 10000 0 0 0 0 0 5 0
1 5 .5 .015 .0004 2500 0 0 0 0 0 5. 3
.0000001 .01 -0.6 32.2 1 2 10 -1.5
2 1 2 0
4 2 3 4 5 20.
100 5 5
400 5 5
150 5 5
150 5
80 5
.8 5.2 200
```

Solution to the five channel system with a lateral inflow

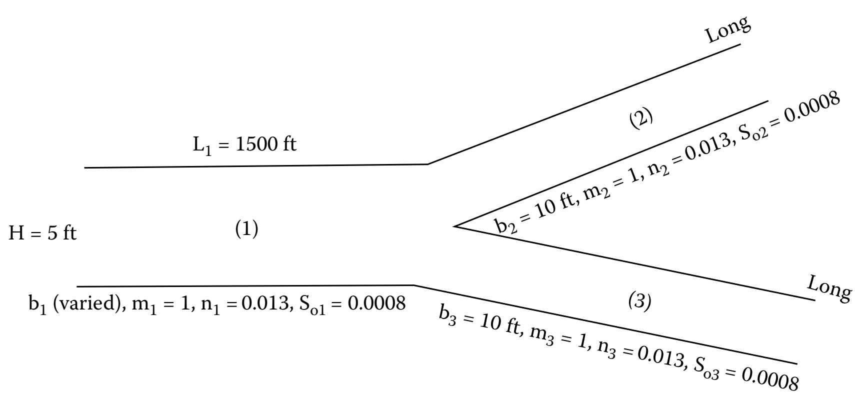
Solution to the five channel problem

```
----------------------------------------------
No  Ty     b     m     n       So      L    dz
----------------------------------------------
 1  -3  5.00   .00  .015  .000300   2000.  .00
 2  -2  5.00  1.00  .015  .000400      0.  .00
 3   1  4.00  1.30  .015  .000500   3000.  .00
 4   0  6.00  1.00  .015  .000200  10000.  .00
 5   3  5.00   .50  .015  .000400   2500.  .00
-------------------------------------------
  db   dm     Yu     Yd       Q  Downstream
-------------------------------------------
 .00  .00   .600  5.429  100.00     Spatial
5.00  .50  5.596  4.267  400.00    Branch 1
 .00  .00  4.611  5.449  149.52        Gate
 .00  .00  4.666  4.666  131.56     Uniform
 .00  .00  4.643  5.000   98.91    Reservo.

Hydraulic jump in Channel 1 with conjugate depths: .617 6.043
At position, x = 3.3
```

Note in this problem that the side slope of channel (2) at its upstream end is $m_2 = 0.5$ but that channel (1) at its downstream end has a side slope of m = 0 (i.e., is rectangular). Thus, there is actually a transition between channel (1) and (2). This could be handle by designating a length of transition, i.e., give LT different from zero in the input, in which case the GVF solution proceeds through the transition, and then the rest of the channel. It can also be handled as is in this input data, by letting the energy equation account for the different channel sizes, and using a minor loss coefficient, which is given as zero for this problem. Note that channel 2 has a zero length in the input but its transition length is given as 200 ft.

In providing the initial guess values for the Newton method to start the solution for multiple channel problems, such as the above examples, it is important that they are reasonably good. These often require more than just good judgement in estimating unknown variables from those that are known. It is always good to solve the normal and critical depths associated with the assumed flow rates so that it is possible to predict what type of GVF profiles will occur and approximately what the depths will be. The need for a "good" guess is particularly true if any of the depths are close to the critical depth. It will be more difficult to get a solution for a problem in which one or more downstream channels end in a free overfall that produces a critical flow, than for problems with other downstream boundary conditions. If the GVF computations begin below the critical depth during any iteration, the Newton iteration will likely fail. Problems involving hydraulic jumps are also more difficult to solve. For example, if the position of the hydraulic jump becomes negative, or moves into the next channel, it may be that the problem is not properly defined, e.g., the gate upstream may be submerged and the upstream flow is actually subcritical. If the channels that branch from a single channel have a composite capacity that is considerable larger than the upstream channel, they may cause a critical flow in it. If this occurs, a control is established at this point, and the problem will need to be separated into two problems. You might try doubling the size of the channel downstream from the junction in Problem #2 of the seven channel system to appreciate how this condition may occur. In brief, it would take considerable additional logic to that in program SOLGBRO.FOR to provide solutions to all problems involving multiple channels under an uneducated use on the part of the user. If in attempting to obtain a solution, it is observed that the depth at the downstream end of a channel during a Newton iteration gives values considerably less than the upstream depths of the channel this channel supplies, and then the next Newton iteration produces a large residual, and/or fails to converge, then it is likely that the upstream channel limits the flow rate, e.g., critical flow in the upstream channel control. In other words, there is no solution to the given equations. If upon increasing the size of the upstream channel a solution results, then the problem needs to be posed as one in which the flow rate is specified in the upstream channel equal to its critical flow under the specific energy available.

Consider a simple example of an upstream channel branching into two identical downstream channels to help illustrate how having more capacity downstream results in a critical flow in the upstream



channel that limits the flow rate. In this example, the two downstream channels have $b_2 = b_3 = 10$ ft, with side slopes $m_2 = m_3 = 1$, bottom slopes $S_{o2} = S_{o3} = 0.0008$, and Manning's $n_2 = n_3 = 0.013$. Both of these channels are assumed to be long, so uniform flow will always exit in them. The upstream channel is supplied by a reservoir with a head H = 5 ft ($K_e = 0$) and is 1500 ft long, with $m_1 = 1$, $S_{o1} = 0.0008$, and $n_1 = 0.013$. A series of solutions will be obtained in which the bottom width of the upstream channel will be successively reduced.

The input data to program SOLGBRO to obtain the solution for $b_1 = 20$ ft is given below.

```
1 3
1 20 1 .013 .0008 1500 0 0 0 0 0 0 -1
1 10 1 .013 .0008 9000 0 0 0 0 0 0 0
1 10 1 .013 .0008 9000 0 0 0 0 0 0 0
.0000001 .001 5 32.2 0
3 1 2 3 0.
700 4.1 3.4
350 4.1
350 4.1
```

The solution is given below.

```
Solution to the three channel problem
----------------------------------------------
No  Ty      b     m     n       So      L   dz
----------------------------------------------
 1  -1  20.00  1.00  .013  .000800  1500.  .00
 2   0  10.00  1.00  .013  .000800  9000.  .00
 3   0  10.00  1.00  .013  .000800  9000.  .00
------------------------------------------
 db   dm     Yu     Yd       Q  Downstream
------------------------------------------
.00  .00  4.197  3.564  730.41      Branch
.00  .00  4.131  4.131  365.21     Uniform
.00  .00  4.131  4.131  365.21     Uniform
```

The bottom width is successively reduced by 0.1 ft, and when $b_1 = 19.6$ ft SOLGBRO failed to obtain a solution. Partial results from these solutions are contained in the table below.

b_1 (ft)	Q_1 (cfs)	Y_{u1} (ft)	Y_{d1} (ft)	Y_{c1} (ft)	$E_{d1} = E_{u2\,\&\,3}$
20.0	730.41	4.197	3.564	3.268	4.739
19.9	727.37	4.197	3.514	3.269	4.728
19.8	724.30	4.197	3.453	3.270	4.717
19.7	721.22	4.197	3.359	3.270	4.705
19.6	(Solution fails)				
19.67	720.29	4.197	3.292	3.270	4.702

Because of the side slope (m = 1), the upstream channel must be more than twice as wide as the two downstream channels for its area to be equal to their composite areas for any depth. This fact is reflected in channel 1 losing more energy than it gains from its bottom slope for all the solutions given in the above table. For example, when $b_1 = 20$ ft the specific energy at its end is 4.739 ft, whereas at its beginning E = 5.00 ft, or it loses 0.261 ft of head. To have its depth constant at $Y_1 = 4.197$ ft ($E_1 = 5.0$ ft) when its $Q_1 = 730.41$ cfs, its bottom width would need to be $b_1 = 20.57$ ft. However, it would then provide more head to the downstream channels so their flow rates would be larger. If the entire system's specific energy were 5.0 ft, then $Q_2 = Q_3 = 400.782$ cfs ($Y_2 = Y_3 = 4.359$ ft), $Q_1 = 801.56$ cfs, and a bottom width $b_1 = 22.505$ ft ($Y_{o1} = 4.212$ ft) would be required. For all of these solutions, the downstream depth in channel 1 Y_{d1}, is only slightly larger than its critical depth, and when $b_1 = 19.67$ ft, this difference $3.292 - 2.270 = 0.022$ ft is too small to distinguish from the critical conditions in light of the fact that the GVF equation gives $dY/dx = 4$ at Y_c.

A better way of examining the relationship between the variables in this channel system is to obtain a series of solutions in which the uniform depths in the downstream channels are varied systematically. The table below contains such an analysis in which $Y_{o2} = Y_{o3}$ have been changed in increments of 0.25 ft starting with 2 ft and ending with 6 ft. Manning's equation is solved for the flow rate Q in the downstream channels in column 3. This flow rate is doubled to get the flow rate in channel 1 in column 5. Next, the critical depth Y_c and the bottom width b_1 in columns 6 and 7 are solved by solving the critical flow and the specific energy equations simultaneously for the given Q and E. Finally, starting a GVF computation slightly above the critical depth, the M_2-GVF profile is solved to the entrance of channel 1 to get its upstream depth Y_u in the second from the last column, and the velocity head is added to this to get H of the reservoir in the last column.

$m = 1.0$, $b = 10.0$, $n = 0.0130$, $S_o = 0.00080$ (From Program SOLYB.FOR)

Downstream Channels				Upstream Channel						
Y	A	Q	E	Q	Yc	b	A	E	Yu	H
2.00	24.00	111.14	2.333	222.29	1.594	18.620	32.23	2.333	2.186	2.557
2.25	27.56	136.74	2.632	273.49	1.803	18.954	37.42	2.632	2.509	2.909
2.50	31.25	164.81	2.932	329.61	2.013	19.273	42.84	2.932	2.700	3.179
2.75	35.06	195.35	3.232	390.69	2.223	19.580	48.47	3.232	2.955	3.490
3.00	39.00	228.37	3.532	456.75	2.435	19.877	54.33	3.532	3.209	3.799
3.25	43.06	263.91	3.833	527.83	2.647	20.168	60.40	3.833	3.461	4.108
3.50	47.25	301.99	4.134	603.98	2.860	20.452	66.68	4.134	3.712	4.416
3.75	51.56	342.62	4.436	685.25	3.074	20.731	73.19	4.436	3.963	4.724
4.00	56.00	385.85	4.737	771.70	3.289	21.007	79.90	4.737	4.212	5.032
4.25	60.56	431.70	5.039	863.39	3.504	21.279	86.84	5.039	4.461	5.339
4.50	65.25	480.20	5.341	960.39	3.720	21.548	93.99	5.341	4.709	5.646
4.75	70.06	531.38	5.643	1062.76	3.936	21.815	101.36	5.643	4.956	5.952
5.00	75.00	585.29	5.946	1170.57	4.153	22.080	108.94	5.946	5.203	6.259
5.25	80.06	641.95	6.248	1283.90	4.370	22.343	116.75	6.248	5.449	6.565
5.50	85.25	701.40	6.551	1402.80	4.588	22.605	124.77	6.551	5.647	6.873
5.75	90.56	763.68	6.854	1527.35	4.807	22.865	133.01	6.854	5.940	7.177
6.00	96.00	828.81	7.157	1657.63	5.025	23.125	141.46	7.157	6.184	7.483

The results in this table represent limiting values; or values of the bottom width b, or specific energy E at the junction, or the reservoir head H, equal or less than those values that produce a critical flow at the downstream end of channel 1. Notice for smaller depths that the limiting width b is less than twice the downstream channels widths, which is 20 ft. However, for larger depths, and the corresponding larger flow rates, the upstream channel width must be more than twice the downstream width. This analysis indicates that for H = 5 ft the upstream channel width b would need to be about 21 ft, or larger, to prevent a critical flow from occurring, whereas SOLGBRO still found a solution with b = 19.67 ft. This difference occurs because high numerical accuracy cannot be achieved in solving the GVF ODE extremely close to the critical depth, and furthermore we would only be fooling ourselves if the ODE were accurately solved since it is not valid near the critical depth where the one-dimensional flow assumption that its derivation is based on cannot be justified.

4.22 OTHER DEPENDENT VARIABLES IN GVF COMPUTATIONS

It is possible to regard other variables than Y (or x) as the dependent variable or unknown. If the change in the bottom width b of a trapezoidal channel is to satisfy some criteria, it may be the dependent variable, and the ODE for GVFs would be written such that db/dx were on the left of the equal sign. The dependent variable may be the side slope m in which case the ODE should show dm/dx as the derivative. For a circular section, the diameter D may be considered the dependent variable, or for any channel the change in the position of the bottom z may need to satisfy other criteria, in which event the ODE should be written for dz/dx. Any of these as dependent variables would be appropriate in the design of transitions, at least for subcritical flows. In other words, the problem of transition design, under subcritical flow conditions, can be viewed as a problem of solving an ODE, just as computing GVF profile depths can be view as a problem of solving an ODE. In the case of a transition design, however, there is a choice of more than one variable that may be considered the dependent variable, i.e., the variable for which the solution is obtained based on the other variable(s) specified. Of the variables mentioned above, b or m for a trapezoidal channel, or D for a circular channel, or z for the change in elevation of the bottom, z is likely the best dependent variable to use. Once the dependent variable is selected, then it is necessary that the other variables be specified. That is, if z is selected as the dependent variable, then it is necessary to specify how b, m (for a trapezoidal channel), and the w.s. elevation change with x. It is common in the design of transitions to specify that the bottom width b changes as defined by two reversed parabolas that join at the center of the transition.

To develop the ordinary differential equation that gives dz/dx for use in the design of a channel transition remember that S_o, the slope of the channel bottom at any point is the negative of dz/dx. Solving Equation 4.7 for $-S_o$, for the case of no lateral inflow or outflow, and calling it dz/dx gives the following ODE:

$$\frac{dz}{dx} = \frac{S_{WS}\left(1 - F_r^2\right) - S_f + (Q^2/(gA^3))(\partial A/\partial x)}{F_r^2} \tag{4.36}$$

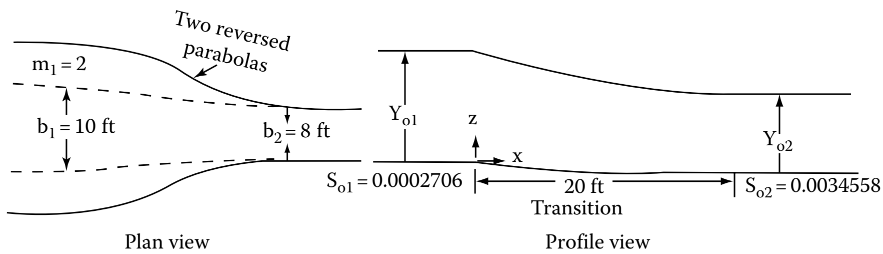
in which S_{ws} is the slope of the water surface. The depths Y and z are related through the following equation that involves the slope of the water surface:

$$Y = Y_1 - x\,S_{ws} - z \tag{4.37}$$

in which Y_1 is the upstream depth, and it is assumed that z equals zero at the beginning of the transition. Equation 4.26 can be solved numerically using the same methods as used for solving ODEs for gradually varied flows or spatially varied flows in which Y is considered the dependent variable. In fact, this is still a GVF, but the application is one of design rather than one of analysis.

EXAMPLE PROBLEM 4.65

A transition from a trapezoidal channel with a bottom width of b = 10 ft, a side slope of m = 2, and a bottom slope of S_{o1} = 0.0002706 to a rectangular channel with a bottom width of 8 ft, and a bottom slope of S_{o2} = 0.0034558 is to be designed. Manning's n for both channels equals 0.013. The criteria are as follows: (1) the transition is to be 20 ft long, (2) the side slope m is to vary linearly across the transition, (3) the bottom width is to be defined by two reversed parabolas, and (4) the water surface is to follow a straight line through the transition. The design flow rate for this channel is 400 cfs.



Solution

Solving Manning's equation for both the upstream and the downstream normal depths produces Y_{o1} = 5.0 ft and Y_{o2} = 4.5 ft, respectively. The upstream and the downstream velocity heads for these normal depths are 0.248 and 1.917 ft, respectively. Using the average slope of the channels to estimate the drop in the energy line across the 20 ft long transition indicates a drop of 0.0373 ft, and therefore the change in the water surface will be about 1.706 ft. The slope S_{ws} of the water surface across the transition should be 0.0853. A listing of a computer program for solving this problem using ODESOL described in Appendix C is given below. The input to this program and the solution obtained follows. It should be noted that using the average of S_{o1} and S_{o2} for the slope of the energy line is only an estimate of the actual loss of energy. The actual loss will be determined from the solution since S_f is part of the solution of the ODE. Based on this slope, the change in the bottom slope of the channel across the transition would be $\Delta z = 5 + 0.248 - 0.0373 - 1.917 - 4.5 = -1.205$ ft. The final solution gives the actual change in the bottom, Δz. In this case, the close agreement between this computed value and the last value for z in the table below indicates that the assumption of the composite frictional slope equaling the average of the two channel slopes is good. In the computer program listing you should note how the two reversed parabolas are defined.

FORTRAN program EPRB4_49.FOR to solve dz/dx

```
      EXTERNAL DZX
      REAL Z(1),ZPRIME(1),XP(1),YP(1,1),WK1(1,13)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV,XM,BM,B1,B2,DBH,XEND,DFM,
     &FM1,SWS,Y1,Q2G,QFN,B,A,Y
      WRITE(6,*)'GIVE:IOUT,TOL,DELX,B1,B2,FM1,Q,FN,SWS,Y1,XBEG,
     &XEND'
      READ(5,*)IOUT,TOL,DELX,B1,B2,FM1,Q,FN,SWS
      *,Y1,XBEG,XEND
      XM=.5*(XBEG+XEND)
      DM=.5*(B1+B2)
      DBH=2.*(B1-B2)/(XEND-XBEG)**2
      DFM=FM1/ABS(XEND-XBEG)
      Q2G=Q*Q/32.2
      QFN=Q*FN/1.49
      X=XBEG
      Z(1)=.0
      E=Y1+Q2G/2./((B1+FM1*Y1)*Y1)**2
      WRITE(IOUT,110)
110   FORMAT(1X,60('-'),/,8X,'x',9X,'z',9X,'y',9X,
      *'b',9X,'H',9X,'A',/,1X,60('-'))
      WRITE(IOUT,100) X,Z,Y1,B1,E,A
100   FORMAT(1X,5F10.3,F10.2)
2     XZ=X+DELX
      CALL ODESOL(Z,ZPRIME,1,X,XZ,TOL,.1,HMIN,1,XP,
      *YP,WK1,DZX)
```

```
      X=XZ
      E=Y+Q2G/(2.*A*A)+Z(1)
      WRITE(IOUT,100) X,Z,Y,B,E,A
      IF(X .LT. XEND) GO TO 2
      STOP
      END
      SUBROUTINE DZX(X,Z,ZPRIME)
      REAL Z(1),ZPRIME(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV,XM,BM,B1,B2,DBH,XEND,DFM,
     &FM1,SWS,Y1,Q2G,QFN,B,A,Y
      IF(X .GT. XM) GO TO 10
      B=B1-DBH*X*X
      DAX=-(2.*DBH*X+DFM*Y)*Y
      GO TO 20
10    B=B2+DBH*(XEND-X)**2
      DAX=-(2.*DBH*(XEND-X)+DFM*Y)*Y
20    FM=FM1-DFM*X
      Y=Y1-SWS*X-Z(1)
      A=(B+FM*Y)*Y
      A3=A**3
      T=B+2.*FM*Y
      FR2=Q2G*T/A
      P=B+2.*Y*SQRT(FM*FM+1.)
      SF=(QFN*(P/A)**.66666667/A)**2
      ZPRIME(1)=(SWS*(1.-FR2)-SF+Q2G*DAX/A3)/FR2
      RETURN
      END
```

Input needed by program to solve the above problem:

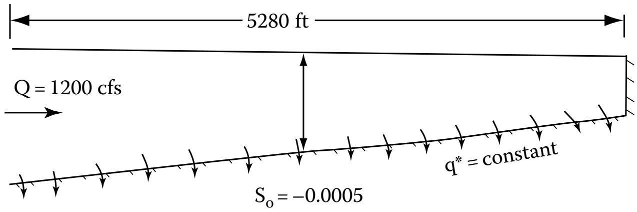
```
6 .001 2 10 8 2 400 .013 .0853 5 0 20
```

Solution to this problem:

x (ft)	z (ft)	Y (ft)	b (ft)	E (ft)
0.0	0.000	5.000	10.000	5.248
2.0	0.492	4.337	9.960	5.247
4.0	0.656	4.001	9.840	5.245
6.0	0.665	3.821	9.640	5.243
8.0	0.575	3.742	9.360	5.242
10.0	0.404	3.744	9.000	5.239
12.0	0.173	3.805	8.640	5.235
14.0	−0.097	3.905	8.360	5.231
16.0	−0.408	4.047	8.160	5.226
18.0	−0.770	4.240	8.040	5.220
20.0	−1.205	4.509	8.000	5.213

EXAMPLE PROBLEM 4.66

As a means of preventing damages to a river under flood conditions, it is proposed that the excess flood waters be spread over adjacent lands by means of a rectangular channel that has a side weir along its entire length. Under flood conditions, a steady-state flow rate of $Q_o = 1200$ cfs will be diverted into this channel. The channel will have a Manning's n = 0.013 and an adverse slope of $S_o = -0.0005$ and will be 1 mile long. Design this channel if the discharge coefficient for the side weir is $C_d = 0.52$.



Solution

The discharge per unit length of this channel is q_o = 1200/5280 = 0.227 cfs/ft. The position of the weir crest below the water surface $Y - H_w$ is obtained from $q_o = C_d/(2g(2/3)(Y - H_w)^{1.5}$, or $Y - H_w$ = 0.1883 ft. The hydraulically most efficient section, which also result in the most economical section, is one in which the depth of flow is 1/2 the bottom width. This criteria will be adopted. The flow rate at any position x in the channel will be $Q = Q_o(1 - x/L)$. Also, it is decided that at the beginning of the channel the bottom width should be 20 ft giving an entrance velocity of 6 fps into the channel. To obtain the ODE that defines the change in the bottom width b with respect to x substitute the following into Equation 4.3: dY/dx = (1/2)db/dx, $A = bY = b^2/2$ so $\partial A/\partial x = b(db/dx)$. Solving db/dx gives

$$\frac{db}{dx} = \frac{2(S_o - S_r + Qq_o/(gA^2)}{1 - 3Q^2b/(gA^2)} = \frac{2(S_o - S_r + 4Qq_o/(gb^4)}{1 - 24Q^2/(gb^5)}$$

The solution to this problem as well as a listing of the FORTRAN program used to obtain this solution is given below:

x (ft)	b (ft)	Q (cfs)	Y (ft)	H_w (ft)
0.0	20.000	1200.0	10.00	9.81
400.0	19.452	1109.1	9.73	9.54
800.0	18.915	1018.2	9.46	9.27
1200.0	18.394	927.3	9.20	9.01
1600.0	17.891	836.4	8.95	8.76
2000.0	17.411	745.5	8.71	8.52
2400.0	16.956	654.5	8.48	8.29
2800.0	16.528	563.6	8.26	8.08
3200.0	16.126	472.7	8.06	7.87
3600.0	15.748	381.8	7.87	7.69
4000.0	15.388	290.9	7.69	7.51
4400.0	15.038	200.0	7.52	7.33
4800.0	14.685	109.1	7.34	7.15
5200.0	14.312	18.2	7.16	6.97
5280.0	14.233	0.0	7.12	6.93

FORTRAN program EPRB4_66.FOR to solve above problem

```
      REAL B(1),XP(1),BPRIME(1),YP(1,1),WK1(1,13)
      EXTERNAL DBX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV
      COMMON /TRAN/ C,G,XO,FN,SO,QO,QS,FL
      WRITE(6,* )'GIVE IOUT,TOL,FL,DELX,BO,QO,FN,
      *SO,XBEG,XEND,g'
      READ(5,*)IOUT,TOL,FL,DELX,BO,QO,FN,SO,XBEG,XEND,G
```

```
      QS=QO/FL
      C=1.
      H1=-.01
      IF(G.GT.30.) C=1.486
      B(1)=BO
      Q=QO
      X=XBEG
      Y=BO/2.
      WRITE(IOUT,100) X,B,Q,Y,Y-.18828
2     XZ=X+DELX
      IF(XZ.GT.XEND) XZ=XEND
      CALL ODESOL(B,BPRIME,1,X,XZ,TOL,H1,0.,1,
      *XP,YP,WK1,DBX)
      X=XZ
      Q=QO*(1.-X/FL)
      Y=B(1)/2.
      WRITE(IOUT,100) X,B,Q,Y,Y-.18828
100   FORMAT(F8.0,F8.3,F8.1,2F8.2)
      IF(DELX .LT. 0.) GO TO 8
      IF(X .LT. XEND) GO TO 2
      GO TO 99
8     IF(X .GT. XEND) GO TO 2
99    STOP
      END
      SUBROUTINE DBX(X,B,BPRIME)
      REAL B(1),BPRIME(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV
      COMMON /TRAN/ C,G,XO,FN,SO,QO,QS,FL
      A=B(1)**2/2
      P=2.*B(1)
      Q=QO*(1-X/FL)
      SF=(FN*Q*(P/A)**.66666667/(C*A))**2
      A2=A*A*G
      BPRIME(1)=2.*(SO-SF+Q*QS/A2)/(1.-Q*Q*B(1)/(A2*A)))
      RETURN
      END
```

Input data needed to solve above problem:

```
3 .001 5280 400 20 1200 .013 -.0005 0 5280 32.2
```

Note that there are many other options available. Instead of specifying that Y = b/2, the criteria might have been to let Y vary according to some specified function of x. Likewise, the beginning 20 ft width of the channel was arbitrary.

4.23 VARIED FLOW FUNCTION

From a historical viewpoint, the use of the varied flow function, as an alternative means for obtaining solutions to GVF profiles, is covered in this section. This method was developed to reduce the number of hand computations required in solving GVF problems, but as computers have become widely used it no longer has advantages over the computational methods discussed earlier in this chapter. So you might skip reading this section of the book if you feel satisfied in using the methods already covered. Chow (1959) provides some insight into the efforts of individuals to develop this method in its various forms. Now, these tedious hand integrations that required many days of work by dedicated individuals to develop the tables of numbers for the varied flow function can be accomplished very easily by means of a computer. In fact, the need for developing such a table has passed since the needed integration can be accomplished specifically for the problem being solved, and also simply because too much effort is involved in looking up values from a table.

The development of the varied flow function method of computing GVF profiles was based on the observation that when only modest changes in depth occur in a given channel the ratio of S_f/S_o can be related to the ratio of the normal depth Y_o to the depth of flow Y raised to a nearly constant power, or $S_f/S_o = (Y_o/Y)^N$ in which N is nearly constant. Likewise, the Froude number squared, F_r^2 can be related to the ratio of the critical depth Y_c to the actual depth raised to a nearly constant power, or $F_r^2 = (Y_c/Y)^M$. Based on these observations, Equation 4.4 can be written as

$$\frac{dY}{dx} = S_o \frac{1 - S_f/S_o}{1 - F_r^2} = S_o \frac{1 - (Y_o/Y)^N}{1 - (Y_c/Y)^M} = S_o \frac{1 - 1/Y'^N}{1 - 1/Y_c'^M} \tag{4.38}$$

in which two dimensionless depths $Y' = Y/Y_o$ and $Y_c' = Y/Y_c$ are given in the term after the final equal sign. Using an average depth for the GVF, Y_{av} to obtain the exponents, N and M can be evaluated from the following equations, respectively:

$$N = \frac{\text{Log}(S_f/S_o)}{\text{Log}(Y_o/Y_m)} \tag{4.39}$$

and

$$M = \frac{\text{Log}\left(F_r^2\right)}{\text{Log}(Y_c/Y_{av})} \tag{4.40}$$

By noting that $dY = Y_o dY'$, Equation 4.36 can be rewritten as

$$\frac{dY}{dx} = \frac{Y_o}{S_o} \frac{Y'^N - Y'^N/Y_c'^M}{Y'^N - 1} dY' = \frac{Y_o}{S_o}\left(1 - \frac{1}{1 - Y'^N} + \left(\frac{Y_c}{Y_o}\right)^M \frac{Y'^{N-M}}{1 - Y'^N}\right) dY' \tag{4.41}$$

By defining $J = N/(N-M+1)$ and $w = Y'^{N/J}$ (or $w^J = Y'^N$) so that $dw = (N/J)Y'^{N/J-1}dY'$, the last term that multiplies $(Y_c/Y_o)^M$ in Equation 4.41 becomes $(J/N)dw/(1 - w^J)$ or upon integrating both sides Equation 4.41 becomes

$$x = Y_o/S_o\left\{\left[\int dY' - \int dY'/(1 - Y'^N) + (Y_c/Y_o)^M (J/N)\int dw/(1 - w^J)\right\}\right. + \text{Constant} \tag{4.42}$$

Since the last two integrals are of the same form, just different variables, they can be defined as the **varied flow function**, i.e.,

$$F(Y', N) = \int dY'/(1 - Y'^N) \quad \text{and} \quad F(w, J) = \int dw/(1 - w^J) = \int dw/(1 - Y'^N)$$

A table can be produced that provides values for $F(Y', N)$ and $F(w, J)$, in which the columns in this table are for different values of N (or J) and the rows correspond to different values of Y' (or w). By using such a table the length of a GVF profile with depths Y_1 and Y_2 at its two ends is then given by

$$L = x_2 - x_1 = \frac{Y_o}{S_o}\left\{[Y_2' - Y_1'] - [F(Y_2', N) - F(Y_1', N)] + \left(\frac{Y_c}{Y_o}\right)^M \frac{J}{N}[F(w_2, J) - F(w_1, N)]\right\} \quad (4.43)$$

The depths Y_1 and Y_2 (and the value for $Y_1' = Y_1/Y_o$ and $Y_2' = Y_2/Y_o$) can be the ends of the GVF profile, but for better accuracy, or if it is desired to be able to plot the profile of water depths, then several steps can be used with $Y_2 - Y_1$ equaling the ΔY increment used.

Using hand methods, the following steps are followed in obtaining a solution to a GVF problem:

1. The normal depth Y_o and the critical depth Y_c are computed.
2. The exponents N and M are computed from Equations 4.39 and 40, and J computed from $J = N/(N - M + 1)$.
3. The dimensionless variables $Y' = Y/Y_o$ and $w = Y'^{N/J}$ are computed corresponding to the end depths Y_1 and Y_2.
4. Values of the varied.flow.functions are obtained from a table that gives these values. (Such a table is given in Chow's book in Appendix D, or could be generated by numerically integrating on the varied flow function.)
5. These values are substituted into Equation 4.43.

In adapting this varied.flow.function method to computer computations the quantities $[F(Y'_2, N) - F(Y'_1, N)]$ and $[F(w_2, J) - F(w_1, J)]$ in Equation 4.43 are evaluated by numerically integrating the varied flow function using Simpson's rule, the trapezoidal rule, or some other appropriate numerical integration method. In other words,

$$F(Y_2, N) - F(Y_1, N) = \int_{Y_1'}^{Y_2'} \frac{dY'}{1 - Y'^N} \quad \text{(numerically integrated)}$$

$$F(w_2, J) - F(w_1, J) = \int_{w_1}^{w_2} \frac{dw}{1 - w^J} \quad \text{(numerically integrated)}$$

and the values produced from these numerical integrations are substituted into Equation 4.43 along with the other needed values to compute L. The steps that such a computer program needs to accomplish consist of the following:

1. Read in the problem variables: b, m, n, S_o, Y_1, Y_2, etc.
2. Compute the normal and critical depths.
3. Compute N, M, $J = N/(N - M + 1)$, and $(Y_c/Y_o)^M(J/N)$ based on the average depth $Y_{av} = (Y_1 + Y_2)/2$.
4. Carry out the numerical integrations of the above varied flow functions.
5. Compute the length of the GVF profile from Equation 4.43.

EXAMPLE PROBLEM 4.67

At the toe of a dam spillway, the depth $Y_1 = 0.15$ m. The channel has the following properties downstream from the spillway: b = 6 m, m = 2, $S_o = 0.0008$, and n = 0.02. For a flow rate $Q = 15\ m^3/s$, determine the location of the hydraulic jump. Carry out the computations using first, third, and tenth steps using the varied flow function.

Solution

It is first necessary to find the normal depth and from this its conjugate depth so that the ending depth Y_2 of the GVF profile can be determined. Solving Manning's equation and then the momentum equation $M_1 = M_2$ gives $Y_o = 1.289\,\text{m}$ and $Y_2 = 0.424\,\text{m}$. The following computer program based on the steps defined above can solve the position of the hydraulic jump using first, third, and tenth steps with the following input, respectively:

```
6 1 .15 .424 .02 .0008 6 2 15 9.81
6 3 .15 .424 .02 .0008 6 2 15 9.81
6 10 .15 .424 .02 .0008 6 2 15 9.81
```

Listing of program VARIFUN that solved the GVF problem using the varied flow function method

```
      PARAMETER (NINC=20) ! NINC is the number of increments
     &used in the numerical integration
      WRITE(*,*)' IOUT,N,Y1,Y2,n,So,b,m,Q,g' ! N=No. of steps
      READ(*,*) IOUT,N,Y1,Y2,FN,SO,B,FM,Q,G
      FNINC=NINC
      SM=2.*SQRT(FM*FM+1.)
      SS=SQRT(SO)
      DELY=(Y2-Y1)/FLOAT(N)
      C=1.486
      IF(G.LT.30.) C=1.
      YO=(FN*Q/(1.5*B)/(C*SS))**.6
10    A=(B+FM*YO)*YO
      P=B+SM*YO
      F=FN*Q*P**.66666667-C*SS*A**1.66666667
      DF=(2.*FN*Q*SM/P**.33333333-5.*C*SS*(B+2.*FM*YO)*A**.
     &66666667)/3.
      YO=YO-F/DF
      IF(ABS(F/DF).GT. .000001) GO TO 10
      YC=((Q/(1.5*B))**2/G)**.333333333
20    A=(B+FM*YC)*YC
      T=B+2.*FM*YC
      F=Q*Q*T-G*A**3
      DF=2.*Q*Q*FM-3.*G*T*A*A
      YC=YC-F/DF
      IF(ABS(F/DF).GT. .000001) GO TO 20
      WRITE(IOUT,100) YO,YC
100   FORMAT(' Yo=',F10.3,' Yc=',F10.3,/,1X,48('-')/5X,'Depth,Y
     &Position,x S. Energy,E Momentum,M',/1X,48('-'))
      X=0.
      A=(B+FM*Y1)*Y1
     WRITE(IOUT,110) Y1,X,Y1+(Q/A)**2/(2.*G),
     &(.5*B+FM*Y1/3.)*Y1*Y1+Q*Q/G/A
      DO 40 I=1,N
      YB=Y1+DELY*FLOAT(I-1)
      YE=YB+DELY
      YA=.5*(YB+YE)
      A=(B+FM*YA)*YA
      SF=(FN*Q/C/A*((B+SM*YA)/A)**.66666667)**2
      EN=ALOG(SF/SO)/ALOG(YO/YA)
      EM=ALOG(Q*Q*(B+2.*FM*YA)/(G*A**3))/ALOG(YC/YA)
      EJ=EN/(EN-EM+1.)
      ENJ=EN/EJ
      YP1=YB/YO
      DYP=(YE/YO-YP1)/FNINC
```

```
      DW=((YE/YO)**ENJ-YP1**ENJ)/(6.*FNINC)
      DYP2=DYP/2.
      DYP6=DYP/6.
      SYP=0.
      SW=0.
      FP1=1./(1.-YP1**EN)
      COE=(YC/YO)**EM*(EJ/EN)
      DO 30 J=1,NINC
      YP=YP1+FLOAT(J)*DYP
      FP2=1./(1.-YP**EN)
      FPM=4./(1.-(YP-DYP2)**EN)+FP1+FP2
      SYP=SYP+DYP6*FPM
      SW=SW+DW*FPM
30    FP1=FP2
      X=X+YO/SO*(FNINC*DYP-SYP+COE*SW)
      A=(B+FM*YE)*YE
40    WRITE(IOUT,110) YE,X,YE+(Q/A)**2/(2.*G),
     &(.5*B+FM*YE/3.)*YE* YE+Q*Q/G/A
110   FORMAT(F12.3,F12.1,2F12.3)
      END
```

The solutions provided by this program consist of the following three tables:

$Y_o = 1.289$, $Y_c = 0.785$.

Depth, Y	Position, x	S. Energy, E	Momentum, M
0.150	0.0	12.992	24.340
0.424	41.2	1.784	8.489

(Using 3 steps)

$Y_o = 1.289$, $Y_c = 0.785$.

Depth, Y	Position, x	S. Energy, E	Momentum, M
0.150	0.0	12.992	24.340
0.241	12.8	4.927	14.844
0.333	26.8	2.665	10.700
0.424	41.2	1.784	8.489

(Using 10 steps)

$Y_o = 1.289$, $Y_c = 0.785$.

Depth, Y	Position, x	S. Energy, E	Momentum, M
0.150	0.0	12.992	24.340
0.177	3.7	9.201	20.443
0.205	7.5	6.860	17.604
0.232	11.5	5.322	15.450
0.260	15.6	4.264	13.766
0.287	19.7	3.509	12.419
0.314	24.0	2.955	11.322
0.342	28.3	2.539	10.417
0.369	32.6	2.222	9.662
0.397	36.9	1.976	9.027
0.424	41.2	1.784	8.489

Note that for this problem, the position of the hydraulic jump is determined at a distance of 41.2 m downstream from the toe of the spillway regardless of whether 1, 3, or 10 increments are used. The reason why the solution for the position is insensitive to the number of steps is that the exponents N and M do not change significantly with the range of depths in this problem, and thus it is possible to used fewer steps than needed when using the "direct step" method described earlier in this chapter.

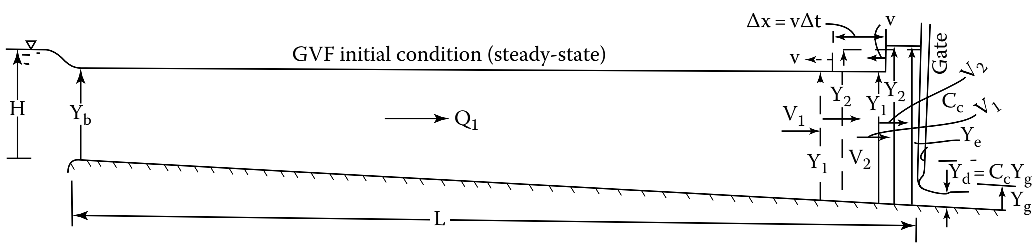
4.24 MOVING WAVES

4.24.1 Quasi-Unsteady Analysis

In Chapter 3, the velocity, etc., of the constant height moving waves were solved by using the continuity and momentum equations from the viewpoint of a moving observer so a steady state hydraulic jump appeared. In this section, a quasi-unsteady analysis of such moving waves is described. In most real situations, the depth and velocity upstream and downstream from moving waves are not constant, and therefore the height of the wave is also not constant. The quasi-unsteady analysis of waves solves the steady-state equations repeatedly for a series of time steps at an increment Δt apart to obtain a time-dependent solution of the wave's velocity and height, etc., as a function of time. This time-dependent solution ignores the inertial effects associated with the acceleration/deceleration of the water in the channel, and thus the analysis is referred to as quasi-unsteady. Chapters 6 and 7 describe methods for solving unsteady problems that do not ignore inertia, i.e., the St. Venant equations (unsteady one-dimensional equations) will be used.

4.24.2 Downstream Controlled Waves

As was noted in Chapter 3, such moving waves may move upstream, their movement being caused by a downstream control, and these are referred to as "Downstream Controlled Waves" (DCW), or they move downstream, this being caused by an upstream control, and these are referred to as "Upstream Controlled Waves" (UCW). The quasi-unsteady analysis of DCW and UCW are essentially the same; both types will be dealt with in this section, but the basic ideas will be covered first using DCW. As an example of a DCW, consider a channel whose supply comes from a constant head reservoir and contains a gate at its downstream end, as illustrated below.



Assume the gate's setting has not changed for a long time so that a steady state M_1-GVF exists in the channel. The solution to this steady state flow is referred to as the **initial condition**. Then suddenly the gate is instantly closed further, causing a DCW as shown in the above sketch. Upstream of this wave, the flow rate Q_1 is constant but the depth Y_1 and the velocity V_1 immediately upstream therefrom will vary with the position of the wave (and consequently with time) because a GVF exists in the channel. Likewise, the depth Y_2 and the velocity V_2 immediately downstream from the wave will vary with the position of the wave (and thus with time.) It will be assumed that the flow rate $Q_2 = V_2A_2$ downstream from the moving wave does not change with the position along the channel. Q_2 in general will vary with time, however, but at any time t this assumption requires that the flow rate immediately downstream from the wave equals the flow rate passing the gate. This assumption of Q_2 not being a function of x may not be true in real situations, but the equations available from the quasi-unsteady theory simply do not allow Q_2 to be a function of position x and t. But the St. Venant equations of Chapter 6 and 7 will allow this dependency to be handled.

The solution will consist of solving steady state equations for a series of time steps. For each such time step, the following five equations, etc., are available. (They are numbered 1 through 5 since they are equations given previously and the d following the number denotes DCW.)

Continuity from the viewpoint of an observer moving with the wave:

$$F_1 = (v + V_1)A_1 - (v + Q_2/A_2)A_2 = 0 \tag{1d}$$

Momentum from the viewpoint of an observer moving with the wave:

$$F_2 = (Ah_c)_1 + \frac{(v+V_1)^2 A_1}{g} - (Ah_c)_2 - \frac{(v+V_2)^2 A_2}{g} = 0 \tag{2d}$$

Energy across the gate (stationary observer):

$$F_3 = Y_e + \frac{Q_2^2}{2gA_2^2} - Y_d - \frac{Q_2^2}{2gA_d^2} = 0 \tag{3d}$$

GVF from gate to the position of the wave:

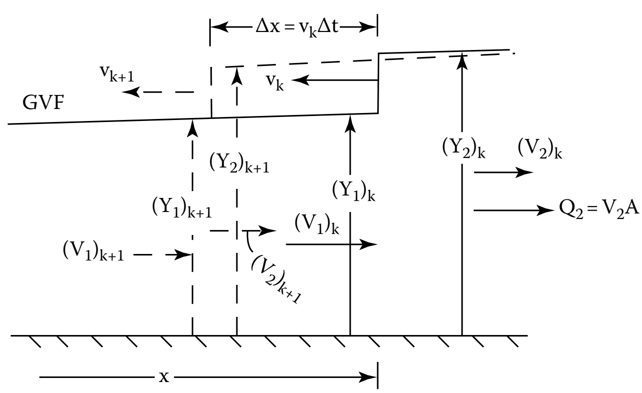
$$F_4 = Y_2 - Y_{2ode}(Y_e) = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f}{1 - F_r^2} \tag{4d}$$

i.e., the GVF ODE is solved starting with the position of the gate where the depth is Y_e to where the wave currently is.

Interpolation of $Y_1(x)$ from the initial GVF profile

$Y_1(x)$ obtained by interpolation as a function of the wave position x and from Y_1,

$$V_1 = \frac{Q_1}{A_1(x)} \tag{5d}$$



There are two approaches to solving these equations. To use terminology from the numerical solution of partial differential equations, (1) an **explicit** and (2) an **implicit** method. The explicit method solves the equations as if they apply at the current time step t. In our problem, this means that the position x of the wave is known, since it is where the wave was based on the previous time step solution, or at the gate for the first time step. In the implicit method, the equations apply at some time between t and t + Δt, typically midway between, or at t + Δt/2.

4.24.2.1 Explicit Method

Let us examine the explicit method first. Since the position x is considered a known variable, the first task in the explicit method is to use Equation 4.5d alone, i.e., by interpolation of the initial steady state

GVF solution to determine the depth Y_1 immediately upstream from the wave based on its position x, and then solve the upstream velocity V_1 by dividing the known flow rate Q_1 by the area associated with this depth, or $V_1 = Q_1/A_1(x)$. Next, Equations 4.1d through 4.4d above are solved simultaneously for the four variables: v, Y_2, Y_e, and Q_2. Notice that three of these equations are algebraic and the fourth is an ODE. Thus, the solution will be based on using the Newton method as in previous applications in this chapter. The difference is that now the process starts with interpolating Y_1 with the x known and then the solving of the four equations will be repeated for a number of time steps Δt apart. We might select how many time steps to use, but the number of such steps cannot be more than required for the wave to arrive at the upstream reservoir, because at this time, a different problem will occur.

In order to implement the above process, it is necessary that the initial condition be known, i.e., the flow rate and the depths throughout the channel upstream from the gate. Typically, this initial condition assumes that the steady state flow exists at time zero. For the gate controlling the flow in a channel fed by a constant head reservoir, this steady state solution comes from solving the following three simultaneous equations for Q_1, Y_b, and Y_e. (Now note Y_1 represents the flow at the beginning of the channel and not the depth immediately upstream from the wave.)

$$F_1 = Y_e + \frac{Q_1^2}{2gA_g^2} - Y_e + \frac{Q_1^2}{2gA_d^2} = 0 \quad \text{Energy across gate} \tag{1i}$$

$$F_2 = H - Y_b - \frac{Q_1^2}{2gA_b^2} = 0 \quad \text{Energy at reservoir} \tag{2i}$$

$$F_3 = Y_b - Y_{bode}(Y_e) = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f}{1 - F_r^2} \quad \text{ODE from gate to reservoir} \tag{3i}$$

The program WAVEMOV is designed to solve a DCW using this explicit method. An excellent means for understanding how the above method solves moving waves is to study this program. The array X(4) (x[4] in the C-program) is used for the four unknown variables in the order v = X(1), Y_2 = X(2), Y_e = X(3), and Q_2 = X(4). The array F is used to store the values of the four equations, i.e., Equations 4.1d through 4.4d above, that need to be solved simultaneously. The array F1 contains the values of these equations with the unknowns incremented so that the elements of the Jacobian D(4,4) can be numerically evaluated. The subroutine FUN2 (void function fun2 in the C-program) is designed to evaluate the four equations. The subroutine ODESOLF (void function rukust in the C-program) solves the ODE, and the subroutine SOLVEQ solves the linear system of equations needed to implement the Newton method as in previous programs. Let us examine the logic in the main program. First, after a prompt to provide data for the initial condition, the following variables are read: G=acceleration of gravity, H=head of the reservoir, Ke = entrance loss coefficient, Cc = gate's contraction coefficient, Yg = initial height of the gate above the bottom, L (FL) = length of the channel, n (FN) = Manning's n, b = bottom width of the trapezoidal channel, m (FM) = side slope, So = bottom slope of the channel, Q = guess of the initial flow rate, Yb = guess for the depth at the beginning of the channel, and Ye = guess of the depth at the end of the channel immediately upstream from the gate. The first task is to solve the three equations immediately above for the initial condition. The subroutine FUN1 provides the values of the three equations for the initial condition solution. The unknowns for the initial condition are used in the array X as follows: Q_1 = X(1), Y_b = X(2), Y_e = X(3). For this solution, the same arrays X, F, F1, and D are used except now only the first three elements are used. After the solution for Q_1, Y_b, and Y_e has been obtained, the subroutine ODESOLF (rukust in the C-program) is called 14 times to provide the depths of the GVF profile at 15 equally spaced positions along the length of the channel and then these are stored in the array Yo(15) for the initial condition that will be used later to interpolate Y_1 in the DO 16 loop, and thereafter the initial condition is written to an output file. Following a prompt in the program, the next is to read in information to define the time-dependent part of the problem. The following are read: NT = number of time steps, DT = Δt the time increment in seconds, and Y_g = the new gate setting.

After providing guesses for the Newton method to solve the unknown variables v, Y_2, Y_e, and Q_2, the DO 50 K=1,NT loop repeats the following tasks for the NT time steps:

1. Finds the Δx increment number M corresponding to x and then uses the linear interpolation of Yo to evaluate Y_1 corresponding to the current position x of the wave.
2. Solves the above four equations 1 through 4 for the new wave velocity v, the depth immediate downstream from the wave Y_2, the depth immediately upstream from the gate Y_e, and the flow rate Q_2 in the channel between the position of the wave and the gate.
3. Writes this solution to the first output file.
4. Solves the GVF between the gate and the current position x (variable DIST in the Fortran program) of the wave and writes these depths to a second output file.
5. Finds the new position x of the wave by subtracting vΔt from the current position, i.e., the statement `40 DIST=DIST-DT*X(1)` accomplishes this task.

The statement following the label `40 IF(DIST.LT.0.)` STOP checks if the wave has arrived at the upstream end of the channel before all the specified time steps have been completed, and if so, stops the solution process. Notice that the index K of the time step is in the block COMMON/TRAS/, and is used in the subroutine FUN2 so that when the wave is at the gate (the instant after the gate's position is dropped) the fourth equation F_4 = F(4) = X(3) − Ye (or Ye − Ye is equated to zero), and there is no need to solve the ODE as is done for all other time steps. This saves the additional logic of solving only the three equations 1–3 for the first time step.

Program WAVEMOV.FOR

```
C Uses an explicit method to solve a moving wave
C upstream from a gate (DCW)
      LOGICAL NFS
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,
     &HMIN,CUN,FMS,FM2,Q,A1,V1,HCA1,K,Ye,DIST
      REAL X(4),F(4),F1(4),D(4,4),Yo(15),YSTOR(16),Y(1)
     &,DY(1),XP(1),YP(1,1),W(1,13)
      INTEGER*2 INDX(4)
      EQUIVALENCE (v,X(1)),(Y2,X(2)),(Qe,X(4))
      WRITE(*,*)' For initial condition give:=,
     &=g,H,Ke,Cc,Yg,L,n,b,m,So, Guess for Q,Y1,Y2'
      H11=-.5
      HMIN=1.E-5
      TOL=1.E-5
      READ(*,*) G,H,FKe,Cc,Yg,FL,FN,B,FM,So,Q,Y2,X(3)
      FKe=FKe+1.
      X(1)=Q
      FMS=2.*SQRT(FM*FM+1.)
      FM2=2.*FM
      Yd=Cc*Yg
      G2=2.*G
      Ad=((b+FM*Yd)*Yd)**2*G2
      Ad1=Ad
```

```
      Cu=1.486
      IF(G.LT.20.) Cu=1.
      CUN=FN/Cu
C Q=X(1); Yb (upstream)=X(2); Ye (downstream)= X(3)
      NCT=0
10    SUM=0.
      CALL FUN1(F,X)
      DO 12 I=1,3
      XX=X(I)
      X(I)=1.005*X(I)
      CALL FUN1(F1,X)
      DO 11 J=1,3
11    D(J,I)=(F1(J)-F(J))/(X(I)-XX)
12    X(I)=XX
      CALL SOLVEQ(3,1,4,D,F,1,DD,INDX)
      DO 14 I=1,3
      X(I)=X(I)-F(I)
14    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      IF(SUM.GT.5.E-5 .AND. NCT.LT.20) GO TO 10
      WRITE(3,130)(X(I),I=1,3)
130   FORMAT(' Steady-State Q=',F8.1,', Y1=',F8.3,', Ye=',F8.3)
      Ye=X(3)
      Y(1)=X(3)
      Yo(15)=X(3)
      Q1=X(1)
      DL=FL/14.
      XX=FL
      NFS=.TRUE.
      DO 16 I=14,1,-1
      XX1=XX-DL
      CALL ODESOLF(Y,DY,1,XX,XX1,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      NFS=.FALSE.
      XX=XX1
16    Yo(I)=Y(1)
      WRITE(3,100) (DL*FLOAT(I-1),Yo(I),I=15,1,-1)
100  FORMAT(' Initial Condition (x,Y)',/,8(F8.0,F8.3),/,7(F8.0,F8.3))
      WRITE(*,*)' Give:No. time steps,Dt,New Yg'
      READ(*,*) NT,DT,Yg
C v=X(1); Y2=X(2); Ye=X(3); Q2=X(4)
      Yd=Cc*Yg
      Ad=((B+FM*Yd)*Yd)**2*G2
      X(4)=Q1*SQRT(Ad/Ad1)
      X(3)=1.2*Yo(15)
      X(2)=.95*X(3)
      X(1)=1.2*SQRT(G*(B+FM*X(3))*X(3)/(B+FM2*X(3)))
      DIST=FL-.1
      H11=-.5
      WRITE(3,111)
111   FORMAT(/,' Time x v Y2 Ye=,= Q Y1 V1',/,1X,62('-'))
      DO 50 K=1,NT
```

```
      M=IFIX(DIST/DL)+2
      IF(M.LT.2) THEN
      Y1=Yo(1)
      ELSE
      FAC=(DIST-FLOAT(M-2)*DL)/DL
      Y1=Yo(M-1)+FAC*(Yo(M)-Yo(M-1))
      ENDIF
      A1=(B+FM*Y1)*Y1
      HCA1=(.5*B+FM*Y1/3.)*Y1**2
      V1=Q1/A1
      IF(K.EQ.1) DIST=FL
      NCT=0
20    SUM=0.
      Q=X(4)
      IF(K.EQ.1) Ye=X(3)
      CALL FUN2(F,X)
      DO 22 I=1,4
      XX=X(I)
      X(I)=1.005*X(I)
      CALL FUN2(F1,X)
      DO 21 J=1,4
21    D(J,I)=(F1(J)-F(J))/(X(I)-XX)
22    X(I)=XX
      CALL SOLVEQ(4,1,4,D,F,1,DD,INDX)
      DO 24 I=1,4
      X(I)=X(I)-F(I)
24    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,SUM
      IF(SUM.GT.5.E-5 .AND. NCT.LT.20) GO TO 20
      WRITE(3,110) IFIX(DT*FLOAT(K-1)),DIST,X,Y1,V1
110   FORMAT(I7,F8.1,3F8.3,F8.2,2F8.3)
      IF(K.EQ.1) GO TO 40
      NFS=.FALSE.
      Y(1)=X(3)
      XX=FL
      J=0
30    J=J+1
      XX1=XX-DL
      IF(XX1.LT.DIST) XX1=DIST
      NFS=.TRUE.
      CALL ODESOLF(Y,DY,1,XX,XX1,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      YSTOR(J)=Y(1)
      XX=XX1
      NFS=.FALSE.
      IF(XX1.GT.DIST) GO TO 30
      WRITE(4,120) IFIX(DT*FLOAT(K-1)),(DL*FLOAT(15-I),
     &YSTOR(I),I=1,J)
120   FORMAT(I6,8(F8.0,F8.3),/,7(F8.0,F8.3))
40    DIST=DIST-DT*X(1)
      IF(DIST.LT.0.) STOP
```

```
50    CONTINUE
      END
      SUBROUTINE FUN1(F,X)
      REAL Y(1),DY(1),XP(1),YP(1,1),W(1,13),X(4),F(4)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,HMIN,CUN,
     &FMS,FM2,Q,A1,V1,HCA1,K,Ye,DIST
      REAL X(4),F(4),F1(4),D(4,4),Yo(15),YSTOR(16),Y(1),DY(1),
     &XP(1),YP(1,1),W(1,13)
      A=(B+FM*X(3))*X(3)
      F(1)=X(3)+(X(1)/A)**2/G2-Yd-X(1)**2/Ad
      F(2)=H-X(2)-FKe*(X(1)/((B+FM*X(2))*X(2)))**2/G2
      Y(1)=X(3)
      CALL ODESOLF(Y,DY,1,FL,0.,TOL,H11,HMIN,1,XP,YP,W,DYX,.TRUE.)
      F(3)=X(2)-Y(1)
      RETURN
      END
      SUBROUTINE FUN2(F,X)
      REAL Y(1),DY(1),XP(1),YP(1,1),W(1,13),X(4),F(4)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,HMIN,CUN,
     &FMS,FM2,Q,A1,V1,HCA1,K,Ye,DIST
      REAL X(4),F(4),F1(4),D(4,4),Yo(15),YSTOR(16),Y(1)DY(1),
     &XP(1),YP(1,1),W(1,13)
      A=(B+FM*X(2))*X(2)
      F(1)=(X(1)+V1)*A1-(X(1)+X(4)/A)*A
      F(2)=HCA1+(X(1)+V1)**2*A1/G-(.5*B+FM*X(2)/3.)
     &*X(2)**2-(X(1)+X(4)/A)**2*A/G
      AE=(B+FM*X(3))*X(3)
      F(3)=X(3)+(X(4)/AE)**2/G2-Yd-X(4)**2/Ad
      IF(K.EQ.1) THEN
      F(4)=X(3)-Ye
      ELSE
      Y(1)=X(3)
      CALL ODESOLF(Y,DY,1,FL,DIST,TOL,H11,HMIN,1,XP,YP,W,DYX,.TRUE.)
      F(4)=X(2)-Y(1)
      ENDIF
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1)
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,
      &HMIN,CUN,FMS,FM2,Q,A1,V1,HCA1,K,Ye,DIST
      REAL X(4),F(4),F1(4),D(4,4),Yo(15),YSTOR(16),Y(1),
     &DY(1),XP(1),YP(1,1),W(1,13)
      A=(B+FM*Y(1))*Y(1)
      SF=(Q*CUN*((B+FMS*Y(1))/A)**.6666667/A)**2
      DY(1)=(SO-SF)/(1.-Q**2*(B+FM2*Y(1))/(G*A**3))
      RETURN
      END
```

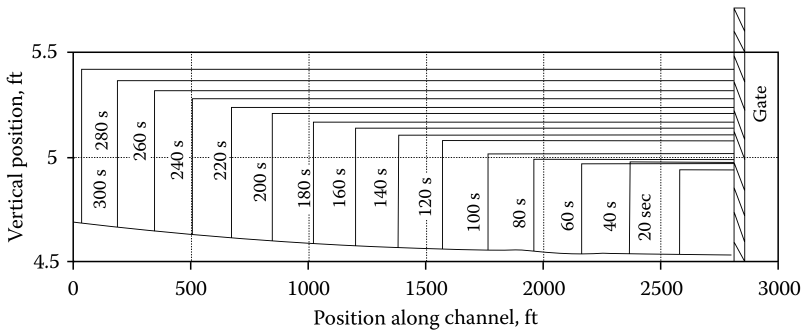
```
Program WAVEMOVR.C
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
float h,fke,yd,ad,g,g2,fl,b,fm,so,tol,*h11,cun,fms,\
   fm2,q,a1,v1,hca1,ye,dist;
int k;
FILE *fil1,*fil2; char fnam[20];
extern void rukust(int neq,float *dxs,float xbeg,\
   float xend,float\ error,float *y,float *ytt);
extern void solveq(int n,float **a,float *b,int itype,\
   float *dd,int *indx);
void fun1(float *f,float *x){
   float a,y[1],ytt[1];
   a=(b+fm*x[2])*x[2];
   f[0]=x[2]+pow(x[0]/a,2.)/g2-yd-x[0]*x[0]/ad;
   f[1]=h-x[1]-fke*pow(x[0]/((b+fm*x[1])*x[1]),2.)/g2; y[0]=x[2];
   rukust(1,h11,fl,0.,tol,y,ytt);
   f[2]=x[1]-y[0];} // End fun1
void fun2(float *f,float *x){
   float a,ae,y[1],ytt[1];
   a=(b+fm*x[1])*x[1]; f[0]=(x[0]+v1)*a1-(x[0]+x[3]/a)*a;
   f[1]=hca1+pow(x[0]+v1,2.)*a1/g-\
      (.5*b+fm*x[1]/3.)*x[1]*x[1]-\pow(x[0]+x[3]/a,2.)*a/g;
   ae=(b+fm*x[2])*x[2];
   f[2]=x[2]+pow(x[3]/ae,2.)/g2-yd-x[3]*x[3]/ad;
   if(k==1) f[3]=x[2]-ye; else {y[0]=x[2];
   rukust(1,h11,fl,dist,tol,y,ytt); f[3]=x[1]-y[0];} }
   // End fun2
void slope(float x,float *y,float *dydx){
   float a,sf;
   a=(b+fm*y[0])*y[0];
   sf=pow(q*cun*pow((b+fms*y[0])/a,.66666667)/a,2.);
   dydx[0]=(so-sf)/(1.-q*q*(b+fm2*y[0])/(g*pow(a,3.)));
   return;} // End slope
void main(void){ int i,j,jj,nt,nct,m,indx[4];
   float cu,cc,yg,fn,ad1,sum,xx,q1,dl,xx1,fac,dt,y1,*dd,yo[15],\
      ystor[16],y[1],ytt[1];
   float x[4],f[4],f1[4],**d; *h11=-.5;tol=1.e-5;
   printf("For initial condition give: g,H,Ke,Cc,Yg,L,n,b,m,So,\
      Guess for Q,Yb,Ye\n");
   scanf("%f %f %f %f %f %f %f %f %f %f %f %f %f",\
      &g,&h,&fke,&cc,&yg,&fl,&fn,&b,&fm,&so,&q,&x[1],&x[2]);
   fke+=1.;x[0]=q; fms=2.*sqrt(fm*fm+1.); fm2=2.*fm; yd=cc*yg;
   g2=2.*g;ad=pow((b+fm*yd)*yd,2.)*g2; ad1=ad;
   cu=1.486; if(g<20.) cu=1.; cun=fn/cu;
// q=x[0]; yb (upstream)=x[1]; ye (downstream)=x[2]
   nct=0;
   d=(float**)malloc(3*sizeof(float*));
   for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
   printf("Give output filenam1\n");scanf("%s",fnam);
```

```
  if((fil1=fopen(fnam,"w"))==NULL){
  printf("Cannot open output file\n");exit(0);}
L10: sum=0.; fun1(f,x);
   for(i=0;i<3;i++){ xx=x[i]; x[i]*=1.005; fun1(f1,x);
      for(j=0;j<3;j++) d[j][i]=(f1[j]-f[j])/(x[i]-xx);
         x[i]=xx;}
   solveq(3,d,f,1,dd,indx);
   for(i=0;i<3;i++){x[i]-=f[i];sum+=fabs(f[i]);}
   if((sum>5.e-5)&&(++nct<20)) goto L10;
   printf("Steady-State Q=%8.1f, Y1=%8.3f, Ye=%8.3f\n",\
      x[0],x[1],x[2]);
   fprintf(fil1,"Steady-State Q=%8.1f, Y1=%8.3f, Ye=%8.3f\n",\
      x[0],x[1],x[2]);
   for(i=0;i<3;i++) free (*d);
   d=(float**)malloc(4*sizeof(float*));
   for(i=0;i<4;i++)d[i]=(float*)malloc(4*sizeof(float));
   ye=x[2];y[0]=x[2];yo[14]=x[2]; q1=x[0];dl=fl/14.;xx=fl;
   for(i=13;i>=0;i–){xx1=xx-dl; rukust(1,h11,xx,xx1,tol,y,ytt);
      xx=xx1;yo[i]=y[0];}
   for(i=14;i>6;i–)fprintf(fil1,"%8.0f %8.3f",\
      dl*(float)i,yo[i]);fprintf(fil1,"\n");
   for(i=6;i>=0;i–)fprintf(fil1,"%8.0f %8.3f",\
      dl*(float)i,yo[i]);fprintf(fil1,"\n");
   printf("Give: No. time steps, Dt,New Yg\n");
   scanf("%d %f %f",&nt,&dt,&yg);
   printf("Give output filenam2\n");scanf("%s",fnam);
   if((fil2=fopen(fnam,"w"))==NULL){
     printf("Cannot open output file\n");exit(0);}
// v=x[0];y2=x[1];ye=x[2];q2=x[3]
   yd=cc*yg; ad=pow((b+fm*yd)*yd,2.)*g2;
   x[3]=q1*sqrt(ad/ad1);x[2]=1.2*yo[14];
   x[1]=.95*x[2];x[0]=1.2*sqrt(g*(b+fm*x[2])*x[2]/(b+fm2*x[2]));
   dist=fl-.1; *h11=-.5;
   for(k=1;k<=nt;k++){ m=(int)(dist/dl)+1;
      if(m<1) y1=yo[0];
      else {fac=(dist-(float)(m-1)*dl)/dl;
         y1=yo[m-1]+fac*(yo[m]-yo[m-1]);}
      a1=(b+fm*y1)*y1; hca1=(.5*b+fm*y1/3.)*y1*y1; v1=q1/a1;
      if(k==1) dist=fl;nct=0;
L20:sum=0.; q=x[3]; if(k==1) ye=x[2]; fun2(f,x);
   for(i=0;i<4;i++){xx=x[i];x[i]*=1.005;fun2(f1,x);
      for(j=0;j<4;j++)d[j][i]=(f1[j]-f[j])/(x[i]-xx);x[i]=xx;}
   solveq(4,d,f,1,dd,indx);
   for(i=0;i<4;i++){x[i]-=f[i];sum+=fabs(f[i]);}
   printf("%d %f\n",++nct,sum);
   if((sum>1.e-5)&&(nct<20)) goto L20;
   fprintf(fil1,"%7d %7.1f ",(int)(dt*(float)(k-1)),dist);
   for(j=0;j<3;j++)fprintf(fil1,"%7.3f",x[j]);
   fprintf(fil1,"%8.2f",x[3]);
   fprintf(fil1,"%8.3f %7.3f\n",y1,v1);
   if(k>1) { y[0]=x[2];xx=fl; j=0; *h11=-.5;
```

```
L30:xx1=xx-dl; if(xx1<dist) xx1=dist;
   rukust(1,h11,xx,xx1,tol,y,ytt);ystor[j++]=y[0];xx=xx1;
   if(xx1>dist)goto L30;jj=j-1;if(jj>7)jj=7;
   fprintf(fil2,"%6d",(int)(dt*(float)(k-1)));
   for(i=0;i<=jj;i++)
      fprintf(fil2,"%8.0f %7.3f",dl*(float)(14-i),ystor[i]);
   fprintf(fil2,"\n");
   if(j>8){  for(i=8;i<j;i++)fprintf(fil2,"%8.0f %7.3f",\
      dl*(float)(14-i),ystor[i]);
      fprintf(fil2,"\n");}}
   dist-=dt*x[0]; if(dist<0.) goto L50;} // end for(k
   L50:fclose(fil1);fclose(fil2); exit(0);}
```

EXAMPLE PROBLEM 4.68

A 2800 ft long trapezoidal channel with b = 10 ft, m = 1.5, n = 0.013, and S_o = 0.001 is feed by a constant head reservoir with H = 5 ft (K_e = 0). The gate, with a contraction coefficient C_c = 0.6, has it tip 2.5 ft above the channel bottom for a long time. Suddenly, the gate is dropped to Y_g = 1.5 ft. Track the resulting moving wave upstream from the gate using the explicit solution method and 20 s time increments.



Solution

The input to program WAVEMOV consists of

```
32.2 5. 0. .6 2.5 2800 .013 10 1.5 .001 296 4.8 7.4
30 20 1.5
```

The output files are

Steady state `Q = 356.6,` Y_1 `= 4.691,` Y_e `= 7.262`

Initial condition (x, Y)

```
2800. 7.262 2600. 7.068 2400. 6.874 2200. 6.681 2000. 6.490 1800. 6.299
1600. 6.110 1400. 5.923
1200. 5.737 1000. 5.554 800. 5.373 600. 5.196 400. 5.023 200. 4.854 0.
4.691
```

Time	X	V	Y_2	Y_e	Q	Y_1	V_1
0	2800.0	10.662	7.631	8.715	229.45	7.262	2.350
20	2586.8	10.513	7.481	7.693	214.06	7.055	2.456
40	2376.5	10.291	7.294	7.714	214.39	6.852	2.567
60	2170.7	10.068	7.113	7.737	214.74	6.653	2.682
80	1969.3	9.844	6.938	7.761	215.12	6.460	2.803
100	1772.4	9.620	6.769	7.787	215.52	6.273	2.929
120	1580.0	9.395	6.607	7.814	215.94	6.091	3.059
140	1392.1	9.170	6.451	7.843	216.39	5.915	3.194
160	1208.7	8.945	6.302	7.874	216.87	5.745	3.334
180	1029.8	8.722	6.159	7.908	217.38	5.581	3.478

(continued)

Time	X	V	Y_2	Y_e	Q	Y_1	V_1
200	855.4	8.500	6.024	7.943	217.93	5.423	3.626
220	685.4	8.280	5.895	7.981	218.51	5.272	3.777
240	519.8	8.063	5.773	8.021	219.12	5.126	3.932
260	358.5	7.848	5.659	8.064	219.77	4.988	4.090
280	201.6	7.638	5.551	8.109	220.46	4.855	4.250
300	48.8	7.434	5.451	8.158	221.19	4.731	4.409

2nd File

20	2800.	7.494	2600.	7.481												
40	2800.	7.516	2600.	7.317	2400.	7.294										
60	2800.	7.538	2600.	7.340	2400.	7.142	2200.	7.113								
80	2800.	7.562	2600.	7.364	2400.	7.166	2200.	6.968	2000.	6.938						
100	2800.	7.588	2600.	7.390	2400.	7.192	2200.	6.994	2000.	6.796	1800.	6.769				
120	2800.	7.615	2600.	7.417	2400.	7.219	2200.	7.021.	2000.	6.824	1800.	6.626	1600.	6.607		
140	2800.	7.645	2600.	7.446	2400.	7.248	2200.	7.050	2000.	6.853	1800.	6.655	1600.	6.459		
	1400.	6.451														
160	2800.	7.676	2600.	7.477	2400.	7.279	2200.	7.081	2000.	6.884	1800.	6.686	1600.	6.489		
	1400.	6.302														
180	2800.	7.709	2600.	7.511	2400.	7.312	2200.	7.114	2000.	6.917	1800.	6.719	1600.	6.523		
	1400.	6.326	1200.	6.159												
200	2800.	7.744	2600.	7.546	2400.	7.348	2200.	7.150	2000.	6.952	1800.	6.755	1600.	6.558		
	1400.	6.361	1200.	6.165	1000.	6.024										
220	2800.	7.782	2600.	7.584	2400.	7.386	2200.	7.188	2000.	6.990	1800.	6.792	1600.	6.595		
	1400.	6.399	1200.	6.202	1000.	6.007	800.	5.895								
240	2800.	7.823	2600.	7.624	2400.	7.426	2200.	7.228	2000.	7.030	1800.	6.832	1600.	6.635		
	1400.	6.439	1200.	6.242	1000.	6.047	800.	5.851	600.	5.773						
260	2800.	7.865	2600.	7.667	2400.	7.468	2200.	7.270	2000.	7.073	1800.	6.875	1600.	6.678		
	1400.	6.481	1200.	6.285	1000.	6.089	800.	5.894	600.	5.699	400.	5.659				
280	2800.	7.910	2600.	7.712	2400.	7.513	2200.	7.315	2000.	7.117	1800.	6.920	1600.	6.723		
	1400.	6.526	1200.	6.329	1000.	6.133	800.	5.938	600.	5.743	400.	5.551				
300	2800.	7.959	2600.	7.760	2400.	7.562	2200.	7.364	2000.	7.166	1800.	6.968	1600.	6.771		
	1400.	6.574	1200.	6.377	1000.	6.181	800.	5.986	600.	5.791	400.	5.597	200.	5.451		

4.24.2.2 Implicit Method

The explicit method extrapolates the wave velocity v from the current time step t ahead in time by Δt to find the next position x of the wave at time $t + \Delta t$. Implicit methods utilize the wave velocity at the advanced time, as well as at the current time. If we denote the wave velocity with a subscript k corresponding to time t, then this velocity at the current time step is v_k, and at the advanced time step it is v_{k+1}. A better estimate of the movement Δx of the wave during the time increment Δt from t to $t + \Delta t$ is $\Delta x = \Delta t(v_k + v_{k+1})/2$, or the change in wave position is the product of the time increment and the average wave velocity at the current and advanced time steps. However, since v_{k+1} will depend upon the other variables at the advanced time step, such as the position of the wave, additional variables must be added to the list of unknowns. In the implicit method described in this section, the position x of the wave at the advanced time step will be the only variable added to the list of unknown variables, so the unknowns become: v, Y_2, Y_e, Q_2, and x_{k+1}. The position of the wave at time t is x_k. Variables $(Y_1)_{k+1}$ and $(V_1)_{k+1}$ might be added to this list of unknowns; however, since a direct relationship exists between these and x_{k+1} from the interpolation of the initial condition, they will not be added to the list. Thus, rather than solving four simultaneous equations as was done in the explicit method, the implicit method will solve five simultaneous equations for the above five unknown variables. The fifth equation added to the previous first four equations is

$$F_5 = x_k - \Delta t(v_{k+1} + v_k)/2 - x_{k+1} = 0 \tag{6d}$$

The program WAVEMOVI.FOR (WAVEMVRI.C) implements this implicit method for a DCW. At time t = 0, the solution is needed to obtain the height of the wave that the closure of the gate causes, but when the wave is still at the gate. Rather than handling this solution by having a special equation when K = 1 as was done in program WAVEMOV, program WAVEMOVI solves the three variables v, Y_2, and V_2, at the gate with the lowered gate position as was done in Chapter 3. The three equations solved to give these three variables are the continuity and momentum equations from the viewpoint of an observer moving with the wave, and the energy equation across the gate as seen by a stationary observer. The subroutine FUN0 has been added to the program to evaluate these three equations. Note that in program WAVEMOVI, the arrays' sizes have been increased to 5. The variables in array X(5) are as follows: v = X(1), Y_2 = X(2), Y_e = X(3), Q_2 = X(4), and x = X(5), where both v and x are at the advanced time step, i.e., their subscript would be k+1 if it were given. Note also that Equation 4.6d above has been added to subroutine FUN2 so this subroutine now returns five equation values. Once the new solution has been obtained for the advanced time step, the solution for x (from X(5)) is transferred to the variable XPOS in the third statement from the end of the DO 50 loop, so this variable represents x_k.

Program WAVEMOVI.FOR

```
C Uses an Implicit method to solve a moving
C wave upstream from a gate (DCW)
      LOGICAL NFS
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,HMIN,CUN,F
     &MS,FM2,Q,A1,Q1,V1,HCA1,WV1,XPOS,Yo(15),DL,DT5
      REAL X(5),F(5),F1(5),D(5,5),YSTOR(16),Y(1),DY(1),XP(1),
     &YP(1,1),W(1,13)
      INTEGER*2 INDX(5)
      EQUIVALENCE (v,X(1)),(Y2,X(2)),(Qe,X(4))
      WRITE(*,*)' For initial condition give:' ' g,H,Ke,Cc,Yg,
     &L,n,b,m,So, Guess for Q,Y1,Y2'
      H11=-.5
      HMIN=1.E-5
      TOL=1.E-5
      READ(*,*) G,H,FKe,Cc,Yg,FL,FN,B,FM,So,Q,Y2,X(3)
      FKe=FKe+1.
      X(1)=Q
      FMS=2.*SQRT(FM*FM+1.)
      FM2=2.*FM
      Yd=Cc*Yg
      G2=2.*G
      Ad=((b+FM*Yd)*Yd)**2*G2
      Ad1=Ad
      Cu=1.486
      IF(G.LT.20.) Cu=1.
      CUN=FN/Cu
C Q=X(1); Yb (upstream)=X(2); Ye (downstream)= X(3)
      NCT=0
```

```
10    SUM=0.
      CALL FUN1(F,X)
      DO 12 I=1,3
      XX=X(I)
      X(I)=1.005*X(I)
      CALL FUN1(F1,X)
      DO 11 J=1,3
11    D(J,I)=(F1(J)-F(J))/(X(I)-XX)
12    X(I)=XX
      CALL SOLVEQ(3,1,5,D,F,1,DD,INDX)
      DO 14 I=1,3
      X(I)=X(I)-F(I)
14    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' 1ST NCT=',NCT,SUM
      IF(SUM.GT.5.E-5 .AND. NCT.LT.20) GO TO 10
      WRITE(3,130)(X(I),I=1,3)
130   FORMAT(' Steady-State Q=',F8.1,', Y1=',F8.3,', Ye=',F8.3)
      Y(1)=X(3)
      Yo(15)=X(3)
      Q1=X(1)
      DL=FL/14.
      XX=FL
      NFS=.TRUE.
      DO 15 I=14,1,-1
      XX1=XX-DL
      CALL ODESOLF(Y,DY,1,XX,XX1,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      NFS=.FALSE.
      XX=XX1
15    Yo(I)=Y(1)
      WRITE(3,100) (DL*FLOAT(I-1),Yo(I),I=15,1,-1)
100   FORMAT(' Initial Condition (x,Y)',/,8(F8.0,F8.3),/,7(F8.0,F8.3))
      WRITE(*,*)' Give:No. time steps,Dt,New Yg'
      READ(*,*) NT,DT,Yg
      DT5=.5*DT
C At gate variables: v=X(1), Y2=X(2), V2=X(3)
      HCA1=(.5*B+FM*X(3)/3.)*X(3)**2
      A1=(B+FM*X(3))*X(3)
      V1=Q1/A1
      Yd=Cc*Yg
      Ad=((B+FM*Yd)*Yd)**2*G2
      X(2)=1.1*X(3)
      Q2=Q1*SQRT(Ad/Ad1)
      X(3)=Q2/((B+FM*X(2))*X(2))
      X(1)=9.*(V1-X(3))
      NCT=0
16    SUM=0.
      CALL FUN0(F,X)
      DO 18 I=1,3
      XX=X(I)
      X(I)=1.005*X(I)
```

```
      CALL FUN0(F1,X)
      DO 17 J=1,3
17    D(J,I)=(F1(J)-F(J))/(X(I)-XX)
18    X(I)=XX
      CALL SOLVEQ(3,1,5,D,F,1,DD,INDX)
      DO 19 I=1,3
      X(I)=X(I)-F(I)
19    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' 2ND NCT=',NCT,SUM,(X(I),I=1,3)
      IF(SUM.GT.5.E-5 .AND. NCT.LT.20) GO TO 16
      WRITE(3,140)(X(I),I=1,3)
140   FORMAT(' Wave at gate, v=',F9.3,', Y2=',F8.3,', V2=',F8.3)
      WV1=X(1)
C v=X(1); Y2=X(2); Ye=X(3); Q2=X(4); Position x of wave=X(5)
      WRITE(3,111)
111   FORMAT(/,'  Time  v  Y2 Ye Q',
     &' x  V2  Y1', ' V1',/,1X,70('-'))
      X(4)=X(3)*((B+FM*X(2))*X(2))
      WRITE(3,110) 0,WV1,X(2),X(2),X(4),FL,X(3),Yo(15),V1
      X(3)=X(2)
      XPOS=FL
      X(5)=XPOS-WV1*DT
      H11=-.5
      DO 50 K=1,NT
      NCT=0
20    SUM=0.
      Q=X(4)
      CALL FUN2(F,X)
      DO 22 I=1,5
      XX=X(I)
      X(I)=1.005*X(I)
      CALL FUN2(F1,X)
      DO 21 J=1,5
21    D(J,I)=(F1(J)-F(J))/(X(I)-XX)
22    X(I)=XX
      CALL SOLVEQ(5,1,5,D,F,1,DD,INDX)
      DO 24 I=1,5
      X(I)=X(I)-F(I)
24    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,SUM
      IF(SUM.GT.5.E-5 .AND. NCT.LT.20) GO TO 20
      M=IFIX(X(5)/DL)+2
      IF(M.LT.2) THEN
      Y1=Yo(1)
      ELSE
      FAC=(X(5)-FLOAT(M-2)*DL)/DL
      Y1=Yo(M-1)+FAC*(Yo(M)-Yo(M-1))
      ENDIF
      V1=Q1/((B+FM*Y1)*Y1)
      WRITE(3,110) IFIX(DT*FLOAT(K)),X,X(4)/((B+FM*X(2))*X(2)),
     &Y1,V1
```

```
110   FORMAT(I7,4F8.3,F8.1,3F8.3)
      IF(K.EQ.1) GO TO 40
      NFS=.FALSE.
      Y(1)=X(3)
      XX=FL
      J=0
30    J=J+1
      XX1=XX-DL
      IF(XX1.LT.X(5)) XX1=X(5)
      NFS=.TRUE.
      CALL ODESOLF(Y,DY,1,XX,XX1,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      YSTOR(J)=Y(1)
      XX=XX1
      NFS=.FALSE.
      IF(XX1.GT.X(5)) GO TO 30
      WRITE(4,120) IFIX(DT*FLOAT(K-1)),(DL*FLOAT(15-I),YSTOR(I),
     &I=1,J)
120   FORMAT(I6,8(F8.0,F8.3),/,7(F8.0,F8.3))
40    XPOS=X(5)
      WV1=X(1)
      X(5)=X(5)-DT*X(1)
      IF(X(5).LT.0.) STOP
50    CONTINUE
      END
      SUBROUTINE FUN1(F,X)
      REAL Y(1),DY(1),XP(1),YP(1,1),W(1,13),X(5),F(5)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,HMIN,CUN,
     &FMS,FM2,Q,A1,Q1,V1,HCA1,WV1,XPOS,Yo(15),DL,DT5
      A=(B+FM*X(3))*X(3)
      F(1)=X(3)+(X(1)/A)**2/G2-Yd-X(1)**2/Ad
      F(2)=H-X(2)-FKe*(X(1)/((B+FM*X(2))*X(2)))**2/G2
      Y(1)=X(3)
      CALL ODESOLF(Y,DY,1,FL,0.,TOL,H11,HMIN,1,XP,YP,W,DYX,.TRUE.)
      F(3)=X(2)-Y(1)
      RETURN
      END
      SUBROUTINE FUN2(F,X)
      REAL Y(1),DY(1),XP(1),YP(1,1),W(1,13),X(5),F(5)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,HMIN,CUN,
     &FMS,FM2,Q,A1,Q1,V1,HCA1,WV1,XPOS,Yo(15),DL,DT5
      A=(B+FM*X(2))*X(2)
      M=IFIX(X(5)/DL)+2
      IF(M.GT.15) M=15
      IF(M.LT.2) THEN
      Y1=Yo(1)
      ELSE
      FAC=(X(5)-FLOAT(M-2)*DL)/DL
```

```
      Y1=Yo(M-1)+FAC*(Yo(M)-Yo(M-1))
      ENDIF
      A1=(B+FM*Y1)*Y1
      HCA1=(.5*B+FM*Y1/3.)*Y1**2
      V1=Q1/A1
      F(1)=(X(1)+V1)*A1-(X(1)+X(4)/A)*A
      F(2)=HCA1+(X(1)+V1)**2*A1/G-(.5*B+FM*X(2)/3.)*X(2)
     &**2-(X(1)+X(4)/A)**2*A/G
      AE=(B+FM*X(3))*X(3)
      F(3)=X(3)+(X(4)/AE)**2/G2-Yd-X(4)**2/Ad
      Y(1)=X(3)
      CALL ODESOLF(Y,DY,1,FL,X(5),TOL,H11,HMIN,1,XP,YP,W,DYX,
     &.TRUE.)
      F(4)=X(2)-Y(1)
      F(5)=XPOS-DT5*(X(1)+WV1)-X(5)
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1)
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,HMIN,CUN,
     &FMS,FM2,Q,A1,Q1,V1,HCA1,WV1,XPOS,Yo(15),DL,DT5
      A=(B+FM*Y(1))*Y(1)
      SF=(Q*CUN*((B+FMS*Y(1))/A)**.6666667/A)**2
      DY(1)=(SO-SF)/(1.-Q**2*(B+FM2*Y(1))/(G*A**3))
      RETURN
      END
      SUBROUTINE FUN0(F,X)
      REAL X(5),F(5)
      COMMON /TRAS/ H,FKe,Yd,Ad,G,G2,FL,B,FM,SO,TOL,H11,HMIN,CUN,
     &FMS,FM2,Q,A1,Q1,V1,HCA1,WV1,XPOS,Yo(15),DL,DT5
      A=(B+FM*X(2))*X(2)
      F(1)=(X(1)+V1)*A1-(X(1)+X(3))*A
      F(2)=HCA1+(X(1)+V1)**2*A1/G-(.5*B+FM*X(2)/3.)*X(2)**2-
     &(X(1)+X(3))**2*A/G
      F(3)=X(2)+X(3)**2/G2-Yd-(X(3)*A)**2/Ad
      RETURN
      END
```

Program WAVEMVRI.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
float h,fke,yd,ad,g,g2,fl,b,fm,so,tol,*h11,cun,fms,fm2,q,a1,\
   q1,v1, hca1,wv1,xpos,yo[15],dl,dt5;
FILE *fil1,*fil2; char fnam[20];
extern void rukust(int neq,float *dxs,float xbeg,\
   float xend,float error,float *y,float *ytt);
extern void solveq(int n,float **a,float *b,int itype,\
   float *dd,int *indx);
void fun1(float *f,float *x){float a,y[1],ytt[1];
   a=(b+fm*x[2])*x[2];
   f[0]=x[2]+pow(x[0]/a,2.)/g2-yd-x[0]*x[0]/ad;
```

```
   f[1]=h-x[1]-fke*pow(x[0]/((b+fm*x[1])*x[1]),2.)/g2;
   y[0]=x[2]; rukust(1,h11,fl,0.,tol,y,ytt);
   f[2]=x[1]-y[0];} // End fun1
void fun2(float *f,float *x){
   float a,ae,y1,fac,y[1],ytt[1]; int m;
   a=(b+fm*x[1])*x[1]; m=(int)(x[4]/dl)+1; if(m>14)m=14;
   if(m<1) y1=yo[0];
   else {fac=(x[4]-(float)(m-1)*dl)/dl;
      y1=yo[m-1]+fac*(yo[m]-yo[m-1]);}
   a1=(b+fm*y1)*y1; hca1=(.5*b+fm*y1/3.)*y1*y1; v1=q1/a1;
   f[0]=(x[0]+v1)*a1-(x[0]+x[3]/a)*a;
   f[1]=hca1+pow(x[0]+v1,2.)*a1/g-\
      (.5*b+fm*x[1]/3.)*x[1]*x[1]-\pow(x[0]+x[3]/a,2.)*a/g;
   ae=(b+fm*x[2])*x[2];
   f[2]=x[2]+pow(x[3]/ae,2.)/g2-yd-x[3]*x[3]/ad; y[0]=x[2];
   rukust(1,h11,fl,x[4],tol,y,ytt); f[3]=x[1]-y[0];
   f[4]=xpos-dt5*(x[0]+wv1)-x[4]; } // End fun2
void funo(float *f,float *x){ float a;
   a=(b+fm*x[1])*x[1]; f[0]=(x[0]+v1)*a1-(x[0]+x[2])*a;
   f[1]=hca1+pow(x[0]+v1,2.)*a1/g-\
      (.5*b+fm*x[1]/3.)*x[1]*x[1]-\pow(x[0]+x[2],2.)*a/g;
   f[2]=x[1]+x[2]*x[2]/g2-yd-pow(x[2]*a,2)/ad; } // End funo
void slope(float x,float *y,float *dydx){
   float a,sf;
   a=(b+fm*y[0])*y[0];
   sf=pow(q*cun*pow((b+fms*y[0])/a,.66666667)/a,2.);
   dydx[0]=(so-sf)/(1.-q*q*(b+fm2*y[0])/(g*pow(a,3.)));
   return;} // End slope
void main(void){ int i,j,k,jj,nt,nct,m,indx[4];
   char *fmt="%7d %7.3f %7.3f %7.3f %7.2f %7.1f %7.3f \
      %7.3f %7.3f\n";
   float cu,cc,yg,fn,sum,xx,xx1,fac,dt,y1,ad1,q2,*dd,ystor[16],\
      y[1],\ytt[1];
   float x[5],f[5],f1[5],**d; *h11=-.5;tol=1.e-5;
   printf("For initial condition give: g,H,Ke,Cc,Yg,L,n,b,m,So,\
      Guess for Q,Yb,Ye\n");
   scanf("%f %f %f %f %f %f %f %f %f %f %f %f %f",\
      &g,&h,&fke,&cc,&yg,\&fl,&fn,&b,&fm,&so,&q,&x[1],&x[2]);
   fke+=1.;x[0]=q; fms=2.*sqrt(fm*fm+1.); fm2=2.*fm; yd=cc*yg;
   g2=2.*g;ad=pow((b+fm*yd)*yd,2.)*g2;ad1=ad; cu=1.486;
   if(g<20.) cu=1.; cun=fn/cu;
// q=x[0]; yb (upstream)=x[1]; ye (downstream)=x[2]
   nct=0;
   d=(float**)malloc(3*sizeof(float*));
   for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
   printf("Give output filenam1\n");scanf("%s",fnam);
   if((fil1=fopen(fnam,"w"))==NULL){
      printf("Cannot open output file\n");exit(0);}
L10: sum=0.; fun1(f,x);
   for(i=0;i<3;i++){xx=x[i]; x[i]*=1.005; fun1(f1,x);
      for(j=0;j<3;j++) d[j][i]=(f1[j]-f[j])/(x[i]-xx); x[i]=xx;}
   solveq(3,d,f,1,dd,indx);
```

```
   for(i=0;i<3;i++){x[i]-=f[i]; sum+=fabs(f[i]);}
   if((sum>5.e-5)&&(++nct<20)) goto L10;
   printf("Steady-State Q=%8.1f, Y1=%8.3f, Ye=%8.3f\n",\
      x[0],x[1],x[2]);
   fprintf(fil1,"Steady-State Q=%8.1f, Y1=%8.3f, Ye=%8.3f\n",\
      x[0],x[1],x[2]);
   y[0]=x[2];yo[14]=x[2]; q1=x[0];dl=fl/14.;xx=fl;
   for(i=13;i>=0;i–){xx1=xx-dl; rukust(1,h11,xx,xx1,tol,y,ytt);
      xx=xx1;yo[i]=y[0];}
   for(i=14;i>6;i–)
      fprintf(fil1,"%8.0f %8.3f",dl*(float)i,yo[i]);
   fprintf(fil1,"\n");
   for(i=6;i>=0;i–)
      fprintf(fil1,"%8.0f %8.3f",dl*(float)i,yo[i]);
   fprintf(fil1,"\n");
   printf("Give: No. time steps, Dt,New Yg\n");
   scanf("%d %f %f",&nt,&dt,&yg);
   printf("Give output filenam2\n");
   scanf("%s",fnam); dt5=.5*dt;
   if((fil2=fopen(fnam,"w"))==NULL){
      printf("Cannot open output file\n");exit(0);}
// At gate variables: v=x[0]; y2=x[1]; v2=x[2]
   hca1=(.5*b+fm*x[2]/3.)*x[2]*x[2]; a1=(b+fm*x[2])*x[2];
   v1=q1/a1;yd=cc*yg;ad=pow((b+fm*yd)*yd,2.)*g2;
   x[1]=1.1*x[2]; q2=q1*sqrt(ad/ad1);
   x[2]=q2/((b+fm*x[1])*x[1]);x[0]=9.*(v1-x[2]);
   nct=0; L16: sum=0.; funo(f,x);
   for(i=0;i<3;i++){xx=x[i]; x[i]*=1.005;funo(f1,x);
      for(j=0;j<3;j++) d[j][i]=(f1[j]-f[j])/(x[i]-xx); x[i]=xx;}
   solveq(3,d,f,1,dd,indx);
   for(i=0;i<3;i++){x[i]-=f[i]; sum+=fabs(f[i]);}
   if((sum>5.e-5)&&(++nct<20)) goto L16;
   fprintf(fil1," Wave at gate, v=%9.3f, Y2=%8.3f, V2=%8.3f\n",\
      x[0],x[1],x[2]);
   wv1=x[0]; for(i=0;i<3;i++) free(*d);
   d=(float**)malloc(5*sizeof(float*));
   for(i=0;i<5;i++)d[i]=(float*)malloc(5*sizeof(float));
// v=x[0];y2=x[1];ye=x[2];q2=x[3]; Position x of wave=x[4]
   fprintf(fil1,"n  Time  v  Y2  Ye  Q  x  V2  Y1  \
   V1\n");
   for(i=0;i<70;i++) fprintf(fil1,"-"); fprintf(fil1,"\n");
   x[3]=x[2]*((b+fm*x[1])*x[1]);
   fprintf(fil1,fmt,0,wv1,x[1],x[1],x[3],fl,x[2],yo[14],v1);
   x[2]=x[1]; xpos=fl; x[4]=xpos-wv1*dt;*h11=-.5;
   for(k=1;k<=nt;k++){ nct=0;
L20:sum=0.; q=x[3]; fun2(f,x);
      for(i=0;i<5;i++){xx=x[i];x[i]*=1.005;fun2(f1,x);
         for(j=0;j<5;j++)d[j][i]=(f1[j]-f[j])/(x[i]-xx);x[i]=xx;}
      solveq(5,d,f,1,dd,indx);
      for(i=0;i<5;i++){x[i]-=f[i]; sum+=fabs(f[i]);}
   printf("%d %f\n",++nct,sum);
   if((sum>5.e-5)&&(nct<20)) goto L20;
   m=(int)(x[4]/dl)+1;
```

```
   if(m<2) y1=yo[0]; else {fac=(x[4]-(float)\(m-1)*dl)/dl;
      y1=yo[m-1]+fac*(yo[m]-yo[m-1]);}
   v1=q1/((b+fm*y1)*y1);
   fprintf(fil1,fmt,(int)(dt*k),x[0],x[1],x[2],x[3],x[4],x[3]/\
      ((b+fm*x[1])*x[1]),y1,v1);
   if(k>1) { y[0]=x[2];xx=fl; j=0; *h11=-.5;
L30:xx1=xx-dl; if(xx1<x[4]) xx1=x[4];
      rukust(1,h11,xx,xx1,tol,y,ytt);ystor[j++]=y[0];xx=xx1;
      if(xx1>x[4])goto L30;jj=j-1;if(jj>7)jj=7;
      fprintf(fil2,"%6d",(int)(dt*(float)(k-1)));
      for(i=0;i<=jj;i++)
      fprintf(fil2,"%8.0f %7.3f",dl*(float)(14-i),ystor[i]);
      fprintf(fil2,"\n");
      if(j>8){for(i=8;i<j;i++)
            fprintf(fil2,"%8.0f\ %7.3f",dl*(float)(14-i),ystor[i]);
         fprintf(fil2,"\n");}}
   xpos=x[4];wv1=x[0]; x[4]-=dt*x[0]; if(x[4]<0.) goto L50;}
   // end for(k
L50:fclose(fil1);fclose(fil2); exit(0);}
```

EXAMPLE PROBLEM 4.69

Repeat the previous example problem, but solve it using the implicit method described above.

Solution

The input to Program WAVEMOVI is identical to that given in the previous problem. The first output files consists of

File 1

Steady state `Q = 356.6,` Y_1 `= 4.691,` Y_e `= 7.262`

Initial condition (x, Y)

```
2800. 7.262 2600. 7.068 2400. 6.874 2200. 6.681 2000. 6.490 1800.
 6.299  1600. 6.110 1400. 5.923
1200. 5.737 1000. 5.554  800. 5.373  600. 5.196  400. 5.023 200.
 4.854  0. 4.691
```

Wave at gate, `v = 10.732,` Y_2`= 7.673,` V_2`= 1.295`

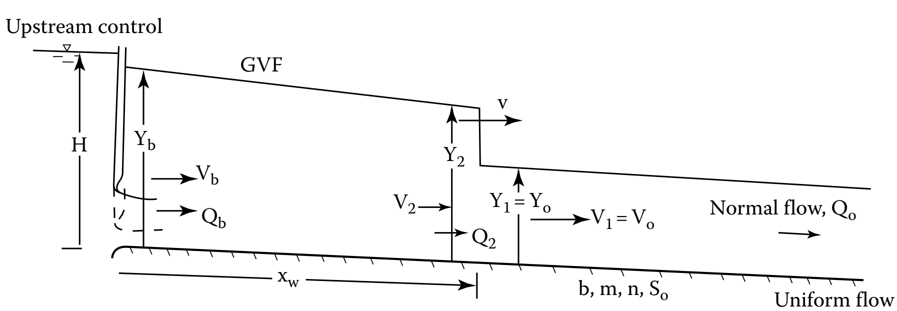
Time	v	Y_2	Y_e	Q	x	V_2	Y_1	V_1
0	10.732	7.673	7.673	213.748	2800.0	1.295	7.262	2.350
20	10.514	7.482	7.693	214.057	2587.5	1.348	7.056	2.455
40	10.295	7.297	7.714	214.358	2379.4	1.403	6.854	2.565
60	10.074	7.117	7.736	214.733	2175.8	1.459	6.658	2.680
80	9.852	6.944	7.760	215.103	1976.5	1.517	6.467	2.799
100	9.630	6.777	7.785	215.495	1781.7	1.577	6.282	2.923
120	9.408	6.616	7.812	215.913	1591.3	1.638	6.102	3.051
140	9.186	6.462	7.841	216.356	1405.3	1.700	5.928	3.184
160	8.964	6.314	7.872	216.829	1223.8	1.764	5.759	3.322
180	8.743	6.173	7.904	217.332	1046.8	1.828	5.597	3.464
200	8.524	6.038	7.939	217.867	874.1	1.893	5.440	3.609
220	8.307	5.910	7.976	218.434	705.8	1.959	5.290	3.759
240	8.092	5.789	8.016	219.035	541.8	2.025	5.145	3.911
260	7.880	5.675	8.057	219.669	382.1	2.091	5.007	4.067
280	7.672	5.568	8.102	220.345	226.6	2.156	4.876	4.223
300	7.470	5.468	8.149	221.063	75.2	2.221	4.752	4.381

File 2

20	2800.	7.515	2600.	7.317	2400.	7.297								
40	2800.	7.538	2600.	7.339	2400.	7.141	2200.	7.117						
60	2800.	7.561	2600.	7.363	2400.	7.165	2200.	6.967	2000.	6.944				
80	2800.	7.587	2600.	7.388	2400.	7.190	2200.	6.993	2000.	6.795	1800.	6.777		
100	2800.	7.614	2600	7.415	2400.	7.217	2200.	7.019	2000.	6.822	1800.	6.625	1600.	6.616
120	2800.	7.642	2600.	7.444	2400.	7.246	2200.	7.048	2000.	6.851	1800.	6.653	1600.	6.462
140	2800.	7.673	2600.	7.475	2400.	7.277	2200.	7.079	2000.	6.881	1800.	6.684	1600.	6.487
	1400.	6.314												
160	2800.	7.706	2600.	7.507	2400.	7.309	2200.	7.111	2000.	6.914	1800.	6.716	1600.	6.519
	1400.	6.323	1200.	6.173										
180	2800.	7.741	2600.	7.542	2400.	7.344	2200.	7.146	2000.	6.948	1800.	6.751	1600.	6.554
	1400.	6.357	1200.	6.161	1000.	6.038								
200	2800.	7.778	2600.	7.579	2400.	7.381	2200.	7.183	2000.	6.985	1800.	6.788	1600.	6.591
	1400.	6.394	1200.	6.198	1000.	6.002	800.	5.910						
220	2800.	7.817	2600.	7.618	2400.	7.420	2200.	7.222	2000.	7.024	1800.	6.827	1600.	6.630
	1400.	6.433	1200.	6.237	1000.	6.041	800.	5.846	600.	5.789				
240	2800.	7.859	2600.	7.660	2400.	7.462	2200.	7.264	2000.	7.066	1800.	6.868	1600.	6.671
	1400.	6.474	1200.	6.278	1000.	6.082	800.	5.887	600.	5.693	400.	5.675		
260	2800.	7.903	2600.	7.705	2400.	7.506	2200.	7.308	2000.	7.110	1800.	6.913	1600.	6.715
	1400.	6.518	1200.	6.322	1000.	6.126	800.	5.931	600.	5.736	400.	5.568		
280	2800.	7.950	2600.	7.752	2400.	7.554	2200.	7.355	2000.	7.157	1800.	6.960	1600.	6.762
	1400.	6.566	1200.	6.369	1000.	6.173	800.	5.978	600.	5.783	400.	5.589	200.	5.468

4.24.3 Upstream Controlled Waves

Waves that move downstream with a velocity v are caused by an upstream control, and as in Chapter 3, we will denote them as UCW. Consider first an UCW caused by suddenly increasing the flow rate coming into a channel by instantly opening a gate further, as shown in the sketch below, and as was handled in Chapter 3. In real applications, there will be a GVF between the upstream control and the position of the wave since the new depth caused by raising the gate will generally not equal the normal depth, i.e., the flow will not be uniform downstream from the gate to the position x_w of the wave. Rather, because of the larger flow rate, a steeper slope of energy line is needed than the bottom slope of the channel, resulting in a decreasing depth from the beginning of the channel to the position of the wave, generally. Thus, the wave will not have a constant height $Y_2 - Y_1$, or move with a constant velocity v. The available equations are



Continuity from the viewpoint of an observer moving with the wave

$$F_1 = (v - V_1)A_1 - (v - V_2)A_2 = 0 \tag{1u}$$

Momentum from the viewpoint of an observer moving with the wave

$$F_2 = (h_cA)_1 + \frac{(v - V_1)^2 A_1}{g} - (h_cA)_2 + \frac{(v - V_2)^2 A_2}{g} \tag{2u}$$

Energy across the gate (or other upstream control condition)

$$F_e = H - Y_b - (1 + K_e)\frac{Q_b^2}{2gAb^2} = 0 \tag{3u}$$

GVF from the gate to the position of the wave

$$F_4 = Y_2 - Y_{2ode}(Y_b) = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f + Qq_o^*/(gA^2)}{1 - F_r^2} \tag{4u}$$

(The u after the equation number stands for UCW.)

Equations 4.1u and 4.2u are written with v first within the parentheses because v is larger than V_1 or V_2, but the equations are also valid if v comes second within the ( ), or $(V_i - v)$. Note that Equations 4.1d and 2d could be written as Equations 4.1u and 4.2u, i.e., the + changed to a—if v in Equations 4.1d and 4.2 d were taken as a negative values, which is justified since v for a DCW is opposite to the positive x axis. Also, note that V_2 in Equations 4.1u and 4.2u can be replaced by Q_2/A_2. Equation 3u applies for a gate, but will need to be altered to reflect the upstream control if other than a gate. We might assume that the entrance loss coefficient K_e times the downstream velocity head or $K_e\{V_b^2/(2g)\}$ gives the headloss caused by the gate when the flow downstream from the gate is submerged, as shown in the above sketch. The next chapter provides better methods for handling submerged flows past gates. Notice that the ODE solved in connection with Equation 4.4u contains the lateral outflow term $Qq_o^*/(gA^2)$, since in general the flow rate Q will vary from the beginning of the channel to the wave and in real situations, the nature of this variation will likely depend upon how far the wave has moved downstream from the gate, with the flow rate nearly constant when x_w is small and a considerable drop in Q as x_w becomes large. For lack of empirical data to define this variation we will assume a linear variation, or

$$Q = Q_b + (x/x_w)(Q_2 - Q_b) \ = \ Q_b - (x/x_w)(Q_b - Q_2).$$

Under this assumption, the lateral outflow term $q_o^* = (Q_b - Q_2)/x_w$. q_o^* is not actually an outflow, but since the flow rate Q varies with x, it must be included when solving the ODE for the GVF from the beginning of the channel to the position x_w of the wave.

Consider the simplest case in which the flow rate past the gate is initially Q_o. Assume the channel is very long; then the initial condition consists of a uniform flow throughout the channel. Suddenly, the flow rate at the beginning of the channel is instantly increased to $Q_b > Q_o$. This increase in Q will cause an UCW to occur. As it first forms, the increased depth $Y_2 = Y_b > Y_o$ and the wave velocity v will be determined by solving Equations 4.1u and 4.2u simultaneously, or Equation 3.36 followed by Equation 3.30. As the wave moves downstream from the gate, the depth Y_2 immediately upstream from the wave will differ from Y_b, and also the wave speed v will change as dictated by the GVF. (A GVF will occur since in general the increased flow rate Q_b will cause a depth Y_b different from the normal depth associated with this flow rate.) Since Q_b is specified, Equation 4.3u is not available, so three variables can be solved from Equations 4.1u, 4.2u, and 4.4u. Typically, the three unknown variables are v, Y_2, and Q_2 (or V_2). If H is specified rather than Q_b and K_e, then Equation 4.3u would be solved first for Q_b.

Programs WAVEUCE and WAVEUCI are designed to obtain solutions for v, Y_2, and Q_2 from Equations 4.1u, 4.2u, and 4.4u, using the explicit and implicit methods, respectively. The array X in program WAVEUCE.FOR contains the unknowns as follows: v = X(1), Y_2 = X(2), and Q_2 = X(3). (In program WAVEUCE.C, the subscripts are reduced by 1.) The arrays F(3) and F1(3) are used to hold the values of the three equations, and D(3,3) is for the Jacobian in implementing a Newton method. In implementing the implicit method in Program WAVEUCI, these arrays are dimensioned by 4 because x_w is added as an additional unknown in x_w = X(4), and the additional equation is added to the other three simultaneous equations to solve

$$F_S = (x_w)^{k-1} + \Delta t(v_{k-1} + v_k)/2 - (x_w)^k = 0 \tag{5u}$$

In both programs, Manning's equation is solved first using the specified channel geometry (for a trapezoidal section), and n and S_o for the normal depth Y_2 by the Newton method. Thereafter, the new flow rate Q_b is read, Equation 3.36 is solved for the new depth $Y_b = Y_2$ by the Newton method, and from Y_2 the wave velocity v is solved from Equation 3.30. Then for the number of specified time steps NT, with an increment DT = Δt apart, the three or four equations above are solved simultaneously to determine the time-dependent movement of the wave. Note in subroutine SLOPE that the assumption is made that the flow rate Q varies linearly with x between the beginning of the channel Q_b and the position of the wave. In Program WAVEUCE (the explicit method), this position x_w is given by DIST. In program WAVEUCI (the implicit method) this position x_w = X(4) is the fourth value being solved and the previous time step position of the wave $(x_w)_{k-1}$ is in XPOS, and the previous time step wave velocity v_{k-1} is the value of WV1.

When the wave is a longer distance downstream, then the flow rate Q_2 immediately upstream from the wave will approach the initial steady-state flow rate Q_o, and the upstream and downstream Froude number, as seen by the moving observer, will approach unity, e.g., the height of the wave gets smaller and the critical flow conditions occur from the moving observer's viewpoint. In practice, the moving wave disappears, and is reflected in the failure in solving the system of equations. Both programs check when the Froude number immediately downstream from the wave, as seen by the moving observer, gets close to unity and allows the user to terminate the solution when this occurs before the specified number of time step solutions have been obtained.

Program WAVEUCE.FOR

```
C Solves UCW with upstream gate instantly raised with uniform
C flow downstream
      INTEGER*2 INDX(3)
      REAL F(3),F1(3),D(3,3)
      COMMON B,FM,FM2,FMS,FM3,BH,Yb,Vo,Ao,HCA1,G,G2,Qb,DXS,DIST,
     &X(3),CUN,SO
      WRITE(*,*)' Give: g,b,m,n,So,Qo & Guess for Yo'
      READ(*,*) G,B,FM,FN,SO,Qo,Yo
      BH=.5*B
      FM3=FM/3.
      FM2=2.*FM
      FMS=2.*SQRT(FM*FM+1.)
      G2=2.*G
      CU=1.486
      IF(G.LT.20.) CU=1.
      CUN=FN/CU
      QN=FN*Qo
      CUS=CU*SQRT(So)
      NCT=0
10    FF=QN*(B+FMS*Yo)**.6666667-CUS*((B+FM*Yo)*Yo)**1.6666667
      NCT=NCT+1
```

```
      IF(MOD(NCT,2).EQ.0) GO TO 15
      FF1=FF
      YY=Yo
      Yo=1.005*Yo
      GO TO 10
15    DIF=(Yo-YY)*FF1/(FF-FF1)
      Yo=YY-DIF
      IF(ABS(DIF).GT.1.E-5 .AND. NCT.LT.20) GO TO 10
      Ao=(B+FM*Yo)*Yo
      Vo=Qo/Ao
      HCA1=(BH+FM3*Yo)*Yo*Yo
      WRITE(3,100) Qo,Yo,Vo
      WRITE(6,100) Qo,Yo,Vo
100   FORMAT(' Qo =',F8.1,', Normal Depth Yo =',F8.3,', Vo =',F8.3)
      WRITE(*,*)' Give: New Q2, Dt, NT & Guess for Yb'
      READ(*,*) Qb,DT,NT,Yb
      HC1=HCA1/Ao
      NCT=0
17    A=(B+FM*Yb)*Yb
      HCA2=(BH+FM3*Yb)*Yb*Yb
      FF=G*(HCA2/Ao-HC1)/(1.-Ao/A)-((Qb-Qo)/(A-Ao)-Vo)**2
      NCT=NCT+1
      IF(MOD(NCT,2).EQ.0) GO TO 18
      FF1=FF
      Yb2=Yb
      Yb=1.005*Yb
      GO TO 17
18    DIF=(Yb-Yb2)*FF1/(FF-FF1)
      Yb=Yb2-DIF
      IF(ABS(DIF).GT.1.E-5 .AND. NCT.LT.30) GO TO 17
C v=X(1); Y2=X(2); Q2=X(3)
      A=(B+FM*Yb)*Yb
      X(1)=(Qb-Qo)/(A-Ao)
      WRITE(6,105) Qb,Yb,A,SQRT(ABS(G*A/(B+FM2*Yb))),X(1)
      WRITE(3,105) Qb,Yb,A,SQRT(G*A/(B+FM2*Yb)),X(1)
      105 FORMAT(' For New Q=',F8.2,/' Starting Depth Yb=',F8.3,
     &' A=',F8.2,' Celerity =',F8.3,' Wave Velocity=',F8.3)
      X(2)=Yb
      X(3)=.5*(Qo+Qb)
      WRITE(3,108)
108   FORMAT(/' Time Pos. x Wave V. Y2 Q2 Fr1',= Fr2',/53('-'))
      WRITE(3,110) 0,0.,X(1),Yb,Qb,(X(1)-Vo)/SQRT(G*Ao/(B+FM2*Yo)),
     &(X(1)-Qb/A)/SQRT(G*A/(B+FM2*Yb))
      DXS=.5
      DIST=0.
      DO 50 K=1,NT
      DIST=DIST+DT*X(1)
      NCT=0
20    SUM=0.
      CALL FUN(F)
      DO 22 I=1,3
```

```
      XX=X(I)
      X(I)=1.005*X(I)
      CALL FUN(F1)
      DO 21 J=1,3
21    D(J,I)=(F1(J)-F(J))/(X(I)-XX)
22    X(I)=XX
      CALL SOLVEQ(3,1,3,D,F,1,DD,INDX)
      DO 24 I=1,3
      X(I)=X(I)-F(I)
24    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,SUM,X
      IF(SUM.GT. 1.E-4 .AND. NCT.LT.20) GO TO 20
      A=(B+FM*X(2))*X(2)
      FR1=(X(1)-Vo)/SQRT(G*Ao/(B+FM2*Yo))
      FR2=(X(1)-X(3)/A)/SQRT(G*A/(B+FM2*X(2)))
      WRITE(3,110) IFIX(DT*FLOAT(K)),DIST,X,FR1,FR2
      IF(FR1.LT.1.07 .AND. K.LT.NT) THEN
      WRITE(*,*)' Froude No. close to 1. Give 1 to try=,= another
     &step, 0=STOP'
      READ(*,*) NCT
      IF(NCT.NE.1) STOP
      ENDIF
50    CONTINUE
110   FORMAT(I5,F8.1,2F8.3,F8.2,2F8.3)
      END
      SUBROUTINE FUN(F)
      REAL F(3),Y(1),YTT(1)
      COMMON B,FM,FM2,FMS,FM3,BH,Yb,Vo,Ao,HCA1,G,G2,Qb,DXS,
     &DIST,X(3),CUN,SO
      A2=(B+FM*X(2))*X(2)
      F(1)=(X(1)-Vo)*Ao-X(1)*A2+X(3)
      F(2)=HCA1+(X(1)-Vo)**2*Ao/G-(BH+FM3*X(2))*X(2)**2-(X(1)-
     &X(3)/A2)**2*A2/G
      Ab=(B+FM*X(3))*X(3)
      Y(1)=Yb
      CALL RUKUST(1,DXS,0.,DIST,1.E-4,Y,YTT)
      F(3)=X(2)-Y(1)
      RETURN
      END
      SUBROUTINE SLOPE(XX,Y,DY)
      REAL Y(1),DY(1)
      COMMON B,FM,FM2,FMS,FM3,BH,Yb,Vo,Ao,HCA1,G,G2,Qb,
     &DXS,DIST,X(3),CUN,SO
      qs=(Qb-X(3))/DIST
      Q=Qb+XX/DIST*(X(3)-Qb)
      A=(B+FM*Y(1))*Y(1)
      SF=(Q*CUN*((B+FMS*Y(1))/A)**.6666667/A)**2
      DY(1)=(SO-SF+Q*qs/(G*A**2))/(1.-Q**2*(B+FM2*Y(1))/(G*A**3))
      RETURN
      END
```

```
Program WAVEUCE.C
// Solves UCW with upstream gate instantly raised with uniform
// flow downstream; uses explicit method.
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
float b,fm,fm2,fms,fm3,bh,yb,vo,ao,hca1,g,g2,qb,*dxs,dist,cun,\
   so,x[3];
FILE *fil1; char fnam[20];
extern void rukust(int neq,float *dxs,float xbeg,\
   float xend,float error,float *y,float *ytt);
extern void solveq(int n,float **a,float *b,int itype,\
   float *dd, int *indx);
void fun(float *f){ float y[1],ytt[1],a2;
   a2=(b+fm*x[1])*x[1]; f[0]=(x[0]-vo)*ao-x[0]*a2+x[2];
   f[1]=hca1+pow(x[0]-vo,2.)*ao/g-(bh+fm3*x[1])*x[1]*x[1]-
      pow(x[0]-x[2]/a2,2.)*a2/g;
   y[0]=yb; rukust(1,dxs,0,dist,1.e-4,y,ytt);
   f[2]=x[1]-y[0]; } // End fun
void slope(float xx,float *y,float *dy){float qs,q,a,sf;
   qs=(qb-x[2])/dist; q=qb+xx/dist*(x[2]-qb); a=(b+fm*y[0])*y[0];
   sf=pow(q*cun*pow((b+fms*y[0])/a,.6666667)/a,2.);
   dy[0]=(so-sf+q*qs/(g*a*a))/(1.-q*q*(b+fm2*y[0])/\
      (g*pow(a,3.)));} // End slope
void main(void){ int nct,i,j,k,nt,indx[3];
   float cu,cus,qn,fn,qo,yy,ff,ff1,yo,a,hca2,dif,hc1,dt,yb2,\
   sum,xx,fr1,fr2,f[3],f1[3],**d,*dd;
   char *fmt="%5d %7.1f %7.3f %7.3f %7.2f %7.3f %7.3f\n";
   d=(float**)malloc(3*sizeof(float*));
   for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
   printf("Give: g,b,m,n,So,Qo & Guess for Yo\n");
   scanf("%f %f %f %f %f %f %f",&g,&b,&fm,&fn,&so,&qo,&yo);
   bh=.5*b;fm3=fm/3.;fm2=2.*fm;fms=2.*sqrt(fm*fm+1.);
   g2=2.*g;cu=1.486;if(g<20.)cu=1.;
   cun=fn/cu;qn=fn*qo;cus=cu*sqrt(so); nct=0;
L10:ff=qn*pow(b+fms*yo,.6666667)-cus*pow((b+fm*yo)*yo,1.666667);
   if(++nct%2){ff1=ff; yy=yo; yo*=1.005; goto L10;}
   dif=(yo-yy)*ff1/(ff-ff1); yo=yy-dif;
   if((fabs(dif)>1.e-5) && (nct<20)) goto L10;
   ao=(b+fm*yo)*yo;vo=qo/ao;hca1=(bh+fm3*yo)*yo*yo;
   printf("Give output filenam1\n");scanf("%s",fnam);
   if((fil1=fopen(fnam,"w"))==NULL){
     printf("Cannot open output file\n"); exit(0);}
   printf("Qo=%8.1f, Normal Depth Yo=%8.3f, Vo=%8.3f\n",\
      qo,yo,vo);
   fprintf(fil1,"Qo=%8.1f, Normal Depth Yo=%8.3f, Vo=%8.3f\n",\
      qo,yo,vo);
   printf("Give: New Q2, Dt, Nt & Guess for Yb\n");
   scanf("%f %f %d %f",&qb,&dt,&nt,&yb);
   hc1=hca1/ao; nct=0;
L17:a=(b+fm*yb)*yb; hca2=(bh+fm3*yb)*yb*yb;
   ff=g*(hca2/ao-hc1)/(1.-ao/a)-pow((qb-qo)/(a-ao)-vo,2.);
```

```
   if(++nct%2){ff1=ff;yb2=yb;yb*=1.005;goto L17;}
   dif=(yb-yb2)*ff1/(ff-ff1); yb=yb2-dif;
   if((fabs(dif)>1.e-5)&&(nct<30)) goto L17;
// v=x[0]; Y2=x[1]; Q2=x[2]
   a=(b+fm*yb)*yb; x[0]=(qb-qo)/(a-ao);
   printf("For New Q %8.2fn Starting Depth Yb=%8.3f A=%8.2f\
      Celerity=%8.3f Wave Velocity=%8.3f\n",\
      qb,yb,a,sqrt(fabs(g*a/(b+fm2*yb))),x[0]);
   fprintf(fil1,"For New Q %8.2fn Starting Depth Yb=%8.3f \
      A=%8.2f Celerity=%8.3f Wave Velocity=%8.3f\n", \
      qb,yb,a,sqrt(fabs(g*a/(b+fm2*yb))),x[0]);
   x[1]=yb; x[2]=.5*(qo+qb);
   fprintf(fil1,"nTime Pos. x Wave V. Y2 Q2 Fr1 Fr2\n");
   for(i=0;i<53;i++) fprintf(fil1,"-"); fprintf(fil1,"\n");
   fprintf(fil1,fmt,0,0.,x[0],yb,qb,\
      (x[0]-vo)/(g*ao/(b+fm2*yo)),(x[0]-qb/a)/(g*a*(b+fm2*yb)));
  *dxs=.5; dist=0.;
   for(k=1;k<=nt;k++){ dist+=dt*x[0]; nct=0;
L20: sum=0.; fun(f);
      for(i=0;i<3;i++){xx=x[i]; x[i]*=1.005; fun(f1);
         for(j=0;j<3;j++)d[j][i]=(f1[j]-f[j])/(x[i]-xx);
         x[i]=xx;}
      solveq(3,d,f,1,dd,indx);
      for(i=0;i<3;i++){x[i]-=f[i];sum+=fabs(f[i]);}
      printf("NCT=%d,%f\n",++nct,sum);
      if((sum>5.e-5)&&(nct<20)) goto L20;
      fr1=(x[0]-vo)/sqrt(g*ao/(b+fm2*yo));
      a=(b+fm*x[1])*x[1];
      fr2=(x[0]-x[2]/a)/sqrt(g*a/(b+fm2*x[1]));
      fprintf(fil1,fmt,(int)(dt*(float) k),\
         dist,x[0],x[1],x[2],fr1,fr2);
      if((fr1<1.1)&&(k<nt)){
         printf("Froude No.=%f is close to 1. Give 1 to try \
            another step, 0=STOP\n",fr1);
         scanf("%d",&nct); if(nct==0) {fclose(fil1);exit(0);}} }
   fclose(fil1);}
```

Program WAVEUCI.FOR

```
C Solves UCW with upstream gate instantly raised with uniform flow
downstream
C Uses the implicit method
      INTEGER*2 INDX(4)
      REAL F(4),F1(4),D(4,4)
      COMMON B,FM,FM2,FMS,FM3,BH,Yb,Vo,Ao,HCA1,G,G2,Qb,DXS,XPOS,X(4),
     &CUN,SO,DT5,WV1
      WRITE(*,*)' Give: g,b,m,n,So,Qo & Guess for Yo'
      READ(*,*) G,B,FM,FN,SO,Qo,Yo
      BH=.5*B
      FM3=FM/3.
      FM2=2.*FM
      FMS=2.*SQRT(FM*FM+1.)
```

```
      G2=2.*G
      CU=1.486
      IF(G.LT.20.) CU=1.
      CUN=FN/CU
      QN=FN*Qo
      CUS=CU*SQRT(So)
      NCT=0
10    FF=QN*(B+FMS*Yo)**.6666667-CUS*((B+FM*Yo)*Yo)**1.6666667
      NCT=NCT+1
      IF(MOD(NCT,2).EQ.0) GO TO 15
      FF1=FF
      YY=Yo
      Yo=1.005*Yo
      GO TO 10
15    DIF=(Yo-YY)*FF1/(FF-FF1)
      Yo=YY-DIF
      IF(ABS(DIF).GT.1.E-5 .AND. NCT.LT.20) GO TO 10
      Ao=(B+FM*Yo)*Yo
      Vo=Qo/Ao
      HCA1=(BH+FM3*Yo)*Yo*Yo
      WRITE(3,100) Qo,Yo,Vo
      WRITE(6,100) Qo,Yo,Vo
100   FORMAT(' Qo =',F8.1,', Normal Depth Yo =',F8.3,', Vo =',F8.3)
      WRITE(*,*)' Give: New Q2, Dt, NT & Guess for Yb'
      READ(*,*) Qb,DT,NT,Yb
      DT5=.5*DT
      HC1=HCA1/Ao
      NCT=0
17    A=(B+FM*Yb)*Yb
      HCA2=(BH+FM3*Yb)*Yb*Yb
      FF=G*(HCA2/Ao-HC1)/(1.-Ao/A)-((Qb-Qo)/(A-Ao)-Vo)**2
      NCT=NCT+1
      IF(MOD(NCT,2).EQ.0) GO TO 18
      FF1=FF
      Yb2=Yb
      Yb=1.005*Yb
      GO TO 17
18    DIF=(Yb-Yb2)*FF1/(FF-FF1)
      Yb=Yb2-DIF
      IF(ABS(DIF).GT.1.E-5 .AND. NCT.LT.30) GO TO 17
C   v=X(1); Y2=X(2); Q2=X(3); Position of wave x=X(4)
      A=(B+FM*Yb)*Yb
      X(1)=(Qb-Qo)/(A-Ao)
      WRITE(6,105) Qb,Yb,A,SQRT(ABS(G*A/(B+FM2*Yb))),X(1)
      WRITE(3,105) Qb,Yb,A,SQRT(G*A/(B+FM2*Yb)),X(1)
105   FORMAT(' For New Q=',F8.2,/' Starting Depth Yb=',F8.3,' A=',
     &F8.2, Celerity=',F8.3,' Wave Velocity=',F8.3)
      X(2)=Yb
      X(3)=.5*(Qo+Qb)
      X(4)=DT*X(1)
      WRITE(3,108)
```

```
108    FORMAT(/' Time Wave V. Y2 Q2 x Fr1 Fr2'/,1X,53('-'))
       WRITE(3,110) 0,X(1),Yb,Qb,0.0,(X(1)-Vo)/SQRT(G*Ao/
      &(B+FM2*Yo)),(X(1)-Qb/A)/SQRT(G*A/(B+FM2*Yb))
       DXS=.5
       XPOS=0.
       WV1=X(1)
       DO 50 K=1,NT
       NCT=0
20     SUM=0.
       CALL FUN(F)
       DO 22 I=1,4
       XX=X(I)
       X(I)=1.005*X(I)
       CALL FUN(F1)
       DO 21 J=1,4
21     D(J,I)=(F1(J)-F(J))/(X(I)-XX)
22     X(I)=XX
       CALL SOLVEQ(4,1,4,D,F,1,DD,INDX)
       DO 24 I=1,4
       X(I)=X(I)-F(I)
24     SUM=SUM+ABS(F(I))
       NCT=NCT+1
       WRITE(*,*)' NCT=',NCT,SUM,X
       IF(SUM.GT. 5.E-4 .AND. NCT.LT.20) GO TO 20
       A=(B+FM*X(2))*X(2)
       FR1=(X(1)-Vo)/SQRT(G*Ao/(B+FM2*Yo))
       FR2=(X(1)-X(3)/A)/SQRT(G*A/(B+FM2*X(2)))
       WRITE(3,110) IFIX(DT*FLOAT(K)),X,FR1,FR2
       WV1=X(1)
       XPOS=X(4)
       X(4)=X(4)+DT*WV1
110    FORMAT(I5,2F8.3,F8.2,F8.1,2F8.3)
       IF(FR1.LT.1.07 .AND. K.LT.NT) THEN
       WRITE(*,*)' Froude No. close to 1. Give 1 to try another step,
      &0=STOP'
       READ(*,*) NCT
       IF(NCT.NE.1) STOP
       ENDIF
50     CONTINUE
       END
       SUBROUTINE FUN(F)
       REAL F(4),Y(1),YTT(1)
       COMMON B,FM,FM2,FMS,FM3,BH,Yb,Vo,Ao,HCA1,G,G2,Qb,DXS,XPOS,X(4),
      &CUN,SO,DT5,WV1
       A2=(B+FM*X(2))*X(2)
       F(1)=(X(1)-Vo)*Ao-X(1)*A2+X(3)
       F(2)=HCA1+(X(1)-Vo)**2*Ao/G-(BH+FM3*X(2))*X(2)**2-(X(1)-X(3)/
      &A2)**2*A2/G
       Y(1)=Yb
       CALL RUKUST(1,DXS,0.,X(4),1.E-4,Y,YTT)
       F(3)=X(2)-Y(1)
       F(4)=XPOS+DT5*(WV1+X(1))-X(4)
```

```
  RETURN
  END
  SUBROUTINE SLOPE(XX,Y,DY)
  REAL Y(1),DY(1)
  COMMON B,FM,FM2,FMS,FM3,BH,Yb,Vo,Ao,HCA1,G,G2,Qb,DXS,XPOS,X(4),
 &CUN,SO,DT5,WV1
  qs=(Qb-X(3))/X(4)
  Q=Qb+XX/X(4)*(X(3)-Qb)
  A=(B+FM*Y(1))*Y(1)
  SF=(Q*CUN*((B+FMS*Y(1))/A)**.6666667/A)**2
  DY(1)=(SO-SF+Q*qs/(G*A**2))/(1.-Q**2*(B+FM2*Y(1))/(G*A**3))
  RETURN
  END
```

EXAMPLE PROBLEM 4.70

The flow rate Q_o = 250 cfs has been released for a long time into a trapezoidal channel with b = 10 ft, m = 1.5, n = 0.013, and S_o = 0.0005. Suddenly, the inflow at the beginning of the channel is doubled to Q_b = 500 cfs. Solve the movement of the wave in the channel for 115 s using 5 s time steps. Assume the flow rate Q varies linearly from Q_b to Q_2 immediately upstream from the wave. Use the explicit method.

Solution

The input to program WAVEUCE to solve this problem consists of

```
32.2 10 1.5 .013 .0005 250 3.4
500 5 25 4.3
```

The output from the program is

Q_o = 250.0, Normal depth Y_o = 3.563, V_o = 4.573
For new Q = 500.00
Starting depth Y_b = 4.302 A = 70.78, Celerity = 9.975, Wave velocity = 15.514

Time	Pos. X	Wave V.	Y_2	Q_2	F_{r1}	F_{r2}
0	0.0	15.517	4.302	500.00	1.186	0.847
5	77.6	15.464	4.281	491.63	1.181	0.851
10	154.9	15.415	4.260	483.55	1.175	0.855
15	232.0	15.367	4.240	475.73	1.170	0.858
20	308.8	15.320	4.220	468.14	1.165	0.862
25	385.4	15.274	4.200	460.76	1.160	0.865
30	461.8	15.229	4.181	453.56	1.155	0.869
35	537.9	15.185	4.162	446.54	1.150	0.872
40	613.8	15.142	4.144	439.66	1.146	0.876
45	689.6	15.099	4.126	432.91	1.141	0.879
50	765.0	15.056	4.107	426.28	1.136	0.882
55	840.3	15.014	4.089	419.76	1.132	0.886
60	915.4	14.972	4.072	413.36	1.127	0.889
65	990.3	14.931	4.054	407.03	1.123	0.893
70	1064.9	14.889	4.036	400.73	1.118	0.896
75	1139.4	14.848	4.019	394.58	1.114	0.899
80	1213.6	14.808	4.001	388.45	1.110	0.903
85	1287.6	14.767	3.983	382.38	1.105	0.906
90	1361.5	14.726	3.966	376.35	1.101	0.910
95	1435.1	14.685	3.948	370.37	1.096	0.914
100	1508.5	14.644	3.931	364.42	1.092	0.917
105	1581.7	14.603	3.913	358.50	1.087	0.921
110	1654.8	14.562	3.895	352.61	1.083	0.924
115	1727.6	14.521	3.878	346.74	1.078	0.928

EXAMPLE PROBLEM 4.71

Repeat the previous example problem but use the implicit method.

Solution

The input to program WAVEUCI is the same as in the previous example problem, namely,

```
32.2 10 1.5 .013 .0005 250 3.4
500 5 25 4.3
```

The output from Program WAVEUCI is

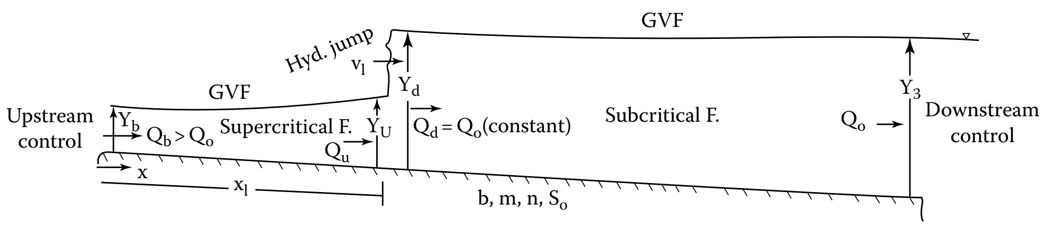
Q_o = 250.0, Normal depth Y_o = 3.563, V_o = 4.573
For new Q = 500.00
Starting depth Y_b = 4.302 A = 70.78 Celerity = 9.975 Wave velocity = 15.514

Time	Wave.V.	Y_2	Q_2	X	F_{r1}	F_{r2}
0	15.514	4.302	500.00	.0	1.186	0.847
5	15.464	4.281	491.64	77.4	1.181	0.851
10	15.415	4.260	483.57	154.6	1.175	0.855
15	15.367	4.240	475.76	231.6	1.170	0.858
20	15.320	4.220	468.19	308.3	1.165	0.862
25	15.275	4.201	460.82	384.8	1.160	0.865
30	15.230	4.181	453.63	461.1	1.155	0.869
35	15.186	4.163	446.61	537.1	1.151	0.872
40	15.142	4.144	439.74	612.9	1.146	0.875
45	15.099	4.126	433.00	688.5	1.141	0.879
50	15.057	4.108	426.38	763.9	1.137	0.882
55	15.015	4.090	419.87	839.1	1.132	0.886
60	14.973	0.072	413.45	914.1	1.127	0.889
65	14.931	4.054	407.12	988.8	1.123	0.893
70	14.890	4.036	400.86	1063.4	1.118	0.896
75	14.849	4.019	394.72	1137.7	1.114	0.899
80	14.808	4.001	388.59	1211.9	1.110	0.903
85	14.768	3.984	382.53	1285.8	1.105	0.906
90	14.727	3.966	376.51	1359.5	1.101	0.910
95	14.686	3.949	370.53	1433.1	1.096	0.913
100	14.645	3.931	364.59	1506.4	1.092	0.917
105	14.605	3.914	358.68	1579.5	1.088	0.921
110	14.563	3.896	352.65	1652.5	1.083	0.924
115	14.522	3.878	346.78	1725.2	1.079	0.928
120	14.480	3.860	340.93	1797.7	1.074	0.932
125	14.439	3.842	335.10	1870.0	1.070	0.936

4.25 MOVING HYDRAULIC JUMP

The moving waves we have examined so far have a subcritical flow both upstream and downstream therefrom when viewed by a stationary observer. The wave speeds are relatively large, and it is only because the moving observer moves with this velocity that he/she sees a supercritical flow on one side of the wave. If a condition changes that causes a hydraulic jump to move, then the same technique of writing steady-state equations from the viewpoint of an observer moving with this hydraulic jump can be used. Since supercritical conditions exist upstream from the hydraulic jump, one would expect the velocity of the jump movement to be smaller than the velocities of the previous moving waves we examined, because this velocity does not need to be large enough to cause a supercritical flow when viewed by moving with the wave. Also, the likely effects of inertia are more important for a moving hydraulic jump than the previous waves, and therefore if its effects are ignored, then the computed movements will deviate more from what can be observed in real situations. If the jump moves upstream, the upstream Froude number will increase, and if the jump moves downstream, the upstream Froude number will decrease as seen by the moving observer.

Consider a hydraulic jump that moves because the flow rate has been increased at the upstream end of the channel as shown in the sketch. Downstream movements give a positive v (or v_j) and upstream movements result in v being negative. In general, both upstream and downstream of the jump there will be gradually varied flows. If the channel is very long, then the flow downstream will be uniform, but this is a special case in which the depth downstream from the jump is constant. Depending upon the control at the downstream end of the channel, either an M_1- or an M_2-GVF may exit initially. Upstream of the jump, an M_3-GVF will exist. The change in total volume of water in the channel will equal the inflow minus the outflow integrated over time. In fact, accounting for the movement of the wave, such volume balances apply both upstream and downstream from the hydraulic jump. Because of larger depths downstream than upstream from the jump, the conservation of the mass principle will cause the jump to move upstream if depths along the upstream and downstream GVF profiles remain essentially constant for small increases in the flow rate Q_b, i.e., the increase in the flow rate accumulates in the plug of water above the original profile. However, the momentum principle suggests that the jump should move downstream for increasing upstream flow rates because M increases with Q if the depths remain constant. In brief, the movement of the hydraulic jump is difficult to predict intuitively, and is complex. The solution process is further complicated by the fact that the equations contain multiple roots.



The continuity and momentum equations are available from the viewpoint of an observer moving with the jump; two GVF equations are available, one upstream and one downstream of the jump; an equation that forces the change in total volume in the channel upstream from the jump to equal the difference between the inflow and outflow times time; a similar volume balance downstream from the jump; and if the implicit method is used, then a seventh equation comes from matching the jump's new position with that computed by adding the product of its average velocity times the time increment to its previous position. These seven equations are

$$F_1 = (V_u - v)A_1 - |V_d - v|A_d = 0 \tag{1j}$$

$$F_2 = (h_cA)_u + \frac{(V_u - v)^2 A_u}{g} - (h_cA)_d - \frac{(V_d - v)^2 A_d}{g} = 0 \tag{2j}$$

$$F_3 = Y_u - Y_{uode}(Y_b) = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f + Qq^*/(gA^2)}{1 - F_r^2} \tag{3j}$$

$$F_4 = (x_j)_{k-1} + \Delta t(v_{k-1} + v_k)/2 - (x_j)_k = 0 \tag{4j}$$

$$F_5 = Y_d - Y_{dode}(Y_3) = 0 \quad \frac{dY}{dx} = \frac{S_o - S_f + Qq^*/(gA^2)}{1 - F_r^2} \tag{5j}$$

$$F_6 = \int_0^{x_j} A\,dx - (\mathbb{V}_1)_{k-1} - \{2Q_b - (Q_u)_{k-1} - (Q_u)_k\}\Delta t/2 = 0 \tag{6j}$$

$$F_7 = \int_{x_j}^{L} A dx - (\mathbb{V}_2)_{k-1} - \{2Q_d + (Q_d)_{k-1} - (Q_3)_{k-1} - (Q_3)_k\}\Delta t/2 = 0 \tag{7j}$$

in which $(\mathbb{V}_1)_{k-1}$ and $(\mathbb{V}_2)_{k-1}$ are the volumes upstream and downstream from the jump, respectively, for the previous time step, starting with sub $k - 1 = 0$ being determined by integration of the original steady state GVF profiles upstream and downstream from the jump, or

$$(\mathbb{V}_1)_0 = \int_0^{x_j} A\,dx, \quad (\mathbb{V}_2)_0 = \int_{x_j}^{L} A\,dx \qquad \text{from the initial condition}$$

These seven equations allow seven variables to be solved for each new time step. The seven logical variables to select are (1) the depth immediately upstream from the jump Y_u, (2) the velocity of the jump v, (which will also be denoted by v_j), (3) the flow rate immediately upstream from the jump Q_u (or the velocity V_u) upstream from the jump, (4) the new position of the jump $(x_j)_k$, (5) the depth immediately downstream from the jump Y_d, (6) the flow rate immediately downstream from the jump Q_d, and (7) the flow rate Q_3 leaving the end of the channel. It will be assumed that the flow rate from the jump to the end of the channel does not change, i.e., $Q_3 = Q_o$.

To obtain the initial condition for a quasi-unsteady solution, the original steady-state problem needs to be solved that will determine the position of the hydraulic jump $(x_j)_0$, the depth upstream Y_u, and the depth downstream Y_d of the hydraulic jump. These three variable are solved from the following three equations as has been described previously in this chapter.

$$F_1 = (h_cA)_u + \frac{Q_p^2}{gA_u} - (h_cA)_d - \frac{Q_p^2}{gA_d} = 0$$

$$F_2 = Y_u - Y_{uode}(Y_b) = 0$$

$$F_3 = Y_d - Y_{dode}(Y_3) = 0$$

To have the hydraulic jump move, something must change either at the upstream end or at the downstream end of the channel, i.e., either the inflow or the outflow will be initiated at some intermediate position along the channel. Program WAVEMJP7.FOR is designed to solve a problem in which the flow rate is changed at the beginning of the channel. It uses the implicit method, and calls on ODESOLF as the ODE solver. It allows two upstream boundary conditions: (1) that the head H remains constant, and if the flow rate is increased the depth Y_b will increase according to the energy principle; and (2) that the depth Y_b remains constant as the flow rate Q_b is increased. As with UCW, without experimental data to describe how the flow rate varies from the beginning of the channel to the jump, one might use a linear interpolation for the Q upstream from the jump. Program WAVEMJP7 permits the user to select (1) a linear interpolation so that Q varies linearly from Q_b to Q_u (which is being solved from the Equations 1j through 4j), (2) a quadratic interpolation in which the change in flow rate is smallest at the beginning according to $Q = Q_b - Cx$, in which $C = (Q_b - Q_u)/x_j^2$, and (3) a quadratic interpolation in which the change in flow rate is largest at the beginning according to $Q = Q_u + Cz$, in which $z = x_j - x$ and $C = (Q_b - Q_u)/x_j^2$. The same options apply for the interpolation of the flow rate Q downstream from the jump, so Q varies between Q_d and Q_o.

Briefly, the structure of the program is as follows:

- After reading in the problem specifications, the steady-state initial condition is solved giving the initial location of the jump $(x_j)_0$ (in variable XPOS), the depths upstream Y_u, and the downstream Y_d from the jump, respectively. The array X stores these unknowns as $Y_u = X(1)$, $Y_d = X(2)$, and $(x_j)_0 = X(3)$. The subroutine FUN1 supplies the values of the three equations, as given above, to use the Newton method to solve these unknowns. Within the subroutine FUN1, a numerical integration using the trapezoidal rule determines the volumes of water upstream and downstream from the jump and stores these in VOL1 and VOL2. The sum of these, stored in VOLo, is the initial total volume of water in the channel. The subroutine FUN1 is also used to print out the steady state GVF profiles when IPRINT = 1, and to print these profiles FUN1 is called by the main program one additional time after the steady state solution has been obtained.
- With the steady-state solution complete, the user is prompted to give the new flow rate Q_b the time step DT to use for the quasi-unsteady solution, the number of time steps NT to complete, and finally the variable INIT which allows the selection of the method to use in interpolating Q upstream from the jump.
- Equations 4.1j through 4.7j are solved simultaneously for the number of time steps NT in the DO 50 K=1, NT. Now, the unknowns are stored in the array X(7) as follows: $Y_u = X(1)$, $v = X(2)$, $Q_u = X(3)$, $x = X(4)$, $Y_d = X(5)$, $Q_d = X(6)$, and $Q_3 = X(7)$. The subroutine FUN2 supplies the values of the seven equations, 4.1j through 4.7j, as the main program uses the Newton method to solve the equations. In supplying the equations, both the upstream and the downstream GVF profiles are numerically integrated within FUN2 so that the overall conservation of volume can be satisfied. Since volumes are much larger than the other variables being solved, the results of the volume calculations in Equations 4.6j and 4.7j (F(6) and F(7)) are divided by the variable FLB, which equals the product of the channel length times it bottom width, or FLB = FL*B.
- Upon completing each new time step solution, the results from the solution are written to the output file.

Program WAVEMJP7.FOR

```
      INTEGER*2 INDX(7)
      REAL F(7),F1(7),D(7,7)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/ B,FM,FM2,FM3,FMS,BH,CUN,Qo,Q2G,X(7),Yb,Y3,XPOS,
     &DT5,XPOO,FL,Qb,DXS,G,VJUMP,So,DX,TOL,H11,HMIN,Ab,A3,VOL1,
     &VOL2,VOLM1,VOLM2,QUM,QDM,Q3M,TIME,FLB,IUP,INIT,IPRINT
      WRITE(*,*)' Give: Qo,Ke,g,b,m,n,So,L,Yb,Y3 & Guess',' for
     &Yu,Yd,x'
      READ(*,*) Qo,FKE,G,B,FM,FN,So,FL,Yb,Y3,(X(I),I=1,3)
      WRITE(*,*)' Give Upstr. BC: 1=H(Const.), 2=Yb(Const.)'
      READ(*,*) IBC
      IPRINT=0
      FLB=FL*B
      H11=.5
      HMIN=1.E-5
      TOL=1.E-5
      BH=.5*B
      FKe=FKe+1.
      FM3=FM/3.
      FMS=2.*SQRT(FM*FM+1.)
      FM2=2.*FM
```

```
      Ab=(B+FM*Yb)*Yb
      A3=(B+FM*Y3)*Y3
      G2=2.*G
      CU=1.486
      IF(G.LT.20.) CU=1.
      CUN=FN/CU
      H=Yb+FKe*(Qo/((B+FM*Yb)*Yb))**2/G2
      Q2G=Qo*Qo/G
      INIT=0
C Solve steady-state variables Yu=X(1); Yd=X(2); x=X(3)
      NCT=0
10    SUM=0.
      CALL FUN1(F)
      DO 14 I=1,3
      XX=X(I)
      X(I)=1.005*X(I)
      CALL FUN1(F1)
      DO 12 J=1,3
12    D(J,I)=(F1(J)-F(J))/(X(I)-XX)
14    X(I)=XX
      CALL SOLVEQ(3,1,7,D,F,1,DD,INDX)
      DO 15 I=1,3
      X(I)=X(I)-F(I)
15    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,SUM,(X(I),I=1,3)
      IF(SUM.GT. 1.E-3 .AND. NCT.LT.20) GO TO 10
      XPOS=X(3)
      IPRINT=1
      CALL FUN1(F)
      VOLM1=VOL1
      VOLM2=VOL2
      QUM=Qo
      QDM=Qo
      Q3M=Qo
      WRITE(3,100) Qo,H,VOL1,VOL2,(X(I),I=1,3)
      WRITE(6,100) Qo,H,VOL1,VOL2,(X(I),I=1,3)
100   FORMAT(' Steady-State Jump: Qo=',F7.1,', H=',F7.3,', Vol1=',
     &E12.6,', VOL2=',E12.6,/,' Yu=',F8.3,', Yd=',F8.3,', x=',F8.2)
      WRITE(*,*)' Give: Qb,Dt,Nt & INIT',' for interp.,1=linear,
     &2=with -x**2,3=with -(xw-x)**2'
      READ(*,*) Qb,DT,NT,INIT
      IF(IBC.GT.1) GO TO 19
      NCT=0
      Yb=Yb*Qb/Qo
17    FF=H-Yb-(Qb/((B+FM*Yb)*Yb))**2/G2
      NCT=NCT+1
      IF(MOD(NCT,2).EQ.0) GO TO 18
      FF1=FF
      YY=Yb
      Yb=1.005*Yb
      GO TO 17
```

```
18    DIF=(Yb-YY)*FF1/(FF-FF1)
      Yb=YY-DIF
      IF(ABS(DIF).GT.1.E-5 .AND. NCT.LT.30) GO TO 17
      Ab=(B+FM*Yb)*Yb
19    IF(IBC.EQ.1) THEN
      WRITE(3,104) Qb,Yb,DT,INIT
104   FORMAT(' New Flow rate Qb=',F8.1,', New Depth Yb=',F8.3,',
     & Dt=',F8.1,', INIT=',I2,/)
      ELSE
      H=Yb+FKe*(Qb/Ab)**2/G2
      WRITE(3,105) Qb,Yb,H,DT,INIT
105   FORMAT(' New Flow rate Qb=',F8.1,', Yb=',F8.3,', H=',
     &F8.3,', Dt=',F8.1,', INIT=',I2,/)
      ENDIF
C Yu=X(1),v=X(2),Qu=X(3),x=X(4),Yd=X(5),Qd=X(6),Q3=X(7)
      DT5=.5*DT
      WRITE(3,108)
108   FORMAT(' Time Yu Jump Vel. Qu x Yd',' Qd Q3 Vd Vu VOL1
     &VOL2',/,1X,92('-'))
      VJUMP=0.
      X(5)=X(2)
      X(2)=1.
      X(3)=Qo
      X(4)=XPOS+DT
      X(6)=Qo+.1*(Qb-Qo)
      X(7)=Qo
      XPOO=XPOS
      DO 50 K=1,NT
      TIME=DT*FLOAT(K)
      NCT=0
20    SUM=0.
      CALL FUN2(F)
      DO 24 I=1,7
      XX=X(I)
      IF(ABS(X(I)).GT. 0.1) THEN
      X(I)=1.005*X(I)
      ELSE
      X(I)=X(I)+.002
      ENDIF
      CALL FUN2(F1)
      DO 22 J=1,7
22    D(J,I)=(F1(J)-F(J))/(X(I)-XX)
24    X(I)=XX
      CALL SOLVEQ(7,1,7,D,F,1,DD,INDX)
      DO 26 I=1,7
      X(I)=X(I)-F(I)
26    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,SUM
      IF(SUM.GT. .008 .AND. NCT.LT.20) GO TO 20
      CALL FUN2(F)
```

```
      VOLM1=VOL1
      VOLM2=VOL2
      WRITE(3,110)IFIX(TIME),X,X(6)/((B+FM*X(5))*X(5)),Qb/
     &((B+FM*X(1))*X(1)),INT4(VOL1),INT4(VOL2)
110   FORMAT(I5,2F8.3,2F8.2,F8.3,2F8.2,2F8.3,2I8)
      VJUMP=X(2)
      XPOS=X(4)
      IF(XPOS.GE.FL .OR. XPOS.LT. 0.) STOP
      QUM=X(3)
      QDM=X(6)
      Q3M=X(7)
50    X(4)=X(4)+DT*VJUMP
      END
      SUBROUTINE FUN1(F)
      LOGICAL NFS
      REAL F(7),Y(1),XP(1),YP(1,1),W(1,13),Yo(15)
      EXTERNAL DYX
      COMMON /TRAS/ B,FM,FM2,FM3,FMS,BH,CUN,Qo,Q2G,X(7),Yb,Y3,
     &XPOS,DT5,XPOO,FL,Qb,DXS,G,VJUMP,So,DX,TOL,H11,HMIN,Ab,A3,
     &VOL1,VOL2,VOLM1,VOLM2,QUM,QDM,Q3M,TIME,FLB,IUP,INIT,IPRINT
      F(1)=(BH+FM3*X(1))*X(1)**2+Q2G/((B+FM*X(1))*X(1))
     &-(BH+FM3*X(2))*X(2)**2-Q2G/((B+FM*X(2))*X(2))
      H11=2.
      NFS=.TRUE.
      VOL1=0.
      AR1=Ab
      XX1=0.
      DX=X(3)/14.
      DX2=.5*DX
      Y(1)=Yb
      IF(IPRINT.GT.0) Yo(1)=Yb
      DO 10 I=1,14
      XX=XX1+DX
      CALL ODESOLF(Y,DY,1,XX1,XX,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      IF(IPRINT.GT.0) Yo(I+1)=Y(1)
      NFS=.FALSE.
      AR2=(B+FM*Y(1))*Y(1)
      VOL1=VOL1+DX2*(AR1+AR2)
      AR1=AR2
10    XX1=XX
      IF(IPRINT.GT.0) THEN
      WRITE(3,101)(DX*FLOAT(I-1),Yo(I),I=1,15),VOL1
101   FORMAT(' GVF profile Upstream of',' Jump',/,
     &3(5(F8.1,F8.3),/),' Volume=',E12.6,/)
      ELSE
      F(2)=X(1)-Y(1)
      ENDIF
      H11=-.001
      NFS=.TRUE.
      VOL2=0.
      AR1=A3
      DX=(FL-X(3))/14.
```

```
      DX2=.5*DX
      XX1=FL
      Y(1)=Y3
      IF(IPRINT.GT.0) Yo(15)=Y3
      DO 20 I=1,14
      XX=XX1-DX
      CALL ODESOLF(Y,DY,1,XX1,XX,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      Yo(15-I)=Y(1)
      NFS=.FALSE.
      AR2=(B+FM*Y(1))*Y(1)
      VOL2=VOL2+DX2*(AR1+AR2)
      AR1=AR2
20    XX1=XX
      IF(IPRINT.GT.0) THEN
      WRITE(3,102)(X(3)+DX*FLOAT(I-1),Yo(I),I=1,15),VOL2
102   FORMAT(' GVF profile Downstream of=,= Jump',/,
     &3(5(F8.1,F8.3),/),' Volume=',E12.6,/)
      ELSE
      F(3)=X(2)-Y(1)
      ENDIF
      RETURN
      END
      SUBROUTINE FUN2(F)
      LOGICAL NFS
      REAL F(7),Y(1),XP(1),YP(1,1),W(1,13)
      EXTERNAL DYX
      COMMON /TRAS/ B,FM,FM2,FM3,FMS,BH,CUN,Qo,Q2G,X(7),Yb,
     &Y3,XPOS,DT5,XPOO,FL,Qb,DXS,G,VJUMP,So,DX,TOL,H11,HMIN,Ab,
     &A3,VOL1,VOL2,VOLM1,VOLM2,QUM,QDM,Q3M,TIME,FLB,IUP,INIT,IPRINT
C Yu=X(1),v=X(2),Qu=X(3),x=X(4),Yd=X(5),Qd=X(6),Q3=X(7)
      Au=(B+FM*X(1))*X(1)
      Ad=(B+FM*X(5))*X(5)
      F(1)=X(3)-X(6)+X(2)*(Ad-Au)
      F(2)=(BH+FM3*X(1))*X(1)**2+(X(3)/Au-X(2))**2*Au/G-
     &(BH+FM3*X(5))*X(5)**2-(X(6)/Ad-X(2))**2*Ad/G
      NFS=.TRUE.
      IUP=1
      VOL1=0.
      AR1=Ab
      XX1=0.
      DX=X(4)/15.
      DX2=.5*DX
      Y(1)=Yb
      DO 10 I=1,15
      XX=XX1+DX
      CALL ODESOLF(Y,DY,1,XX1,XX,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      NFS=.FALSE.
      AR2=(B+FM*Y(1))*Y(1)
      VOL1=VOL1+DX2*(AR1+AR2)
      AR1=AR2
10    XX1=XX
      F(3)=X(1)-Y(1)
```

```
      F(4)=XPOS+DT5*(VJUMP+X(2))-X(4)
      H11=-.001
      NFS=.TRUE.
      IUP=0
      VOL2=0.
      AR1=A3
      DX=(FL-X(4))/15.
      DX2=.5*DX
      XX1=FL
      Y(1)=Y3
      DO 20 I=1,15
      XX=XX1-DX
      CALL ODESOLF(Y,DY,1,XX1,XX,TOL,H11,HMIN,1,XP,YP,W,DYX,NFS)
      NFS=.FALSE.
      AR2=(B+FM*Y(1))*Y(1)
      VOL2=VOL2+DX2*(AR1+AR2)
      AR1=AR2
20    XX1=XX
      F(5)=X(5)-Y(1)
      F(6)=(VOLM1-VOL1+DT5*(2.*Qb-QUM-X(3)))/FLB
      F(7)=(VOLM2-VOL2+DT5*(QDM-Q3M+X(6)-X(7)))/FLB
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1)
      COMMON /TRAS/ B,FM,FM2,FM3,FMS,BH,CUN,Qo,Q2G,X(7),Yb,Y3,
     &XPOS,DT5,XPOO,FL,Qb,DXS,G,VJUMP,So,DX,TOL,
     &H11,HMIN,Ab,A3,VOL1,VOL2,VOLM1,
      &VOLM2,QUM,QDM,Q3M,TIME,FLB,IUP,INIT,IPRINT
      A=(B+FM*Y(1))*Y(1)
      IF(IABS(INIT).GT.0) GO TO 10
      SF=(Qo*CUN*((B+FMS*Y(1))/A)**.6666667/A)**2
      DY(1)=(So-SF)/(1.-Qo**2*(B+FM2*Y(1))/(G*A**3))
      RETURN
10    IF(IUP.EQ.1) THEN
      Q1=Qb
      Q2=X(3)
      XP=XX
      XJ=X(4)
      ELSE
      Q1=X(6)
      Q2=X(7)
      XP=XX-X(4)
      XJ=FL-X(4)
      ENDIF
      IF(IABS(INIT).EQ.1) THEN
      qs=(Q1-Q2)/XJ
      Q=Q1+XP/XJ*(Q2-Q1)
      ELSE IF(IABS(INIT).EQ.2) THEN
      C=(Q1-Q2)/XJ**2
      qs=2.*C*XP
      Q=Q1-C*XP**2
```

```
      ELSE
      C=(Q1-Q2)/XJ**2
      z=XJ-XP
      qs=2.*C*z
      Q=Q2+C*z**2
      ENDIF
      SF=(Q*CUN*((B+FMS*Y(1))/A)**.6666667/A)**2
      DY(1)=(So-Sf+Q*qs/(G*A**2))/(1.-Q**2*(B+FM2*Y(1))/(G*A**3))
      RETURN
      END
```

WAVEMJP7.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
extern void rukust(int neq,float *dxs,float xbeg,\
   float xend,float error,float *y,float *ytt);
extern void solveq(int n,float **a,float *b,int itype,\
   float *dd, int *indx);
int iup,init,iprint;
float b,fm,fm2,fm3,fms,bh,cun,qo,q2g,x[7],yb,y3,xpos,dt5,\
   xpoo,fl, qb,*dxs,g,vjump,so,dx,tol;
float ab,a3,vol1,vol2,volm1,volm2,qum,qdm,q3m,time,flb;
FILE *fil1; char fnam[20];
void fun1(float *f){
   int i,j; float xx1,ar1,ar2,xx,dx,dx2,y[1],ytt[1],yo[15];
   f[0]=(bh+fm3*x[0])*x[0]*x[0]+q2g/((b+fm*x[0])*x[0])-\
      (bh+fm3*x[1])*x[1]*x[1]-q2g/((b+fm*x[1])*x[1]);
  *dxs=2.;vol1=0.;ar1=ab;xx1=0.;dx=x[2]/14.;dx2=.5*dx;y[0]=yb;
   if(iprint)yo[0]=yb;
   for(i=1;i<15;i++){xx=xx1+dx;rukust(1,dxs,xx1,xx,tol,y,ytt);
      if(iprint) yo[i]=y[0];
      ar2=(b+fm*y[0])*y[0];vol1+=dx2*(ar1+ar2);ar1=ar2;xx1=xx;}
   if(iprint){fprintf(fil1,"GVF profile Upstream of Jump\n");
      for(j=0;j<3;j++){
         for(i=5*j;i<5*(j+1);i++)
            fprintf(fil1,"%8.1f %7.3f",dx*(float)i,yo[i]);
            fprintf(fil1,"\n");}}
   else f[1]=x[0]-y[0];
   *dxs=-.001; vol2=0.;ar1=a3;dx=(fl-x[2])/14.;dx2=.5*dx;
   xx1=fl;y[0]=y3;if(iprint)yo[14]=y3;
   for(i=1;i<15;i++){
      xx=xx1-dx;rukust(1,dxs,xx1,xx,tol,y,ytt);yo[14-i]=y[0];
      ar2=(b+fm*y[0])*y[0];vol2+=dx2*(ar1+ar2);ar1=ar2;xx1=xx;}
   if(iprint){fprintf(fil1,"GVF profile Downstream of Jump\n");
      for(j=0;j<3;j++){
        for(i=5*j;i<5*(j+1);i++)
           fprintf(fil1,"%8.1f %7.3f",x[2]+dx*(float)i,yo[i]);
           fprintf(fil1,"\n");}}
   else f[2]=x[1]-y[0];} // End fun1
void fun2(float *f){
   int i,j; float au,ad,xx1,ar1,ar2,xx,dx,dx2, y[1],ytt[1];
```

```
   au=(b+fm*x[0])*x[0]; ad=(b+fm*x[4])*x[4];
   f[0]=x[2]-x[5]+x[1]*(ad-au);
   f[1]=(bh+fm3*x[0])*x[0]*x[0]+pow(x[2]/au-x[1],2.)*au/g-\
      (bh+fm3*x[4])*x[4]*x[4]-pow(x[5]/ad-x[1],2.)*ad/g;
   iup=1;*dxs=2.;vol1=0.;ar1=ab;xx1=0.;dx=x[3]/15.;
   dx2=.5*dx; y[0]=yb;
   for(i=0;i<15;i++){xx=xx1+dx;rukust(1,dxs,xx1,xx,tol,y,ytt);
      ar2=(b+fm*y[0])*y[0];vol1+=dx2*(ar1+ar2);ar1=ar2;xx1=xx;}
   f[2]=x[0]-y[0];f[3]=xpos+dt5*(vjump+x[1])-x[3];
  *dxs=-.001;iup=0;vol2=0.;ar1=a3;dx=(fl-x[3])/15.;
   dx2=.5*dx;xx1=fl;y[0]=y3;
   for(i=0;i<15;i++){xx=xx1-dx;rukust(1,dxs,xx1,xx,tol,y,ytt);
      ar2=(b+fm*y[0])*y[0];vol2+=dx2*(ar1+ar2); ar1=ar2;xx1=xx;}
   f[4]=x[4]-y[0];f[5]=(volm1-vol1+dt5*(2.*qb-qum-x[2]))/flb;
   f[6]=(volm2-vol2+dt5*(qdm-q3m+x[5]-x[6]))/flb; } // End fun2
void slope(float xx,float *y,float *dy){
   float a,sf,q1,q2,xp,xj,qs,q,c,z;
   a=(b+fm*y[0])*y[0];if(init) goto L10;
   sf=pow(qo*cun*pow((b+fms*y[0])/a,.6666667)/a,2.);
   dy[0]=(so-sf)/(1.-qo*qo*(b+fm2*y[0])/(g*a*a*a)); return;
L10: if(iup){q1=qb;q2=x[2];xp=xx;xj=x[3];}
   else {q1=x[5];q2=x[6];xp=xx-x[3];xj=fl-x[3];}
   if(abs(init)==1){qs=(q1-q2)/xj;q=q1+xp/xj*(q2-q1);}
   else if(abs(init)==2) { c=(q1-q2)/xj/xj;qs=2.*c*xp;
      q=q1-c*xp*xp;}
   else {c=(q1-q2)/xj/xj;z=xj-xp;qs=2.*c*z;q=q2+c*z*z;}
   sf=pow(q*cun*pow((b+fms*y[0])/a,.6666667)/a,2.);
   dy[0]=(so-sf+q*qs/(g*a*a))/(1.-q*q*(b+fm2*y[0])/(g*a*a*a)); }
  //End dyx
void main(void){ int ibc,i,j,k,nct,nt,indx[7];
   float fke,fn,cu,h,sum,xx,yy,dif,ff1,ff,dt,g2,*dd,f[7],\
      f1[7],**d;
   printf("Give: Qo,Ke,g,b,m,n,So,L,Yb,Y3 & Guess for \
      Yu,Yd,x\n");
   scanf("%f %f %f %f %f %f %f %f %f %f",\
      &qo,&fke,&g,&b,&fm,&fn,&so,\&fl,&yb,&y3);
   for(i=0;i<3;i++)scanf("%f",&x[i]);
   printf("Give Upstr. BC: 1=H(Const.), 2=Yb(Const.)\n");
   scanf("%d",&ibc); iprint=0;flb=fl*b;*dxs=.5;tol=1.e-5;
   bh=.5*b;fke+=1.;fm3=fm/3.;fms=2.*sqrt(fm*fm+1.);
   fm2=2.*fm;ab=(b+fm*yb)*yb;a3=(b+fm*y3)*y3;g2=2.*g;
   cu=1.486;if(g<20.) cu=1; cun=fn/cu;
   h=yb+fke*pow(qo/ab,2.)/g2;q2g=qo*qo/g;init=0;
// Solve steady-state variables Yu=x[0]; yd=x[1]; x=x[2]
   d=(float**)malloc(3*sizeof(float*));
   for(i=0;i<3;i++)d[i]=(float*)malloc(3*sizeof(float));
   printf("Give output filenam1\n");scanf("%s",fnam);
   if((fil1=fopen(fnam,"w"))==NULL){
      printf("Cannot open output file\n");exit(0);}
   nct=0;
   do{ sum=0.; fun1(f);
```

```
   for(i=0;i<3;i++){xx=x[i];x[i]+=1.005; fun1(f1);
      for(j=0;j<3;j++) d[j][i]=(f1[j]-f[j])/(x[i]-xx);
      x[i]=xx;}
   solveq(3,d,f,1,dd,indx);
   for(i=0;i<3;i++){x[i]-=f[i]; sum+=fabs(f[i]);}
   printf(" NCT=%d SUM=%f\n",++nct,sum);
   }while((sum>.04)&&(nct<20));
   xpos=x[2]; iprint=1; fun1(f); volm1=vol1; volm2=vol2;
   qum=qo;qdm=qo;q3m=qo;
   printf("Steady-State Jump: Qo=%7.1f, H=%7.3f, Vol1=%12.6e,\
      Vol2=%12.6e\n Yu=%8.3f, Yd=%8.3f, x=%8.2f\n",\
      qo,h,vol1,vol2,x[0],x[1],x[2]);
   fprintf(fil1,"Steady-State Jump: Qo=%7.1f, H=%7.3f, Vol1=\
      %12.6e,\ Vol2=%12.6en Yu=%8.3f, Yd=%8.3f ,\
      x=%8.2f\n",qo,h,vol1,vol2,x[0],x[1],x[2]);
   printf("Give: Qb,Dt,Nt & INIT for interp.,1=linear,2=\
      with -x**2,3=with -(xw-x)**2\n");
   scanf("%f %f %d %d",&qb,&dt,&nt,&init); dt5=.5*dt;
   for(i=0;i<3;i++) free (d); free(*d);
   d=(float**)malloc(7*sizeof(float*));
   for(i=0;i<7;i++)d[i]=(float*)malloc(7*sizeof(float));
   if(ibc==1){nct=0;yb=yb*qb/qo;
L17: ff=h-yb-fke*pow(qb/((b+fm*yb)*yb),2.)/g2;
      if(++nct%2){ff1=ff;yy=yb;yb*=1.005;goto L17;}
         dif=(yb-yy)*ff1/(ff-ff1);
         yb=yy-dif;
         if((fabs(dif)>1.e-5)&&(nct<30)) goto L17;
         ab=(b+fm*yb)*yb;
          fprintf(fil1,"New Flow rate Qb=%8.1f, New Depth Yb=\
             %8.3f, Dt=%8.1f, INIT=%2d\n\n",qb,yb,dt,init);}
      else
         fprintf(fil1,"New Flow rate Qb=%8.1f, Yb=%8.3f, H=\
            %8.3f, Dt=%8.1f, INIT=%2dn\n",qb,yb,h,dt,init);
// Yu=x[0],v=x[1],Qu=x[2],x=x[3],Yd=x[4],Qd=x[5],Q3=x[6]
      fprintf(fil1,"n Time Yu Jump Vel.  Qu  x  Yd  Qd  \
         Q3  Vd Vu VOL1 VOL2\n");
      for(i=0;i<92;i++) fprintf(fil1,"-"); fprintf(fil1,"\n");
      vjump=0.;x[4]=x[1];x[1]=1.;x[2]=qo;x[3]=xpos+dt;
      x[5]=qo+.1*(qb-qo);x[6]=qo;xpoo=xpos;
      for(k=1;k<=nt;k++){ time=(float)k*dt; nct=0;
         do{sum=0.; fun2(f);
            for(i=0;i<7;i++){xx=x[i];
               if(fabs(x[i])>0.1) x[i]*=1.005; else x[i]+=.002;
               fun2(f1);
               for(j=0;j<7;j++) d[j][i]=(f1[j]-f[j])/(x[i]-xx);
               x[i]=xx;}
            solveq(7,d,f,1,dd,indx);
            for(i=0;i<7;i++){x[i]-=f[i];sum+=fabs(f[i]);}
         }while((sum>.008)&&(++nct<20));
      fun2(f);volm1=vol1;volm2=vol2;
      fprintf(fil1,"%5d %7.3f %7.3f %7.2f %7.2f %7.3f %7.2f\
         %7.2f %7.3f %7.3f %7d %7d\n",(int)time,x[0], \
```

```
        x[1],x[2],x[3],x[4],x[5],x[6],x[5]/((b+fm*x[4])*x[4]),\
        qb/((b+fm*x[0])*x[0]),(long)vol1,(long)vol2);
     vjump=x[1];xpos=x[3];
     if((xpos>=fl)||(xpos<0.)){fclose(fil1); exit(0);}
     qum=x[2];qdm=x[5];q3m=x[6];x[3]+=dt*vjump; }
   // End for(k
   fclose(fil1); }
```

EXAMPLE PROBLEM 4.72

Initially, a flow rate Q_o = 500 cfs enters a trapezoidal channel with b = 10 ft, m=1.5, n = 0.013, S_o = 0.0012, and a length L = 2000 ft. At the channel's beginning, the depth is Y_b = 2.0 ft. Suddenly, the flow rate entering the channel is increased to Q_b = 600 cfs, with the depth at the channel's beginning remaining at Y_b = 2 ft. The channel discharges into a reservoir at its downstream end whose water surface elevation is 6 ft above the channel bottom. Determine the initial position of the hydraulic jump, and its movement after the flow rate is increased. Obtain two solutions; the first based on the assumption that H at the upstream end of the channel remains constant, and the second for which Y_b is held constant.

Solution

The input to Program WAVEMJP7 using a linear interpolation for Q consists of

```
500 0 32.2 10 1.5 .013 .0012 2000 2 6 2.5 4.1 250
```
1/ (for the second solution, this is changed to 2 to select Y_b = constant.)
```
600 5 100 1
```

The output consists of (for H = constant):

GVF profile upstream of jump

```
 .0   2.000  21.1 2.053  42.2 2.105  63.3 2.158  84.4 2.211
105.5 2.265 126.6 2.318 147.7 2.372 168.8 2.426 189.9 2.481
211.0 2.537 232.1 2.593 253.2 2.650 274.3 2.708 295.4 2.767
```

Volume = 0.954002E + 04

GVF profile downstream of jump

```
 295.4 4.436  417.1 4.513  538.9 4.598  660.6 4.691  782.4 4.790
 904.2 4.894 1025.9 5.004 1147.7 5.118 1269.4 5.236 1391.2 5.357
1513.0 5.481 1634.7 5.608 1756.5 5.737 1878.2 5.868 2000.0 6.000
```

Volume = 0.156187E + 06

Steady state jump: `Qo= 500.0, H= 7.743, Vol1= .954002E+04, VOL2= .156187E+06`
`Yu= 2.767, Yd= 4.436, x= 295.37`

Time	Y_u	Jump Vel.	Q_u	X	Y_d	Q_d	Q_3	V_d	V_u	VOL1	VOL2
5	2.547	0.347	480.66	296.24	4.436	494.09	498.55	6.688	17.044	10088	156176
10	2.725	0.491	516.62	298.34	4.471	534.44	518.40	7.154	15.627	10595	156205
15	2.836	0.515	537.86	300.85	4.501	555.90	532.07	7.372	14.844	10958	156304
20	2.909	0.498	551.32	303.38	4.526	568.34	542.28	7.480	14.359	11236	156429
.	.	.	.	.	.	.	.	.	.	.	.
300	3.237	0.010	599.48	330.85	4.691	599.78	600.79	7.504	12.478	13155	157332
305	3.237	0.009	599.50	330.90	4.691	599.79	600.78	7.504	12.477	13158	157327
310	3.237	0.009	599.51	330.94	4.691	599.80	600.76	7.505	12.475	13160	157322
.	.	.	.	.	.	.	.	.	.	.	.
495	3.242	0.003	599.84	331.95	4.690	599.94	600.30	7.508	12.451	13213	157206
500	3.242	0.003	599.84	331.96	4.690	599.94	600.29	7.509	12.451	13214	157205

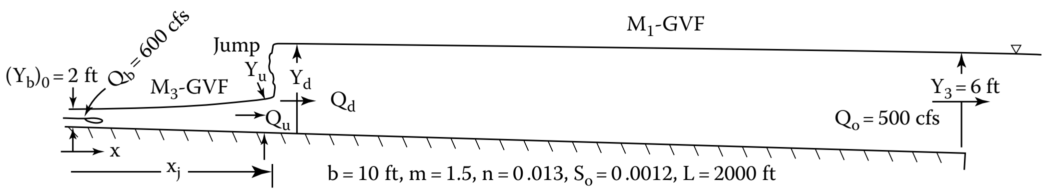
This solution shows that the hydraulic jump starts at a position 295 ft downstream from the beginning of the channel, and stabilizes again after moving down to a position at 332 ft. The steady state solution for Q = 600 cfs gives $Y_u = 3.20$ ft, $Y_d = 4.75$ ft, and $x_j = 472$ ft. The maximum velocity of the jump is 1/2 ft/s and this occurs 15 s after the flow rate is increased to 600 cfs. This solution assumes that the upstream head H is constant, and if the entrance loss coefficient $K_e = 0$ (as specified), then this head is 7.743 ft.

The following solution holds the upstream depth Y_b constant at 2 ft as the flow rate is increased to 600 cfs, and thus H is increased to 10.269 ft, resulting in a larger value for momentum at the beginning of the channel, and this causes a larger velocity of jump; now starting at 2.165 fps rather than 0.347 fps. Now at 500 s (the end of the solution), the jump has moved to a position of $x_j = 461$ ft, and is nearly stabilized here since its velocity is only $v_j = 0.055$ fps.

Since the solution indicates that the increased flow rate to 600 cfs soon becomes near constant throughout the entire length of channel, it is clear that ignoring the inertial effect makes this solution only an rough approximation of what will actually occur.

EXAMPLE PROBLEM 4.73

Solve the previous example problem assuming that the flow rate at the downstream end of the channel remains constant and equal to $Q_o = 500$ cfs. To obtain this solution, since there is one less unknown, i.e., six rather than seven, one less equation is needed. Combine the two equations that make the changes in volumes upstream and downstream of the jump equal to the net inflow volume into a single equation that keeps the total volume in the channel equal to the original total volume plus $(Q_b - Q_o)t$.



Solution

Equations 6j and 7j are replaced by the following single equation:

$$F_6 = \int_0^{x_j} A\,dx + \int_{x_j}^{L} A\,dx - \mathbb{V}_o - (Q_b - Q_u)t = 0 \tag{6j}$$

in which $\mathbb{V}_o$ is determined by integration of the original steady state GVF profiles upstream and downstream from the jump, or

$$\mathbb{V}_o = \int_o^{x_j} A\,dx + \int_{x_j}^{L} A\,dx \quad \text{from the initial condition}$$

Program WAVEMJP6.FOR (listing not given here, but is on the CD in the folder PROGRAM_HWK) is designed solve this problem. The input to Program WAVEMJP6 using a linear interpolation for Q consists of

`500 0 32.2 10 1.5 .013 .0012 2000 2 6 2.5 4.1 250`
`1/` (for the second solution, this is changed to 2 to select Y_b = constant.)
`600 5 30 1`

The output consist of (for H=constant):

Steady state jump: `Qo= 500.00, H= 7.743, Vol= .165728E+06`
`Yu= 2.767, Yd= 4.436, x= 295.37`

GVF profile upstream of jump

```
   .0 2.000  21.1 2.053  42.2 2.105  63.3 2.158  84.4 2.211
105.5 2.265 126.6 2.318 147.7 2.372 168.8 2.426 189.9 2.481
211.0 2.537 232.1 2.593 253.2 2.650 274.3 2.708 295.4 2.767
```

Volume= .954002E+04

GVF profile downstream of jump

```
 295.4 4.436  417.1 4.513  538.9 4.598  660.6 4.691  782.4 4.790
 904.2 4.894 1025.9 5.004 1147.7 5.118 1269.4 5.236 1391.2 5.357
1513.0 5.481 1634.7 5.608 1756.5 5.737 1878.2 5.868 2000.0 6.000
```

Volume= .156187E+06
New flow rate Qb= 600.0, New depth Yb= 2.379, Dt= 5.0, INIT = 1

Time	Y_u	Jump Vel.	Q_u	x	Y_d	Q_d	V_d	V_u
5	2.578	.397	487.00	296.36	4.436	502.13	6.797	16.785
10	2.453	.105	460.34	297.62	4.454	464.62	6.253	17.881
15	2.334	−.189	435.32	297.41	4.469	427.15	5.722	19.042
20	2.221	−.480	411.94	295.74	4.481	390.20	5.208	20.269
25	2.113	−.761	390.28	292.63	4.488	354.31	4.718	21.557
30	2.013	−1.027	370.38	288.16	4.492	320.07	4.257	22.898
.	.	.	.	.	.	.	.	.
70	1.464	−2.161	274.38	218.08	4.426	153.82	2.089	33.594
.	.	.	.	.	.	.	.	.
145	0.9145	−2.085	199.05	56.46	4.244	75.93	1.093	57.690
150	0.881	−2.076	194.31	46.06	4.231	71.44	1.033	60.148

(and for Yb = constant)

Steady state jump: Qo= 500.00, H= 7.743, Vol= .165728E+06
Yu= 2.767, Yd= 4.436, x= 295.37

GVF profile upstream of jump

```
 .0 2.  000  21.1 2.053  42.2 2.105  63.3 2.158  84.4 2.211
105.5 2.265 126.6 2.318 147.7 2.372 168.8 2.426 189.9 2.481
211.0 2.537 232.1 2.593 253.2 2.650 274.3 2.708 295.4 2.767
```

Volume= .954002E+04
GVF profile downstream of jump

```
 295.4 4.436  417.1 4.513  538.9 4.598  660.6 4.691  782.4 4.790
 904.2 4.894 1025.9 5.004 1147.7 5.118 1269.4 5.236 1391.2 5.357
1513.0 5.481 1634.7 5.608 1756.5 5.737 1878.2 5.868 2000.0 6.000
```

Volume= .156187E+06
New flow rate Qb= 600.0, Yb= 2.000, H= 10.269, Dt= 5.0, INIT= 1

Time	Y_u	Jum Vel.	Q_u	X	Y_d	Q_d	V_d	V_u
5	1.859	0.008	374.79	295.39	4.485	375.20	5.001	25.234
10	1.785	−0.354	356.62	294.53	4.494	337.99	4.493	26.511
15	1.713	−0.658	339.79	291.93	4.498	302.97	4.022	27.865

(continued)

Time	Y_u	Jum Vel.	Q_u	X	Y_d	Q_d	V_d	V_u
20	1.644	−0.979	324.42	287.77	4.498	270.73	3.594	29.277
25	1.579	−1.235	310.62	282.23	4.494	241.81	3.214	30.723
30	1.519	−1.450	298.41	275.52	4.488	216.60	2.885	32.180
.	.	.	.	.	.	.	.	.
70	1.177	−2.008	241.52	201.38	4.402	122.58	1.677	43.317
.	.	.	.	.	.	.	.	.
145	0.709	−1.926	173.27	53.64	4.230	55.21	0.798	76.549
150	0.679	−1.921	168.73	44.02	4.218	50.79	0.737	80.199

Notice the vast differences in the solutions. By not allowing a greater flow rate to exit from the end of the channel, the jump is forced upstream, and as this jump moves upstream, the flow rates immediately upstream and downstream from the jump decrease considerably below the original 500 cfs. What this solution shows is that the extra volume coming into the channel accumulates downstream from the jump, and this forces the jump to move upstream. The GVF equations in combination with the momentum and the continuity equations that the moving observer sees forces the depth to decrease as this occurs, and it does not take long until the jump moves upto the beginning of the channel. Contracting these solutions with the previous ones illustrates how assumptions affect the outcome. By not properly accounting for the volumes both upstream and downstream from the jump, this flow does not pass through the jump.

PROBLEMS

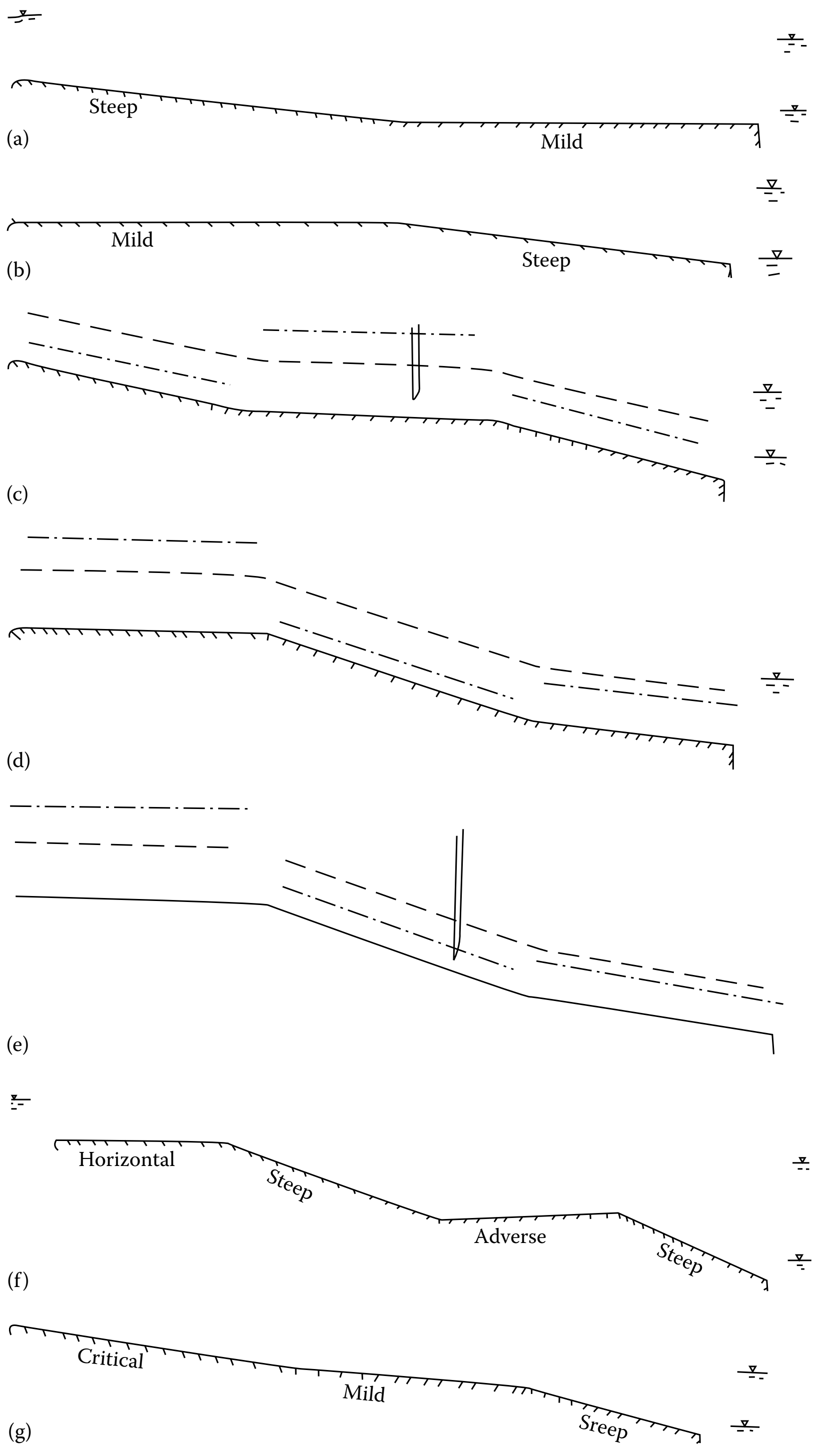
4.1 A 80 ft long linear transition occurs from a rectangular section with b = 8 ft to a trapezoidal section with b = 10 ft and m = 1.5. For a flow rate of Q = 400 cfs, determine the importance of the nonprismatic term in the ODE for GVF in comparison to the other terms in the numerator of this equation at the midpoint of the transition if the depth of flow here is 4.5 ft. The bottom slope of the transition at this point is $S_o = 0.0008$. Does this term tend to make the depth increase or decrease in the downstream direction? Rationalize this effect on the basis of the energy and the hydraulic grade lines. (n = 0.013)

4.2 A flow rate q* = 2 cfs/ft leaves from a side weir of a trapezoidal channel with b = 10 ft, n = 0.013, and m = 1.0. At the beginning of the side weir, the flow rate Q = 450 cfs, and the depth of flow at this position is 4.5 ft. How does the magnitude of the outflow term in the GVF-equation compare with the frictional loss term S_f.? Does this later outflow term tend to increase or decrease the depth in the downstream direction? Rationalize this effect on the basis of the energy and hydraulic grade lines. (n = 0.013)

4.3 A transition reduces a trapezoidal channel with b = 4 m and m = 2 to a rectangular section with b = 3.5 m over a length of 20 m. Both the change in the bottom width and the side slope are linear. If the depth of flow at the middle of the transition is 2 m when the flow rate is Q = 80 m^3/s, determine the magnitude of the term in the numerator of the ODE for GVF flows that describe the nonprismatic channel effects. How does this term's magnitude compare with S_f? Is the effect of the nonprismatic term to increase or to decrease the depth of flow in the downstream direction? (Assume $S_f = S_o$.)

4.4 A transition reduces a rectangular channel with b = 6 m to a rectangular channel with b = 4 m over a length of 18 m. There is a lateral outflow weir on one side of this transition. Determine how the lateral outflow q* must vary across the transition such that the slope S_f equals the bottom slope S_o across the transition. Notice this requirement should result in a near constant depth across the side weir.

4.5 Sketch in and properly label all possible GVF profiles in the prismatic channels depicted below. In some of these situations, two reservoir depths are shown. For these situations, determine and label GVF profiles for all reservoir depths between these two limits.



4.6 A trapezoidal channel has its bottom slope changed from $S_{o1} = 0.035$ to $S_{o2} = 0.00036$. The bottom width of the channel is b = 3 m and its side slope is 1.5. For a flow rate of 25 m^3/s, determine the location of the hydraulic jump. Manning's n for this channel equals 0.014. Also solve the problem using Chezy's equation if e = 0.0012 m and $\nu = 1.003 \times 10^{-6}$ m^2/s.

4.7 A trapezoidal channel with b = 6 ft and m = 2 has its bottom slope changed abruptly from $S_{o1} = 0.0135$ to $S_{o2} = 0.00014$. Its roughness coefficient is n = 0.014. If the flow rate is Q = 300 cfs, determine the location of the hydraulic jump.

4.8 A circular channel with a diameter D = 4 m and a Manning's n = 0.013 changes from a bottom slope of $S_{o1} = 0.018$ to $S_{o2} = 0.00051$. For a flow rate of 20 m^3/s, locate the position of the hydraulic jump.

4.9 Water enters a circular channel with a diameter D = 5 m from a reservoir whose water surface is 3 m above the bottom of the channel. The channel has a bottom slope $S_o = 0.025$ and a roughness coefficient n = 0.012. The entrance loss coefficient is $K_e = 0.15$. Determine the water surface profile into this channel including the depth till it is within 1% of the normal depth.

4.10 Note that the magnitude of the denominator in the GVF ODE has an influence on the length of the GVF profiles, as well as on the numerator. When Froude numbers are close to unity, then the profiles are short because the dY/dx is large. When the Froude numbers are large, then the dY/dx are smaller because $S_o - S_f$ is divided by a negative value larger than the unity in magnitude. Determine how the portion of the length of S_2-GVF profiles from 1.2 to 1.01 of the normal depth varies with the bottom slope S_o from 0.04 to .1 for a pipe with a diameter D = 5 ft and n = 0.012 if the flow rate is Q = 550 cfs. How does the total length of these profiles vary with these bottom slopes?

4.11 In the previous problem, it was discovered that S_2-GVF profile lengths tend to decrease in length with increasing bottom slopes. Investigate how the lengths of S_3-GVFs vary with the upstream depth in a flat very wide rectangular channel with n = 0.013 if the flow rate is q = 20 cfs/ft. Let the upstream depths vary from $Y_1 = 0.5$ to $Y_1 = 2.0$.

4.12 Water discharges from a gate in a rectangular channel at a depth of 2 ft. If the flow rate per unit width is q = 40 cfs/ft, and the channel is 10 ft wide, has a bottom slope of $S_o = 0.0008$, and n = 0.014, determine the GVF profiles to completely describe the water surfaces. Assume the channel if very long.

4.13 The channel in the previous problem terminates in a free overfall 1000 ft downstream from the gate. Define the GVF profiles completely. The water level at the end of the channel is well below the critical depth in this channel. Also solve the problem using Chezy's equation if e = 0.004 ft and $\nu = 1.217 \times 10^{-5}$ ft^2/s.

4.14 In Problems 4.12 and 4.13, there is a reservoir 1000 ft downstream with a water surface at a depth of 6.5 ft. Completely define the water surface profile throughout the channel.

4.15 A gate in a trapezoidal channel 500 ft downstream from a break in grade from $S_{o1} = 0.0005$ to $S_{o2} = 0.05$ caused the depth just upstream from it to be 10 ft for a flow rate of 450 cfs. The channel is trapezoidal in cross section with b = 10 ft and m = 1.5. The roughness coefficient is n = 0.013. Determine the GVF profiles and the location of the hydraulic jump. What is the minimum depth upstream from the gate that might occur and still have the gate touch the water under steady flow conditions? If $C_c = 0.6$, what is the height of the gate above the channel bottom for this minimum depth condition? Note that this condition can only occur if the gate is raised from a lower position. Assume the gate is being closed and is lower just below where it touches the flowing water surface. After steady flow conditions occur, what is the depth upstream of the gate?

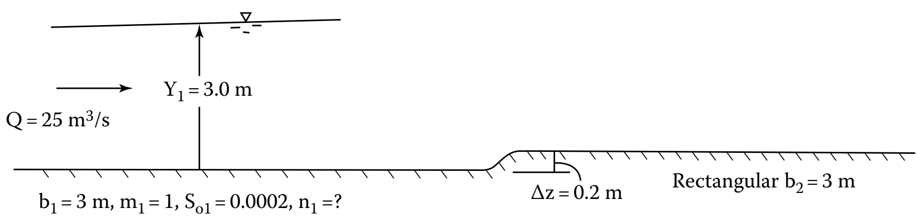
4.16 A gate in a trapezoidal channel 150 m downstream from a break in grade from $S_{o1} = 0.00045$ to $S_{o2} = 0.045$ caused the depth just upstream from it to be 4.0 m for a flow rate of 15 m^3/s. The channel is trapezoidal in cross section with b = 3 m and m = 1.5. The roughness coefficient is n = 0.013. Determine the GVF profiles and the location of the hydraulic jump. What is the minimum depth upstream from the gate that might occur and still have the gate touch the water under steady flow conditions? If $C_c = 0.6$, what is the height of the gate above the channel bottom for this minimum depth condition? Note that this condition can only occur if the gate is raised from a lower position. Assume the gate is being closed and is lower just below where it touches the flowing water surface. After steady flow conditions occur, what is the depth upstream of the gate?

4.17 Solve Problem 4.15 (or 4.16) but use Chezy's equation with e = 0.001 ft.

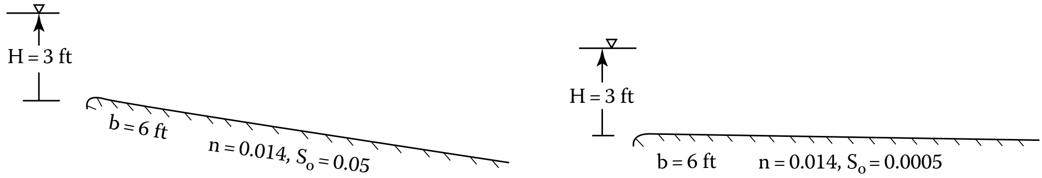
4.18 The channel in Problem 4.15 is feed by water from a reservoir that is 1000 ft upstream from the break in grade. If the depth of water in the reservoir is 5 ft above the bottom of the channel, determine the flow rate that will be flowing down the channel if the entrance loss coefficient is $K_e = 0.12$.

4.19 A gate in a main channel is used to control the flow rate into a side channel. The main channel is trapezoidal with b = 5 m and m = 2, and the side channel is 2 m wide and rectangular in shape. The bottom of the side channel rises 1 m above the bottom of the main channel, and it has a bottom slope $S_{o2} = 0.001$ and $n_2 = 0.018$. If 4 m^3/s is diverted in the side channel, and the main channel contains a flow rate of 30 m^3/s, and has a bottom slope $S_{o1} = 0.00095$ and $n_1 = 0.020$, determine the GVF profiles and how far they extend in all directions. This includes what happens downstream, as well as upstream from the gate.

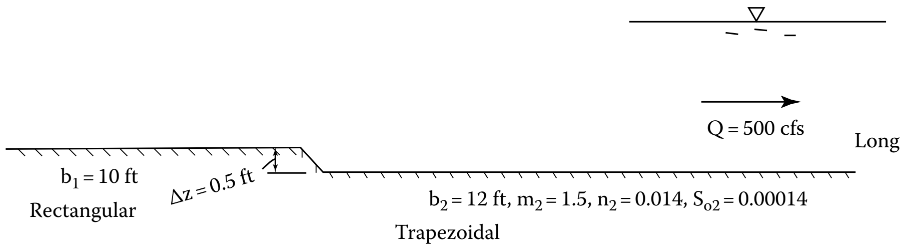
4.20 A transition occurs between an upstream trapezoidal channel with $b_1 = 3$ m and $m_1 = 1$ to a rectangular channel with $b_2 = 3$ m. The bottom rises by $\Delta z = 0.2$ m through the transition. If the flow rate is Q = 25 m^3/s and the depth in the upstream channel is $Y_1 = 3$ m (if not effected by backwater effects), determine the depth in the rectangular channel. What is the critical depth in the rectangular channel? If the slope of the upstream trapezoidal channel is $S_{o1} = 0.0002$, what is its Manning's roughness coefficient n if the depth of 3 m is its normal (or uniform) depth?



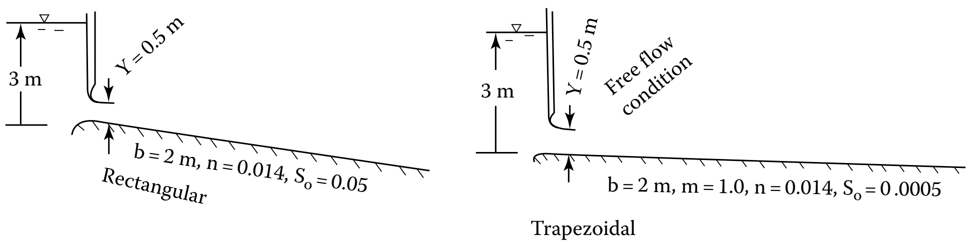
4.21 A rectangular channel is supplied by a reservoir whose water surface is H = 3 ft above the bottom of the channel, and the channel has a bottom width b = 6 ft, a Mannings roughness n = 0.014, and a bottom slope of $S_o = 0.05$. What is the flow rate into the channel? Another identical channel is supplied by a similar reservoir with H = 3 ft, but its bottom slope is $S_o = 0.0005$. What is the flow rate into this channel? Both channels are very long.



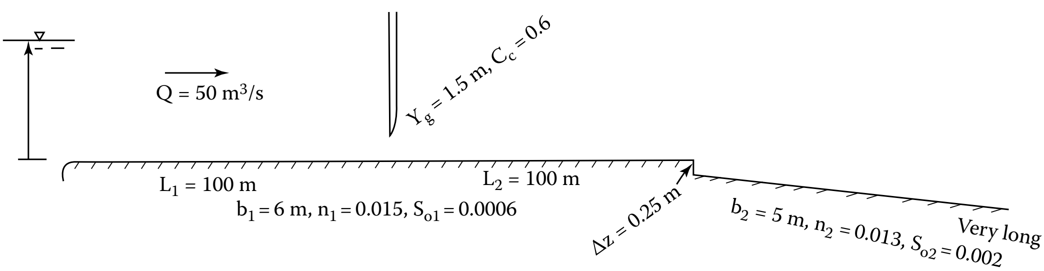
4.22 A rectangular channel with a bottom width $b_1 = 10$ ft smoothly changes to a trapezoidal channel with a bottom width of $b_2 = 12$ ft, $m_2 = 1.5$, $n_2 = 0.014$, and a bottom slope of $S_{o2} = 0.00014$. The channel's bottom drops 0.5 ft through the transition. The downstream channel is very long. If the flow rate is Q = 500 cfs, what is the depth at the end of the rectangular channel? What is the critical depth in the rectangular channel? If the drop in the channel bottom were 0.8 ft, what is the depth at the end of the rectangular channel? If the drop in the channel bottom were 1.8 ft, what is the depth at the end of the rectangular channel, and what happens then?



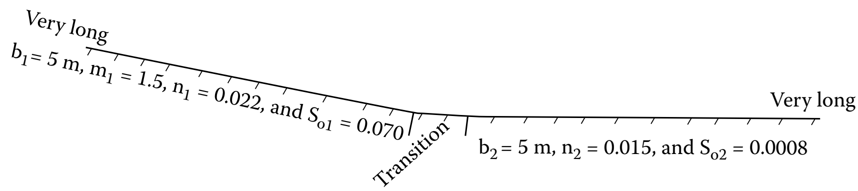
4.23 The sketches below show a rectangular and a trapezoidal channel receiving water from a gate at its beginning with the reservoir water depth of 3 m behind the gate. The rectangular channel has a bottom width b = 2 m, a Manning's n = 0.014, and a bottom slope $S_o = 0.05$, and the trapezoidal channel has a bottom width b = 2 m, a side slope m = 1, a Manning's n = 0.014, and a bottom slope $S_o = 0.0005$. What flow rate occurs in each channel if the depth immediately downstream from the gate in both cases is 0.5 m? (The trapezoidal channel ends in a free overfall at a not too long a distance downstream so the gate is free flowing.)



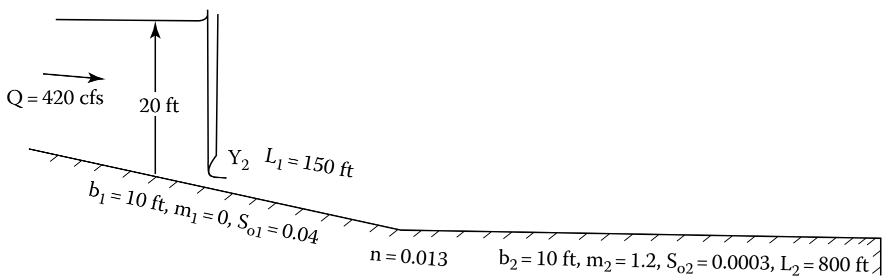
4.24 A flow rate of Q = 50 m^3/s enters a rectangular channel with a bottom width b_1 = 6 m, a roughness coefficient n_1 = 0.015, and a bottom slope of S_{o1} = 0.0006 from a reservoir with a water surface elevation H above the channel bottom. Downstream, at a distance of L_1 = 100 m, there is a sluice gate with a contraction coefficient C_c = 0.6 whose bottom is Y_g = 1.5 m above the channel bottom. At an additional distance of L_2 = 100 m, the channel reduces to b_2 = 5 m, and its bottom slope increases to S_{o2} = 0.002, and this downstream channel has a Manning's roughness coefficient n_2 = 0.013. At this transition, the channel bottom drops by Δz = 0.25 m. Determine the following: (a) the depth immediately upstream from the gate, (b) the head H of the reservoir above the channel bottom (ignore entrance losses), (c) the depth Y_4 in the downstream channel immediately downstream from the change in channel sizes, (d) the depth in the channel immediately upstream from the change in channel sizes, (e) the location of the hydraulic jump, (f) the energy and power loss in the hydraulic jump, and (g) the combined force on the transition and the drop between the two channels.



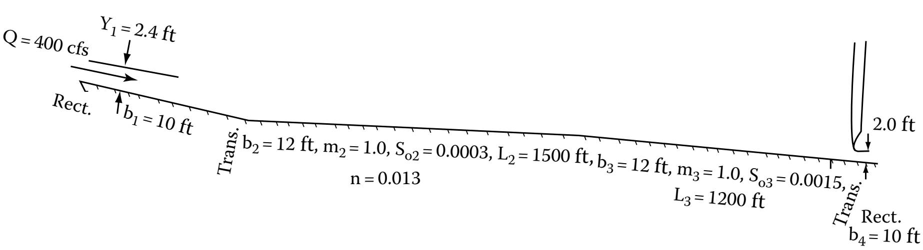
4.25 A break in grade from a trapezoidal channel with $b_1 = 5$ m, $m_1 = 1.5$, $n_1 = 0.022$, and $S_{o1} = 0.07$ to a rectangular channel with $b_2 = 5$ m, $n_2 = 0.015$, and $S_{o2} = 0.0008$ occurs. Write the system of equation that needs to be solved for the flow rate that will cause the hydraulic jump to be exactly at the end of the transition between the two channels and give the depths at the upstream and downstream ends of the transition and in the downstream channel. Make up a table that shows the flow rate Q, the normal depth Y_{o1}, the depth at the end of the transition Y_2, and the depth at the downstream end of the channel Y_{o2}, as a function of the side slope of the downstream channel m_2, for $m_2 = 0$ to $m_2 = 0.23$. What occurs when m_2 becomes larger than 0.2358? Also solve the problem using Chezy's equation if $e_1 = 0.01$ m and $e_2 = 0.0035$ m ($\nu = 1.007 \times 10^{-6}$ m²/s).



4.26 A flow rate of 420 cfs passes a gate in a steep rectangular channel with a bottom width $b_1 = 10$ ft and a bottom slope $S_{o1} = 0.04$. The depth upstream from the gate is 20 ft, and at a distance $L_1 = 150$ ft downstream from the gate the channel changes to a trapezoidal channel with $b_2 = 10$ ft, $m_2 = 1.2$, and a bottom slope $S_{o2} = 0.0003$. Downstream, at a further distance of $L_2 = 800$ ft, the channel ends in a free overfall. Both channels have $n = 0.012$. Determine the GVF profiles that exist, the force on the gate, and the position of the hydraulic jump.

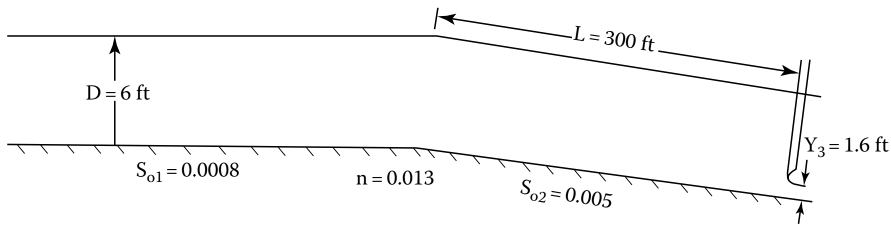


4.27 A flow rate of Q = 400 cfs is coming down a steep rectangular channel with a bottom width of $b_1 = 10$ ft. Through a short smooth transition, the channel becomes trapezoidal and mild with a bottom width $b_2 = 12$ ft, a side slope $m_2 = 1$, and a bottom slope $S_{o2} = 0.0003$. After a length of $L_2 = 1500$ ft, the bottom slope of this channel changes to $S_{o3} = 0.0015$. At a further distance of $L_3 = 1200$ ft, there is a short smooth transition to a rectangular section with $b_4 = 10$ ft that contains a vertical gate that produces a depth of 2.0 ft downstream from it. All channels have a Manning's roughness coefficient n = 0.013. Determine the types of GVF profiles that exist and the depths at their beginnings and ends. Also, solve this problem using Chezy's equation with e = 0.0035 ft and $\nu = 1.217 \times 10^{-5}$ ft²/s.



4.28 A long pipe with a 6 ft diameter has a break in grade from $S_{o1} = 0.0008$ to $S_{o2} = 0.005$. Its Manning's roughness coefficient is $n = 0.013$. At a distance 300 ft downstream from the break in grade a gate exists in the pipe that produces a depth $Y_3 = 1.6$ ft downstream from it. Determine the following, if the flow rate $Q = 100$ cfs: (a) the force on the gate, (b) the GVF profiles.

Solve this problem again if everything is the same as above except the slope of the downstream channel is changed from 0.005 to $S_{o2} = 0.013$.



4.29 By taking the differential of the specific energy function $E = Y + Q^2/(2gA^2)$ show that Equation 4.8 allows for the nonprismatic term that consists of $\partial A/\partial x|_Y$ in the numerator of Equation 4.6.

4.30 Write a program or develop computer software that is capable of solving GVF's in circular channels with diameters that change with position x.

4.31 A channel consisting of a rectangular cross section and a bottom width of $b = 4$ m is used to carry water between two lakes whose water surface elevations vary. The channel between the lakes is 1500 m long, has a bottom slope of $S_o = 0.0005$, and $n = 0.014$. Develop two separate stage-discharge relationships. The first in which the downstream reservoir water surface elevation is 2.5 m above the downstream bottom of the channel, and the upstream reservoir water surface varies, and the second in which the upstream reservoir water surface elevation is constant at 2.5 m above the channel bottom and the downstream water surface elevation varies.

4.32 A sluice gate with a contraction coefficient $C_c = 0.58$ exists in a trapezoidal channel with $b = 7$ ft and $m = 2$. At the gate, a smooth short transition with a minor loss coefficient of 0.12 takes the channel to a rectangular section with a bottom width of 7 ft. The channel is supplied by a reservoir that is 1500 ft upstream from the gate, and its water surface elevation is 3.5 ft above the bottom of the channel. The entrance loss coefficient into the channel is 0.18. The channel has a bottom slope $S_o = 0.00085$, and a Manning's $n = 0.014$. Determine the discharge through the channel as a function of the gate setting, i.e., a discharge curve that shows the expected flow through the channel as a function of the position that the gate is above the channel bottom, y_G. How much free board is needed to prevent the channel from overtopping when the gate is completely closed?

4.33 Develop a computer program that solves the GVF ordinary differential equation, Equation 4.8, by starting the solution with the Euler method and continues the solution utilizing the Hamming's method. Use this program to solve Example No. 3 in Appendix C and compare with the results given there.

4.34 Modify the computer listing in Appendix C under Problem No. 3 that calls on the subroutine ODESOL to solve the problem. Instead, call on the ISML subroutine DVERK for the solution. If ISML is available to you, then run Problem No. 3, and compare the results.

4.35 Modify the computer listing in Appendix C that utilizes the ISML subroutine DVERK, but instead, use the subroutine ODESOL to obtain a solution to this problem.

4.36 A gate in a trapezoidal channel creates a depth of 1.5 ft immediately downstream from it. The channel has the following properties: $b = 10$ ft, $m = 1.5$, $n = 0.014$, and $S_o = 0.0005$. If a flow rate of 800 cfs exists in the channel, determine the position where a hydraulic jump will occur. Assume that the channel is very long and nothing exists downstream from the gate.

4.37 A distance 3000 ft downstream from the gate in the previous problem there is a second gate that causes the depth downstream from it to be 2.5 ft. Now determine the location of the hydraulic jump between the two gates. The flow rate is 800 cfs.

4.38 A transition is to enlargen a rectangular channel from $b_1 = 5$ ft to a trapezoidal channel with $b_2 = 6$ ft and $m_2 = 1.5$. The slopes of the two channels are 0.004 and 0.00012, respectively. The design flow rate is 50 cfs, and the channels both have a Manning's n = 0.012. Use the approach of solving an ODE for dz/dx and have the bottom width through the transition follow the equation of two reversed parabolas joined at the mid point of the transition. The transition is to be 30 ft long.

4.39 A trapezoidal channel with a bottom width b = 12 ft, a side slope m = 1, a bottom slope of $S_o = 0.00085$, and a Manning's n = 0.012, takes water from a reservoir. Under design conditions, it is expected that the reservoir will have its water surface elevation 8.5 ft above the bottom of the channel's inlet, and the entrance loss coefficient is $K_e = 0.09$. The design calls for a transition to a circular section 1500 ft downstream, with the pipe on a steep slope, and with a diameter D = 12 ft. Assume that the depth entering the channel is uniform, and compute this depth and the flow rate in the channel. Under this assumption, will an M_2 or an M_1 GVF profile occur in the trapezoidal channel? What would the depth be at its downstream end? (The elevation of the bottom of the channel through the transition remains constant.)

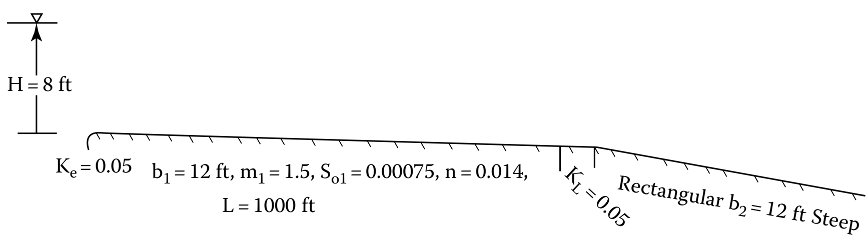
4.40 The pipe in the previous problem is laid on a bottom slope $S_o = 0.03$, and it has a Manning's n = 0.012. After a long distance downstream, a transition again takes the channel back to its original trapezoidal cross section, and the bottom slope equals $S_o = 0.001$ for a considerable distance thereafter. Will the hydraulic jump occur upstream, downstream, or within the transition under the assumption that the flow rate you computed in the previous problem is correct. Locate the position of this hydraulic jump.

4.41 Solve the GVF profile upstream from the transition to the pipe in Problem 4.39 and determine what the actual flow rate will be in this channel.

4.42 Solve the GVF profile in the pipe based on the flow rate you determined for Problem 4.41.

4.43 Locate the position of the hydraulic jump based on the flow rate you determined in Problem 4.41 if at its end, the pipe discharges into a reservoir whose water surface is 11.5 ft above its bottom, and the downstream channel is 3000 ft long.

4.44 An upstream trapezoidal channel with a bottom width of $b_1 = 12$ ft, and $m_1 = 1.5$, $n_1 = 0.014$, and a bottom slope of $S_{o1} = 0.00075$ changes to a steep rectangular channel with $b_2 = 12$ ft at a distance L = 1000 ft downstream from where it is supplied water by a reservoir with a head H = 8 ft above its bottom. Both the entrance and the transition loss coefficients are $K_e = 0.05$ and $K_L = 0.05$, respectively. Determine the flow rate in the channel, the depth at its entrance, the depth at the beginning of the transition, and the depth at the end of the transition.



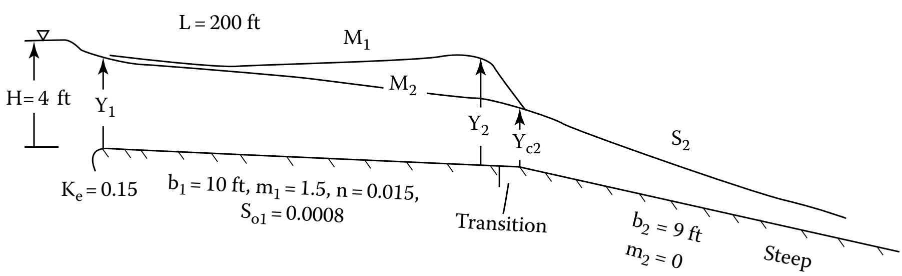
4.45 The same as the previous problem except that a gate exits at a distance L = 1000 ft downstream from the entrance, and the gate is set so it produces a depth of 2 ft immediately downstream from it.

4.46 Keeping the bottom of the channel at the same elevation through the first transition in Problems 4.39 through 4.41 is not a good design. Determine how much the bottom of the channel should change in elevation through the transition in order to keep the depth uniform throughout the entire upstream trapezoidal channel. How much should the bottom elevation change through the second transition? Assume that the transitions do not cause extra losses.

4.47 An upstream reservoir with a head H = 5 m supplies a trapezoidal channel with b_1 = 10 m, a side slope of m_1 = 1, a Manning's n = 0.014, and a bottom slope of S_{o1} = 0.0008. Downstream at a distance of L = 200 m, the channel becomes steep and changes to a rectangular channel with a bottom width of b_2 = 9 m. Determine the discharge and depths at the beginning and the end of the upstream trapezoidal channel. What is the depth at the beginning of the rectangular channel?

4.48 Modify the computer program SOLGVF.FOR so that it will accommodate a rectangular channel from the reservoir to the break in grade, and have the steep channel downstream from the break in grade with the same bottom width as the upstream channel (SOLGVF3.FOR).

4.49 Water enters a trapezoidal channel with b_1 = 10 ft, m_1 = 1.5, n = 0.015, and S_{o1} = 0.0008, that is 200 ft long, at which position the channel becomes rectangular with a bottom width b_2 = 9 ft, and this portion of the channel is steep as shown in the sketch below. The water surface of the reservoir is 4 ft above the bottom of the channel at its beginning, and the entrance loss coefficient is K_e = 0.15. Determine whether an M_1 or an M_2 GVF profile exists upstream from the break in grade, and solve the flow rate into the channel, as well as the depth Y_1 at the beginning of the upstream channel, and the depth at its end Y_2 before the transition to the rectangular channel begins. Solve this problem using TK-Solver, (or another applications program) in addition to using a computer program that utilizes the Newton method to solve the ODE and algebraic equations simultaneously. What are the flow rate, and depths if the width of the steep channel is 10, 11, and 12 ft?



4.50 When water flows in a channel between two reservoirs, the two equations that govern the problem are the upstream energy equations, and the ODE for the GVF throughout the channel. The velocity head at the end of the channel will be dissipated as it enters the downstream reservoir. Therefore, the depth at the downstream end of the channel will equal the head H_2 of the downstream reservoir above the channel bottom. Write a computer program that solves this type of problem for the flow rate Q and the upstream depth Y_1. Use this program to solve the following problem: The channel is 2000 ft long and is horizontal, the upstream reservoir has a head H_1 = 5 ft, and the downstream reservoir has a head H_2 = 4.5 ft, Manning's n = 0.013, the bottom width of the trapezoidal channel is b = 10 ft, and its side slope is m = 1. The entrance loss is K_e = 0.1.

4.51 Often, a channel between two reservoirs has a horizontal bottom, i.e., S_o = 0, and it is used to convey water from either reservoir to the other reservoir depending upon which reservoir has the higher water surface elevation. Note that the problem is the same regardless of which direction the flow is in; its just a matter of always selecting the reservoir with the higher water surface as the upstream reservoir. Modify the program you developed in the previous problem to print out a table of values that show how the flow rate and the upstream depth vary with the level of the downstream reservoir, with its water surface starting at $H_2 = 5.0$ ft, and dropping to $H_2 = 3.5$ ft in increments of 0.1 ft. The channel has the size, etc., of the channel in the previous problem.

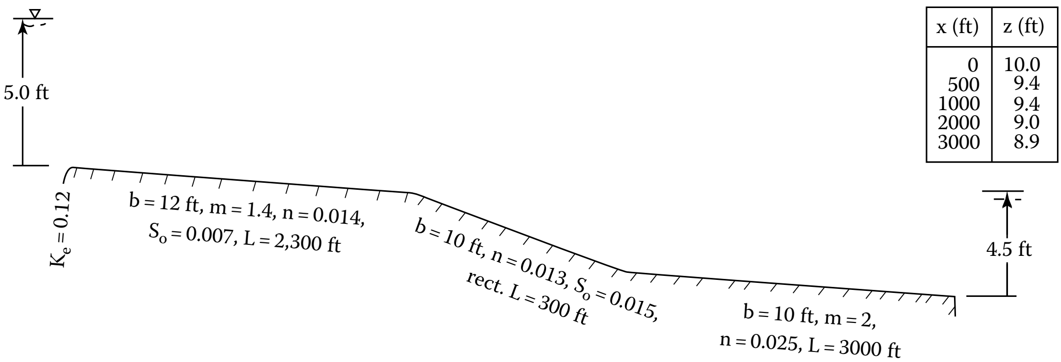
4.52 Modify the program you developed in Problem 4.50 so that rather than solving the flow rate, the flow rate is specified, and the bottom width b is an unknown along with the depth at the upstream end of the channel. Use this program to find what width of trapezoidal channel with a side slope of m = 1, $S_o = 0.0$, and n = 0.013 is required to convey Q = 200 cfs between two reservoirs 2000 ft apart if the upstream reservoir level is 5.0 ft above the channel bottom, and the downstream reservoir level is 4.5 ft above the channel bottom. The entrance loss coefficient is $K_e = 0.1$.

4.53 A trapezoidal channel is to convey water between two reservoirs that are 1020 m apart. The water may flow either way depending upon which reservoir has the higher water surface, and therefore the channel is to have a horizontal bottom. The channel is to have a side slope m = 1.5, and a Manning's n = 0.014. The entrance loss coefficient equals $K_e = 0.15$. The channel is to be sized (be determined) so that when the upstream reservoir has its water surface elevation 2 m above the channel bottom, and the downstream reservoirs water surface is 1.2 m above the channel bottom, 15 m^3/s is carried by the channel. With the bottom width you determine, make three tables that show the flow rate between the reservoirs for an upstream reservoir head of (a) $H_1 = 2$ m, (b) $H_1 = 1.5$ m, and (c) $H_1 = 1.0$ m. In each table have the downstream reservoir's head vary in increments of 0.1 m, starting 0.1 m below that of the upstream reservoir. (Since Q = 0 when the reservoir water surface elevations are equal, you can add another point to each of these tables without solving a problem.)

4.54 The bottom slope of a trapezoidal channel changes as given by the table below that provides the elevation of the channel bottom at several positions x. Modify one of the programs that handles GVF problems so that this table of x z values is read into the program and used to compute S_o at any position x, using a linear interpolation between two consecutive values. Using this program, solve the GVF in a 3000 ft long trapezoidal channel with b = 10 ft, m = 1, and n = 0.013 for a flow rate Q = 500 cfs when the depth at its downstream end is 5.0 ft. (Notice that the first 300 ft of channel has an adverse bottom slope.)

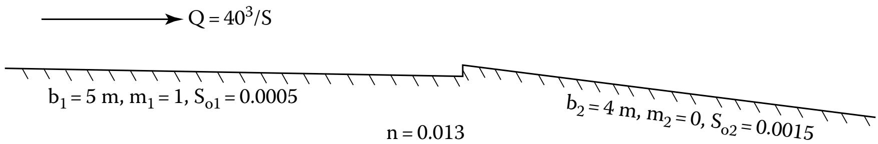
x (ft)	z (ft)
0	50.0
300	50.1
1000	49.0
2000	48.0
3000	47.5

4.55 A channel consists of three sections. The first is trapezoidal with b = 12 ft, m = 1.4, n = 0.014, and $S_o = 0.0007$, and is 2300 ft long. This section connects into a steep rectangular section that is 300 ft long. The second section has a width b = 10 ft, n = 0.013, and $S_o = 0.015$. The third section is also trapezoidal, built of earthen material, with a bottom width b = 10 ft, a side slope m = 2, and n = 0.025. Its bottom slope varies as given by the table of (x, z) coordinates below. At the upstream end, the channel is fed by a reservoir with a water surface elevation 5 ft above its bottom ($K_e = 0.12$), and at the downstream end the channel discharges into another reservoir with a water surface 4.5 ft above the channel bottom. The loss coefficient between the second and third section is $K_L = 0.1$. Find the flow rate and the water surface profile thorugh the channel system.



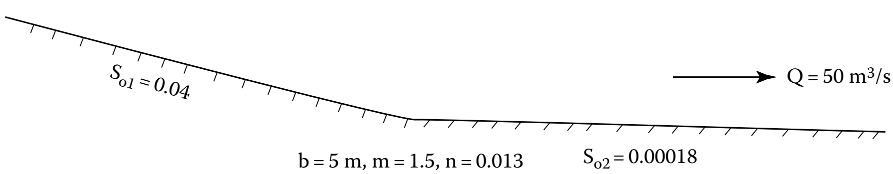
x (ft)	z (ft)
0	10.0
500	9.4
1000	9.4
2000	9.0
3000	8.9

4.56 A channel that contains a flow rate $Q = 40\,m^3/s$ changes from a trapezoidal shape with $b_1 = 5\,m$, $m_1 = 1$, and a bottom slope $S_{o1} = 0.0005$ to a rectangular section with $b_2 = 4\,m$, and a bottom slope $S_{o2} = 0.0015$. The bottom rises by $\Delta z = 0.15\,m$ through the transition. Manning's roughness for both channels is $n = 0.013$. (a) Determine what the depths will be immediately upstream and immediately downstream from the transition. (b) How much must the channel rise or fall through the transition for both channels to flow at a uniform depth?

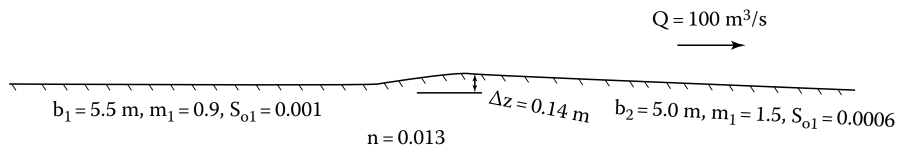


4.57 A flow rate $Q = 50\,m^3/s$ flows in a trapezoidal channel with a bottom width $b = 5\,m$, a side slope $m = 1.5$, and a Manning's $n = 0.013$. Upstream from a break in grade, the bottom slope is $S_{o1} = 0.04$, and downstream therefrom $S_{o2} = 0.00018$. (a) Determine whether the hydraulic jump will occur upstream or downstream from the break in grade. (b) What are the depths immediately upstream and downstream from the hydraulic jump? (c) How much energy per unit weight of fluid is lost through the hydraulic jump, and how much horsepower does this represent? (d) Locate the position of the hydraulic jump.

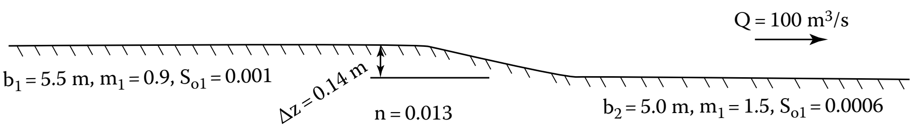
(To eliminate the need for you to solve Manning's equation, the normal depths in the two channels are $Y_{o1} = 0.749\,m$ and $Y_{o2} = 3.213\,m$, respectively.)



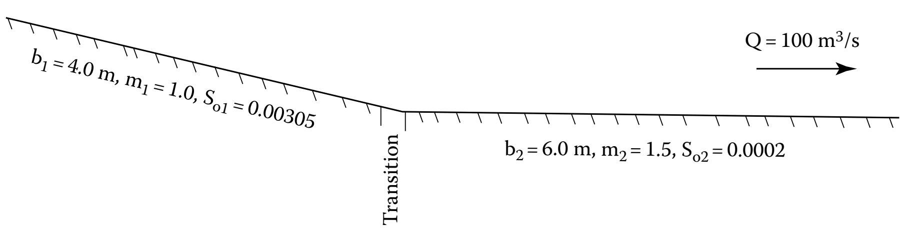
4.58 The size of a long trapezoidal channel is rapidly changed through a short smooth transition from $b_1 = 5.5\,m$, $m_1 = 0.9$, and $S_{o1} = 0.001$ to $b_2 = 5.0\,m$, $m_2 = 1.5$, and $S_{o2} = 0.0006$. Both upstream and downstream of the transition, the value of Manning's $n = 0.013$. Through the transition, the bottom of the channel rises by an amount $\Delta z = 0.14\,m$. For a flow rate $Q = 100\,m^3/s$, determine the following: (a) the depth immediately upstream and immediately downstream from the transition, (b) the type of GVF profile(s) that will occur, and sketch and label it(them) below, (c) the length(s) of the GVF profile(s), and (d) the force on the transition.



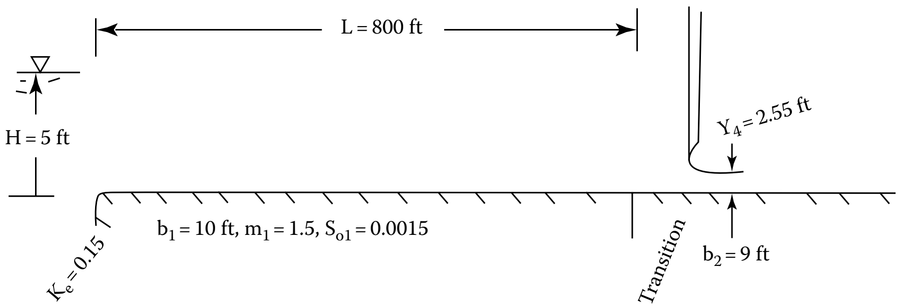
4.59 The size of a long trapezoidal channel is rapidly changed through a short smooth transition from $b_1 = 5.5$ m, $m_1 = 0.9$, and $S_{o1} = 0.001$ to $b_2 = 5.0$ m, $m_2 = 1.5$, and $S_{o2} = 0.0006$. Both upstream and downstream of the transition, the value of Manning's is $n = 0.013$. Through the transition, the bottom of the channel drops by an amount $\Delta z = 0.14$ m. For a flow rate $Q = 100\,m^3/s$ determine the following: (a) the depth immediately upstream and immediately downstream from the transition; (b) the type of GVF profile(s) that will occur, and sketch and label it (them) below; (c) the length(s) of the GVF profile(s); and (d) the force on the transition. (Note that this problem with the dropping bottom is quite different from the previous problem with the rising bottom.)



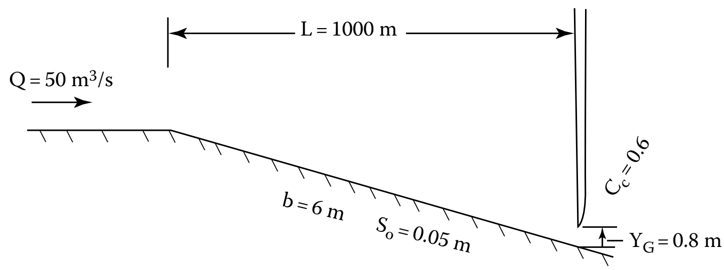
4.60 A break in grade occurs from an upstream steep trapezoidal channel to a downstream mild trapezoidal channel. Upstream, the channel consists of: $b_1 = 4.0$ m, $m_1 = 1.0$, and $S_{o1} = 0.00305$ to $b_2 = 6.0$ m, $m_2 = 1.5$, and $S_{o2} = 0.0002$. Both channels have a Manning's $n = 0.013$. The flow rate is $Q = 100\,m^3/s$. (a) Determine whether the hydraulic jump will occur upstream, within, or downstream from the transition. (b) What are the depths immediately upstream and downstream from the hydraulic jump? (c) What is the force on the transition?, and (d) Locate the position of the hydraulic jump.



4.61 A channel with a bottom width of $b_1 = 10$ ft, a side slope $m_1 = 1.5$, a roughness coefficient $n = 0.013$, and a bottom slope $S_{o1} = 0.0015$ has a gate at a position $L = 800$ ft downstream from the beginning of the channel. The gate cause a depth of flow $Y_4 = 2.55$ ft immediately downstream from it. The channel is supplied by a reservoir with a water surface elevation $H = 5$ ft above the channel bottom, and the minor loss coefficient at the channel entrance is $K_e = 0.15$. At the position of the downstream gate, the channel becomes rectangular with a bottom width $b_2 = 9$ ft. Solve the flow rate in the channel, and the depths at its beginning and at its end just before there is a transition to the rectangular section at the gate. Also solve the problem using Chezy's equation with $e = 0.004$ ft and $\nu = 1.217 \times 10^{-5}$ ft²/s.



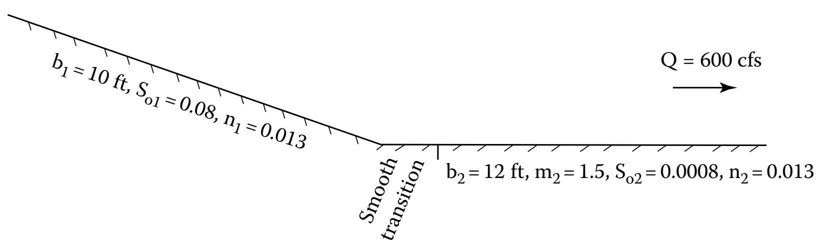
4.62 If a gate is located 1000 m downstream from the break in grade and is set at $Y_g = 0.8$ m above the channel bottom, and its contraction coefficient is $C_c = 0.6$, the flow rate in the channel is $Q = 50\,m^3/s$, and the channel is rectangular with b = 6 m, n = 0.014, and a bottom slope S_o = 0.05, describe how you would locate where the hydraulic jump will occur, and then determine this location.



4.63 A rectangular channel with a bottom width of b = 5 m receives its water from a reservoir with a water surface H = 2.8 m above the channel's bottom. The channel has a bottom slope of S_o = 0.0008 for a length of L = 100 m, and then it becomes a steep channel. Determine the flow rate in the channel and the depth at the channel's entrance and at the break in grade. Ignore minor losses and use n = 0.015.

4.64 A flow rate from a dam spillway enters a stilling basin whose width expands from 10 to 15 ft, and the side slope increases from 0 to 2 over a distance of 1000 ft. The flow rate over the spillway is Q = 1000 cfs, and the channel through the stilling basin has a bottom slope S_o = 0.0005, and a Manning's n = 0.014. The depth of flow at the beginning of the stilling basin is 2 ft. Determine the water surface profile(s) through the length of the 1000 foot long stilling basin.

4.65 A steep rectangular channel with a bottom width b_1 = 10 ft, bottom slope S_{o1} = 0.08, and a Manning's roughness coefficient n_1 = 0.013 changes to a trapezoidal channel with a bottom width b_2 = 12 ft, a side slope m_2 = 1.5, bottom slope S_{o2} = 0.0008, and a Manning's roughness coefficient n_2 = 0.013 through a smooth transition just beyond the change in grade. Determine whether the hydraulic jump will occur in the steep channel, in the transition, or in the mild channel when the flow rate is Q = 600 cfs. Determine the position of the hydraulic jump and the power lost through the hydraulic jump.



4.66 Assume that a uniform flow occurs in the three downstream channels of a four channel system that join at a junction. Channel 1 flows into the junction and the other three channels flow out from the junction. All channels are trapezoidal with n = 0.014 and the following parameters: $b_1 = 15$ m, $m_1 = 1.2$, $S_{o1} = 0.0009$, $b_2 = 3$ m, $m_2 = 1.0$, $S_{o2} = 0.0007$, $b_3 = 3$ m, $m_3 = 1.0$, $S_{o3} = 0.0008$, $b_4 = 2$ m, $m_4 = 1.5$, $S_{o4} = 0.0010$. Determine the depth and the flow rates in all the channels if channel #1 is supplied by a reservoir with a water surface elevation 3 m above it bottom, and all minor loss coefficients equal 0.1 for the following four cases: (a) the upstream channel is short so it has only one depth, (b) also find b_1 for channel 1 which is long and also contains a uniform flow, (c) also find the minimum width b_1 of channel 1 so that it will allow a uniform flow to occur in the three downstream channels, and (d) channel 1 is $L_1 = 500$ m long ($b_1 = 15$ m) and may have a different depth at its upstream and downstream ends.

4.67 A gate with a contraction coefficient $C_c = 0.7$ has its bottom set 1.5 ft above the bottom of the channel. At this position, the channel is rectangular with a width of 8 ft. A short distance upstream, the channel is trapezoidal with $b_1 = 10$ ft, $m_1 = 1.5$, n = 0.012, and $S_{o1} = 0.001$. If the channel is 1500 ft long and is supplied by a reservoir with a water surface elevation H = 5 ft above the channel bottom ($K_e = 0.1$) determine the discharge into the channel, and the depth at its beginning and just upstream from the gate.

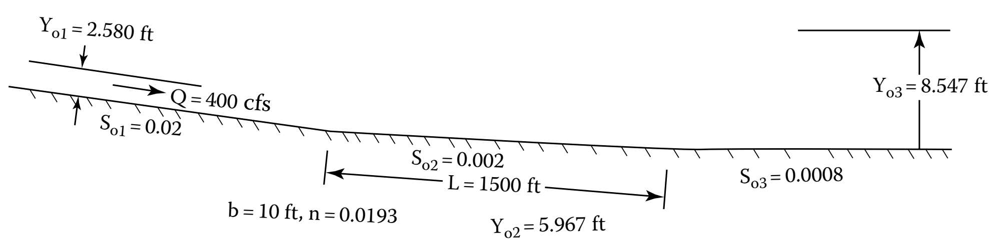
4.68 Modify the program SOLGATE, developed in Example Problem 4.10 so that it uses the third alternative discussed in this problem, namely, including the term for a nonprismatic channel while solving the ODE through the transition. Then solve this and the next example problems with this program and compare the results with those obtained in the example problems. Make the transitions 20 m and 20 ft long in these problems respectively, so that the total lengths from the reservoir to the gate are 620 m and 1020 ft, respectively.

4.69 Modify the program SOLGATE, developed in Example Problem 4.10 so that it uses the second alternative discussed in this problem, namely, solving four, rather than three equations simultaneously, and then solve the same problems as in the previous problem.

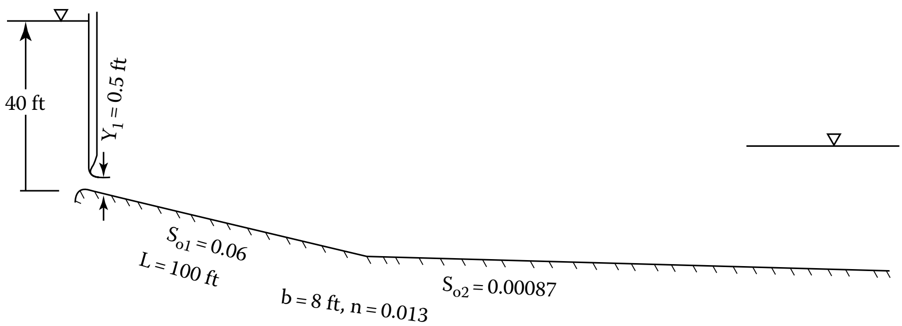
4.70 Example Problem 4.11 obtained a series of solutions in which the gate was raised from 3 ft in increments of 0.2 ft until the gate cleared the water surface. In addition to raising the gate as in this example problem, study the effects that different bottom slopes have on the solutions.

4.71 Solve the bottom width needed so that a trapezoidal channel with a side slope of m = 1.5, $S_{o1} = 0.00085$, and n = 0.013 will take a flow rate of Q = 300 cfs from a reservoir with a head of 5.2 ft ($K_e = 0.15$), if downstream from the reservoir at a distance of L = 1200 ft there is a gate with a width of 10 ft, and its bottom is set 3.0 ft above the channel bottom. The gate's contraction coefficient is $C_c = 0.6$.

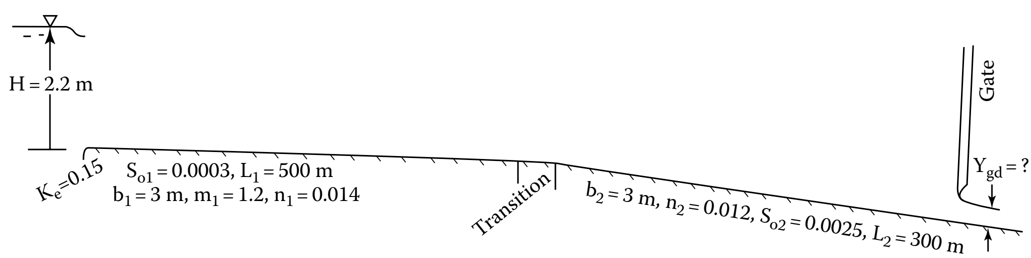
4.72 A rectangular channel has a bottom width of b = 10 ft and a Manning's n = 0.013 over its entire length. At its beginning, the channel has a bottom slope of $S_{o1} = 0.02$. Its bottom slope then changes to $S_{o2} = 0.002$, and at a distance 1500 ft downstream from this change in grade, the bottom of the channel changes to a bottom slope $S_{o3} = 0.0008$. The flow rate is Q = 400 cfs, the channel is very long with the upstream steep slope, and is also very long with the downstream flattest slope. Determine what the GVF profiles will be, and sketch them in below, with a proper label for each. Then solve the position where the hydraulic jump will occur.



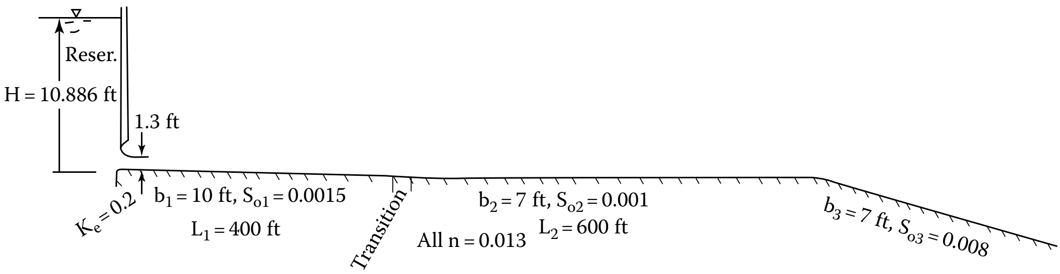
4.73 A rectangular channel with a bottom width b = 8 ft and a Manning's n = 0.013 is supplied water by a gate at a reservoir whose water surface is 40 ft above the channel bottom. For a length of 100 ft downstream from the gate the slope of the channel bottom is $S_{o1} = 0.06$, and at this position the bottom slope changes to $S_{o2} = 0.00087$. Thereafter, the channel is very long. Just downstream from the gate the depth of flow is $Y_1 = 0.5$ ft. Show the GVF profiles that will occur, and locate the position of the hydraulic jump.



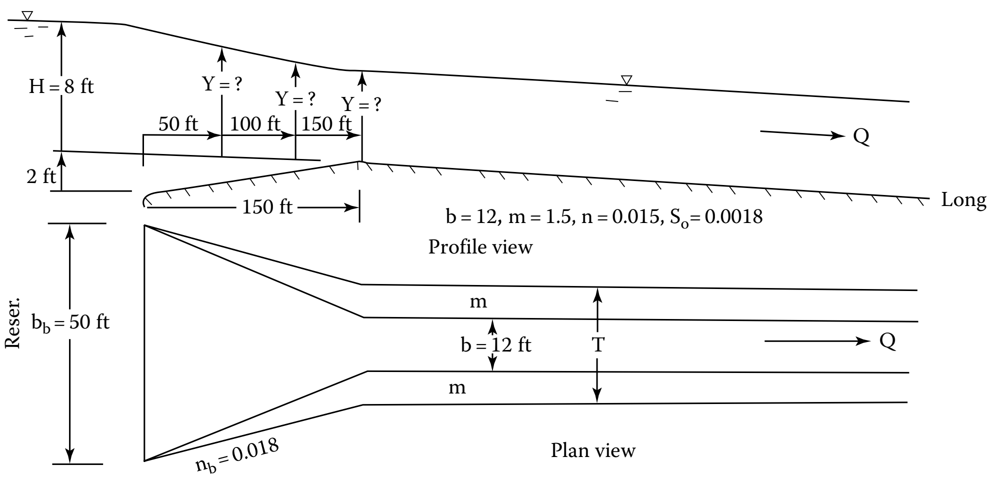
4.74 A trapezoidal channel with $b_1 = 3$ m, $m_1 = 1.2$, $n_1 = 0.014$, and a bottom slope $S_{o1} = 0.0003$ receives water from a constant head reservoir. After 500 m, there is a smooth transition to a rectangular channel with $b_2 = 3.0$ m, $n_2 = 0.012$, and $S_{o2} = 0.0025$. At a position $L_2 = 300$ m downstream in the second channel there is a gate. Determine what depth the gate should produce downstream from it if the reservoir head is H = 2.2 m, and a uniform flow is to exist in the upstream trapezoidal channel. The entrance loss coefficient is $K_e = 0.15$.



4.75 A reservoir with its water surface elevation 10.886 ft above the bottom of a 10 ft wide rectangular channel produces a depth of 1.3 ft downstream from the gate as shown. The gate and the entrance loss coefficient is $K_e = 0.2$. This channel has a bottom slope $S_{o1} = 0.0015$ and 400 ft downstream from the reservoir changes through a smooth transition into a 7 ft wide rectangular channel with a bottom slope $S_{o2} = 0.001$. This second channel is $L_2 = 600$ ft long, and after 600 ft the bottom slope changes to $S_{o3} = 0.008$. Determine the depths at positions between channels, and prove whether the hydraulic jump will occur in channel 1, in channel 2, or in the transition between these channels. Also determine the length of the GVF in channel 3 until it is within 1 % of the normal depth. (All n's = 0.013)



4.76 An inlet structure from a reservoir to a trapezoidal channel consists of a rising bottom and converging side walls as shown below by the profile and plan views. The beginning width of the structure is 50 ft and it is rectangular here, and it is 150 ft long and ends with the same size as the channel, namely, a bottom width of 12 ft and a side slope of m = 1.5. The downstream channel has a Manning's n = 0.015 and a bottom slope $S_o = 0.0018$. If the reservoir head is 8 ft, and you ignore all entrance losses through the inlet structure, what are the flow rate and the depth in the downstream channel? Compute the depths at the positions x = 50 ft, x = 100 ft, and x = 150 ft within the inlet structure.



4.77 In the previous problem, the losses through the inlet structure were ignored. Account for these losses by solving the GVF through the inlet structure over its 150 ft length and adverse slope, and determine the flow rate and the depths throughout the inlet and the downstream channel. (How many equations do you need to solve simultaneously?)

4.78 A 300 ft long culvert is a pipe with a diameter of 10 ft. It has a Manning's roughness coefficient n = 0.018, a bottom slope of $S_o = 0.018$, and an entrance loss coefficient of $K_e = 0.10$. If the water depth upstream of the culvert is $H_1 = 8$ ft, and the downstream depth is $H_2 = 7$ ft, determine the flow rate through the culvert, and the water surface profile through its length.

4.79 A circular culvert with an 8 ft diameter is 300 ft long and is laid on a bottom slope $S_o = 0.005$. Its Manning's roughness coefficient is n = 0.022, and the entrance loss coefficient is $K_e = 0.25$. For a constant downstream head $H_2 = 6$ ft, generate a table that gives the depth at its beginning, and the upstream head H_1 for flow rates from 50 cfs to 500 cfs in increments of 25 cfs.

4.80 If both the upstream and downstream heads H_1 and H_2 of the culvert in the previous problem are 6 ft, what is the flow rate Q? How much does this differ from the uniform flow under an upstream head $H_1 = 6$ ft? How much does it differ if the uniform depth in the culvert were 6 ft?

4.81 Solve the (d) part of Example Problem 4.21. In this problem, the solution obtained from CULVERTU indicated that the effect of the S_1 extended the culvert's entrance, and therefore its flow is controlled by downstream conditions ($D = 5$ ft, $S_o = 0.018$, $n = 0.013$, $K_e = 0.2$, $L = 80$ ft, $H_1 = 4.5$ ft, and $H_2 = 5.5$ ft).

4.82 A circular culvert with a diameter $D = 8$ ft, and an entrance loss coefficient $K_e = 0.15$, a Manning's $n = 0.015$ and a bottom slope $S_o = 0.02$, is 250 ft long. Solve the flow rate and the depths throughout the culvert if the upstream head $H_1 = 7$ ft, and the downstream depth is (a) $H_2 = 3$ ft, (b) $H_2 = 5$ ft, and (c) $H_2 = 13$ ft.

4.83 What is the maximum flow rate that can be achieved in the culvert of the previous two problems if the upstream head $H_1 = 6$ ft? How effective would lowering H_2 be in increasing the flow rate above the uniform flow that would occur with $H_1 = 6$ ft?

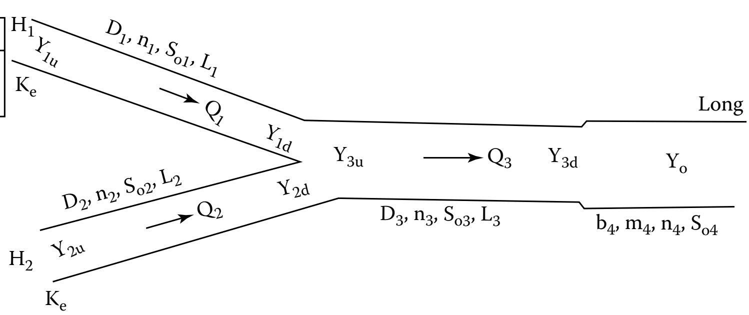
4.84 A circular culvert with diameter $D = 5$ ft is laid with a bottom slope of $S_o = 0.003$, and has a Manning's $n = 0.013$. The entrance loss coefficient is $K_e = 0.1$ and the culvert is 100 ft long. If the flow rate is $Q = 100$ cfs, solve the depths through the culvert and the upstream reservoir head H_1 if (a) $H_2 = 2$ ft, (b) $H_2 = 4$ ft, (c) $H_2 = 5.1$ ft, and (d) $H_2 = 5.5$ ft.

4.85 Modify program CULVERTU so that it solves the flow in a box culvert with an upstream control, rather than in a circular culvert. A box culvert has a rectangular shape with a width b and a height He. Using your modified program solve the conditions through a culvert if a flow rate $Q = 150$ cfs occurs in a 400 ft long culvert with $b = 7$ ft, $He = 4$ ft, a bottom slope $S_o = 0.0075$, a Manning's $n = 0.013$, and an entrance loss coefficient $K_e = 0.12$, if the water depth at the downstream end of the culvert is 5.2 ft above its bottom.

4.86 Modify program CULVERTD so that is solves the flow in a box culvert with a downstream control, rather than in a circular culvert. A box culvert has a rectangular shape with a width b and a height He. Using your modified program, solve the conditions through a culvert if the following apply: (a) $b = 7$ ft, $He = 4$ ft, $H_1 = 3.8$ ft, $H_2 = 3.9$ ft, $K_e = 0.12$, $L = 400$ ft, $n = 0.013$, $S_o = 0.0008$, (b) the same as above except that the downstream depth is now above the top of the culvert with $H_2 = 4.05$ ft, (c) the same as (b) except use the Darcy–Weisbach equation rather than Manning's equation for the portion of the culvert that contains full flow. In using the Darcy–Weisbach equation, use four times the hydraulic radius in place of the pipe diameter, i.e., $D = 4R_h$.

4.87 A culvert consists of two pipes that join a third pipe as shown in the sketch below. The first pipe is supplied by a reservoir with a head of $H_1 = 4.8$ ft, and the second pipe is supplied by a reservoir with a head of $H_2 = 5.0$ ft. Downstream for the third pipe, which is the end of the culvert, the flow discharges into a trapezoidal channel with the following properties: $b_4 = 10$ ft, $m_4 = 1.5$, $n_4 = 0.022$, and $S_{o4} = 0.0003$. This channel is very long. The properties of the three culvert pipes are given in the table, and the entrance loss coefficients are $K_e = 0.1$. Set up the governing equations.

No.	D (ft)	n	S_o	L (ft)
1	6	0.013	0.001	1000
2	8	0.013	0.0009	800
3	10	0.013	0.0005	1200

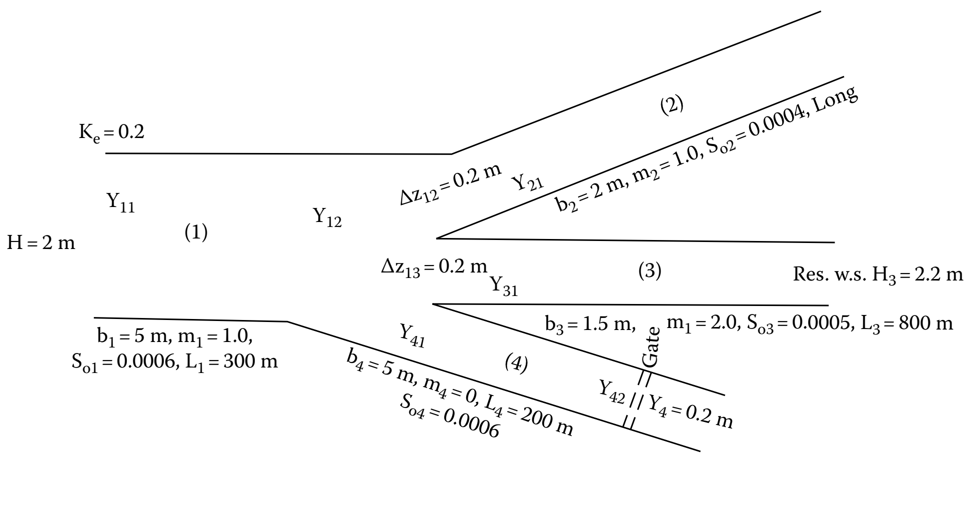


4.88 Find the flow rate in each of the three culverts, and the depths at the beginning and at the end of each of the pipes that form the culvert system of the previous problem. (To do previous problem you will need to develop a program that sets up the needed 10 equations to solve the following 10 unknowns: Q_1, Q_2, Q_3, Y_{1u}, Y_{1d}, Y_{2u}, Y_{2d}, Y_{3u}, Y_{3d}, Y_o).

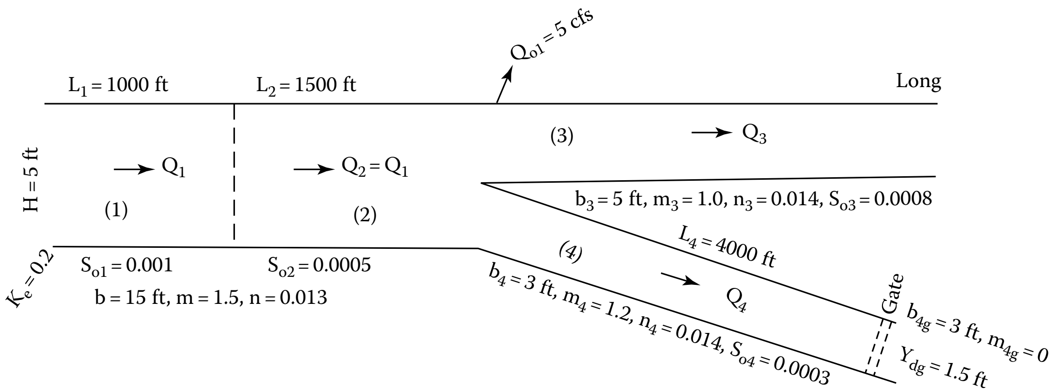
4.89 Assume that in the three pipe Y junction culvert system of the previous problem, the flow rates in Culverts 1 and 2 are known respectively to be Q_1 = 80 cfs and Q_2 = 170 cfs, and that the upstream heads H_1 and H_2 are unknown. Solve these for the heads as well as the depths at the beginning and the end of each of the pipes that form the culvert system. (Notice that now it is not necessary to solve a system of equations.)

4.90 Solve Problem 4.87 if the pipe for culvert #1 has its diameter increased from 6 to 8 ft, with all sizes and conditions the same including H_1 = 4.8 ft and H_2 = 5 ft. Why is the total flow rate increased by a very small amount?

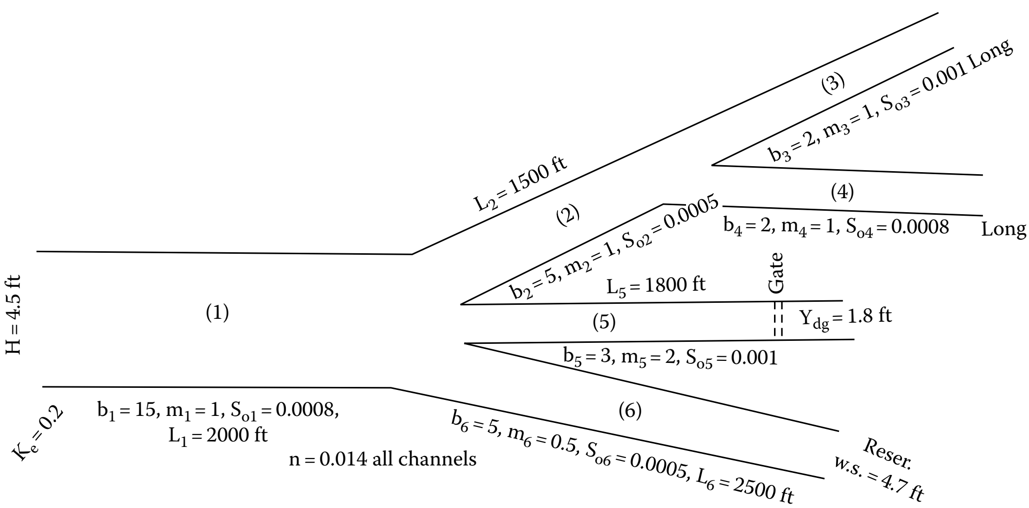
4.91 The sketch below shows a four channel system that receives water from an upstream reservoir with a head H = 2.0 m, and eventually delivers the water through three channels that branch off in different directions from the main channel. The properties of the channel are given in the table below. Channel 2 is long, channel 3 ends in a reservoir, and channel 4 is controlled by a gate that produces a depth Y_4 = 0.2 m downstream from it. Upstream from the gate, channel 4 has a bottom slope S_{o4} = 0.0006. The bottom of channel 2 is Δz_{12} = 0.2 m above the bottom of channel 1 and channel 3's bottom is 0.2 m below the bottom of channel 1. Write out the equations whose solution gives the flow rates in the four channels, and their depths at both the upstream and downstream ends, where the depth is unknown. All n = 0.014, and K_e = 0.2. Obtain a solution for the unknowns by first using a bottom width for channel 4 of b_4 = 3 m, and then b_4 = 5 m.



4.92 The four channel system shown below is supplied by a reservoir whose water surface elevation is 5 ft above the channel bottom. After a distance of 1000 ft, the bottom slope of channel 1 changes from S_{o1} = 0.001 to S_{o2} = 0.0005. Channel 4 contains a gate 4000 ft downstream from its beginning that causes a depth of Y_{dg} = 1.5 ft downstream from it. An outflow of Q_{o1} occurs at the junction of channel 2 with channels 3 and 4. The sizes of the channel are given on the sketch. Find the flow rates and depths throughout this channel system.



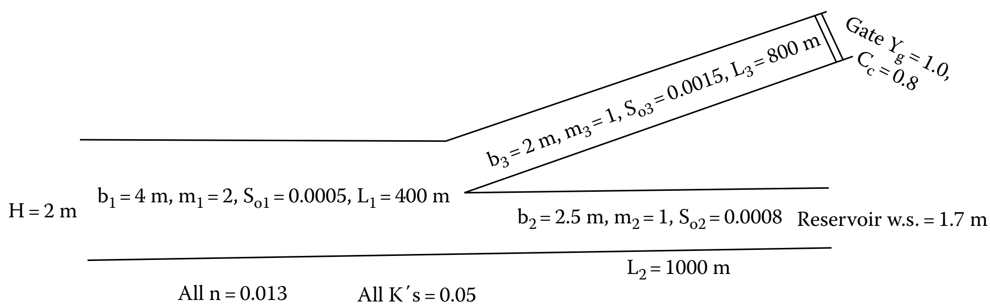
4.93 The six channel system shown below is supplied by a reservoir with a head of H = 4.5 ft (with an entrance loss coefficient), $K_e = 0.2$. All channels are trapezoidal, as shown on the sketch. Channels 3 and 4 are long with no downstream controls. Channel 5 has a gate 1800 downstream from its beginning and is set to produce a depth of 1.8 ft downstream from it.Channel 6 discharges into a reservoir whose water surface is 4.7 ft above the channel bottom at a distance of 2500 ft downstream from its beginning. Solve the flow rates in all channels, and their depths at the beginning and ends of the channels, if their flow is not uniform; if so, find their normal depth.



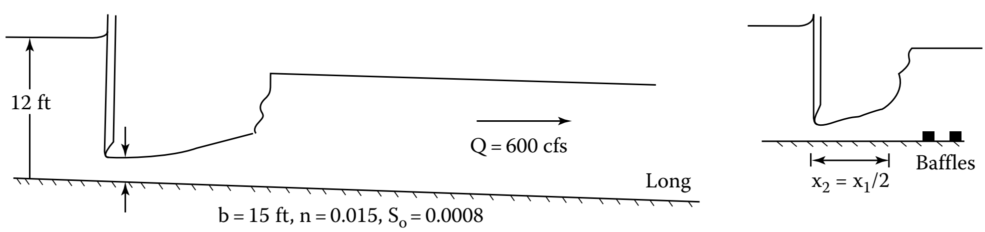
4.94 Solve the previous problem for reservoir heads varying from H = 3.0 ft to H = 5.0 ft in increments of 0.5 ft. Try obtaining a solution for H = 5.5 ft. Explain why the solution is not obtained.

4.95 In the previous problem, it was discovered that when H = 5.5 ft the critical conditions exist at the end of channel 2 and limit the flow rates into channels 3 and 4. Solve the problem for reservoir heads of H = 5.3 ft, H = 5.5 ft, H = 5.75 ft, and H = 6.0 ft.

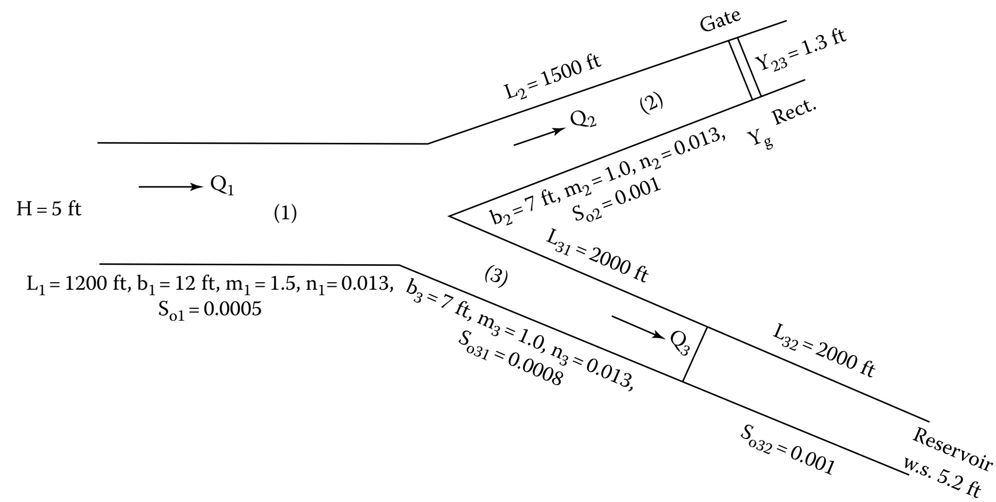
4.96 A main trapezoidal channel with a bottom width of $b_1 = 4$ m, a side slope $m_1 = 2$, and a bottom slope of $S_{o1} = 0.0005$ receives water from a reservoir whose water surface is 2 m above the bottom of the channel at its entrance. At a distance of $L_1 = 400$ m, this channel divides into two trapezoidal channels with $b_2 = 2.5$ m, $m_2 = 1$ and $S_{o2} = 0.0008$, and $b_3 = 2.0$ m, $m_3 = 1$, and $S_{o3} = 0.0015$. At a position of $L_2 = 1000$ m downstream, the one branch channel discharges into a reservoir with a water surface elevation 1.7 m above the channel bottom, and at a distance $L_3 = 800$ m, the other branch channel contains a gate whose contraction coefficient is $C_c = 0.8$, with its position $Y_g = 1.0$ m above the channel bottom. Manning's roughness coefficient is n = 0.013 for all three channels, and all the minor loss coefficients are K = 0.05 for the entrance from the upstream reservoir, as well as the branches between the channels. Write out the system of equations whose solution will give the flow rates in each of these three channels, as well as their depths at both their beginnings and at their ends. Then solve this system of equations.



4.97 A rectangular channel that is 15 ft wide, has a Manning's roughness n = 0.015, and a bottom slope of $S_o = 0.0008$. It contains a gate near it upstream end that causes a depth upstream there from equal to 12 ft when the flow rate is Q = 600 cfs. The channel is very long. Determine where the hydraulic jump will occur. How much energy and horsepower is dissipated by the jump? If the jump is to occur at a position x equal to one-half that computed above, what force must be applied by the baffles at the bottom of the channel in the region within the length of the jump.



4.98 A main trapezoidal channel with $b_1 = 12$ ft, $m_1 = 1.5$, $n_1 = 0.013$, and $S_{o1} = 0.0005$ branches into two channels at a distance $L_1 = 1200$ ft downstream from the reservoir that supplies it with a head H = 5 ft. The first branch channel has $b_2 = 7$ ft, $m_2 = 1.0$, $n_2 = 0.013$, and $S_{o2} = 0.001$. At a distance $L_2 = 1500$ downstream from the branch this channel changes to a rectangular section where a gate exits. At the gate $b_{22} = 6$ ft, and the depth of water immediately downstream from this gate is $Y_{23} = 1.3$ ft. The other branch channel has $b_3 = 7$ ft, $m_3 = 1.0$, $n_3 = 0.013$, and a bottom slope $S_{o31} = 0.0008$ for a length $L_{31} = 2000$ ft. At this position, the bottom slope changes to $S_{o32} = 0.001$, and a distance $L_{32} = 2500$ ft further downstream this channel discharges into a reservoir whose water surface elevation is 5.2 ft above the channel bottom. Determine the flow rates in the three channels, as well as the depths at their beginnings and their ends, i.e., solve Q_1, Q_2, Q_3, Y_{u1}, Y_{d1}, Y_{u2}, Y_{d2}, Y_{u3}, Y_{m3}.



4.99 Solve Example Problem 4.30 if a gate exists at the downstream end of channel #4 and has a depth of 1.5 ft downstream from the gate.

4.100 Solve Example Problem 4.31 if a gate exists at the downstream end of channel #4 and has a depth of 1.5 ft downstream from the gate.

4.101 The program that solves Example Problems 4.30, etc., requires an estimate of the flow rates and the upstream and downstream depths in each of the channels. Remove this READ statement and add the statements necessary that will generate starting values for these variable for the Newton method.

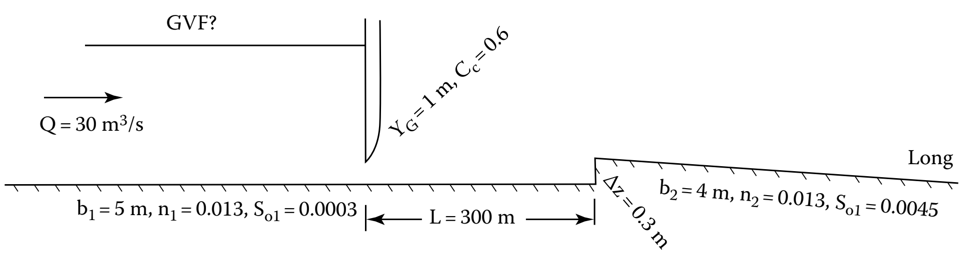
Answer: Possible code to accomplish this consists of the following that is added just before the Newton iteration begins in the MAIN program:

```
      IPERM=.TRUE.
      Y=.8*H
      DX=KL(1)*SO(1)*(CC/FN(1))**2
      12 F(1)=H-Y-DX*(AR(1,Y)/PERM)**.6666667
      Y=Y+.01
      F(2)=H-Y-DX*(AR(1,Y)/PERM)**.6666667
      DIF=.01*F(1)/(F(2)-F(1))
      Y=Y-DIF-.01
      IF(ABS(DIF).GT. .0001) GO TO 12
      X(1)=SQRT((H-Y)/KL(1))*AR(1,Y)
      WRITE(*,*)' Y=',Y,X(1)
      DIF=0.
      DO 14 I=2,NO
      F(I)=CC/FN(I)*AR(I,Y)**1.6666667/PERM**
     &.6666667*SQRT(SO(I))
      IF(ICTL(I).EQ.1) F(I)=.7*F(I)
      IF(ICTL(I).EQ.2) THEN
      IF(DB(I).LT.0.) THEN
      F(I)=.8*F(I)
      ELSE
      F(I)=1.2*F(I)
      ENDIF
      ENDIF
      ENDIF
14    DIF=DIF+F(I)
```

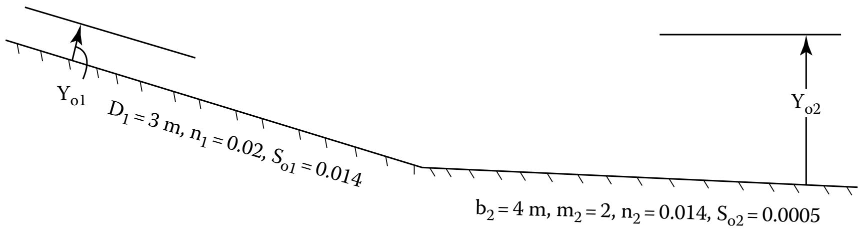
```
      DX=(X(1)-DIF)/2.
      X(1)=X(1)-DX
      DIF=X(1)/DIF
      X(NO2+1)=Y
      DO 16 I=1,NO
      IF(I.GT.1) THEN
      X(I)=DIF*F(I)
      IF(ICTL(I).EQ.1) THEN
      X(I+NO2)=1.2*Y
      ELSE IF(ICTL(I).EQ.2) THEN
      X(I+NO2)=.75*Y
      ELSE
      X(I+NO2)=YG(I)
      ENDIF
      ENDIF
16    X(I+NO)=Y
      IPERM=.FALSE.
```

4.102 The fourth channel in Example Problem 4.30 is replaced by a pipe with a diameter D = 5 ft. Solve all flow rates and depths.

4.103 A flow rate of Q = 30 m³/s flows in a rectangular channel that contains a sluice gate that is set at a height Y_G = 1 m above the channel bottom (C_c = 0.6). This channel has the following properties: b_1 = 5 m, n_1 = 0.013, S_{o1} = 0.0003. At a distance L = 300 m downstream from the gate, the bottom of the channel rises Δz = 0.3 m and the section changes smoothly to a rectangular channel with a width of b_2 = 4 m and a bottom slope of S_{o2} = 0.0045 (Manning's roughness coefficient is also n_2 = 0.013). This downstream portion of the channel is very long. Do the following: (a) determine the depth upstream from the gate; (b) identify the GVF profile upstream from the gate and compute its length; (c) determine the depth, the specific energy, the momentum function, and the Froude number in the downstream channel (i.e., determine these immediately downstream from the rise in bottom); (d) determine the depth, the specific energy, the momentum function, and the Froude number immediately upstream from the rise in the channel bottom; (e) determine the force on the hump and the transition; (f) sketch in and label the GVF profiles that will occur; (g) compute the change in the water surface elevation across the hump; (h) describe in words, equations, and/or sketches how you would locate the position of the hydraulic jump.



4.104 Determine at what flow rate the hydraulic jump will occur exactly at the break in grade between the upstream circular section with a diameter D_1 = 3 m, n_1 = 0.02, and S_{o1} = 0.044 and the downstream trapezoidal channel with b_2 = 4 m, m_2 = 2, n_2 = 0.014, and S_{o2} = 0.0002. For a flow rate of Q = 25 m³/s prove whether the hydraulic jump will occur upstream or downstream from the break in grade. Long channels occur both upstream and downstream from this break in grade.

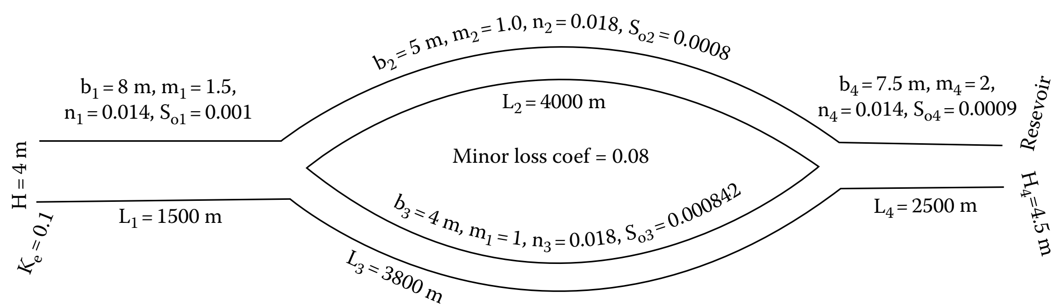


4.105 A rectangular channel with a bottom width of b = 4 m receives its water from a reservoir with a water surface H = 2.5 m above the channel's bottom. The channel has a bottom slope of S_o = 0.001, n = 0.013, and a length of L = 100 m, and then it becomes a steep channel. Determine the flow rate in the channel and the depth at the channel's entrance and at the break in grade. Ignore minor losses. (You should only complete one cycle of iterations to clearly demonstrate that you understand the process required to obtain the solution.)

4.106 A parallel system of channels exists as in Example Problem 4.35. All specifications are as given in this example problem, except at the upstream junction there is an outflow of 5 m^3/s and channel 1 has its width increased from 5 to 7 m. Solve the depths and flow rates throughout this system of channels. Why should the upstream channel be enlarged with this outflow? What would the flow rates be if the side slope of the downstream channel 5 at the gate were not reduced to 0?

4.107 A parallel system of channels exists as in Example Problem 4.35. All specifications are as given in this example problem, except at the upstream junction there is an outflow of 10 m^3/s and at the downstream junction there is an inflow of 5 m^3/s, and the width of channel 1 is b_1 = 7 m. Solve the depths and the flow rates throughout this system of channels.

4.108 A parallel system of channels exists as in Example Problem 4.35. All specifications are as given in this example problem, except at the downstream end of channel 5 there is a break in grade to a steep channel that is rectangular and has a bottom width of 5 m, and the width of Channel 1 is 7 m. Solve the depths and flow rates throughout this system of channels. Also solve the system of channels shown below.



Answer:

Input data to program SOLPAR (see Problem 4.111)

```
5
1 7.0 1.5 .014 .0008 500 0 .08
1 2.0 1.  .014 .0015 400 0 .08
1 1.5 1.3 .014 .0015 400 0 .08
1 3.0 1.0 .014 .0015 400 0 .08
1 5.0 1.5 .014 .001  800 0 .08
5 0. -1.5 1.97 2
.000001 0.005 2.5 0. 0. 9.81
42. 11. 11. 20. 2.18 2.3 2.3 2.3 2.5 2.18 1.97
```

Solution to the five channels at junction

No.	b	m	n	S_o	L	d_z	Y_u	Y_d	Q
1	7.00	1.50	0.014	0.000800	500.0	0.00	2.307	2.172	45.16
2	2.00	1.00	0.014	0.001500	400.0	0.00	2.273	2.367	13.62
3	1.50	1.30	0.014	0.001500	400.0	0.00	2.270	2.367	14.36
4	3.00	1.00	0.014	0.001500	400.0	0.00	2.268	2.367	17.17
5	5.00	1.50	0.014	0.001000	800.0	0.00	2.367	2.026	45.16

4.109 A parallel system of channels exists as in Example Problem 4.35. All specifications are as given in this example problem, except that the downstream channel 5 discharges into a reservoir whose water surface elevation is 3.7 m above the bottom of the channel. Solve the depths and the flow rates throughout this system of channels.

4.110 A parallel system of channels exists as in the previous problem above but now contains an inflow of 5 m^3/s at both the upstream and downstream junctions. Solve the depths and flow rates throughout this system of channels.

4.111 A parallel system of channels exists as in Problem 4.106 above but now channel 2 has its bottom raised by an amount of $\Delta z = 0.1$ m, and channel 3 has its bottom dropped by an amount of $\Delta z = 0.1$ m. There is still an inflow of 5 m^3/s at both the upstream and the downstream junctions. Note that as the bottom changes elevation at the upstream junction, the slopes of the channels must also change. Solve the depths and flow rates throughout this system of channels. To solve this problem, modify the program SOLPARG.FOR so that it only handles a series of parallel channels between an upstream channel and a downstream channel.

Answer:

Program listing SOLPAR.FOR

```
      PARAMETER (N=12,M=25)
      LOGICAL*2 IPERM,NTRAN
      REAL F(M),D(M,M),FN(N),SO(N),L(N),B(N),FM(N),DZ(N),K
     &L(N),X(M),LT
      INTEGER*2 INDX(M),ITYP(N),ICTL
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NO,NOM,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,
     &DZ,KL,QL,DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QOUT1,QN,
     &Q2G,BB,FMM,IPERM,NTRAN,JI
      DATA IN,IOUT/2,3/
      IPERM=.FALSE.
      NTRAN=.TRUE.
C ITYP=1 for trap. channels, otherwise channel is circular.
C ICTL=1,2 OR 3 for types of downs. controls, gate=1,crit=2,
C reser=3. YG is depth behind gate for gate; = 0 for crit
C DB and DFM are changes (+ or -) across transition.
C IN UNKNOWN VECTOR X (NO-1) Q's COMES 1ST; (NO) UPSTREAM
C DEPTHS NEXT & 2 DOWNSTREAM DEPTHS (for 1st and NO's
C channel) LAST i.e.
C Q(I)=X(I) for I=1 to NO-1;YU(I)=X(I+NO-1) for I=1 to NO;
C YD(1)=X(NEQS-1 & YD(2)=X(NEQS). Note other YD's = YU(NO).
      READ(IN,*) NO,(ITYP(I),B(I),FM(I),FN(I),SO(I),L(I),DZ(I),
     &KL(I),I=1,NO),LT,DB,DFM,YG,ICTL
      NOM=NO-1
      NEQS=2*NO+1
      NO2=NEQS-2
```

```
      READ(2,*) TOL,ERR,H,QOUT1,QOUT2,G
      READ(2,*) (X(I),I=1,NEQS) ! GUESSES
C unknown (NO-1 Q'S 1ST, Yu next,Yd(1),Yd(NO)
      IF(G.GT. 30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      DO 10 I=1,NO
10    KL(I)=(KL(I)+1.)/G2
      NCT=0
20    QL=X(1)-QOUT1-QOUT2
      DO 30 I=1,NEQS
      F(I)=FUN(I)
      DO 30 J=1,NEQS
      DX=.015*X(J)
      IF(J.EQ.1) QL=QL+DX
      X(J)=X(J)+DX
      D(I,J)=(FUN(I)-F(I))/DX
      X(J)=X(J)-DX
      IF(J.EQ.1) QL=QL-DX
30    CONTINUE
      CALL SOLVEQ(NEQS,1,M,D,F,1,DD,INDX)
      DIF=0.
      DO 40 I=1,NEQS
      X(I)=X(I)-F(I)
40    DIF=DIF+ABS(F(I))
      NCT=NCT+1
      WRITE(*,101) NCT,DIF,(X(I),I=1,NEQS)
101   FORMAT(' NCT =',I3,' SUM =',E12.5,/(8F10.3))
      IF(NCT.LT.30 .AND. DIF.GT.ERR) GO TO 20
      WRITE(IOUT,100) NO
100   FORMAT(' Solution to',I3,' Channel at Junction',/,
     &1X,65('-'),/,' No b m n So L dz Yu',' Yd Q',/,
     &1X,65('-'))
      DO 50 I=1,NO
      IF(I.EQ.NO) THEN
      YD=X(NEQS)
      QQ=X(1)-QOUT1-QOUT2
      ELSE
      QQ=X(I)
      IF(I.EQ.1) THEN
      YD=X(NEQS-1)
      ELSE
      YD=X(NO2)
      ENDIF
      ENDIF
50    WRITE(IOUT,110) I,B(I),FM(I),FN(I),SO(I),L(I),DZ(I),
     &X(I+NOM),YD,QQ
110   FORMAT(I3,F7.2,F5.2,F6.3,F8.6,F7.0,F6.2,2F7.3,F8.2)
```

```
      END
      FUNCTION FUN(II)
      PARAMETER (N=12,M=25)
      EXTERNAL DYX
      LOGICAL*2 IPERM,NTRAN
      REAL FN(N),SO(N),L(N),B(N),FM(N),DZ(N),KL(N),X(M),
     &Y(1),DY(1),W(1,13),XP(1),YP(1,1),LT
      INTEGER*2 ITYP(N),ICTL
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/NO,NOM,NO2,NEQS,ITYP,ICTL,FN,SO,L,LT,B,FM,
     &DZ,KL,QL,DB,DFM,YG,X,H,G,G2,CC,TOL,PERM,TOPW,QOUT1,QN,
     &Q2G,BB,FMM,IPERM,NTRAN,JI
      H1=-.05
      HMIN=.0000001
      IF(II.EQ.1) THEN
      FUN=H-X(NO)-KL(1)*(X(1)/AR(1,X(NO)))**2
      ELSE IF(II.GT.1 .AND. II.LT.NO) THEN
      FUN=X(NEQS-1)+(X(1)/AR(1,X(NEQS-1)))**2/G2-X(NOM+II)
     &-KL(II)*(X(II)/AR(II,X(NOM+II)))**2+DZ(II)
      ELSE IF(II.GE.NO .AND. II.LE.NO2) THEN
      JI=II-NOM
      XX=L(JI)
      XZ=0.
      IPERM=.TRUE.
      IF(JI.EQ.NO) THEN
      XX=XX+LT-1.E-5
      QQ=QL
      ELSE
      QQ=X(JI)
      ENDIF
      IF(JI.EQ.1) THEN
      Y(1)=X(NEQS-1)
      ELSE IF(JI.EQ.NO) THEN
      Y(1)=X(NEQS)
      ELSE
      Y(1)=X(NO2)
      ENDIF
      IF(ICTL.EQ.2) Y(1)=1.2*Y(1)
      QN=(FN(JI)*QQ/CC)**2
      Q2G=QQ**2/G
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      FUN=X(NOM+JI)-Y(1)
      ELSE IF(II.EQ.NO2+1) THEN
      FUN=X(1)-QOUT1
      DO 20 I=2,NOM
20    FUN=FUN-X(I)
      ELSE
      NTRAN=.FALSE.
      BB=B(NO)+DB
      FMM=FM(NO)+DFM
      IF(ICTL.EQ.1) THEN
```

```
      FUN=X(NEQS)+(QL/AR(NO,X(NEQS)))**2/G2-YG-KL(NO)
     &*(QL/AR(NO,YG))**2
      NTRAN=.TRUE.
      ELSE IF(ICTL.EQ.2) THEN
      AA=AR(NO,X(NEQS))
      IPERM=.FALSE.
      FUN=QL**2*TOPW-G*AA**3
      ELSE
      FUN=X(NEQS)-YG
      ENDIF
      NTRAN=.TRUE.
      ENDIF
      RETURN
      END
```

C FUNCTION AR and subroutine DYX the same as in SOLPARG.FOR

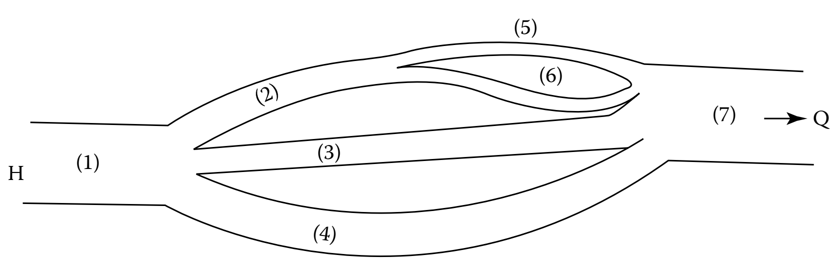
Input data to solve problem

```
5
1 5.0 1.5 .014 .0008 500 0 .08
1 2.0 1.  .014 .00175 400 .1 .08
1 1.5 1.3 .014 .00125 400 -.1 .08
1 3.0 1.0 .014 .0015 400 0 .08
1 5.0 1.5 .014 .001  800 0 .08
5 0. 0. 3.7 3
.000001 .1 2.5 -5. -5. 9.81
35. 14. 12. 14. 2.2 2.3 2.7
2.5 3. 2.3 3.7
```

Solution to the five channels at a junction

No.	b	m	n	S_o	L	d_z	Y_u	Y_d	Q
1	5.00	1.50	0.014	0.000800	500.0	0.00	2.202	2.346	42.56
2	2.00	1.00	0.014	0.001750	400.0	0.10	2.355	2.977	24.81
3	1.50	1.30	0.014	0.001250	400.0	–0.10	2.477	2.977	0.29
4	3.00	1.00	0.014	0.001500	400.0	0.00	2.415	2.977	22.46
5	5.00	1.50	0.014	0.001000	800.0	0.00	2.977	3.700	52.56

4.112 A main channel divides into three parallel channels. A distance downstream from this junction channel #2 divides into two parallel channels. Further downstream from this second junction all channels join at a third junction forming a system of seven channels. If the sizes of all channels are known, and their bottom slopes, lengths and roughness coefficients are also known, define the variables that are unknown, and provide the system of equations whose solution will give all flow rates and depths through this system of channels.



No.	b (ft)	m	n	S_o	L (ft)
1	12	1.0	0.014	0.0005	800
2	6	0.0	0.014	0.0004	1400
3	4	0.0	0.014	0.0004	2800
4	4	0.0	0.014	0.0004	2800
5	3	0.0	0.014	0.0004	1400
6	3	0.0	0.014	0.0004	1400
7	10	0.0	0.014	0.0003	2000

4.113 The channel of the previous problem has the geometry, etc., as given in the table below, and is supplied by a reservoir with a water surface elevation H = 4.5 ft at the upstream end, and a gate exists at the downstream end of channel 7 that produces a depth of 2.3 ft downstream from it. Solve this problem for the flow rates and the depths in all channels.

No.	b (ft)	m	n	S_o	L (ft)
1	12	2.0	0.014	0.0006	1200
2	8	1.0	0.014	0.0009	600
3	5	1.5	0.014	0.0008	1800
4	6	0.5	0.014	0.00085	1800
5	6	0.5	0.014	0.00065	1500
6	4	1.0	0.014	0.0006	1200
7	12	0	0.014	0.001	800

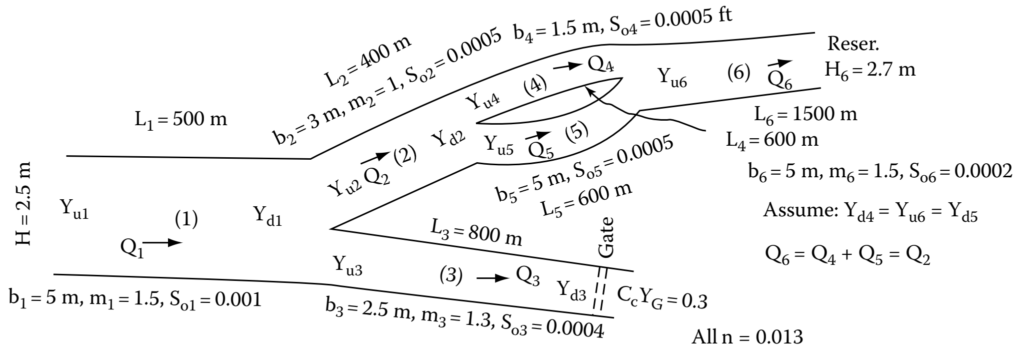
4.114 The channel configuration given in Problem 4.112 has the sizes, etc., as provided in the table below. The head of the upstream reservoir is H = 5 ft. Consider the following three cases: (a) a reservoir exists at the downstream end of channel 7 with a water surface elevation of 5 ft, and (b) channel 7 is very long, and (c) at a distant of 2000 ft downstream, channel 7 discharges into a reservoir whose water surface is below the bottom of the channel.

No.	b (ft)	m	n	S_o	L (ft)
1	12	1.0	0.014	0.0005	800
2	6	0.0	0.014	0.0004	1400
3	4	0.0	0.014	0.0004	2800
4	4	0.0	0.014	0.0004	2800
5	3	0.0	0.014	0.0004	1400
6	3	0.0	0.014	0.0004	1400
7	10	0	0.014	0.0003	2000

4.115 The treatment of channels in parallel assumed that the depth of flow at the end of each of two or more channels that flow into a single, equals the upstream depth in the downstream channel. Modify the program SOLPARG, or write a computer program that does not make this assumption, but rather allows different depths to exist in all channels that join at all junctions. Then solve Example Problems 4.35 and 4.36 with this program, and compare the solution with those obtained based on this assumption.

4.116 The sketch shows a system of six channels supplied by a reservoir at the head of channel 1 with a water surface elevation H = 2.5 m above the bottom of channel 1. Assume at junctions where several channels discharge into a single channel, that the depths are constant, i.e., $Y_{d4} = Y_{d5} = Y_{u6}$. Also note that $Q_6 = Q_4 + Q_5 = Q_2$, and therefore only one of Q_2 or Q_4 needs to be considered unknown.

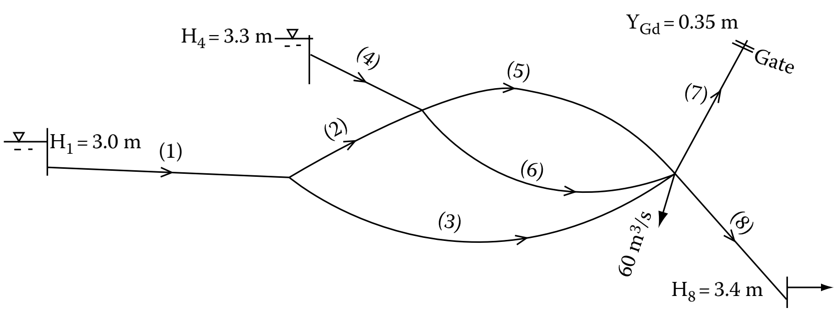
Identify how many unknowns there are and then write out as many equations that will solve these unknowns. Then solve the unknowns.



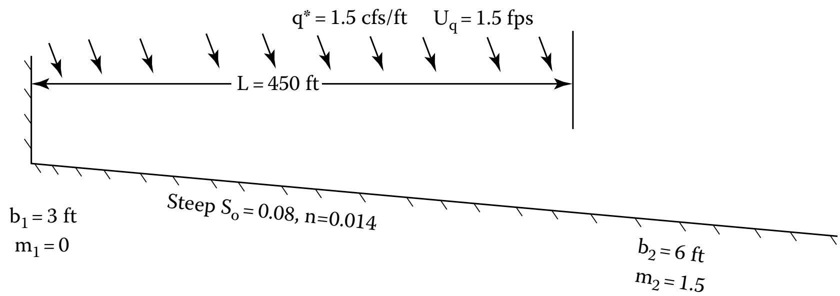
4.117 Modify the previous problem to eliminate the assumption that the upstream depths at the junction between channel 4, 5, and 6 are equal to the downstream depths. Also, do not necessarily eliminate Q_6 as an unknown. What additional equations are needed to solve the additional unknowns? What is the solution to this problem?

4.118 The sketch below shows a system of eight channels. Channels 1 and 4 receive water from upstream reservoirs with water surface elevations $H_1 = 3.0$ and $H_4 = 3.3$ m above the channel bottoms respectively. The flow rate in channel 7 is controlled by a gate in that channel that produces a depth of $Y_{Gd} = 0.35$ m downstream from it. A large flow rate of 60 m^3/s is taken from the third junction, i.e., the downstream junction. The size and the slopes of the channel are provided in the table below. Set up the equations that describe this problem. Obtain a solution for the flow rates in the eight channels, and the depths at their upstream and downstream ends.

b	m	S_o	L
8	2	0.0008	1000
5	1	0.0006	800
6	1	0.0005	2000
2	0	0.0003	800
4	1	0.0005	1200
4	1	0.0005	1200
5	1	0.0006	1500
4	0	0.0005	1500



4.119 A lateral inflow at a rate of $q^* = 1.5$ cfs/ft of length supplies water to an expanding trapezoidal channel at its beginning over a length of 450 ft. At the beginning, the channel is rectangular with a bottom width $b_1 = 3$ ft, and the bottom width and the side slope increase linearly to be $b_2 = 6$ ft and $m_2 = 1.5$ at $x = 450$ ft where the inflow stops. Thereafter, the size of the channel remains constant. The lateral inflow has a component of velocity $U_q = 1.5$ fps in the direction of the channel flow. Solve the position where the flow becomes critical, and the depth at this position. What are the depths at the beginning and the end of this lateral inflow section?



4.120 In the previous problem, investigate how the position of the critical flow changes with the slope of the channel bottom S_o and the component of velocity U_q. In other words, solve x_c for S_o = 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, and let U_q vary from 0 to 1.5 at least for S_o = 0.02.

4.121 A trapezoidal channel receives it water supply over a length of 150 ft. The amount of this lateral inflow is 4.0 cfs/ft of length, and this lateral inflow comes in with a velocity of 20 fps and at an angle of 60° from the direction of the channel. Over this lateral inflow section, the channel size changes from b_1 = 5 ft and m_1 = 0.5, to b_2 = 12 ft and m_2 = 1.5. The channel is very long, has a bottom slope of S_o = 0.02, and a Manning's n = 0.013. Downstream from the lateral inflow section the channel retains the same size and the bottom slope . Solve the depth over the 150 ft length of inflow.

If the downstream channel has a bottom slope of S_o = 0.0002, then solve the depths throughout the inflow length.

4.122 A collection channel such as that which receives water from the overflow spillway at the Hoover dam receives a lateral inflow of 20 cfs/ft of length over the 200 ft long crest of the spillway. This inflow enters at right angles to the flow in the main channel. The slope of the main channel is zero, it is 30 ft wide, and has a side slope of 1. Its Manning's n = 0.015. Determine the depths of water through the main channel if the slope of the channel beyond the collection length is S_o = 0.001.

4.123 The same as the previous problem except that the incoming lateral flow has a velocity component of 2.0 ft per second in the direction of the main channel flow.

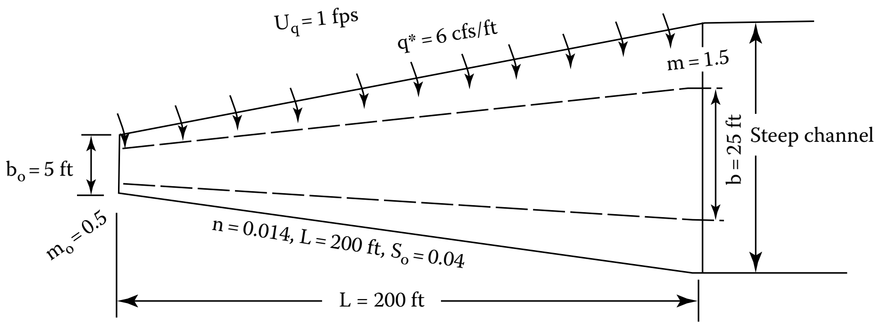
4.124 The same as Problem 4.122 except that the slope of the channel bottom is S_o = 0.05.

4.125 A lateral inflow occurs in a trapezoidal channel with a varying bottom width from b_1 = 4 ft to b_2 = 10 ft at its end, but the side slope is constant and is m = 1. The section of inflow is 300 ft long, and downstream therefrom the channel is the same as at the end of the inflow length. The slope of the channel bottom is S_o = 0.025 and it roughness coefficient is n = 0.014. If a lateral inflow of q* = 1.5 cfs/ft, determine (a) the position where the flow changes from the subcritical to the supercritical flow, (b) the GVF profile upstream from this point, (c) the GVF profile downstream from this point to the end of the spatially varied flow, and (d) the GVF profile downstream from the end of the spatially varied flow.

4.126 Repeat the previous problem for a lateral inflow q* = 3.0 cfs/ft.

4.127 An expanding channel is supplied by a constant inflow of q* = 3.5 cfs/ft over a 400 ft length. Over this length, the channel changes from b_1 = 2 ft and m_1 = 0 to b_2 = 12 ft and m_2 = 1.5. The bottom slope of the channel is S_o = 0.028 and its Manning's roughness coefficient is n = 0.013. Determine whether a critical flow section will occur within the spatially varied flow, and solve the GVF profiles that exit.

4.128 A lateral inflow q* = 6 cfs/ft provides the flow of a channel with a starting width of b_o = 5 m and m_o = 0.5. At the end of the lateral inflow portion of the channel, it has expanded to a bottom width of b_e = 25 ft and a side slope of m_e = 1.5, linearly. Manning's n = 0.014 and the bottom slopes at S_o = 0.04. At the end of the inflow channel, which is L = 200 ft long, the channel retains its size and shape for a long distance downstream therefrom. The lateral inflow has a velocity component of U_q = 1.5 fps in the direction of the channel flow. Determine the profile of water through the first 200 ft length of this channel. How does the position where the critical flow occurs change with the velocity component U_q? What effect does q* have on this position?



4.129 Solve the depth, etc., across the lateral inflow portion of the channel in the previous problem with parameters the same except that the slope of the channel bottom is S_o = 0.025. The lateral inflow is constant and equal to q* = 6 cfs/ft.

4.130 Modify Equation 4.17 to include the possibility that the component of the incoming velocity in the direction of the main channel flow is not zero.

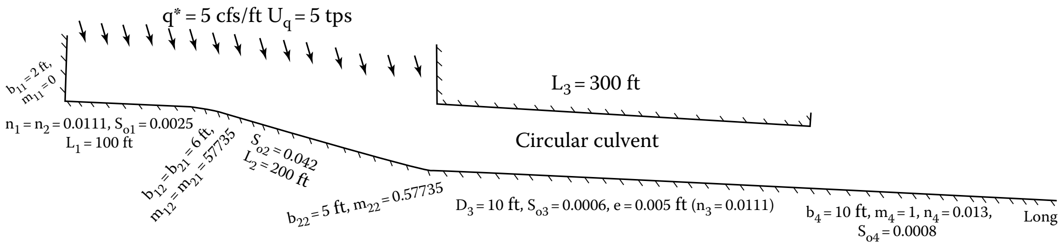
4.131 In Example Problem 4.41 solve the critical position x_c, for cases where its inflow has a velocity component in the direction of the main flow of U_q = −1 fps, U_q = 2 fps, and U_q = 5 fps. Then solve the spatially varied flow problem for these three cases.

4.132 Solve Example Problem 4.41 for a constant lateral inflow q* = 3.5 cfs/ft.

4.133 At the beginning of a trapezoidal channel, its bottom width is b_o = 5 m and at the end of a 100 m length of inflow at a rate of q* = 1.5 m^2/s, the bottom width is 7.5 m. The side slope varies from m_o = 0 to m_e = 2 at the end of the inflow length. The channel roughness coefficient is n = 0.015. Locate the position where the critical flow will occur for bottom slopes varying from S_o = 0.03 to S_o = 0.10 in increments of .005. Repeat the above solutions for situations in which the component of the velocity in the direction of the main channel vary from zero, i.e., U_q = 0 to U_q = 3 m/s in increments of 1 m/s. Plot both the position x_c and the critical depth Y_c here as a function of S_o with separate curves for U_q and explain why the relationship that you observe from these graphs occurs.

4.134 An upstream collection channel receives a lateral inflow over a length of 300 ft. The first portion of this channel has a bottom slope S_{o1} = 0.0025 and starts with b_{11} = 2 ft and m_{11} = 0, and at 100 ft, its bottom width is b_{12} = 6 ft and m_{12} = 0.57735. The changes in b and m are linear over this 100 ft length. The second portion of this channel (which is 200 ft long) has a bottom slope S_{o2} = 0.042, has b_{21} = 6 ft, and m_{21} = 0.57735 and at its end at position 300 ft has b_{22} = 5 ft and m_{22} = 0.57735. Channels 1 and 2 have $n_1 = n_2$ = 0.0111. At this 300 ft position, the circular culvert is 300m long and has a diameter of D_3 = 10 ft. The local loss coefficient between the upstream channel and the culvert varies according to the depth in the culvert; if $Y_{31} \le 5$ then KL = 0.1 (constant); if $5 < Y_{31} < 10$ then $K_L = 0.1 + 0.08(Y_{31} - 5)$; and if $Y_{31} \ge 10$ then K_L = 0.5 (constant). The equivalent sand roughness for use in the Darcy–Weisbach equation for the culvert is e = 0.005 in. (or Manning's n = 0.0111), and it has a bottom slope S_{o3} = 0.0008. Downstream from the culvert, there is a long trapezoidal channel with b_4 = 10 ft, m_4 = 1.0, n_4 = 0.013, and S_{o4} = 0.0008. The local loss coefficient between the culvert and the downstream channel is based on the formula $h_L = (V_3 - V_4)^2/(2g)$, where V_3 is the velocity in the culvert, and V_4 is the velocity in the downstream channel. If the lateral inflow over the first 300 ft of channel consists of q* = 5 cfs/ft, with a velocity component of Uq = 5 ft/s in the direction of the channel flow, solve the depth, etc., throughout this channel system.

Repeat the computations but for a lateral inflow of q* = 2.5 cfs/ft, with a velocity component of Uq = 5 ft/s.



4.135 Water discharges from the bottom of a channel by means of grates running parallel to the direction of flow for a distance of 60 ft. The width and the space of the grates result in 1% of the bottom width of the channel being open, and the discharge coefficient for these grates is Cd = 0.58 (for use in an orifice type equation). The channel has a trapezoidal cross section with b = 12 ft and m = 1.4. The channel is very long upstream and downstream from this outflow length and has a bottom slope S_o = 0.00083, and a Manning's n = 0.014. Determine the amount of discharge through the grates, and the water surface profiles if the flow rate downstream from the grates is Q_2 = 550 cfs.

4.136 Solve the previous problem, but in this case, the flow rate in the upstream channel is known and is Q_1 = 800 cfs.

4.137 If the flow rate upstream from the grates in Problem 4.135 is 800 cfs, determine the length of the grates, with a 10% opening ratio that would be needed to discharge the total flow rate.

4.138 Repeat Problem 4.135 except the grates have a 20% opening ratio.

4.139 Instead of using a rectangular channel as in Example Problem 4.66, a trapezoidal channel is to be used to discharge the 1200 cfs uniformly across the 1 mile long side weir. Size the beginning of the channel so that the average velocity in the channel at this point will be 6 fps and make the section the most hydraulically efficient, which consists of 1/2 of a hexagon.

4.140 The side discharge from the channel in Example Problem 4.66 is by means of pipes that penetrate the channel side wall at an elevation equal to the channel bottom. These pipes are to be spaced on a 10 ft interval over the mile-long channel, and will have a length so their discharges can be computed from an orifice type equation with a discharge coefficient of Cd = 0.32. The pipe diameters can change every 200 ft along the channel. Size these pipes.

4.141 A gutter along side a road way has its cross section defined as 1/2 a triangular section with a side slope of m = 3. The inflow from the crown of the road supplies an inflow of 0.01 cfs/ft. The gutter has a bottom slope of S_o = 0.002 and a Manning's n = 0.021. If the spacing between storm drain inlets is 1000 ft, and each inlet receives all of the gutter flow at that point, compute the GVF profile between storm drain inlets.

4.142 The storm inlets in the previous problem consist of bars running at 90° to the direction of the gutter and have 1/2 of the side of the gutter open, and have a discharge coefficient Cd = 0.45 (for use in an orifice type equation). Determine what length these grates should have to discharge all the water in the gutter.

4.143 A transition from a trapezoidal channel with a bottom width of b = 15 ft and m = 1.5 is to reduce the size of the channel to a rectangular shape with b = 10 ft over a length of 40 ft. The normal depth in the trapezoidal channel is 7.5 ft, and the normal depth in the rectangular channel is 4.8 ft. If the flow rate is 450 cfs, design this transition by (a) having the sides of the channel defined by two reversed parabolas, and (b) have the water surface follow a straight line through the transition.

4.144 At a position 1500 ft downstream from the beginning of a trapezoidal channel, the lateral outflow at a rate of q_o* = 1.5 cfs/ft begins and continues at this constant rate over a 50 ft length. Over this 50 ft length of outflow, the channel reduces in size from b_1 = 10 ft and m_1 = 2 to $b_2 = 8$ and $m_2 = 0$. The other channel parameters are $S_{o1} = 0.0008$, $n_1 = 0.014$, $S_{o2} = 0.0008$, $n_2 = 0.014$, $S_{ot} = 0.02$.

At a distance 3000 ft downstream from the end of the outflow there is a gate that causes the depth downstream from it to be at 2.5 ft. At the upstream end, the channel is supplied by a reservoir whose water surface elevation is 5.0 ft above the channel bottom. The width of the channel at the gate is 7.5 ft. Assume an entrance loss coefficient of $K_e = 0.08$ and a loss coefficient of $K_L = 0.1$ at the transition just in front of the gate. Solve the flow rate into this channel and the depth at its entrance and at the end of the channel.

Solution: $Q = 361.3$ cfs, $Y_1 = 4.74$ ft, $Y_2 = 5.91$ ft, $Q_e = 286.3$ cfs.

4.145 Repeat the previous problem except that there is an inflow of $q^* = 1.5$ cfs/ft.

4.146 In Problem 4.144 assume the same channel exists with the same lateral outflow, except that the channel discharges into a reservoir at it downstream end with a water surface elevation that is 4.8 ft above the bottom of the channel.

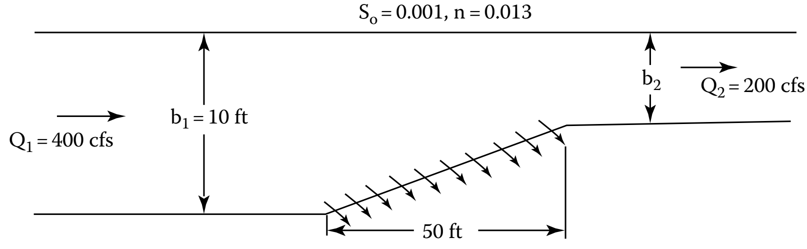
4.147 By differentiation of the closed-form solution for the profile over the lateral outflow from a side weir, Equation 4.16, prove that it is a solution for the ODE that describes this flow.

4.148 Use the equation for the closed-form solution of the GVF across a side weir to solve Example Problem 4.37. Since this theory applies only for a rectangular channel, and this channel is trapezoidal with b = 8 ft and m = 1, use a width b that will produce the same normal depth upstream from the $Q_l = 600$ cfs.

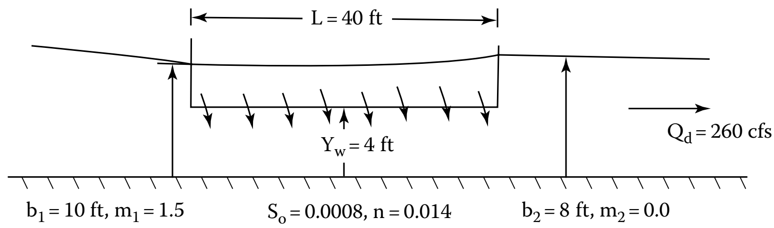
4.149 A lateral outflow section that is 30 m long exists midway through a trapezoidal channel with an outflow $q_o^* = 0.5\ m^2/s$. At the downstream end of the channel, a gate exist that causes the depth downstream from it to be 1 m. The other channel dimensions, etc., are $b_1 = 6$ m, $m_1 = 1.8$, $S_{o1} = 0.00085$, $n_1 = 0.0135$, $L_1 = 600$ m, $b_2 = 5$ m, $m_2 = 1.0$, $S_{o2} = 0.0012$, $n_2 = 0.013$, $L_2 = 1600$ m, $L_t = 30$ m, $S_{ot} = 0.005$, $b_3 = 4.5$ m, $Y_3 = 1$ m, H = 3 m (upstream reservoir head), $K_e = 0.1$, $K_L = 0.1$. Solve the flow rate into the channel, the depth at its beginning and at its end. Also plot the profile of the water surface depths through the channel.

Solution: $Q = 54.6\ m^2/s$, $Y_1 = 2.83$ m, $Y_2 = 5.32$ m, $Q_e = 39.6\ m^2/s$.

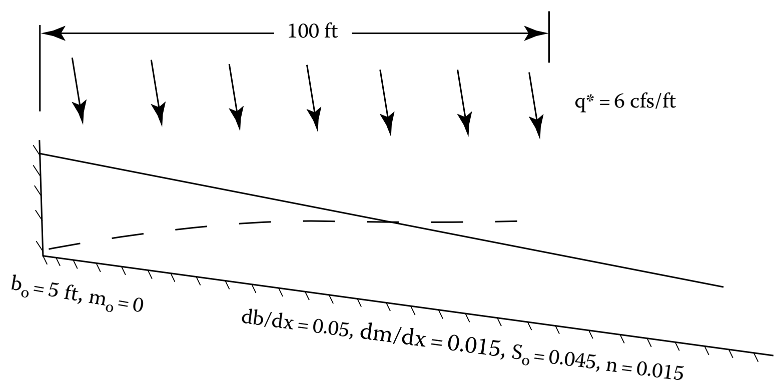
4.150 Determine the width of the rectangular channel downstream from the side weir in the channel shown below so that the water surface remains parallel to the bottom of the channel over the side weir, and that the energy line is also parallel to the channel bottom. The side weir is 50 ft long and the upstream channel is rectangular also with a bottom width of $b_1 = 10$ ft. The upstream flow rate is $Q_1 = 400$ cfs, and the downstream flow rate should be $Q_2 = 200$ cfs. The entire channel, including that upstream and downstream from the side weir has a bottom slope $S_o = 0.001$ and a Manning's roughness coefficient n = 0.013. (*Hint*: In solving the downstream bottom width b_2, you should also solve the depth of flow Y and the reduction in bottom width db/dx.) What height above the channel bottom should the weir's crest be placed if its discharge coefficient is $C_d = 0.45$ in the equation $q^* = C_d(2/3)(2g)^{1/2}(Y - Y_w)^{1.5}$?



4.151 Over the length of a side weir that is 4 ft above the channel bottom, the channel reduces from an upstream trapezoidal section with $b_1 = 10$ ft, and a side slope $m_1 = 1.5$ to a rectangular section with a bottom width of $b_2 = 8$ ft. The side weir is 40 ft long and has a discharge coefficient $C_D = 0.45$. If the flow rate in the downstream channel is $Q_d = 260$ cfs, what is the flow rate upstream, and how does the depth change across the side weir? What type of GVF exists upstream from the side weir?



4.152 A constant inflow of q* = 6 cfs/ft of length occurs into a trapezoidal channel over a length of 100 ft. The channel has a bottom slope of S_o = 0.045 and a Manning's value of n = 0.015. At its beginning, the bottom width is b_o = 5 ft and m_o = 0, and at the end of the lateral inflow length the bottom width is b_e = 10 ft and m_e = 1.5, and the channel keeps this size downstream therefrom. For a lateral inflow velocity, components U_q from 0 to 7 fps in the direction of the main channel flow determine where the critical depth will occur, and what the depths are at the beginning and at the end of the lateral inflow section. Also, solve the problem using Chezy's equation with a roughness coefficient e = 0.004 ft and $\nu = 1.217 \times 10^{-5}$ ft²/s.



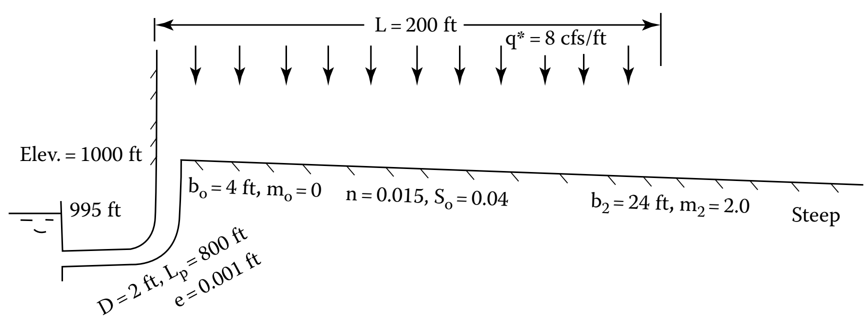
4.153 A lateral inflow section that is 50 m long exists midway through a trapezoidal channel with an inflow q* = 1.3 m²/s. At the downstream end of the channel, a gate exist that causes the depth downstream from it to be 2 m. The other channel dimensions, etc., are b_1 = 6 m, m_1 = 1.0, S_{o1} = 0.00085, n_1 = 0.013, L_1 = 1000 m, b_2 = 8 m, m_2 = 1.5, S_{o2} = 0.001, n_2 = 0.013, L_2 = 1000 m, L_t = 50 m, S_{ot} = 0.001, b_3 = 7.5 m, Y_3 = 2 m, H = 3.5 m (reservoir head), K_e = 0.1, K_L = 0.1. Solve the flow rate into the channel, the depth at its beginning, and at its end. Also plot the profile of the water surface depths through the channel.
Solution: Q = −0.16 m²/s, Y_1 = 3.50 m, Y_2 = 4.54 m, Q_e = 74.84 m²/s.

4.154 The same channel as is in the Example Problem 4.49, except that at the downstream end of this channel, after the channel reduces to a rectangular section, there is a free overfall that produces a critical depth. The rate of the lateral outflow over the 50 m long transition is 0.5 m²/s. The variables for this channel are b_1 = 5 m, m_1 = 1.5, S_{o1} = 0.001, n_1 = 0.013, L_1 = 950 m, b_2 = 4 m, m_2 = 1.0, S_{o2} = 0.001, n_2 = 0.013, L_2 = 500 m, L_t = 50 m, K_e = 0.05, K_L = 0.05. At the end of the channel, where critical flow occurs, the channel is rectangular with a width of 4 m. Obtain values for the depth at 50 m increments, and at a closer spacing across the lateral outflow section.
Solution: Q = 86.4 m³/s, Y_1 = 3.02 m, Y_2 = 4.24 m, Q_e = 61.4 m³/s.

4.155 The same channel as is in the Example Problem 4.49, except that at the downstream end of this channel, after the channel reduces to a rectangular section, there is reservoir with a water surface elevation of 4.8 m above the bottom of the channel. The rate of lateral inflow over the 50 m long transition is 0.5 m^2/s. The variables for this channel are as follows: b_1 = 5 m, m_1 = 1.5, S_{o1} = 0.001, n_1 = 0.013, L_1 = 950 m, b_2 = 4 m, m_2 = 1.0, S_{o2} = 0.001, n_2 = 0.013, L_2 = 500 m, L_t = 50 m, q^* = 0.5 m^2/s, K_e = 0.05, K_L = 0.05. Obtain values for the depth at 50 m increments, and at a closer spacing across the lateral outflow section.
Solution: Q = 46.4 m^3/s, Y_1 = 3.40 m, Y_2 = 4.65 m, Q_e = 71.4 m^3/s.

4.156 A collection channel such as that which receives water from the overflow spillway at the Hoover dam receives a lateral inflow of 0.6 m^3/s per meter of length over a 60 m long crest of the spillway. At the beginning of the main channel it is 3 m wide, and at its end, it is 9 m wide. Its beginning is rectangular in shape but at its end, the side slope of the trapezoidal section is 1. Both b and m vary linearly with x. Manning's roughness coefficient is n = 0.015. Determine the depths of water throughout the main channel for the following four cases: (a) the channel downstream from this channel continues with a width of b = 9 m and m = 1, and has a bottom slope of S_o = 0.001, and n = 0.013, the bottom is horizontal over the collection portion and the lateral inflow enters at right angles to the flow in the main channel; (b) the same collection channel as in (a), but the channel downstream therefrom is steep; (c) the downstream channel is steep, and the same size as the end of the collection channel, and the lateral inflow has a velocity of 3 m/s and is directed at an angle of 45° to the direction of the main channel flow; (d) the same as in (c) except the collection portion of the channel has a slope of S_o = 0.03.

4.157 A lateral inflow at a rate of q^* = 8 cfs/ft occurs over the first 200 ft length of a channel with a bottom slope of S_o = 0.04, and a Manning's roughness coefficient n = 0.015. At its beginning, the channel has a bottom width b_o = 4 ft, a side slope m_o = 0, and at the 200 ft position the bottom width is b_2 = 24 ft and m_2 = 2.0. Thereafter, the channel does not change size, but keeps the same bottom slope. At the beginning of the channel, a 2 ft diameter pipe that is 800 ft long (and an equivalent sand roughness e = 0.001 ft) takes water to a reservoir with a water surface elevation of 995 ft. The bottom of the channel at its beginning is at elevation 1000 ft. Determine the depths, etc., throughout the channel, and also determine what flow is being carried by the pipeline. Obtain two solutions to this problem; the first assumes that the inflow enters at right angles to the main channel flow, i.e., U_q = 0, and second where U_q = 2 fps.



4.158 A parking lot in the shape of a rectangle with a length of 800 ft in the north–south direction and 1200 ft in the east–west direction has a slope in the east–west direction of 0.0014, and is flat in the north-south direction. It is made of a black top with a Chezy's equivalent sand roughness e = 0.009 ft. It is drained by a gutter running in the north–south direction that consists of 1/2 of a trapezoidal channel with a bottom width of 4 ft, and a side slope of m = 3.

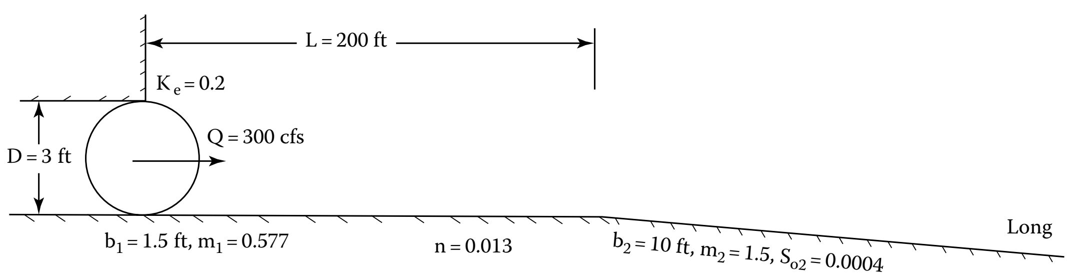
The vertical side extends upward so it cannot be overtopped. The depth of this gutter, i.e., the channel at the end of the parking lot is 1.5 ft below the level of the parking area at this end. At its end, the gutter discharges into a channel of the same size (i.e., 1/2 a trapezoid with b = 4 ft and m = 3). This downstream channel has a bottom slope of $S_o = 0.001$, and a Manning's roughness coefficient,n = 0.015. A high-intensity rainfall of 1.5 in./h occurs for a long enough time to establish a steady-state flow over the parking lot. Determine (a) the water surface depths over the parking lot, and (b) through the 800 ft length of gutter. Use your best judgment about how to handle the two directional flow over the parking lot using one-dimensional hydraulics.

4.159 Example Problem 4.48 determines the length of a tile drain that will accommodate a lateral inflow of $q^* = 0.00009339\,m^2/s$. The bottom slope of the tile is $S_o = 0$, i.e., it is flat. For this situation, answer the question regarding how significant the lateral inflow term is by solving the problem based on the equation $dY/dx = (S_o - S_f)/(1 - F_r^2)$, but have the flow rate given by $Q = xq^*$.

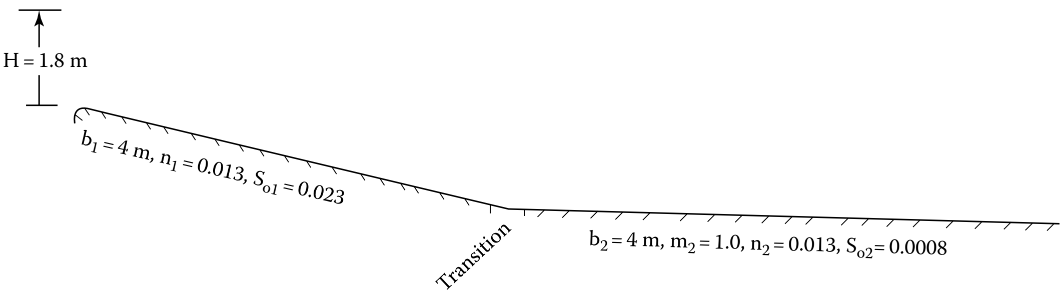
4.160 The difference in lengths determined in the last problem and Example Problem 4.48 is not large. Solve the bottom slope that will result in the same length of the tile as in the last problem, but include the lateral inflow term in the numerator of the ODE for GVFs. Then solve the adverse bottom slope that will cause the computed length to be that obtained in Example Problem 4.48, but neglect the lateral inflow term.

4.161 A tile drain with a diameter of 8 in. continues for a long distance downstream of the lateral inflow portion of it that receives seepage inflow. Manning's n for the pipe is n = 0.012. The no inflow portion of the pipe has a bottom slope $S_{o2} = 0.0005$. The lateral inflow to the upper portion of the pipe is $q^* = 0.001$ cfs/ft. Solve the following cases: (a) The upstream lateral inflow portion of the tile has the bottom slope $S_{o1} = 0.0005$. For this case, the length of the upstream portion of the pipe is 150 ft. (b) The upper portion of the pipe has an adverse slope $S_{o1} = -0.0001$. Determine what length of the drain will result in the pipe being full (or $Y_1 = 0.93D$) at its beginning. (c) The upstream portion of the tile is flat, $S_{o1} = 0$, and the downstream portion has a slope of $S_{o2} = 0.00055$. Determine what length of the drain will result in the pipe being full at its beginning. (d) The same as case (c) except that the downstream pipes bottom slope is $S_{o2} = 0.0006$. (Note that to solve cases (b) through (d), the program TDRAIN will need to be modified to solve for the uniform flow depth, rather than the critical depth to obtain the solutions.)

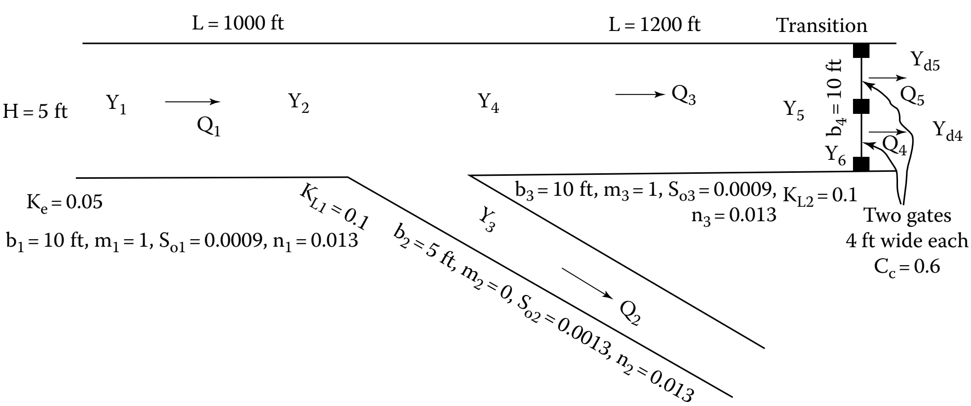
4.162 A 3 ft diameter pipe that contains a flow rate Q = 300 cfs discharges into an open channel as shown below. At the discharge point, the channel has a bottom width of $b_1 = 1.5$ ft, and a side slope $m_1 = 0.577$ to conform closely to the shape of the bottom half of the pipe. A 200 ft long transition increases the bottom width and the side slope linearly to $b_2 = 10$ ft and $m_2 = 1.5$, respectively. The transition has a horizontal bottom. At the end of the transition, the bottom slope of the channel is $S_{o2} = 0.0004$, and the Manning's roughness for the transition and the long prismatic downstream channel is n = 0.013. Describe what the flow looks like as it moves through the channel transition and into the long prismatic channel downstream from the transition. Assume that the loss coefficient from the pipe to channel flow is $K_e = 0.2$.



4.163 Water enters a steep rectangular channel with a bottom width $b_1 = 4$ m, a Manning's roughness coefficient $n_1 = 0.013$, and a bottom slope $S_{o1} = 0.023$ from a reservoir whose water surface is $H = 1.8$ m above the channel bottom. Downstream therefrom the channel changes into a trapezoidal channel through a short smooth transition with $b_2 = 4$ m, $m_2 = 1.0$ m, $n_2 = 0.013$, and $S_{o2} = 0.0008$. Do the following: (a) Sketch in and label the possible GVF profiles. (b) Determine whether the hydraulic jump will occur upstream, within, or downstream from the break in grade. (The normal depths for the steep and the mild channels are $Y_{o1} = 1.35$ m and $Y_{o2} = 2.50$ m, respectively, and the upstream channel is sufficiently long for a uniform flow to exist before the break in grade.) (c) Compute the power in kilowatts and horsepower that is dissipated in the hydraulic jump. (d) Using a single step, locate the position of the hydraulic jump.



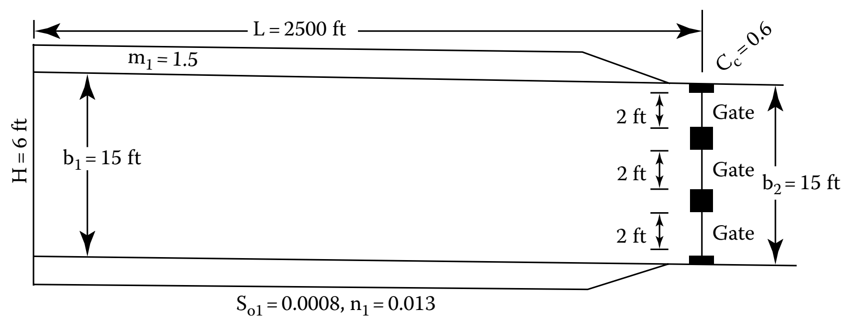
4.164 The sketch below shows a main channel that has a side channel from its side 1000 ft downstream from a reservoir that supplies the main channel with a head $H = 5$ ft. At a distance further downstream in the main channel, two gates exist that are each 4 ft wide. The main channel changes to a rectangular section with a bottom width of $b_4 = 10$ ft through a transition just upstream from the gates. Write out the system of equations that will need to be solved in order to obtain the flow rates in each of the channels and the depths at both their upstream and downstream ends. Solve the problem if the depths of flow downstream from the gates are $Y_{d4} = 1.4$ ft and $Y_{d5} = 1.8$ ft, respectively.



4.165 Solve the five flow rates, and six depths in the previous problem if the two gates have contraction coefficients of $C_c = 0.6$, and have their positions set at $Y_{G1} = 1.0$ ft and $Y_{G2} = 0.8$ ft. Gate 1 discharges Q_4 and gate 2 discharges Q_5.

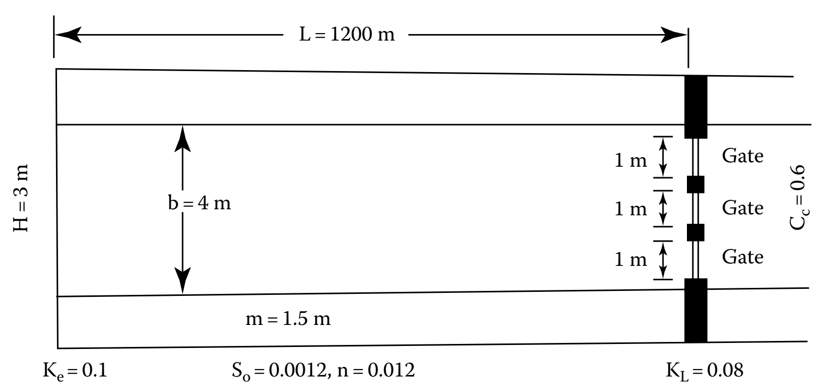
4.166 A trapezoidal channel with $b_1 = 15$ ft, $m_1 = 1.5$, $S_{o1} = 0.0008$, and $n_1 = 0.013$ contains three gates 2500 ft downstream from its beginning. At its beginning, the channel is supplied by a reservoir whose water surface is 6 ft above its bottom. There is a transition to a rectangular channel with $b_2 = 15$ ft just upstream from the gates, and each gate is 2 ft wide and is a vertical gate with a contraction coefficient of $C_c = 0.6$. For the five cases of different gate settings given below, determine the total flow rate in the upstream channel, the flow rate passing each gate, the depth Y_1 at the beginning of the channel, the depth Y_2 at the end of the trapezoidal channel, and the depth Y_3 immediately upstream from the gates after the transition to the rectangular channel.

Case	**1**	**2**	**3**	**4**	**5**
Y_{G1}	0.5	2.0	2.0	1.5	3.0
Y_{G2}	1.5	2.0	1.5	1.5	2.5
Y_{G3}	1.0	2.0	1.0	2.0	3.1



4.167 A trapezoidal channel with a bottom width b = 4 m and side slope m = 1.5 takes water from a lake with a constant water surface elevation 3 m above the channel bottom. The slope of the channel bottom is $S_o = 0.0012$ and its Manning's roughness coefficient is n = 0.012. At a distance 1200 m downstream from the beginning of the channel there are three vertical gates (with $C_c = 0.6$) used to control the flow. Each gate is 1 m wide, and its position above the bottom of the channel is known. Give the equations whose solution gives the flow rates past each gate, as well as solves the other unknowns. The entrance loss at the reservoir is $K_e = 0.1$, and the loss coefficient from the trapezoidal channel to the 4 m wide rectangular channel immediately upstream from the gates is $K_L = 0.08$. For the gate setting given below solve the flow rates, etc.

Case	**1**	**2**	**3**	**4**
Y_{G1}	0.2	0.4	0.6	0.8
Y_{G2}	0.4	0.4	0.4	0.6
Y_{G3}	0.6	0.6	0.7	0.9



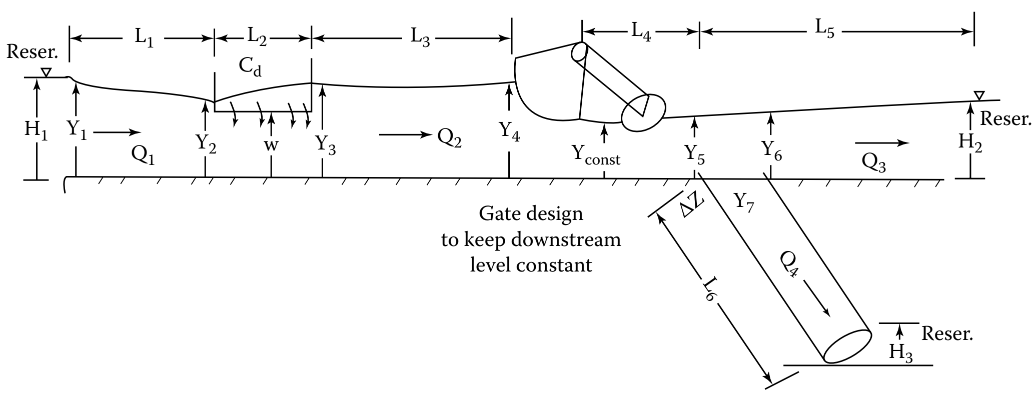
Solution:

Y_{G1}	Y_{G2}	Y_{G3}	Q	q_1	q_2	q_3	Y_{u1}	Y_{d1}	Y_{rect}
0.2	0.4	0.6	6.249	1.053	2.096	3.099	2.997	4.436	4.434
0.4	0.4	0.6	7.272	2.077	2.096	3.099	2.995	4.435	4.428
0.6	0.4	0.7	8.756	3.071	2.096	3.589	2.993	4.433	4.422
0.8	0.6	0.9	11.676	4.033	3.099	4.544	2.988	4.427	4.408

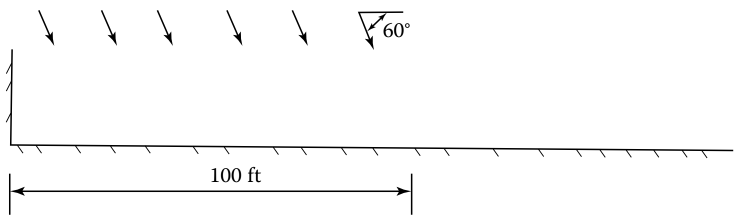
4.168 In Example Problem 4.33 assume that the gate in channel 3 can be set at different positions that causes the depth Y_{3dg} downstream from it to vary. Make up a table showing the depth and the flow rates with Y_{3dg} starting at 1.2 ft and decreasing in increments of 0.1 ft until $Y_{3dg} = 0.1$ ft. Note that as the gate becomes completely closed, $Q_3 = 0$, and therefore the problem no longer involves two channels, but becomes a problem of a channel reducing in size at some position.

4.169 In the sketch below, a channel system is shown that consists of the following elements: (1) The upstream length of the channel has a bottom width b_1, side slope m_1, Manning's coefficient n_1, a bottom slope S_{o1}, and a length L_2. (This section receives water from a reservoir with a water surface elevation H above the channel bottom, and at its end there is a side weir.) (2) A side weir with a length L_2 over which the bottom width and the side slope reduce from b_1 and m_1 to b_2 and m_2, respectively. The weir is w above the channel bottom and it has a discharge coefficient C_d, and this length of channel has properties of n_2 and S_{o2}. (3) A length of channel L_3 between the end of the side weir and upstream from a special "level control" gate maintains a constant depth of water downstream from it. The properties of this third length of channel consist of b_3, m_3, n_3, and S_{o3}. (4) A length of channel L_4 between the gate and a position where a circular channel diverts flow from the side of the channel. The properties of this length of channel are: b_4, m_4, n_4, and S_{o4}. (5) A length of channel from the circular channel diversion of length L_5 that delivers the water into a reservoir whose water surface elevation is maintained constant at H_2. The properties of this channel are b_5, m_5, n_5, and S_{o5}. (6) The circular channel with a length L_6 that delivers its water into a reservoir whose water surface elevation is constant at H_3 above its bottom. The properties of this circular channel are D_6, n_6, and S_{o6}. The beginning of the circular channel is Δz above the main channel.

Set up the system of equations that will solve this problem giving the flow rates in each section, and as well as the depths throughout the channel system.



4.170 The beginning of a trapezoidal channel receives its water supply over a length of 100 ft. The amount of this lateral inflow is 5.0 cfs/ft of length, and this lateral inflow comes in with a velocity of 20 fps and at an angle of 60° from the direction of the channel. Over this lateral inflow section, the channel size changes from $b_1 = 4$ ft and $m_1 = 0.5$ to $b_2 = 12$ ft and $m_2 = 1.5$. The channel is very long, has a bottom slope of $S_o = 0.0006$, and a Manning's $n = 0.013$. Solve the depth through the 100 ft length of inflow.



4.171 Same as the previous problem except that the channel has a bottom slope of $S_o = 0.08$.

4.172 The discussion of the closed-form solution equation for a lateral inflow in rectangular channels referred to comparisons of the results from this theory with results from numerical solutions for different bottom slopes in a rectangular channel. Obtain both the theoretical and the numerical solutions referred to. The problem statement is as follows: A 3 ft wide rectangular channel with $n = 0.013$ receives a constant lateral inflow of $q^* = 1.5\,m^2/s$. Obtain numerical solutions and also solutions based on the closed-form equation for the following cases: (a) the bottom slope of the channel is $S_o = 0.0005$ and the length of the inflow is $L = 10$ m, (b) the bottom slope of the channel is $S_o = 0.25$ and $L = 20$ m, and (c) the bottom slope is $S_o = 0.15$ and $L = 20$ m.

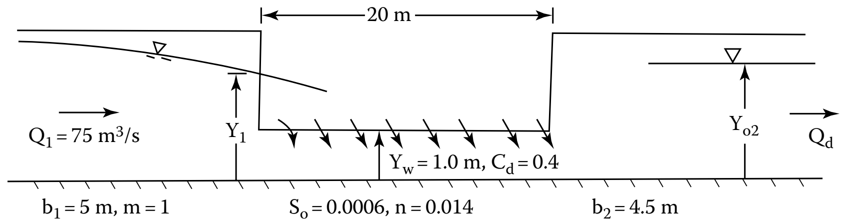
4.173 Simplify program GRATMILD so that is applies only to prismatic rectangular channels, and then solve Example Problem 4.55 with this modified program.

4.174 A 4 ft wide and 2 ft long rack at the bottom of a channel has a discharge coefficient Cd = 0.4, and covers one-half of the bottom area. The channel's bottom slope is $S_o = 0.0005$ and $n = 0.013$. Solve the problem if (a) the rack is in a rectangular channel and the upstream flow rate is 60 cfs; (b) the rack is in a trapezoidal channel with $b = 4$ and $m = 1.5$, and the upstream flow rate is 90 cfs; (c) the rack is in a triangular channel with a side slope $m = 3$, and the upstream flow rate is 60 cfs.

4.175 Repeat the three parts of the previous problem with the exception that the channel over the rack (grate) is not prismatic so the bottom width varies from 4.0 to 3.0 ft over the length of the rack. For the triangular channel, assume m varies from 4.0 to 3.0 over the rack.

4.176 The side weir channel in Example Problem 4.37 is nonprismatic, reducing from an upstream trapezoidal channel with $b = 8$ ft and $m = 1$, to a rectangular channel at the end of the side weir with $b = 6$ ft. Obtain a series of solutions in which the height of the side weir varies from 1.3 to 3.0 ft in increments of 0.1 ft. (As in the example problem, the upstream flow rate is $Q = 600$ cfs, $n = 0.013$, and $So = 0.001$ for all channels, and the discharge coefficient is 0.675.)

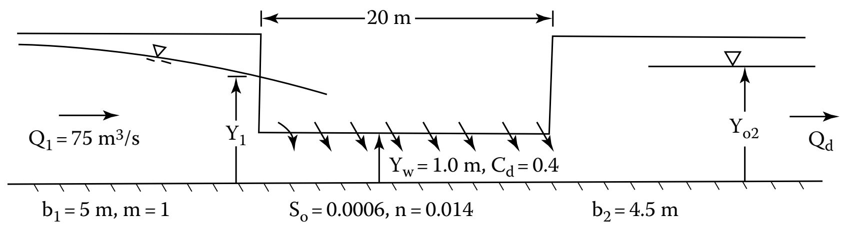
4.177 A side weir with its crest 1 m above the channel bottom has a discharge coefficient $C_d = 0.4$, and is 20 m long. The channel reduces from a trapezoidal channel with $b_1 = 5$ m and $m = 1$ to a rectangular section with $b_2 = 4.5$ m of the length of the side weir. The channel has a bottom slope $S_o = 0.0008$ and $n = 0.014$. Part (a): If the upstream flow rate is $Q_1 = 75\,m^3/s$, analyze what happens to the water depths across the weir length and upstream therefrom? Part (b): If the depth at the end of the side weir were not controlled by downstream conditions, what would the depth at the end of the weir be? Part (c): The downstream rectangular channel now has a width $b_2 = 4$ m. Starting with side weir heights of 2.5 m, obtain solutions decreasing this height by 0.1 m until a solution is not possible. What happens if you try and solve the problem with $Y_w = 2.0$ assuming that the depth at the beginning of the side weir is just below critical ($Y_c = 2.402$ m)?



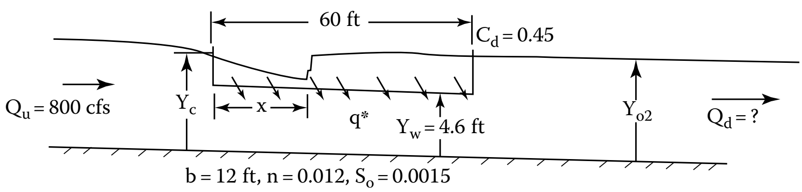
4.178 To get acquainted with Program GVFALL that is designed to solve problems governing a single ODE for GVFs, solve the following problem in prismatic channels. (The next problem deals with nonprismatic channels. (1) Solve the lateral outflow from the side weir in Example Problem 4.39 with a downstream flow rate of $30\,m^3/s$. (b = 5 m, n = 0.016, S_o = 0.0005, $Y_w = 3.1$ m, $C_d = 0.6$, L = 15 m) (2) Solve (1) starting at the upstream end with Q and Y found therefrom. (3) Solve (1) but specify a constant outflow $q_o^* = 0.22\,m^2/s$. (4) Solve (3) but starting from the upstream end. (5) Solve problem (1) but instead of outflow, there is an inflow $q^* = 0.6\,m^2/s$ over the 150 m length and no flow at the beginning. (6) Resolve (5) but start at the upstream end. (7) Solve (1) but rather than a side weir, assume that a grate that exists at the bottom of the channel has the fraction of the bottom open equal to 0.045, and a discharge coefficient $C_d = 0.6$. (8) Resolve (7) but start at the upstream end. (9) Solve problem (1) but instead of a side weir, the outflow is from seepage and equal to q_o^* = $0.1\,m^2/s$. (10) Resolve (9) but start at the upstream end. (11) Solve the M_2-GVF upstream from the side weir in (1) (12) Solve the lateral inflow Example Problem 4.38 (b = 6 ft, m = 1, n = 0.015, $S_o = 0.0008$, with an inflow that varies linearly from $q_1^* = 4$ to $q_2^* = 0$ cfs/ft over the 150 ft length, and an upstream flow rate $Q_1 = 100$ cfs.) (13) Resolve (12) starting at the upstream end. (14) Solve the supercritical flow through the side weir of Example Problem 4.35. (15) A lateral outflow occurs from a circular channel with D = 10 ft, the side weir is at the centerline, i.e., $Y_w = D/2$, and L = 50 ft, with $C_d = 0.4$, $S_o = 0.001$, n = 0.013, and the downstream flow rate is $Q_d = 350$ cfs. (16) Solve the M_2-GVF 1000 ft upstream from the side weir in (15).

4.179 Use Program GVFALL to solve the following problems: (1) Example Problem 4.44, with the side weir in the nonprismatic channel ($b_1 = 10$ ft, $m_1 = 1.2$, $b_2 = 6$ ft, $m_2 = 0$, $S_o = 0.001$, n = 0.013, L = 100 ft, $Y_w = 3$ ft, $C_d = 0.45$, $Q_d = 145$ cfs). (2) Solve (1) starting at the upstream end using Y_1 and Q_1 from (1). (3) Solve the GVF downstream from x_c in Example Problem 4.43. (4) Solve the GVF upstream from x_c in Example Problem 4.45 (EP 45 consists of an inflow with $q^* = 5$ cfs/ft with $U_q = 1$ fps, in a channel with b varying from 4 to 12 ft, and m varying from 0 to 2 over 200 ft, with n = 0.014 and $S_o = 0.035$. $x_c = 155.4$ ft with $Y_c = 4.47$ ft).

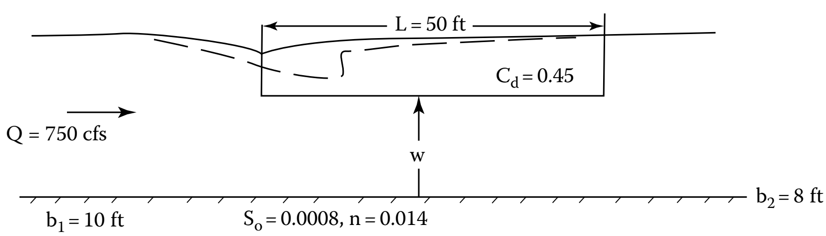
4.180 A flow rate of $Q_u = 800$ cfs occurs in a 12 ft wide rectangular channel upstream from a 60 ft long side weir with a discharge coefficient of $C_d = 0.45$. The channel retains the same width throughout the side weir, and has a Manning's roughness coefficient n = 0.012, and a bottom slope $S_o = 0.0015$. The weir's crest is 4.6 ft above the channel bottom. Assume that the critical depth, or slightly below Y_c, occurs at the beginning of the side weir, and determine the following: (a) the discharge in the downstream channel Q_d, (b) the depth upstream from the hydraulic jump Y_1, (c) the depth downstream from the hydraulic jump Y_2, (d) the normal depth in the downstream channel Y_{o2} (assume that the channel is very long and its flow is not affected by any downstream control), and (e) the position where the hydraulic jump occurs, x.



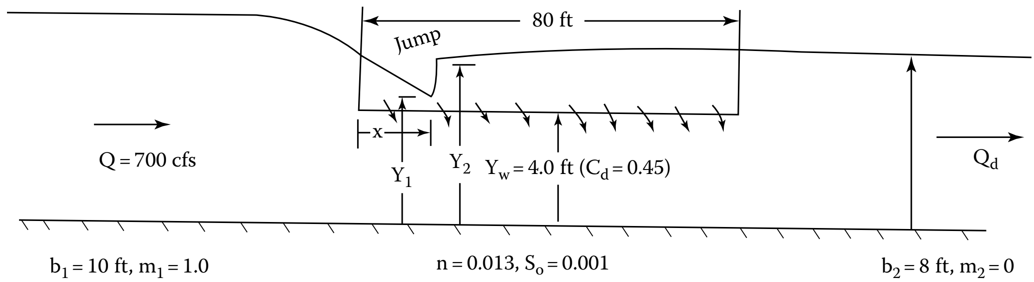
4.181 A rectangular channel contains a 60 ft long side weir whose height is w = 4.65 ft above the channel bottom which has n = 0.013 and $S_o = 0.0008$. At the beginning of the weir, the width of the channel is $b_1 = 12$ ft and at its end the width is $b_2 = 10$ ft. If the flow rate upstream of the side weir is $Q_u = 420$ cfs, determine the flow rate in the downstream channel Q_d, the depth at the beginning of the side weir Y_1, and the depth at the end of the side weir Y_{o2} if the channel is very long. The discharge coefficient for the side weir is $C_d = 0.45$.



4.182 A 50 ft long side weir is to be placed in a rectangular channel with an upstream width of $b_1 = 10$ ft and a downstream width of $b_2 = 8$ ft. The channel has a bottom slope $S_o = 0.0008$ and n = 0.014. The flow in the upstream channel is Q = 750 cfs. The weir has a discharge coefficient $C_d = 0.45$. Determine the depths across the lateral outflow length of the channel for weir heights ranging from 8 ft above the channel bottom to 4.6 ft above the channel bottom.



4.183 A side weir that is 4.0 ft high and 80 ft long exists on the side of a channel whose size changes from $b_1 = 10$ ft and $m_1 = 1.0$ to a rectangular section with $b_2 = 8$ ft, as shown in the sketch. If the flow rate upstream is 700 cfs, determine the flow rate and the depth in the downstream channel, the depth upstream and downstream from the hydraulic jump, and the position x of the hydraulic jump. Assume that the depth passes just below the critical at the beginning of the side weir. The discharge coefficient for the weir is $C_d = 0.45$.



4.184 In the previous problem, there is a gate at the distance 1000 ft downstream from the end of the side weir whose setting produces a depth of Y_{Gd} = 1.6 ft. Now, determine the depths and flow rates, etc. Investigate what happens as the gate is lowered to positions to cause Y_{Gd} = 1.6, 1.55, 1.5, etc.

4.185 A flow rate of Q_u=20 m^3/s occurs in a 3 m wide rectangular channel upstream from an 18 m long side weir with a discharge coefficient C_d = 0.45. The channel retains the same width through the side weir, and has a Manning's n = 0.012 and a bottom slope S_o = 0.0015. Investigate the different flow conditions that will occur across the length of the side weir as its height varies. In this investigation, recognize that three cases can occur as described in this chapter.

4.186 In the Problem 4.183, shorten the length of the side weir from 80 to 50 ft, and raise the height of the side weir from 4.0 to 4.1 ft, and then obtain a solution for the gate set so that it produces depths downstream from it of Y_{Gd} = 1.2, 1.1, 1.0, 0.9, 0.8, 0.7, 0.6, and 0.5 ft, respectively. For these solutions, the incoming flow rate is still Q_u = 700 cfs, the upstream channel is trapezoidal with b_1 = 10 ft, m_1 = 1, and after the 50 ft long side weir, the channel is rectangular with b_2 = 8 ft. The gate is 1000 ft downstream from the end of the latter outflow, and the discharge coefficient for the side weir is C_d = 0.45, and for all channels n = 0.013 and S_o = 0.001.

4.187 A triangular gutter with a side slope m = 5, n = 0.014, and a bottom slope S_o = 0.0015 is 150 m long. The grates at both ends of this gutter readily taken in all the flow accumulated in the gutter. If the lateral inflow to the gutter is $q^* = 1.0 \times 10^{-3}$ m^2/s determine (1) the flow rate entering the upstream grate and the depth here, (2) the flow rate entering the downstream grate and the depth here, (3) the position where the flow separates from moving upstream to downstream and the depth here.

4.188 Resolve the previous problem if the gutter is horizontal, S_o = 0. Thereafter, obtain a series of solutions to determine how Y_{cl}, Y_{cr}, X_s, and Y_s change with S_o in this channel ($q^* = 1.0 \times 10^{-3}$ m^2/s.)

4.189 Resolve the previous problem, but keep the bottom slope S_o = 0.0015 constant, and vary the lateral inflow between $q^* = 1.0 \times 10^{-2}$ m^2/s to $q^* = 1.0 \times 10^{-4}$ m^2/s.

4.190 A triangular gutter with a vertical side and the other side with a slope m = 3, that is 1000 ft long, has a Manning's n = 0.021 and a bottom slope S_o = 0.002 that receives a lateral inflow of q^* = 0.01 cfs/ft. The drains at both ends of this gutter readily receive all the inflow. Determine the depths at both ends of the gutter, the flow rates at these ends, and the position and the depth that divides the flow from moving up slope to down slope.

4.191 Determine the length of grate necessary to receive the flow from the gutter of the previous problem if one-half of the area of the bottom of the b = 3 ft wide grate is open, and its discharge coefficient is C_d = 0.45.

4.192 Since an explicit equation gives the critical depth in a triangular gutter, it is possible to eliminate Equations 4.21a and b from the system of simultaneous equations that must be solved so that Equations 4.21c and d can be used to solve Y_s and X_s when solving problems of lateral inflow into gutters with critical depths at both ends. Modify program GUTTER so that only two equations are solved, and then use this program to solve Example Problem 4.52.

4.193 Modify the program GUTTER so that the lateral inflow q* can vary as a function of x across the gutter. Then use this program to solve the 15 problems with the lateral inflows given in the table below. The gutter is triangular with a side slope m = 4 and is L = 280 m long. Manning's n = 0.013 for this gutter, and it has a bottom slope of S_o = 0.0009. The coefficients in the table are for the equation $q^* = a_o x + a_1$, in which x begins at the upstream (left) end of the gutter.

	Coefficients		
Prob. No.	a_o	a_1	Explanation
1	4.0×10^{-4}	0	A constant inflow that result in a total Q = 0.112 m³/s
2	6.0×10^{-4}	0	A constant inflow that result in a total Q = 0.168 m³/s
3	8.0×10^{-4}	0	A constant inflow that result in a total Q = 0.224 m³/s
4	2.0×10^{-4}	1.42857×10^{-6}	q* at beg. 1/2 of No. 1 with increase with x so Q = 0.112 m³/s
5	0	2.85714×10^{-6}	q* at beg. 0, with increase with x so Q = 0.112 m³/s
6	6.0×10^{-4}	-1.42857×10^{-6}	q* at beg. 0.0006, with decrease with x so Q = 0.112 m³/s
7	8.0×10^{-4}	-2.85714×10^{-6}	q* at beg. 0.0008, with decrease with x so Q = 0.112 m³/s
8	3.0×10^{-4}	2.142857×10^{-6}	q* at beg. 1/2 of No. 2 with increase with x so Q = 0.168 m³/s
9	0	4.28571×10^{-6}	q* at beg. 0, with increase with x so Q = 0.168 m³/s
10	9.0×10^{-4}	-2.14286×10^{-6}	q* at beg. 0.0009, with decrease with x so Q = 0.168 m³/s
11	12.0×10^{-4}	-4.28571×10^{-6}	q* at beg. 0.0012, with decrease with x so Q = 0.168 m³/s
12	4.0×10^{-4}	2.857143×10^{-6}	q* at beg. 1/2 of No. 3 with increase with x so Q = 0.224 m³/s
13	0	5.714286×10^{-6}	q* at beg. 0, with increase with x so Q = 0.224 m³/s
14	12.0×10^{-4}	-2.85714×10^{-6}	q* at beg. 0.0012, with decrease with x so Q = 0.224 m³/s
15	16.0×10^{-4}	-5.71429×10^{-6}	q* at beg. 0.0016, with decrease with x so Q = 0.224 m³/s

4.194 Obtain a series of solutions in which the lateral inflow q* changes what enters a triangular gutter which has long grates at both its ends. The gutter has a side slope m = 4, is L = 280 m long, with Manning's n = 0.013, and a bottom slope S_o = 0.0009. Use this series of solutions to determine how x_s, Y_s, Y_1, and Y_r vary with q* for this channel. Also give the total inflow into the gutter in this table of solutions.

4.195 Generate a series of solutions for the same gutter as in the previous problem (i.e., L = 280 m, n = 0.013), except have its bottom slope vary from 0.0018 to 0.0001, and for each of these bottom slopes let the lateral inflow vary as in the previous problem. Plot the results of these solutions on four graphs, each of which contains a series of lines for S_o and has the lateral inflow q* plotted along its abscissa. The ordinates of the four separate graphs should be (1) the position where the reverse flow x_s occurs, (2) the depth Y_s at this position, (3) the critical depth at the beginning of the gutter Y_1, and (4) the critical depth at the end of the gutter Y_r.

4.196 Find the lengths of grates needed to discharge the flow from each of the lateral inflows used in the previous problem under the assumption that the flow through the grates is supercritical. The grates have a bottom width of b = 0.3 m, and have the same n and bottom slope as the gutter, i.e., n = 0.013 and S_o = 0.0009, and the side slope beyond the flat bottom is m = 4.

4.197 Program GRATMILD, that is designed to solve the upstream depth, the downstream depth, and the downstream flow rate, given the upstream flow rate entering a grate call on the subroutine RUKUST as the ODE-Solver. Program GRATMILD assumes that the flow is subcritical across the entire grate. Modify this program so that it uses DVERK, or ODESOL, as the solver. Use your modified program to solve the following problem with the flow rate coming into the upstream end of the grate varying from $Q_l = 1.1\,m^3/s$ to $0.5\,m^3/s$. The grate has a $b = 0.3\,m$, a Manning's $n = 0.012$, and a bottom slope $S_o = 0.0009$. One-half of the bottom width is open and the discharge coefficient is $C_d = 0.45$. The grate is $L = 0.5\,m$ long.

4.198 The program GUTTER solves four equations simultaneously for four unknown variables. For a triangular channel, the first two equations, i.e., the critical flow equations at the beginning (left), and at the end (right) of the gutter are explicit and can be solved to provide these critical depths based on any value of x_s. Modify the program you developed for Problem 4.192 that allows for the lateral inflow q* to be a function of the position x, but solves only two equations simultaneously. Use this program to solve the problems of that problem.

4.199 Determine the depths at the beginning and at the end of a 1 m long grate that has one-half its 0.5 m wide bottom open and this opening has a discharge coefficient $C_d = 0.45$ for a series of incoming flow rates Q_{in} from $0.5\,m^3/s$ to smaller values until the flow across the grate approaches critical conditions. The side slope of the grate beyond its bottom width of $b = 0.5\,m$ is $m = 4$, Manning's $n = 0.013$, and the bottom slope of the grate is $S_o = 0.0009$.

4.200 Modify program GUTTER so that rather than having the lateral inflow q* into the gutter equal to a constant value, have this lateral inflow be a linear function of the position x along the gutter according to $q^* = q^*_b + x(q^*_e - q^*_b)/L$, in which q^*_b is the lateral inflow at $x = 0$ (the beginning of the gutter), and q^*_e is the lateral inflow at $x = L$ (at the end of the gutter). (Notice that if $q^*_e = q^*_b$, the case of q* equal constant is specified.) Then solve Example Problem 4.52 for the following two cases: (a) $q^*_b = 0.005$ cfs/ft and $q^*_e = 0.02$ cfs/ft, and (b) $q^*_b = 0.02$ cfs/ft and $q^*_e = 0.005$ cfs/ft. ($L = 800$ ft, $m = 4$, $n = 0.013$, $S_o = 0.001$) For these two cases, solve the problem if the gutter is horizontal, $S_o = 0$.

4.201 Modify program GUTTER so that the cross section of the gutter is trapezoidal rather than triangular. With this modified program solve Example Problem 4.52 if the bottom width of the gutter is $b = 2$ ft and the side slope is $m = 1$. Since the gutter is larger, also increase the lateral inflow q* to 0.1 cfs/ft and resolve Problem 4.52.

4.202 If the bottom rack (grate) of Example Problem 4.55 is 1/2 a triangle rather than a rectangular channel, determine the depth at it entrance, the depth at its end and the flow rate leaving the grate, if the flow rate entering the grate is the same flow rate of $Q_{in} = 60$ cfs as in the example problem ($b = 4$ ft, $n = 0.013$, $S_o = 0.0004$, fraction = 0.5, and $C_d = 0.40$, $L = 2$ ft).

4.203 A trapezoidal channel with a side slope of $m = 1$, a bottom width of $b = 1.2\,m$, and a length of 0.7 m contains a grate across its bottom with an opening equal to one-half the width and a discharge coefficient $C_d = 0.45$. The channel has a bottom slope $S_o = 0.0004$ and $n = 0.013$. The flow rate at the beginning of the grate is $Q_{in} = 2.8\,m^3/s$. Determine the flow rate leaving at the end of the grate, and the depths at its beginning and its end.

4.204 Determine the length of grate required to discharge a flow rate of $Q_r = 0.2\,m^3/s$ coming from the right side of a triangular gutter with $m = 4$, $n = 0.013$, and $Q_l = 0.055\,m^3/s$ coming from the left side of a gutter. The bottom slope of the grate (and gutter) is $S_o = 0.001$ and it has a bottom width $b = 1.2\,m$. One-half of the area of the bottom is open and the discharge coefficient is $C_d = 0.4$. Resolve the length of grate required assuming that $Q_l = 0$, and the total flow rate is coming from the right side of the gutter $Q_r = 0.25$

4.205 Obtain a series of solutions to determine how the depths at the beginning and at the end of a 0.5 m long grate vary with the incoming flow rate Q_{in}. No flow exists in the channel downstream from the grate, i.e., $Q_{out} = 0$. The grate is 1.5 m wide, has one-quarter of the area on the bottom open, and its discharge coefficient is $C_d = 0.4$. It is in a triangular gutter with $m = 4$ and $S_o = 0$. The bottom is flat and assume $n = 0.014$. Start this series of solutions for a depth just above that which causes F_r to equal unity.

4.206 Repeat Example Problem 4.56, except rather than specifying $Q_{out} = 0$, use a flow rate of $Q_{out} = 1$ cfs passing the end of the grate into the downstream gutter.

4.207 Modify program GUTTER4T so that it solves six unknowns rather than four. In other words, add variables x_m and Y_m (the position and the depth in the grate where Q = 0) to the list of unknowns. Use your modified program to solve Example Problem 4.57.

4.208 Modify program GUTTER4T so that it accounts for the lateral inflow over the length of grate. Assume this inflow equals that over the gutter's length. With this modified program, solve Example Problem 4.57 with a bottom slope of $S_o = 0.00005$, and compare the results with those when $q^* = 0$ over the grate's length.

4.209 Obtain a series of solutions to the same inflow to a triangular gutter–grate system with no flow leaving the end of the grate, as in Example Problem 4.63, except have the bottom slope S_o vary starting with $(S_o)_{beg} = 0.0002$ and go to $(S_o)_{end} = 0.0018$ in increments of $\Delta S_o = 0.0001$, i.e., obtain 17 solutions. The specifications for this gutter–grate system are n = 0.013, m = 4, length of gutter receiving inflow is 800 ft, length of grate that discharges flow is 1 ft, and its width is 4 ft, with one-half of its bottom open, and a discharge coefficient $C_d = 0.45$. The latter inflow over the gutter length is $q^* = 0.011$ cfs/ft. The inflow $Q_{in} = 0$ cfs at the beginning of the gutter.

4.210 Obtain a series of solutions to the same inflow to a triangular gutter–grate system with no flow leaving the end of the grate, as in Example Problem 4.63, except have Manning's n vary starting with $n_{beg} = 0.012$ and go to $n_{end} = 0.031$ in increments of $\Delta n = 0.001$, i.e., obtain 20 solutions. The specifications for this gutter–grate system are $S_o = 0.00026$, m = 4, length of gutter receiving inflow is 800 ft, length of grate that discharges flow is 1 ft, and its width is 4 ft, with one-half its bottom open, and a discharge coefficient $C_d = 0.45$. The latter inflow over the gutter length is $q^* = 0.011$ cfs/ft. The inflow $Q_{in} = 0$ cfs at the beginning of the gutter.

4.211 In solving Example Problem 4.63, and other example problems involving flow out from grates the cross-sectional area of the grate is defined as $A = 0.5mY^2$ where m is the side slope and applies for both the upstream gutter as well as the grate. Yet, the discharge through the grate is defined by the orifice flow, i.e., $q_{out} = C_d b(f)Y^{.5}$, where b is the given width for the grate, which for Example Problem 4.63 was specified as 4 ft. In other words, the discharge coefficient must account for a varying depth at any position x along the grate. Modify Program GUTGRAT1 so that the cross-sectional area along the grate is defined by a rectangular portion with a width b that contains the opening, and a triangular side attached to this with a side slope m_g, which may be different than the side slope m for the upstream gutter. With this program solve the same problem as given in Example Problem 4.63 with b = 4 and $m_g = 1$.

4.212 Obtain a series of solutions with the program you developed in the previous problem (or modify it with an additional DO loop) so that the side slope m of the upstream gutter varies. Obtain two such series of solutions; the first in which m starts with 4 and decreases to 2.7 in increments of −0.1, and the second in which m again starts with 4 and increases to 5 in increments of +0.1. The rest of the specifications are as in the previous problem (or Example Problem 4.63), e.g., n = 0.013, $S_o = 0.00026$, L = 800 ft, $L_G = 1$ ft, $Q_{in} = 0$, $q^* = 0.011$ cfs, and the width of the grate is b = 1 ft, with a side slope of $m_g = 1$.

4.213 For the gutter–grate system of Example Problem 4.63, investigate how the following variables affect the depths at the beginning of the gutter, at its end, and at the end of the grate: b, m_g, Q_{in}, C_d, and q^*. The properties of the original gutter–grate system are L = 800 ft, $L_{grate} = 1$ ft, $S_o = 0.00026$, m = 4 (for upstream triangular gutter), n = 0.013, b = 4 ft, $m_g = 1$, $Q_{in} = 0$, f = 0.5, and $C_d = 0.45$.

4.214 Program GUTGRAT1, which is used to solve Example Problem 4.63, assumes that the lateral inflow q^* occurs only over the length of the gutter, and not over the length of the grate. In other words, the assumption is that the outflow from the grate includes the actual outflow plus any lateral inflow that may be occurring over its length. (The same assumption is made for the other program dealing with gutter–grate problems.) Modify this program so that the same lateral inflow q^* continues over the length of the grate, and use your modified program to resolve Example Problem 4.63.

4.215 The lateral inflow is a linear function of the distance from the beginning of a gutter–grate in which no flow leaves the end of the grate. The problem is the same as Example Problem 4.63 with the exception that the lateral inflow is given by $q^* = 0.011 + 0.00001x$, where x is the position from the beginning of the gutter. This lateral inflow continues over the length of the grate, as well as the gutter. (The problem specifications are $L = 800\,ft$, $L_G = 1\,ft$, $n = 0.013$, $S_o = 0.00026$, $m = 4$, $b = 4\,ft$, $f = 0.5$, $C_d = 0.45$)

4.216 A triangular gutter that is 150 m long with $n = 0.014$, $m = 5$, and $S_o = 0.0001$ discharges into a grate that is $L_G = 0.4\,m$ long with a width $b = 1.2\,m$, with one-quarter of its bottom open and a discharge coefficient $C_d = 0.4$. Start with a lateral inflow of $q^* = 0.001\,m^2/s$ and an inflow at the beginning of the gutter of $Q_{in} = 0.05\,m^3/s$, and solve the depth at the beginning of the gutter, the depth between the gutter and the grate, and the depth at the end of the grate. The channel terminates at the end of the grate. Then obtain a series of solutions in which the inflow Q_{in} varies to 0 and then to $0.1\,m^3/s$, and the lateral inflow equals $q^* = 0.001$, 0.0012, 0.0015, and $0.002\,m^2/s$.

4.217 Assume the gutter–grate system in the previous problem consists of an infinite series so that the depth at the end of each grate is the depth at the beginning of the next gutter, and therefore the flow rate past the end of the grate is the flow rate into the next gutter, etc. The lateral inflow over the 150 m length of each gutter is $q^* = 0.001\,m^2/s$, the bottom slope is $S_o = 0.0001$, $m = 5$, and $n = 0.014$. The grates have a length $L_G = 0.4\,m$, a bottom width $b = 1.2\,m$, with $f = 0.25$ of its area open, and a discharge coefficient $C_d = 0.4$. Solve Y_l, Y_r, and $Q_{out} = Q_{in}$. How do these variables change if $f = 0.24$, 0.22, and 0.2?

4.218 Program GUTGRTN.FOR has considerable logic in it so that extra computations do not take place in evaluating elements of the Jacobian matrix that are zero. This logic, etc., results in making the program more computational efficient. Modify program GUTGRTN so that it uses the usual statements in evaluating the Jacobian matrix numerically by reevaluating all equations each time one of the unknowns is incremented and then evaluates the partial derivatives with respect to that unknown as $\partial F_i/\partial x = (FF_i - F_i)/\Delta x$.

4.219 Program GUTGRTN calls on a standard linear algebra solver SOLVEQ that does not take advantage of the fact that the Jacobian matrix is banded with most of the elements equal to zero. Modify program GUTGRTN so that the solution to the linear algebra needed to obtain the correction for the Newton method takes advantage of the sparseness of the Jacobian matrix. Verify that your program works by solving Example Problem 4.64.

4.220 Example Problem 4.64 solves the depths, etc., that result from having three gutter–grates in series. Increase the number of these gutter–grates to 5, and solve the same problem.

4.221 Four triangular gutter–grates, all with Manning's $n = 0.014$, side slopes $m = 5$, and bottom slope $S_o = 0.0001$, have a lateral inflow $q^* = 0.001\,m^2/s$ over the length of the gutters, which are $L = 150m$ long. The grates are all $L_G = 0.4\,m$ long. Solve the depths throughout the series of four gutter–grates, and the flow rates past the grate into the next gutter if the width of the grates are $b = 1.2\,m$, one-quarter of their bottom areas are open, and their discharge coefficients are $C_d = 0.45$. No flow passes the last, and the fourth grate. Repeat the solution with the lateral inflow increased to $q^* = 0.0012\,m^2/s$.

4.222 Resolve the previous problem with the lengths of the gutters being as follows: $L_1 = 180\,m$, $L_2 = 150\,m$, $L_3 = 130\,m$, $L_4 = 110\,m$. Obtain the solution for $q^* = 0.001\,m^2/s$ and $q^* = 0.0012\,m^2/s$. Also resolve the previous problem keeping the lengths of the gutters the same with $L = 150\,m$, but change the bottom slopes as follows: $S_{o1} = 0$, $S_{o2} = 0.0001$, $S_{o3} = 0.0005$, $S_{o4} = 0.001$.

4.223 Solve Example Problem 4.64 with the lengths of the three gutters as follows: $L_1 = 900\,ft$, $L_2 = 800\,ft$, and $L_3 = 700\,ft$. For this solution, the length of all of the grates are $L_G = 1.0\,ft$, and the other variables are as in Problem 4.64. Then obtain a second solution in which all of the gutter lengths are $L = 800\,ft$, but the lengths of the three grates are $L_{G1} = 0.9\,ft$, $L_{G2} = 1.\,ft$, and $L_{G3} = 1.1\,ft$.

4.224 Obtain a series of solutions for the three gutter–grate system of Example Problem 4.64 with a bottom slope $S_o = 0.0003$ in which the lateral inflows into the gutters vary as given below. This gutter–grate system consists of three 800 ft long triangular gutters followed by three 1 ft long grates, all with a side slope m = 4, n = 0.013, and S_o = 0.0003. The width of the grates is b = 4 ft, and they have one-half their bottoms open, with a discharge coefficient C_d = 0.45. Start the series of solutions with q1* = 0.012 cfs/ft and q2* = q3* = 0.011 cfs/ft. Then increase q1* to .02 cfs/ft in small increments with q2* = q3* = 0.011 cfs/ft held constant. Next decrease q2* = q3* in small increments until they equal .088 cfs/ft with q1* = 0.02 cfs/ft held constant. Finally decrease q1* in small increments with q2* = q3* = 0.008 cfs/ft held constant until the solution process fails. To solve this problem you might consider modifying program GUTGRTN so that it will repeat a new solution when requested with different values for the lateral inflow.

4.225 Three triangular gutter–grates exist in series. All gutters have a length L = 800 ft, and all grates have a length of 1.0 ft. The side slopes of all gutter–grates is m = 4, and their Manning's n = 0.012. One-half of the bottom of the grates is open and their discharge coefficients are all C_d = 0.45. The bottom slopes of the gutter–grates vary according to S_{o1} = 0.0003, S_{o2} = 0.0005, and S_{o3} = 0.0008. Solve the depths throughout this system and the flow rates passed beyond the first grates into their downstream gutters if the lateral inflow into gutters 1 and 2 is 0.012 cfs/ft and into gutter 3 is 0.011 cfs/ft.

4.226 Obtain a series of solutions for a five gutter–grate system that has bottom slopes as follows: S_{01} = 0.0003, S_{02} = 0.0005, S_{03} = 0.0008, S_{04} = 0.001, S_{05} = 0.001, in which the lateral inflows into the gutters vary as given below. This gutter–grate system consists of three 800 ft long triangular gutters followed by three 1 ft long grates, and the last two gutters are 900 ft long with grates that are 1 ft long. All gutters and grates have a side slope m = 4 and n = 0.013. The width of the grates is b = 4 ft, and the fractions of the bottoms that are open are f_1 = 0.4, f_2 = 0.4, f_3 = 0.45, f_4 = 0.5, and f_5 = 0.5. All of the grates have a discharge coefficient C_d = 0.45. Start the series of solutions with the lateral inflow into gutters 1, 2, and 3 equal to 0.012 cfs/ft, and that into gutters 4 and 5 equal to 0.011 cfs/ft. Then increase the lateral inflow into gutters 1 and 2 (with q* into gutters 3, 4, and 5 constant as above) in increments of 0.005 cfs/ft until q1* = q2* = 0.02 cfs/ft.

4.227 The program GUTGRTN used to solve Example Problem 4.64 (and other problems) assumes that the discharge from the grate is the actual discharge minus any that inflow, or in other words, the lateral inflow over the gutter does not continue over the length of the grate. Modify program GUTGRTN so that the specified lateral inflow q^*_i for gutter i continues to occur over the length of grate i. With this modified program resolve Example Problem 4.64.

4.228 In the previous problem you modified computer program GUTGRTN so that the lateral inflow q_i^* into gutter i continued over the length of the grate that follows this gutter. Resolve Problem 4.225 using this modified program.

4.229 A series of six gutters followed by grates occur with the lengths and the characteristics given in the table below. Obtain a solution to this system of gutter and grates that gives the depths throughout the system, and the flow rate passing beyond each grate. All gutters are triangular with a side slope m = 4, and have a Manning's n = 0.012 (lengths are in ft, and q* in cfs/ft).

No.	S_o	L (Gutter)	L_G (Grate)	q* (Gutter)	b (Grate)	Fraction	C_d
1	0.0001	700	0.3	0.012	4	0.5	0.45
2	0.0003	800	1.3	0.012	4	0.45	0.45
3	0.0003	900	1.5	0.014	4	0.4	0.45
4	0.0003	1000	1.7	0.015	4	0.4	0.45
5	0.0003	1100	1.7	0.016	4	0.4	0.45
6	0.0003	1200	1.2	0.017	4	0.4	0.45

4.230 The table below gives the lengths of a series of five gutters, which are followed by five intake grates. The bottom slope of the entire gutter–grate system is S_o = 0.0001. All five grates have a flat bottom that is 1.2 m wide and all grates are 0.5 m long, and the portion of this bottom which is open for outgoing flow is one-quarter. The side slope m of the gutters (as well as the grates beyond the flat bottom) is m = 4.5. The discharge coefficient for all grates is $C_d = 0.40$. Solve the depths and flow rates across this system if the flow past the end of the last grate and the lateral inflow to all gutters is $q^* = 0.0012\,m^2/s$.

No.	**1**	**2**	**3**	**4**	**5**
L (m)	200	180	160	140	140

4.231 Add an additional gutter and grate to the end of the system of the previous problem. This final gutter has a length of 140 m.

4.232 A system of six gutter–grates exist with the lengths in the table below.

Manning's n for all gutter–grates is 0.013, their bottom slopes $S_o = 0.0009$, and the side slope for the system is m = 4. The grates have a flat bottom that is 0.5 m wide with 40% of this area open and their discharge coefficients are C_d.= 0.3 (the bars are normal to the direction of flow). If the lateral inflow into the grates is $q^* = 0.0012\,m^2/s$, find the depths and flow rates throughout this system.

No.	**1**	**2**	**3**	**4**	**5**	**6**
L (gutter)	280	250	200	180	160	140
L (grate)	0.8	0.8	0.9	1.0	1.0	1.0

Also solve the system of gutter–grates if the lengths of the last three gutters is 200 m. Then solve the problem if all of the gutters have a length of 280 m, and all the grates have a length of 1.0 m. From this last solution, some of the gutters with longer depths will continue to increase in the downstream direction, therefore, obtain a solution to this series in which the length of the last four grates is increased to 1.2 m. As a final solution to this system, increase the lengths of the first two grates to 1.2 m, also so that all grate lengths are 1.2 m.

4.233 Add additional cases to the previous problem in which the length of the grates increase so that eventually they all have a length of 1.4 m, except grate 1, and make its length as long as possible to still have a subcritical flow through the system of six gutter–grates. What happens when the depth at the end of any grate (or the beginning of the following gutter) approaches the critical depth?

4.234 Write a computer program to solve a series of gutter–grates in which the critical depth occurs at the end of the first gutter, i.e., the intake capacity of the downstream grates is such that the critical flow occurs at the beginning of grate1, but the subcritical flow occurs throughout the rest of the downstream gutter–grates. Use this program to solve the six series of gutter–grates of the last two problems in which all gutters have a length of 280 m, and all grates have a length of 1.4 m. Then use this program to solve additional cases in which $(L_G)_1$= 1.5, 1.6, and 1.7 m.

4.235 Program GUTGRTN.FOR and the other program designed for solving a series of gutter–grates that use other ODE solvers, generate the Jacobian matrix by utilizing the fact that many of the elements are 0, and others are known as described in the text. Modify program GUTGRTN so that this modified program generates all the elements of the Jacobian matrix by obtain two solutions of the equation vector; the first without any unknown incremented, and the second with each of the n unknowns incremented, as successive row of this matrix are evaluated numerically. Then use the program to obtain solution(s) to one or more of the problems involving the series of gutter–grates that you previously solved.

4.236 Obtain several solutions to the system of six gutter–grates of the previous problem in which the length of the last grate is lengthened over the previous solution. Why do you fail to obtain a solution when the final grate gets too long?

4.237 Another gutter–grate combination is attached to the end of the five gutter–grates given in Problem 4.226, so that six combinations exist. This sixth gutter has a length L = 600 ft, followed by a grate with a length $L_G = 0.6$ ft. The bottom slope over this last gutter–grate equals $S_o = 0.0006$ and the side slope of the gutter and grate are m = 4, as for the other combinations. The discharge coefficient is $C_d = 0.45$ and the fraction of the 4 ft wide opening at the bottom of the grate is 0.45. Solve the depths throughout this system, and the discharge from its final grate.

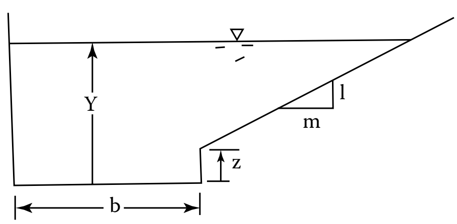
4.238 Develop a computer program that will solve the problem in which the last grate of a series of gutter–grates is long enough that it will receive all of the flow at the end of the last gutter, i.e., critical conditions exist at the end of the last gutter as the flow enters the last grate. Then solve the six gutter–grates of Problem 4.232, except that the last grate is long enough to discharge all the flow entering it under supercritical conditions. Also solve this system of gutter–grates, if grates 4 and 5 have their lengths reduced from 1.7 to 1.6 ft.

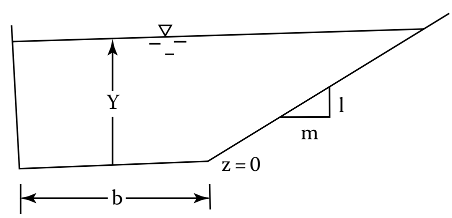
4.239 Develop a program to solve the position of the hydraulic jump, the depth upstream and downstream therefrom, and the depth at the end of the final grate of the previous problem, if this grate is 3 ft long. Then solve this problem, i.e., a flow rate of $Q_u = 27.551$ cfs, the bottom slope $S_o = 0.0003$, n = 0.012, the fraction of opening = 0.5, and $C_d = 0.45$. This grate is triangular with m = 4, and it is assumed that the discharge occurs over a width of 4 ft across the bottom of the grate.

4.240 In Example Problem 4.55, assume that there is zero flow leaving the end of the grate, but the inflow is still $Q_{in} = 60$ cfs. Now determine the depths at the beginning and at the end of the rectangular channel that contains a grate in its bottom that has one-half the width open and a discharge coefficient, $C_e = 0.4$. The grate is also 2 ft long, and the channel has a bottom width b = 4 ft, a bottom slope of $S_o = 0.0005$, and n = 0.013 as in the example problem.

4.241 Solve Problem 4.203 assuming that a zero flow leaves the end of the grate, i.e., solve the depths at the beginning and at the end of the grate if the inflow is $Q_{in} = 2.8\,m^3/s$ but $Q_{out} = 0$. (The channel is now rectangular with b = 1.2 m, n = 0.013, $S_o = 0.0004$, fraction opening = 0.5, $C_d = 0.45$, and L = 0.7 m.) Instead of a grate atn the bottom of this channel, assume that there is a side weir with a discharge coefficient of $C_d = 0.42$. Obtain several solutions in which the height of this side weir is changed. The flow rate at the beginning of the outflow section is $Q_{in} = 2.8\,m^3/s$. Identify the type of GVF upstream for each of these solution if the upstream channel has the same size and slope, etc., through the weir section.

4.242 Modify program GUTTER so that the gutter can have a flat bottom with a width b and a rise z as shown on the sketch. Notice that if both b and z are zero, the shape of the gutter is triangular as handled by GUTTER, and if z is zero, then the shape of the gutter becomes a flat bottom gutter as shown on the second sketch below. With this program, solve the depths at the beginning and at the end, as well as the position where the flow separates from moving toward the left to the right and the depth at this position, of an 800 ft long gutter with b = 2 ft, m = 4, z = 0, n = 0.013, $S_o = 0.001$, if the lateral inflow is q* = 0.011 cfs/ft. Assume the grates (storm drain inlets) at the ends of the gutter can readily accept all the inflow. What are the depths if the gutter is horizontal?





4.243 For Homework Problem 4.192 you are to take advantage of the fact that explicit equations give the critical depth in triangular gutters, and thus reduce to two the number of unknown that need to be solved in program GUTTER. Modify the program you wrote for the previous problem that handles a gutter with a flat bottom with a width b and a vertical rise z on one side, so that it only solves two equations simultaneously for the two unknowns, Y_s and X_s.

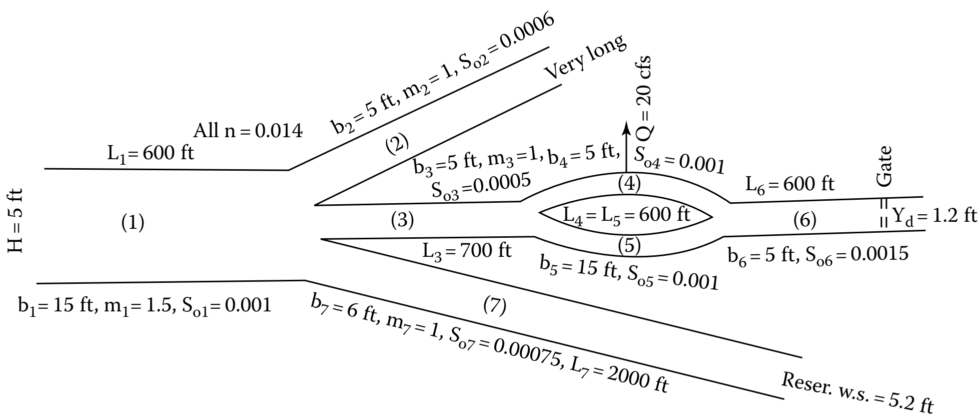
Then use this program to solve the problem of flow in a gutter with b = 2 ft, m = 4, n = 0.013, S_o = 0.001 that is 800 ft long and has a lateral inflow q* = 0.011 cfs/ft. (In writing the program write the special solver of the linear 2 × 2 system of equations. Also note that the first column of the Jacobian matrix contains a 1 in both rows 1 and 2, and thus the Jacobian can be solved as a one-dimensional array.

4.244 Since the ODE for the lateral inflow can be solved continuously from the left end of the gutter, where the critical depth occurs, to the right end, where the flow is again critical, the depth Y_s where the flow separates from moving from the left to the right can be eliminated as an unknown variable. Thus, the number of unknowns can be reduced to only one, X_s. Write a computer program that solves the problem of the lateral inflow into a triangular gutter with a flat bottom using only one ODE equation. Then solve the two problems: (1) Lateral inflow of q* = 0.011 cfs/ft into a triangular gutter with b = 0, m = 4, n = 0.013, S_o = 0.001, i.e., 800 ft long, and (2) lateral inflow of q* = 0.02 cfs/ft in a flat bottom triangular gutter with b = 2 ft, m = 4, n = 0.013, S_o = 0.001, z = 0 ft that is 800 ft long.

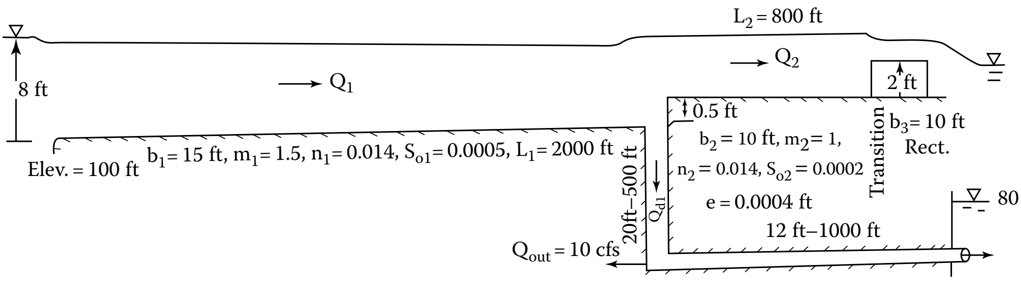
4.245 Obtain a series of solutions (for the previous problem) in which the lateral inflow q* varies from 0.002 m^2/s to 0.02 m^2/s into triangular gutter with a flat bottom with b = 0.4 m and a side slope of m =3.5, and a vertical rise z = 0.15 m. The gutter is 150 m long and has a bottom slope S_o= 0.0008 and a Manning's roughness coefficient n = 0.013.

4.246 From the series of solution obtained in the previous problem determine the minimum lengths of grates needed to accept the flow from both ends of the gutter.

4.247 A channel system is shown in the sketch below. Write out the equations, whose solution will provide the depths and flow rates throughout the system. Then solve this problem. What occurs at the downstream end of parallel channel, such as 4 and 5, if the channel to which it is connected has much more capacity than they do? Try obtaining a solution to this problem if the width of channel 6 is increased to 6 ft. Also solve the problem using Chezy's equation at the hydraulic equation if all roughness coefficients are e = 0.004 ft and ν = 1.217 × 10^{-5} ft^2/s.



4.248 The channel system is as in the sketch, takes water from a reservoir with a water surface 8 ft above the channel bottom. The entrance loss coefficient is K_e = 0.15. The upstream channel is trapezoidal with b_1 = 15 ft, m_1 = 1.5, n_1 = 0.014, and a bottom slope S_{o1} = 0.0005. This channel is 2000 ft long. At its downstream end, a 20 in. diameter pipe has its intake at the bottom of the channel. After 500 ft of this pipe length there is a withdrawning of Q_{out} = 10 cfs from it and the diameter reduces to 12 in. The 12 in. pipe terminates in a reservoir with a water surface elevation of 80 ft. Both pipes have an equivalent sand roughness e = 0.0004 ft for use in the Darcy–Weisbach, Colebrook–White equations. The bottom elevation of the channel at its very beginning is 100 ft. At the location where the pipe takes water from the channel, its bottom rises 0.5 ft and the channel changes size to b_2 = 10 ft, m_2 = 1, n_2 = 0.014, and the bottom



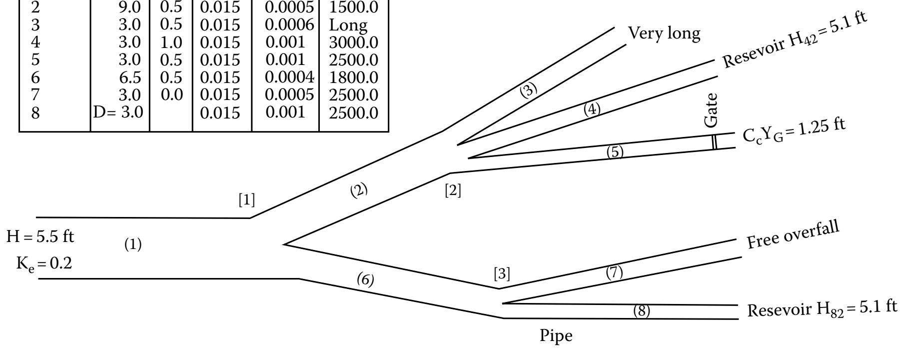
slope is $S_{o2} = 0.0002$. A distance 800 ft downstream, a 2 ft high broad crested weir exists, and thereafter the channel empties into a reservoir with a water surface at about the level of the crest of the weir. Do the following: (a) Identify the unknown variables that will need to be solved. (b) Write out the system of equations that will provide this solution. (c) Obtain the solution giving the flow rates in the two channels and two pipes, as well as the depths at the beginning and at the end of each channel. How does the flow rate in the first channel compare with what would exist if its flow were uniform?

4.249 The channel system as in the previous problem except that the pipe sizes have been enlarged to D_1 = 36 in. (3 ft), and D_2 = 24 in. (2 ft), and there is no rise in the bottom of the channel between 1 and 2, i.e., the position where the pipe takes water from the channel bottom is flat. The outflow at the junction of the two pipes is Q_{out} = 20 cfs. Solve the flow rates and the depth throughout the system.

4.250 The same channel system as in the previous two problems with the exception that the broad crested weir does not exist in the rectangular channel downstream from channel 2, but rather after the transition, the channel discharges into a reservoir with a water surface elevation of 106.5 ft. D_1 = 36 in., and D_2 = 24 in., and there is a rise of 0.5 ft atn the bottom of the channel between 1 and 2, Δz = 0.5 ft. The outflow at the junction of the two pipes is Q_{out} = 20 cfs.

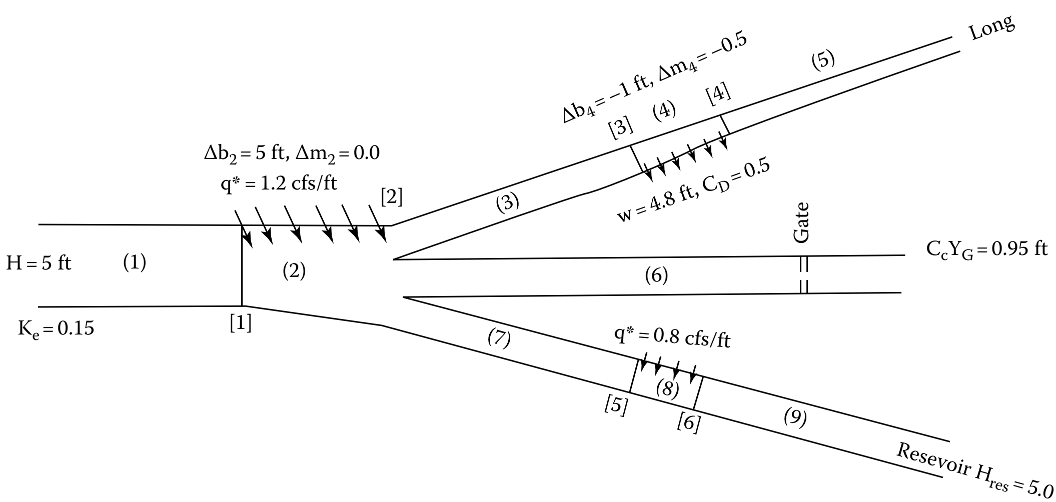
4.251 A branched system of eight channels has the configuration shown in the sketch below. The upstream main channel receives its water from a reservoir with a water surface elevation H = 5.5 ft above the channel bottom, and 1000 ft downstream branches into two channels, as shown. The entrance loss to the main channel is K_e = 0.2, and all other minor losses are zero. The sketch indicates what the downstream condition for each of the final branched channel is. All the channels are trapezoidal with the dimensions given in the table below except channel 8 which is a 6 ft diameter pipe. The gate at the downstream of channel 5 is set so that it produces a depth of 1.25 ft downstream from it. Write out the system of equations that need to be solved and then solve the flow rates, the upstream, and the downstream depths in all eight channels.

Channel	b (ft)	m	n	S_o	L (ft)
1	12.0	1.0	0.015	0.0006	1000.0
2	9.0	0.5	0.015	0.0005	1500.0
3	3.0	0.5	0.015	0.0006	Long
4	3.0	1.0	0.015	0.001	3000.0
5	3.0	0.5	0.015	0.001	2500.0
6	6.5	0.5	0.015	0.0004	1800.0
7	3.0	0.0	0.015	0.0005	2500.0
8	D= 3.0		0.015	0.001	2500.0



4.252 A branched system of channels is shown in the sketch below. The upstream main channel receives the water supply for the system from a reservoir whose water surface elevation is H = 5 ft above the channel bottom. Throughout the length of channel 2, the lateral inflow in the amount of q* = 1.5 cfs/ft occurs, and a lateral inflow of q* = 0.8 cfs/ft occurs into channel 8. The inflow into channel 2 has a component of velocity U_q = 2.0 fps in the direction of the main channel flow. A side weir with a height of 4.8 ft, and a discharge coefficient of 0.5 exists along channel 4. The sketch shows the conditions that exist downstream of the channels at the ends of the branches, and the table below gives the geometries of the channels. The width of channel 2 increases from b_2 = 10 ft given in the table to b_2 = 15 ft over its length and the m_2 remains constant at 1.5. The width of channel 4 reduces from 4 ft to 3 ft over its length and its side slope reduces from 0.5 to 0 (rectangular) at its end. (a) Write out the system of equations that need to be solved, and then solve the flow rates, the upstream, and the downstream depths in all nine channels. (b) How does the flow rate in channel 1 (i.e., the flow from the reservoir) compare with what would be taken from the reservoir if channel 1 were very long? (c) If the lateral inflow in channel 2 did not exist, what would the flow rate into the channel system be from the reservoir? (d) If the velocity component U_q of the lateral inflow were zero, what is the flow rate into the channel system?

Channel	b (ft)	m	n	S_o	L
1	10.0	1.5	0.015	0.0005	1500.0
2	10.0	1.5	0.015	0.0004	100.0
3	4.0	0.5	0.015	0.0004	1800.0
4	4.0	0.5	0.015	0.0004	30.0
5	3.0	0.0	0.015	0.001	1000.0
6	8.0	1.0	0.015	0.0009	3000.0
7	6.0	0.5	0.015	0.001	1000.0
8	8.0	1.0	0.015	0.0008	50.0
9	6.0	0.5	0.015	0.001	3000.0

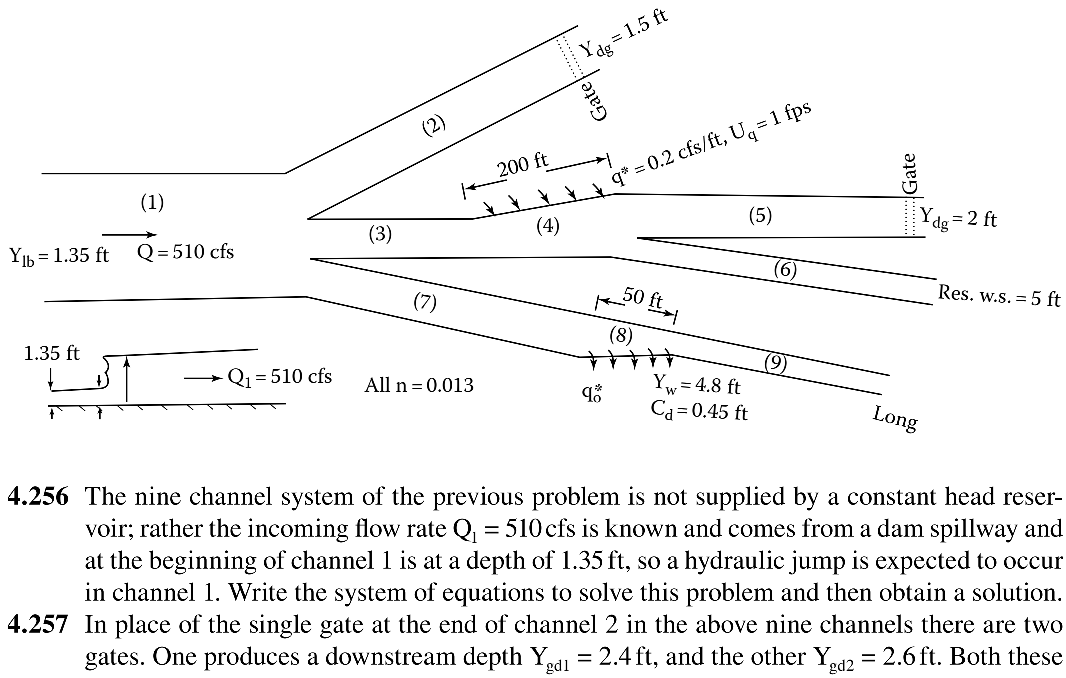


4.253 Solve the same channel system as in the previous problem except that the flow coming into channel 1 equals 300 cfs and is at a depth of 1 ft, being discharged from a vertical gate. In addition to solving case (a) with a lateral inflow of q* = 1.2 cfs/ft (U_q = 2 fps), solve the (c) and (d) parts of the previous problem in which q* = 0 in channel 2, and $U_q = 0$ in channel 2, respectively. Also solve case (a) again, but increase the flow rate to 355 cfs.

4.254 Systematically determine the effect of varying lateral inflow rates of q* and velocity components U_q of this inflow into channel 2 of the previous problem by solving a series of problems in which q* varies from 0 to 6 cfs/ft over the 100 ft length of this channel, and U_q varies from 0 to 6 fps in the direction of the channel flow.

4.255 A nine channel system is shown in the sketch below. The system is supplied by a constant head reservoir, with H = 5.2 ft. First identify the unknowns, and then give the system of equations that can be used to solve these. Obtain a solution giving the flow rates and depths through this channel system. (The sketch applies specifically to the next problem.)

Channel	b (ft)	m	S_o	L	Comments
1	18	1.0	0.001	1000	Contains a jump
2	6	1.0	0.0007	2500	Ends in a gate
3	7	0.0	0.0005	1000	
4	6–10	0.0	0.001	200	Bottom width expands from 6 to 10 ft
5	4	1.0	0.0006	2000	Ends in a gate
6	4	0.0	0.0002	2000	Discharges into a reservoir ws =5 ft
7	6	0.0	0.0008	1500	
8	5–3	0.0	0.001	50	Bottom reduces from 5 to 3 ft
9	3	0.0	0.0002		Long

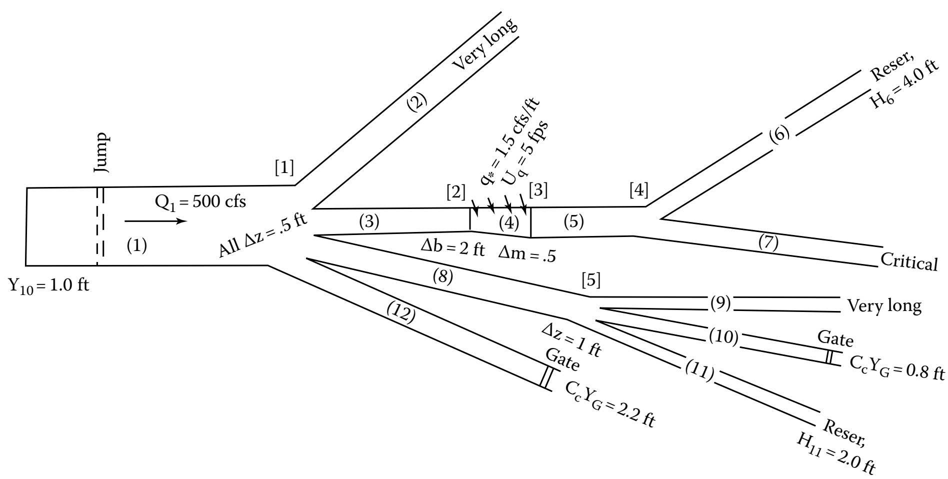


4.256 The nine channel system of the previous problem is not supplied by a constant head reservoir; rather the incoming flow rate $Q_1 = 510$ cfs is known and comes from a dam spillway and at the beginning of channel 1 is at a depth of 1.35 ft, so a hydraulic jump is expected to occur in channel 1. Write the system of equations to solve this problem and then obtain a solution.

4.257 In place of the single gate at the end of channel 2 in the above nine channels there are two gates. One produces a downstream depth $Y_{gd1} = 2.4$ ft, and the other $Y_{gd2} = 2.6$ ft. Both these gates are 2 ft wide with a 2 ft wide pier between them. Solve both the problems in which the system is supplied by a constant head reservoir with H = 5.2 ft, and the system in which the depth at the beginning of channel 1 is $Y_{1b} = 1.35$ ft, and its flow rate is $Q_1 = 510$ cfs. What does the solution indicate about the position of the hydraulic jump now? Assume that each of the gates is 3 ft wide and resolve the problem.

4.258 In the nine channel system of Problem 4.255, the flow into channel 7 is shut off. Now solve the flow rates and the depths in the other channels.

4.259 A 12 channel system is shown in the sketch below that starts downstream from a steep channel that produces a depth of 1 ft at the beginning of the 20 ft wide upstream channel 1, and the flow rate is $Q_1 = 500$ cfs. At the first junction of channel 1 with channels 2, 3, 8, and 12 there is a rise of 0.5 ft, i.e., channel 1 is lower by 0.5 ft than the other channels at this junction. In channel 4, which is 50 ft long, a lateral inflow of 1.5 cfs/ft occurs. Channel 11 is a pipe with a 3 ft diameter, and its bottom elevation at its beginning is 1 ft above the other three channels at this junction. The sizes, the slopes, etc., for the 12 channels is given in the table below, and the downstream boundary conditions are shown on the sketch. Do the following: (a) write out the system of equations that need to be solve to get flow rates and depths throughout this channel system, (b) solve this problem, (c) solve the critical depth in channel 1, and then in increments reduce the size of the rise in bottom at junction 1 (i.e., the amount that channels 2, 3, 8, and 12 are above the bottom of channel 1) until a solution is not possible for the entire system. Explain what is occurring and what is needed for a solution.



Channel	b (ft)	m	n	S_o	L (ft)	Δz	Downstream B.C.
1	20	1.5	0.016	0.0003	2500.0		
2	10	1.0	0.015	0.0006	Long	0.5	Uniform flow
3	8	1.0	0.015	0.001	500.0	0.5	
4	10–12	1–1.5	0.015	0.001	50.0		Lateralinflow q* = 1.5cfs/ft, U_q =5
5	12	1.5	0.015	0.001	1000.0		
6	4	0.	0.014	0.0015	2000.0		Reservoir with w.s. elev = 4, 0 ft
7	3	0.5	0.015	0.0005	2000.0		Free overfall
8	6	1.0	0.017	0.0003	1500.0	0.5	
9	3	0.5	0.018	0.0005	Long		Uniform flow
10	3	0.	0.016	0.001	2000.0		Gate set so depth downstream = 0.8 ft
11	D=3		0.014	0.001	1800.0	1.0	Reservoir with w.s. elev = 2.0 ft
12	5	1.0	0.015	0.0025	1500.0		Gate set so depth downstream = 2.2 ft

4.260 Determine values of N and M in Equation 4.38 for a very wide rectangular channel if Chezy's equation is used, and Chezy's C is constant.

4.261 Integrate the equation developed in the previous problem. Use this integrated equation to solve the S_2-GVF profile in the 12 ft wide steep rectangular channel of Example Problem 4.34, for a gate setting that produces a $Y_4 = 2.4$ ft (i.e., $Q = 391.84$ cfs, $Y_1 = 4.41$ ft, $Y_2 = 4.11$ ft, $Y_c = 3.21$ ft), and use a depth of $Y_u = 2.402$ ft upstream from the hydraulic jump using 16 steps, and compare these results with those obtained by numerically solving this GVF. Also use this integrated equation to solve Example Problem 4.4, and compare the results with the numerical solution of this example problem with the given 8 ft wide rectangular channel. (Note that the integrated equation assumes an infinitely wide channel.)

4.262 Write a computer program that will generate a table of varied-flow-function values, F(Y′,N). The values of Y′ should range from 0 to 5 in increments of 0.05, and the values of N should be 2.0, 2.25, 2.5, 2.75, 3.0, 3.25, 3.5, 3.75, and 4.0. Use the values in this table to solve the Example Problem 4.67.

4.263 Using the varied-flow-function method, determine the length of the M_1-GVF profile upstream from a dam that creates a depth of $Y_2 = 4$ m immediately upstream from it. The channel has the following properties: b = 5 m, m = 1.5, $S_o = 0.0014$, and n = 0.025. The flow rate in the channel is 20 m^3/s.

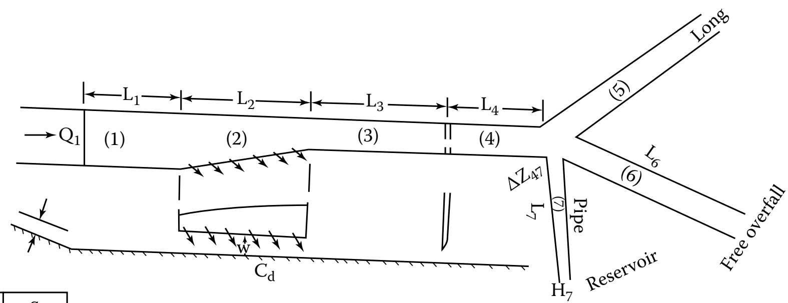
4.264 Write a computer program that utilizes the varied flow function, F(Y’,N), in solving GVF-problems, but that computes the depths over a change in position along the channel. In other words, consider x the independent variable and Y the dependent variable. Use this program to solve previous problems in this chapter, as designated by the instructor.

4.265 A channel with a bottom width b = 10 ft and a side slope m = 1.5 contains a gate 2000 ft downstream from a reservoir with a head H = 4.5 ft. At the position of the gate, the channel changes to a rectangular shape with b = 8 ft. Upstream of the gate, the slope of the channel is $S_{o1} = 0.0008$ and downstream from the gate the slope is $S_{o2} = 0.0005$. Manning’s coefficient is n = 0.013 for the entire channel. The channel is very long downstream from the gate. Starting with a depth Y_2 immediately downstream from the gate, and incrementing this depth by 0.2 ft until the gate becomes submerged, make a table giving the following values: (1) the depth Y_2, (2) the flow rate Q, (3) the depth at the beginning of the channel Y_{beg}, (4) the depth in the trapezoidal channel upstream from the gate at the beginning of the transition to a rectangular section Y_{1t} (5) the depth in the rectangular channel immediately upstream from the gate Y_{1r} (6) the normal depth downstream from the hydraulic jumpY_3 (7) the depth immediately upstream from the hydraulic jump Y_{2d} and (8) the depth upstream from the transition in the channel downstream from the gate if the depth from (7) should occur at the end of the transition Y_{2u}. If this last depth becomes less than Y_2, then the flow past the gate will become submerged.
Answer: Table giving solution

Y_2	Q	Y_{beg}	Y_{1t}	Y_{1r}	Y_3	Y_{2d}	Y_{2u}
0.20	35.07	4.496	6.094	6.088	1.184	0.367	0.422
0.40	68.85	4.486	6.075	6.054	1.746	0.595	0.717
0.60	101.20	4.469	6.046	5.999	2.171	0.781	0.977
0.08	132.00	4.447	6.006	5.924	2.517	0.941	1.215
1.00	161.12	4.419	5.956	5.830	2.809	1.079	1.437
1.20	188.39	4.388	5.897	5.717	3.059	1.202	1.644
1.40	213.67	4.352	5.828	5.585	3.275	1.309	1.837
1.60	236.78	4.314	5.752	5.434	3.461	1.403	2.015
1.80	257.56	4.274	5.668	5.266	3.620	1.485	2.178
2.00	275.84	4.234	5.579	5.082	3.754	1.555	2.323
2.20	291.49	4.195	5.488	4.882	3.866	1.613	2.450
2.40	304.42	4.159	5.399	4.669	3.956	1.660	2.557
2.60	314.64	4.129	5.316	4.448	4.025	1.967	2.644

The solution solves five equations for the unknowns: Q, Y_{beg},Y_{1t},Y_{1r}, and Y_3. Thereafter, the depth Y_{2d} is obtained by finding the conjugate depth to Y_3, and finally Y_{2u} is obtained by solving the energy equation $E_{2u} = E_{2d}$ in which 2u is in the rectangular section and 2d is in the trapezoidal section. (See program SOLGVFS4.FOR.)

4.266 In the sketch below, a plan and partial profile view of a channel system is shown that consists of the seven channel components with lengths $L_1 = 700\,m$, $L_2 = 60\,m$, $L_3 = 800\,m$, $L_4 = 600\,m$, $L_6 = 1200\,m$, $L_7 = 1000\,m$. All are trapezoidal channels except channel 7 which consists of a pipe. The sizes of these channel are given in the table below. Channel 2 consist of a side weir with a crest $Y_w = 2\,m$ above the channel bottom and a discharge $C_d = 0.45$. A gate separates channels 3 and 4 and the gate's bottom is $Y_G = 0.2\,m$ above the channel bottom, and it has a contraction coefficient $C_c = 0.58$. The circular channel has its bottom $\Delta z_{47} = 0.18\,m$ above the channel that it branches from. It discharges into a reservoir with a constant water surface $H_7 = 1.3\,m$ above its bottom and channel 6 ends in a free overfall, and channel 5 is very long. The flow rate coming into the channel system is $Q_1 = 80\,m^3/s$ as it comes over the crest of a dam spillway at a depth of 0.25 m. Set up and solve the system of equations for this channel system. Also solve the problem using Chezy's equation as the hydraulic equation with $\nu = 1.217 \times 10^{-5}\,ft^2/s$, and all the roughness coefficients are e = 0.004 ft.

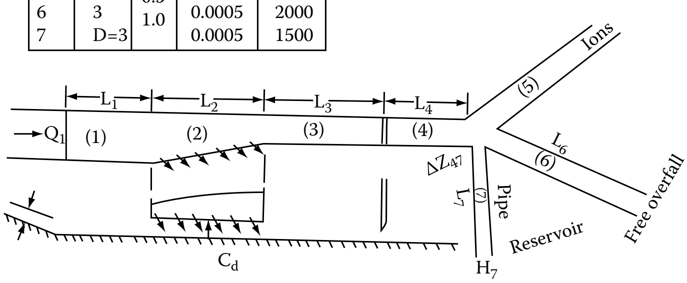


All n's = 0.013

No.	b (m)	m	S_o
1	15	1.5	0.0002
2			0.0003
3	10	1.0	0.0004
4	10	1.0	0.0004
5	3.0	0.0	0.0005
6	1.3	0.0	0.0005
7	D=3		0.0004

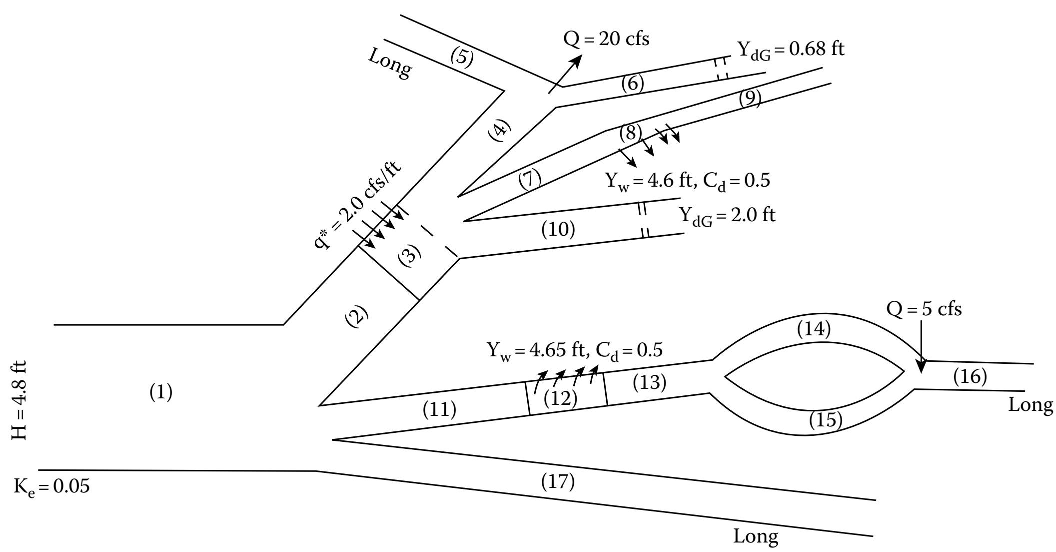
4.267 Solve the same channel system as in the previous problem, except use size and slopes in the table below, and the following: $Y_w = 2.4\,m$ and $C_d = 0.45$, depth in the upstream steep channel = 1.4 m, flow rate in the channel 1 $Q_1 = 89.2\,m^3/s$, $H_7 = 2\,m$, depth downstream from gate between channel 3 and 4 = 1.5 m and all Manning's n = 0.014. First use $\Delta z = 0.5\,m$; then lower the grate so that the depth is 1.3 m, and obtain a series of solutions for $\Delta z = 0.5$, 0.4, 0.3, 0.2, and 0.15 m.

No.	b (m)	m	S_o	L (m)
1	4	2.5	0.0006	1000
2			0.0006	50
3	3	1.5	0.0006	1500
4	3	1.0	0.0004	1200
5	2	0.5	0.0038	long
6	3	1.0	0.0005	2000
7	D=3		0.0005	1500



4.268 Given the 17 channel system shown in the sketch below, the height of the side weir along channel 12 is shown as 4.65 ft above the channel bottom. Obtain a series of solutions in which the weir height is decreased to 3.85 ft in increments of 0.05 ft, and examine how the flow rates, etc., change throughout the channel system due to this changing discharge from channel 12. Bottom widths and lengths are in feet as given in the table below. All Manning's n = 0.013.

No.	b	m	S_o	L
1	25	2.0	0.001	2000
2	20	1.5	0.0005	800
3	20	1.5	0.0005	50
4	6	1.25	0.0009	1300
5	1.4	1.2	0.0007	Long
6	4	1.5	0.001	1500
7	5	1.5	0.00085	1700
8	5	1.5	0.001	150
9	3.5	0	0.0005	Long
10	5	1.0	0.0006	2000
11	4	1.5	0.0006	1000
12	4	1.5	0.0006	150
13	4	0	0.0006	500
14	2	1.0	0.0005	1000
15	2	0.	0.0005	1000
16	4	0	0.0008	Long
17	4	1.5	0.0004	Long



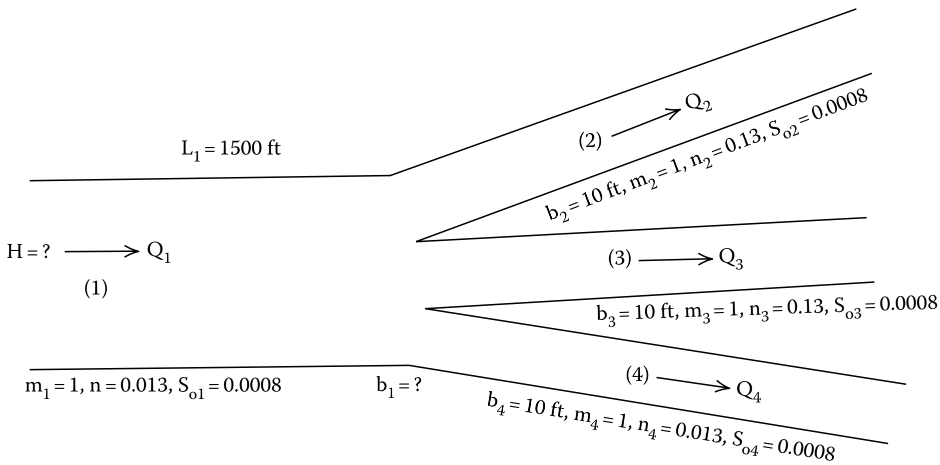
4.269 Assume in the channel system of the previous problem, that rather than being long, channel 17 is 3000 ft long and ends in a free overfall. Obtain a series of solutions in which the head of the supply reservoir varies from H = 4.0 ft to H = 5.0 ft in increments of 0.1 ft, and determine how the flow rates and the depths throughout the channel system are effected.

4.270 In the 17 channel system of the previous two problems investigate what happens as the widths of channels 2 and 3 are reduced in size by obtaining a series of solutions with b_2 and b_3, both simultaneously reduced from the present 20 ft widths. What occurs when the widths of these two channels reduce to about 17.5 ft? (The lateral inflow to channel 3 is q* = 2.0 cfs/ft.)

4.271 Verify the table of solutions given for the three channel branched system in which the depths in the downstream 10 ft wide channels were systematically increased from 2 to 6 ft with an increment of 0.25 ft. This table of solutions is given at the end of Section 4.21 just before Section 4.22. This table solves the width of the upstream channel so the critical flow occurs at its downstream end, and then solves its upstream depth and the head H of the reservoir head needed.

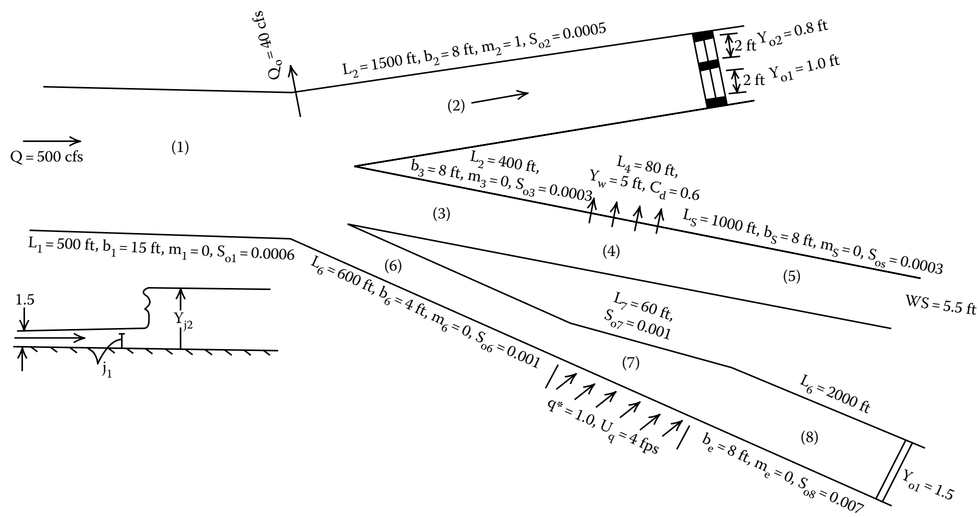
4.272 Investigate the effect of having different bottom slopes in the upstream channel by repeating the tables of solutions of the previous problem for different values of S_{o1}.

4.273 Solve the depth and the bottom width of the upstream channel, etc., for a series of depths in the downstream channels from 2 to 6 ft, as in the previous problem, which involves a three channel system, except make this a four channel system in which the upstream channel branches into three identical downstream channels with: b = 10 ft, m = 1, n = 0.013, and $S_o = 0.0008$. Also let $n_1 = 0.013$, $m_1 = 1$, $S_{o1} = 0.0008$, and $L_1 = 1500$ ft.



4.274 Obtain a series of solutions, as in the previous two problems for a four channel system in which channel 2 is as in the previous problem, i.e., $b_2 = 10$ ft, $m_2 = 1.2$, $n_2 = 0.014$, and $S_{o2} = 0.0008$, but channel 3 is one-half this width, i.e., $b_3 = 5$ ft, $m_3 = 1.2$, $n_3 = 0.014$, and $S_{o3} = 0.0008$. The upstream channel 1 is 1800 ft long and has $n_1 = 0.014$, $m_1 = 1.2$, and $S_{o1} = 0.0008$.

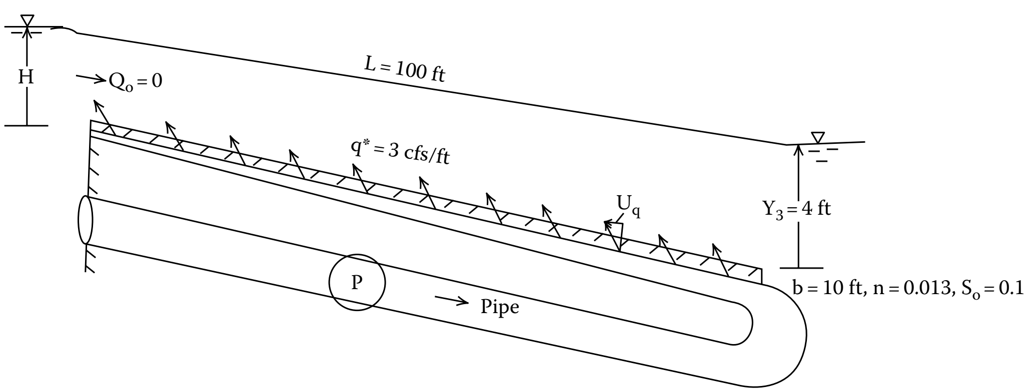
4.275 A channel system exists downstream from a dam spillway that is supplying it with $Q_1 = 500$ cfs as shown in the sketch below. Two gates exist downstream in channel 2 which have depths of 1.0 ft, and 0.8 ft, respectively, downstream from them. At a distance $L_3 = 400$ ft in channel 3, there is an 80 ft long side weir with a discharge coefficient $C_d = 0.6$ that is $Y_w = 5$ ft above the channel bottom. Downstream from this side weir, the channel discharges into a reservoir whose water surface is WS = 5.5 ft above the channel bottom. At the end of channel 6, which is 600 ft long, there is a lateral inflow of $q^* = 1.0$ cfs/ft over a 60 foot length with a component of velocity $U_q = 4$ fps in the direction of the channel flow. Further 2000 ft downstream in this channel, there is a gate that has a depth of 2.0 ft immediately downstream from it. A flow rate of $Q_o = 40$ cfs is taken out from the junction of channels 1, 2, 3, and 6. The depth of the 500 cfs coming from the steep upstream spillway channel upstream of the beginning of channel 1 is $Y_{o1} = 2.0$ ft. List the unknowns that you would solve in this problem, and then write out the system of equations whose solution will provide values for these. Finally obtain a solution using SOLGBRO.



4.276 Solve the previous problem except there is a single gate across the entire end of channel 2 that produces a depth of 0.9 ft downstream from it.

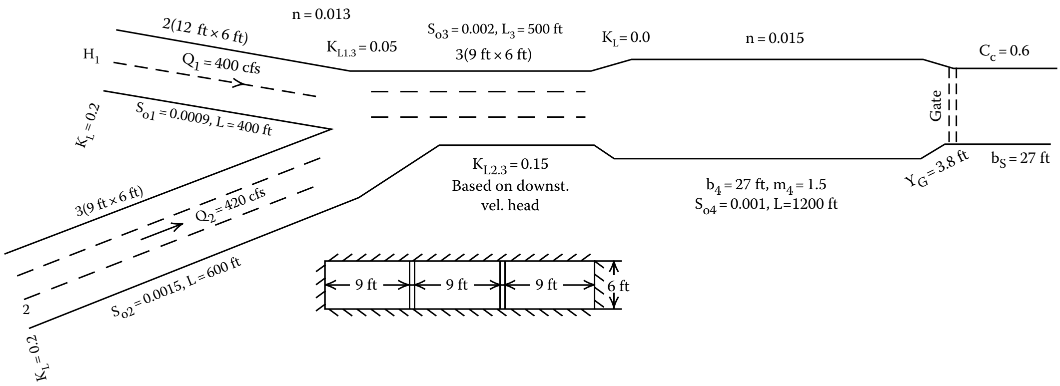
4.277 Same as the previous problem except have the channel reduce to a rectangular section at the gate position at the end of channel 2 with a gate width of 8 ft.

4.278 A means for allowing fish to migrate up a chute (channel) is proposed to consist of having a lateral inflow be pumped into the channel with a component of its velocity directed up the chute. Assume that the depth of water at the bottom of the chute is $Y_e = 4$ ft above the bottom, and the channel has a bottom slope of $S_o = 0.1$, and that the amount of lateral inflow pumped into the channel is $q^* = 3$ cfs/ft. (a) Solve the depths, etc., over a 100 ft length of chute that is rectangular with a bottom width $b = 10$ ft and a Manning's $n = 0.013$, if the components of velocity up the channel are $U_q = 10$ fps and $U_q = 20$ fps.



Obtain these two solutions based on the assumption that the flow rate into the top of the chute is $Q_o = 0$ cfs. (b) For the latter $U_q = 20$ fps obtain similar solutions for bottom slopes of $S_o = 0.12$ and $S_o = 0.13$. What do you observe happens as the bottom slope of the chute is increased? (c) Solve the flow rate Q_o at the beginning of the chute that will result in the downstream depth $Y_e = 4$ ft being at the critical flow. Solve the depths, etc., throughout the chute with this flow rate entering the chute at its upstream end.

4.279 A culvert system consists of two culverts that branch into a single culvert as shown in the sketch. Downstream from the culverts, the channel is trapezoidal with $b_4 = 27$ ft, $m_4 = 1.5$, and a bottom slope $S_{o4} = 0.001$. This trapezoidal channel is 1200 ft long at which position there is a short smooth transition to a rectangular section that contains a gate. The first branch of the culvert system consists of two rectangular sections each 12 ft wide and 6 ft high. The second branch of the culvert system consists of three rectangular sections each 9 ft wide and 6 ft high. The culvert into which these two branches join consists of three rectangular sections each 9 ft wide and 6 ft high, i.e., the same as branch two. The culverts all have a Manning's n = 0.013, and the trapezoidal channel has $n_4 = 0.015$. The bottom slopes and lengths of the three culverts are $S_{o1} = 0.0009$, $L_1 = 400$ ft; $S_{o2} = 0.0015$, $L_2 = 600$ ft, $S_{o3} = 0.002$, and $L_3 = 500$ ft. If the flow rate in branch one is $Q_1 = 400$ cfs, in branch two $Q_2 = 420$ cfs, and the gate is set so its bottom is $Y_G = 3.8$ ft above the channel bottom (its contraction coefficient is $C_c = 0.6$), determine the depths throughout the system including what the reservoir heads H_1 and H_2 must be that feed culvert branches one and two. (The two entrance loss coefficients are $K_e = 0.2$, the loss $K_{L1\text{-}3} = 0.05$ and $K_{L2\text{-}3} = 0.15$.)



4.280 If rather than specifying the flow rates Q_1 and Q_2 in the culvert system of the previous problem, the upstream reservoir water surfaces H_1 and H_2 are specified, how would you solve the problem? (You should identify all the unknown variables, and then give the equations that you would use to solve these unknowns). If $H_1 = 4.6$ ft and $H_2 = 4.7$ ft, determine the flow rates and the depths throughout the system.

4.281 Discuss what will happen as the gate is lowered in the previous problem. Solve the problem with the gate at a position 2.0 ft above the channel bottom.

4.282 Solve Example Problem 4.68, but rather than closing the gate from $Y_g = 2.5$ to 1.5 ft, close the gate entirely, i.e., set $Y_g = 0$ so all the flow past the gate is stopped at t = 0 (H = 5 ft, $K_e = 0$, b = 10 ft, m = 1.5, n = 0.013, $S_o = 0.001$, and L = 2800 ft).

4.283 Modify Program WAVEMOV.FOR so that in place of calling on the ODE solver ODESOLF, it calls on the solver RUKUST. With this modified program solve Example Problem 4.68.

4.284 Modify Program WAVEMOVI.FOR (the program that uses the implicit method) so that in place of calling on the ODE solver ODESOLF it calls on the solver RUKUST. With this modified program solve Example Problem 4.69.

4.285 Solve Example Problem 4.68, but rather than instantly closing the gate to a new setting of $Y_g = 1.5$ ft, the flow rate past the gate is controlled so that it equals one-quarter of the initial flow rate.

4.286 Modify Program WAVEMOVR.C so that in place of calling on the ODE solver RUKUST, it calls on the solver ODESOLC. With this modified program solve Example Problem 4.68.

4.287 A 1000 m long trapezoidal channel with b = 4 m, m = 1, n = 0.014, and $S_o = 0.0012$ is supplied by a constant head reservoir with a head of H = 2 m and $K_e = 0.15$. Initially, the gate has its bottom $Y_g = 0.9$ above the channel bottom. Its contraction coefficient is $C_c = 0.7$. Then at t = 0, the gate is suddenly closed so $Y_g = 0.2$ m. Using the explicit method, solve the moving wave that occurs upstream from the gate.

4.288 Resolve the previous problem but use the implicit, rather than the explicit method.

4.289 Resolve Problem 4.287, but rather than keeping the gate setting constant for time after $t = 0$, control the flow rate so the amount passing the gate equals one-third of the initial amount.

4.290 Resolve Problem 4.287, except at time $t = 0$ the gate is completely closed.

4.291 A rectangular channel with $b = 12$ ft, $n = 0.012$, and $S_o = 0.0008$ is 2000 ft long. It is supplied by a reservoir whose head is constant at $H = 6$ ft above the channel bottom and the entrance loss coefficient is $K_e = 0.12$. Initially and for a long time, a free outflow has been occurring at the end of the channel. Suddenly a vertical gate ($C_c = 0.6$) is lowered so its height is $Y_g =$ 1.5 ft above the channel bottom. Solve the moving wave with an increment $\Delta t = 20$ s using both the explicit and the implicit methods.

4.292 Solve the wave in the previous problem, but rather than holding the gate at a specified distance above the channel bottom, the flow rate leaving the end of the channel is controlled to be one-third of its initial free outflow.

4.293 Solve Example Problem 4.68 except initially the flow ends in a free overfall at the end of the channel, and then the gate is lowered to $Y_g = 1.5$ ft, to cause the wave to form upstream therefrom.

4.294 Repeat the previous problem but rather than fixing the position of the gate, the flow rate at the end of the channel is reduced instantly to one-half its initial value at time $t = 0$, and held constant thereafter.

4.295 Repeat the previous problem but rather than holding the flow rate constant, it is reduced to 0 cfs in 60 s.

4.296 A long trapezoidal with $b = 4$ m, $m = 1.2$, $n = 0.014$, and $S_o = 0.00035$ has been receiving a flow rate $Qo = 8.5\,m^3/s$ for a long time, when suddenly this flow rate is doubled to $17\,m^3/s$. Solve the UCW that will result using 5 s time steps.

4.297 Solve Example Problem 4.70 with the flow rate initially increasing from 250 to 500 cfs as in that problem, but thereafter increase the flow rate that enters the channel by 0.2 cfs/s up to the time of 60 s and for later times hold it constant at 510 cfs. Assume that during this time of increasing flow rate, and thereafter, that the head of the reservoir H is constant, but that the depth Yb at the beginning of the channel will vary. Use the explicit method. (To solve this problem, you can modify program WAVEUCE so that it adds Equation 4.3u to the system of equations being solved, and Yb is added as an additional unknown variable.)

4.298 Resolve Example Problem 4.70, but rather than having a uniform flow in the channel as the initial condition, the channel is 2000 ft long and ends in a free overfall.

4.299 Resolve Example Problem 4.70, but rather than having a uniform flow in the channel as the initial condition, the channel is 2000 ft long and discharges into a reservoir with a water surface elevation at 4.5 ft.

4.300 Programs WAVEUCE and WAVEUCI assume that the flow rate varies linearly from the upstream end of the channel to the position of the wave, and that the depth Y_b at the upstream end of the channel does not change from that determined by solving the new depth Y_b caused by the new specified flow rate Q_b. Make the following different assumptions: (1) the flow rate is constant from the beginning of the channel to the position of the wave and this value is specified as the new Q_b, and (2) that the depth Y_b varies as needed to allow for the GVF. The upstream control to allow this condition would be a device that supplies the specified flow rate regardless of the depth at the beginning of the channel. Assume as in these programs, that the flow rate downstream from the moving wave is constant and equal to the value Q_o that is specified as the initial flow rate. Solve Example Problem 4.70.

4.301 In Example Problem 4.73, the assumption was made that the flow rate at the end of the channel remains constant at all times and is equal to the original flow rate, i.e., 500 cfs. Write the program WAVEMJP6 and verify the solutions obtained for this problem.

4.302 The assumption in Example Problem 4.73 was that the flow rate at the end of the channel does change. Impose the additional assumption that the flow rate downstream from the jump remains constant and equal to the original flow rate. This means that rather than solving the downstream GVF, the depth Y_d can be obtained by interpolation of the original GVF profile based on the position of the jump. Thus, two additional unknowns are eliminated by this assumption, the flow rate Q_d and the depth Y_d. The remaining four unknown variables are: Y_u, v, Q_u, x_j. Write a program to solve Example Problem 4.73 based on these assumptions.

4.303 Use the program you wrote for the previous problem to solve Example Problem 4.72 in which the original flow rate $Q_o = 480$ cfs is suddenly increased to $Q_b = 550$ cfs. In solving this problem, assume that the normal depth occurs downstream from the jump, e.g., the depth downstream from the jump will always be Y_o associated with $Q_o = 480$ cfs.

4.304 Repeat the previous problem but rather than specifying the normal depth downstream from the jump, assume that the channel ends at 800 ft in a free overfall, e.g., that a critical flow occurs at the end of the channel.

4.305 Modify program WAVEMJP7 so that rather than giving the initial flow rate Q_o, the initial upstream head H is specified. With this program solve Example Problem 4.72 with H = 7.743 to see if the same solution results.

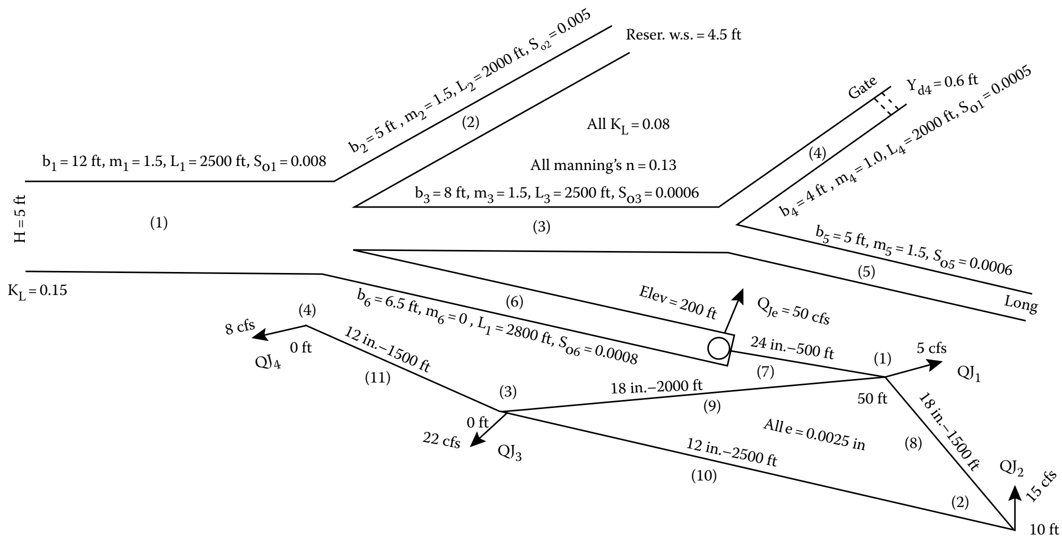
4.306 Modify program WAVEMJP7 so that rather than having a GVF downstream from the jump assume that the depth is always normal downstream from the jump. What this assumption entails is that the equation from the downstream GVF is replaced by an algebraic equation for a uniform flow (Manning's equation). Use the program to solve Example Problem 4.72, with the downstream depth $Y_d = Y_o$, based on Q_d, rather than the specified $Y_3 = 6$ ft.

4.307 Compare the solution you obtained in the previous problem with a solution in which you specify the depth at the end of the 2000 ft long channel equal to the normal depth Y_o associated with the initial flow rate $Q_o = 500$ cfs. Note, the latter solution can be obtained from program WAVEMJP7 with Y_3 specified equal to Y_o.

4.308 Modify program WAVEMJP7 so that at the end of the channel with a length L, the depth will be normal for the flow rate at the end of the channel, but since the flow rate can vary between the jump and the end of the channel, a GVF will exist in this portion of the channel. In other words, rather than specifying the depth Y_3 at the downstream end of the channel, this depth will be one of the unknown variables and will satisfy Manning's equation for the flow rate at the end of the channel. Solve Example Problem 4.72 with this program. Also obtain a solution from program WAVEMJP7 in which you specify Y_3 equal to the normal depth associated with $Q_o = 500$ cfs, $Y_c = 4.079$ ft, and compare the results.

4.309 Modify the program you developed in the previous problem so that rather than having the normal depth at the end of the channel with length L, it ends in a free overfall. In other words, have the depth Y_3 satisfy the critical flow equation rather than Mannings' equation. Solve Example Problem 4.72 assuming it ends in a free overfall at a position L = 2000 ft downstream from its beginning.

4.310 The six branched channel system shown below supplies a five pipe network, whose diameters and lengths are given on the sketch. The pipes are numbered 7, 8, 9, 10, and 11. The entrance loss coefficient to Channel # 1 is $K_e = 0.15$ and the loss coefficients between the upstream and for all downstream channels is $K_L = 0.08$. In addition to the flow rate in pipe (7), there is an outflow of $QJ_e = 50$ cfs from the end of channel 6. Do the following: (a) For design purposes, the demands at the four nodes of the network are $QJ_1 = 5$ cfs, $QJ_2 = 15$ cfs, $QJ_3 = 22$ cfs, $QJ_4 = 8$ cfs. Solve the flow rates in all channels, their depths and the flow rates in the pipes. (b) Solve all the flow rates and the depths in the channels, and the flow rates in the pipes if the demand at node 4 is unknown, but the pressure here is 40 psi. (c) Assume that actually the HGL elevation at the nodes of the pipe network are H_1 = 190 ft, H_2 = 160 ft, H_3 = 140 ft, and H_4 = 95 ft. Now solve all the flow rates, the depths and the discharges QJ_1, QJ_2, QJ_3, and QJ_4 from the pipe network.



REFERENCES

Bakhmeteff, B. A., *Hydraulics of Open Channels*, McGraw-Hill Book Company, Inc., New York, 1932.

Chow, Ven Te, 1959. *Open-Channel Hydraulics*, McGraw-Hill Book Company, Inc., New York, Section 10-2 "Method of Direct Integration", pp 252–262 and Appendix Tables D and E.

de Marchi, G. 1934, Saggio di teoria di funzionamente degli stramazzi laterali. *L'Energia Elletrica*, 11(11), 849–860.

5 Common Techniques Used in Practice and Controls

5.1 INTRODUCTION

The contents of this chapter can be divided into (1) solving problems in natural or irregular channels and (2) structures used to control and measure channel flows, including the design of transitions. The subject of flow in natural channels will be dealt with first and thereafter controls and other topics will be covered as they are used in both man-made and natural channels. Currently the program HEC-2 "Water Surface Profiles" developed by the U.S. Army Corps of Engineers, Hydrologic Engineering Center, is quite widely used in practice in solving open channel flows in natural channel. Therefore, after describing the principles and techniques that are applicable to irregular channels, the use of the HEC2 software package will be dealt with briefly assuming that it is available to you, the reader, along with its manual describing the input data needed.

5.2 RESISTANCE TO FLOW IN NATURAL STREAMS AND RIVERS

Calculations relating depth to flow rate in natural channels generally uses Manning's equation in current U.S. practice. Thus these calculations require values for Manning's roughness coefficient n. The processes that contribute to flow resistance in natural streams and rivers are complex, and those dominating under normal flow conditions are different from those dominating under high-flow flood conditions. Studies have shown that many factors influence flow resistance and energy dissipation. The flow in high-gradient mountain streams and rivers exhibits large amounts of turbulence, which under large flows results in sprays into the air and frequent breaking waves, whereas the flow in a river meandering through a flat plain behaves more like the flow in a man-made trapezoidal channel with a mild slope. This complex nature of the hydraulics of natural streams and rivers is the result of many interacting factors, including bed grain size, bed forms, vegetal effects, sediment scour and deposition, debris such as trees and branches and tree roots, beaver dams, cross-section irregularities, meandering, suspended materials, etc. to name but a few, all of which change with time and flow conditions. Because of unknown relationships between these many variables, hydraulic computations in natural channels must be considered approximate. Furthermore, Manning's equation is likely to be appropriate only for those natural flows that are not too different from the data that have been used in the past, which has lead to the wide use of this equation. In this section a brief discussion is given related to how values of n might be obtained for natural channels.

Two widely used guidelines for selecting Manning's n are Chow's (1959) open channel book and USGS water supply paper by Barnes (1967). Barnes gives verified n values with color photographs, and descriptive data for 50 streams, most of which are flowing near full. Research and field data have clearly demonstrated that n is not related solely to the size of bed roughness, especially in natural channels with steep gradients. Greater resistance occurs when the depth of flow is small in natural channels for the size of roughness, and therefore n is related to the hydraulic radius of the cross section. Limerinos (1970) gives n values for 11 streams. He also provides the following equation that provides values of n as function of the hydraulic radius of the flow, and a bed particle size that equals or exceeds that of 84% of the particles sizes, d_{84}. This equation is,

$$n = \frac{0.02926R_h^{2/4}}{1.16 + 2Ln(R_h/d_{84})} \tag{5.1}$$

Equation 5.1 is clearly an empirical equation that has been developed for a narrow range of natural channels, and its use is confined within this range. Solving Equation 5.1 for d_{84} less than 0.5 ft with R_h greater than 1 ft produces values of n less than 0.01, which is unrealistic. Also note that if d_{84} is larger than R_h, which may be true for some steep gradient mountain streams with large rock on its bottom, that the natural logarithm becomes negative, and when the ratio $R_h/d_{84} = 0.56$, then the denominator of Equation 5.1 becomes 0, resulting in an infinite value for n. For $R_h/d_{84} < 0.56$, Equation 5.1 produces negative values for n. Thus it appears that the range of use of Equation 5.1 is restricted roughly to d_{84} between 0.15 and 0.5 ft, with R_h between 0.25 and 1.0 ft, with the requirement that d_{84} be larger as R_h becomes larger.

For the case of steep gradient mountain streams, large amounts of energy dissipation occurs due to wake turbulence and the formation of localized hydraulic jumps downstream from large roughness elements (boulders that typically protrude up to or above the water surface). Peterson (1960) and a series of his graduate students who have attempted to quantify resistance in high-gradient streams have referred to this as tumbling flows, as a classification beyond turbulent flow. For some reason, nature develops bed roughnesses sufficiently large in mountain streams to maintain much of the flow under subcritical conditions, with localized supercritical flows only over the tops of large rocks, etc. where a localized hydraulic jump contributes to the energy loss. Under large flow conditions, these mountain streams are characterized by turbulence that dissipate energy by throwing portions of the flow into the air as a jet very similar to a flip-buck at the end of a spillway is used to dissipate energy. In the natural stream, the jet is not clearly defined being localized, and the plunge pool consists of a subsequent boulder that results in another ill-defined jet that throws a spray into the air.

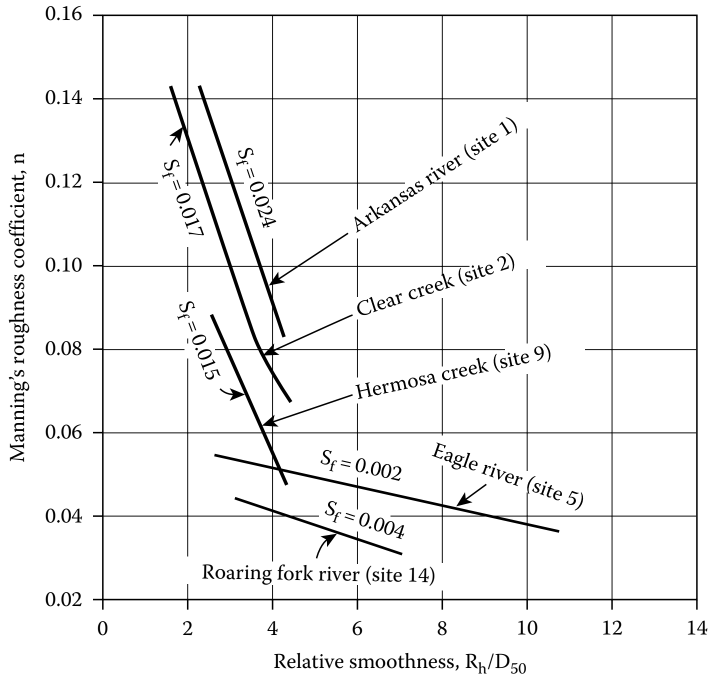
In a series of papers, backed up by much field data from flow measurements made on the large gradient river that flows through the **Rocky Mountain Hydraulic Laboratory**, Jarrett (1990) shows that Manning's n varies inversely with the depth of flow, and directly with the magnitude of the bottom slope. He gives the following equations:

$$n = \frac{0.315S_f^{.38}}{R_h^{.16}} \quad \text{for SI units (i.e., } R_h \text{ in m)} \tag{5.2}$$

and

$$n = \frac{0.39S_f^{.38}}{R_h^{.16}} \quad \text{for ES units (i.e., } R_h \text{ in ft)} \tag{5.2a}$$

These equation apply only to higher-gradient mountain streams and rivers with bottom slopes of 0.002 and larger to as large as 0.052. The slope of the channel bottom S_o can be substituted in place of the slope of the energy line S_f in Equations 5.2.



Relation of manning's roughness coefficient, n to relative smoothness and friction slope, R_h/D_{50}

In arriving at these equations, Jarrett gives the above figure that relates Manning's n to the relative smoothness defined as the hydraulic radius divided by the mean grain size of the bed materials, D_{50}, which indicates that n varies directly with the slope S_o and inversely with R_h/D_{50}. This is called relative smoothness since it is the inverse of the relative roughness.

Note from this figure that as the slope S_o becomes larger that n rapidly increases in size as R_h/D_{50} decreases. This figure can be useful in deciding if Manning's equation should be modified for how the depth changes with flow rate, etc. in a given river or stream. It shows that when S_o is small such as 0.004 and 0.002 that the dependency of n on R_h is small and Manning's equation could probably be assume to have an n dependent only on the size of the bed roughness, but when $S_o = 0.015$ and larger that n does depend strongly enough on R_h that Equations 5.1 and 5.2 should be used to modify its value.

Using the above relationships for n, result in the following equations in place of Manning's equations:

$$Q = 3.17AR_h^{.83}S_o^{.12} = \frac{3.17A^{1.83}S_o^{.12}}{P^{.83}} \quad \text{for SI units} \tag{5.3}$$

and

$$Q = 3.81AR_h^{.83}S_o^{.12} = 3.81A^{1.83}S_o^{.12}/P^{.83} \quad \text{for ES units} \tag{5.3a}$$

Note from these equations how the flow rate is given as a function of the slope of energy line (or bottom slope) raised to the 0.12 power, rather than the 0.5 power as given by Manning's equation.

Traditional head losses under turbulent flow conditions, whether in closed conduits or open channels, have been taken as proportional to the velocity squared (or flow rate squared if the area of flow remains constant). According to Equations 5.2 and 5.3, however, this proportionality constant is 1/0.12 = 8.33 for steep gradient mountain streams. In other words, according to Manning's equation (with n constant) a doubling of velocity causes a fourfold increase in head loss, but according to Equations 5.3 a doubling of the velocity results in 323 (=$2^{8.33}$) times the head loss. Clearly for Equations 5.3 to be valid, much of the energy dissipation occurs through processes other than fluid friction on the channel walls, such as the continual addition of large-scale turbulence that is dissipated very similar to that occurring in a hydraulic jump. The application of Equations 5.2 and 5.3 are also no doubt limited to a range of velocities as well as channel types and conditions.

Manning's equation will be used below in the computations for natural channels to illustrate the techniques and methods. However, when Manning's equation is not applicable for a given situation, then it should be viewed as an equation of the form

$$V = \left(\frac{C_u}{n}\right) R_h^{e1} S_o^{e2} = C R_h^{e1} S_o^{e2} \quad \text{or}$$

$$Q = \left(\frac{C_u}{n}\right) A R_h^{e1} S_o^{e2} = C A R_h^{e1} S_o^{e2}$$

where the parameters C_u, e1, and e2 can be changed, if appropriate, to values that are better for the given natural river or stream, when available data allows this to be done.

5.3 TECHNIQUES USED FOR SOLVING STEADY FLOWS IN IRREGULAR CHANNELS

If the geometry of a channel changes with the position along the channel, then the flow will be nonuniform and generally gradually varied. Uniform flow never occurs in irregular channels, and solutions for depths and velocities in such channels are obtained using extensions of the methods described in the previous chapter for solving GVF problems. Since information about the properties of the cross sections of an irregular channel are given at positions along the channel (or at given x values), only those methods that treat x as the independent variable are directly applicable. This means that one of the forms of the GVF equation needs to be used that contains dx in the denominator of the derivative. Equation 4.8 considers the specific energy E as the dependent variable is easier to use generally than an equation that considers the depth Y as the dependent variable, such as Equation 4.6. Equation 4.8, or

$$\frac{dE}{dx} = S_o - S_f \tag{4.8}$$

will be used in the subsequent sections. An advantage in using dE/dx instead of dY/dx is that it is not necessary to define terms that give the changes in the cross-sectional area as a function of x (the terms that contains $\partial A/\partial x$), or terms that involve lateral inflow or outflow (the terms that contain dQ/dx or q). (A proof of this fact is required in Problem 5.4.) There is a trade-off for not having to evaluate $\partial A/\partial x$, however. Since S_f in Equation 4.8 depends upon the depth, Y, as defined by the uniform flow equation (Manning's or Chezy's equation), the use of Equation 4.8 requires that the specific energy equation $E = Y + Q^2/(2gA^2)$ also be solved.

Two approaches to solving Equation 4.8 in irregular channel will be dealt with subsequently. The first is adapted for hand computations and is easily implemented in a spreadsheet, and is described in most other books that cover the subject of open channel flow. The second approach utilizes ODE solvers as in Chapter 4. The major difference in the second approach and that used for regular channels is that now algebraic equations are not available that define the relationship between the area, A, the wetted Perimeter, P, and eventually the slope of the energy line, S_f as functions of the of depth Y. Without these functional relationships a "table look-up" technique is used to define the relationship between needed variables. Before these methods can be effectively described, it is necessary to first discuss effective methods for defining the geometry of irregular channels.

5.3.1 Defining Irregular Channel Properties

The cross-sectional properties of rivers and other natural channels are generally measured at known stations, or positions along the channel. (See Appendix A for more detailed computational techniques to determine needed values, such as area, A, wetted perimeter, P, etc. from cross-section data.) Three common methods used to describe the cross section at these stations are as follows: (1) Providing a number of pairs of values (x_b, z) that give the horizontal distance from bank to a point on the cross section's bottom and the corresponding elevation, (2) providing pairs of values as in (1) but giving the vertical distance downward from the top of the channel in place of the elevation as the second value (x_b, y_t), and (3) providing pairs of values (T, y_b) that give the width of the channel at various distances upward from the bottom of the channel. Several other possible variations may be used, but these three methods are illustrated in Figure 5.1.

Method # 2 can be converted to method # 1 by subtracting the second value of each pair from the elevation of the reference bank point, $y_t = z_{min} - z$ or method # 1 can be converted to method # 2 by subtracting the second of each pair from the elevation of the bank, $z = z_{top} - y_t$. Method # 3 provides data from which it is convenient to obtain cross-sectional areas, wetted perimeters, hydraulic radii, and conveyance values (as well as top widths which are given) as a tabular function of the depth of water in the channel. The natural channel component of program CHANNEL accepts any of the above three descriptions of an irregular channel's cross section and provides a table of values with constant depth increments giving the area, A, the wetted perimeter, P, and the top width, T for each of these increments. A variety of methods may be used to develop such tables of variable dependency. Below, such a method is described that using data in the form of method # 1.

The procedure consists of the following steps: (1) Determine the point I_{min} that has the smallest value of elevation, z_{min}. This is the bottom of the channel. Assign x_{L1} and x_{R1} the x_b corresponding to this minimum point. (Subscripts L and R denote left and right sides, respectively.) (2) Establish indexes IR (and IR1 = IR − 1) and IL (and IL1 = IL + 1) that reference the next data point values on the right and left sides, respectively, of point I_{min}. If two points have the same minimum elevation, then $IR = I_{min} + 2$ where I_{min} is the first minimum value, or otherwise $IR = I_{min} + 1$. The index IL is initially given the value $I_{min} - 1$. The x_b values from the pairs of data provided to give this cross section, corresponding to the indexes, IL, IL1, IR, and IR1 will be designated as X_{IL}, X_{IL1}, X_{IR}, and X_{IR1}, respectively.

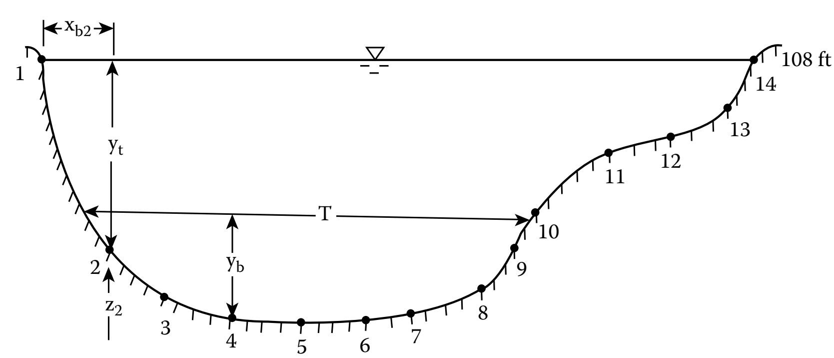


FIGURE 5.1 Typical irregular channel cross section.

Method # 1 Dist. from Bank and Elevation Bottom

pt.	x_b	z
1	0	108
2	4	102
3	8	100.2
4	12	99.6
5	16	99.6
6	20	99.7
7	24	100.0
8	28	100.5
9	32	101.4
10	34.2	103.4
11	36	104.9
12	40	105.3
13	44	106.2
–14	45.5	108.0

Method # 2 Dist. from Bank and Depth Below Top

pt.	x_b	y_t
1	0	0
2	4	6
3	8	7.8
4	12	8.4
5	16	8.4
6	20	8.3
7	24	8.0
8	28	7.5
9	32	6.6
10	34.2	4.6
11	36	3.1
12	40	2.7
13	44	1.8
14	45.5	0

Method # 3 Dist. Across Channel and Height Above

T	y_b
0	4.0
0.4	14.7
1.0	21.2
2.0	26.0
4.0	30.0
5.5	35.5
6.4	41.0
7.0	44.0
8.4	45.5

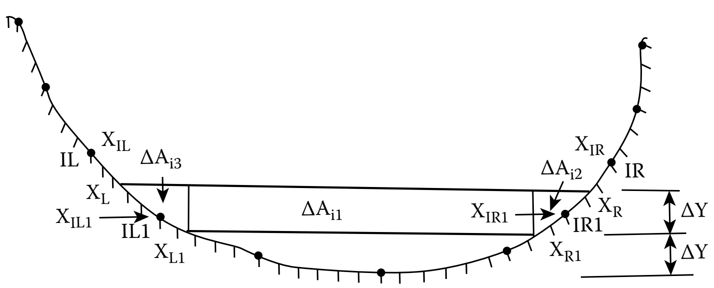
(3) Starting with the channel bottom, designate a depth Y = 0 and give the area, A, corresponding to this depth a value of 0, and if only one minimum occurs then both the top width, T, and the perimeter are given zero also, otherwise they are given values equal to the distance between the two minimum points. (4) Increment the depth by a constant amount, ΔY. Should this depth plus z_{min} be larger than the z corresponding with IR then increment IR (and IR1) by one. Likewise, should this depth plus z_{min} be larger than the z corresponding with IL, then decrement IL (and IL1) by one. (5) By interpolating between X_{IL}, and X_{IL1}, and also between X_{IR}, and X_{IR1} determine x_L and x_R, respectively. Letting ΔA_{i1}, ΔA_{i2}, and ΔA_{i3} be the incremental area shown on the sketch below, compute the new incremental area $\Delta A_1 = \Delta A_{i1} + \Delta A_{i2} + \Delta A_{i3}$ using the following formulas:

$$\Delta A_{i1} = \Delta Y(x_{R1} - x_{L1})$$

$$\Delta A_{i2} = \frac{\Delta Y(x_R - x_{R1})}{2}$$

$$\Delta A_{i3} = \frac{\Delta Y(x_{L1} - x_L)}{2}$$

Compute the incremental wetted perimeter from



$$\Delta P_i = \left\{\Delta Y^2 + (x_R - x_{R1})^2\right\}^{1/2} + \left\{\Delta Y^2 + (x_{L1} - x_L)^2\right\}^{1/2}$$

add ΔA_i and ΔP_i to the previously accumulated area and wetted perimeter and compute the top width from

$$T_i = x_R - x_L$$

(6) Assign values $x_{R1} = x_R$ and $x_{L1} = x_L$ and repeat steps 4 and 5 until Y is at the top of the channel.

The above procedure for obtaining a table that gives A, P, and T as functions of depth, Y, in a channel is implemented in the following FORTRAN and C programs. In these programs, the PARAMETER (const int) N = 26 establishes that this table will contain 26 entries. The program is used to obtain a table of geometric properties for the channel's cross section shown in Figure 5.1. Note that this program will use either linear, or quadratic interpolation depending upon whether the logical variable LINEAR is read as .TRUE. (1 in the C-program) or .FALSE. (0 in the C-program) in the input. When quadratic interpolation is used the Lagrange formula described in Appendix A is used. Linear interpolation was used for this cross section. However, quadratic interpolation will allow for the curvature of some cross sections to be better, provided the data is relatively smooth. If the channel's cross-sectional data are irregular, then the use of quadratic interpolation may result in assumed geometries that are too wide, or too narrow, between input data points, and then linear interpolation should be used.

FORTRAN program CHTABL that makes table of geometry

```
      PARAMETER (N=26)
      LOGICAL*2 LINEAR
      REAL X(20),Z(20),Y(N),A(N),P(N),T(N)
      READ(3,*) NVALUE,IOUT,LINEAR
      IF(NVALUE.GT.20) THEN
      WRITE(*,100)
100   FORMAT(' Program dimension to allow',' only 20 coordinate
     &pairs')
      STOP
      ENDIF
      READ(3,*)(X(I),Z(I),I=1,NVALUE)
      ZSTA=Z(1)
      IMIN=1
      YMAX=ZSTA
      DO 10 I=2,NVALUE
      IF(Z(I).GE.ZSTA) GO TO 5
      ZSTA=Z(I)
      IMIN=I
5     IF(Z(I).GT.YMAX) YMAX=Z(I)
10    CONTINUE
      IF(Z(IMIN+1).LT.ZSTA+1.E-5) THEN
      T(1)=X(IMIN+1)-X(IMIN)
      P(1)=T(1)
```

```
      IR1=IMIN+1
      XR1=X(IMIN+1)
      ELSE
      T(1)=0.
      P(1)=0.
      IR1=IMIN
      XR1=X(IMIN)
      ENDIF
      XL1=X(IMIN)
      AA=0.
      PP=P(1)
      A(1)=0.
      Y(1)=0.
      IL1=IMIN
      IL=IL1-1
      IR=IR1+1
      DY=(YMAX-ZSTA)/FLOAT(N-1)
      DY2=.5*DY
      DYS=DY*DY
      ILM=1
      IRM=NVALUE
      IF(.NOT.LINEAR) THEN
      IRM=IRM-1
      ILM=2
      ENDIF
      DO 40 I=2,N
      Y(I)=DY*FLOAT(I-1)
      YZ=Y(I)+ZSTA
      DO 20 WHILE (IR.LT.IRM.AND. Z(IR).LT.YZ)
      IR1=IR
20    IR=IR+1
      DO 30 WHILE (IL.GT.ILM.AND. Z(IL).LT.YZ)
      IL1=IL
30    IL=IL-1
      IF(LINEAR) THEN
      XR=X(IR1)+(YZ-Z(IR1))/(Z(IR)-Z(IR1))*(X(IR)-X(IR1))
      XL=X(IL1)+(YZ-Z(IL1))/(Z(IL)-Z(IL1))*(X(IL)-X(IL1))
      ELSE
      ILP=IL-1
      IRP=IR+1
      XR=(YZ-Z(IR))*(YZ-Z(IRP))*X(IR1)/((Z(IR1)-Z(IR))*
     &(Z(IR1)-Z(IRP)))+(YZ-Z(IR1))*(YZ-Z(IRP))*X(IR)/((Z(IR)-
     &Z(IR1)*(Z(IR)-Z(IRP)))+(YZ-Z(IR1))*(YZ-Z(IR))*X(IRP)/
     &((Z(IRP)-Z(IR1))*(Z(IRP)-Z(IR)))
      XL=(YZ-Z(IL))*(YZ-Z(ILP))*X(IL1)/((Z(IL1)-Z(IL))*
     &(Z(IL1)-Z(ILP)))+(YZ-Z(IL1))*(YZ-Z(ILP))*X(IL)/((Z(IL)-
     &Z(ILA))*(Z(IL)-Z(ILP)))+(YZ-Z(IL1))*(YZ-Z(IL))*X(ILP)/
     &((Z(ILP)-Z(IL1))*(Z(ILP)-Z(IL)))
      ENDIF
```

```
      T(I)=XR-XL
      AA=AA+DY*(XR1-XL1)+DY2*(XR-XR1+XL1-XL)
      A(I)=AA
36    PP=PP+SQRT(DYS+(XR-XR1)**2)+SQRT(DYS+(XL1-XL)**2)
      P(I)=PP
      XR1=XR
40    XL1=XL
      DO 310 I=1,N
310   WRITE(IOUT,311) I,Y(I),A(I),P(I),T(I)
311   FORMAT(I4,5F10.3)
      RETURN
      END
```

Program CHTABL.C

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
void main(void){
 const int n=26; int linear,nvalue,imin,ir1,ir,il1,il,irm,ilm,ilp,\
   irp,i;
 float aa,pp,zsta,ymax,dy,dy2,dys,yz,xr,xl,xr1,xl1;
 float x[20],z[20],y[26],a[26],p[26],t[26];
 char fnam[20]; FILE *fili,*filo;
 printf("Give name of file that contains input data\n");
   scanf("%s",fnam);
 if((fili=fopen(fnam,"r"))==NULL){printf("File does not exist.\n");
   exit(0);}
 printf("Give name of output file\n"); scanf("%s",fnam);
 if((filo=fopen(fnam,"w"))==NULL){printf("File cannot be opened\n");
   exit(0);}
 fscanf(fili,"%d %d",&nvalue,&linear);
 if(nvalue>20){printf("program dimension to allow only\
   20 coordinate pairs");exit(0);}
 for(i=0;i<nvalue;i++) fscanf(fili,"%f %f",&x[i],&z[i]);
 zsta=z[0]; imin=0; ymax=zsta;
 for(i=1;i<nvalue;i++){if(z[i]<zsta){zsta=z[i];imin=i;}
 if(z[i]>ymax) ymax=z[i];}
 if(z[imin+1]<zsta+1.e-5){t[0]=x[imin+1]-x[imin];p[0]=t[0];
   ir1=imin+1;xr1=x[imin+1];}
   else {t[0]=0.;p[0]=0.;ir1=imin;xr1=x[imin];}
 xl1=x[imin]; aa=0.; pp=p[0]; a[0]=0.; y[0]=0.; il1=imin;
   il=il1-1; ir=ir1+1;
 dy=(ymax-zsta)/(float)(n-1); dy2=.5*dy; dys=dy*dy; ilm=0;
   irm=nvalue-1;
 if(linear==0){irm=irm-1; ilm=1;}
 for(i=1;i<n;i++){y[i]=dy*(float)i; yz=y[i]+zsta;
  while ((ir<irm)&&(z[ir]<yz)){ir1=ir;ir++;}
  while ((il>ilm)&&(z[il]<yz)){il1=il;il--;}
 if(linear){xr=x[ir1]+(yz-z[ir1])/(z[ir]-z[ir1])*(x[ir]-x[ir1]);
```

```
  xl=x[il1]+(yz-z[il1])/(z[il]-z[il1])*(x[il]-x[il1]);} else{
   ilp=il-1;irp=ir+1;
r=(yz-z[ir])*(yz-z[irp])*x[ir1]/((z[ir1]-z[ir])*(z[ir1]-z[irp]))+\
  (yz-z[ir1])*(yz-z[irp])*x[ir]/((z[ir]-z[ir1])*(z[ir]-z[irp]))+\
  (yz-z[ir1])*(yz-z[ir])*x[irp]/((z[irp]-z[ir1])*(z[irp]-z[ir]));
l=(yz-z[il])*(yz-z[ilp])*x[il1]/((z[il1]-z[il])*(z[il1]-z[ilp]))+\
  (yz-z[il1])*(yz-z[ilp])*x[il]/((z[il]-z[il1])*(z[il]-z[ilp]))+\
  (yz-z[il1])*(yz-z[il])*x[ilp]/((z[ilp]-z[il1])*(z[ilp]-z[il]));}
t[i]=xr-xl; aa=aa+dy*(xr1-xl1)+dy2*(xr-xr1+xl1-xl); a[i]=aa;
pp=pp+sqrt(dys+pow(xr-xr1,2))+sqrt(dys+pow(xl1-xl,2)); p[i]=pp;
xr1=xr; xl1=xl;}
fprintf(filo," no. depth area perimeter top width\n");
for(i=0;i<n;i++)
 fprintf(filo,"%4d %9.3f %9.3f %9.3f %9.3f\n",i,y[i],a[i],p[i],t[i]);
}
```

Input for cross-section in Figure 5.1.

```
14 4.true.
0 108 4 102 8 100.2 12 99.6 16 99.6 20 99.7 24 100
28 100.5 32 101.4 34.2 103.4 36 104.9 40 105.3
44 106.2 45.5 108.0
```

Output from above computer program

Point No.	Depth Y, ft	Area A, sqft	Perimeter P, ft	Top Width T, ft
1	0.000	0.000	4.000	4.000
2	0.336	2.921	13.420	13.387
3	0.672	8.250	18.417	18.336
4	1.008	14.924	21.564	21.387
5	1.344	22.486	23.913	23.627
6	1.680	30.801	26.263	25.867
7	2.016	39.747	27.923	27.384
8	2.352	49.136	29.241	28.501
9	2.688	58.824	30.190	29.169
10	3.024	68.725	31.093	29.762
11	3.360	78.824	31.996	30.356
12	3.696	89.124	32.900	30.950
13	4.032	99.626	33.821	31.566
14	4.368	110.338	34.749	32.194
15	4.704	121.261	35.678	32.821
16	5.040	132.394	36.607	33.448
17	5.376	143.850	38.134	34.744
18	5.712	156.115	41.848	38.261
19	6.048	169.259	43.783	39.979
20	6.384	182.981	45.717	41.696
21	6.720	197.206	47.233	42.980
22	7.056	211.732	48.074	43.484
23	7.392	226.427	48.915	43.988
24	7.728	241.292	49.756	44.492
25	8.064	256.326	50.598	44.996
26	8.400	271.529	51.439	45.500

5.3.2 Use of Cubic Splines to Define Cross Section

The technique used above might be viewed as a crude numerical integration that provided the area (perimeter and top width) as a function of depth using differential elements with a height ΔY and a width equal to the distance from the left to the right bank at this elevation in the channel. Taking the elements in this direction has the advantage that a table is produced that give A, P, and T as a function of Y, and this table can serve the same role as equations relating these variables; the essential difference is that instead of doing the arithmetic of solving the equation, a table look-up, interpolation is used. The elements might be taken also in the vertical direction, especially if the objective is only to find the area, flow rate, average velocity, etc. for a known depth. Use of vertical elements makes it easier to handle problems for which specified velocity distribution are given as a function of depth. Cubic splines will be used next to define the shape of the natural section using (X_j, Z_j) (X = horizontal position, Z = elevation) pairs at points along the section where there is a significant change in the shape.

Splines are interpolating polynomials that are developed so that derivatives of the function, as well as the function, are continuous as the input data points are approached from both sides. A quadratic spline, or second degree polynomial approximation, maintains first derivatives continuous across data points; a cubic spline, or third degree polynomial approximation, maintains first and second derivatives continuous across data points. The difference between splines and regular polynomial interpolation is that the polynomial developed for a spline applies only over a single ΔX interval from X_j to X_{j+1} regardless of the degree of approximation, whereas a regular 2nd degree interpolating polynomial applies over $2\Delta X$, and 3rd degree applies over $3\Delta X$, etc. Spline interpolation is described in Appendix B. Cubic splines are popular because they provide a good "smoothing of data," and can be obtained using a small amount of arithmetic. In brief, a third degree polynomial is defined by computing a single shape (weighting) factor, $a = (X_{j+1} - x)/(X_{j+1} - X_j)$ and the following interpolating equation

$$z = aZ_j + bZ_{j+1} + \left\{\frac{(a^3 - a)(X_{i+1} - X_i)^2}{6}\right\} Z''_j + \left\{\frac{(b^3 - b)(X_{j+1} - X_j)^2}{6}\right\} Z''_{j+1}$$

where $b = 1 - a$ and Z''_j and Z''_{j+1} are the second derivatives of the function evaluated at points, j and j + 1, respectively. Note that a = 1 at X_j and a = 0 at X_{j+1}, whereas b = 0 at X_j and b = 1 at X_{j+1}. Thus if the second derivatives are obtained a simple equation provides a third degree polynomial approximation over each ΔX interval between consecutive data. (That third degree terms occur is clear because a and b occur to the 3 power in the above equation.) Furthermore, these second derivatives can be solved from a tridiagonal system of linear equations. In other words, after solving the second derivatives, Z'', and upon selection the correct pair of second derivatives Z''_j and Z''_{j+1} for the interval, in which the interpolation is to occur, and computing a for the position x within that interval the above interpolation equation provides the elevation of the channel bottom at the position x.

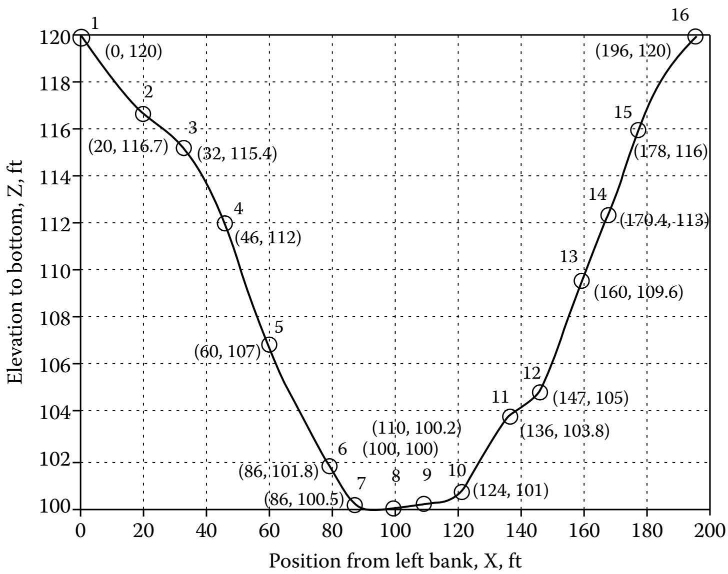
With z defined as a function of x the area can be obtained by summing widths time heights of vertical elements, or $A = \Sigma\Delta x(WS - z)$, where WS is the elevation of the water surface. The wetted perimeter $P = \Sigma\{\Delta x^2 + \Delta z^2\}^{1/2}$. If a known velocity distribution with depth y from the bottom at that vertical section is given, then the flow rate within that Δx vertical element can be obtained by integrating this velocity distribution between z and WS or $q_i = \int v\,dy$, etc. The total flow rate is $Q = \Delta x \Sigma q_i$.

EXAMPLE PROBLEM 5.1

The cross section of a flow in a river is given by the table of X (distance from left bank when looking upstream) and Z (elevation of point). Assume that the velocity distribution is given by the following equation as a function of the position y above the bottom of the river bottom at any position:

$$v(y) = 2.1\,\mathrm{Ln}(2.2y + 1) - 0.12y \text{ (fps)}$$

X (ft)	Z (ft)
0	120.0
20	116.7
32	115.4
46	112.0
60	107.0
80	101.8
86	100.5
100	100.0
110	100.2
124	101.0
136	103.8
147	105.0
160	109.6
170.4	113
178	116.0
196	120.0



Use a cubic spline to define the bottom of the channel from the given data and compute the following: (a) the flow rate, Q, (b) the area of flow, the wetted perimeter, and the average velocity, (c) the kinetic energy correction coefficient, α, and (d) the momentum flux correction coefficient, β. (In obtaining Q, α and β integrate vertically from the channel bottom to the water surface elevation at 120 ft for each 1 ft increment of channel width, and sum the result of these 196 vertical integrations. In other words determine q and other parameters for each unit width of channel.)

Solution

$A = 2209.0\,\text{ft}^2$, $P = 201.3\,\text{ft}$, $Q = 9970.5$ cfs, $V = 4.514$ fps, $\alpha = 1.210$, and $\beta = 1.081$.

The FORTRAN program below solves this problem using the following input: 3 16 196 120 (3 for INPUT unit, 16 for No. of pairs of (X, Z) given, 196 for the number of vertical sections to use, or $\Delta x = 1'$, and 120 for the water surface elevation). In addition, an input file read "READ (*, *) (X(I), Z(I), I = 1, N)" using logical unit 3 puts the above 16 pairs in arrays X and Z. The first task of the program is to solve for the second derivatives in array D2Y at the $n - 2 = 14$ point where two segments of input data join, or at X_j, $j = 2, 3, \ldots, 14$. This is accomplished in the program to statement labeled 20, and the code above this statement implements a solution to the tridiagonal linear system of equations as described in Appendix B. The loop DO 50 solves for the flow rate Q by calling on SIMPR (a numerical integration using Simpson's rule also described in Appendix B) to obtain the flow rate in each vertical differential element. The area and wetted perimeter are computed as described above in this same Do loop. The DO 90 has a very similar loop DO 80 within it that integrates v^3 and v^2 where the exponent is IE = 3 and = 2, respectively, to determine α and β, respectively. To solve a problem with a different velocity distribution in the vertical, the function subprogram EQUAT will need to be modified.

Program SPLINENA.FOR for obtaining A, P, V, α and β in a natural section

```
      EXTERNAL EQUAT
      COMMON IE
      CHARACTER*5 QUANT(2)/'Beta ','Alpha'/
      REAL X(41),Z(41),D(41),D2Z(41)
      WRITE(*,*)' Give: INPUT,N(No. pts),NX(No. delx), WS-elev'
      ITY=0
      IE=1
      READ(*,*) INPUT,N,NX,WS
      NM=N-1
      IF(INPUT.EQ.0 .OR. INPUT.EQ.5) WRITE(*,*)' Give',N,
     &' pairs of x y'
      READ(INPUT,*) (X(I),Z(I),I=1,N)
      IF(ITY.EQ.0) THEN
      D2Z(1)=0.
      D(1)=0.
      ELSE
      IF(INPUT.EQ.0 .OR. INPUT.EQ.5) WRITE(*,*)
     &' Give end derivatives'
      READ(*,*) DY1,DYN
      D2Z(1)=-.5
      D(1)=3.*((Z(2)-Z(1))/(X(2)-X(1))-DY1)/(X(2)-X(1))
      ENDIF
      DO 10 J=2,NM
      JM=J-1
      JP=J+1
      FAC=(X(J)-X(JM))/(X(JP)-X(JM))
      FA1=FAC*D2Z(JM)+2.
      D2Z(J)=(FAC-1.)/FA1
10    D(J)=(6.*((Z(JP)-Z(J))/(X(JP)-X(J))-(Z(J)-Z(JM))/(X(J)-
     &X(JM)))/(X(JP)-X(JM))-FAC*D(JM))/FA1
      IF(ITY.EQ.0) THEN
      D2YN=0.
      DN=0.
      ELSE
      D2YN=.5
      DN=3.*(D2YN-(Z(N)-Z(NM)))/(X(N)-X(NM))**2
      ENDIF
      D2Z(N)=(DN-D2YN*D(NM))/(D2YN*D2Z(NM)+1.)
      DO 20 J=NM,1,-1
20    D2Z(J)=D2Z(J)*D2Z(J+1)+D(J)
      J=1
      JP=2
      DX=X(2)-X(1)
      AR=0.
      Q=0.
      DELX=(X(N)-X(1))/FLOAT(NX)
      DELX5=.5*DELX
      DELXS=DELX**2
      Zb1=Z(1)
```

```
      DO 50 I=1,NX
      XX=DELX*FLOAT(I)-DELX5
30    IF(XX.LE.X(JP).OR.JP.EQ.N) GO TO 40
      J=JP
      JP=JP+1
      DX=X(JP)-X(J)
      GO TO 30
40    A=(X(JP)-XX)/DX
      B=1.-A
      Zb=A*Z(J)+B*Z(JP)+((A*A-1.)*A*D2Z(J)+(B*B-1.)*B*
     &D2Z(JP))*DX*DX/6.
      AR=AR+DELX*(WS-Zb)
      IF(I.EQ.1) THEN
      P=SQRT(DELX5**2+(Zb1-Zb)**2)
      ELSE
      P=P+SQRT(DELXS+(Zb1-Zb)**2)
      ENDIF
      CALL SIMPR(EQUAT,0.,WS-Zb,SQ,1.E-4,50)
      Zb1=Zb
50    Q=Q+SQ*DELX
      P=P+SQRT(DELX5**2+(Z(N)-Zb)**2)
      V=Q/AR
      WRITE(*,*) ' Area =',AR,' P =',P,' Q =',Q,' V =',V
      DO 90 K=2,1,-1
      IE=K+1
      J=1
      JP=2
      DX=X(JP)-X(J)
      FS=0.
      DO 80 I=1,NX
      XX=FLOAT(I)-DELX5
60    IF(XX.LE.X(JP).OR.JP.EQ.N) GO TO 70
      J=JP
      JP=JP+1
      DX=X(JP)-X(J)
      GO TO 60
70    A=(X(JP)-XX)/DX
      B=1.-A
      Zb=A*Z(J)+B*Z(JP)+((A*A-1.)*A*D2Z(J)+(B*B-1.)*B*D2Z(JP))*
     &DX*DX/6.
      CALL SIMPR(EQUAT,0.,WS-Zb,SQ,1.E-4,50)
80    FS=FS+SQ*DELX
90    WRITE(*,*) QUANT(K),' =',FS/(Q*V**K)
      END
      FUNCTION EQUAT(y)
      COMMON IE
      EQUAT=(2.1*ALOG(2.2*y+1.)-.12*Y)**IE
      RETURN
      END
```

The above example problem has specified how the velocity in the vertical direction varies by giving an algebraic equation. If data gave this variation, another spline function could be used to define this velocity distribution. If this were done then EQUAT could be modified to use this second spline function for the velocity in the vertical, and there would be two arrays to store the second derivatives for the two splines. Of course, there would also need to be another pair of arrays, such as YV and V for example, into which this additional data would be read, with YV the independent variable for the position above the channel bottom, and V the dependent velocity at this position. A third spline function could be used if data were available giving the variation of the velocity across the channel, or in the x direction, especially if the variation were defined as a multiplier of the vertical velocity at any depth y. For each such additional set of splines, the algorithm that solves the tridiagonal system of equations would be called to obtain the second derivatives at the points given by that independent variable. In other words, splines are a viable alternative to use to define the geometric and hydraulic properties of natural channels.

If one were to use splines to generate a table of values, as above, for use later to provide the relationships of A, P and T to Y, then it becomes necessary to repeat the solution process used in Example Problem 5.1 repeatedly for increments of depth from the first to the last depth entries in the table. During this by hand is not feasible; however, since the same process is repeated, it can readily be programmed for a computer solution, and represents a viable method that can provide more accurate answers. Since data giving channel shapes are generally not that accurate, however, more sophistication in processing the data may not be justified.

Splines can also be used with horizontal differential elements taken across the channel from the left to the right banks at any depth y, or elevation z. The horizontal positions X would be taken as the dependent variable and the elevation as the independent variable, i.e., the reversed role of variables used in program SPLINENA. Furthermore, to use splines with horizontal elements the input (X_j, Z_j) data might be divided into two groups with the division occurring at the bottom of the section where the smallest Z occurs. Separate splines could be obtained from these two groups of data, one for the left side of the section and the other for the right side of the section. The top width at any z then equals the difference between the two x values from the two splines and this difference gives the width of the horizontal element that would be integrated in the vertical direction to get the area.

Since the interpolation based on splines occurs only between two of the original data points, it is not necessary however to separate the input data into two groups to the left of and to the right of the bottom of the channel. One could determine the second derivatives at the point of all of the data, and then find the point where the minimum z (channel bottom) occurs, and then use splines to interpolate both to the left and right of the bottom point. The interpolation on the left side would move to decreasing input point numbers as the depth increased, and the interpolation on the right side would move to increasing point numbers with increasing depths. See the homework problems for exercise in writing such a computer program.

In what follows, the approach described earlier is used, not that it is superior to using splines (probably the reverse is true), but because it is both easier conceptually and computationally, and from a practical viewpoint, it is difficult to justify the greater smoothness provided by splines in the light of accuracy of most data and parameters that are available for natural channels.

5.3.3 Solving Manning's and Energy Equations in Natural Channels

Since the cross section of a natural channel generally is not constant over a long distance, the flow is seldom uniform. However, for reference purposes, it is often desirable to be able to solve the uniform flow equation in a natural channel. It may be desirable to determine what flow rate a given constant natural cross section will convey given its Manning's roughness coefficient, its bottom slope, and the depth of flow. Likewise, the normal depth may be sought given the other variables of the channel. Furthermore, the specific energy equation may be solved for any of the variables associated with it, or the momentum equation may be solved for any unknown associated with it based on the assumption that the other variables are known. It is desirable to know that the critical depth is in a natural channel. In brief, it may be desirable to solve any of the equations in a natural cross section that were solved in channel with a specified geometry. This section describes how these type of solutions can be effectively obtained.

For channels of known geometry equations, give the area, top width, and wetted perimeter as a function of the depth in the channel. For a natural channel, such equations are not available generally. In place of the equations, a table of values giving the area, top width, and wetted perimeter for discrete values of depth Y can be obtained, as described previously, and the desired values obtained by a "table look-up," i.e., interpolating values between a couple of entries. Thus for natural channel the equations are replaced by interpolating values in this table of geometric properties of the channel. Thus in a computer program that is used to solve an equation such as Manning's equation, it will be necessary to replace a statement that provides the area, A, and one that provides wetter perimeter, P by the necessary logic needed to obtain these values by interpolation in the table. To have the table available for this purpose, it must be stored in memory. One way of thinking about how this may be done is to consider having a SUBROUTINE of FUNCTION subprogram perform the task of computing the A, P, T, etc. This subprogram could have the logic within it to use the appropriate algebraic equations for different types of channel cross sections. Thus if the channel is trapezoidal then A is computed from $A = (b + mY)Y$ whereas if the channel is circular then $A = D^2/4(\beta - \cos\beta \sin \beta)$ is used, etc. For a natural channel, rather than obtaining these values by solving the correct equation the logic exist to first find the depth entries that bracket the depth for which A, P, etc. is desired, and after this position in the table has been identified, then the values desired are obtained by interpolation.

The program NATURAL.FOR, whose listing is given below (and also the C program NATREG.C) are designed to solve the following equations for a natural channel: (1) Manning's equation, $Q = C_n/nA(A/P)^{2/3}S^{1/2}$; (2) the specific energy equation $E = Y + (1 + K)(Q/A)^2/(2g)$, (3) the momentum equation $M = Ah_c + Q^2/(gA)$, (4) the critical flow equation $Q_2T/(gA^3) = 1$, (5) the critical flow equation $E_c = Y_c + (Q/A)^2/(2g)$, (6) the energy and Manning's equations simultaneously, and (7) the energy and the critical flow equations simultaneously. The FORTRAN program consists of the following subroutines (and what each's role is) (the C-program has similar functions, but it also solve trapezoidal and circular channels, which you are to do with the FORTRAN program as a homework problem):

EQS: This subroutine evaluates the above seven possible equations depending upon whether ICASE equals 1, 2, 3, …, 7. To evaluate these equations, the area (the FORTRAN variable AA), the perimeter (PP), and top width (TT) must be available. These become available to EQS by calling subroutine AREA.

AREA: This subroutine evaluates the area, the perimeter, and top width by using the logic described above in implementing a "table look-up." Notice it returns the area in AA, as it argument, and PP and TT through the common block TRAS.

CHTABL: This subroutine is essentially the same as the program by this name described previously. Its role is to read in the pairs of x, z values that define the shape of the natural channel and for a number of entries for depth Y determine the area, perimeter and top width values, and store these values in a table. The depths of this table are stored in the array Y(26), the corresponding areas, perimeters, and top widths are stored in the arrays A(26), P(26), and T(26), respectively. This subroutine generates 26 such lines of values in the table.

Ahcn: Since the first moment of area is a little complicated to obtain this FUNCTION SUBPROGRAM is included to evaluate Ah_c.

The main program is designed to allow its user to select the type of problem he wants solved, and then provide the known variables. It assumes that prior to executing NATURAL, the user has prepared the natural cross-sectional data in the format needed by program CHTABL and that these data are stored on a file. The length of the program has been kept small, and therefore, rather than internally generating a guess(es) for the unknown variable(s), the user must supply a guess(es) for the Newton method to start the iterative solution with. Thus the main program in addition to interacting with its user, it implements a solution to the problem specified by using the Newton method. For the problems that have two equations to solve simultaneously two unknowns are selected, and the Jacobian matrix (a 2×2 in this case) is defined and then the linear algebra of solving the linear system of equations is implemented.

Listing of program NATURAL.FOR

```
      CHARACTER *1 C(6,7)/'Y','Q','n','S',' ',' ','Y','Q','E',
     &'K',' ',' ','Y','Q','M',' ',' ',' ','Y','Q',' ',' ',' ','
     &','Y','E',' ',' ',' ',' ','Y','Q','H','K','n','S','Y','Q',
     &'H','K',' ',' '/,CI(0:5)/'0','1','2','3','4','5'/
      CHARACTER*4 PR(7)/'UNIF','ENER','MOME','CRIT','CRI2','E+UN',
     &'E+CR'/
      CHARACTER TX*11,FMT1*18/"(1X,A1,' =',F10.2)"/,T1*11/
     &' 1 number  '/
      INTEGER*2 JU(2),NON(7)/4,4,3,2,2,6,4/
      REAL JAC(2,2),FF(2),FJ(2)
      LOGICAL*2 IFN
      COMMON /TRAS/ V(6),Y(26),A(26),P(26),T(26),EN1,ES1,g2,F,F1,
     &TT,PP,g,EN2,ES2,Cu,DY2,N2,IY,ICASE
      N2=25
      IY=13
      EN1=.66666667
      ES1=.5
      EN2=2.*EN1
      ES2=2.*ES1
      WRITE(*,*)' Are ES = 1, or SI=2 units used (Give 1 or 2)'
      READ(*,*) IU
      Cu=1.486
      g=32.2
      IF(IU.EQ.2) THEN
      g=9.81
      Cu=1.
      ENDIF
      g2=2.*G
1     CALL CHTABL
5     WRITE(*,*)' Give the number for the type of Problem'
      DO 10 I=1,7
10    WRITE(*,"(I3,' ',A4)") I,PR(I)
      READ(*,*) ICASE
      TX=T1
      IF(ICASE.GT.5) TX=' 2 numbers '
      WRITE(*,"(' Give',A11,'for the unkn')") TX
      DO 20 I=1,NON(ICASE)
```

```
20    WRITE(*,"(I3,'  ',A1)") I,C(I,ICASE)
      IF(ICASE.GT.5) THEN
      READ(*,*) JU
      ELSE
      READ(*,*) IU
      ENDIF
      WRITE(*,*)' Provide values for known plus',' guess for
     &unknown'
      DO 30 I=1,NON(ICASE)
      WRITE(*,"(1X,A1,' = ',\)") C(I,ICASE)
30    READ(*,*) V(I)
      NCT=0
      IF(ICASE.LT.6) THEN
40    IFN=.TRUE.
      DD=.002*V(IU)
41    CALL EQS(FF)
      IF(IFN) THEN
      F1=F
      V(IU)=V(IU)+DD
      IFN=.FALSE.
      GO TO 41
      ENDIF
      DIF=DD*F1/(F-F1)
      NCT=NCT+1
      V(IU)=V(IU)-DD-DIF
      IF(ABS(DIF).GT. .005 .AND. NCT.LT.15) GO TO 40
      IF(NCT.EQ.15) WRITE(*,*)' Failed to converge',DIF
      ELSE
C  Solves 2 equations
42    CALL EQS(FJ)
      DO 44 I=1,2
      DD=.005*V(JU(I))
      V(JU(I))=V(JU(I))+DD
      CALL EQS(FF)
      DO 43 J=1,2
43    JAC(J,I)=(FF(J)-FJ(J))/DD
44    V(JU(I))=V(JU(I))-DD
      XX=JAC(2,1)/JAC(1,1)
      JAC(2,2)=JAC(2,2)-XX*JAC(1,2)
      FJ(2)=FJ(2)-XX*FJ(1)
      F1=FJ(2)/JAC(2,2)
      V(JU(2))=V(JU(2))-F1
      F=(FJ(1)-F1*JAC(1,2))/JAC(1,1)
      V(JU(1))=V(JU(1))-F
      NCT=NCT+1
      IF(ABS(F1)+ABS(F).GT. .05 .AND. NCT.LT.15) GO TO 42
      IF(NCT.EQ.15) WRITE(*,*)' Failed to converge',F1,F
      ENDIF
      WRITE(*,*)' Solution:'
      DO 50 I=1,NON(ICASE)
```

```
      II=5-IFIX(ALOG10(ABS(V(I)+.1)))
      IF(II.LT.0) II=0
      IF(II.GT.5) II=5
      FMT1(17:17)=CI(II)
50    WRITE(*,FMT1) C(I,ICASE),V(I)
      WRITE(*,*)' Give: 0-stop, ','1-another problem same channel,
     &2-dif. c.'
      READ(*,*) II
      IF(II.LT.1 .OR. II.GT. 2) STOP
      IF(II.EQ.1) GO TO 5
      GO TO 1
      END
      SUBROUTINE EQS(FF)
      COMMON /TRAS/ V(6),Y(26),A(26),P(26),T(26),EN1,ES1,g2,F,
     &F1,TT,PP,g,EN2,ES2,Cu,DY2,N2,IY,ICASE
      REAL FF(2)
      CALL AREA(AA)
      GO TO (1,2,3,4,5,6,7),ICASE
1     F=V(3)*V(2)-Cu*AA*(AA/PP)**EN1*V(4)**ES1
      RETURN
2     F=V(3)-V(1)-(1.+V(4))*(V(2)/AA)**2/g2
      RETURN
3     F=V(3)-V(2)**2/(g*AA)-Ahcn(V(1)-DY2,AA)
      RETURN
4     F=g*AA-TT*(V(2)/AA)**2
      RETURN
5     F=V(2)-V(1)-AA/(2.*TT)
      RETURN
6     FF(1)=V(3)-V(1)-(1.+V(4))*(V(2)/AA)**2/g2
      FF(2)=V(5)*V(2)-Cu*(AA/PP)**EN1*AA*V(6)**ES1
      RETURN
7     FF(1)=V(3)-V(1)-(1.+V(4))*(V(2)/AA)**2/g2
      FF(2)=g*AA-TT*(V(2)/AA)**2
      RETURN
      END
      SUBROUTINE AREA(AA)
      COMMON /TRAS/ V(6),Y(26),A(26),P(26),T(26),EN1,ES1,g2,F,
     &F1,TT,PP,g,EN2,ES2,Cu,DY2,N2,IY,ICASE
      YY=V(1)
      DO 10 WHILE (YY.GT.Y(IY+1) .AND. IY.LE.N2)
10    IY=IY+1
      DO 20 WHILE (YY.LT.Y(IY) .AND. IY.GT.1)
20    IY=IY-1
      IM=IY-1
      IP=IY+1
      FAC=(YY-Y(IY))/(Y(IP)-Y(IY))
      TT=T(IY)+FAC*(T(IP)-T(IY))
      PP=P(IY)+FAC*(P(IP)-P(IY))
      IF(IY.EQ.1) THEN
      AA=A(IY)+FAC*(A(IP)-A(IY))
      ELSE
```

```
      AA=(YY-Y(IY))*(YY-Y(IP))*A(IM)/((Y(IM)-Y(IY))*(Y(IM)-Y(IP)))
     &+(YY-Y(IM))*(YY-Y(IP))*A(IY)/((Y(IY)-Y(IM))*(Y(IY)-Y(IP)))
     &+(YY-Y(IM))*(YY-Y(IY))*A(IP)/((Y(IP)-Y(IM))*(Y(IP)-Y(IY)))
      ENDIF
      RETURN
      END
      SUBROUTINE CHTABL
      LOGICAL*2 LINEAR
      REAL X(20),Z(20)
      COMMON /TRAS/ V(6),Y(26),A(26),P(26),T(26),EN1,ES1,g2,F,
     &F1,TT,PP,g,EN2,ES2,Cu,DY2,N2,IY,ICASE
      N=26
      READ(3,*) NVALUE,IOUT,LINEAR
      IF(NVALUE.GT.20) THEN
      WRITE(*,100)
100   FORMAT(' Program dimension to allow only 20',
     &' coordinate pairs')
      STOP
      ENDIF
      READ(3,*)(X(I),Z(I),I=1,NVALUE)
      ZSTA=Z(1)
      IMIN=1
      YMAX=ZSTA
      DO 10 I=2,NVALUE
      IF(Z(I).GE.ZSTA) GO TO 5
      ZSTA=Z(I)
      IMIN=I
5     IF(Z(I).GT.YMAX) YMAX=Z(I)
10    CONTINUE
      IF(Z(IMIN+1).LT.ZSTA+1.E-5) THEN
      T(1)=X(IMIN+1)-X(IMIN)
      P(1)=T(1)
      IR1=IMIN+1
      XR1=X(IMIN+1)
      ELSE
      T(1)=0.
      P(1)=0.
      IR1=IMIN
      XR1=X(IMIN)
      ENDIF
      XL1=X(IMIN)
      AA=0.
      PP=P(1)
      A(1)=0.
      Y(1)=0.
      IL1=IMIN
      IL=IL1-1
      IR=IR1+1
      DY=(YMAX-ZSTA)/FLOAT(N-1)
      DY2=.5*DY
      DYS=DY*DY
```

```
      ILM=1
      IRM=NVALUE
      IF(.NOT.LINEAR) THEN
      IRM=IRM-1
      ILM=2
      ENDIF
      DO 40 I=2,N
      Y(I)=DY*FLOAT(I-1)
      YZ=Y(I)+ZSTA
      DO 20 WHILE (IR.LT.IRM .AND. Z(IR).LT.YZ)
      IR1=IR
20    IR=IR+1
      DO 30 WHILE (IL.GT.ILM .AND. Z(IL).LT.YZ)
      IL1=IL
30    IL=IL-1
      IF(LINEAR) THEN
      XR=X(IR1)+(YZ-Z(IR1))/(Z(IR)-Z(IR1))*(X(IR)-X(IR1))
      XL=X(IL1)+(YZ-Z(IL1))/(Z(IL)-Z(IL1))*(X(IL)-X(IL1))
      ELSE
      ILP=IL-1
      IRP=IR+1
      XR=(YZ-Z(IR))*(YZ-Z(IRP))*X(IR1)/((Z(IR1)-Z(IR))*(Z(IR1)-
     &Z(IRP)))+(YZ-Z(IR1))*(YZ-Z(IRP))*X(IR)/((Z(IR)-Z(IR1))*
     &(Z(IR)-Z(IRP)))+(YZ-Z(IR1))*(YZ-Z(IR))*X(IRP)/((Z(IRP)-
     &Z(IR1))*(Z(IRP)-Z(IR)))
      XL=(YZ-Z(IL))*(YZ-Z(ILP))*X(IL1)/((Z(IL1)-Z(IL))*(Z(IL1)-
     &Z(ILP)))+(YZ-Z(IL1))*(YZ-Z(ILP))*X(IL)/((Z(IL)-Z(IL1))*
     &(Z(IL)-Z(ILP)))+(YZ-Z(IL1))*(YZ-Z(IL))*X(ILP)/((Z(ILP)-
     &Z(IL1))*(Z(ILP)-Z(IL)))
      ENDIF
      T(I)=XR-XL
      AA=AA+DY*(XR1-XL1)+DY2*(XR-XR1+XL1-XL)
      A(I)=AA
36    PP=PP+SQRT(DYS+(XR-XR1)**2)+SQRT(DYS+(XL1-XL)**2)
      P(I)=PP
      XR1=XR
40    XL1=XL
      WRITE(IOUT,300)
300   FORMAT(' No.    Depth      Area Perimeter   Top Width')
      DO 50 I=1,N
50    WRITE(IOUT,311) I,Y(I),A(I),P(I),T(I)
311   FORMAT(I4,5F10.3)
      RETURN
      END
      FUNCTION Ahcn(YY,AA)
      COMMON /TRAS/ V(6),Y(26),A(26),P(26),T(26),EN1,ES1,g2,F,
     &F1,TT,PP,g,EN2,ES2,Cu,DY2,N2,IY,ICASE
      AASS=0.
      A1=0.
      DO 10 K=1,IY
      AASS=AASS+(YY-Y(K))*(A(K)-A1)
```

```
10    A1=A(K)
      Ahcn=AASS+(V(1)-(V(1)-Y(IY))/2.)*(AA-A1)
      RETURN
      END
```

Program NAT-REG.C (designed to solve steady problem in natural and prismatic channels)

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
 float v[6],y[26],a[26],p[26],t[26],en1,es1,g2,f,f1,tt,pp,g,en2,\
   es2,cu,dy2,dia,b,fm,aa;
 int n2,iy,icase,itype;
 void area(void){float yy,fac,beta; int iy,im,ip;
 yy=v[0];
 if(itype==1){
 while ((yy>y[iy+1]) && (iy<n2)) iy=iy+1;
 while ((yy<y[iy+1]) && (iy>1)) iy=iy-1;
 im=iy-1; ip=iy+1; fac=(yy-y[iy])/(y[ip]-y[iy]); tt=t[iy]+fac*\
   (t[ip]-t[iy]);
 pp=p[iy]+fac*(p[ip]-p[iy]);
 if(iy==1) aa=a[iy]+fac*(a[ip]-a[iy]); else
 aa=(yy-y[iy])*(yy-y[ip])*a[im]/((y[im]-y[iy])*(y[im]-y[ip]))+\
   (yy-y[im])*(yy-y[ip])*a[iy]/((y[iy]-y[im])*(y[iy]-y[ip]))+\
   (yy-y[im])*(yy-y[iy])*a[ip]/((y[ip]-y[im])*(y[ip]-y[iy]));}
 else if(itype==3) {beta=acos(1.-2.*yy/dia); tt=dia*sin(beta);
  pp=beta*dia; aa=.25*dia*dia*(beta-cos(beta)*sin(beta));} else {
 tt=b+2.*fm*yy; pp=b+2.*sqrt(fm*fm+1.)*yy; aa=(b+fm*yy)*yy;}
} // End of area
void chtabl(void){
 const int n=26; int linear,nvalue,imin,ir1,ir,il1,il,irm,ilm,ilp,\
   irp,i,iout;
 float zsta,ymax,dy,dy2,dys,yz,xr,xl,xr1,xl1;
 float x[20],z[20];
 char fnam[20]; FILE *fili,*filo;
 printf("Give name of file that contains input data\n");
   scanf("%s",fnam);
 if((fili=fopen(fnam,"r"))==NULL){printf("File does not exist.n");
   exit(0);}
 fscanf(fili,"%d %d %d",&nvalue,&iout,&linear);
 if(iout!=6){printf("Give name of output file\n");
   scanf("%s",fnam);
 if((filo=fopen(fnam,"w"))==NULL){
   printf("File cannot be opened\n");exit(0);}}
 if(nvalue>20){printf("program dimension to allow only \
   20 coordinate pairs");exit(0);}
 for(i=0;i<nvalue;i++) fscanf(fili,"%f %f",&x[i],&z[i]);
 zsta=z[0]; imin=0; ymax=zsta;
 for(i=1;i<nvalue;i++){if(z[i]<zsta){zsta=z[i];imin=i;}
 if(z[i]>ymax) ymax=z[i];}
```

```
 if(z[imin+1]<zsta+1.e-5){t[0]=x[imin+1]-x[imin];p[0]=t[0];
   ir1=imin+1;xr1=x[imin+1];}
    else {t[0]=0.;p[0]=0.;ir1=imin;xr1=x[imin];}
 xl1=x[imin]; aa=0.; pp=p[0]; a[0]=0.; y[0]=0.; il1=imin;
   il=il1–1; ir=ir1+1;
 dy=(ymax-zsta)/(float)(n-1); dy2=.5*dy; dys=dy*dy; ilm=0;
   irm=nvalue-1;
 if(linear==0){irm=irm-1; ilm=1;}
 for(i=1;i<n;i++){y[i]=dy*(float)i; yz=y[i]+zsta;
  while ((ir<irm)&&(z[ir]<yz)){ir1=ir;ir++;}
  while ((il>ilm)&&(z[il]<yz)){il1=il;il–;}
 if(linear){xr=x[ir1]+(yz-z[ir1])/(z[ir]-z[ir1])*(x[ir]-x[ir1]);
    xl=x[il1]+(yz-z[il1])/(z[il]-z[il1])*(x[il]-x[il1]);} else \
      {ilp=il-1;irp=ir+1;
 xr=(yz-z[ir])*(yz-z[irp])*x[ir1]/((z[ir1]-z[ir])*(z[ir1]-\
   z[irp]))+(yz-z[ir1])*(yz-z[irp])*x[ir]/((z[ir]-z[ir1])\
   *(z[ir]-z[irp]))+(yz-z[ir1])*(yz-z[ir])*x[irp]/\
   ((z[irp]-z[ir1])*(z[irp]-z[ir]));
 xl=(yz-z[il])*(yz-z[ilp])*x[il1]/((z[il1]-z[il])*(z[il1]-\
   z[ilp]))+(yz-z[il1])*(yz-z[ilp])*x[il]/((z[il]-z[il1])\
   *(z[il]-z[ilp]))+(yz-z[il1])*(yz-z[il])*x[ilp]/\
   ((z[ilp]-z[il1])*(z[ilp]-z[il]));}
 t[i]=xr-xl; aa=aa+dy*(xr1-xl1)+dy2*(xr-xr1+xl1-xl); a[i]=aa;
 pp=pp+sqrt(dys+pow(xr-xr1,2))+sqrt(dys+pow(xl1-xl,2)); p[i]=pp;
 xr1=xr; xl1=xl;}
if(iout!=6){fprintf(filo," no. depth area perimeter top width\n");
 for(i=0;i<n;i++)fprintf(filo,"%4d %9.3f %9.3f %9.3f %9.3f\n",i+1,\
   y[i],a[i],p[i],t[i]);fclose(filo);}
fclose(fili);} //End of chtabl
 float ahcn(float yy){float cosb,beta,aass,a1; int k;
 if(itype==2) return ((0.5*b+fm*yy/3.)*yy*yy);
 else if(itype==3){cosb=1.-2.*yy/dia; beta=acos(cosb);
 return.5*dia*(dia*dia/6.*sin(pow(beta,3.)-aa*cosb));} else {
  aass=0.; a1=0.;for(k=0;k<iy;k++) {aass+=(yy-y[k])*(a[k]-a1);
    a1=a[k];}
 return aass+(v[1]-(v[1]-y[iy])/2.)*(aa-a1);}
} // End of ahcn
 void eqs(float *ff){area();
switch (icase){
case 1: f=v[2]*v[1]-cu*aa*pow(aa/pp,en1)*pow(v[3],es1); break;
case 2: f=v[2]-v[0]-(1.+v[3])*pow(v[1]/aa,2.)/g2; break;
case 3: f=v[2]-pow(v[1],2.)/(g*aa)-ahcn(v[0]-dy2); break;
case 4: f=g*aa-tt*pow(v[1]/aa,2.); break;
case 5: f=v[1]-v[0]-aa/(2.*tt); break;
case 6: ff[0]=v[2]-v[0]-(1.+v[3])*pow(v[1]/aa,2.)/g2;
 ff[1]=v[4]*v[1]-cu*pow(aa/pp,en1)*aa*pow(v[5],es1); break;
case 7: ff[0]=v[2]-v[0]-(1.+v[3])*pow(v[1]/aa,2.)/g2;
 ff[1]=g*aa-tt*pow(v[1]/aa,2.);}
} // End of eqs
void main(void){
 float g,dd,dif,xx;
```

```
 int iu,i,j,nct,ifn,ii;
 char *c[]={"YQnS ","YQEK ","YQM ","YQ ","YE ","YQHKnS","YQHK "};
 char ci[6]="012345";
 char *pr[]={"UNIF","ENER","MOME","CRIT","CRI2","E+UN","E+CR"};
 char *fmt1=" %c = %10.2fn"; char tx[11],t1[11]=" 1 number ";
 int ju[2],non[7]={4,4,3,2,2,6,4};
 float jac[2][2],ff[2],fj[2];
 n2=25;
 iy=13;
 en1=.66666667;
 es1=.5;
 en2=2.*en1;
 es2=2.*es1;
printf(" Are ES = 1, or SI=2 units used (give 1 or 2)n");
  scanf("%d",&iu);
 cu=1.486;
 g=32.2;
 if(iu==2){g=9.81;cu=1.;} g2=2.*g;
L1:printf("Give: 1,2 or 3 for type of channel; 1=natural, 2=trap.,\
   3=circle\n");scanf("%d",&itype);
 if(itype==1) chtabl(); else if (itype==3){
   printf(" Give its diameter\n");scanf("%f",&dia);}
 else{printf("Give its bottom width b, and side slope m\n");
   scanf("%f %f",&b,&fm);}
L5:printf("Give the number for the type of problem\n");
 for(i=0;i<7;i++) printf("%3d %4s\n",i+1,pr[i]);
   scanf("%d",&icase);
 strcpy(tx,t1);
 if(icase>5) strcpy(tx," 2 numbers ");
 printf(" Give %s for the unkn\n",tx);
 for(i=0;i<non[icase-1];i++) printf("%3d %c\n",i+1,c[icase-1][i]);
 if(icase>5){scanf("%d %d",&ju[0],&ju[1]);ju[0]–;ju[1]–;}
   else {scanf("%d",&iu);iu–;}
 printf("Provide values for known plus guess for unknown\n");
 for(i=0;i<non[icase-1];i++){printf(" %c = ",c[icase-1][i]);
   scanf("%f",&v[i]);}
 nct=0;
 if(icase<6) {
do {ifn=1; dd=.002*v[iu];
L41:eqs(ff);
 if(ifn) {f1=f; v[iu]+=dd; ifn=0; goto L41;}
 dif=dd*f1/(f-f1); nct++; v[iu]-=(dd+dif);
} while ((fabs(dif)>.005)&& (nct<15));
if(nct==15) printf(" failed to converge %f\n",dif);}
else {
// Solves 2 equations
do {eqs(fj);
 for(i=0;i<2;i++){dd=.005*v[ju[i]]; v[ju[i]]+=dd; eqs(ff);
  for(j=0;j<2;j++) jac[j][i]=(ff[j]-fj[j])/dd; v[ju[i]]-=dd;
 xx=jac[1][0]/jac[0][0]; jac[1][1]-=xx*jac[0][1]; fj[1]-=xx*fj[0];
 f1=fj[1]/jac[1][1]; v[ju[1]]-=f1; f=(fj[0]-f1*jac[0][1])/jac[0][0];
```

```
 v[ju[1]]-=f; nct++;}
} while (((fabs(f1)+fabs(f))>0.05) && (nct<15));}
 if(nct==15)printf(" failed to converge %f %f\n",f1,f);
printf(" Solution:n");
for(i=0;i<non[icase-1];i++){ii=5-(int)log10(fabs(v[i]+.10));
 if(ii<0) ii=0; if(ii>5) ii=5; fmt1[10]=ci[ii];
printf(fmt1, c[icase-1][i],v[i]);}
printf(" Give: 0-stop,1-another problem same channel,2-dif. ");
scanf("%d",&ii);
 if((ii<1)||(ii>2)) exit(0);
 if(ii==1) goto L5; goto L1;
}
```

As an exercise in using program NATURAL, assume it is desired to find the uniform flow rate corresponding to a normal depth of $Y_o = 6$ in the natural channel whose geometry is defined by the data in the listing of program CHTABL and whose Manning's roughness coefficient is n = 0.03, and whose bottom slope is $S_o = 0.001$. Assume the data as listed in CHTABL is stored on the file NATURAL.DAT. To obtain the solution the first response after executing the program is to type 1 for ES units. Next select 1 to select UNIF as the type of problem, and the third response is 2 to select Q as the unknown. Finally the following values are given to the variables of the problem: Y = 6, Q = 300, n = 0.03 and S = 0.001, and NATURAL produces the solution for Q = 607.5 cfs when using quadratic interpolation to for the look-up table, and Q = 628.4 cfs when using linear interpolation of the input channel geometry pairs of x and z values.

5.3.4 Hand Solution to GVF-Flow by the Standard Step Method

A solution to Equation 4.8 can be accomplished by hand computations, or in keeping with the computer age by using a spreadsheet. The solution procedures involves the following steps: (1) Obtaining a tabular solution that gives the areas and wetted perimeters as a function of the depth at positions along the channel using a technique such as described in the previous section. (2) Start at some position in the river, or natural channel, where the depth of flow is known as some function of the flow rate. The solution will proceed upstream from this position. The sum of the depth and the elevation of the channel bottom are called the river stage, or $h = z_{bot} + Y$. At this starting position, the stage h is known and placed at the top of the table that will be used to compute stages at other stations. (3) The velocity head is computed from the known flow rate, and the area corresponding to this known stage, and this velocity head is added to the stage to give the elevation of the energy line or a value to the total head. The wetted perimeter is also determined at this station. (4) A guess for the stage h at the next upstream station is made based on sound judgment. (5) Based on this guessed value of the stage, the area and wetted perimeter for the upstream station are determined. (6) The average slope of the energy line between the two stations is computed. This average slope, which will be denoted as $\overline{S}_f$ is commonly taken as the average of the values of S_f at the two stations, or $\overline{S}_f = (S_{fi} + S_{fi+1})/2$. (7) This average value of the slope of the energy line, $\overline{S}_f$ is multiplied by the length between the two stations and added to the beginning station's total head to determine the total head at the next station to establish another value for total head at the next upstream station. (8) This second computed total head is compared with the guessed value, stage plus velocity head (i.e., the first total head), and steps (4) through (8) are repeated until the two total heads are in close enough agreement. When the agreement is satisfactory, the process is repeated for the next upstream station.

This process is known as the standard step method, probably because it is commonly used over the entire length between consecutive stations where cross sections for the river are known. It does involve the above described iteration, or trial and error process, and it could easily be applied to

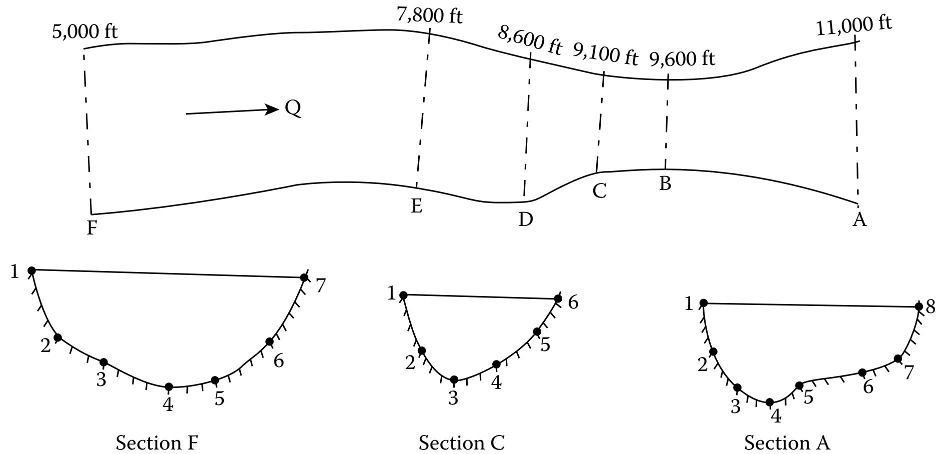


FIGURE 5.2 Plan view of a small river between position x = 5,000 ft and x = 11,000 ft.

positions between stations where the geometry of the river are known by using interpolation of the cross-sectional data at the two stations. Using shorter distances would generally improve the accuracy of the GVF solution, especially when stations for which cross-sectional data are available only at long distances apart.

The standard step method can be understood best by following through the above steps in a small example. Figure 5.2 gives the plan view of a river from position 5,000 to 11,000 ft. Along this reach of river, there are six sections, identified as A, B, C, D, E, and F, for which surveys have been made that give the river cross sections. Cross sections A, C, and F are shown in Figure 5.2, and Table 5.1 gives the x_d distance from the right bank (when looking up stream) and the elevation of the river bottom as pairs for all six sections. Using the methods described above, these data have been used to provide the areas and wetted perimeters as a function of the depths that are contained in Table 5.2. Some values not needed for the solution have been removed in making up Table 5.2.

A spreadsheet solution (using EXCEL, program RIVER1.XLS.) is shown in Table 5.3 for the above river flow problem. The spreadsheet contains values for the flow rate Q, Manning's n, the acceleration of gravity, and the constant C in Manning's equation. Whenever these values are referred to in equations, they are made global variables (by preceding both their row and column with a $). This spreadsheet is designed so that the user enters the values for the **Position**, **Stage**, **Depth**, **Area, and Perimeter**. The column for the **Vel. Head** is computed by dividing the flow rate

TABLE 5.1
Cross-Section Data for the Six Stations along the River Shown in Figure 5.2

	Section A 11,000 ft		Section B 9,600 ft		Section C 9,100 ft		Section D 8,600 ft		Section E 7,800 ft		Section F 5,000 ft	
Point No.	x_p (ft)	z (ft)	x_p (ft)	z (ft)	x_p (ft)	z (ft)	x_p (ft)	z (ft)	x_p (ft)	z (ft)	x_p (ft)	z (ft)
1	0	505.0	0	506.8	0	508.7	0	508.9	0	509.0	0	511.8
2	2.4	500.0	3.0	504.4	3.0	502.3	4.0	504.1	4.2	504.0	4.0	505.0
3	6.0	497.1	6.0	500.2	7.0	499.8	8.0	502.0	13.0	501.9	12.6	502.8
4	12.0	495.8	12.0	501.0	13.5	500.7	12.5	500.7	20.0	500.9	20.0	501.1
5	16.0	498.0	18.5	505.2	20.0	503.6	20.3	503.8	28.8	503.5	29.0	502.7
6	24.0	498.5	21.0	506.0	24.0	508.7	38.0	508.9	36.8	505.5	37.0	505.0
7	32.0	500.5	23.8	506.8					42.0	509.0	44.2	511.8
8	34.0	505.0										

TABLE 5.2
Tables Giving Areas, A and Perimeter, P as a Function of the Depth Y for the Six Sections of River

Section A
No. = 1, x = 11000 ft, El. = 495.8 ft

No.	Y	A	P
1	0.000	0.000	0.000
2	0.368	0.436	2.502
.	.	.	.
19	6.624	130.630	36.294
20	6.992	142.328	37.105
21	7.360	154.152	37.916
22	7.728	166.101	38.727
23	8.096	178.174	39.537
24	8.464	190.373	40.348
25	8.832	202.698	41.159
26	9.200	215.147	41.970

Section B
No. = 2, x = 9600 ft, El. = 500.2 ft

No.	Y	A	P
1	0.000	0.000	0.000
2	0.264	0.286	2.322
.	.	.	.
22	5.544	59.972	22.882
23	5.808	65.109	24.173
24	6.072	70.564	25.557
25	6.336	76.351	26.940
26	6.600	82.469	28.324

Section C
No. = 3, x = 9100 ft, El. = 499.8 ft

No.	Y	A	P
1	0.000	0.000	0.000
2	0.356	0.559	3.267
.	.	.	.
22	7.476	114.894	28.535
23	7.832	122.882	29.381
24	8.188	131.029	30.226
25	8.544	139.335	31.072
26	8.900	147.800	31.918

Section D
No. = 4, x = 8600 ft, El. = 500.7 ft

No.	Y	A	P
1	0.000	0.000	0.000
2	0.328	0.322	2.070
.	.	.	.
23	7.216	128.668	37.417
24	7.544	139.974	39.028
25	7.872	151.744	40.640
26	8.200	163.976	42.252

Section E
No. = 5, x = 7800 ft, El. = 500.7 ft

No.	Y	A	P
1	0.000	0.000	0.000
2	0.324	0.545	3.435
.	.	.	.
23	7.128	182.080	43.288
24	7.452	195.078	44.291
25	7.776	208.319	45.295
26	8.100	221.805	46.298

Section F
No. = 6, x = 5000 ft, El. = 501.1 ft

No.	Y	A	P
1	0.000	0.000	0.000
2	0.428	0.914	4.357
.	.	.	.
18	7.276	189.832	42.711
19	7.704	206.487	43.831
20	8.132	223.443	44.950
21	8.560	240.702	46.070
22	8.988	258.261	47.190
23	9.416	276.123	48.310
24	9.844	294.286	49.430
25	10.272	312.751	50.550
26	10.700	331.518	51.670

by the area squaring this and then dividing the result by 2 g. This velocity head is added to the Stage to get the next column of total head, **T.Head**. The next column for $\mathbf{S_f}$ is computed using Manning's equation. The first row is then copied into the second row, and the column **Av. Sf** is obtained as the average of the previous columns from rows 1 and 2, the **Length** obtained as the difference in **position** values from rows 1 and 2. By multiplying this length by the average slope of the energy line (**Av. Sf**), the head loss column **hf** is obtained and this headloss is added to the **T.Head** in the last column in line 1 to get the value for **T.Head** in the last column of line 2. (Actually because of the heading line 1 is line 4 in the spreadsheet.) The processes used in the spreadsheet is to repeatedly enter values for **Stage**, h, **Depth**, Y, **Area**, A, and **Perimeter**, P until the **T.Head** obtained as the sum of the stage and velocity head equals the last column of **T.Head** obtained by adding the headloss to the previous stage. The values for A, P, and h are determined by linear interpolation (done by hand) from the entries in the table for that section in Table 5.2.

You should use the above discussion as a guide for developing a spreadsheet solution, but your spreadsheet can contain a number of variations and enhancements over the one shown above. Notice that the numbers in the lower portion of this spreadsheet give the bottom elevations of the sections

TABLE 5.3
Spreadsheet Solution to Previous GVF-Profile in the River of Figure 5.1 (EXCEL RIVER1.XLS)

Q = 500, n = 0.032, g = 32.2, 2 g = 64.4, Cu = 1.486

Position	Stage	Depth	Area	Perimeter	Vel. Head	T.Head	Sf	Av. Sf	Length	hf	T.Head
11,000	503	7.2	149.0111304	37.5633913	0.174830308	503.1748303	0.000831424				503.175
9,600	506.544	63.44	76.53639394	26.98193939	0.662701798	507.2067018	0.004928751	0.002880087	1400	4.032122364	507.2071224
9,100	508.439	8.639	141.5939185	31.29775843	0.193626564	508.6326266	0.0007728	0.002850775	500	1.425387712	508.6325101
8,600	508.86	8.16	162.4842927	42.05541463	0.147038605	509.0070386	0.000724308	0.000748554	500	0.374276829	509.0067869
7,800	509.325	8.425	235.3326235	47.30409568	0.070095457	509.3950955	0.000246484	0.000485396	800	0.388316681	509.3951036
5,000	509.949	8.849	252.5584276	46.82626168	0.060859785	510.0098598	0.000192152	0.000219318	2800	0.614089913	510.0091935
495.8	6.992	142.328	37.105	7.36	154.152	37.916					
500.2	6.336	76.351	26.94	6.6	82.469	28.324					
499.8	8.544	139.335	31.072	8.9	147.8	31.918					
500.7	7.872	151.744	40.64	8.2	163.976	42.252					
500.9	7.776	208.319	45.295	8.1	221.805	46.298					
501.1	8.56	240.702	46.07	8.988	258.261	47.19					

in column A, and then in columns B and E, the two depths that are anticipated to bracket the depth that will occur at this sections. The first columns after the depths are the corresponding areas and the second columns thereafter contain the corresponding perimeters. These partial tables of depth versus area and perimeters are used for linear interpolation to determine the areas and perimeters in the above portion of the spreadsheet corresponding to the depths in column C. (In this spreadsheet, the depth of 8.425 at Position 7800 is greater than 8.1 ft, the largest value obtained in the previous table for Section E, and therefore extrapolation occurs in finding the area and perimeter at this position.) If the smallest z of 500.9 ft is the top of the river bank, then the water would flow over it bank top at this position. The idea used in the above spreadsheet is that you repeatedly try values in column 2 for the stage until the two columns for "T. Head" are the same, and the equations defining the columns carry out the computations as soon as a new value is entered. After you are satisfied, the same process is continued for the next line. It should not be hard for you to figure out what the formulas are that compute the values in columns C through L. Once the correct formulas are defined for Row 4, they can be copied in Row 5, etc. until the GVF-Solution is completed.

After the two **T.Head**s agree close enough, the third line is added. This is done by "copying" line 2 into line 3. The same process is completed for line 3, that is stages (or depths) are repeatedly tried until the two **T.Head**s are nearly equal. There is little justification in making these values equal to 0.01 ft or less since the computations are over such long distances and therefore not very precise. The fourth line is next copied from the third line, etc. until the final station is arrived at, which in this problem is at position 5000 ft.

The above spreadsheet could be significantly improved by storing the tables giving the areas and perimeters as a function of depth, or better yet let the spreadsheet compute the values in these tables. Instead of having to enter areas and perimeters, these could then be obtained from the given depth by interpolation. The user could then rapidly try different values of depth to satisfy the total head requirements. An alternative would be to write a computer program that implements the above procedure. While such a computer program is certainly a viable means for solving GVF problems in irregular sections, a better approach is to use an ODE solver as was done in solving GVF problems in regular channels whose geometric properties can be defined by algebraic equations. The next section describes this approach.

5.3.5 Using an ODE Solver to Compute GVF-Profile in Irregular Channels

In Chapter 4, it was shown that an alternative to solving the ODE for gradually varied flow that contains dY/dx was to solve the equation that gives dE/dx, i.e., solve Equation 4.8. When dealing with a channel with irregular cross sections, the approach will be modified by using a "table look-up" to supply values for the needed variables rather than solving these from equations. Since "table look-up" implies that tables of relationships are available, these tables will need to be generated internally in the program as needed.

A computer program that solves the GVF problems in irregular channels will need to contain the following components or if you prefer thinking in terms of tasks, the following tasks:

(1) A main program that defines the problem to be solved by reading in problem specification and then calls on the differential equation solver appropriately. (2) A subroutine that defines the ODE that is to be solved. This subroutine will be called by the differential equation solver as needed. (3) A subroutine or statements that obtains the depth corresponding to the given specific energy E. This component is needed since E is considered the dependent variable in Equation 4.8, yet the area, wetted perimeter, etc. are dependent upon the depth Y. These three components are needed in a program that solves GVF problems in channels whose geometric properties are defined by equations, and these components for irregular channels need only minor modifications to those that apply for regular channels. (4) A subroutine that reads in data that defines the geometry of the channel and generates from this data a table of values that gives the area, A, wetted perimeter, P, and the specific energy, E (for the given flow rate), as a function of depth, Y. (5) A subroutine that

appropriately interpolates values from this table. Also this interpolation subroutine will need to interpolate between tables that apply at two consecutive sections so that correct values for variables can be obtained for any x.

Below listings of a FORTRAN and a C program are given that solve GVF problems in irregular channels. These programs calls on the differential equation solver ODESOL. With minor changes, the FORTRAN program could utilize the solver DVERK, or RUKUST or another differential equation solver, and the C program could use RUKUST.

FORTRAN program GVFNAT.FOR designed to solve GVF problem in natural channels

```
      PARAMETER (N=26)
      REAL SE(1),DE(1),XP(1),YP(1,1),WK1(1,13)
      CHARACTER*2 UNIT
      LOGICAL*2 LINEAR
      EXTERNAL DEX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRANS/ QN,FN,SO
      COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),XSTA1,XSTA2,
     &ZSTA1,YY,ZSTA2,PSTA,ASTA,Q2G,LINEAR,IP1,IP2,IR,IR1,IL,IL1,
     &IOTAB
      WRITE(*,*)'GIVE:UNIT,IOUT,IOTAB,NSTAT,TOL,DELX,YB,Q'',FN,
     &XBEG,XEND'
      READ(*,*) UNIT,IOUT,IOTAB,NSTAT,TOL,DELX,YB,Q,FN,XBEG,XEND
      LINEAR=.TRUE.
      H1=-.1
      HMIN=1.E-4
      CC=1.486
      G=32.2
      IF(UNIT.EQ.'SI' .OR. UNIT.EQ.'si') THEN
      CC=1.
      G=9.81
      ENDIF
      G2=2.*G
      QN=(FN*Q/CC)**2
      Q2G=Q*Q/G2
      XS=XBEG
      CALL CHTABL(1,1)
      CALL CHTABL(2,2)
      NSTA=2
      SO=(ZSTA2-ZSTA1)/(XSTA1-XSTA2)
      YY=YB
      IP1=N/2
      IP2=N/2
      CALL INTERP(XS,YY,.TRUE.)
      SE(1)=YY+Q2G/ASTA**2
      WRITE(IOUT,15) Q,FN
15    FORMAT(' Solution to Gradually Varied Flow in',' natural
     &channel with',/,' Q =',F10.2,' n =',F7.4,/,1X,71('-'),/,
     &6X,'X',11X,'Y',9X,'E',' Perimeter Area',' Velocity Bot.
     &Slope',/,1X,71('-'))
      WRITE(IOUT,40) XS,YY,SE(1),PSTA,ASTA,Q/ASTA,SO
40    FORMAT(1X,6F10.2,F10.7)
```

```
20    XZ=XS+DELX
      CALL ODESOL(SE,DE,1,XS,XZ,TOL,H1,HMIN,1,XP,YP,WK1,DEX)
      WRITE(IOUT,40) XZ,YY,SE(1),PSTA,ASTA,Q/ASTA,SO
      XS=XZ
      IF(XZ.LE.XSTA2 .AND. NSTA.LT.NSTAT) THEN
      DO 30 I=1,N
      Y(I,1)=Y(I,2)
      A(I,1)=A(I,2)
      P(I,1)=P(I,2)
30    E(I,1)=E(I,2)
      ZSTA1=ZSTA2
      XSTA1=XSTA2
      IP1=IP2
      NSTA=NSTA+1
      CALL CHTABL(NSTA,2)
      SO=(ZSTA2-ZSTA1)/(XSTA1-XSTA2)
      ENDIF
      IF(XZ.GT.XEND) GO TO 20
      STOP
      END
      SUBROUTINE DEX(X,SE,EPRIME)
      PARAMETER (N=26)
      REAL SE(1),EPRIME(1)
      LOGICAL*2 LINEAR
      COMMON /TRANS/ QN,FN,SO
      COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),
     &XSTA1,XSTA2,ZSTA1,YY,ZSTA2,PSTA,ASTA,Q2G,LINEAR,
     &IP1,IP2,IR,IR1,IL,IL1,IOTAB
      CALL INTERP(X,SE(1),.FALSE.)
      SF=QN*(ABS(PSTA/ASTA)**.666666667/ASTA)**2
      EPRIME(1)=SO-SF
      RETURN
      END
      SUBROUTINE CHTABL(NOSEC,M)
      PARAMETER (N=26)
      LOGICAL*2 LINEAR,IBOTSC
      REAL X(20),H(20)
      COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),
     &XSTA1,XSTA2,ZSTA1,YY,ZSTA2,PSTA,ASTA,Q2G,LINEAR,
     &IP1,IP2,IR,IR1,IL,IL1,IOTAB
      READ(3,*) NSTA,NVALUE,XSTA,LINEAR
      IF(M.EQ.1) THEN
      XSTA1=XSTA
      ELSE
      XSTA2=XSTA
      ENDIF
      IF(NVALUE.GT.20) THEN
      WRITE(*,100)
100   FORMAT(' Program dimension to allow only 20',' coordinate
     &pairs')
      STOP
```

```
      ENDIF
      IF(NSTA.NE.NOSEC) THEN
      WRITE(*,110) NSTA,NOSEC
110   FORMAT(' Next station with data is',I4,/,
     &' Expecting data for station',I5)
      STOP
      ENDIF
      READ(3,*)(X(I),H(I),I=1,NVALUE)
      ZSTA=H(1)
      IMIN=1
      YMAX=ZSTA
      DO 10 I=2,NVALUE
      IF(H(I).GE.ZSTA) GO TO 5
      ZSTA=H(I)
      IMIN=I
5     IF(H(I).GT.YMAX) YMAX=H(I)
10    CONTINUE
      IF(H(IMIN+1).LT.ZSTA+1.E-5) THEN
      T(1,M)=X(IMIN+1)-X(IMIN)
      P(1,M)=T(1,M)
      IR1=IMIN+1
      XR1=X(IMIN+1)
      ELSE
      T(1,M)=0.
      P(1,M)=0.
      IR1=IMIN
      XR1=X(IMIN)
      ENDIF
      XL1=X(IMIN)
      AA=0.
      PP=P(1,M)
      A(1,M)=0.
      Y(1,M)=0.
      IL1=IMIN
      IL=IL1-1
      IR=IR1+1
      DY=(YMAX-ZSTA)/FLOAT(N-1)
      DY2=.5*DY
      DYS=DY*DY
      ILM=1
      IRM=NVALUE
      IF(.NOT.LINEAR) THEN
      IRM=IRM-1
      ILM=2
      ENDIF
      IF(M.EQ.1) THEN
      ZSTA1=ZSTA
      ELSE
      ZSTA2=ZSTA
      ENDIF
      DO 40 I=2,N
```

```
      Y(I,M)=DY*FLOAT(I-1)
      YZ=Y(I,M)+ZSTA
      DO 20 WHILE (IR.LT.IRM .AND. H(IR).LT.YZ)
      IR1=IR
20    IR=IR+1
      DO 30 WHILE (IL.GT.ILM .AND. H(IL).LT.YZ)
      IL1=IL
30    IL=IL-1
      IF(LINEAR) THEN
      XR=X(IR1)+(YZ-H(IR1))/(H(IR)-H(IR1))*(X(IR)-X(IR1))
      XL=X(IL1)+(YZ-H(IL1))/(H(IL)-H(IL1))*(X(IL)-X(IL1))
      ELSE
      ILP=IL-1
      IRP=IR+1
      XR=(YZ-H(IR))*(YZ-H(IRP))*X(IR1)/((H(IR1)-H(IR))*(H(IR1)-
     &H(IRP)))+(YZ-H(IR1))*(YZ-H(IRP))*X(IR)/((H(IR)-H(IR1))*
     &(H(IR)-H(IRP)))+(YZ-H(IR1))*(YZ-H(IR))*X(IRP)/((H(IRP)-
     &H(IR1))*(H(IRP)-H(IR)))
      XL=(YZ-H(IL))*(YZ-H(ILP))*X(IL1)/((H(IL1)-H(IL))*(H(IL1)-
     &H(ILP)))+(YZ-H(IL1))*(YZ-H(ILP))*X(IL)/((H(IL)-H(IL1))*
     &(H(IL)-H(ILP)))+(YZ-H(IL1))*(YZ-H(IL))*X(ILP)/((H(ILP)-
     &H(IL1))*(H(ILP)-H(IL)))
      ENDIF
      T(I,M)=XR-XL
      AA=AA+DY*(XR1-XL1)+DY2*(XR-XR1+XL1-XL)
      A(I,M)=AA
      IF(AA.LT. 1.E-5) GO TO 35
      VS=Q2G/AA**2
      IF(VS.GT. 5.*Y(I,M)) GO TO 35
      E(I,M)=Y(I,M)+VS
      GO TO 36
35    E(I,M)=0.
36    PP=PP+SQRT(DYS+(XR-XR1)**2)+SQRT(DYS+(XL1-XL)**2)
      P(I,M)=PP
      XR1=XR
40    XL1=XL
      IF(IOTAB.GT.0) THEN
      IOTT=IOTAB
      IBOTSC=.FALSE.
      IF(IOTAB.GT.20) THEN
      IOTT=IOTAB-20
      IBOTSC=.TRUE.
      ENDIF
      IF(IBOTSC) WRITE(*,312) NSTA,XSTA,ZSTA
      WRITE(IOTT,312) NSTA,XSTA,ZSTA
312   FORMAT(/,' Station no.=',I4,' Position, x =',F10.1,' Elev.
     &Bot=',F8.2,/,' No.      Depth Area Perimeter Top Width S.',
     &' Energy',/,1X,55('-'))
      DO 310 I=1,N
      IF(IBOTSC) WRITE(*,311) I,Y(I,M),A(I,M),P(I,M),T(I,M),E(I,M)
310   WRITE(IOTT,311) I,Y(I,M),A(I,M),P(I,M),T(I,M),E(I,M)
```

```
311   FORMAT(I4,5F10.3)
      IF(IOTAB.EQ.6 .OR.IBOTSC) READ(*,*)
      ENDIF
      RETURN
      END
      SUBROUTINE INTERP(X,SE,KNWY)
      PARAMETER (N=26)
      LOGICAL*2 LINEAR,KNWY,TWICE
      COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),
     &XSTA1,XSTA2,ZSTA1,YY,ZSTA2,PSTA,ASTA,Q2G,LINEAR,
     &IP1,IP2,IR,IR1,IL,IL1,IOTAB
      NCT=0
      TWICE=.FALSE.
      FAX=(X-XSTA1)/(XSTA2-XSTA1)
      IF(FAX.GT. .99) FAX=1.
      IF(FAX.LT. .01) FAX=0.
      JP1=IP1-1
      JP2=IP2-1
10    IF(IP1.GE.N .OR. Y(IP1,1).GE.YY) GO TO 20
      JP1=IP1
      IP1=IP1+1
      GO TO 10
20    IF(IP1.LT.3 .OR. Y(JP1,1).LE.YY) GO TO 30
      IP1=IP1-1
      JP1=IP1-1
      GO TO 20
30    IF(FAX.LT.1.E-5) GO TO 50
      IF(IP2.GE.N .OR. Y(IP2,2).GE.YY) GO TO 40
      JP2=IP2
      IP2=IP2+1
      GO TO 30
40    IF(IP2.LT.3 .OR. Y(JP2,2).LE.YY) GO TO 50
      IP2=IP2-1
      JP2=IP2-1
      GO TO 40
50    FA1=(YY-Y(JP1,1))/(Y(IP1,1)-Y(JP1,1))
      FA2=(YY-Y(JP2,2))/(Y(IP2,2)-Y(JP2,2))
      ASTA=(1.-FAX)*(A(JP1,1)+FA1*(A(IP1,1)-A(JP1,1)))+FAX*
     &(A(JP2,2)+FA2*(A(IP2,2)-A(JP2,2)))
      PSTA=(1.-FAX)*(P(JP1,1)+FA1*(P(IP1,1)-P(JP1,1)))+FAX*
     &(P(JP2,2)+FA2*(P(IP2,2)-P(JP2,2)))
      IF(KNWY) GO TO 70
      F=SE-YY-Q2G/ASTA**2
      IF(TWICE) GO TO 60
      TWICE=.TRUE.
      F1=F
      YY=YY-.01
      GO TO 50
60    DIF=.01*F1/(F1-F)
      NCT=NCT+1
      TWICE=.FALSE.
```

```
      YY=YY+.01-DIF
      IF(NCT.LT.10 .AND. ABS(DIF).GT. .00001) GO TO 10
      IF(NCT.EQ.10) WRITE(*,80) DIF,YY,SE,Q2G
80    FORMAT(' Failed to converge, DIF=',E10.4,' YY=',3E10.4)
70    RETURN
      END
```

Program GVFNATN.C

```
//This program allows n to vary linearly between stations. Now n
//is input with station and not in the original data.
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include "odesolc.h"
 const int n=25;
 float qn,fn[2],so;
 float y[26][2],t[26][2],a[26][2],p[26][2],e[26][2],xsta1,xsta2,\
   zsta1,yy,zsta2,psta,asta,fna,q2g;
 int linear,ip1,ip2,ir,ir1,il,il1,iotab; char fnam[20];
 FILE *fili,*filo,*fil1;
 void interp(float x,float se,int knwy){int linear,twice,nct,jp1,jp2;
  float fax,fa1,fa2,f,f1,dif;
 fax=(x-xsta1)/(xsta2-xsta1); nct=0;
 if(fax> .99) fax=1.;
 if(fax< .01) fax=0.;
 jp1=ip1–1;
 jp2=ip2–1;
do{twice=1;
 while ((ip1<n) && (yy>y[ip1][0])){jp1=ip1;ip1++;}
 while ((ip1>1) && (yy<y[jp1][0])){ip1=jp1;jp1--;}
 if(fax<1.e-5) goto L50;
 while ((ip2<n) && (yy>y[ip2][1])){jp2=ip2;ip2++;}
 while ((ip2>1) && (yy<y[jp2][1])){ip2=jp2;jp2--;}
L50: fa1=(yy-y[jp1][0])/(y[ip1][0]-y[jp1][0]);
 fa2=(yy-y[jp2][1])/(y[ip2][1]-y[jp2][1]);
 asta=(1.-fax)*(a[jp1][0]+fa1*(a[ip1][0]-a[jp1][0]))+fax*\
   (a[jp2][1]+fa2*(a[ip2][1]-a[jp2][1]));
 psta=(1.-fax)*(p[jp1][0]+fa1*(p[ip1][0]-p[jp1][0]))+fax*\
   (p[jp2][1]+fa2*(p[ip2][1]-p[jp2][1]));
 fna=(1.-fax)*fn[0]+fax*fn[1];
 if(knwy) return;
 f=se-yy-q2g/(asta*asta);
 if(twice){
 twice=0;
 f1=f;
 yy=yy-.01;
 goto L50;}
 dif=.01*f1/(f1-f);
 nct++;
 yy=yy+.01-dif;
}while ((nct<10) && (fabs(dif)> .0001));
```

```
 if(nct==10) printf(" Failed to converge, dif= %10.4e \
   yy= %10.4e %10.4e\n",dif,yy,se,q2g);
} //End of interp
 void slope(float x,float *se,float *eprime){
 interp(x,*se,0);
 eprime[0]=so-qn*pow(fna*pow(fabs(psta/asta),.666666667)/asta,2.);
} // End of dex
void chtabl(int nosec,int m){
 int nvalue,imin,irm,ilm,ilp,irp,i,nsta;
 float xsta,zsta,ymax,dy,dy2,dys,yz,xr,xl,xr1,xl1,aa,pp,vs;
 float x[20],h[20];
 fscanf(fili,"%d %d %f %f %d",&nsta,&nvalue,&xsta,&fn[m],&linear);
 if(m) xsta2=xsta; else xsta1=xsta;
 if(nvalue>20){printf("program dimension to allow only \
   20 coordinate pairs");exit(0);}
 if(nsta!=nosec){printf("Next station with data is %4d\n Expecting\
   data for station
%4d",nsta,nosec);exit(0);}
 for(i=0;i<nvalue;i++) fscanf(fili,"%f %f",&x[i],&h[i]);
 zsta=h[0]; imin=0; ymax=zsta;
 for(i=1;i<nvalue;i++){if(h[i]<zsta){zsta=h[i];imin=i;}
 if(h[i]>ymax) ymax=h[i];}
 if(h[imin+1]<zsta+1.e-5){t[0][m]=x[imin+1]-x[imin];p[0][m]=t[0][m];
   ir1=imin+1;}
    else {t[0][m]=0.;p[0][m]=0.;ir1=imin;}
 xr1=x[ir1];xl1=x[imin]; aa=0.; pp=p[0][m]; a[0][m]=0.; y[0][m]=0.;
   il1=imin; il=il1-1; ir=ir1+1;
 dy=(ymax-zsta)/(float)n; dy2=.5*dy; dys=dy*dy; ilm=0;
   irm=nvalue-1;
 if(linear==0){irm=irm-1; ilm=1;}
 if(m) zsta2=zsta; else zsta1=zsta;
 for(i=1;i<=n;i++){y[i][m]=dy*(float)i; yz=y[i][m]+zsta;
  while ((ir<irm)&&(h[ir]<yz)){ir1=ir;ir++;}
  while ((il>ilm)&&(h[il]<yz)){il1=il;il--;}
 if(linear){xr=x[ir1]+(yz-h[ir1])/(h[ir]-h[ir1])*(x[ir]-x[ir1]);
    xl=x[il1]+(yz-h[il1])/(h[il]-h[il1])*(x[il]-x[il1]);}
      else {ilp=il-1;irp=ir+1;
 xr=(yz-h[ir])*(yz-h[irp])*x[ir1]/((h[ir1]-h[ir])*(h[ir1]-\
   h[irp]))+(yz-h[ir1])*(yz-h[irp])*x[ir]/\
 ((h[ir]-h[ir1])*(h[ir]-h[irp]))+(yz-h[ir1])*(yz-h[ir])*x[irp]/\
   ((h[irp]-h[ir1])*(h[irp]-h[ir]));
 xl=(yz-h[il])*(yz-h[ilp])*x[il1]/((h[il1]-h[il])*(h[il1]-\
   h[ilp]))+(yz-h[il1])*(yz-h[ilp])*x[il]/\
 ((h[il]-h[il1])*(h[il]-h[ilp]))+(yz-h[il1])*(yz-h[il])*x[ilp]/\
   ((h[ilp]-h[il1])*(h[ilp]-h[il]));}
 t[i][m]=xr-xl; aa=aa+dy*(xr1-xl1)+dy2*(xr-xr1+xl1-xl); a[i][m]=aa;
 if(aa>1.e-4){vs=q2g/(aa*aa); if(vs<5.*y[i][m]) e[i][m]=y[i][m]+vs;
   else e[i][m]=0.;}
 pp=pp+sqrt(dys+pow(xr-xr1,2))+sqrt(dys+pow(xl1-xl,2));
   p[i][m]=pp;
```

```
 xr1=xr; xl1=xl;}
 if(iotab>0) {fprintf(filo,"Station no.= %4d Position, x=%10.1f \
   Elev. bot=%8.2f\n",nsta,xsta,zsta);
 fprintf(filo," No. Depth Area Perimeter Top Width S. Energy\n \
------------------------------------------------------\n");
 for(i=0;i<=n;i++)fprintf(filo,"%4d  %9.3f  %9.3f  %9.3f  %9.3f
%9.3f\n",i+1,y[i][m],a[i][m],p[i][m],t[i][m],e[i][m]);}
} //End of chtabl
void main(void){
 float se[1],de[1],xp[1],yp[1][1],wk1[1][13],tol,delx,yb,q,xbeg,\
   xend,hy,hmin,cc,g,g2,xs,h1,xz;
 char unit[2];
 char *fmt40="%10.2f %9.2f %9.2f %9.2f %9.2f %9.2f %9.7f %6.4f\n";
 int nstat,nsta,nvalue,i;
 printf("Give name of file that contains input data\n");
   scanf("%s",fnam);
 if((fili=fopen(fnam,"r"))==NULL){printf("File does not exist.\n");
   exit(0);}
 printf("Give name of output file\n"); scanf("%s",fnam);
 if((fil1=fopen(fnam,"w"))==NULL){printf("File cannot be opened\n");
   exit(0);}
 printf("Give: UNIT,IOTAB,NSTAT,TOL,DELX,YB,Q,XBEG,XEND\n");
 scanf("%s %d %d %f %f %f %f %f %f",&unit,&iotab,&nstat,&tol,&delx,\
   &yb,&q,&xbeg,&xend);
if(iotab){printf("Give name of file cross-section data\n");
  scanf("%s",fnam);
 if((filo=fopen(fnam,"w"))==NULL){printf("File cannot be opened\n");
   exit(0);}}
    h1=-.1; hmin=1.e-4; cc=1.486; g=32.2;
  if((unit=="SI") || (unit=="si")) {cc=1.; g=9.81;}
  g2=2.*g; qn=pow(q/cc,2.); q2g=q*q/g2; xs=xbeg; chtabl(1,0);
    chtabl(2,1);
  nsta=2; so=(zsta2-zsta1)/(xsta1-xsta2); yy=yb; ip1=n/2; ip2=n/2;
  interp(xs,yy,1);
  se[0]=yy+q2g/(asta*asta);
  fprintf(fil1," Solution to gradually varied flow in natural\
    channel with\n q = %10.2f\n \
-----------------------------------------------------------\n\
     x  Y  E   Perimeter   Area   Velocity Bot.   slope   n\n\
-----------------------------------------------------------\n",q);
  fprintf(fil1,fmt40,xs,yy,se[0],psta,asta,q/asta,so,fn[0]);
 do{xz=xs+delx;
  odesolc(se,xs,xz,tol,h1,hmin,1);
  fprintf(fil1,fmt40,xz,yy,se[0],psta,asta,q/asta,so,fna);
  xs=xz;
  if((xz<=xsta2) && (nsta<nstat)) {
 for(i=0;i<n;i++){y[i][0]=y[i][1]; a[i][0]=a[i][1];
   p[i][0]=p[i][1]; e[i][0]=e[i][1];}
  fn[0]=fn[1]; zsta1=zsta2; xsta1=xsta2; ip1=ip2; nsta=nsta+1;
  chtabl(nsta,1); so=(zsta2-zsta1)/(xsta1-xsta2);}
```

```
}while (xz>xend); fclose(fili); fclose(fil1);
  if(iotab) fclose(filo);
}
```

File odesolc.h

```
extern void odesolc(float *y,float x1,float x2,float err,float h1,\
   float hmin,int stor);
int ngood=0,nbad=0,nbetw=0,ibetw=0,nstor=1,nv=1;
float dxbetw=0.000001;
```

Input variables to program

`UNIT`: Should be given as "`ES` for English units, or "`SI`" for international units.
`IOUT`: Is the logical unit for solution output. If 6 solution is printed on terminal. Another value will result in a filename being requested. Logical unit 3 is used for input so 3 cannot be given.
`IOTAB`: Logical unit for station geometry tables to be written to. If 0 no such tables written. If 6 tables will be written to screen and held there until a key is pressed.
`NSTAT`: Is the number of input stations for which cross-sectional data will be given. Data for this geometry will be in the file whose logical unit is `IOUT`.
`TOL`: Error parameter for ODESOL to use.
`DELX`: Increment between printed solution values of GVF-profile. This value will generally be negative because solution is moving upstream starting at downstream station.
`YB`: Beginning value for depth Y at starting station.
`Q`: Flow rate.
`FN`: Value of Manning's n.
`XBEG`: beginning x position for GVF-solution, i.e. starting x.
`XEND`: Ending x for GVF-solution.

In the FORTRAN listing, the main program occurs first, followed by subroutine DEX. Subroutine DEX defines the derivative dE/dx for the differential equation solver ODESOL. This task is accomplished by calling Subroutine INTERP passing to it the position X and specific energy SE (supplied as the dependent variable to DEX by ODESOL). X is used by INTERP to interpolate properly between sections where tables are stored that gives the relationship of area, perimeter, and specific energy to depth, and SE gives INTERP the specific energy for which the corresponding area, ASTA, and wetted perimeter, PSTA, are desired. (The STA within these FORTRAN variable names denote station, or section x.) The values of ASTA and PSTA are returned to subroutine DEX in the named COMMON/INF/ and DEX uses these values in the Manning's equation to compute S_f (SF is the FORTRAN variable name) for which dE/dx (EPRIME is the FORTRAN variable) is computed.

Subroutine CHTABL, which occurs third in the listing, reads in cross-sectional data that defines the channel geometry at given sections, and from this data creates the tables that establish the relationship between the depth, Y, and area, A, perimeter, P, top width, T and the specific energy E. These tables are for constant increments of depth Y. Subroutine CHTABL is almost identical to the program whose listing was given previously. Since two tables that provide this data at two consecutive input sections are needed for interpolation, the arrays that store this data now have two subscripts, the first of which is dimensioned to N = 26 and the second to 2 for the two tables. Thus the two tables giving the needed relationships between variables are Y(N,2), T(N,2), A(N,2), P(N,2), and E(N,2), and these are contained in the named COMMON/INF/. When the second subscript for these arrays is 1, values from the first section are provided, and when the second subscript is 2, values from the second section are provided. The second argument M in CHTABL determines whether data will be stored for the first or second section, e.g., M is the second subscript in the above arrays in CHTABL when the data is computed and stored in them. The first argument of CHTABL is the section number and is used to verify that the correct input data is being read for each new section. The main program calls CHTABL twice, telling it to create tables for sections # 1 and # 2.

Subroutine INTERP does the two-way interpolation needed both within individual section geometry tables, and between these tables. The third argument passed to INTERP is a logical variable (KNWY), which when true, tells INTERP that the depth Y is known. When KNWY is false, then INTERP is told that the depth Y should be determined that corresponds to the given values of specific energy, SE, which is passed as the second argument. The Newton method is used to solve for the depth Y when SE is given. Thus this subroutine performs tasks (3) and (5) described above, depending upon whether KNWY is false or true, respectively.

The main program carries out the additional task of checking whether the position x for which a solution is to be obtained lies between the sections for which tables are stored in memory. When X becomes less than the position of the section for which the second table is stored, then the arrays for Y, A, P, and E with the second subscript 2 are copied into these arrays with the second subscript 1, and CHTABL is called to fill in new values with the second subscript 2, i.e., table 2 becomes table 1 and a new table 2 is computed that applies for the next upstream section. The slope of the channel bottom is also updated to represent the slope between these new sections.

The use of the above program will be illustrated by solving the problem of the GVF in the small river shown in Figure 5.2. The solution to this problem will be obtained and written to an output table at an increment of $\Delta x = 200$ ft between the downstream section at x = 11,000 ft and the upstream section where x = 5,000 ft Thus the input to the program for DELX will be −200 and XBEG will be given a value of 11,000 and XEND, a value of 5,000. Manning's n equals 0.032 and therefore FN will be given a value of 0.032. A value of 0.00001 is appropriate for TOL. The first line of input read by this program to solve this problem thus is:

```
'ES' 2 4 6 .00001 -200 7.2 500 .032 11000 5000
```

The program interpolates linearly between sections in determining the needed variables, but it allows for either linear or quadratic interpolation of the original input data for the cross sections in generating the tables that define the relationship between depth, area, wetted perimeter, and specific energy. Linear interpolation will be used here.

The input data consist of the following and the file that contains this data will be requested when the program is ready to read this data:

```
1 8 11000 .TRUE.
0 505. 2.4 500 6 497.1 12 495.8 16 498 24 498.5 32 500.5 34 505
2 7 9600 .TRUE.
0 506.8 3 504.4 6 500.2 12 501.0 18.5 505.2 21 506. 23.8 506.8
3 6 9100 .TRUE.
0 508.7 3 502.3 7 499.8 13.5 500.7 20 503.6 24 508.7
4 6 8600 .TRUE.
0 508.9 4 504.1 8 502 12.5 500.7 20.3 503.8 38 508.9
5 7 7800  .TRUE.
0 509. 4.2 504. 13 501.9 20 500.9 28.8 503.5 36.8 505.5 42 509
6 7 5000 .TRUE.
0 511.8 4 505 12.6 502.8 20 501.1 29 502.7 37 505 44.2 511.8
```

This input consists of six pairs of two lines each for the six input sections A through F shown in Figure 5.2. The first line of each pair contains (1) the number for the station starting with 1 for the downstream station and increasing consecutively by one for each new station, (2) the number of pairs of values given on the second line of each pair, (3) the x position of this section, and (4) the logical variable .TRUE. to indicate that linear interpolation is to be used. The second line of each pair gives the pairs of data (x_b, z) that define the cross sections. These are the values in Table 5.1.

The solution to this problem consists of the following table. Note there are differences between this solution and that obtained using the spreadsheet that implements the standard step method. The major source of these differences is associated with solving the slope of the energy line more frequently. According to Manning's equation S_f varies inversely as the area to the 10/3 power, and with the wetted perimeter to the 4/3 power. Thus computing S_f on a smaller increment will produce different changes in depth, etc. than when these computations are done only at the input sections and an average value used. This occurs even though linear interpolation is used between the input sections to determine the area and perimeter at each position x.

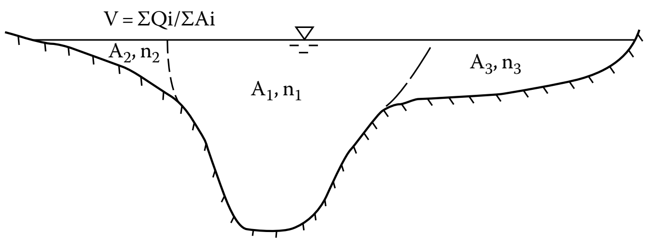
Solution to Gradually Varied Flow in Natural Channel with Q = 500.00 n = .0320

X	Y	E	Perimeter	Area	Velocity	Bot. Slope
11000.00	7.20	7.37	37.56	149.01	3.36	0.0031429
10800.00	6.72	6.96	35.39	126.38	3.96	0.0031429
10600.00	6.34	6.66	33.13	108.34	4.62	0.0031429
10400.00	6.10	6.52	31.06	95.50	5.24	0.0031429
10200.00	6.03	6.54	29.45	87.67	5.70	0.0031429
10000.00	6.11	6.66	28.38	83.35	6.00	0.0031429
9800.00	6.27	6.86	27.81	80.98	6.17	0.0031429
9600.00	6.49	7.10	27.71	79.75	6.27	0.0031429
9400.00	7.48	7.82	31.14	107.51	4.65	0.0008000
9200.00	8.03	8.28	31.02	124.85	4.00	0.0008000
9000.00	8.43	8.63	30.77	136.33	3.67	0.0008000
8800.00	8.26	8.43	37.66	152.63	3.28	0.0018000
8600.00	8.07	8.22	41.57	158.78	3.15	0.0018000
8400.00	8.18	8.30	43.22	178.41	2.80	0.0002500
8200.00	8.25	8.35	44.59	196.52	2.54	0.0002500
8000.00	8.29	8.37	45.80	213.64	2.34	0.0002500
7800.00	8.31	8.38	46.91	230.06	2.17	0.0002500
7600.00	8.35	8.42	46.92	231.68	2.16	0.0000714
7400.00	8.38	8.46	46.93	233.23	2.14	0.0000714
7200.00	8.42	8.49	46.92	234.73	2.13	0.0000714
7000.00	8.46	8.53	46.91	236.18	2.12	0.0000714
6800.00	8.49	8.56	46.90	237.59	2.10	0.0000714
6600.00	8.52	8.59	46.88	238.94	2.09	0.0000714
6400.00	8.56	8.62	46.86	240.25	2.08	0.0000714
6200.00	8.59	8.65	46.83	241.53	2.07	0.0000714
6000.00	8.62	8.68	46.79	242.77	2.06	0.0000714
5800.00	8.65	8.71	46.76	243.98	2.05	0.0000714
5600.00	8.68	8.74	46.71	245.15	2.04	0.0000714
5400.00	8.70	8.77	46.67	246.29	2.03	0.0000714
5200.00	8.73	8.80	46.62	247.40	2.02	0.0000714
5000.00	8.76	8.82	46.57	248.47	2.01	0.0000714

5.3.6 Multiple Parallel Channels and Flood Plain Storage

Most larger rivers store water outside of their normal channel during times of flood. Flow may occur in area outside of the normal channel, but when it does the roughness coefficient will be larger generally than the main channel. Problems of this nature can be handled by assuming that the irregular section consists of multiple parts (generally a right and left bank channel and the main central channel), each of which have different roughness coefficients, but flow in all component channels is caused by the same sloping energy line. For such channels, the velocity head will be modified by a kinetic energy correction coefficient, α. Thus the position of the mean energy lines will be at a distance $\alpha V^2/(2\,g)$ above the mean river stage at any position. V is the average velocity obtained by dividing the total flow rate occurring in the main channel plus the side channels by the total area of main plus side channels, or

$$V = \frac{\sum Q_i}{\sum A_i} \tag{5.4}$$



The kinetic energy correction coefficient is defined as the actual kinetic energy flux accounting for the velocity distribution that occurs at a cross section divided by the kinetic energy flux computed by average values. The kinetic energy flux computed from average values equals $(V^2/2\,g)Q$ and the actual kinetic energy flux is obtained by integrating the point velocity distribution cubed over the cross-sectional area, and dividing by $2\,g$. Thus α is given by

$$\alpha = \frac{\int v^3 dA}{AV^3} \tag{5.5}$$

In computing the kinetic energy correction coefficient for multiple connected parallel channels, the integral is replaced by a summation, or

$$\alpha = \frac{\sum \left\{ v_i^3 A_i \right\}}{AV^3} \tag{5.6}$$

where v_i and A_i are the velocities and areas of the separate channels. For a situation with a main channel and two side channels, the summation in Equation 5.6 will be from channels 1 through 3. Equation 5.6 cannot be used to compute α because the separate velocities in the different channel are not known. What is known is the roughness coefficients and areas for all channels. Therefore Equation 5.6 needs to be manipulated so that α can be computed from these known quantities. Replacing the individual velocities v_i by Q_i/A_i and V by Equation 5.4 allows Equation 5.6 to be rewritten as

$$\alpha = \frac{\sum \left\{ (Q_i/A_i)^3 A_i \right\}}{\left(\sum Q_i\right)^3 \Big/ \left(\sum A_i\right)^2} = \frac{\sum (Q_i^3/A_i^2)\left(\sum A_i\right)^2}{\left(\sum Q_i\right)^3} \tag{5.7}$$

Since the slope of the energy line S_f will be the same in all channels and this is obtained from the uniform flow equation (Manning's equation) it can be noted that

$$\sqrt{S_f} = \frac{\{n_1Q_1P_1^{2/3}\}}{(C_uA_1^{5/3})} = \frac{\{n_2Q_2P_2^{2/3}\}}{(C_uA_2^{5/3})} = \frac{\{n_3Q_3P_3^{2/3}\}}{(C_uA_3^{5/3})} = \frac{\{nQP^{2/3}\}}{(C_uA^{5/3})}$$

where subscripts 1, 2, and 3 are for the three separate channels, and variables without a subscript represent these variables for the total combined channels. Defining the conveyance as $K = C_uA^{5/3}/(nP^{2/3})$ allows the last equation to be written as

$$\frac{Q_1}{K_1} = \frac{Q_2}{K_2} = \frac{Q_3}{K_3} = \frac{Q}{K} = S_f^{1/2}$$

This equation allows Equation 5.7 to be written as (since $Q = K\sqrt{S_fQ}$)

$$\alpha = \frac{A^2}{\left(\sum K_i\right)^3}\sum\frac{K_i^3}{A_i^2} \tag{5.8}$$

which is in a form that permits the kinetic energy coefficient to be computed from known information, i.e., the geometries of the component channels and their roughness coefficients.

The procedures previously described can be readily adapted to connected parallel channels by simply adjusting velocity heads by means of Equation 5.8. When using the standard step method implemented in the spreadsheet, the modification is to first compute α and then multiple the velocity head by α. When using an ODE solver, the modifications to the computer program consist of adding the necessary code to evaluate α from Equation 5.8 and then including this value in the specific energy equation, or

$$E = Y + \alpha\left(\frac{V^2}{2g}\right) = Y + \alpha\left\{\frac{(Q/A)^2}{2g}\right\} \tag{5.9}$$

In order to use Equation 5.8, it is necessary that the data given to define the cross section of the irregular channel be divided into parts so that the individual areas of the different roughness channels can be computed, and from these, the individual conveyance values computed.

5.3.7 Implementation of Solution for Compound Channel

In this section, methods for obtaining computer solutions in channels that have right and left flood plain areas, as well as the main channel, are described. The description deals with the changes that are required for the previous computer program GVFNAT. The additional considerations beyond that used in the previous program can be outlined as follows (As you read these you will want to examine the program listing GVFNATM.FOR.):

1. The program will need to read separate n values for each channel division (main, left, and right channels). A homework problem requests that GVFNAT be modified to allow Manning's n to vary from station to station, and interpolate n for the position x between stations. If n is to vary between stations as well as for the three-component parts of the cross section at a station, then FN should be a two dimensional array; one subscript for the two stations and the other to 3 for the 3 divisions at any stations.
2. The program will need to read x positions where the divisions of the channel at every input station change from the left to the main channel and the main channel to the right side. These values are XLD = x between the left side and the main, and XRD = x between the

main channel and the right side. These x positions should be judged appropriately recognizing that as the flow expands into the side channels more of the flow will behave as if the channel's roughness is closer to that given the main channel than the side channels. Thus XLD will generally be less than where the left channel first actually begins, and XRD will be greater than where the right channel first begins.

3. In making up tables that give A, P, and T as a function of Y (and/or E) it will be necessary to determine separate areas, A_{main}, A_{left}, and A_{right} and separate wetted perimeters P_{main}, P_{left}, P_{right}, when depths get above where side channels receive water. From these and the n values of the individual channels portions, individual conveyances, K_i will need to be computed. $K_i = C_u A_i^{5/3} \big/ \left(n_i P_i^{2/3}\right)$.
4. The energy correction coefficient α will need to be evaluated at each input station, and this value will need to be interpolated for the x as the solution proceeds. This energy correction value at any x will be used in the specific energy equation, or $E = Y + \alpha V^2/(2g)$.

By comparing listing GVFNATM with GVFNAT, you will note the following changes (mainly additions):

1. The array FN(2,3) ALPHA(N,2) (for α) RKS(N,2) (for K), are added to common blocks, The variable FNA is also added to the common block for the value of n at any position x. Also XMID, XLD, XRD, and QSTA (for the fraction of the total flow in the main channel) are added to common. The two-dimensional array RKS contains the conveyance of the main channel divided by the sum of conveyances in the three channel. Thus it actually provides the fraction of the flow in the main channel to the total flow. This quantity is used because the program computes the slope of the energy line S_f using the main channel, and to do this, it needs a value that gives the fraction of the total flow in the main channel because the solution is for a total flow rate.
2. In the main program when station 2 values are transferred into station 1 values ALPHA and RKS arrays are also included.
3. In the subroutine CHTABL (FN(M,I),I=1,3),XLD,XRD are added to the READ statement for the information preceding the x, z coordinate pairs.
4. A number of statements are added to subroutine CHTABL to perform the added tasks described above. Most of these added statements are at the end of this subroutine.
5. The subroutine INTERP contains added statements that interpolates QSTA from the two-dimensional array RKS, and this QSTR is used in computing the slope of the energy line S_f in subroutine DEX that supplies the value of dE/dx for whatever position the ODE solver ODESOL passes via the variable argument X. In other words, S_f is solved using Q_1 (the flow rate in the main channel) and $QSTA = K_1/(K_1 + K_2 + K_3)$.

Program Listing GVFNATM.FOR (Handles compound section of natural channel)

```
C This program handles the case of a channel divided into:
C (1) a main channel (2) a left side, and (3) a right side.
C n is different in each portion of section
C and can vary linearly between stations.
      PARAMETER (N=26)
      REAL SE(1),DE(1),XP(1),YP(1,1),WK1(1,13)
      CHARACTER*2 UNIT
      LOGICAL*2 LINEAR
      EXTERNAL DEX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRANS/ QN,FN(2,3),SO
      COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),ALPA(N,2)
```

```
     &,RKS(N,2),CC,XSTA1,XSTA2,ZSTA1,YY,ZSTA2,PSTA,ASTA,FNA
     &,Q2G,XMID,XLD,XRD,QSTA,LINEAR,IP1,IP2,IR,IR1,IL,IL1,IOTAB
      WRITE(*,*)' GIVE: UNIT,IOUT,IOTAB,NSTAT,TOL,DELX,YB,',
     &'Q,XBEG,XEND'
      READ(*,*) UNIT,IOUT,IOTAB,NSTAT,TOL,DELX,YB,Q,XBEG,XEND
      LINEAR=.TRUE.
      H1=-.1
      HMIN=1.E-4
      CC=1.486
      G=32.2
      IF(UNIT.EQ.'SI' .OR. UNIT.EQ.'si') THEN
      CC=1.
      G=9.81
      ENDIF
      G2=2.*G
      QN=(Q/CC)**2
      Q2G=Q*Q/G2
      XS=XBEG
      CALL CHTABL(1,1)
      CALL CHTABL(2,2)
      NSTA=2
      SO=(ZSTA2-ZSTA1)/(XSTA1-XSTA2)
      YY=YB
      IP1=N/2
      IP2=N/2
      CALL INTERP(XS,YY,.TRUE.)
      SE(1)=YY+Q2G/ASTA**2
      WRITE(IOUT,15) Q
15    FORMAT(' Solution to Gradually Varied Flow in natural',
     &'channel with',/,' Q =',F10.2,/,1X,87('-'),/,4X,'X',
     &9X,'Y',7X,'E',' Per. Area Velocity Bot. S. n-main n-left',
     &' n-right Qm/Qt',/,1X,87('-'))
      WRITE(IOUT,40) XS,YY,SE(1),PSTA,ASTA,Q/ASTA,SO,(FN(1,I),
     &I=1,3),QSTA
40    FORMAT(1X,F8.0,5F8.2,F10.7,4F7.4)
20    XZ=XS+DELX
      CALL ODESOL(SE,DE,1,XS,XZ,TOL,H1,HMIN,1,XP,YP,WK1,DEX)
      WRITE(IOUT,40) XZ,YY,SE(1),PSTA,ASTA,Q/ASTA,SO,FNA,QSTA
      XS=XZ
      IF(XZ.LE.XSTA2 .AND. NSTA.LT.NSTAT) THEN
      DO 30 I=1,N
      Y(I,1)=Y(I,2)
      A(I,1)=A(I,2)
      P(I,1)=P(I,2)
      ALPA(I,1)=ALPA(I,2)
      RKS(I,1)=RKS(I,2)
30    E(I,1)=E(I,2)
      DO 31 I=1,3
31    FN(1,I)=FN(2,I)
      ZSTA1=ZSTA2
      XSTA1=XSTA2
```

```
      IP1=IP2
      NSTA=NSTA+1
      CALL CHTABL(NSTA,2)
      SO=(ZSTA2-ZSTA1)/(XSTA1-XSTA2)
      ENDIF
      IF(XZ.GT.XEND) GO TO 20
      STOP
      END
      SUBROUTINE DEX(X,SE,EPRIME)
      PARAMETER (N=26)
      REAL SE(1),EPRIME(1)
      LOGICAL*2 LINEAR
      COMMON /TRANS/ QN,FN(2,3),SO
      COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),ALPA(N,2)
     &,RKS(N,2),CC,XSTA1,XSTA2,ZSTA1,YY,ZSTA2,PSTA,ASTA,FNA
     &,Q2G,XMID,XLD,XRD,QSTA,LINEAR,IP1,IP2,IR,IR1,IL,IL1,IOTAB
      CALL INTERP(X,SE(1),.FALSE.)
      SF=QN*(QSTA*FNA*ABS(PSTA/ASTA)**.666666667/ASTA)**2
      EPRIME(1)=SO-SF
      RETURN
      END
      SUBROUTINE CHTABL(NOSEC,M)
      PARAMETER (N=26)
      LOGICAL*2 LINEAR,IBOTSC
      REAL X(20),H(20)
      COMMON /TRANS/ QN,FN(2,3),SO
      COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),ALPA(N,2)
     &,RKS(N,2),CC,XSTA1,XSTA2,ZSTA1,YY,ZSTA2,PSTA,ASTA,FNA
     &,Q2G,XMID,XLD,XRD,QSTA,LINEAR,IP1,IP2,IR,IR1,IL,IL1,IOTAB
      READ(3,*) NSTA,NVALUE,XSTA,(FN(M,I),I=1,3),XLD,XRD,LINEAR
C NSTA-Station No. starting at downstream end with 1, NVALUE-No.
C of pts to define sec., XSTA-x position of station, n-main,n-left,
C n-right, XLD-x at start of left channel, XRD-x at start of right
C channel, LINEAR .true. if linear interpolation, .false. otherwise
      IF(M.EQ.1) THEN
      XSTA1=XSTA
      ELSE
      XSTA2=XSTA
      ENDIF
      IF(NVALUE.GT.20) THEN
      WRITE(*,100)
100   FORMAT(' Program dimension to allow only 20',
     &' coordinate pairs')
      STOP
      ENDIF
      IF(IOTAB.GT.0) THEN
      IOTT=IOTAB
      IBOTSC=.FALSE.
      IF(IOTAB.GT.20) THEN
      IOTT=IOTAB-20
      IBOTSC=.TRUE.
```

```
      ENDIF
      ENDIF
      IF(NSTA.NE.NOSEC) THEN
      WRITE(*,110) NSTA,NOSEC
110   FORMAT(' Next station with data is',I4,/,' Expecting data
     &for station',I5)
      STOP
      ENDIF
      AL=.001
      AR=.001
      PL=0.
      PR=0.
      FK2=0.
      FK3=0.
      READ(3,*)(X(I),H(I),I=1,NVALUE)
      ZSTA=H(1)
      IMIN=1
      YMAX=ZSTA
      DO 10 I=2,NVALUE
      IF(H(I).GE.ZSTA) GO TO 5
      ZSTA=H(I)
      IMIN=I
5     IF(H(I).GT.YMAX) YMAX=H(I)
10    CONTINUE
      IF(H(IMIN+1).LT.ZSTA+1.E-5) THEN
      T(1,M)=X(IMIN+1)-X(IMIN)
      XMID=.5*(X(IMIN+1)+X(IMIN))
      P(1,M)=T(1,M)
      IR1=IMIN+1
      XR1=X(IMIN+1)
      ELSE
      T(1,M)=0.
      P(1,M)=0.
      IR1=IMIN
      XR1=X(IMIN)
      XMID=XR1
      ENDIF
      XL1=X(IMIN)
      AA=0.
      PP=P(1,M)
      A(1,M)=0.
      Y(1,M)=0.
      IL1=IMIN
      IL=IL1-1
      IR=IR1+1
      DY=(YMAX-ZSTA)/FLOAT(N-1)
      DY2=.5*DY
      DYS=DY*DY
      ILM=1
      IRM=NVALUE
      IF(.NOT.LINEAR) THEN
```

```
      IRM=IRM-1
      ILM=2
      ENDIF
      IF(M.EQ.1) THEN
      ZSTA1=ZSTA
      ELSE
      ZSTA2=ZSTA
      ENDIF
      IF(IBOTSC) WRITE(*,312) NSTA,XSTA,ZSTA,(FN(M,I),I=1,3)
      WRITE(IOTT,312) NSTA,XSTA,ZSTA,(FN(M,I),I=1,3)
312   FORMAT(/,' Station no.=',I4,' x =',F10.1,' Elev. Bot=',
     &F8.2,' n''s=',3F7.4,/,' No. Depth Amain Aleft Aright  A ',
     &'Pmain Pleft Pright P Top S-En. alpha',/,1X,&92('-'))
      DO 40 I=2,N
      Y(I,M)=DY*FLOAT(I-1)
      YZ=Y(I,M)+ZSTA
      DO 20 WHILE (IR.LT.IRM .AND. H(IR).LT.YZ)
      IR1=IR
20    IR=IR+1
      DO 30 WHILE (IL.GT.ILM .AND. H(IL).LT.YZ)
      IL1=IL
30    IL=IL-1
      IF(LINEAR) THEN
      XR=X(IR1)+(YZ-H(IR1))/(H(IR)-H(IR1))*(X(IR)-X(IR1))
      XL=X(IL1)+(YZ-H(IL1))/(H(IL)-H(IL1))*(X(IL)-X(IL1))
      ELSE
      ILP=IL-1
      IRP=IR+1
      XR=(YZ-H(IR))*(YZ-H(IRP))*X(IR1)/((H(IR1)-H(IR))*(H(IR1)-
     &H(IRP)))+(YZ-H(IR1))*(YZ-H(IRP))*X(IR)/((H(IR)-H(IR1))*
     &(H(IR)-H(IRP)))+(YZ-H(IR1))*(YZ-H(IR))*X(IRP)/((H(IRP)-
     &H(IR1))*(H(IRP)-H(IR)))
      XL=(YZ-H(IL))*(YZ-H(ILP))*X(IL1)/((H(IL1)-H(IL))*(H(IL1)-
     &H(ILP)))+(YZ-H(IL1))*(YZ-H(ILP))*X(IL)/((H(IL)-H(IL1))*
     &(H(IL)-H(ILP)))+(YZ-H(IL1))*(YZ-H(IL))*X(ILP)/((H(ILP)-
     &H(IL1))*(H(ILP)-H(IL)))
      ENDIF
      T(I,M)=XR-XL
      IF(XL.GE.XLD .AND. XR.LE.XRD) THEN
      AA=AA+DY*(XR1-XL1)+DY2*(XR-XR1+XL1-XL)
      PP=PP+SQRT(DYS+(XR-XR1)**2)+SQRT(DYS+(XL1-XL)**2)
      PT=PP
      AT=AA
      ALPA(I,M)=1.
      RKS(I,M)=1.
      ELSE
      IF(XL.LT.XLD) THEN
      AL=AL+DY*(XLD-XL1)+DY2*(XL1-XL)
      PL=PL+SQRT(DYS+(XL1-XL)**2)
      AA=AA+DY*(XMID-XLD)
      ELSE
```

```
      PP=PP+SQRT(DYS+(XL1-XL)**2)
      AA=AA+DY*(XMID-XL1)+DY2*(XL1-XL)
      ENDIF
      IF(XR.GT.XRD) THEN
      AR=AR+DY*(XR1-XRD)+DY2*(XR-XR1)
      PR=PR+SQRT(DYS+(XR-XR1)**2)
      AA=AA+DY*(XRD-XMID)
      ELSE
      PP=PP+SQRT(DYS+(XR-XR1)**2)
      AA=AA+DY*(XR1-XMID)+DY2*(XR-XR1)
      ENDIF
      AT=AA+AL+AR
      PT=PP+PL+PR
      FK1=CC*AA*(AA/PP)**.66666667/FN(M,1)
      IF(AL.LT. 0.) AL=ABS(AL)
      IF(AR.LT. 0.) AR=ABS(AR)
      IF(PL.GT. .001) FK2=CC*AL*(AL/PL)**.66666667/FN(M,2)
      IF(PR.GT. .001) FK3=CC*AR*(AR/PR)**.66666667/FN(M,3)
      ALPA(I,M)=AT*AT*(FK1*(FK1/AA)**2+FK2*(FK2/AL)**2+FK3*
     &(FK3/AR)**2)/(FK1+FK2+FK3)**3
      RKS(I,M)=FK1/(FK1+FK2+FK3)
      ENDIF
      IF(AA.LT. 1.E-5) GO TO 35
      VS=Q2G/AT**2
      IF(VS.GT. 5.*Y(I,M)) GO TO 35
      E(I,M)=Y(I,M)+VS*ALPA(I,M)
      GO TO 36
35    E(I,M)=0.
36    P(I,M)=PP
      A(I,M)=AA
      XR1=XR
      XL1=XL
      IF(IBOTSC) WRITE(*,311)I,Y(I,M),AA,AL,AR,AT,PP,PL,PR,PT,
     &T(I,M),E(I,M),ALPA(I,M)
40    WRITE(IOTT,311) I,Y(I,M),AA,AL,AR,AT,PP,PL,PR,PT,T(I,M),
     &E(I,M),ALPHA(I,M),ALPA(I,M)
311   FORMAT(I3,F7.3,4F8.2,4F7.2,3F7.3)
      IF(IOTAB.EQ.6 .OR.IBOTSC) READ(*,*)
      RETURN
      END
      SUBROUTINE INTERP(X,SE,KNWY)
      PARAMETER (N=26)
      LOGICAL*2 LINEAR,KNWY,TWICE
      COMMON /TRANS/ QN,FN(2,3),SO
      COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),ALPA(N,2)
     &,RKS(N,2),CC,XSTA1,XSTA2,ZSTA1,YY,ZSTA2,PSTA,ASTA,FNA
     &,Q2G,XMID,XLD,XRD,QSTA,LINEAR,IP1,IP2,IR,IR1,IL,IL1,IOTAB
      NCT=0
      TWICE=.FALSE.
      FAX=(X-XSTA1)/(XSTA2-XSTA1)
      IF(FAX.GT. .99) FAX=1.
```

```
      IF(FAX.LT. .01) FAX=0.
      JP1=IP1-1
      JP2=IP2-1
10    IF(IP1.GE.N .OR. Y(IP1,1).GE.YY) GO TO 20
      JP1=IP1
      IP1=IP1+1
      GO TO 10
20    IF(IP1.LT.3 .OR. Y(JP1,1).LE.YY) GO TO 30
      IP1=IP1-1
      JP1=IP1-1
      GO TO 20
30    IF(FAX.LT.1.E-5) GO TO 50
      IF(IP2.GE.N .OR. Y(IP2,2).GE.YY) GO TO 40
      JP2=IP2
      IP2=IP2+1
      GO TO 30
40    IF(IP2.LT.3 .OR. Y(JP2,2).LE.YY) GO TO 50
      IP2=IP2-1
      JP2=IP2-1
      GO TO 40
50    FA1=(YY-Y(JP1,1))/(Y(IP1,1)-Y(JP1,1))
      FA2=(YY-Y(JP2,2))/(Y(IP2,2)-Y(JP2,2))
      ASTA=(1.-FAX)*(A(JP1,1)+FA1*(A(IP1,1)-A(JP1,1)))+FAX*
     &(A(JP2,2)+FA2*(A(IP2,2)-A(JP2,2)))
      PSTA=(1.-FAX)*(P(JP1,1)+FA1*(P(IP1,1)-P(JP1,1)))+FAX*
     &(P(JP2,2)+FA2*(P(IP2,2)-P(JP2,2)))
      QSTA=(1.-FAX)*(RKS(JP1,1)+FA1*(RKS(IP1,1)-RKS(JP1,1)))
     &+FAX*(RKS(JP2,2)+FA2*(RKS(IP2,2)-RKS(JP2,2)))
52    FNA=(1.-FAX)*FN(1,1)+FAX*FN(2,1)
      IF(KNWY) GO TO 70
      F=SE-YY-Q2G/ASTA**2
      IF(TWICE) GO TO 60
      TWICE=.TRUE.
      F1=F
      YY=YY-.01
      GO TO 50
60    DIF=.01*F1/(F1-F)
      NCT=NCT+1
      TWICE=.FALSE.
      YY=YY+.01-DIF
      IF(NCT.LT.10 .AND. ABS(DIF).GT. .00001) GO TO 10
      IF(NCT.EQ.10) WRITE(*,80) DIF,YY,SE,Q2G
80    FORMAT(' Failed to converge, DIF=',E10.4,' YY=',3E10.4)
70    RETURN
      END
```

An example of a problem involving a natural channel that consists of a main channel plus a left and right side when the depth of flow becomes larger is used to illustrate the type of information needed, how this is prepared so GVFNATM can solve the problem, and what the solution consists of. The cross-sectional data for this channel are provided below at stations 2000 ft apart. The downstream station is as 10,000 ft, and the upstream station is at position 0, and the steady state gradually varied flow profile in this channel is to be determined for specified flow rates and downstream depths.

The first step in solving the problem is to prepare the data for the six cross sections on a file. For this problem, this data file will consist of the following:

from keyboard

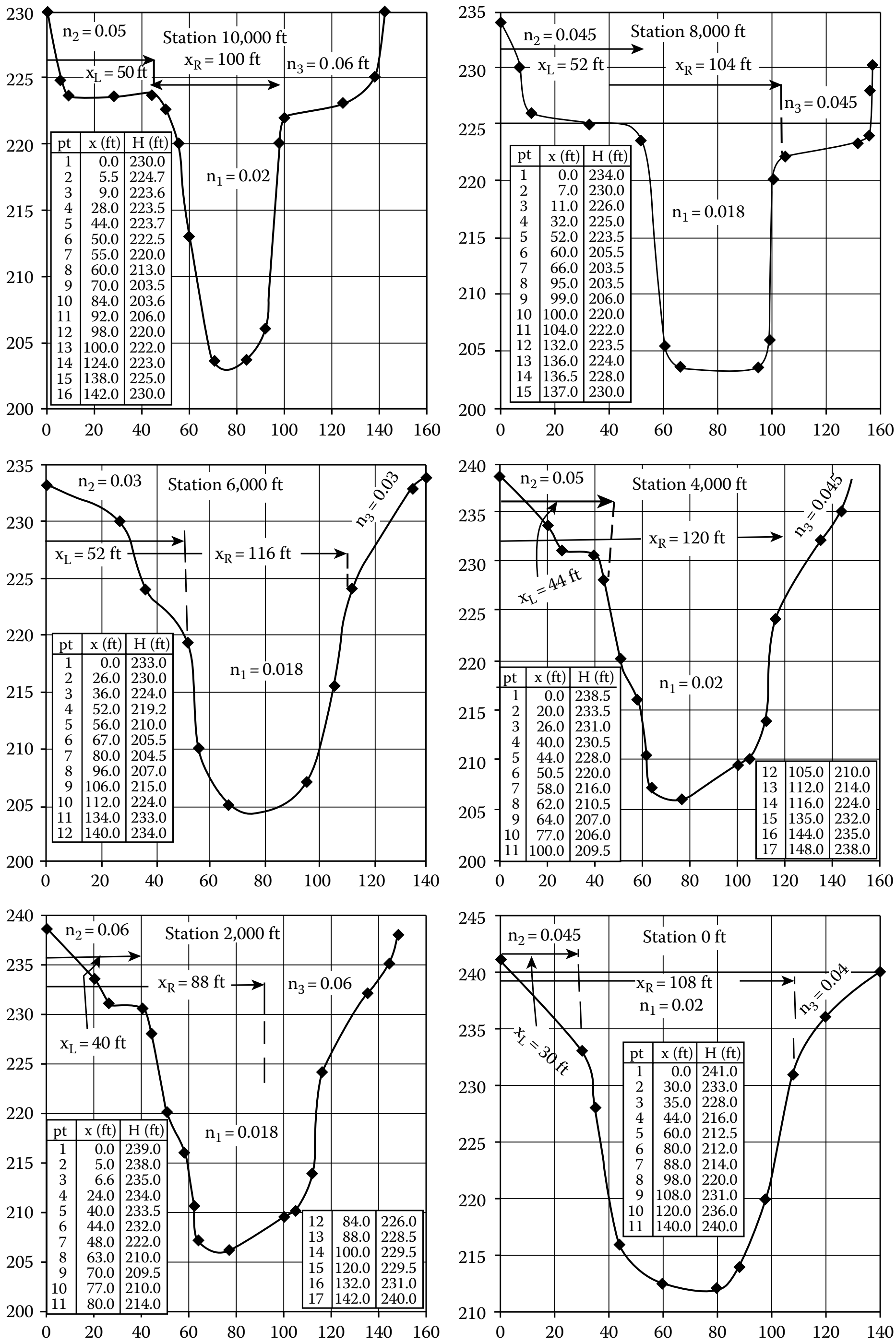
```
UNIT IOUT IOTAB NSTAT TOL  DELX   YB   Q   XBEG XEND
'ES'  2     4    6   1.E-5 -200 26.5 6500 10000  0
```

in file

```
1 16 10000 .02 .05 .06 50 100. TRUE.
0 230 5.5 224.7 9 223.6 28 223.5 44 223.7 50 222.5 55 220 60 213
  70.5 203.5
  84 203.6 92 206 98 220 100 222 124 223 138 225 142 230
2 15 8000 .018 .045 .045 52 104 .TRUE.
0 234 7 230 11 226 32 225 52 223.5 60 205.5 66 203.5 95 203.5 99
  206 100 220
  104 222 132 223.5 136 224 136.5 228 137 230
3 12 6000 .018 .03 .03 52 116 .TRUE.
0 233 26 230 36 224 52 219.2 56 210 67 205 80 204.5 96 207 106
  215.5 112 224
  134 233 140 234
4 17 4000 .02 .05 .045 44 120 .TRUE.
0 238.5 20 233.5 26 231 40 230.5 44 228 50.5 220 58 216 62 210.5
  64 207 77 206
  100 209.5 105 210 112 214 116 224 135 232 144 235 148 238
5 17 2000 .018 .06 .06 40 88 .TRUE.
0 239 5 238 6.6 235 24 234 40 233.5 44 232 48 222 63 210 70 209.5
  77 210
  80 214 84 226 88 228.5 100 229.5 120 229.5 132 231 142 240
6 11 0 .02 .045 .04 30 108 .TRUE.
0 241 30 233 35 228 44 216 60 212.5 80 212 88 214 98 220 108 231
  120 236 140 240
```

'ES' given for UNIT is a 2 character string denoting ES units, 2 for IOUT indicates that logical unit 2 should be used for writing the solution. Logical unit 3 is for input and must therefore not be given, 4 for IOTAB indicates that output of section tables should use unit 4, (a zero will result in this data not written to an output file.) 6 for NSTAT indicates that data for 6 sections will be in the input file.

For each of the six stations, there are two lines of input (the second of which can actually use two or more lines). The first line consists of (1) the station number, starting with the downstream end, (2) the number of pairs of x, z coordinates that will be given on the second line to define the cross section, (3) the position of this station, (4) the Manning's n for the main channel, (5) the Manning's n for the left-side channel, (6) the Manning's n for the right-side channel. (Left and right are defined by looking upstream in the direction of the computations.), (7) the x distance where the left channel ends and the main channel begins, (8) the x distance where the main channel ends and the right side begins, and (9) .TRUE. if linear interpolation is to be used between the x, z pairs in generating the tables of A, P, etc.; .FALSE. if quadratic interpolation is to be used for this purpose. The second line consists of the (x, z) pairs that defined the shape of the cross section at this station.



Assume that the downstream depth is known at station = 10,000 ft to be 27.5 ft, and that the flow rate is Q = 6,500 cfs. This depth results in a stage elevation of 26.5 + 203.5 = 230.0 ft, since the lowest elevation of this station is 203.5 ft. Program GVFNATM provides the prompt **Give: UNIT, IOUT, IOTAB, NSTAT, TOL, DELX, YB, Q, XBEG, XEND** for input from the keyboard. The following is given with the meanings described above: "ES" 2 4 5 1.e–5 –200 26.5 6,500 10,000 0. The solution to the problem is provided in the table below.

Solution to Gradually Varied Flow in Natural Channel with Q = 6500.00

X	Y	E	Per.	Area	Velocity	Bot. S.	n-Main	n-Left	n-Right	Qm/Qt
10000.0	26.50	27.14	65.03	1012.93	6.42	0.0000000	0.0200	0.0500	0.0600	0.9079
9800.0	26.56	27.17	66.22	1033.94	6.29	0.0000000	0.0198	0.0495	0.0585	0.9096
9600.0	26.61	27.20	67.40	1055.30	6.16	0.0000000	0.0196	0.0490	0.0570	0.9113
9400.0	26.66	27.23	68.59	1076.51	6.04	0.0000000	0.0194	0.0485	0.0555	0.9130
9200.0	26.71	27.25	69.77	1097.58	5.92	0.0000000	0.0192	0.0480	0.0540	0.9149
9000.0	26.75	27.28	70.95	1118.54	5.81	0.0000000	0.0190	0.0475	0.0525	0.9168
8800.0	26.79	27.30	72.14	1139.38	5.70	0.0000000	0.0188	0.0470	0.0510	0.9187
8600.0	26.83	27.32	73.32	1160.12	5.60	0.0000000	0.0186	0.0465	0.0495	0.9207
8400.0	26.87	27.34	74.51	1180.76	5.50	0.0000000	0.0184	0.0460	0.0480	0.9228
8200.0	26.91	27.36	75.69	1201.31	5.41	0.0000000	0.0182	0.0455	0.0465	0.9249
8000.0	26.94	27.38	76.87	1221.78	5.32	0.0000000	0.0180	0.0450	0.0450	0.9271
7800.0	26.87	27.30	76.72	1234.04	5.27	0.0005000	0.0180	0.0435	0.0435	0.9309
7600.0	26.79	27.21	76.57	1246.07	5.22	0.0005000	0.0180	0.0420	0.0420	0.9347
7400.0	26.72	27.13	76.42	1257.88	5.17	0.0005000	0.0180	0.0405	0.0405	0.9384
7200.0	26.64	27.05	76.27	1269.46	5.12	0.0005000	0.0180	0.0390	0.0390	0.9419
7000.0	26.56	26.96	76.12	1280.82	5.07	0.0005000	0.0180	0.0375	0.0375	0.9454
6800.0	26.48	26.88	75.96	1291.94	5.03	0.0005000	0.0180	0.0360	0.0360	0.9486
6600.0	26.41	26.79	75.81	1302.84	4.99	0.0005000	0.0180	0.0345	0.0345	0.9518
6400.0	26.33	26.71	75.66	1313.52	4.95	0.0005000	0.0180	0.0330	0.0330	0.9548
6200.0	26.25	26.62	75.51	1323.96	4.91	0.0005000	0.0180	0.0315	0.0315	0.9577
6000.0	26.17	26.54	75.36	1334.17	4.87	0.0005000	0.0180	0.0300	0.0300	0.9605
5800.0	26.04	26.40	76.99	1343.29	4.84	0.0007500	0.0182	0.0320	0.0315	0.9643
5600.0	25.91	26.26	78.62	1352.11	4.81	0.0007500	0.0184	0.0340	0.0330	0.9681
5400.0	25.78	26.13	80.25	1360.64	4.78	0.0007500	0.0186	0.0360	0.0345	0.9719
5200.0	25.65	26.00	81.87	1368.87	4.75	0.0007500	0.0188	0.0380	0.0360	0.9756
5000.0	25.52	25.86	83.50	1376.82	4.72	0.0007500	0.0190	0.0400	0.0375	0.9792
4800.0	25.39	25.73	85.13	1384.48	4.69	0.0007500	0.0192	0.0420	0.0390	0.9828
4600.0	25.26	25.60	86.76	1391.87	4.67	0.0007500	0.0194	0.0440	0.0405	0.9864
4400.0	25.13	25.46	88.39	1398.98	4.65	0.0007500	0.0196	0.0460	0.0420	0.9898
4200.0	25.00	25.33	90.02	1405.81	4.62	0.0007500	0.0198	0.0480	0.0435	0.9933
4000.0	24.87	25.20	91.65	1412.38	4.60	0.0007500	0.0200	0.0500	0.0450	0.9966
3800.0	24.50	24.87	89.11	1324.40	4.91	0.0017500	0.0198	0.0510	0.0465	0.9940
3600.0	24.12	24.55	86.57	1238.15	5.25	0.0017500	0.0196	0.0520	0.0480	0.9919
3400.0	23.74	24.23	84.03	1153.58	5.63	0.0017500	0.0194	0.0530	0.0495	0.9905
3200.0	23.34	23.91	81.48	1070.61	6.07	0.0017500	0.0192	0.0540	0.0510	0.9897
3000.0	22.94	23.61	78.70	989.59	6.57	0.0017500	0.0190	0.0550	0.0525	0.9895
2800.0	22.52	23.31	75.62	910.47	7.14	0.0017500	0.0188	0.0560	0.0540	0.9901
2600.0	22.08	23.02	72.35	832.85	7.80	0.0017500	0.0186	0.0570	0.0555	0.9916
2400.0	21.61	22.76	69.12	756.89	8.59	0.0017500	0.0184	0.0580	0.0570	0.9932
2200.0	21.10	22.51	65.80	682.09	9.53	0.0017500	0.0182	0.0590	0.0585	0.9955
2000.0	20.53	22.30	62.31	607.80	10.69	0.0017500	0.0180	0.0600	0.0600	0.9978
1800.0	20.75	22.19	65.50	675.29	9.63	0.0012500	0.0182	0.0585	0.0580	0.9976
1600.0	20.85	22.05	68.56	737.73	8.81	0.0012500	0.0184	0.0570	0.0560	0.9973
1400.0	20.86	21.89	71.53	796.41	8.16	0.0012500	0.0186	0.0555	0.0540	0.9976
1200.0	20.82	21.72	74.43	852.10	7.63	0.0012500	0.0188	0.0540	0.0520	0.9980
1000.0	20.74	21.54	77.28	905.22	7.18	0.0012500	0.0190	0.0525	0.0500	0.9986
800.0	20.63	21.35	80.09	956.05	6.80	0.0012500	0.0192	0.0510	0.0480	0.9989
600.0	20.50	21.15	82.86	1004.74	6.47	0.0012500	0.0194	0.0495	0.0460	0.9992
400.0	20.36	20.95	85.61	1051.41	6.18	0.0012500	0.0196	0.0480	0.0440	0.9995
200.0	20.20	20.74	88.33	1096.14	5.93	0.0012500	0.0198	0.0465	0.0420	0.9997
0.0	20.03	20.54	91.02	1138.99	5.71	0.0012500	0.0200	0.0450	0.0400	0.9999

5.3.8 System of Natural Streams and Rivers

The previous materials in this chapter dealing with channels having irregular cross sections have been restricted to a single length of a river or a stream. In previous chapters, when dealing with regular channels, techniques were discussed to handle more complicated situations involving the branching of one channel into several channels, or more than a single control. In the case of a channel branching into several channels, any or all of these branches might have gates or other devices in them. A channel could get its water supply from a reservoir, but because of downstream conditions, the amount of flow into that channel may be reduced (or increased) from that obtained from assuming uniform flow because of downstream conditions. A simple example is the case where a gate exists downstream of a channel supplied by a reservoir. The solution to this problem was obtained by simultaneously solving three simultaneous equations. Two or these were algebraic equations (the energy equation across the gate, and the energy equation at the channel entrance), and one ODE, the GVF equation. If a transition existed upstream from the gate, then a fourth equation was added, the energy equation from the beginning to the end of the transition.

Another possible situation that was handled consisted of the problem of locating the position of a hydraulic jump downstream from a gate in a finite length of channel downstream therefrom. For this case, if the water level at the end of the channel is below critical depth so that critical depth is produce here, there are five simultaneous equations to solve. Three of these are algebraic equations—the energy across the gate, the critical flow equation at the end of the channel, and the momentum equation across the hydraulic jump, e.g., $M_1 = M_2$. The fourth and fifth equations are ODEs, one that defines the M_3-GVF upstream from the hydraulic jump, and the other that defines the M_2-GVF downstream from the jump to the end of the channel.

In other words, when dealing with a regular channel, it was practical to handle problems whose solution depended upon solving systems of equations. These systems of equation could include ordinary differential equations as well as algebraic equations. The same can be done when dealing with natural streams and rivers. Seldom in practice is there a single length of river with but one control where the downstream depth is known, for example. Rather real situations involve the type of conditions that were handled for regular channel. The principles utilized in solving problems dealing with a system of channel with regular cross sections are equally applicable for a system of natural streams and river with irregular sections. The only difference is that the computations become more involved because geometric variables must be obtained now from a "table look-up" and interpolation rather than from algebraic equations.

When the technique of calling on an ODE solver to provide a solution to a GVF profile is employed for irregular channels, the methods discussed in previous chapters for solving simultaneous systems of algebraic and ordinary differential equations carry over directly. While the computations are more involved, and more data is needed such as pair of values to define all irregular sections rather than parameters such as the bottom width, side slope, etc., the number of equations involved may be reduced. An example is the above cited case where a transition exists upstream from a gate. The GVF solution between the two irregular sections automatically accounts for the transition. (Note the same could be done in the regular channel case. The energy equation across the transition could be eliminated by using a nonprismatic term in the ODE.)

5.3.9 HEC-2 Water Surface Profiles

The HEC-2 computer program developed by the U.S. Army Corps of Engineers, Hydrologic Engineering Center, that has been in existence since about 1976, is widely used by practicing engineers and others to determine steady-state water surface profiles in natural channels such as rivers. It handles these GVF computations based on one-dimensional hydraulics and evaluates frictional losses with the Manning's equation. This program uses the standard step method described earlier and is designed to compute GVF-profiles for supercritical as well as subcritical conditions. Since subcritical flows are controlled by downstream conditions, these GVF problems are solved starting at some downstream position where the depth is known (controlled) and proceed upstream. GVF profiles for supercritical flows proceed from an upstream section where the depth is known in a downstream direction. HEC-2 has many capabilities and features not described in earlier sections. The effects of various obstructions, such as bridges, culverts, weirs, and structures, in the channel and flood plain are accommodated. In other words, much effort has been expended in developing a software package that allows solutions to problems in irregular channels. As a hydraulic engineer, you should become acquainted with the capabilities and use of HEC-2.

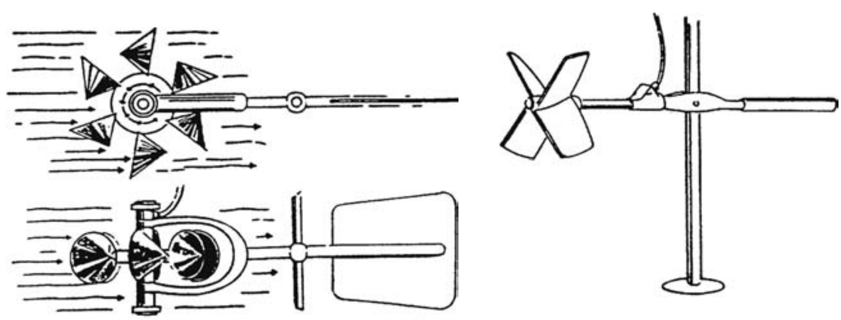
5.4 WATER MEASUREMENT IN CHANNELS

In this section, several devices commonly used to measure flow rates in channels, ditches, streams, and rivers will be described. For our discussions, these descriptions will be brief and the reader needs to consult a publication devoted specifically to a given device to get more information related to its operation, maintenance, and calibration.

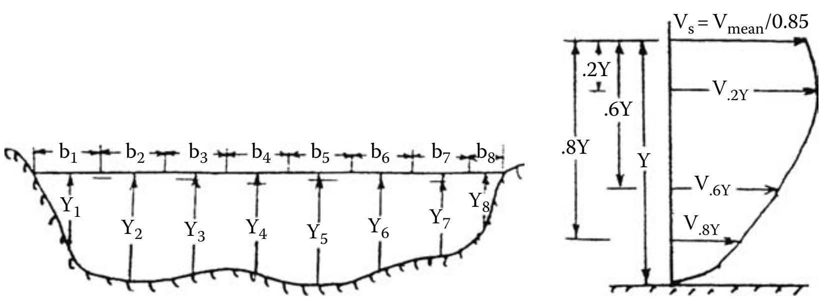
A rough estimate of the flow rate in a straight uniform channel is available from an object that floats. A piece of wood, apple, lemon, etc. may be thrown into the stream and the time recorded for this object to travel a known distance downstream. The average cross-sectional area multiplied by 0.85 times the surface velocity of the float gives the flow rate.

5.4.1 Current Meters

Flow rates in streams and rivers are often measured by dividing the cross-sectional area in several subareas and measuring the velocities in each of these with a current meter or other device that measures point velocity. In the past, most current meters consisted of a wheels fitted with cupped vanes and mounted on an axis about which the wheel is free to rotate in direct response to the velocity of the flow by the meter. In the last decade, electronic devices such as acoustic Doppler velocity or magnetic flux meters have become available that record velocities at the point of the meters. These meter may be used in shallow water where wading is possible supported by a vertical rod held in the hands of the observer. In deeper water, or more rapidly moving water, the rod is replaced by a cable attached to the meter, which is suspended from a boat, bridge, or cable car suspended across the river.



Price (left) and Hoff (right) current meters that have been used widely in past



The flow rate is the sum of products of the individual subareas multiplied by the mean velocity in each subarea, as shown below. Various methods may be used for obtaining the mean velocity of each subarea. The two most commonly used methods are the following: (1) Making a single velocity measurement at a position equal to 0.6 the mean depth of the subarea. (This method assumes that the velocity at this position represents the mean velocity.) (2) Making two velocity measurements, at 0.2 of the mean depth and the other at 0.8 of the mean depth, and averaging these two velocities as the mean velocity. Using the $(V_{.2Y} + V_{.8Y})/2$ is probably better in most streams than the single velocity $V_{.6Y}$. The flow rate $Q = A_1V_1 + A_2V_2 + \cdots + A_nV_n = \sum A_iV_i$.

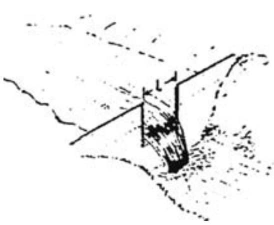
5.4.2 Weirs

A weir is the simplest of the practical devices used to measure open channel flow rates. Weirs may be divided into two general categories: (1) sharp-crested and (2) broad-crested. When properly constructed and operated, it is also one of the most accurate. In its simplest form, a sharp-crested weir consists of a bulkhead of steel, timber, or concrete at right angles to the direction of flow across the stream. If there is concern of seepage and erosion around the sides of the weir, it can be placed in a "weir box" that consists of walls, which extend upstream and basically line the stream for a short distance. The crest and sides of weirs should be straight and sharp-edged, and usually 4–8 mm (one-eight to one-quarter inch) in thickness. The crest of the weir should be horizontal. Below are sketches of four common sharp-crested weirs being applied to measure flows in small ditches or streams, along with the appropriate equation used to determine the discharge, Q, as a function of the head, H, over the weir. The coefficients in these equations vary with a number of factors as well as the head H on the weir. A commonly used formula for determining the coefficient, C, is the Rehbock equation,

$$C = 0.605 + 0.08\frac{H}{P} + \frac{1}{305H} \tag{5.10}$$

where H is the head on the weir and P is the height of the weir above the channel bottom.

The first weir shown is a rectangular contracted weir. The name rectangular contracted weir comes from the location and shape of the opening through which the water flows. It is simple to construct and measures flow accurately. It is one of the most popular weirs because of its simplicity. Recommended sizes for rectangular weir are as follows:



Rectangular contracted weir

$Q = C_3\frac{2}{3}\left(b - \frac{2H}{10}\right)\sqrt{2g}\,H^{3/2}$

C_3 may be taken equal to C for a first approximation

Cipoletti weir

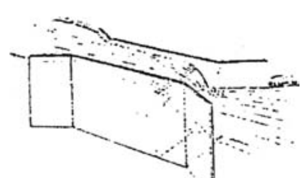
Same equation as below for sharp edged, or rectangular suppressed weir

Triangular weir

$Q = \frac{8}{15}C_2 \tan\propto \sqrt{2g}\,H^{5/2}$

$= 4.28\ C_2 \tan\propto\ H^{5/2}$

C_2 may be taken equal to C for a first approximation



Sharp edged weir

$Q = C\frac{2}{3}b\sqrt{2g}\,H^{3/2} = C_1bH^{3/2}$

Flow Rate		Max. Head, H		Crest Length, b	
m^3/s	cfs	m	ft	m	ft
0.0085–0.056	0.30–2.00	0.23	0.75	0.31	1.0
0.056–0.071	2.00–2.50	0.23	0.75	0.46	1.5
0.056–0.170	2.50–6.00	0.31	1.00	0.61	2.0
0.141–0.368	5.00–13.0	0.38	1.25	0.91	3.0
0.227–0.566	8.00–20.0	0.43	1.40	1.22	4.0

The most common type of triangular, or V-notch, weir has a 90° notch, and it has a greater range of capacity than other types for a given size. Because it causes a larger head loss, its use is commonly for flow rates less than 4 cfs (0.11 m^3/s).

The Cipolletti, or trapezoidal, weir with a side slope of 1/4, is named for the Italian engineer who designed it. Cipolletti proposed giving the sides such a slope that the increased area would be just equal to the decrease in discharge due to the fluid contracting from the sides of the weir. This contraction is commonly takes to be 0.1 time H on both sides, as reflected in the above equation for the rectangular contracted weir. Recommended sizes for Cipolletti weirs are as follows:

Flow Rate		Max. Head, H		Crest Length, b	
m^3/s	cfs	m	ft	m	ft
0.0085–0.065	0.30–2.30	0.23	0.75	0.31	1.0
0.0566–0.113	2.00–4.00	0.26	0.85	0.46	1.5
0.085–0.198	3.00–7.00	0.31	1.02	0.61	2.0
0.142–0.396	5.00–14.0	0.38	1.24	0.91	3.0
0.227–0.623	8.00–22.0	0.43	1.40	1.22	4.0

Rectangular suppressed, or just sharp-edged, weirs have no end contractions. A suppressed weir may be in the form of a flume of uniform cross section, with a vertical weir plate extending upward from the bottom. The length of the flume should be at least 10 times the length of the weir crest, b, in order to remove turbulence in the approach section. Ventilation under the water falling over the crest of the weir should be provided to maintain the pressure in this region equal to atmospheric pressure. The height of the crest above the bottom should be at least twice the maximum head to be measured. For good accuracy, the head should not be greater than 0.61 m (2.0 ft) or less than 0.06 m (0.2 ft).

The following are general guidelines associated with installation and operation of weirs:

1. The weir should be located at the downstream end of a pool where the streamlines will smoothly flow into the weir with an approach velocity no larger than 0.15 m/s (0.5 fps).
2. The face of the weir should be vertical and at a right angle to the direction of the flow.
3. The distance of the crest of the weir above the bottom of the pool should be a least two times (and preferably three times) the water depth, or head H, over the crest of the weir. The distance between the sides of the weir to sides of the pool should be at least twice the head, H, over the weir.
4. The scale that records H should be sufficiently upstream or on one side of the weir so that the velocity head does not create any significant drawdown of the water surface.
5. The crest should be placed high enough so the water will fall freely from the crest of the weir with a visible air pocket under the weir overflow jet.

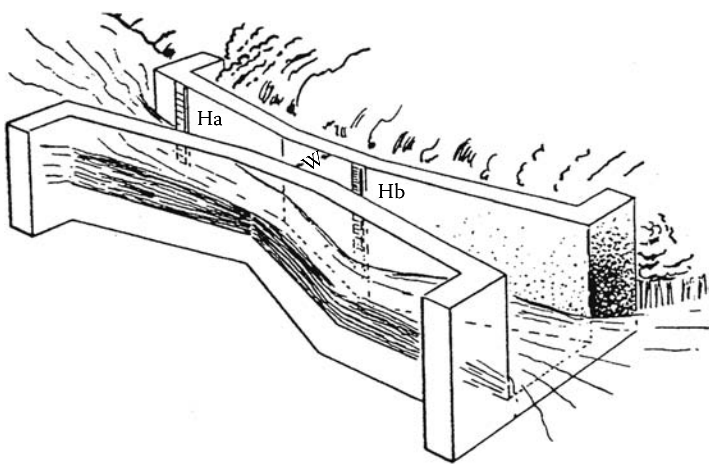
6. The head, H, should be less than 1/3 the length of the weir crest, b.
7. The weir should not be allowed to collect sediment upstream from it, and material should protect the banks of dirt ditches from erosion.

The following are a list of things to avoid with weirs and other measurement devices:

1. Don't locate a weir below a curve in a ditch of stream that will cause the water to flow predominately to one side of the weir crest.
2. Don't locate the weir close to other structures such as headgates where a significant velocity of approach exists.
3. Don't allow water downstream from the weir to backup and affect the flow over the weir crest.
4. Don't attempt to use too small a weir. Use a size so that the head meets the above requirement, namely H less then P/2 (and preferably P/3).
5. Don't allow the pool to fill up with sediment or debris.

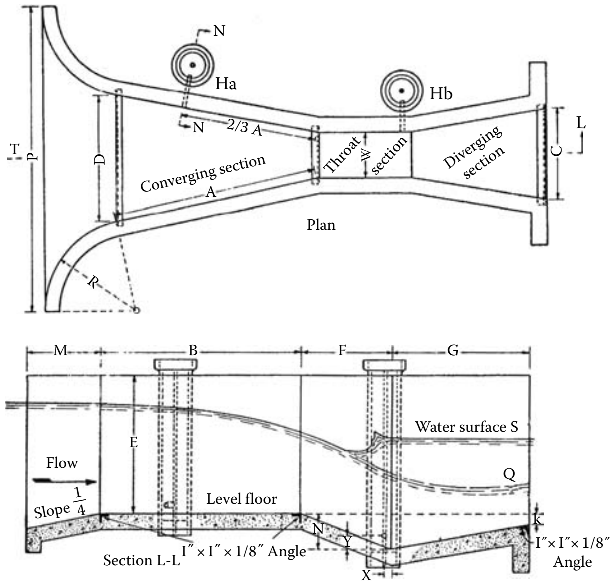
5.4.3 Parshall Flumes and Cutthroat Flumes

Parshall flumes are widely used to measure flows in canals, ditches, and streams. A Parshall flume consists of a converging section, a throat, and a diverging section, with the floor of the converging or upstream section level. The floor of the throat is inclined downward with a slope of 9–24 (or 0.375) within the diverging or downstream section beyond the throat and is inclined upward at a slope of 1–6 (.167) downstream therefrom. Throat widths are from 0.31 to 2.4 m (1–8 ft). A Parshall flume is a critical flow meter since it creates critical flow as the basis for the equation, or calibration, that gives the flow rate as a function of the head in a stilling well, or scale in the upstream portion of the flume. When properly installed and maintained, the accuracy of measurements with Parshall flumes is 2% to 5%. This requires that the proper size is selected and that it is flowing under "free flow" conditions.



Dimensions and Capacities for Parshall Flumes with Different Throat Widths in ES Units

W (ft)	Flowrate (cfs)	A (ft)	2A/3 (ft)	B (ft)	C (ft)	D (ft)	E (ft)	F (ft)	G (ft)	K (ft)	N (ft)	R (ft)	M (ft)	P (ft)	X (ft)	Y (ft)
0.25	0.03–1.9	1.531	1.021	1.500	0.583	0.849	2.0	0.5	1.0	0.083	0.1875	1.333	1.000	2.521	0.0833	0.125
0.50	0.05–3.9	2.438	1.625	2.000	1.293	1.302	2.0	1.0	2.0	0.250	0.375	1.333	1.000	2.958	0.167	0.250
0.75	0.09–8.9	2.885	1.923	2.833	1.500	1.885	2.5	1.0	2.5	0.250	0.375	1.333	1.000	3.542	0.167	0.250
1.0	0.11–16.1	4.500	3.000	4.406	2.000	2.771	3.0	2.0	3.0	0.250	0.750	1.667	1.250	4.896	0.167	0.250
1.5	0.15–24.6	4.750	3.167	4.656	2.500	3.365	3.0	2.0	3.0	0.250	0.750	1.667	1.250	5.500	0.167	0.250
2.0	0.42–33.1	5.000	3.333	4.906	3.000	3.958	3.0	2.0	3.0	0.250	0.750	1.667	1.250	6.083	0.167	0.250
3.0	0.61–50.4	5.500	3.667	5.396	4.000	5.156	3.0	2.0	3.0	0.250	0.750	1.667	1.250	7.292	0.167	0.250
4.0	1.3–67.9	6.000	4.000	5.885	5.000	6.354	3.0	2.0	3.0	0.250	0.750	2.000	1.500	8.896	0.167	0.250
5.0	1.6–85.6	6.500	4.333	6.375	6.000	7.552	3.0	2.0	3.0	0.250	0.750	2.000	1.500	10.104	0.167	0.250
6.0	2.6–103.5	7.000	4.667	6.865	7.000	8.750	3.0	2.0	3.0	0.250	0.750	2.000	1.500	11.292	0.167	0.250
7.0	3.0–121.4	7.500	5.000	7.354	8.000	9.948	3.0	2.0	3.0	0.250	0.750	2.000	1.500	12.500	0.167	0.250
8.0	3.5–139.5	8.000	5.333	7.844	9.000	11.146	3.0	2.0	3.0	0.250	0.750	2.000	1.500	13.688	0.167	0.250



Plan and profile views of a concrete Parshall Flume. W = size of the flume (throat width), A = length of side wall of converging section, 2/3 A = distance back from end of crest to gage point of connection to stilling well, B = axial length of converging section, C = width of downstream end of flume, D = width of upstream end of flume, E = depth of flume, F = length of throat, G=length of diverging section, M = length of approach floor, N = depth of depression in throat below crest, P = width between ends of curved wing walls, R = radius of curved wing wall, X = horizontal distance to Hb gate point from low point in throat, Y = vertical distance to Hb gage point from low point in throat. See Table above and/or below for actual dimensions.

Throat width 3 in.
$Q = 0.992\ H_a^{1.547}$ (cfs)

Throat width 6 in.
$Q = 2.06\ H_a^{1.58}$ (cfs)

Throat width 9 in.
$Q = 3.07\ H_a^{1.53}$ (cfs)

Throat width 12 ft–8 ft
$Q = 4WH_a^{1.522W^{.026}}$ (cfs)

Throat width 10 ft–50 ft
$Q = (3.6875W + 2.5)H_a^{1.6}$ (cfs)

In the above equations, Q is the free discharge in cfs. W is the width of throat in ft., and H_a is the gage reading in ft. When the ratio of the gage reading H_b to H_a exceeds the limits of 0.6 for 3-, 6-, and 9-in. flumes, 0.7 for 1- to 8-ft. flumes, and 0.8 for 10- to 50-ft. flumes, the flow becomes submerged and the discharge must be reduced (see Fig. 4–7, Chow, 1959.)

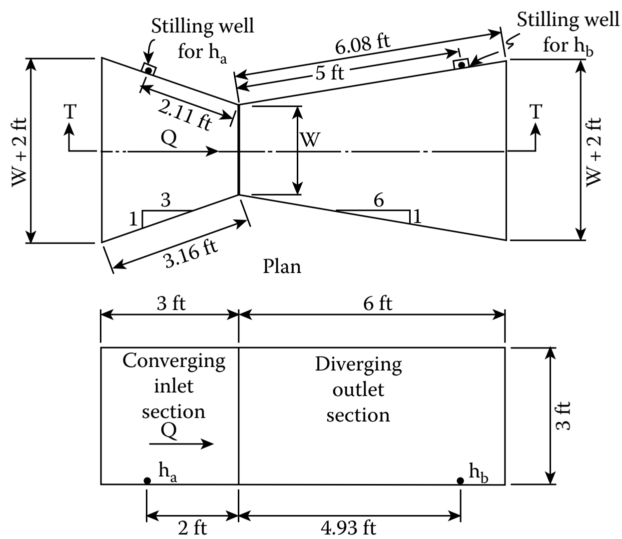
Dimensions and Capacities for Parshall Flumes with Different Throat Widths in SI Units

W (m)	Flow Rate m³/s	A (m)	2A/3 (m)	B (m)	C (m)	D (m)	E (m)	F (m)	G (m)	K (m)	N (m)	R (m)	M (m)	P (m)	X (m)	Y (m)
0.076	0.0008–0.0538	0.467	0.311	0.457	0.178	0.259	0.610	0.152	0.305	0.025	0.057	0.406	0.305	0.768	0.025	0.038
0.152	0.0014–0.1104	0.743	0.495	0.610	0.394	0.397	0.610	0.305	0.610	0.076	0.114	0.406	0.305	0.902	0.051	0.076
0.229	0.0025–0.2520	0.879	0.586	0.863	0.457	0.575	0.762	0.305	0.762	0.076	0.114	0.406	0.305	1.080	0.051	0.076
0.305	0.0031–0.4559	1.372	0.914	1.343	0.610	0.845	0.914	0.610	0.914	0.076	0.229	0.508	0.381	1.492	0.051	0.076
0.457	0.0042–0.6966	1.448	0.965	1.419	0.762	1.026	0.914	0.610	0.914	0.076	0.229	0.508	0.381	1.676	0.051	0.076
0.610	0.0119–0.9373	1.524	1.016	1.495	0.914	1.206	0.914	0.610	0.914	0.076	0.229	0.508	0.381	1.854	0.051	0.076
0.914	0.0173–1.4272	1.676	1.118	1.645	1.219	1.572	0.914	0.610	0.914	0.076	0.229	0.508	0.381	2.223	0.051	0.076
1.219	0.0368–1.9227	1.829	1.219	1.794	1.524	1.937	0.914	0.610	0.914	0.076	0.229	0.610	0.457	2.712	0.051	0.076
1.524	0.0453–2.4239	1.981	1.321	1.943	1.829	2.302	0.914	0.610	0.914	0.076	0.229	0.610	0.457	3.080	0.051	0.076
1.829	0.0736–2.9308	2.134	1.423	2.092	2.134	2.667	0.914	0.610	0.914	0.076	0.229	0.610	0.457	3.442	0.051	0.076
2.134	0.0850–3.4377	2.286	1.524	2.241	2.438	3.032	0.914	0.610	0.914	0.076	0.229	0.610	0.457	3.810	0.051	0.076
2.438	0.0991–3.9502	2.438	1.625	2.391	2.743	3.397	0.914	0.610	0.914	0.076	0.229	0.610	0.457	4.172	0.051	0.076

The cutthroat flume was developed as a replacement to the Parshall flume because it was much simpler and less costly to construct. It has a flat floor and only a converging and then a diverging section. It, like the Parshall flume, may be classified as a critical flow meter. Under free flow conditions, the flow rate, Q, through a cutthroat flume depends only upon the upstream depth of flow, H_a, the depth of water in the stilling well above its level floor. The following basic equation can be used to determine this flow rate:

$$Q = CH_a^{1.56} \quad \text{in which} \quad C = 3.50W^{1.025}$$

where W is the throat width in feet.



Plan and sectional view of rectangular cutthroat measuring flume

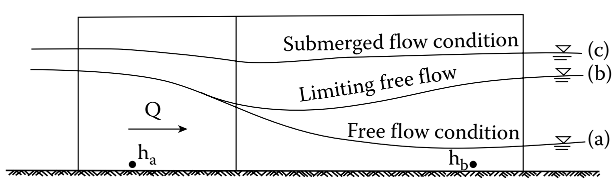
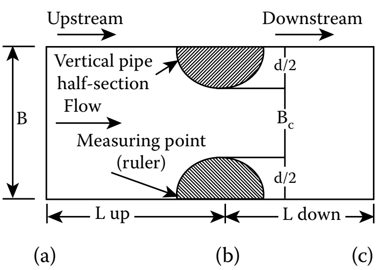


Illustration of flow conditions in a rectangular cutthroat measuring flume

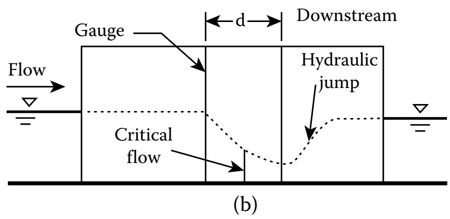
A sketch of a cutthroat flume is given above. The one varying dimension indicated on this sketch is the flume size, or throat width, W. The length of the converging and diverging sections are the same for each flume size, as well as the location of the points for upstream depth measurement, H_a, and the depth measurement, H_b, that is needed only if submerged conditions exist. More information related to cutthroat flumes is contained in “Design and Calibration of Submerged Open Channel Flow Measurement Structure—Part 3 Cutthroat Flumes,” Utah Water Research Laboratory, College of Engineering, Utah State University, Logan, Utah 84322-8200.

5.4.4 Other Critical Flow Flumes

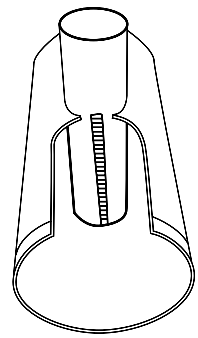
Since pipes are readily available, two halves of a pipe can be placed vertically in a rectangular channel to create a critical flow flume. The two sketches below shows such a flow measuring device, which its inventors have named an S–M (Samani and Magallanez) flume after their last names (Samani and Yousaf 1991, Samani et al. 2006).



Top view of S–M flume



Side view of S–M flume



Circular flume (described below)

One of the half pipes can act as the stilling well for the flume with the water entering this stilling well from a small hole drilled at it front at the bottom of the flume. Theoretically, the flow rate can be determined by solving the energy equation between the upstream end of the flume and its throat where critical flow occurs. The head H measured in the stilling well will not be exactly equal to the total head upstream, and furthermore two-dimensional effects are present, so the flow rate is determined from the empirical equation,

$$Q = 0.701(g)^{1/2}B_c^{0.91}H^{1.59}$$

The size of the S–M flume is determined according to the range of flow rates that it is anticipated to measure according to the table below.

	B = Width of Flume (ft)					
B_c/B	1	2	3	4	5	6
0.40	0.04–5.0	0.2–9.0	0.3–14.0	0.3–18.0	0.4–22.0	0.4–26.0
0.50	0.05–6.0	0.3–11.0	0.4–17.0	0.5–22.0	0.6–27.0	0.7–32.0
0.60	0.8–7.0	0.5–14.0	1.0–20.0	1.3–26.0	1.5–32.0	2.0–38.0

If the open channel flow is in a pipe rather than a rectangular or trapezoidal channel, then a smaller diameter pipe can be inserted vertically through the top of the pipe (third sketch above), and a scale to measure the head installed within this vertical pipe as shown in the sketch, to create a circular-flume critical flow measuring device. The ratio of the diameter of the smaller vertical pipe to the larger pipe should be about 1/3, or d/D = 0.25 to d/D = 0.32. If one assumes that the upstream head equals the critical specific energy,

$$E_1 = Y_1 + \frac{Q^2}{2gA_1^2} = E_c = Y + \frac{Q^2}{2gA^2} = Y + \frac{A}{2T}$$

where the unsubscripted Y, A, and T apply at the throat where the smaller vertical pipe causes the flow to become critical. At this section the cross-sectional area A is given by

$$A = 0.25D^2[\beta - \sin\beta\cos\beta] - (Y - Y_n)d - 0.25d^2[\alpha - \sin\alpha\cos\alpha]$$

where D is the diameter of the pipe in which the flow occurs, d is the diameter of the inserted vertical pipe, β is as defined previously, i.e., $\beta = \cos^{-1}(1 - 2Y/D)$, $\alpha = \sin^{-1}(d/D)$, and Y_n is the depth in the larger pipe that is completely removed by the smaller vertical pipe, and is computed from $Y_n = 0.5D[1 - \cos(\alpha)]$. Since the only portion of this area that will vary with the depth is the first term, and Yd, this equation for area can be written as

$$A = 0.25D^2[\beta - \sin(\beta)\cos(\beta)] - dY - A_m$$

where $A_m = 0.25d^2[\alpha - \sin(\alpha)\cos(\alpha)] - dY_n = 0.25[d^2/\tan(\alpha) + \alpha D^2] - 0.5dD$.

The calibration of this circular flume shows that the head measured in the vertical pipe is related to the upstream energy by $H = E_1/0.96$, and that the actual flow rate is related to that computed from the above equations by

$$\frac{Q_m}{Q} = 1.057 + 0.2266\left(\frac{H}{D}\right)$$

where

Q_m is the measured flow rate

Q is the flow rate computed by solving the above critical flow equations

EXAMPLE PROBLEM 5.2A

A circular measuring flume contains a vertical 0.3 ft diameter pipe inserted in a 1 ft diameter pipe. Develop the rating table for this flume. (Assuming critical flow occurs.)

Solution

For a series of upstream-specific energies the above equations need to be solved for Y and Q, and because of the nature of the equations A, T, and β needed to be added as unknowns. Below a TK-Solver model accomplishes this task. The last two columns in the table sheet give the flow rate Q_m corresponding to the head H measured in the vertical pipe. To have the calibration apply for different sizes the values given to D1 (diameter of the pipe) and D2 (diameter of the vertical inserted pipe) can be changed and F10 pressed to "list solve" for the new table values.

```
════ VARIABLE SHEET ══════
St Input───── Name──── Output───
   1          D1
   .3         D2
              Am       -.0022814
              Alpha    .30469265
L  .5         E
LG 1.3271706  Beta
LG .37938858  Y
LG .16173261  A
LG .67046975  T
LG .45074897  Q
L             H        .52083333
L             Qm       .52751152
```

```
════════════════ RULE SHEET ═══════════════
  Alpha=ASIN(D2/D1)
  Am=.25*(D2^2/tan(Alpha)+Alpha*D1^2)-.5*D1*D2
  A=D1^2/4*(Beta-sin(Beta)*cos(Beta))-Y*D2-Am
  E=Y+A/(2*T)
  T=D1*sin(Beta)-D2
  cos(Beta)=1-2*Y/D1
  Q=A*sqrt(64.4*(E-Y))
  H=E/.96
  Qm=Q*(1.057+.2266*E/D1
```

════════════════════ TABLE: ═════════════════════

Title: Circular Flume-for D = 1 ft, and vertical pipe d = 0.3 ft.

Element	E	Q	Y	A	Beta	T	H	Qm
1	0.3	0.1483	0.2342	7.201E-2	1.0102	0.547	0.3125	0.1668
2	0.325	0.1777	0.2527	8.234E-2	1.0533	0.5691	0.3385	0.2009
3	0.35	0.2097	0.2711	9.299E-2	1.0952	0.589	0.3646	0.2383
4	0.375	0.2441	0.2894	0.1039	1.136	0.6069	0.3906	0.2788
5	0.4	0.2809	0.3076	0.1151	1.1758	0.623	0.4167	0.3224
6	0.45	0.3615	0.3437	0.1382	1.2529	0.6499	0.4688	0.4189
7	0.5	0.4507	0.3794	0.1617	1.3272	0.6705	0.5208	0.5275
8	0.55	0.5481	0.4146	0.1856	1.3991	0.6853	0.5729	0.6477
9	0.6	0.6529	0.4492	0.2095	1.4691	0.6948	0.625	0.7789
10	0.65	0.7644	0.4833	0.2333	1.5373	0.6994	0.6771	0.9205
11	0.7	0.8819	0.5166	0.2566	1.604	0.6994	0.7292	1.072
12	0.75	1.0045	0.5491	0.2793	1.6692	0.6952	0.7813	1.2325
13	0.8	1.1316	0.5808	0.3012	1.733	0.6869	0.8333	1.4013
14	0.85	1.2623	0.6114	0.322	1.7955	0.6749	0.8854	1.5774
15	0.9	1.3959	0.6409	0.3417	1.8565	0.6595	0.9375	1.7601

5.5 DESIGN OF TRANSITIONS

Because grades of channels need to correspond approximately to the slope of the terrain and it is not good economics to keep the size of a channel constant if its bottom slope changes, transitions between channels of different sizes are common. An improperly designed channel transition can cause the entire system to perform badly. An improperly designed transition from a large mild channel to a smaller steep channel can easy act as a "choke," causing the flow to back up in the upstream channel, and even overtop it if there is not adequate freeboard or reduce the flow rate that the system can carry. There are numerous other situations that can result in unacceptable performance of a channel system that are the consequences of inadequate design of transitions. For an individual who is not familiar with open channel hydraulic principles, the resulting occurrences would likely seem strange and unexpected. To such untrained individuals, transitions in open channels would also seem unimportant and not worthy of the added expense associated with implementing a design with curved sides, and a change in bottom or elevation. They might ask why not just change the section from one size to the other abruptly? Trying to correct a situation that was not anticipated in the original channel design is generally very costly, and the design engineers might well find themselves in litigation and liable for damages resulting from improper channel design. The subject of transition design is of vital importance.

The design of a transition that is to operate with supercritical flows is a completely different subject than the design of a transition for subcritical flows. In supercritical flows, a change in the size of a channel creates cross waves, or oblique hydraulic jumps, and the major concern of the design is to minimize, or eliminate such cross waves from forming. The best is to avoid transitions in supercritical flows if possible, especially if the supercritical flow is upstream from the transition. In subcritical flows energy considerations dictate what will happen, and the main concern is to maintain the depths as near as possible to uniform depths both in the upstream and downstream channels for the design flow rate.

5.5.1 Subcritical Transitions

The design of transitions between channels of two different sizes for upstream subcritical flow will be approached in two different manners below. First, such designs will be approached from the viewpoint of accomplishing the design by "hand" computations. Second, such design will pose the problem as an ordinary differential equation, and an ODE solver will be utilized.

5.5.1.1 Design of Transitions by Hand Computations

Traditional methods used for subcritical transition designs can be accomplished by completing the following steps:

1. Determine the normal depths (uniform flow depths) for both the upstream and downstream channels for the design flow rate (generally the largest flow rate that the channel is anticipated to carry under usual conditions). These computations solve for Y_{o1} and Y_{o2} from Manning's equation, or Chezy's equation, depending upon the equation being used to define uniform flow. Computations of these depths assumes that the size, bottom slope, and roughness coefficients are known for both the channel upstream and downstream from the transition.
2. Decide upon the length, L, of the transition. A rule of thumb for determining this length is that angles of side changes should not exceed 12.5°. This rule is of particular importance if the size increases in the downstream direction, or else the water will separate from the channel sides and result in excessive headlosses. Thus, for example, if the transition is from a 4 m wide rectangular channel to a 6 m wide trapezoidal channel with a side slope of 1, and the anticipated depth in the downstream channel is 3 m, then this rule would give a minimum length of

$$L = \frac{T_2 - T_1}{\tan 12.5°} = \frac{(b_2 + 2mY_2 - b_1)}{\tan 12.5°} = \frac{(12 - 4)}{0.222} = 36.1\ \text{m}$$

3. Compute the velocity heads in both upstream and downstream channels and estimate the headloss through the transition. This estimate of headloss includes frictional plus minor losses.
4. Decide upon the nature of the side contractions or enlargements. Two reversed parabolas connected at the midpoint for each side is a commonly used function for this purpose because it provides a smooth transition and is easy to work with. Such reversed parabolas give a width (and top width) at the midpoint equal to the average of the upstream and downstream widths, or $b_m = (b_1 + b_2)/2$, and upstream from this midpoint the amount of reduction (or enlargement) in width is proportional to the square of the ratio of distance from the beginning to the midpoint length, and downstream the same enlargement (or reductions) in width occur except starting from the end of the transition moving upstream.
5. Divide the transition into a reasonable number of equal length intervals, and determine the channel width and other geometric properties at each section. For the example in step 2, let us assume that L = 40 m is selected, and the transition is divided into eight sections (intervals) spaced at 5 m. Furthermore assume that the transition will be trapezoidal with the side slope m varying linearly from 0 to 1 across the transition. Using reversed parabolas gives the following bottom widths and side slopes at each stations through the transitions.

Sta. No.	Dist. x (m)	Side Slope (m)	Change in Width Δb (m)	Bottom Width b (m)
Beg.	0	0		= 4.00
1	5	0.125	$(1/4)^2(1) = 0.0625$	4 + 0.0625 = 4.0625
2	10	0.250	$(2/4)^2(1) = 0.2500$	4 + 0.25 = 4.25
3	15	0.375	$(3/4)^2(1) = 0.5625$	4 + 0.5625 = 4.5625
4	20	0.500	(6 – 4)/2 = 1.000	4 + 1.000 = 5.00
5	25	0.625	0.5625	6 – 0.5625 = 5.4375
6	30	0.750	0.25	6 – 0.25 = 5.75
7	35	0.875	0.0625	6 – 0.0625 = 5.9375
End	40	1.000		= 6.00

6. Specify the shape of the water surface profile. Two reversed parabolas connected at the midpoint is common again for this profile. From this shape determine the elevation of the water surface at each station.
7. Assuming that the headloss determined in Step # 3 is essentially uniformly distributed along the transition (i.e., the slope of the energy line is constant from the beginning to the end of the transition), determine the elevation of the energy line at each station.
8. Compute the velocity head at each station by subtracting the elevation from step # 6 from the EL-elevation from step # 7, and then compute the velocity at each station.
9. Compute the area and depth of water at each station.
10. Compute the elevation of the channel bottom at each station by subtracting the depth from step # 9 from the elevation of the water surface from step # 6.
11. If the profile of the channel bottom is unacceptable, adjust side contractions, and/or the water surface profile and repeat steps 7 through 10 until the bottom is satisfactory.

EXAMPLE PROBLEM 5.2

A transition occurs from an upstream rectangular channel with a bottom width of $b_1 = 4$ m, a Manning's $n_1 = 0.013$ and a bottom slope $S_{o1} = 0.002$ to a trapezoidal channel with $b_2 = 6$ m, $m_2 = 1.5$, $n_2 = 0.015$ and $S_{o2} = 0.00075$. Assume that the elevation of the channel bottom at the beginning of the transition equals 100 m. The design flow rate is 50 m³/s.

Solution

Solving Manning's equation twice in completing step # 1 gives: Y_{o1} = 3.171 m and Y_{o2} = 4.016. Step # 2 gives L_{min} = {(6 + 2(1.5)4.016 − 4}/tan 12.5° = 63.375 m, Use L = 65 m. An appropriate number of section increments is 10. The velocity heads at the ends of the transition are as follows: $(V^2/2g)_1$ = 0.792 m, and $(V^2/2g)_2$ = 0.055 m. The headloss through the transition might be estimated by using the product of the length and the average slope of the upstream and downstream bottom slopes plus a loss coefficient time the upstream velocity head, or $h_L = L(S_{o1} + S_{o2})/2 + K_L V_1^2/(2g)$ = 65(.002 + 0.00035)/2 + 0.1 (.792) = 0.156 m. The beginning elevation of the EL (energy line) equals 100 + 3.171 + 0.792 = 103.963 m, and EL_2 = 103.963 − 0.156 = 103.807 m. Subtracting the velocity heads from these energy line elevations give ws-elev$_1$ = 103.963 − 0.792 = 103.171 m and ws-elev$_2$ = 103.807 − 0.055 = 103.752 m. After determining the bottom widths according to two reversed parabolas, and also finding the water surface elevation between these end values result in the table below.

Sta.	x (m)	m	b (m)	Elev-EL (m)	EL-ws (m)	Vel. H (m)	V (m/s)	A (sq-m)	Y (m)	z (m)
Beg	0.0	0.000	4.000	103.963	103.171	0.792	3.942	12.684	3.171	100.000
2	6.5	0.150	4.040	103.947	103.183	0.765	3.874	12.908	2.886	100.297
3	13.0	0.300	4.160	103.932	103.218	0.714	3.744	13.356	2.689	100.528
4	19.5	0.450	4.360	103.916	103.276	0.641	3.545	14.104	2.559	100.717
5	26.0	0.600	4.640	103.901	103.357	0.544	3.266	15.311	2.495	100.862
6	32.5	0.750	5.360	103.885	103.566	0.319	2.500	19.996	2.706	100.860
7	39.0	0.900	5.640	103.869	103.648	0.222	2.086	23.974	2.905	100.743
8	45.5	1.050	5.840	103.854	103.706	0.148	1.704	29.346	3.193	100.513
9	52.0	1.200	5.960	103.838	103.741	0.097	1.383	36.156	3.541	100.199
End	65.0	1.500	6.000	103.807	103.752	0.055	1.035	48.288	4.016	99.736

The drop of 0.31 m in the position of the channel bottom z may not be ideal, and either the manner in which the bottom width is varied, or the way that the water surface elevation is determined might be changed.

The above method of implementing the design of a subcritical transition through a systematic table can easily be done using a spreadsheet, or writing a computer program. Below is a FORTRAN program that determines the positions of the channel bottom following the computations described above. The input for this program to solve the above example is as follows:

50 3.171 4.016 4 0 6 1.5 100 65 .156 9.81 10

FORTRAN program that solves Subcritical Transition (TRANSIT.FOR)

```
      WRITE(*,*)' Give: Q,Yo1,Yo2,b1,m1,b2,m2,z1,L,HL,G,NSTA'
      READ(*,*) Q,YO1,YO2,B1,FM1,B2,FM2,Z1,FL,HL,G,NSTA
      FNSTM=FLOAT(NSTA)/2.
      DX=FL/FLOAT(NSTA)
      G2=2.*G
      A1=(B1+FM1*YO1)*YO1
      A2=(B2+FM2*YO2)*YO2
      VH1=(Q/A1)**2/G2
      VH2=(Q/A2)**2/G2
      WS1=Z1+YO1
      EL1=WS1+VH1
      EL2=EL1-HL
      WS2=EL2-VH2
      DEL=HL/FLOAT(NSTA)
```

```
      DWS=(WS2-WS1)/2.
      DB=(B2-B1)/2.
      DM=(FM2-FM1)/FLOAT(NSTA)
      WRITE(3,100)
100   FORMAT(' Sta   x  m  b  elev-EL  EL-ws  Vel.H  V  A  Y  z')
      WRITE(3,110) 'Beg',0.,FM1,B1,EL1,WS1,VH1,SQRT(G2*VH1),A1,YO1,Z1
110   FORMAT(1X,A3,F6.1,2F8.3,2F10.3,5F8.3)
      DO 10 I=2,NSTA-1
      FM=FM1+FLOAT(I-1)*DM
      EL=EL1-FLOAT(I-1)*DEL
      IF(I.LE.NSTA/2) THEN
      B=B1+(FLOAT(I-1)/FNSTM)**2*DB
      WS=WS1+(FLOAT(I-1)/FNSTM)**2*DWS
      ELSE
      B=B2-(FLOAT(NSTA-I)/FNSTM)**2*DB
      WS=WS2-(FLOAT(NSTA-I)/FNSTM)**2*DWS
      ENDIF
      VH=EL-WS
      V=SQRT(G2*VH)
      A=Q/V
      Y=(SQRT(B*B+4.*FM*A)-B)/(2.*FM)
      Z=WS-Y
10    WRITE(3,120) I,DX*FLOAT(I-1),FM,B,EL,WS,VH,V,A,Y,Z
120   FORMAT(I4,F6.1,2F8.3,2F10.3,5F8.3)
      WRITE(3,110)'End',FL,FM2,B2,EL2,WS2,VH2,SQRT(G2*VH2),
     &(B2+FM2*YO2)*YO2,YO2,EL1-HL-YO2
      END
```

5.5.1.2 Transition Design by Solving an ODE Problem

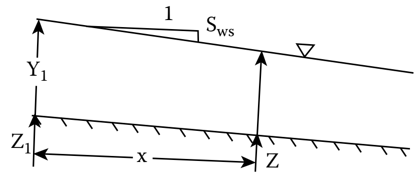
The problem of transition design can be viewed as a problem of solving an ordinary differential equation, just the same as computing GVF-profiles can be viewed as solving an ordinary differential equation. In the case of transition design, you have the choice of several variables that may be considered the dependent variable, e.g., the variable that will be solved. This variable may be the bottom width, b, as a function of x, the elevation of the water surface, ws-elev, as a function of x, or the elevation of the channel bottom, z, as a function of x. The bottom elevation, z, is probably the most viable of these variables. After the dependent variable is selected, the other variables must be specified. In other words, if z is selected as the dependent variable, then it is necessary to specify how b and the water surface elevation, ws-elev, change with x. In the hand design procedure described above, these are specified as steps 5 and 6, and if desired, two reversed parabolas may be used for these functional dependencies on x.

To develop the ordinary differential equation that will be used to design a channel transition with z as the dependent variable take the derivative of the energy equation with respect to x, or differentiate,

$$H = z + Y + \frac{Q^2}{2gA^2} \tag{5.13}$$

and relate y and z through the equation (for designs that call for the slope of the water surface to vary linearly from the beginning to the end of the transition)

$$Y = Y_1 + z_1 - xS_{ws} - z \tag{5.14}$$



where S_{ws} is the slope of the water surface, which is assume to be constant. After differentiation of Equation 5.13 with respect to x and substituting for dY/dx from Equation 5.14, the following first-order ordinary differential equation for z as a function of x results:

$$\frac{dz}{dx} = \frac{S_{ws}(1 - F_r^2) - S_f + \left[Q^2/(gA^3)\right](\partial A/\partial x)}{F_r^2} \tag{5.15}$$

This equation can be solved numerically using the same methods as used for solving the ODE for gradually varied flows, or for spatially varied flows. The solution is obtained by calling on an ODE solver such as ODESOL or DVERK. The program that call on this solver needs to have a main program that call on the solver, and a subroutine that defines the derivative dz/dx.

EXAMPLE PROBLEM 5.3

A transition from a trapezoidal channel with a bottom width of $b_1 = 10$ ft, a side slope of $m_1 = 2$, and a bottom slope of $S_{o1} = 0.0002706$ to a rectangular channel with a bottom width of $b_2 = 8$ ft and a bottom slope of $S_{o2} = 0.0034558$ is to be designed. The criteria are as follows: (a) the transition is to be 20 ft long, (b) the side slope m is to vary linearly across the transition, (c) the bottom width is to be defined by two reversed parabolas, and (d) the water surface is to follow a straight line through the transition. The design flow rate for this channel is 400 cfs, and the Manning's roughness coefficient is n = 0.013 for both upstream and downstream channels.

Solution

Solving Manning's equation for both the upstream and downstream channels produces Y_{o1} = 5.00 ft, and Y_{o2} = 4.00 ft, respectively. The upstream and downstream velocity heads are 0.248 and 1.917 ft, respectively. Using the average slope of the channels to estimate the drop in the energy line through the 20 ft long transition (and assuming there are no additional minor losses) gives a headloss, $h_L = 0.037$ ft, and therefore the change in the water surface will be 1.706 ft. The slope of the water surface, S_{ws}, through the transition should be 0.0853. A computer program that utilizes ODESOL to solve this problem is listed below. The prompt for input and the input for this problem consist of:

Input data

```
GIVE: IOUT,TOL,DELX,B1,B2,FM1,Q,FN,SWS,Y1,XBEG,XEND
       6 .0001 2   10  8  2 400 .013 .0853 5 0 20
```

FORTRAN program EXPR4-3.FOR listing

```
      EXTERNAL DZX
      REAL Z(1),ZPRIME(1),XP(1),YP(1,1),WK1(1,13)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV,XM,BM,B1,B2,DBH,XEND,
     &DFM,FM1,SWS,Y1,Q2G,QFN,B,A,Y
      WRITE(6,*)'GIVE:IOUT,TOL,DELX,B1,B2,FM1,Q,','FN,SWS,Y1,
     &XBEG,XEND'
      READ(5,*) IOUT,TOL,DELX,B1,B2,FM1,Q,FN,SWS,Y1,XBEG,XEND
```

```
      XM=.5*(XBEG+XEND)
      DM=.5*(B1+B2)
      DBH=2.*ABS(B1-B2)/(XEND-XBEG)**2
      DFM=FM1/ABS(XEND-XBEG)
      Q2G=Q*Q/32.2
      QFN=Q*FN/1.49
      X=XBEG
      Z(1)=.0
      E=Y1+Q2G/2./((B1+FM1*Y1)*Y1)**2
      WRITE(IOUT,100) X,Z,Y1,B1,E
100   FORMAT(1X,5F10.3)
2     XZ=X+DELX
      CALL ODESOL(Z,ZPRIME,1,X,XZ,TOL,.1,HMIN,1,XP,YP,WK1,DZX)
      X=XZ
      E=Y+Q2G/(2.*A*A)+Z(1)
      WRITE(IOUT,100) X,Z,Y,B,E
      IF(X .LT. XEND) GO TO 2
      STOP
      END
      SUBROUTINE DZX(X,Z,ZPRIME)
      REAL Z(1),ZPRIME(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV,XM,BM,B1,
     &B2,DBH,XEND,DFM,FM1,SWS,Y1,Q2G,QFN,B,A,Y
      IF(X .GT. XM) GO TO 10
      B=B1-DBH*X*X
      DAX=-(2.*DBH*X+DFM*Y)*Y
      GO TO 20
10    B=B2+DBH*(XEND-X)**2
      DAX=-(2.*DBH*(XEND-X)+DFM*Y)*Y
20    FM=FM1-DFM*X
      Y=Y1-SWS*X-Z(1)
      A=(B+FM*Y)*Y
      A3=A**3
      T=B+2.*FM*Y
      FR2=Q2G*T/A3
      P=B+2.*Y*SQRT(FM*FM+1.)
      SF=(QFN*(P/A)**.66666667/A)**2
      ZPRIME(1)=(SWS*(1.-FR2)-SF+Q2G*DAX/A3)/FR2
      RETURN
      END
```

The solution is:

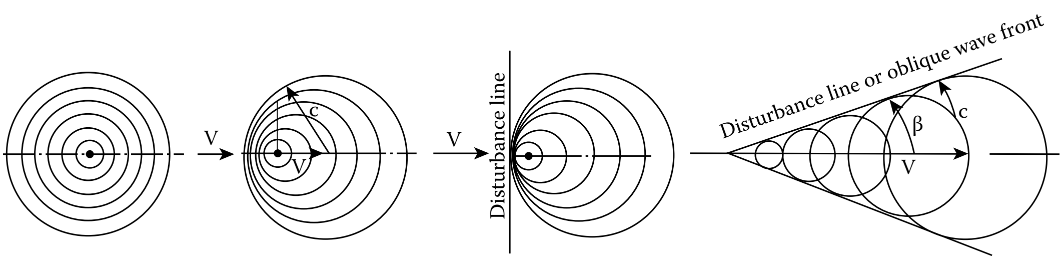
x	z	Y	b	h
0.000	0.000	5.000	10.000	5.248
2.000	0.492	4.338	9.960	5.248
4.000	0.656	4.003	9.840	5.246
6.000	0.665	3.823	9.640	5.244
8.000	0.575	3.743	9.360	5.242
10.000	0.404	3.743	9.000	5.239
12.000	0.174	3.803	8.640	5.235
14.000	–0.096	3.902	8.360	5.231
16.000	–0.406	4.042	8.160	5.227
18.000	–0.767	4.233	8.040	5.221
20.000	–1.201	4.496	8.000	5.215

The actual headloss across the transition from the above solution is $h_L = 5.248 - 5.214 = 0.034$ ft. The discrepancy between this value and that assumed from the average slope of the channel bottoms, or 0.0373, accounts for the reason why the depth at the end of the transition is 4.496 ft rather than 4.5 ft. A more accurate solution could now be obtained by using a headloss of 0.034 ft, which translates into a slope of water surface $S_{ws} = 0.0851$, instead of 0.0853 as used in the above solution. An alternative way of looking at the problem is that differential equation (Equation 5.15) giving dz/dx, which is solved simultaneously with the algebraic equation (Equation 5.11) giving Y. The Newton method could be used to solve these two simultaneous equations, but this approach is not merited since one adjustment based on the actual drop in energy line gives sufficient accuracy for practical engineering purposes.

5.5.2 Supercritical Transitions

The criteria that dictates the design of a supercritical transition is to eliminate, or minimize, the oblique waves that will result when the direction of a supercritical flow is changed. Since supercritical flows in channels can cause problem such as overtopping of sides due to the standing waves, etc., it is important to understand how supercritical flows behave through transition, bends, and control structures. Before getting into the actual design of transitions for supercritical flows, a discussion of the behavior of channel flows with Froude numbers larger than unity follows.

A simple physical explanation why a change in wall direction results in a standing oblique wave in supercritical flows, but not in subcritical flows, is that the celerity of a gravity wave is always less than the supercritical velocity. Therefore, any disturbance from the walls will be washed downstream. The speed, of celerity, of the gravity waves propagates in all direction so that at different times they move to circles of ever larger radii, but as the velocity of flow increases the effects are washed downstream. For flows with Froude numbers larger than unity disturbances are washed downstream so that they form a wave front or disturbance line, as shown below. When the velocity of the flow and celerity are equal this disturbance line is at right angles to the direction of flow. When the velocity of the flow is larger than the celerity, giving a Froude number greater than unity, the disturbance line is washed backward past the object that causes the disturbance because the fluid is moving faster than the signal gets to it that an object exists.

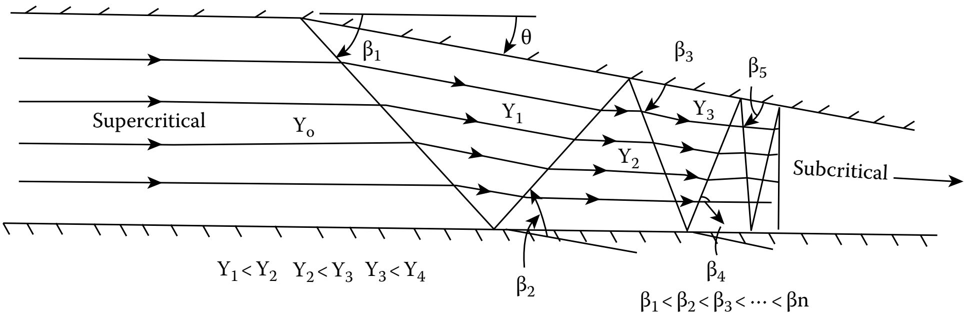


Stationary $F_r = 0$ | Subcritical flow $F_r < 1$ | Critical flow $F_r = 1$ | Supercritical flow $F_r > 0$

If the angle β denotes the angle of the disturbance line from the direction of the channel velocity, V, then

$$\sin\beta = \frac{c}{V} = \frac{\sqrt{gY}}{V} = \frac{1}{F_r} \tag{5.16}$$

An analogous phenomena occurs in supersonic gas flows. In gas flows, it is the celerity of a small amplitude pressure wave, and not the celerity of a small amplitude gravity wave that is compared with the velocity of the flow. An object moving through air, or air moving past the object, at a Mach number larger than unity will cause an oblique shock wave similar to the oblique standing wave. In the gas flow, an abrupt change in pressure takes place across the wave. In the channel flow, an abrupt change in depth takes place.



Channel flow with wall deflected inward by an angle θ

5.5.2.1 Channel Contractions

Consider what happens when one wall of a channel suddenly deflects inward toward the flow with an angle θ as shown above. The fluid particles upstream from the oblique wave receive no signal that something is happening downstream since their velocities are larger than the celerity, which provides them with this information. They therefore do not adjust gradually to downstream conditions but suddenly they flow into the oblique wave. Thus the streamlines are abruptly bent through an angle θ and the depth is suddenly increased. A short distance downstream from the position where the wall changes its direction, the oblique wave strikes the opposite wall and is reflected back. The depth upstream from this reflected wave is larger than the upstream depth in the undisturbed flow, the celerity in this region is larger, the Froude number is smaller, and therefore from the above equation $\beta_2 > \beta_1$. This process is repeated with each succeeding β getting larger and approaching $90° = \pi/2$ radian in the limit, if the convergent of the channel is long enough. When this occurs, the line of disturbance becomes normal to the direction of the flow, as shown in the previous sketch for critical flow. At this position, a hydraulic jump (a wave normal to the direction of flow) takes the flow to a subcritical condition. The problem is that sides cannot continue to converge and still allow the channel to accommodate the flow rate without the depth approaching infinity as the width of the channel approaches zero.

If both sides of the channel deflect inward with the same angle, θ, then the oblique waves will meet at the centerline of the channel and each wave will be reflected by the other to form the wave pattern shown below, i.e., duplicate the above pattern on both sides of the centerline. If the angles of inward deflection of the two channel walls are not equal a more complex asymmetrical wave pattern will be produced.

In the discussion that follows, it will be assumed that (1) the channels both upstream and downstream from the transition are rectangular and (2) the upstream channel is wider than the downstream channel so the transition reduces the size of the channel. The means for minimizing the height of oblique waves in the channel downstream of the transition is to make the length of the transition correct so that the positive oblique wave strikes the seat of a negative wave. Changes in bottom width are therefore made abrupt as shown in the sketch below. In other words, the sides of the channel are not changed gradually as is done for subcritical transitions, but rather occur abruptly so that a well-defined positive oblique wave occurs and a well-defined seat for a negative wave exists where the reduction in bottom width stops and the two sides of the channel again become parallel.

Thus the design calls for obtaining the length L where the oblique wave that originates at the beginning of the transition hit the point where the transition ends.

5.5.3 Design of Supercritical Transitions

To analyze this standing wave, a control volume will be taken that has a width of unity parallel to the wave and has its upstream section immediately in front of the change in water depth and its downstream section just downstream from where the depth increases, as shown in the sketch below by the rectangle abcd in the plan view. From symmetry note that the flow rate entering normal to face bc equal the flow rate leaving face ad. Therefore the continuity principles applied to this control volume gives

$$q = Y_1 V_1 \sin\beta = Y_2 V_2 \sin(\beta - \theta) \tag{5.17}$$

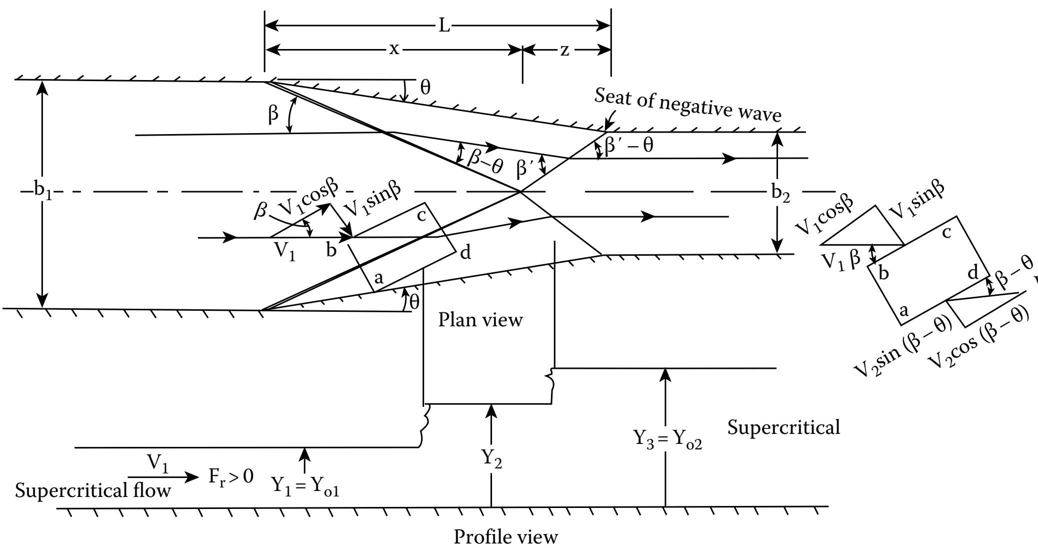
Applying the momentum principle in the tangential direction (or direction parallel to the oblique wave) results in the following equation since the forces on the two opposite faces in this direction are equal:

$$V_1 \cos\beta = V_2 \cos(\beta - \theta) \tag{5.18}$$

Applying the momentum principle in the normal direction to the oblique wave gives

$$\frac{\gamma Y_1^2}{2} - \frac{\gamma Y_2^2}{2} = \frac{\gamma q}{g}\left\{V_2 \sin(\beta - \theta) - V_1 \sin\beta\right\} \tag{5.19}$$

The flow rate per unit width q in the above equation is that normal to the direction of the oblique wave and is not the flow rate per unit width in the upstream channel. Since the depth Y_2 downstream from the wave is larger than the upstream depth Y_1 the left side of the equal sign is negative, and therefore $V_2 \sin(\beta - \theta) + V_1 \sin\beta$ and the deflection of the streamlines occurs in the direction shown.



With q defined by Equation 5.17, Equation 5.19 can be written as

$$\frac{(Y_1 - Y_2)(Y_1 + Y_2)}{2} = \frac{q^2}{gY_1Y_2}(Y_1 - Y_2) \tag{5.20}$$

After dividing by $(Y_1 - Y_2)$ and writing as a quadratic equation for Y_2, and solving with the quadratic formula the following results:

$$\frac{Y_2}{Y_1} = \frac{-1 + \left\{1 + 8q^2/(gY_1^3)\right\}^{1/2}}{2} = \frac{1}{2}\left\{\left\{-1 + \left[1 + \frac{8[V_1 \sin\beta]^2}{gY_1}\right]\right\}^{1/2}\right\} \tag{5.21}$$

Note that the first form of Equation 5.21 is that for a hydraulic jump in a rectangular channel. The difference is that the flow rate per unit width q is defined differently. Another useful form of Equation 5.21 is obtained by solving for sin β (which can be obtained more directly from Equation 5.20). The result is

$$\sin\beta = \frac{\left\{(1/2)(Y_2/Y_1)(Y_2/Y_1 + 1)\right\}^{1/2}}{F_{r1}} \tag{5.21a}$$

where $F_{r1} = V_1/\sqrt{gY_1}$.

Equation 5.17 can be used to obtain another equation for the depth ratio, or

$$\frac{Y_2}{Y_1} = \frac{V_1 \sin\beta}{V_2 \sin(\beta - \theta)} = \frac{V_1 \sin\beta}{[V_1 \cos\beta/\cos(\beta-\theta)]\sin(\beta-\theta)} = \frac{\tan\beta}{\tan(\beta-\theta)} \tag{5.22}$$

Note that the last form of this equation can also be obtained by dividing Equation 5.17 by Equation 5.18. Then by equating Equation 5.22 to Equation 5.21, the following implicit equation results that gives the angle β as a function of the upstream Froude number and the angle of the channel wall deflection θ.

$$\frac{\tan\beta}{\tan(\beta-\theta)} = \frac{1}{2}\left\{-1 + [1 + 8(F_{r1}\sin\beta)^2]^{1/2}\right\} \tag{5.23}$$

Thus with θ and $F_{r1} = V_1/\sqrt{gY_1}$ known β can be determined. Then from Equation 5.21 the depth ratio Y_2/Y_1 can be solved, and from Equation 5.18 V_2 determined.

It is useful to be able to solve for θ as a function of β and Y_2/Y_1. An equation that allows this can be obtained from the trigonometric identity

$$\tan(\beta - \theta) = \frac{\tan\beta - \tan\theta}{1 + \tan\beta\tan\theta}$$

Upon substituting this identity into the last expression of Equation 5.22 gives

$$\frac{Y_2}{Y_1} = \frac{\tan\beta(1 + \tan\beta\tan\theta)}{\tan\beta - \tan\theta}$$

then upon solving for tan θ results in

$$\tan\theta = \frac{((Y_2/Y_1) - 1)\tan\beta}{(Y_2/Y_1) + \tan^2\beta} \tag{5.24}$$

After the oblique standing wave arrives at the centerline of the transition, it encounters a symmetric wave from the opposite wall, resulting in another oblique wave downstream therefrom that increases the depth from Y_2 to Y_3. The same equations apply to this oblique wave as given above with the exception that the angle β', shown in the above sketch replaces β, and the ratio of depths become Y_3/Y_2.

From the geometry of the transition, additional equations can be added to the above equations that describe the behavior of the oblique wave. With x equal to the distance from the beginning of the transition to where the first oblique waves meet at the center of the channel, and z the distance from this point to the end of the transition, we get the following four equations:

$$L = x + z \quad \text{or} \quad z = L - x \tag{5.25}$$

$$L = \frac{0.5(b_1 - b_2)}{\tan\theta} \tag{5.26}$$

$$\frac{b_1}{2} = x\tan\beta \tag{5.27}$$

$$\frac{b_2}{2} = z\tan(\beta' - \theta) \tag{5.28}$$

If we assume that the upstream and downstream depths Y_1 and Y_3, as well as the flow rate Q, the bottom widths b_1 and b_2, are known so that F_{r1} can be computed then there are the following 7 unknown variables associated with the design of a supercritical transition:

$$L \quad \theta \quad Y_2 \quad \beta \quad \beta' \quad x \quad \text{and} \quad z$$

The seven equations needed to solve for these seven unknowns are: Equations 5.21a, 5.22a, and 5.24 through 5.28 applied for the oblique wave after the meeting of the two waves at the centerline of the channel, or

$$\frac{Y_3}{Y_2} = \frac{\tan\beta'}{\tan(\beta' - \theta)} \tag{5.22a}$$

Since the simultaneous solution of these seven equations is difficult without the use of computers, the typical procedure followed in the past involves a trial and error approach. Following the steps below generally results in rapid convergence to the proper length to use for a supercritical transition:

1. Solve for the normal depths in both the upstream and downstream channels, to get Y_1 and Y_3. (These will be based upon knowing Q, b_1, b_2, S_{o1}, S_{o2}, n_1, and n_2.) Then compute the upstream Froude number, F_{r1}.
2. Assume a depth Y_2 in the section of flow immediately downstream from the oblique wave and before this oblique wave is reinforced at the center of the channel.
3. Calculate the angle β of this oblique wave from Equation 5.21a, or from

$$\beta = \sin^{-1}\left[\left\{\left[\frac{1}{2}\frac{Y_2}{Y_1}\left(\frac{Y_2}{Y_1}+1\right)\right]^{1/2} \Big/ F_r^2\right\}\right]$$

4. Calculate the angle θ of the direction of the channel side contraction from Equation 5.24, or from

$$\theta = \tan^{-1}\frac{((Y_2/Y_1)-1)\tan\beta}{(Y_2/Y_1)+\tan^3\beta}$$

5. Calculate the length of the transition, L from Equation 5.26, or

$$L = \frac{0.5(b_1 - b_2)}{\tan\theta}$$

6. Calculate the distance x from the beginning of the transition to where the oblique waves meet at the center from Equation 5.27, or from

$$x = \frac{b_1}{2\tan\beta}$$

7. Calculate the distance z from where the oblique wave crosses at the centerline to the end of the transition from Equation 5.25, or from

$$z = L - x$$

8. Compute the angle β′, the angle of the direction of the oblique wave after having meet at the centerline of the channel from Equation 5.28, or from

$$\frac{b_2}{2} = z\tan(\beta' - \theta) \quad \text{or} \quad \beta' = \theta + \tan^{-1}\left\{\frac{b_2}{2z}\right\}$$

9. Compute the ratio of depths Y_3/Y_2 from Equation 5.22a, or from

$$\frac{Y_3}{Y_2} = \frac{\tan\beta'}{\tan(\beta' - \theta)}$$

10. Check $(Y_3/Y_2)Y_2 = Y_3$
11. Repeat steps 2 through 10 until close enough agreement is reached.

With use of computers, the seven equations that control the variables involved in the design of a supercritical transition can easily be solved. Below a listing is given of a FORTRAN function subprogram, FUN, that will replace that function in the SOLVE program whose listing is given on the CD in the folder PROGRAM_HWK. When this subprogram is attached to SOLVE, it will solve the problems that require the simultaneous solution of the above seven equations. The use of software packages such as TK-Solver, MATLAB, or MATHCAD are other viable options for solving such problems.

FUNCTION subprogram to use with SOLVE from Chapter 4.
Designs supercritical transitions that reduce size or rectangular sections (see program PRB5_66.FOR or PRB5_67.FOR on the CD in the folder PROGRAM_HWK.)

```
      FUNCTION FUN(I,X1,DX,J,N,KN)                  Known variables
      REAL X(10),X1(10),KN(20)                        KN(1) = b1
      DO 10 K=1,N                                     KN(2) = b2
```

```
10    X(K)=X1(K)                                              KN(3) = Fr1
      IF(J .GT. 0) X(J)=X(J)+DX                               KN(4) = Y1
      RY=X(3)/KN(4)                                           KN(5) = Y3
      GO TO (1,2,3,4,5,6,7),I
1     FUN=SIN(X(4))-SQRT(.5*RY*(RY+1.))/KN(3)
      GO TO 50
2     FUN=TAN(X(2))-(RY-1.)*TAN(X(4))/(RY+TAN(X(4))**2)
      GO TO 50                                           Unknown variables
3     FUN=X(1)-.5*(KN(1)-KN(2))/TAN(X(2))                     X(1) = L
      GO TO 50                                                X(2) = θ
4     FUN=X(1)-X(6)-X(7)                                      X(3) = Y2
      GO TO 50                                                X(4) = β
5     FUN=.5*KN(2)-X(7)*TAN(X(5)-X(2))                        X(5) = β'
      GO TO 50                                                X(6) = x
6     FUN=.5*KN(1)-X(6)*TAN(X(4))                             X(7) = z
      GO TO 50
7     FUN=KN(5)/X(3)-TAN(X(5))/TAN(X(5)-X(2))
50    RETURN
      END
```

EXAMPLE PROBLEM 5.4

Design a transition for a 10 ft wide rectangular channel to a 5 ft wide rectangular channel. The design flow rate is Q = 500 cfs, and the slopes of the channels are such that the normal depth in the 10 ft wide channel is 2 ft, and the normal depth in the downstream channel is 5 ft.

Solution

Using the program listed above the following input was used to solve the problem:

10 5 3.11528 2 5 /values of knowns/
220 .14 3 .4 .5 11 7 /guesses for unknowns/

The solution produced is: L = 13.156 ft, θ = 0.1878 rad, Y_2 = 3.371 ft, β = 0.504 rad, β' = 0.737 rad, x = 9.066 ft, and z = 4.090 ft.

The rule and variable sheets from TK-Solver for this problem are given below:

```
==================== VARIABLE SHEET ====================
St Input───── Name──── Output──── Unit────── Comment──
              B
              Bp
              theta
              Y2
              L
              x
              z
   10         b1
   5          b2
   32.2       g
   2          Y1
   25         V1
   5          Y3
              Fr1     3.1152799
              Fr3     1.5762208
```

```
================ RULE SHEET ================
S Rule————————————————————————————————————————
  SIN(B)^2=(Y2/Y1)*(Y2/Y1+1)/(2.*V1^2/(g*Y1))
  TAN(theta)=(Y2/Y1 1)*TAN(B)/(Y2/Y1+TAN(B)^2)
  L=x+z
  L=.5*(b1 b2)/TAN(theta)
  .5*b1=x*TAN(B)
  .5*b2=z*TAN(Bp theta)
  Y3/Y2=TAN(Bp)/TAN(Bp theta)
  Fr1=V1/sqrt(g*Y1)
  Fr3=b1*Y1*V1/(b2*Y3)/sqrt(g*Y3)
```

Extension of Problem

You should obtain a solution to this problem using an available software package that will accommodate nonlinear equations. As an additional exercise, determine the slopes of the upstream and downstream channels if the Manning's n = 0.013. Then add these equations to the system of equations being solved, and solve the problem of determining the length of the transition for a range of flow rates from 400 to 600 cfs in 50 cfs increments. How well would the transition designed for Q = 500 cfs serve this range of flow rates?

Solution to Extension

Solving Manning's equation twice gives $S_{o1} = 0.02973$ and $S_{o2} = 0.015492$. With minor modifications to the SOLVE FORTRAN program that places a DO statement therein to solve for the above 5 flow rates, or using the "list" capabilities of TK-SOLVE or MATLAB the following table of solutions is obtained:

Prb. No.	Q (cfs)	Y_1 (ft)	Y_3 (ft)	L (ft)	θ rad.	Y_2 (ft)	β rad.	β′ rad.	x (ft)	z (ft)
1	400.0	1.72	4.17	14.003	0.177	2.829	0.492	0.669	9.337	4.666
2	450.0	1.86	4.59	13.524	0.183	3.105	0.498	0.706	9.187	4.337
3	500.0	2.00	5.00	13.156	0.188	3.371	0.504	0.736	9.066	4.090
4	550.0	2.13	5.41	12.837	0.192	3.632	0.510	0.764	8.949	3.887
5	600.0	2.67	5.81	12.588	0.196	3.885	0.514	0.785	8.849	3.738

Note that the length of the transition becomes smaller with increasing flow rates, such that it is about 10% shorter for a 50% increase in flow rates. The depth changes across the oblique waves increase with increasing flow rate. The reason why larger flow rates (e.g., larger upstream depths) result in shorter transition lengths is that the celerity is larger with a larger angle β.

The variable and rule sheets from TK-Solver might look as follows (the variable sheet solves the problem with Q = 400 cfs.)

```
============== VARIABLE SHEET ============
St Input———— Name——— Output——— Unit—————
             B        .49204545
             Bp       .67052674
             theta    .17701062
             Y2       2.8309561
             L        13.975629
             x        9.3280379
             z        4.6475907
             Y1       1.7211621
             V1       23.240112
             Y3       4.174029
             Fr1      3.1217592
             Fr3      1.6532132
   10        b1
   5         b2
```

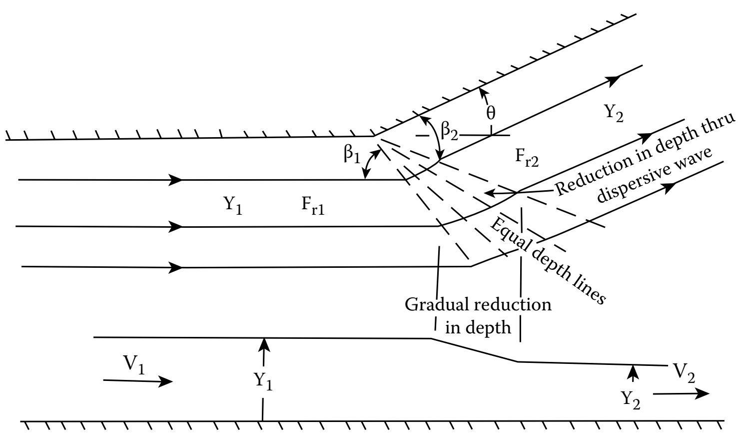
```
   32.2      g
   400       Q
   1.486     Cu
   .013      n
   .02973    So1
   .015492   So2

St Input———— Name——— Output——— Unit—————
═══════════════ RULE SHEET ═══════════════
S Rule————————————————————————————————————
  V1=Q/(b1*Y1)
SIN(B)^2=(Y2/Y1)*(Y2/Y1+1)/(2.*V1^2/(g*Y1))

TAN(theta)=(Y2/Y1 1)*TAN(B)/(Y2/Y1+TAN(B)^2)
  L=x+z
  L=.5*(b1 b2)/TAN(theta)
  .5*b1=x*TAN(B)
  .5*b2=z*TAN(Bp theta)
  Y3/Y2=TAN(Bp)/TAN(Bp theta)
  Fr1=V1/sqrt(g*Y1)
  Fr3=b1*Y1*V1/(b2*Y3)/sqrt(g*Y3)
Q=Cu/n*(b1*Y1)^1.666667/(b1+2*Y1)^.6666667*sqrt(So1)

Q=Cu/n*(b2*Y3)^1.666667/(b2+2*Y3)^.6666667*sqrt(So2)
```

5.5.4 Channel Enlargements

Consider next a channel wall that abruptly deflects outward from the direction of flow by an angle θ as shown below. At the point where the wall changes direction the disturbance line will extend into the channel with an angle β_1. This angle will be defined by the upstream velocity, V_1, and depth Y_1 identical to that for an inward deflection of the wall by the equation

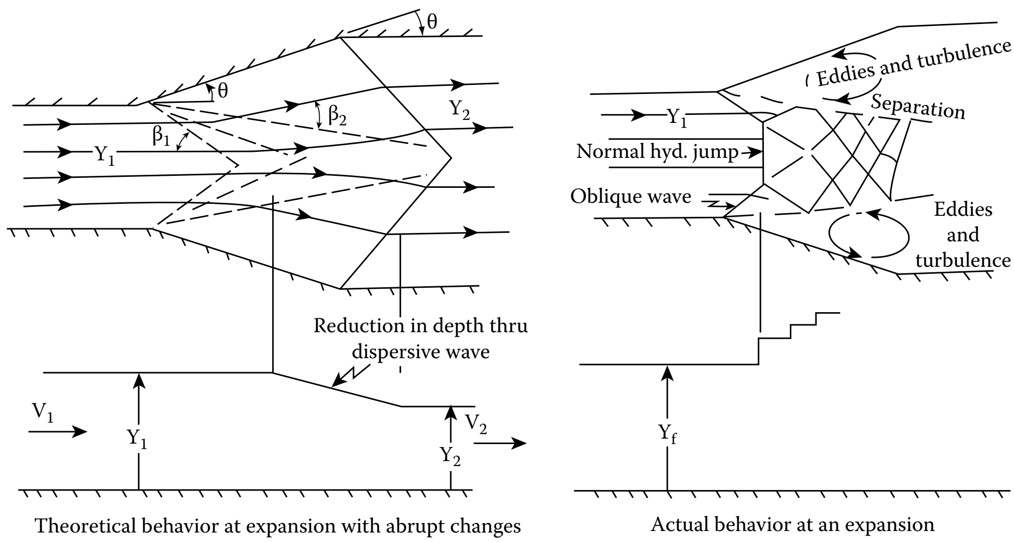
$$\sin\beta_1 = \frac{\sqrt{gY_1}}{V_1} = \frac{c_2}{V_2} = \frac{1}{F_{r1}} \tag{5.29}$$

However, there is no such occurrence as a "negative" hydraulic jump, or "negative" oblique wave. Rather than being a positive wave it is dispersive, and the depth decrease smoothly, even though the reduction in depth is rapid. The streamlines curve smooth through the zone of the dispersion until their direction is again in the direction of the outward deflecting wall. Thereafter the depth is constant again. The radial line that defines the terminus of the zone in which the depth decreases is at an angle of β_2 from the direction of the streamlines, or the direction of the deflected wall. This angle is given by



$$\sin\beta_2 = \frac{\sqrt{gY_2}}{V_2} = \frac{c_2}{V_2} = \frac{1}{F_{r2}} \tag{5.30}$$

Since a wave does not occur in this situation, an approximation of what occurs can ignore energy losses through the expansion. Equating $E_1 = E_2$, dividing by Y_1 and multiplying by 2 results in



Sketch showing streamlines and depth in supercritical flows through an expansion with abrupt angle changes

$$2 + \frac{V_1^2}{gY_1} = 2\frac{Y_2}{Y_1} + \frac{V_2^2}{gY_2}\frac{Y_2}{Y_1}(2 + F_{r2}^2) \quad \text{or} \quad \frac{Y_2}{Y_1} = \frac{2 + F_{r1}^2}{2 + F_{r2}^2} \tag{5.31}$$

The problem with the above analysis is that the fluid will separate from the walls unless the angle of outward deflection is very small. Theoretically if the fluid did deflect in the direction of the wall, then when the side of the channel walls become parallel again an oblique wave would form, giving the flow pattern depicted in the left portion of the sketch above. However, the real flow will look more like that in the right portion of the sketch. Physical and laboratory observations indicate that the flow within the separated portions behaves similarly to flows confined to contractions within sidewalls that contract as the separated flow does.

Even though it is not possible to precisely predict the behavior of supercritical flows at expansions on the basis of one-dimensional hydraulics, expansions in supercritical flow often occur in situations where large velocities from steep chutes, from gates, or spillways enter less steep channels whose widths must be sufficient to carry the flow rate. To prevent problems resulting from overtopping of the walls of the expansion or failure due to larger than anticipated forces on these walls the design should be conservative.

It is best to gradually increase the width of the channel with supercritical flows occurring upstream when an expansion is necessary. Otherwise separation is certain. If the length of the expansion is longer than needed, its cost will be increased. Using experimental results coupled with analytical studies, Rouse et al. (1951) suggest that the first portion of the dimensionless width $b' = b/b_1$ of such an expansion can be governed by the following equation:

$$b' = \frac{b}{b_1} = 1 + 0.25\left(\frac{x}{b_1 F_{r1}}\right)^{1.5} = 1 + 0.25(x')^{1.5} \tag{5.32}$$

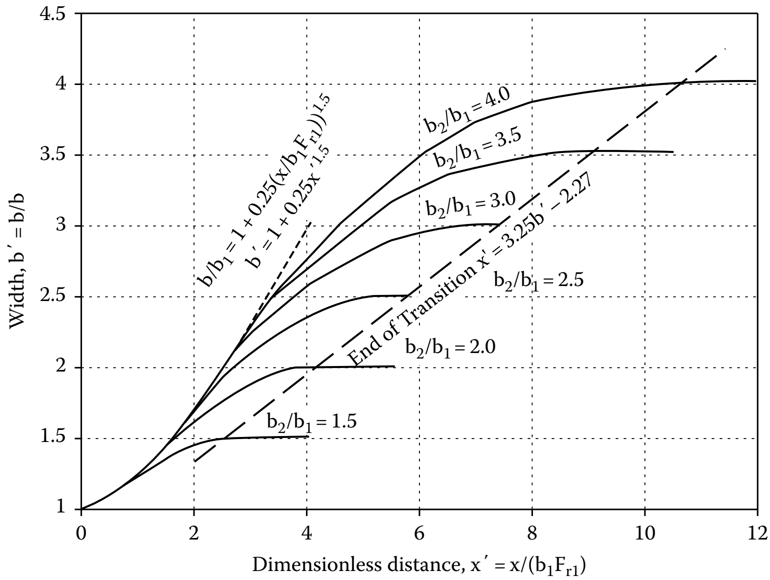


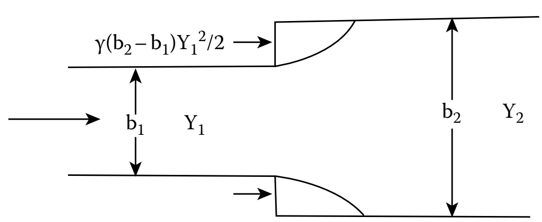
FIGURE 5.3 Transition shapes for supercritical expansion.

where x is measured in the downstream direction from the beginning of the transition. Note that the nature of the equation results in very small increases in the width as x is small, especially for flows with large Froude numbers, F_{r1} upstream. After expanding the width according to this equation for part of the transition length they recommend a well proportioned reversal of the wall curvature so that the bottom width takes the shape shown by one of the dimensionless curves shown in the figure above, depending upon the ratio of the downstream width b_2 to the upstream width b_1. Using this guide the length of expansion transitions is given by the equation

$$x' = 3.25b' - 2.27 \quad \text{or} \quad x = b_1F_{r1}\left(3.25\frac{b_2}{b_1} - 2.72\right) \tag{5.33}$$

which is shown in Figure 5.3 by a dashed line.

Our understanding of what occurs at enlargements when the upstream flow is supercritical can be improved by examining possible flow patterns through an abrupt enlargement of a rectangular channel. Let the bottom width abruptly increases from b_1 to b_2 as illustrated below in plan view.



The reason for taking an abrupt enlargement is that it can be readily analyzed. Since there is an unknown amount of energy loss as the larger upstream velocity, V_1, moves into the smaller velocity region of the downstream channel, the momentum principle will be used. We would expect the depths in the wake areas immediately downstream from the enlargement walls to be close to the upstream depth, thus applying the momentum principle $\Sigma F = (\gamma/g)Q^2(1/A_2 - 1/A_1)$ results in

$$\frac{\gamma b_1 Y_1^2}{2} + \frac{\gamma(b_2 - b_1)Y_1^2}{2} - \frac{\gamma b_2 Y_2^2}{2} = \frac{\gamma Q^2}{g}\left(\frac{1}{A_2} - \frac{1}{A_2}\right)$$

dividing by b_1 and letting $r = b_2/b_1$ results in

$$\frac{rY_1^2}{2} - \frac{rY_2^2}{2} = \frac{Q^2}{b_1^2 g}\left(\frac{1}{rY_2} - \frac{1}{Y_1}\right) = \frac{q_1^2}{g}\left(\frac{1}{rY_2} - \frac{1}{Y_1}\right)$$

let $y = Y_2/Y_1$ and divide by Y_1^2 results in the following dimensionless momentum equation:

$$r - ry^2 = 2F_{r2}^2\left\{\frac{1}{ry} - 1\right\} \tag{5.34}$$

This is a cubic equation that can be rewritten as

$$y^3 - \left(\frac{2F_{r1}^2}{r} + 1\right)y + \frac{2F_{r1}^2}{r^2} = 0 \tag{5.35}$$

Now write the energy equation across the abrupt enlargement to determine the energy loss, or $\Delta E = E_1 - E_2$, gives

$$\Delta E = Y_1 + \frac{Q^2}{2g(b_1Y_1)^2} - Y_2 - \frac{Q^2}{2g(b_2Y_2)^2} = Y_1 - Y_2 + \frac{Q^2}{2g(b_1Y_1)^2}\left(1 - \frac{1}{r^2y^2}\right)$$

Noting that since $F_{r1}^2 = (V_1^2/gY_1)$ that $Y_1 = V_1^2/(gF_{r1}^2)$ and $Y_2 = yV_1^2/(gF_{r1}^2)$ that the following dimensionless equation results:

$$\frac{\Delta E}{V_1^2/(2g)} = 1 - \frac{1}{(ry)^2} + \frac{2}{F_{r1}^2} - \frac{2y}{F_{r1}^2} \tag{5.36}$$

To solve a problem that specifies what the flow rate is and gives the depth Y_1 upstream of an abrupt enlargement one would first solve Equation 5.35 for the appropriate dimensionless depth y and from this obtain the downstream depth $Y_2 = yY_1$. Next this value of y would be substituted into Equation 5.36 and the energy loss computed.

EXAMPLE PROBLEM 5.5

Obtain a series of solutions in which the ratio $r = b_2/b_1$ varies from 1.2 to 2.2 and the upstream Froude number also varies. In these solutions obtain all roots of Equation 5.35. Also compute the energy loss that occurs through the enlargement. Explain which solutions are not feasible and why. Repeat this table of solutions for both subcritical and supercritical flow upstream.

Solution

An effective means for obtaining the three possible roots is to utilize the LAGUER subroutine described previously. A program to accomplish this is listed below. With the inputs shown below the following two tables are obtained for (a) upstream subcritical flow and (b) upstream supercritical flow, respectively.

FORTRAN listing to solve for downstream depths through an abrupt enlargement and the energy loss through it

Program EXPR5_5.FOR

```
      PARAMETER (ND=3,EPS=1.E-6)
      REAL YPA(3),EP(2),FR2(2),R(6),FR(6)
      COMPLEX C(ND+1),AD(51),Z1
      DATA R/1.2,1.4,1.6,1.8,2.0,2.2/,YPA/-2.,.302,2./
      READ(*,*) FR
      EPS1=2.*EPS*EPS
      DO 40 J=1,6
      DO 40 I=1,6
      II=3
      C(4)=CMPLX(1.,0.)
      C(2)=CMPLX(-2.*FR(I)/R(J)-1.,0.)
      C(3)=CMPLX(0.,0.)
      C(1)=CMPLX(2.*FR(I)/(R(J)*R(J)),0.)
      DO 20 K=1,ND+1
20    AD(K)=C(K)
      DO 30 K=ND,1,-1
      Z1=CMPLX(YPA(K),0.)
      CALL LAGUER(AD,ND,Z1,EPS,.FALSE.)WRITE(*,100) K,Z1
      YP=REAL(Z1)
      IF(YP.LE.0) GO TO 30
      II=II-1
      EP(II)=1.-1./(R(J)*YP)**2+2./FR(I)-2.*YP/FR(I)
      FR2(II)=FR(I)/(YP**3*R(J)**2)
30    YPA(K)=YP
40    WRITE(3,110) R(J),FR(I),YPA,EP,FR2
100   FORMAT(I5,2E12.5)
110   FORMAT(2F6.1,7F8.4)
      END
```

Upstream Subcritical Flows

		Roots of Equation 5.35			Energy Loss		Downstream	
b_2/b_1	F_{r1}	Y_2/Y_1	Y_2/Y_1	Y_2/Y_1	$\Delta E/\{V_1^2/(2g)\}$		F_{r2}	F_{r2}
1.2	0.1	−1.1353	0.1205	1.0148	−29.1979	0.0298	39.6410	0.0665
1.2	0.2	−1.2474	0.2159	1.0315	−6.0598	0.0321	13.8050	0.1265
1.2	0.3	−1.3453	0.2949	1.0504	−2.2860	0.0346	8.1258	0.1798
1.2	0.4	−1.4333	0.3617	1.0715	−1.1158	0.0375	5.8686	0.2258
1.2	0.5	−1.5139	0.4189	1.0951	−0.6334	0.0406	4.7245	0.2644
1.2	0.6	−1.5889	0.4679	1.1210	−0.3985	0.0441	4.0681	0.2958
1.4	0.1	−1.1112	0.0899	1.0212	−43.8960	0.0859	70.1693	0.0479
1.4	0.2	−1.2062	0.1620	1.0442	−10.0518	0.0905	23.9835	0.0896
1.4	0.3	−1.2906	0.2219	1.0687	−4.1710	0.0953	14.0013	0.1254
1.4	0.4	−1.3674	0.2726	1.0948	−2.2272	0.1003	10.0705	0.1555
1.4	0.5	−1.4384	0.3160	1.1224	−1.3725	0.1055	8.0821	0.1804
1.4	0.6	−1.5047	0.3534	1.1512	−0.9289	0.1109	6.9331	0.2006

(*continued*)

(continued)

Upstream Subcritical Flows

		Roots of Equation 5.35			Energy Loss		Downstream	
b_2/b_1	F_{r1}	Y_2/Y_1	Y_2/Y_1	Y_2/Y_1	$\Delta E/\{V_1^2/(2g)\}$		F_{r2}	F_{r2}
1.6	0.1	−1.0938	0.0697	1.0241	−60.6959	0.1462	115.133	0.0364
1.6	0.2	−1.1760	0.1266	1.0493	−14.6290	0.1519	38.4804	0.0676
1.6	0.3	−1.2500	0.1743	1.0757	−6.3522	0.1578	22.1281	0.0941
1.6	0.4	−1.3180	0.2150	1.1030	−3.5288	0.1638	15.7318	0.1164
1.6	0.5	−1.3812	0.2500	1.1312	−2.2500	0.1698	12.5000	0.1349
1.6	0.6	−1.4406	0.2805	1.1602	−1.5676	0.1759	10.6239	0.1501
1.8	0.1	−1.0808	0.0557	1.0251	−79.5563	0.2037	170.496	0.0286
1.8	0.2	−1.1530	0.1019	1.0511	−19.7572	0.2098	58.3821	0.0532
1.8	0.3	−1.2187	0.1410	1.0777	−8.7997	0.2161	33.0372	0.0740
1.8	0.4	−1.2796	0.1746	1.1050	−4.9944	0.2223	23.1837	0.0915
1.8	0.5	−1.3366	0.2039	1.1327	−3.2421	0.2285	18.2152	0.1062
1.8	0.6	−1.3903	0.2295	1.1609	−2.2929	0.2347	15.3255	0.1184
2.0	0.1	−1.0708	0.0455	1.0253		0.2562	264.697	0.0232
2.0	0.2	−1.1349	0.0838	1.0511	−25.4179	0.2625	84.8912	0.0431
2.0	0.3	−1.1940	0.1166	1.0774	−11.4977	0.2687	47.3063	0.0600
2.0	0.4	−1.2490	0.1450	1.1040	−6.6098	0.2748	32.7770	0.0743
2.0	0.5	−1.3008	0.1699	1.1309	−4.3365	0.2809	25.4704	0.0864
2.0	0.6	−1.3499	0.1919	1.1580	−3.0939	0.2869	21.2199	0.0966
2.2	0.1	−1.0629	0.0379	1.0250		0.3036	378.658	0.0192
2.2	0.2	−1.1205	0.0702	1.0503	−31.6004	0.3097	119.328	0.0357
2.2	0.3	−1.1740	0.0981	1.0759	−14.4370	0.3157	65.5641	0.0498
2.2	0.4	−1.2242	0.1226	1.1016	−8.3672	0.3216	44.8895	0.0618
2.2	0.5	−1.2716	0.1441	1.1275	−5.5261	0.3274	34.5230	0.0721
2.2	0.6	−1.3167	0.1632	1.1535	−3.9641	0.3331	28.4974	0.0808

Supercritical Flow Upstream

		Roots of Equation 5.35			Loss Energy		Downstream	
r	F_{r1}	y	y	y	$\Delta E/\{V_1^2/(2g)\}$		F_{r2}	F_{r2}
1.2	1.5	−2.1175	0.6885	1.4290	−0.0497	−0.0879	3.1920	0.3569
1.2	2.0	−2.3486	0.7313	1.6174	−0.0299	0.1172	3.5517	0.3283
1.2	2.5	−2.5546	0.7555	1.7991	−0.0210	0.1462	4.0259	0.2981
1.2	3.0	−2.7422	0.7708	1.9714	−0.0161	0.1737	4.5499	0.2719
1.2	3.5	−2.9156	0.7811	2.1345	−0.0131	0.1993	5.0996	0.2499
1.2	4.0	−3.0776	0.7886	2.2890	−0.0109	0.2230	5.6639	0.2316
1.4	1.5	−1.9790	0.5360	1.4430	−0.1571	0.1644	4.9694	0.2547
1.4	2.0	−2.1885	0.5796	1.6090	−0.0985	0.1940	5.2414	0.2450
1.4	2.5	−2.3759	0.6069	1.7690	−0.0705	0.2218	5.7047	0.2304
1.4	3.0	−2.5471	0.6254	1.9216	−0.0546	0.2474	6.2563	0.2157
1.4	3.5	−2.7056	0.6387	2.0669	−0.0444	0.2709	6.8551	0.2022
1.4	4.0	−2.8539	0.6485	2.2053	−0.0373	0.2924	7.4820	0.1903
1.6	1.5	−1.8712	0.4365	1.4346	−0.2985	0.2307	7.0431	0.1984

(continued)

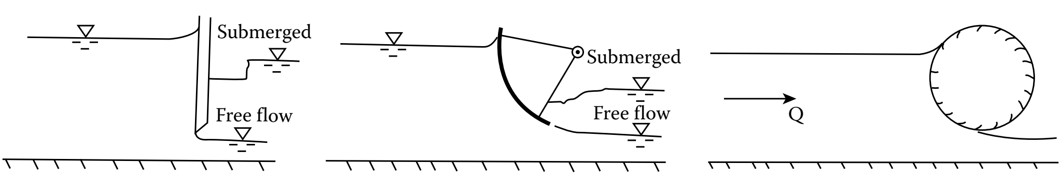
Supercritical Flow Upstream

		Roots of Equation 5.35			Loss Energy		Downstream	
r	F_{r1}	y	y	y	$\Delta E/\{V_1^2/(2g)\}$		F_{r2}	F_{r2}
1.6	2.0	−2.0633	0.4775	1.5858	−0.1905	0.2589	7.1738	0.1959
1.6	2.5	−2.2358	0.5046	1.7311	−0.1376	0.2848	7.5990	0.1882
1.6	3.0	−2.3936	0.5237	1.8699	−0.1070	0.3083	8.1612	0.1792
1.6	3.5	−2.5400	0.5376	2.0024	−0.0872	0.3298	8.7977	0.1703
1.6	4.0	−2.6772	0.5483	2.1289	−0.0735	0.3494	9.4787	0.1619
1.8	1.5	−1.7848	0.3655	1.4192	−0.4639	0.2878	9.4787	0.1619
1.8	2.0	−1.9625	0.4035	1.5589	−0.2989	0.3141	9.3938	0.1629
1.8	2.5	−2.1225	0.4295	1.6930	−0.2170	0.3379	9.7412	0.1590
1.8	3.0	−2.2692	0.4481	1.8211	−0.1691	0.3595	0.2897	0.1533
1.8	3.5	−2.4056	0.4621	1.9435	−0.1380	0.3791	0.9473	0.1471
1.8	4.0	−2.5336	0.4729	2.0606	−0.1163	0.3970	1.6703	0.1411
2.0	1.5	−1.7139	0.3122	1.4018	−0.6483	0.3371	2.3272	0.1361
2.0	2.0	−1.8794	0.3473	1.5321	−0.4200	0.3614	1.9363	0.1390
2.0	2.5	−2.0288	0.3718	1.6570	−0.3057	0.3834	2.1575	0.1374
2.0	3.0	−2.1662	0.3898	1.7764	−0.2385	0.4032	2.6622	0.1338
2.0	3.5	−2.2941	0.4035	1.8906	−0.1948	0.4211	3.3205	0.1295
2.0	4.0	−2.4142	0.4142	2.0000	−0.1642	0.4375	4.0711	0.1250
2.2	1.5	−1.6548	0.2706	1.3841	−0.8486	0.3800	5.6370	0.1169
2.2	2.0	−1.8097	0.3031	1.5065	−0.5515	0.4024	4.8339	0.1209
2.2	2.5	−1.9500	0.3263	1.6237	−0.4019	0.4226	4.8719	0.1207
2.2	3.0	−2.0793	0.3435	1.7358	−0.3137	0.4409	5.2978	0.1185
2.2	3.5	−2.1998	0.3567	1.8431	−0.2562	0.4574	5.9332	0.1155
2.2	4.0	−2.3132	0.3672	1.9460	−0.2160	0.4724	6.6941	0.1121

In examining the solutions in the above tables based on upstream subcritical flow, it can be observed that supercritical flow is not possible downstream. This is reflected by having a negative energy loss. Thus only the second root is valid. For the upstream supercritical root a negative energy loss occurs if the downstream flow is supercritical. Thus the conclusion is that the flow must be subcritical downstream, and again only the second root is possible.

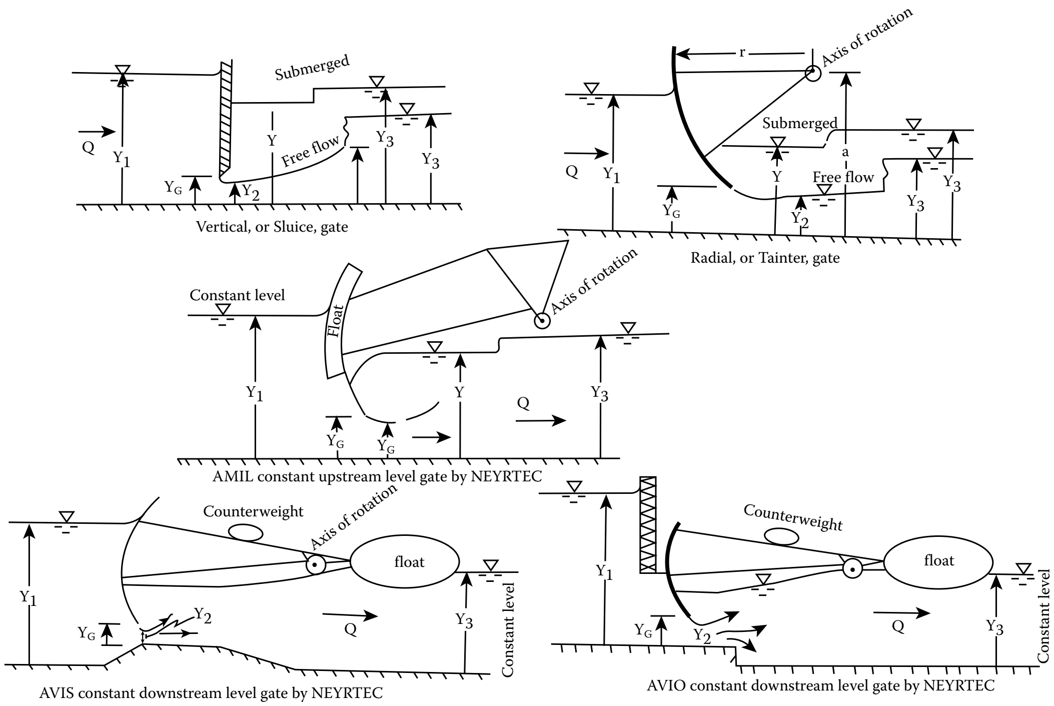
5.6 GATES

Gates are used to control the depth and flow rate in channels. A broad classification of gates is **overflow**, **underflow**, or **both over- and underflow**. Common types of underflow gates are illustrated in the sketch below and consist of (a) vertical (or a sluice gate), radial (known as Tainter gates) and drum gates.



Vertical gate Radial (Tainter) gate

Variations of these exit and have been developed for such purposes as automatically adjust their positions to maintain constant depths upstream, or downstream from the gate, by floats attached to the gate or other means. Three such gates are AMIL, AVIS, and AVIO developed by French engineers from NEYRTEC. The AMIL gate is designed to maintain a constant upstream water depth and uses a single float on the upstream face of the gate leaf to adjust the gate's height. The AVIS and AVIO gates are designed to maintain a constant downstream depth by means of a float attached downstream from the gates axis of rotation. The liquid flow past a gate is accelerated rapidly and the curvatures of the streamlines are too large to ignore and therefore the classification of these flows is rapidly varied flow. The methods of one-dimensional hydraulic must therefore be complemented with experimentally determined coefficients. Better theory for these flows can be developed using two-dimensional or three-dimensional formulations of the flow situations.



From the hydraulic viewpoint, the two most importance items of interest are (1) the relationship between the flow rate and the upstream head if the gate is free flowing, and the difference between the upstream and downstream heads if the flow is submerged, and (2) the forces on the gate. Forces on a gate are obtained by use of the momentum principle, which is discussed in Chapter 3, and will not be dealt with in this chapter. Practical means for determining the flow rate past a gate are based on the following weir formula if it is an overflow structure:

$$Q = C_d\left(\frac{2}{3}\right)b(2g)^{1/2}H^{3/2} = C_{d1}bH^{3/2} \tag{5.37}$$

and the following orifice formula if it is an underflow structure:

$$Q = C_d bY_G(2gY_1)^{1/2} = C_{d1}bY_G Y_1^{1/2} \tag{5.38}$$

where

H is the head over the crest of the gate

Y_G is the height the gate is above the channel bottom

Y_1 is the depth upstream from the gate

b is the width of the gate
C_d is a discharge coefficient that accounts for the contraction of the fluid after it leaves the gate, losses that occur as the flow passes, and two- (three-)dimensional flow effects

If accurate values for C_d are desired, they must be obtained experimentally for that particular gate operating under its field conditions. Both of these formulas assume that free flow occurs. The form of Equation 5.37 is still valid if the flow over a gate is submerged by high downstream water depths, but Equation 5.38 must be modified when the flow past an underflow gate is submerged by the downstream water depth to the form

$$Q = C_d b Y_G \left\{2g(Y_1 - Y_2)\right\}^{1/2} \tag{5.39}$$

where Y_2 is the downstream depth, which is larger than the depth that would occur if the flow were free flowing, and is generally larger than the height of the gate above the channel bottom.

For a gate under free flowing conditions, an approximation of the discharge coefficient can be obtained assuming that no headloss occurs across the gate, i.e., $E_1 = E_2$, which for a rectangular gate per unit width becomes

$$Y_1 + \frac{q^2}{2gY_1^2} = Y_2 + \frac{q^2}{2gY_2^2}$$

The downstream depth Y_2 can be replaced by the contraction coefficient C_c times the height of the gate Y_G, or $Y_2 = C_c Y_G$. Solving for q this energy equation becomes

$$q = Y_1 Y_2 \left\{\frac{2g}{Y_2 - Y_1}\right\}^{1/2} = C_c Y_G Y_1 \left\{\frac{2g}{C_c Y_G + Y_1}\right\}^{1/2} = C_u Y_G \left\{\frac{2gY_1}{C_c Y_G / Y_1 + 1}\right\}^{1/2}$$

Comparing this result with Equation 5.38 the discharge coefficient is given by

$$C_d = \frac{C_c}{(1 + C_c Y_G Y_1)^{1/2}} \quad \text{or} \quad C_{d1} = \left\{\frac{2g}{1 + C_c Y_G / Y_1}\right\}^{1/2} \tag{5.40}$$

In practice the values for C_d would be expected to be slightly smaller than given by Equation 5.40. The subject of submerged flow past vertical gates is dealt with in a subsequent section.

For radial gates, the discharge coefficient would be expected to be a function of its radius, r, and the distance the axis of rotation of the gate is above the channel bottom, a. Using as dimensionless parameters, $a' = a/r$, $Y_1' = Y_1/r$ and $Y_G' = Y_G/r$, and using results from Toch (1955), the discharge coefficient of Tainter gates can be approximated by the following equation when free flow conditions exist:

$$C_d = 0.489 + .11a' + .116(a')^2 - .246Y_G' + .261(Y_g')^2 + .105Y_1' - .025(Y_1')^2 - .391a'Y_G' + .0149a'Y_1' \tag{5.41}$$

For AVIO and AVIS gates, the discharge past the gate is expected to be related to the square root of the difference between the upstream depth Y_1 and the downstream depth Y_3 since they behave as orifices. An appropriate formula for steady-state flow conditions is

$$Q = KbR^2 \left(\frac{\Delta z}{\Delta z_{max}}\right) \{Y_1 - Y_3\}^{\frac{1}{2}} \tag{5.42}$$

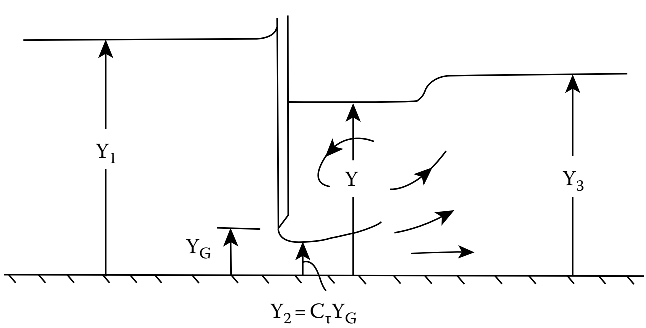
where K is a constant depending on the gate type and whether ES or SI units are used. R is the float radius (ft or m), and Δz and Δz_{max} are the decrement and maximum decrement. Typically Δz_{max} equals 5% of the float radius R, or $\Delta z_{max} = 0.05R$, and the decrement $\Delta z = 0$ corresponds to a closed gate. Values of K are:

Gate Type	ES	SI
AVIS	2.3	4.1
High-head AVIO	1.8	1.0
Low-head AVIO	3.6	2.0

5.7 SUBMERGED FLOW DOWNSTREAM FROM VERTICAL GATES

Often in practice a series of gates along a channel system are used to control flow rates and keep depths sufficiently large to divert water into the turnout structures between these gates. When this occurs, then the flow downstream from the gates likely submerges the bottoms of the gates. Whenever the momentum function for the flow immediately downstream from a gate is larger than the momentum function associated with the alternate depth (supercritical depth) to the depth upstream on the gate, then the flow downstream of the gate will be submerged. When submerged flow exists by a gate, the flow is subcritical downstream of the gate as well as subcritical upstream. When this occurs, the gate acts as a loss device in the channel, and the amount of this loss will depend not only upon the position of the gate and the flow rate, but also on the type of gate and its specific design. Generally either field data, or laboratory model studies, are needed to define this loss and the variables (or dimensionless parameters) it depends upon. For flow situations in which a clearly defined jet flow occurs beneath the gate similar to free flow except it is over ridden by the downstream subcritical depth as shown in the sketches below, then the results of the following analysis, which uses both the energy and momentum principles, will give reasonable answers.

The treatment in the following sections deals specifically with submerged flow past vertical gates, such as sluice gates. However, when the conditions are similar to those described for other gates, then these result can be used for other types of gates. These basic conditions are that the fluid emerges from the bottom of the gate as a jet that flows under the downstream flow with an over-riding roller, and that submergence depth immediately downstream from the gate is less than the depth downstream a short from the gate. This difference in depth is caused by the high-velocity fluid emerging from below the gate.



The special specific energy per unit weight of fluid in the jet emerging beneath the gate can be taken equal to that upstream or

$$E_1 = E_2 \quad \text{or} \quad Y_1 + \frac{q^2}{2gY_1^2} = Y + \frac{q^2}{2gY_2^2} \tag{5.43}$$

Note that depth Y (not Y_2) represents the elevation head above the channel bottom immediately downstream from the gate at section 2. For this reason, it is called the special specific energy. Under some conditions, submerged flow downstream from a gate is influenced significantly by the two- and three-dimensional effects that use of a one-dimensional hydraulic equation, such as Equation 5.43, can be questioned. Therefore, what follows can be refined by using experimental coefficients for gates of various designs. Rewriting Equation 5.43 gives

$$Y_1 - Y = \frac{q^2}{2g}\frac{Y_1^2 - Y_2^2}{Y_1^2 Y_2^2} \tag{5.44}$$

Applying the momentum function $m_2 = m_3$ in a like manner gives

$$\frac{Y^2}{2} + \frac{q^2}{gY_2} = \frac{Y_3^2}{2} + \frac{q^2}{gY_3} \tag{5.45}$$

or solving for q^2/g results in

$$\frac{q^2}{g} = \frac{1}{2}\frac{\left(Y_3^2 - Y^2\right)(Y_2 Y_3)}{Y_3 - Y_2} \tag{5.46}$$

Substituting this momentum function expression for q^2/g into Equation 5.44 results in the following quadratic equation for the depth Y if Y_1, Y_2 and Y_3 are known:

$$\frac{Y_3\left(Y_1^2 - Y_2^2\right)}{4Y_1^2 Y_2(Y_3 - Y_2)}\left(Y_3^2 - Y^2\right) + Y - Y_1 = 0 \tag{5.47}$$

Equation 5.47 might be solved by the quadratic formula and after Y is determined its value can be substituted into either Equations 5.44 or 5.46 and q solved.

Available laboratory data by Rajaratnam allow the applicability of both the special specific energy and special momentum functions to be ascertained. His data were obtained in an 18 in. wide, 16 ft long rectangular flume with a smooth aluminum bed and Plexiglas's sides. The gate was a 1/4 in. thick aluminum plate with a sharp lower edge. The table below provides a comparison of this laboratory data for submerged flow. The laboratory data give the position of a vertical gate Y_G in a rectangular channel, the flow rate per unit width q, the depth Y immediately downstream from the gate, the depth upstream from the gate Y_1, and the tailwater depth Y_3. To determine the depth Y_2 emerging from the gate the contraction coefficient was computed from $C_c = 0.59 + 0.022Y_G/Y_1$ (using the computed Y_1), and Equation 5.44 is solved for the depth Y_1 upstream from the gate in the 7th column of the table below. The difference between these computed values and the observed values for the upstream depth Y_1 are given in column 8 with the percent difference given in the next column. The average absolute difference is 1.1%. The columns in the table before the energy loss columns are used to check the momentum Equation 5.45. For this check the downstream depth Y_3 was computed from Equation 5.45 using Y_2 as computed above, and the measured depth Y. The average difference 2.3%. The last three columns deal with energy loss from upstream of the gate to the downstream position 3. More about energy loss will be given later, since this is an important item associated with submerged flow past gates. The first such column gives the loss obtained from the laboratory data, the second such column using the computed Y_1 and Y_3 (along with the laboratory q to obtain this energy loss) and the last column gives the percent difference between the laboratory energy loss that the energy loss computed. This last column shows the largest difference between the laboratory data and that given by the above special energy and momentum equations.

Comparison of Computed Upstream Depth from Modified Energy across Gate with Rajaratnam Laboratory Data

Obs No.	Lab. Y_G (ft)	Y_2 (ft)	Lab. q (cfs/ft)	Lab. Y (ft)	Lab. Y_1 (ft)	Comp. Y_1 (ft)	Dif. (ft)	Dif. %	Lab. Y_3 (ft)	Comp. Y_3 (ft)	Dif. (ft)	Dif. %	Energy Loss 1 to 3 ΔE_L	ΔE	% dif
1	0.250	0.149	0.735	0.867	1.217	1.241	−0.024	−2.01	0.930	0.971	−0.041	−4.38	0.283	0.267	5.58
2	0.250	0.148	0.735	1.183	1.550	1.561	−0.011	−0.70	1.214	1.265	−0.051	−4.18	0.334	0.294	11.81
3	0.250	0.148	0.735	1.433	1.808	1.812	−0.004	−0.23	1.473	1.503	−0.029	−2.00	0.334	0.308	7.57
4	0.250	0.148	0.730	1.650	2.025	2.025	0.000	0.01	1.676	1.711	−0.035	−2.08	0.348	0.313	10.03
5	0.250	0.148	0.723	2.125	2.475	2.494	−0.019	−0.77	2.140	2.173	−0.033	−1.52	0.335	0.321	4.04
6	0.250	0.149	0.735	0.833	1.200	1.207	−0.007	−0.61	0.915	0.940	−0.025	−2.79	0.281	0.263	6.32
7	0.250	0.148	0.950	1.450	2.092	2.085	0.006	0.31	1.535	1.564	−0.029	−1.87	0.554	0.519	6.30
8	0.250	0.148	0.937	1.975	2.608	2.595	0.013	0.51	2.030	2.060	−0.030	−1.47	0.577	0.534	7.42
9	0.250	0.148	0.955	1.071	1.704	1.710	−0.006	−0.34	1.183	1.217	−0.034	−2.88	0.516	0.488	5.40
10	0.250	0.148	0.955	0.858	1.500	1.495	0.005	0.34	0.996	1.031	−0.035	−3.56	0.496	0.457	7.97
11	0.250	0.149	0.960	0.675	1.308	1.315	−0.007	−0.53	0.845	0.881	−0.036	−4.25	0.452	0.424	6.08
12	0.250	0.149	0.960	0.529	1.158	1.166	−0.008	−0.66	0.742	0.768	−0.027	−3.61	0.401	0.384	4.36
13	0.250	0.149	0.960	0.400	1.042	1.033	0.009	0.87	0.675	0.678	−0.003	−0.51	0.348	0.337	3.42
14	0.500	0.303	0.965	0.525	0.650	0.647	0.003	0.38	0.590	0.609	−0.019	−3.28	0.053	0.034	36.05
15	0.500	0.301	0.965	0.742	0.883	0.882	0.001	0.10	0.817	0.819	−0.003	−0.34	0.064	0.060	5.50
16	0.500	0.300	0.965	0.942	1.100	1.090	0.010	0.90	1.008	1.011	−0.003	−0.27	0.089	0.077	13.82
17	0.500	0.299	0.965	1.267	1.425	1.421	0.004	0.25	1.308	1.325	−0.016	−1.24	0.115	0.096	16.94
18	0.500	0.300	0.962	1.000	1.158	1.149	0.009	0.80	1.058	1.067	−0.008	−0.79	0.098	0.081	17.70
19	0.500	0.298	0.950	1.475	1.633	1.627	0.006	0.38	1.490	1.525	−0.035	−2.37	0.142	0.101	29.00
20	0.500	0.303	0.965	0.596	0.725	0.726	−0.001	−0.19	0.662	0.679	−0.016	−2.49	0.057	0.043	23.89
21	0.333	0.199	1.035	0.600	1.017	1.003	0.014	1.34	0.785	0.780	0.005	0.59	0.221	0.212	4.00
22	0.333	0.200	1.035	0.400	0.800	0.790	0.010	1.22	0.620	0.621	−0.001	−0.21	0.163	0.152	6.31

23	0.333	0.199	1.035	0.725	1.142	1.133	0.009	0.78	0.878	0.886	−0.008	−0.89	0.255	0.238	6.31
24	0.333	0.198	1.025	1.175	1.592	1.584	0.008	0.50	1.271	1.288	−0.017	−1.36	0.317	0.292	7.84
25	0.333	0.198	1.025	1.458	1.883	1.870	0.013	0.71	1.533	1.554	−0.020	−1.34	0.348	0.314	9.68
26	0.167	0.099	0.811	1.108	2.133	2.156	−0.023	−1.08	1.248	1.269	−0.021	−1.66	0.881	0.883	−0.27
27	0.167	0.099	0.805	1.633	2.608	2.668	−0.059	−2.28	1.720	1.747	−0.027	−1.59	0.886	0.919	−3.62
28	0.167	0.099	0.823	0.558	1.608	1.634	−0.026	−1.59	0.815	0.829	−0.014	−1.72	0.782	0.794	−1.53
29	0.167	0.099	0.710	0.417	1.225	1.213	0.012	1.01	0.685	0.666	0.019	2.79	0.529	0.534	−1.12
30	0.167	0.099	0.523	0.175	0.583	0.593	−0.010	−1.68	0.404	0.399	0.005	1.34	0.166	0.180	−8.50
31	0.167	0.099	0.518	0.242	0.725	0.655	0.070	9.69	0.477	0.433	0.043	9.07	0.238	0.209	12.20
32	0.083	0.049	0.341	0.360	1.100	1.101	−0.001	−0.11	0.496	0.512	−0.016	−3.23	0.598	0.584	2.39
33	0.083	0.049	0.341	0.767	1.508	1.510	−0.001	−0.09	0.841	0.852	−0.011	−1.33	0.666	0.656	1.46
34	0.083	0.049	0.374	0.850	1.725	1.745	−0.020	−1.14	0.910	0.943	−0.033	−3.65	0.813	0.800	1.66
35	0.083	0.049	0.374	1.617	2.467	2.513	−0.046	−1.86	1.642	1.669	−0.027	−1.65	0.825	0.843	−2.28
36	0.083	0.049	0.640	1.017	3.508	3.643	−0.134	−3.83	1.190	1.237	−0.047	−3.95	2.314	2.402	−3.79

Source: Rajaratnam, N. and Subramanya, K., *J. Irrigat. Drain. Div*., ASCE, 93(IR3), 167–186, September 1967.

Notes related to table: The average difference of 1.1% between the laboratory data, and the upstream depth computed from the special energy equation suggests the theory involved is applicable. The modestly large percent difference of 2.3% associated with the use of the special momentum function also suggest its theory is sound. Because of the larger fluctuations of the downstream depth Y_3, than those with Y_1 one would suspect larger errors in the laboratory value of Y_3. The larger percent differences between the computed and measured energy losses, are in part due to the fact that the magnitude of the energy loss is smaller. The largest difference of 36% is associated with a measured $\Delta E = 0.053$, and might be associated with accuracy of measurement in the 2nd digit beyond the decimal point. Thus the conclusion is that this laboratory data validates the use of the special energy and momentum equation for submerged flow past vertical gates.

5.7.1 Dimensionless Forms of Equations

It is useful, however, to develop a dimensionless form of Equation 5.47. Let $y = Y/Y_1$, $y_2 = Y_2/Y_1$ and $y_3 = Y_3/Y_1$. Dividing Equation 5.47 by Y_1 gives the following dimensionless quadratic equation for y:

$$\frac{1}{4}\frac{y_3}{y_2}\frac{\left(1-y_2^2\right)}{y_2-y_3}\left(y^2-y_3^2\right)+y-1=0 \tag{5.48}$$

$$\text{Letting } b=\frac{4y_2(y_3-y_2)}{y_3\left(1-y_2^2\right)}=\frac{4y_2(1-y_2/y_3)}{1-y_2^2} \tag{5.49}$$

allows Equation 5.48 to be written as

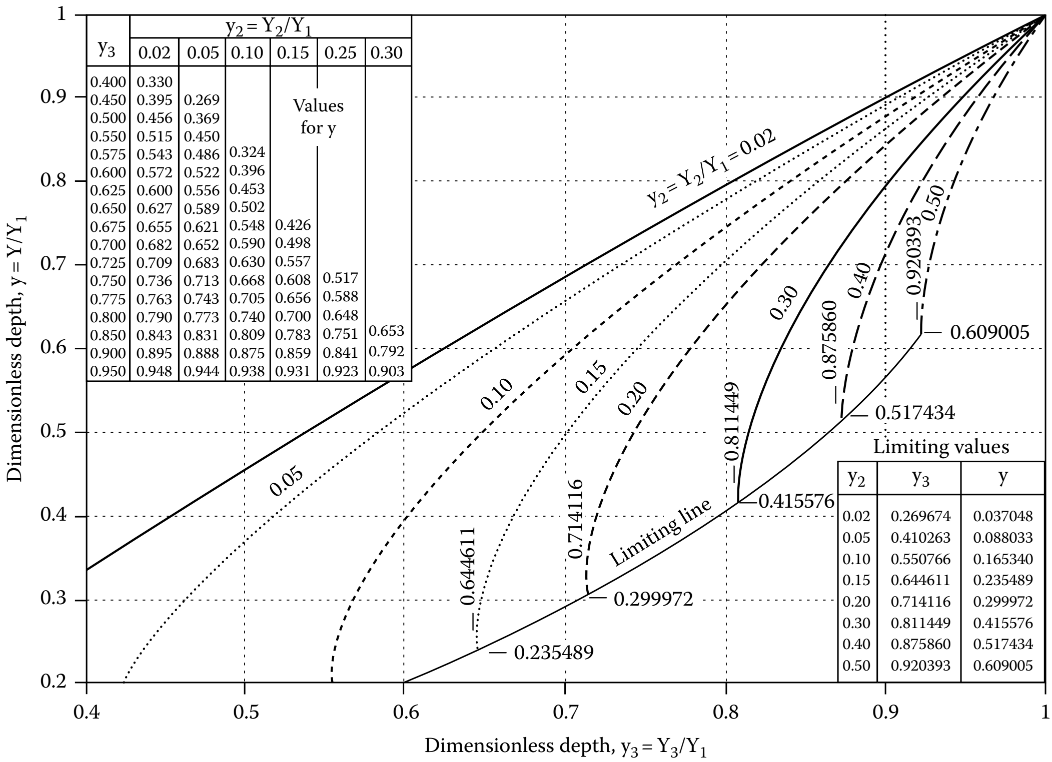
$$y^2-by-\left(y_3^2-b\right)=0 \tag{5.50}$$

which can be solved directly by the quadratic formula to give

$$y=\frac{\left\{b+\sqrt{b^2+4\left(y_3^2-b\right)}\right\}}{2} \tag{5.51}$$

Figure 5.4 gives the solution to Equation 5.51 with the dimensionless depth y_3 as the abscissa, the dimensionless depth y as the ordinate, and y_2 as a series of curves. This graph also shows the limiting values of y_3 and y for the y_2 curves when the argument of the square root in Equation 5.51 becomes negative. In other words at these, and smaller values, of y_3 no simultaneous solution exist for the special specific energy and momentum Equations 5.43 and 5.45. For example, if $y_2 = 0.2$, then for $y_3 \le 0.7141$ (for which value $y = 0.30$), it is not possible to solve quadratic Equation 5.51.

Another means that one might use to determining the maximum depth that can exist downstream from a gate before it becomes submerged is to determine the conjugate depth to that of the downstream depth Y_3, i.e., solve for the depth upstream of the hydraulic jump, Y_{u3}. If this depth is larger than Y_2, the flow will be free flowing since the depth will need to increase along an M_3 GVF profile downstream of the gate. If Y_{u3} is less than Y_2, then one might conclude the gate will be submerged. However, the limiting depth Y_3 determine through this procedure will not agree with that determined when the quantity under the square root in Equation 5.51 is zero. The discrepancy is due to using the special momentum and energy equations to define submerged conditions when the square root is set to zero. While the design of the gate will likely have an influence on what downstream depth actually separates free flow from submerged flow, general observations of gates operating under conditions just before submergence occurs suggest that it is possible for the computed Y_{u2} to be less than Y_2 before submergence occurs because the high velocity of the flow from under the gate is able to prevent the jump for being trapped at the gate resulting in submerged flow. On the other hand, going from a submerged condition toward a free flowing conditions may retain submergence after Y_{u2} becomes larger than Y_2. Notice how rapidly the different curves for different value of y_2 rise from the limiting line. This rapid rise indicates that once submergence occurs, a very small increase in the downstream depth represented by the abscissa y_3 will cause a large increase in the depth of submergence y. In other words, the depth of submergence Y is greatly affected by small changes in the downstream conditions especially when submergence first occurs. Thus one would expect unstable operating conditions behind gates when submergence is in its beginning stages. Another item of interest that can be gleaned from examining Figure 5.4 is that generally the larger the dimensionless depth y_2 is the larger the difference will be between the dimensionless submergence depth y and the downstream depth y_3. These two depths (y and y_3) are equal along a straight



y_3	$y_2 = Y_2/Y_1$					
	0.02	0.05	0.10	0.15	0.25	0.30
0.400	0.330					
0.450	0.395	0.269				
0.500	0.456	0.369				
0.550	0.515	0.450				
0.575	0.543	0.486	0.324			
0.600	0.572	0.522	0.396			
0.625	0.600	0.556	0.453			
0.650	0.627	0.589	0.502			
0.675	0.655	0.621	0.548	0.426		
0.700	0.682	0.652	0.590	0.498		
0.725	0.709	0.683	0.630	0.557		
0.750	0.736	0.713	0.668	0.608	0.517	
0.775	0.763	0.743	0.705	0.656	0.588	
0.800	0.790	0.773	0.740	0.700	0.648	
0.850	0.843	0.831	0.809	0.783	0.751	0.653
0.900	0.895	0.888	0.875	0.859	0.841	0.792
0.950	0.948	0.944	0.938	0.931	0.923	0.903

Values for y

Limiting values

y_2	y_3	y
0.02	0.269674	0.037048
0.05	0.410263	0.088033
0.10	0.550766	0.165340
0.15	0.644611	0.235489
0.20	0.714116	0.299972
0.30	0.811449	0.415576
0.40	0.875860	0.517434
0.50	0.920393	0.609005

FIGURE 5.4 Solution of dimensionless depth Y immediately downstream from gate based on dimensionless depth y_3 and y_2. (Assumption is that jet flow occurs beneath gate so that special use of energy and momentum equations are valid.)

line from the ordinate of 0.4 to the upper right hand corner of the graph where both ordinate and abscissa are 1. (This line is not shown on Figure 5.4.) Thus the further the curves are below this line the bigger the difference in depths y and y_3. Obviously all of the curves on Figure 5.4 must be below this line since Y_3 must always be larger than Y if the size of the channel does not change. Furthermore, neither Y nor Y_3 can be larger than Y_1.

A dimensionless form of the special momentum Equation 5.46 is

$$\frac{q^2}{gY_1^3} = F_{r1}^2 = \frac{\left(y_3^2 - y^2\right)y_3}{2\left(y_3/y_2 - 1\right)} = \frac{\left(y_3^2 - y^2\right)y_2 y_3}{2(y_3 - y_2)} \tag{5.52}$$

and a plot of this equation (with y solved by Equation 5.51) is given in right portion of Figure 5.5. A dimensionless form of the special energy Equation 5.44 is

$$\frac{q^2}{gY_1^3} = F_{r1}^2 = \frac{2(1-y)}{\left(1/y_2^2 - 1\right)} = \frac{2(1-y)y_2^2}{\left(1 - y_2^2\right)} \tag{5.53}$$

and this equation is plotted in left portion of Figure 5.5. Using a common ordinate, F_{r1}^2, allows for q to be obtained from the graph using y_2 (upstream depth, gate setting (and contraction coefficient) and flow depth immediately downstream from the gate) after y has been obtained from Figure 5.4 instead of solving Equation 5.53, or from the graph using y_3 and y_2 (the dimensionless momentum downstream from gate, i.e., the upstream depth, the gate setting [and its contraction coefficient] and the downstream depth Y_3), instead of solving Equations 5.51 and 5.52.

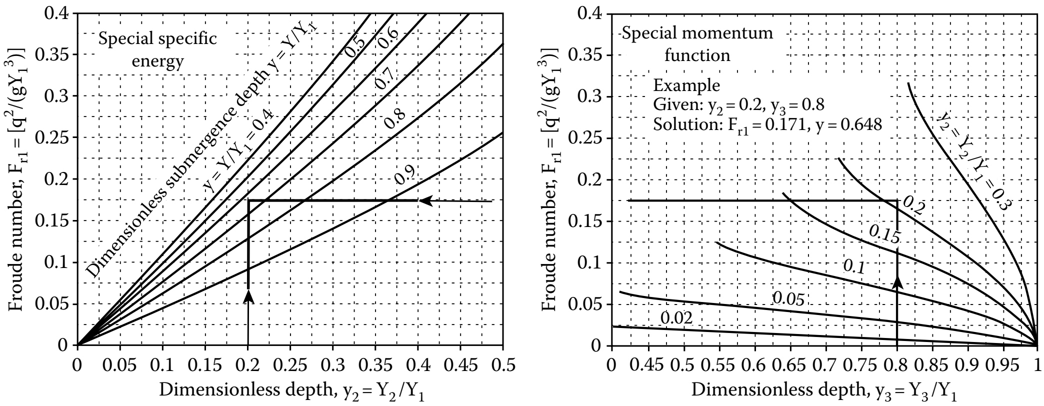


FIGURE 5.5 Special dimensionless-specific energy and momentum functions.

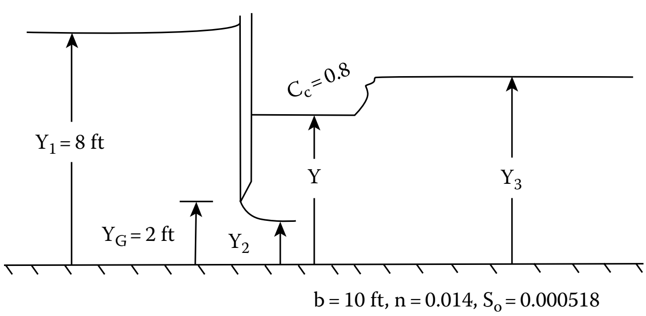
On the graph of the special dimensionless momentum function of Figure 5.5, the curves for the six values of y_2 terminate when the argument of the square root becomes zero, because small values of y_3 would result in attempting to take the square root of a negative value. The same is true for the curves on Figure 5.4. These limiting values define the minimum downstream depth required to maintain a submerged condition. With b defined by Equation 5.49 the limiting values of y_3 associated with any y_2 can be obtained by solving the equation

$$f(y_3) = b^2 + 4(y_3^2 - b) = 0$$

The solutions for several values of y_2 are given in the lower table insert in Figure 5.4.

EXAMPLE PROBLEM 5.6

The contraction coefficient for a gate in a rectangular channel is 0.8 and the gate is 2 ft above the channel bottom when the depth upstream from the gate is 8 ft. The gate is submerged. Downstream from the gate the channel is rectangular with a bottom width b = 10 ft, a Manning's roughness coefficient, n = 0.014, and a bottom slope S_o = 0.000518. Determine the flow rate and the submergence depth Y immediately downstream from the gate assuming the flow beneath the gate consists of jet action into the downstream channel.



Solution

The solution to this problem requires Equations 5.44, 5.46 and Manning's equation be solved simultaneously for Y, Y_3 = Y_o and q. This is done below using TK-Solver with the solution Y = 3.62 ft, Y_3 = 5.86 ft, and q = 27.4 cfs/ft. However by hand the following can be done: First obtain the dimensionless depth y_2 = 0.8(2)/8 = 0.2, but since y_3 depends upon the flow rate start with a guess such as q = 30 cfs/ft (F_{r1}^2 = .0546, F_{r1} = 0.2336), or Q = 300 cfs, and then solve

Manning's equation to give $Y_3 = 6.28$ ft, or $y_3 = 0.785$. Using Figure 5.5, or the table inserted in the upper part of Figure 5.4, gives, $y = 0.612$, and using Figure 5.5, or solving Equation 5.53 for the Froude number squared $F_{r1}^2 = .03233$ ($F_{r1} = 0.1798$) gives $q = 23.1$ cfs. Repeating the above process with different guesses for q eventually gives: $q = 27.4$ cfs/ft, $Y_3 = 5.86$ ft, and $y = 0.453$ with $Y = 3.62$ ft. It is interesting to ask the question: How much must the downstream depth be lowered for free flow to occur? Using the limiting values in the lower table in Figure 5.4, or setting the argument of the square-root in Equation 5.51 to zero, gives: $y_3 = 0.714116$ ($Y_3 = 5.713$ ft), $y = 0.299972$. Substituting y into Equation 5.53, (or y_3 into Equation 5.52) gives $F_{r1}^2 = .05833$ from which $q = 31.0$ cfs assuming that the upstream depth does not change, i.e., is 8 ft. On the other hand equating $E_1 = E_2$ across the gate with $Y_2 = 1.6$ ft and $Y_1 = 8$ ft gives $q = 33.15$ cfs/ft, i.e., there is a small discrepancy in the values of q obtained by these two methods. Solving Manning's equation to give S_o corresponding to $Q = 310$ cfs, and a depth of 5.713 ft gives $S_o = 0.000707$.

Let us now address the question: "When will submerged flow first occur and when will the flow be free flowing?" Submerged flow will not occur until at least the momentum function per unit width m_3 associated with the downstream depth Y_3 becomes equal to, or larger, than the momentum function m_2, which is based on the alternate depth to the upstream depth Y_1. As long as m_3 is smaller than m_2, the conjugate depth to Y_3, which will be denoted as, Y_{uj} for upstream of jump, will be larger than Y_2 and an M_3 GVF-profile will exist downstream from the gate. As depth Y_{uj} decreases toward Y_2, the length of this GVF-profile decreases, and when $Y_{uj} = Y_2$ it will have a zero length, i.e., the hydraulic jump will exist immediately downstream from the gate. Limiting values for the dimensional downstream depth y_3 and submergence depth y are shown in Figure 5.4. Let us add to these the limiting flow rate Q_{limit}, i.e., for flow rates smaller or equal to Q_{limit} the gate will be submerged for limiting $(Y_3)_{limit}$, and free flow conditions will occur when $Q > Q_{limit}$ and $Y_3 < (Y_3)_{limit}$. To determine this limiting flow rate assume the position of the hydraulic jump has just moved up to the position of the gate so $Y_{uj} = Y_2$ (or $y_{uj} = y_2$), and remember the limiting values for y_3 and y are defined so that the square root in Equation 5.51 is zero. Furthermore, let us use the upstream Froude number $F_{r1} = \sqrt{q^2/(gY_1^3)}$ as the dimensionless expression for the limiting flow rate and note that the Froude number squared associated with the downstream depth Y_3 is $F_{r3}^2 = F_{r1}^2/y_3^3$. Thus writing the hydraulic jump equation $Y_{uj}/Y_3 = Y_2/Y_3 = y_2/y_3 = \left[(1+8F_{r1}^2/y_3)^{1/2} - 1\right]/2$, and then solving for the upstream Froude number gives

$$(F_{r1})_{limit} = \left\{ (y_3)_{limit}^3 \frac{\left[\left((2y_2/(y_3)_{limit}) + 1\right)^2 - 1\right]}{8} \right\}^{1/2}$$

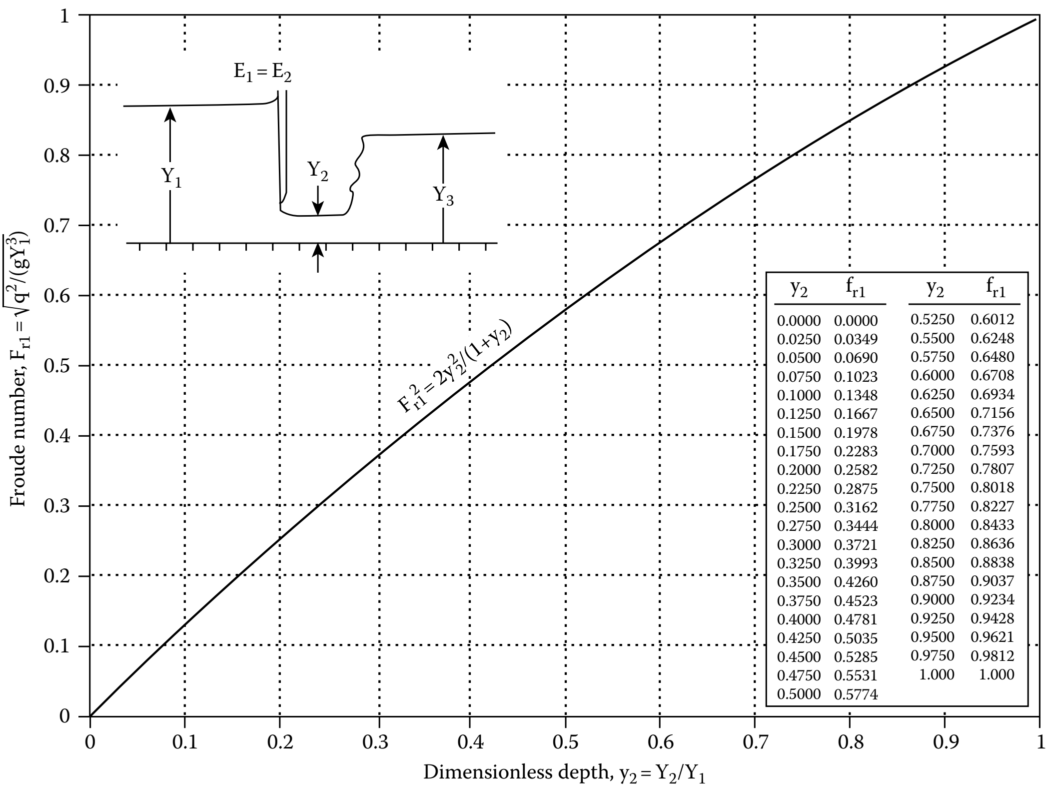
This limiting upstream Froude number $(F_{r1})_{limit}$ is plotted on Figure 5.6 against y_2 using the left ordinate. The right ordinate is for the limiting dimensionless downstream depth $(y_3)_{limit}$, the submergence depth y_{limit} and the dimensionless-specific energy (which is defined later) Δe_{limit}. If the actual upstream Froude number F_{r1} equals, or is less than, $(F_{r1})_{limit}$ then submergence will take place as soon as the jump arrives at the position of the gate, i.e., the down stream depth $y_3 = (y_3)_{limit}$.

EXAMPLE PROBLEM 5.7

What downstream depth Y_3 and flow rate Q will result in the gate in Example Problem 5.6 first becoming submerged? Assume the upstream depth remains constant at $Y_1 = 8$ ft.

Solution

Since $y_2 = 1.6/8 = 0.2$, the dimensionless downstream depth from Figure 5.7 is $y_3 = 0.714$, and $Y_3 = 0.714(8) = 5.71$ ft. The limiting Froude number $(F_{r1})_{limit} = 0.256$, so $q_{limit} = 0.256(gY_1^3)^{1/2} = 0.256(32.2 \times 512)^{1/2} = 32.81$ cfs/ft. If flow rates are larger than $Q_{limit} = 328.1$ cfs (rounding causes



y_2	f_{r1}	y_2	f_{r1}
0.0000	0.0000	0.5250	0.6012
0.0250	0.0349	0.5500	0.6248
0.0500	0.0690	0.5750	0.6480
0.0750	0.1023	0.6000	0.6708
0.1000	0.1348	0.6250	0.6934
0.1250	0.1667	0.6500	0.7156
0.1500	0.1978	0.6750	0.7376
0.1750	0.2283	0.7000	0.7593
0.2000	0.2582	0.7250	0.7807
0.2250	0.2875	0.7500	0.8018
0.2500	0.3162	0.7750	0.8227
0.2750	0.3444	0.8000	0.8433
0.3000	0.3721	0.8250	0.8636
0.3250	0.3993	0.8500	0.8838
0.3500	0.4260	0.8750	0.9037
0.3750	0.4523	0.9000	0.9234
0.4000	0.4781	0.9250	0.9428
0.4250	0.5035	0.9500	0.9621
0.4500	0.5285	0.9750	0.9812
0.4750	0.5531	1.000	1.000
0.5000	0.5774		

FIGURE 5.6 Solution of dimensionless-specific energy across a gate.

some of the difference between this and 310 cfs), then submergence will not occur until after the jump has arrived at the gate. For flow rates equal to, or less than, 328.1 cfs the gate will become submerged as soon as the jump arrives at the position of the gate, i.e., $Y_3 = 5.71$ ft.

The solution for Problem 6 gives a flow rate Q = 274 cfs, which is less than this limiting value. The solved for downstream depth of $Y_3 = 5.86$ ft produces a conjugate depth of $Y_{uj} = 5.86/2[1+8(27.4)^2/(32.2(5.86)^3]^{1/2} = 1.14$, which is less than $Y_2 = 1.6$ ft, i.e. so submerged conditions beyond limiting conditions are expected. Assume the flow rate is Q = 340 cfs, then the normal downstream depth is $Y_3 = 7.076$ ft, $y_3 = 0.885$, which is greater than $(y_3)_{limit} = 0.714$ ($(Y_3)_{limit} = 5.713$ ft). However, if the downstream channels slope were steeper ($S_o = 0.000850$) or n smaller (n = 0.01093) so the normal depth were 5.713 ft for this flow rate, then the conjugate depth would be 1.70 ft, and the gate would be free flowing with the hydraulic jump a short distance downstream therefrom.

According to the theory used, as soon as submergence occurs the limiting dimensionless submergence depth y shown in Figure 5.7 (and also given in Figure 5.4) will occur. If downstream conditions cause Y_3 to be larger than the limited value $(Y_3)_{limit}$, or cause Q to be less than Q_{limit} then the submergence depth increases above this limiting value, i.e., $Y > Y_{limit}$. Actually an increase in Y_3, when the gate is submerged causes a reduction in the flow rate because it creates greater energy losses at the gate.

EXAMPLE PROBLEM 5.8

Backwater effects downstream from the gate in the channel of Example Problem 5.6 cause the depth a short distance downstream from the gate to be $Y_3 = 6$ ft. Assume that the depth upstream remains at $Y_1 = 8$ ft, and determine the submergence depth Y and the flow rate Q.

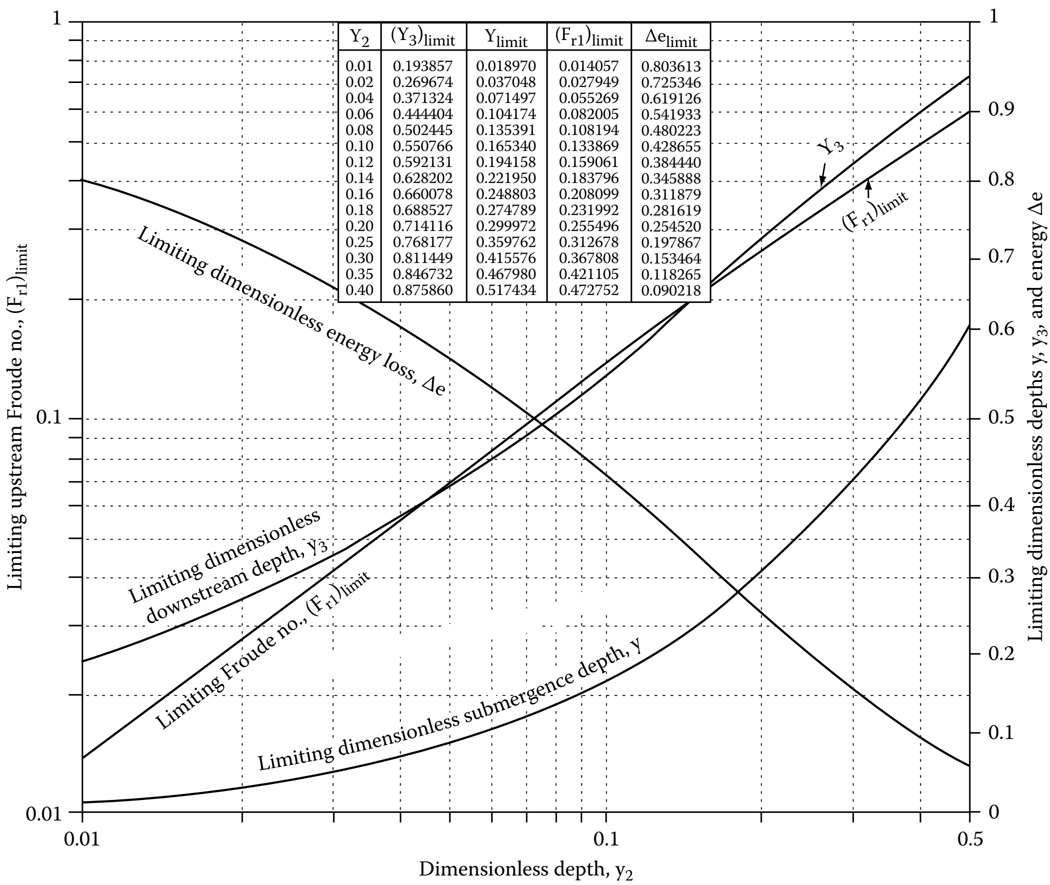


FIGURE 5.7 Limiting values for submergence to first occur.

Solution

Now submergence beyond the limiting values has occurred so Equation 5.51 (or Equation 5.47) can be used to solve for the submergence depth, or $b = 4y_2(1 - y_2/y_3)/(1 - y_2^2) = 4(.2)(1 - .2/.75)/(1 - .04) = 0.611$, $y = .5\left[b + \{b^2 + 4(y_3^2 - b)\}^{1/2}\right] = .5\left[.611 + \{.3735 + 4(-.0486)\}^{1/2}\right] = .517$, and the submergence depth is $Y = 0.517(8) = 4.137$ ft. The upstream Froude number can be obtained from Equations 5.52 or 5.53 and the flow rate therefrom (or Equations 5.44 or 5.46 can be used to solve q directly.) Using Equation 5.53, $F_{r1}^2 = 2y_2(1 - y)/(1 - y_2^2) = 2(.04)(1 - .517)/(.96) = .0402$ or $q = [.0402(32.2(8))^3]^{1/2} = 25.76$ cfs/ft and $Q = 257.6$ cfs. Note that the increase in downstream depth from the limiting value of 5.86–6 ft or 0.14 ft, has resulted in the submergence depth increasing from the limiting value of $Y_{limit} = 0.3(8) = 2.40$ to 4.137 ft or a 1.74 ft increase (this multifold increasing depth effect is apparent from the steepness of the curves on Figure 5.4 to as they intersect the bottom limiting line), and the flow rate has been reduced from the limiting value of $Q = 328.1$ cfs to 257.6 cfs, or a 70.5 cfs reduction. The energy loss now is $\Delta E = E_1 - E_3 = 8.161 - 6.286 = 1.875$ ft. Of course, this problem has been simplified because in an actual channel system the depth upstream from the gate will increase, depending upon where this water supply comes from, as the flow rate is decreased.

To summarize submerged flow will occur when the momentum function m_3 associated with the depth Y_3 becomes equal to (or larger than) the momentum function m_2, which is based on the alternate depth to the upstream depth Y_1. In other words, submerged conditions can be determined by going through the following computational steps: (1) if Y_1 and q are known solve for Y_2, of if Y_2

and q are known solve for Y_1, or if Y_1 and Y_2 are known solve for q from the energy equation $E_2 = E_1$, (2) compute m_2 from the known Y_2, or other variable obtained in step 1, (3) compute m_3 from the known Y_3. If $m_3 \geq m_2$, then the downstream depth will submerge the gate.

Adding additional mathematics to these steps gives, first, from $E_2 = E_1$:

$$Y_2 + \frac{q^2}{2gY_2^2} = Y_1 + \frac{q^2}{2gY_1^2} \quad \text{or} \quad \frac{q^2}{2g}\left(\frac{Y_1^2 - Y_2^2}{Y_1^2 Y_2^2}\right) = Y_1 - Y_2 \tag{5.54}$$

Dividing by Y_1 and using the dimensionless depth, $y_2 = Y_2/Y_1$ gives (*Note*: The root $y_2 = Y_2/Y_1 = 1$ is eliminated)

$$1 - y_2 = \frac{q^2}{2gY_1^3}\left(\frac{1}{y_2^2} - 1\right) = \frac{F_{r1}^2}{2}\left(\frac{1}{y_2^2} - 1\right) \tag{5.55}$$

or solving for the upstream Froude number squared gives

$$F_{r1}^2 = \frac{2y_2^2}{1 + y_2} \quad \text{or} \quad 2y_2^2 - F_{r1}^2 y_2 - F_{r1}^2 = 0 \tag{5.56}$$

Next the momentum function immediately downstream from the gate is

$$m_2 = \frac{Y_2^2}{2} + \frac{q^2}{gY_2} \tag{5.57}$$

and dividing by Y_1^2 and defining $m_2' = m_2/Y_1^2$ results in

$$m_2' = \frac{y_2^2}{2} + \frac{q^2}{gy_1^3 y_2} = \frac{y_2^2}{2} + \frac{F_{r1}^2}{y_2} = \frac{y_2^2}{2} + \frac{2y_2}{1 + y_2} \tag{5.58}$$

where $y_2 = Y_2/Y_1$. Likewise the momentum function downstream from a jump, if it occurs, is

$$m_3 = \frac{Y_3^2}{2} + \frac{q^2}{gY_3} \tag{5.59}$$

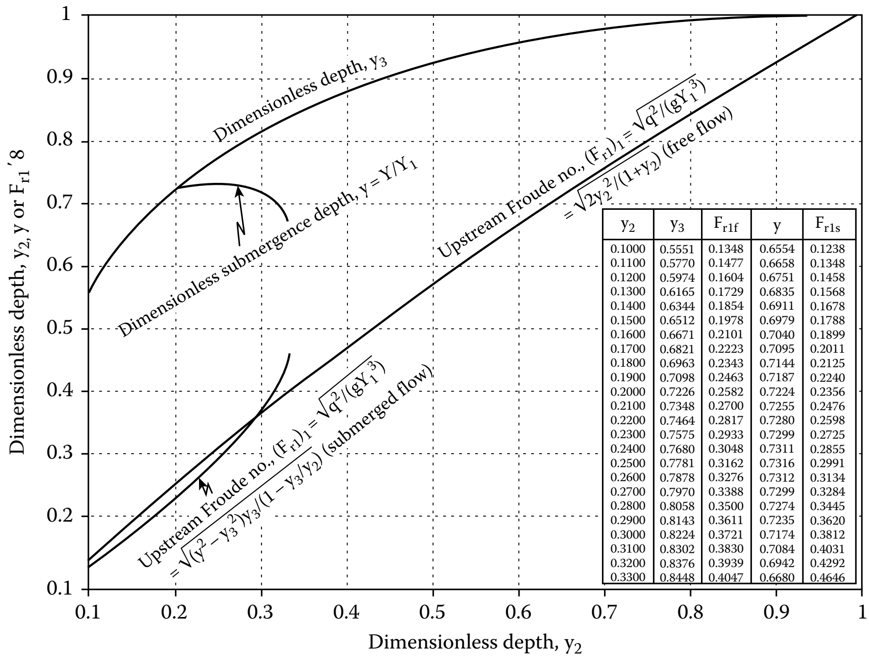
and dividing by Y_1^2 and defining $m_3' = m_3/Y_1^2$ results in

$$m_3' = \frac{y_3^2}{2} + \frac{q^2}{gy_1^3 y_3} = \frac{y_3^2}{2} + \frac{F_{r1}^2}{y_3} = \frac{y_3^2}{2} + \frac{2y_2^2}{y_3(1 + y_2)} \tag{5.60}$$

where $y_3 = Y_3/Y_1$. Since if $m_3' < m_2'$ free flow will occur, equate m_2' to m_3' to give the following implicit equation:

$$f(y_3) = \frac{y_3^2 - y_2^2}{2} + \frac{2y_2}{1 + y_2}\left(\frac{y_2}{y_3} - 1\right) = 0 \tag{5.61}$$

With any gate position Y_G and its contraction coefficient C_c known, the downstream dimensionless depth can be solved from $y_2 = Y_2/Y_1 = C_c Y_G/Y_1$, and then Equation 5.61 can be solved for y_3. The curve labeled "Dimensionless depth, y_3" on Figure 5.8 is a solution of Equation 5.61, and the inserted table provides data for this solution for y_2 from 0.1 to 0.33. The curve labeled



y_2	y_3	F_{r1f}	y	F_{r1s}
0.1000	0.5551	0.1348	0.6554	0.1238
0.1100	0.5770	0.1477	0.6658	0.1348
0.1200	0.5974	0.1604	0.6751	0.1458
0.1300	0.6165	0.1729	0.6835	0.1568
0.1400	0.6344	0.1854	0.6911	0.1678
0.1500	0.6512	0.1978	0.6979	0.1788
0.1600	0.6671	0.2101	0.7040	0.1899
0.1700	0.6821	0.2223	0.7095	0.2011
0.1800	0.6963	0.2343	0.7144	0.2125
0.1900	0.7098	0.2463	0.7187	0.2240
0.2000	0.7226	0.2582	0.7224	0.2356
0.2100	0.7348	0.2700	0.7255	0.2476
0.2200	0.7464	0.2817	0.7280	0.2598
0.2300	0.7575	0.2933	0.7299	0.2725
0.2400	0.7680	0.3048	0.7311	0.2855
0.2500	0.7781	0.3162	0.7316	0.2991
0.2600	0.7878	0.3276	0.7312	0.3134
0.2700	0.7970	0.3388	0.7299	0.3284
0.2800	0.8058	0.3500	0.7274	0.3445
0.2900	0.8143	0.3611	0.7235	0.3620
0.3000	0.8224	0.3721	0.7174	0.3812
0.3100	0.8302	0.3830	0.7084	0.4031
0.3200	0.8376	0.3939	0.6942	0.4292
0.3300	0.8448	0.4047	0.6680	0.4646

FIGURE 5.8 Solution of dimensionless parameter equations that separate submerged and free flow past a gate. (Submerged flow will occur for $Y_3 \geq Y_1 y_3$.)

"Upstream Froude No." (free flow) is a plot of Equation 5.56 on this same graph. This latter curve and the third column solve the energy $E_1 = E_2$, so if the flow rate and upstream depth are known, y_2 can be determined. With dimensionless depth y_2 known, the dimensionless depth y_3 can be determined that will result in a jump a very short distance downstream from the gate. Thus the graph, Figure 5.8, eliminates the need to solve Equations 5.56 and 5.61. If the actual downstream depth Y_3 is equal or greater than $y_3 Y_1$, then submerged flow occurs, otherwise free flow occurs.

Once the gate is submerged then the above analysis is no longer valid, since it is based on $E_1 = E_2$ across the gate. In other words, Equations 5.56 and 5.61 are valid only if $Y_3 < y_3 Y_1$. Free flow will occur if $Y_3 < y_3 Y_1$, where y_3 is the solution to Equation 5.61. These steps are illustrated by the following example problems.

EXAMPLE PROBLEM 5.9

A sluice (or vertical) gate in a rectangular channel has a contraction coefficient $C_c = 0.6$ and is set at a $Y_G = 1.7$ ft above the channel bottom. The depth on the upstream side of the gate is 5 ft. What is the maximum depth that can be allowed downstream from the gate without causing submerged flow?

Solution

Dimensionless depth $y_2 = 0.6(1.7)/5 = 0.204$. Substituting y_2 in Equation 5.56, or using Figure 5.8, gives, $F_{r1}^2 = q^2/(gY_1^3) = .06913 (F_{r1} = .263)$, from which q = 16.68 cfs/ft. Next solve Equation 5.61 for (or use Figure 5.8 to get) $y_3 = 0.728$ or $Y_3 = 0.728(5) = 3.64$ ft. Downstream depths less than 3.61 ft will allow flow past the gate to be free flowing. Note from Figure 5.8 that y is essentially equal to y_3, so that should submerged flow occur, then there should be little difference in the submergence depth Y immediately downstream from the gate and the downstream channel depth Y_3. However, for this submerged flow the upstream Froude number, $F_{r1} = 0.240$, gives q = 15.25 cfs, which is less than if the gate is free flowing. (Notice that as a bonus, with a minor amount of additional computations, the flow rate per unit width is obtained.)

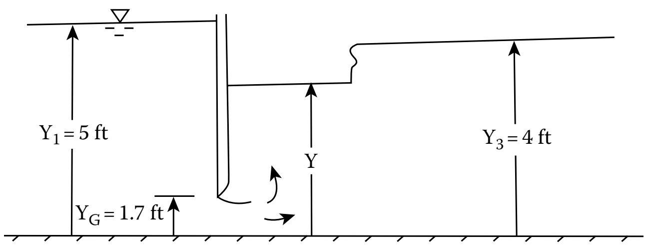
Assume the gate is raised to $Y_G = 2.5$ ft and Y_1 remains constant at 5 ft. Then $y_2 = 0.3$, $y_3 = 0.8224$ ($Y_3 = 4.11$ ft), and q = 23.61 cfs/ft, but the corresponding y for submerged flow is y = 0.7174 (Y = 3.59 ft), or Y is 0.52 ft less than Y_3. If submerged flow occurs, then the upstream Froude number = 0.3812, which is slightly larger than if the gate is free flowing.

EXAMPLE PROBLEM 5.10

Determine the depth immediately downstream from the gate, and the flow rate past the gate if in the previous problem the downstream depth is $Y_3 = 4$ ft ($Y_G = 1.7$ ft). Assume the upstream depth does not change, i.e., is still $Y_1 = 5$ ft.

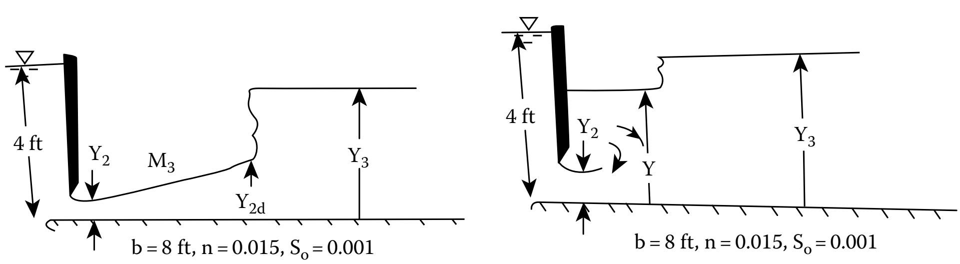
Solution

First compute $y_3 = 4/5 = 0.8$ and substitute this value and the value $y_2 = 0.204$ into Equation 5.49 to get b = 0.6343. Next substitute this b into Equation 5.51 to get y = 0.643, or Y = 0.643(5) = 3.22 ft. Now, however, the flow rate is given by Equation 5.53 (or Equation 5.52), or is given by, $q^2 = gY_1^3\{2(1-y)/(1/y_2^2-1)\} = 32.2(125)\{2(.357)/(1/.204-1)\} = 736.5$, or q = 27.14 cfs/ft, or if the upstream depth on the gate does not change then the submerging of the gate caused the flow rate per unit width of channel to reduce from 36.93 to 27.14 cfs/ft. Depending upon how this upstream channel gets its water, etc. this depth Y_1 will vary. The actual steady state solution would involve using the methods of solving algebraic and ODEs simultaneously described in Chapter 4. The difference is that logic is needed to determine whether free flow, or submerged flow, exists by the gate depending upon the downstream depth (or condition). When submerged flow occurs then algebraic equation (Equations 5.43 or 5.44) and Equation 5.45 (or Equation 5.46) replace $E_1 = E_2$, and the additional variable Y (depth immediately downstream from the gate) is added.



EXAMPLE PROBLEM 5.11

A gate controls the flow into an 8 ft wide rectangular channel that has a bottom slope of S_o = 0.001, and a Manning's n = 0.015. The gate is at the reservoir entrance, which keeps the depth of water at the upstream side of the gate constant at 4 ft. Consider steady-state flows into this channel with the gate setting changed so that the depth immediately downstream from the gate varies so that the ratio of this depth to the upstream depth $y_2 = Y_2/Y_1$ starts at 0.1 and increases to 0.30 in increments of 0.02 and then the gate setting is lowered to repeat these depth ratio, and identify at what gate heights it becomes submerged, or free flowing, both when increasing its height above the bottom and decreasing its heights. In other words, make a table that gives the flow rate Q, the channel depth upstream and downstream from a jump and/or the submergence depth Y as a function of the gate setting.



Free flowing conditions

Submerged conditions

Solution

Assume that at the smaller depth ratios that free flow occurs in which the depth Y_2 immediately downstream from the gate is less than the conjugate depth Y_{2d} to the normal depth in the downstream channel. Under this assumption, the first lines in table given below, which gives the solution, are obtained as follows: (1) the third column giving the Froude numbers for free flow are copied from the insert table in Figure 5.8 corresponding to the dimensionless depths in column 1, (2) Since $F_{rf} = q/(gY_1^3)^{1/2} = q/45.396$, the fourth column for q is obtained by multiplying the third column by 45.396, (3) the fifth column, giving the flow rate Q, is obtained by multiplying the fourth column by the 8 ft width, (4) Column 6 gives the normal depths in the channel and is obtained by solving Manning's equation using the flow rates in column 5, (5) Column 7 that gives the conjugate depth upstream from the hydraulic jump, Y_{2d}, is obtained by solving the hydraulic jump equation ($m_2 = m_3$) using the depth in column 6 as the downstream depth.

These computations continue until the depth in column 7 is no longer greater than the depth Y_2 in column 2 (which is four times the value in column 1), or as long as $Y_{2d} > Y_2 = Y_1 y_2$ the flow is free flowing. When Y_{2d} becomes equal to Y_2, then the hydraulic jump has moved up to the position of the gate and any increases in the gate's height will result in submergence. This condition occurs when $y_2 = 0.25$ and $Y_2 = 1.0$ ft, which is greater than $Y_{2d} = 0.986$ ft in the solution table below.

Solution to Problem 5.11: (b = 8 ft, n = 0.015, $S_o = 0.001$, $H = Y_1 = 4$ ft)

y_2 (1)	Y_2 (ft) (2)	F_{r1} (3)	q (cfs/ft) (4)	Q(cfs) (5)	Y_3 (ft) (6)	Y_{2d} (ft) (7)	y (8)	Y (ft) (9)	Jump x (ft)
Rising gate									
0.100	0.400	0.1348	6.121	48.97	1.725	0.584			34.6
0.120	0.480	0.1604	7.280	58.24	1.943	0.653			33.1
0.140	0.560	0.1854	8.418	67.34	2.149	0.715			
0.160	0.640	0.2101	9.537	76.30	2.346	0.772			26.0
0.180	0.720	0.2343	10.638	85.10	2.534	0.826			20.9
0.200	0.800	0.2582	11.721	93.77	2.715	0.875			14.9
0.220	0.880	0.2817	12.787	102.30	2.891	0.922			8.3
0.240	0.960	0.3048	13.837	110.69	3.060	0.965			1.0
0.250	1.000	0.3162	14.355	114.84	3.143	0.986	Submergence begins		0.0
0.250	1.000	0.2936	13.328	106.62	2.978		0.311	1.242	
0.260	1.040	0.2973	13.495	107.96	3.005		0.346	1.385	
0.280	1.120	0.3054	13.862	110.89	3.064		0.405	1.621	
0.300	1.200	0.3134	14.229	113.83	3.123		0.454	1.817	
Lowering gate							0.454	1.817	
0.300	1.200	0.3134	14.229	113.83	3.123		0.454	1.621	
0.280	1.120	0.3054	13.862	110.89	3.064		0.346	1.385	
0.260	1.040	0.2973	13.495	107.96	3.005		0.311	1.242	
0.250	1.000	0.2936	13.328	106.62	2.978		0.266	1.065	
0.240	0.960	0.2907	13.198	105.59	2.957		Submergence ends		
0.220	0.880	0.2834	12.864	102.91	2.904	0.695			
0.220	0.880	0.2817	12.787	102.30	2.891	0.922			8.3
0.200	0.800	0.2582	11.721	93.77	2.715	0.875			14.9
0.180	0.720	0.2343	10.638	85.10	2.534	0.826			20.9
0.160	0.640	0.2101	9.537	76.30	2.346	0.772			26.0
0.140	0.560	0.1854	8.418	67.34	2.149	0.715			30.2
0.120	0.480	0.1604	7.280	58.24	1.943	0.653			33.1
0.100	0.400	0.1348	6.121	48.97	1.725	0.584			34.6

Free flow

1. Solve $E_1 = E_2$ for $q(F_{r1}^2 = q^2/(gY_1^3) = 2y_2^2(1 - y_2)/(1 - y_2^2) = 2y_2^2/(1 + y_2)$
2. Solve Manning's equation for Y_3
3. Solve momentum equation $m_{2d} = m_3$ for Y_{2d}

Submerged flow (unknowns: Q, Y, and Y_3)
Equations:

1. Manning's equation downstream $F_1 = nQ - CA_3(A_3/P_3)^{2/3}S_o^{1/2} = 0$
2. Equation 5.44 (energy) $F_2 = Y_1 - Y - (Q/b)^2/(2g)(Y_1^2 - Y_2^2)/(Y_1Y_2)^2 = 0$
3. Equation 5.46 (momentum) $F_3 = 2(Q/b)^2/g - Y_2Y_3(Y_3^2 - Y^2)/(Y_3 - Y_2) = 0$

Alternative Equation 5.2 (Equation 5.47)

1. Equation 5.46 (The dimensionless forms of these equations might also be used.)
   The first line in the above table with $y_2 = 0.25$ in column 1 represents what the conditions are immediately after submergence has taken place. However, with the water submerging the downstream side of the gate, the flow rate will be reduced thus causing an unsteady flow in which the flow rate reduces and the downstream depth Y_3 becomes less. After steady-state flow has again established itself for this gate setting the second entry for $y_2 = 0.25$ applies. This second entry for $y_2 = 0.25$ is obtained by simultaneously solving the equations that apply for submerged flow, i.e., Equations 5.52 and 5.50 (or the dimensional form of these equations, Equations 5.43 and 5.45) are solved with Manning's equation for q, y, and y_3. Note that when submergence occurs the flow rate is reduced from 114.83 to 90.64 cfs. For all entries in the above table that are for submerged conditions Equations 5.52, 5.50 and Manning's equation ($8q = 1.486/nA_3^{1.6666667}/P^{.6666667}S_o^{1/2}$) are solved simultaneously for q, y, and y_3. In solving three equations simultaneously, it is necessary to caution be exercised that an incorrect root is not accepted. The depth Y downstream from the gate must be subcritical.

When the gate is lowered (i.e., the lower portion of the above table), it should be noted that the gate remains submerged until the depth ($Y_2 = 4y_2$) > Y_{2d} (the conjugate depth to the downstream depth Y_3). Free flow begins again as the gate is lower to produce $y_2 = 0.20$. Note that submergence remains for lower setting of the gate when it is being closed than when it is being opened.

When submerged flow takes place by a gate, the gate acts as a loss device in the channel. Of interest is the amount of this loss, or the difference between the upstream-specific energy E_1 and that in the flow downstream from the gate, E_3. This loss can be determined from the special specific energy (Equation 5.43) and momentum function Equation 5.45. Let e_1 be the dimensionless-specific energy upstream from the gate, which is made nondimensional by dividing by the upstream depth Y_1, then

$$e_1 = \frac{E_1}{Y_1} = \frac{Y_1 + q^2/\left(2gY_1^2\right)}{Y_1} = 1 + \frac{F_{r1}^2}{2} \tag{5.62}$$

and the downstream dimensionless depth e_3 is defined by

$$e_3 = \frac{E_3}{Y_1} = \frac{Y_3 + q^2/\left(2gY_3^2\right)}{Y_1} = y_3 + \frac{F_{r1}^2}{2y_3^2} \tag{5.63}$$

The upstream Froude number squared divided by 2 can be obtained from Equation 5.44 or in dimensionless form from Equation 5.53, or

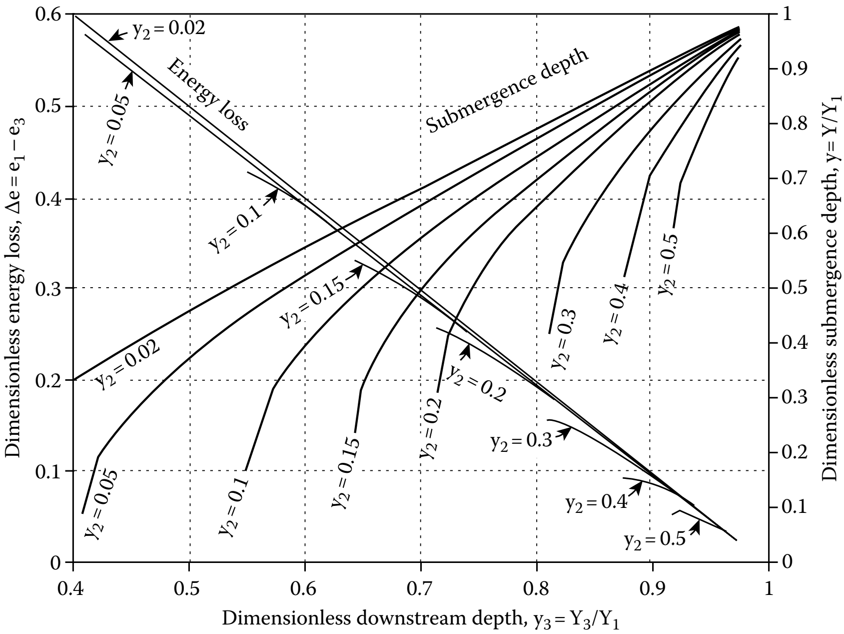


FIGURE 5.9 Dimensionless solution to loss of energy past submerged gate, $\Delta e = e_1 - e_3$, and submergence depth, $y = Y/Y_1$.

$$\frac{F_{r1}^2}{2} = \frac{y_2^2(1-y)}{1-y_2^2} \tag{5.64}$$

Thus the loss of energy in dimensionless form across the submerged gate becomes

$$\Delta e = e_1 - e_3 = 1 - y_3 + \left(1 - \frac{1}{y_3^2}\right)\frac{F_{r1}^2}{2} \tag{5.65}$$

Figure 5.9 show how this loss varies with the dimensionless downstream depth y_3 and the dimensionless depth beneath the gate y_2. Note from the plot that Δe decreases with increasing y_3 but is almost independent of y_2 and linear. Using the straight line diagonally across Figure 5.9 gives

$$\Delta e = 1 - y_3 \quad \text{(approximate)} \tag{5.65a}$$

Equation 5.65a can also be obtained by dropping the last term in Equation 5.65 that multiplies one-half the Froude number squared. Since the upstream Froude number is generally quite small, dropping this term allows a first approximation to the energy loss to be obtained from the very simple dimensionless equation (Equation 5.65a). The influence of this last term becomes significant as $1/y_3^2$ becomes large. However, the limiting depth y_3 between submerged and free flow conditions limits the magnitude of this term. Where the curves start on Figure 5.9 is at the limiting depth between free and submerged flow conditions. The effect of y_2 is that submergence occurs for smaller values of y_3 as y_2 decreases; or stated otherwise, limiting y_3 increases with y_2, i.e., as the gate is raised the downstream depth y_3 must be larger for submergence to occur. (See homework problem as an exercise in obtaining a table of values of Δe.)

It should be recognized that in using the dimensionless equations, whose variables have been made dimensionless by dividing by the upstream depth, Y_1, that a variable for the upstream condition

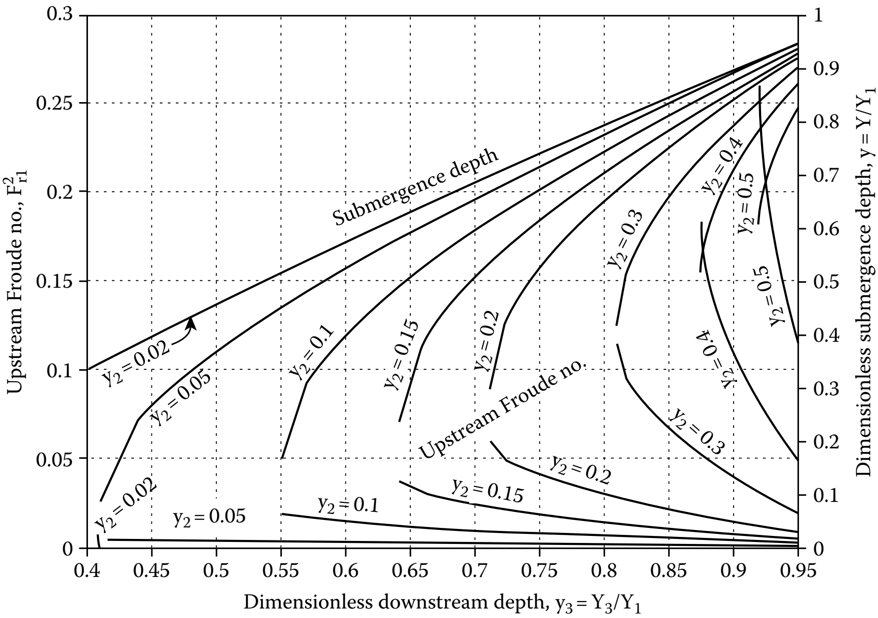


FIGURE 5.10 Variations of upstream Froude number squared with dimensionless depth y_2 and y_3.

has been eliminated, i.e., the variables, Y_1, Y_2, Y, and Y_3 have been reduced to three parameters, y_2, y, and y_3. Because of this, the dimensionless energy loss given by Equation 5.65 (or Figure 5.9) does not involve the upstream depth (or flow rate). However, the upstream conditions can be obtained that are associated with any y_2, y, and y_3 by solving for the upstream Froude number squared from Equation 5.52 (or Equation 5.53, which will give the same value). The upstream Froude number F_{r1} is a dimensionless representation of the upstream conditions. Figure 5.10 is a plot similar to Figure 5.9 except that the left ordinate gives the upstream Froude number, F_{r1}, rather than the dimensionless energy loss, Δe. Note that for the smaller values of the dimensionless depth y_2 that the upstream Froude number squared is small, and becomes even smaller as the downstream dimensionless depth y_3 becomes larger. As the depth y_2 get large, there is a very rapid change in F_{r1} with changes in y_3.

To acquire a still better understanding of the relationships between the dimensionless parameters F_{r1}^2, y_2, y, and y_3, it is useful to eliminate y between Equations 5.53 and 5.52 so as to produce a relationship of F_{r1}^2, y_2, and y_3 on a graph. This task can be accomplished by solving Equation 5.53 for y giving, $y = 1 - F_{r1}^2\left(1 - y_2^2\right)/\left(2y_2^2\right)$, and then replacing y in Equation 5.52 with this result. A plot of these results is shown on the first graph in Figure 5.11 in which the upstream Froude number squared is plotted on the abscissa, and the downstream dimensionless depth y_3 as the ordinate and the depth of flow from the gate y_2 as a series of curves. The lowest starting point for each such curve is the limiting value when the square root in Equation 5.51 becomes negative, or where real solutions to Equation 5.48 are first possible. The solid portion of each such curve stops when y equals y_2, and the dashed portions show the solution until y becomes zero. Thus the range for which submergence is possible is quite limited, i.e., for example a gate setting that produces $y_2 = 0.5$ restricts y_3 within the range of 0.920–0.931, and restricts F_{r1}^2 between 0.260 and 0.333. For a gate setting that produces $y_2 = 0.1$, as another example, y_3 can vary from 0.55 to 0.58, and F_{r1}^2 only between 0.0168 and 0.020. In brief, the first graph in Figure 5.11 reveals that very limited combinations of these variables are necessary for submerged flow to occur and this reinforces the result shown in Figure 5.8.

The second graph in Figure 5.11 shows a similar plot using the upstream Froude number squared as the abscissa but the dimensionless submergence depth y as the ordinate. The portion of this graph

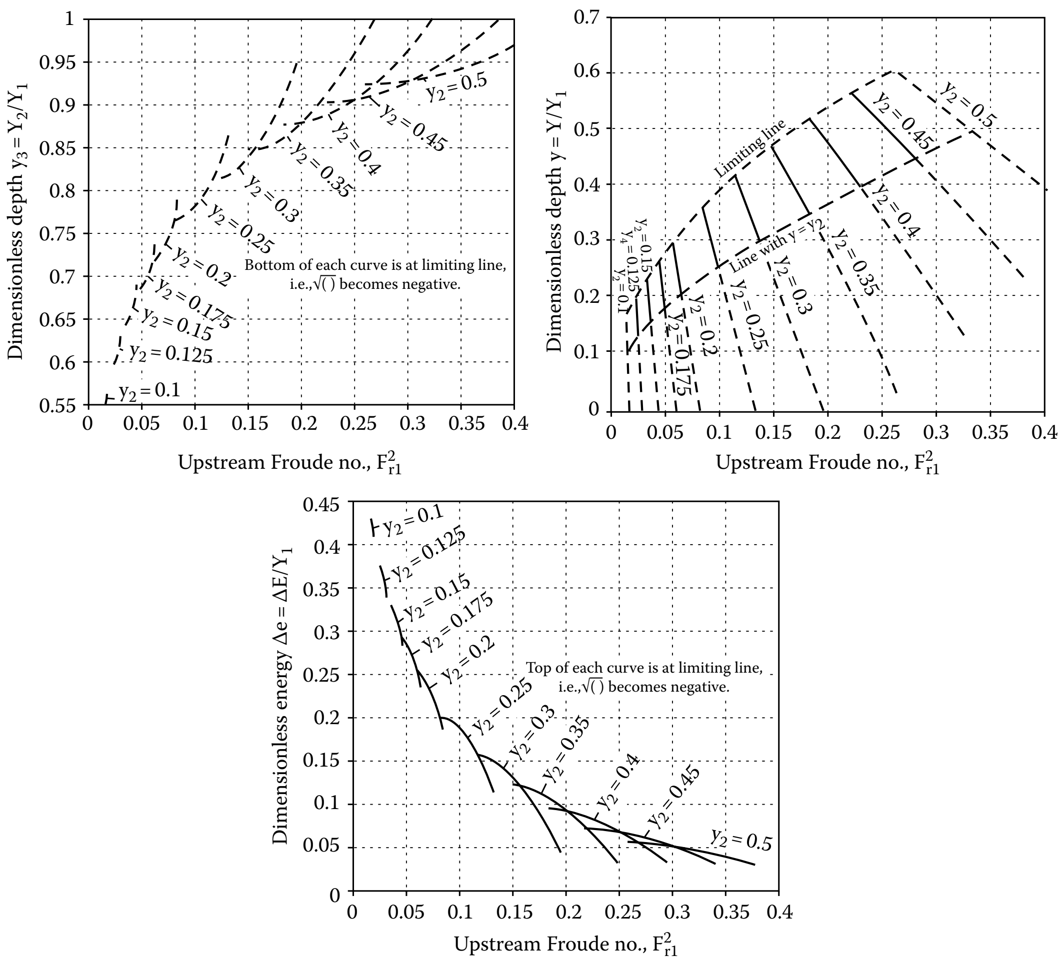


FIGURE 5.11 Dimensionless variables displayed as functions of the upstream Froude number squared.

between the limiting line and the line for which $y = y_2$ represents when submerged flow is possible. The third graph in Figure 5.11 gives the dimensionless loss in specific energy Δe as the ordinate.

EXAMPLE PROBLEM 5.12

A rectangular channel contains a sluice gate 800 m downstream from where it is supplied water by a constant level reservoirs whose depth is H = 2 m above the channel bottom. Upstream from the gate, the channel has a bottom width b_1 = 5 m and downstream from the gate the bottom width is b_2 = 4 m. Both the upstream and downstream portions of the channel have a Manning's n = 0.013, and a bottom slope S_o = 0.001 and the downstream channel is long. Generate a table of solutions starting with a dimensionless depth of flow below the gate tip of $y_2 = Y_2/Y_1 = 0.05$, varying this dimensionless depth until beyond when submergence occurs and back again.

Solution

The solution to the entries in the table will depend upon whether the flow behind the gate is free flowing, or whether the gate is submerged. For free flow conditions, the equations that need to be solve simultaneously for the unknowns, Q, Y_{beg} (depth immediately downstream from reservoir), Y_1 (depth upstream from gate), and Y_3 (normal depth downstream from hydraulic jump, or normal depth in downstream channel) are as follows:

$$F_1 = nQ - C_u A_3 \left(\frac{A_3}{P_3}\right)^{2/3} S_o^{1/2} = 0 \quad \text{(Manning's equation in downstream channel)}$$

$$F_2 = H - Y_{beg} - (1+K_e)\frac{(Q/A_{beg})^2}{(2g)} = 0 \quad \text{(Energy equation between reservoir and channel)}$$

$$F_3 = Y_1 + \frac{(Q/A_1)^2}{(2g)} - Y_2 - \frac{(Q/A_2)^2}{(2g)} = 0 \quad \text{(Energy across gate)}$$

$$F_4 = Y_{beg} - Y_{beg}(Y_1)_{ode} = 0 \quad \text{(GVF-solution upstream of gate to reservoir)}$$

After this solution has been obtained, the hydraulic jump equation is solved using the depth Y_3 downstream from the jump to get the depth Y_{2d} upstream from the jump. If this depth is greater than Y_2, the depth of flow under the tip of the gate, then the flow will be free flowing. However, as this depth Y_{2d} approaches Y_2, the hydraulic jump will move up to the position of the gate and begin to submerge the gate.

Upon being submerged, the additional submergence depth must be added to the list of unknowns that must be solved simultaneously, i.e., this depth replaces Y_{2d} as an unknown.

Solution Problem 5.12: (b_1 = 5m, b_2 = 4m, n = .013, S_o = .001, L = 800 m)

y_2	Y_2 m	Y_1 m	F_{r1}	q m²/s	Q m³/s	Y_{beg} m	Y_3 m	Y_{2d} m	Y (subm) m
0.05	0.139	2.783	0.0552	0.802	4.012	1.991	0.659	0.319	
0.08	0.207	2.762	0.0818	1.176	5.879	1.981	0.852	0.410	
0.10	0.274	2.735	0.1077	1.526	7.629	1.967	1.019	0.484	
0.125	0.338	2.702	0.1330	1.850	9.250	1.950	1.166	0.546	
0.15	0.400	2.664	0.1576	2.147	10.734	1.932	1.295	0.599	
0.175	0.459	2.622	0.1816	2.415	12.077	1.912	1.409	0.643	
0.20	0.515	2.576	0.2050	2.656	13.279	1.891	1.509	0.680	
0.25	0.620	2.478	0.2500	3.055	15.276	1.850	1.671	0.738	
0.30	0.713	2.376	0.2925	3.356	16.780	1.811	1.791	0.779	
0.35	0.796	2.275	0.3326	3.575	17.875	1.777	1.877	0.808	
0.36	0.812	2.256	0.3403	3.610	18.052	1.771	1.891	0.812	
0.361	0.814	2.254	0.3410	3.614	18.069	1.771	1.892	0.813	Submergence begins
0.361	0.814	2.255	0.3405	3.611	18.057	1.771	1.891		0.819
0.375	0.844	2.252	0.3417	3.617	18.085	1.770	1.893		0.923
0.40	0.896	2.240	0.3464	3.637	18.186	1.767	1.901		1.062
0.45	0.995	2.211	0.3576	3.683	18.416	1.759	1.919		1.262
0.50	1.092	2.185	0.3680	3.722	18.612	1.751	1.934		1.408
0.55	1.190	2.163	0.3767	3.753	18.764	1.746	1.946		1.524
0.60	1.287	2.145	0.3837	3.776	18.878	1.741	1.955		1.618
0.50	1.092	2.185	0.3680	3.722	18.612	1.751	1.934		1.408
0.40	0.896	2.240	0.3464	3.637	18.186	1.767	1.901		1.062
0.375	0.844	2.252	0.3417	3.617	18.085	1.770	1.893		0.923
0.36	0.812	2.255	0.3405	3.611	18.057	1.771	1.891		0.810
0.36	0.812	2.256	0.3403	3.610	18.052	1.771	1.891	0.812	Free flow begins again
0.35	0.796	2.275	0.3326	3.575	17.875	1.777	1.877	0.808	
0.30	0.713	2.376	0.2925	3.356	16.780	1.811	1.791	0.779	
0.20	0.515	2.576	0.2050	2.656	13.279	1.891	1.509	0.680	
0.10	0.274	2.735	0.1077	1.526	7.629	1.967	1.019	0.484	
0.05	0.139	2.783	0.0552	0.802	4.012	1.991	0.659	0.319	

The unknowns become Q, Y_{beg}, Y_1, Y_3, and Y. The equation that can be used to solve these five variables are as follows (or their dimensionless equivalences could be used):

$$F_1 = nQ - C_u A_3 \left(\frac{A_3}{P_3}\right)^{2/3} S_o^{1/2} = 0 \quad \text{(Manning's equation in downstream channel)}$$

$$F_2 = H - Y_{beg} - (1 + K_e)\frac{(Q/A_{beg})^2}{(2g)} = 0 \quad \text{(Energy equation between reservoir and channel)}$$

$$F_3 = Y_1 + \frac{(Q/A_1)^2}{(2g)} - Y - \frac{(Q/A_2)^2}{(2g)} = 0 \quad \text{(Energy across gate)}$$

$$F_4 = \frac{q_2^2}{g} - \frac{0.5\left(Y_3^2 - Y^2\right)Y_3 Y_2}{(Y_3 - Y_2)} = 0 \quad \text{(Momentum downstream from gate)}$$

$$F_5 = Y_{beg} - Y_{beg}(Y_1)_{ode} = 0 \quad \text{(GVF-solution upstream of gate to reservoir)}$$

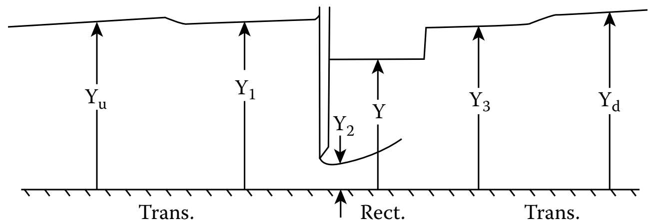
(A computer program that solves this problem is listed with the solution provided to Problem 5.87 at the end of this chapter.)

The solution to this problem is given in the table above.

5.8 SERIES OF SUBMERGED GATES

Channel systems frequently have a series of gates along them to control the depths and flow rates. When such a series of gates does exist, it is most likely that the flow past the gates will be submerged. Should the flow past a gate be free flowing, then this gate acts to control the flow both upstream and downstream from it, i.e., the problem is divided into two problems; the problem upstream from this gate and the one downstream therefrom. In this section, methods will be discussed that allow for the flow rate that will exist in a channel and the depths throughout the channel system to be solved assuming steady state conditions. The assumption will be that the flow by all gates is submerged so that the flow rate will be downstream controlled, and each gate in the system will act as a local loss device.

To formulate this problem of a series of submerged gate flows, we will assume that there is a short smooth transition both upstream and downstream from the gate that takes the channel from its upstream shape to a rectangular shape by the gate and then back to its downstream shape, and that the submerged flows are confined to the rectangular portion of the channel at the gates. The notation shown in the sketch below will be used. The depth in the channel upstream from the transition is denoted by Y_u; the depth downstream of the upstream transition immediately upstream from the gate is Y_1; the jet flow past the gate is Y_2, and its magnitude equals the gate height Y_G time a contraction coefficient C_c; the submergence depth is Y; the depth in a short distance downstream from the gate but still in the rectangular channel is Y_3; and the depth just beyond the downstream transition is Y_d. At each gate, there are four equations that can be written; two are regular energy equations across the upstream and downstream transitions, one is the special energy equation across the gate, and the fourth is the special momentum equation downstream from the gate. Not using possible dimensionless versions these four equations are as follows:



$$F_1 = Y_u + \frac{Q^2}{2gA_u^2} - Y_1 - \frac{q^2}{2gY_1^2} = 0 \quad \text{Energy across upstream transition} \tag{5.66}$$

$$F_2 = Y_1 + \frac{q^2}{2gA_1^2} - Y - \frac{q^2}{2gY_2^2} = 0 \quad \text{Special energy across gate} \tag{5.67}$$

$$F_3 = \frac{Y^2}{2} + \frac{q^2}{gY_2} - \frac{Y_3^2}{2} - \frac{q^2}{gY_3} = 0 \quad \text{Special Momentum downstream from gate} \tag{5.68}$$

$$F_4 = Y_3 \frac{q^2}{2gY_3^2} - Y_d - \frac{Q^2}{2gA_d^2} = 0 \quad \text{Energy across downstream transition} \tag{5.69}$$

At each such submerged gate, one might consider the downstream depth Y_d established by a downstream condition, i.e., another submerged gate, a downstream reservoir, a free overfall, etc. Thus at each gate there are four unknown variables that can be solved from the four equations: Y_3, Y, Y_1, and Y_u. For example, if there are four submerged gates in the system there will be 4 × 4 equations available, and these can be used to solve for the 4 × 4 variables as listed below. At the beginning of the channel system, there will be another unknown, the depth Y_{1r}, but the energy equation from the reservoir to the beginning of the channel provides an additional equation, or

$$F_o = H - Y_{1r} - \frac{(1 + K_e)Q^2}{2gA_{1r}^2} = 0 \tag{5.70}$$

This energy equation might be considered the equation associated with the unknown flow rate Q. There will be a gradually varied flow equation that uses as its starting depth Y_u from the next downstream gate, or if this is the last gate in the series, then the downstream boundary condition, and end at the depth Y_d for this gate, or if this is the first gate, the depth Y_{1r} at the beginning of the channel. This equation can be denoted as

$$F_5 = Y_{di} - Y_{GVF}(Y_{ui+1}) = 0 \tag{5.71}$$

Thus in general if N_{gate} exist there will be $5N_{gate} + 2$ equations.

We will consider three types of downstream boundary conditions: (1) The final channel is very long so that uniform flow occurs in it. For this condition, the finally GVF equation will be replaced by the uniform flow equation, e.g., Manning's equation, (2) A reservoir into which the last channel discharges. Since the velocity head will be dissipated, the height of the reservoir's water surface elevation above the channel bottom will establish the final depth Y_e and the final GVF equation will begin with this depth. (3) A free overfall occurs at the end of the channel. For this third boundary condition, and additional unknown will be introduced, the critical depth, $Y_c = Y_e$, and the critical flow equation in the downstream channel can be added to the list of available equations. For this third boundary condition, there will be $5N_{gate} + 3$ equations, and this many unknowns.

The solution to this combined system of algebraic and ordinary differential equations can be solved using the Newton method and with the techniques described in previous chapters. The FORTRAN and C programs SUBMESER are designed to obtain such solutions. These programs do **not** accommodate free flow at any gate. In using them it is therefore necessary that the conditions specified in the problem will cause a submerged flow past all of the gates in the channel. In these programs, the two-dimensional array V(5,10) contains the values of the variables being solved by each gate, e.g., V(1,I) = Y_d, V(2,I) = Y_3, V(3,I) = Y, V(4,I) = Y_1, and V(5,I) = Y_u. The submerged jet depth $Y_2 = C_cY_GI$, with the contraction coefficient COEF1 = $C_c = 0.59 + 0.0224Y_G/Y_1I$. In addition, the depth in the channel at the reservoir Y_{or}, which is the variable Y1 in the program (or y1 in the C program), and the flow rate Q (Q or q in the programs) constitute two additional unknowns. If the third boundary condition of critical flow is specified, then Y_e (YE or ye in the programs is an additional unknown. The subroutine (void function) FUNCT (funct) is used to generate the system of equations and store the values of these equations in the array F past as its argument. Since this subroutine will need to also evaluate the GVF equations, it calls on the ODE solver ODESOL, and this solver in turn calls on subroutine DYX to provide the ODE it is to solve. The main program implements the Newton method starting with the statement 20 SUM=0. (sum=0.) The input consists of the following on the first line: (1) NGATE (the number of submerged gate in the series of gates). The number of channels NC = NGATE + 1), (2) IBCE (the number of the downstream boundary condition); IBCE = 1 specifies uniform flow (i.e., a long channel), IBCE = 2 specifies a downstream reservoir, in which case Y_e (program variable YE) will be its depth above the bottom of the channel at its end; IBCE = 3 specifies critical flow at the channel's end, (3) G (the acceleration of gravity), (4) Q (an estimate, or starting value, for the flow rate), (5) H (the depth of the upstream reservoir above the channel bottom), (6) FKE (the upstream minor local loss coefficient), (7) Y1 (an estimate of the upstream depth, Y_{1r}), (8) YE (an estimate of the downstream depth if uniform flow occurs, i.e., IBCE = 1; the depth of the downstream reservoir above the channel if IBCE = 2; and an estimate of the critical depth if IBCE = 3), (9) TOL (the error criteria that ODESOL is to use in solving the ODE).

The next line of input consists of pairs of values (YG(I),BG(I),I=1,NGATE), consisting of the height of the gate and the channel's bottom width at each of the NGATE. The next group of input data provide the geometries of each of the channels starting with the upstream most channel. The first line contains the following five items for the channel upstream of the first transition to the first gate: the bottom width b; the side slope m of the trapezoidal section; the length of this channel; and its bottom slope S_o. The remaining lines of this input contain the same information for the remaining channels, the last of which will be for the channel downstream from the final gate.

To providing reasonable starting values for the Newton method, the last input requires estimates for the following three variables at each gate starting with the upstream most gate: Y_1, Y, and Y_3.

Program SUBMESER.FOR (solves flow through series of submerged gates)

```
      COMMON NGOOD,NBAD,KMAX,KCOUNT,DXSAVE
COMMON/TRAS/V(5,10),YG(10),BG(10),B(11),FM(11),
     &FN(11),FL(11),So(11),H,FKE,TOL,QN,HMIN
     &,H1,SOC,BC,FMC,FMS,FM2,Q2G,Y1,YE,G,
     &Q,Cu,NGATE,IBCE,NEQ,NC
      REAL FF(57),F(57),DJ(57,57)
      INTEGER*2 INDX(57)
      READ(2,*) NGATE,IBCE,G,Q,H,FKE,Y1,YE,TOL
      NC=NGATE+1
C For each gate V(1,I)=Yd, V(2,I)=Y3, V(3,I)=Y,
C V(4,I)=Y1,    V(5,I)=Yu
C IBCE=1 (Long downstream channel so YE is uniform
```

```
C depth & V(1,NGATE)=YE
C IBCE=2(Downstream reservoir with
C w.s. elevation=YE
C (specified))
C IBCE=3(Crit. depth at end of channel NC,YE=Critical depth)
      READ(2,*)(YG(I),BG(I),I=1,NGATE)
      READ(2,*)(B(I),FM(I),FN(I),FL(I),So(I),I=1,NC)
C Guesses for Q, and depth Y1,Y,Y3 for each gate
      DO 10 I=1,NGATE
      READ(2,*) V(4,I),V(3,I),V(2,I)
      V(5,I)=1.005*V(4,I)
10    V(1,I)=1.005*V(2,I)
      IF(G.LT.25.) THEN
      Cu=1
      ELSE
      Cu=1.486
      ENDIF
      G2=2.*G
      Q2G=Q*Q/G2
      FKE=1.+FKE
      H1=-.05
      HMIN=.0005
      NEQ=5*NGATE+2
      IF(IBCE.EQ.3) NEQ=NEQ+1
      NCT=0
      WT=.2
20    SUM=0.
      CALL FUNCT(F)
      DO 40 I=1,NGATE
      DO 40 J=1,5
      IJ=5*(I-1)+J
      XX=V(J,I)
      V(J,I)=1.005*XX
      CALL FUNCT(FF)
      DO 30 K=1,NEQ
30    DJ(K,IJ)=(FF(K)-F(K))/(V(J,I)-XX)
40    V(J,I)=XX
      IJ=IJ+1
      XX=Y1
      Y1=1.005*Y1
      CALL FUNCT(FF)
      DO 45 K=1,NEQ
45    DJ(K,IJ)=(FF(K)-F(K))/(Y1-XX)
      Y1=XX
      IJ=IJ+1
      XX=Q
      Q=1.005*Q
      Q2G=Q*Q/G2
      CALL FUNCT(FF)
```

```
      DO 50 K=1,NEQ
50    DJ(K,IJ)=(FF(K)-F(K))/(Q-XX)
      Q=XX
      Q2G=Q*Q/G2
      IF(IBCE.GT.2) THEN
      XX=YE
      YE=1.005*YE
      CALL FUNCT(FF)
      DO 55 K=1,NEQ
55    DJ(K,NEQ)=(FF(K)-F(K))/(YE-XX)
      YE=XX
      ENDIF
      CALL SOLVEQ(NEQ,57,DJ,F,1,DD,INDX)
      DO 60 I=1,NGATE
      DO 60 J=1,5
      IJ=5*(I-1)+J
      V(J,I)=V(J,I)-WT*F(IJ)
60    SUM=SUM+ABS(F(IJ))
      IJ=IJ+1
      Y1=Y1-WT*F(IJ)
      SUM=SUM+ABS(F(IJ))
      IJ=IJ+1
      Q=Q-WT*F(IJ)
      Q2G=Q*Q/G2
      SUM=SUM+ABS(F(IJ))
      IF(IBCE.EQ.3) THEN
      YE=YE-F(NEQ)
      SUM=SUM+ABS(F(NEQ))
      ENDIF
      NCT=NCT+1
      IF(SUM.LT.5.) WT=1.
      WRITE(*,200) NCT,SUM,((V(J,I),J=1,5),I=1,NGATE),Y1,Q,YE
200   FORMAT(' NCT=',I4,' SUM=',E12.3,/(10F8.3))
      IF(WT.LT.1.) THEN
      WRITE(*,*)' Current Weight factor=',WT,' Give new one'
      READ(*,*) WT
      ENDIF
      IF(NCT.LT.40 .AND. SUM.GT. 1.E-4) GO TO 20
      DO 70 I=1,NGATE
70    WRITE(3,100) I,(V(J,I),J=1,5)
      IF(IBCE.EQ.1) YE=V(1,NGATE)
      WRITE(3,101) Y1,Q,YE
100   FORMAT(I4,5F8.3)
101   FORMAT(' Y1=',F8.3,' Q=',F10.3,' Ye=',F8.3)
      END
      SUBROUTINE FUNCT(F)
      REAL F(57)
      COMMON NGOOD,NBAD,KMAX,KCOUNT,DXSAVE
      COMMON/TRAS/V(5,10),YG(10),BG(10),B(11),FM(11),FN(11),FL(11),
```

```
     &So(11),H,FKE,TOL,QN,HMIN,H1,SOC,BC,FMC,FMS,FM2,Q2G,Y1,YE,G,
     &Q,Cu,NGATE,IBCE,NEQ,NC
      REAL Y(1),DY(1),W(1,13),XP(1),YP(1,1)
      EXTERNAL DYX
      F(1)=H-Y1-FKE*Q2G/((B(1)+FM(1)*Y1)*Y1)**2
      DO 10 I=1,NGATE
      J=I+1
      IG=5*(I-1)+1
      BC=B(I)
      FMC=FM(I)
      FM2=2.*FM(I)
      FMS=2.*SQRT(FM(I)**2+1.)
      SOC=So(I)
      QS1=Q2G/BG(I)**2
      QMG=2.*QS1
      QN=(FN(I)*Q/Cu)**2
      Y2=(.59+.022*YG(I)/V(4,I))*YG(I)
      H11=V(4,I)+QS1/V(4,I)**2
      F(IG+1)=V(5,I)+Q2G/((BC+FMC*V(5,I))*V(5,I))**2-H11
      F(IG+2)=H11-V(3,I)-QS1/Y2**2
      F(IG+3)=.5*(V(3,I)**2-V(2,I)**2)+QMG/Y2-QMG/V(2,I)
      F(IG+4)=V(2,I)+QS1/V(2,I)**2-V(1,I)-Q2G/((B(J)+
     &FM(J)*V(1,I))*V(1,I))**2
      Y(1)=V(5,I)
      CALL ODESOL(Y,DY,1,FL(I),0.,TOL,H1,HMIN,1,XP,YP,W,DYX)
      IF(I.EQ.1) THEN
      F(IG+5)=Y1-Y(1)
      ELSE
      F(IG+5)=V(1,I-1)-Y(1)
      ENDIF
10    CONTINUE
      IF(IBCE.EQ.1) THEN
      IG=IG+6
      F(IG)=FN(NC)*Q*(B(NC)+2.*V(1,NGATE)*SQRT(FM(NC)**
     &2+1.))**.66666667-Cu*((B(NC)+FM(NC)*V(1,NGATE))*
     &V(1,NGATE))**1.666667*SQRT(So(NC))
      ELSE
      Y(1)=YE
      IF(IBCE.EQ.3) F(NEQ-1)=.5*Q2G*(B(NC)+2.*FM(NC)*YE)-
     &G*((B(NC)+FM(NC)*YE)*YE)**3
      BC=B(NC)
      FMC=FM(NC)
      FM2=2.*FM(NC)
      FMS=2.*SQRT(FM(NC)**2+1.)
      SOC=So(NC)
      QN=(FN(NC)*Q/Cu)**2
      CALL ODESOL(Y,DY,1,FL(NC),0.,TOL,H1,HMIN,1,XP,YP,W,DYX)
      F(NEQ)=V(1,NGATE)-Y(1)
      ENDIF
      RETURN
      END
```

```
      SUBROUTINE DYX(X,Y,DY)
      REAL Y(1),DY(1)
      COMMON/TRAS/V(5,10),YG(10),BG(10),B(11),FM(11),FN(11),FL(11),
     &So(11),H,FKE,TOL,QN,HMIN,H1,SOC,BC,FMC,FMS,FM2,Q2G,Y1,YE,G,
     &Q,Cu,NGATE,IBCE,NEQ,NC
      P=BC+FMS*Y(1)
      A=(BC+FMC*Y(1))*Y(1)
      SF=QN*((P/A)**.6666667/A)**2
      DY(1)=(SOC-SF)/(1.-2.*Q2G*(BC+FM2*Y(1))/A**3)
      RETURN
      END
```

Program SUBMESER.C (Solves flow by series of submerged gates)

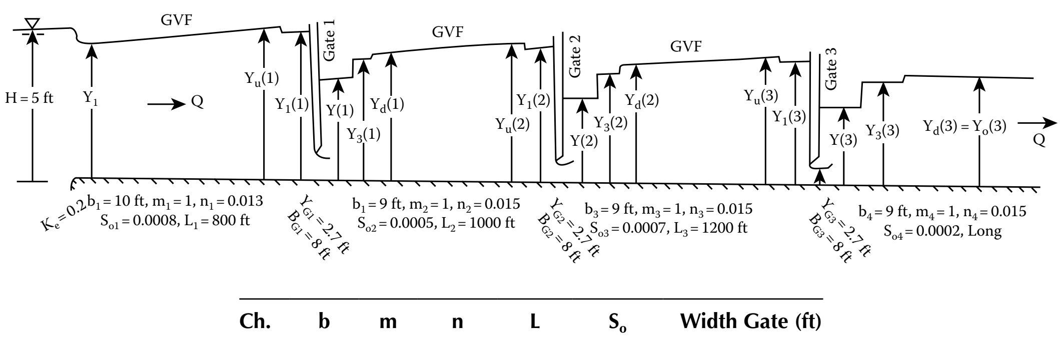
```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "odesolc.h"
float v[5][10],yg[10],bg[10],b[11],fm[11],fn[11],fl[11],so[11],h,\
  fke,tol,qn;
float soc,bc,fmc,fms,fm2,q2g,y1,ye,g,g2,q,cu;
int ngate,ibce,neq,nc;
extern void solveq(int n,float **a,float *b,int itype,float dd,\
  int *indx);
 char fnam[20];
 FILE *fili,*filo;
 void slope(float x,float *y,float *dy){float p,a,sf;
 p=bc+fms*y[0]; a=(bc+fmc*y[0])*y[0];
   sf=qn*pow(pow(p/a,.6666667)/a,2.);
 dy[0]=(soc-sf)/(1.-2.*q2g*(bc+fm2*y[0])/pow(a,3.));
} // End of slope
void funct(float *fe) {int i,j,ig;float h11,qs1,qmg,y2,\
  hmin=.005,h1=-.1,y[1];
 fe[0]=h-y1-fke*q2g/pow((b[0]+fm[0]*y1)*y1,2.);
 for(i=0;i<ngate;i++){j=i+1; ig=5*i;
 bc=b[i]; fmc=fm[i]; fm2=2.*fm[i]; fms=2.*sqrt(fm[i]*fm[i]+1.);
   soc=so[i];
 qs1=q2g/(bg[i]*bg[i]); qmg=2.*qs1; qn=pow(fn[i]*q/cu,2.);
 y2=(.59+.022*yg[i]/v[3][i])*yg[i]; h11=v[3][i]+qs1/(v[3][i]*v[3][i]);
 fe[ig+1]=v[4][i]+q2g/pow((bc+fmc*v[4][i])*v[4][i],2.)-h11;
 fe[ig+2]=h11-v[2][i]-qs1/(y2*y2);
 fe[ig+3]=.5*(v[2][i]*v[2][i]-v[1][i]*v[1][i])+qmg/y2-qmg/v[1][i];
 fe[ig+4]=v[1][i]+qs1/(v[1][i]*v[1][i])-v[0][i]-q2g/pow\
   ((b[j]+fm[j]*v[0][i])*v[0][i],2.);
 y[0]=v[4][i];
 odesolc(y,fl[i],0.,tol,h1,hmin,1);
 if(i==0) fe[ig+5]=y1-y[0]; else fe[ig+5]=v[0][i-1]-y[0];}
 if(ibce==1) {ig+=6;
 fe[ig]=fn[ngate]*q*pow(b[ngate]+2.*v[0][ngate-1]*sqrt(fm[ngate]*\
   fm[ngate]+1.),.66666667)
 -cu*pow((b[ngate]+fm[ngate]*v[0][ngate-1])*v[0][ngate-1],\
   1.666667)*sqrt(so[ngate]);}
```

```
 else {y[0]=ye;if(ibce==3)
 fe[neq-2]=.5*q2g*(b[ngate]+2.*fm[ngate]*ye)-\
   g*pow((b[ngate]+fm[ngate]*ye)*ye,3.);
 bc=b[ngate]; fmc=fm[ngate]; fm2=2.*fm[ngate];
   fms=2.*sqrt(fm[ngate]*fm[ngate]+1.);
 soc=so[ngate]; qn=pow(fn[ngate]*q/cu,2.);
 odesolc(y,fl[ngate],0.,tol,h1,hmin,1);
 fe[neq-1]=v[0][ngate-1]-y[0];}
} // End of funct
void main(void){
 float ff[57],f[57],**dj;
 int indx[57],i,j,k,ij,nct;
 float xx,sum,wt,dd;
 printf("Give name of file that contains input datan");
   scanf("%s",fnam);
 if((fili=fopen(fnam,"r"))==NULL){printf("File does not exist.n");
   exit(0);}
 printf("Give name of output filen"); scanf("%s",fnam);
 if((filo=fopen(fnam,"w"))==NULL){printf("File cannot be openedn");
   exit(0);}
 fscanf(fili,"%d %d %f %f %f %f %f %f %f",&ngate,&ibce,&g,&q,&h,\
   &fke,&y1,&ye,&tol);
 nc=ngate+1;
// Each gate v(0,i)=yd, v(1,i)=y3, v(2,i)=y, v(3,i)=y1, v(4,i)=yu;
//ibce=1 (long downstream channel so ye is uniform depth & \
  v(0,ngate)=ye;
//ibce=2 (downstream reservoir with w.s. elevation = ye \
  (specified));
//ibce=3 (critical depth at end of channel nc, ye = critical depth);
 for(i=0;i<ngate;i++) fscanf(fili,"%f %f",&yg[i],&bg[i]);
 for(i=0;i<nc;i++) fscanf(fili,"%f %f %f %f %f",&b[i],&fm[i],\
   &fn[i],&fl[i],&so[i]);
// Guesses for depth y1,y,y3 for each gate;
 for(i=0;i<ngate;i++){fscanf(fili,"%f %f %f",&v[3][i],&v[2][i],\
   &v[1][i]);
 v[4][i]=1.005*v[3][i]; v[0][i]=1.005*v[1][i];}
 if(g<25.) cu=1; else cu=1.486; g2=2.*g; q2g=q*q/g2; fke=1.+fke;
 neq=5*ngate+2;
 if(ibce==3) neq=neq+1; nct=0; wt=.2;
dj=(float**)malloc(neq*sizeof(float*));
  for(i=0;i<neq;i++)dj[i]=(float*)malloc(neq*sizeof(float));
do{sum=0.; funct(f);
 for(i=0;i<ngate;i++) {for(j=0;j<5;j++){ij=5*i+j;
 xx=v[j][i]; v[j][i]*=1.005; funct(ff);
  for(k=0;k<neq;k++) dj[k][ij]=(ff[k]-f[k])/(v[j][i]-xx);
    v[j][i]=xx;}}
 ij++;xx=y1;y1*=1.005; funct(ff);
   for(k=0;k<neq;k++) dj[k][ij]=(ff[k]-f[k])/(y1-xx);y1=xx;
 ij++;xx=q;q*=1.005;q2g=q*q/g2; funct(ff);
  for(k=0;k<neq;k++) dj[k][ij]=(ff[k]-f[k])/(q-xx);
    q=xx; q2g=q*q/g2;
```

```
 if(ibce>2){xx=ye;ye*=1.005; funct(ff);
   for(k=0;k<neq;k++) dj[k][neq-1]=(ff[k]-f[k])/(ye-xx); ye=xx;}
 solveq(neq,dj,f,1,dd,indx);
 for(i=0;i<ngate;i++) for(j=0;j<5;j++){ij=5*i+j;
  v[j][i]-=wt*f[ij]; sum+=fabs(f[ij]);}
 y1-=wt*f[++ij]; sum+=fabs(f[ij]);
 q-=wt*f[++ij]; q2g=q*q/g2; sum+=fabs(f[ij]);
 if(ibce==3){ye-=f[neq]; sum+=fabs(f[neq]);}
 nct++; if(sum<5.) wt=1.;
 printf("NCT=%4d SUM= %12.3fn",nct,sum);for(i=0;i<ngate;i++){
   for(j=0;j<5;j++)printf("%8.3f",v[j][i]);printf("n");}
     printf("%8.3f %7.3f %7.3fn",y1,q,ye);
 if(wt<1.) {printf(" Current weight factor=%6.1f Give new onen",wt);
   scanf("%f",&wt);}
} while((nct<40) && (sum> 1.e-4));
 for(i=0;i<ngate;i++){for(j=0;j<5;j++) fprintf(filo,"%8.3f",v[j][i]);
   fprintf(filo,"n");}
 if(ibce==1)ye=v[0]0[];fprintf(filo,"Y1=%8.3f Q=%10.3f
   Ye%8.3fn",y1,q,ye);
 free(dj);fclose(fili);fclose(filo);
}
```

EXAMPLE PROBLEM 5.13

A channel system receives its water from an upstream reservoir (the entrance loss coefficient is $K_e = 0.2$) and contains three sluice gates at distances of 800, 1800, and 3000 ft downstream from the reservoir. The channel downstream from the third gate is long enough so uniform flow exits in it. The channel is trapezoidal except at the gates where it reduces to a rectangular section. Numbering the channel upstream from the gate by the number of the gate, the channel has the properties given in the table below. Solve for the flow rate and the depths through the channel system if all three gates have their tips 2.7 ft above the channel bottom, and the reservoir supplying the system has a head H = 5 ft.



Ch.	b	m	n	L	S_o	Width Gate (ft)
1	10.0	1.0	0.013	800	0.0008	8
2	9.0	1.0	0.015	1000	0.0005	8
3	9.0	1.0	0.015	1200	0.0007	8
4	9.0	1.0	0.015	long	0.0002	8

Note: b and L in feet.

Solution

A few computations suggest that a flow rate of 200 cfs should be a starting guess for the Newton method to converge to a solution. The input to program SUBMESER for this problem consists of the following (SUBMESER.IN):

```
3 1 32.2 200 5 .2 4.8 3.84 .0001
2.7 8 2.7 8. 2.7 8.
10. 1. .013 800 .0008
9. 1. .015 1000 .0005
9. 1. .015 1200 .0007
9. 1. .015 10000 .0002
4.8 3.0 4.5
4.8 3.0 4.5
4.8 3.0 4.93
```

The solution, i.e., output from program SUBMESER, consists of:

```
     Yd      Y3      Y       Y1      Yu
1   4.689   4.473   3.181   5.351   5.516
2   4.383   4.131   2.770   4.888   5.072
3   4.385   4.133   2.772   4.891   5.075
  Y1= 4.910, Q= 161.083, Ye=  3.840
```

The first three rows of depths in the above output are given in the following order upstream from each gate: Y_d, Y_3, Y, Y_1, and Y_u.

5.9 DESIGN OF SIDE WEIRS

Previously the treatments of side weir have been analyses of given weirs. Now our attention will be the design of side weirs. In Section 4.15, we learned that three possible GVF-profiles might occur across the length of a side weir. It is often desirable to determine the height of a side weir so that the depth at its beginning is above critical so that Case #2 (in which a hydraulic jump occurs within the length of the side weir) or Case #3 (in which a hydraulic jump occurs downstream from the side weir) does not occur. By making the height of the side weir crest larger than this critical height a hydraulic jump can be prevented and Case #1 will occur in which the flow remains subcritical throughout the side weir. In this section, this critical side weir height will be determined, and additional results will be presented to assist in the design of side weirs.

If the water depth at entrance of the side weir is equal to the critical depth Y_c, then the denominator of the governing equation will vanish. The spatially GVF equation (Equation 4.13) across a side weir is as given below. (The second part substitutes the weir equation for the lateral outflow.)

$$\frac{dY}{dx}=\frac{S_o-S_f+\dfrac{Q^2}{gA^3}\left.\dfrac{\partial A}{\partial x}\right|_Y+\dfrac{Qq_o^*}{gA^2}}{1-F_r^2}=\frac{S_o-S_f+\dfrac{Q^2}{gA^3}\left.\dfrac{\partial A}{\partial x}\right|_Y+\dfrac{(2/3)C_d\sqrt{2g}Q(Y-H_W)^{2/3}}{gA^2}}{1-F_r^2}$$

If critical depth occurs at the beginning of the side weir, then the indetermination case of 0/0 exists and the numerator will vanish and the governing equation becomes (the rational for including the lateral outflow and nonprismatic terms is that these apply at the beginning of the lateral outflow section):

$$S_o-S_f+\frac{Q^2}{gA^3}\left.\frac{\partial A}{\partial x}\right|_Y+\frac{(2/3)C_d\sqrt{2g}Q(Y-H_W)^{2/3}}{gA^2}=0$$

Since the flow is critical here, Y_c is substituted for Y in the above equation giving

$$S_o - S_c + \frac{Q_o^2}{gA_c^3}\frac{\partial A}{\partial x}\bigg|_Y + \frac{(2/3)C_d\sqrt{2g}Q_o(Y_c - H_{wc})^{2/3}}{gA_c^2} = 0$$

where Q_0 is the flow rate in the channel upstream from the side weir, and the flow rate at the beginning of the lateral outflow. Solving for H_{wc} yields the following equation:

$$H_{wc} = Y_c - \left[\frac{gA_c^2(S_c - S_o) - \dfrac{Q_o^2}{gA_c^3}\dfrac{\partial A}{\partial x}\bigg|_Y}{(2/3)C_d\sqrt{2g}Q_o}\right]^{2/3} \tag{5.72}$$

As mentioned above, if the actual height is greater than H_{wc}, the flow will be subcritical over the entire length of the side weir; otherwise the flow will be supercritical either along the upstream portion of its length, or over its entire length, and a hydraulic jump will occur.

In the equations that follow, it will be assumed that the channel is prismatic so that Equation 5.72 reduces to

$$H_{wc} = Y_c - \left[\frac{gA_c^2(S_c - S_o)}{(2/3)C_d\sqrt{2g}Q_o}\right]^{2/3} \tag{5.72a}$$

The use of Equations 5.72 and 5.72a requires that S_c be determined. Since the equations are based on the critical flow at the beginning of the side weir the, critical flow equation can be used to determine the relationship between the depth at the beginning of the side weir and the flow rate Q_o, or $Q^2T/(gA^3) = 1$, and Manning's equation can be used to compute S_c. Then Equation 5.72 becomes

$$H_{wc} = Y_c - \left[\frac{gA_c^2\left\{\left(\dfrac{nQ_o}{C_u}\right)^2\left(\dfrac{P_c^{4/3}}{A_c^{10/3}}\right) - S_o\right\}}{\dfrac{2}{3}C_dQ_o\sqrt{2g}}\right]^{2/3} = Y_c - \left[\frac{gA_c^2\left\{\left(\dfrac{n}{C_u}\right)^2\left(\dfrac{gA_c^3}{T_c}\right)\left(\dfrac{P_c^{4/3}}{A_c^{10/3}}\right) - S_o\right\}}{\dfrac{2}{3}C_d\sqrt{\dfrac{gA_c^3}{T_c}}\sqrt{2g}}\right] \tag{5.73}$$

By combining terms this equation becomes

$$H_{wc} = Y_c - \left[\frac{3}{2^{1.5}C_d}\right]^{2/3}\left[\left(\frac{n^2g}{C_u^2}\right)\left(\frac{A_c^{1/6}P_c^{4/3}}{T_c^{1/2}}\right) - S_o(A_cT_c)^{1/2}\right]^{1/2} \tag{5.74}$$

If the channel is rectangular so that $T_c = b$, $P_c = b + 2Y_c$ and $A_c = bY_c$, then Equation 5.74 becomes

$$H_{wc} = Y_c - \left[\frac{3}{2^{1.5}C_d}\right]^{2/3}\left[\left(\frac{n^2g}{C_u^2}\right)\left(\frac{(bY_c)^{1/6}(b + 2Y_c)^{4/3}}{b^{1/2}}\right) - S_obY_c^{1/2}\right]^{2/3} \tag{5.75}$$

For a rectangular channel, the critical depth Y_c can be replaced by $Y_c = \sqrt[3]{q^2/g}$ where $q = Q_o/b$. The graph in Figure 5.12 shows how the critical side weir height H_{wc} varies with the upstream channel

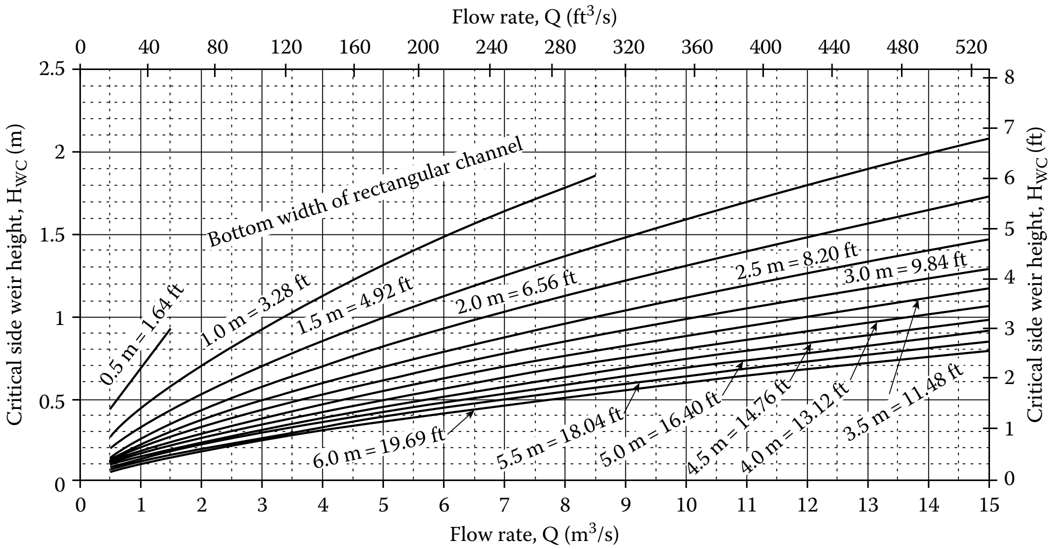


FIGURE 5.12 A plot that shows how the critical side weir height varies with the flow rate in the upstream channel and the bottom width of a rectangular channel. For this plot Manning's n = 0.016 and the channel has a bottom slope $S_o = 0.0005$, and the discharge coefficient for the side weir is $C_d = 0.45$.

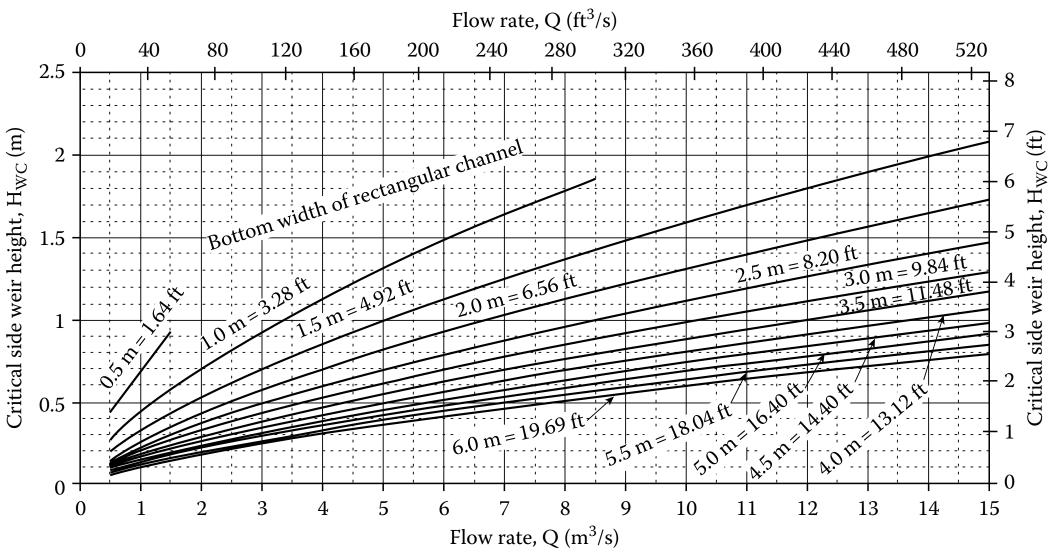


FIGURE 5.13 A plot that shows how the critical side weir height varies with the flow rate in the upstream channel and the bottom width of a rectangular channel. For this plot Manning's n = 0.016 and the channel has a bottom slope $S_o = 0.001$, and the discharge coefficient for the side weir is $C_d = 0.45$.

flow rate and the bottom width of a rectangular channel with a Manning's n = 0.016, and a bottom slope $S_o = 0.0005$ and a side weir discharge coefficient of $C_d = 0.45$.

Figure 5.13 is the same plot with the exception that the slope of the bottom channel has been doubled from 0.0005 to 0.001. In comparing the graphs, there are only modest differences, indicating that the slope of the channel bottom has a very small effect on the critical side weir height.

The small differences in the values of H_{wc} for these two solutions is illustrated in the table below.

	H_{wc} (m)					
	b = 1.0 m		b = 3.0 m		b = 6.0 m	
S_o	Q = 1.0	Q = 8.5	Q = 1.0	Q = 15.0	Q = 1.0	Q = 15.0
0.0005	0.4383	1.8681	0.1920	1.3025	0.0960	0.7986
0.001	0.4416	1.8642	0.1985	1.3140	0.1048	0.8177

The next step in the design of a side weir, after specifying its height above the channel bottom is to determine its length so that it will discharge a desired amount. If the flow is subcritical across the side weir, then the spatially varied flow equation will need to be solved starting at the downstream end of the weir; but in order to do this, its length must be known. This length can be obtained by starting with a reasonable guess and use the Newton method to converge to the length that will provide the specified outflow from the channel. It will, therefore, be assumed that the following are known: (1) the flow rate at the beginning of the side weir, Q_o, (2) the flow rate at the end of the side weir, Q_2, (3) the depth at the end of the side weir, Y_2 (if uniform flow exist downstream then Manning's equation can yield this depth using Q_2, or perhaps a GVF solution yields this depth), and (4) the height of the side weir, and its discharge coefficient. Then the Newton method is implemented so that the spatially varied flow solutions starting at the downstream end of the side weir eventually gives the flow rate at the beginning of the side weir. Or mathematically

$$L^{(m+1)} = L^{(m)} - \frac{\Delta L F(L)}{F(L+\Delta L) - F(L)} \tag{5.76}$$

where F(L) represents the spatially varied solution starting at the downstream end of the weir and ending at its upstream end with a flow rate Q_{GVF} so the $F(L) = Q_{GVF} - Q_o$, which will be driven toward zero through the use of Equation 5.76. The details of the solution implementation are provided in the program listing in the following example problem.

EXAMPLE PROBLEM 5.14

Design a side weir with a discharge coefficient $C_d = 0.45$ that will discharge 40% of the upstream flow of $Q_o = 10\,m^3/s$. The channel containing this weir is trapezoidal with b = 2 m and m = 1.5, and has an n = 0.016 and a bottom slope $S_o = 0.0005$. The flow is to remain subcritical, and the height of the weir above the channel bottom should be 5% larger than that needed to produce critical flow at its beginning.

Also design this side weir so that the channel changes to rectangular at its end with a bottom width of $b_2 = 2$ m (e.g., only the side slope changes across the side weir).

Solution

The normal upstream depth is $Y_o = 1.664$ m, so $E_o = 1.755$ m, $A_o = 7.482\,m^2$, $P_o = 8.000$ m, $T_o = 6.992$, and the critical depth in this channel associated with $Q = 10\,m^3/s$ is $Y_c = 1.048$ m so $S_c = 0.003259$, $A_c = 3.743$ m, $P_c = 5.778$ m, $T_c = 5.144$ m. As a first step in the solution process, we solve Equation 5.72 to find the critical height of the side weir as $H_{wc} = 0.95462$ m. Adding 5% to this gives the height of the side weir as $H_w = 1.0024$ m. Next under the assumption that uniform flow exists downstream from the weir, we solve Manning's equation for Y_2 using for a flow rate of $6\,m^3/s$ in the downstream channel, and get $Y_2 = 1.298$ m. Program LENSIDEW is designed to implement the Newton method in solving for lengths of the side weirs. The input to this program consists of

```
3 1.e-5 -.5 1.298 6 10 .016 .0005 2 1.5 2 1.5 35 1.0024 .45 9.81
5 10 .15
```

The solution from the program consists of: (*Note*: The length of side weir needed is L = 40.1 m.)
Length of side weir needed = 40.14

x (m)	Y (m)	Q (m^3/s)	q* (m^2/s)	F_r^2
40.1	1.298	6.00	0.0359	0.401
37.6	1.283	6.47	0.1973	0.442
35.1	1.267	6.94	0.1807	0.486
32.6	1.250	7.37	0.1639	0.529
30.1	1.233	7.76	0.1472	0.572
27.6	1.216	8.11	0.1310	0.614
25.1	1.198	8.42	0.1154	0.655
22.6	1.182	8.69	0.1008	0.694
20.1	1.165	8.92	0.0873	0.732
17.6	1.150	9.12	0.0752	0.767
15.1	1.136	9.30	0.0647	0.800
12.6	1.123	9.45	0.0559	0.830
10.1	1.113	9.58	0.0488	0.856
7.6	1.105	9.69	0.0435	0.879
5.1	1.099	9.80	0.0397	0.897
2.6	1.095	9.89	0.0374	0.912
0.1	1.093	9.99	0.0362	0.923

Program LENSIDEW.FOR (uses Newton method to solve for length of side weir)

```
      EXTERNAL DYX,FUN
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV
      COMMON /TRAN/CMA,CDG,HW,G,XO,BO,FMO,FN,SO,Q,QS1,DB,DM,DELX,
     &ADELX5,YB,TOL,QSTART,QB,NPRINT,IOUT,MPRT
      WRITE(6,* )'GIVE IOUT,TOL,DELX,YB,QB,','QSTART,FN,SO,BO,FMO,
     &B2,FM2,Est. L','Hw,Cd,g,MPRT,MAX,ERR'
      READ(5,*)IOUT,TOL,DELX,YB,QB,QSTART,FN,SO,BO,FMO,B2,FM2,
     &XBEG,HW,CD,G,MPRT,MAX,ERR
      CDG=.666667*CD*SQRT(2.*G)
      CMA=1.
      IF(G.GT.30.) CMA=1.486
      DB=(B2-BO)/XBEG
      DM=(FM2-FMO)/XBEG
      ADELX5=.5*DELX
      QS1=CDG*(YB-HW)**1.5
      NPRINT=0
C Newton Method
      FL=XBEG
      NCT=0
10    FF=FUN(FL)
      DIF=FF/(FUN(FL+1.)-FF)
      FL=FL-DIF
      DB=(B2-BO)/FL
      DM=(FM2-FMO)/FL
      NCT=NCT+1
      X=FL
      IF(NCT.LT.MAX.AND.ABS(DIF).GT.ERR) GO TO 10
      IF(NCT.EQ.MAX) WRITE(*,*)' Didnot converge'
      WRITE(IOUT,110) FL
110   FORMAT(' Length of side weir needed=',F8.2)
```

```
      WRITE(*,110) FL
      NPRINT=1
      C Call to print out w.s. profile
      DIF=FUN(FL)
      END
      SUBROUTINE DYX(X,Y,YPRIME)
      REAL Y(1),YPRIME(1)
      COMMON /TRAN/CMA,CDG,HW,G,XO,BO,FMO,FN,SO,QO,QS1,DB,DM,DELX,
     &ADELX5,YB,TOL,QSTART,QB,NPRINT,IOUT,MPRT
      B=BO+DB*X
      FM=FMO+DM*X
      A=(B+FM*Y(1))*Y(1)
      T=B+2.*FM*Y(1)
      P=B+2.*SQRT(FM*FM+1.)*Y(1)
      IF(Y(1).GT.HW) THEN
      QS=CDG*(Y(1)-HW)**1.5
      ELSE
      QS=0.
      ENDIF
      Q=QO-.5*(X-XO)*(QS1+QS)
      SF=(FN*Q*(P/A)**.66666667/(CMA*A))**2
      A2=A*A*G
      FR2=Q*Q*T/(A*A2)
      YPRIME(1)=(SO-SF+Q*QS/A2+Q*Q/(A*A2)*Y(1)*(DB+Y(1)*
     &DM))/(1.-FR2)
      RETURN
      END
      FUNCTION FUN(XX)
      EXTERNAL DYX
      REAL Y(1),XP(1),YP(1,1),WK1(1,13),YPRIME(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV
      COMMON /TRAN/CMA,CDG,HW,G,XO,BO,FMO,FN,SO,Q,QS1,DB,DM,
     &DELX,ADELX5,YB,TOL,QSTART,QB,NPRINT,IOUT,MPRT
      H1=.01
      Q=QB
      XO=XX
      X=XX
      Y(1)=YB
      IF(NPRINT.GT.0) WRITE(IOUT,100) X,Y,QB,QS1,SQRT(Q**2*
     &(B+2.*FM*Y(1))/(G*((B+FM*Y(1))*Y(1))**3))
      MP=0
2     XZ=X+DELX
      CALL ODESOL(Y,YPRIME,1,X,XZ,TOL,H1,0.,1,XP,YP,WK1,DYX)
      IF(Y(1).GT.HW) THEN
      QS2=CDG*(Y(1)-HW)**1.5
      ELSE
      QS2=0.
      ENDIF
      XO=XZ
      Q=Q-ADELX5*(QS1+QS2)
      QS1=QS2
      X=XZ
      B=BO+DB*X
      FM=FMO+DM*X
      IF(NPRINT.EQ.0) GO TO 5
```

```
      MP=MP+1
      IF(MOD(MP,MPRT).EQ.0)
      WRITE(IOUT,100) X,Y,Q,QS1,SQRT(Q**2*(B+2.*FM*Y(1))/(G*((B+
     &FM*Y(1))*Y(1))**3))
100   FORMAT(1X,F8.1,F10.3,F10.2,F10.4,F10.3)
5     IF(X .GT. 0.) GO TO 2
99    FUN=QSTART-Q
      RETURN
      END
```

In providing a guess for the length to this program, it is best to underguess than overguess because too large a length might create critical depth before arriving at the beginning of the side weir, and cause the computations to abort as dY/dx becomes infinite.

The second part of the problem assumes that the side slope will decrease from 1.5 to 0 at the end of the side weir. Now the downstream normal depth is computed as $Y_2 = 2.656$ m. The input to solve this part might consist of

```
3 1.e-5 -.05 2.656 6 10 .016 .0005 2 0 2 1.5 1.5 1.0024 .45 9.81
5 10 .15
```

The solution from the program is:

Length of side weir needed = 1.46

x (m)	Y (m)	Q (m^3/s)	q* (m^2/s)	F_r^2
1.5	2.656	6.00	1.8397	0.224
1.4	2.654	6.27	2.8217	0.104
1.3	2.653	6.55	2.8169	0.114
1.2	2.650	6.84	2.8112	0.125
1.1	2.648	7.12	2.8041	0.136
1.0	2.644	7.40	2.7954	0.149
0.9	2.640	7.68	2.7846	0.164
0.8	2.635	7.95	2.7710	0.180
0.7	2.628	8.23	2.7537	0.198
0.6	2.619	8.50	2.7313	0.219
0.5	2.607	8.78	2.7017	0.242
0.4	2.591	9.04	2.6618	0.270
0.3	2.569	9.31	2.6067	0.302
0.2	2.538	9.56	2.5283	0.339
0.1	2.491	9.81	2.4123	0.384

Notice that the length of side weir has reduced to only 1.46 m. Why the very large reduction in length? Two factors contribute to this: (1) the much larger downstream depth in the rectangular channel than in the trapezoidal channel results in much larger side discharges. (*Note*: The side discharge varies to the 1.5 power of the height over the weir) and (2) the effect of the nonprismatic term in the GVF equation.

In using program LENSIDEW, it is necessary to provide a reasonable guess for the length of side weir needed because in carrying out the solution of the spatially varied flow the depth may get too close to critical depth otherwise, resulting in a division by near zero in evaluating dY/dx. Also one does not want to specify too small a convergence value (ERR in the program) because this cannot be achieved. The value of 0.15 used above appears to be quite easily achieved, but also suggests that the length of 1.46 is likely not good to two digits beyond the decimal point. In solving the ODE, the outflow is obtained by taking the average to the q* on an increment of DELX (the third item of input). In other words, the ODE solver ODESOL is called upon on this increment, and this calling is repeated twice for each Newton iteration. The value given to DELX should be larger for longer lengths of side weir and is negative because the GVF solution takes place from the end to the beginning of the side weir.

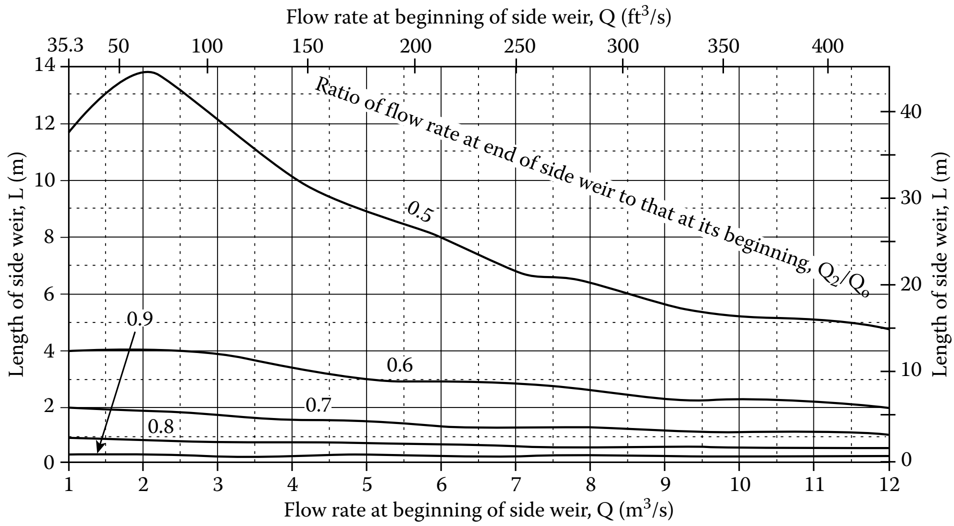


FIGURE 5.14 Lengths of side weir needed to discharge varying ratios of downstream to upstream channel flow, under varying upstream flow rates. The following were specified in obtaining this graph: the channel is rectangular with a constant width b = 2 m, the side weir height is 10% above the critical height, the bottom slops is constant, $S_o = 0.0005$, with Manning's n = 0.016, and the discharge coefficient $C_d = 0.45$.

There are too many variables that influence the length of side weirs needed to attempt to display all these influences graphically. However, the graph in Figure 5.14 illustrates that the lengths vary over a wide range. In obtaining the series of solutions needed to plot this graph, the following were specified: (1) the channel is rectangular with a bottom width of b = 2 m and this does not change (e.g., the channel is prismatic both upstream, across the side weir, and downstream there from), (2) the height of the weir's crest is 10% above the critical value determined from Equation 5.72, or $H_w = 1.1H_{wc}$, (3) normal depth occurs downstream of the side weir, and (4) the following parameters apply: n = 0.016, $S_o = 0.0005$, and $C_d = 0.45$. The five lines on this graph are for ratios downstream to upstream flow rate from 0.9 to 0.5. Notice that the lengths of side weirs are small when the downstream flow rate is a large fraction of that upstream. For example with $Q_2/Q_o = 0.9$ and $Q_o = 1.0\,m^3/s$ the length L = 0.4 m. As the upstream flow rate increases to $12\,m^3/s$, the length decreases to 0.2 m. When the side weir discharges one-half the flow in the upstream channel, e.g., $Q_2/Q_o = 0.5$, then when $Q_o = 1.0\,m^3/s$ the length is L = 11.7 m, and when $Q_o = 12.0\,m^3/s$ the length is L = 4.6 m. Notice that the lengths decrease with increasing upstream flow rates.

5.10 OPTIMAL DESIGN OF TRAPEZOIDAL CHANNELS CONSIDERING TOTAL COSTS

In this section, we will deal with the optimal design of a channel by minimizing the total cost, including the lining cost, the excavation and the right-of-way costs. The approach handles two slightly different channel configurations as shown in the sketch in Figure 5.15. The details in the mathematics that follow assume the channel is trapezoidal.

Referring to Figure 5.15 above, the total cost per unit of length of the channel is defined by the following:

$$\begin{aligned} C &= C_r e_r(b + 2e_b SM - c_o m e_r) + 2C_b e_b SM \times Z + C_e(b + 2e_b SM + 2mZ) \\ &\quad + C_x\left[(b + 2e_b SM + mZ)Z + e_r(b + 2e_b SM - c_o m e_r)\right] \end{aligned} \tag{5.77}$$

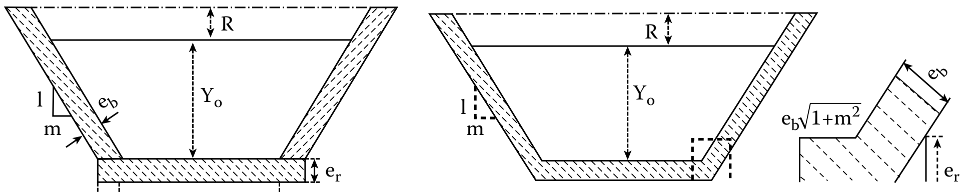


FIGURE 5.15 Variables that define trapezoidal channel's cross-sections.

where z = y_o + R and $SM = \sqrt{1+m^2}$ and the other variables are the following: Y_o is the uniform depth (length), b is the bottom width (length), m is the sides slope, e_b is the thickness of sides lining (length), e_r is the thickness of the channel bottom (length), R is the free board (length), C_x is the unit cost of excavated materials ($/length^3), C_e is the unit cost of the right-of-way ($/length^2), C_b and C_r are the unit costs of the lining respectively of the channel sides and its bottom, c_0 is constant, which is 0 or 1 depending on the cross section profile type. If the profile is type 1, then c_o = 0, and if the type is 2, c_o = 1. While C is referred to as the total cost, it does not represent all the costs associated with a channel. A channel system needs control structures. Seldom is the natural terrain at a constant slope, so that grading is required. The right-of-way cost C_e above is just for the channel's top width, but often the actual width of a right-of-way is much wider as need for a maintain road, etc. However, such needed width is constant, and therefore not appropriate to include in an analysis to find the optimal channel dimensions.

The minimum cost is for a specified flow rate, Q, and the channel dimensions must satisfy Manning's equation:

$$Q = \frac{C_u}{n}\frac{A^{5/3}}{p^{2/3}}\sqrt{S_o} = \frac{C_u}{n}\ R_h^{2/3}\sqrt{S_o} \tag{5.78}$$

where C_u = 1 for SI units and C_u = 1.486 for ES units, n is the Manning's coefficient A = (b + mY_o)Y_o is the cross section area, $P = b + 2Y\sqrt{1+m^2}$ is the wetted perimeter, R_h = A/P is the hydraulic radius and S_o is the bottom slope.

The optimization model to be solved can be written as follows:

$$\begin{aligned}\text{Min } C &= C_r e_r(b + 2e_b SM - c_o m e_r) + 2C_b e_b SMZ + C_e(b + 2e_b SM + 2mZ)\\ &\quad + c_x[(b + 2e_b XM + mZ) + e_r(b + 2e_b SM - c_o m e_r)]\end{aligned} \tag{5.79}$$

Subjected to satisfying Manning's equation, or

$$g = AR_h^{2/3} - \frac{nQ}{\sqrt{S_o}} = 0$$

This optimization problem can be solved assuming that the unknown variables are as follows: (1) the uniform depth Y_o, (2) the bottom width b, and (3) the channel sides slope m. (m can be considered a design parameter that may either be specified, or be solved for.)

Using the Lagrangian method, the optimization problem becomes

$$\begin{aligned}L &= C_r e_r(b + 2e_b SM - c_o m e_r) + 2C_b e_b SM \times Z + C_e(b + 2e_b SM + 2mZ)\\ &\quad + C_x\left[(b + 2e_b SM + m \times Z) \times Z + e_r(b + 2e_b SM - c_o m e_r)\right] + \lambda\left(AR_H^{2/3} - \frac{nQ}{C_u\sqrt{S_o}}\right)\end{aligned}$$

The Lagrangian function depends on the four variables Y_o, b, m, and λ, and its minimum is reach if all its partial derivatives with respect to each of the variables Y_o, b, m, and λ are equated to zero.

$$\frac{\partial L}{\partial Y_o} = 2C_b e_b SM + C_x(b + 2e_b SM + 2mZ) + 2mC_e + \frac{1}{3}\lambda R_H^{2/3}(5B - 4R_H SM) = 0 \tag{5.80}$$

$$\frac{\partial L}{\partial b} = C_r e_r + C_x(Z + e_r) + C_e + \frac{1}{3}\lambda R_H^{2/3}(5Y_o - 2R_H) = 0 \tag{5.81}$$

$$\frac{\partial L}{\partial m} = \frac{2me_b}{SM}\left[e_r(C_r + C_x) + Z(C_x + C_b) + C_e\right] + Z(Z\,C_x + 2C_e) - c_o e_r^2(C_x + C_r)$$

$$+\frac{1}{3}\lambda R_h^{2/3} Y_o\left(5Y_o - \frac{4m}{SM}R_H\right) = 0 \tag{5.82}$$

$$\frac{\partial L}{\partial \lambda} = AR_H^{2/3} - \frac{nQ}{\sqrt{S_o}} = 0 \tag{5.83}$$

This system of equations may be written as

$$2C_b e_b SM + C_x(b + 2e_b SM + 2mZ) + 2mC_e = -\frac{1}{3}\lambda R_H^{2/3}(5B - 4R_H SM) \tag{5.84}$$

$$C_r e_r + C_x(Z + e_r) + C_e = -\frac{1}{3}\lambda R_H^{2/3}(5Y_o - 2R_H) \tag{5.85}$$

$$\frac{2me_b}{SM}\left[e_r(C_r + C_x) + Z(C_x + C_b) + C_e\right] + Z(ZC_x + 2C_e) - c_o e_r^2(C_x + C_r)$$

$$= -\frac{1}{3}\lambda R_h^{2/3} Y_o\left(5Y_o - \frac{4m}{SM}R_H\right) \tag{5.86}$$

$$AR_H^{2/3} - \frac{nQ}{C_u\sqrt{S_o}} = 0 \tag{5.87}$$

To eliminate the Lagrange multiplier λ, the fourth variable, divide Equations 5.84 and 5.86 by Equation 5.85 to get the following system of three nonlinear equations:

$$F_1 = \frac{2C_b e_b SM + C_x(b + 2e_b SM + 2mZ) + 2mC_e}{C_r e_r + C_x(Z + e_r) + C_e} - \frac{(5B - 4R_h SM)}{(5Y_o - 2R_h)} = 0$$

$$F_2 = \frac{\frac{2me_b}{SM}\left[e_r(C_r + C_x) + Z(C_x + C_b) + C_e\right] + Z(ZC_x + 2C_e) - c_o e_r^2(C_x + C_r)}{C_r e_r + C_x(Z + e_r) + C_e}$$

$$-\frac{Y_o\left(5Y_o - \frac{4m}{SM}R_h\right)}{5Y_o - 2R_h} = 0$$

$$F_3 = AR_h^{2/3} - \frac{nQ}{C_u\sqrt{S_o}} = 0 \tag{5.88}$$

If the channel side slope, m, is known, then equation F_2 is dropped from the above system of equations, and the remaining two equations are solved, by a method such as the Newton method, to determine the uniform depth, Y_o, and the channel bottom width, b. The following Fortran program OPTIMALN.FOR implements this solution.

Definitions of Variables in Program: Y(1) = Channel maximum Capacity channel m^3/s, Y(2) = Channel bottom slope, Y(3) = Manning's coefficient, Y(4) = Channel free board (m), Y(5) = channel sides lining thickness (m), Y(6) = channel bottom lining thickness (m) Y(7) = sides lining unit cost MU/m^3, Y(8) bottom lining unit cost MU/m^3, Y(9) excavation unit cost MU/m^3, Y(9) right of way MU/m^2, C0 = coefficient, which is 0 or 1 depending on the shape of the cross section, X(1) = uniform depth (m), X(2) = bottom width (m), X(3) = side slope, YY(1) = mean velocity m/s, YY(2) = channel total Cost, D = Jacobian matrix.

Called functions: `AR` = cross section, `PER` = wetted perimeter, `TOP` = Top width, `RH` = hydraulic radius, `DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)` function, evaluate the derivative off the objective function with respect of the three variables, `COST(COUT,X)` a function that evaluate the channel cost.

```
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER (NV=10)
      EXTERNAL AR,TOP,RH,PER,FUN
      CHARACTER*40 VAR1(5),ANS*1
      DIMENSION X(3),YY(2)
      COMMON /ALL/ Y(10),C0,QNS
      DATA var1/' Uniform depth (m) =','Bottom width (m)= ',' Side
     &slope = ',' Flow velocity (m/s) =',' Canal total cost ($) = '/
      WRITE(*,'(//30X,A\)') ' Profile of type 1 or 2:  '
      READ(*,*) ITYPE
      C0=0.
      IF(ITYPE.EQ.2) C0=1
      WRITE(*,'(//,2X,A2)') '  '
      CALL KNOWN(Y,NV)
2     WRITE(*,'(//10X,A\)')' Is the canal Side Slope known or
     &not ? [Y/N] '
      READ(*,'(A1)') ANS
      IF(ANS.EQ.'n'.OR.ANS.EQ.'N') THEN
      N=3
C Initial value of the side slope
      X(3)=1./SQRT(3.)
      ELSE
      N=2
      WRITE(*,'(/20X,A\)') ' Side Slope  m = '
      READ(*,*) X(3)
      ENDIF
      QNS=Y(1)*Y(3)/SQRT(Y(2))
      SM=SQRT(1.+X(3)*X(3))
C Dimensions of the most efficient section are used to get
C the initial value of the variable Y0 and b
      X(1)=(4.*(QNS/(2*SM-X(3)))**3)**.125
      X(2)=2.*X(1)*(SM-X(3))
      CALL NEWT(X,N)
```

```
      YY(1)=Y(1)/AR(X(1),X(2),X(3))
      CALL COST(YY,X)
      DO 10 I=1,3
10    WRITE(*,'(15X,A30,2X,F12.6)') VAR1(I),X(I)
      DO 20 I=1,2
20    WRITE(*,'(15X,A30,2X,F12.3)') VAR1(I+3),YY(I)
      END
      SUBROUTINE NEWT(X,N)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION D(3,3),F(3),X(3)
      COMMON /ALL/ Y(10),C0,QNS
      DATA EPS,FAC,KMAX/.0000001,.001,25/
      ITER=0
4     ITER=ITER+1
      DO 10 I=1,N
10    F(I)=FUN(X,I)
      DO 30 I=1,N
      DX=FAC*(X(I)+1.)
      X(I)=X(I)-DX
      DO 20 J=1,N
20    D(J,I)=(F(J)-FUN(X,J))/DX
30    X(I)=X(I)+DX
      CALL SOLVE(D,F,N)
      SUM=0.0
      DO 40 I=1,N
      X(I)=X(I)-F(I)
      IF(X(I).LT.0.) X(I)=0.001
40    SUM=SUM+F(I)*F(I)
      SUM=SQRT(SUM)
      IF(ITER.GT.KMAX) THEN
      WRITE(*,100) ITER
      ELSE
      IF(SUM.GT.EPS) GO TO 4
      ENDIF
      RETURN
100   FORMAT(//20X,' Does not converge after:',I4,' ITERATIONS'/)
      RETURN
      END
      FUNCTION FUN(X,I)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(3)
      COMMON /ALL/ Y(10),C0,QNS
      GO TO (4,6,8),I
4     FUN=AR(X(1),X(2),X(3))*RH(X(1),X(2),X(3))**(2./3.)-QNS
      RETURN
6     CALL DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)
      FUN=DCY*DFB-DFY*DCB
      RETURN
8     CALL DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)
```

```
      FUN=DCM*DFB-DFM*DCB
      RETURN
      END
      SUBROUTINE DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(3)
      COMMON /ALL/ Y(10),C0,QNS
      EQUIVALENCE (Y(3),RV),(Y(4),EB),(Y(5),ER),(Y(6),CB) ,
     &(Y(7),CR),(Y(8),CX),(Y(9),CE)
      Z=X(1)+RV
      SM=SQRT(1.+X(3)*X(3))
      B1=X(2)+2.*SM*EB
      RH1=RH(X(1),X(2),X(3))
      DCY=2.*SM*CB*EB+CX*TOP(Z,B1,X(3))+2.*CE*X(3)
      DCB=CR*ER+CX*(Z+ER)+CE
      DCM=2.*EB*X(3)*(ER*(CR+CX)+Z*(CB+CX)+CE)/SM+Z*
     &(Z*CX+2.*CE)-C0*ER**2*(CR+CX)
      DFY=5.*TOP(X(1),X(2),X(3))-4.*RH1*SM
      DFB=5.*X(1)-2.*RH1
      DFM=X(1)*(5.*X(1)-4.*RH1*X(3)/SM)
      RETURN
      END
      SUBROUTINE SOLVE(D,F,N)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION D(3,3),F(3)
      DO 20 K=1,N-1
      DO 20 I=K,N-1
      XMULT=D(I+1,K)/D(K,K)
      DO 10 J=K+1,N
10    D(I+1,J)=D(I+1,J)-XMULT*D(K,J)
20    F(I+1)=F(I+1)-XMULT* F(K)
      F(N)=F(N)/D(N,N)
      F(N-1)=(F(N-1)- F(N)*D(N-1,N))/D(N-1,N-1)
      IF(N.GT.2) THEN
      F(N-2)=(F(N-2)-F(N-1)*D(N-2,N-1)-F(N)*D(N-2,N))/D(N-2,N-2)
      ENDIF
      RETURN
      END
      SUBROUTINE COST(YY,X)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(3),YY(2)
      COMMON /ALL/ Y(10),C0,QNS
      EQUIVALENCE (Y(4),RV),(Y(5),EB),(Y(6),ER),(Y(7),CB),
     &(Y(8),CR),(Y(9),CX),(Y(10),CE)
      Z=X(1)+RV
      SM=SQRT(1.+X(3)*X(3))
      B1=X(2)+2.*SM*EB
      YY(2)=CR*ER*(B1-C0*X(3)*ER)+2.*CB*EB*SM*Z+CX*(AR(Z,
     &B1,X(3))+ ER*(B1-C0*X(3)*ER))+CE*TOP(Z,B1,X(3))
      RETURN
      END
```

```
      SUBROUTINE KNOWN(Y,NV)
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*40 VAR(10)
      DIMENSION Y(NV)
      DATA var/' Q Canal maximum Capacity (m3/s)    ',
     &' S0 Bottom slope                       ',
     &' n Manning coefficient                 ',
     &' Rv Free board (m)                     ',
     &' Eb Sides lining thickness (m)         ',
     &' Er Bottom lining thickness(m)         ',
     &' Cb Sides lining unit cost  ($/m3)     ',
     &' Cr Bottom lining unit cost ($/m3)     ',
     &' Cx Excavation unit($/m3)              ',
     &' Ce Right of way unit cost ($/m2)      '/
      DO 10 I=1,NV
      WRITE(*,'(15X,A45,2X,A3\)') VAR(I),'= '
10    READ(*,*) Y(I)
      RETURN
      END
      FUNCTION AR(Y,B,FM)
      IMPLICIT REAL*8(A-H,O-Z)
      AR=(B+FM*Y)*Y
      RETURN
      END
      FUNCTION TOP(Y,B,FM)
      IMPLICIT REAL*8(A-H,O-Z)
      TOP=B+2.*FM*Y
      RETURN
      END
      FUNCTION PER(Y,B,FM)
      IMPLICIT REAL*8(A-H,O-Z)
      PER=B+2.*Y*SQRT(1.+FM*FM)
      RETURN
      END
      FUNCTION RH(Y,B,FM)
      IMPLICIT REAL*8(A-H,O-Z)
      RH=AR(Y,B,FM)/PER(Y,B,FM)
      RETURN
      END
```

EXAMPLE PROBLEM 5.15

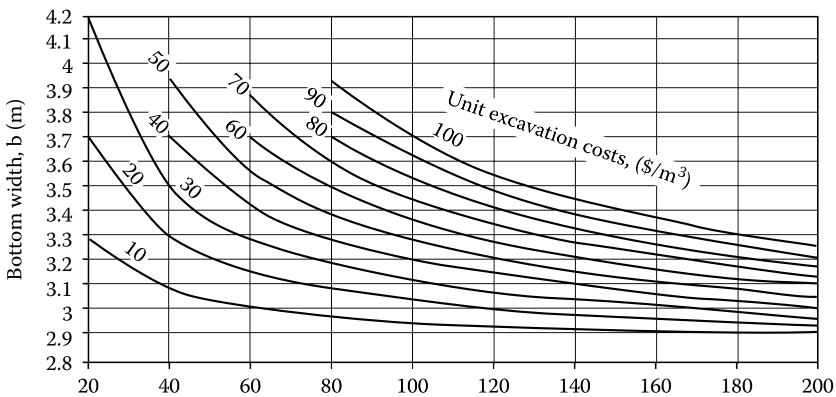
Assume that the design flow rate is $Q = 20\,m^3/s$ and the values of bottom slope, Manning's coefficient and the free board are $S_o = 0.0002$, $n = 0.016$, and $R = 0.5\,m$, respectively. The thicknesses of the lining for the sides and the bottom are the same with $e_b = e_r = 0.1\,m$. The unit cost of lining of the bottom and the sides is $C_b = C_r = \$100$ and the unit costs of the excavation and the right-of-way are respectively $C_x = \$60$ and $C_e = \$15$. What is the uniform depth, Y_o, the channel bottom width, b, and the channel sides slope, m?

Solution

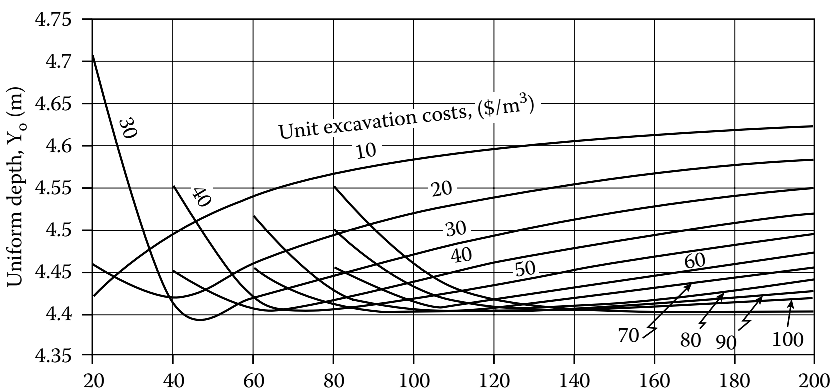
The input needed to obtained the solution using the above program consists of:

$$Y_o = 3.257\text{ m},\quad b = 4.609\text{ m},\quad m = 0.198,\quad V = 1.169\text{ m/s}\quad\text{and}\quad\text{cost} = \$1493.31.$$

Because of the number of variables involved it is not practical to graphically display all the relationships between these variables that exist. However, Figure 5.16 contains five graphs that show how the bottom width, b, the normal depth Y_o, the side slope, m, the velocity V, and the total cost are related to the unit costs of lining the channel and the unit excavation cost. To obtain these relationships, the following were specified: (1) the unit side and bottom lining costs were made equal, (2) the thickness of the bottom and sides were assumed equal and equal to 0.15 m, e.g., $e_b = e_r = 0.15$ m, (3) the flow rate is $Q = 20\,m^3/s$, (4) Manning's n = 0.016, the free board R = 0.5 m, (6) the right-of-way cost is \$15/m, and (7) the bottom slope is $S_o = 0.0002$. Also the series of solutions used to obtain these plots were for the type 2 channel shown in Figure 5.15. The following general observations can be made: (1) the bottom width, b, of the channel decreases with increasing lining costs, and increases with increasing excavation costs. For larger lining costs, the rate of decrease is smaller. On the other hand, the uniform depth, Y_o, generally increases with increasing lining costs and decreases with increasing excavation costs. The exception to these latter trends is when the lining costs are small. It is also interesting to note that the optimal side slopes of the channel are relatively small, all less than the hydraulic optimal side slope of $1/\sqrt{3} = 0.577$, and as the lining costs become small, especially with larger excavation costs, m approaches zero, indicating that a rectangular channel is the optimal shape. In fact with smaller lining cost, and larger excavating costs used to obtain these graphs, the methodology described above failed to find a solution. Therefore, the curves on these graphs end where no solution was available. It is also interesting to note the very small range of velocities from just larger than 1 m/s to just smaller than 1.2 m/s that result from these optimal design solutions. These velocities tend to converge

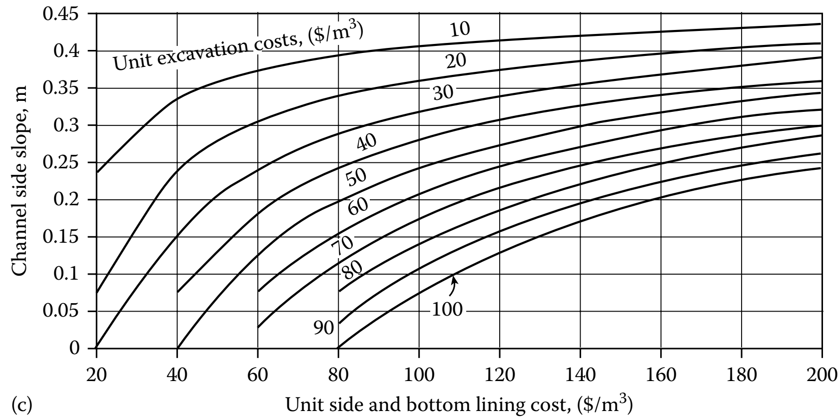


(a)

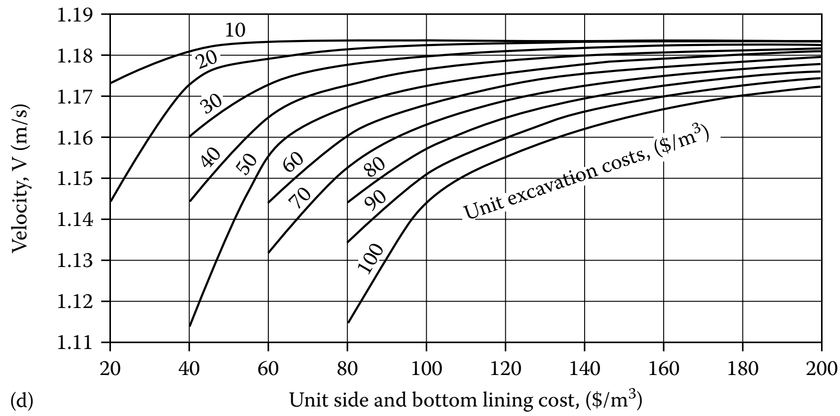


(b)

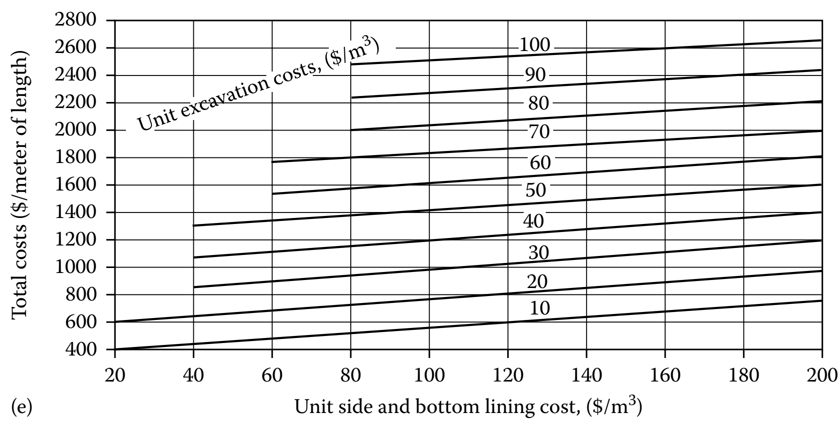
FIGURE 5.16 Five graphs that show how optimal b, Y_o, m, V, and C vary with lining and excavation costs. The following parameters are specified for the series of solutions that gives these graphs: $Q = 20\,m^3/s$, n = 0.016, $e_b = e_r = 0.15$ m, R = 0.5 m, C_e = \$15/m and $S_o = 0.0002$.



(c)



(d)



(e)

FIGURE 5.16 (continued)

toward 1.18 m/s as the lining costs become large. Since the curves on the last graph in Figure 5.16 are nearly straight lines, and these lines are nearly equally spaced, we can conclude that there is essentially a linear relationship between the total costs and the lining cost and the excavation cost. A least squared regression analysis gives the following such linear equation:

$$C = 131.91 + 21.603C_x + 131.906C_b$$

Remember this approximate equations is limited to the parameters used in obtaining the series of solution whose results are plotted in Figure 5.16.

5.10.1 Additional Optional Analyses

The velocities from the above series of solutions are sufficiently small to be acceptable, but these velocities were obtained by specifying a relatively flat channel with a bottom slope of $S_o = 0.0002$. As the slope of the bottom of the channel increases, one would expect that the velocities resulting from an optimal solution would also increase. To show this trend, the five graphs shown in Figure 5.17 were obtained by plotting the results from series of solutions with the same parameters as the graphs in Figure 5.16 with the exception that the bottom slope of the channel has been increased tenfold to $S_o = 0.002$. Again the optimal velocities are confined to a relatively small range from 2.62 to 2.82 m/s, but are over two times as large as those obtained when $S_o = 0.0002$. Clearly for steeper channels, the velocities obtained from the above optimization can become too large and the above methodology needs to be modified.

If the maximum velocity is limiting one more constraint must be added that specifies the maximum allowable velocity:

$$\begin{aligned} \mathrm{MicC} = {} & C_r e_r(b + 2e_b SM - c_o m e_r) + 2C_b e_b SMZ + C_e(b + 2e_b SM + 2mZ) \\ & + c_x[(b + 2e_b SM + mZ) + e_r(b + 2e_b SM - c_o m e_r)] \end{aligned} \tag{5.89}$$

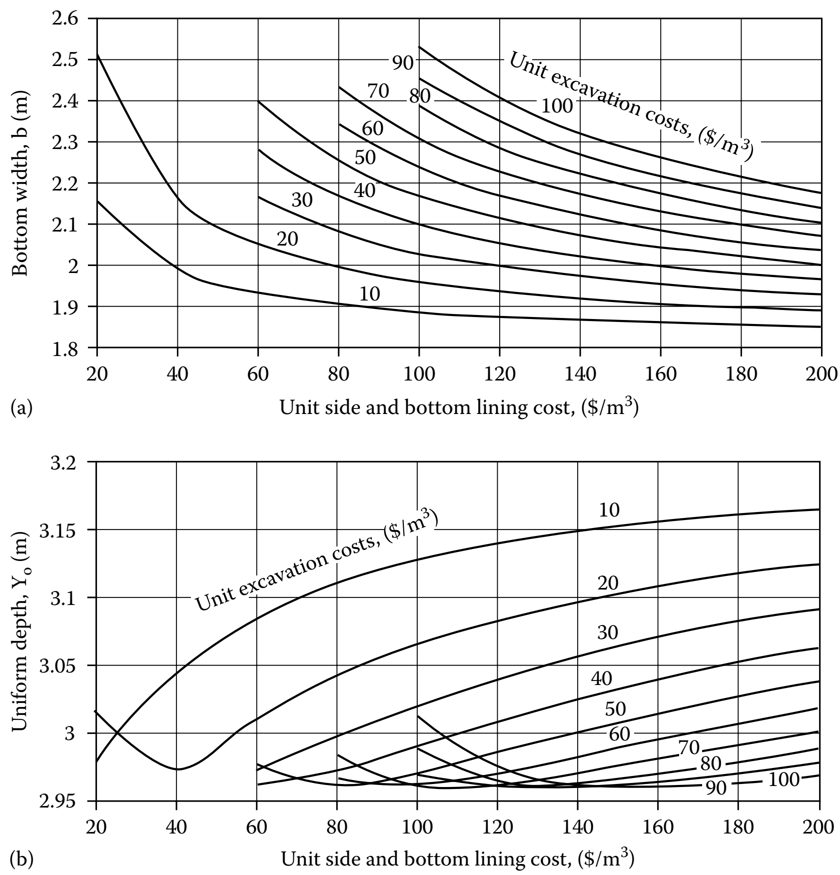


FIGURE 5.17 Five graphs that show how optimal b, Y_o, m, V, and C vary with lining and excavation costs. The following parameters are specified for the series of solutions that gives these graphs: $Q = 20\ m^3/s$, $n = 0.016$, $e_b = e_r = 0.15$ m, $R = 0.5$ m, $C_e = \$15/m$ and $S_o = 0.002$.

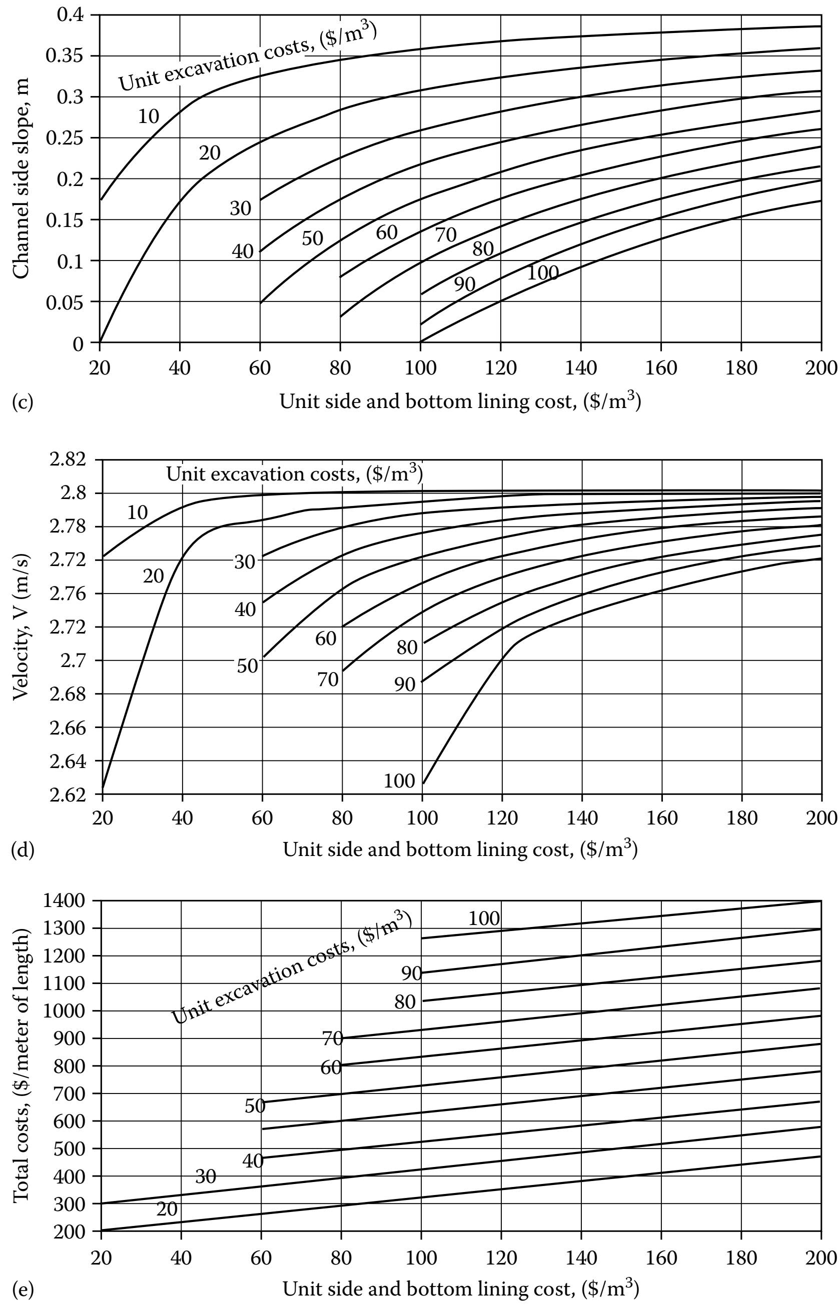


FIGURE 5.17 (continued)

subjected to

$$g_1 = AR_h^{2/3} - \frac{nQ}{\sqrt{S_o}} = 0$$

and

$$g_2 = A - \frac{Q}{V_{max}} = 0$$

Again applying Lagrangian method we get the function:

$$L = C_r e_r (b + 2e_b SM - c_o m e_r) + 2C_b e_b SM \times Z + C_e (b + 2e_b SM + 2mZ)$$
$$+ C_x \left[(b + 2e_b SM + mZ)Z + e_r (b + 2e_b SM - c_o m e_r)\right] + \lambda_1 \left(AR_H^{2/3} - \frac{nQ}{\sqrt{S_o}} \right) + \lambda_2 \left(A - \frac{Q}{V_{max}} \right) \tag{5.90}$$

The partial derivatives of the function L with respect of the three variables are as follows:

$$\frac{\partial L}{\partial Y_o} = 2C_b e_b SM + C_x (b + 2e_b SM + 2mZ) + 2mC_e + \frac{1}{3}\lambda_1 R_H^{2/3}(5B - 4R_H SM) + \lambda_2 B = 0 \tag{5.91}$$

$$\frac{\partial L}{\partial b} = C_r e_r + C_x (Z + e_r) + C_e + \frac{1}{3}\lambda_1 R_H^{2/3}(5Y_o - 2R_H) + \lambda_2 Y_o = 0 \tag{5.92}$$

$$\frac{\partial L}{\partial m} = \frac{2m_{e_b}}{SM}\left[e_r (C_r + C_x) + Z(C_x + C_b) + C_e\right] + Z(ZC_x + 2C_e) - c_o e_r^2 (C_x + C_r)$$
$$+ \frac{1}{3}\lambda_1 R_H^{2/3} Y_o \left(5Y_o - \frac{4m}{SM} R_H \right) + \lambda_2 Y_o^2 = 0 \tag{5.93}$$

$$\frac{\partial L}{\partial \lambda_1} = AR_h^{2/3} - \frac{nQ}{C_u \sqrt{S_o}} = 0 \tag{5.94}$$

$$\frac{\partial L}{\partial \lambda_2} = A - \frac{Q}{V_{max}} = 0 \tag{5.95}$$

Upon eliminating the Lagrange multipliers λ_1 et λ_2, the following system of nonlinear equations results:

$$AR_H^{2/3} - \frac{nQ}{C_u \sqrt{S_o}} = 0$$

$$A - \frac{Q}{V_{max}} = 0$$

$$AR_h^{2/3} - \frac{nQ}{C_u \sqrt{S_o}} = 0$$

$$A - \frac{Q}{V_{max}} = 0 \tag{5.96}$$

$$\frac{DCY \times (5Y_o - 2R_H) - DCB \times (5B - 4SMR_H)}{DCM \times (5Y_o - 2R_H) - DCB \times Y_o\left(5Y_o - \frac{4m}{SM}R_H\right)} - \frac{2Y_oSM - B}{Y_o^2\left(\frac{2m}{SM} - 1\right)} = 0$$

where

$$DCY = 2C_be_bSM + C_x(b + 2e_bSM + 2mZ) + 2mC_e$$

$$DCB = C_re_r + C_x(Z + e_r) + C_e$$

$$DCM = \frac{2me_b}{SM}\left[e_r(C_r + C_x) + Z(C_x + C_b) + C_e\right] + Z(ZC_x + 2C_e) - c_o + e_r^2(C_r + C_x)$$

The program given above must be modified to solve the nonlinear system of equations given by Equation 5.96. The first change concern the function FUN that evaluates the equations to be solved and the subroutine DIFF that evaluates the partial derivatives of the cost C with respect of the three variables, namely Y_o, b, and m, Manning's equation and the maximum velocity equation. The dimension of the vector Y must be increased to 11 since we need to specify a value for the maximum velocity. The EQUIVALENCE and COMMON/ALL/ need to be changed as shown below.

```
      FUNCTION FUN(X,I)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(3)
      COMMON /ALL/ Y(11),C0,C1,QNS
      GO TO(2,4,6),I
2     FUN=AR(X(1),X(2),X(3))*RH(X(1),X(2),X(3))**(2./3.)-QNS
      RETURN
4     FUN=AR(X(1),X(2),X(3))-C1
      RETURN
6     CALL DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)
      B=TOP(X(1),X(2),X(3))
      SM=SQRT(1.+X(3)**2)
      FUN=SM*(DCM*DFB-DCB*DFM)/(2.*X(3)-SM)-(DCY*DFB-DCB*DFY)*
     &X(1)**2/(2.*X(1)*SM-B)
      RETURN
      END
      SUBROUTINE DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(3)
      COMMON /ALL/ Y(11),C0,C1,C2,QNS
      EQUIVALENCE (Y(4),RV),(Y(6),EB),(Y(7),ER),(Y(8),CB),
     &(Y(9),CR),(Y(10),CX),(Y(11),CE)
      Z=X(1)+RV
      SM=SQRT(1.+X(3)*X(3))
      B1=X(2)+2.*SM*EB
      RH1=RH(X(1),X(2),X(3))
      DCY=2.*SM*CB*EB+CX*TOP(Z,B1,X(3))+2.*CE*X(3)
      DCB=CR*ER+CX*(Z+ER)+CE
```

```
      DCM=2.*EB*X(3)*(ER*(CR+CX)+Z*(CB+CX)+CE)/SM+Z*(Z*CX+2.*CE)-
     &C0*ER**2*(CR+CX)
      DFY=5.*TOP(X(1),X(2),X(3))-4.*RH1*SM
      DFB=5.*X(1)-2.*RH1
      DFM=X(1)*(5.*X(1)-4.*RH1*X(3)/SM)
      RETURN
      END
```

The constant c_1 is the wetted area or the ratio of the channel capacity by the maximum velocity and c_2 is the value of the wetted perimeter.

If the channel side slope is known, and the flow velocity is greater than the maximum allowable velocity, the optimal solution does not depend on the components of the channel cost. This is due to the fact when the value of the channel side slope is known, the two constraints, namely Manning's equation and the maximum velocity equation define the values of the uniform depth and the bottom width regardless of the unit costs C_b, C_r, C_e, C_x, and the free board.

The maximum velocity constraint might be written as

$$A = \frac{Q}{V_{max}} = C_1; \quad \text{or} \quad by_o + mY_o^2 = C_1 \tag{5.97}$$

To evaluate the wetted perimeter use Manning's equation

$$\frac{Qn}{C_u\sqrt{S_o}} = AR_h^{2/3} = \frac{C_1^{5/3}}{P^{2/3}}$$

Solving for P gives

$$P = \frac{C_1^{5/2}}{\left(\dfrac{Qn}{C_u\sqrt{S_o}}\right)^{3/2}} = b + 2Y_o\sqrt{1+m^2} = C_2 \tag{5.98}$$

The following two equations allow Y_0 and b to be solved:

$$\begin{aligned} bY_o + mY_o^2 &= C_1 \\ b + 2Y_o\sqrt{1+m^2} &= C_2 \end{aligned} \tag{5.99}$$

Or by combining these two equations, the bottom width can be eliminated, and Y_o solved from the quadratic equation:

$$Y_o^2\left(2\sqrt{1+m^2} - m\right) - C_2Y_o + C_1 = 0 \tag{5.100}$$

The discriminant of this equation is

$$\Delta^2 = c_2^2 - 4c_1\left(2\sqrt{1+m^2} - m\right) \tag{5.101}$$

A solution of this equation exists if Δ is positive, which requires that the following inequality holds:

$$\frac{c_2^2}{4c_1} \geq 2\sqrt{1+m^2} - m$$

The solution Equation 5.100 is as follows:

$$Y_o = \frac{C_2 \pm \sqrt{C_2^2 - 4C_1\left(2\sqrt{1+m^2} - m\right)}}{2\left(2\sqrt{1+m^2} - m\right)} \tag{5.102}$$

To find out for which value of the side slope m, the discriminant Δ is zero Δ^2 is set to zero.

$$\Delta^2 = c_2^2 - 4c_1\left(2\sqrt{1+m^2} - m\right) = 0$$

This equation might be written as

$$\left(\frac{C_2^2}{4C_1} + m\right)^2 - 4(1+m^2) = 0$$

After simplifying the following second order equation results:

$$3m^2 - \frac{2c_2^2}{4c_1}m - \frac{c_2^4}{16c_1^2} + 4 = 0 \tag{5.103}$$

The solution to this equation is

$$m = \frac{1}{3}\left(\frac{c_2^2}{4c_1} \pm 2\sqrt{\frac{c_2^4}{16c_1^2} - 3}\right) \tag{5.104}$$

The discriminant of this relation is zero if the following relation holds:

$$\frac{c_2^2}{4c_1} = \sqrt{3}$$

The corresponding value of the channel side slope is exactly equal to $1/\sqrt{3} \approx 0.5774$.

PROBLEMS

5.1 From the linear graph given by Jarrett for the Arkansas River (site 1), determine the relationship between Manning's n and the relative smoothness R_h/D_{50}. Using this relationship, what value for Manning's n should be used if the river stage is large so that R_h/D_{50} equals 6? For a smaller river stage that causes R_h/D_{50} to equal 2.0 what should n be taken as? (These n are to assume that the regular Manning's equation is to be used.) Using these values of n, accepting the constant 3.81 (for ES units) proposed by Jarrett in the modified Manning's equation, and assume $D_{50} = 0.4$ ft, determine the exponents of hydraulic radius R_h and the slope S_o that might apply for the Arkansas River for flows associated with R_h/D_{50} between these values.

5.2 Using Jarrett's proposed modification to Manning's equation for steep gradient mountain streams, determine what flow rate is likely occurring if the cross section of a mountain stream with a bottom slope of $S_o = 0.018$ can be approximated by a trapezoid with b = 4 ft and m = 1, and the depth of flow is Y = 3 ft. What is the average velocity of this flow? If a value of n = 0.03 is reasonable for the bed material in this stream how much energy is being dissipated per 1000 ft of stream length in addition to that from fluid friction? What power is associated with this loss?

5.3 The flow rate in a mountain stream is determined equal to 20 m^3/s. The slope of this stream is $S_o = 0.025$, and its cross section can be approximated by a trapezoid with b = 4 m and a side slope of m = 1.5. Using the modified Manning's equation proposed by Jarrett, determine the depth of flow. What is the corresponding average velocity of flow? Thereafter make up a table of values in which the flow rate varies from 10 to 30 m^3/s in increments of 2.5 m^3/s that gives (a) the corresponding depth based on Jarrett's equation and (b) the value of n that results in this same depth if Manning's equation were to be used.

5.4 Prove that when the ODE for GVFs is taken in the form $dE/dx = S_o - S_f$ that the terms involving the changing cross section $\partial A/\partial x$ and the term for lateral inflow or outflow that occur in the equation that define dY/dx are taken into account provided E is appropriately defined in a nonprismatic channel. Appropriately accounting for lateral inflow (outflow) requires that the flow rate be a function of x in the specific energy equation $E = Y + (Q/A)^2/(2g)$.

5.5 Define a trapezoidal channel with a bottom width of b = 5 m and m = 1, by using 6 pairs of x and z (elevation) as if it were an irregular section with a bottom elevation of 5 m. The six pairs of values are as follows: (0,10), (2.5,7.5), (5,5), (10,5), (12.5,7.5), (15,10). Using linear interpolation, obtain a table that gives A, P, and T related to Y using the method described in Section 5.3.1. Select depths midway between a couple of the entries in this table and obtain A, P, and T for these midpoint using linear interpolation. Compute the correct values from the equations that give these values as a function of depth. Why do you get identically the same values for P and T but not A as those obtain by interpolating at the midpoints?

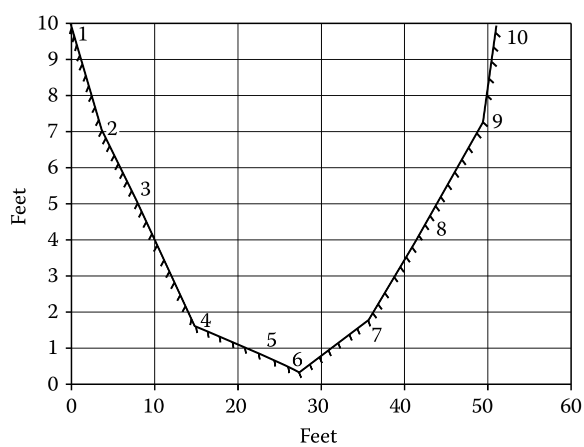
5.6 Treat a circular section with a 10 ft diameter as an irregular channel section. Starting with a depth of 9 ft, and then depths of 7.5, 5 2.5 0 2 3 5 6 8 and 9 again, compute the values of x_b, and z for the bottom of the circle at an elevation of 10 ft.

Answer	x_b	2	.67	0	.67	5	9	9.583	10	9.899	9	8
	Z	19	17.5	15	12.5	10	12	13	15	16	18	19

Using these pairs of values compute A, P, and T for 26 increments of Y between 0 and 9 by using both linear and quadratic interpolation, and compare the result with the exact values obtained from the appropriate algebraic equations. (You may try using a decrement of 0.125 ft, and repeat the above, and compare how much closer the interpolated values are to the actual for this smaller ΔY.)

5.7 Using hand computations compute: A, P, and T for 13 increments of depth for an irregular channel section whose x_p and z values are as follows:

x_p (ft)	0	3	7	13.5	20	24
z (ft)	508.7	502.3	499.8	500.7	503.6	508.7



5.8 The cross section of a natural channel is defined by the x z pairs of values given in the table below. The other table below was obtained from program CHANNEL using this data. If the depth of water in this channel is measured to be 8.4 ft, the Manning's roughness coefficient is n = 0.025, and the bottom slope is $S_o = 0.001$, what is the flow rate Q?

pt.	x (ft)	z (ft)
1	0	10
2	4	7
3	8	5.1
4	15	1.5
5	24	0.7
6	27.5	0.3
7	36	1.8
8	42.7	4.5
9	49.5	7.3
10	51.2	10.0

Depth	Top Width	Area	Perim.
0.00	0.00	0.00	0.00
0.49	7.45	2.41	7.52
0.97	15.53	7.98	15.65
1.46	23.09	17.35	23.29
1.94	28.76	29.92	29.05
2.43	33.08	44.92	33.48
2.91	36.04	61.68	36.60
3.40	37.65	79.55	38.49
3.88	37.91	97.87	39.88
4.37	37.17	116.08	42.11
4.85	36.39	133.92	45.42
5.34	38.58	152.10	47.81
5.82	40.77	171.34	50.21
6.31	42.98	191.65	52.62
6.79	45.16	213.02	55.01
7.28	46.88	235.34	56.98
7.76	48.20	258.40	58.62
8.25	49.29	282.04	60.09
8.73	50.16	306.15	61.42
9.22	50.79	330.63	62.63
9.70	51.20	355.37	63.79

5.9 Assume the above table in the previous problem is stored in arrays; Y(21) for the depth Y, T(21) for the top width T, A(21) for the area A, and P(21) for the wetted perimeter, P. Write the computer code that (a) finds the entries in the table that bracket a value for a specified depth YY and (b) then interpolates the values to give the top width, the area and the perimeter corresponding to this depth.

5.10 If the flow rate in the channel whose cross section is defined by the data in Problem 5.7 is Q = 350 cfs, determine the average velocity, velocity head, and specific energy for depths of 6 and 8 ft (for the same flow rate). What would the depth be if uniform flow occurred and n = 0.014 and $S_o = 0.0008$?

5.11 A natural channel's cross section is defined by the 9 pairs of (x, z) values in the table below. Use cubic splines to define this channel's cross section, and use some plotting program to plot the cross section. Using this spline definition of the channel bottom determine: (a) the flow rate, Q, (b) the kinetic energy correction coefficient, α, and (c) the momentum flux correction coefficient, β if the velocity distribution in the vertical direction at any position is given by $v(y) = 2.1Ln(2.2y + 1) - 0.12y$ and the depth is $Y = 12$ ft.

Point	1	2	3	4	5	6	7	8	9
x (ft)	3	7	10	16	24	34	43.5	49	54
z (ft)	16	10	4	1	0	2.5	10	16	18

5.12 The data below provide the cross section of a natural channel. Solve for the (1) the area, (2) the wetted perimeter, (3) the flow rate, (4) the average velocity, (5) the kinetic energy correction coefficient, α, and (6) the momentum correction coefficient, β if (a) the velocity varies with the position y as in the previous problem, i.e., $v(y) = 2.1Ln(2.2y + 1) - 0.12y$, and (b) the velocity varies according to $v(y) = 3.0Ln(3.2y + 1) - 0.25y$. The channel is full to its top, and $n = 0.025$.

Point	1	2	3	4	5	6	7	8	9	10
x (ft)	0.0	12.5	17.5	26.0	45.0	70.0	85.0	92.0	95.0	98.0
z (ft)	12.0	9.0	5.0	0.5	−0.2	0.5	2.5	5.0	8.0	12.0

5.13 Investigate the effects of the constants in the equation used in the previous problem that provide the velocity as a function of the depth, i.e., write this equation as: $v(y) = aLn(by + 1) - cy$ and then plot curves for v versus y for different values of a, b and c. For example, use Case 1: $a = 2.1$, $b = 2.2$, $c = 0.12$; Case 2: $a = 2.5$, $b = 2.2$, $c = 0.12$; Case 3: $a = 3.0$, $b = 2.2$, $c = 0.12$; Case 4: $a = 2.1$, $b = 2.2$, $c = 0.20$; Case 5: $a = 2.5$, $b = 2.2$, $c = 0.20$; Case 6: $a = 3.0$, $b = 2.2$, $c = 0.20$; Case 7: $a = 2.1$, $b = 2.2$, $c = 0.30$; Case 8: $a = 2.5$, $b = 2.2$, $c = 0.30$; Case 9: $a = 3.0$, $b = 2.2$, $c = 0.30$; Case 10: $a = 2.1$, $b = 2.2$, $c = 0.50$; Case 11: $a = 2.5$, $b = 2.2$, $c = 0.50$; Case 12: $a = 3.0$, $b = 2.2$, $c = 0.50$, etc.

5.14 The data below defines a divided channel consisting of an overflow left and right sides, and the main channel. The main channel begins with point 5 and ends with point 13. Assume the velocity distribution is given by the equation $v(y) = 2.5Ln(2.3y + 1) - 0.15y$ for all three portions of the channel and determine the following: The area of the flow, A, the flow rate, Q, the average velocity, V, the kinetic energy correction coefficient for all three portions of the channel as well as the composite value, α_l, α_m, α_r, and α, and the momentum correction coefficients for the three portions of the channel as well as the composite value, β_l, β_m, β_r, β, if the depth of flow is $Y = 13.3$ m.

Point	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
x(m)	0.0	0.5	4.0	36.0	43.0	50.0	600.	76.0	85.0	100	108	116	120.0	128.0	152.0	158.0
z(m)	16.0	15.0	14.0	13.7	13.0	9.0	5.0	2.6	2.2	4.0	7.0	12.0	13.5	13.8	13.9	15.5

5.15 Repeat the previous problem except that the velocity profile in the two side channels is given by $v(y) = 1.8Ln(2.0y + 1) - 0.1y$ and that in the main channel is still given by $v(y) = 2.5Ln(2.3y + 1) - 0.15y$.

5.16 Develop a computer program that could be used to replace program CHTABL to make a table giving the areas, wetted perimeters, and top widths as a function of depths but base the values obtained for these on cubic splines interpolation. This program would use splines interpolation much as program SPLINENA does, but would consider the elevation coordinate z the independent variable, and the x positions of the two banks as the dependent variable that will be interpolated for a number of increments of z (or the depth y) between the bottom of the natural channel to its top. The program should determine the top width as the difference between the x on the right and the left sides of the channel, and the wetted perimeter could be obtained by accumulating the changes in the two side lengths much the same as program CHTABL does. However, since a numerical integration routine such as SIMPR can easily be utilized with the spline interpolation, obtain the area by calling on a numerical integration routine rather than using the technique used in program CHTABL. In using cubic spline interpolation, obtain one array of 2nd derivatives at the points of the original data.

5.17 Using cubic splines to obtain a table relating A, P, and T to the depth Y as in the previous problem, with the exception that two cubic splines will be used; one for the left side of the channel, and the other for the right side of the channel. In other words, divide the x z coordinates that define the channel's cross-sectional geometry into two parts with the smallest elevation, or bottom of the channel separating these to arrays of pairs of data.

5.18 Use the program of the two previous problems to generate a table that relates A, P, and T to the depth Y for a channel in which the following pairs of values define the x and z (elevations) of the channel.

```
(0 108), (4 102), (8 100.2), (12 99.6), (16 99.5), (20 99.7),
  (24 100), (28 100.5), (32 101.4),
(34.2 103.4), (36 104.9), (40 105.3), (44 106.2), (45.5 108.0)
```

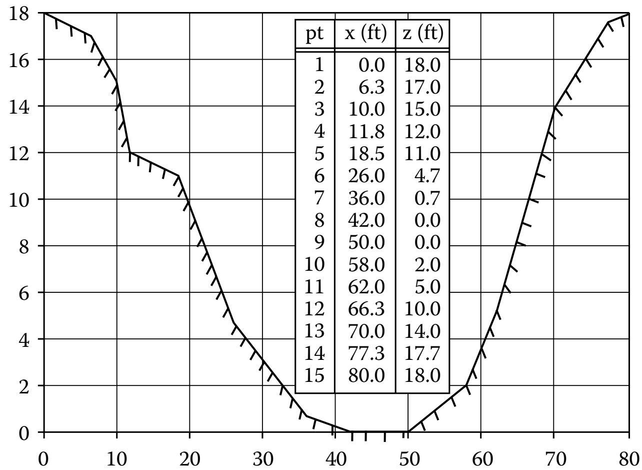
5.19 The cross section for a natural channel is defined by the x and z (elevation) data pairs given below. Solve the following problems using the CHANNEL program: (The known variables are given values and the unknown is followed by and = ?.)

Point	1	2	3	4	5	6	7	8	9	10	11	12
x (ft)	2.0	5.4	6.2	7.0	9.0	12.0	15.0	20.0	25.0	27.1	29.0	30.6
z (ft)	9.0	8.0	7.6	6.0	3.7	2.4	1.7	3.3	7.5	8.7	8.8	9.0

(a) Uniform flow: Y = 5 ft, n = 0.03, S_o = 0.001, Q = ?
(b) Uniform flow: Q = 200 cfs, n = 0.03, S_o = 0.001, Y = ?
(c) Uniform flow: Y = 5 ft, S_o = 0.001, Q = 200 cfs, n = ?
(d) Uniform flow: Y = 6 ft, Q = 200 cfs, n = 0.03, S_o = ?
(e) Specific energy: Q = 200 cfs, E = 5.5 ft, K = 0, Y = ?
(f) Specific energy: E = 5.5 ft, Y = 5.2 ft, K = 0, Q = ?
(g) Specific energy: Y = 5.5 ft, Q = 250 cfs, K = 0, E = ?
(h) Specific energy: Q = 200 cfs, E = 5.5 ft, K = 0, Y = ?
(i) Uniform + energy: n = 0.03, S_o = 0.001, H = 5.6 ft, K = 0, Y = ?, Q = ?

5.20 A natural channel cross section is defined by the x z coordinates given in the table below. If the flow rate is Q = 4000 cfs, and the Manning's roughness coefficient is n = 0.025, and the bottom slope is S_o = 0.001, what is the depth of flow in this channel?

pt.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x (ft)	0.0	6.3	10.0	11.8	18.5	26.0	36.0	42.0	50.0	58.0	62.0	66.3	70.0	77.3	80.0
z (ft)	18	17.0	15.0	12.0	11.0	4.7	0.7	0.0	0.0	2.0	5.0	10.0	14.0	17.7	18.0



5.21 For the natural channel of the previous problem, what is the uniform depth of flow Y_o if $Q = 4000$ cfs, Manning's $n = 0.025$, and the bottom slope is $S_o = 0.001$? If under uniform flow conditions the depth is 12 ft, what is the flow rate if $n = 0.024$ and $S_o = 0.001$?

5.22 For the same natural channel as in the previous two problems, obtain the variable that is followed by an =? based on the other variables whose values are given.

(a) Uniform flow: $Y = 16$ ft, $n = 0.020$, $S_o = 0.0008$, $Q = ?$
(b) Uniform flow: $Y = 10$ ft, $n = 0.020$, $S_o = 0.0008$, $Q = ?$
(c) Uniform flow: $Q = 3000$ cfs, $n = 0.020$, $S_o = 0.0008$, $Y = ?$
(d) Uniform flow: $Y = 17$ ft, $Q = 4000$, $S_o = 0.0008$, $n = ?$
(e) Uniform flow: $Y = 16$ ft, $Q = 3500$ cfs, $n = 0.020$, $S_o = ?$
(f) Specific energy: $Y = 16$ ft, $Q = 3500$ cfs, $K = 0.0$, $E = ?$
(g) Specific energy: $Y = 16$ ft, $E = 16.3$ ft, $K = 0.0$, $Q = ?$
(h) Uniform + energy: $H = 17.2$ ft, $K = 0$, $n = 0.02$, $S_o = 0.0008$, $Y = ?$, $Q = ?$
(i) Uniform + energy: $H = 13.0$ ft, $K = 0$, $n = 0.02$, $S_o = 0.0008$, $Y = ?$, $Q = ?$

5.23 Modify the program NATURAL so that it will also accommodate problems of the same type it now can handle but for which the channel may be either trapezoidal or circular in cross section in addition to a natural section.

Use your modified program to solve the following problems:

(a) Uniform flow: $Q = 50\,m^3/s$, $b = 3$ m, $m = 1.5$, $n = 0.014$, $S_o = 0.001$, $Y = ?$
(b) Specific energy: $Q = 50\,m^3/s$, $b = 3$ m, $m = 1.5$, $Y = 3$ m, $E = ?$
(c) Uniform + energy: $H = 1.5$ m, $K = 0$, $b = 3$ m, $m = 1.5$, $n = 0.014$, $S_o = 0.001$, $Y = ?$, $Q = ?$
(d) Specific energy: $b = 10$ ft, $m = 1.5$, $Q = 400$ cfs, $Y = 6$ ft, $E = ?$
(e) Momentum: $b = 10$ ft, $m = 1.5$, $Q = 400$ cfs, $Y = 6$ ft, $M = ?$
(f) Momentum: $b = 3$ m, $m = 1.5$, $Q = 70\,m^3/s$, $Y = 2$ m, $M = ?$
(g) Momentum: $D = 10$ m, $Q = 70\,m^3/s$, $Y = 2$ m, $M = ?$

5.24 Fill in the data that is missing for Section A in Table 5.2.

5.25 Fill in the data that is missing for Section E in Table 5.2.

5.26 Develop a spreadsheet that accomplishes a solution to the small river shown in Figure 5.2. For this solution, the flow rate is $Q = 500$ cfs, and Manning's $n = 0.032$. The downstream stage at Section A is 7.2 ft. This spreadsheet should duplicate the results shown in Table 5.3.

5.27 Develop a spreadsheet that accomplishes a solution to the small river shown in Figure 5.2 for a flow rate $Q = 400$ cfs, a Manning's $n = 0.028$, and a downstream stage of $h = 7.0$ ft.

5.28 Develop a spreadsheet that accomplishes a solution to the small river shown in Figure 5.2 for a flow rate Q = 500 cfs, and a downstream stage of h = 7.2 ft. The roughness coefficients of the channel are different at each section as given below.

Section	A	B	C	D	E	F
n	0.032	0.025	0.022	0.032	0.035	0.033

5.29 Using the computer program GVFNAT, duplicate the given solution to the GVF in the small river that is shown Figure 5.2.

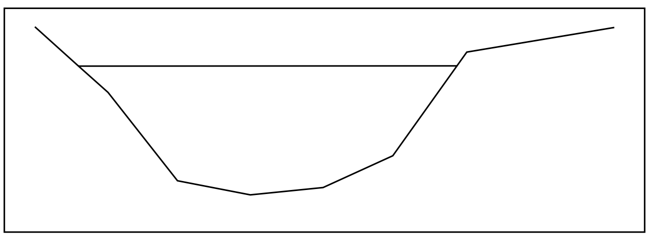
5.30 Using the computer program GVFNAT, obtain a solution for the GVF in the small river shown in Figure 5.2 for a flow rate of 400 cfs and a downstream depth of 7.0 ft.

5.31 Modify the computer program GVFNAT so that it will allow for the Manning's roughness coefficients to be different at each of the input sections along the river. Use linear interpolation for the roughness coefficient for position between the input sections.

5.32 Modify the spreadsheet solution so that it can accomplish a solution for a channel that contains (a) a main channel with a roughness coefficient n_1, a right side channel with a roughness coefficient n_2, and a left side channel with a roughness coefficient n_3. Decide how you wish to define the geometries of the three connected channels, and how the kinetic energy correction coefficient can be computed best.

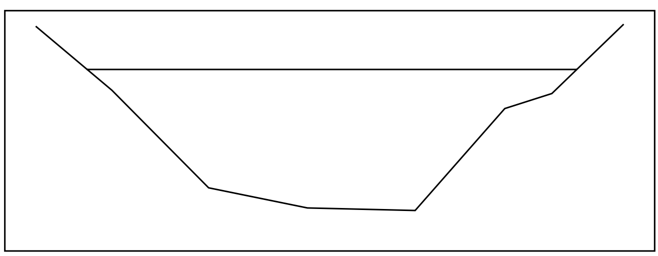
5.33 Below is data that defines the cross sections of a natural channel at stations: 8000, 6500, 4000, 2000, and 0 ft. The Manning's roughness coefficient for the entire channel is n = 0.015. For flow rates of (a) Q = 1500 cfs, (b) Q = 2000 cfs, and (c) Q = 2500 cfs determine the depths, etc. at positions spaced at 200 ft apart starting at station 1 at 8000 ft up to station 5 at 0 ft. The depth at the downstream end of the channel is as follows for the three above flow rates: Q = 1500 cfs, Y_1 = 10 ft, Q = 2000 cfs, Y_1 = 11 ft, Q = 2500 cfs, Y_1 = 12 ft.

x_p (ft)	z (ft)
0	200.0
5	195.0
10	188.0
15	187.0
20	187.5
25	190.0
30	200.0



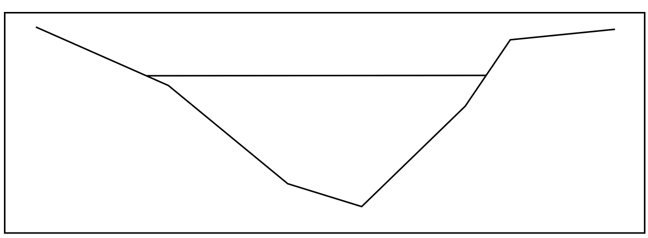
Station 1, x = 8000 ft

x_p (ft)	z (ft)
0	201.6
4	196.8
9.5	188.9
15	187.3
20.8	187.1
25.8	195.2
28.2	196.3
32	204.1



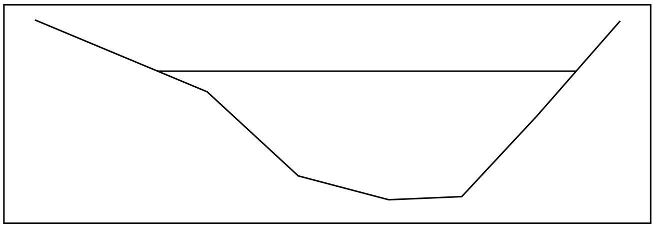
Station 2, at position 6500 ft

x_p (ft)	z (ft)
0	204.1
8	199.2
15.5	190.5
20	188.6
26.2	197.5
28.8	203.1
35	204.1



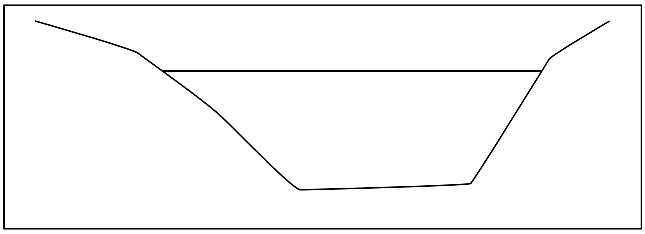
Section 3, at position 4000 ft

x_p (ft)	z (ft)
0	205.9
10.1	200.2
15.5	193.4
20.7	191.5
25	191.8
30	199.2
34	205.9



Section 4, position 2000 ft

x_p (ft)	z (ft)
0	207.6
5.8	205.2
9.5	201.6
15.4	194.0
25.6	194.5
30.2	204.7
33.5	207.6



Section 5, position 0 ft

5.34 For the natural channel in the previous problem obtain the solution for the flow rate of Q = 2000 cfs using (a) an increment of 100 ft, and (b) and increment of 400 ft. Compare these results with those obtained from the previous problem and make comments about the amount of difference between these solutions.

5.35 Solve Problem 5.33 except using a Manning's roughness coefficient n = 0.032.

5.36 The tables below provide the cross-sectional data for four stations along a stream at 1000 m intervals. Do the following: (a) Using cubic spline functions plot the four cross sections. (b) For the geometry of each section, solve for the uniform flow rate Q_o if the depth of water fills the channel to 80% and using the slope between the sections, and a Manning's n = 0.025. (c) Solve the gradually varied flow in this stream if the flow rate is Q = 90 m³/s and the depth at the downstream end of the channel, or at station 4000 m is Y_{end} = 3.8 m.

Station 3000 m			**Station 2000 m**			**Station 1000 m**			**Station 0 m**		
pt.	**X (m)**	**Z (m)**	**pt.**	**X (m)**	**Z (m)**	**pt.**	**X (m)**	**Z (m)**	**pt.**	**X (m)**	**Z (m)**
1	0.0	5.0	1	−1.0	6.5	1	0.0	8.0	1	0.0	9.0
2	1.7	3.8	2	3.5	5.0	2	4.5	6.7	2	3.0	8.3
3	3.5	2.7	3	5.0	4.2	3	7.0	5.0	3	4.7	7.5
4	7.5	2.5	4	7.5	3.5	4	9.0	3.6	4	5.0	6.5
5	9.5	1.6	5	9.7	2.7	5	10.1	3.0	5	6.5	5.5
6	11.5	1.3	6	12.5	2.0	6	14.5	2.8	6	8.0	5.0
7	13.5	0.8	7	17.0	2.6	7	15.5	3.5	7	12.0	4.6
8	16.0	1.25	8	20.0	3.7	8	15.9	5.0	8	17.0	4.65
9	18.0	2.1	9	23.0	4.5	9	17.0	5.5	9	19.0	5.2
10	20.5	2.7	10	25.5	6.2	10	20.0	6.0	10	20.0	6.5
11	22.5	4.0	11	28.0	6.5	11	23.0	6.5	11	22.0	8.2
12	24.2	5.0				12	25.1	6.8	12	23.5	9.0
						13	27.5	8.0			

S_o = 0.0015 to next station

5.37 The computer program GVFNAT solves the ODE $dE/dx = S_o - S_f$ by interpolating the area, A, and wetted perimeter, P, from two adjacent stations where cross-sectional data are given to define the geometric properties of the channel. After obtaining values for A and P, the value of S_f is computed in the subroutine DEX. Modify this program so the tables at the input sections contain only the following three columns of values: (1) the depth Y, (2) the corresponding slope of energy line, S_f, and (3) the corresponding specific energy, E. (Note that if the depth were not printed out as part of the solution it would not be necessary to have the column for the depth Y.)

5.38 The listing of the subroutine DEX from the program GVFNAT.FOR is given below. Modify the statements in this subroutine to solve a GVF problem in a natural channel if Equation 5.2, proposed by Jarrett, applies in defining Manning's roughness coefficient, i.e., $n = .39S_f^{.38}/R_h^{.16}$. Also indicate what statement you would change in the main program, and how it should read.

```
 SUBROUTINE DEX(X,SE,EPRIME)
 PARAMETER (N=26)
 REAL SE(1),EPRIME(1)
 LOGICAL*2 LINEAR
 COMMON /TRANS/ QN,FN,SO
 COMMON /INF/Y(N,2),T(N,2),A(N,2),P(N,2),E(N,2),XSTA1,XSTA2,
&ZSTA1,YY,ZSTA2,PSTA,ASTA,Q2G,LINEAR,IP1,IP2,IR,IR1,IL,IL1,
&IOTAB
 CALL INTERP(X,SE(1),.FALSE.)
 SF=QN*(ABS(PSTA/ASTA)**.666666667/ASTA)**2
 EPRIME(1)=SO-SF
 RETURN
 END
```

5.39 Develop a computer program that accepts as input data cross-sectional data that contains the values of x_b (the horizontal distance from a bank to a point on the section's bottom), and the elevation of that point, e.g., (x_b, z) pairs of data, and thereafter identifies the point numbers that separate the right, main, and left channels. From this input, it computes the areas, perimeters, and conveyances of the three separate channels, and from this data solves for the kinetic energy correction coefficient, α.

5.40 The additional headlosses caused by rapid changes in the cross section of channel are commonly not accounted for. Such conditions often exist at bridge piers, or other obstructions in a river. Modify your spreadsheet solution developed for Problem 5.26 to include such minor losses. These losses may be given directly by giving values to h_e, or computed from one of the minor head loss equations $h_e = KV^2/2g$, or $h_e = K (V_i - V_{i+1})^2/2g$.

5.41 Options available in HEC-2 allow for the average slope of the energy to be defined by any of the following three equations in addition to the average of the values at the ends of the reach over which the change in stage is being computed for, as was done in the spreadsheet solution:

Average conveyance

$S_f = (Q_i + Q_{i+1})/(K_i + K_{i+1})$ in which the K's are the conveyance values

Geometric mean friction slope

$$Sf = \{S_{fi} \times S_{fi+1}\}^{1/2}$$

or

Harmonic mean friction slope

$$S_f = \frac{\{2S_{fi} \times S_{fi+1}\}}{\{S_{fi} + S_{fi+1}\}}$$

Implement these average values for S_f in your spreadsheet solution to Problem 5.26 and compare any differences that result.

5.42 Explain why the program GVFNAT does not need to use any of the “average values” for the slope of the energy line.

5.43 Prepare the input data required for HEC-2 to solve the GVF profile in the small river of Figure 5.2, and obtain a solution.

5.44 Use the program GVFNAT, obtain a solution to the GVF-profile for the stream in the previous problem, i.e., the small river of Figure 5.2 (n = 0.032).

5.45 Using the cross-sectional data at the six stations from 10,000 ft of 0 ft for the compound natural channel that was used to illustrate the use of program GVFNATM obtain solutions for flow rates of Q = 9,000, 10,000, 11,000, 12,000 cfs. Use a downstream depth of Y_b = 28.0 ft for the three flow rates. What can you conclude from this solutions regarding the amount of water this channel can convey?

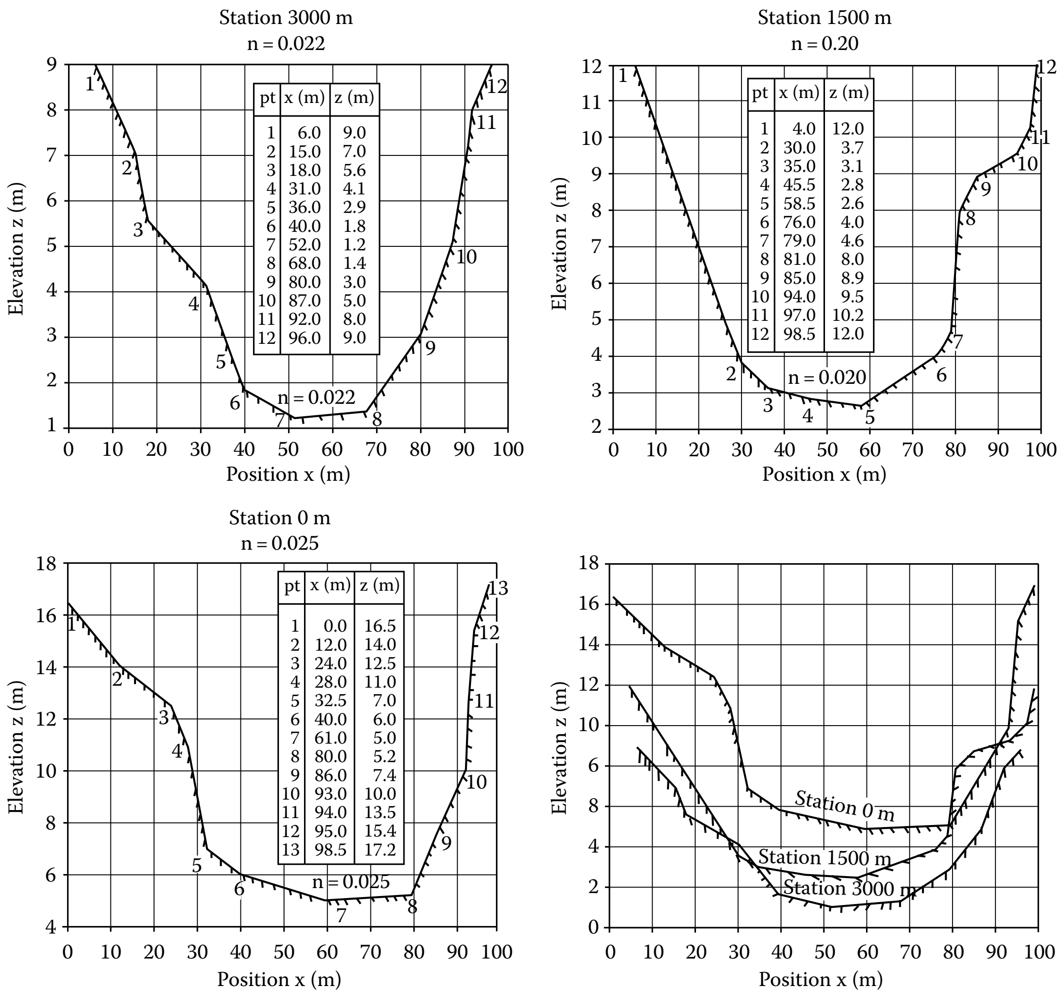
5.46 Survey data gives the following pairs of (x_b, z) at three sections at positions x = 5000, 2000, and 0 ft, respectively, for a small natural stream that discharges into a lake whose water surface elevation is 9 ft above the channel bottom here (where x = 5000 ft). If Manning’s n equals 0.028 for this stream, use the standard step method to compute the water level that would be expected at the other two stations for a flow rate Q = 400 cfs. (x_b and z are given in feet.)

Section # 1 x = 5000 ft		
pt.	x_b	z
1	0.0	150.0
2	4.0	146.0
3	12.0	141.0
4	22.0	139.0
5	32.0	140.8
6	40.0	145.0
7	48.0	150.0

Section # 2 x = 2000 ft		
pt.	x_b	z
1	0.0	155.1
2	0.7	149.1
3	4.0	144.1
4	10.0	141.8
5	16.0	143.0
6	24.0	145.1
7	32.0	145.5
8	40.0	147.1
9	47.0	149.6
10	49.0	152.3
11	50.0	155.1

Section # 3 x = 0 ft		
pt.	x_b	z
1	0.0	155.6
2	8.0	150.8
3	11.9	148.8
4	16.0	146.9
5	24.0	144.5
6	30.0	143.6
7	40.0	144.8
8	45.6	147.6
9	54.6	150.9
10	60.5	152.6
11	68.0	155.6

5.47 A river whose cross sections are defined at three stations at 3000, 1500, and 0 m discharges into the ocean that keeps its downstream depth at 7.8 m regardless of the flow rate. Notice that Manning’s n varies being 0.022 at the downstream station, 0.020 at the middle station, and equal to 0.025 at the upstream station. Obtain the gradually varied solutions if the flow rate is (a) Q = 1400 m^3/s, (b) Q = 2000 m^3/s, and (c) Q = 2500 m^3/s.



5.48 The cross-sectional data for a divided river is shown in the tables below. The judged positions that divide the left side from the main channel and the main channel from the right side of the channel are given in the second table as well as the Manning's n for these three portions of the river. For a flow rate of Q = 5300 cfs and a depth of Y_{beg} = 14 ft at station 5000, obtain the gradually varied profile up to station 0 m. Use cubic splines to show the shape of the six cross sections defined by the data in the first table below.

Table of x z pairs of coordinates that define cross sections of a divided river at six stations.

	Sta-5000 ft		Sta-4000 ft		Sta-3000 ft		Sta-2000 ft		Sta-1000 ft		Sta-0 ft	
Pt.	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**
1	0.0	20.0	0.0	21.5	0.0	22.0	0.0	24.0	0.0	24.0	0.0	27.0
2	12.0	18.0	10.4	17.5	10.0	18.0	18.0	20.0	20.0	21.0	20.0	25.5
3	26.0	14.0	24.5	12.5	20.0	15.5	36.0	16.0	25.0	20.0	40.0	23.0
4	40.0	12.5	34.0	12.0	28.0	14.5	40.0	13.5	32.0	17.0	46.0	21.0
5	52.0	10.0	38.0	10.0	40.0	13.5	44.0	8.0	40.0	13.8	50.0	18.0

(*continued*)

(continued)

	Sta-5000 ft		Sta-4000 ft		Sta-3000 ft		Sta-2000 ft		Sta-1000 ft		Sta-0 ft	
Pt.	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**	**x (ft)**	**z (f)**
6	56.0	8.0	43.0	6.0	48.0	13.0	56.0	3.8	53.5	12.0	52.0	13.0
7	58.0	4.0	46.0	2.7	54.0	10.0	68.0	3.5	60.0	8.5	56.0	8.0
8	64.0	0.7	52.0	1.9	60.0	5.0	80.0	5.0	65.0	5.0	60.0	6.5
9	72.0	0.0	60.0	1.8	66.0	2.0	94.0	10.0	72.0	4.0	68.0	5.8
10	80.0	0.5	68.0	1.0	74.0	2.1	98.5	14.8	80.0	4.0	80.0	6.5
11	90.0	4.0	80.0	3.5	80.0	4.0	104.0	16.8	91.0	6.0	88.0	8.0
12	100.0	10.0	81.0	5.5	94.5	8.0	120.0	18.5	95.0	10.0	92.0	10.0
13	112.0	15.0	83.0	10.0	96.0	10.0	136.0	20.0	98.0	12.0	96.0	16.0
14	128.0	18.0	86.0	13.0	98.5	14.0	144.0	23.5	104.0	12.5	98.0	19.0
15	140.0	20.0	96.0	15.0	108.0	16.0			113.0	14.0	104.0	20.5
16			108.0	17.0	120.0	17.0			120.0	17.5	120.0	23.0
17			116.0	18.5	132.0	19.5			134.0	20.0	138.0	28.0
18			127.0	19.0	136.0	22.0			142.0	24.0		
19			135.0	19.5								
20			139.0	21.5								

	Mannings n				
Station (ft)	**M**	**L**	**R**	**Left-Main (ft)**	**Main-Right (ft)**
5000	0.022	0.040	0.045	50.0	110.0
4000	0.020	0.038	0.040	50.0	90.0
3000	0.025	0.035	0.045	48.0	100.0
2000	0.018	0.050	0.050	36.0	104.0
1000	0.024	0.035	0.040	45.0	100.0
0	0.022	0.050	0.045	44.0	100.0

5.49 A canal consisting of the natural earth takes water from a much larger river by means of three 6-foot wide vertical gates at the canal's beginning. Cross-sectional data for this canal are given below. Computer solutions of the GVF profiles for flow rates of Q = 35, 70, 100, 150, 180 and 220 cfs are to be obtained. At station 1800 ft judgment suggests that the following are reasonable depths to specify for each of these flow rates:

Flow rate, Q (cfs)	35	70	100	150	180	220
Depth at 1800 ft, Y (ft)	1.5	1.8	2.0	2.3	2.5	3.0

Using an increment of 50 ft, determine the depths from position 1800 ft to the gates at position 0 ft for each of the above flow rates. For the last section at the gates assume the channel is rectangular and the bottom of the gates is at an elevation of 85.05 ft.

Cross-section data (all values in ft)

Station	x	z	x	z	x	z	x	z	x	z	x	z	x	z	x	z	x	z	x	z	x	z	n
1800	0	86.93	1	83.00	13	82.95	23	83.08	33	82.99	38	83.83	39	84.99	40	86.37	41	86.93					0.025
1600	0	86.09	0.1	85.65	0.2	84.89	0.6	83.41	5	83.1	10	82.89	15	82.95	20	83.1	21.5	83.04	24	86.09			0.025
1400	3	86.53	7	84.96	11	83.46	15	83.27	20	83.17	25	83.38	30	84.09	31.5	84.94	32	86.35	35	87.09			0.025
1200	0	87.52	4.5	85.41	10	84.29	15	84.24	20	84.0	25	83.54	30	84.03	30.2	84.45	32.8	86.35	34.3	87.44			0.028
1000	0	88.10	2.5	87.10	3.5	86.19	7	85.51	10	85.37	15	84.0	20	85.23	25	85.77	26	87.09	30	88.89			0.028
800	0	87.26	2.5	85.72	3	84.75	7	84.5	11	84.26	15	84.22	20	84.22	22	84.21	22.5	85.67	27	87.57			0.028
600	0	87.87	3	87.18	3.5	86.04	6	85.91	10	85.69	15	85.6	20	85.61	25	86.22	26.5	87.11	27	87.93	30	88.59	0.028
400	0	87.71	3.4	86.36	3.5	85.16	8	84.9	13	84.55	18	84.56	23	84.98	24.3	86.34	27	87.69					0.028
200	0	87.99	2	86.64	2.5	85.25	5	84.35	10	84.44	15	84.67	18.8	85.29	19.5	86.66	23.1	88.74					0.028
20	0	88.26	2	86.91	2.5	85.52	5	84.62	10	84.71	15	84.94	18.8	85.56	19.5	86.93	23.1	89.01					0.028

A flume is to be place at position 250 ft downstream from the gates. What size flume should be used? (You can select either a Parshall or Cutthroat flume.) The flow rates will rarely exceed 100 cfs and for flow rates up to 220 cfs, their duration will be relatively short, so the accuracy of these measurements need not be as precise as for flow rates less than 100 cfs. Determine the bottom elevation that this flume should be placed at.

5.50 The following data were collected from current meter measurement at a river cross section. Determine the flow rate.

Section No.	1	2	3	4	5	6
Position (ft)	1.2	3.5	6.5	9.5	12.5	14.0
Depth (ft)	2.3	5.8	6.4	5.6	2.1	1.1
Vel., $V_{.2Y}$ (fps)	4.1	4.6	4.8	4.5	4.0	2.1
Vel, $V_{.8Y}$ (fps)	3.8	4.4	4.5	4.2	3.5	1.4

5.51 The following data were collected from current meter measurement at a river cross section. Determine the flow rate.

Section No.	1	2	3	4	5	6
Position (m)	1.0	3.0	6.0	9.0	12.0	14.1
Depth (m)	1.3	1.8	1.4	1.6	1.1	0.7
Vel., $V_{.2Y}$ (m/s)	1.1	1.7	1.9	1.7	1.0	0.9
Vel, $V_{.8Y}$ (m/s)	0.0	1.4	1.5	1.3	0.8	0.6

5.52 Current meter data obtained from a section in an earthen canal in which the depth and velocity at 0.6 times this depth were measured with a Marsh–McBirney current meter are given below. What is the flow rate in the canal?

Position (ft)	3.5	4.0	4.5	5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0
Depth (ft)	0	0.2	0.3	0.4	0.5	0.7	0.8	0.8	1.1	1.1	1.1	1.2
Velocity (fps)	0	0.35	1.00	1.28	1.47	1.62	1.70	1.76	1.93	1.62	1.70	1.93
	9.5	10.0	10.5	11.0	11.5	12.0	12.5	13.0	13.5	14.0	14.5	15.0
	1.4	1.4	1.4	1.5	1.4	1.4	1.4	1.4	1.5	1.3	1.3	1.2
	2.03	2.08	2.16	2.22	2.09	2.02	2.27	2.31	2.12	1.92	2.11	2.12
	15.5	16.0	16.5	17.0	17.5	18.0	18.5	19.0	19.5	20.0	20.5	
	1.2	1.2	1.1	1.1	1.1	0.9	0.9	0.7	0.5	0.3	0.2	
	1.89	2.12	1.27	1.92	1.63	1.74	1.53	1.09	1.21	0.90	0.62	

5.53 What head, H, would you expect to exists on a rectangular sharp crested suppressed weir that is 4 m long, when a flow rate of $Q = 0.5\,m^3/s$ passes over it?

5.54 A flow rate from 0.5 to 2.0 cfs is to be measured as it flows from a large tank. Size the appropriate sharp crested weir.

5.55 A 90° V-notched weir is used to measure water flowing from a reservoir into a ditch. The head over the bottom of the weir is 0.42 m. Determine the flow rate, Q.

5.56 What length of Cipoletti weir should be used if it is to measure flow rates to $Q = 0.75\,m^3/s$.

5.57 The upstream stilling well in a Parshall flume with a 6 in. throat width is 4.5 in. What is the flow rate?

5.58 A Cutthroat flume is to be used to measure a flow rate of up to $1\,m^3/s$ in a canal that is 3 m wide. Select the proper size flume, and indicate what the depth of water will be in the upstream stilling well when the capacity flow rate of $Q = 1\,m^3/s$ occurs.

5.59 Design the transition for a channel with a Manning's roughness coefficient of n = 0.013 for a flow rate of Q = 500 cfs if the upstream channel is trapezoidal with $b_1 = 20$ ft, $m_1 = 1.5$, and $S_{o1} = 0.0001$, and the downstream channel is rectangular with $b_2 = 16$ ft, and $S_{o2} = 0.001$.

5.60 Write the algebraic equations for the two reversed parabolas that define the change in the bottom width b through the transition in the previous problem. Provide an expression for T as a function of x assuming that the depth of flow varies linearly between the beginning of the transition to its end in the previous problem.

5.61 Design a transition that changes a rectangular channel with a bottom width of $b_1 = 3$ m to a trapezoidal channel with a bottom width of $b_2 = 5$ m and a side slope $m_2 = 1.5$ for a flow rate of $Q = 30\,m^3/s$. Both channels have a Manning's roughness coefficient n = 0.012, and the upstream channel has a bottom slope $S_{o1} = 0.0014$, and the downstream channel has a bottom slope $S_{o2} = 0.00045$.

5.62 Using the computer program whose listing is given for solving subcritical transitions to obtain a transition design between an upstream channel with $b_1 = 4$ m, $m_1 = 0.5$, $S_{o1} = 0.0009$ and $n_1 = 0.014$ and a downstream channel with $b_2 = 6$ m, $m_2 = 1.5$, $S_{o2} = 0.00025$ and $n_2 = 0.0135$. The channel is designed for a flow rate that will have a normal depth of $Y_{o1} = 2.4$ m in the upstream channel.

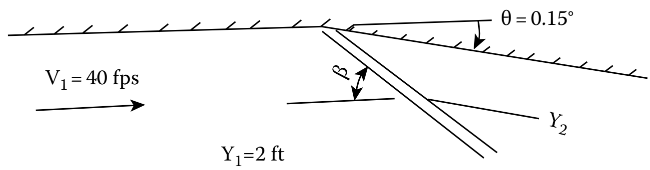
5.63 Program TRANSIT, which is designed to design the shape of a subcritical transition, requires the depths in the upstream and downstream channels, Y_{o1} and Y_{o2}, be provided as input data. Replace Y_{o1} and Y_{o2} with n_1, S_{o1}, n_2, and S_{o2} as input data. Also the head loss HL is given as input in program TRANSIT. Replace this input with a local loss coefficient K_L in the equation $H_L = K_L\left|V_1^2 - V_2^2\right|/(2g)$. With this modified program solve Example Problem 5.2 with $n_1 = n_2 = 0.015$ and $S_{o1} = 0.002663$ and $S_{o2} = 0.000077$ and $K_L = 0.212$. With this modified program design the transition between the two prismatic channels below that is to be 100 ft long and carry a flow rate Q = 900 cfs? The local loss coefficient is $K_L = 0.20$.

Channel	b (ft)	m	S_o	n
Upstream	10	1	0.0012	0.013
Downstr.	15	1.5	0.00015	0.013

5.64 For the second part of the previous problem, determine how the depth at the beginning of the transition varies from the normal depth in the upstream channel as the flow rate varies from the design flow rate of 900 cfs. Start with a flow rate of 100 cfs and end with a flow rate of 1600 cfs in this solution table in increments of 50 cfs. For each such flow rate, also determine how far upstream the M_1 or M_2 GVF exists upstream to where the depth becomes within 1% of the normal depth. (For each Q, assume the downstream channel is long so that normal conditions exist in it.)

5.65 Write a computer program that uses the 11 iterative steps outlined in the text to solve for the length of a supercritical transition for rectangular channels. Then use this program to solve the supercritical transition of Example Problem 5.4. Manning's n for both upstream and downstream channels is 0.013.

5.66 The sides of a converging rectangular channel deflect inward with an angle $\theta = 15°$. Upstream the velocity and depth are $V_1 = 40$ fps, and $Y_1 = 2.00$ ft, respectively. Determine the angle β of the oblique wave and the depth Y_2 immediate downstream from this wave.



5.67 A supercritical flow occurs along a wall that abruptly deflect inward toward the flow with an angle of 20°. The depths upstream and downstream from the oblique wave are observed to be 2.00 and 4.80 ft, respectively. Determine what the flow rate per unit width, q, is in this channel flow.

5.68 The inward angle of deflection θ for a wall that retains an open channel flow is 15°. The upstream depths and velocity are $Y_1 = 1$ m and $V_1 = 6.5$ m/s. Determine the angle of the oblique wave that will propagate from the point on the wall where this deflection occurs. What is the depth immediately downstream from the oblique wave?

5.69 Make a plot that shows the relationship between the three variables that appear in Equation 5.23, namely β, θ, and F_{r1}.

5.70 Derive Equations 5.22 through 5.24 that apply across the oblique wave that is formed as the waves reflect each other across the centerline of a rectangular channel. These equations will involve the angle β'.

5.71 Write a computer program that designs supercritical contracting transitions in rectangular channels. As input this program should read the following variables: Q, b_1, S_{o1}, n_1, b_2, S_{o2}, and n_2.

5.72 Modify the program in the previous problem so that it solves for the length of the supercritical transition for a range of flow rates. In place of the input Q, this program should read beginning and ending flow rates and the number of entries that should be included in the solution table, i.e., in place of each Q it should read Q_1, Q_2, and N.

5.73 Design a supercritical transition between the following two rectangular channels: Upstream the following occur: $Y_1 = 1$ m, $F_{r1} = 3.5$, $b_1 = 4$ m. Downstream the following occur: $Y_3 = 1.8$ m, $b_2 = 2.5$ m.

5.74 An abrupt outward wall deflection of 20° occurs in a supercritical flow, which causes a gradual drop in the water surface from 4.80 to 2.00 ft. Determine the flow rate per unit width in the upstream channel, q_1 if angle $\beta_1 = 45°$. What is the angle subtending through the region at this deflection point through which the changes in water depth occurs.

5.75 It is necessary to design a rectangular channel enlargement from an upstream width of b_1 = 10 m to a downstream width of b_2 = 25 m. Upstream the velocity and depth are V_1 = 17.5 m/s and 1.5 m, respectively. Determine the length of transition and its width every 10 m using the shape recommended by Rouse.

5.76 Make up a table of values for the dimensionless depths y_2 and y_3 that represent the limiting values for which the solution to the dimensionless depth y no longer exists for submerged flow behind a gate (i.e., when the argument for the square root of Equation 5.51 becomes negative, and gives the corresponding values for the dimensionless depth y).

5.77 The depth upstream from a gate that has submerged flow at its downstream end is Y_1 = 3 m. If the gate has a contraction coefficient of C_c = 0.65 is set at Y_G = 0.5 m above the channel bottom, and the downstream depth in the downstream channel is Y_3 = 2.5 m determine the submergence depth immediate downstream from the gate. What is the flow rate per unit width passing the gate? (Solve this problem both using the equations that contain actual depths and secondly using the equations with dimensionless depths and verify that the answers are the same. Also use appropriate graphs to verify the algebraic solutions.)

5.78 Write a computer program to generate the valves for the dimensionless depth y that are in the table insert in Figure 5.4.

5.79 Rework Example Problem 5.6 with the bottom slope of the channel equal to S_o = 0.0008 instead of = 0.000518.

5.80 Write a computer program or use a software package that will generate a table that provides the limiting depth data that could be used to plot Figure 5.7. The data generated should provide the following values corresponding to the dimensionless depth $y_2 = (C_cY_G/Y_1)$ caused by the gate that will first result in a gate becoming submerged: (1) the limiting dimensionless downstream depth $(y_3)_{limit}$, (2) the limiting value of the submerged depth y_{limit}, (3) the limiting upstream Froude number (F_{r1}), and the limiting dimensionless energy loss caused by the submergences, Δe_{limit}.

5.81 Make a table of values that gives values for the loss of specific energy in dimensionless form($\Delta e = (E_1 - E_3)/Y_1$ as a function of y_2 and y_3 for submerged flow past a gate.

5.82 A sluice gate has a contraction coefficient of C_c = 0.6 and its position above the channel bottom is Y_G = 0.5 m. If the depth upstream from the gate is Y_1 = 3 m, determine what downstream channel depth will be limiting between free and submerged flow by the gate. What is the corresponding flow rate past the gate per unit width of gate? If the depth is increased by 0.1 m above this limiting value, and the upstream depth does not change determine the submergence depth immediately downstream from the gate, and the corresponding flow rate.

5.83 If the flow rate per unit width in a 10 ft wide rectangular channel is q = 9.32 cfs/ft (when the gate is free flowing), the depth upstream from the gate is Y_1 = 4 ft, and the height of the gate times its contraction coefficient is C_cY_G = 0.8 ft, what is the maximum downstream depth allowed so that the flow past the gate is free flowing? If the downstream channel is very long and has a Manning's roughness coefficient n = 0.015 and a bottom slope S_o = 0.0004, estimate the depth of submergence on the gate (do this under the assumption that Y_1 does not change). How much energy is dissipated under this latter submerged flow condition?

5.84 A vertical gate that has a contraction coefficient of C_c = 0.6 is set 1.5 ft above the bottom in a rectangular channel. If the depth upstream from the gate is 5 ft and the flow rate per unit width is q = 15 cfs/ft, determine (a) the downstream channel depth Y_3 that limits the conditions downstream from the gate between free flow and submerged flow. (b) If the downstream depth Y_3 is increased 0.5 ft from this limiting value, then determine: (1) the submergence depth Y, (2) the flow rate q, and (3) the amount of specific energy per unit width of channel that will be dissipated.

5.85 Example Problem 5.8, assume that 2000 ft downstream from the gate the channel discharges into a reservoir whose water surface is 4 ft above the channel bottom. This channel is rectangular with a bottom width b = 8 ft has a bottom slope S_o = 0.0008 and a Manning's roughness coefficient n = 0.0135. Determine the conditions at the gate (whether the gate is submerged or free flowing with a hydraulic jump downstream, and if so where) the depths downstream from the gate and the flow rate past the gate assuming that the depth upstream of the gate remains at $Y_1 = 5$ ft. (Let Y_2 vary from 0.75 to 1.5 ft.)

5.86 A gate controls the flow into a 10 ft wide rectangular channel that has a bottom slope of $S_o = 0.0008$ and $n = 0.013$. The gate receives the water directly from a reservoir whose depth is 5 ft above the channel bottom. Make up a table that shows steady-state flow conditions for the gates positions so it produces $y_2 = Y_2/Y_1$ varying from 0.1 to 0.3 starting at the lowest position and then repeat these computations except when the gate is being lowered instead of being raised. Identify when the gate becomes submerged/free flowing.

Solution Problem 5.86

y_2	F_{r1}	q (cfs/ft)	Q (cfs)	Y_3 (ft)	Y_2 (ft)	Y_{2d} (ft)	Y (ft)	y	x-jump (ft)
0.100	0.500	0.1348	8.555	85.55	2.058	0.779			75.0
0.120	0.600	0.1604	10.173	101.73	2.317	0.871			74.2
0.140	0.700	0.1854	11.765	117.65	2.561	0.955			70.7
0.160	0.800	0.2101	13.329	133.29	2.794	1.032			65.0
0.180	0.900	0.2343	14.867	148.67	3.017	1.104			57.5
0.200	1.000	0.2582	16.381	163.81	3.232	1.171			48.4
0.220	1.100	0.2817	17.871	178.71	3.439	1.234			37.9
0.240	1.200	0.3048	19.337	193.37	3.640	1.293			26.4
0.260	1.300	0.3276	20.782	207.82	3.835	1.349			13.9
0.270	1.350	0.3388	21.496	214.96	3.930	1.376			7.4
0.280	1.400	0.3500	22.205	222.05	4.024	1.402			0.7
0.290	1.450	0.3611	22.909	229.09	4.117	1.428	Submerged		0.0
0.290	1.450	0.3291	20.881	208.81	3.848		1.780	0.356	
0.300	1.500	0.3323	21.083	210.83	3.875		1.933	0.387	
0.310	1.550	0.3357	21.297	212.97	3.903		2.069	0.415	
0.320	1.600	0.3391	21.514	215.14	3.932		2.192	0.448	
0.210	1.050	0.2838	18.007	180.07	3.462		0.517	0.127	
0.210	1.050	0.2700	17.129	171.29	3.336	1.203	Free flowing		
0.290	1.450	0.3611	22.909	229.09	4.117	1.428			0.0
0.280	1.400	0.3500	22.205	222.05	4.024	1.402			0.7
0.270	1.350	0.3388	21.496	214.96	3.930	1.376			7.4
0.260	1.300	0.3276	20.782	207.82	3.835	1.349			13.9
0.250	1.250	0.3162	20.062	200.62	3.738	1.322			20.3
0.240	1.200	0.3048	19.337	193.37	3.640	1.293			26.4
0.220	1.100	0.2817	17.871	178.71	3.439	1.234			37.9
0.200	1.000	0.2582	16.381	163.81	3.232	1.171			48.4
0.150	0.750	0.1978	12.550	125.50	2.679	0.994			68.1
0.100	0.500	0.1348	8.555	85.55	2.058	0.779			75.0

Procedure-free flow conditions:

1. Write down columns 1 and 2 from insert in Figure 5.7
2. Compute q and Q = bq
3. Solve Manning's equation for normal depth Y_3
4. Solve for conjugate depth Y_2 to normal depth Y_3.

5.87 Repeat the previous problem except rather than having the gate immediately downstream from the reservoir with a constant head of 5 ft, the gates exists at a position L = 1500 ft downstream from this constant head reservoir. The channel upstream from the gate has b = 10 ft, $S_o = 0.0008$, and $n = 0.013$, also. The entrance loss for the channel at the supply reservoir is $K_e = 0.08$.

Procedure-submerged flow conditions:

1. Based on y_2 (Y_2), solve Equations 5.40, 5.43, and Manning's equation (or 5.34, 5.35 and Manning's equation) simultaneously for Y_3, Y, and Q.
2. Solve for F_{r1}.

Solution Problem 5.82 (b_1 = 10′, b_2 = 10′, n = 0.013, S_o = 0.0008)

y_2	Y_2 (ft)	Y_1 (ft)	F_{r1}	q (cfs/ft)	Q (cfs)	Y_{beg} (ft)	Y_3 (ft)	Y_{2d} (ft)	Y (subm) (ft)
0.10	0.600	6.001	0.1348	11.247	112.469	4.912	2.482	0.928	
0.12	0.711	5.922	0.1604	13.112	131.122	4.879	2.762	1.022	
0.14	0.817	5.833	0.1854	14.825	148.250	4.843	3.011	1.102	
0.16	0.918	5.737	0.2101	16.383	163.832	4.805	3.232	1.171	
0.18	1.014	5.635	0.2343	17.788	177.884	4.766	3.428	1.231	
0.20	1.106	5.528	0.2582	19.045	190.448	4.728	3.600	1.282	
0.22	1.192	5.418	0.2817	20.159	201.593	4.690	3.751	1.326	
0.24	1.273	5.306	0.3048	21.141	211.407	4.654	3.883	1.363	
0.26	1.350	5.193	0.3276	21.999	219.990	4.620	3.997	1.395	
0.27	1.387	5.137	0.3388	22.385	223.854	4.603	4.048	1.409	
0.28	1.423	5.081	0.3500	22.745	227.451	4.588	4.095	1.422	Submergence
0.28	1.502	5.365	0.2928	20.646	206.465	4.673	3.817		begins
0.29	1.549	5.342	0.2975	20.843	208.432	4.665	3.843		1.969
0.30	1.596	5.319	0.3023	21.039	210.389	4.658	3.869		2.091
0.32	1.687	5.271	0.3119	21.418	214.179	4.643	3.920		2.201
0.34	1.776	5.225	0.3213	21.773	217.735	4.629	3.967		2.396
0.32	1.687	5.271	0.3119	21.418	214.179	4.643	3.920		2.566
0.30	1.596	5.319	0.3023	21.039	210.389	4.658	3.869		2.396
0.28	1.502	5.365	0.2928	20.646	206.465	4.673	3.817		2.201
0.26	1.406	5.406	0.2843	20.275	202.750	4.686	3.767		1.969
0.24	1.301	5.422	0.2810	20.128	201.275	4.691	3.747		1.673
0.24	1.273	5.306	0.3048	21.141	211.407	4.654	3.883	1.363	1.190
0.22	1.192	5.418	0.2817	20.159	201.593	4.690	3.751	1.326	Submergence
0.20	1.106	5.528	0.2582	19.045	190.448	4.728	3.600	1.282	ends
0.18	1.014	5.635	0.2343	17.788	177.884	4.766	3.428	1.231	
0.16	0.918	5.737	0.2101	16.383	163.832	4.805	3.232	1.171	
0.14	0.817	5.833	0.1854	14.825	148.250	4.843	3.011	1.102	
0.12	0.711	5.922	0.1604	13.112	131.122	4.879	2.762	1.022	
0.10	0.600	6.001	0.1348	11.247	112.469	4.912	2.482	0.928	

FORTRAN (SOLGVFSU.FOR) listing to solve above problem

```
C Solves submerged flow by gate downstream from reservoir.
      LOGICAL*2 SUBMER
      INTEGER*2 INDX(5)
      REAL F(5),D(5,5),X(5),KL2,KE1,KL,KE
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
      COMMON /TRAS/B1,B2,H,G,G2,KL2,KE1,FL,TOL,FN,SO,CC,QN,
     &Q2G,X,yG2,SUBMER
      EQUIVALENCE (Q,X(1)),(Ybeg,X(2)),(Y1,X(3)),(Y3,X(4)),
     &(Y,X(5))
      WRITE(*,*)' GIVE:IOUT,TOL,ERR,FN,SO,B1,B2,y2,H,L,g,KL,KE'
      READ(*,*) IOUT,TOL,ERR,FN,SO,B1,B2,yG2,H,FL,G,KL,KE
C  yG2 is the dimensionless depth of flow under the gate
C    y2=Y2/Y1
      SUBMER=.FALSE.
```

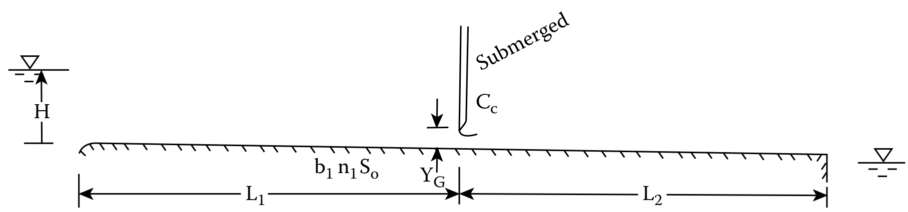
```
      N=4
      IF(G.GT.30.) THEN
      CC=1.486
      ELSE
      CC=1.
      ENDIF
      G2=2.*G
      KL2=1.+KL2
      KE1=1.+KE
      WRITE(*,*)' GIVE guess for: Q,Ybeg,Y1,Y3,Y'
      READ(*,*) X
2     NCT=0
1     DO 10 I=1,N
      F(I)=FUN(I)
      DO 10 J=1,N
      DX=.005*X(J)
      X(J)=X(J)+DX
      D(I,J)=(FUN(I)-F(I))/DX
10    X(J)=X(J)-DX
      CALL SOLVEQ(N,5,D,F,1,DD,INDX)
      SUM=0.
      DO 20 I=1,N
      X(I)=X(I)-F(I)
20    SUM=SUM+ABS(F(I))
      NCT=NCT+1
      WRITE(*,*)' NCT=',NCT,SUM
      WRITE(*,*) (X(I),I=1,N)
      IF(NCT.LT.30 .AND. SUM.GT. ERR) GO TO 1
      Fr1=Q/(B1*Y1)/SQRT(G*Y1)
      Y2=yG2*Y1
      IF(.NOT.SUBMER) Y=.5*Y3*(SQRT(1.+8.*(Q/(B2*Y3))**2/
     &(G*Y3))-1.)
      WRITE(IOUT,100) yG2,Y2,Y1,Fr1,Q/B1,Q,Ybeg,Y3,Y
100   FORMAT(F5.2,2F8.3,F8.4,5F8.3)
      IF(Y.LT.Y2 .AND. .NOT.SUBMER) THEN
      SUBMER=.TRUE.
      N=5
      ELSE IF(Y.LT.Y2 .AND. SUBMER) THEN
      SUBMER=.FALSE.
      N=4
      ENDIF
      WRITE(*,110) Y2,Q,Ybeg,Y1,Y3,Y,SUBMER
110   FORMAT(F10.3,F10.2,4F8.3,L8,/,' Give new y2 ')
      READ(*,*) yG2
      IF(yG2.GT. 0.) GO TO 2
      END
      FUNCTION FUN(II)
      EXTERNAL DYX
      LOGICAL*2 SUBMER
      REAL X(5),W(1,13),KL2,KE1,Y(1),DY(1),XP(1),YP(1,1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE
```

```
      COMMON /TRAS/B1,B2,H,G,G2,KL2,KE1,FL,TOL,FN,SO,
     &CC,QN,Q2G,X,yG2,SUBMER
      Y2=yG2*X(3)
      H1=-.05
      HMIN=.001
      Q2G=X(1)*X(1)/G2
      Q2S1=Q2G/B1**2
      Q2S2=Q2G/B2**2
      IF(SUBMER) THEN
      GO TO (1,2,3,3,5),II
      ELSE
      GO TO (1,2,3,5),II
      ENDIF
1     AA=B2*X(4)
      FUN=FN*X(1)-CC*AA*(AA/(B2+2.*X(4)))**.6666667*SQRT(SO)
      RETURN
2     FUN=H-X(2)-KE1*Q2S1/X(2)**2
      RETURN
3     IF(SUBMER) THEN
      IF(II.EQ.3) THEN
      FUN=X(3)+Q2S1/X(3)**2-X(5)-Q2S2/Y2**2
      ELSE
      FUN=2.*Q2S2-.5*(X(4)**2-X(5)**2)*Y2*X(4)/(X(4)-Y2)
      ENDIF
      ELSE
      FUN=X(3)+Q2S1/X(3)**2-Y2-Q2S2/Y2**2
      ENDIF
      RETURN
5     Y(1)=X(3)
      XX=FL
      XZ=0.
      IND=1
      QN=(FN*X(1)/CC)**2
      Q2G=X(1)*X(1)/G
      CALL ODESOL(Y,DY,1,XX,XZ,TOL,H1,HMIN,1,XP,YP,W,DYX)
      FUN=X(2)-Y(1)
      RETURN
      END
      SUBROUTINE DYX(XX,Y,DY)
      REAL Y(1),DY(1),KL2,KE1,X(5)
      LOGICAL*2 SUBMER
      COMMON/TRAS/B1,B2,H,G,G2,KL2,KE1,FL,TOL,
     &FN,SO,CC,QN,Q2G,X,yG2,SUBMER
      YY=ABS(Y(1))
      P=B1+2.*YY
      A=B1*YY
      SF=QN*((P/A)**.666666667/A)**2
      DY(1)=(SO-SF)/(1.-Q2G*B1/A**3)
      RETURN
      END
```

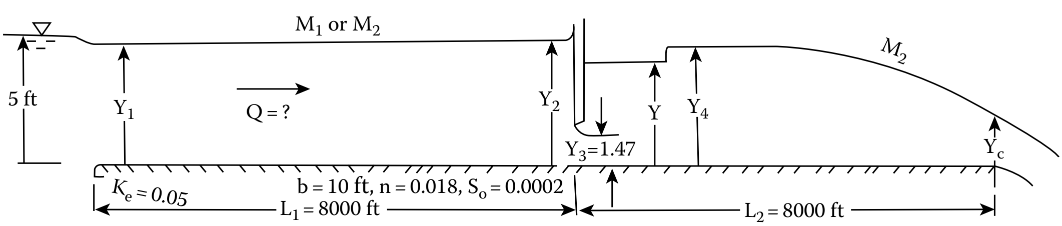
```
Input: 3 .000001 .0005 .013 .0008 10 10 .1 5 1500 32.2 0 .08
90 4.95 6.12 1.85 .7
```

5.88 Modify the computer program (or the use of the software package), or the program listing given in the answer to the previous problem so that there can be a trapezoidal channel upstream from the gate with a bottom width b_1 and a side slope m_1. Also allow for different bottom slopes and Manning's roughness coefficients upstream and downstream from the gate. Allow for a minor loss coefficient to be given to describe the loss in head through the transition from the upstream trapezoidal channel to the rectangular section at the gate and downstream therefrom. Solve the following problem: A gate exists at a distance L = 2000 ft downstream from where a trapezoidal channel with b_1 = 10 ft, and m_1 = 1.5 receives water from a reservoir with H = 4.5 ft, where the minor entrance loss coefficient is K_e = 0.08. At the gate, the channel reduces to a rectangular channel with b_2 = 9 ft. Upstream the bottom slope and Manning's n are S_{o1} = 0.0008 and n_1 = 0.013, respectively, and downstream these are S_{o2} = 0.001 and n_2 = 0.013. Starting with a dimensionless depth of flow below, the gate of $y_2 = Y_2/Y_1 = 0.2$ raise the gate in increments until after it becomes submerged (and beyond) and then lower the gate again. For each of these gate settings, determine (a) the flow rate, Q, (b) the depth in the channel by the reservoir, Y_{beg}, (c) the depth at the end of the trapezoidal channel, Y_{1t}, (d) the depth immediately upstream from the gate in the rectangular channel, Y_{1r}, (e) the depth in the downstream channel Y_3 (assuming this downstream channel is very long), and (f) the depth Y_{2d} immediately upstream from the hydraulic jump when free flow occurs, or the submergence depth Y when the flow past the gate is submerged.

5.89 A mild 3000 ft long rectangular channel contains a gate at its mid position that has submerged flow past it. At the downstream end, the channel terminates in a free overfall, and at its upstream end, it is supplied by a reservoir whose water surface elevation is H ft above the channel bottom. The width, b, the roughness coefficient, n, and the bottom slope of the channel are known. For a known gate setting, Y_G and contraction coefficient, C_c, write out the system of equations that will need to be solve to determine the flow rate and the depths throughout this channel. Also identify the variables that these equations will solve.



5.90 In the previous problem the following values are given: b = 10 ft, n = 0.018, S_o = 0.0002, and L_1 = 8000 ft, L_2 = 8000 ft, and H = 5 ft (the entrance loss coefficient is K_e = 0.05.) The gate is set so it produces a jet downstream from it with a depth of 1.47 ft, i.e., the height of the gate is $Y_G = 1.47/C_c$. Solve for the flow rate, Q, the depth Y_1 immediately downstream from the reservoir, the depth Y_2 immediately upstream from the gate, the submergence depth Y, the depth Y_4 a short distance downstream from the gate, and the critical depth at the end of the channel where the free overfall occurs.



5.91 A gate in a 6 ft wide rectangular channel is to be used as a flow measurement device. It has been determined that this gate has a constant contraction coefficient $C_c = 0.65$ for all settings, and that the head loss it creates can be computed using the local loss $K_L = 0.05$ (constant—not effect by gate setting) multiplied by the downstream velocity head, or the jet velocity head if the gate is submerged. For use by a water master, develop a "rating table" that gives the flow rate as a function of the gates height Y_G above the channel bottom and the upstream depth Y_1 when the gate is free flowing, and as a function of (1) the gate height Y_G, (2) the submerged depth Y, and the upstream depth Y_1 when the gate is submerged. Associated with the tables for submerged flow also provide tables that give values of the downstream depth Y_3. In using the tables that apply when the gate is submerged, it is necessary that there be sufficient head difference across the gate so that a "jet flow" does exist immediately downstream from the bottom of the gate. The depths Y_1 and Y (when submerged) will be determined from stilling wells, and the height of the gate will be determined from a calibrated scale attached to its opening mechanism.

5.92 Develop rating tables, that provide the flow rate past a vertical gate with the same contraction coefficient and local loss coefficient, i.e., $C_c = 0.65$ and $K_L = 0.05$, but this gate supplied water to a downstream rectangular channel from the side of an upstream much larger river, so that the river's depth at the position of the gate is for all practical purposed equal to the energy line. For this situation in addition to providing rating tables for free and submerged conditions also provide a rating table that gives the flow rate as a function of the difference in water surface elevations in the river and in the channel immediately downstream from the gate when the gate is completely opened.

5.93 In Example Problem 5.13, with three submerged gates, assume there is a reservoir at the end of the fourth channel whose water surface elevation is 5.2 ft above the channel bottom. Now what is the flow rate and the depths throughout the channel?

In addition to the downstream reservoir the third gate in Example Problem 5.13 is lowered so that it is 2.0 ft above the channel bottom. Now what is the flow rate coming from the reservoir and the depths throughout the channel system.

5.94 Modify the program SUBMESER used to solve a series of submerged gates so that flow rates can be extracted between gates, i.e., the flow rate passing each succeeding gate in the series is not necessarily the same as that passing the previous gate in the series.

5.95 A series of six gates exist along a channel system to control the flow rates and the depths. These sluice gates are set so their bottom are the following distances above the channel bottom, and the widths of the rectangular channel at the gates are as follows:

Gate No.	1	2	3	4	5	6
Height of gate (ft)	1.5	2.2	1.8	1.6	1.4	1.3
Width of ch. (ft)	12	10	10	10	10	10

The channel is supplied by a reservoir at its upstream end with a water surface elevation 6 ft above the channel bottom and the entrance loss coefficient is $K_e = 0.15$. At its downstream end, the channel discharges into a reservoir whose water surface is 6.5 ft above the channel bottom. The size of the channels between the gates are given in the table below. (b and L are in feet.) Solve for the flow rate and the depths throughout this channel system.

Ch.	b	m	n	L	S_o
1	12	1.5	0.014	1000	0.0004
2	12	1.5	0.014	1300	0.0006
3	10	1.2	0.015	1500	0.0005
4	10	1.2	0.015	1000	0.0004
5	8	1.5	0.014	1600	0.0006
6	8	1.5	0.014	2000	0.0005
7	8	1.5	0.014	1000	0.0008

5.96 Solve the previous problem with all gates raised by 0.3 ft.

5.97 Write out the system of equations that is being solved in Example Problem 5.12. Identify the unknowns in the list of variables.

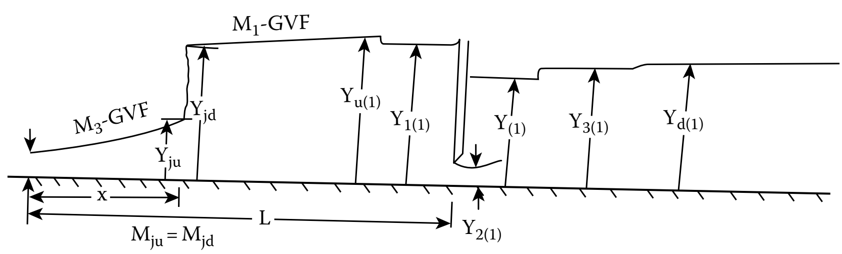
5.98 For the above Problem 5.95 that contains six gates in seven channels sections write out the equations that need to be solved and identify the unknowns in the list of variables.

5.99 Solve for the flow rate into and the depth throughout the channel system in the above problem with the six gates, and the same gate setting except now there is a outflow from the main channel just before each of the six gates equal to 20 cfs. To solve this problem, use the modified program you were to produce in Problem 5.94.

5.100 Program SUBMESER is designed only to handle problems in which the flow by all of the series of gates is submerged. If the flow is "free flowing" past a gate, this gate acts as a control creating two problems: one upstream from it and one downstream from it. Modify your modified program SUBMESER that allows discharges upstream from the gates so that the downstream boundary condition could be a gate with free flow past it. Notice that this program will substitute the energy equation across this downstream gate for the critical flow equation that is handled if IBCE=3 in program SUBMESER.

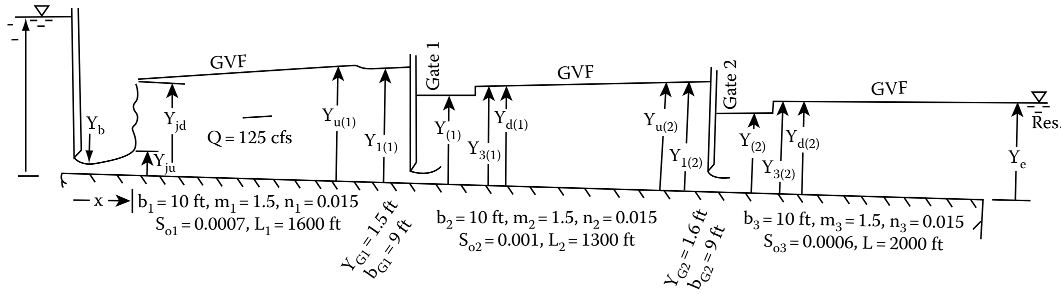
Use this program to obtain a solution to the six-gate system of Problem 5.95 but with outflows of 20 cfs just upstream from the first 5 gates and having the last, or 6th, gate with free flow past it and a depth of 1 ft downstream from it.

5.101 When free flow occurs past a gate, but the series of gates downstream therefrom have submerged flow by them the flow rate becomes a known by solving the upstream problem. As shown in the sketch below the three variables: (1) the depth Y_{ju} (upstream from the hydraulic jump), (2) Y_{jd} (downstream from the hydraulic jump), and (3) x (the position of the hydraulic jump) become unknowns in place of the two variables: (1) depth Y_{1r} and (2) the flow rate Q when the upstream end of a series of submerged gates is feed by a reservoir. Now of course the energy equation at the reservoir is not available. What are the additional equations that allow these variables to be solved? Modify program SUBMESER to accommodate these upstream conditions.



5.102 A flow rate of 125 cfs comes from a gate that supplies the channel from an upstream reservoir into a channel system that contains two submerged gates, and at its end, the channel discharges into a reservoir with a water surface elevation $Y_e = 5.5$ ft above the channel bottom. The distance between the supply reservoir and the first submerged gate is 1600 ft. The water from the supply reservoir comes into the trapezoidal channel with depths that range from $Y_b = 0.16$ to $Y_b = 0.1$ ft. If the first submerged gate is set 1.5 ft above the channel bottom, and the second 1.6 ft above the channel bottom, determine the position of the hydraulic jump, x, and the depth through the channel system. At the submerged gates, the width of the rectangular channels is 9 ft and upstream and downstream from the gates there is a transition to a trapezoidal channel. The sizes, etc. of the channels are given in the table below. For each of these depths Y_b, what is the water surface elevation in the supply reservoir if the width of this upstream supply gate is 10 ft?

No.	b (ft)	m	n	L (ft)	S_o
1	10	1.5	0.015	1600	0.0007
2	10	1.5	0.015	1300	0.001
3	10	1.5	0.015	2000	0.0008

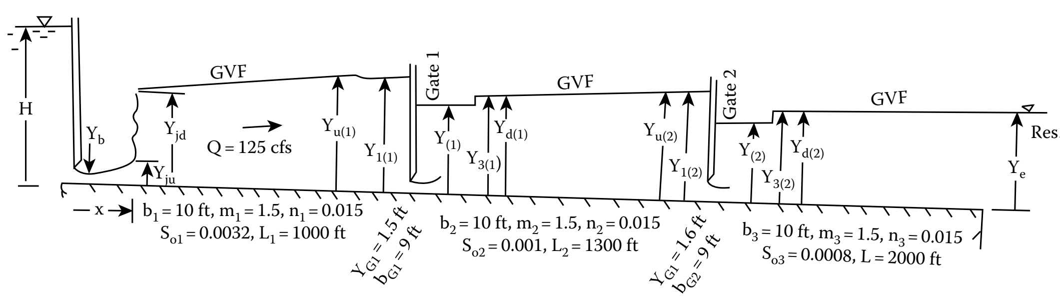


5.103 Resolve the previous problem with the bottom slope of the middle channel equal to 0.0008 and 0.0006 and determine the effects this has on the depths and the position of the hydraulic jump. Then increase the flow rate to Q = 150 cfs and determine its effects.

5.104 A series of six gates exist in the channel that eventually discharges into a reservoir whose water surface is 6.0 ft above the channel bottom. At the upstream gate the channel is rectangular and 12 ft wide and shortly downstream therefrom becomes trapezoidal with $b_1 = 12$ ft, and $m_1 = 1.5$. Upstream from gate 1, the depth is 5.795 ft and downstream from this gate in the trapezoidal section the depth is 0.709 ft. The properties of the channels between the gates are given in the table. Solve for the flow rate and depths throughout the system. The other gates have the widths and produce the jet depths given below. $b_{G2} = 10'$, $Y_{d2} = 2.3'$, $b_{G3} = 10'$, $Y_{d3} = 2.0$, $b_{G4} = 10'$, $Y_{d4} = 1.8'$, $b_{G5} = 10'$, $Y_{d5} = 1.5'$, $b_{G6} = 10'$, $Y_{d6} = 1.5'$.

No.	b (ft)	m	n	L (ft)	So
1	12	1.5	0.014	2800	0.00455
2	10	1.2	0.015	1500	0.0005
3	10	1.2	0.015	1000	0.0004
4	10	1.5	0.014	1600	0.0006
5	11	1.5	0.015	2000	0.0005
6	10	1.5	0.014	1000	0.0008

5.105 The channel system shown below contains three gates and three channels. The downstream channel discharges into a reservoir with a water surface elevation of $Y_e = 6$ ft. The sizes of the channel and the setting of the gates are given on the sketch. If the flow rate released from the upstream reservoirs is (a) Q = 80 cfs with a reservoir head of H = 2.603 ft, and (b) Q = 100 cfs, with a reservoir head of H = 3.701 ft (if the gate is free flowing), what are the depths throughout the system. As a part (c) determine the effects of lowering the water surface elevation of the downstream reservoir if the flow rate is 100 cfs as in part (b).



5.106 A series of six gates exist in a channel similar to that in Problem 5.95 except that the first channel is shorter and much flatter. The properties of the channel between gates is given in the table below. The widths of the channels at the gates and the heights of the gates are as in Problem 5.97, namely $b_{G1} = 10'$, $Y_{d1} = 2.3'$, $b_{G2} = 10'$, $Y_{d2} = 2.3'$, $b_{G3} = 10'$, $Y_{d3} = 2.0$, $b_{G4} = 10'$, $Y_{d4} = 1.8'$, $b_{G5} = 10'$, $Y_{d5} = 1.5'$, $b_{G6} = 10'$, $Y_{d6} = 1.5'$. As in Problem 5.97 the system discharges into a reservoir whose water surface is 6 ft above the channel bottom, and the entrance loss coefficient is $K_e = 0.05$. (a) The head of the reservoir that supplies the system is H = 18 ft. Determine the flow rate into the system, and the depths throughout. (b) Specify flow rates of 170, 150, 125, 100, and 75 cfs and find what reservoir head will produce these flow rates and the depths throughout the channel system.

No.	b (ft)	m	n	L (ft)	So
1	12	1.5	0.014	1300	0.0003
2	10	1.2	0.015	1500	0.0005
3	10	1.2	0.015	1000	0.0004
4	10	1.5	0.014	1600	0.0006
5	11	1.5	0.015	2000	0.0005
6	10	1.5	0.014	1000	0.0008

5.107 Compute the critical side weir heights for a weir in the side of a rectangular channel with a bottom width of b = 2 m, a bottom slope $S_o = 0.0005$, a Manning's n = 0.016 and for entering flow rates of $Q_o = 1$ to $12\,m^3/s$. Also determine the normal depths downstream of the side weir if the ratio of downstream to upstream flow rates varies from 0.9 to 0.5.

5.108 Obtain the series of values needed to plot the curves in Figure 5.14.

5.109 Solve for the length of side weirs required for the conditions as described below. The following are the same for all cases. The upstream channel is trapezoidal with a bottom width b = 2 m, and a side slope m = 1.5. The bottom slope of the channel is $S_o = 0.0005$, and it has a Manning's n = 0.016. The weir's discharge coefficient is $C_d = 0.45$, and its crest is set 1.2 times the critical crest height H_{wc}. Also for all cases the flow rate upstream from the side weir is $Q_o = 10\,m^3/s$, and that downstream from the side weir is $6\,m^3/s$, or $4\,m^3/s$ is discharged from the side of the weir. Case 1: The channel remains prismatic through the length of the side weir, and thereafter so that b = 2 m and m = 1.5. Uniform flow exists downstream from the side weir. Case 2: The side slope of the channel changes from m = 1.5 to 1.0 over the length of the side weir. A GVF profile exists downstream from the side weir that causes the depth to equal $Y_2 = 1.4$ m at the end of the side weir. Case 3: The same as Case 2 with the exception that the depth at the end of the side weir is $Y_2 = 1.45$ m. Case 4: The same as Case 2 with the exception that the depth at the end of the side weir is $Y_2 = 1.5$ m, Case 5: The same as Case w with the exception that the depth at the end of the side weir is $Y_2 = 1.55$ m. Case 6: The side width of the channel reduces from 1.5 to m = 0.75 and the depth downstream from the side weir is normal.

5.110 Generate the series of values needed to plot the curves, etc. on Figure 5.16.

5.111 What is the least cost trapezoidal channel designed for a flow rate of $Q = 10\,m^3/s$, its bottom slope is $S_o = 0.0005$ and its Manning's n = 0.015. The channel is to be designed with 0.5 m of freeboard, and the following cost apply. The lining costs of the bottom and sides is $90 per meter of length, the excavation costs are $10 per cubic meter, and the right-of-way costs are $25 per meter of length. The sides and bottoms are to have thicknesses of 0.09 m.

5.112 The program OPTIMALN is designed to only handle problems that use SI units. Modify this program so that it will also accommodate ES units. With this modified program solve for the least cost trapezoidal channel of type 2 for a design flow rate of $Q = 700\,ft^3/s$, a bottom slope $S_o = 0.0002$, a Manning's n = 0.016. The thickness of the sides and bottom lining is to be 0.3 ft, the free board is 1.5 ft, and the costs are as follows: (a) the costs of lining both the bottom and sides is \$30/ft^3, (b) the excavation cost is \$20/ft^3, and (c) the right-of-way cost is \$5/ft^2.

A possible way of modifying the program given in the text is as shown by the program listed below.

Program OPTIMAES.FOR

```
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER (NV=10)
      EXTERNAL AR,TOP,RH,PER,FUN
      CHARACTER*40 VAR1(5),VAR2(5),ANS*1
      DIMENSION X(3),YY(2)
      COMMON /ALL/ Y(10),C0,QNS
      DATA VAR1/' Uniform depth (m) =','Bottom width (m)= ',
     &' Side slope  = ',' Flow velocity (m/s) =',' Canal total
     &cost  ($)  = '/
      DATA VAR2/' Uniform depth (ft)=','Bottom width (ft)= ',
     &' Side slope  = ',' Flow velocity (fps) =',' Canal total
     &cost  ($)  = '/
      WRITE(*,'(//30X,A\)') ' Profile of type 1 or 2:  '
      READ(*,*) ITYPE
      WRITE(*,*)' Give 1 for SI units, or 2 for ES units'
      READ(*,*) ICU
      CU=1.
      IF(ICU.GT.1) CU=1.486
      C0=0.
      IF(ITYPE.EQ.2) C0=1.
      WRITE(*,'(//,2X,A2)') '  '
      CALL KNOWN(Y,NV,ICU)
2     WRITE(*,'(//10X,A\)')' Is the canal Side Slope known or
     &not ? [Y/N] '
      READ(*,'(A1)') ANS
      IF(ANS.EQ.'n'.OR.ANS.EQ.'N') THEN
      N=3
C   Initial value of the side slope
      X(3)=1./SQRT(3.)
      ELSE
      N=2
      WRITE(*,'(/20X,A\)') ' Side Slope  m = '
      READ(*,*) X(3)
      ENDIF
      QNS=Y(1)*Y(3)/(CU*SQRT(Y(2)))
      SM=SQRT(1.+X(3)*X(3))
C Dimensions of the most efficient section are used to get
C the initial value of the variable Y0 and b
      X(1)=(4.*(QNS/(2.*SM-X(3)))**3)**.125
      X(2)=2.*X(1)*(SM-X(3))
      CALL NEWT(X,N)
      YY(1)=Y(1)/AR(X(1),X(2),X(3))
      CALL COST(YY,X)
      DO 10 I=1,3
      IF(ICU.EQ.1) THEN
```

```
      WRITE(*,'(15X,A30,2X,F12.6)') VAR1(I),X(I)
      ELSE
      WRITE(*,'(15X,A30,2X,F12.6)') VAR2(I),X(I)
      ENDIF
10    CONTINUE
      DO 20 I=1,2
      IF(ICU.EQ.1) THEN
      WRITE(*,'(15X,A30,2X,F12.3)') VAR1(I+3),YY(I)
      ELSE
      WRITE(*,'(15X,A30,2X,F12.3)') VAR2(I+3),YY(I)
      ENDIF
20    CONTINUE
      END
      SUBROUTINE NEWT(X,N)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION D(3,3),F(3),X(3)
      COMMON /ALL/ Y(10),C0,QNS
      DATA EPS,FAC,KMAX/.0000001,.001,25/
      ITER=0
 4    ITER=ITER+1
      DO 10 I=1,N
10    F(I)=FUN(X,I)
      DO 30 I=1,N
      DX=FAC*(X(I)+1.)
      X(I)=X(I)-DX
      DO 20 J=1,N
20    D(J,I)=(F(J)-FUN(X,J))/DX
30    X(I)=X(I)+DX
      CALL SOLVE(D,F,N)
      SUM=0.0
      DO 40 I=1,N
      X(I)=X(I)-F(I)
      IF(X(I).LT.0.) X(I)=0.001
40    SUM=SUM+F(I)*F(I)
      SUM=SQRT(SUM)
      IF(ITER.GT.KMAX) THEN
      WRITE(*,100) ITER
      ELSE
      IF(SUM.GT.EPS) GO TO 4
      ENDIF
      RETURN
 100  FORMAT(//20X,' Does not converge after:',I4,' ITERATIONS'/)
      RETURN
      END
      FUNCTION FUN(X,I)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(3)
      COMMON /ALL/ Y(10),C0,QNS
      GO TO (4,6,8),I
 4    FUN=AR(X(1),X(2),X(3))*RH(X(1),X(2),X(3))**.66666667-QNS
      RETURN
```

```
 6    CALL DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)
      FUN=DCY*DFB-DFY*DCB
      RETURN
 8    CALL DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)
      FUN=DCM*DFB-DFM*DCB
      RETURN
      END
      SUBROUTINE DIFF(X,DCY,DCB,DCM,DFY,DFB,DFM)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(3)
      COMMON /ALL/ Y(10),C0,QNS
      EQUIVALENCE (Y(3),RV),(Y(4),EB),(Y(5),ER),(Y(6),CB) ,
     &(Y(7),CR),(Y(8),CX),(Y(9),CE)
      Z=X(1)+RV
      SM=SQRT(1.+X(3)*X(3))
      B1=X(2)+2.*SM*EB
      RH1=RH(X(1),X(2),X(3))
      DCY=2.*SM*CB*EB+CX*TOP(Z,B1,X(3))+2.*CE*X(3)
      DCB=CR*ER+CX*(Z+ER)+CE
      DCM=2.*EB*X(3)*(ER*(CR+CX)+Z*(CB+CX)+CE)/SM+Z*(Z*CX+2.*CE)-C0*
     &ER**2*(CR+CX)
      DFY=5.*TOP(X(1),X(2),X(3))-4.*RH1*SM
      DFB=5.*X(1)-2.*RH1
      DFM=X(1)*(5.*X(1)-4.*RH1*X(3)/SM)
      RETURN
      END
      SUBROUTINE SOLVE(D,F,N)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION D(3,3),F(3)
      DO 20 K=1,N-1
      DO 20 I=K,N-1
      XMULT=D(I+1,K)/D(K,K)
      DO 10 J=K+1,N
10    D(I+1,J)=D(I+1,J)-XMULT*D(K,J)
20    F(I+1)=F(I+1)-XMULT* F(K)
      F(N)=F(N)/D(N,N)
      F(N-1)=(F(N-1)- F(N)*D(N-1,N))/D(N-1,N-1)
      IF(N.GT.2) THEN
      F(N-2)=(F(N-2)-F(N-1)*D(N-2,N-1)-F(N)*D(N-2,N))/D(N-2,N-2)
      ENDIF
      RETURN
      END
      SUBROUTINE COST(YY,X)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(3),YY(2)
      COMMON /ALL/ Y(10),C0,QNS
      EQUIVALENCE (Y(4),RV),(Y(5),EB),(Y(6),ER),(Y(7),CB),(Y(8),CR),
     &(Y(9),CX),(Y(10),CE)
      Z=X(1)+RV
      SM=SQRT(1.+X(3)*X(3))
```

```
      B1=X(2)+2.*SM*EB
      YY(2)=CR*ER*(B1-C0*X(3)*ER)+2.*CB*EB*SM*Z+CX*(AR(Z,B1,X(3))+
     &ER*(&B1-C0*X(3)*ER))+CE*TOP(Z,B1,X(3))
      RETURN
      END
      SUBROUTINE KNOWN(Y,NV,ICU)
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*40 VAR(10),VAR2(10)
      DIMENSION 4Y(NV)
      DATA VAR/' Q Canal maximum Capacity (m3/s)   ',
     &' S0 Bottom slope                    ',
     &' n Manning coefficient              ',
     &' Rv Free board (m)                  ',
     &' Eb Sides lining thickness (m)      ',
     &' Er Bottom lining thickness(m)      ',
     &' Cb Sides lining unit cost  ($/m3)',
     &' Cr Bottom lining unit cost ($/m3)',
     &' Cx Excavation unit ($/m3)        ',
     &' Ce Right of way unit cost ($/m2) '/
      DATA VAR2/' Q Canal maximum Capacity (ft3/s)  ',
     &' S0 Bottom slope                    ',
     &' n Manning coefficient              ',
     &' Rv Free board (ft)                 ',
     &' Eb Sides lining thickness (ft)     ',
     &' Er Bottom lining thickness(ft)     ',
     &' Cb Sides lining unit cost  ($/ft3)',
     &' Cr Bottom lining unit cost ($/ft3)',
     &' Cx Excavation unit ($/ft3)         ',
     &' Ce Right of way unit cost ($/ft2) '/
      DO 10 I=1,NV
      IF(ICU.EQ.1) THEN
      WRITE(*,'(15X,A45,2X,A3\)') VAR(I),'= '
      ELSE
      WRITE(*,'(15X,A45,2X,A3\)') VAR2(I),'= '
      ENDIF
10    READ(*,*) Y(I)
      RETURN
      END
      FUNCTION AR(Y,B,FM)
      IMPLICIT REAL*8(A-H,O-Z)
      AR=(B+FM*Y)*Y
      RETURN
      END
      FUNCTION TOP(Y,B,FM)
      IMPLICIT REAL*8(A-H,O-Z)
      TOP=B+2.*FM*Y
      RETURN
      END
      FUNCTION PER(Y,B,FM)
      IMPLICIT REAL*8(A-H,O-Z)
```

```
PER=B+2.*Y*SQRT(1.+FM*FM)
RETURN
END
FUNCTION RH(Y,B,FM)
IMPLICIT REAL*8(A-H,O-Z)
RH=AR(Y,B,FM)/PER(Y,B,FM)
RETURN
END
```

REFERENCES

Barnes, H.H. Jr., 1967. Roughness characteristics of natural channels, U.S.G.S., Water-Supply Paper 1849, Washington, DC.

Chow, V.T., 1959. *Open-Channel Hydraulics*, McGraw-Hill, New York, 680 pp.

Jarrett, R.D., 1990. Hydrologic and hydraulic research in mountain rivers, *Water Resources Bulletin, American Water Resources Association*, 26(3):419–429, June.

Limerinos, J.T., 1970. Determination of the manning coefficient from measured bed roughness in natural channels, U.S.G.S., Water-Supply Paper 1898-B, Washington, DC.

Rajaratnam, N. and Subramanya, K. 1967. Flow equation for the sluice gate, *Journal of the Irrigation and Drainage Division*, 93(IR3), 167–186, September.

Rouse, H., B.V. Bhoota, and E.-Y. Hsu, 1951. Design of channel expansions, 4th paper in high-velocity flow in open channels; A symposium, *Transactions of the American Society of Civil Engineers*, 116:326–346.

Samani, Z. and M. Yousaf, 1991. For the circular flume see "Hydraulic characteristics of circular flume," *Journal of Irrigation and Drainage Engineering*, 117(4):558–566, July/August 1991, ASCE.

Samani, Z., H. Magallanez, and R. Skaggs, 2006. A simple flow measuring device for farms, SR-IWM-6, Southern Regional Water Program, New Mexico State University, Las Cruces, NM.

Toch, A., 1955. Discharge characteristics of tainter gates, *Transactions of the American Society of Civil Engineers*, 120:290.

6 Unsteady Flows

6.1 WHEN SHOULD FLOW BE HANDLED AS UNSTEADY?

Most natural flows in streams and rivers change slowly with time. Also, man-made channels and canals have gates that permit a greater or lesser flow through their structures in response to changing demands in water requirement. An important problem that has not been solved, and will probably never be completely solved is how controls are to be operated in time to optimize benefits to the water users, while minimizing waste, and anticipating changes in demand caused by weather conditions, crop requirements, and supply limitations. Thus, the flow in man-made channels is often controlled so as to be unsteady. The fact is that in the real world, most open-channel flows are not steady state, but unsteady and often only modestly so.

However, the design of most channels has been for some steady-state flow rate, generally the largest that the channel is expected to carry. Thus, the study of unsteady flows in open channels is directed more toward the analysis of "what if" questions, than in the design of channel systems. As computers are able to quickly perform the numerous calculations associated with unsteady open-channel hydraulics, there will be greater emphasis on the unsteady flow, and the design of channel systems will likely be studied through computer simulations for many possible unsteady occurrences, that might possibly occur in their operation. The intent of this chapter is to provide an introduction to the concepts associated with unsteady open-channel hydraulics, and provide a minimum background into currently used methods for solving such problems. The methods covered in this chapter are only appropriate when uniform flow exists initially throughout the entire channel. Furthermore, the unsteady conditions, such as depth and velocity, should not be vastly different from these steady-state values. Since, the resulting equations obtained from the simplifying assumptions are simple, the material in this chapter allows you to gain an appreciation for the behavior of different types of unsteady flows, to understand how these unsteady effects are propagated, and to realize what causes an effect. The next chapter deals with numerically solving the two partial differential equations, the St. Venant equations, that govern unsteady flows. The material in this chapter is necessary background for setting up unsteady problems for numerical solutions. Therefore, in addition to learning about the simplified method of characteristics, the subject of this chapter, and how it can readily be used to solve some types of unsteady open-channel flows, you will also gain clear concepts of unsteady open-channel flows.

Considerable research and computer software development is currently underway throughout the world related to unsteady open-channel hydraulics so that it will probably be only a matter of a few years until unsteady solutions will be done on a routine basis.

The question posed in the above heading has no single answer. A general response, however, is that channel systems are currently designed based on steady-state flows. They will most likely be operated under unsteady conditions, and therefore a complete study of new or existing systems should check for various possible unsteady conditions under which they may be called to perform to ensure the soundness of their designs.

6.2 BASIC ONE-DIMENSIONAL EQUATIONS FOR UNSTEADY CHANNEL FLOWS (THE ST. VENANT EQUATIONS)

The equations that describe unsteady flows in open channels are called the St. Venant equations. These equations will be developed in this section. They consist of two partial differential equation, (PDEs); one that satisfies mass conservation, (e.g., the continuity equation), and one that is obtained from Newton's second law. This second equation is actually the momentum equation for one-dimensional hydraulics, and for most applications is also the energy equation, but it will be referred to as the equation of motion since the momentum and the energy equations are both derived from Newton's second law of motion, $\mathbf{F} = m\mathbf{a}$. The two independent variables in these equations will be x (for position) and t (for time); i.e., the variables of the problem will be dependent on both the position in the channel and on time. Two alternate pairs of dependent variables will be used. The first pair consists of the flow rate Q and the depth Y. They will be referred to as the Q-Y set of St. Venant equations. The other pair consists of the average velocity V and the depth Y. These will be referred to as the V-Y set of the St. Venant equations. In addition, these two pairs of equations can be combined to produce other forms of the St. Venant equations. Such a variation, that is most useful, consists of V and c as dependent variables, e.g., the celerity of a small amplitude gravity wave replaces the depth Y as the dependent variable.

6.2.1 Q-Y Set of St. Venant Equations

The continuity equation based on the one-dimensional hydraulics of the open-channel flow was derived in Equation 1.17. It is duplicated below as Equation 6.1:

$$\frac{\partial Q}{\partial x} - q^*(x,t) + T\frac{\partial Y}{\partial t} = 0 \tag{6.1}$$

Since, TdY equals the differential change in area, an alternative way of writing Equation 6.1 is

$$\frac{\partial Q}{\partial x} - q^* + \frac{\partial A}{\partial t} = 0 \tag{6.1a}$$

Generally, the added parenthesis to q^* that denotes it varies with x and t, are not included, nor are parenthesis included to denote that the other variables, such as Q, V, are functions of x and t, except when this needs to be emphasized, because this is understood when dealing with unsteady flows. In other words, x and t are the independent variables, and the other variables such as Y, Q, V, etc. are dependent on these.

The equation of motion can be obtained using the same summation of forces that leads to Equation 4.5 with the exception that now, the flow rate Q is a function of time t as well as x, or Q(x, t), and consequently when the momentum function is differentiated, the chain rule of calculus must be used when taking the derivative of terms involving Q. The differentiation of the momentum function $M = Ah_c + Q^2/(gA)$ with respect to the position x along the channel gives

$$\frac{dM}{dx} = A\frac{\partial Y}{\partial x} + \frac{2Q}{gA}\left(\frac{\partial Q}{\partial x} + \frac{\partial Q}{\partial t}\frac{dt}{dx}\right) - \frac{Q^2}{gA^2}\left(\left.\frac{\partial A}{\partial x}\right|_{Y,t} + Y\frac{\partial Y}{\partial t}\right)$$

in which $dt/dx = 1/V$ and $\partial A/\partial x|_{Y,t}$ is the same quantity defined in Chapter 4 as $\partial A/\partial x|_Y$. The extra t subscript now occurs because of the time-dependent nature of the equation. Substituting dM/dx into the second equation in Equation 4.5, dividing by $\gamma A dx$, and simplifying, results in

$$\frac{2Q}{gA^2}\frac{\partial Q}{\partial x}+\frac{1}{gA}\frac{\partial Q}{\partial t}+\left(1-F_r^2\right)\frac{\partial Y}{\partial x}-\frac{Q^2}{gA^3}\frac{\partial A}{\partial x}\bigg|_{Y,t}-S_o+S_f+\frac{Qq^*}{gA^2}+F_q=0 \tag{6.2}$$

in which F_q is defined as given after Equation 4.6, namely,

$$F_q = 0 \quad \text{for bulk lateral outflow}$$

$$F_q=\frac{Vq^*}{2gA}=\frac{Qq^*}{2gA^2} \quad \text{for seepage outflow, and}$$

$$F_q=\frac{(V-U_q)q^*}{gA}+\frac{h_c}{A}\frac{\partial A}{\partial x}\bigg|_{Y,t} \quad \text{for lateral inflow}$$

in which U_q is the velocity component of the inflow in the direction of the main channel flow, and q^* represent the lateral inflow or outflow per unit length of main channel with q^* negative for lateral outflow, and positive for lateral inflow. It is worthwhile noting that Equation 6.2 is general in that it applies to flow in nonprismatic, as well as in prismatic channels, and allows for either lateral inflow or outflow.

When dealing with steady-state problems, q^* was defined equal to dQ/dx. However, when dealing with equations, such as Equations 6.1 and 6.2, for unsteady problems, the partial derivative of Q with respect to x has an expanded meaning. The other dependent variable t is being held constant in taking $\partial Q/\partial x$. The interpretation of $\partial Q/\partial x$ is therefore how the flow rate changes with position at any given time, and in general this will not be zero, even if there is no lateral inflow or outflow, since a greater or lesser flow rate will be entering the channel rather than leaving it at any time, i.e., fluid volume will be going into or coming out of the channel storage. The continuity Equation 6.1 shows this, that is, even if $q^* = 0$, the change in Q in the x direction for a time held constant equals the negative increase in the surface area with respect to time. For situations involving spatially varied flow, (lateral inflow or outflow), then $\partial Q/\partial x$ includes q^*, as well as the negative of $\partial A/\partial t$ as shown by Equation 6.1. The term q^* represents the actual physical amount of lateral inflow, or outflow (when its magnitude is negative) from the main channel flow.

The term $\partial A/\partial x|_{Y,t}$ is evaluated as in steady problems, namely, the equation that defines the cross section of the channel is differentiated with respect to x with both Y and t held constant. Since this geometry will not change with respect to time unless we are dealing with a channel whose bed may scour or receive sediment deposition, the extra subscript t may be considered redundant for fixed bed channels, but it does remind us of this fact. For example, when dealing with trapezoidal channels, $\partial A/\partial x|_{Y,t} = Y(db/dx) + Y^2(dm/dx)$ with Y evaluated for the time t and the position x that are applicable.

We need to also remind ourselves that Equations 6.1 and 6.2 are based on the assumptions of one-dimensional hydraulics, namely, that we are dealing with average quantities at any position along the channel and at any instant of time. In other words, they do not account for difference in velocity from the top to the bottom of the flow, or from the left to the right side of the flow, nor do they allow for any differences in depth from the left to the right side of the channel. The average velocity V at any point, at any time, is defined as the volumetric flow rate Q divided by the cross-sectional area A. Also, the depth of flow Y will always equal the sum of the pressure head, p/γ, plus the distance from the channel bottom to all points in the cross section, i.e., the pressure distribution is hydrostatic for all times and at all positions. Because of the one-dimensional flow assumptions, questions can be raised about the exact nature of the St. Venant equations, and what terms should, or should not, be included, especially when dealing with problems involving significant amounts of lateral inflow or outflow, because under these circumstances, it may be questionable whether one-dimensional hydraulics is appropriate.

It should be noted that Equation 4.6, that describes a gradually varied flow, is a special case of Equation 6.2 for which the terms involving the time derivative vanish from the equation. Likewise, if the partial derivative with respect to t is zero in Equation 6.1, its variables can be separated and both sides integrated to give the continuity equation for steady-state open channel problems, namely, $Q = Q_o + \int q^* dx$ in which Q_o represents the constant of integration, i.e., the flow coming into the spatially varied flow section. Thus Equations 6.1 and 6.2 are the more general equations for one-dimensional open-channel hydraulics.

6.2.2 V-Y Set of St. Venant Equations

Replacing Q by VA in Equations 6.1 and 6.2 and simplifying gives the following equations involving V and Y as the dependent variables instead of Q and Y:

$$A\frac{\partial V}{\partial x} + VT\frac{\partial Y}{\partial x} + V\left.\frac{\partial A}{\partial x}\right|_{Y,t} - q^* + T\frac{\partial Y}{\partial t} = 0 \quad \text{for the continuity equation, and} \tag{6.3}$$

$$\frac{V}{g}\frac{\partial V}{\partial x} + \frac{\partial Y}{\partial x} - S_o + S_f + F_q + \frac{1}{g}\frac{\partial V}{\partial t} = 0 \quad \text{for the equation of motion} \tag{6.4}$$

6.2.3 Derivation Based Directly on Newton's Second Law of Motion

Before leaving the subject of the development of the equations for unsteady flows for one-dimensional open channel hydraulics, there is merit in developing the equation of motion directly from Newton's second law of motion, **F** = m**a**. Following the procedure used in Chapter 1 in the development of the momentum, Equation 1.20, we note that **a** = d**V**/dt, and then replace **F** by the resultant force **R** and associate dt with the mass m and interpret the result as the mass flow rate ρQ, and finally substituting into Newton's second law gives

$$\mathbf{R} = \rho Q\, d\mathbf{V}$$

Since we are dealing with only one direction in one-dimensional open-channel hydraulics, the bolding of R and V denoting vectors can be dropped and the resultant R is the summation of all forces in the x direction ΣF_x, and dV needs to be expanded by the chair rule of calculus since V is a function of both x and t, namely, $dV = (\partial V/\partial x)dx + (\partial V/\partial t)dt$, and Newton's second law becomes

$$\sum F_x = \rho Q\left(\frac{\partial V}{\partial x}dx + \frac{\partial V}{\partial t}dt\right) = \rho AV\frac{\partial V}{\partial x}dx + \rho AV\frac{\partial V}{\partial t}dt$$

replacing Vdt in the last term of this equation by the differential distance dx, gives

$$\sum F_x = \rho AV\frac{\partial V}{\partial x}dx + \rho A\frac{\partial V}{\partial t}dx = (M_x + M_t)dx$$

in which M_x and M_t are defined as the momentum flux changes (not the momentum functions defined earlier) with respect to x and t, respectively. The difference between the two pressure forces acting on the two ends of a control volume dx long is $F_{p1} - F_{p2} = -\gamma\{\partial(Ah_c)/\partial x\}dx = \gamma A(dY/dx)dx$. Including this resultant pressure force with the component of weight in the x direction, and the resisting shear forces, and force due to a nonprismatic channel as in Chapter 4 in the development of Equation 4.5 in that chapter for ΣF_X results in

$$-\gamma A\frac{\partial Y}{\partial x}dx + \gamma AS_o\,dx - \gamma R_h S_f P\,dx - \gamma h_c\,dA\Big|_{Y,t} = \rho AV\frac{\partial V}{\partial x}dx + \frac{\partial V}{\partial t}dx - \rho q^* U_q\,dx$$

in which the last term represents the momentum flux leaving the differential control volume from the lateral inflow (it has the negative sign for entering). Dividing by $-\gamma A dx$ and bringing all terms to the same side of the equation gives

$$\frac{\partial Y}{\partial x} - S_o + S_f - \frac{h_c}{A}\frac{\partial A}{\partial x}\Big|_{Y,t} + \frac{V}{g}\frac{\partial V}{\partial x} + \frac{1}{g}\frac{\partial V}{\partial t} - \frac{q^* U_q}{gA} = 0$$

This is the equation of motion, Equation 6.4 again. The equation in which Q and Y are considered the dependent variables can be obtained by replacing the average velocity V by Q/A such that

$$\frac{V}{g}\frac{\partial V}{\partial x} = \frac{Q}{gA}\frac{\partial(Q/A)}{\partial x} = \frac{Q}{gA}\left(-\frac{Q}{A^2}\frac{\partial A}{\partial x} + \frac{1}{A}\frac{\partial Q}{\partial x}\right)$$

and

$$\frac{1}{g}\frac{\partial V}{\partial t} = \frac{1}{g}\frac{\partial(Q/A)}{\partial t} = \frac{1}{gA}\frac{\partial Q}{\partial t} - \frac{Q}{gA}\frac{\partial A}{\partial t} = \frac{1}{gA}\frac{\partial Q}{\partial t} - \frac{Qq^*}{gA^2} - \frac{Q}{gA^2}\frac{\partial Q}{\partial x}$$

Upon substituting of these results into the equation above, the following equation of motion results

$$(1-F_r^2)\frac{\partial Y}{\partial x} - S_o + S_f - \frac{Q^2}{gA^3}\frac{\partial A}{\partial x}\Big|_{Y,t} + \frac{1}{gA}\frac{\partial Q}{\partial t} + \frac{2Q}{gA^2}\frac{\partial Q}{\partial x} + \frac{h_c}{A}\frac{\partial A}{\partial x}\Big|_{Y,t} - \frac{q^* U_q}{gA} + \frac{Qq^*}{gA^2} = 0$$

6.2.4 No Lateral Inflow or Outflow and Prismatic Channels

If there is no lateral inflow or outflow, and the channel is prismatic, then Equations 6.3 and 6.4 simplify to the following two equations:

$$A\frac{\partial V}{\partial x} + VT\frac{\partial Y}{\partial x} + T\frac{\partial Y}{\partial t} = 0 \quad \text{as the continuity equation, and} \qquad (6.3a)$$

$$\frac{V}{g}\frac{\partial V}{\partial x} + \frac{\partial Y}{\partial x} - S_o + S_f + \frac{1}{g}\frac{\partial V}{\partial g} = 0 \quad \text{as the equation of motion} \qquad (6.4a)$$

Likewise, if the terms associated with the spatially varied flow, and that account for the nonprismatic channel effect, are deleted, then Equations 6.1 and 6.2 simplify to

$$\frac{\partial Q}{\partial x} + T\frac{\partial Y}{\partial t} = 0 \quad \text{as the continuity equation, and} \qquad (6.1a)$$

$$\frac{2Q}{gA^2}\frac{\partial Q}{\partial x} + \frac{1}{gA}\frac{\partial Q}{\partial t} + (1-F_r^2)\frac{\partial Y}{\partial x} - S_o + S_f = 0 \quad \text{as the equation of motion} \qquad (6.2a)$$

If the channel is of rectangular shape, then Q can be replaced by qb and A by bY and Equations 6.1a and 6.2a become the following two equations:

$$\frac{\partial q}{\partial x}+\frac{\partial Y}{\partial t}=0 \quad \text{as the continuity equation, and} \tag{6.1b}$$

$$\frac{2q}{gY^2}\frac{\partial q}{\partial x}+\frac{1}{gY}\frac{\partial q}{\partial t}+(1-F_r^2)\frac{\partial Y}{\partial x}-S_o+S_f=0 \quad \text{as the equation of motion} \tag{6.2b}$$

Since q = VY, Equations 6.1b and 6.2b, or Equation 6.3a and 6.4a can be written as the following for a rectangular channel:

$$Y\frac{\partial V}{\partial x}+V\frac{\partial Y}{\partial x}+\frac{\partial Y}{\partial t}=0 \quad \text{as the continuity equation in a rectangular channel and} \tag{6.3b}$$

$$\frac{V}{g}\frac{\partial V}{\partial x}+\frac{\partial Y}{\partial x}-S_o+S_f+\frac{1}{g}\frac{\partial V}{\partial t}=0 \quad \text{as the equation of motion} \tag{6.4b}$$

Equation 6.4b is identical to Equation 6.4a and Equation 6.3b can be obtained by dividing Equation 6.3a by the top width T = b for a rectangular channel.

6.2.5 Changes in Dependent Variables

The units associated with the dependent variables in either the Q Y or V Y forms of the St. Venant equations, given above, are different. For flow in a rectangular channel, it is possible to replace Y by the celerity c of a small amplitude gravity wave in the V Y form of Equations 6.3b and 6.4b, such that the two dependent variables V and c have the same units. Note that since $c^2 = gY$ for a rectangular section,

$$\frac{\partial Y}{\partial x}=\frac{2c}{g}\frac{\partial c}{\partial x} \quad \text{and} \quad \frac{\partial Y}{\partial t}=\frac{2c}{g}\frac{\partial c}{\partial t}$$

Substituting these into Equations 6.3b and 6.4b gives the following for the continuity equation and the equation of motion:

$$c\frac{\partial V}{\partial x}+2V\frac{\partial c}{\partial x}+2\frac{\partial c}{\partial t}=0 \quad \text{(Continuity, rectangular channel)} \tag{6.5}$$

$$V\frac{\partial V}{\partial x}+2c\frac{\partial c}{\partial x}+\frac{\partial V}{\partial t}=g(S_o-S_f) \quad \text{(Motion, rectangular channel)} \tag{6.6}$$

6.3 DETERMINATION OF MATHEMATICAL TYPE OF St. VENANT EQUATIONS

Before discussing methods for solving the St. Venant equations, which describe unsteady open-channel flows, it is good to look into what type of partial differential equations they are. Partial differential equations are classified as (1) elliptic, (2) parabolic, or (3) hyperbolic, and each of these types of equations describe problems with unique features. Furthermore, the methods for solving problems described by PDEs are different depending upon the type of the PDE. The typical classification of PDEs given in most elementary text books is only for second-order PDEs. Therefore, this common classification will be described first as background to classify the two simultaneous PDEs that we call the St. Venant equations.

6.3.1 Second-Order PDEs

A second-order PDE is classified as hyperbolic, parabolic, or elliptic according to whether real and distinct, coincident, or imaginary characteristics exist, respectively. That is, if the differential equation is of the form,

$$af_{xx} + bf_{xy} + cf_{yy} = d(x, y, f, f_x, f_y)$$

in which the common notation of using subscripts to denote partial derivatives is used, e.g., f_x is $\partial f/\partial x$, f_{xx} is $\partial^2 f/\partial x^2$, etc. and a, b, c may be any functions of x and y and, as indicated by the parenthesis, d may also be a function of the first derivatives. The type of the equation is

Hyperbolic if $b^2 - 4ac > 0$
Parabolic if $b^2 - 4ac = 0$
Elliptic if $b^2 - 4ac < 0$

The simplest common examples of these three types of equations are

1. The hyperbolic wave equation, $f_{xx} - f_{yy} = 0$
2. The one-dimensional time-dependent parabolic equation, $f_{xx} - f_t = 0$
3. The elliptical Laplace equation for potential fluid flows, $f_{xx} + f_{yy} = 0$

Characteristics represent curves across which derivatives are discontinuous. In the case of elliptic equations, the characteristics are imaginary (i.e., do not exist), and therefore derivatives above any curve in the region are uniquely determined by the functions below that curve. In other words, there are no curves across which derivatives are discontinuous. In the case of hyperbolic equations, the characteristics are real and distinct and, therefore, derivatives (being discontinuous across characteristics) above any curve in the region are not uniquely determined by the function below that curve.

In the case of a second-order PDE, as given above, the continuity of derivatives can be ascertained by first taking the differentials of f_x and f_y by the chain rule,

$$d(f_x) = f_{xx}\,dx + f_{xy}\,dy$$

$$d(f_y) = f_{xy}\,dx + f_{yy}\,dy$$

and second, by combining these differentials with the PDE to form the following system of equations for the unknowns f_{xx}, f_{xy}, and f_{yy} written in matrix notation as

$$\begin{bmatrix} a & b & c \\ dx & dy & 0 \\ 0 & dx & dy \end{bmatrix} \begin{Bmatrix} f_{xx} \\ f_{xy} \\ f_{yy} \end{Bmatrix} = \begin{Bmatrix} d \\ d(f_x) \\ d(f_y) \end{Bmatrix}$$

Third, we note that a unique solution for the second derivative does not exist if the determinant of the matrix is zero. Therefore, if the determinant is zero, then real characteristics exist across which derivatives are discontinuous. Therefore, fourth, we set the determinant to zero, or

$$a(dy)^2 - b(dx\,dy) + c(dx)^2 = 0$$

to form the characteristic equation. Finally we note, that since this is a quadratic equation, the determinant will be zero provided that its discriminate,

$$b^2 - 4ac > 0.$$

If the discriminate is greater than zero, then the equation is hyperbolic, and two real and distinct characteristics exist. If the discriminate

$$b^2 - 4ac = 0$$

a single solution exists, and the characteristics are coincident, giving what is called a parabolic PDE. If the discriminate is negative, i.e.,

$$b^2 - 4ac < 0$$

then, the determinate does not vanish, and no real characteristics exist; the equation is elliptic.

This process has developed the means for typing second-order PDEs. To determine the type of systems of PDEs of the first order, consider the following two general simultaneous quasi-linear PDEs with dependent variables f and g:

$$a_1 f_x + b_1 f_y + c_1 g_x + d_1 g_y = e_1$$

and

$$a_2 f_x + b_2 f_y + c_2 g_x + d_2 g_y = e_2$$

in which the ten coefficients a_1, b_1, …, e_2 may be functions of x, y, f, and g. Using the chain rule to define the differentials, df and dg give

$$df = f_x\,dx + f_y\,dy \quad \text{and} \quad dg = g_x\,dx + g_y\,dy$$

which when combined with the PDEs give the system of equations:

$$\begin{bmatrix} a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ dx & dy & 0 & 0 \\ 0 & 0 & dx & dy \end{bmatrix} \begin{Bmatrix} f_x \\ f_y \\ g_x \\ g_y \end{Bmatrix} = \begin{Bmatrix} e_1 \\ e_2 \\ df \\ dg \end{Bmatrix}$$

Equating the determinate of the matrix to zero leads to the characteristic equation, which expanding around the bottom row gives

$$dx\begin{bmatrix} a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \\ dx & dy & 0 \end{bmatrix} + dy\begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ dx & dy & 0 \end{bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$$

that can be further expanded to

$$(dx)^2(b_1d_2 - b_2d_1) - dx\,dy(a_1d_2 - a_2d_1) + dy\,dx(b_1c_2 - b_2c_1) + (dy)^2(a_1c_2 - a_2c_1) = 0$$

or

$$(a_1c_2 - a_2c_1)(dy)^2 - (a_1d_2 - a_2d_1 - b_1c_2 + b_2c_1)dx\,dy + (b_1d_2 - b_2d_1)(dx)^2 = 0$$

The characteristics of this last equation are real and distinct (hyperbolic), identical (parabolic), or imaginary (elliptic) according to whether the discriminant

$$(a_1d_2 - a_2d_1 - b_1c_2 + b_2c_1)^2 - 4(a_1c_2 - a_2c_1)(b_1d_2 - b_2d_1)$$

is positive, zero, or negative. Thus, this last equation provides a means for classifying systems of first-order PDEs.

As an example, let us consider the two simultaneous first-order equations for potential fluid flows, which upon combining by differentiation, result in Laplace's equation,

$$f_x - g_y = 0 \quad \text{and} \quad f_y + g_x = 0$$

For these equations, $a_1 = 1$, $b_1 = 0$, $c_1 = 0$, $d_1 = -1$, $a_2 = 0$, $b_2 = 1$, $c_2 = 1$, and $d_2 = 0$, which upon substitution in the above, give a negative, so the equations are elliptic.

Now, let us consider the St. Venant equations,

$$2Vc_x + cV_x + 2c_t = 0 \quad \text{and} \quad VV_x + 2cc_x + V_t = g(S_o - S_f)$$

For these equations, $a_1 = 2V$, $b_1 = 2$, $c_1 = c$, $d_1 = 0$, $a_2 = 2c$, $b_2 = 0$, $c_2 = V$, and $d_2 = 1$, so that the discriminate equals $16c^2$, which is larger than zero, and therefore the St. Venant equations are hyperbolic.

6.4 TAKING ADVANTAGE OF THE EQUATION CHARACTERISTICS

Since the St. Venant equations are of the hyperbolic type with two real and distinct characteristics, it is possible to take advantage of this fact. Doing so is commonly referred to as the "method of characteristics." The method of characteristics replaces the two PDE's that constitute the St. Venant equations with an equivalent system of four ordinary differential equation, ODE's. This conversion of the equations can be done in the following steps. First the two PDE's are added after multiplying Equation 6.4a by g and the continuity equation by λ. Combining Equations 6.3a and 6.4a by this addition gives

$$g\frac{\partial Y}{\partial x} + V\frac{\partial V}{\partial x} + \frac{\partial V}{\partial t} + g(S_o - S_f) + \lambda\left(T\frac{\partial Y}{\partial t} + A\frac{\partial V}{\partial x} + TV\frac{\partial Y}{\partial x}\right) = 0 \tag{6.7}$$

Doing this is valid since any linear combination of the original equations still produces the same solution as the original two equations. However since there are two equations λ would be expected to have two values, and as we shall shortly see this is true. This equation can be rewritten as follows:

$$\lambda T\left\{\left(\frac{g}{\lambda T}+V\right)\frac{\partial Y}{\partial x}+\frac{\partial Y}{\partial t}\right\}+\left\{(\lambda A+V)\frac{\partial V}{\partial x}+\frac{\partial V}{\partial t}\right\}=g(S_o-S_f) \tag{6.8}$$

The two quantities within the { } brackets are very similar to the total or substantial derivative, $dF/dt = (\partial F/\partial x)(dx/dt) + \partial F/\partial t = (\partial F/\partial x)V + \partial F/\partial t$. In fact if dx/dt is defined for the two derivatives as

$$\frac{dx}{dt}=\frac{g}{\lambda T}+V \quad \text{and} \quad \frac{dx}{dt}=\lambda A+V$$

respectively, and these two values are equated to each other and the parameter λ solved for results in

$$\lambda=\pm\left\{\frac{g}{AT}\right\}^{1/2} \tag{6.9}$$

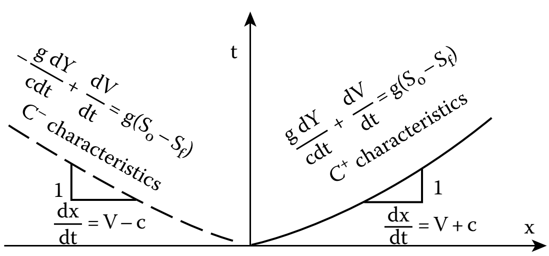
Thus the terms in Equations 6.7 and 6.8 involving λ can be evaluated as follows: $g/(\lambda T) = \pm g\{(AT)\}^{1/2}/(Tg^{1/2}) = \pm(gA/T)^{1/2} = \pm c$, the celerity of a small amplitude gravity wave, $\lambda A = \pm\{g/(AT)\}^{1/2}A = \pm(gA/T)^{1/2} = \pm c$, and $\lambda T = +T[g/(AT)]^{1/2} = +g/(gA/T)^{1/2} = +g/c$, and therefore the two equations for dx/dt (the equations that define the characteristics) become

$$\frac{dx}{dt}=V+c \quad \text{and} \quad \frac{dx}{dt}=V-c \quad \text{or} \quad \frac{dx}{dt}=V\pm c \tag{6.10}$$

respectively, and the above partially differential equation can be written as the following two ordinary differential equations:

$$\pm\frac{g}{c}\frac{dY}{dt}+\frac{dV}{dt}=g(S_o-S_f) \tag{6.11}$$

in which the positive sign in Equation 6.10 is used with the positive sign in Equation 6.11, or $dY/dt = (\partial Y/\partial x)(V+c)+\partial Y/\partial t$, and when using the negative sign $dV/dt = (\partial V/\partial x)(V-c)+\partial V/\partial t$. Thus Equations 6.10 and 6.11 represent 4 ordinary differential equations that can replace the St. Venant partial differential equations. The solution to these equations involves solving the ordinary differential Equations 6.11 along the two characteristic lines defined by Equations 6.10.



To gain some experience in what this means we will first accomplish the same thing but using Equations 6.5 and 6.6 that were developed just for a rectangular channel and are duplicated below:

$$c\frac{\partial V}{\partial x}+2V\frac{\partial c}{\partial x}+2\frac{\partial c}{\partial t}=0 \quad \text{(Continuity, rectangular channel)} \tag{6.5}$$

$$V\frac{\partial V}{\partial x}+2c\frac{\partial c}{\partial x}+\frac{\partial V}{\partial t}=g(S_o-S_f)\quad \text{(Motion, rectangular channel)} \tag{6.6}$$

If Equations 6.5 and 6.6 are added, and then Equation 6.5 is subtracted from Equation 6.6 then the following two equations result:

$$(V+c)\frac{\partial V}{\partial x}+(V+c)\frac{\partial(2c)}{\partial x}+\frac{\partial(V+2c)}{\partial t}=g(S_o-S_f)$$

and

$$(V-c)\frac{\partial V}{\partial x}-(V-c)\frac{\partial(2c)}{\partial x}+\frac{\partial(V-2c)}{\partial t}=g(S_o-S_f)$$

that can be combined and written as

$$(V\pm c)\frac{\partial(V\pm 2c)}{\partial x}+\frac{\partial(V\pm 2c)}{\partial t}=g(S_o-S_f) \tag{6.12}$$

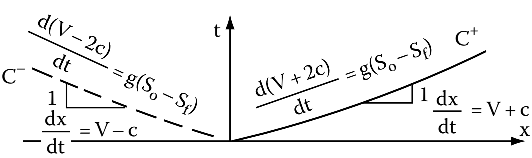
and as above, if the characteristics are defined by

$$\frac{dx}{dt}=V\pm c \tag{6.13}$$

then, Equations 6.12 can be defined as the two ordinary differential equations.

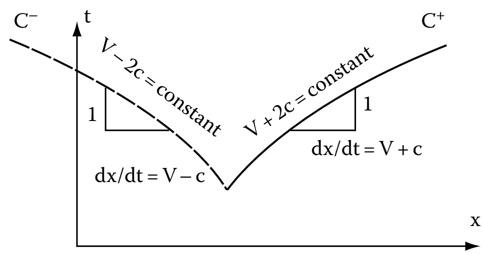
$$\frac{d(V\pm 2c)}{dt}=g(S_o-S_f) \tag{6.14}$$

The interpretation of Equations 6.13 and 6.14 is similar to that given for Equations 6.10 and 6.11. Namely, the two Equations 6.13 define positive C^+ characteristics ($dx/dt = V + c$) and C^- characteristics ($dx/dt = V - c$), respectively, and along these two characteristics the two Equations 6.14 ($d(V + 2c)/dt = g(S_o - S_f)$ and $d(V - 2c)/dt = g(S_o - S_f)$) apply, respectively. The designation positive and negative comes from that $dx/dt = V + c$ is a positive value, i.e., has a positive slope in the xt-plane, whereas $dx/dt = V - c$ is negative for subcritical flow in which $V < c$. That is to say two ordinary differential equations apply along curves (characteristics), in the xt-plane, that are defined by two other ordinary differential equations as illustrated in the sketch below.



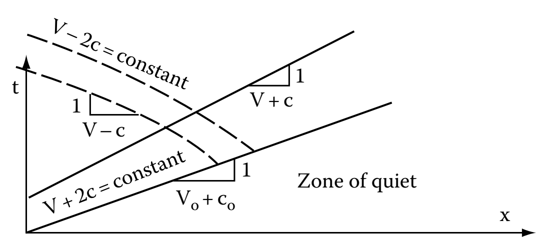
6.5 SOLUTION TO UNSTEADY FLOWS THAT DEVIATE ONLY SLIGHTLY FROM UNIFORM CONDITIONS

If the right side of Equations 6.14 are zero, i.e., S_0 and S_f are equal to each other, then integration of these equations gives $V + 2c =$ constant along each characteristics defined by $dx/dt = V + c$ and $V - 2c =$ constant along the each characteristic $dx/dt = V - c$. The characteristics $dx/dt = V + c$ will be referred to as the C^+ characteristics and $dx/dt = V - c$ will define the C^- characteristics. Since these characteristics are defined as the derivatives dx/dt their slopes at any point in the xt plane will be given by the values of V + c and V − c respectively, as shown in the xt plane below, and are positive and negative as noted below for subcritical flow.



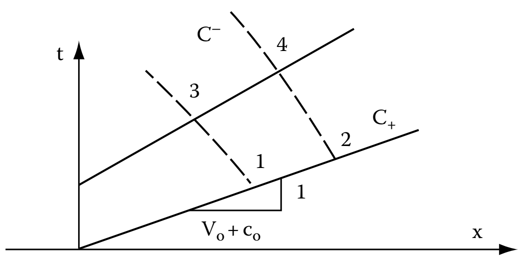
To provide some physical interpretation to the C^+ and C^- characteristics consider a channel that contains a uniform flow, such as water entering a mild rectangular channel from a reservoir. Everywhere along this channel the depth will be the uniform depth Y_o, with the corresponding celerity c_o (a constant), and the velocity equal the constant V_o. Starting at time zero, however, the water elevation in the reservoir begins to drop causing an unsteady flow in the channel to commence at this time. The C^+ characteristic through the origin in the xt plane will be a straight lines with an inverse slope $dx/dt = V_o + c_o$. The magnitude $V_o + c_o$ is called the inverse slope because the slope defined in mathematics is the change in the vertical coordinate per unit value of the horizontal coordinate. In the xt plane t is the vertical coordinate, and x is the horizontal coordinate, and $V_o + c_o$ represents the change in the horizontal coordinate per unit change in the vertical coordinate, i.e., it is the reciprocal of the mathematical slope. In this problem notice that the speed, at which the effects of the decreasing elevation of the reservoir water surface is propagated down the channel, equals the sum of the flow velocity V_o and the celerity of a small amplitude gravity wave c_o. Thus at any time t the distance that the effect will have moved down the channel equals $t(V_o + c_o)$. In the xt plane this is the x value of the C^+ characteristic through the origin corresponding to time t, e.g., the intersection of the $dx/dt(0,0) = V_o + c_o$ with a horizontal line through time t. The (0,0) attached to dx/dt denotes that this characteristic passes through the point (0,0) in the xt plane.

The triangular region below and to the right of the beginning C^+ characteristic $dx/dt(0, 0) = V_o + c_o$ is referred to as the **zone of quiet** since it is unaffected by the transient, or unsteady flow. The speed at which the zone of quite is receding equals $V_o + c_o$ and this is the rate at which the transient is propagating into the otherwise uniform flow.



First, we will deal only with what are called **negative** or **dispersive** waves. The above situation causes a negative wave because the depth decreases in time. A positive wave results when the depth increases with time. An alternative, and perhaps more informative way of understanding the difference between negative and positive waves is to examine the nature of the family of C^+ characteristics. In the case of dispersive waves these characteristics, fan out or diverge further from each other with increasing time. This fan effect of the C^+ characteristics is due to having each subsequent characteristic that intersects the t axis with a smaller inverse slope than the previous one, i.e., $dx/dt(0, t_2) < dx/dt(0, t_1)$ where $t_2 > t_1$. The physical meaning of the diverging of the family of C^+ characteristics is that small amplitude waves that originate later in a smaller depth of flow will travel slower than those that started their travel along the channel sooner, and as time increases the distance separating these separate waves will therefore increase.

A **positive** wave results if the depth increases with time. For such waves each subsequent C^+ characteristic will have a larger inverse slope dx/dt than its predecessor. This large inverse slope comes from the fact that the celerity c is larger for increasing depths. Consequently small waves that originate later will eventually out run those that started their journey down the channel earlier in time. As these waves over take earlier waves they tend to form a surge, i.e., a larger than small amplitude wave. Once a surge forms it traps the train of continuous waves in it and they are unable to speed on ahead. In the xt plane the formation of this surge is shown by the intersection of consecutive C^+ characteristics. We will deal only with **negative** or **dispersive** waves in the next couple of sections in this book. Thereafter, **positive** waves will be discussed, and means provided to determine the time and location where a surge is likely to occur. The assumption that we are currently working under that $S_o - S_f$ can be ignored, is more valid for problems containing negative waves than those that contain positive waves because the formation of a surge is associated with more rapid changes in depths with respective to position and time. An important principle associated with characteristics under the assumption that the right of Equation 6.14 is zero, or $S_o - S_f = 0$ is that if any one of the characteristics in either the positive, C^+, or negative C^- family is a straight line, then all other characteristics of this family must be straight lines. This principle would dictate all other C^+ characteristics for the example problem above will be straight lines because the C^+ characteristic through the origin $dx/dt(0, 0) = V_o + c_o$ is a straight line because of the initial uniform flow conditions that exist at time zero. Proof of this principle is quite easy. This proof consists of considering a pair of adjacent C^+ characteristics, and a pair of adjacent C^- characteristics as shown in the sketch below. To keep track of the intersections of these characteristics the numbers 1, 2, 3, and 4 are used. From Equation 6.14 with the positive signs it follows that



$$V_3 + 2c_3 = V_4 + 2c_4 \tag{a}$$

Likewise from Equation 6.14 with the negative signs it follows that

$$V_3 - 2c_3 = V_1 - 2c_1 \tag{b}$$

$$V_4 - 2c_4 = V_2 - 2c_2 \tag{c}$$

But since $V_1 = V_2 = V_o$ and $c_1 = c_2 = c_o$ it follows that $V_1 - 2c_1 = V_2 - 2c_2$ and therefore

$$V_3 - 2c_3 = V_4 - 2c_4 \tag{d}$$

Adding Equations (a) and (d) produces $2V_3 = 2V_4$, or $V_3 = V_4$. Subtracting Equation (d) from Equation (a) produces $4c_3 = 4c_4$, or $c_3 = c_4$. Therefore the slope dx/dt of the characteristic passing through points 3 and 4 is constant since this $dx/dt = V_3 + c_3 = V_4 + c_4$. Not only is the slope of the characteristic constant, i.e., a straight line, but anywhere along it, the velocity V is constant, and the celerity c is constant, which means the depth is also constant along it. Since the unit flow rate q = VY it is also constant along any C^+ characteristic.

It should be noted that the above proof was based on having $S_o - S_f = 0$. If this difference is not zero, then V + 2c and V − 2c will not be constant along the C^+ and C^- characteristics respectively, and the above principle does not apply. As the difference $S_o - S_f$ deviate from zero as the unsteady flow begins to have depths and velocities different than V_o and Y_o then the C^+ characteristic will also in practice deviate from being straight. However, if the channel has a relatively flat slope, then the magnitude of S_f will not be too different from the magnitude of S_o for transient conditions that perturb the flow only modestly. In any event the results from simpler solutions that follow in this chapter can be used as a preliminary indication of the unsteady behavior of the flow.

Above, we have considered situations that produce straight lines for the positive characteristics. Other situations will produce straight lines for the C^- characteristics, and these are given later. If the initial C^- characteristic with an inverse slope $dx/dt = V_o - c_o$ is a straight line, i.e., $V_o - c_o$ is constant, then all members of this family must be straight lines following a similar proof to that above.

6.6 BOUNDARY CONDITIONS

A solution of unsteady problems can be completed throughout the entire xt plane provided that the slope of all C^+ characteristics can be obtained at the origin, (along the t axis). These slopes are obtained from the so-called boundary conditions that exist at the origin (x = 0) for any time t. There are three different types of boundary conditions that commonly occur, namely: (1) the depth is specified at the origin as a function of time, e.g., Y(0, t) is given, (2) the velocity is specified at the origin as a function of time, e.g., V(0, t) is given, and (3) the flow rate per unit width of channel is specified at the origin as a function of time, e.g., q(0, t) is given. Of these, the latter is most common in practice, but its implementation in the solution process is also the most difficult. Typically, a rate of change of the depth Y, the velocity V, or the unit flow rate q is given, i.e., dY/dt, dV/dt, or dq/dt is specified. (4) A head H(0, t) is known at the upstream or the downstream end of the channel. How each of these four boundary conditions permits the slope dx/dt(0, t) of the C^+ characteristics to be obtained is described under the four separate headings below.

6.6.1 Depth at Origin Specified, Y(0, t)—Known

Consider moving along a C^- characteristic from the origin at any time t to the correct point on the C^+ characteristic through the origin. As shown on the sketch below, along this C^- characteristic $V(0, t) - 2c(0, t) = V_o - 2c_o$, or

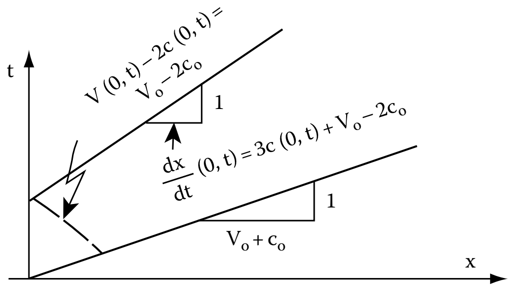
$$V(0,t) = 2c(0,t) + V_o - 2c_o \tag{6.15}$$

Since the slope of the C^+ characteristic through this point (0, t) in the xt plane is $dx/dt(0, t) = V + c$, its magnitude can be determined from the known way in which the depth varies at the origin as a function of time, or

$$\frac{dx}{dt}(0,t) = 3c(0,t) + V_o - 2c_o \tag{6.16}$$

Since $c = (gY)^{1/2}$ it follows that

$$\frac{dx}{dt}(0,t) = 3\sqrt{gY(0,t)} + V_o - 2c_o \tag{6.17}$$



It is interesting to note that the V – 2c that is constant along every C^- characteristic must have the same value as this constant along all other C^- characteristics, because all negative characteristics that we are considering intersect with the initial C^+ characteristic.

6.6.2 Velocity at Origin Specified, V(0, t)—Known

If the velocity at the origin is known as a function of time then the slope of C^+ characteristics through the origin can be determined by solving Equation 6.15 for c(0, t) and substituting the result in place of c in the equation dx/dt = V + c, to give

$$\frac{dx}{dt}(0,t)=\frac{3}{2}V(0,t)-\frac{1}{2}V_o+c_o \tag{6.18}$$

6.6.3 Flow Rate q at the Origin Specified, q(0, t)—Known

When the flow rate per unit width in the rectangular channel is given at the origin as a function of time, then this value must be equal to the product V(0, t)xY(0, t) = V(0, t)xc^2(0, t)/g. Substituting from Equation 6.15 for V(0, t) results in the following cubic equation that gives c(0, t):

$$gq(0,t)-c^2(0,t)\left\{2\left[c(0,t)-c_o\right]+V_o\right\}=0 \tag{6.19}$$

After finding the appropriate root for c(0, t) from this equation, this value can be substituted into Equation 6.16 to find the slope of the C^+ characteristic, or the velocity V(0, t) can be evaluated from Equation 6.15, and the sum of c(0, t) and V(0, t) obtained to give dx/dt(0, t). See Example Problem 6.2 that follows for possible techniques for solving this cubic equation.

6.6.4 Upstream Reservoir Water Surface Specified, H(0, t)—Known

If the reservoir's water surface elevation at the downstream end of a channel is given as a function of time, this change in elevation H(0, t) is equivalent to Y(0, t) since the velocity head in the channel is dissipated upon entering the reservoir. However, if the reservoir's water surface elevation that supplies a channel changes with time, the channel depth will be less than the reservoir depth H above the channel bottom by the velocity head plus the entrance loss, or H = Y + (1 + K_e)V^2/(2g). Since V(0, t) = 2c(0, t) + V_o – 2c_o and Y = c^2/g, this energy equation can be written with c as the unknown as

$$H=\frac{c^2}{g}+\frac{1+K_e}{2g}(2c+V_o-2c_o)^2 \tag{6.20}$$

Upon solving for c, Y, V and q = VY can be determined.

EXAMPLE PROBLEM 6.1

Water enters a very wide rectangular channel and flows through it under uniform conditions at a depth $Y_o = 4.5$ ft, and a velocity $V_o = 3$ fps. Suddenly at time $t = 0$ the depth in the channel at its beginning falls at a rate of $dY/dt = -0.005$ ft/s. Ignoring the difference between S_o and S_f determine the following: (a) The distance downstream from the channel entrance that this decreasing depth has been felt after 60, 120, and 180 s, (b) The time when the depth will be 4.2 ft at a position 1000 ft downstream from the channel entrance, and (c) The position where the depth will be 4.3 ft at time t = 80 s.

Solution

The (a) part of this problem requires that the inverse slope of the initial C^+ characteristics $dx/dt(0, 0) = V_o + c_o$ be determined to define the "zone of quiet." To determine this inverse slope $c_o = \sqrt{gY_o} = 12.037$ fps. Therefore, the effects of the decreasing depth at the origin propagate down the channel with a speed $V_o + c_o = 15.037$ fps. Therefore at t = 60 s, the water will just begin dropping at a position 60(15.037) = 902.2 ft, and at times 120 and 180 the positions will be 1804.5 and 2706.7 ft, respectively. The time t for the (b) part of the problem can be determined by adding t_1, which is the time when the depth at the origin will be 4.2 ft, to $\Delta t = 1000/dx/dt$, the time it takes this wave to move from the origin to a position 1000 ft downstream. Time $t_1 = (Y_o - Y)/|dY/dt| = 0.3/.005 = 60$ s, and the inverse slope $dx/dt(0,60) = 3c(0,60) + V_o - 2c_o = 3\sqrt{32.2(4.2)} + 3 - 2(12.057) = 13.813$ fps, and therefore $\Delta t = 1000/13.813 = 72.4$ s, or t = 132.4 s. In the (c) part of this problem we need to find the time t_1 for the depth at the origin to be 4.3 ft, and then subtract time t_1 from the given 80 s, to find Δt. With Δt known, the distance x can be obtain by multiplying Δt by $dx/dt(0, t_1)$. The time $t_1 = 0.2/0.005 = 40$ s, and $dx/dt(0, t_1) = 3\sqrt{32.2(4.3)} + 3 - 2(12.037) = 14.227$, and therefore the position is $x = (80 - 40) \times (14.227) = 569.1$ ft.

EXAMPLE PROBLEM 6.2

A rectangular channel contains a uniform flow with a depth $Y_o = 4$ ft, and $V_o = 5$ fps. A gate at its upstream end is used to control the flow into the channel. Make up a table that gives the depth and velocity at the gate as a function for the flow rate past the gate.

Solution

This problem requests that conditions be determined at the origin of the xt-plane, i.e., Y(0, t) and V(0, t) are to be determined as a function of the flow rate q(0, t). The third type B.C. Equation 6.19 needs to be solved. Since this equation is a third degree polynomial, a convenient way of solving this equation is to modify the computer program that uses Laguerre's method from Chapter 3 for finding roots of a polynomial whose listing was given in Chapter 3. The modified main program is given below, after the solution table is given.

Flow Rate q (cfs/ft)	18	16	14	12	10	8	6	4	2	0
Depth Y (ft)	3.876	3.747	3.613	3.474	3.328	3.174	3.010	2.835	2.644	2.432
Velocity (fps)	4.644	4.270	3.875	3.454	3.005	2.520	1.993	1.411	0.756	0.000
dx/dt (fps)	15.82	15.25	14.66	14.03	13.36	12.63	11.84	10.97	9.984	8.849

Listing of FORTRAN program to solve above problem (EPRB6_2.FOR)

```
      PARAMETER (ND=3,EPS=1.E-6)
      COMPLEX C(ND+1),ROOTS(ND),AD(51),Z1,Z2,Z3
      EPS1=2.*EPS*EPS
10    WRITE(6,*)' Give: Vo,Yo,q,g'
      READ(5,*) VO,YO,Q,G
      CO=SQRT(G*YO)
      C(4)=CMPLX(1.,0.)
      C(3)=CMPLX(.5*VO-CO,0.)
      C(2)=CMPLX(0.,0.)
      C(1)=CMPLX(-G*Q/2.,0.)
```

```
      DO 20 J=1,ND+1
20    AD(J)=C(J)
      DO 30 J=ND,1,-1
      Z1=CMPLX(0.,0.)
      CALL LAGU(AD,J,Z1,EPS)
      IF(ABS(AIMAG(Z1)).LE.EPS1*ABS(REAL(Z1)))
     &Z1=CMPLX(REAL(Z1),0.)
      ROOTS(J)=Z1
      Z2=AD(J+1)
      DO 30 JJ=J,1,-1
      Z3=AD(JJ)
      AD(JJ)=Z2
30    Z2=Z1*Z2+Z3
      DO 50 J=2,ND
      Z1=ROOTS(J)
      DO 40 I=J-1,1,-1
      IF(REAL(ROOTS(I)).LE.REAL(Z1)) GO TO 50
40    ROOTS(I+1)=ROOTS(I)
      I=0
50    ROOTS(I+1)=Z1
      DO 60 I=1,ND
      IF(ABS(AIMAG(ROOTS(I))).GT. .0001) GO TO 60
      COT=REAL(ROOTS(I))
      DXDT=3.*COT+VO-2.*CO
      DEPTH=COT*COT/G
      IF(DEPTH.LT.1.E-5) THEN
      V=0.
      ELSE
      V=Q/DEPTH
      ENDIF
      WRITE(6,100) COT,DEPTH,V,DXDT
100   FORMAT('Wave speed, c=',F10.3, 'Depth=',F10.3,'Velocity=',F10.3,
      *'Slope dx/dt=',F10.3)
60    CONTINUE
      WRITE(6,*)' Give 1 if to solve another problem, otherwise 0'
      READ(5,*) I
      IF(I.EQ.1) GO TO 10
      STOP
      END
      SUBROUTINE LAGU(C,ND,Z1,EPS)
      COMPLEX C(ND+1),Z1,DX,ZO,Z2,Z3,Z4,Z5,DZ,SS,Z6,Z7,Z8,ZERO,XX,FF
      ZERO=CMPLX(0.,0.)
      DO 20 ITER=1,50
      Z2=C(ND+1)
      Z3=ZERO
      Z4=ZERO
      DO 10 J=ND,1,-1
      Z4=Z1*Z4+Z3
      Z3=Z1*Z3+Z2
10    Z2=Z1*Z2+C(J)
      IF(CABS(Z2).LE.1.E-8) THEN
```

```
      DX=ZERO
      ELSE IF(CABS(Z3).LE.1.E-8.AND.CABS(Z4).LE.1.E-8) THEN
      DX=CMPLX(CABS(Z2/C(ND+1))**(1./FLOAT(ND)),0.)
      ELSE
      Z5=Z3/Z2
      Z8=Z5*Z5
      DZ=Z8-2.*Z4/Z2
      XX=(ND-1)*(ND*DZ-Z8)
      YY=ABS(REAL(XX))
      ZZ=ABS(AIMAG(XX))
      IF(YY.LT.1.E-12 .AND. ZZ .LT.1.E-12) THEN
      SS=ZERO
      ELSE IF (YY.GE.ZZ) THEN
      FF=(1./YY)*XX
      SS=SQRT(YY)*CSQRT(FF)
      ELSE
      FF=(1./ZZ)*XX
      SS=SQRT(ZZ)*CSQRT(FF)
      ENDIF
      Z6=Z5+SS
      Z7=Z5-SS
      IF(CABS(Z6).LT.CABS(Z7)) Z6=Z7
      DX=FLOAT(ND)/Z6
      ENDIF
      ZO=Z1-DX
      IF(Z1.EQ.ZO) RETURN
      Z1=ZO
      IF(CABS(DX).LE.EPS*CABS(Z1)) RETURN
20    CONTINUE
      WRITE(6,*)' FAILED TO CONVERGE'
      RETURN
      END
```

The two program below provide alternative methods for solving the implicit cubic equation resulting from $q = (c^2/g)(2c + V_o - 2c_o)$. Program EPRB6_2A.FOR takes advantage of the fact that only one real root exists (the other two are complex imaginary) and uses the method for solving a cubic equation described in the CRC handbook, and used to solve the energy equation in Chapter 2 and the momentum equation in Chapter 3 for rectangular channel. Program EPRB6_2B.FOR uses the Newton method to solve the equation.

Let us examine how the cubic equation is solved. When written as a cubic equation, the boundary condition that gives c as a function of the specified unit flow rate is $c^3 + (V_o/2 - c_o)c^2 - gq = 0$. For a subcritical flow, the coefficient for the squared term is negative, and likewise the constant is negative. The argument of arc cosine is limited between ±1, and with both the above coefficients negative, the argument is less than −1. Therefore, only one real root exists, and for our above cubic equation can be found by defining the following new variables: $S = (V_o/2 - c_o)^2/9$, $T = [(V_o/2 - c_o)^3 - 6.75gq]/27$, and finally $STS = [(T^2 - S^3)^{1/2} + |T|]^{1/3}$. From these variables, the celerity that is the only positive real root of the cubic equation is given by

$$c = \frac{T}{|T|}\left\{STS + \frac{S}{STS}\right\} - \frac{1}{3}\left(\frac{1}{2}V_o - c_o\right)$$

Program EPRB6_2A.FOR

```
C  Only one real root exits for equation
1     WRITE(*,*)' Give: Yo,Vo,q & g'
      READ(*,*) Yo,Vo,q,g
      IF(Yo.LT.1.E-5) STOP
      co=SQRT(g*Yo)
      VOC=(.5*Vo-co)
      S=VOC**2/9.
      T=(VOC**3-6.75*g*q)/27.
      STS=(SQRT(T*T-S**3)+ABS(T))**.33333333
      c=-T/ABS(T)*(STS+S/STS)-VOC/3.
      Y=c*c/g
      V=2.*c+Vo-2.*co
      WRITE(6,100) Yo,Vo,q,g,c,Y,V,V+C
100   FORMAT(' Yo=',F8.4,' Vo=',F8.3,' q=',F8.3,' g=',F8.2,/
     &' c=',F9.5,' Y=',F9.5,' V=',F9.5,' dx/dt=',F9.5)
      GO TO 1
      END
```

Program EPRB6_2B.FOR

```
C  Uses Newton Method to solve q=(c**2/g)*(2c+Vo-2co)
1     WRITE(*,*)' Give: Yo,Vo,q & g'
      READ(*,*) Yo,Vo,q,g
      IF(Yo.LT.1.E-5) STOP
      co=SQRT(g*Yo)
      VOC=(Vo-2.*co)
      GQ=g*q
      c=.9*co
      M=0
5     F=c*c*(2.*c+VOC)-GQ
      M=M+1
      IF(MOD(M,2).NE.0) THEN
      FF=F
      CC=c
      c=1.005*c
      GO TO 5
      ENDIF
      DIF=(c-CC)*FF/(F-FF)
      c=CC-DIF
      IF(ABS(DIF).GT.1.E-6 .AND. M.LT.30) GO TO 5
      IF(M.EQ.30) WRITE(*,*)' Failed to converge',DIF,c
      Y=c*c/g
      V=2.*c+Vo-2.*co
      WRITE(6,100) Yo,Vo,q,g,c,Y,V,V+C
100   FORMAT(' Yo=',F8.4,' Vo=',F8.3,' q=',F8.3,' g=',F8.2,/
     &' c=',F9.5,' Y=',F9.5,' V=',F9.5,' dx/dt=',F9.5)
      GO TO 1
      END
```

Problems that can be solved readily using the methods just described generally fall within the following three categories depending upon what is requested. In illustrating these three types of problems, the type 1 boundary condition Y(0, t) will be used; however V(0, t), or q(0, t), might also be the boundary condition, so there are actually nine types of problems; three basic types with three possible boundary conditions under each basic type.

1. Y = given, x = given, t = ?. This type of problem states "at the following position(s) determine the times when a specified depth and corresponding velocity (or flow rate) will occur" in a channel caused by a dispersive wave. This type of problem is the simplest to solve because the solution consists of the direct steps: (a) find the time t_1 when the given depth occurs at the origin, which is obtained from $t_1 = |Y_o - Y|/|(dY/dt)| = |Y_o - Y|\,|(dt/dY)|$ if the rate of change in depth dY/dt is constant, (b) determine the slope of this C^+ characteristic from $dx/dt = 3c(0, t) + V_o - 2c_o$, (c) then find the time $\Delta t = x/(dx/dt)$ (or $\Delta t = x/(3c + V_o - 2c_o)$), and (d) add these two times to get the requested time, or $t = t_1 + \Delta t$.
2. t = given, Y = given, x = ?. This type of problem indicates that at "selected times determine where specified depths will occur." Problems of this type can be solved by completing the following steps: (a) The times are determined at the origin (x = 0) when the specified depths will occur. These times are denoted by t_1 and as in type 1 problems are obtain from $t_1 = |Y_o - Y|/|(dY/dt)| = |Y_o - Y|\,|(dt/dY)|$ when the rate of depth change is constant. If the boundary condition specifies a constant rate of change of velocity then $t_1 = |V_o - V|/|(dV/dt)| = |V_o - V|\,|(dt/dV)|$. If the flow rate per unit width is specified to vary linearly with time, then $t_1 = |q_o - q|/|(dq/dt)| = |q_o - q|\,|(dt/dq)|$. (b) The slopes $dx/dt(0, t_1)$ corresponding to these times are determined from one of the three boundary condition, Equations 6.17, 6.18, or 6.19, depending on which utilizes the given information about the cause of the unsteady flow. (c) The times Δt are computed from $\Delta t = t - t_1$. (d) The requested positions are computed from $x = (dx/dt)\,\Delta t$.
3. t = given, x = given, Y = ?. This type of problem gives both times (generally at a designated increment apart) and selected positions and requests that the corresponding depths and velocities (or flow rates) be determined corresponding to each pair of x and t. Problem of this type are solved by noting that the given t is the sum of t_1 and Δt, or $t = t_1 + \Delta t$. Since t_1 can be determined from the information about the cause of the unsteady flow from one of the boundary conditions it is written in terms of the unknown celerity c, which is used to determine the wanted depth after it is solved for. For example, if the time it takes for a given depth change is given, i.e., $dt/dY(0, t)$ and this value is constant, then $t_1 = |(Y_o - Y)|/|(dY/dt)| = |(dt/dY)||(Y_o - c^2/g)|$. The value of Δt is obtained from $\Delta t = x/(dx/dt) = x/(3c + V_o - 2c_o)$. Equating the sum of t_1 and Δt to the given t,

$$\frac{|Y_o - Y|}{|dY/dt|} + \frac{x}{dx/dt} = t = \frac{|Y_o - c^2/g|}{|dY/dt|} + \frac{x}{3c + V_o - 2c_o} \tag{6.21}$$

or expressing the result as a third degree polynomial gives the following equation:

$$\frac{3}{g}\frac{dt}{dY}c^3 + \frac{dt}{dY}\frac{(V_o - 2c_o)}{g}c^2 + 3\left(t - \frac{dt}{dY}Y_o\right)c + \left(t - \frac{dt}{dY}Y_o\right)(V_o - 2c_o) - x = 0 \tag{6.21a}$$

This cubic equation is solved for the appropriate root of c (from the context of the problem the appropriate root can be selected, and if the Newton method is used with a "good" initial guess, the solution will generally converge to the appropriate root), or the original Equation 6.21 on your calculator, or an applications software package. Thereafter, the depth Y is determined from $Y = c^2/g$, and the velocity corresponding to this depth can be obtained from $V = 2(c - c_o) + V_o$, and if the flow rate per unit width of the channel is wanted, it is obtained as the product of V and Y or $q = VY$.

It is important to note that Equation 6.21 applies only for times after the flow begins to drop at the specified positions. Prior to this time, the flow remains uniform. Therefore, as a preliminary step, it is necessary to determine the time when the transient arrives at the positions x.

EXAMPLE PROBLEM 6.3

The flow downstream from a gate is uniform with $V_o = 4$ fps and $Y_o = 6$ ft when suddenly the flow rate at the gate is decreased at a constant rate of $dq/dt = -0.003$ $(ft/s)^2$ for 1 h and then held constant again. At a position 5280 ft downstream from the gate determine the times when the depths will decrease from the uniform depth at increments of 0.1 ft, i.e., when Y = 5.9, 5.8, 5.7, etc.

Solution

This problem is of type #2, as described above. Therefore, the first step is to determine the times t_1 at the origin when the depths will be 5.9, 5.8 ft, etc. This is done by solving the boundary condition Equation 6.15 that gives the velocity V in terms of c (e.g., the depth Y), or $V = 2(c - c_o) + V_o$. The q at the origin is given by $q = VY$, and therefore $t_1 = (q_o - q)/|dq/dt|$. To find Δt to add to t_1 to get the total time, divide the distance x by dx/dt, which is equal to V + c. The results from this solution are given below, along with the few programming statements needed to obtain the solution.

Table giving solution to problem

Y (ft)	c (fps)	V (fps)	q (ft²/s)	t_1 (s)	dx/dt	t (s)
6.000	13.900	4.000	24.000	0	17.900	295
5.900	13.783	3.767	22.227	591	17.551	892
5.800	13.666	3.533	20.490	1170	17.199	1477
5.700	13.548	3.296	18.788	1737	16.844	2051
5.600	13.428	3.057	17.121	2293	16.486	2613
5.500	13.308	2.817	15.491	2836	16.124	3164
5.400	13.186	2.573	13.897	3368	15.760	3703
5.300	13.064	2.328	12.339	3887	15.392	4230
5.200	12.940	2.080	10.818	4394	15.020	4745
5.100	12.815	1.830	9.335	4888	14.645	5249
5.000	12.689	1.578	7.889	5370	14.266	5740
4.900	12.561	1.323	6.482	5839	13.884	6220
4.800	12.432	1.065	5.113	6296	13.497	6687
4.700	12.302	0.805	3.782	6739	13.107	7142
4.600	12.170	0.542	2.492	7169	12.712	7585
4.500	12.037	0.276	1.240	7587	12.313	8015
4.400	11.903	0.007	.029	7990	11.910	8434
4.300	11.767	−0.265	−1.142	8381	11.501	8840
4.200	11.629	−0.541	−2.271	8757	11.089	9233
4.100	11.490	−0.819	−3.359	9120	10.671	9615

FORTRAN listing to obtain solution

Program UNTYPE1.FOR

```
READ(5,*)YO,VO,DY,DQDT,X,G,IOUT
QO=YO*VO
CO=SQRT(G*YO)
DO 10 I=1,20
Y=YO-DY*FLOAT(I-1)
C=SQRT(G*Y)
V=2.*(C-CO)+VO
Q=Y*V
T1=(QO-Q)/DQDT
DXDT=V+C
```

```
      T=T1+X/DXDT
10    WRITE(IOUT,100)Y,C,V,Q,IFIX(T1+.5)
      *,DXDT,IFIX(T+.5)
100   FORMAT(4F8.3,I8,F8.3,I8)
      END
```

```
Input: 6 4.003 5280 32.2 3
```

EXAMPLE PROBLEM 6.4

Uniform flow exists downstream from a gate as in the previous problem with $V_o = 4$ fps and $Y_o = 6$ ft when suddenly, the depth of flow in the channel at the gate is decreased at a rate of 1.5 ft/h for 3 h. On a 30 s increment determine the depth of flow, the velocity, and the flow rate q at a position x = 1500 ft downstream from the gate.

Solution

This problem is of type #3 as described above. Therefore, the uniform celerity is first determined; $c_o = \sqrt{gY_o} = 13.900$ fps, and therefore $dx/dt|_{t=0} = 17.900$ fps, and dividing this into x = 1500 ft gives a time $\Delta t = 83.8$ s for the time for the wave to first arrive at the given position. Therefore, starting with t = 90 Equation 6.21 is solved, i.e., the following equation is solved for different ts

$$\frac{3}{g}\frac{dt}{dY}c^3 + \frac{dt}{dY}\frac{(V_o - 2c_o)}{g}c^2 + 3\left(t - \frac{dt}{dY}Y_o\right)c + \left(t - \frac{dt}{dY}Y_o\right)(V_o - 2c_o) - x = 0$$

in which dt/dY = 3600/1.5 = 2400, $V_o = 4$, $Y_o = 6$, and $c_o = 13.900$, with the results given in the table below. (The equation can be effectively solved by using the computer program in Chapter 3 that implements Laguerre's method, by the Newton method, or by available software such as Mathcad, TK-Solver, or MATLAB.)

t (s)	c (fps)	Y (ft)	V (fps)	Q (cfs/ft)
90	13.897	5.997	3.994	23.95
120	13.882	5.985	3.996	23.92
150	13.853	5.973	3.936	23.51
180	13.853	5.960	3.908	23.29
—	—	—	—	—
3600	12.104	4.550	0.409	1.86
3630	12.088	4.538	0.376	1.71
—	—	—	—	—
7170	10.012	3.113	−3.777	−11.76
7200	9.993	3.101	−3.814	−11.82

Program UNTYPE4.FOR

```
      READ(*,*) Yo,Vo,DT,DTDY,x,G,TE
      co=SQRT(G*Yo)
      TS=x/(Vo+co)
      TS=DT*(IFIX(TS/DT)+1.)
      N=(TE-TS)/DT+.5
      C1=3.*DTDY/G
      VOC=Vo-2.*co
      C2=DTDY*VOC/G
      DYo=Yo*DTDY
      C=.95*co
      DO 10 I=0,N
      T=TS+DT*FLOAT(I)
```

```
      C3=T-DYo
      M=0
1     F=((C1*C+C2)*C+3.*C3)*C+VOC*C3-x
      M=M+1
      IF(MOD(M,2).EQ.0) GO TO 2
      F1=F
      CC=C
      C=1.005*C
      GO TO 1
2     DIF=(C-CC)*F1/(F-F1)
      C=CC-DIF
      IF(ABS(DIF).GT.1.E-6.AND. M.LT.30) GO TO 1
      Y=C*C/G
      V=2.*(C-co)+Vo
10    WRITE(3,100) IFIX(T),C,Y,V,V*Y
100   FORMAT(I5,3F8.3,F8.2)
      END
```

```
Input: 6 4 30 2400 1500 32.2 7200
```

The flow starts reversing itself at this position at about time t = 3970 s.

EXAMPLE PROBLEM 6.5

The flow in a rectangular channel has been uniform with Y_o = 5 ft and V_o = 3 fps. At time zero, the gate at the downstream end is suddenly opened further to increase the flow rate by dq/dt = 0.01 $(ft/s)^2$. Determine the depth and velocity at the down end of the channel as a function of time, and provide tables that give the depth, the velocity, and the flow rate in the channel at positions x = 500 ft and x = 1000 ft as a function of time. Use the Newton method implemented in computer programs to solve the problem.

Solution

This is a common problem that: (1) gives uniform flow conditions at t = 0, (2) gives a flow rate increased at the downstream end to cause a dispersive wave, and (3) asks that q, V, and Y at selected positions be found as a function of time. At the origin the following Equation needs to be solved by the Newton method.

$$f(c) = 2c^3 + (V_o - 2c_o)c^2 - gq = 0 \tag{1}$$

Note that in this problem, the origin is taken at the downstream end of the channel, and therefore V and q are negative since x is positive in the opposite direction than the direction of flow. For problems with the cause of the disturbance at the downstream end of the channel, it is common to take the positive x-axis in the opposite direction to that of the flow so that the C^+ characteristics are straight lines. However, as illustrated later, the x-axis can be taken in the direction of flow for a problem with downstream control making the C^- characteristics straight lines when a uniform flow exists initially. The program that accomplishes this solution at x = 0 along with the solution to the problem are given below.

Program EPRB6_5.FOR

```
      LOGICAL IFST
      WRITE(6,*)' Give: Vo,Yo,dq/dt,NT,DT,g,IOUT'
      READ(5,*) VO,YO,DQT,NT,DT,G,IOUT
      CO=SQRT(G*YO)
      QO=VO*YO
      VOC=VO-2.*CO
      C=CO
```

```
      C001=.001*CO
      DC=5.*C001
      DO 10 K=1,NT
      NCT=0
      T=DT*FLOAT(K-1)
      Q=QO-T*DQT
      C=C-DC
1     IFST=.FALSE.
2     F=(2.*C+VOC)*C*C-G*Q
      IF(IFST) GO TO 5
      F1=F
      C=C-C001
      IFST=.TRUE.
      GO TO 2
5     DIF=C001*F1/(F1-F)
      C=C-DIF+C001
      NCT=NCT+1
      IF(ABS(DIF).GT..0001.AND. NCT.LT.20) GO TO 1
      Y=C*C/G
      V=2.*C+VOC
10    WRITE(IOUT,100) IFIX(T),C,Y,V,Q
100   FORMAT(I5,3F10.3,F10.2)
      END
```

Solution at x = 0 ft (the origin)

Time (s)	c (fps)	Y (ft)	V (fps)	q (cfs/ft)
0	12.689	5.000	−3.000	−15.00
30	12.649	4.969	−3.079	−15.30
60	12.609	4.937	−3.160	−15.60
90	12.568	4.905	−3.241	−15.90
120	12.526	4.873	−3.324	−16.20
150	12.484	4.840	−3.409	−16.50
180	12.441	4.807	−3.495	−16.80
210	12.397	4.773	−3.583	−17.10
240	12.353	4.739	−3.672	−17.40
270	12.307	4.704	−3.763	−17.70
300	12.261	4.669	−3.856	−18.00
330	12.213	4.633	−3.950	−18.30
360	12.165	4.596	−4.047	−18.60
390	12.116	4.559	−4.146	−18.90
420	12.065	4.521	−4.247	−19.20
450	12.013	4.482	−4.351	−19.50
480	11.960	4.442	−4.457	−19.80
510	11.905	4.402	−4.566	−20.10
540	11.849	4.361	−4.678	−20.40
570	11.792	4.318	−4.794	−20.70
600	11.732	4.275	−4.913	−21.00

At another position, the total time t that is considered known, consists of

$$\text{(known) } t = t_1 + \Delta t$$

or

$$t = \frac{|q - q_o|}{|dq/dt|} + \frac{x}{dx/dt}$$

where $dx/dt = 3c(0, t) + V_o - 2c_o$ or

$$f(c) = \frac{|2c^3/g + (V_o - 2c_o)c^2/g - q_o|}{|dq/dt|} + \frac{x}{3c + V_o - 2c_o} - t = 0$$

as the equation that needs to be solved by the Newton method at positions 500 and 1000 ft. Note that for this problem, and most problems, V_o, V and q_o are negative. The computer program for solving for these requested quantities at x = 500 ft and x = 1000 ft (or at any other position) is given below. The solution at these two positions follows the computer listing.

Program EPRB6_5A.FOR

```
      LOGICAL IFST
      WRITE(6,*)' Give: Vo,Yo,x,dq/dt,NT,DT,g,IOUT'
      READ(5,*) VO,YO,X,DQT,NT,DT,G,IOUT
      CO=SQRT(G*YO)
      QO=ABS(VO*YO)
      VOC=VO-2.*CO
      DXO=VO+CO
      C=CO
      C001=.001*CO
      DC=5.*C001
      DO 10 K=1,NT
      NCT=0
      T=DT*FLOAT(K-1)
      IF(T*DXO.LT.X) THEN
      V=VO
      Y=YO
      ELSE
      C=C-DC
1     IFST=.FALSE.
2     F=(ABS(2.*C+VOC)*C*C/G-QO)/ABS(DQT)+X/(3.*C+VOC)-T
      IF(IFST) GO TO 5
      F1=F
      C=C-C001
      IFST=.TRUE.
      GO TO 2
5     DIF=C001*F1/(F1-F)
      C=C-DIF+C001
      NCT=NCT+1
      IF(ABS(DIF).GT..0001.AND. NCT.LT.20) GO TO 1
      Y=C*C/G
      V=2.*C+VOC
      ENDIF
      Q=V*Y
10    WRITE(IOUT,100) IFIX(T),C,Y,V,Q
100   FORMAT(I5,3F10.3,F10.2)
      END
```

Position x = 500 ft

Time (s)	c (fps)	Y (ft)	V (fps)	q (cfs/ft)
0	12.689	5.000	−3.000	−15.00
30	12.689	5.000	−3.000	−15.00
60	12.678	4.992	−3.022	−15.08
90	12.639	4.961	−3.099	−15.38
120	12.599	4.930	−3.178	−15.67
150	12.559	4.899	−3.259	−15.96
180	12.519	4.867	−3.340	−16.26
210	12.477	4.835	−3.423	−16.55
240	12.435	4.802	−3.507	−16.84
270	12.393	4.769	−3.592	−17.13
300	12.349	4.736	−3.679	−17.42
330	12.305	4.702	−3.767	−17.71
360	12.260	4.668	−3.857	−18.00
390	12.214	4.633	−3.949	−18.30
420	12.168	4.598	−4.042	−18.58
450	12.120	4.562	−4.137	−18.87
480	12.071	4.525	−4.234	−19.16
510	12.022	4.488	−4.333	−19.45
540	11.971	4.451	−4.434	−19.74
570	11.920	4.412	−4.538	−20.02
600	11.867	4.373	−4.644	−20.31

Position x = 1000 ft

Time (s)	c (fps)	Y (ft)	V (fps)	q (cfs/ft)
0	12.689	5.000	−3.000	−15.00
30	12.689	5.000	−3.000	−15.00
60	12.689	5.000	−3.000	−15.00
90	12.689	5.000	−3.000	−15.00
120	12.667	4.983	−3.042	−15.16
150	12.629	4.953	−3.119	−15.45
180	12.590	4.923	−3.196	−15.74
210	12.551	4.892	−3.275	−16.02
240	12.511	4.861	−3.355	−16.31
270	12.471	4.830	−3.436	−16.59
300	12.430	4.798	−3.518	−16.88
330	12.388	4.766	−3.601	−17.16
360	12.346	4.734	−3.685	−17.45
390	12.303	4.701	−3.771	−17.73
420	12.259	4.667	−3.859	−18.01
450	12.215	4.634	−3.947	−18.29
480	12.170	4.600	−4.037	−18.57
510	12.124	4.565	−4.129	−18.85
540	12.077	4.530	−4.222	−19.13
570	12.030	4.494	−4.317	−19.40
600	11.982	4.458	−4.414	−19.68

EXAMPLE PROBLEM 6.6

A 4 m wide rectangular channel is supplied by a reservoir whose water surface elevation is H = 2 m above the channel bottom. The channel has a bottom slope S_o = .0006, and n = 0.014. The entrance loss coefficient is K_e = 0.2. At time zero, the head of the reservoir begins to decrease at a of rate dH/dt = 0.005 m/s. Determine the depth, the velocity, and the flow rate per unit width at the beginning of the channel and at positions x = 100, 200 and 300 m downstream.

Solution

Equation 6.20 provides a means for solving the celerity, the depth, and the velocity at the origin, x = 0. However, a more general equation for any position can be obtained from noting that the time $t = t_1 + \Delta t$ or,

$$t = \frac{H_o - H}{|dH/dt|} + \frac{x}{3c + V_o - 2c_o} \quad \text{in which } H = \frac{c^2}{g} + \frac{(1+K_e)}{2g}(2c + V_o - 2c_o)^2$$

Note, when x = 0, this equation reduces to Equation 6.20 that applies at the origin. The program below implements the Newton method to solve the above equation for any position x. The solution to the problem is given in the tables that follow.

Listing of program HVARYX.FOR to solve dispersive waves in which H decreases

```
      REAL X[ALLOCATABLE](:)
      COMMON G,EK,Voc,XX,Ho,dHdt,TIME
      WRITE(*,*)' Give:IOUT,Yo,Vo,Ho,dHdt,','Ke,g,NT,DT,NS'
      READ(*,*) IOUT,Yo,Vo,Ho,dHdt,EK,G,NT,IDT,NS
      ALLOCATE(X(NS))
      WRITE(*,*)' Give',NS,' positions for x'
      READ(*,*) X
      co=SQRT(G*Yo)
      dcdt=SQRT(G)*ABS(dHdt)
      Voc=Vo-2.*co
      Voco=Vo+co
      EK=(1.+EK)/2.
      DO 50 I=1,NS
      c=co
      XX=X(I)
      WRITE(IOUT,90) I,XX
90    FORMAT(' At station',I3,', x =',F8.1,/,' Time  H  c  Y  V',,
     &'  q',/,1x,45('_'))
      TIM1=XX/Voco
      II=IFIX(TIM1)/IDT
      WRITE(IOUT,100)(K*IDT,Ho,c,c*c/G,2.*c+Voc,c*c/G*(2.*c+Voc),
     &K=0,II)
      DO 50 K=II+1,NT
      TIME=K*IDT
      M=0
      c=c-dcdt*(TIME-TIM1)
10    F=FUN(c)
      DIF=F/((F-FUN(.995*c))/(.005*c))
      c=c-DIF
      M=M+1
      IF(ABS(DIF).GT.1.E-5.AND.M.LT.20)GO TO 10
      IF(M.EQ.20)WRITE(*,*)'Failed to converge',K,DIF
      H=(c*c+EK*(2.*C+Voc)**2)/G
      WRITE(IOUT,100) K*IDT,H,c,c*c/G,2.*c+Voc,c*c/G*(2.*c+Voc)
50    TIM1=TIME
100   FORMAT(I5,5F8.3)
      END
```

```
      FUNCTION FUN(c)
      COMMON G,EK,Voc,XX,Ho,dHdt,TIME
      H=(c*c+EK*(2.*c+Voc)**2)/G
      FUN=TIME-ABS(Ho-H)/ABS(dHdt)-XX/(3.*c+Voc)
      RETURN
      END
```

A program that only solves for conditions at the original may consist of:

Listing of program HVARY.FOR

```
      FUN(G,H,EK,Voc,c)=G*H-c*c-EK*2.*c+Voc)**2
      WRITE(*,*)'Give:IOUT,Yo,Vo,Ho,dHdt,Ke,g,NT,DT'
      READ(*,*) IOUT,Yo,Vo,Ho,dHdt,EK,G,NT,IDT
      co=SQRT(G*Yo)
      Voc=Vo-2.*co
      c=co
      EK=(1.+EK)/2.
      WRITE(IOUT,100) 0,Ho,c,c*c/G,2.*c+Voc,c*c/G*(2.*c+Voc)
      DO 50 K=1,NT
      M=0
      H=Ho-FLOAT(K*IDT)*dHdt
      c=.98*c
10    F=FUN(G,H,EK,Voc,c)
      DIF=F/((FUN(G,H,EK,Voc,1.01*c)-F)/(.01*c))
      c=c-DIF
      M=M+1
      IF(ABS(DIF).GT.1.E-5.AND.M.LT.20)GO TO 10
      IF(M.EQ.20)WRITE(*,*)'Failed to converge',K,DIF
50    WRITE(IOUT,100) K*IDT,H,c,c*c/G,2.*c+Voc,c*c/G*(2.*c+Voc)
100   FORMAT(I5,5F8.3)
      END
```

Solution obtained from HVARYX

At station 1, x = 0.0

Time	H	c	Y	V	q
0	2.000	4.230	1.824	1.692	3.086
10	1.950	4.191	1.791	1.614	2.891
20	1.900	4.151	1.756	1.533	2.693
30	1.850	4.109	1.721	1.450	2.497
40	1.800	4.067	1.686	1.366	2.302
50	1.750	4.023	1.650	1.278	2.109
60	1.700	3.979	1.614	1.189	1.918
70	1.650	3.933	1.576	1.097	1.729
80	1.600	3.885	1.539	1.002	1.542
90	1.550	3.836	1.500	0.904	1.356
100	1.500	3.785	1.461	0.802	1.172

At station 2, x = 100.0

Time	H	c	Y	V	q
0	2.000	4.230	1.824	1.692	3.086
10	2.000	4.230	1.824	1.692	3.086
20	1.985	4.219	1.814	1.670	3.029

(continued)

Time	H	c	Y	V	q
30	1.937	4.180	1.781	1.593	2.837
40	1.888	4.141	1.748	1.514	2.647
50	1.840	4.101	1.715	1.434	2.459
60	1.792	4.060	1.681	1.352	2.273
70	1.745	4.018	1.646	1.269	2.089
80	1.697	3.976	1.611	1.183	1.907
90	1.649	3.932	1.576	1.096	1.727
100	1.602	3.887	1.540	1.006	1.550

At station 3, x = 200.0

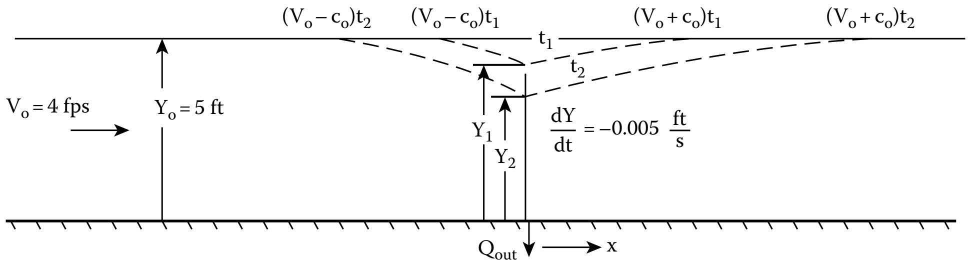
Time	H	c	Y	V	q
0	2.000	4.230	1.824	1.692	3.086
10	2.000	4.230	1.824	1.692	3.086
20	2.000	4.230	1.824	1.692	3.086
30	2.000	4.230	1.824	1.692	3.086
40	1.971	4.208	1.805	1.647	2.973
50	1.924	4.170	1.773	1.573	2.788
60	1.878	4.132	1.741	1.497	2.605
70	1.831	4.094	1.708	1.419	2.424
80	1.785	4.054	1.676	1.340	2.246
90	1.740	4.014	1.642	1.260	2.070
100	1.694	3.973	1.609	1.178	1.816

At station 4, x = 300.0

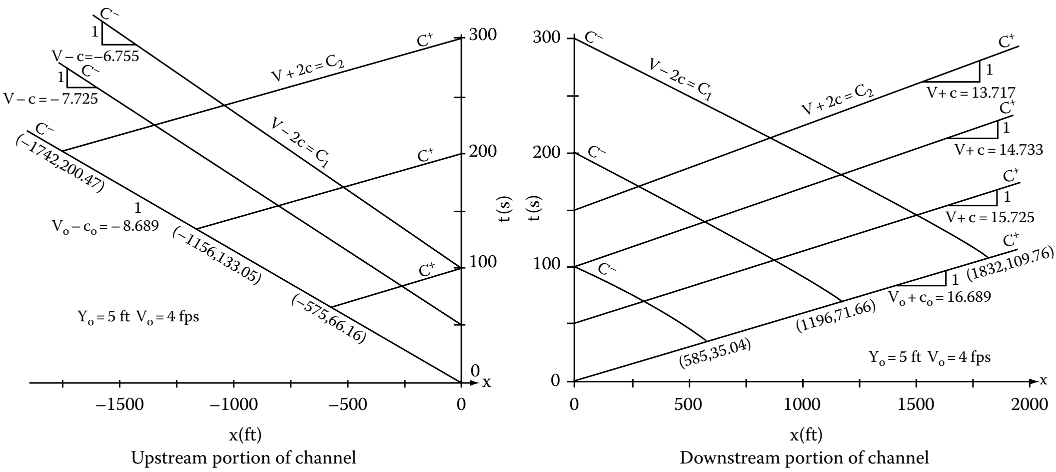
Time	H	c	Y	V	q
0	2.000	4.230	1.824	1.692	3.086
10	2.000	4.230	1.824	1.692	3.086
20	2.000	4.230	1.824	1.692	3.086
30	2.000	4.230	1.824	1.692	3.086
40	2.000	4.230	1.824	1.692	3.086
50	2.000	4.230	1.824	1.692	3.086
60	1.958	4.197	1.796	1.626	2.921
70	1.912	4.161	1.765	1.554	2.742
80	1.868	4.124	1.734	1.480	2.566
90	1.823	4.087	1.702	1.405	2.392
100	1.779	4.049	1.671	1.329	2.221

6.6.5 Understanding Characteristics Better

The understanding of both the positive and the negative characteristics is vital in solving problems in this chapter, and in the next chapter, where numerical solutions of the St. Venant equations will be covered. The following few pages are intended to enhance this understanding. Consider a channel, as shown in the sketch below, that contains a uniform flow at a depth Y_o and a velocity V_o (and a unit flow rate $q_o = Y_oV_o$) prior to the time when at a position somewhere near the middle of this channel, a time varying outflow is started at time zero. To simplify the problem, to begin with, assume that the outflow causes a linear change in depth with time at the position where it occurs, i.e., at x = 0, a rate of depth change with time dY/dt is specified and is a negative constant. We will consider the case where dY/dt is negative, so as to cause negative waves both upstream and downstream from this position, but as long as dY/dt is not so large (or lasts so long) as to result in a surge, as described later, the equations, etc., that follow, apply for increasing as well as decreasing depths. As equations, etc., are given below, the numbers associated with the problem in which $Y_o = 5$ ft ($c_o = 12.689$ fps), $V_o = 4$ ft/s, and dY/dt = −0.005 ft/s will generally also be given.



Notice that, the effect of decreasing the depth at x = 0 propagates both upstream and downstream. The downstream rate of propagation is $(dx/dt)_o^+ = V_o + c_o = 4 + 12.689 = 16.689$ fps since the disturbance moves with a velocity c_o and it is carried downstream with a velocity V_o. The rate of upstream propagation, however, is $(dx/dt)_o^- = V_o - c_o = 4 - 12.689 = -8.689$ fps. For a subcritical flow, as we are restricting our considerations here, $V_o - c_o$ is negative since c_o is larger in magnitude than V_o. Since the movement is opposite to the direction of the positive x axis that is taken in the direction of the channel flow for both the upstream and the downstream portions of the channel with $(dx/dt)_o^-$ negative for the upstream portion of the channel we get the proper sign for x, i.e., as time increases the position $x = t(dx/dt)_o^-$ of the disturbance is negative. For example, at time t = 100 s the upstream position of the disturbance is at x = 100(−8.689) = −868.9 ft, and the downstream position of the disturbance is at x = 100(16.689) = 1668.9 ft. The (+) or (−) superscript on (dx/dt), which has now been added, denotes whether its value is positive or negative, respectively, and this notation is consistent with the designation of the positive or the negative characteristics. It also helps because now, we are dealing with four characteristics; two for both the downstream and the upstream portions of the channel. Since in developing the equations, we called the characteristics with an inverse slope V + c, the C^+ (positive characteristics), and those with an inverse slope V − c the C^- (negative characteristics), the xt-planes for the upstream and the downstream portions of the channel will be as shown in the two graphs below. As before, the downstream portion of the channel, and the C^+ characteristics were straight lines, since the inverse slope of the initial characteristic of this family was $(dx/dt)_o^+ = V_o + c_o = 16.689$ fps. However, now the C^- characteristics are straight lines for the upstream portion of the channel, and the C^+ characteristics are not straight lines. In this upstream portion of the channel, the inverse slope of the initial C^- characteristic is $(dx/dt)_o^- = V_o - c_o = -8.689$ fps. The figures below display both the positive and the negative characteristics for both the upstream and the downstream portions of the channel. In both these graphs, the given characteristics apply for the problem with $Y_o = 5$ ft and $V_o = 4$ fps. The C^- characteristics in the graph for the downstream portion of the channel and the C^+ characteristics in the graph for the upstream portion of the channel have been obtained by numerically solving the differential equations $(dx/dt)_d^- = V - c$ and $(dx/dt)_u^+ = V + c$, respectively, as described subsequently. (*Note*: The added subscripts d and u denote the downstream and the upstream propagations, respectively.)



It is pertinent to examine why the C^- characteristics for the downstream portion, and the C^+ characteristics for the upstream portion, of the channel are not straight lines. Note, in both cases, these characteristics exist in the portion of the channel in which the depth is less than Y_o, thus $c < c_o$. As just noted, the inverse slope of the C^- characteristics within the downstream portion of the channel is given by $(dx/dt)_d^- = V - c$, and have a negative magnitude, since the magnitude of c is larger than the magnitude of V. With c decreasing with time, the inverse slope dx/dt will change. The inverse slope of the C^+ characteristics within the upstream portion of the channel is given by $(dx/dt)_u^+ = V + c$, and are positive. Again, with c changing, the slope dx/dt at any position, (x, t) will also change. As discussed earlier, the C^+ characteristics for the downstream portion of the channel are straight lines, since $(dx/dt)_o^+ = V_o + c_o$ (a constant) is the inverse slope of the first such characteristics, and we have proven earlier that if one line is straight, all other lines of that family must be straight. For the upstream portion of the channel, the inverse slope of the initial C^- characteristic is $(dx/dt)_o^- = V_o - c_o$ (a constant), and consequently all of the negative characteristics for the upstream portion of the channel will also be straight lines. For the downstream portion of the channel, the boundary conditions are as given previously by Equations 6.15 through 6.20. However, for the upstream portion of the channel, the boundary conditions will be different, and must be obtained by noting that when we move along a C^+ characteristic V + 2c is constant, and therefore $V(0, t) + 2c(0, t) = V_o + 2c_o$, or

$$V(0,t) = V_o + 2c_o - 2c(0,t) = 29.377 - 2c(0,t)$$

The inverse slope of the C^- characteristics, which for this upstream portion of the channel are $dx/dt = V - c$, are given by

$$(dx/dt)_u^- = V - c = V_o + 2c_o - 3c(0,t) = V_o + 2c_o - 3(gY)^{1/2} = 29.377 - 3c(0,t) = 29.377 - 3(gY(0,t))^{1/2}$$

This becomes the boundary condition for the upstream portion of the channel if the depth is specified as a function of time at x = 0. If the velocity V(0, t) is specified as a function of time, the boundary condition is

$$V(0,t) = \frac{V_o}{2} + c_o - \frac{V}{2} = 14.689 - .5V(0,t) \quad \text{and} \quad \left(\frac{dx}{dt}\right)_u^- = \frac{V(0,t)}{2} - \left(\frac{V_o}{2} + c_o\right) = 0.5V(0,t) - 14.689$$

Since, q(0, t) = VY = (V_o + $2c_o$ − 2c(0, t))c(0, t)^2/g, the boundary condition, if q(0, t) is specified, becomes the following cubic equation in c(0, t), which must be solved for the appropriate root.

$$gq(0,t) = \{V_o + 2c_o - 2c(0,t)\}c(0,t)^2 = \{29.377 - 2c\}c^2$$

Below are tables that solve the unsteady flow in both the downstream and the upstream portions of the channel for the problem given above in which the depth at the origin has been specified to decrease at a rate of dY/dt = −0.005 ft/s. At the origin (x = 0), the depths for the downstream and the upstream portions of the channel are the same corresponding to any given time. However, the velocities in the downstream portion of the channel decrease with time, whereas the velocities in the upstream portion of the channel increase with time. Thus, the unit flow rate at x = 0 for the upstream portion q_u is larger than q_d for the downstream portion of the channel. The difference in q = s is the outflow Δq that must occur at x = 0 at the given time, i.e., $\Delta q = q_u - q_d$. We are assuming this outflow Δq is a point outflow, but of course in practice, it must take place over some finite length of channel. You should study the results in these two solution tables to fully appreciate what is happening and why. See a homework problem to write computer programs to generate these solution tables. For example, the depth at x = 0 for both the upstream and the downstream portions of the channel at t = 100 s is Y = 5 − 100(.005) = 4.5 ft, so c = (32.2 × 4.5)^2 =12.037 fps. For the downstream channel at x = 0 and t = 100 s, $V_d = 2c + V_o - 2c_o$ = 2(12.037) − 21.377 = 2.698 fps, and the unit flow rate q_d = VY = 12.139 cfs/ft. The slope of the C^+ characteristics in the downstream channel $(dx/dt)_d^+ = V + c$ so at t = 100 s, $(dx/dt)_d^+ = 14.735$ fps. At position x = 1000 ft and at t = 100 s, the equation $t = t_1 + \Delta t = (5 - c^2/g)/.005 + 1000/(3c + V_o - 2c_o)$ must be solved for c giving c = 12.449 fps. So, Y = 12.449^2/32.2 = 4.813 ft, and V = 2c − 21.377 = 3.521 fps. Finally, the slope of the C^+ characteristic through (1000, 100) is $(dx/dt)^+$ = V + c = 3.521 + 12.449 = 15.970 fps.

Time-dependent changes downstream of the falling depth dY/dt = −0.005 fps.

	x = 0 (Origin)					x = 1000 ft				
t (s)	**Y (ft)**	**c (fps)**	**V (fps)**	**dx/dt**	**q (sfps)**	**Y (ft)**	**c (fps)**	**V (fps)**	**dx/dt**	**q (sfps)**
0	5.0000	12.6886	4.0000	16.6886	20.0000					
10	4.9500	12.6250	3.8728	16.4978	19.1703					
20	4.9000	12.5611	3.7449	16.3060	18.3502					
30	4.8500	12.4968	3.6164	16.1132	17.5398	No change until $t = x/(V_o + c_o)$ = 1000/(4 + 12.6886) = 59.92 sec				
40	4.8000	12.4322	3.4873	15.9195	16.7389					
50	4.7500	12.3673	3.3574	15.7247	15.9478					
60	4.7000	12.3020	3.2269	15.5289	15.1665	4.9996	12.6881	3.9991	16.6872	19.9939
70	4.6500	12.2364	3.0957	15.3321	14.3949	4.9529	12.6286	3.8801	16.5087	19.2177
80	4.6000	12.1705	2.9638	15.1342	13.6333	4.9062	12.5690	3.7608	16.3298	18.2177
90	4.5500	12.1041	2.8311	14.9352	12.8815	4.8596	12.5092	3.6411	16.1503	17.6945
100	4.5000	12.0374	2.6977	14.7352	12.1398	4.8131	12.4491	3.5211	15.9703	16.9475
110	4.4500	11.9704	2.5636	14.5340	11.4080	4.7667	12.3890	3.4008	15.7897	16.2104
120	4.4000	11.9029	2.4287	14.3317	10.6864	4.7203	12.3286	3.2801	15.6087	15.4831
130	4.3500	11.8351	2.2931	14.1282	9.9749	4.6741	12.2681	3.1590	15.4271	14.7657
140	4.3000	11.7669	2.1567	13.9236	9.2736	4.6280	12.2074	3.0377	15.2451	14.0582
150	4.2500	11.6983	2.0194	13.7177	8.5826	4.5820	12.1466	2.9160	15.0625	13.3608
160	4.2000	11.6293	1.8814	13.5107	7.9018	4.5360	12.0855	2.7939	14.8795	12.6733
170	4.1500	11.5598	1.7425	13.3024	7.2315	4.4902	12.0244	2.6716	14.6959	11.9960
180	4.1000	11.4900	1.6028	13.0928	6.5716	4.4445	11.9630	2.5489	14.5120	11.3288
190	4.0500	11.4197	1.4623	12.8820	5.9223	4.3990	11.9016	2.4260	14.3275	10.6718
200	4.0000	11.3490	1.3209	12.6699	5.2835	4.3535	11.8399	2.3027	14.1426	10.0249

Time-dependent changes upstream of the falling depth dY/dt = −0.005 fps.

	x = 0 (Origin)					x = −1000 ft				
t (s)	Y (ft)	c (fps)	V (fps)	dx/dt	q (sfps)	Y (ft)	c (fps)	V (fps)	dx/dt	q (sfps)
0	5.0000	12.6886	4.0000	−8.6886	20.0000					
10	4.9500	12.6250	4.1272	−8.4978	20.4297					
20	4.9000	12.5611	4.2551	−8.3060	20.8498					
30	4.8500	12.4968	4.3836	−8.1132	21.2602					
40	4.8000	12.4322	4.5127	−7.9195	21.6611					
50	4.7500	12.3673	4.6426	−7.7247	22.0522					
60	4.7000	12.3020	4.7731	−7.5289	22.4335					
70	4.6500	12.2364	4.9043	−7.3321	22.8051					
80	4.6000	12.1705	5.0362	−7.1342	23.1667	No change until $t = x/(V_o\text{-}c_o) = -1000/(4 - 12.6886) = 115.09$ sec				
90	4.5500	12.1041	5.1689	−6.9352	23.5185					
100	4.5000	12.0374	5.3023	−6.7352	23.8602					
110	4.4500	11.9704	5.4364	−6.5340	24.1920					
120	4.4000	11.9029	5.5713	−6.3317	24.5136	4.9804	12.6637	4.0497	−8.6141	20.1691
130	4.3500	11.8351	5.7069	−6.1282	24.8251	4.9408	12.6133	4.1506	−8.4627	20.5074
140	4.3000	11.7669	5.8433	−5.9236	25.1264	4.9015	12.5630	4.2511	−8.3119	20.8369
150	4.2500	11.6983	5.9806	−5.7177	25.4174	4.8626	12.5130	4.3511	−8.1619	21.5074
160	4.2000	11.6293	6.1186	−5.5107	25.6982	4.8240	12.4633	4.4506	−8.0127	21.4698
170	4.1500	11.5598	6.2575	−5.3024	25.9685	4.7858	12.4138	4.5496	−7.8642	21.7732
180	4.1000	11.4900	6.3972	−5.0928	26.2284	4.7479	12.3646	4.6479	−7.7167	22.0680
190	4.0500	11.4197	6.5377	−4.8820	26.4778	4.7105	12.3158	4.7456	−7.5701	22.3543
200	4.0000	11.3490	6.6791	−4.6699	26.7166	4.6734	12.2672	4.8427	−7.4245	22.6321

For the upstream portion of the channel at position x = −1000 ft, the wave has not reached here at t = 100 s, i.e., it has traveled to position $100(V_o - c_o) = 100(-8.689) = -868.9$ ft. Therefore, let us solve the conditions at t = 150 s. At time t = 150 s, the following apply: First, the cubic equation $150 = (5 - c^2/g)/.005 - 1000/(V_o + 2c_o - 3c)$ is solved for c = 12.513 fps. The depth $Y = (12.513)^2/32.2 = 4.863$ ft, and the velocity $V = V_o + 2c_o - 2c = 29.377 - 2(12.513) = 4.351$ fps. The unit flow rate is $q = VY = 4.351 \times 4.863 = 21.159$ cfs/ft, and the slope of the straight line C^- characteristic through this point (−1000, 150) is $(dx/dt)_u = V - c = 4.351 - 12.513 = -8.162$ fps. At x = 0 ft and t = 100 s, the upstream unit flow rate is $q_u = V_u Y_u = 5.302(4.5) = 23.859$ cfs/ft and the unit outflow at this time equals $\Delta q = 23.859 - 12.139 = 11.720$ cfs/ft.

The C^- characteristics for the downstream portion of the channel, and the C^+ characteristics for the upstream portion of the channel as given in the above graphs, were obtained by numerically solving the ordinary differential equations $(dx/dt)_d^- = V - c$ and $(dx/dt)_d^+ = V + c$, respectively. An explanation of how this is done is given below.

First, consider solving the C^- characteristics for the downstream portion of the channel by numerically solving the ODE, (dx/dt) = V − c. (dx/dt) gives the inverse slope of these characteristics at any point (x, t). Since in this downstream portion of the channel the C^+ characteristics are straight lines, it is possible to solve c at any point with the equation $t = t_1 + \Delta t = |(Y_o - c^2/g)|/|dY/dt| + x/(3c + V_o - 2c_o)$. Upon solving c, the velocity at this point can be obtained from $V = 2c + V_o - 2c_o$, and the inverse slope $(dx/dt)_d^- = V - c$ (a negative value for subcritical flow). The table below gives the results from numerically solving the ODE, $(dx/dt)_d^- = V - c$ starting at t = 300 s and decreasing t by 10 s until this C^- characteristic intersects with the initial C^+ characteristic for the downstream portion of the channel. (See a homework problem to write a program to generate this table.)

This table applies to the downstream portion of the channel. The ODE dx/dt = V-c is solved numerically to get x for each t.

t (s)	x (ft)	c (fps)	Y (ft)	V (fps)	q (cfs/ft)	dx/dt (fps)
300	0.00	10.616	3.500	−.145	−.508	−10.761
290	106.86	10.765	3.599	0.153	0.550	−10.612
280	212.27	10.906	3.694	0.434	1.604	−10.471
270	316.31	11.039	3.785	0.702	2.655	−10.338
260	419.05	11.167	3.873	0.957	3.707	−10.210
250	520.53	11.290	3.958	1.202	4.759	−10.087
240	620.81	11.408	4.042	1.439	5.815	−9.969
230	719.93	11.522	4.123	1.667	6.873	−9.855
220	817.92	11.633	4.202	1.888	7.935	−9.744
210	914.83	11.740	4.280	2.103	9.002	−9.637
200	1010.67	11.845	4.357	2.312	10.073	−9.533
190	1105.49	11.946	4.432	2.516	11.150	−9.431
180	1199.30	12.046	4.506	2.714	12.232	−9.331
170	1292.12	12.143	4.579	2.909	13.320	−9.234
160	1383.99	12.238	4.651	3.099	14.413	−9.139
150	1474.91	12.331	4.722	3.285	15.513	−9.046
140	1564.91	12.422	4.792	3.468	16.618	−8.955
130	1654.01	12.512	4.862	3.647	17.730	−8.865
120	1742.22	12.600	4.930	3.823	18.848	−8.777
110	1829.56	12.686	4.998	3.996	19.973	−8.691
109.76	1831.67	12.689	5.000	4.000	20.000	−8.689

That these C characteristics for the downstream portion of the channel are not straight lines is shown by the fact that dx/dt is not constant. Since the value of dx/dt decreases with time, these C characteristics are concave downward. It is also worth noting that the velocity V decreases with increasing time (and with decreasing position x), so that for this example problem, it is negative (−0.145 fps) at (0, 300).

Now, consider solving the C^+ characteristics for the upstream portion of the channel by numerically solving the ODE, $(dx/dt)_u^+ = V + c$. In this upstream portion of the channel, the C characteristics are straight lines, and so by utilizing them it is possible to solve c at any point in the xt plane with the implicit equation $t = t_1 + \Delta t = |(Y_o - c^2/g)|/|dY/dt| + |x|/|(V_o + 2c_o - 3c)|$. Using this solved c, the velocity at this point (x, t) is given by $V = V_o + 2c_o - 2c$. The table below provides the results of this numerical solution for the C^+ characteristics that starts on the t-axis at 300 s, and goes to where it intersects the initial C characteristic. (See a homework problem to write a program to generate this table.)

This table applies to the upstream portion of the channel. The ODE dx/dt = V + c is solved numerically to get x for each t.

t (s)	x (ft)	c (fps)	Y (ft)	V (fps)	q (cfs/ft)	dx/dt (fps)
300	.00	10.616	3.500	8.145	28.508	18.761
290	−185.19	11.054	3.795	7.269	27.585	18.323
280	−366.92	11.342	3.995	6.694	26.741	18.036
270	−546.08	11.575	4.161	6.228	25.912	17.802
260	−723.07	11.777	4.307	5.823	25.083	17.600
250	−898.15	11.958	4.441	5.460	24.250	17.419
240	−1071.49	12.125	4.566	5.127	23.410	17.252
230	−1243.23	12.280	4.683	4.818	22.561	17.097
220	−1413.47	12.425	4.795	4.526	21.703	16.952
210	−1582.29	12.563	4.902	4.251	20.835	16.814
200.47	−1741.83	12.689	5.000	4.000	20.000	16.689

The slope of these C^+ characteristics is positive as shown by the last column of this table. The fact that dx/dt is not constant indicates that these characteristics are not straight lines, and are concave upward in the xt plane.

The above discussion assumed that the depth was decreased at x = 0 as a function of time. In practice, the unit outflow Δq will more likely be known as a function of time. In other words, the difference between the upstream and the downstream unit flow rates will be specified to change with time, i.e., $\Delta q = q_u - q_d$ will be known for any time t. The boundary condition at x = 0, can therefore be obtained by replacing the upstream and the downstream unit flow rates by the proper velocities and the depths expressed in terms of the celerity c at x = 0, or $\Delta q = (V_o + 2c_o - 2c)c^2/g - (2c + V_o - 2c_o)c^2/g$ that simplifies to the following cubic equation for the boundary condition in which Δq is given: $g\dfrac{\Delta q}{4} + (c - c_0)c^2 = 0$. (Note that (0, t) could be added to c to denote at x = 0 and any time, t.)

The table below provides the solution to the above problem with $Y_o = 5$ ft and $V_o = 4$ fps, in which the outflow Δq has been specified to increase at a constant rate of $d(\Delta q)/dt = 0.1$ cfs/ft/s for 100 s, and thereafter remains constant at $\Delta q = 10$ cfs/ft, i.e., one-half the initial unit flow rate is outflow at x = 0. Therefore, this table provides the conditions at the origin and at two positions downstream and two positions upstream. It will be a useful exercise for you to write a computer program to generate this table of values as requested in a homework problem at the end of this chapter. It is an interesting, and a significant, fact from the above equation $g\Delta q/4 + (c - c_o)c^2 = 0$ that the celerity c(0, t) (and depth) depends only upon Δq and the initial celerity c_o, and not on time t (or the rate of change of outflow $(d(\Delta q)/dt)$. The affected lengths of the upstream and downstream portions of the channel are dependent upon the outflow $(d(\Delta q)/dt)$. The affected lengths increase with time, and the associated expanding volumes must supply the outflow. Since c(0, t) is only a function of Δq and c_o, it is well to consider non-dimensionalizing this equation by dividing both sides of it by the celerity c_c associated with critical flow of the initial unity flow rate q_o. This critical flow will have a depth $Y_c = (q_o^2/g)^{1/3} = [(Y_oV_o)^2/g]^{1/3}$, and also because $c_c = (gY_c)^2$, $c_c = (gq_o)^{1/3}$, or $c_c^3 = gq_o$. Dividing the equation for the depth at the origin by c_c^3 gives

$$\frac{g\Delta q}{4gq_0} + \frac{(c - c_o)c^2}{c_c^3} = 0$$

The table giving the unsteady solution when the unit outflow rate Δq is changed by d(q)/dt = 0.1 cfs/ft/s

	At the Origin, x = 0						
t (s)	Δq (fss)	c (fps)	Y (ft)	V-dow (fps)	q-dow (fss)	q-up (fss)	V-up (fps)
0	0	12.689	5.000	4.000	20.000	20.000	4.000
10	1	12.638	4.960	3.899	19.341	20.341	4.101
20	2	12.587	4.920	3.797	18.681	20.681	4.203
30	3	12.535	4.880	3.693	18.018	21.018	4.307
40	4	12.482	4.838	3.587	17.354	21.354	4.413
50	5	12.428	4.797	3.479	16.687	21.687	4.521
60	6	12.373	4.754	3.369	16.018	22.018	4.631
70	7	12.317	4.712	3.257	15.346	22.346	4.743
80	8	12.260	4.668	3.143	14.672	22.672	4.857
90	9	12.202	4.624	3.027	13.995	22.995	4.973
100	10	12.143	4.579	2.908	13.316	23.316	5.092
110							
120							
130	No change since t > t_2 = 100, i.e. del q = 10 (ft/s)**2						
140							
150							
160							
170							
180							
190							
200							

	At x = 500 ft				At x = 1000 ft			
t (s)	c (fps)	Y (ft)	V (fps)	q (fss)	c (fps)	Y (ft)	V (fps)	q (fss)
0	12.831	5.113	4.286	21.913	12.962	5.218	4.547	23.728
10	12.784	5.076	4.192	21.276	12.918	5.182	4.459	23.107
20	12.737	5.038	4.096	20.637	12.873	5.147	4.369	22.486
30	12.688	5.000	4.000	19.997	12.828	5.110	4.278	21.864
40	12.639	4.961	3.902	19.356	12.782	5.074	4.187	21.242
50	12.590	4.922	3.802	18.714	12.735	5.037	4.094	20.619
60	12.539	4.883	3.701	18.071	12.688	5.000	3.999	19.995
70	12.488	4.843	3.598	17.426	12.640	4.962	3.904	19.371
80	12.436	4.803	3.494	16.780	12.592	4.924	3.807	18.746
90	12.383	4.762	3.388	16.132	12.543	4.886	3.709	18.120
100	12.329	4.720	3.280	15.483	12.493	4.847	3.609	17.494
110	12.274	4.678	3.170	14.832	12.443	4.808	3.508	16.867
120	12.218	4.636	3.059	14.180	12.391	4.768	3.406	16.239
130	12.161	4.593	2.945	13.527	12.339	4.729	3.301	15.611
140	12.143	4.579	2.908	13.316	12.286	4.688	3.196	14.982
150	12.143	4.579	2.908	13.316	12.233	4.647	3.088	14.352
160	12.143	4.579	2.908	13.316	12.178	4.606	2.979	13.722
170	12.143	4.579	2.908	13.316	12.143	4.579	2.908	13.316
180	12.143	4.579	2.908	13.316	12.143	4.579	2.908	13.316
190	12.143	4.579	2.908	13.316	12.143	4.579	2.908	13.316
200	12.143	4.579	2.908	13.316	12.143	4.579	2.908	13.316

(continued)

	At x = −500 ft				At x = −1000 ft					
t (s)	**c (fps)**	**Y (ft)**	**V (fps)**	**q (fss)**	**c (fps)**	**Y (ft)**	**V (fps)**	**q (fss)**	**x-do (ft)**	**x-up (ft)**
0	12.943	5.202	3.492	18.164	13.151	5.371	3.076	16.520	0	0
10	12.900	5.168	3.578	18.489	13.112	5.339	3.153	16.836	167	−87
20	12.856	5.133	3.665	18.812	13.073	5.308	3.231	17.150	334	−174
30	12.812	5.098	3.753	19.132	13.034	5.276	3.310	17.461	501	−261
40	12.768	5.063	3.842	19.449	12.994	5.244	3.389	17.770	668	−348
50	12.723	5.027	3.932	19.764	12.954	5.212	3.468	18.076	834	−434
60	12.677	4.991	4.022	20.076	12.914	5.179	3.548	18.379	1001	−521
70	12.632	4.955	4.114	20.386	12.874	5.147	3.629	18.680	1168	−608
80	12.585	4.919	4.207	20.692	12.833	5.115	3.710	18.978	1335	−695
90	12.539	4.882	4.300	20.995	12.792	5.082	3.792	19.273	1502	−782
100	12.491	4.846	4.395	21.295	12.751	5.050	3.875	19.565	1669	−869
110	12.444	4.809	4.490	21.592	12.710	5.017	3.958	19.854	1836	−956
120	12.395	4.772	4.586	21.885	12.668	4.984	4.041	20.140	2003	−1043
130	12.347	4.734	4.684	22.174	12.626	4.951	4.125	20.422	2170	−1130
140	12.297	4.697	4.782	22.460	12.584	4.918	4.209	20.701	2336	−1216
150	12.248	4.659	4.881	22.741	12.541	4.885	4.294	20.976	2503	−1303
160	12.198	4.621	4.982	23.018	12.499	4.852	4.380	21.248	2670	−1390
170	12.147	4.582	5.083	23.291	12.456	4.818	4.465	21.516	2837	−1477
180	12.143	4.579	5.092	23.316	12.413	4.785	4.552	21.780	3004	−1564
190	12.143	4.579	5.092	23.316	12.369	4.752	4.638	22.039	3171	−1651
200	12.143	4.579	5.092	23.316	12.326	4.718	4.725	22.295	3338	−1738

If we define $c' = c/c_c$ and $c'_o = c_o/c_c$, this equation becomes the following cubic equation:

$$c'^3 - c'_o c'^2 + 0.25\Delta q' = 0$$

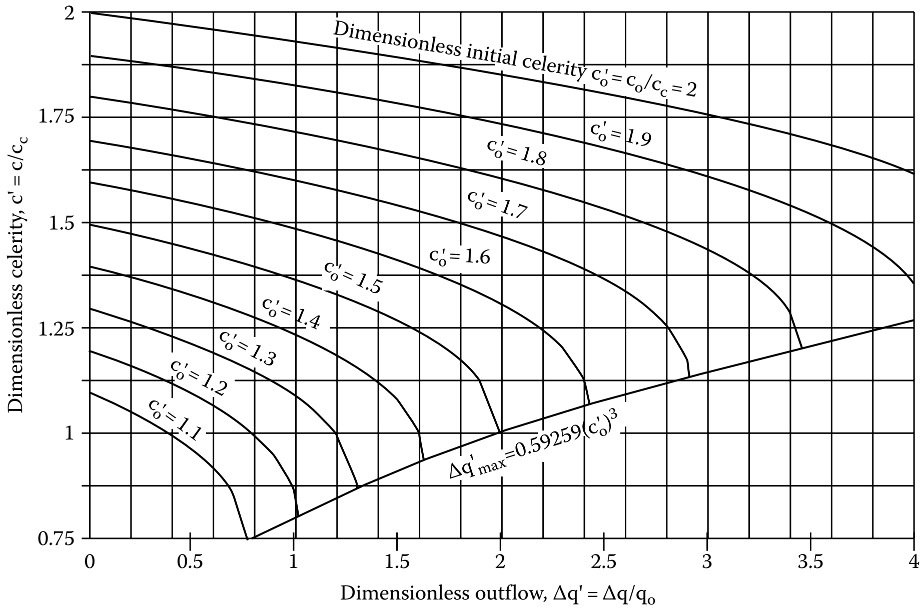
in which $\Delta q' = \Delta q/q_o$. The solution to this cubic equation can be obtained using the methodology described for solving the cubic-specific energy equation in Chapter 2, or the cubic momentum function equation in Chapter 3 for rectangular channels. Using the method that involved the arc-cosine and cosine, we note that the first root is negative, and the third root is associated with a supercritical flow, so the second root is the appropriate root. Thus, the desired root can be obtained by first solving θ from

$$\theta = \cos^{-1}\left(\frac{3.375\Delta q}{q_o c_o'^3} - 1\right) = \cos^{-1}\left(\frac{3.375\Delta q'}{c_o'^3} - 1\right)$$

and thereafter solving the dimensionless celerity $c' = c/c_c$ from

$$c' = \frac{1}{3}c'_o\left\{1 - 2\cos\left(\frac{\theta + 2\pi}{3}\right)\right\}$$

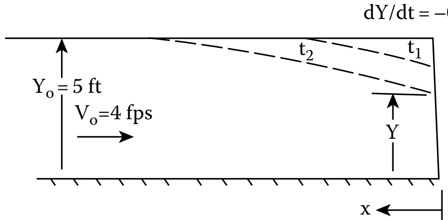
Since the cosine is restricted between plus and minus 1, the quantity $3.375\Delta q'/c_o^3$ is limited between 0 and 2. For the lower limit, $\Delta q' = 0$ and $\theta = \pi$, and $c' = c'_o\{1 - 2(-1)\}/3 = c'_o$, i.e., the flow remains uniform. For the upper limit $\Delta q'_{max} = 2c_o'^3/3.375 = 0.59259c_o'^3$, which for our example problem ($Y_o = 5'$ and $q_o = 20$ cfs/ft) results in $\Delta q'_{max} = 1.8798$, or about 1.8 times the uniform flow rate can be withdrawn, but obviously for a limited time. In practice, the unit dimensionless outflow $\Delta q'$ will be less than $\Delta q'_{max}$ because the theory that is being used and assumes $S_f - S_o = 0$ ignores the added frictional losses associated with larger flow rates.

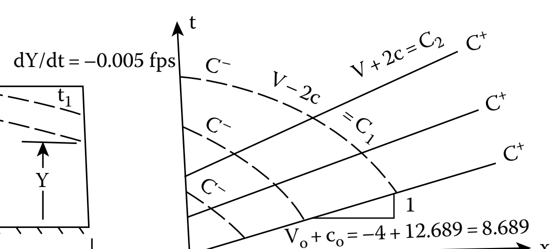


If one desires to solve the original cubic equation that gives c as a function of the point outflow ($c^3 - c_oc^2 + g\Delta q/4 = 0$) by the above method, then $\theta = \cos^{-1}(3.375g\Delta q/c_o^3 - 1)$ and $c = c_o\{1 - 2\cos[(\theta + 2\pi)/3]\}/3$. The graph above gives the relationship of c' to $\Delta q'$ and c_o'.

The following few paragraphs will show that identical results are obtained for the upstream portion of the channel using x positive in the downstream direction, and letting x be positive in the upstream direction as is typically done, and is illustrated by the previous Example Problems 6.2 and 6.5. In other words, it is typical to solve the unsteady upstream portion of the channel flow as a separate problem, and since the flow of interest is upstream of the cause of the unsteady flow to let x = 0 be at the downstream end and have the x positive pointing upstream. When the x-axis is positive upstream, then the C^+ characteristics are straight lines with an inverse slope $(dx/dt)^+ = V + c$. However, since V is in the opposite direction to x, it has a negative value. Thus, these positive characteristics have a magnitude equal to $c - |V|$ (a positive value), the magnitude of which is the same as the slope of the C^- characteristics resulting for the upstream portion of the channel when x is positive in the downstream direction. However, as shown above, the inverse slope of these negative characteristics is $(dx/dt)_u^- = V - c$ (with V now positive). The difference is now $(dx/dt)_u^-$ is negative, and must be so since x decreases with increasing time.

Just to demonstrate that the two approaches give identical results let us resolve some values given in the solution table for the upstream portion of the channel, but use x positive in the upstream direction.





Let us solve c, Y, V and dx/dt at t = 150 s at the origin x = 0 and at x = 1000 ft.

At x = 0: $Y = 5 + 150(-0.005) = 4.25$ ft, $c = (gY)^{1/2} = 11.698$ fps, $V = 2c + V_o - 2c_o = 2(11.698) - 4 - 2(12.689) = -5.981$ fps, and $dx/dt = V + c = -5.981 + 11.698 = 5.717$ fps. (Note, in the previous solution table that solved the upstream portion of the channel that $V = V_o + 2c_o - 2c = 29.377 - 2(11.698) = 5.981$ fps (positive) and $(dx/dt)_u^- = V - c = 5.981 - 11.698 = -5.717$ fps (negative). The change in sign is required since the x was positive in the downstream direction.)

At x = 1000 ft (previously x = −1000 ft): $t = t_1 + \Delta t$ results in the following implicit equation:

$$t = \frac{Y_o - Y}{|dY/dt|} + \frac{x}{3c + V_o - 2c_o} \quad \text{or}$$

$$150 = \frac{(5 - c^2/32.2)}{0.005} + \frac{1000}{(3c - 29.377)},$$

that gives c = 12.513 fps and $Y = c^2/g = 4.863$ ft. The velocity $V = 2c + V_o - 2c_o = 2c - 29.377 = -4.351$ fps, and $dx/dt = V + c = -4.351 + 12.513 = 8.162$ fps. (Note in the previous table that solved the upstream portion of the channel that the implicit equation that gives c is $t = (Y_o - c^2/g)/|dY/dt| + x/(V_o + 2c_o - 3c)$ or

$$t = \frac{(Y_o - c^2/g)}{|dY/dt|} + \frac{x}{(V_o + 2c_o - 3c)} \quad \text{or}$$

$$150 = \frac{(5 - c^2/32.2)}{0.005} + \frac{(-1000)}{29.377 - 3c},$$

that gives c = 12.513 fps, and $V = V_o + 2c_o - 2c = 29.377 - 2(12.513) = 4.351$ fps (positive) and $(dx/dt)_u^- = V - c = 4 - 12.162 = -8.162$ fps $= V - c = 4 - 12.162 = -8.162$ fps (negative.)

EXAMPLE PROBLEM 6.7

Initially uniform flow at a depth of $Y_o = 2$ m and a velocity $V_o = 2.5$ m/s occurs in a rectangular channel. Suddenly at a position near the middle of the channel, a point outflow Δq begins. The rate of change of this outflow $d(\Delta q)/dt = .05 - .0003t$ for 90 s, and thereafter remains constant. Make up a table that gives Y, V, and q at the following positions: x = 0 (origin), x = 150 m, x = 300 m, x = −50 m, and x = −100 m.

Solution

First the equation that gives Δq is obtained by integrating the change $d(\Delta q)/dt = .05 - .0003t$. Assuming that at t = 0, $\Delta q = 0$, we obtain $\Delta q = (.05 - .00015t)t$. Now at the origin x = 0 and at t = 10 s, $\Delta q = 0.485\ (m/s)^2$ and the equation $g\Delta q/4 = (c_o - c)c^2 = (9.81)(.485)/4 = 1.189$ is solved for c giving c = 4.367 m/s, and $Y = c^2/g = 1.944$ m. For the downstream portion of the flow, the velocity is given by $V = 2c + V_o - 2c_o = 2(4.367) + 2.5 - 2(4.429) = 2.375$ m/s, and $q = VY = 4.618\ m^2/s$. For the upstream portion of the channel, but still at x = 0, $V = V_o + 2c_o - 2c = 2.5 + 2(4.429) - 2(4.367) = 2.625$ m/s, and $q = 5.103\ (m/s)^2$. These values are shown in the table below, and the difference between the upstream and the downstream unit flow rates equals the Δq corresponding to time t, or for our problem 5.103 − 4.618 = 0.485.

At downstream positions (such as 150 and 300 m) the sum of t_1 and Δt must equal the time for which the values are sought, or $t = t_1 + \Delta t$ in which both t_1 and Δt are expressed in term of the celerity c. $\Delta t = x/(dx/dt)$ or using the downstream velocity $\Delta t = x/(3c + V_o - 2c_o)$ and since we are seeking the C^+ characteristics that passes through the point x and t, Δq is first expressed as a function of c, or $\Delta q = (4/g)(c_o - c)c^2$, and then the quadratic equation that gives Δq as a function of t is solved for t_1. In general, this polynomial equation is $\Delta q = at^2 + bt + c$, and using the quadratic formula $t_1 = \{-b + (b^2 - 4ac)^{1/2}\}/(2a)$. For our problem a = −.00015, b = .05, c = 0 − Δq. Note that this Δq is not the Δq associated with time t at the origin in the second column of the solution table below, but the Δq at the time t_1. For example, solving c by the above process for x = 150 m and t = 60 s gives c = 4.207 m/s, which is obtained using the Newton method. This same procedure is followed for upstream distances, with the difference that for these positions $\Delta t = x/(V_o + 2c_o - 3c)$. You should verify several values in the table below to make sure that the solution process is understood.

	At the Origin, x = 0								At x = 150 m			
t (s)	Δq	c (m/s)	Y (m)	V-dow (m/s)	q-dow (mss)	q-up (mss)	V-up (m/s)	c (m/s)	Y (m)	V (m/s)	q (mss)
0	0.000	4.429	2.000	2.500	5.000	5.000	2.500	4.429	2.000	2.500	5.000
10	0.485	4.367	1.944	2.375	4.618	5.103	2.625	4.429	2.000	2.500	5.000
20	0.940	4.305	1.889	2.251	4.253	5.193	2.749	4.429	2.000	2.500	5.000
30	1.365	4.244	1.836	2.128	3.907	5.272	2.872	4.380	1.956	2.402	4.697
40	1.760	4.183	1.783	2.007	3.579	5.339	2.993	4.322	1.904	2.285	4.350
50	2.125	4.123	1.733	1.887	3.269	5.394	3.113	4.264	1.853	2.169	4.020
60	2.460	4.064	1.684	1.769	2.979	5.439	3.231	4.207	1.804	2.054	3.706
70	2.765	4.007	1.637	1.655	2.710	5.475	3.345	4.150	1.756	1.942	3.410
80	3.040	3.952	1.592	1.545	2.460	5.500	3.455	4.095	1.710	1.831	3.131
90	3.285	3.900	1.550	1.440	2.233	5.518	3.560	4.041	1.665	1.724	2.870
100								3.989	1.622	1.619	2.626
110	Δq remains constant at 3.3 (m/s)**2 so other variables are also							3.939	1.581	1.518	2.401
120	the same							3.900	1.550	1.440	2.233
130								3.900	1.550	1.440	2.233

	At x = 300 m				At x = −50 m			
t (s)	c (m/s)	Y (m)	V (m/s)	q (mss)	c (m/s)	Y (m)	V (m/s)	q (mss)
0	4.429	2.000	2.500	5.000	4.429	2.000	2.500	5.000
10	4.429	2.000	2.500	5.000	4.429	2.000	2.500	5.000
20	4.429	2.000	2.500	5.000	4.429	2.000	2.500	5.000
30	4.429	2.000	2.500	5.000	4.409	1.982	2.541	5.035
40	4.429	2.000	2.500	5.000	4.361	1.939	2.637	5.112
50	4.392	1.966	2.425	4.769	4.315	1.898	2.730	5.180
60	4.337	1.917	2.314	4.437	4.271	1.859	2.818	5.238
70	4.282	1.869	2.205	4.121	4.229	1.823	2.901	5.288
80	4.228	1.822	2.097	3.821	4.190	1.790	2.978	5.331
90	4.175	1.777	1.990	3.536	4.154	1.759	3.050	5.366
100	4.122	1.732	1.886	3.267	4.121	1.731	3.116	5.396
110	4.071	1.690	1.784	3.014	4.091	1.706	3.176	5.420
120	4.022	1.649	1.684	2.777	4.064	1.684	3.231	5.439
130	3.974	1.609	1.588	2.556	4.040	1.664	3.279	5.455

	At x = −100 m					
t (s)	c (m/s)	Y (m)	V (m/s)	q (mss)	x-do (m)	x-up (m)
0	4.429	2.000	2.500	5.000	0	0
10	4.429	2.000	2.500	5.000	69	−19
20	4.429	2.000	2.500	5.000	139	−39

(continued)

	At x = −100 m					
t (s)	c (m/s)	Y (m)	V (m/s)	q (mss)	x-do (m)	x-up (m)
30	4.429	2.000	2.500	5.000	208	−58
40	4.429	2.000	2.500	5.000	277	−77
50	4.429	2.000	2.500	5.000	346	−96
60	4.396	1.970	2.567	5.056	416	−116
70	4.357	1.935	2.645	5.118	485	−135
80	4.320	1.903	2.719	5.172	554	−154
90	4.285	1.872	2.788	5.220	624	−174
100	4.253	1.844	2.854	5.261	693	−193
110	4.222	1.817	2.915	5.296	762	−212
120	4.194	1.793	2.971	5.327	832	−232
130	4.167	1.770	3.024	5.354	901	−251

6.7 MAXIMUM POSSIBLE FLOW RATES

It is intuitive that there is a maximum flow rate that can take place in a channel that initially is under uniform flow conditions even if there is no control downstream from that point to limit the flow. An example of such a situation would be to have a channel discharging into a reservoir whose water surface elevation was at the same level as the uniform depth in the channel, and then suddenly at time zero have the reservoir level drop more rapidly than the water level can drop in the channel. This maximum flow rate that the channel can supply can be obtained, based on the assumptions inherent in the method of characteristics, by finding the maximum value, q_{max} (which may have a negative magnitude because of the direction of the x axis) that the equation for the flow rate boundary condition equation will allow. For the present, a channel with the unsteady flow controlled downstream will be considered. Rewriting Equation 6.19 without the arguments (0, t) gives

$$q = VY = \frac{(2c + V_o - 2c_o)c^2}{g} \tag{6.19a}$$

Equation 6.19a can be written as a function of the two variables, the celerity, c and the flow rate, q or,

$$f(c,q) = gq - c^2(2c + V_o - 2c_o) = 0 \tag{6.19b}$$

A plot of Equation 6.19b with f(c, q) on the ordinate and c on the abscissa and q as a series of lines shows a concave downward family of curves (see a subsequent illustrative example problem for such a plot) with a maximum value of each curve. As the flow rate q increases, the maximum value of this function decreases, and at some value of q, which is q_{max}, the two positive roots for c become imaginary, i.e., there is no real solution of c from Equation 6.19b. The maximum flow rate, q_{max} is determined when the curve of a given q has a zero slope at the f(c, q) = 0 horizontal axis. This flow rate can be determined by taking the derivative of Equation 6.19b with respect to c, and setting this derivative equal to zero, or,

$$\frac{\partial f}{\partial c} = -6c^2 + 2(2c_o - V_o)c = 0 \quad \text{or}$$
$$2c_o - V_o - 3c = 0 \tag{6.22}$$

In principle, this operation gives a second equation that can be solved simultaneously with Equation 6.19b for the two unknowns, q_{max} and c_{min}. However, because of the symmetry of the curve so that the maximum value of f is independent of q, which is reflected in Equation 6.22 not involving q, Equation 6.22 can be solved directly for c_{min}, and this value substituted into Equation 6.19a to solve q_{max}. Doing this produces

$$c_{min} = \frac{1}{3}(2c_o - V_o) \tag{6.23}$$

and q_{min} (or q_{max} if one ignores the fact that q is negative) is given by

$$\begin{aligned} q_{min} &= -\frac{(2c_o - V_o)(2c_o - V_o)^2}{(27g)} = -\frac{(2c_o - V_o)^3}{(27g)} - \frac{c_{min}^3}{g} \\ \text{or} \quad |q_{max}| &= \frac{c_{min}}{g} \end{aligned} \tag{6.24}$$

When the minimum depth occurs, the inverse slope, dx/dt, of that positive characteristic becomes zero, i.e., these characteristics becomes a vertical line. This means V + c is zero, or V = c, or critical flow is taking place. Under critical flow conditions, the velocity in the channel washes away the effect of any disturbance downstream as fast as it can propagate upstream. To prove that dx/dt is zero when the depth is critical, substitute c_{min} for c in Equation 6.16, that gives dx/dt, or replace c in $dx/dt = 3c + V_o - 2c_o$ by $(2c_o - V_o)/3$. You should also note that the relationship between critical q_c and Y_c is maintained, which can be verified by substituting $(gY_c)^{1/3}$ for q_{min} in Equation 6.24 (since $q_c = (gY_c)^{1/3}$. Then replace $c_{min}{}^2/g$ for Y_c.

EXAMPLE PROBLEM 6.8

Under uniform flow the depth and velocity in a rectangular channel are: $Y_o = 4$ ft, and $V_o = 3$ fps. This channel ends in a free overfall a short distance downstream from a gate that is currently set so that uniform flow occurs. If the gate is suddenly and completely opened determine the maximum discharge that will take place from the channel immediately after the gate is opened.

Solution

To solve this problem the celerity associated with uniform flow is first computed, $c_o = (gY_o)^{1/2} = 11.349$ fps. Since the origin is at the gate's position and the x axis is positive in the upstream flow direction $V_o = -3$ fps. Substituting these values into Equation 6.23 gives: $c_{min} = (2(11.349) + 3)/3 = 8.566$ fps. Thus the depth of flow at x = 0 in the channel will be $Y = c_{min}{}^2/g = (8.566)^2/32.2 = 2.28$ ft. Substituting into Equation 6.24 gives: $q_{min} = -\{2(11.349) + 3\}^3/(27g) = -19.520$ cfs/ft, or $q_{max} = 19.520$ cfs/ft. The above mathematics can be readily understood by plotting f(c, q) against c for several values of q. This has been done in Figure 6.1 below for q = −16, −18, and −20 cfs/ft. Note that the maximum point for the curve q = −20 cfs/ft is below the f(c, q) = 0 horizontal axis. If a curve for q = −19.52 were plotted, this curve would just touch the zero horizontal axis. That dx/dt = 0 for these critical conditions can be noted since $dx/dt = 3c_{min} + V_o - 2c_o = 3(8.566) - 3 - 2(11.349) = 0$.

A special case, that is designated as a dam break problem, occurs when the velocity $V_o = 0$, i.e., the upstream channel is a reservoir. For this case, replacing the c's in Equation 6.23 by $(gY)^{1/2}$ gives the minimum depth as

$$Y_{min} = \frac{4}{9}Y_o$$

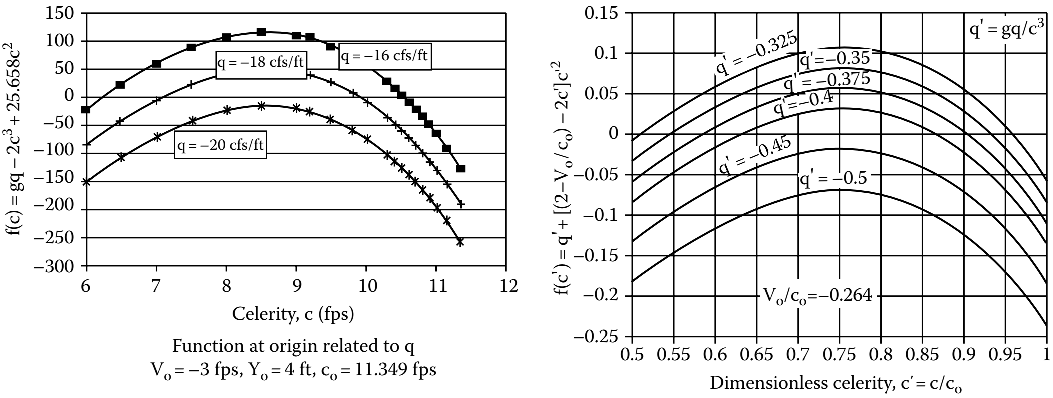


FIGURE 6.1 Function of celerity and flow rate per unit width, q, at origin for a rectangular channel.

as the depth by the breached dam, or a gate that is suddenly opened. From Equation 6.15 the velocity is

$$V_{max} = 2c + V_o - 2c_o = 2(c - c_o) = 2\left(\frac{2}{3}c_o - c_o\right) = -\frac{2}{3}c_o = -\frac{2}{3}(gY_o)^{1/2}$$

From Equation 6.24

$$q_{max} = -\frac{8c_o^3}{27g} = \frac{8\sqrt{gY_o^3}}{27}$$

Note, the maximum velocity can also be obtained by dividing q_{max} by Y_{min}. While the assumptions of $g(S_o - S_f) = 0$ are violated for such extreme flow conditions, and three-dimensional flow effects are important, the above results are useful indicators of flow conditions when a gate is instantly opened or a dam fails. Also, the nature of these curves points out that caution is needed when solving Equation 6.19 for c when the boundary condition specifies q(0, t). It is easy to specify a flow rate for which no solution exists, i.e., the magnitude of q is too large $|q| > q_{max} > |q_{min}|$.

It is useful to have the above equations that give maximum possible flow rates, and the associated minimum possible depths in the dimensionless form. Let a dimensionless celerity be defined by $c' = c/c_o$. Next, divide Equation 6.19b by c_o^3 to give

$$f(c', q') = q' - c'^2\left(2c' + \frac{V_o}{c_o} - 2\right) = q' + \left\{2(1-c) - \frac{V_o}{c_o}\right\}c'^2 = 0 \tag{6.25}$$

in which the dimensionless flow rate per unit width is defined as $q' = gq/c_o^3$. Taking the derivative of this function with respect to c′ and setting the result to zero gives

$$\frac{df}{dc} = 2\left(2 - \frac{V_o}{c_o}\right)c' - 6c'^2 = 0$$

Upon dividing by 2c′ and solving c′ gives

$$c'_{min} = \frac{1}{3}\left(2 - \frac{V_o}{c_o}\right) \tag{6.26}$$

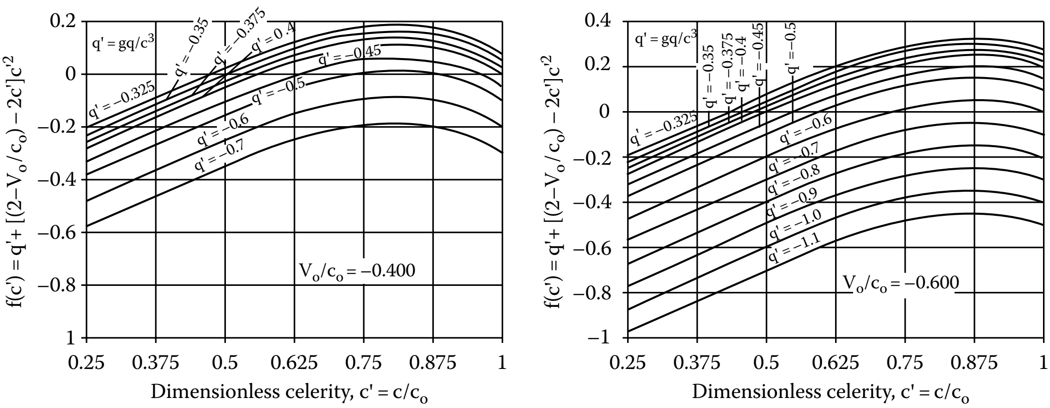


FIGURE 6.2 Graphs that give the dimensionless function $f(c',q')$ for a rectangular channel. (Roots are where the curves cross the zero horizontal axis. Only the smaller root has physical significance.)

Substituting Equation 6.26 into Equation 6.25 and solving the dimensionless flow rate gives

$$q'_{max} = -\frac{1}{27}\left(2 - \frac{V_o}{c_o}\right)^3 \tag{6.27}$$

A plot of the dimensionless functions, Equation 6.25 for different dimensionless flow rates is given in the 2nd graph on Figure 6.1, which applies for the dimensionless ratio $V_o/c_o = -3/11.349 = -0.264$ for the above problem, and similar graphs for $V_o/c_o = -0.4$ and -0.6 are in Figure 6.2. Not only are such graphs useful in determining maximum possible flow rates, but it can also be used to solve the cubic Equation 6.19 for values of c, or for obtaining starting values for an iterative solution, such as with the Newton method.

6.7.1 Maximum Point Outflow, Δq

In this section, the maximum unit point outflow Δq that can be extracted from an intermediate position along a rectangular channel will be determined when initially a uniform flow exists with a depth Y_o (and associated celerity, c_o) and velocity V_o (and therefore an initial unit flow rate $q_o = Y_oV_o$). Previously, it was shown that a unique relationship exists between the point outflow and celerity (or depth) for any initial celerity c_o, or $g\Delta q/4 = c^2(c_o - c)$. Writing this equation as a function of the two unknowns Δq and c equated to zero provides the first equation, or

$$F_1 = \frac{1}{4}g\Delta q - c^2(c_o - c) = 0$$

The second needed equation is the critical flow equation $q^2/(gY^3) = 1$, or $V^2/(gY) = 1$, which can be written using the unit flow rate q_d downstream of the point outflow, the unit flow rate upstream of the point outflow q_u or the velocity V_u upstream, or thirdly by noting that when critical flow occurs, the upstream velocity V_u will equal the celerity c, or this C^- characteristic becomes a vertical line in the xt-plane. Using these three options produces the following three equations for the second needed equation.

$$F_2 = (q_d + \Delta q)^2 - gY^3 = \left[\frac{c^2}{g}(2c + V_o - 2c_o) + \Delta q\right]^2 - \frac{c^6}{g^3}$$

$$F_2 = V_u^2 - gY = (V_o + 2c_o - 2c)^2 - c^2 = (V_o + 2c_o)^2 - 4(V_o + 2c_o)c + 3c^2 = 0$$

$$F_2 = V_u - c = (V_o + 2c_o - 2c) - c = V_o + 2c_o - 3c = 0$$

(Note that one of the factors of the second equation is the third equation, and only the first equation contains both unknowns, i.e., the other two can be solved for c without involving the first equation.) The third is the simplest of these three equations; in fact, it allows the celerity (and therefore the critical depth) to be determined from the initial uniform flow conditions by the following explicitly equation,

$$c = \frac{1}{3}(V_o + 2c_o)$$

Furthermore, this expression for c can be substituted into the above equation F_1 to provide the following equation giving the maximum unit outflow as a function of the initial conditions:

$$\Delta q_{max} = \frac{4}{27g}(V_o + 2c_o)^2(c_o - V_o)$$

EXAMPLE PROBLEM 6.9

A rectangular channel initially contains a uniform flow throughout its length with $V_o = 4$ fps and $c_o = 12$ fps (or $Y_o = 4.472$ ft.) Determine the maximum unit point outflow that can be taken from an intermediate position along this channel. What fraction of this Δq_{max} comes from the downstream reverse flow?

Solution

Solving the equation for the outflow gives, $\Delta q_{max} = [4/(27 \times 32.2)](4 + 2 \times 12)^2(12 - 4) = 28.857$ cfs/ft, and the equation for c gives, $c_c = [4 + 2 \times 12]/3 = 9.333$ fps, or $Y_c = 2.705$ ft. The upstream flow rate $q_u = V_uY_c = [V_o + 2c_o - 2c]Y_c = [28 - 2(9.333)]2.705 = 25.250$ cfs/ft. Therefore, the downstream reverse flow is $q_d = q_u - \Delta q = 25.250 - 28.857 = -3.607$ cfs/ft. Alternatively, $q_d = V_dY_c = [2c + V_o - 2c_o]Y_c = -1.333(2.705) = -3.607$ cfs/ft. Therefore, the fraction of the outflow coming from the downstream reverse flow is 3.607/28.857 = .125.

You should verify that the same results are obtained by solving the above equation F_1 simultaneously with all three of the above F_2 equations.

6.8 EXTENDING THE METHODS TO NONRECTANGULAR CHANNELS

6.8.1 Trapezoidal Channels

The methods described above for solving unsteady dispersive waves problems in rectangular channels can be extended with modifications to nonrectangular channels. A variant form of the characteristics, Equations 6.11, can be obtained by multiplying the continuity Equation 6.3a by $l = (g/AT)^{1/2}$ and by adding the result to the equation of motion, Equation 6.4a, to get the equation associated with the positive characteristic C^+. The equation associated with the negative characteristic is obtained by subtracting the result from Equation 6.4a, to give the following two equations:

$$\frac{\partial V}{\partial t} + (V \pm c)\frac{\partial V}{\partial x} \pm \left(\frac{gT}{A}\right)^{1/2}\left\{\frac{\partial Y}{\partial t} + (V \pm c)\frac{\partial Y}{\partial x}\right\} = g(S_o - S_f) \tag{6.28}$$

A stage variable w, that was introduced by Escoffier, is defined by

$$w = \int_e^Y \sqrt{\frac{gT}{A}}dY = g\int_e^Y \frac{dy}{c} = \int_e^Y \sqrt{\frac{gA}{T}}\left(\frac{T}{A}\right)dy = \int_e^Y \frac{c\,dA}{A}$$

The above equation can be written as

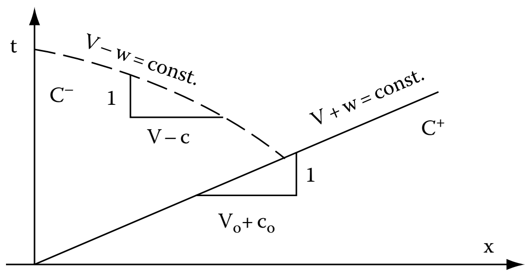
$$\frac{\partial V}{\partial t}+(V\pm c)\frac{\partial V}{\partial x}\pm\left\{\frac{\partial w}{\partial t}+(V\pm c)\frac{\partial w}{\partial x}\right\}=g(S_o-S_f) \tag{6.29}$$

which can be written as the following four ODEs:

$$\frac{d(V\pm w)}{dt}=g(S_o-S_f) \tag{6.30}$$

$$\frac{dx}{dt}=V\pm c \tag{6.31}$$

If the right hand side of Equation 6.30 is assumed to be zero, then along the C^+ and C^- characteristics defined with slopes V + c and V − c in the xt plane, respectively, the values of V + w and V − w are constant. The only complexity is that w is defined by the above integral.



To make the method useful in obtaining solutions to dispersive unsteady waves type problems that are close to uniform flow conditions, dimensionless variables associated with the stage function w will be developed, and tables for trapezoidal and circular channels of the dimensionless stage function w′ will be provided that allow values for w to be obtained so that Equations 6.30 and 6.31 will be nearly as useful as Equations 6.13 and 6.14 are in solving problems in rectangular channels.

6.8.2 Dimensionless Variables

As has been done previously, a dimensionless depth in a trapezoidal channel will be defined as $Y' = mY/b$. Then, the top width T and the area A are as given below as functions of Y':

$$T=b(1-2Y')$$

$$A=\left(\frac{b^2}{m}\right)(Y'+Y'^2)$$

and the celerity c squared becomes

$$c^2=\frac{gA}{T}=\frac{gb(Y'+Y'^2)}{m(1+2Y')}=\left(\frac{gb}{m}\right)\left(\frac{mY}{b}\right)\frac{1+Y'}{1+2Y'}=c_r^2\,\frac{1+Y'}{1+2Y'}$$

in which $c_r = (gY)^{1/2}$, the celerity of a small amplitude gravity wave in a rectangular channel with the depth of flow in this trapezoidal channel. If a dimensionless celerity in a trapezoidal channel is defined by

$$c'^2 = \frac{Y' + Y'^2}{1 + 2Y'} \tag{6.32}$$

then, the celerity c can also be obtained from this dimensionless c' by

$$c = c'\left(\frac{gb}{m}\right)^{1/2} \quad \text{or} \quad c' = c\sqrt{\frac{m}{(gb)}} \tag{6.33}$$

From the definition of the stage variable w above, it becomes

$$w = g\int_{\varepsilon}^{Y} \frac{dy}{c} = \sqrt{\frac{gb}{m}}\int_{\varepsilon}^{Y'} \frac{dy'}{c'}$$

Now, define a dimensionless stage variable w' as

$$w' = \frac{w}{\sqrt{gb/m}} = \int_{\varepsilon}^{Y'} \frac{1}{c'}dy' = \int_{\varepsilon}^{Y'} \sqrt{\frac{1 + 2y'}{y' + y'^2}}\, dy' \tag{6.34}$$

Table 6.1 contains values of both c' and w' as a function of the dimensionless depth Y'. This table makes it possible to obtain w' and then in turn obtain w for use in solving unsteady flows in a trapezoidal channel through the use of Equations 6.29 and 6.30 with $g(S_o - S_f) = 0$. Likewise, if a w is determined from these equations, the corresponding values for the celerity c and depth Y can be obtained.

Boundary condition equations that give the slope of C^+ characteristics along the t axis that replace Equations 6.16 through 6.19 when dealing with a trapezoidal channel can be obtained as follows: First consider moving from the initial C^+ characteristic along a C^- characteristic to the origin such that

$$V(0,t) - w(0,t) = V_o - w_o \tag{6.35}$$

Since the inverse slope $dx/dt(0, t) = V + c$ along the C^+ characteristics, this inverse slope can be obtained from the following equation:

$$\frac{dx}{dt} = w(0,t) + c + V_o - w_o \tag{6.36}$$

If the velocity at the origin is known as a function of time, i.e., V(0, t) is given, then Equation 6.35 can be used first to obtain the value of w(0, t). This value is used next to determine the corresponding depth Y(0, t), and from this determined depth, the corresponding value of c(0, t) is computed, and finally Equation 6.36 is used to determine the slope dx/dt.

When the flow rate Q(0, t) is known, then the procedure involves utilizing this known flow rate to determine the corresponding depth from the continuity equation,

$$A(Y) - \frac{Q(0,t)}{w(0,t) + V_o - w_o} = 0 \tag{6.37}$$

TABLE 6.1
Dimensionless Variables Associated with Stage Variable, w, for Trapezoidal Channels

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.0100	.0995	.1380	.1040	.3083	.5930	.1980	.4122	.8533
.0120	.1089	.1572	.1060	.3110	.5995	.2000	.4140	.8581
.0140	.1175	.1749	.1080	.3137	.6059	.2020	.4159	.8629
.0160	.1255	.1913	.1100	.3164	.6122	.2040	.4177	.8677
.0180	.1330	.2068	.1120	.3190	.6185	.2060	.4195	.8725
.0200	.1401	.2214	.1140	.3216	.6248	.2080	.4212	.8773
.0220	.1468	.2354	.1160	.3242	.6310	.2100	.4230	.8820
.0240	.1531	.2487	.1180	.3267	.6371	.2120	.4248	.8867
.0260	.1592	.2615	.1200	.3292	.6432	.2140	.4265	.8914
.0280	.1651	.2739	.1220	.3317	.6493	.2160	.4283	.8961
.0300	.1707	.2858	.1240	.3342	.6553	.2180	.4300	.9008
.0320	.1762	.2973	.1260	.3366	.6612	.2200	.4317	.9054
.0340	.1814	.3085	.1280	.3391	.6672	.2220	.4334	.9100
.0360	.1865	.3194	.1300	.3414	.6730	.2240	.4351	.9146
.0380	.1915	.3299	.1320	.3438	.6789	.2260	.4368	.9192
.0400	.1963	.3403	.1340	.3462	.6847	.2280	.4385	.9238
.0420	.2009	.3503	.1360	.3485	.6904	.2300	.4402	.9284
.0440	.2055	.3602	.1380	.3508	.6962	.2320	.4419	.9329
.0460	.2099	.3698	.1400	.3531	.7018	.2340	.4435	.9374
.0480	.2142	.3792	.1420	.3554	.7075	.2360	.4452	.9419
.0500	.2185	.3885	.1440	.3576	.7131	.2380	.4468	.9464
.0520	.2226	.3975	.1460	.3599	.7187	.2400	.4484	.9509
.0540	.2266	.4065	.1480	.3621	.7242	.2420	.4500	.9553
.0560	.2306	.4152	.1500	.3643	.7297	.2440	.4517	.9598
.0580	.2345	.4238	.1520	.3664	.7352	.2460	.4533	.9642
.0600	.2383	.4323	.1540	.3686	.7406	.2480	.4548	.9686
.0620	.2420	.4406	.1560	.3707	.7460	.2500	.4564	.9730
.0640	.2457	.4488	.1580	.3729	.7514	.2520	.4580	.9773
.0660	.2493	.4569	.1600	.3750	.7568	.2540	.4596	.9817
.0680	.2528	.4648	.1620	.3771	.7621	.2560	.4611	.9860
.0700	.2563	.4727	.1640	.3791	.7674	.2580	.4627	.9904
.0720	.2597	.4804	.1660	.3812	.7726	.2600	.4642	.9947
.0740	.2631	.4881	.1680	.3832	.7779	.2620	.4658	.9990
.0760	.2664	.4956	.1700	.3853	.7831	.2640	.4673	1.003
.0780	.2697	.5031	.1720	.3873	.7883	.2660	.4688	1.008
.0800	.2729	.5105	.1740	.3893	.7934	.2680	.4704	1.012
.0820	.2761	.5178	.1760	.3913	.7985	.2700	.4719	1.016
.0840	.2792	.5250	.1780	.3932	.8036	.2720	.4734	1.020
.0860	.2823	.5321	.1800	.3952	.8087	.2740	.4749	1.025
.0880	.2853	.5391	.1820	.3971	.8138	.2760	.4764	1.029
.0900	.2883	.5461	.1840	.3991	.8188	.2780	.4778	1.033
.0920	.2913	.5530	.1860	.4010	.8238	.2800	.4793	1.037
.0940	.2942	.5598	.1880	.4029	.8288	.2820	.4808	1.041
.0960	.2971	.5666	.1900	.4048	.8337	.2840	.4822	1.045
.0980	.2999	.5733	.1920	.4066	.8386	.2860	.4837	1.050
.1000	.3028	.5799	.1940	.4085	.8435	.2880	.4852	1.054
.1020	.3055	.5865	.1960	.4104	.8484	.2900	.4866	1.058

TABLE 6.1 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Trapezoidal Channels

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.2920	.4880	1.062	.3840	.5483	1.239	.4760	.5999	1.400
.2940	.4895	1.066	.3860	.5495	1.243	.4780	.6010	1.403
.2960	.4909	1.070	.3880	.5507	1.247	.4800	.6020	1.406
.2980	.4923	1.074	.3900	.5519	1.250	.4820	.6031	1.410
.3000	.4937	1.078	.3920	.5531	1.254	.4840	.6041	1.413
.3020	.4951	1.082	.3940	.5542	1.257	.4860	.6052	1.416
.3040	.4965	1.086	.3960	.5554	1.261	.4880	.6062	1.419
.3060	.4979	1.090	.3980	.5566	1.265	.4900	.6072	1.423
.3080	.4993	1.094	.4000	.5578	1.268	.4920	.6083	1.426
.3100	.5007	1.098	.4020	.5589	1.272	.4940	.6093	1.429
.3120	.5021	1.102	.4040	.5601	1.275	.4960	.6103	1.433
.3140	.5034	1.106	.4060	.5613	1.279	.4980	.6114	1.436
.3160	.5048	1.110	.4080	.5624	1.283	.5000	.6124	1.439
.3180	.5062	1.114	.4100	.5636	1.286	.5020	.6134	1.442
.3200	.5075	1.118	.4120	.5647	1.290	.5040	.6144	1.446
.3220	.5089	1.122	.4140	.5659	1.293	.5060	.6154	1.449
.3240	.5102	1.126	.4160	.5670	1.297	.5080	.6164	1.452
.3260	.5115	1.130	.4180	.5682	1.300	.5100	.6174	1.455
.3280	.5129	1.134	.4200	.5693	1.304	.5120	.6185	1.459
.3300	.5142	1.138	.4220	.5705	1.307	.5140	.6195	1.462
.3320	.5155	1.142	.4240	.5716	1.311	.5160	.6205	1.465
.3340	.5168	1.145	.4260	.5727	1.314	.5180	.6215	1.468
.3360	.5181	1.149	.4280	.5738	1.318	.5200	.6225	1.472
.3380	.5195	1.153	.4300	.5750	1.321	.5220	.6235	1.475
.3400	.5208	1.157	.4320	.5761	1.325	.5240	.6244	1.478
.3420	.5221	1.161	.4340	.5772	1.328	.5260	.6254	1.481
.3440	.5234	1.165	.4360	.5783	1.332	.5280	.6264	1.484
.3460	.5246	1.169	.4380	.5794	1.335	.5300	.6274	1.488
.3480	.5259	1.172	.4400	.5805	1.339	.5320	.6284	1.491
.3500	.5272	1.176	.4420	.5816	1.342	.5340	.6294	1.494
.3520	.5285	1.180	.4440	.5827	1.345	.5360	.6304	1.497
.3540	.5297	1.184	.4460	.5838	1.349	.5380	.6313	1.500
.3560	.5310	1.187	.4480	.5849	1.352	.5400	.6323	1.503
.3580	.5323	1.191	.4500	.5860	1.356	.5420	.6333	1.507
.3600	.5335	1.195	.4520	.5871	1.359	.5440	.6342	1.510
.3620	.5348	1.199	.4540	.5882	1.363	.5460	.6352	1.513
.3640	.5360	1.202	.4560	.5893	1.366	.5480	.6362	1.516
.3660	.5373	1.206	.4580	.5904	1.369	.5500	.6371	1.519
.3680	.5385	1.210	.4600	.5914	1.373	.5520	.6381	1.522
.3700	.5397	1.214	.4620	.5925	1.376	.5540	.6391	1.525
.3720	.5410	1.217	.4640	.5936	1.379	.5560	.6400	1.529
.3740	.5422	1.221	.4660	.5946	1.383	.5580	.6410	1.532
.3760	.5434	1.225	.4680	.5957	1.386	.5600	.6419	1.535
.3780	.5446	1.228	.4700	.5968	1.390	.5620	.6429	1.538
.3800	.5459	1.232	.4720	.5978	1.393	.5640	.6438	1.541
.3820	.5471	1.236	.4740	.5989	1.396	.5660	.6448	1.544

(*continued*)

TABLE 6.1 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Trapezoidal Channels

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.5680	.6457	1.547	.6620	.6881	1.688	.7560	.7270	1.821
.5700	.6467	1.550	.6640	.6889	1.691	.7580	.7278	1.824
.5720	.6476	1.553	.6660	.6898	1.694	.7600	.7286	1.826
.5740	.6485	1.556	.6680	.6906	1.697	.7620	.7293	1.829
.5760	.6495	1.560	.6700	.6915	1.700	.7640	.7301	1.832
.5780	.6504	1.563	.6720	.6923	1.703	.7660	.7309	1.835
.5800	.6514	1.566	.6740	.6932	1.706	.7680	.7317	1.837
.5820	.6523	1.569	.6760	.6941	1.708	.7700	.7325	1.840
.5840	.6532	1.572	.6780	.6949	1.711	.7720	.7333	1.843
.5860	.6541	1.575	.6800	.6957	1.714	.7740	.7341	1.846
.5880	.6551	1.578	.6820	.6966	1.717	.7760	.7349	1.848
.5900	.6560	1.581	.6840	.6974	1.720	.7780	.7357	1.851
.5920	.6569	1.584	.6860	.6983	1.723	.7800	.7364	1.854
.5940	.6578	1.587	.6880	.6991	1.726	.7820	.7372	1.857
.5960	.6587	1.590	.6900	.7000	1.728	.7840	.7380	1.859
.5980	.6597	1.593	.6920	.7008	1.731	.7860	.7388	1.862
.6000	.6606	1.596	.6940	.7016	1.734	.7880	.7396	1.865
.6020	.6615	1.599	.6960	.7025	1.737	.7900	.7403	1.867
.6040	.6624	1.602	.6980	.7033	1.740	.7920	.7411	1.870
.6060	.6633	1.605	.7000	.7042	1.743	.7940	.7419	1.873
.6080	.6642	1.608	.7020	.7050	1.746	.7960	.7427	1.875
.6100	.6651	1.611	.7040	.7058	1.748	.7980	.7434	1.878
.6120	.6660	1.614	.7060	.7066	1.751	.8000	.7442	1.881
.6140	.6669	1.617	.7080	.7075	1.754	.8020	.7450	1.884
.6160	.6678	1.620	.7100	.7083	1.757	.8040	.7457	1.886
.6180	.6687	1.623	.7120	.7091	1.760	.8060	.7465	1.889
.6200	.6696	1.626	.7140	.7100	1.763	.8080	.7473	1.892
.6220	.6705	1.629	.7160	.7108	1.765	.8100	.7480	1.894
.6240	.6714	1.632	.7180	.7116	1.768	.8120	.7488	1.897
.6260	.6723	1.635	.7200	.7124	1.771	.8140	.7496	1.900
.6280	.6732	1.638	.7220	.7132	1.774	.8160	.7503	1.902
.6300	.6741	1.641	.7240	.7141	1.777	.8180	.7511	1.905
.6320	.6750	1.644	.7260	.7149	1.779	.8200	.7519	1.908
.6340	.6758	1.647	.7280	.7157	1.782	.8220	.7526	1.910
.6360	.6767	1.650	.7300	.7165	1.785	.8240	.7534	1.913
.6380	.6776	1.653	.7320	.7173	1.788	.8260	.7541	1.916
.6400	.6785	1.656	.7340	.7181	1.791	.8280	.7549	1.918
.6420	.6794	1.659	.7360	.7189	1.793	.8300	.7557	1.921
.6440	.6802	1.662	.7380	.7197	1.796	.8320	.7564	1.923
.6460	.6811	1.665	.7400	.7205	1.799	.8340	.7572	1.926
.6480	.6820	1.668	.7420	.7214	1.802	.8360	.7579	1.929
.6500	.6829	1.671	.7440	.7222	1.804	.8380	.7587	1.931
.6520	.6837	1.674	.7460	.7230	1.807	.8400	.7594	1.934
.6540	.6846	1.676	.7480	.7238	1.810	.8420	.7602	1.937
.6560	.6855	1.679	.7500	.7246	1.813	.8440	.7609	1.939
.6580	.6863	1.682	.7520	.7254	1.815	.8460	.7617	1.942
.6600	.6872	1.685	.7540	.7262	1.818	.8480	.7624	1.945

TABLE 6.1 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Trapezoidal Channels

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.8500	.7632	1.947	.9420	.7964	2.065	1.034	.8280	2.178
.8520	.7639	1.950	.9440	.7971	2.068	1.036	.8286	2.181
.8540	.7646	1.952	.9460	.7978	2.070	1.038	.8293	2.183
.8560	.7654	1.955	.8480	.7985	2.073	1.040	.8300	2.186
.8580	.7661	1.958	.9500	.7992	2.075	1.042	.8306	2.188
.8600	.7669	1.960	.9520	.7999	2.078	1.044	.8313	2.190
.8620	.7676	1.963	.9540	.8006	2.080	1.046	.8320	2.193
.8640	.7683	1.965	.9560	.8013	2.083	1.048	.8326	2.195
.8660	.7691	1.968	.9580	.8020	2.085	1.050	.8333	2.198
.8680	.7698	1.971	.9600	.8027	2.088	1.052	.8339	2.200
.8700	.7706	1.973	.9620	.8034	2.090	1.054	.8346	2.202
.8720	.7713	1.976	.9640	.8041	2.093	1.056	.8353	2.205
.8740	.7720	1.978	.9660	.8048	2.095	1.058	.8359	2.207
.8760	.7728	1.981	.9680	.8055	2.098	1.060	.8366	2.210
.8780	.7735	1.984	.9700	.8062	2.100	1.062	.8372	2.212
.8800	.7742	1.986	.9720	.8069	2.103	1.064	.8379	2.214
.8820	.7750	1.989	.9740	.8076	2.105	1.066	.8386	2.217
.8840	.7757	1.991	.9760	.8083	2.108	1.068	.8392	2.219
.8860	.7764	1.994	.9780	.8090	2.110	1.070	.8399	2.222
.8880	.7771	1.996	.9800	.8097	2.112	1.072	.8405	2.224
.8900	.7779	1.999	.9820	.8103	2.115	1.074	.8412	2.226
.8920	.7786	2.002	.9840	.8110	2.117	1.076	.8418	2.229
.8940	.7793	2.004	.9860	.8117	2.120	1.078	.8425	2.231
.8960	.7800	2.007	.9880	.8124	2.122	1.080	.8431	2.233
.8980	.7808	3.009	.9900	.8131	2.125	1.082	.8438	2.236
.9000	.7815	2.012	.9920	.8138	2.127	1.084	.8444	2.238
.9020	.7822	2.014	.9940	.8145	2.130	1.086	.8451	2.241
.9040	.7829	2.017	.9960	.8151	2.132	1.088	.8457	2.243
.9060	.7836	2.020	.9980	.8158	2.135	1.090	.8464	2.245
.9080	.7844	2.022	1.0000	.8165	2.137	1.092	.8470	2.240
.9100	.7851	2.025	1.002	.8172	2.139	1.094	.8477	2.250
.9120	.7858	2.027	1.004	.8179	2.142	1.096	.8483	2.252
.9140	.7865	2.030	1.006	.8185	2.144	1.098	.8490	2.255
.9160	.7872	2.032	1.008	.8192	2.147	1.100	.8496	2.257
.9180	.7879	2.035	1.010	.8199	2.149	1.102	.8503	2.259
.9200	.7887	2.037	1.012	.8206	2.152	1.104	.8509	2.262
.9220	.7894	2.040	1.014	.8212	2.154	1.106	.8516	2.264
.9240	.7901	2.042	1.016	.8219	2.157	1.108	.8522	2.266
.9260	.7908	2.045	1.018	.8226	2.159	1.110	.8529	2.269
.9280	.7915	2.047	1.020	.8233	2.161	1.112	.8535	2.271
.9300	.7922	2.050	1.022	.8239	2.164	1.114	.8541	2.274
.9320	.7929	2.053	1.024	.8246	2.166	1.116	.8548	2.276
.9340	.7936	2.055	1.026	.8253	2.169	1.118	.8554	2.278
.9360	.7943	2.058	1.028	.8259	2.171	1.120	.8561	2.281
.9380	.7950	2.060	1.030	.8266	2.174	1.122	.8567	2.283
.9400	.7957	2.063	1.032	.8273	2.176	1.124	.8573	2.285

(continued)

TABLE 6.1 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Trapezoidal Channels

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
1.126	.8580	2.288	1.220	.8873	2.395	1.314	.9155	2.500
1.128	.8586	2.290	1.222	.8879	2.398	1.316	.9161	2.502
1.130	.8592	2.292	1.224	.8885	2.400	1.318	.9166	2.504
1.132	.8599	2.295	1.226	.8891	2.402	1.320	.9172	2.506
1.134	.8605	2.297	1.228	.8898	2.404	1.322	.9178	2.508
1.136	.8612	2.299	1.230	.8904	2.406	1.324	.9184	2.510
1.138	.8618	2.301	1.232	.8910	2.409	1.326	.9190	2.513
1.140	.8624	2.304	1.234	.8916	2.411	1.328	.9196	2.515
1.142	.8631	2.306	1.236	.8922	2.413	1.330	.9202	2.517
1.144	.8637	2.308	1.238	.8928	2.415	1.332	.9207	2.519
1.146	.8643	2.311	1.240	.8934	2.418	1.334	.9213	2.521
1.148	.8650	2.313	1.242	.8940	2.420	1.336	.9219	2.523
1.150	.8656	2.315	1.244	.8946	2.422	1.338	.9225	2.526
1.152	.8662	2.318	1.246	.8952	2.424	1.340	.9231	2.528
1.154	.8668	2.320	1.248	.8958	2.427	1.342	.9237	2.530
1.156	.8675	2.322	1.250	.8964	2.429	1.344	.9242	2.532
1.158	.8681	2.325	1.252	.8970	2.431	1.346	.9248	2.534
1.160	.8687	2.327	1.254	.8976	2.433	1.348	.9254	2.536
1.162	.8694	2.329	1.256	.8982	2.436	1.350	.9260	2.539
1.164	.8700	2.332	1.258	.8988	2.438	1.352	.9266	2.541
1.166	.8706	2.334	1.260	.8994	2.440	1.354	.9271	2.543
1.168	.8712	2.336	1.262	.9000	2.442	1.356	.9277	2.545
1.170	.8719	2.338	1.264	.9006	2.444	1.358	.9283	2.547
1.172	.8725	2.341	1.266	.9012	2.447	1.360	.9289	2.549
1.174	.8731	2.343	1.268	.9018	2.449	1.362	.9294	2.552
1.176	.8737	2.345	1.270	.9024	2.451	1.364	.9300	2.554
1.178	.8744	2.348	1.272	.9030	2.453	1.366	.9306	2.556
1.180	.8750	2.350	1.274	.9036	2.456	1.368	.9312	2.558
1.182	.8756	2.352	1.276	.9042	2.458	1.370	.9317	2.560
1.184	.8762	2.354	1.278	.9048	2.460	1.372	.9323	2.562
1.186	.8768	2.357	1.280	.9054	2.462	1.374	.9329	2.564
1.188	.8775	2.359	1.282	.9060	2.464	1.376	.9335	2.567
1.190	.8781	2.361	1.284	.9066	2.467	1.378	.9340	2.569
1.192	.8787	2.364	1.286	.9072	2.469	1.380	.9346	2.571
1.194	.8793	2.366	1.288	.9078	2.471	1.382	.9352	2.573
1.196	.8799	2.368	1.290	.9084	2.473	1.384	.9358	2.575
1.198	.8806	2.370	1.292	.9090	2.475	1.386	.9363	2.577
1.200	.8812	2.373	1.294	.9096	2.478	1.388	.9369	2.579
1.202	.8818	2.375	1.296	.9102	2.480	1.390	.9375	2.582
1.204	.8824	2.377	1.298	.9108	2.482	1.392	.9380	2.584
1.206	.8830	2.379	1.300	.9113	2.484	1.394	.9386	2.586
1.208	.8836	2.382	1.302	.9119	2.486	1.396	.9392	2.588
1.210	.8842	2.384	1.304	.9125	2.489	1.398	.9398	2.590
1.212	.8849	2.386	1.306	.9131	2.491	1.400	.9403	2.592
1.214	.8855	2.388	1.308	.9137	2.493	1.402	.9409	2.594
1.216	.8861	2.391	1.310	.9143	2.495	1.404	.9415	2.596
1.218	.8867	2.393	1.312	.9149	2.497	1.406	.9420	2.599

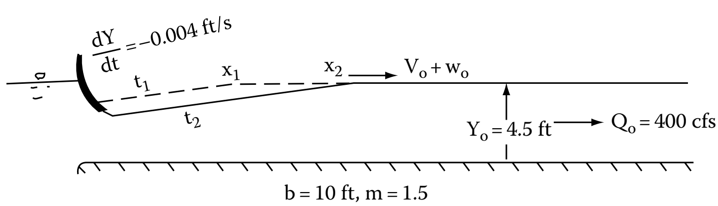
TABLE 6.1 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Trapezoidal Channels

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
1.408	.9426	2.601	1.442	.9522	2.637	1.476	.9616	2.672
1.410	.9432	2.603	1.444	.9527	2.639	1.478	.9622	2.674
1.412	.9437	2.605	1.446	.9533	2.641	1.480	.9627	2.676
1.414	.9443	2.607	1.448	.9538	2.643	1.482	.9633	2.678
1.416	.9449	2.609	1.450	.9544	2.645	1.484	.9638	2.680
1.418	.9454	2.611	1.452	.9550	2.647	1.486	.9644	2.683
1.420	.9460	2.613	1.454	.9555	2.649	1.488	.9649	2.685
1.422	.9466	2.616	1.456	.9561	2.651	1.490	.9655	2.687
1.424	.9471	2.618	1.458	.9566	2.653	1.492	.9660	2.689
1.426	.9477	2.620	1.460	.9572	2.655	1.494	.9666	2.691
1.428	.9482	2.622	1.462	.9577	2.658	1.496	.9671	2.693
1.430	.9488	2.624	1.464	.9583	2.660	1.498	.9677	2.695
1.432	.9494	2.626	1.466	.9589	2.662	1.500	.9682	2.697
1.434	.9499	2.628	1.468	.9594	2.664	1.502	.9688	2.699
1.436	.9505	2.630	1.470	.9600	2.666	1.504	.9693	2.701
1.438	.9510	2.632	1.472	.9605	2.668	1.506	.9699	2.703
1.440	.9516	2.634	1.474	.9611	2.670	1.508	.9700	2.705

The solution of the depth from Equation 6.37 involves a trial process. However, it can be accomplished because both the area A and the stage variable w are a function of the depth Y. In the case of the stage variable w, it is necessary that this functional dependency either be obtained through the use of Table 6.1, or an integration of its defining equation.

EXAMPLE PROBLEM 6.10

Water enters a trapezoidal channel with b = 10 ft, m = 1.5, from a reservoir such that for a long time the depth has been constant throughout the channel at 4.5 ft and the flow rate has been Q = 400 cfs. Suddenly, at the reservoir, the depth of water in the channel is decreased at a rate of dY/dt = −0.004 ft/s. Determine when the depth will be 4.2 ft at a position 2000 ft downstream in the channel, and how far downstream from the beginning of the channel the effect of the transient is noticeable at this time.



Solution

The area and the top width under uniform flow conditions are: A = 75.375 ft² and T = 23.5 ft, and therefore c_o = 10.163 fps, and $V_o = Q_o/A$ = 5.307 fps. Dimensionless variables that can now be computed are

$Y'_o = mY_o/b = 0.675$, $c'_o = \sqrt{m/(gb)}c = 0.06825c = .6937$. From Table 6.1, $w'_o = 1.707$, and it should be noted that c′ in this table agrees with what was computed. Now $w_o = w'_o/0.06825 = 25.011$ fps, and since V − w is constant along a C^- characteristic, it follows that $V(0, t) = w(0, t) + V_o - w_o$, and therefore the slope of any C^+ characteristic can be obtained from

$$\frac{dx}{dt} = V + c = w(0,t) + c(0,t) + V_o - w_o$$

Therefore, we have to find w and c associated with Y = 4.2 ft (Y′ = 0.630). From Table 6.1 c′ = 0.674, and w′ = 1.641, giving c = 9.876 fps, and w = 24.044 fps, and dx/dt = 14.216 fps. Therefore, the time at 2000 for the depth to be 4.2 ft will equal.

$t = t_1 + \Delta t = 0.3/0.004 + 2000/14.216 = 75 + 140.69 = 215.7$ s. The slope of the original C^+ characteristic is $dx/dt(0,0) = V_o + c_o = 15.47$ fps, and therefore the distance x = 215.7(15.47) = 3337 ft.

EXAMPLE PROBLEM 6.11

In the previous problem, the flow rate entering the channel is suddenly decreased at a rate of dQ/dt(0, t) = −0.50 cfs/s instead of the depth. Determine the same quantities.

Solution

In this problem, it is necessary to first determine the flow rate that will occur at the origin when the depth here is 4.2 ft. This can be accomplished by first finding the corresponding velocity from Equation 6.35. This determined velocity is next multiplied by the area A to give Q, and from this Q, the time t_1 is computed. Since the depth is known Δt is determined as in the previous problem. Thus V(0, t) = dx/dt − c = 14.216 − 9.876 = 4.340, and Q = 4.340(10 × 4.2 + 1.5 × 4.2 × 4.2) = 297.1 cfs. Time $t_1 = (400 - 297.1)/0.5 = 205.8$ s. Adding this to the previous Δt gives 220.5 s. The position of the first reduction in depth is x = 220.5(15.47) = 3411.1 ft.

EXAMPLE PROBLEM 6.12

The flow rate in Example Problem 6.10 decreases at a rate of dQ/dt = −0.5 cfs/s at the reservoir. Make a table that shows the depth in the channel at both its beginning and at a position 2000 ft downstream therefrom as a function of time.

Solution

In order to solve the depth at the origin, with the flow rate known here as a function of time, it is necessary to solve the equation $F = A - Q/(w + V_o - w_o) = 0$ by trial using the definition of w as related to the depth Y for each entry in the table. At another position x, the sum of $t_1 + \Delta t = t$ (the time for which this line in the table is to apply) must be solved. When known values are substituted for t_1 and Δt, this produces the equation: $F = (Q_o - Q)/|dQ/dt| + x/(w + V_o - w_o + c) - t = 0$, in which Q = VA, and $V = V(0, t) = w(0, t) + V_o - w_o$, and the area A is obtained from the equation A = [b + mY(0, t)]Y(0, t). The computer programs whose listings are given below carry out this solution. The program is given twice; first in FORTRAN and then in PASCAL. The first portion of the programs read in the problem variables, then the stage function w is integrated over the range of depths from 0.01 to Y_o. Thereafter, the two equations above are solved, and the time, the flow rate, the depth, and the velocity associated with each of these times are printed. The solutions are given below.

For Station x = 0.00 ft

Time (s)	Q	Y	V
0	400.0	4.500	5.307
60	370.0	4.416	5.040
120	340.0	4.329	4.762
180	310.0	4.239	4.470
240	280.0	4.146	4.165
300	250.0	4.048	3.843
360	220.0	3.946	3.503
420	190.0	3.838	3.142
480	160.0	3.725	2.756
540	130.0	3.605	2.341
600	100.0	3.476	1.891

For Station at x = 2000.0			
Time (s)	Q	Y	V
0	400.0	4.500	5.307
60	400.0	4.500	5.307
120	400.0	4.500	5.307
180	375.8	4.433	5.093
240	347.3	4.351	4.831
300	319.0	4.266	4.559
360	290.8	4.180	4.276
420	262.7	4.090	3.981
480	234.9	3.997	3.674
540	207.3	3.901	3.353
600	180.0	3.801	3.016

Listing of FORTRAN program to solve problem, YTIMEX

```
      PARAMETER (N=200)
      REAL W(N),X(20)
      FW(Y)=SQRT(G*(B+FM2*Y)/((B+FM*Y)*Y))
      WRITE(6,*)' Give:Yo,Qo,b,m,dQ/dt,Tstart,Tend,Dt,g,IOUT,
     &Stations'
      READ(5,*)
      YO,QO,B,FM,DQT,TST,TE,DT,G,IOUT,NSTA
      IF(NSTA.LT.1) GO TO 1
      WRITE(6,*)'Give x distances for',NSTA
      READ(5,*)(X(I),I=1,NSTA)
1     DY=(YO-.01)/FLOAT(N-1)
      FM2=2.*FM
      DY2=DY/10.
      DYH=DY2/2.
      Y=.01
      WF1=FW(Y)
      W(1)=0.
      DO 10 I=2,N
      SUM=0.
      DO 5 J=1,10
      WF=FW(Y+DY2*FLOAT(J))
      SUM=SUM+DYH*(WF1+WF)
5     WF1=WF
      W(I)=W(I-1)+SUM
10    Y=Y+DY
      VO=QO/((B+FM*YO)*YO)
      VOW=VO-W(N)
      WRITE(IOUT,110)
      IM=N-1
      NT=(TE-TST)/DT+1.99
      DO 40 I=1,NT
      T=TST+DT*FLOAT(I-1)
      IT=T+.99
      Q=QO-ABS(DQT)*T
      F=(B+FM*YO)*YO-Q/(W(N)+VOW)
```

```
20    F1=F
      Y=.01+FLOAT(IM-1)*DY
      F=(B+FM*Y)*Y-Q/(W(IM)+VOW)
      IF(F.LE.0.) GO TO 30
      IM=IM-1
      IF(IM.GT.0) GO TO 20
30    FAC=F/(F1-F)
      YY=Y-FAC*DY
      V=W(IM)-FAC*(W(IM+1)-W(IM))+VOW
40    WRITE(IOUT,100) IT,Q,YY,V
100   FORMAT(I8,F8.1,2F8.3)
110   FORMAT(' Time(s)',5X,'Q',7X,'Y',7X,'V'
      *,/,1X,32('-'))
      DXO=VO+SQRT(G*(((B+FM*YO)*YO)/(B+FM2*YO)))
      DO 60 K=1,NSTA
      IM=N-1
      WRITE(IOUT,*)' For Station at x =',X(K)
      F=X(K)/DXO-TST
      DO 60 I=1,NT
      T=TST+DT*FLOAT(I-1)
      IT=T+.99
      IF(T*DXO.LE.X(K)) THEN
      WRITE(IOUT,100) IT,QO,YO,VO
      ELSE
45    F1=F
      V=W(IM)+VOW
      Y=.01+FLOAT(IM-1)*DY
      A=(B+FM*Y)*Y
      Q=V*A
F=(QO-Q)/ABS(DQT)+X(K)/(V+SQRT(G*A/(B+FM2*Y)))-T
      IF(F.GT.0.) GO TO 48
      IM=IM-1
      IF(IM.GT.0) GO TO 45
48    FAC=F/(F1-F)
      YY=Y-FAC*DY
      V=W(IM)-FAC*(W(IM+1)-W(IM))+VOW
      Q=V*(B+FM*YY)*YY
      WRITE(IOUT,100) IT,Q,YY,V
      ENDIF
60    CONTINUE
      STOP
      END
```

Listing of PASCAL program to Solve Problem

```
Program Ytimex;
Const N=200;
Var w:array [1..200] of real;x:array [1..20] of real;
 I,J,K,IT,IM,NT,Nsta:integer;
 Yo,Qo,b,m,dQt,Tst,Te,Dt,g,DY,DY2,FM2,DYH,WF1,WF,SUM,VOW,
 Q,F,FAC,T,YY,F1,V,A,Vo,dxo:real;
Function FW(Y:real):real;
```

```
 Begin FW:=Sqrt(g*(b+FM2*Y)/((b+m*Y)*Y)) End;
Label L20,L30,L45,L48; Var Y:real;
BEGIN
 Writeln('Give: Yo,Qo,b,m,dQt,Tstart,Tend,Dt,g,Nsta');
 Readln(Yo,Qo,b,m,dQt,Tst,Te,Dt,g,Nsta);
 If Nsta>0 Then begin
  Writeln('Give x distances for',Nsta:3,' stations');
  For I:=1 to Nsta-1 do Read(x[I]);Readln(x[Nsta]) end;
    DY:=(Yo-0.01)/(N-1);FM2:=2*m;DY2:=DY/10;DYH:=DY2/2;
 Y:=0.01; WF1:=FW(Y);w[1]:=0;
 For I:=2 to N do Begin SUM:=0; For J:=1 to 10 do begin
  WF:=FW(Y+DY2*J);SUM:=SUM+DYH*(WF+WF1);WF1:=WF end;
 w[I]:=w[I-1]+SUM; Y:=Y+DY End;
 Vo:=Qo/((b+m*Yo)*Yo); VOW:=Vo-w[N];
 Writeln(' Time(s) Q Y V');
 Writeln('----------');
 IM:=N-1;NT:=Trunc((Te-Tst)/DT+1.9);
 For I:=1 to NT do Begin
  T:=Tst+DT*(I-1); IT:=Trunc(T); Q:=Qo-abs(dQt)*T;
  F:=(b+m*Yo)*Yo-Q/(w[N]+VOW); L20:F1:=F;Y:=0.01+(IM-1)*DY;
  F:=(b+m*Y)*Y-Q/(w[IM]+VOW); If F <= 0 then GoTo L30;IM:=IM-1;
  If IM > 0 then GoTo L20; L30:FAC:=F/(F1-F);YY:=Y-FAC*DY;
  V:=w[IM]-FAC*(w[IM+1]-w[IM])+VOW;
  Writeln(IT:8,Q:8:1,YY:8:3,V:8:3) End;
 dxo:=Vo+sqrt(g*(((b+m*Yo)*Yo)/(b+FM2*Yo)));
 For K:=1 to Nsta do Begin
  IM:=N-1;Writeln('For Station at x =',x[K]:10:0);
  F:=x[K]/dxo-Tst; For I:=1 to NT do begin T:=Tst+DT*(I-1);
  IT:=Trunc(T+0.99);
   If T*dxo <= x[K] then Writeln(IT:8,Qo:8:1,Yo:8:3,Vo:8:3) else
   Begin L45:F1:=F; V:=w[IM]+VOW;
     Y:=0.01+(IM-1)*DY;A:=(b+m*Y)*Y; Q:=V*A;
   F:=(Qo-Q)/abs(dQt)+x[K]/(V+sqrt(g*A/(b+FM2*Y)))-T;
   If F>0 then GoTo L48;IM:=IM-1; If IM>0 then GoTo L45;
L48:FAC:=F/(F1-F);YY:=Y-FAC*DY;V:=w[IM]-FAC*(w[IM+1]-w[IM])+VOW;
   Q:=V*(b+m*YY)*YY;Writeln(IT:8,Q:8:1,YY:8:3,V:8:3) End; end;
 End;
END.
```

Program to solve Problem YTIMEX.C

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
const N=200; float w[200],x[20],g,b,fm2,fm;
float fw(float y){return sqrt(g*(b+fm2*y)/((b+fm*y)*y));}
void main(void){int i,h,nsta,im,nt,it,j,k; FILE *out; char fname[20];
 float yo,qo,dqt,tst,te,dy,dy2,wf1,sum,vo,dt,wf,q,vow,t,yy,v,dxo,f,\
   f1,fac,a,dyh,y;
 printf("Give:Yo,Qo,b,m,dQ/dt,Tstart,Tend,Dt,g,Stations\n");
 scanf("%f %f %f %f %f %f %f %f %f %d",&yo,&qo,&b,&fm,&dqt,&tst,&te,\
   &dt,&g,&nsta);
```

```
 if(nsta>0){printf("Give x distances for %d stations\n",nsta);
   for(i=0;i<nsta;i++)scanf("%f",&x[i]);}
 printf("Give name for output file\n");scanf("%s",fname);
   out=fopen(fname,"wt");
 dy=(yo-.01)/(float)(N-1);fm2=2.*fm;dy2=dy/10.;dyh=dy2/2.;
   y=.01;wf1=fw(y);w[0]=0.;
 for(i=1;i<N;i++){sum=0.; for(j=1;j<=10;j++)
   {wf=fw(y+dy2*(float)j);sum+=dyh*(wf1+wf);wf1=wf;}
 w[i]=w[i-1]+sum; y+=dy;}
 vo=qo/((b+fm*yo)*yo); vow=vo-w[N-1];
 fprintf(out,"Time(s)   Q   Y   V\n");im=N-2;nt=(te-tst)/dt+1.99;
 for(i=0;i<nt;i++){t=tst+dt*(float)i; it=t+.99; q=qo-fabs(dqt)*t;
 f=(b+fm*yo)*yo-q/(w[N-1]+vow); L20:f1=f; y=.01+(float)im*dy;
 f=(b+fm*y)*y-q/(w[im]+vow); if(f>0.){im–;if(im>=0) goto L20;}
 fac=f/(f1-f); yy=y-fac*dy; v=w[im]-fac*(w[im+1]-w[im])+vow;
 fprintf(out,"%8d %7.1f %7.3f %7.3f\n",it,q,yy,v);} //End for i
 dxo=vo+sqrt(g*(((b+fm*yo)*yo)/(b+fm2*yo)));
 for(k=0;k<nsta;k++)\
   {im=N-2;fprintf(out,"\nFor Station at x =%8.1f\n",x[k]);
  f=x[k]/dxo-tst;
 for(i=0;i<nt;i++){t=tst+dt*(float)i; it=t+.99;
 if(t*dxo<=x[k]) fprintf(out,"%8d %7.1f %7.3f %7.3f\n",it,qo,yo,vo);
    else
 {L45:f1=f;v=w[im]+vow; y=.01+(float)im*dy; a=(b+fm*y)*y; q=v*a;
  f=(qo-q)/fabs(dqt)+x[k]/(v+sqrt(g*a/(b+fm2*y)))-t;
  if(f<=0.){im–;if(im>=0) goto L45;}
 fac=f/(f1-f); yy=y-fac*dy; v=w[im]-fac*(w[im+1]-w[im])+vow;
   q=v*(b+fm*yy)*yy;
 fprintf(out,"%8d %7.1f %7.3f %7.3f\n",it,q,yy,v);}
 }} //End of for k & for i
}
```

EXAMPLE PROBLEM 6.13

Previously, it has been shown that, if a point outflow takes place at a intermediate position in a rectangular channel with an initially uniform flow, the depth at this point is uniquely determined by the magnitude of this outflow at any given time. Prove that for a nonrectangular channel the same is true, i.e., the relationship $\Delta Q = 2A(w_o - w)$ holds in which ΔQ is the point outflow. For a trapezoidal channel with b = 3 m, and m = 1.2, with an initial uniform flow of $Y_o = 2$ m and $Q_o = 12\,m^3/s$, make up a table that gives the depth, the velocity and the flow rate at this point, as well as these variables at positions x = 100 m, x = 200 m, x = −50 m, and x = −100 m for several time steps if the rate of change of the point outflow is $d(\Delta Q)/dt = 0.1\ (m/s)^2$ for 50 s and remains constant at $Q = 5\,m^3/s$, thereafter.

Solution

The solution of the requested flow variables involves three parts: (1) at the origin where the outflow takes place, (2) at the downstream positions, and (3) at the upstream positions.

At the origin, x = 0: Here, the point outflow ΔQ equals the difference between the upstream and the downstream flow rates, or $\Delta Q = Q_u - Q_d = A(V_u - V_d) = A[V_o + w_o - w - (w + V_o - w_o)] = 2A(w_o - w)$. (Note A and w do not have subscripts since the upstream and the downstream depths are the same at x = 0.) Thus, we have proven that for a given initial uniform flow in a nonrectangular channel that produces w_o there is a depth that produces w that satisfies the equation $F = .5\Delta Q + A(w - w_o) = 0$. Therefore, at x = 0, this equation must be solved for a depth Y with $\Delta Q = \{d(\Delta Q)/dt\}t$ computed for any time step until ΔQ not longer changes. Once Y is known, the other variables are computed from $V_d = w + V_o - w_o$, $Q_d = AV_d$, $V_u = V_o + w_o - w$, $Q_u = AV_u$. The first table below gives these computed values for times using t = 10 s. For example at t = 40 s: Q = .1(40) = 4 m^3/s; the solution of the above implicit equation gives Y = 1.927 m, and A = 8.310 m^2, V_d = 0.916 m/s, V_u = 1.306 m/s, Q_d = 9.4 m^3/s, and Q_u = 13.4 m^3/s. (See a home work problem to write a computer program to carry out these computations.)

At downstream positions where x is positive: At these positions we seek the C^+ characteristic along which $t_l + \Delta t = t$, in which $t_l = \Delta Q/\{d(\Delta Q)/dt\}$ and $\Delta t = x/(V_d + c_d)$. Since, as shown above, there is a unique relationship between ΔQ and the depth, this equation is solved for Y, or the Newton Method obtains Y that satisfies: $F = \Delta Q/\{d(\Delta Q)/dt\} + x/(V_d + c_d) - t = 0$. Note, ΔQ in this equation is the outflow associated with the C^+ characteristics through the point (x, t) and is not the ΔQ at time t, thus in the tables below it is denoted by ΔQ_c. For example, at t = 40 s and x = 100 m, the solution of the above implicit equation produces Y = 1.967 m, from which A = 8.577 m^2, V = 1.023 m/s, Q = 11.4 m^3/s. The constant outflow along the C^+ characteristic through point (100, 40) of the xt plane is ΔQ = 1.865 m^3/s. The second table below provides these values for t = 0, 10, 20, …, 70 s. Note that the initial uniform flow conditions exist prior to the time when the wave reaches the position x = 100 or x = 200 m. The position of the wave is shown in the last column of this table.

At upstream positions, where x is negative: At these positions we seek the C^- characteristic (that is a straight line) along which Y, V_u, Q_u, and w are constant such that $t_l + \Delta t = t$, in which $t_l = \Delta Q/\{d(\Delta Q)/dt\}$ and $\Delta t = x/(V_u - c_u)$. Note that Δt is not the same as for downstream positions at time t, i.e., the depth that satisfies the equation $F = \Delta Q/\{d(\Delta Q)/dt\} + x/(V_u - c_u) - t = 0$ will not equal the depth that satisfies the downstream implicit equation for the same time t. This means that, the C^- characteristic we are now seeking originates at a different time t from x = 0. For example, at t = 40 s and x = −50 m, the solution of the above implicit equation produces: Y = 1.965 m, from which V_u = 1.204 m/s, Q_u = 12.7 m^3/s, c_u = 3.66 m/s, and ΔQc = 1.963 m^3/s. It should be noted that when the upstream flow becomes critical, or when $|V_u| = c$, or $Q_u^2T/(gA^3) = 1$, it limits the amount of the point outflow.

$$Y_o = 2.000, \quad Q_o = 12.0, \quad V_o = 1.111, \quad A_o = 10.800, \quad c_o = 3.686, \quad w_o = 8.996$$

At the origin, x = 0

t (s)	ΔQ (cms)	Y (m)	V_d (m/s)	V_u (m/s)	Q_d (cms)	Q_u (cms)	C (m/s)	W (m/s)
0	.000	2.000	1.111	1.111	12.0	12.0	3.686	9.00
10	1.000	1.982	1.064	1.158	11.3	12.3	3.672	8.95
20	2.000	1.964	1.016	1.206	10.7	12.7	3.658	8.90
30	3.000	1.946	.967	1.256	10.0	13.0	3.644	8.85
40	4.000	1.927	.916	1.306	9.4	13.4	3.629	8.80
50	5.000	1.908	.863	1.359	8.7	13.7	3.614	8.75
60	5.000	1.908	.863	1.359	8.7	13.7	3.679	8.95
70	5.000	1.908	.863	1.359	8.7	13.7	3.667	8.90

	x = 100 m							x = 200 m						
t (s)	ΔQ (cms)	Y (m)	V (m/s)	Q (cms)	c (m/s)	ΔQc (cms)	w (m/s)	Y (m)	V (m/s)	Q (cms)	c (m/s)	ΔQc (cms)	w (m/s)	x_w (m)
0	.000	2.000	1.111	12.0	3.69	.000	9.00	2.000	1.111	12.0	3.69	.000	9.00	0
10	1.000	2.000	1.111	12.0	3.69	.000	9.00	2.000	1.111	12.0	3.69	.000	9.00	48
20	2.000	2.000	1.111	12.0	3.69	.000	9.00	2.000	1.111	12.0	3.69	.000	9.00	96
30	3.000	1.984	1.069	11.4	3.67	.892	8.95	2.000	1.111	12.0	3.69	.000	9.00	144
40	4.000	1.967	1.023	10.8	3.66	1.865	8.91	2.000	1.111	12.0	3.69	.000	9.00	192
50	5.000	1.949	.975	10.1	3.65	2.836	8.86	1.986	1.074	11.5	3.67	.789	8.96	240
60	5.000	1.931	.926	9.5	3.63	3.806	8.81	1.969	1.029	10.9	3.66	1.736	8.91	−288
70	5.000	1.912	.875	8.9	3.62	4.774	8.76	1.952	.983	10.2	3.65	2.681	8.87	336

(*continued*)

(continued)

	x = −50 m							x = −100 m						
t (s)	ΔQ (cms)	Y (m)	V (m/s)	Q (cms)	c (m/s)	ΔQc (cms)	w (m/s)	Y (m)	V (m/s)	Q (cms)	c (m/s)	ΔQc (cms)	w (m/s)	x_w (m)
0	.000	2.000	1.111	12.0	3.69	.000	9.00	2.000	1.111	12.0	3.69	.000	9.00	0
10	1.000	2.000	1.111	12.0	3.69	.000	9.00	2.000	1.111	12.0	3.69	.000	9.00	−26
20	2.000	1.999	1.114	12.0	3.68	.055	8.99	2.000	1.111	12.0	3.69	.000	9.00	−51
30	3.000	1.982	1.159	12.4	3.67	1.011	8.95	2.000	1.111	12.0	3.69	.000	9.00	−77
40	4.000	1.965	1.204	12.7	3.66	1.963	8.90	1.998	1.116	12.0	3.68	.106	8.99	−103
50	5.000	1.948	1.251	13.0	3.65	2.912	8.86	1.982	1.159	12.4	3.67	1.020	8.95	−129
60	5.000	1.930	1.299	13.3	3.63	3.856	8.81	1.966	1.203	12.7	3.66	1.929	8.90	−154
70	5.000	1.912	1.348	13.6	3.62	4.797	8.76	1.949	1.247	13.0	3.65	2.832	8.86	−180

Note: While, it is not possible to integrate the stage variable w (or w′) in the closed form to obtain an algebraic equation that gives w′ as a function of y′, it is possible to take the data in Table 6.1 and fit it to a model by a least squared regression analysis. Once such an equation is available, it can be used instead of looking up values for w′ corresponding to y′, or by finding y′ corresponding to w′ from the table, to implicitly solve y′ for a given w′. Such a regression fit using a 6 degree polynomial using the data in Table 6.1 for a trapezoidal channel gives

$$w' = 0.140785 + 5.254419y' - 11.42496y'^2 + 19.398805y'^3 - 18.48886y'^4 + 8.998029y'^5 - 1.7405045y'^6$$

Average absolute difference = 0.003468

The multiple regression correlation coefficient for this fit is $R^2 = 0.9999$, with the major deviations occurring for small values of y′, which would seldom be used. Thus, the above equation might be used to replace Table 6.1 especially in light of the approximations used in the theory and if changes in w′ are relative small.

More accurate estimates of w′ might be obtained by dividing Table 6.1 into several parts and performing a regression fit for each part of this table. Using three parts, Part 1 with y′ from 0.01 to 0.5080, Part 2 with y′ from 0.5100 to 1.008, and Part 3 with y′ from 1.010 to 1.508 gives the following three sixth order polynomial equations (with the average absolute difference between the w′ in the table and that given after each equation.)

Part 1 ($y' = 0.01$ to $y' = 0.508$)

$$w' = 0.07378655 + 8.18346421y' - 48.52012353y'^2 + 228.8879673y'^3 - 618.51160685y'^4 + 864.136452y'^5 - 483.52952655y'^6$$

Average absolute difference = 0.0012065 = 1.2065×10^{-3}

Part 2 ($y' = 0.51$ to $y' = 1.008$)

$$w' = 0.32126859 + 3.25239439y' - 3.24689267y'^2 + 3.396349093y'^3 - 2.739827y'^4 + 0.95440249y'^5 - 0.1664955y'^6$$

Average absolute difference = 0.000003748 = 3.7485×10^{-6}

Part 3 ($y' = 1.01$ to $y' = 1.508$)

$$w' = 0.39850342 + 2.74650447y' - 1.85174341y'^2 + 1.32995644y'^3 - 0.64462275y'^4 + 0.18036973y'^5 - 0.02192338y'^6$$

$$\text{Average absolute difference} = 0.00001001 = 1.0009 \times 10^{-5}$$

Fitting the data in Table 6.2, which follows, for a circular channel with a fourth degree polynomial equation produces a multiple regression correlation coefficient equally close to unity. This equation is

$$w' = -0.3047431 + 6.0278815y' - 10.9985247y'^2 + 11.9441034y'^3 - 4.97205264y'^4$$

$$\text{Average absolute difference} = 0.01116496 = 1.116496 \times 10^{-2}$$

Using three parts: Part #1 $y' = 0$ to $y' = .3206$

$$w' = 0.1951268 + 10.8143864y' - 61.1991232y'^2 + 206.7081028y'^3 - 261.8789472y'^4$$

$$\text{Average absolute difference} = 0.0067979 = 6.79793 \times 10^{-2}$$

Using three parts: Part #2 $y' = 0.322$ to $y' = .6566$

$$w' = 0.5187091 + 3.9167468y' - 4.332116y'^2 + 3.6181778y'^3 - 1.4055572y'^4$$

$$\text{Average absolute difference} = 0.0000079713 = 7.9713 \times 10^{6}$$

Using three parts: Part #3 $y' = .6580$ to $y' = .9926$

$$w' = -0.5316203 + 9.2891735y' - 14.5119655y'^2 + 12.0485209y'^3 - 3.9610566y'^4$$

$$\text{Average absolute difference} = 0.000086512 = 8.651189 \times 10^{-5}$$

6.8.3 Circular Channels

The utilization of the stage function w works equally well for circular sections. In this section, the development of dimensionless variables for use in solving unsteady problems in circular channels will be discussed. In a circular channel, the celerity of a gravity wave can be written in terms of the auxiliary angle β as follows:

$$c^2 = \frac{gD}{4}\frac{[\beta - \cos(\beta)\sin(\beta)]}{\sin(\beta)}$$

or if a dimensionless celerity is defined by

$$c = c'\sqrt{gD}$$

TABLE 6.2
Dimensionless Variables Associated with Stage Variable, w, for Circular Sections

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.0066	.0664	.0684	.0644	.2086	.6197	.1288	.2972	.8746
.0014	.0306	.0911	.0658	.2109	.6264	.1302	.2989	.8793
.0028	.0432	.1291	.0672	.2132	.6330	.1316	.3006	.8839
.0042	.0529	.1582	.0686	.2154	.6395	.1330	.3022	.8886
.0056	.0611	.1827	.0700	.2176	.6460	.1344	.3038	.8932
.0070	.0684	.2044	.0714	.2198	.6524	.1358	.3055	.8978
.0084	.0749	.2239	.0728	.2220	.6587	.1372	.3071	.9024
.0098	.0809	.2419	.0742	.2242	.6650	.1386	.3087	.9069
.0112	.0865	.2586	.0756	.2263	.6712	.1400	.3103	.9114
.0126	.0918	.2743	.0770	.2284	.6774	.1414	.3119	.9159
.0140	.0967	.2892	.0784	.2305	.6835	.1428	.3135	.9204
.0154	.1015	.3033	.0798	.2326	.6895	.1442	.3151	.9249
.0168	.1060	.3168	.0812	.2347	.6955	.1456	.3167	.9293
.0182	.1104	.3297	.0826	.2367	.7015	.1470	.3183	.9337
.0196	.1145	.3422	.0840	.2388	.7073	.1484	.3198	.9381
.0210	.1186	.3542	.0854	.2408	.7132	.1498	.3214	.9425
.0224	.1225	.3658	.0868	.2428	.7190	.1512	.3230	.9468
.0238	.1263	.3771	.0882	.2448	.7247	.1526	.3245	.9511
.0252	.1299	.3880	.0896	.2468	.7304	.1540	.3261	.9554
.0266	.1335	.3986	.0910	.2487	.7361	.1554	.3276	.9597
.0280	.1370	.4090	.0924	.2507	.7417	.1568	.3291	.9640
.0294	.1404	.4191	.0938	.2526	.7472	.1582	.3307	.9682
.0308	.1437	.4289	.0952	.2545	.7528	.1596	.3322	.9725
.0322	.1470	.4385	.0966	.2564	.7582	.1610	.3337	.9767
.0336	.1502	.4480	.0980	.2583	.7637	.1624	.3352	.9808
.0350	.1533	.4572	.0994	.2602	.7691	.1638	.3367	.9850
.0364	.1564	.4662	.1008	.2621	.7744	.1652	.3382	.9892
.0378	.1594	.4751	.1022	.2639	.7798	.1666	.3397	.9933
.0392	.1623	.4838	.1036	.2658	.7850	.1680	.3412	.9974
.0406	.1652	.4924	.1050	.2676	.7903	.1694	.3427	1.002
.0420	.1681	.5008	.1064	.2694	.7955	.1708	.3441	1.006
.0434	.1709	.5090	.1078	.2712	.8007	.1722	.3456	1.010
.0448	.1736	.5171	.1092	.2730	.8058	.1736	.3471	1.0137
.0462	.1763	.5251	.1106	.2748	.8109	.1750	.3485	1.0177
.0476	.1790	.5330	.1120	.2766	.8160	.1764	.3500	1.0217
.0490	.1817	.5408	.1134	.2784	.8211	.1778	.3514	1.0257
.0504	.1843	.5484	.1148	.2801	.8261	.1792	.3529	1.0297
.0518	.1868	.5560	.1162	.2819	.8311	.1806	.3543	1.0336
.0532	.1894	.5634	.1176	.2836	.8360	.1820	.3558	1.0376
.0546	.1919	.5708	.1190	.2854	.8409	.1834	.3572	1.0415
.0560	.1944	.5780	.1204	.2871	.8458	.1848	.3586	1.0454
.0574	.1968	.5852	.1218	.2888	.8507	.1862	.3601	1.0493
.0588	.1992	.5922	.1232	.2905	.8555	.1876	.3615	1.0532
.0602	.2016	.5992	.1246	.2922	.8603	.1890	.3629	1.0571
.0616	.2040	.6061	.1260	.2939	.8651	.1904	.3643	1.0609
.0630	.2063	.6130	.1274	.2956	.8698	.1918	.3657	1.0647

TABLE 6.2 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Circular Sections

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.1932	.3671	1.0686	.2576	.4279	1.2307	.3220	.4836	1.3720
.1946	.3685	1.0724	.2590	.4292	1.2339	.3234	.4848	1.3749
.1960	.3699	1.0762	.2604	.4304	1.2372	.3248	.4859	1.3778
.1974	.3713	1.0799	.2618	.4317	1.2404	.3262	.4871	1.3807
.1988	.3727	1.0837	.2632	.4329	1.2437	.3276	.4883	1.3836
.2002	.3741	1.0875	.2646	.4342	1.2469	.3290	.4894	1.3864
.2016	.3754	1.0912	.2660	.4354	1.2501	.3304	.4906	1.3893
.2030	.3768	1.0949	.2674	.4367	1.2533	.3318	.4918	1.3921
.2044	.3782	1.0986	.2688	.4379	1.2565	.3332	.4929	1.3950
.2058	.3796	1.1023	.2702	.4391	1.2597	.3346	.4941	1.3978
.2072	.3809	1.1060	.2716	.4404	1.2629	.3360	.4952	1.4006
.2086	.3823	1.1097	.2730	.4416	1.2661	.3374	.4964	1.4035
.2100	.3836	1.1133	.2744	.4428	1.2692	.3388	.4975	1.4063
.2114	.3850	1.1170	.2758	.4441	1.2724	.3402	.4987	1.4091
.2128	.3863	1.1206	.2772	.4453	1.2756	.3416	.4999	1.4119
.2142	.3877	1.1242	.2786	.4465	1.2787	.3430	.5010	1.4147
.2156	.3890	1.1278	.2800	.4477	1.2818	.3444	.5022	1.4175
.2170	.3904	1.1314	.2814	.4490	1.2849	.3458	.5033	1.4203
.2184	.3917	1.1350	.2828	.4502	1.2881	.3472	.5045	1.4231
.2198	.3930	1.1386	.2842	.4514	1.2912	.3486	.5056	1.4258
.2212	.3944	1.1421	.2856	.4526	1.2943	.3500	.5068	1.4286
.2226	.3957	1.1457	.2870	.4538	1.2974	.3514	.5079	1.4314
.2240	.3970	1.1492	.2004	.4550	1.3004	.3528	.5091	1.4341
.2254	.3983	1.1527	.2898	.4562	1.3035	.3542	.5102	1.4369
.2268	.3997	1.1562	.2912	.4574	1.3066	.3556	.5113	1.4396
.2282	.4010	1.1597	.2926	.4587	1.3096	.3570	.5125	1.4423
.2296	.4023	1.1632	.2940	.4599	1.3127	.3584	.5136	1.4451
.2310	.4036	1.1667	.2954	.4611	1.3157	.3598	.5148	1.4478
.2324	.4049	1.1701	.2968	.4623	1.3187	.3612	.5159	1.4505
.2338	.4062	1.1736	.2982	.4635	1.3218	.3626	.5170	1.4532
.2352	.4075	1.1770	.2996	.4647	1.3248	.3640	.5182	1.4559
.2366	.4088	1.1805	.3010	.4658	1.3278	.3654	.5193	1.4586
.2380	.4101	1.1839	.3024	.4670	1.3308	.3668	.5205	1.4613
.2394	.4114	1.1873	.3038	.4682	1.3338	.3682	.5216	1.4640
.2408	.4127	1.1907	.3052	.4694	1.3368	.3696	.5227	1.4667
.2422	.4140	1.1941	.3066	.4706	1.3398	.3710	.5239	1.4693
.2436	.4152	1.1975	.3080	.4718	1.3427	.3724	.5250	1.4720
.2450	.4165	1.2008	.3094	.4730	1.3457	.3738	.5261	1.4747
.2464	.4178	1.2042	.3108	.4742	1.3487	.3752	.5273	1.4773
.2478	.4191	1.2075	.3122	.4753	1.3516	.3766	.5284	1.4800
.2492	.4203	1.2109	.3136	.4765	1.3545	.3780	.5295	1.4826
.2506	.4216	1.2142	.3150	.4777	1.3575	.3794	.5307	1.4853
.2520	.4229	1.2175	.3164	.4789	1.3604	.3808	.5318	1.4879
.2534	.4241	1.2208	.3178	.4801	1.3633	.3822	.5329	1.4905
.2548	.4254	1.2241	.3192	.4812	1.3662	.3836	.5340	1.4932
.2562	.4267	1.2274	.3206	.4824	1.3691	.3850	.5352	1.4958

(*continued*)

TABLE 6.2 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Circular Sections

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.3864	.5363	1.4984	.4508	.5876	1.6131	.5152	.6388	1.7182
.3878	.5374	1.5010	.4522	.5887	1.6154	.5166	.6399	1.7204
.3892	.5385	1.5036	.4536	.5898	1.6178	.5180	.6411	1.7225
.3906	.5397	1.5062	.4550	.5909	1.6202	.5194	.6422	1.7247
.3920	.5408	1.5088	.4564	.5920	1.6226	.5208	.6433	1.7269
.3934	.5419	1.5114	.4578	.5931	1.6249	.5222	.6444	1.7291
.3948	.5430	1.5140	.4592	.5942	1.6273	.5236	.6456	1.7312
.3962	.5442	1.5165	.4606	.5953	1.6296	.5250	.6467	1.7334
.3976	.5453	1.5191	.4620	.5965	1.6320	.5264	.6476	1.7356
.3990	.5464	1.5217	.4634	.5976	1.6343	.5278	.6489	1.7377
.4004	.5475	1.5242	.4648	.5987	1.6367	.5292	.6501	1.7399
.4018	.5486	1.5260	.4662	.5998	1.6390	.5306	.6512	1.7420
.4032	.5497	1.5293	.4676	.6009	1.6413	.5320	.6523	1.7442
.4046	.5509	1.5319	.4690	.6020	1.6437	.5334	.6535	1.7463
.4060	.5520	1.5344	.4704	.6031	1.6460	.5348	.6546	1.7485
.4074	.5531	1.5370	.4718	.6042	1.6483	.5362	.6557	1.7506
.4088	.5542	1.5395	.4732	.6053	1.6506	.5376	.6569	1.7527
.4102	.5553	1.5420	.4746	.6065	1.6529	.5390	.6580	1.7549
.4116	.5565	1.5445	.4760	.6076	1.6552	.5404	.6591	1.7570
.4130	.5576	1.5470	.4774	.6087	1.6575	.5418	.6603	1.7591
.4144	.5587	1.5495	.4788	.6098	1.6598	.5432	.6614	1.7612
.4158	.5598	1.5520	.4802	.6109	1.6621	.5446	.6626	1.7634
.4172	.5609	1.5545	.4816	.6120	1.6644	.5460	.6637	1.7655
.4186	.5620	1.5570	.4830	.6131	1.6667	.5474	.6648	1.7676
.4200	.5631	1.5595	.4844	.6142	1.6690	.5488	.6660	1.7697
.4214	.5643	1.5620	.4858	.6153	1.6713	.5502	.6671	1.7718
.4228	.5654	1.5645	.4872	.6165	1.6735	.5516	.6683	1.7739
.4242	.5665	1.5670	.4886	.6176	1.6758	.5530	.6694	1.7760
.4256	.5676	1.5694	.4900	.6187	1.6781	.5544	.6706	1.7781
.4270	.5687	1.5719	.4914	.6198	1.6803	.5558	.6717	1.7801
.4284	.5698	1.5744	.4928	.6209	1.6826	.5572	.6729	1.7822
.4298	.5709	1.5768	.4942	.6220	1.6848	.5586	.6740	1.7843
.4312	.5720	1.5793	.4956	.6231	1.6871	.5600	.6752	1.7864
.4326	.5731	1.5817	.4970	.6243	1.6893	.5614	.6763	1.7884
.4340	.5743	1.5841	.4984	.6254	1.6916	.5620	.6775	1.7905
.4354	.5754	1.5866	.4998	.6265	1.6938	.5642	.6766	1.7926
.4368	.5765	1.5890	.5012	.6276	1.6961	.5656	.6796	1.7946
.4382	.5776	1.5914	.5026	.6287	1.6983	.5670	.6809	1.7967
.4396	.5787	1.5939	.5040	.6298	1.7005	.5684	.6821	1.7988
.4410	.5798	1.5963	.5054	.6310	1.7027	.5698	.6833	1.8008
.4424	.5809	1.5987	.5068	.6321	1.7049	.5712	.6844	1.8029
.4438	.5820	1.6011	.5082	.6332	1.7072	.5726	.6856	1.8049
.4452	.5831	1.6035	.5096	.6343	1.7094	.5740	.6867	1.8069
.4466	.5843	1.6059	.5110	.6354	1.7116	.5754	.6879	1.8090
.4480	.5854	1.6083	.5124	.6366	1.7138	.5768	.6891	1.8110
.4494	.5865	1.6107	.5138	.6377	1.7160	.5782	.6902	1.6130

TABLE 6.2 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Circular Sections

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.5796	.6914	1.8151	.6454	.7485	1.9066	.7112	.8119	1.9910
.5810	.6926	1.8171	.6468	.7498	1.9084	.7126	.8133	1.9927
.5824	.6938	1.8191	.6482	.7510	1.9103	.7140	.8148	1.9945
.5838	.6949	1.8211	.6496	.7523	1.9122	.7154	.8162	1.9962
.5852	.6961	1.8231	.6510	.7536	1.9140	.7168	.8177	1.9979
.5866	.6973	1.8251	.6524	.7549	1.9159	.7182	.8192	1.9996
.5880	.6985	1.8272	.6538	.7561	1.9177	.7196	.8206	2.0013
.5894	.6996	1.8292	.6552	.7574	1.9196	.7210	.8221	2.0030
.5908	.7008	1.8312	.6566	.7587	1.9214	.7224	.8236	2.0047
.5922	.7020	1.8331	.6580	.7600	1.9233	.7238	.8251	2.0064
.5936	.7032	1.8351	.6594	.7613	1.9251	.7252	.8266	2.0081
.5950	.7044	1.8371	.6608	.7626	1.9269	.7266	.8281	2.0098
.5964	.7056	1.8391	.6622	.7639	1.9288	.7280	.8296	2.0115
.5976	.7068	1.8411	.6636	.7652	1.9306	.7294	.8311	2.0132
.5992	.7080	1.8431	.6650	.7665	1.9324	.7308	.8326	2.0149
.6006	.7092	1.8451	.6664	.7678	1.9343	.7322	.8342	2.0165
.6020	.7103	1.8470	.6678	.7692	1.9361	.7336	.8357	2.0182
.6034	.7115	1.8490	.6692	.7705	1.9379	.7350	.8372	2.0199
.6048	.7127	1.8510	.6706	.7718	1.9397	.7364	.8388	2.0216
.6062	.7139	1.8529	.6720	.7731	1.9415	.7378	.8403	2.0232
.6076	.7152	1.8549	.6734	.7745	1.9433	.7392	.8419	2.0249
.6090	.7164	1.8568	.6748	.7758	1.9451	.7406	.8435	2.0265
.6104	.7176	1.8588	.6762	.7771	1.9469	.7420	.8450	2.0282
.6118	.7188	1.8607	.6776	.7785	1.9487	.7434	.8466	2.0299
.6132	.7200	1.8627	.6790	.7798	1.9505	.7448	.8482	2.0315
.6146	.7212	1.8646	.6804	.7812	1.9523	.7462	.8498	2.0332
.6160	.7224	1.8666	.6818	.7825	1.9541	.7476	.8514	2.0348
.6174	.7236	1.8665	.6832	.7839	1.9559	.7490	.8530	2.0365
.6188	.7249	1.8704	.6846	.7852	1.9577	.7504	.8546	2.0381
.6202	.7261	1.8724	.6860	.7866	1.9595	.7518	.8562	2.0397
.6216	.7273	1.8743	.6874	.7880	1.9613	.7532	.8579	2.0414
.6230	.7285	1.8762	.6888	.7893	1.9630	.7546	.8595	2.0430
.6244	.7298	1.8781	.6902	.7907	1.9648	.7560	.8612	2.0446
.6258	.7310	1.8901	.6916	.7921	1.9666	.7574	.8628	2.0462
.6272	.7322	1.8820	.6930	.7935	1.9683	.7588	.8645	2.0479
.6286	.7335	1.8839	.6944	.7949	1.9701	.7602	.8661	2.0495
.6300	.7347	1.8856	.6958	.7962	1.9719	.7616	.8678	2.0511
.6314	.7359	1.8677	.6972	.7976	1.9736	.7630	.8695	2.0527
.6328	.7372	1.8696	.6986	.7990	1.9754	.7644	.8712	2.0543
.6342	.7384	1.8915	.7000	.8004	1.9771	.7658	.8729	2.0559
.6356	.7397	1.8934	.7014	.8019	1.9789	.7672	.8746	2.0575
.6370	.7409	1.8953	.7028	.8033	1.9806	.7686	.8763	2.0591
.6384	.7422	1.8972	.7042	.8047	1.9824	.7700	.8781	2.0607
.6398	.7434	1.8990	.7056	.8061	1.9841	.7714	.8798	2.0623
.6412	.7447	1.9009	.7070	.8075	1.9858	.7728	.8816	2.0639
.6426	.7460	1.9028	.7084	.8090	1.9876	.7742	.8833	2.0655
.6440	.7472	1.9047	.7098	.8104	1.9893	.7756	.8851	2.0671

(continued)

TABLE 6.2 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Circular Sections

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.7770	.8869	2.0687	.8414	.9825	2.1378	.9058	1.131	2.1992
.7784	.8886	2.0702	.8428	.9850	2.1392	.9072	1.136	2.2005
.7798	.8904	2.0718	.8442	.9875	2.1406	.9086	1.140	2.2017
.7812	.8922	2.0734	.8456	.9901	2.1421	.9100	1.145	2.2029
.7826	.8941	2.0749	.8470	.9926	2.1435	.9114	1.150	2.2042
.7840	.8959	2.0765	.8484	.9952	2.1449	.9128	1.154	2.2054
.7854	.8977	2.0781	.8498	.9978	2.1463	.9142	1.159	2.2066
.7868	.8996	2.0796	.8512	1.000	2.1477	.9156	1.164	2.2078
.7882	.9014	2.0812	.8526	1.003	2.1491	.9170	1.169	2.2090
.7896	.9033	2.0827	.8540	1.006	2.1505	.9184	1.174	2.2102
.7910	.9052	2.0643	.8554	1.008	2.1519	.9198	1.180	2.2114
.7924	.9071	2.0858	.8568	1.011	2.1533	.9212	1.185	2.2126
.7938	.9090	2.0874	.8582	1.014	2.1546	.9226	1.190	2.2137
.7952	.9109	2.0889	.8596	1.017	2.1560	.9240	1.196	2.2149
.7966	.9128	2.0904	.8610	1.020	2.1574	.9254	1.202	2.2161
.7980	.9148	2.0920	.8624	1.022	2.1588	.9268	1.208	2.2172
.7994	.9167	2.0935	.8638	1.025	2.1601	.9282	1.213	2.2184
.8008	.9187	2.0950	.8652	1.028	2.1615	.9296	1.220	2.2195
.8022	.9207	2.0965	.8666	1.031	2.1629	.9310	1.226	2.2207
.8036	.9227	2.0981	.8680	1.034	2.1642	.9324	1.232	2.2218
.8050	.9247	2.0996	.8694	1.037	2.1656	.9338	1.239	2.2230
.8064	.9267	2.1011	.8708	1.040	2.1669	.9352	1.246	2.2241
.8078	.9288	2.1026	.8722	1.043	2.1683	.9366	1.252	2.2252
.8092	.9308	2.1041	.8736	1.046	2.1696	.9380	1.260	2.2263
.8106	.9329	2.1056	.8750	1.050	2.1709	.9394	1.267	2.2274
.8120	.9349	2.1071	.8764	1.053	2.1723	.9408	1.274	2.2205
.8134	.9370	2.1086	.8778	1.056	2.1736	.9422	1.282	2.2296
.8148	.9391	2.1101	.8792	1.059	2.1749	.9436	1.290	2.2307
.8162	.9413	2.1116	.8806	1.063	2.1762	.9450	1.298	2.2318
.8176	.9434	2.1131	.8820	1.066	2.1775	.9464	1.307	2.2329
.8190	.9456	2.1146	.8834	1.069	2.1789	.9478	1.315	2.2339
.8204	.9477	2.1160	.8848	1.073	2.1802	.9492	1.324	2.2350
.8218	.9499	2.1175	.8862	1.076	2.1815	.9506	1.334	2.2361
.8232	.9521	2.1190	.8876	1.080	2.1828	.9520	1.343	2.2371
.8246	.9544	2.1205	.8890	1.084	2.1841	.9534	1.353	2.2381
.8260	.9566	2.1219	.8904	1.087	2.1854	.9548	1.364	2.2392
.8274	.9589	2.1234	.8918	1.091	2.1866	.9562	1.375	2.2402
.8288	.9611	2.1248	.8932	1.095	2.1879	.9576	1.386	2.2412
.8302	.9634	2.1263	.8946	1.099	2.1892	.9590	1.397	2.2422
.8316	.9658	2.1277	.8960	1.102	2.1905	.9604	1.410	2.2432
.8330	.9681	2.1292	.8974	1.106	2.1917	.9618	1.422	2.2442
.8344	.9704	2.1306	.8988	1.110	2.1930	.9632	1.436	2.2452
.8358	.9728	2.1321	.9002	1.115	2.1943	.9646	1.450	2.2461
.8372	.9752	2.1335	.9016	1.119	2.1955	.9660	1.464	2.2471
.8386	.9776	2.1349	.9030	1.123	2.1968	.9674	1.480	2.2481
.8400	.9801	2.1364	.9044	1.127	2.1900	.9688	1.496	2.2490

TABLE 6.2 (continued)
Dimensionless Variables Associated with Stage Variable, w, for Circular Sections

Y′	c′	w′	Y′	c′	w′	Y′	c′	w′
.9702	1.513	2.2499	.9772	1.617	2.2544	.9842	1.771	2.2586
.9716	1.531	2.2509	.9786	1.643	2.2553	.9856	1.813	2.2593
.9730	1.551	2.2518	.9800	1.671	2.2561	.9870	1.859	2.2601
.9744	1.571	2.2527	.9814	1.701	2.2570	.9884	1.913	2.2609
.9758	1.593	2.2535	.9828	1.735	2.2578	.9898	1.975	2.2616

then the dimensionless celerity is defined by

$$c' = \frac{1}{2}\sqrt{\frac{\beta - \cos\beta\sin\beta}{\sin\beta}} \tag{6.38}$$

and the stage variable w can be defined by

$$w = g\int\frac{dY}{c} = (gD)\times\frac{1}{(gD)^{1/2}}\int\frac{\sin\beta}{2c'}d\beta = \sqrt{gD}\int\frac{\sin\beta}{\sqrt{\beta/\sin\beta - \cos\beta}}d\beta$$

Thus, if a dimensionless stage function w′ is defined by the following equation, then it is possible to develop a table of dimensionless variables in much the same way as was done for trapezoidal channels.

$$w' = \frac{w}{\sqrt{gD}} = \int_{\varepsilon}^{\beta}\frac{\sin\beta}{\sqrt{\beta/\sin\beta - \cos\beta}}d\beta \tag{6.39}$$

Table 6.2 contains values for c′ and w′ as defined above for a circular channel. In this table, the angle β is replaced by the dimensionless depth Y′ = Y/D, through the relationship that 2Y/D = 1 − cos β. The use of Table 6.2 is identical to that of Table 6.1 in solving unsteady flow problems in circular sections.

EXAMPLE PROBLEM 6.14

Water is flowing through a circular channel with a diameter D = 5 m at a rate of $Q_o = 5.5\,m^3/s$ and at a velocity of $V_o = 0.3$ m/s. At its downstream end, the flow rate is suddenly increased at a rate of $dQ/dt = 0.005\,m^3/s^2$. Determine when the depth will equal 4.1 m at the downstream end of this channel. At a position 1000 m upstream from its end, when will the flow rate equal $8.0\,m^3/s$?

Solution

The first step of the solution requires that the uniform depth be determined by dividing the flow rate by the velocity giving an area $A_o = 18.333\,m^2$. The corresponding normal depth and the top width are: $Y_o = 4.410$ m, and $T_o = 3.226$ m. Therefore, $c_o = (gA/T)^{1/2} = 7.467$ m/s. Since the origin of the xt plane is at the downstream end of the channel, the uniform velocity (and flow rate) will be negative, and dx/dt(0,0) = 7.167 m/s. To determine when the depth will equal 4.1 m at the origin, Y′ corresponding to this depth is computed to be .810 m. Entering Table 6.2 with this dimensionless depth produces $c' = 0.932$, and $w' = 0.704$; thus $w = 4.931$, and $V = 4.931 - .3 - 5.113 = -0.482$. The area corresponding to $Y = 4.1$ m is $A = 17.232\,m^2$, and therefore the flow rate is $Q = -8.306\,m^3/s$. The time for this flow rate to occur is $t_1 = (8.306 - 5.5)/0.005 = 561.2$ s. The second part of the problem requires that $Q = VA = (w + V_o - w_o)A$ be solved for the depth Y (or Y′) to get t_1 and then determine dx/dt at this time and divide this value into 1000 m to determine Δt. The total time is finally determined as the sum of t_1 and Δt. After several tries $Y = 4.072$ m satisfies the continuity equation as follows: $Y' = .8144$, $c' = .939$, $w' = .706$ $w = 4.944$. The area $A = 17.123$ and $V = 4.944 - 5.412 = -0.4685$ fps; giving $Q = 8.02\,m^3/s$. Next, $c = 6.576$ so $dx/dt = 6.107$ m/s so $\Delta t = 163.7$ s, and $t_1 = 500$ s, so the total time is $t = 663.7$ s.

6.9 MAXIMUM FLOW RATES IN NONRECTANGULAR CHANNELS

In a previous section, simple relationships were obtained that defined the limiting flow rate, and the minimum depth that can be obtained in a rectangular channel. In this section, these relationships will be extended to nonrectangular channels based on the methods of characteristics that assume $g(S_o - S_f) = 0$. The flow rate is given by Q = AV and from the equation $c^2 = gA/T$, and the relationship that gives the velocity at the origin, this can be put in the following function of Y and Q equal to the zero form:

$$\begin{aligned} F(Y,Q) &= A(w + V_o - w_o) - Q = 0 \quad \text{or} \\ F(Y,Q) &= Tc^2(w + V_o - w_o) - gQ = 0 \end{aligned} \tag{6.40}$$

Taking the derivative of Equation 6.40 with respect to Y and setting the result to zero to find extreme values gives

$$\frac{\partial F}{\partial Y} = Tc^2\frac{\partial w}{\partial Y} + (w + V_o - w_o)\left\{T\left(2c\frac{\partial c}{\partial Y}\right) + c^2\frac{\partial T}{\partial Y}\right\} = 0 \tag{6.41}$$

From the definition of the stage variable w,

$$\frac{\partial w}{\partial Y} = \left(\frac{gT}{A}\right)^{1/2} = c\left(\frac{T}{A}\right) = \frac{g}{c} \quad \text{and for a trapezoidal channel}$$

$$\frac{\partial T}{\partial Y} = 2m, \quad \text{and} \quad 2c\frac{\partial c}{\partial Y} = g\left(1 - \frac{2mA}{T^2}\right)$$

Substituting these into the above derivative that was set to zero gives

$$gcT + (w + V_o - w_o)\left\{gT\left(1 - \frac{2mA}{T^2}\right) + 2mc^2\right\}$$

Upon dividing this result by gT and simplifying gives

$$\begin{aligned} &c + (w + V_o - w_o)\{1\} = 0 \quad \text{or} \\ &w + c = w_o - V_o \end{aligned} \tag{6.42}$$

Since for the type of problems for which extreme flow is of interest, the positive x-axis will point upstream, Equation 6.42 indicates that the sum of the stage variable w and the celerity c corresponding to the minimum depth, or maximum flow rate will be the sum of the stage variable w_o and the magnitude of the velocity V_o under the initial uniform flow. Thus, to solve the minimum depth, the value of $w_o + |V_o|$ is first computed, and thereafter different values of Y are selected until w + c associated with these, equals this value. Equation 6.42 was derived using a trapezoidal channel. For a circular channel,

$$\frac{\partial w}{\partial Y} = \sqrt{\frac{gT}{A}} = c\left(\frac{T}{A}\right) = \frac{g}{c} \quad \text{the same as for a trapezoidal channel, but}$$

$$\frac{\partial T}{\partial Y} = D\cos\beta\frac{\partial \beta}{\partial Y} = \frac{2D\cos\beta}{T}, \quad \text{and} \quad 2c\frac{\partial c}{\partial Y} = g\left(1 - \frac{2DA\cos\beta}{T^3}\right)$$

which when substituted in the derivative that is set to zero, Equation 6.40, and simplified results in the identical results, or Equation 6.41 also applies for a circular section. We might consider a similar development of extreme flow conditions using dimensionless variables. In other words, replace the variables in Equation 6.40 with their dimensionless equivalents, or for a trapezoidal channel using $Y = bY'/m$, $T = b(1 + 2Y')$, $A = (b^2/m)(Y' + Y'^2)$, $c^2 = (gb/m)c'^2$ and $w = (gb/m)^{1/2}w'$ in which $c'^2 = (Y' + Y'^2)/(1 + 2Y')$, taking the derivative of this equation with respect to Y' and setting the result to zero. The same could be done for a circular section. However, this will be left as an exercise for the reader. A simpler approach is to convert Equation 6.42 to its dimensionless equivalent directly. Defining $V'_o = V_o/(gb/m)^{1/2}$ for a trapezoidal channel and $V'_o = V_o/(gD)^{1/2}$ for a circular section and dividing Equation 6.42 by $(gb/m)^{1/2}$ for trapezoidal channels or $(gD)^{1/2}$ for a circular section result in

$$w' + c' = w'_o - V'_o \tag{6.43}$$

for either a trapezoidal or a circular section. While it is understood that this w and $c = -V_c$ are critical conditions, one may wish to add a c subscript to them to emphasize this fact.

Upon changing Equation 6.40 to its dimensionless equivalent produces the following equation:

$$f(Y', Q') = (1 + 2Y')c'^2\{w' - w'_o + V'_o\} - Q' = Tc'^2V' - Q' = 0 \tag{6.44}$$

or

$$f(Y', Q') = (Y' + Y'^2)\{w' - w'_o + V'_o\} - Q' = A'V' - Q' = 0$$

in which dimensionless $Q' = m^{3/2}Q/(g^{1/2}b^{5/2}) = Q\left\{m^3/(gb^5)\right\}^{1/2} = (1 + 2Y')c'^2(w' + V'_o - w'_o)$ for a trapezoidal channel and $Q' = Q/(g^{1/2}D^{5/2}) = Q/(gD^5)^{1/2} = \sin(\beta)(1 + 2Y')c'^2(w' + V'_o - w'_o) = A'V' = .25(\beta - \cos\beta\sin\beta)(w' + V'_o - w'_o)$ for a circular channel. From Equation 6.43, it can be seen that critical conditions, i.e., the dimensionless depth Y'_c, and the dimensionless velocity $V'_c = -c'$, are determined uniquely by the initial conditions, i.e., Y'_o (which fixes the value of w'_o) and V'_o (or Q'_o). Furthermore, since w_c is uniquely determined by Y_c and $c' = \left\{Y'_c + Y'^2_c)/(1 + 2Y'_c)\right\}^{1/2}$ for a trapezoidal channel, or $c' = 0.5\{\beta/\sin\beta - \cos\beta\}^{1/2}$ where $\beta = \cos^{-1}(1 - 2Y'_c)$ for a circular channel, it is only a function of the dimensionless critical depth Y'_c, which means that for any value of $w'_o - V'_o$, the corresponding value of Y'_c can be determined. This value of Y'_c produces a sum $w' + c'$ that equals the value $w'_o - V'_o$. After this value for Y'_c has been determined it can in turn be used

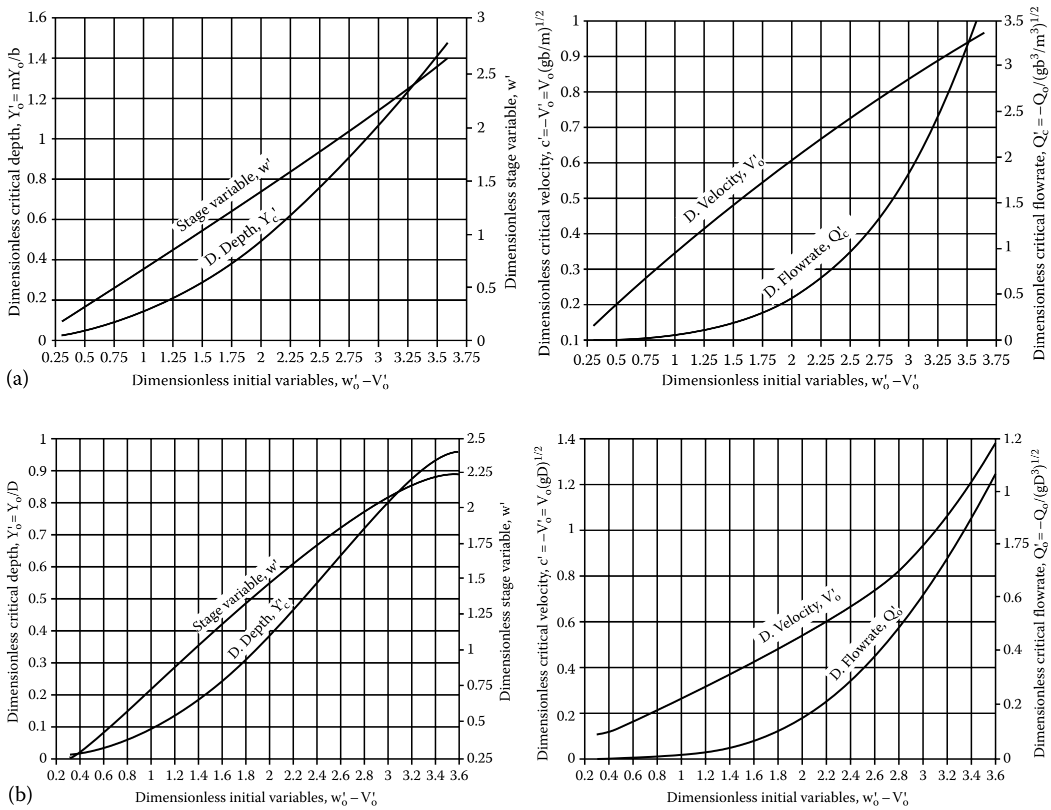


FIGURE 6.3 (a) Dimensionless critical depth and stage variable related to initial $w_o' - V_o'$ for a trapezoidal channel. (b) Dimensionless critical depth and stage variable related to initial $w_o' - V_o'$ for a circular channel.

to compute the critical dimensionless velocity $V_c' = -c'$ from the above equation for a trapezoidal, or a circular channel. Once V_c' is computed, the dimensionless flow rate can be obtained from $Q_c' = -|Q_{max}| = A_c'V_c'$ in which $A_c' = Y_c' + Y_c'^2$ for a trapezoidal, or $A_c' = 0.25(\beta - \cos\beta\sin\beta)$ for a circular channel. The critical value Y_c' has been determined for a series of values for $w_o' - V_o'$ and plotted on the first graph in Figure 6.3a using the left ordinate. Using the right ordinate on this figure, the corresponding value for the critical dimensionless stage variable w' has been plotted. The second graph on Figure 6.3a gives the relationship of the dimensionless critical velocity $c' = -V_c'$ on the left ordinate and the dimensionless critical flow rate Q_c' on the right ordinate, for trapezoidal channels. Figure 6.3b contains the same graphs except for circular, rather than trapezoidal channels.

An alternative for displaying these dimensionless relationships is given in Figure 6.4a and b for trapezoidal and circular channels, respectively. The difference between these plots and those in Figure 6.3a and b is that the dimensionless initial (normal) depth Y_o' is plotted on the abscissa rather than $w_o' - V_o'$, and different curves apply for different magnitude values of the initial dimensionless velocity V_o'. On these latter graphs, the effects of V_o' can be seen by the spread between the curves. When using Figure 6.3a, or b, to solve problems, w_o' must first be determined, whereas this value is not required when using Figure 6.4a, or b. The use of Figures 6.3 and 6.4 limit the accuracy of the results to about two digits. To achieve better precision, Tables 6.1 and 6.2 are required to solve extreme conditions problems. An alternative to get greater accuracy is to implement solutions to these problems in a computer model.

Program MAXFLOW is designed to obtain the maximum flow rate (or minimum if one considers that Q is negative) and the minimum depth for trapezoidal and circular channels.

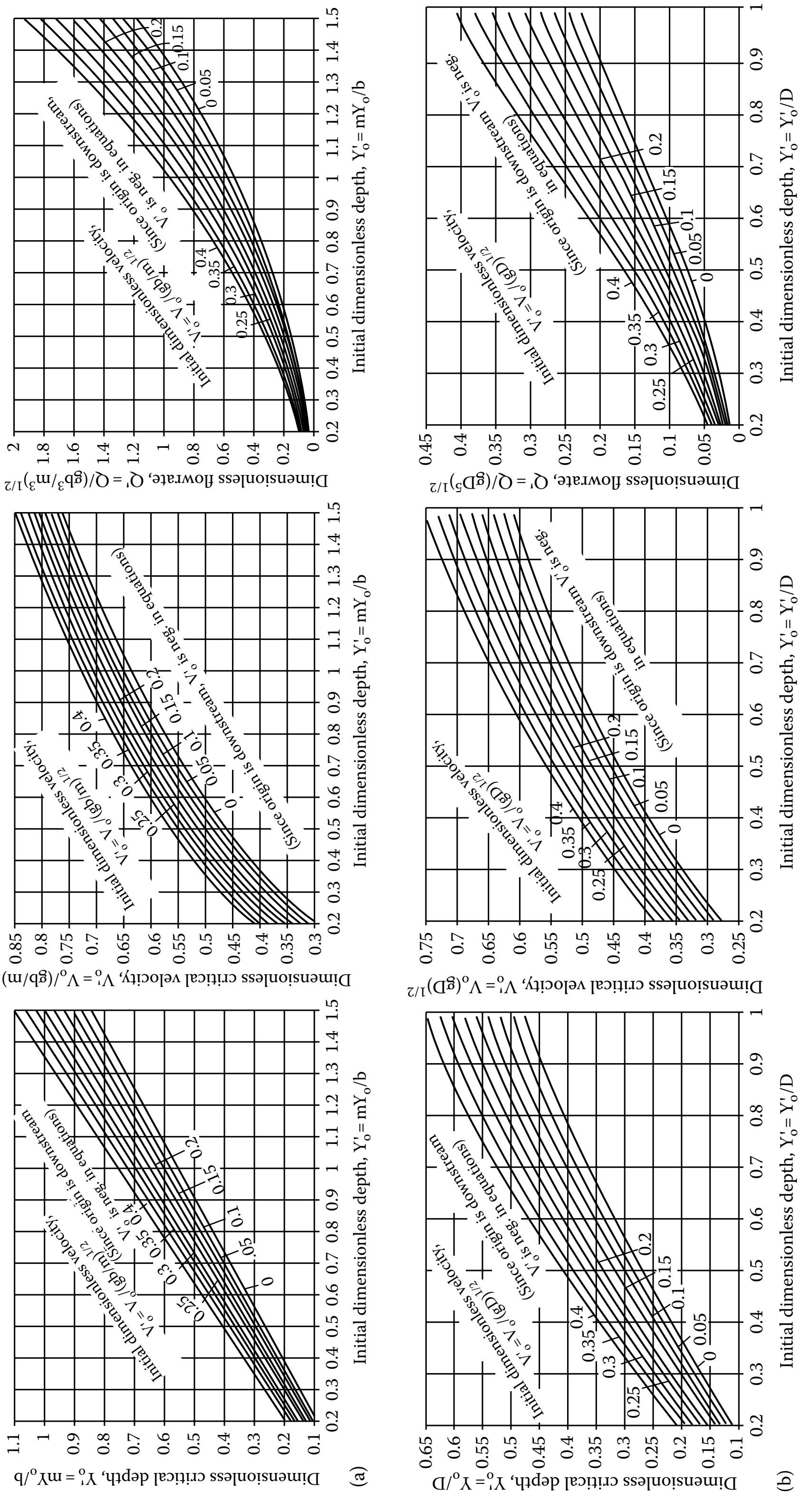


FIGURE 6.4 (a) Dimensionless critical values (Y'_c, V'_c, and Q'_c) related to initial dimensionless depth, Y'_o and velocity, V'_o for a trapezoidal channel. (b) Dimensionless critical values (Y'_c, V'_c, and Q'_c) related to initial dimensionless depth, Y'_o and velocity, V'_o for a circular channel.

Program MAXFLOW.FOR

```
      REAL F[ALLOCATABLE](:),W[ALLOCATABLE](:),X(2)
      WRITE(*,*)' Give:(2=trap,1=cir),IOUT,Yo,','Vo(neg),g,
     &(b & m) or D'
      READ(*,*) ITY,IOUT,YO,VO,G,(X(I),I=1,ITY)
      IF(ITY.EQ.1) THEN
      D=X(1)
      YPO=YO/D
      YP1=.0066
      BET1=ACOS(1.-2.*YP1)
      DY=.0014
      YPE=YO/D
      RC=FC(ITY,BET1,CP)
      W1=BET1*RC
      VO=VO/SQRT(G*D)
      CONVY=D
      ELSE
      B=X(1)
      FM=X(2)
      YPO=YO*FM/B
      DY=.002
      YPE=FM*YO/B
      YP1=.01
      RC=FC(ITY,YP1,CP)
C     W1=.138
      W1=YP1*RC
      VO=VO/SQRT(G*B/FM)
      CONVY=B/FM
      ENDIF
      DY1=DY/10.
      DYH=DY1/2.
      YP11=YP1
      N=(YPE-YP1)/DY+2.
      ALLOCATE(F(N+5),W(N+5))
      W(1)=W1
      DO 10 I=2,N
      SUM=0.
      IF(ITY.EQ.1) THEN
      YP=YP1+DY
      BET2=ACOS(1.-2.*YP)
      DBET=(BET2-BET1)/10.
      DBE2=.5*DBET
      DO 4 J=1,10
      RC1=FC(ITY,BET1+DBET*FLOAT(J),CP)
      SUM=SUM+DBE2*(RC+RC1)
4     RC=RC1
      BET1=BET2
      ELSE
      DO 5 J=1,10
      YP=YP1+DY1*FLOAT(J)
      RC1=FC(ITY,YP,CP)
```

```
      SUM=SUM+DYH*(RC+RC1)
5     RC=RC1
      ENDIF
      YP1=YP
      W(I)=W(I-1)+SUM
10    F(I)=W(I)+CP
      IYP=ABS((YPO-YP11)/DY)+1.
      K=IYP
      WO=W(IYP)
      SUM=WO-VO
15    IF(SUM.GT.F(K)) GO TO 16
      K=K-1
      IF(K.GT.0) GO TO 15
16    FAC=(SUM-F(K))/(F(K+1)-F(K))
      YPM=YP11+DY*(FLOAT(K)+FAC)
      IF(ITY.EQ.1) THEN
      BET1=ACOS(1.-2.*YPM)
      RC1=FC(ITY,BET1,CP)
      ELSE
      RC1=FC(ITY,YPM,CP)
      ENDIF
      CP2=CP*CP
      WP=W(K)+FAC*(W(K+1)-W(K))
      VP=WP+VO-WO
      IF(ITY.EQ.1) THEN
      BETA=ACOS(1.-2.*YPM)
      QP=SIN(BETA)*CP2*VP
      Q=QP*SQRT(G*D**5)
      V=SQRT(G*D)*VP
      ELSE
      QP=(1.+2.*YPM)*CP2*VP
      Q=QP*SQRT(G*B**5/FM**3)
      V=SQRT(G*B/FM)*VP
      ENDIF
      WRITE(IOUT,100) YPM*CONVY,V,Q,YPM,VP,QP
100   FORMAT(' Min. Depth=',F9.3,' Velocity=',F9.3,' Max.
     &Flowrate=',F10.2,/,2F9.5,F10.5)
      DEALLOCATE(F,W)
      END
      FUNCTION FC(ITY,YP,CP)
      IF(ITY.EQ.1) THEN
      SINB=SIN(YP)
      CP=SQRT(YP/SINB-COS(YP))
      FC=SINB/CP
      CP=.5*CP
      ELSE
      FC=SQRT((1.+2.*YP)/(YP+YP*YP))
      CP=1./FC
      ENDIF
      RETURN
      END
```

Program MAXFLOW.C

```
#include <math.h>
#include <stdlib.h>
#include <stdio.h>
float fc(int ity,float yp,float *cp){float cosb,beta,sinb,tfc;
 if(ity==1){sinb=sin(yp); *cp=sqrt(yp/sinb-cos(yp));
 tfc=sinb/(*cp); *cp*=.5;} else {tfc=sqrt((1.+2.*yp)/(yp+yp*yp));
   *cp=1./tfc;}
 return tfc;} // End fc
void main(void){int k,i,j,n,ity,iyp;
   float yp,rc1,bet1,bet2,dbet,dbe2,vp,v;
 float x[2],b,m,d,ypo,yp1,dy,ype,rc,w1,yo,vo,convy,dy1,dyh,yp11,sum,\
   wo,ypm,fac,qp,q,beta,cp2,wp,g,*cp,*f,*w;
 printf("Give: (2=trap,1=cir),Yo,Vo(neg),g,(b & m) or D\n");
 scanf("%d %f %f %f",&ity,&yo,&vo,&g);
   for(i=0;i<ity;i++) scanf("%f",&x[i]);
 if(ity==1){d=x[0];ypo=yo/d;yp1=.0066;bet1=acos(1.-2.*yp1);
   dy=.0014;ype=yo/d;rc=fc(ity,bet1,cp);
  w1=bet1*rc;vo=vo/sqrt(g*d); convy=d;} else {
 b=x[0];m=x[1];ypo=yo*m/b;dy=.002;ype=m*yo/b;
   yp1=.01;rc=fc(ity,yp1,cp);
  w1=yp1*rc;vo=vo/sqrt(g*b/m);convy=b/m;}
 dy1=.1*dy; dyh=.5*dy1; yp11=yp1;n=(ype-dy1)/dy+2.;
 f=(float *)calloc(n+5,sizeof(float));
   w=(float *)calloc(n+5,sizeof(float));
 w[0]=w1;for(i=1;i<n;i++){sum=0.;if(ity==1){yp=yp1+dy;
   bet2=acos(1.-2.*yp);
  dbet=(bet2-bet1)/10.;dbe2=.5*dbet;
    for(j=1;j<11;j++){rc1=fc(ity,bet1+dbet*((float)j),cp);
  sum+=dbe2*(rc+rc1);rc=rc1;}bet1=bet2;} else {for(j=1;j<11;j++){
   yp=yp1+dy1*(float)j;rc1=fc(ity,yp,cp); sum+=dyh*(rc+rc1);
     rc=rc1;}}
 yp1=yp; w[i]=w[i-1]+sum;f[i]=w[i]+(*cp);}
 iyp=fabs((ypo-yp11)/dy); k=iyp; wo=w[iyp];sum=wo-vo;
   while((sum<=f[k])&&(k>-1))k-;
 fac=(sum-f[k])/(f[k+1]-f[k]); ypm=yp11+dy*((float)(k+1)+fac);
 if(ity==1){bet1=acos(1.-2.*ypm);
   rc1=fc(ity,bet1,cp);} else rc1=fc(ity,ypm,cp);
 cp2=(*cp)*(*cp);wp=w[k]+fac*(w[k+1]-w[k]);vp=wp+vo-wo;
 if(ity==1){beta=acos(1.-2.*ypm);qp=sin(beta)*cp2*vp;
   q=qp*sqrt(g*pow(d,5.));v=sqrt(g*d)*vp;}
 else {qp=(1.+2.*ypm)*cp2*vp;q=qp*sqrt(g*pow(b,5.)/pow(m,3.));
   v=sqrt(g*b/m)*vp;}
 printf("Min. Y= %10.3f Vel.= %9.3f Max. Q= %12.2f \n %8.5f\
   %8.5f %8.5f\n",ypm*convy,v,q,ypm,vp,qp);
 free(f);free(w);}
```

EXAMPLE PROBLEM 6.15

A trapezoidal channel with a bottom width b = 10 ft, and a side slope of m = 1.0 initially has a uniform depth of Y_o = 4.2 ft, and a velocity V_o = −3.517 fps. (The velocity is negative because the origin for the unsteady solution is at the downstream end of the channel.) The gate at the downstream end of this channel is suddenly opened allowing the channel to discharge freely into a reservoir whose water surface is well below the water in the channel. Determine the flow rate into the reservoir.

Solution
First, compute $Y'_o = (1)(4.2)/10 = 0.42$ and entering Table 6.1 with this value gives $w'_o = 1.304$, and $w_o = (gb/m)^{1/2} w'_o = 17.944(1.304) = 23.399$ fps. The dimensionless initial velocity is $V'_o = V_o/(gb/m)^{1/2} = -3.517/17.944 = -0.196$, or $w'_o - V'_o = 1.304 + 0.196 = 1.5$. Thus, we seek a dimensionless depth Y′ so that the corresponding w′ + c′ = 1.5. Trying several values of Y′ in Table 6.1 results in the following:

Y′	c′	w′	c′+w′
.270	.472	1.016	1.488
.272	.473	1.020	1.493
.274	.475	1.025	1.500
Solution (Y'_{min} = .274)			

Thus, the minimum depth when the maximum flow rate occurs is $Y_{min} = (b/m)Y'_{min} = 2.74$ ft, the corresponding velocity is $V_{max} = w + V_o - w_o = 17.944(1.025) - 3.517 - 23.399 = -8.523$ fps and the maximum flow rate is $Q = VA = -8.523(10 + 2.74)2.74 = -8.523(34.908) = 297.5$ cfs. Note that if 1.5 is entered on the abscissa on the first graph of Figure 6.3a Y′ = .27 can be read. As a useful learning exercise, you should program Equation 6.42 (or Equation 6.43) into an equation solver such as that available in MathCad, TK-Solver or MATLAB and by iterative providing values of w by using Table 6.1, find the values that result in this equation equaling zero.

6.9.1 Maximum Outflow ΔQ That Can Be Taken at an Intermediate Position

As with a rectangular channel, the maximum outflow that can occur from a nonrectangular channel occurs when the condition immediately upstream from the point of the outflow becomes critical. In other words, the critical depth and the critical velocity can be obtained by equating the upstream velocity to the celerity, or solve the implicit equation,

$$V_u = c \quad \text{or} \quad F = (V_o + w_o - w) - c = 0 \quad \text{or} \quad F = (V_o + w_o - w) - \sqrt{gA/T} = 0 \quad \text{or}$$

$$w + c = w_o + V_o \quad \text{or} \quad w' + c' = w'_o + V'_o \quad \text{(The same as Equation 6.43 when noting } V_o \text{ is now positive)}$$

Thereafter, the outflow can be obtained from,

$$\Delta Q = 2A(w_o - w)$$

EXAMPLE PROBLEM 6.15A
Initially a trapezoidal channel with b = 10 ft and m = 1.5 contains a uniform flow at a depth of $Y_o = 5$ ft, and a flow rate $Q_o = 400$ cfs. What is the maximum point outflow that can be taken from an intermediate position along this channel?

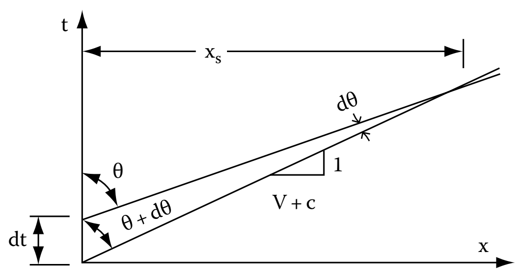
Solution
The equations given above are solved with the following results: $V_o = 4.571$ ft/s, $c_o = 10.616$ ft/s, $w_o =$ 26.339 ft/s and from equating V_d to c, the following are obtained: Y = 3.565 ft, c = 9.226 fps, A = 54.71 ft², $Y'_o = .75$, $w'_o = 1.813$, $w_o = 26.561$, Y′ = .5348, w′ = 1.495, w = 21.906, ΔQ = 509.33 cfs, $V_u = 9.226$ fps, $Q_u = 504.76$ cfs, $V_d = -.084$ fps, $Q_d = -4.57$ cfs, with w = 21.684 fps.

6.10 POSITIVE WAVES

The illustrative problems dealt with up to this point in this chapter have all involved negative or dispersive waves, i.e., the C^+ characteristics spread apart because the inverse slope of each succeeding characteristic is smaller than the previous one. In this section, problems will be discussed in which depths increase at the origin as a function of time with the effect that positive waves are produced. The effect of this increasing depth is that the succeeding C^+ characteristics have larger inverse slopes than their predecessors, and therefore as shown on the sketch below, will intersect with its predecessor that passes through the t-axis a small time dt earlier. The shortest distance from the origin where the intersection of adjacent characteristics occurs fixes a point that will trap all subsequent waves issuing from the increasing depth at the origin, and as these waves are trapped, they combine to form a moving surge in the channel. Such surges are called by such names as **hydraulic bores** when they occur in a river, or **seiches** when they occur in an estuary.

It is important to be able to predict the position at which the surge will occur. In what follows, an equation will be developed that allows both the position and the time of the occurrence to be determined for a positive wave in a rectangular channel. To obtain this equation, consider a C^+ characteristic that starts at a dt time later from the origin than its predecessor characteristic in the xt plane, as shown above. The angle between these two characteristics, $d\theta$ is given by

$$d\theta = \frac{\sin(\theta)dt}{x_s/\sin(\theta)} = \frac{\sin^2(\theta)}{x_s}$$



The tan θ equals the inverse slope of the first characteristic, or tan $\theta = V + c$, which if the differential is taken of both sides produces $\sec^2\theta \, d\theta = d(V + c)$, or $d\theta$ also is given by $d\theta = d(V + c) \cos^2\theta$. Therefore,

$$\frac{\sin^2(\theta)dt}{x_s} = d(V+c)\cos^2(\theta)$$

Or solving position x_s where the two characteristics intersect gives

$$x_s = \frac{\tan^2(\theta)}{d(V+c)/dt} = \frac{(v+c)^2}{d(V+c)/dt} = \frac{(3c(0,t)+V_o-2c_o)^2}{3(dc(0,t)/dt)} \qquad (6.45)$$

If a change in velocity, rather than a change in depth, with respect to time is specified, then replace dc/dt with dV/dt.

The position where the surge will take place is the minimum distance x_s obtained as the intersection of all possible pairs of adjacent C^+ characteristics. This minimum distance is generally at the intersection with the initial characteristic through the origin. Reasons for this can be seen by examining Equation 6.45. Note that for a positive wave, the numerator of Equation 6.5 increases with time, and since this is squared, the distance x_s will get larger in time if the rate of change of the celerity c with time is constant. Only if the rate of change of c increases in time sufficiently to overcome the squared increasing effect in the numerator of Equation 6.45 will the minimum x_s occur at the intersection of the subsequent pair of adjacent characteristics. If dc/dt does increase in time, then it is necessary to create a table of values for x_s and select the smallest value. Otherwise, x_s is computed using $(V_o + c_o)^2$ in the numerator of Equation 6.45, and dc/dt is evaluated at t = 0.

EXAMPLE PROBLEM 6.16

A rectangular channel discharges into a reservoir. At time zero and earlier, the flow in the channel has been uniform with $Y_o = 3$ m, and the velocity of flow $V_o = 1.5$ m/s. Suddenly, the water depth in the reservoir begins to rise causing the water at the end of the channel to increase at a constant rate dY/dt = 0.017 m/s. Determine the position where, and at what time a surge would be expected to form in the channel.

Solution

Since, the rate of rise in the water surface is constant, the surge will form at the front of the undisturbed zone of quiet, and therefore the distance from the end of the channel where it will first form is given by $x_s = (V_o + c_o)^2/[3dc/dt(0, 0)]$. The speed $c_o = 5.425$ m/s, and dc/dt can be obtained by differentiation of $c^2 = gY$, giving $dc/dt = (0.5g/c_o)(dY/dt) = 0.01537$, and therefore $x_s = (-1.5 + 5.425)^2/(3 \times 0.01537) = 334$ m. The time of this surge formation is 334/3.925 = 85 s.

When the flow rate q(0, t) is given as a function of time at the origin, rather than the depth, it is necessary to express the denominator in Equation 6.45, i.e., dc/dt, in terms of this known dq/dt. This relationship can be obtained from $q = VY = (c^2/g)V$, or $gq = Vc^2$ by taking the derivative with respect to time, or

$$g\frac{dq}{dt} = V\left(2c\frac{dc}{dt}\right) + c^2\frac{dV}{dt}$$

but $V = 2c + V_o - 2c_o$, so dV/dt = 2dc/dt and

$$g\frac{dq}{dt} = 2c(V + c)\frac{dc}{dt}$$

Assuming that the surge will form at the front of the zone of quiet, we can replace c and V by c_o and V_o, and then Equation 6.45 can be written as

$$x_s = \frac{2c_o(V_o + c_o)^3}{3g(dq/dt)} \tag{6.45a}$$

To obtain the equation that gives the position x_s where the surge first forms when dealing with a nonrectangular channel, it is probably easier to start with the equation that uses the slope of the C^+ characteristics, or

$$x_s = \frac{\left(\dfrac{dx}{dt}\right)^2}{\dfrac{d}{dt}\left(\dfrac{dx}{dt}\right)}$$

Noting that, $dx/dt = w + c + V_o - w_o$ and that, unless unusual situations occur, the surge will first form at the leading edge of the wave so that subzero will be used on all variables, the numerator can be evaluated by squaring the inverse slope of the initial C^+ characteristic or $(V_o + c_o)^2$, and the denominator by noting that the derivative of dx/dt with respect to time equals (dw/dY + dc/dY) (dY/dt). As noted earlier, dw/dy = g/c, and for a nonrectangular channel dc/dY can be evaluated by differentiating $c^2/g = A/T$ that gives

$$\frac{dc}{dY} = \frac{g}{2c}\left[1 - \frac{AdT}{T^2 dY}\right]$$

and the denominator becomes

$$\frac{d}{dt}\left(\frac{dx}{dt}\right) = \frac{g}{c}\left[1 + \frac{1}{2}\left(1 - \frac{A}{T^2}\frac{dT}{dY}\right)\right]\frac{dY}{dt} = \frac{g}{2c}\left[3 - \frac{A}{T^2}\frac{dT}{dY}\right]\frac{dY}{dt}$$

in which dT/dY will need to be evaluated for the particular channel geometry. For a trapezoidal channel dT/dY = 2m, and for a circular channel dT/dY will need to be evaluated using the chain rule, or dT/dY = (dT/dβ)(dβ/dY). From cosβ = 1 – 2Y/D, dβ/dY = 2/(D sin β) and from T = D sin β, dT/dβ = D cos β, and therefore dT/dY = 2/tan β. Substitution of these quantities into the above equation that gives the position of the surge, gives the following equation for a nonrectangular channel (note that dY/dt = (c/g)dV/dt):

$$x_s = \frac{2c(V+c)^2}{g\left[3 - \frac{A}{T^2}\frac{dT}{dY}\right]\left(\frac{dY}{dt}\right)} = \frac{2(V+c)^2}{\left[3 - \frac{A}{T^2}\frac{dT}{dY}\right]\left(\frac{dV}{dt}\right)} = \frac{2(V+c)^2\left\{c\left[3T - (A/T)(dT/dY)\right] + 2V(V+c)\right\}}{\left[3 - \frac{A}{T^2}\frac{dT}{dY}\right]\left(\frac{dQ}{dt}\right)} \tag{6.45b}$$

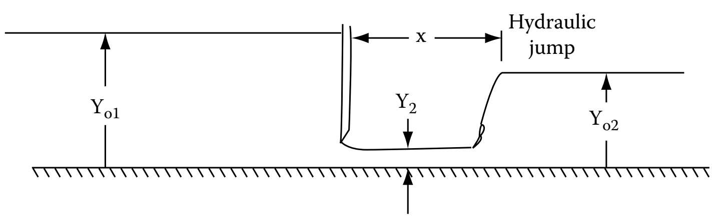
(It is left as an exercise for you to obtain the last expression that gives x_s from a specified dQ/dt.)

It might seem logical to move with the surge once it has formed and write the continuity and the momentum equations from this viewpoint as was done earlier. Doing this results in the equations $(v - V_1)Y_1 = (v - V_o)Y_o$ and $(1/2)(Y_1/Y_o)(Y_1/Y_o + 1) = (V_o + v)/(gY_o)$. There are three unknowns, V_1, Y_1, and v. A third equation can be obtained from the unsteady flow, namely, $t_1 + x/dx/dt = t$, in which t_1 and dx/dt will be evaluated depending upon what is specified at the origin. For example, if the change in flow rate dq/dt is given at x = 0, then this third equation is $(q_o - Y_1V_1)/(dq/dt) + x/(3(gY_1)^{1/2} + V_o - 2c_o) = t$. With v known from the solution of these equations, the increase position x for the surge can be computed by integration of $\int vdt$. However, the problem is that these equations will always produce a solution that gives $Y_1 = Y_o$ and $V_1 = V_o$, which implies that the height of the surge remains zero. The reason for this can be understood by noting that the flow upstream of a hydraulic jump must be supercritical, which means that $v - V_o$ (with V_o negative so that the difference actually is the sum of the magnitudes of the uniform velocity and the velocity of the surge) must be larger than c_o. But the movement of the effect of the increasing depth propagates only with a velocity of $V_o + c_o$. The magnitude of v at the time of the initial formation of the surge is $v = c_o + V_o$, and therefore at this time, the observer moving with the surge sees an incoming velocity equal to c_c, or a Froude number of unity. For him to see a velocity greater than c_o, he would have to move faster than the wave thus leaving it behind. Therefore, we can conclude that the method of characteristics does not allow a solution of the surge after it has formed, but is only useful in predicting when and where the surge may first form. In other words, this theory would always assume a surge of zero height.

6.11 CONTROL STRUCTURES

A control structure whose setting is changed typically creates a positive wave to the flow on one side of it and a negative wave on the other side. Such an example is a gate in a channel. As the gate is raised increasing the discharge past it, this increasing flow rate results in a negative or a dispersive wave upstream from it and a positive wave downstream from it. If the gate is closed further in decreasing the flow past it, a positive wave will result upstream, and if the rate of closure is rapid enough, a surge will form. Downstream from the closing gate, a dispersive wave will generally result if the amount of flow rate decrease is relatively small. If the gate is dropped instantly and the amount of flow rate decrease is quite large, then a surge will instantly be caused to form that will move in the downstream direction. Since control structures, such as gates are common, and the upstream and downstream unsteady flows are linked together by the continuity and energy equations at the gate, it is worth exploring how both the upstream and the downstream flows change due to gates whose settings are changed in time. At first let us make the following assumptions: (1) that the simplified theory associated with the method of characteristics in which $S_o - S_f$ can be ignored are valid, and (2) that the rate of gate change is small so that surges are not formed immediately, as when a gates setting is instantly changed. (Later, we will consider instant gate changes.)

To examine such situations, assume that a uniform flow exists for a long distance upstream from a gate, such as shown in the sketch below. In making this assumption, we recognize that actually, an M_1 GVF-profile will exist upstream from the gate, but for flat channels, the depth upstream will change only modestly over long distances. Furthermore, assume that the depth between the gate and the hydraulic jump downstream therefrom is constant, i.e., an M_3 GVF-profile does not exist downstream of the gate. Finally, assume that the flow is uniform downstream from the hydraulic jump for a long distance.



Consider first, the case in which the gate opening increases slowly with time. As the flow rate past the gate increases with time due to this enlargement in the opening, the velocity will increase in the channel upstream from the gate and the depth will decrease. This upstream time-dependent problem can be handled by the methods described earlier in this chapter using the characteristics in the xt plane with the x axis positive upstream with its origin at the gate. Downstream from the gate, the increasing flow rate will change the stationary hydraulic jump into a moving surge. An approximate analysis of this moving surge is possible by using the continuity and momentum function equations based upon the viewpoint of an observer moving with the speed of the surge v, as was done in Chapter 3. This observer will not actually see a steady-state flow since the depths both upstream and downstream from the hydraulic jump will be increasing. However, for a first approximation of situations in which changes are small in time, the use of the steady-state conditions will suffice. The equations available are

1. The characteristic equations, and the boundary condition equations derived therefrom for the upstream flow.
2. The energy equation applied immediately upstream to immediately downstream from the gate from the viewpoint of a stationary observer.
3. The continuity equation across the moving hydraulic jump from the view point of an observer moving with the same velocity as the hydraulic jump.
4. The momentum equation across the moving hydraulic jump from the view point of a moving observer.

5. The characteristic equations applied to the flow downstream of the hydraulic jump. For these equations, the positive x axis is in the direction of the flow. These five equations allow for the following variables to be typically solved for (1) the upstream depth as it changes in time (with the associated velocity and flow rate), (2) the depth immediately downstream from the gate as it changes in time, (3) the time-dependent velocity downstream from the moving hydraulic jump, (4) the time-dependent depth downstream from the hydraulic jump, and (5) the velocity of this hydraulic jump movement that will also change in time. The details will vary depending upon the problem, and what is known.

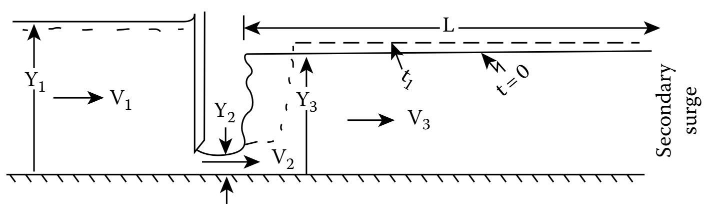
EXAMPLE PROBLEM 6.17

A uniform flow with a velocity of 1.0 fps and a depth of 5.0 ft has existed upstream from a gate for a long time in a rectangular channel. Suddenly, the discharge past the gate per unit width of channel is increased at a constant rate of 1 cfs/min, e.g., $dq/dt = (1/60)\ (ft/s)^2$. Determine the following: (1) the depth and the velocity immediately upstream from the gate after 100 s, (2) the depth and the velocity immediately downstream from the gate at 30 s, (3) the depth and the velocity immediately downstream from the moving hydraulic jump, (4) the velocity of the jump's movement, and (5) the position and the time when a secondary positive wave is likely to form.

Solution

Upstream from the gate the wave is dispersive, and the methods used previously can be used to find the depth and the velocity upstream from the gate at t = 100 s. At time zero, the depth immediately downstream from the gate can be obtained by setting $E_1 = E_2$ giving $E_1 = 5.0155 = Y_2 + q^2/(2gY_2^2)$. Synthetic division can be used in solving Y_2 as follows:

1	−5.0155	0	0.3882 | 5
	5.	−.0776	−0.3882 |
1	−0.0155	−0.07764	0



$Y_2 = (0.0155 + \{(.0155)^2 + 4(0.07764)\}^{1/2})/2 = 0.2865$ ft and the velocity $V_2 = q/Y_2 = 17.452$ fps. Now that the depth is known immediately downstream from the gate, the hydraulic jump equation can be used to solve the depth conjugate $Y_3 = 2.189$ ft, $V_3 = 2.284$ fps, $c_3 = 8.577$ fps.

At time 30 s, the flow rate q = −5.5 cfs/ft, and solving the equation $q = (c^2/g)*(2c - V_o - 2c_o)$ for c gives c = 12.634 fps, with a corresponding depth $Y_1 = 4.957$ ft. Based on this depth, the depth and the velocity downstream from the gate are, $Y_2 = 0.318$ ft, and $V_2 = 17.321$ fps, respectively. Next, to find the depth Y_3, the velocity V_3 downstream of the moving hydraulic jump, and the velocity of this surges movement, we need to solve the continuity equation, the momentum equation based on what a moving observer sees, and the unsteady equation for the downstream flow simultaneously. These three equations are

$$F_1 = (V_2 - v)Y_2 - (V_3 - v)Y_3 = 0 \quad \text{(continuity)} \tag{1}$$

$$F_2 = \frac{1}{2}\left(\frac{Y_3}{Y_2}\right)\left(\frac{Y_3}{Y_2}+1\right) - \frac{(V_2 - v)^2}{(gY_2)} = 0 \quad \text{(momentum)} \tag{2}$$

$$F_3 = 2(gY_3)^{1/2} + V_{o3} - 2c_{o3} - V_3 = 0 \quad \text{(unsteady Char. Equation)} \tag{3}$$

Using the Newton method to solve these three equations gives

$$Y_3 = 2.260 \text{ ft}, \quad V_3 = 2.552 \text{ fps}, \quad \text{and} \quad v = 0.137 \text{ fps}.$$

The position where a secondary surge might be expected to occur in the downstream flow due to the rising water surface downstream from the moving hydraulic jump can be predicted from

$$X_s = \frac{(V_o + c_o)^2}{1.5\{g/Y\}^{1/2}(dY/dt)} = \frac{(2.284 + 8.577)^2}{1.5\{g/2.189\}^{1/2}(2.260 - 2.189)/30} = 8.664\,\text{ft}$$

EXAMPLE PROBLEM 6.18

Develop a computer program to carry out the computations in Example Problem 6.17 for a series of equally spaced time increments, and run this program to solve the depths and velocities upstream and downstream from the gate over a period of 5 min in 30 s increments. The flow conditions at time zero are as described in Example Problem 6.14, and the rate of discharge past the gate is increased at a rate dq/dt = 0.01667 cfs/ft/s.

Solution

The listing of computer programs in both FORTRAN and PASCAL are given below to solve this problem. Thereafter, the output table from running either of these programs using the following input data:

```
5 −1 −.016667 30. 11 32.2 3
```

gives the solution in the table provided below the program listings.

FORTRAN listing to solve unsteady flow both upstream and downstream from gate, UNSTGT

```
      REAL F(3),D(3,3),X(3)
      INTEGER*2 INDX(3)
      COMMON V2,Y2,V3,Y3,V,G,VOC3,VOC,Q,C
      EQUIVALENCE (X(1),V3),(X(2),Y3),(X(3),V)
      WRITE(6,*)' Give:Yo,Vo,dq/dt,Dt,N,g,IOUT'
      READ(5,*)YO,VO,DQT,DT,N,G,IOUT
      C=SQRT(G*YO)
      QO=YO*VO
      V=0.
      VOC=VO-2.*C
      WRITE(IOUT,100)
      DO 40 K=1,N
      T=DT*FLOAT(K-1)
      Q=QO+DQT*T
  2   FF=FUN(4)
      C=C-.005
      DIF=.005*FF/(FF-FUN(4))
      C=C-DIF+.005
      IF(ABS(DIF).GT.1.E-5) GO TO 2
      Y1=C*C/G
      V1=2.*C+VOC
      B=V1*V1/(2.*G)
      Y2=.5*(B+SQRT(B*B+4.*Y1*B))
      V2=ABS(Q)/Y2
      IF(K.EQ.1) THEN
      Y3=.5*Y2*(SQRT(1.+8.*Q*Q/(G*Y2**3))-1.)
      V3=ABS(QO)/Y3
```

```
      VOC3=V3-2.*SQRT(G*Y3)
      ENDIF
      NCT=0
 10   DO 20 I=1,3
      F(I)=FUN(I)
      DO 20 J=1,3
      X(J)=X(J)-.01
      D(I,J)=100.*(F(I)-FUN(I))
 20   X(J)=X(J)+.01
      CALL SOLVEQ(3,1,3,D,F,1,DD,INDX)
      NCT=NCT+1
      DIF=0.
      DO 30 I=1,3
      DIF=DIF+ABS(F(I))
 30   X(I)=X(I)-F(I)
      IF(NCT.LT.20 .AND. DIF.GT.1.E-6)GO TO 10
      IF(NCT.EQ.20)WRITE(6,*)' FAILED TO CONV. K=',K
 40   WRITE(IOUT,110) IFIX(T),Q,Y1,V1,Y2,V2,Y3,V3,V
100   FORMAT(' Time q Upstream gate Downstr. gate Beyond Hyd
      *. J. Vel.H.J.',/,' sec cfs ',3(' Y(ft) V(fps) '),' v(fps)
      *',/,1X,72('-'))
110   FORMAT(I5,F9.3,7F8.4)
      STOP
      END
      FUNCTION FUN(I)
      COMMON V2,Y2,V3,Y3,V,G,VOC3,VOC,Q,C
      GO TO (1,2,3,4), I
  1   FUN=(V2-V)*Y2-(V3-V)*Y3
      RETURN
  2   FUN=.5*Y3/Y2*(Y3/Y2+1.)-(V2-V)**2/(G*Y2)
      RETURN
  3   FUN=2.*SQRT(G*Y3)+VOC3-V3
      RETURN
  4   FUN=C*C*(2.*C+VOC)-G*Q
      RETURN
      END
```

PASCAL listing to solve unsteady flow both upstream and downstream from gate

```
Program Unsteady_gate;
Var F,X:array[1..3] of real;D:array[1..3,1..3] of real;
    indx:array[1..3] of integer;
    V1,Y1,B,V2,Y2,YO,VO,qo,q,DQT,DT,T,g,VOC,VOC3,c,DIF,FF:real;
    K,I,J,N,NCT:integer; OUT:Text;Filen:string[12];
Function FUN(II:integer):real;Begin
 Case II of
  1:FUN:=(V2-X[3])*Y2-(X[1]-X[3])*X[2];
  2:FUN:=X[2]/Y2*(X[2]/Y2+1)/2-sqr(V2-X[3])/(G*Y2);
  3:FUN:=2*sqrt(G*X[2])+VOC3-X[1];
  4:FUN:=c*c*(2*c+VOC)-g*q;
 End;End;
{$I SOLVEQ.PAS}
BEGIN
 Writeln('Give:Yo,Vo,dq/dt,Dt,N,g');
 Readln(Yo,Vo,DQT,Dt,N,g);
 Writeln('Give output filename');Readln(Filen);
 Assign(OUT,Filen);Rewrite(OUT);
```

```
c:=sqrt(g*Yo);qo:=Yo*Vo;X[3]:=0;VOC:=Vo−2*c;
Writeln
(OUT,' Time q Upstream gate Downstr. gate Beyond Hyd. J. Vel.H.J.');
Writeln(OUT,' sec (cfs/ft) Y(ft) V(fps) Y(ft) V(fps) Y(ft)
  V(fps) v(fps)');
Writeln(OUT,'----------------------------------------------');
For K:=1 to N do Begin
  T:=Dt*(K−1); q:=qo+DQT*T;
  Repeat FF:=FUN(4);c:=c−0.005;DIF:=0.005*FF/(FF-FUN(4));
   c:=c−DIF+0.005; until abs(DIF)<1.e−5;
  Y1:=c*c/g; V1:=2*c+VOC;B:=V1*V1/(2*g);
  Y2:=(B+sqrt(B*B+4*Y1*B))/2; V2:=abs(q)/Y2;
  If K=1 then begin
   X[2]:=0.5*Y2*(sqrt(1+8*q*q/(g*Y2*sqr(Y2)))−1);
   X[1]:=abs(qo)/X[2];VOC3:=X[1]−2*sqrt(g*X[2]) end;
  NCT:=0;Repeat
  For I:=1 to 3 do Begin
   F[I]:=FUN(I);
   For J:=1 to 3 do begin X[J]:=X[J]−0.01;
   D[I,J]:=100*(F[I]−FUN(I));X[J]:=X[J]+0.01 end;End;
 SOLVEQ(3); NCT:=NCT+1;DIF:=0;
 For I:=1 to 3 do Begin DIF:=DIF+abs(F[I]);X[I]:=X[I]−F[I] End;
until (NCT>20) or (DIF<1E−6);
Writeln(OUT,Trunc(T):5,q:9:3,Y1:9:4,V1:9:4,Y2:9:4,V2:9:4,/
  X[2]:9:4, X[1]:9:4,X[3]:9:4);
End;
END.
```

Solution to Example Problem 6.18

		Upstream Gate		Downstream Gate		Beyond Hyd. J.		Vel. H. J.
Time (s)	Q (cfs/ft)	Y_1 (ft)	V_s (fps)	Y_2 (ft)	V_2 (ft)	Y_3 (ft)	V_3 (fps)	V (fps)
0	−5.0000	5.0000	−1.0000	.2865	17.4513	2.1892	2.2840	0.0000
30	−5.5000	4.9569	−1.1096	.3175	17.3207	2.2596	2.5521	0.1374
60	−6.0000	4.9132	−1.2212	.3491	17.1878	2.3250	2.7970	0.2545
90	−6.5000	4.8689	−1.3350	.3812	17.0525	2.3858	3.0220	0.3542
120	−7.0000	4.8238	−1.4511	.4138	16.9147	2.4427	3.2299	0.4385
150	−7.5001	4.7780	−1.5697	.4471	16.7742	2.4961	3.4227	0.5093
180	−8.0001	4.7315	−1.6908	.4810	16.6309	2.5464	3.6023	0.5679
210	−8.5001	4.6841	−1.8147	.5156	16.4847	2.5938	3.7702	0.6155
240	−9.0001	4.6359	−1.9414	.5510	16.3353	2.6386	3.9274	0.6529
270	−9.5001	4.5868	−2.0712	.5871	16.1825	2.6811	4.0751	0.6808
300	−10.0001	4.5367	−2.2043	.6240	16.0262	2.7213	4.2141	0.6999

In examining this solution over time, it should be recognized that the effects of an M_3 GVF-profile downstream from the gate are ignored, as well as the effects of the difference between the slope of the energy line and the slope of the channel bottom. The former effect could be included by solving the GVF equation as an additional equation in which the length over which this solution occurs could be predicted based on the velocity of the moving hydraulic jump, v. The latter effect is most important in the supercritical flow length downstream from the gate. However, the above approach to the problem provides a rough, but reasonable, estimate to the time-dependent movement of the hydraulic jump, and changes in depth and velocity for a short time after the discharge by the gate is changed.

EXAMPLE PROBLEM 6.19

Solve the same problem as given above as Example Problem 6.17, except that, the rate dq/dt at which the flow rate past the gate is increased linearly with time until 250 s, and held constant thereafter. In other words, for 250 s dq/dt = at + b. The rate of linear increase of dq/dt is such that $a = 1.0 \times 10^{-4}$ and $b = 1.0 \times 10^{-3}$ $(ft/s)^2$. Also predict the position downstream from the gate where a secondary surge will likely develop, and the time at which this surge will first appear. In predicting x_s, note that since the rate dq/dt is not constant but increases with time, the position may not, and very likely is not, at the leading edge of the unsteady flow, or at the zone of quiet. Therefore, x_s should be computed for each time, and the value that is the smallest will represent the location of the formation of the surge.

Solution

Integration of dq/dt with respect to time gives q as a function of time, or $q = q_o + (at + b)t$, in which $a = 1.0 \times 10^{-4}$ and $b = 1.0 \times 10^{-3}$. The computer programs above can be modified slightly to solve this problem. These modifications might consist of: First, instead of reading dq/dt, the values of a and b above need to be read, and from these, the flow rate per unit width q is computed from the above integrated equation, (Note that since the q given in these programs first applies for the flow upstream from the gate, a and b will be read as negative values.) second, a statement needs to be added to compute the position of the secondary surge from the equation:

$$x_s = \frac{(V+c)^2}{1.5(g/Y)^{1/2}(\Delta Y/\Delta t)}$$

and third, the variable for x_s needs to be added to the written statement. Using the following line of input data:

```
5, -1, -1E-4, -1E-3, 30, 11, 32.2, 3
```

results in the solution given in the following table.

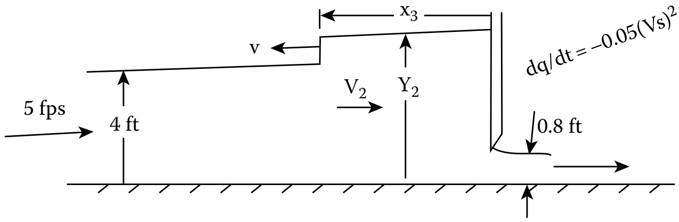
		Upstream Gate		Downstream Gate		Beyond Hyd. J		Vel. H. J.	X-Surge
Time (s)	q (cfs/ft)	Y_1 (ft)	V_1 (fps)	Y_2 (ft)	V_2 (fps)	Y_3 (ft)	V_3 (fps)	v (fps)	Xs (ft)
0	−5.000	5.0000	−1.0000	0.2865	17.4513	2.1892	2.2840	0.0000	—
30	−5.120	4.9897	−1.0261	0.2939	17.4202	2.2066	2.3507	0.0350	34,914
60	−5.420	4.9639	−1.0919	0.3125	17.3418	2.2487	2.5109	0.1169	15,233
90	−5.900	4.9220	−1.1987	0.3427	17.2146	2.3123	2.7497	0.2326	10,918
120	−6.560	4.8635	−1.3488	0.3851	17.0361	2.3929	3.0478	0.3650	9,463
150	−7.400	4.7872	−1.5458	0.4404	16.8026	2.4857	3.3852	0.4961	9,099
180	−8.420	4.6918	−1.7946	0.5100	16.5083	2.5864	3.7440	0.6086	9,325
210	−9.620	4.5749	−2.1028	0.5958	16.1454	2.6909	4.1092	0.6862	9,958
240	−11.000	4.4334	−2.4812	0.7006	15.7019	2.7959	4.4688	0.7131	10,939
270	−12.560	4.2623	−2.9468	0.8285	15.1598	2.8983	4.8133	0.6718	12,271
300	−14.300	4.0535	−3.5278	0.9870	14.4890	2.9956	5.1348	0.5386	14,028

The position of a secondary surge will be at about 9100 ft at a time of 150 s, the minimum distance, x_s in the solution table above.

EXAMPLE PROBLEM 6.20

The flow upstream from a gate in a rectangular channel has been at a depth of $Y_o = 4$ ft and a velocity of $V_o = 5$ fps for some time, when suddenly the gate is dropped so as to produce a depth of 0.8 ft downstream from it and a surge upstream from it. Determine the depth and the velocity immediately upstream from the gate and the speed of the surge. After this initial drop, the gate is gradually closed further so as to decrease the flow past the gate at a rate dq/dt = 0.01 $(ft/s)^2$.

Use the methods of this chapter to approximately track the movement of the surge and the depth upstream from the gate.



Solution

First, a steady-state solution is called for based on the continuity and the momentum equations from the viewpoint of a moving observer, as is described in Chapter 3. Designating the steady-state upstream depth and the velocity by an s subscript, the three equations that solve Y_2, V_2, and v are

$$F_1 = (|V_2| + v)Y_2 - (|V_s| + v)Y_s = 0 \tag{1}$$

$$F_2 = .5\left(\frac{Y_2}{Y_s}\right)\left(\frac{Y_2}{Y_s} + 1\right) - \frac{(|V_s| + v)^2}{(gY_s)} = 0 \tag{2}$$

$$F_3 = Y_2 + \frac{V_2^2}{(2g)} - Y_d - \frac{(Y_2V_2)^2}{(2gY_d{}^2)} = 0 \tag{3}$$

The solution gives: $Y_2 = 4.848$ ft, $V_2 = 2.701$ fps, and v = 8.140 fps, and can be obtained using any software that is designed to solve systems of nonlinear algebraic equations.

To apply the methods of this chapter to the unsteady flow assume that they are applicable to the expanding region between the moving surge and the gate and that the initial depth and the velocity solved above, i.e., the Y_2 and V_2 represent $Y_o = 4.848$ ft and $V_o = -2.701$ fps, i.e., the initial conditions for the unsteady solution and those that have been used in the equations of this chapter. The equations that are available are the first two equations above, i.e., the continuity and the momentum equations from a moving observer's viewpoint, plus the equation that indicates that V − 2c is constant along negative characteristics, the equation that indicates that $t = t_1 + \Delta t$, that the new flow rate past the gate $q_d = q_o + (dq/dt)t$, and finally again, the energy across the gate. In other words, there are six available equations for each new time step. These equations are

$$F_1 = (|V_2| + v)Y_2 - (|V_s| + v)Y_s = 0 \tag{4}$$

$$F_2 = .5\left(\frac{Y_2}{Y_s}\right)\left(\frac{Y_2}{Y_s} + 1\right) - \frac{(|V_s| + v)^2}{(gY_s)} = 0 \tag{5}$$

$$F_3 = V_2 - 2(gY_2)^{1/2} - V_o + 2(gY_o)^{1/2} = 0 \tag{6}$$

$$F_4 = t - \frac{(V_2Y_2 - q_o)}{|dq/dt|} - \frac{x}{\left\{3(gY_2)^{1/2} + V_o - 2(gY_o)^{1/2}\right\}} = 0 \tag{7}$$

$$F_5 = V_oY_o - q_d = 0 \quad \left(\text{in which } q_d = q_o - \left|\frac{dq}{dt}\right| t\right) \tag{8}$$

$$F_6 = Y_o + \frac{V_o^2}{(2g)} - Y_d - \frac{(q_d/Y_d)}{(2g)} = 0 \tag{9}$$

These six equations can be repeatedly solved for each new time step for the following six variables: Y_2, V_2, v, Y_o, V_o, and Y_d. (Note that Y_2 and V_2 from the above solution, for after the instant drop of the gate, have been renamed as Y_o and V_o to be consistent with the notation used for the initial uniform flow conditions, but also that these variables actually change as the gate is closed further to provide the specified dq/dt.) The position x_s of the moving surge can be determined by multiplying the speed of the surge v by the time increment, or $x = \Sigma\, v\Delta t$.

The above solution procedures are implemented in the FORTRAN program SURGMO6.FOR, the listing of which is given below. This program first solves the three Equations 1, 2, and 3 that describe the beginning surge size, the velocity, etc., by the Newton method. Thereafter, for each time step, (The DO 80 K=1,NT) the six Equations 4 through 9 are solved using the Newton method giving Y_2, V_2, v, Y_o, V_o, and Y_d for that time step. Note that before solving these equations that include Equations 6 and 7 from the characteristics, V_o and V_2 are made negative to reflect that the x axis is pointing upstream. q_o is also negative. The product of Y_2 and V_2 obtain from the solution of Equations 1 through 3. dq/dt is read as a positive value, and therefore to make q_d less negative than q_o its value is multiplied by time t which is added to q_o.

Program SURGMO6.FOR

```
      LOGICAL*2 ISTEAD
      INTEGER*2 INDX(6)
      REAL F(6),F1(6),D(6,6)
      COMMON g,G2,qo,qd,T,dqdt,Xs,Ys,Vs,X(6),ISTEAD
      EQUIVALENCE (Y2,X(1)),(V2,X(2)),(v,X(3)),(Yo,X(4)),
     &(Vo,X(5)),(Yd,X(6))
      WRITE(*,*) ' Give: Yo,Vo,Yd,IOUT,NT,DELT,dqdt,g'
      READ(*,*) Yo,Vo,Yd,IOUT,NT,DELT,dqdt,g
      ISTEAD=.TRUE.
      Ys=Yo
      Vs=Vo
      G2=2.*g
      Xs=0.
      Y2=1.2*Yo
      V2=.55*Vo
      v=1.8*Vo
      NCT=0
5     CALL FUNCT(F1)
      DO 20 J=1,3
      X1=X(J)
      X(J)=1.005*X1
      CALL FUNCT(F)
      DO 10 I=1,3
10    D(I,J)=(F(I)-F1(I))/(X(J)-X1)
20    X(J)=X1
      CALL SOLVEQ(3,1,6,D,F1,1,DD,INDX)
      NCT=NCT+1
      DIF=0.
      DO 30 J=1,3
      X(J)=X(J)-F1(J)
30    DIF=DIF+ABS(F1(J))
      IF(NCT.LT.20 .AND. DIF.GT.1.E-5) GO TO 5
      ISTEAD=.FALSE.
      V2=-V2
      Vo=V2
      qo=Vo*X(1)
```

```
      Yo=Y2
      WRITE(IOUT,100) (X(I),I=1,4),Vs,Yd,qo
100   FORMAT(' Steady Solution: Y2=',F8.3,' V2=',F8.3,' v=',
     &F8.3,/,' Yo=',F8.3,' Vo=',F8.3,' Yd=',F8.3,' qd=',F9.3)
      WRITE(IOUT,110)
110   FORMAT(1X,76('-'),/,' Time Y2 V2 v',' Yo Vo Yd x qd'/,
     &1X,76('-'))
      WRITE(IOUT,120) 0.,X,Xs,qo
      DO 80 K=1,NT
      NCT=0
      T=DELT*FLOAT(K)
      qd=qo+ABS(dqdt)*T
      v1=v
      Xs=Xs+DELT*v
40    CALL FUNCT(F1)
      DO 60 J=1,6
      X1=X(J)
      X(J)=1.005*X1
      CALL FUNCT(F)
      DO 50 I=1,6
50    D(I,J)=(F(I)-F1(I))/(X(J)-X1)
60    X(J)=X1
      CALL SOLVEQ(6,1,6,D,F1,1,DD,INDX)
      NCT=NCT+1
      DIF=0.
      DO 70 J=1,6
      X(J)=X(J)-F1(J)
70    DIF=DIF+ABS(F1(J))
      WRITE(*,*)' NCT',NCT,DIF
      IF(NCT.LT.20 .AND. DIF.GT.1.E-5) GO TO 40
80    WRITE(IOUT,120) T,X,Xs,qd
120   FORMAT(F5.1,6F9.3,F9.1,F9.3)
      END
      SUBROUTINE FUNCT(F)
      REAL F(6)
      LOGICAL*2 ISTEAD
      COMMON g,G2,qo,qd,T,dqdt,Xs,Ys,Vs,X(6),ISTEAD
      F(1)=(ABS(X(2))+X(3))*X(1)-(ABS(Vs)+X(3))*Ys
      F(2)=.5*(X(1)/Ys)*(X(1)/Ys+1.)-(ABS(Vs)+X(3))**2/(g*Ys)
      IF(ISTEAD) THEN
      F(3)=X(1)+X(2)**2/G2-X(6)-((X(1)*X(2))/X(6))**2/G2
      ELSE
      F(3)=X(2)-X(5)+2.*(SQRT(g*X(4))-SQRT(g*X(1)))
      F(4)=T-ABS(X(1)*X(2)-qo)/ABS(dqdt)-Xs/(3.*SQRT(g*X(1))
     &+X(5)-2.*SQRT(g*X(4)))
      F(5)=X(4)*X(5)-qd
      F(6)=X(4)+X(5)**2/G2-X(6)-(qd/X(6))**2/G2
      ENDIF
      RETURN
      END
```

Input to program to solve problem: `4 5 .8 3 20 30 .01 32.2`

Solution output:

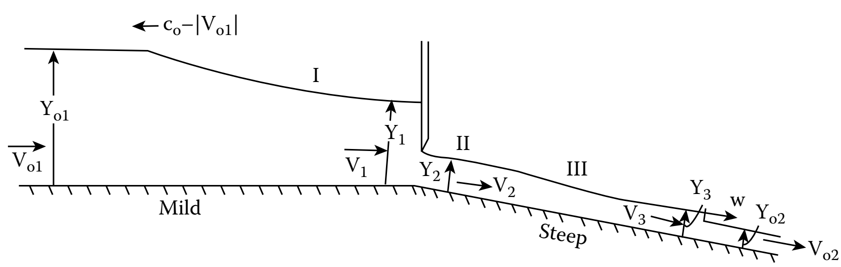
Steady Solution: Y2 = 4.848 V2 = `−2.701 v = 8.140`

`Yo = 4.848 Vo = 5.000 Yd = .800 qd= −13.096`

Time	Y_2	V_2	v	Y_o	V_o	Y_d	x	q_d
0.0	4.848	−2.701	8.140	4.848	−2.701	0.800	0.0	−13.096
30.0	4.843	−2.715	8.129	4.879	−2.623	0.777	244.2	−12.796
60.0	4.838	−2.728	8.119	4.909	−2.546	0.755	488.1	−12.496
90.0	4.833	−2.740	8.108	4.939	−2.470	0.733	731.6	−12.196
120.0	4.828	−2.753	8.098	4.968	−2.395	0.711	974.9	−11.896
150.0	4.823	−2.766	8.088	4.997	−2.321	0.690	1217.8	−11.596
180.0	4.819	−2.778	8.078	5.026	−2.248	0.668	1460.5	−11.296
210.0	4.814	−2.791	8.068	5.055	−2.176	0.647	1702.8	−10.996
240.0	4.809	−2.803	8.058	5.083	−2.104	0.627	1944.9	−10.696
270.0	4.805	−2.815	8.049	5.111	−2.034	0.606	2186.6	−10.396
300.0	4.800	−2.827	8.039	5.139	−1.965	0.586	2428.1	−10.096
330.0	4.796	−2.839	8.030	5.166	−1.896	0.566	2669.3	−9.796
360.0	4.791	−2.850	8.020	5.193	−1.829	0.546	2910.1	−9.496
390.0	4.787	−2.862	8.011	5.220	−1.762	0.526	3150.7	−9.196
420.0	4.782	−2.873	8.002	5.247	−1.696	0.507	3391.1	−8.896
450.0	4.778	−2.884	7.993	5.273	−1.630	0.488	3631.1	−8.596
480.0	4.774	−2.896	7.984	5.300	−1.565	0.469	3870.9	−8.296
510.0	4.769	−2.907	7.975	5.326	−1.502	0.450	4110.4	−7.996
540.0	4.765	−2.918	7.966	5.351	−1.438	0.431	4349.7	−7.696
570.0	4.761	−2.928	7.958	5.377	−1.376	0.412	4588.7	−7.396
600.0	4.757	−2.939	7.949	5.402	−1.314	0.394	4827.4	−7.096

6.12 PARTIAL INSTANT OPENING OF GATES IN RECTANGULAR CHANNELS

Now, let us cover an analysis that examines the flows resulting from the operation of gates in which the gate's position is instantly changed. Assume that a gate has been set so as to permit a flow rate per unit width q_1 past it in a rectangular channel for a long time and is then suddenly raised to increase the flow rate to q_2. To allow the theory to have a semblance to reality, assume that the slope of the channel changes at the gate, so initially, near-uniform flows occur both upstream and downstream from the gate, so that as the gate is raised, the situation shown in the sketch below occurs, with uniform flow at a constant depth Y_{o1} and at a constant velocity V_{o1} upstream from where the effect of opening the gate has propagated to, and Y_{o2} and V_{o2} downstream from where the effect of opening the gate is felt. Upstream from the gate, there will be a zone I with a length that increases at a rate $c_{o1} - |V_{o1}|$ that is unsteady. Downstream from the gate, there will be a zone II in which the depth Y_2 and the velocity V_2 will be assumed to be constant. There will be a zone III governed by the characteristics with a constant height wave at the leading edge moving at a velocity w over the initial downstream flow at a depth Y_{o2} and V_{o2}.



Typically, the depth Y_2 will be known from the new setting of the gate along with the variables Y_{o1}, V_{o1}, Y_{o2}, and V_{o2}, so that the following variables are unknown: Y_1, V_1, V_2, Y_3, V_3, and w. Thus, six equations are needed. These are

1. Energy across the gate (stationary observer)

$$Y_1 + \frac{V_1^2}{2g} = Y_2 + \frac{V_2^2}{2g} \tag{6.46}$$

2. Continuity across the gate (stationary observer)

$$V_1 Y_1 = V_2 Y_2 \tag{6.47}$$

3. Continuity across the wave moving with a velocity w (moving observer)

$$(w - V_3)Y_3 = (w - V_{02})Y_{02} \tag{6.48}$$

4. Momentum across the wave (moving observer)

$$\frac{1}{2}Y_3^2 + \frac{(w - V_3)^2 Y_3}{g}\ \frac{1}{2}Y_{02}^2 + \frac{(w - V_{02})^2 Y_{02}}{g} \tag{6.49}$$

5. and 6 Characteristic equations for the unsteady flow in zones I and III. These equations are obtained by noting that V − 2c is constant along the C^- characteristics. However, since V is negative with x pointing upstream, and as it will be more convenient to let V be positive, we will let −|V| − 2c = constant, so the equation in zone I is

$$V_1 + 2c_1 = V_{o1} + 2c_{o1} \quad \text{or} \quad V_1 + 2\sqrt{gY_1} = V_{o1} + 2\sqrt{Y_{o1}} \tag{6.50}$$

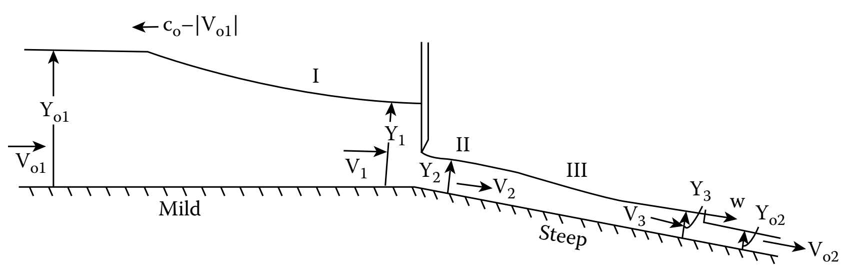
and in zone III is

$$V_2 + 2c_2 = V_3 + 2c_3 \quad \text{or} \quad V_2 + 2\sqrt{gY_2} = V_3 + 2\sqrt{gY_3} \tag{6.51}$$

Because Equation 6.47 is simple, it might be eliminated along with V_2 as an unknown by substituting $V_2 = V_1(Y_1/Y_2)$ in Equations 6.46 and 6.51. An even simpler approach, especially if the solution is done by hand, is available by noting that since the gate controls both the upstream and the downstream flows, one needs to solve Equations 6.46, 6.47, and 6.50 for Y_1, V_1, and V_2 and then Equations 6.48, 6.49, and 6.51 for w, V_3, and Y_3, for separate solutions.

EXAMPLE PROBLEM 6.21

Initially, the depth upstream and downstream from a gate is 8 and 2 ft respectively. Suddenly, the gate is raised so as to produce a depth of Y_2 = 3.5 ft downstream from the gate. Assuming that initially, a uniform flow exists upstream and downstream from the gate, use the characteristic to determine the new depth Y_1 and the velocity V_1 immediately upstream from the gate, the velocity V_2 immediately downstream from the gate, the depth Y_3, and the velocity V_3 at the wave that will move downstream from the gate and the velocity of this wave w.



Solution

First, the energy equation $8 + q_1^2/(2g8^2) = 2 + q_1^2/(2g2^2)$ across the gate needs to be solved for q_1 = 40.603 cfs/ft. Next, solve $V_{o1} = q_1/8$ = 5.075 fps, and $V_{o2} = q_1/2$ = 20.302 fps. Now, the six equations can be solved for the six unknowns. Three separate TK-Solver models are given below to solve this problem. The first solves all six equations simultaneously; the second solves the three equations that govern the flow upstream from the gate; and the third, the three equations that govern the flow downstream from the gate. Note, the two approaches yield the same solution: Y_1 = 6.144 ft, V_1 = 9.045 fps, V_2 = 15.877 fps, V_3 = 20.890 fps, Y_3 = 2.096 ft, and w = 28.613 fps. The flow rate after the gate is raised is $q_2 = V_1Y_1$ = (9.045)(6.144) = 55.569 cfs/ft or $q_2 = V_2Y_2$ = (15.877)(3.5) = 55.569 cfs/ft, or the increased flow rate is Δq = 14.966 cfs/ft. To the moving observer, the flow is supercritical downstream from the wave and the subcritical upstream therefrom, i.e., $F_{ro2}^2 = (w - V_{o2})^2/(gY_{o2}) = 1.0727$ and $F_{r3}^2 = (w - V_3)^2/(gY_3) = 0.9327$. (See Homework Problem 6.103 for a Fortran Program to solve this problem.)

TK-Solver model for entire problem, GATEUP.TK (or GATEUP1.TK that eliminates V_2 as an unknown

```
========== VARIABLE SHEET ========== and combines the continuity
equation with 1st eq.)
St Input------   Name------   Output------   Unit
                 Y1           6.1439597
                 V1           9.0444734
                 V2           15.876823
                 w            28.613414
                 V3           20.680377
                 Y3           2.0954622
   32.2          g
   3.5           Y2
   20.301724     Vo2
   2             Yo2
   5.075431      Vo1
   8             Yo1
========== RULE SHEET ==========
S Rule------------------------
  Y1+V1^2/(2*g)=Y2+V2^2/(2*g)
  V1*Y1=V2*Y2
  (w-V3)*Y3=(w-Vo2)*Yo2
  .5*Y3^2+(w-V3)^2*Y3/g=.5*Yo2^2+(w-Vo2)^2*Yo2/g
  V1+2*sqrt(g*Y1)=Vo1+2*sqrt(g*Yo1)
  V2+2*sqrt(g*Y2)=V3+2*sqrt(g*Y3)
```

TK-Solver model for problem upstream from gate, GATEUP2.TK

```
========== VARIABLE SHEET ==========
St Input------   Name------   Output------   Unit
                 Y1           6.1439597
                 V1           9.0444734
                 V2           15.876823
   32.2          g
   3.5           Y2
   5.075431      Vo1
   8             Yo1
```

```
========== RULE SHEET ==========
S Rule-------------------------
* Y1+V1^2/(2*g)=Y2+V2^2/(2*g)
* V1*Y1=V2*Y2
* V1+2*sqrt(g*Y1)=Vo1+2*sqrt(g*Yo1)
```

TK-Solver model for problem downstream from gate, GATEUP3.TK

```
========== VARIABLE SHEET ==========
St Input-----   Name-----   Output-----   Unit
                w           28.613405
                V3          20.680365
                Y3          2.0954593
   32.2         g
   3.5          Y2
   20.301724    Vo2
   2            Yo2
   15.8768      V2
========== RULE SHEET ==========
S Rule-------------------------
  (w-V3)*Y3=(w-Vo2)*Yo2
  .5*Y3^2+(w-V3)^2*Y3/g=.5*Yo2^2+(w-Vo2)^2*Yo2/g
  V2+2*sqrt(g*Y2)=V3+2*sqrt(g*Y3)
```

EXAMPLE PROBLEM 6.22
Assume the gate in the previous problem has been operating for a long time in which it produces a depth of 3.5 ft downstream from it, and now suddenly the gate is lowered to decrease the flow rate past it, so it produces a depth of 2 ft downstream from it. Solve the resulting flows based on the assumptions that allow the characteristic method to be used to solve the unsteady flow.

Solution
With the gate instantly lowered, both an upstream and a downstream wave will be created and shown in the sketch. Typically, as in this problem, the new depth downstream from the gate as well as the depth Y_{o1} and the velocity V_{o1}, and the downstream depth Y_{o1} and the velocity V_{o1} are known, and the seven unknown variables are: Y_1, V_1, V_2, Y_3, V_3, w_u, and w_d. The seven equations that allow for these to be solved are

1. Energy across the gate (stationary observer)

$$Y_1 + \frac{V_1^2}{2g} = Y_2 + \frac{V_2^2}{2g}$$

2. Continuity across the gate (stationary observer)

$$V_1Y_1 = V_2Y_2$$

3. Continuity across the upstream wave (moving observer)

$$(w_u + V_{o1})Y_{o1} = (w_u + V_1)Y_1$$

4. Momentum across the upstream wave (moving observer)

$$\frac{Y_{o1}^2}{2} + \frac{(w_u + V_{o1})^2 Y_{o1}}{g} = \frac{Y_1^2}{2} + \frac{(w_u + V_1)^2 Y_1}{g}$$

5. Continuity across the downstream wave (moving observer)

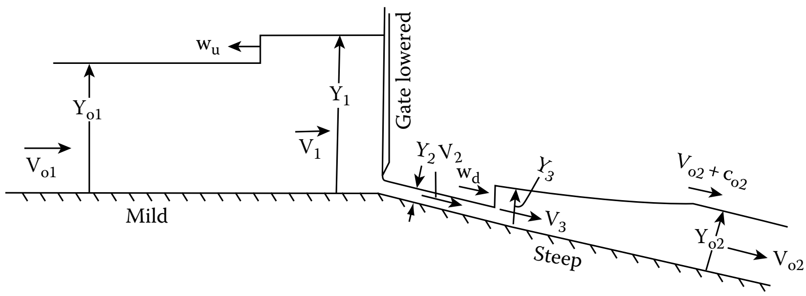
$$(V_2 - w_d)Y_2 = (V_3 - w_d)Y_3$$

6. Momentum across the downstream wave (moving observer)

$$\frac{Y_2^2}{2} + \frac{(V_2 - w_d)^2 Y_2}{g} = \frac{Y_3^2}{2} + \frac{(V_3 - w_d)^2 Y_3}{g}$$

7. Characteristic equation for unsteady flow

$$V_3 - 2c_3 = V_{o2} - 2c_{o2} \text{ or } V_3 - 2(gY_3)^{1/2} = V_{o2} - 2(gY_{o2})^{1/2}$$



A TK-Solver model is given below. An alternative is to solve the upstream and the downstream portions of the problem separately. In so doing, the first four equations would be solved for the first four unknowns for the variables upstream of the gate including V_2, and then the last three equations for the last three unknowns.

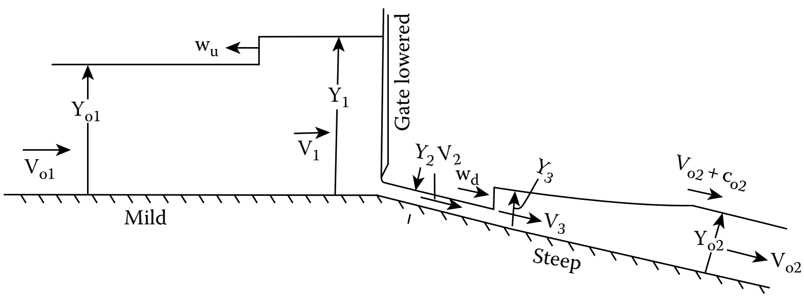
Solving all 7 equations, model GATEDW.TK (see GATEDW1.TK to solve first 4 equations followed by

```
==========VARIABLE SHEET========== GATEDW2.TK to solve last 3
equations.)
St Input————  Name————  Output————  Unit
              Y1        7.9855713
              V1        5.0790966
              V2        20.279744
              Wu        8.1506047
              Wd        8.2580442
              V3        15.42769
              Y3        3.3534989
  2           Y2
  15.8768     Vo2
  3.5         Yo2
  9.0445      Vo1
  6.144       Yo1
  32.2        g

=========== RULE SHEET ==========
S Rule————————————————————————————
* Y1+V1^2/(2*g)=Y2+V2^2/(2*g)
* V1*Y1=V2*Y2
* (Wu+Vo1)*Yo1=(Wu+V1)*Y1
* .5*Yo1^2+(Wu+Vo1)^2*Yo1/g=.5*Y1^2+(Wu+V1)^2*Y1/g
* (V2-Wd)*Y2=(V3-Wd)*Y3
* .5*Y2^2+(V2-Wd)^2*Y2/g=.5*Y3^2+(V3-Wd)^2*Y3/g
* V3-2*sqrt(g*Y3)=Vo2-2*sqrt(g*Yo2)
```

6.13 PARTIAL INSTANT CLOSING OF GATES IN TRAPEZOIDAL CHANNELS

The use of characteristics in solving the unsteady flows created by the instant partial closure or opening of gates can be extended to nonrectangular channels by using the stage variable w. As has been shown previously, to handle nonrectangular channels, w replaces 2c. In this section, we illustrate the use of these methods for the instant partial closure of a gate in a trapezoidal channel, but the same approach can be used for a circular channel (or other shaped channels), and by using the appropriate equations the unsteady flow caused by the instant partial opening of a gate can be solved.

When a gate is instantly partially closed in a trapezoidal channel, a surge is formed both upstream and downstream from the gate as shown in the sketch in Example Problem 6.19, which is duplicated below. Typically, the new depth Y_2, downstream from the gate, will be known along with the previous steady uniform depths and the velocities upstream and downstream from the gate, Y_{o1}, V_{o1}, Y_{o2}, and V_{o2}, before it was partially closed,. Actually, the initial flow rate Q_{o1}, might be given, with either the upstream, or the downstream depth. If so, then by solving the energy equation across the gate, these quantities can be obtained. Also, the velocity V_2, or the new flow rate Q_2 might be given rather than Y_2. If the variables Y_{o1}, V_{o1}, Y_{o2}, V_{o2}, and Y_2 are known, the following seven variables will be unknown, as in Example Problem 6.19: Y_1, V_1, V_2, Y_3, V_3, w_u, and w_d. The seven equations needed to solve these unknowns are given below. Note, these equations are the same as those used in Example Problem 6.19 with w replacing 2c in the characteristic equation and areas replacing depths in the continuity equations.



1. Energy across the gate (stationary observer)

$$F_1 = Y_1 + \frac{V_1^2}{2g} - Y_2 - \frac{V_2^2}{2g} = 0 \tag{6.52}$$

2. Continuity across the gate (stationary observer)

$$F_2 = V_1A_1 - V_2A_2 = 0 \tag{6.53}$$

3. Continuity across the upstream wave (moving observer)

$$F_3 = (w_u + V_{o1})A_{o1} - (w_u + V_1)A_1 = 0 \tag{6.54}$$

4. Momentum across the upstream wave (moving observer)

$$F_4 = \frac{1}{2}bY_{o1}^2 + \frac{1}{3}mY_{o1}^3 + \frac{(w_u + V_{o1})^2 A_{o1}}{g} - \frac{1}{2}bY_1^2 - \frac{1}{3}mY_1^3 - \frac{(w_u + V_1)^2 A_1}{g} \tag{6.55}$$

5. Continuity across the downstream wave (moving observer)

$$F_5 = (v_2 - w_d)A_2 - (V_3 - w_d)A_3 \tag{6.56}$$

6. Momentum across the downstream wave (moving observer)

$$F_6 = \frac{1}{2}bY_2^2 + \frac{1}{3}mY_2^3 \frac{(w_d - V_2)^2 A_2}{g} - \frac{1}{2}bY_3^2 - \frac{1}{3}mY_3^3 + \frac{(w_d - V_3)^2 A_3}{g} \tag{6.57}$$

7. Characteristic equation for unsteady flow

$$F_7 = (V_3 - w_3) - (V_{o2} - w_{o2}) \tag{6.58}$$

in which the w's in the last equation, Equation 6.58, are the stage variables associated with depths Y_3 and Y_{o2}, respectively, and w_u and w_d are the velocities of the upstream and the downstream surges, respectively.

The program GATETR is designed to solve these seven equations using the Newton method. The program requires two lines of input; the first for the known variables (including the acceleration of gravity, g), and the second for guesses for the unknown variables. As has been done in previous programs, the subroutine FUN evaluates the equations based on the current values for the unknowns that are stored in the array X. In order to evaluate the stage variables w, it first generates a table for w versus Y for the given trapezoidal channel in the same manner as program YTIMEX does, and then depending upon the value for the depth interpolates in this table to get the corresponding value of w.

Listing of program GATETR.FOR

```
      PARAMETER (N=200)
      CHARACTER*2 UNK(7)/'Y1','V1','V2','Y3','V3','Wu','Wd'/
      INTEGER*2 INDX(7)
      REAL F(7),FF(7),D(7,7)
      COMMON B,FM,BH,FM3,Y2,W(200),X(7),DY,VOW,Yo1,Vo1,Yo2,Vo2,G,G2,
     &Ao1,Ao2,Ao1G,Ao2G,A2,A2G,FMON1,FMON2
      FW(Y)=SQRT(G*(B+FM2*Y)/((B+FM*Y)*Y))
      WRITE(*,*)' Give: g,b,m,Yo1,Vo1,Yo2,Vo2,Y2'
      READ(*,*) G,B,FM,Yo1,Vo1,Yo2,Vo2,Y2
      WRITE(*,100) UNK
100   FORMAT(' Give guesses for:',6(A3,','),A3)
      READ(*,*) X
      G2=2.*G
      DY=(Yo2-.01)/FLOAT(N-1)
      FM2=2.*FM
      FM3=FM/3.
      BH=B/2.
      Ao1=(B+FM*Yo1)*Yo1
      Ao1G=Ao1/G
      Ao2=(B+FM*Yo2)*Yo2
      Ao2G=Ao2/G
      A2=(B+FM*Y2)*Y2
      A2G=A2/G
      FMON1=(BH+FM3*Yo1)*Yo1**2
      FMON2=(BH+FM3*Y2)*Y2*Y2
      DY2=DY/10.
```

```
      DYH=DY2/2.
      Y=.01
      WF1=FW(Y)
      W(1)=0.
      DO 10 I=2,N
      SUM=0.
      DO 5 J=1,10
      WF=FW(Y+DY2*FLOAT(J))
      SUM=SUM+DYH*(WF1+WF)
5     WF1=WF
      W(I)=W(I-1)+SUM
10    Y=Y+DY
      VOW=Vo2-W(N)
      NCT=0
12    SUM=0.
      CALL FUN(F)
      DO 20 J=1,7
      XT=X(J)
      X(J)=1.005*X(J)
      CALL FUN(FF)
      DO 15 I=1,7
15    D(I,J)=(FF(I)-F(I))/(X(J)-XT)
20    X(J)=XT
      CALL SOLVEQ(7,1,7,D,F,1,DD,INDX)
      NCT=NCT+1
      DO 30 I=1,7
      X(I)=X(I)-F(I)
30    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,X
110   FORMAT(' NCT=',I3,', SUM=',E12.5,/7F9.3)
      IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 12
      WRITE(*,120)(UNK(I),X(I),I=1,7)
120   FORMAT(A3,' = ',F10.3)
      END
      SUBROUTINE FUN(F)
      REAL F(7)
      COMMON B,FM,BH,FM3,Y2,W(200),X(7),DY,VOW,Yo1,Vo1,Yo2,Vo2,G,G2,
     &Ao1,Ao2,Ao1G,Ao2G,A2,A2G,FMON1,FMON2
      F(1)=X(1)+X(2)**2/G2-Y2-X(3)**2/G2
      A1=(B+FM*X(1))*X(1)
      A3=(B+FM*X(4))*X(4)
      F(2)=X(2)*A1-X(3)*A2
      F(3)=FMON1+(X(6)+Vo1)**2*Ao1G-(BH+FM3*X(1))*X(1)**2-(X(6)
     &+X(2))**2*A1/G
      F(4)=(X(6)+Vo1)*Ao1-(X(6)+X(2))*A1
      F(5)=FMON2+(X(7)-X(3))**2*A2G-(BH+FM3*X(4))*X(4)**2-(X(7)
     &-X(5))**2*A3/G
      F(6)=(X(7)-X(3))*A2-(X(7)-X(5))*A3
      IM=(X(4)-.01)/DY
      FAC=(X(4)-DY*FLOAT(IM))/DY
      IF(IM.GT.199) IM=199
```

```
      IF(IM.LT.1) IM=1
      F(7)=X(5)-(W(IM)+FAC*(W(IM+1)-W(IM)))-VOW
      RETURN
      END
```

Listing of Program GATETR.C

```
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
float
b,m,bh,fm2,fm3,y2,w[200],x[7],dy,vow,yo1,vo1,yo2,vo2,g,g2,ao1,ao2,\
  ao1g,ao2g,a2,a2g,fmon1,fmon2;
float fw(float y) {return (sqrt(g*(b+fm2*y)/((b+m*y)*y)));}
extern void solveq(int n,float **a,float *b,int itype,\
  float *dd,int *indx);
void fun(float *f){float a1,a3,fac; int im;
 a1=(b+m*x[0])*x[0]; a3=(b+m*x[3])*x[3];
   f[0]=x[0]+x[1]*x[1]/g2-y2-x[2]*x[2]/g2;
 f[1]=x[1]*a1-x[2]*a2;
 f[2]=fmon1+pow(x[5]+vo1,2.)*ao1g-(bh+fm3*x[0])*x[0]*x[0]-\
   pow(x[5]+x[1],2.)*a1/g;
 f[3]=(x[5]+vo1)*ao1-(x[5]+x[1])*a1;
 f[4]=fmon2+pow(x[6]-x[2],2.)*a2g-(bh+fm3*x[3])*x[3]*x[3]-\
   pow(x[6]-x[4],2.)*a3/g;
 f[5]=(x[6]-x[2])*a2-(x[6]-x[4])*a3;
 im=(x[3]-.01)/dy; fac=(x[3]-dy*(float)im)/dy;
   if(im>199)im=199;if(im<1)im=1;
 f[6]=x[4]-(w[im-1]+fac*(w[im]-w[im-1]))-vow;
} // End of fun
void main(void){int indx[7],i,j,nct;
  float sum,y,dy2,dyh,wf1,wf,xt,*dd,f[7],ff[7],**d;
char *unk[7]={"Y1","V1","V2","Y3","V3","Wu","Wd"};
d=(float**)malloc(7*sizeof(float*));
  for(i=0;i<7;i++)d[i]=(float*)malloc(7*sizeof(float));
printf("Give: g,b,m,Yo1,Vo1,Yo2,Vo2,Y2\n");
scanf("%f %f %f %f %f %f %f %f",&g,&b,&m,&yo1,&vo1,&yo2,&vo2,&y2);
printf("Give guesses for: ");for(i=0;i<6;i++) printf("%s,",unk[i]);
  printf("%s\n",unk[6]);
for(i=0;i<7;i++)scanf("%f",&x[i]);
g2=2.*g;dy=(yo2-.01)/199.;fm2=2.*m; fm3=m/3.;bh=b/2.;
  ao1=(b+m*yo1)*yo1;ao1g=ao1/g;
ao2=(b+m*yo2)*yo2;ao2g=ao2/g;a2=(b+m*y2)*y2;a2g=a2/g;
  fmon1=(bh+fm3*yo1)*yo1*yo1; fmon2=(bh+fm3*y2)*y2*y2;
dy2=dy/10.;dyh=dy2/2.;y=.01;wf1=fw(y);w[0]=0.;
for(i=1;i<200;i++){sum=0.;for(j=1;j<11;j++){wf=fw(y+dy2*(float)j);
  sum+=dyh*(wf1+wf);wf1=wf;}
 w[i]=w[i-1]+sum; y+=dy;} vow=vo2-w[199]; nct=0;
do{sum=0.; fun(f);for(j=0;j<7;j++){xt=x[j]; x[j]*=1.005; fun(ff);
for(i=0;i<7;i++)d[i][j]=(ff[i]-f[i])/(x[j]-xt);x[j]=xt;}
solveq(7,d,f,1,dd,indx); nct++;
```

```
for(i=0;i<7;i++){x[i]-=f[i];sum+=fabs(f[i]);}
printf("NCT= %d SUM= %f\n",nct,sum);
  for(i=0;i<7;i++)printf("%9.3f",x[i]);printf("\n");
} while((nct<30) && (sum>1.e-5));
for(i=0;i<7;i++) printf("%s %10.3f\n",unk[i],x[i]);
}
```

EXAMPLE PROBLEM 6.23

A gate in a trapezoidal channel with b = 12 ft and m = 1.0 has been set so as to produce a depth of Y_{o1} = 10 ft upstream from it and Y_{o2} = 4 ft downstream from it for a long time. Suddenly, the gate is lowered so as to cause the depth immediately downstream from it to be Y_2 = 2.0 ft. Solve the resulting unsteady flow upstream and downstream from the gate.

Solution

To obtain the initial steady-state velocities, the energy equation across the gate must be solved, i.e. $E_{o1} = E_{o2}$, for the flow rate Q_1 = 1314.92 cfs. Next, $V_{o1} = Q_1/A_{o1}$ = 1314.92/220 = 5.977 fps, and V_{o2} = 1314.92/64 = 20.546 fps. The specific energy is E_1 = 10.555 ft, and the upstream and the downstream Froude numbers are: F_{r1} = 0.402 and F_{r2} = 2.024, respectively. The new flow rate past the gate is V_1A_1 = 2.551(274.78) = 700.9 cfs or V_2A_2 = 25.032(28) = 700.9 cfs. The input to program GATETR is

32.2	12	1	10	5.977	4	20.546	2
13	2	27	3.8	21	8	20	

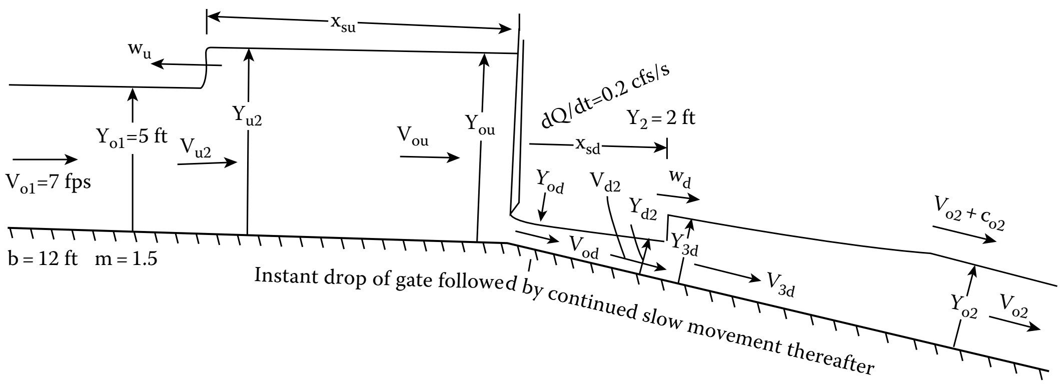
and it produces the solution: Y_1 = 11.629 ft, V_1 = 2.551 fps, V_2 = 25.032 fps, Y_3 = 3.545 ft, V_3 = 19.033 fps, w_u = 11.208 fps, and w_d = 12.839 fps.

6.14 PARTIAL INSTANT CLOSURE FOLLOWED BY SLOW MOVEMENT THEREAFTER

6.14.1 Rectangular Channels

In Sections 6.12 and 6.13, instant closure of a gate occurred when uniform flow existed both upstream and downstream from the gate. In this section, the gate will continue to close after this instant partial closure so that a positive wave will propagate upstream from the gate and a negative wave will propagate downstream from the gate in addition to the moving surges that were caused by the instant closure of the gate. The sketch below depicts the elements of the problem. The upstream moving hydraulic bore's velocity will be identified by w_u, as previously, and it will move over the upstream initial uniform flow conditions with a depth Y_{o1} and a velocity, V_{o1}. The depth and the velocity behind this surge will be denoted by Y_{u2} and V_{u2}. Immediately upstream from the gate, the depth and the velocity will be denoted by Y_{ou} and V_{ou}, and a positive wave connects these two upstream depths. (Positive wave because the case being considered is for the gate continuing to close.) As before, the initial uniform depth and the velocity, Y_{o2} and V_{o2} will exist at the end of the channel because we will assume the channel is very long, and therefore the effect of the gate closure, which moves with a velocity $V_{o2} + c_{o2}$, will never come to the end of the channel. The downstream hydraulic bore resulting from the instant first partial closure of the gate will travel downstream with a velocity w_d. and the depth and the velocity on its downstream side (the larger depth) will be denoted by Y_{3d} and the associated velocity by V_{3d}. On the upstream side of this downstream bore, the depth and the velocity will be denoted by Y_{d2} and V_{d2}, respectively. The depth and the velocity immediately downstream from the gate will be denoted by Y_{od} and V_{od}. These variables are all shown on the sketch below.

In the discussion on how this problem can be solved, only a rectangular channel will be considered, but from previous considerations of nonrectangular channels, it should be apparent as to what is needed to extend the methodology to trapezoidal and circular channels, etc. Furthermore, only the case in which the unit flow rate q is changed, will be considered, as the equations are given, i.e. it will be assumed that dq/dt is known. When dealing only with the instant partial closure of a gate, as in Sections 6.12 and 6.13, there were seven unknown variables, requiring that seven equations be solved simultaneously. With the continued slow closure of the gate, there are four additional variables added to the problem; an additional depth and a velocity upstream from the gate and an additional depth and a velocity downstream from the gate. The variables that can be considered the 11 unknowns are: Y_{u2}, V_{u2}, w_u, Y_{ou}, V_{ou}, V_{od}, Y_{d2}, V_{d2}, Y_{3d}, V_{3d}, and w_d. Notice that, Y_{od} is not included in this list. The boundary condition at the gate will allow it to be eliminated from the above list because if dq/dt is given, then q will be known at any time, and $Y_{od} = q_d/V_{od}$, in which q_d is the flow rate past the gate and is given by $q_d = q_o - |dq/dt|t$ in which q_o is the unit flow rate past the gate immediately after the gate has been instantly partially closed. (If Y_{od} is added to the list of unknowns, then this could be added as an additional equation.)



The equations that allow for the solution of the above 11 unknown variables consist of

$$F_1 = (V_{o1} + w_{o1})Y_{o1} - (V_{u2} + w_u Y_{u2} = 0 \quad \text{(Continuity to moving observer)} \tag{6.59}$$

$$F_2 = \frac{Y_{o1}^2}{2} + \frac{(V_{o1} + w_u)Y_{o1}}{g} - \frac{Y_{u2}^2}{2} - \frac{(V_{u2} + w_u)Y_{u2}}{g} = 0 \quad \text{(Momentum to moving observer)} \tag{6.60}$$

$$F_3 = V_{u2} - 2\sqrt{gY_{u2}} - V_{uo} + 2\sqrt{gY_{uo}} = 0 \quad (C^- \text{–characteristic equation}) \tag{6.61}$$

$$F_4 = V_{uo}Y_{uo} - V_{do}Y_{do} = V_{uo}Y_{uo} - q_d = 0 \quad \text{(Continuity equation, moving observer)} \tag{6.62}$$

$$F_5 = Y_{uo} + \frac{V_{uo}^2}{2g} - Y_{do} - \frac{V_{do}^2}{2g} = 0 \quad \text{(Energy equation, moving observer)} \tag{6.63}$$

$$F_6 = V_{d2} - 2\sqrt{gY_{d2}} - V_{do} - 2\sqrt{gY_{do}} = 0 \quad (C^- \text{–characteristic equation}) \tag{6.64}$$

$$F_7 = (V_{d2} - w_d)Y_{d2} - (V_{d3} - w_d)Y_{d3} = 0 \quad (\text{Contiunity equation, moving observer}) \tag{6.65}$$

$$F_8 = \frac{Y_{d2}^2}{2} + \frac{(V_{d2} + w_d)^2 Y_{d2}}{g} - \frac{Y_{d3}^2}{2} - \frac{(V_{d3} + w_d)^2 Y_{d3}}{g} = 0 \quad (\text{Momentum equation, moving observer}) \tag{6.66}$$

$$F_9 = V_{d3} - 2\sqrt{gY_{d3}} - V_{do} - 2\sqrt{gY_{do}} = 0 \quad (C^- \text{–characteristic equation}) \tag{6.67}$$

$$F_{10} = t - \frac{|q_o - V_{d2}Y_{d2}|}{|dq/dt|} - \frac{X_{sd}}{3\sqrt{gY_{d2}} + V_{org,d} - 2\sqrt{gY_{org,d}}} = (\text{Equation } t = t_1 + \Delta t) \tag{6.68}$$

$$F_{11} = t - \frac{|q_o - V_{u2}Y_{u2}|}{|dq/dt|} - \frac{X_{su}}{3\sqrt{gY_{u2}} + V_{org,d} - 2\sqrt{gY_{org,u}}} = (\text{Equation } t = t_1 + \Delta t) \tag{6.69}$$

In Equations 6.68 and 6.69, the depths with the added subscript org denote the downstream and the upstream depths at the gate, respectively, determined immediately after the instant partial closure of the gate.

The Program GATEDWU.FOR, whose listing is given below, is designed to solve problems in which a gate is initially instantly partially closed with uniform flows both upstream and downstream from the gate prior to this closure, and then thereafter dq/dt is specified, as the gate continues to close slowly. Before solving the above 11 equations for the 11 unknowns, this program solve the six equations for the conditions resulting from the instant gate closure, as described in the previous sections.

Program GATEDWU.FOR

```
      CHARACTER*2 UNK(6)/'Y1','V1','Y3','V3','Wu','Wd'/
      INTEGER*2 INDX(11)
      REAL F(11),FF(11),D(11,11)
      COMMON Y2,X(11),Yo1,Vo1,Yo2,Vo2,G,G2,Yo1S,Vo2S,Yo2S,TIME,qo
     &,dqdT,qd,Yod,XSU,XSD,V2C,VO2CU,VO2CD,INIT
      INIT=0
      WRITE(*,*)' Give: g,Yo1,Vo1,Yo2,Y2,IOUT'
      READ(*,*) G,Yo1,Vo1,Yo2,Y2,IOUT
      Vo2=Vo1*Yo1/Yo2
      WRITE(*,100) UNK
100   FORMAT(' Give guesses for (initialcondition):',/,5(A3,','),A3)
      READ(*,*) (X(I),I=1,6)
      G2=2.*G
      Yo1S=.5*Yo1**2
      Vo2S=Vo2-2.*SQRT(G*Yo2)
      Yo2S=.5*Y2**2
      V2C=Vo1-2.*SQRT(G*Yo1)
      NCT=0
12    SUM=0.
      CALL FUN(F)
      DO 20 J=1,6
      XT=X(J)
      X(J)=1.005*X(J)
```

```
      CALL FUN(FF)
      DO 15 I=1,6
15    D(I,J)=(FF(I)-F(I))/(X(J)-XT)
20    X(J)=XT
      CALL SOLVEQ(6,1,11,D,F,1,DD,INDX)
      NCT=NCT+1
      DO 30 I=1,6
      X(I)=X(I)-F(I)
30    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,(X(I),I=1,6)
110   FORMAT(' NCT=',I3,', SUM=',E12.5,/11F9.3)
      IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 12
      V2=X(2)*X(1)/Y2
      WRITE(*,120)(UNK(I),X(I),I=1,6),' Y2',Y2,' V2',V2
      IF(IOUT.NE.6) THEN
      WRITE(IOUT,119)
119   FORMAT(' Solution to conditions immediately after dropping
     &gate')
      WRITE(IOUT,120)(UNK(I),X(I),I=1,6),' Y2',Y2,' V2',V2
120   FORMAT(A3,' = ',F10.3)
      ENDIF
      VO2CU=X(2)-2.*SQRT(G*X(1))
      VO2CD=X(4)-2.*SQRT(G*X(3))
      WRITE(*,*)' Give: dq/dt, Delt & Number of steps'
      READ(*,*) dqdt,DELT,NUM
C New order of unknowns in array X for time simulation when
  dY/dt is given
C 1=Yu2,2=Vu2,3=wu,4=You,5=Vou,6=Vod,7=Yd2,8=Vd2,9=Y3d,10=V3d &
  11=wd
      qo=X(1)*ABS(X(2))
      X(11)=X(6)
      X(10)=X(4)
      X(9)=X(3)
      X(8)=V2
      X(7)=Y2
      X(4)=X(1)
      X(6)=V2
      X(3)=X(5)
      X(5)=X(2)
      X(4)=X(1)
      INIT=1
      XSUo=0.
      XSDo=0.
      WRITE(IOUT,129)
129   FORMAT(118('-'),/,' t    Yu2  Vu2  wu  You  Vou Yod  Vod Yd2
     &Vd2 Y3d  V3d wd  Xsu Xsd',/,118('-'))
      DO 50 K=1,NUM
      TIME=DELT*FLOAT(K)
      qd=qo-dqdt*TIME
      NCT=0
52    SUM=0.
```

```
      XSU=XSUo+DELT*X(3)
      XSD=XSDo+DELT*X(11)
      CALL FUN(F)
      DO 53 J=1,11
      XT=X(J)
      X(J)=1.005*X(J)
      CALL FUN(FF)
      DO 55 I=1,11
55    D(I,J)=(FF(I)-F(I))/(X(J)-XT)
53    X(J)=XT
      CALL SOLVEQ(11,1,11,D,F,1,DD,INDX)
      NCT=NCT+1
      DO 56 I=1,11
      X(I)=X(I)-F(I)
56    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,X
      IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 52
      XSUo=XSUo+DELT*X(3)
      XSDo=XSDo+DELT*X(11)
50    WRITE(IOUT,130) IFIX(TIME),(X(I),I=1,5),qD/X(6),(X(I),I=6,11),
     &XSU,XSD
130   FORMAT(I4,12F8.3,2F9.1)
      END
      SUBROUTINE FUN(F)
      REAL F(11)
      COMMON Y2,X(11),Yo1,Vo1,Yo2,Vo2,G,G2,Yo1S,Vo2S,Yo2S,TIME,qo,
     &dqdT,qd,Yod,XSU,XSD,V2C,VO2CU,VO2CD,INIT
      IF(INIT.GT.0) GO TO 10
      V2=X(2)*X(1)/Y2
      F(1)=X(1)+X(2)**2/G2-Y2-V2**2/G2
      F(2)=(X(5)+Vo1)*Yo1-(X(5)+X(2))*X(1)
      F(3)=Yo1S+(X(5)+Vo1)**2*Yo1/G-.5*X(1)**2-(X(5)+X(2))
     &**2*X(1)/G
      F(4)=(V2-X(6))*Y2-(X(4)-X(6))*X(3)
      F(5)=.5*X(3)**2+(X(4)-X(6))**2*X(3)/G-Yo2S-(V2-X(6))**2*Y2/G
      F(6)=X(4)-2.*SQRT(G*X(3))-Vo2S
      RETURN
10    F(1)=(Vo1+X(3))*Yo1-(X(2)+X(3))*X(1)
      F(2)=.5*(Yo1**2-X(1)**2)+((Vo1+X(3))**2*Yo1-(X(2)+X(3))
     &**2*X(1))/G
      F(3)=X(2)-X(5)-2.*(SQRT(G*X(1))-SQRT(G*X(4)))
      F(4)=X(4)*X(5)-qd
      F(5)=X(4)+X(5)**2/G2-qd/X(6)-X(6)**2/G2
      F(6)=X(8)-X(6)+2.*(SQRT(G*qd/X(6))-SQRT(G*X(7)))
      F(7)=(X(8)-X(11))*X(7)-(X(10)-X(11))*X(9)
      F(8)=.5*(X(7)**2-X(9)**2)+((X(8)-X(11))**2*X(7)-
     &(X(10)-X(11))**2*X(9))/G
      F(9)=X(10)-2.*SQRT(G*X(9))-Vo2S
      F(10)=TIME-ABS(qo-X(7)*X(8))/dqdt-XSD/(3.*SQRT(G*X(7))-VO2CD)
      F(11)=TIME-ABS(qo-X(1)*X(2))/dqdt-XSU/(3.*SQRT(G*X(1))-VO2CU)
      END
```

EXAMPLE PROBLEM 6.24

Uniform flow exists upstream and downstream from a gate in a rectangular channel. The upstream uniform depth and velocity are: Y_{o1} = 6 ft, and V_{o1} = 9 fps. The downstream uniform depth is: Y_{o2} = 3.5 ft. Suddenly, the gate is partially closed so that it produces a depth of 2 ft immediately downstream from it. First, determine the depth and the velocity immediately upstream from the gate, and the velocity of the hydraulic bore that the gate's instant closure causes. Also, determine the velocity of the hydraulic bore downstream from the gate and the depth and the velocity just downstream from this moving wave. Second, determine the conditions upstream and downstream from the gate over a 400 s time period, using 20 s time steps if the gate is closed further so that the change in unit flow rate after the initial instant closure is dq/dt = 0.01 $(ft/s)^2$.

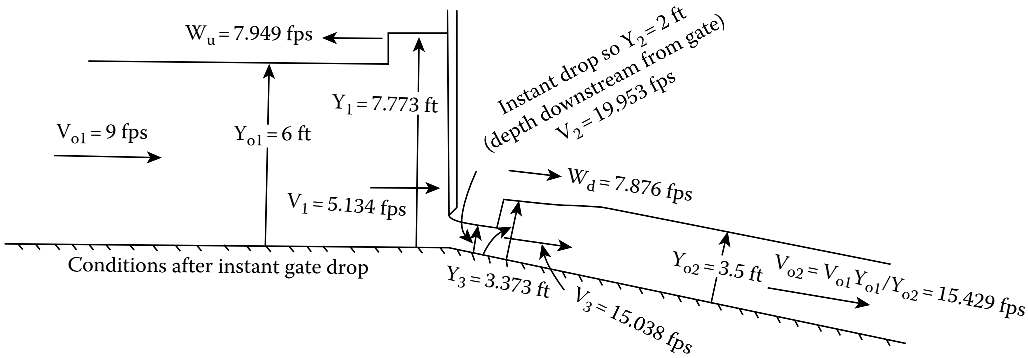
Solution

Program GATEDWU is designed to solve problems of this type. The input to solve the problem consists of the following:
In response to the prompt: Give: g, Y_{o1}, V_{o1}, Y_{o2}, Y_2, I_{OUT}

```
32.2 6 9 3.5 2 3
```

In response to the prompt: Give guesses for (initial condition): Y_1, V_1, Y_3, V_3, W_u, W_d

```
8 5 3.3 15.5 8.1 8.2
```



The solution output consists of

Solution to conditions immediately after dropping gate

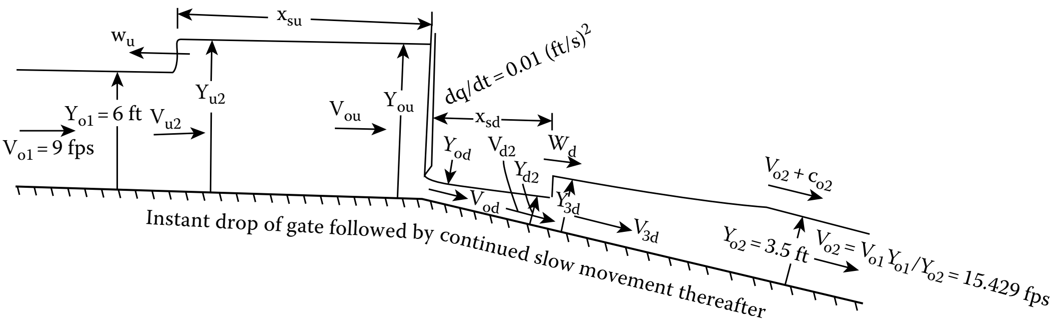
Y1 =	7.773	Wu =	7.949
V1 =	5.134	Wd =	7.876
Y3 =	3.373	Y2 =	2.000
V3 =	15.038	V2 =	19.953

And finally in response to the prompt: Give: dq/dt, Delt and Number of steps

```
.01    20    20
```

The solution output for the second part of the problem when dq/dt decreases by 0.01 $(ft/sec)^2$ consists of: Solution when gate continues to close so dq/dt = .01 $(ft/sec)^2$

t	Y_{u2}	V_{u2}	w_u	Y_{ou}	V_{ou}	Y_{od}	V_{od}	Y_{d2}	V_{d2}	Y_{3d}	V_{3d}	w_d	X_{su}	X_{sd}
0	7.773	5.134	7.949	7.773	5.134	2.000	19.953	2.000	19.953	3.373	15.038	7.876	0.0	0.0
20	7.757	5.168	7.921	7.739	5.131	1.995	19.905	2.007	19.955	3.378	15.054	7.877	158.4	157.5
40	7.740	5.202	7.893	7.704	5.128	1.990	19.857	2.014	19.957	3.383	15.070	7.877	316.3	315.1
60	7.724	5.236	7.865	7.669	5.125	1.984	19.809	2.022	19.959	3.388	15.086	7.877	473.6	472.6
80	7.707	5.271	7.837	7.635	5.122	1.979	19.760	2.029	19.960	3.393	15.102	7.877	630.3	630.2
100	7.690	5.305	7.808	7.600	5.119	1.974	19.711	2.036	19.961	3.398	15.118	7.877	786.5	787.7
120	7.674	5.340	7.780	7.565	5.117	1.969	19.662	2.044	19.962	3.403	15.133	7.877	942.1	945.2
140	7.657	5.375	7.751	7.530	5.114	1.963	19.612	2.051	19.963	3.408	15.149	7.876	1097.1	1102.8
160	7.640	5.411	7.722	7.495	5.111	1.958	19.562	2.058	19.963	3.413	15.164	7.876	1251.5	1260.3
180	7.623	5.446	7.693	7.459	5.109	1.953	19.512	2.066	19.963	3.418	15.180	7.875	1405.4	1417.8
200	7.606	5.482	7.664	7.424	5.106	1.948	19.461	2.073	19.963	3.423	15.195	7.874	1558.7	1575.3
220	7.588	5.518	7.634	7.389	5.103	1.943	19.410	2.081	19.963	3.428	15.210	7.874	1711.4	1732.8
240	7.571	5.554	7.604	7.353	5.101	1.937	19.359	2.088	19.962	3.433	15.226	7.872	1863.4	1890.2
260	7.554	5.591	7.575	7.317	5.099	1.932	19.307	2.096	19.961	3.438	15.241	7.871	2014.9	2047.6
280	7.536	5.628	7.545	7.281	5.096	1.927	19.256	2.103	19.960	3.443	15.256	7.870	2165.8	2205.0
300	7.518	5.665	7.514	7.245	5.094	1.922	19.203	2.111	19.959	3.448	15.270	7.868	2316.1	2362.4
320	7.501	5.702	7.484	7.209	5.092	1.917	19.151	2.119	19.957	3.453	15.285	7.866	2465.8	2519.7
340	7.483	5.740	7.453	7.173	5.090	1.912	19.098	2.126	19.956	3.458	15.300	7.864	2614.9	2677.0
360	7.465	5.778	7.423	7.136	5.088	1.906	19.044	2.134	19.953	3.463	15.315	7.862	2763.3	2834.2
380	7.447	5.816	7.392	7.100	5.086	1.901	18.990	2.142	19.951	3.467	15.329	7.860	2911.1	2991.4
400	7.428	5.854	7.360	7.063	5.084	1.896	18.936	2.150	19.948	3.472	15.344	7.858	3058.3	3148.6



6.14.2 Nonrectangular Channels

This section discusses how to solve conditions upstream and downstream from a gate in nonrectangular channels when the gate is partially instantly closed, and then when this closure is followed by a slow movement thereafter. The variables that will be solved are the same as those given in the previous section for rectangular channels. In other words, the unknowns will be the following 11 variables: Y_{u2}, V_{u2}, w_u, Y_{ou}, V_{ou}, V_{od}, Y_{d2}, V_{d2}, Y_{3d}, V_{3d}, and w_d.

The 11 equations that allow for the solution of these 11 variables are as given below for a trapezoidal channel. There will be some differences in the equations when dealing with circular, or other, cross-sections. These equations are given in the same order as those for a rectangular channel. In order to distinguish the stage variable w from the velocities of the hydraulic bore that moves upstream and downstream, ws will be used to denote the stage variable with a subscript to indicate the position along the channel where this stage variable is to be evaluated from the depth Y with the same subscript. (In other words, ws takes the place of w used earlier, but only in this section.)

$$F_1 = (V_{o1} + w_u)A_{o1} - (V_{u2} + w_u)A_{u2} = 0 \quad \text{(Continuity equation, moving observer)} \qquad (6.70)$$

$$F_2 = \frac{by_{o1}^2}{2} + \frac{my_{o1}^3}{3} + \frac{(V_{o1} + w_u)^2 A_{o1}}{g} - \frac{by_{uz}^2}{2} - \frac{my_{u2}^3}{3} - \frac{(V_{u3} + w_u)^2 A_{u2}}{g} = 0 \quad \text{(Mom., mov. obs.)} \; (6.71)$$

$$F_3 = V_{u2} - ws_{u2} - V_{uo} + ws_{uo} = 0 \quad (C^- \text{–characteristic equation}) \qquad (6.72)$$

$$F_4 = V_{uo}A_{uo} - V_{do}A_{do} = V_{uo}A_{uo} - Q_d = 0 \quad \text{(Continuity across gate)}$$

$$\text{in which } A_{do} = \frac{Q_d}{V_{do}} \quad \text{and} \quad Y_{do} = \frac{(b^2 + 4mA_{do})^{1/2} - b}{2m} \qquad (6.73)$$

$$F_5 = Y_{uo} + \frac{V_{uo}^2}{2g} - Y_{do} - \frac{V_{do}^2}{2g} = 0 \quad \text{(Energy across gate)} \qquad (6.74)$$

$$F_6 = V_{d2} - w_{d2} - V_{do} - w_{do} = 0 \quad (C^- \text{–characteristic equation}) \qquad (6.75)$$

$$F_7 = (V_{d2} - w_d)A_{d2} - (V_{d3} - w_d)A_{d3} = 0 \quad \text{(Continuity equation, moving observer)} \tag{6.76}$$

$$F_8 = \frac{by_{d2}^2}{2} + \frac{my_{d2}^3}{3} + \frac{(V_{d2} - w_d)^2 A_{d2}}{g} - \frac{by_{d3}^2}{2} - \frac{my_{d3}^3}{3} - \frac{(V_{d3} - w_d)^2 A_{d3}}{g} = 0 \quad \text{(Mom., mov. obs.)} \tag{6.77}$$

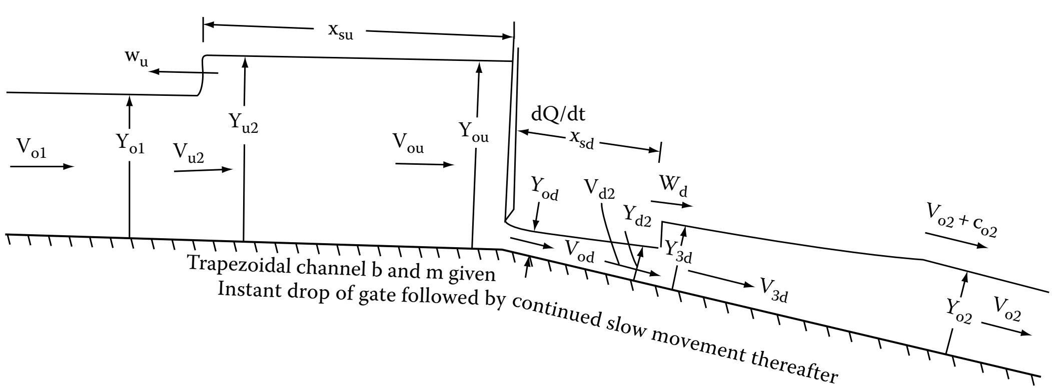
$$F_9 = V_{d3} - ws_{d3} - V_{o2} - ws_{o2} = 0 \quad (C^- - \text{characteristic equation}) \tag{6.78}$$

$$F_{10} = t - \frac{|Q_o - V_{d2}A_{d2}|}{|dQ/dt|} - \frac{x_{sd}}{ws_{d2} + c_{d2} + V_{orgd} - ws_{orgd}} = 0 \quad (\text{Equation } t = t_1 + \Delta t) \tag{6.79}$$

$$F_{11} = t - \frac{|Q_o - V_{u2}A_{u2}|}{|dQ/dt|} - \frac{x_{su}}{ws_{u2} + c_{u2} + V_{orgu} - ws_{orgu}} = (\text{Equation } t = t_1 + \Delta t) \tag{6.80}$$

As in the equations for a rectangular channel, the additional subscript org indicates that these quantities are to be evaluated from the solution obtained the instant after the gate has been dropped to the new position. The equation below Equation 6.74 indicates that since Y_{od} is not included in the list of unknown variables, it must be evaluated by first dividing the velocity into the flow rate Q_d that is passing the gate at this time by the velocity V_{od} here to get the area A_{od}, and thereafter Y_{od} is evaluated by the quadratic formula that relates area to the depth.

The solution procedure for problems in nonrectangular channels follows much the same steps as needed to solve problems in rectangular channels. The differences are that areas are involved in the equations rather than just the depths (which are in the list of unknowns), and therefore these areas will need to be evaluated. Also, since the stage variable is used rather than the celerity in the C^-—characteristic equations, a table of stage variables ws will need to be produced so that a table lookup can be used to evaluate the needed stage variable associated with the depth. In generating this table of stage variable values versus the depth Y, it is necessary that the last entry for depth Y be larger than any Y that will be encountered during the solution. Therefore, the table will need two entries for depth larger than Y_{o1}. As with the rectangular channel, the solution entails two separate solutions: first, the newly created depth immediately after the gate in instantly partially closed, and second, the above problem requiring the solution of 11 equations for 11 unknowns.



The Program GATETRDU.FOR is designed to solve problems in which the change inflow rate dQ/dt is given for problems in trapezoidal channels, after the position of the gate is instantly changed. This program first solves Equations 6.52 through 6.58 for seven unknowns, just like Program GATETR.FOR. Unlike GATETR, however, which only needs to employ the table lookup technique to evaluate stage variables for downstream depths less than Y_{o2}, it is necessary that the table of stage variables be for values larger than Y_{o1}, and therefore values are generated for depths up to $(3/2)Y_{o1}$. Then, it takes this solution for seven unknowns and places the values in the new positions of X( ) for the expanded problem in which 11 variables are unknown. Thereafter, the Newton method is utilized to solve the 11 equations for the new 11 unknowns. Note, the integer INIT is first given the value of 0 to have the subroutine FUN evaluate the seven equations, and thereafter INIT=1 to tell the subroutine FUN to evaluate the 11 equations.

Program GATETRDU.FOR

```
C This program handles: First when a gate in instantly partially
C closed in a C trapezoidal channel, and second thereafter the
C flow rate dQ/dt is given.
C This 1st portion of the program solves 7 eqs, as does GATETR,
C and for the C 2nd portion solves 11 variables in which theory of
C Characteristics is used to C handle the upstream and downstream
C unsteady flows. For the 1st problem the C  unknowns are: 1=Y1,
C 2=V1, 3=V2, 4=Y3, 5=V3, 6=Wu, and 7=Wd
C For the 2nd problem the unknowns are:
C 1=Yu2,2=Vu2,3=wu,4=You,5=Vou,6=Vod,7=Yd2,8=Vd2,9=Y3d,10=V3d,11=wd
      PARAMETER (N=200)
      CHARACTER*2 UNK(7)/'Y1','V1','V2','Y3','V3','Wu','Wd'/
      INTEGER*2 INDX(11)
      REAL F(11),FF(11),D(11,11)
      COMMON B,FM,BH,FM3,Y2,W(N),X(11),DY,VOW,Yo1,Vo1,G,G2,dQdt,
     &TIME,BS,FM2,FM4,Ao1,Ao2,Ao1G,A2,A2G,FMON1,FMON2,Qd,VOWU,
     &VOWD,Qo,XSU,XSD,INIT
      FW(Y)=SQRT(G*(B+FM2*Y)/((B+FM*Y)*Y))
      INIT=0
      WRITE(*,*)' Give: g,b,m,Yo1,Vo1,IOUT'
      READ(*,*) G,B,FM,Yo1,Vo1,IOUT
      IOU1=IOUT+1
      BS=B*B
      FM4=4.*FM
      G2=2.*G
      Aunif=(B+FM*Yo1)*Yo1
      Qunif=Aunif*Vo1
      Eo1=Yo1+Vo1**2/G2
      Yo2=Yo1/2.5
      NCT=0
2     FE=Eo1-Yo2-(Qunif/((B+FM*Yo2)*Yo2))**2/G2
      NCT=NCT+1
      IF(MOD(NCT,2).EQ.0) GO TO 3
      FEE=FE
      Yo2=Yo2+.01
      GO TO 2
3     DIF=FEE/(100.*(FE-FEE))
```

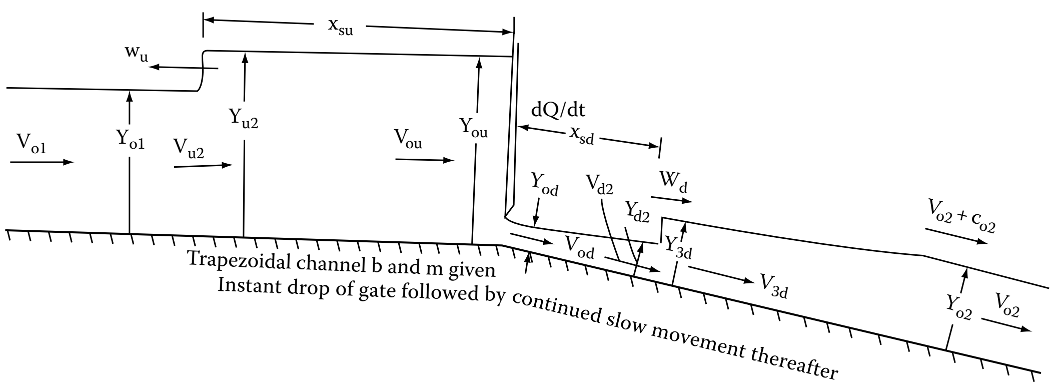
```
      Yo2=Yo2-.01-DIF
      IF(NCT.LT.30 .AND. ABS(DIF).GT.1.E-5) GO TO 2
      Ao2=(B+FM*Yo2)*Yo2
      Vo2=Qunif/Ao2
      WRITE(IOUT,212) Yo1,Vo1,Aunif,Qunif,Yo2,Vo2,Ao2
212   FORMAT(' Uniform upstream and downstream conditions',/,'
     &Yo1=',F7.3,' Vo1=',F7.3,' Ao1=',F7.2,' Qunif=',F8.2,' Yo2=',
     &F7.3,' Vo2=',F7.3,' Ao2=',F8.2)
      WRITE(*,213) Qunif,Yo2,Vo2
213   FORMAT(' Qunif=',F9.2,' Yo2=',F8.3,' Vo2=',F8.3)
      WRITE(*,100) UNK
100   FORMAT(' Give Y2 & guesses for:',6(A3,','),A3)
      READ(*,*) Y2,(X(I),I=1,7)
      N23=2*N/3
      DY=(Yo1-.01)/FLOAT(N23)
      FM2=2.*FM
      FM3=FM/3.
      BH=B/2.
      Ao1=(B+FM*Yo1)*Yo1
      Ao1G=Ao1/G
      A2=(B+FM*Y2)*Y2
      A2G=A2/G
      FMON1=(BH+FM3*Yo1)*Yo1**2
      FMON2=(BH+FM3*Y2)*Y2*Y2
      DY2=DY/10.
      DYH=DY2/2.
      Y=.01
      WF1=FW(Y)
      W(1)=0.
      DO 10 I=2,N
      SUM=0.
      DO 5 J=1,10
      WF=FW(Y+DY2*FLOAT(J))
      SUM=SUM+DYH*(WF1+WF)
5     WF1=WF
      W(I)=W(I-1)+SUM
10    Y=Y+DY
      WRITE(IOUT,338) Y
338   FORMAT(' Largest depth for stage variable look-up
     &table=',F9.2)
      WRITE(*,338) Y
      IM=(Yo2-.01)/DY+1.
      FAC=(Yo2-.01-DY*FLOAT(IM-1))/DY
      VOW=Vo2-W(IM)+FAC*(W(IM+1)-W(IM))
      NCT=0
12    SUM=0.
      CALL FUN(F)
      DO 20 J=1,7
      XT=X(J)
      X(J)=1.005*X(J)
      CALL FUN(FF)
```

```
      DO 15 I=1,7
15    D(I,J)=(FF(I)-F(I))/(X(J)-XT)
20    X(J)=XT
      CALL SOLVEQ(7,1,11,D,F,1,DD,INDX)
      NCT=NCT+1
      DO 30 I=1,7
      X(I)=X(I)-F(I)
30    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,(X(I),I=1,7)
110   FORMAT(' NCT=',I3,', SUM=',E12.5,/11F9.3)
      IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 12
      IF(IOUT.NE.6) THEN
      WRITE(IOUT,119)
119   FORMAT(' Solution to conditions immediately after dropping
     &gate')
      WRITE(IOUT,120)(UNK(I),X(I),I=1,7)
      ENDIF
      WRITE(*,120)(UNK(I),X(I),I=1,7)
120   FORMAT(A3,' = ',F10.3)
      INIT=1
      IM=(X(1)-.01)/DY+1.
      IF(IM.GT.199) IM=199
      FAC=(X(1)-.01-DY*FLOAT(IM-1))/DY
      VOWU=X(2)-W(IM)+FAC*(W(IM+1)-W(IM))
      IM=(X(4)-.01)/DY+1.
      FAC=(X(4)-.01-DY*FLOAT(IM-1))/DY
      VOWD=X(5)-W(IM)+FAC*(W(IM+1)-W(IM))
      Qo=(B+FM*X(1))*X(1)*ABS(X(2))
      WRITE(IOUT,121) Qo
121   FORMAT(' The flow rate past gate after instant drop,
     &Qo=',F9.3)
      WRITE(*,*)' Give: dQ/dt, Delt & Number of steps'
      READ(*,*) dQdt,DELT,NUM
C New order of unknowns in array X for time simulation when dY/dt
  is given
C 1=Yu2,2=Vu2,3=wu,4=You,5=Vou,6=Vod,7=Yd2,8=Vd2,9=Y3d,10=V3d &
  11=wd
      X(11)=X(7)
      X(10)=X(5)
      X(9)=X(4)
      X(8)=X(3)
      X(7)=Y2
      X(4)=X(1)
      VWAVE=X(6)
      X(6)=X(3)
      X(3)=VWAVE
      X(5)=X(2)
      X(4)=X(1)
      XSUo=0.
      XSDo=0.
```

```
      WRITE(IOUT,129)
129   FORMAT(118('-'),/,' t   Yu2  Vu2  wu  You  Vou Yod  Vod  Yd2
     &Vd2 Y3d  V3d wd  XsuXsd',/,118('-'))
      WRITE(IOU1,128)
128   FORMAT(84('-'),/,' t Au2 Qu2 Aou Qou Aod Qod Ad2 Qd2 A3d
     &Q3d',/,84('-'))
      Yd2=(SQRT(BS+FM4*Qo/X(6))-B)/FM2
      WRITE(IOUT,130) 0,(X(I),I=1,5),Yd2,(X(I),I=6,11),XSUo,XSDo
      Au2=(B+FM*X(1))*X(1)
      Qu2=Au2*X(2)
      Aou=(B+FM*X(4))*X(4)
      Qou=Aou*X(5)
      Aod=(B+FM*Yd2)*Yd2
      Qod=Aod*X(6)
      Ad2=(B+FM*X(7))*X(7)
      Qd2=Ad2*X(8)
      A3d=(B+FM*X(9))*X(9)
      Q3d=A3d*X(10)
      WRITE(IOU1,131) 0,Au2,Qu2,Aou,Qou,Aod,Qod,Ad2,Qd2,A3d,Q3d
      DO 50 K=1,NUM
      TIME=DELT*FLOAT(K)
      Qd=Qo-dQdt*TIME
      NCT=0
52    SUM=0.
      XSU=XSUo+DELT*X(3)
      XSD=XSDo+DELT*X(11)
      CALL FUN(F)
      DO 53 J=1,11
      XT=X(J)
      X(J)=1.005*X(J)
      CALL FUN(FF)
      DO 55 I=1,11
55    D(I,J)=(FF(I)-F(I))/(X(J)-XT)
53    X(J)=XT
      CALL SOLVEQ(11,1,11,D,F,1,DD,INDX)
      NCT=NCT+1
      DO 56 I=1,11
      X(I)=X(I)-F(I)
56    SUM=SUM+ABS(F(I))
      WRITE(*,110) NCT,SUM,X
      IF(NCT.LT.30 .AND. SUM.GT.1.E-5) GO TO 52
      XSUo=XSUo+DELT*X(3)
      XSDo=XSDo+DELT*X(11)
      Yd2=(SQRT(BS+FM4*Qd/X(6))-B)/FM2
      WRITE(IOUT,130) IFIX(TIME),(X(I),I=1,5),Yd2,(X(I),I=6,11),
     &XSU,XSD
130   FORMAT(I4,12F8.3,2F9.1)
      Au2=(B+FM*X(1))*X(1)
      Qu2=Au2*X(2)
      Aou=(B+FM*X(4))*X(4)
```

```
      Qou=Aou*X(5)
      Aod=(B+FM*Yd2)*Yd2
      Qod=Aod*X(6)
      Ad2=(B+FM*X(7))*X(7)
      Qd2=Ad2*X(8)
      A3d=(B+FM*X(9))*X(9)
      Q3d=A3d*X(10)
50    WRITE(IOU1,131) IFIX(TIME),Au2,Qu2,Aou,Qou,Aod,Qod,Ad2,Qd2,
     &A3d,Q3d
131   FORMAT(I4,10F8.2)
      END
      SUBROUTINE FUN(F)
      PARAMETER (N=200)
      REAL F(11)
      COMMON B,FM,BH,FM3,Y2,W(N),X(11),DY,VOW,Yo1,Vo1,G,G2,dQdt,
     &TIME,BS,FM2,FM4,Ao1,Ao2,Ao1G,A2,A2G,FMON1,FMON2,Qd,VOWU,
     &VOWD,Qo,XSU,XSD,INIT
C  1=Y1,2=V1,3=V2,4=Y3,5=V3,6=Wu,and 7=Wd
      IF(INIT.GT.0) GO TO 10
      F(1)=X(1)+X(2)**2/G2-Y2-X(3)**2/G2
      A1=(B+FM*X(1))*X(1)
      A3=(B+FM*X(4))*X(4)
      F(2)=X(2)*A1-X(3)*A2
      F(3)=FMON1+(X(6)+Vo1)**2*Ao1G-(BH+FM3*X(1))*X(1)**
     &2-(X(6)+X(2))**2*A1/G
      F(4)=(X(6)+Vo1)*Ao1-(X(6)+X(2))*A1
      F(5)=FMON2+(X(7)-X(3))**2*A2G-(BH+FM3*X(4))*X(4)**2-
     &(X(7)-X(5))**2*A3/G
      F(6)=(X(7)-X(3))*A2-(X(7)-X(5))*A3
      IM=(X(4)-.01)/DY+1.
      IF(IM.GT.199) IM=199
      FAC=(X(4)-.01-DY*FLOAT(IM-1))/DY
      F(7)=X(5)-(W(IM)+FAC*(W(IM+1)-W(IM)))-VOW
      RETURN
C 1=Yu2,2=Vu2,3=wu,4=You,5=Vou,6=Vod,7=Yd2,8=Vd2,9=Y3d,10=V3d,11=wd
10    Au2=(B+FM*X(1))*X(1)
      F(1)=(Vo1+X(3))*Ao1-(X(2)+X(3))*Au2
      F(2)=BH*(Yo1**2-X(1)**2)+FM3*(Yo1**3-X(1)**3)+((Vo1+X(3))**
     &2*Ao1-(X(2)+X(3))**2*Au2)/G
      IM=(X(1)-.01)/DY+1.
      IF(IM.GT.199) IM=199
      FAC=(X(1)-.01-DY*FLOAT(IM-1))/DY
      Wu2=W(IM)+FAC*(W(IM+1)-W(IM))
      IM=(X(4)-.01)/DY+1.
      IF(IM.GT.199) IM=199
      FAC=(X(4)-.01-DY*FLOAT(IM-1))/DY
      Wou=W(IM)+FAC*(W(IM+1)-W(IM))
      F(3)=X(2)-X(5)-Wu2+Wou
      Aou=(B+FM*X(4))*X(4)
      F(4)=Aou*X(5)-Qd
```

```
      Aod=Qd/X(6)
      Yod=(SQRT(BS+FM4*Aod)-B)/FM2
      F(5)=X(4)-Yod+(X(5)**2-X(6)**2)/G2
      IM=(X(7)-.01)/DY+1.
      IF(IM.GT.199) IM=199
      FAC=(X(7)-.01-DY*FLOAT(IM-1))/DY
      Wd2=W(IM)+FAC*(W(IM+1)-W(IM))
      IM=(Yod-.01)/DY+1.
      IF(IM.GT.199) IM=199
      FAC=(Yod-.01-DY*FLOAT(IM-1))/DY
      Wod=W(IM)+FAC*(W(IM+1)-W(IM))
      F(6)=X(8)-X(6)+Wod-Wd2
      Ad2=(B+FM*X(7))*X(7)
      A3d=(B+FM*X(9))*X(9)
      F(7)=(X(8)-X(11))*Ad2-(X(10)-X(11))*A3d
      F(8)=BH*(X(7)**2-X(9)**2)+FM3*(X(7)**3-X(9)**3)+
     &((X(8)-X(11))**2*Ad2-(X(10)-X(11))**2*A3d)/G
      IM=(X(9)-.01)/DY+1.
      IF(IM.GT.199) IM=199
      FAC=(X(9)-.01-DY*FLOAT(IM-1))/DY
      W3d=W(IM)+FAC*(W(IM+1)-W(IM))
      F(9)=X(10)-W3d-VOW
      cd=X(8)/SQRT(G*Ad2/(B+FM2*X(7)))
      cu=X(2)/SQRT(G*Au2/(B+FM2*X(1)))
      IM=(X(7)-.01)/DY+1.
      IF(IM.GT.199) IM=199
      FAC=(X(7)-.01-DY*FLOAT(IM-1))/DY
      Wd2=W(IM)+FAC*(W(IM+1)-W(IM))
      F(10)=TIME-ABS(Qo-Ad2*X(8))/dQdt-XSD/(Wd2-cd-VOWD)
      F(11)=TIME-ABS(Qo-Au2*X(2))/dQdt-XSU/(Wu2-cu-VOWU)
      END
```



EXAMPLE PROBLEM 6.25

A trapezoidal channel with a bottom width b = 12 ft, and a side slope m = 1.5 initially contains a uniform flow both upstream and downstream from a gate. (The bottom slopes of the channel are such that a uniform flow is possible both upstream and downstream from the gate for the flow rate that is occurring.) The uniform depth and the velocity upstream from the gate are: Y_{o1} = 5 ft and V_{o1} = 7 ft/s. The gate is first instantly closed so that it produces a depth of 2 ft immediately downstream from it. Thereafter, its position is adjusted slowly so that the rate of change inflow rate is decreased by dQ/dt = 0.2 cfs/s. Using 20 s time steps, solve the depths and velocities upstream and downstream from the gate over a 400 time period.

Solution

The input to the Program GATETRDU is as follows:
In response to the first prompt, give: g, b, m, Y_{o1}, V_{o1}, I_{OUT}

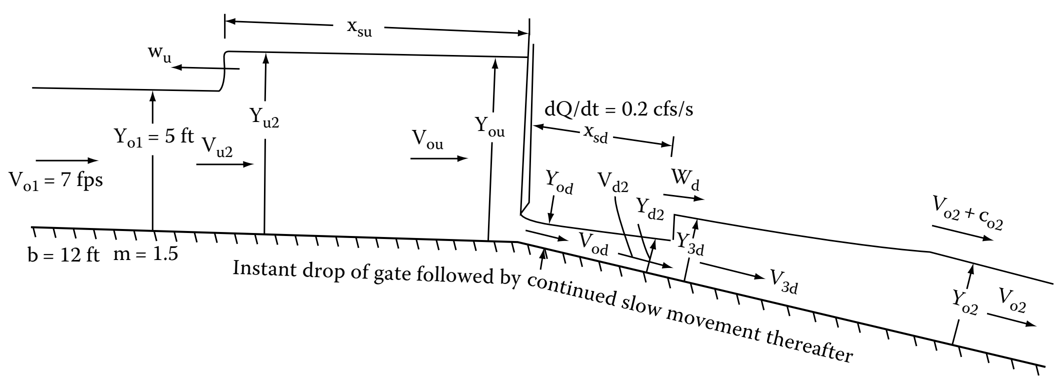
```
32.2 12 1.5 5 7 3
```

And to the prompt, give Y_2 and guesses for: Y_1, V_1, V_2, Y_3, V_3, W_u, W_d. The following is provided:

```
2 6 4 17 3 12.5 6 6
```

And finally to the prompt, give: dQ/dt, Delt and the number of steps for the following is given:

```
.2 20 20
```



The output from program GATETRDU is:

Uniform upstream and downstream conditions

```
Yo1= 5.000 Vo1= 7.000 Ao1= 97.50 Qunif= 682.50 Yo2= 3.146
Vo2= 12.978 Ao2= 52.59
```

Solution to conditions immediately after dropping gate

Y1 =	6.078
V1 =	3.896
V2 =	16.667
Y3 =	2.997
V3 =	12.591
Wu =	5.916
Wd =	6.302

The flow rate past gate after instant drop, Qo = 500.013

t	Y_{u2}	V_{u2}	w_u	Y_{ou}	V_{ou}	Y_{od}	V_{od}	Y_{d2}	V_{d2}	Y_{3d}	V_{3d}	w_d	X_{su}	X_{sd}
0	6.078	3.896	5.916	6.078	3.896	2.000	16.667	2.000	16.667	2.997	12.591	6.302	.0	.0
20	6.063	3.937	5.886	6.047	3.893	1.991	16.624	2.007	16.693	3.005	12.618	6.319	117.7	126.4
40	6.048	3.979	5.856	6.016	3.890	1.982	16.581	2.013	16.718	3.012	12.646	6.336	234.8	253.1
60	6.033	4.022	5.826	5.984	3.888	1.973	16.537	2.020	16.743	3.019	12.672	6.353	351.4	380.2
80	6.018	4.064	5.795	5.953	3.885	1.964	16.492	2.026	16.767	3.027	12.699	6.370	467.3	507.6
100	6.002	4.108	5.764	5.921	3.882	1.955	16.447	2.033	16.792	3.034	12.726	6.387	582.5	635.3
120	5.987	4.151	5.732	5.889	3.880	1.945	16.402	2.039	16.816	3.041	12.752	6.403	697.2	763.4
140	5.971	4.196	5.701	5.857	3.877	1.936	16.356	2.046	16.839	3.048	12.779	6.419	811.2	891.7
160	5.955	4.240	5.669	5.824	3.875	1.927	16.309	2.053	16.862	3.055	12.805	6.435	924.6	1020.4
180	5.939	4.285	5.637	5.791	3.873	1.918	16.262	2.059	16.884	3.062	12.831	6.450	1037.3	1149.4
200	5.923	4.331	5.604	5.758	3.871	1.909	16.214	2.066	16.907	3.069	12.856	6.466	1149.4	1278.8
220	5.906	4.377	5.571	5.725	3.869	1.900	16.166	2.073	16.928	3.076	12.882	6.481	1260.8	1408.4
240	5.890	4.424	5.538	5.692	3.867	1.890	16.117	2.079	16.950	3.083	12.907	6.496	1371.6	1538.3
260	5.873	4.471	5.504	5.658	3.865	1.881	16.067	2.086	16.970	3.090	12.932	6.510	1481.7	1668.5
280	5.856	4.519	5.470	5.624	3.864	1.872	16.016	2.093	16.991	3.097	12.957	6.525	1591.1	1799.0
300	5.839	4.568	5.436	5.589	3.862	1.863	15.965	2.100	17.010	3.104	12.982	6.539	1699.8	1929.8
320	5.822	4.617	5.401	5.555	3.861	1.854	15.913	2.106	17.030	3.110	13.006	6.552	1807.8	2060.8
340	5.804	4.666	5.366	5.520	3.860	1.845	15.861	2.113	17.048	3.117	13.031	6.566	1915.1	2192.1
360	5.786	4.717	5.331	5.484	3.859	1.835	15.808	2.120	17.067	3.124	13.055	6.579	2021.7	2323.7
380	5.768	4.768	5.295	5.449	3.858	1.826	15.753	2.127	17.084	3.130	13.079	6.592	2127.6	2455.5
400	5.750	4.820	5.258	5.413	3.857	1.817	15.698	2.134	17.101	3.137	13.103	6.604	2232.8	2587.6

(*continued*)

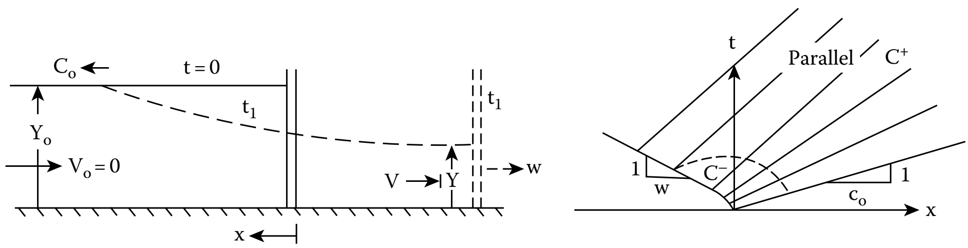
(continued)

t	A_{u2}	Q_{u2}	A_{ou}	Q_{ou}	A_{od}	Q_{od}	A_{d2}	Q_{d2}	A_{3d}	Q_{3d}
0	128.35	500.01	128.35	500.01	30.00	500.01	30.00	500.01	49.44	622.53
20	127.90	503.58	127.41	496.01	29.84	496.01	30.12	502.75	49.60	625.84
40	127.44	507.14	126.47	492.01	29.67	492.01	30.24	505.48	49.75	629.14
60	126.99	510.71	125.53	488.01	29.51	488.01	30.35	508.21	49.91	632.43
80	126.53	514.28	124.59	484.01	29.35	484.01	30.47	510.95	50.06	635.71
100	126.07	517.85	123.64	480.01	29.18	480.01	30.59	513.68	50.21	638.99
120	125.60	521.42	122.69	476.01	29.02	476.01	30.71	516.42	50.36	642.25
140	125.13	524.99	121.73	472.01	28.86	472.01	30.83	519.15	50.51	645.50
160	124.65	528.56	120.77	468.01	28.70	468.01	30.95	521.89	50.66	648.74
180	124.18	532.14	119.81	464.01	28.53	464.01	31.07	524.63	50.81	651.98
200	123.69	535.72	118.84	460.01	28.37	460.01	31.19	527.36	50.96	655.20
220	123.20	539.30	117.87	456.01	28.21	456.01	31.31	530.10	51.11	658.41
240	122.71	542.88	116.89	452.01	28.05	452.01	31.44	532.84	51.26	661.60
260	122.22	546.46	115.91	448.01	27.88	448.01	31.56	535.58	51.40	664.78
280	121.71	550.04	114.92	444.01	27.72	444.01	31.68	538.32	51.55	667.95
300	121.21	553.63	113.93	440.01	27.56	440.01	31.81	541.06	51.69	671.10
320	120.70	557.22	112.93	436.01	27.40	436.01	31.93	543.80	51.84	674.24
340	120.18	560.81	111.93	432.01	27.24	432.01	32.06	546.55	51.98	677.36
360	119.66	564.40	110.93	428.01	27.08	428.01	32.19	549.29	52.12	680.46
380	119.13	567.99	109.91	424.01	26.92	424.01	32.31	552.04	52.26	683.55
400	118.59	571.58	108.90	420.01	26.76	420.01	32.44	554.79	52.40	686.62

6.15 DAM BREAK PROBLEM

The use of these methods can provide rough estimates of what happens when a gate is opened, or if a dam fails. If no flow initially exists downstream from the dam, or if the gate is completely closed, and there is an instant failure of the dam, or an instant opening of the gate, then such situations are commonly referred to as a "dam break problem."

To apply the characteristic method to this problem assume the channel is flat so that a constant depth exists upstream from the gate initially, so Y_o is constant and $V_o = 0$ as shown in the sketch.



To use the methods, let us consider that the gate moves downstream at a variable rate w, but not so fast as to leave the water until it reaches a constant velocity to create the situation shown in the above sketch by the dotted lines at some time t_1. After a short time, the movement of the gate can have a constant velocity w so that the C^+ characteristics become parallel as shown in the sketch of the xt plane above. To determine this constant slope, move along a C^- characteristic from the line traced in the xt-plane by the movement of the gate, which has an inverse slope w to the initial C^+ characteristic, which is shown by the dashed line in the sketch, giving

$$V - 2c = V_o - 2c_o = 0 - 2c_o = -2c_o, \quad \text{or}$$

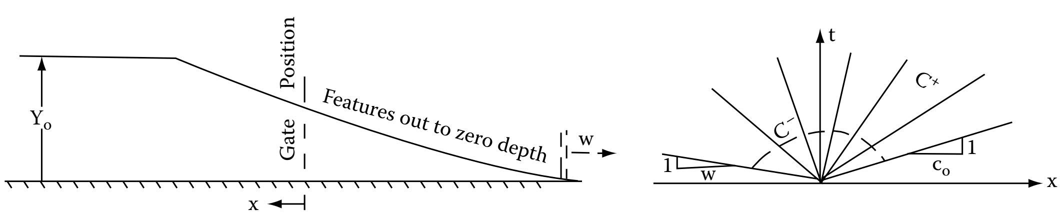
$$V = 2c - 2c_o = 2\left\{(gY)^{1/2} - (gY_o)^{1/2}\right\} = 2g^{1/2}(Y^{1/2} - Y_o^{1/2}\} \quad \text{and}$$

$$\frac{dx}{dt} = V + c = 3c - 2c_o$$

Note, the velocity V of the water is the same as the velocity of the gate, or V = w.

If the gate is moved fast enough to bring the depth Y to zero as shown below, then from the above equations, the velocity of the leading edge of water will feather down to a zero depth at the front (so c = 0) and the magnitude of this velocity is $|V| = 2c_o$, or

$$V = w = 2(0) - 2c_o = 2c_o = 2(gY_o)^{1/2}$$



Since, all the C^+ characteristics are straight lines, their slopes dx/dt can be given by x/t and

$$\frac{x}{t} = V + c = 3c - 2c_o = g^{1/2}\left\{3Y^{1/2} - 2Y_o^{\ 1/2}\right\}$$

The depth at the origin (at any time t) can be determined by letting x = 0 in this equation, or

$$3c - 2c_o = 3(2Y)^{1/2} - 2(gY_o)^{1/2} = 0 \quad \text{or}$$

$$Y_{x=0} = \left(\frac{2}{3}\right)^2 Y_o = \frac{4Y_o}{9}$$

and the velocity at the origin is

$$V_{x=0} = 2c_{x=0} - 2c_o = 2\left(\frac{2c_o}{3}\right) - 2c_o = -\frac{2c_o}{3} = -\left(\frac{2}{3}\right)(gY_o)^{1/2}$$

and the flow rate per unit width is given by

$$|q_{x=0}| = Y_x = 0\,|V_{x=0}| = \left(\frac{4}{9}\right)Y_o\left|\frac{2(gY_o)^{1/2}}{3}\right| = \left(\frac{8}{27}\right)g^{1/2}Y_o$$

PROBLEMS

6.1 A rectangular channel contains a uniform flow with Y_o = 4.5 ft and V_o = 3 fps at time zero. At this time, a gate is closed in such a manner that it reduces the depth at the beginning of the channel at a rate dY/dt = 1 ft/h for a 2 h period. Determine (a) how far the effect of this gate closure is noticeable downstream after 1 h, (b) the depth of water at a position 3000 ft downstream from the beginning of the channel at times 1, 4, and 10 min, and (c) what the velocities V and the flow rates q are at these times (1, 4, and 10 min) at this position (x = 3000 min).

6.2 A rectangular channel contains a flow rate per unit width of 4 m^2/s under uniform flow conditions at a depth of 2 m. At the upstream end, the depth of the flow is decreased so that it takes 2400 s for the depth to drop 1 m. This drop continues for 1200 s. Determine (a) how far the effect of dropping the upstream depth is felt after 120 and 1000 s, and (b) the depth and the velocity of the water at a position 1000 m downstream at times 1, 10, and 35 min.

6.3 A rectangular channel discharges into a small retention pond at its downstream end. For a long time, the pond level has been retained at an elevation of 5 ft above the channel bottom and this depth coincides with the normal depth in the channel. The flow rate per unit width in the channel under this condition is q = 17.5 cfs/ft. Suddenly, the withdrawal from the pond exceeds the inflow such that its level drops at a rate of 1.4 ft/h. Make up tables that show the depths, the velocities, and the discharges at the end of the channel where it discharges into the pond, and also 1500 ft upstream from here as a function of time. (Note that since the depth is known at the downstream end of this channel, x should be selected to be positive upstream with its origin at the end of the channel. Using this x coordinate V and q are negative.)

6.4 The flow rate per unit width into a channel is decreased at a rate of dq/dt = –2 cfs/ft/h. At time zero, the channel contains a uniform flow at a depth of Y_o = 4.0 and V_o = 3 fps. Make up tables that give (a) the depth, (b) the velocity, and (c) the discharge at both x = 0 and at x = 2000 ft, as a function of time.

6.5 Water behind (i.e., upstream from) a gate is essentially at uniform depth, with $Y_o = 3$ m, and $V_o = 1.5$ m/s. Suddenly, the gate must be opened gradually to increase the discharge at a rate of dq/dt = 0.025 $(m/s)^2$ for 60 s and the discharge must be held constant again at this new rate. Determine the depth, the velocity, and the discharge at the gate, and at a position 300 m upstream from the gate for times 0 through 80 s in increments of 20 s.

6.6 A rectangular river is discharging into the ocean at the time when the tide begins to recede. Initially, the river contains a flow rate per unit width of 20 cfs/ft at a depth of 6 ft. If the tide causes the ocean water surface to drop at a rate of 0.5 ft/h, and this rate is constant for some time, determine the discharge in the river at its end and 1 mile upstream therefrom in time increments of 10 min over a period of 1 h.

6.7 The flow downstream from a gate is at uniform flow with $V_o = 3.8$ fps and $Y_o = 5.5$ ft when suddenly, the flow rate per unit width at the gate is decreased at a constant rate of dq/dt = −0.0025 $(ft/s)^2$ for 1.2 h and then held constant again. At a position 2500 ft downstream from the gate, determine the times when the depths will decrease from the uniform depth at increments of 0.1 ft, i.e., when Y = 5.4, 5.3, 5.2, etc.

6.8 A uniform flow exists downstream from a gate as in the previous problem with $V_o = 3.8$ fps and $Y_o = 5.5$ ft, when suddenly the depth of flow in the channel at the gate is decreased at a rate of 1.5 ft/h for 1 h. On a 30 s increment determine the depth of flow, the velocity, and the flow rate q at a position x = 2500 ft downstream from the gate.

6.9 The uniform velocity and the depth downstream from a gate are $V_o = 1.2$ m/s and $Y_o = 2$ m, respectively. The flow rate per unit width at the gate is decreased at a constant rate of 0.001 $(m/s)^2$. At a position 1500 m downstream from the gate make up a table that gives the times when the depth will decrease in increments of 0.1 m.

6.10 The same as in the previous problem except, you are to find the depth, the velocity, and the flow rate at the position 1500 m on a 1 min interval.

6.11 Water enters a mild rectangular channel under uniform flow conditions at a flow rate per unit width of $q_o = 20$ cfs/ft, and a depth $Y_o = 5$ ft. Because of an increase inflow rate into the reservoir its water surface begins to rise at a rate of dY/dt = 0.02 ft/s for 100 s and then remains constant thereafter. Do the following: (a) Make a table that gives the flow rate at the entrance to the channel for times, $t_1 = 30$ s, $t_2 = 100$ s, and $t_3 = 160$ s. (b) What is the depth and the velocity at a position 500 ft downstream from the reservoir at time 50 s? (c) At 160 s, how far has the effect of increasing the reservoir level propagated to?

6.12 Water enters a trapezoidal channel with b = 5 m and m = 1.5 from a reservoir such that for a long time the depth is constant throughout the channel at 2.5 m and the flow rate is Q = 15 m^3/s. Suddenly, at the reservoir, the depth of water in the channel is decreased at a rate dY/dt = −0.002 m/s. Determine when the depth will be 2.2 m at a position 1000 m downstream in the channel, and how far downstream from the beginning of the channel the effect of the transient is noticeable at this time.

6.13 In the previous problem, the flow rate entering the channel is suddenly decreased at a rate dQ/dt(0, t) = −0.015 m^3/s^2 instead of the depth. Determine the same quantities.

6.14 Water flowing under uniform flow conditions in a trapezoidal channel with a bottom width b = 10 ft, and a side slope m = 1.8 into a retention pond whose water surface has been held constant for a long time because the outflow from it was equal to that coming in. Under these uniform conditions, the depth of flow is $Y_o = 4.5$ ft and the velocity is $V_o = 2.5$ fps. The outflow from the pond is suddenly increased so that its water surface begins to drop at a rate dY/dt = −0.001 ft/s. Determine when the depth will be 4.2 ft at a position 2000 ft upstream from the pond, and how far the effect of the increasing flow in the channel will be felt at this time.

6.15 Everything is as in the previous problem except that the flow rate into the pond is increased at a rate of dQ/dt = 0.025 cfs/s. Determine the same quantities as in the previous problem.

6.16 A reservoir with a water surface elevation H = 5 ft above the channel bottom has been supplying a very long rectangular channel with a bottom width b = 10 ft, a bottom slope S_o = 0.0008, and a Manning's roughness coefficient n = 0.014 for a long time. The entrance loss coefficient is K_e = 0.2. Suddenly, the water surface elevation in the reservoir begins to drop at a rate of 0.02 ft/s, i.e., dH/dt = −0.02 fps. Fill in the tables below giving the celerity, the depth, the velocity, and the flow rate per unit width for the times indicated at both the beginning of the channel and at a distance 200 ft downstream therefrom.

At position x = 0 ft

Time (s)	c (fps)	Y (ft)	V (fps)	q (ft**2/s)
0				
20				
40				

At position x = 200 ft

Time (s)	H (ft)	c (fps)	Y (ft)	V (fps)	q (ft**2/s)
0					
20					
40					

6.17 Initially, the velocity and the depths in a rectangular channel are uniform and equal to V_o = 5 fps and Y_o = 4 ft, respectively. At this time, the flow rate per unit width at the downstream end of the channel is increased so that |dq/dt| = 0.05 ft²/s². (a) How long can this rate of increase of flow rate last? (b) Fill in the table below giving the velocities, depths, etc. at the times in the first column at the downstream end of the channel as well as at a position 200 ft upstream therefrom. (c) What is the value of dx/dt when the maximum value of |q| occurs? What does this mean?

	x = 0 ft				x = 200 ft			
Time (s)	q (cfs/ft)	c (fps)	V (fps)	Y (ft)	c (fps)	Y (ft)	V (fps)	q (cfs/ft)
0	−20.00	11.349		4.000	11.349	4.000		
30								
60								
90								
120								

6.18 A rectangular channel initially contains a uniform flow at a depth of Y_o = 4 ft and a velocity V_o = 3 ft/s, when suddenly at it downstream end a gate is raised to increase the flow rate per unit width at a rate of dq/dt = 0.1 (ft/s)² for 90 s. (a) When will the depth be 3.8 ft at a position x = 500 ft upstream from the gate? (b) At this time, how far upstream has the effect of raising the gate propagated? (c) Determine the depth at a position x = 500 ft upstream from the gate at time t = 120 s. What is the corresponding flow rate? (d) What is the maximum flow rate that can occur past the gate?

6.19 In discussing the characteristics both upstream and downstream of a point where an outflow causes the depth to decrease at a rate dY/dt = −0.005 ft/s, when the initial conditions were

$Y_o = 5$ ft and $V_o = 4$ fps, a table was given that provides the following variables at positions x = 0 and x = 1000 ft, for the downstream portion of the channel. Write a computer program or develop a spreadsheet, etc., to duplicate the values in this table. Also, solve the same problem except increase the depth at the origin at a rate of dY/dt = 0.001 ft/s.

6.20 Repeat the previous problem, except write a computer program to generate the values in the table that apply for the channel upstream from the point outflow. Also, solve the same problem except increase the depth at the origin at a rate of dY/dt = 0.001 ft/s.

6.21 In discussing the negative characteristics associated with the portion of the channel downstream from a point outflow, i.e., the shape of the C^- characteristic passing through the t axis at t = 300 s, a table was provided that numerically solved $(dx/dt)^- = V - c$ to obtain the values of x corresponding to t in 10 s decrements of t. Duplicate this table of values by writing a computer program (or use some other software package). Also, obtain the solution tracing out the negative characteristic passing through the t axis at t = 200 s and t = 100 s. Note that your program will need to utilize the fact that the C^+ characteristics for the flow in the portion of the channel downstream from the point outflow are straight lines, and therefore the c corresponding to any coordinates (x, t) can be evaluated from $t = t_1 + \Delta t$, where $\Delta t = x/(dx/dt)^+$. Solve the problem twice; once with dY/dt = −.005 fps and once with dY/dt = 0.001 fps.

6.22 Repeat the previous problem except numerically solve the C^+ characteristics in the portion of the channel upstream from the point outflow, i.e., duplicate the table given in the text that applies for upstream channel in the example given in the text. Using the program you have thus developed, also solve the shape, etc., of the C^+ characteristics through the t axis for times t = 100 s, t = 200 s, and t = 400 s. Also, solve this problem twice; once with dY/dt = −.005 fps and once with dY/dt = 0.001 fps.

6.23 Construct a graph that displays the grid of C^+ and C^- characteristics that apply for a rectangular channel that initially contains a uniform flow with a depth $Y_o = 6$ ft and a velocity $V_o = 3$ fps. At the upstream end of this channel the depth decreases at a constant rate of dY/dt(0, t) = −0.005 ft/s. On this graph, have the C^- characteristics pass through the t-axis starting at t = 50 s and each 25 s thereafter until t = 500 s. Also have the C^+ characteristics pass through these same positions on the t-axis. Have the x-axis extend downstream for 3000 ft. Show the depth and the velocity that occur along each C^+ characteristic, as well as the constant V + 2c for this characteristic. What is the constant V − 2c for all of the C^- characteristics? Use the graph to fill in the table below.

	x = 0 ft		x = 1000 ft		x = 2000 ft	
t (s)	**Y (ft)**	**V (fps)**	**Y (ft)**	**V (fps)**	**Y (ft)**	**V (fps)**
0						
200						
400						
500						

6.24 Construct a graph that displays the grid of C^- and C^+ characteristics that apply for a rectangular channel that initially contains a uniform flow with a depth $Y_o = 6$ ft and a velocity $V_o = 3$ fps. At the downstream end of this channel, the depth decreases at a constant rate of dY/dt(0, t) = −0.005 ft/s. Have the x-axis point in the direction of the flow so that the C^- characteristics are straight lines with an inverse slope dx/dt = V − c, and the C^+ characteristics are not straight lines. On this graph, have both the C^- and C^+ characteristics pass through the t-axis at t = 50, 75, 100, 125, …, 500 s and the x-axis extend upstream to −2000 ft. Give the depth and the velocity that is constant along each C^- characteristic, as well as the constant

V − 2c for this characteristic. What is the constant V + 2c for all of the C^+ characteristics? Use the graph to fill in the table below.

	x = 0 ft		x = −600 ft		x = −1200 ft	
t (s)	Y (ft)	V (fps)	Y (ft)	V (fps)	Y (ft)	V (fps)
0						
200						
400						
500						

6.25 Repeat the previous problem in making a graph of the characteristics in the xt-plane downstream from the source of the disturbance, except rather than specifying dY/dt, specify dq/dt at the origin. The graph should apply for a rectangular channel that initially contains a uniform flow with $Y_o = 6$ ft, $V_o = 3$ fps, and the rate of flow decreased to x = 0 so that dq/dt(0, t) = −0.015 $(ft/s)^2$. From the graph fill in the table below.

	x = 0 ft		x = 1000 ft			x = 2000 ft		
t (s)	Y (ft)	V (fps)	Y (ft)	V (fps)	q (cfs/ft)	Y (ft)	V (fps)	q (cfs/ft)
0								
200								
400								
500								

6.26 For the example problem given in the text in which a point outflow occurs near the center of a channel with $Y_o = 5$ ft and $V_o = 4$ ft/s, the point outflow was increased at a rate of d(Δq)/dt = 0.1 cfs/ft/s for 100 s and then held constant for subsequent times to Δq = 10 cfs/ft of channel width, and variables of the flow were given in a table for x = 0, 500, 1000, −500, and −1000 ft. Write a computer program to duplicate the values in this table.

6.27 Repeat the solution that you obtained from your program of the previous problem except change the point outflow at a rate of d(Δq)/dt = 0.05 cfs/ft/s for 100 s, and then at a rate of d(Δq)/dt = 0.15 cfs/ft/s for 100 s.

6.28 A uniform flow exists in a channel initially at a depth of $Y_o = 2.5$ m and a velocity of $V_o = 2.3$ m/s. Suddenly, at a position near the middle of the channel, a point outflow is started, which varies in time as shown in the table below. Using a time increment of 10 s, determine the depth, the velocity and the q at x = 0 (both upstream and downstream from the point outflow), x = 100 m, x = 250 m, x = −50 m, and x = −120 m.

t	0	5	8	15	20	35	50	90
Δq	0	.2	1.0	1.5	2.0	2.4	2.0	1.8

6.29 When a point outflow Δq takes place near the middle of a channel it was shown in the text that the depth Y(0, t) (and the celerity c(0.t)) at x = 0 does not depend upon time t or the rate of d(Δq)/dt, but only on Δq and the initial steady-state conditions. Write a computer program that will produce the numbers that are needed to make a graph such as the dimensionless graph that gives $c' = c/c_c$ as a function of the dimensionless unit outflow $\Delta q' = \Delta q/q_o$ and the initial dimensionless celerity $c'_o = c_o/c_c$.

6.30 Using the x axis pointing in the upstream direction solve the ODE in the upstream portion of the channel that provides the negative characteristics C^- for the problem with $Y_o = 5$ ft, $V_o = 4$ fps, and dY/dt = −0.005 ft/s in the upstream portion of the channel. These negative characteristics should have identical magnitudes but an opposite sign to those provided in the text for the positive characteristics. (The result should be those in Problem 6.22, with opposite signs.)

6.31 For the Example Problem 6.7, in which the uniform condition are $Y_o = 5$ ft and $V_o = 4$ fps, extract the maximum point outflow per unit width Δq_{max} at a midpoint of the channel, and determine the following at the origin x = 0: c′, Y′, Y, q_d, and q_u, and then verify that $\Delta q_{max} = q_u - q_d$.

6.32 In the text, and also in Problem 6.21, the negative characteristics were obtained by solving the ODE dx/dt = v − c in the xt-plane when x is positive in the downstream direction, and the rate of depth change dY/dt is specified (i.e., the depth is given as a function of time). Develop a similar program to the one developed in Problem 6.21, except have it obtain the position of the negative characteristics when the change in flow rate per unit width dq/dt is specified. Use this program to solve the problem in which the rate of discharge into a rectangular channel with $Y_o = 5.5$ ft, $V_o = 4.5$ fps, and the rate of unit flow rate is dq/dt = .15 cfs/ft/s. (This represents a large rate of change and is a large value for this theory, especially if carried over a long period of time.) Have the negative characteristics pass through the t-axis at 400, 300, 200, and 100 s.

Add the capability to this program to solve the celerity, the depth, the velocity, and the unit flow rate at the origin, and at several specified downstream distances x. Use this modified program to solve the problem with $Y_o = 5.5$ ft, $V_o = 4.5$ fps, and dq/dt = .05 cfs/ft/s, and let the downstream positions be x = 500 ft, x = 1000 ft, and x = 1500 ft. Also, for this second solution, use t = 400 s, t = 300 s, t = 200 s, and t = 100 s.

6.33 Obtain a series of solutions that show how the depth and the velocity change as the flow rate is increased at the downstream end of a channel from its uniform conditions. For these solutions, the uniform flow conditions are: $Y_o = 5.5$ ft and $V_o = 4.5$ fps. Have the flow rate start at the uniform flow rate and increment q by 1.0 cfs/ft until q is very close to the maximum that can be withdrawn as the depth approaches critical depth. Also, obtain a series of solution in which q starts with q_o and decreases in even increments of 1 cfs/ft to q = 0.

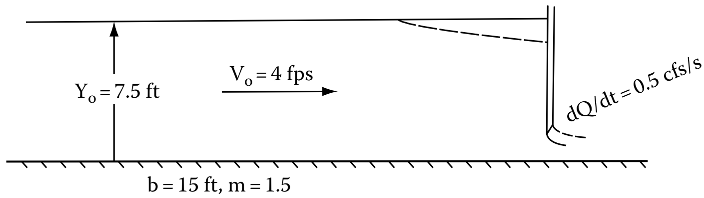
6.34 In the text, and also in Problem 6.22, the positive characteristics were obtained by solving the ODE dx/dt = v + c in the xt-plane when x is positive in the downstream direction, and the rate of depth change dY/dt is specified (i.e., the depth is given as a function of time). Develop a similar program to the one developed in Problem 6.22, except have it obtain the position of the positive characteristics when the rate of flow rate dq/dt is specified. Add the capability to this program to solve for the celerity, the depth, the velocity, and the unit flow rate at the origin, and at several specified upstream distances x. Use this program to solve the problem with $Y_o = 5.5$ ft, $V_o = 4.5$ fps, and dq/dt = .02 cfs/ft/s, and let the upstream positions be x = −500 ft, x = −1000 ft, and x = −1500 ft. Start the positive characteristic at t = 300 s.

6.35 A rectangular channel initially contains a uniform flow with a velocity $V_o = 1.5$ m/s and a depth $Y_o = 2$ m. Determine the maximum point unit outflow Δq that can be taken from an intermediate position along this channel. What fraction of this maximum outflow comes from downstream reverse flow? (Solve this problem using the explicit equations that give Δq_{max} and c_c, and then also solve the problem by using pairs of equations given in the text based on the downstream flow rate and the upstream velocity, as well as the equation that gives Δq.

6.36 A trapezoidal channel with a bottom width b = 8 ft and a side slope m = 1.5 contains a uniform flow initially at $Y_o = 4$ ft and $V_o = 5$ fps. At time 0, the flow rate into the upstream end of the channel is reduced at a rate of 0.5 cfs/s (dQ/dt = −0.5). Fill in the table below the upstream end and at a position 200 ft downstream therefrom.

	x = 0 ft					x = 2000 ft			
Time (s)	**Q (cfs)**	**Y (ft)**	**A (ft**2)**	**c (fps)**	**V (fps)**	**Y (ft)**	**c (fps)**	**V (fps)**	**Q (cfs)**
0									
30									

6.37 A trapezoidal channel with a bottom width of b = 15 ft, and a side slope of 1.5 contains a constant depth of 7.5 ft and a velocity of 4 fps upstream from a gate. At time t = 0, the flow rate past the gate is increased at a rate of $|dQ/dt| = 0.5$ cfs/s. Determine the velocity and the depth at the gate after 60 s. Also, determine the depth, the velocity, and the flow rate at a position 500 ft upstream from the gate at t = 100 s. How long will this rate of flow increase be possible?

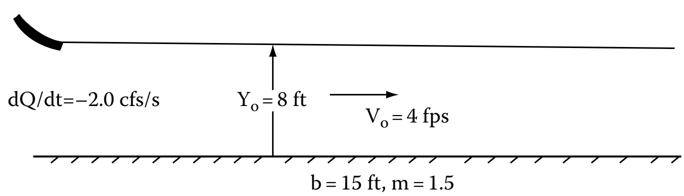


6.38 A rectangular channel receives its supply from an upstream reservoir under uniform flow conditions of $Y_o = 6$ ft and $V_o = 4$ fps. At t = 0, the flow rate into the channel is increased at a rate of dq/dt = .30 $(ft/s)^2$. (a) When and where is a surge likely to develop?

6.39 Water enters a trapezoidal channel with b = 20 ft and m = 1.5, from a reservoir such that for a long time, the depth has been constant throughout the channel at 9.5 ft and the flow rate has been Q = 900 cfs. Suddenly, at the reservoir, the depth of water in the channel is decreased at a rate of dY/dt = −0.004 ft/s. Determine when the depth will be 9.0 ft at a position 2000 ft downstream in the channel, and how far downstream from the beginning of the channel the effect of the transient is noticeable at this time.

6.40 In the previous problem, the flow rate entering the channel is suddenly decreased at a rate of dQ/dt(0, t) = −1.50 cfs/s instead of the depth. Determine the same quantities.

6.41 A gate controls the flow into a trapezoidal channel with a bottom width b = 15 ft, and a side slope m = 1.5. The gate has been set so as to produce a uniform flow at $Y_o = 8$ ft, and $V_o =$ 4 fps through the channel. At time t = 0, the gate is slowly closed so that the flow rate into the channel is decreased at a rate of dQ/dt = −2.0 cfs/s. (a) What will the depth be at the gate 30 s later? (b) At a position 2000 ft downstream what will the depth, the velocity, and the flow rate be at t = 150 s?



6.42 All initial conditions are the same as in the previous problem except that the gate exists at the downstream end, and this gate is rapidly opened. What is the maximum flow rate (ignoring sign) that will take place at the downstream end of the channel, and what is the minimum depth associated with the maximum flow rate?

6.43 The flow rate in Problem 6.41 is decreased at a rate of dQ/dt = −1.4 cfs/s at the reservoir. Make a table that shows the depth in the channel at both its beginning and at a position 3000 ft downstream therefrom as a function of time.

6.44 Water flows through a circular channel with a diameter D = 15 ft at a rate of $Q_o = 200$ cfs, and at a velocity of $V_o = 3.0$ fps. At its downstream end, the flow rate is suddenly increased at a rate of dQ/dt = 0.02 cfs/s. Determine when the depth will equal 5.5 ft at the downstream end of this channel, or whether the minimum depth is reached first, and if so what this depth is and what the limiting flow rate is. At a position 2500 ft upstream from its end, when will the flow rate equal 230 cfs?

6.45 Write a computer program similar to YTIMEX, or develop a computer model, that solves problems in which the variations of Y and V are at the origin and how Y, V, and Q variations at positions x are sought if the time varying flow rate at the origin is specified, but handles circular channels. Use this program to verify the results from the previous problem by interpolating in these solution tables.

6.46 Water flows through a circular channel with a diameter D = 5 m at a rate $Q_o = 5.5\,m^3/s$, and at a velocity of $V_o = 0.3$ m/s. At its upstream end, the flow rate is suddenly decreased at a rate $dQ/dt = -0.005\,m^3/s^2$. Determine when the depth will equal 4.3 m at the upstream end of this channel. At a position 1000 m downstream from its entrance, when will the flow rate equal $5.0\,m^3/s$.

6.47 Use the computer program you developed in Problem 6.45 to solve the previous problem giving how Y, V, and Q vary at the origin and at x = 1000 m, and by interpolating in these solution tables verify the results of the previous problem.

6.48 Water flows through a circular channel with a diameter D = 9 ft at a rate of $Q_o = 90$ cfs and at a velocity of $V_o = 1.5$ ft/s. At its upstream end, the flow rate is suddenly decreased at a rate of $dQ/dt = -0.015\,ft^3/s^2$. Determine when the depth will equal 7.5 ft at the upstream end of this channel. At a position 3000 ft downstream from its entrance, when will the flow rate equal 80 cfs?

6.49 Use the computer program you developed in Problem 6.45 to solve the previous problem giving how Y, V, and Q vary at the origin and at x = 3000 ft, and by interpolating in these solution tables verify the results of the previous problem.

6.50 Develop a program that will integrate the C^- characteristics in the xt-plane with x positive in the downstream direction for a trapezoidal channel. Use this program to obtain the C^- through the t-axis at t = 200 s if the initial uniform flow consists of $Y_o = 5$ ft, $Q_o = 300$ cfs, and the depth drops at x = 0 at a rate of $dY/dt = -0.005$ ft/s. The channel has a bottom width b = 10 ft, and a side slope m = 1.

6.51 Construct a graph that displays the grid of C^+ and C^- characteristics that apply for a trapezoidal channel with a bottom width b = 12 ft and a side slope m = 1.5, that initially contains a uniform flow with a depth $Y_o = 6$ ft and a velocity $V_o = 3$ fps. At the upstream end of this channel, the depth decreases at a constant rate of $dY/dt(0, t) = -0.005$ ft/s. On this graph, have the C^- characteristics pass through the t-axis starting at t = 50 s and for each 25 s thereafter until t = 500 s. Also, have the C^+ characteristics pass through these same positions on the t-axis. Have the x-axis extend downstream for 3000 ft. Show the depth and the velocity that occur along each C^+ characteristic, as well as the constant V + w for this characteristic. What is the constant V − w for all of the C^- characteristics? Use the graph to fill in the table below.

	x = 0 ft			x = 1000 ft			x = 2000 ft		
t (s)	Y (ft)	V (fps)	Q (cfs)	Y (ft)	V (fps)	Q (cfs)	Y (ft)	V (fps)	Q (cfs)
0									
200									
400									
500									

6.52 Construct a graph as in the previous problem showing a grid of positive and negative characteristics for a trapezoidal channel with a bottom width b = 12 ft and a side slope of m = 1.5, if initially the flow is uniform at a depth of $Y_o = 6$ ft and $V_o = 3$ fps. However, rather than specifying a change in the depth at the origin, the rate of flow rate at the origin is changed. For the first 250 s $dQ/dt = -0.75$ cfs/s, and thereafter $dQ/dt = -0.25$ cfs/s.

6.53 Repeat the previous problem except the channel is circular with a diameter D = 12 ft. Initially, there is a uniform flow of Q_o = 400 cfs at a depth Y_o = 8 ft. The rate of change inflow rate at the origin is $(dQ/dt)_1 = -1.0$ cfs/s for the first 250 s, and thereafter $(dQ/dt)_2 = -0.25$ cfs/s.

6.54 Write a program that will integrate the negative characteristics in the x positive portion of the xt plane as in the previous problem, except specify the rate of change in the flow rate dQ/dt rather than the rate of change in the depth dY/dt. The type of channels to be accommodated are trapezoidal in cross-section. Obtain the shape of the negative characteristic that passes through the t-axis at t = 100, 200, and 300 s, if the initial uniform conditions are: Y_o = 5 ft, and Q_o = 300 cfs, and the rate of decrease of flow rate is dQ/dt = −1.0 cfs/s. The channel has b = 10 ft and m = 1.0.

6.55 Develop a similar program to that of Problem 6.50 that integrates for the negative characteristics in the xt-plane except rather than for a trapezoidal channel have it apply to a circular channel. Then, determine the shape of the C^- characteristic through the t-axis at t = 300, 200, and 100 s if the initial uniform flow has Y_o = 5 ft, Q_o = 300 cfs, and the depth decreases at a rate of dY/dt = −0.005 ft/s at the origin. The circular channel has a diameter D = 12 ft.

6.56 Obtain a series of solutions that give the flow rate, the velocity, the celerity, the w, the rate of change of flow rate with depth, dQ/dY, and the Froude number as the depth changes at the downstream end of a channel (where x = 0, and points upstream), if the initial depth is Y_o = 6 ft, and the initial flow rate is Q_o = −450 cfs. The starting depth for this series of solutions should start with 8 ft (or 2 ft above the initial depth), and end with a depth below the critical depth, when the negative velocity in the channel equals the upstream speed of the gravity wave, c. Obtain this series of solutions for both a trapezoidal channel with b = 15 ft and m = 1, and a circular channel with a diameter D = 15 ft. Note that the Froude number equals unity when the maximum flow rate occurs. Note that these series of solutions describe what happen at the origin, (x = 0) and that the time required to get to these new depths does not enter into the equations that give conditions at the origin. If one were interested in what these variables are at some upstream positions, then the time required to get to each of the new depths would be needed.

6.57 Fill in the missing values in the table below. Note that for both the trapezoidal and the circular channels, the initial depth is Y_o = 6 ft, and the initial flow rate is Q_o = |450| cfs (which is negative when x is positive in the upstream direction), and varies by plus or minus 100 cfs from this value, except rather than −550 cfs for the circular channel −495 cfs is used because critical conditions limit the maximum value. For each of the eight situations identify whether a negative, or a positive wave will occur. (Note that a surge might form when a positive wave occurs depending upon how rapidly the new flow rate develops.) For the channels in which positive waves occur, determine the rate of depth increase dY/dt that will result in the surge forming at a position 1000 ft from the control.

Channel	Control	b or D (ft)	m	Y_o (ft)	Q_o (cfs)	Q (cfs)	Y (ft)	V (fps)	c (fps)	A (ft²)	W (fps)	Fr	Wave
Trapezoidal	upstream	15	1	6	450	350							
Trapezoidal	upstream	15	1	6	450	550							
Trapezoidal	downstream	15	1	6	−450	−350							
Trapezoidal	downstream	15	1	6	−450	−550							
Circle	upstream	15		6	450	350							
Circle	upstream	15		6	450	550							
Circle	downstream	15		6	−450	−350							
Circle	downstream	15		6	−450	−495							

6.58 Develop a computer solution to numerically solve the ODE that determines the shape of the positive characteristics in the upstream portion of the xt-plane (where x is negative but x points positive in the downstream flow direction) for a trapezoidal channel. In this plane, the negative characteristics with a slope V − c are straight lines (with V positive) and the positive characteristics are not straight lines, i.e., their slope at any point is given by V + c. Have this program have the rate of flow rate change at the origin dQ/dt equal to a constant. Then, for a trapezoidal channel with a bottom width b = 10 ft, and a side slope m = 1, and containing an initial uniform flow of Y_o = 5 ft and Q_o = 300 cfs, determine the positive characteristics through the t axis at t = 100, 200, and 225 s if the rate of flow rate change is dQ/dt = 0.5 cfs/s.

6.59 Repeat the previous problem except solve the shape of the positive characteristics for a circular section in the xt plane with x negative, i.e., when the control is downstream and the negative characteristics are straight lines. Obtain these characteristics for a circular channel with D = 15 ft, and an initial uniform flow of Y_o = 5 ft and Q_o = 300 cfs, if dQ/dt = 0.1 cfs/s, through the t axis at t = 100, 200, and 300 s.

Also obtain the positive characteristics through t = 300 if the rate of flow rate increase is dQ/dt = .133 cfs/s. What do you note about the first c and V at t = 300? Also, obtain the positive characteristics through the largest time t that is possible if the rate of flow rate increase is dQ/dt = .15 cfs/s. (What is Q_{max}?)

6.60 Develop a computer program to obtain the tables given in Example Problem 6.13. In addition to verifying the values in these tables use this program to solve the following problem: Initially the uniform depth and flow rate in a trapezoidal channel are Y_o = 6 ft and Q_o = 400 cfs, respectively. The channel has a bottom width b = 10 ft, and a side slope m = 1.0. At an intermediate position along this channel, a point outflow is started such that d(ΔQ)/dt is constant and equal to 1.5 cfs/s for 100 s and remains constant at ΔQ = 150 cfs for t > 100 s. Use a time increment of Δt = 10 s in the solution tables and solve the flow variables at two downstream positions, x = 500 ft and x = 1000 ft, and at two upstream position, x = −200 ft and x = −600 ft.

6.61 Determine the maximum point outflow ΔQ that can be taken from a trapezoidal channel with a bottom width b = 4 m and a side slope m = 1.6, if the initial uniform depth is Y_o = 2 m, and the uniform discharge is: (a) Q_o = 50 m^3/s, (b) Q_o = 25 m^3/s, and (c) Q_o = 1 m^3/s. Why would you expect a smaller contribution of ΔQ from the downstream channel for the larger discharges with reverse flow occurring only as Q_o becomes smaller?

6.62 Determine the maximum point outflow that can be taken from an intermediate position along a circular channel with a diameter D = 15 ft that initially contains a uniform flow at a depth Y_o = 7 ft and a flow rate Q_o = 300 cfs.

6.63 The flow rate into a rectangular channel that contains a uniform flow with q_o = 15 cfs/ft and Y_o = 6 ft is suddenly increased so that its water surface at its beginning rises at a rate of dY/dt = .02 ft/s. Determine when and where a surge will form.

6.64 A channel discharges into a downstream reservoir. At time zero, the flow conditions are uniform with a depth Y_o = 5 ft and V_o = 3.0 fps. The reservoir water surface begins to rise at a rate of dY/dt = 0.0095 fps. Determine when and where a surge will form.

6.65 A trapezoidal channel with a bottom width b = 8 ft and a side slope m = 1.5 contains a uniform flow initially at Y_o = 4 ft and V_o = 5 fps. At time 0, the magnitude of the flow rate at the downstream end of the channel is increased at a rate of |dQ/dt| = 0.5 cfs/s. What is the minimum depth that can be created at the downstream end, and what is the maximum flow rate (magnitude) associated with this depth?

6.66 A trapezoidal channel with b = 10 ft, and m = 1 initially contains a uniform flow at a depth of Y_o = 4 ft and a velocity of V_o = 3 fps, when suddenly at its downstream end a gate is raised to increase the flow rate at a rate of dQ/dt = 0.8 cfs/s for 90 s. (a) When will the depth be 3.8 ft at a position x = 250 ft upstream from the gate? (b) At this time, how far upstream has the effect of raising the gate propagated? (c) Determine the depth at a position x = 250 ft upstream from the gate at time t = 90 s. What is the corresponding flow rate? (d) What is the maximum flow rate that can occur past the gate?

6.67 A gate at the upstream end of a rectangular channel suddenly increases the flow per unit width into the channel at a rate of $dq/dt = 1.0\ (ft/s)^2$. Before this time assume that a uniform flow existed in the entire channel with a flow rate per unit width $q_o = 15$ cfs/ft and at a depth of $Y_o = 5$ ft. Determine when and where a surge will form.

6.68 Initially, the velocity and the depths in a rectangular channel are uniform and equal to $V_o = 5$ fps and $Y_o = 4$ ft, respectively. At this time, the depth at the downstream end of the channel is increased at a rate of $dY/dt = 0.01$ ft/s. When and where will a surge first develop?

6.69 The small downstream reservoir into which a trapezoidal channel discharges has its outflow suddenly stopped so that its water surface begins to rise at a rate of $dY/dt = 0.010$ ft/s. Prior to this time, a uniform flow existed in the channel at a rate of 500 cfs. The channel has a bottom width $b = 10.5$ ft, a side slope $m = 2$, and a bottom slope $S_o = 0.00075$ and $n = 0.013$. Determine the following: (a) when and where a surge is likely to first form, (b) the discharge into the reservoir as a function of time, and (c) the discharge as a function of time at a position 500 ft upstream from the reservoir.

6.70 A trapezoidal channel with a bottom width of $b = 10$ ft, and a side slope of $m = 1.5$ contains a flow with a constant depth of 5 ft and a velocity of 4 fps upstream from a gate. Determine the flow rate and the velocity at the gate after 100 s if the depth of water at the gate is decreased at a rate of $dY/dt = -0.015$ ft/s. Also, determine the depth, the velocity and the flow rate at a position 600 ft upstream from the gate at the time of $t = 100$ s.

6.71 A flow upstream from a gate in a rectangular channel is at a constant depth of 5 ft and a velocity of 3 fps when the gate at the downstream end of the channel is raised to increase the flow rate per unit width at a rate of $dq/dt = 0.01$ cfs/s. Determine the following after 200 s: (1) the depth and velocity at the gate, (2) the depth, the velocity and the flow rate at a position 100 ft upstream from the gate, and (3) the position where the effects of opening the gate have propagated to.

Prove that the maximum flow rate per unit width that the channel can deliver past the gate is defined by

$$q_{max} = \frac{(V_o - 2c_o)^3}{(27g)}$$

For this channel, what is this maximum flow rate per unit width? What are the equations that give the depth and the velocity corresponding to this maximum flow rate? Give all three of these values for the above channel, i.e., give: q_{max}, V_{max}, and Y_{max}.

6.72 In the previous problem, what is the maximum rate at which the flow rate per unit width at the gate can be decreased so that a surge will not form at a position 2000 ft (or less than this distance) upstream from the gate?

6.73 A uniform flow with a depth $Y_o = 2$ m and $V_o = 1.5$ m/s exists initially in a trapezoidal channel with $b = 3.5$ m and $m = 1.5$. Determine a constant rate of increase in the flow rate dQ/dt that this channel can sustain over a 10 min time period. What will the depth at the gate be, and what will the flow rate past the gate be at this time of 600 s? How much volume of water will be removed from channel storage at this time, if this rate of increase in the flow rate occurs?

6.74 A circular channel with a diameter $D = 5$ m discharges into a downstream reservoir. Initially, the depth and the velocity throughout the channel are $Y_o = 4.0$ m and $V_o = 2.5$ m/s, respectively. The water surface elevation in the downstream reservoir suddenly begins to fall very rapidly. What will the limiting depths and the flow rates be at the end of the channel? After 180 s how much volume of water has been removed from channel storage? What are the depth, the velocity, and the flow rate in this channel at a position 200 m upstream from the reservoir at a time $t = 180$ s.

6.75 A trapezoidal channel with a bottom width b = 15 ft and a side slope m = 1.5 initially contains a uniform flow rate with a depth of Y_o = 8.00 ft and a velocity of V_o = 4.00 fps. At the downstream gate of this channel, the flow rate is suddenly increased by the gate at a constant rate dQ/dt = 1 cfs/s. Determine: (a) For how long can this rate of increase in the flow rate occur? (b) At this time, what is the depth at the gate, and what is the flow rate past the gate? (c) At 60 s, what are the depth and the flow rate at the gate?, and (d) What are the depth and the flow rates at a position 500 ft upstream from the gate at 60 s?

6.76 Write a computer program, or a computer model, to generate the values for the stage variable w and the celerity c in Table 6.1 for a trapezoidal channel. Add a fourth column that gives the sum of c + w.

6.77 Write a computer program, or a computer model, to generate the values for the stage variable w and the celerity c in Table 6.2 for a circular channel. Add a fourth column that gives the sum of c + w.

6.78 A reservoir, whose water surface elevation has been constant for a long time with H = 6 ft, supplies a trapezoidal channel with b = 15 ft and m = 1.5, a Manning's roughness coefficient n = .014 and a bottom slope of S_o = 0.00085. The entrance loss coefficient is K_e = 0.2. Suddenly, the water surface elevation of the reservoir begins to drop at a constant rate of dH/dt = −0.025 fps. Solve the depths, the velocities, and the flow rates at the beginning of the channel and at a position 500 ft downstream therefrom for several time steps.

6.79 The reservoir that supplies a trapezoidal channel with b = 3 m and m = 1 (with n = .013, and S_o = 0.0007) begins to drop at time zero at a rate of dH/dt = 0.015 m/s, after being constant at H = 2.5 m for a long time. Determine the depth, the velocity, and the flow rate at the beginning of the channel, and at a position 150 m downstream therefrom for a number of time steps.

6.80 The reservoir in the previous problem begins to rise at a rate of dH/dt = .015 m/s. When and where would you expect a surge to first form?

6.81 Starting with the functional relationship, Equation 6.40, that gives a relationship of Y and Q for a nonrectangular channel, and substitute appropriate dimensionless variables for a circular section to obtain a dimensionless relationship similar to Equation 6.44 that applies for a circular section. Set the derivative of this dimensionless equation to zero and verify that Equation 6.43 applies for a circular section.

6.82 Differentiate Equation 6.40 for a circular section and set this derivative to zero, and verify that Equation 6.42 applies for a circular section.

6.83 Write a computer program that will determine the maximum flow rate and the associated minimum depth that can occur past a gate's position in a trapezoidal channel if the gate is suddenly and completely opened, and if prior to the opening of the gate, the depth and the velocity in the channel upstream from the gate are constant under uniform conditions. As input data to the program give: the uniform depth and the velocity, the acceleration of gravity, the bottom width, and the side slope of the channel. Use this program to fill in the last two columns of the table below. Verify your answers using the Dimensional Figure in the book.

Y_c	V_c	b	m	Y_{min}	Q_{max}
4 ft	3 fps	10 ft	1.2		
8 ft	4 fps	20 ft	1.5		
2 m	1.5 m/s	5 m	1.2		
3 m	1.0 m/s	4 m	2.0		

6.84 Write a computer program that will determine the maximum flow rate and the associated minimum depth that can occur past a gate's position in a circular channel if the gate is suddenly and completely opened, and if prior to the opening of the gate, the depth and the velocity in the channel upstream from the gate are constant under uniform conditions. As input data to the program give: the uniform depth and the velocity, the acceleration of gravity, and the diameter of the channel. Use this program to fill in the last two columns of the table below. Verify you answers using the Dimensional Figure in the book.

Y_c	V_c	D	Y_{min}	Q_{max}
4 ft	3 fps	10 ft		
8 ft	4 fps	15 ft		
2 m	1.5 m/s	5 m		
3 m	1.0 m/s	4 m		

6.85 Write a computer program, or model, that provides values needed to plot Figure 6.4a for trapezoidal channels.

6.86 Write a computer program, or model, that provides values needed to plot Figure 6.4b for circular channels.

6.87 For special types of channels, such as a rectangular, triangular and parabolic, the stage variable w is a constant time the celerity c in that channel. Prove the relationships shown in the table below for these three channel types. In the last column of this table, c_r is the celerity in a rectangular channel, $c_r = (gY)^{1/2}$, and it shows that the celerity in a triangular and a parabolic channel are a constant time the celerity in a rectangular channel.

Type of Channel	Celerity, c	Stage Variable, w	c
Rectangular	$(gY)^{1/2}$	$2(gY)^{1/2} = 2c$	$c = c_r$
Triangular	$(gY/2)^{1/2}$	$2(2gY)^{1/2} = 4c$	$c = c_r/\sqrt{2} = .707c_r$
Parabolic	$(2gY/3)^{1/2}$	$(6gY)^{1/2} = 3c$	$c = c_r(2/3)^{1/2} = .8165c_r$

6.88 Write a computer program, or develop a spreadsheet, that generates a table of values for the dimensionless minimum depth, $Y'_{min} = mY_{min}/b$ and the maximum dimensionless flow rate $Q'_{max} = Q_{max}(gm^3/D^5)^{1/2}$ past a gate (or dam) site as a function of the initial dimensionless depth $Y'_o = mY_o/b$ and the initial dimensionless velocity, $V'_o = V_o(gm/b)^{1/2}$, if the gate (or dam) is suddenly removed. This program is to apply for trapezoidal channels. In other words, the values in this table could be used to generate the graphs in Figure 6.4a. Note, if you use TK-Solver to obtain this table, it can also be plotted, or the output from a computer program could be imported into a spreadsheet and plotted.

6.89 Write a computer program, or develop a spreadsheet, which generates a table of values for the dimensionless minimum depth, $Y'_{min} = Y_{min}/D$ and the maximum dimensionless flow rate $Q'_{max} = Q_{max}(g/D^5)^{1/2}$ past a gate (or dam) site as a function of the initial dimensionless depth $Y'_o = mY_o/b$ and the initial dimensionless velocity, $V'_o = V_o(g/D)^{1/2}$, if the gate (or dam) is suddenly removed. This program is to apply for circular channels. In other words, the values in this table could be used to generate the graphs in Figure 6.4b. Note, if you use TK-Solver to obtain this table, it can also be plotted, or the output from a computer program could be imported into a spreadsheet and plotted.

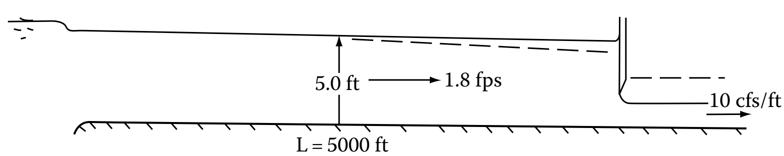
6.90 The equations that provided the maximum flow rate that can be passed by a gate, if it is rapidly opened wide, are based on assumptions upon which the characteristic method is based, i.e., $g(S_o - S_f) = 0$. It stands to reason that Q_{max} cannot be sustained indefinitely. To illustrate this, assume that initially, the uniform depth and the velocity in a trapezoidal channel are $Y_o = 5$ ft and $V_o = 3$ fps. This channel has the following properties: b = 10 ft, m = 1.5, n = 0.013, and $S_o = 0.000151$. For this uniform flow rate $Q_o = 262.5$ cfs, and for the uniform flow rates of $Q_o = 400$, 500, and 600 cfs do the following: (a) determine the minimum depth and the maximum (or minimum if sign is considered) flow rate, Y_{min} and Q_{max}, (b) compute the time it takes for the effects of opening the gate to be felt at a position 1000 ft upstream from the gate for each of these four flow rates, (c) for each of these four cases, compute the volume of water mined from the channel if the flow rate were able to continue at Q_{max} for the time computed in part (b), (d) compute the energy line for the wedge of flow between the gate and the 1000 ft position based on the assumption that S_f is parallel to the water surface and the water surface is a straight line, i.e., $S_f = S_o + (Y_o - Y_{min})/1000$, and based on this slope and the average depth in the 1000 ft wedge compute the flow rate, (e) compare the flow rate in part (e) with that in part (a), and described what is likely to actually occur (e.g., can the channel support Q_{max} indefinitely?) (f) compute the volumes of water in the upper wedges of the channel that have been dried up by the receding downstream depths over the 1000 length upstream from the gate, and compare these volumes with those computed in (c). Again, note the difference, and how these volumes suggest that the maximum flow rate will not be sustained for very long.

6.91 A circular channel that is 2000 m long contains a flow rate of $Q_o = 30\,m^3/s$ at a depth of $Y_o = 3.2$ m. The diameter of the channel is D = 5 m. Determine the maximum rate of inflow increase into this channel that can be accommodated without causing a surge to develop.

6.92 A ship lock consists of a rectangular channel 150 ft wide. The low water level depth in this channel is 50 ft. Its depth needs to be raised to 70 ft for the ships to pass through, and if the lock is 1500 ft long, determine the maximum rate at which it can be filled and not cause a surge to form as it is being filled.

6.93 A fisherman observes the formation of a surge in a river that discharges into the ocean. The river is 100 ft wide, has a depth of flow of 5 ft and contains a flow rate of Q = 900 cfs. If the fisherman is 1.5 miles upstream in the river, predict the rate at which the tide is rising in the ocean.

6.94 Assume that the depth $Y_o = 5.0$ ft and velocity $V_o = 1.8$ fps are constant upstream from a sluice gate to a reservoir that is 5000 ft upstream from the gate. The rate of discharge is to be increased so that after 180 s, the flow rate passing the gate per unit width should be 10.0 cfs/ft. If the gate is to be raised in such a manner as to create a constant rate of drop in the water surface immediately upstream from the gate, determine what this rate of drop should be. When will the effect of increasing the flow rate past the gate first reach the reservoir? Determine what the depth will be at a position 500 ft upstream from the gate after 240 s. Describe qualitatively what the effect of the reservoir is on the flow in the channel after the effect from the increased flow rate past the gate arrives at the reservoir.



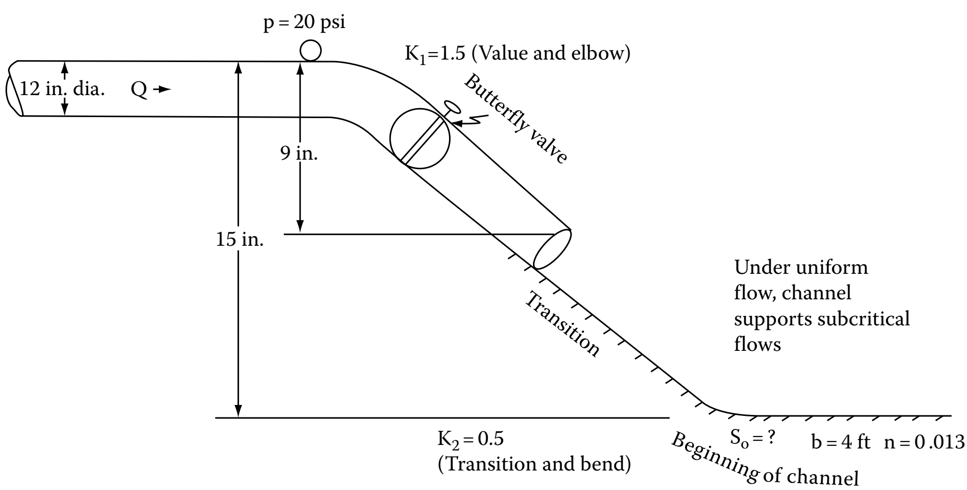
6.95 In the previous problem at time zero the hydraulic jump exists immediately downstream from the gate (i.e., there is a zero length M_3 GVF-profile) and the depth downstream from the hydraulic jump is constant for a long distance. What is this depth immediately downstream from the hydraulic jump? By carrying out computations for 20 and 40 s determine what happens to the hydraulic jump as the flow rate past the gate is increased. For each of these times, what is an estimate of the average rate of movement of the hydraulic jump? From these computations predict where and when an additional surge might occur.

6.96 A trapezoidal channel with b = 10 ft and m = 1.5 discharges into the ocean. No tidal action has kept the water level at its downstream end, and throughout the channel, constant at 5.0 ft for a long time. The corresponding uniform velocity is V_o = 3.0 fps. The tide now begins to "come in" increasing the downstream depth in the channel at a rate dY/dt = .005 ft/s. Determine the following: (a) when and where a surge might first form, (b) the discharge into the ocean at time t = 300 s, and (c) the discharge at 300 s at a position 500 ft upstream from the end of the river.

6.97 Water enters a rectangular channel that is 4 ft wide from a 12-in. diameter pipe as shown in the sketch below. After the bend in the pipe, there is a butterfly valve used to control the flow, and under steady-state conditions the pressure recorded by a gauge at the top of the pipe is 20 psi. At its end, the pipe is flowing full, and under these steady-state conditions, the minor loss coefficient for the combined valve and the pipe bend is K_1 = 1.5. The top of the pipe is 15 ft above the bottom of the channel, and the end of the pipe is 6 ft above the bottom of the channel. The transition and the bend below the end of the pipe has a minor loss coefficient of K_2 = 0.5.

What is the slope of the bottom of the channel if a hydraulic jump forms at a distance of 30 ft downstream from its entrance? The channel is made of concrete with a Manning's roughness coefficient n = 0.013.

An unsteady-state flow is created by closing the butterfly valve slowly so that the rate of discharge is decreased, such that dQ/dt = −0.05 cfs/s. Analyze and describe what will happen under this unsteady flow condition. Assume in this analysis, at least for the next few minutes, that the pipe at its end remains flowing full. What is the depth downstream from the hydraulic jump after 1 min from when the valve starts closing? At this 1 min time, what is the depth and the velocity in the channel immediately upstream from the jump? For this transient analysis, assume that $S_o - S_f$ can be ignored so the problem can be solved using straight C^+ characteristics.



Partial solution: To determine the discharge, the Bernoulli equation can be written between the top of the pipe where the pressure is measured to the bottom of the pipe, giving

$$\frac{20(144)}{62.4}+9+\frac{Q^2}{2gA^2}=2.5\frac{Q^2}{2gA^2} \quad \text{or} \quad 55.154=1.5\frac{Q^2}{2g(\pi/4)^2}$$

giving Q = 38.219 cfs. The depth at the beginning of the channel can now be determined under the assumption that this supercritical flow will widen down the transition with a minor loss coefficient of 0.5, as stated in the problem. (The loss might actually be much larger because a separation of the flow might take place. A model study of this situation may help determine what will actually happen.) Writing the specific energy equation between the end of the pipe and the beginning of the channel gives

$$6+\frac{Q^2}{2gA^2}=Y_2+1.5\frac{Q^2}{2g(bY_2)^2}$$

which upon solving gives $Y_2 = 0.224$ ft with associated $F_{r2} = 15.93$ and $E_2 = 28.476$ ft. Next, it is necessary to solve the GVF equation over a length of 30 ft to determine the depth of flow immediately upstream from the hydraulic jump. However, to explore what the flow situation is, let us assume that the jump occurs at the beginning of the channel. Then, the utilization of the momentum principle allows for the depth downstream from the hydraulic jump to be determined, or $Y_3 = (Y_2/2)[-1+(1+8F_{r2}^2)^{1/2}] = 4.92$ ft with $F_{r3} = 0.15$ and $E_3 = 4.979$ ft. Solving Manning's equation indicates that the slope of the channel bottom should be

$$S_o = 0.000181$$

Since the conjugate depth will be less than 4.92 ft, the actual slope of the channel must be larger. Continuing with the assumption let us examine what the flow conditions would be after 1 min has elapsed. At this time, the flow rate is $Q = 38.219 - 60(0.05) = 35.219$ cfs, and the uniform velocity and the celerity are $V_o = 1.942$ fps and $c_o = 12.287$ fps. From the C^- characteristic $V = 2c - 23.231$, and since $q = YV = c^2/g(2c - 23.231) = 8.805$, it is possible to determine what the celerity and the depth are in the subcritical region at the beginning of the channel. The solution of this equation gives $c = 12.520$ fps and $Y_3 = 4.868$ ft. Again, solving the specific energy equation across the transition gives

$$\frac{(35.219)^2}{64.4(\pi/4^2)}+6=37.224=Y_2+1.5\frac{(35.219)^2}{64.4(16)Y_2^2}$$

giving $Y_2 = 0.221$ ft and its conjugate depth $Y_3 = 4.56$ ft. Since this depth is less than that dictated by the unsteady flow in the downstream channel, or 4.868, the hydraulic jump will move upstream, and the pipe flow will impact upon it. The actual problem requires that this rate of upward movement be determined.

6.98 Spreadsheets are widely used in engineering practice, as well as in business and many other applications, and have been developed to allow their use to replace the use of programming languages. Develop a spread sheet solution for unsteady flow in trapezoidal channels (and possibly also circular channels). This spreadsheet should allow you to give the basic specifications of the problem, and then provide a complete solution giving the depth, velocities, and flow rates at a number of positions along the channel for a number of time increments. As part of the spreadsheet, you will need to generate the stage variable and use these values in solving the problem.

6.99 The equation that gives the position where a surge will first form due to positive waves is given in general by $x_s = (dx/dt)^2/d(dx/dt)/dt$. Derive the versions of this equation that apply for nonrectangular channels that give x_s and contain dY/dt, dV/dt, and dQ/dt for the three boundary conditions in which, (a) the rate of depth change with time is constant, (b) the rate of velocity change with time is constant, and (c) the rate of flow rate change with time is constant, respectively.

6.100 Solve Example Problem 6.17 (see Section 6.11) but rather than specifying the time rate of change of the unit flow rate, dq/dt, the gate is raised at a constant rate over 300 s for its initial setting to a position where it causes the depth downstream from it to vary linearly to 0.6 ft. In other words, the depth Y_2 is specified and begins with the same value $Y_2 = 0.2865$ ft as in that example problem and increases to 0.6 ft.

6.101 This problem is similar to the previous one in that it is a variation of the conditions specified for Example Problem 6.17. Now, rather than starting the depth downstream from the gate as in the example problem, the beginning and the ending depths are specified downstream from the gate. Let the beginning depth behind the gate be 0.25 ft, and the ending depth after 7 min (420 s) be 0.6 ft. As in the previous problem, and in Example Problem 6.17, find the depths and velocities: (1) upstream from the gate, (2) between the gate and the moving hydraulic jump, and (3) downstream from the jump. Also, compute the velocity of the jump's movement and the varying flow rate past the gate. Use 30 s increments to solve this problem.

6.102 In Example Problem 6.17, the gate is placed so as to produce and the given upstream velocity of 1 fps and a depth of 5 ft, and from these steady-state values the depth downstream from the gate is computed as 0.2865 ft. There after the rate of unit flow rate with time is specified as a constant, i.e., dq/dt is specified. Rather than specifying dq/dt, specify the beginning and ending downstream depths of water and solve the problem with starting downstream depth of 0.2865 ft and an ending depth of 0.6 ft downstream from the gate.

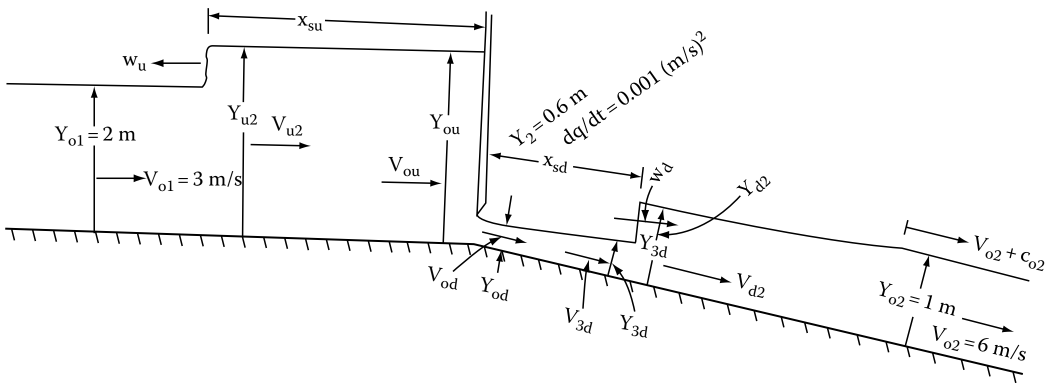
6.103 In Example Problem 6.20, the unsteady portion of the problem was solved using six equations to solve six unknowns that were Y_2, V_2, v, Y_o, V_o, and Y_d, and thereafter the position of the moving surge was determined. The determination changing values for the depth and the velocity Y_o, V_o immediately in front of the gate is not completely consistent with the method of the characteristics of this chapter. Obtain a solution to this problem in which a gate is at first instantly dropped, and thereafter slowly dropped to cause a constant dq/dt, by only solving the first four variables, namely, Y_2, V_2, v, and Y_o using only the first four equations used in solving the example problem. Note that in this solution, since the fifth and sixth equations are not used the solution doesn't result in a constant unit flow rate dq/dt past the gate.

6.104 A check on how well the theory of using the steady-state equation and the concepts with the "Method of Characteristics" account for the overall conservation of water involved is to compute the changes in the unit flow rate across the moving wave and to determine if this change inflow rate Δq equals the flow rate determined by the height of the wave times its velocity. For the situation in Example Problem 6.20 compute (1) the difference in the flow rate represented by the height of the wave times the velocity of the wave or $\Delta q_1 = v(Y_2 - Y_s)$, or $\Delta q_1 = v(Y_2 - 4)$, (2) the difference in the flow rate across the moving wave by $\Delta q_2 = V_sY_s - V_2Y_2$, or $\Delta q_2 = 20 - V_2Y_2$, and (3) by using the average of the depths immediately behind the moving wave and immediately in front of the gate, or $\Delta q_3 = V_sY_s - (V_2Y_2 + V_oY_o)/2$.

6.105 For some time, a gate has been set so as to produce uniform upstream and downstream depths of 2.8 and 0.6 m, respectively. The channel downstream of the gate is steeper than the channel upstream from the gate. The gate is instantly raised so as to produce a new downstream depth of $Y_2 = 1$ m. Determine the new depth Y_1 and the velocity V_1 upstream from the gate and the new velocity V_2 immediately downstream from the gate. Also, determine the velocity of the surge w, that will move downstream and the depth Y_s and the velocity V_s on the downstream side of the surge.

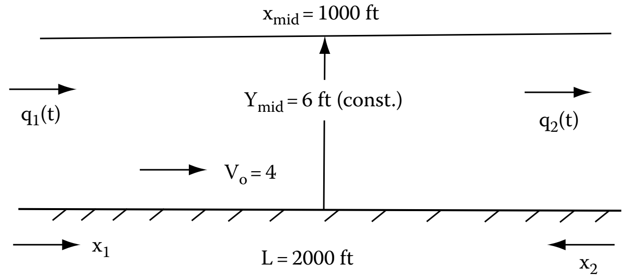
6.106 A gate, with a contraction coefficient $C_c = 0.6$ and a loss coefficient $K_L = 0.05$ exists at the position where a rectangular channel changes from $b_1 = 12$ ft, $S_{o1} = 0.0005$ to $b_2 = 8$ ft and $S_{o2} = 0.00773$. The ns for the upstream and downstream channels are the same and $n_1 = n_2 = 0.013$. For a long time, the flow rate has been Q = 600 cfs and the gate has been set to produce a uniform flow upstream. What is the height of the gate above the channel bottom? Suddenly, the gate is lowered by $\Delta Y_G = 2$ ft, and thereafter it is gradually closed so that the flow rate past the gate is reduced by 0.2 cfs/s, i.e., dQ/dt = −0.2 cfs/s. (1) Determine conditions upstream and downstream from the gate after the initial instant closure of the gate, and (2) for a period of 400 s thereafter, with time steps of 20 s, determine velocities of waves, depths, etc.

6.107 A uniform flow occurs both upstream and downstream from a gate in a rectangular channel with $q = 6\,m^2/s$. Upstream from the gate, the uniform depth is $Y_{o1} = 2\,m$, and the velocity is $V_{o1} = 3\,m/s$. Downstream from the gate, the uniform depth is $Y_{o2} = 1\,m$. The gate is then suddenly closed so as to produce a new depth immediately downstream from it of 0.6 m. Solve the depths and the velocities upstream and downstream from the gate, as well as the velocities of the hydraulic bores caused by this instant partial gate closure. Then, determine the unsteady conditions upstream and downstream from the gate over a 10 min (600 s) time period, with 30 s increments, if the unit flow rate past the gate continues to decrease with $dq/dt = 0.001\ (m/s)^2$, thereafter.

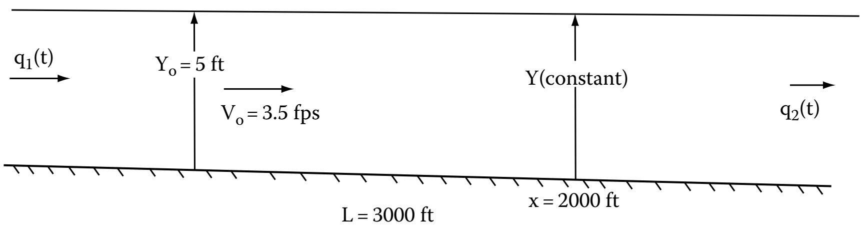


6.108 A section of a 2000 ft long channel between two gates has been flowing at a constant depth of $Y_o = 6$ ft and $V_o = 4$ fps for some time, when suddenly at t = 0, the discharge past the upstream gate $q_1(t)$ is increased at a constant rate of $dq_1/dt = 0.01\ (ft/s)^2$. Assume that the method of characteristics is applicable with $g(S_o - S_f) = 0$, and determine when the discharge at the downstream gate must first begin to be increased so that the depth at the midpoint of the channel (i.e., at $x_1 = 1000$ ft, $x_2 = L - x_1 = 1000$ ft) remains constant at Y = 6 ft. Assume that $dq_2/dt = dq_1/dt = 0.01\ (ft/s)^2$. Fill in the table below giving these discharges q_2 past the downstream gate, and the flow rates past the midpoint, q_{mid} for the times indicated.

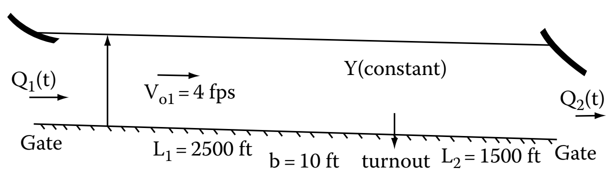
t (s)	q_1 (cfs/ft)	q_2 (cfs/ft)	q_{mid} (cfs/ft)
0	24.0		
30	24.3		
60	24.6		
90	24.9		



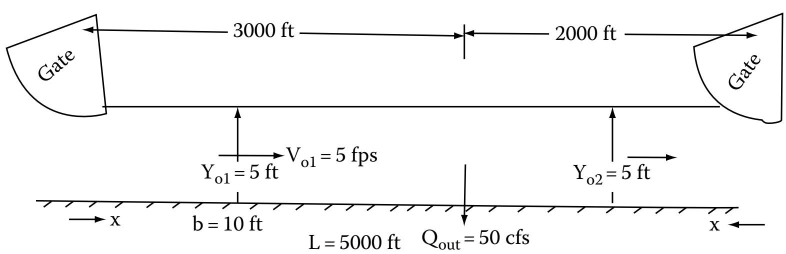
6.109 A uniform flow exists in a rectangular channel at a depth $Y_o = 5$ ft and $V_o = 3.5$ fps. Gates exist 3000 ft apart. It is desirable to maintain the depth constant at a position 2000 ft downstream from the upper gate. If the flow rate per unit width at the upstream gate is to be increased by $dq_1/dt = .015\ (ft/s)^2$, at what time must the flow rate at the downstream gate be increased to keep the depth constant at x = 2000 ft? Make up tables with a 20 s increment that show (a) the flow rate q, (b) the depth Y, and (c) the velocity V at the upstream gate, the downstream gate and at the position x = 2000 for about 160 s. (Make $dq_2/dt = dq_1/dt$).



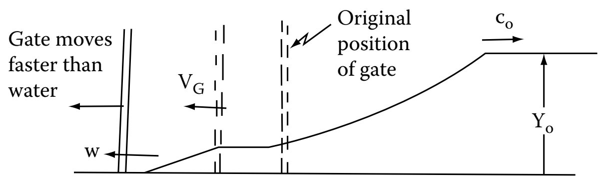
6.110 A turnout located 2500 ft downstream from an upstream gate is taking 100 cfs of water from a 10 ft wide rectangular channel. Another gate exists 1500 ft downstream from the turnout. The depth is to be maintained constant at the turnout, but more water is to be put into the channel. Assume that the depth is constant throughout the channel at Y_o = 5 ft, and is uniform both upstream and downstream of the turnout, with the velocity of V_{o1} = 4 fps upstream of the turnout. If at the downstream gate the rate of increase inflow rate is to be $|dQ/dt|$ = .15 cfs/s for 200 s, determine when more should be released at the upstream gate? (Make $dQ_1/dt = dQ_2/dt$).

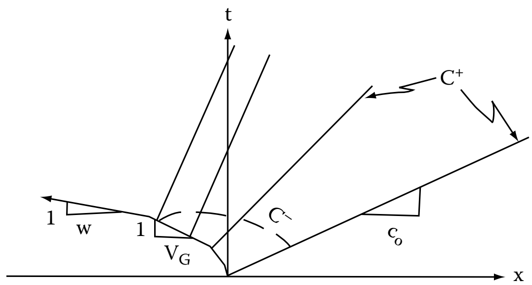


6.111 The depth of flow in a 10 ft wide rectangular channel is constant at Y_o = 5 ft throughout the channel between two gates that are 5000 ft apart. At a position 3000 ft downstream from the upstream gate there is a turnout that is taking 50 cfs from the channel. Upstream from the turnout the velocity is V_o = 5 fps. If the flow rate at the downstream gate is to be increased at a rate of dQ_2/dt = 0.25 cfs/s for 200 s, when must the flow past the upstream gate be increased (also at a rate of dQ_1/dt = 0.25 cfs/s) if the depth at the turnout is to remain constant? Make up tables that apply at the upstream gate, at the downstream gate and immediately upstream from the turnout that gives: (1) the flow rate Q, (2) the depth of flow, and (3) the velocity as a function of time. Use a 30 s time increment.



6.112 A rough estimate of the velocity (and the position) of the leading edge of the water from a dam break as a function of time can be obtained using the method of characteristics as developed in Chapter 6. Obtain this estimate of the velocity by assuming the following: (a) for a short time after the failure, the dam moves downstream at a velocity, w, holding the water back slightly so that it does not outrun the water from the reservoir; and (b) after a very short time, the gate moves so the depth of the leading edge of water approaches zero. Using this as a basis for a warning system, how much time is available to evacuate people from a town 10 miles downstream from a dam if the depth of water in the reservoir behind the dam is 70 ft. What depth and flow rate occur at the dam site after a short time? How fast will the effect of the failed dam move upstream in the reservoir? (Its velocity is initially at zero.) Hint: examine the sketch below. (1 mile = 5280 ft).





6.113 A wide rectangular channel 3000 ft long conveys water between two reservoirs whose water surface elevations are 1.5 ft different in elevation. At time t = 0, the flow is uniform throughout the channel at a rate of q = 20 cfs/ft. The channel's Manning's roughness coefficient is n = 0.013. At time t = 0, the downstream reservoir water level begins to drop at a constant rate dY/dt = −0.02 ft/s. Make up a table that gives estimates of the depth, the velocity and the discharge at the end and the beginning of the channel as a function of time. To obtain these estimates, use the method of characteristics until the effects of lowering the downstream reservoir start increasing the flow into the channel from the upstream reservoir. Note, the limiting flow rate will occur when a steady flow occurs with an M_2-GVF at a critical depth at the end of the channel. What is the difference in the channel storage between this final steady-state condition and when the upstream reservoir first begins to supply more water?

6.114 Solve the moving surge velocity and the position for the channel with the downstream gate of Example Problem 6.17 but with an initial depth of Y_s = 5 ft and a velocity of V_s = 5 fps. Initially, the gate is dropped instantly to the same extent as in the problem mentioned above, i.e., ΔY_d = .655 ft. The magnitude of the flow rate at the gate is changed by dq/dt = .05 (ft/s)^2, as in the example problem.

6.115 The same as in the previous problem but the magnitude of the change inflow rate at the gate of the initial drop is dq/dt = 0.075 (ft/s)^2.

6.116 A gate has been set so as to produce a uniform depth of Y_{o1} = 2.8 m and a velocity V_{o1} = 1.2 m/s upstream from it, and downstream from the gate the channel is steep so a uniform depth is also created. Suddenly, the gate is raised so that it produces a depth Y_2 = 0.8 m downstream from it. Determine the new depth and the velocity upstream from the gate, the velocity immediately downstream from the gate, the speed of the surge, and the depth and the velocity immediately upstream from the surge. What is the Froude number upstream and downstream of the surge that an observer moving with the wave would see?

6.117 A uniform depth of 1.0 ft occurs downstream from a gate that is passing a flow rate of 15 cfs/ft. Suddenly, the gate is raised so that the depth downstream from it is 2 ft. Under the assumption that a uniform flow also exists upstream from the gate before the gate is raised, and that the method of characteristics can be used, solve the depths and the velocities upstream and downstream of the gate, as well as the movement of the wave after the gate is raised.

6.118 The gate of the previous problem has been at the position where it produces a depth of 2 ft downstream from it when suddenly it is lowered to produce a depth of 1 ft downstream from it. Solve the depths and the velocities immediately upstream and downstream from the gate, and the velocities of the waves now. How is this problem different from Example Problem 6.17?

6.119 A gate in a rectangular channel has been operating so as to produce a uniform flow upstream and downstream from it for some time with the flow rate per unit width of $q_1 = 10\,m^2/s$ and a downstream depth of 1 m. Suddenly, the gate is lowered so as to produce a downstream depth of 0.5 m. Determine the new flow rate per unit width and the wave movements upstream and downstream of the gate as well as the associated depths.

6.120 The TK-Solver model GATEDW.TK used to solve Example Problem 6.22 solves seven equations for both the variables upstream and downstream from the gate. Make two models; one to solve the variables upstream from the gate; and the other to solve the variables downstream from the gate, and use these models to solve this example problem.

6.121 Write a computer program to solve problems for a rectangular channel in which a gate is partially and instantly raised and use this program to solve Example Problem 6.21 and Problems 6.107 and/or 6.108.

6.122 Write a computer program to solve problems for a rectangular channel in which a gate is partially and instantly lowered and use this program to solve Example Problem 6.22 and Problems 6.109 and/or 6.110.

6.123 Program GATETR is designed to solve problems associated with instantly partially closing a gate in a trapezoidal channel. Write a similar program that is designed to solve problems associated with instantly raising a gate in a trapezoidal channel. Use this program to solve the Example Problem 6.23 in which the gate has been set at the lower level for a long time, and is then instantly raised so that it now produces a downstream depth of $Y_2 = 4$ ft. (In other words, use the solution from Example Problem 6.23 as the initial conditions, and solve the new depths, etc., if the gate now causes the depth downstream from it to be 4 ft, from an initial depth of 2 ft.)

6.124 Modify the program you developed in the previous problem so that rather than specifying the new depth Y_2 downstream from the gate, the new velocity V_2 downstream from the gate, or the new flow rate Q_2, is specified. Use this program to solve the previous problem in which the flow rate is instantly increased to 150% of the original flow rate. Also, use this program to solve the previous problem in which the velocity V_2 is instantly decreased to 20.624 fps.

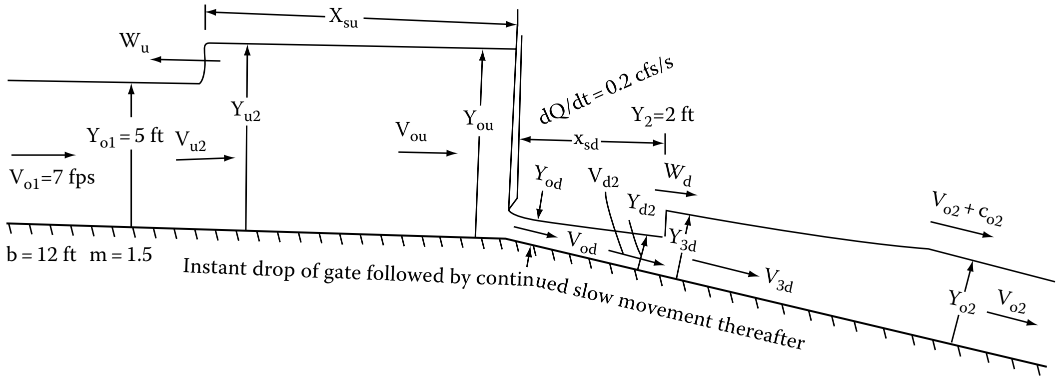
6.125 Modify program GATETR so that, rather than specifying the new depth Y_2 downstream from the gate, the new flow rate Q_2 past the gate is specified. Use this program to solve Example Problem 6.23 in which the new flow rate Q_2 is specified to equal one-half the original flow rate $Q_1 = V_{o1}A_{o1} = V_{o2}A_{o2}$. Also, solve the unsteady problem if the new velocity downstream from the gate is $V_2 = 25.032$ fps.

6.126 A gate in a trapezoidal channel with b = 5 m and m = 1.2 has been set so as to produce a depth of 6 m upstream of the gate when the flow rate is $Q = 95\,m^3/s$ for a long time. Suddenly, the gate is raised so as to produce a depth of $Y_2 = 2.5$ m downstream from the gate. Determine the new flow rate, the velocities upstream and downstream of the gate, the new depth upstream from the gate and the speed of the surge. Also, solve the problem if the flow rate is suddenly increased to $150\,m^3/s$.

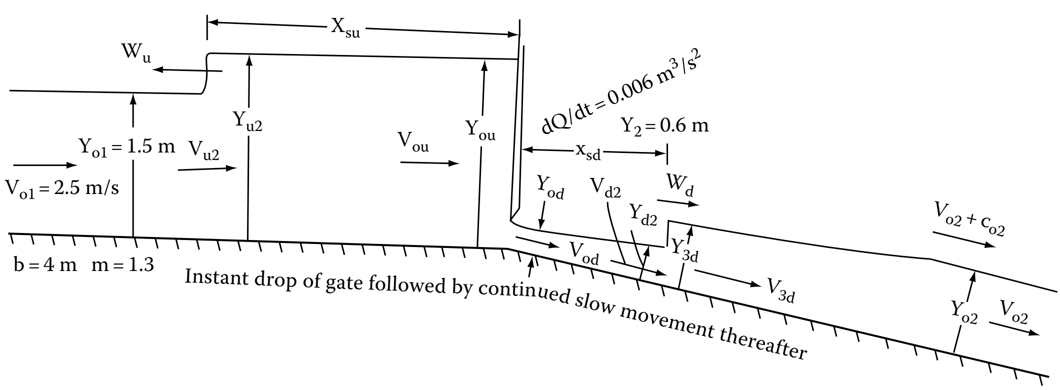
6.127 Program GATETR is designed to solve seven unknown variables associated with the problem in which a gate is instantly lowered in a trapezoidal channel. Modify this program so that it solves this type of problem in a circular channel. With your modified program, solve the new upstream and downstream depths and velocities Y_1, V_1, Y_3, and V_3, the new velocity V_2 past the gate, and the wave speeds w_u and w_d, if with the old gate setting in a 12 ft diameter pipe the upstream depth is $Y_{o1} = 9$ ft and the downstream depth is $Y_{o2} = 4$ ft, and the gate is instantly lowered to a certain extent so as to produce a depth downstream from it of $Y_2 = 3$ ft.

6.128 Initially, a uniform flow exists both upstream and downstream from a gate in a trapezoidal channel with a bottom width b = 12 ft, and a side slope m = 1.5. The upstream depth and velocity are: $Y_{o1} = 5$ ft, and $V_{o1} = 7$ ft/s. If the gate is instantly dropped so that it produces 2.0 ft downstream from it, determine what the flow conditions are at this time.

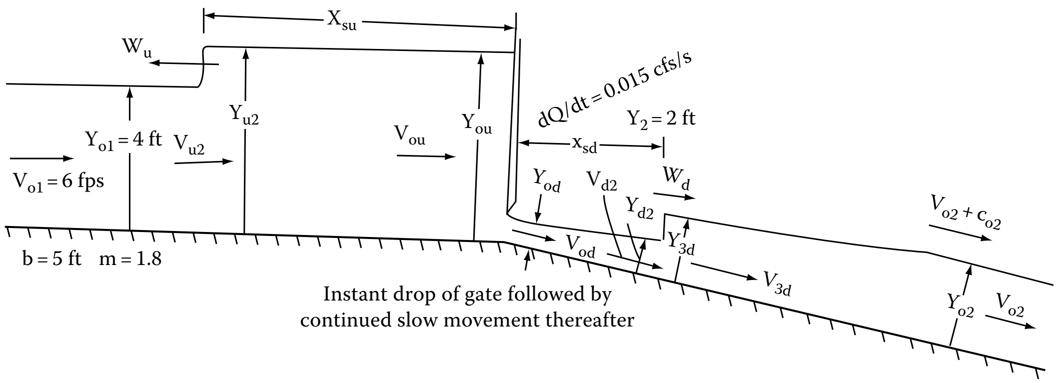
6.129 A uniform flow occurs both upstream and downstream of a gate in the trapezoidal channel of the previous problem, i.e., b = 12 ft, m = 1.5 with an upstream depth of $Y_{o1} = 5$ ft and a velocity $V_{o1} = 7$ fps. Suddenly, the gate is dropped so that it produces a depth of $Y_2 = 2$ ft downstream from it; however, after this instant drop of the gate, the gate continues to drop gradually so that the flow rate is reduced by dQ/dt = 0.2 cfs/s. Using 20 s time steps solve the conditions upstream and downstream from the gate over a 400 s time period.



6.130 Prior to the partial instant gate closure in a trapezoidal channel with a bottom width of b = 4 m and a side slope m = 1.3, uniform depths exist both upstream and downstream from a gate. The depth and velocity upstream from the gate are $Y_{o1} = 1.5$ m and $V_{o1} = 2.5$ m/s. Suddenly, the gate is partially closed so that the depth downstream from the gate is $Y_2 = 0.6$ m. Thereafter, the gate's position is gradually adjusted so that the flow rate past it increases at a rate of dQ/dt = 0.006 m^3/s^2. Solve the upstream and downstream depths and velocities, as well as the velocities of the hydraulic bores resulting from this gate movement for a period of 600 s in 20 s increments.



6.131 A trapezoidal channel with a bottom width b = 5 ft and a side slope of m = 1.8 has a gate near the middle that is set just right so that a uniform flow occurs both upstream and downstream from the gate for a long time. The channel has a steeper slope downstream from the gate than upstream therefrom. The uniform depth and velocity upstream from the gate are: Y_{o1} = 4 ft and V_{o1} = 6 fps. If the loss coefficient of the gate is zero, what will the uniform depth and velocity downstream from the gate be? Suddenly, the gate is partially closed so as to produce a depth of 2 ft downstream from it. What are the depths and the velocities immediately upstream and downstream from the gate just after it is partially closed? What are the velocities of the hydraulic bores upstream and downstream from the gate that this partial closure causes? Thereafter, the gates position is adjusted so that the flow rate is changed with dQ/dt constant and equal to dQ/dt = 0.15 cfs/s. Determine the depths and the velocities both upstream and downstream from the gate through a period of 500 s in 20 s increments.



6.132 Obtain a series of solution to the previous problem in which the rate of flow rate change dQ/dt varies from 0.05 to 0.23 cfs/s in increments of 0.002 cfs/s. Carry out these solutions for a maximum time of 600 s, or until the Froude numbers upstream and downstream from the hydraulic bore upstream from the gate approach unity as seen by a moving observer, i.e., the height of this bore approaches zero and this moving observer sees a critical flow both upstream and downstream from where he is.

6.133 Initially, a flow rate of 25 m^3/s occurs in a circular channel with a diameter D = 4 m upstream of a gate at a depth of Y_{o1} = 2.5 m. The gate is suddenly lowered to produce a depth of Y_2 = 1.3 m. Determine the new depths and velocities upstream and downstream from the gate, as well as the speeds of the waves upstream and downstream from the gate.

6.134 Downstream from a gate in a trapezoidal channel with b = 10 ft and m = 1.5, the depth initially is 3 ft and the steady-state flow rate is Q = 600 cfs. Suddenly, the gate is raised so as to produce a new downstream depth of Y_2 = 3.5 ft. Determine the new depth and velocity upstream from the gate, the new velocity of flow past the gate, the depth Y_3 and V_3, as well as the speed at which the wave moves w.

6.135 Develop a computer program, or model, that can solve problems in which a gate is instantly raised to some extent in a circular channel. With this program, solve the following problem: A 12 ft diameter channel has a depth Y_{o1} = 10 ft upstream from a gate and Y_{o2} = 3 ft downstream from this gate. The gate is suddenly raised so the new downstream depth is Y_2 = 3.5 ft. (a) Find the new upstream and downstream depths and the speed of the downstream wave. (b) Repeat the previous problem but raise the gate so that the new downstream depth is Y_2 = 4.0 ft. (c) Repeat the problem, but initially the downstream depth is Y_{o2} = 2.5 ft (rather than 3.0 ft), and the gate is raised so Y_2 = 3.5 ft.

6.136 Program GATETR which solves the seven unknown variables associated with the partial instant closing of a gate in a trapezoidal channel requires that the problem is specified by giving the initial upstream and downstream depths and velocities, Y_{o1}, V_{o1}, Y_{o2}, V_{o2}, and the new depth Y_2 caused by the partial closure of the gate. Often, flow rates are known rather than velocities. Generalize Program GATETR so that it will accommodate the following for cases of specifying the initial conditions: (1) Y_{o1}, V_{o1}, Y_{o2}, and V_{o2} are given, (2) Y_{o1} and Y_{o2} are given, (3) Y_{o1} and Q_o are given, or (4) Y_{o2} and Q_o. Also, for the new conditions, allow any of the following three specifications: (a) the new depth Y_2 downstream from the gate is given, (b) the new velocity V_2 downstream from the gate is given, or (c) the new flow rate Q_2 past the gate is given. With this program solve Example Problem 6.23 four times, using the four possible specifications for the initial conditions, all by giving $Y_2 = 2$ ft. Also, solve this example problem but rather than specifying Y_2, specify that the partial closure of the gate reduces the flow rate from $Q_o = 1314.92$ cfs to $Q_2 = 800$ cfs, past the gate.

6.137 The more generalized program you developed in the previous problem is designed to solve problems in which an unsteady situation is created by an instant partial closure of a gate in a trapezoidal channel. Develop a similar generalized program that handles the same four initial condition cases, and the same three new condition cases but is designed to solve problems in which the unsteady flow is caused by the instant partial raising of the gate. In other words, generalize the program you developed in Problem 6.136. Using this program, solve Problem 6.136 four different time using the four cases to specify the initial conditions. For these four solutions, use the new condition of specifying $Y_2 = 4$. Obtain two additional solutions to this problem in which the fourth case for initial conditions is used, namely, giving Y_{o2} and Q_o, the first of which specifies a new flow rate of $Q_2 = 1200$ cfs, and the second, a new velocity $V_2 = 22$ fps.

6.138 Write a program similar to the program in Problem 6.50 that solves the C^- characteristic in the xt-plane of a circular channel, except in place of specifying dY/dt, specify the rate of change of the flow rate at the origin, or dQ/dt. For a circular channel with D = 12 ft, and uniform conditions of $Y_o = 5$ ft and $Q_o = 300$ cfs, and dQ/dt = −1.0 cfs/s, find the C^- characteristics through the t-axis at t = 100, 200 and 230 s.

6.139 A rectangular channel initially contains a uniform flow with a velocity $V_o = 1.5$ m/s and a depth of $Y_o = 2$ m. Determine the maximum point unit outflow Δq that can be taken from an intermediate position along this channel. What fraction of this maximum outflow comes from the downstream reverse flow? (Solve this problem using the explicit equations that give Δq_{max} and c_c, and then also solve the problem by using pairs of equations given in the text based on the downstream flow rate and the upstream velocity, as well as the equation that gives Δq.

6.140 A trapezoidal channel with a bottom width b = 10 ft and a side slope m = 1.0 initially contains a flow rate of Q = 580 cfs. Initially, a uniform flow exists both upstream and downstream from a gate when the gate is instantly and partially closed so as to reduce the flow rate by 80% of the initial flow rate, i.e., 0.8(580). If the velocity immediately downstream from the gate before it was partially closed further was $V_{o2} = 15$ fps, determine the new depths and velocities both upstream and downstream from the gate. After this instant partial closure of the gate, it is closed slowly thereafter so that the flow rate change is dQ/dt = 0.2 cfs/s.

6.141 For a long time, a uniform flow existed both upstream and downstream from a gate in a circular channel with a diameter D = 15 ft. The flow rate under these uniform flow conditions is $Q_o = 400$ cfs, and the velocity downstream from the gate is $V_{o2} = 15$ fps. Determine the uniform depth and velocity upstream from the gate. Suddenly, the gate is partially closed so that the flow rate is reduced to 80% of Q_o, or Q_d past the gate equals 320 cfs. Determine the new depths upstream and downstream from the gate immediately after the gate's position has changed; also determine the associated velocities of the hydraulic bores in this circular channel, and the depth and the velocity downstream from the downstream moving hydraulic bore. After this initial partial closure of the gate its position is adjusted so that the flow rate is gradually reduced in time to dQ/dt = 0.2 cfs/s. Determine the conditions both upstream and downstream from the gate for a period of 400 s in increments of 20 s.

6.142 Both upstream and downstream from a gate in a circular channel with a diameter D = 5 m, the flow is uniform. The flow rate under this steady state uniform conditions is $Q_o = 12\,m^3/s$, and the velocity downstream of the gate is $V_{o2} = 6\,m/s$. The gate is suddenly partially closed so that the flow rate is reduced to 75% of the uniform amount or to $Q_d = 9\,m^3/s$. Thereafter, the flow rate is gradually decreased with $dQ/dt = 0.006\,m^3/s^2$. First, solve the uniform condition upstream from the gate, second, solve the conditions upstream and downstream from the gate immediately after the gate is partially closed, and third, obtain the solution over a 400 s period, with a time step of 20 s, for the wave condition that occurs thereafter as the flow rate is gradually decreased.

6.143 A 15 ft diameter pipe contains a gate that has been set for a long time so as to produce a uniform flow both upstream and downstream from it. The upstream uniform depth is $Y_{o1} = 8.5$ ft, and the upstream velocity is $V_{o1} = 3$ fps. Suddenly, the gate is partially closed so that the depth downstream from it is 1.1 ft. Do the following: (1) compute the downstream uniform depth and velocity, (2) determine what happens upstream and downstream from the gate immediately after the partial instant closure, and (3) after the instant partial closure the gate's height is gradually adjusted so that the flow rate continues to decrease at a constant rate of dQ/dt = 0.1 cfs/s; now determine the upstream and downstream conditions including the positions of the hydraulic bores for 400 s in 20 s time increments.

6.144 A pipe with a diameter D = 5.5 m initially has uniform flow both upstream and downstream from a gate, such that $Y_{o1} = 4$ m and $V_{o1} = 1.5$ m/s. The gate is instantly partially closed so that the depth immediately downstream from it is 0.7 m. Determine the conditions upstream and downstream from the gate after the instant partial closure, as well as for 400 s, with 20 s time steps, if the gate gradually closes thereafter so that $dQ/dt = 0.005\,m^3/s^2$.

6.145 A gate has been closed in a rectangular channel and produces a depth of 8 ft upstream therefrom. Downstream from the gate, the channel is dry. Suddenly, the gate is completely opened. Determine how fast the water moves downstream in the dry channel, and how fast the effect of the gate propagates upstream. What is the flow rate per unit width past the gate site?

6.146 Water is being retained behind a gate in a flat rectangular channel to a constant depth of 5 m. There is no water in the channel downstream from the gate. The gate is suddenly and completely opened. Determine the following: (1) the speed at which the water move downstream in the dry channel, (2) the speed at which the effect of opening the gate will propagate upstream, (3) the depth of water at the gate site, and (4) the flow rate past the gate site.

7 Numerical Solution of the St. Venant Equations

7.1 BACKGROUND

This chapter will continue the subject of unsteady open channel flows that was begun in Chapter 6. In Chapter 6 solutions to the St. Venant equations were obtained by making simplifying assumptions so that the C^+ characteristic through the origin of the xt plane was a straight line, i.e., a uniform flow existed in the channel, at time t, and the difference between the slope of the channel bottom and the energy line throughout the length of the channel, and through the time period of the solution were assumed to be equal, or $S_f = S_o$. These assumptions allowed solutions to be obtained from relatively simple algebraic equations that applied along the characteristic lines in the xt plane. These assumptions are restrictive, and rarely duplicate real world unsteady flows. Available "closed-form" solutions to the St. Venant equations are very limited, and not of great practical use. Thus to obtain unsteady solutions to various flow conditions that arise in open channel hydraulics we must turn to numerical methods. This chapter deals with numerical solutions of the St. Venant equations. The intent of this chapter is not to give an exhaustive treatment of the subject. The literature on this subject is vast and would fill volumes to describe the variety of, and large number of, techniques that have been proposed and used. Rather the approach is to provide the reader with a basic working knowledge of numerical methods as they apply to solving hyperbolic partial differential equations such as the continuity equation and equation of motion for unsteady open channel flows that we call the St. Venant equations after the French Engineer De Saint-Venant who proposed them in 1871. There is probably no "best" method for obtaining numerical solution of these equations. Indeed the best method for one problem may prove deficient for another problem. There is no substitute for a good knowledge on the part of the engineer who must obtain solutions to unsteady problems. It is hoped that readers of this text are not found in the camp of engineers who accept results from a computer just because it supplied answers in response to some input. Life is not so simple, especially when dealing with unsteady channel flows.

When solving problems numerically, it must be keep in mind that the solutions can deviate from the real life flows due to many causes, among which are as follows: (1) The accuracy with which the numerical solution solves the mathematical problem governed by the differential equations, and boundary and initial conditions. (2) How accurately the differential equations describe the real processes. (3) How closely the boundary conditions and initial condition describe the problem. (4) How close the parameters, or variables used to define the problem agree with these values in the real processes. In considering these limitations, it is apparent that numerical solutions are certainly not the answer to all problems, and cannot handle all situation, at least not without giving special considerations to what is important in a given situations, and whether the numerical solution addresses these features in an optimal manner. In other words, just because the computer may produce a solution, it may not represent the real unsteady flow that you are modeling.

The question of how accurate a numerical solution may obtain the solution to the initial boundary value problem will be a major topic of the remainder of this chapter. As we begin this chapter, it is well to review what assumptions the St. Venant equations are based upon in addressing the question of how accurate do the differential equations describe the real processes. Fundamental assumptions underlying the St. Venant equations when applied to one-dimensional flows are

1. That one-dimensional hydraulics applies, e.g., wave surfaces vary gradually, pressure distribution are hydrostatic in the vertical, and do not vary in the direction transverse to the direction of flow, the average velocity can be used in place of velocity profiles in the vertical direction, and no vertical or horizontal accelerations are involved.
2. The processes by which fluid frictional losses occur are identical to those for steady-state flows, and therefore the uniform flow equations can be used to compute these locally. In other word, the Manning's or Chezy equation can be used to determine S_f in all positions, and at all times.
3. The variations of depth and velocity across the channel do not affect the propagation of waves, e.g., there is one wave propagation speed that applies for the entire channel cross section. (This speed will generally be different at different cross sections, e.g., x positions.)
4. The slope of the channel bottom at all positions is small enough that S_o, which is usual taken as the tangent of the angle between the channel bed and the horizontal, is equal to the sine of this angle, and equals the angle itself.

When dealing with unsteady flows in basins, storage ponds, or agricultural fields being irrigated, then the flow is two-dimensional. For such two-dimensional flows, the St. Venant equations become three in number; the continuity equation in the x y-coordinate plane (horizontal), and two equations of motion, one in the x-coordinate direction and one in the y-coordinate direction. Two-dimensional unsteady flows will not be dealt with in this text, but the methods described can be extended to such unsteady flow problems.

The reader should be aware that the subject in this chapter is restricted to numerical solutions of the St. Venant equations. There are a number of other methods being used to solve problems associated with unsteady flows in channels, rivers, and streams, including the routing of flood flows through reservoirs. In order to quantify damages resulting from failure of dams, methods and procedures have been developed to simulate the failure mechanism of the dam, quantify the discharge past the dam site, and route the resulting flood in the downstream channel. Other applications involve quantifying the overland flow on watersheds resulting from time and spatially varying rainfall minus infiltration into and then through a network of streams that combine to produce a flood hydrograph. Broadly speaking, these methods can be divided into the following categories: (1) Those that attempted to solve the full dynamic equations of motion numerically, i.e., the St. Venant equations, the subject of this chapter. (2) Those based on kinematic waves theory, which basically omits the equation of motion from the St. Venant equations and uses the continuity equation along with a hydraulic resistance equation such as the Manning's or the Chezy equation. (3) Storage routing techniques, which only use some simplified form of the continuity equation.

To illustrate the simplification possible consider storage routing techniques that are based on the continuity equation stated in the form of the inflow I minus the outflow O equals the rate of change of storage in a length of channel, or

$$I - O = \frac{ds}{dt} = \frac{\Delta S}{\Delta t}$$

in which ΔS is the change in the volume of storage in the time Δt. The "Muskingum" method, initially developed in the 1930s in connection with the design of flood protection in the Muskingum river basin, Ohio, and later enhanced to Muskingum–Cunge, and other methods, has been used widely throughout the world. Briefly in this method the storage S is assumed to be a linear function of the inflow and outflow, or S = K[wI + (1 − w)O], in which K is a storage coefficient with dimensions of time, and w is a weight factor between 0 and .5. Discretizing the above equation from time 1 to time 2 and writing the storage equation for these two times gives

$$I_1 + I_2 - O_1 - O_2 = \frac{2(S_2 - S_1)}{\Delta t},$$

where

$$S_1 = K[wI_1 + (1 - w)O_1$$

$$S_2 = K[wI_2 + (1 - w)O_2$$

Upon substituting the latter two equations into the first gives the outflow as at time 2, O_2, as a linear function of the inflows at the beginning and end of the increment, I_1 and I_2 and the outflow at the beginning of the time increment O_1, or

$$O_2 = C_1 I_1 + C_2 I_2 + C_3 I_3$$

in which $C_1 = (\Delta t/K + 2w)/[\Delta t/K + 2(1 - w)]$, $C_2 = (\Delta t/K - 2w)/[\Delta t/K + 2(1 - w)]$ and $C_3 = (\Delta t/K + 2(1 - w))/[\Delta t/K + 2(1 - w)]$. Since $C_1 + C_2 + C_3 = 1$, these can be interpreted as weighting coefficient. Thus given an inflow hydrograph that provides the I's, an initial flow condition that provides the starting O_1, the routing parameters K and w simple computations, which can be carried out by hand, produce the outflow hydrograph.

While such simplified methods as Muskingum routing have been widely used in the past, they are not included in this book first because it must be limited in scope, but also because with the availability of computers their use is more difficult to justify, even in situations for which limited data is available about channel (or river) properties, inflows, etc.

7.2 METHOD OF CHARACTERISTICS

Chapter 6 used characteristics in the xt plane. It is logical to beginning the subject of numerical solutions based also on these characteristics. Obtaining numerical solutions at intersections of C^+ and C^- characteristics will enhance your understanding of the method of characteristics because as part of the solution you will be locating the x and t coordinates of these intersections. Now the C^+ characteristics are not straight lines because $Y \neq Y_o$ and $S_o \neq S_f$. It will be more expedient to now take the positive x axis in the direction of the channel flow even for problems in which the unsteady flow is caused by a change at the downstream end of the channel because the depths and positions can be read directly from solutions to gradually varied flows to provide the initial condition for the unsteady problem. Since the positions of the C^+ and C^- characteristics will be determined numerically, there is no advantage in having the first C^+ characteristic pass through the origin from which the "zone of quiet" is determined. The C^+ characteristics that will be worked along will start from the fixed points along the x-axis rather than up the t-axis. Furthermore, the channel would not be assumed to extend to infinity in one direction, but be limited in length by both an "upstream" and "downstream" boundary condition. Thus the characteristics for a typical unsteady subcritical flow problem will have a set of C^+ and C^- characteristics as shown in Figure 7.1. The numbering of the intersections in this xt plane will be discussed later since it is a vital component in writing computer code in obtaining a numerical solution. Note this numbering, as shown on Figure 7.1, reflects the region in the xt plane to where the effect has propagated to if the unsteady flow is solely due to a disturbance at the downstream end if steady

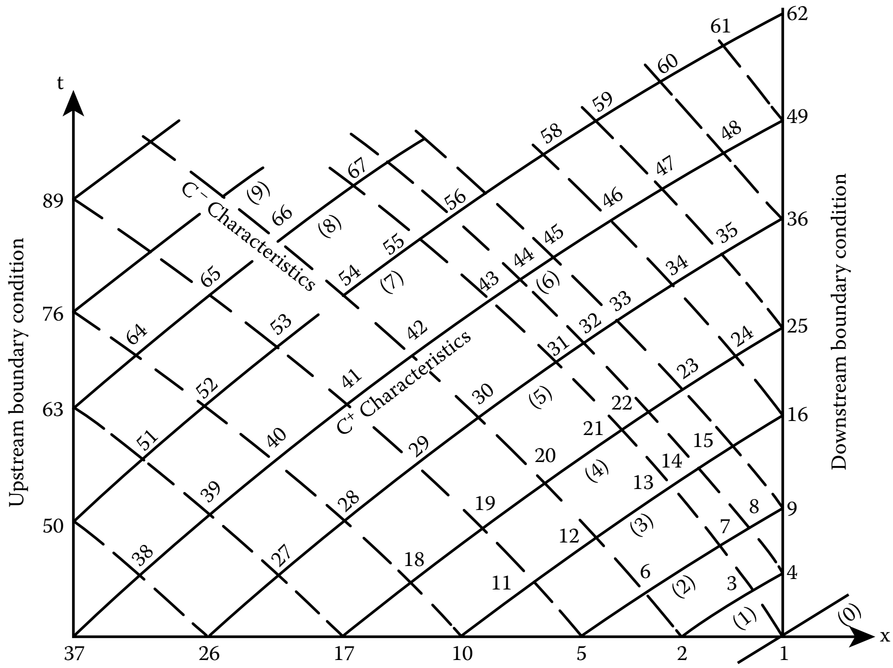


FIGURE 7.1 Intersection of C^+ with C^- characteristic lines to define grid points at which unknown will be solved.

state flow exists initially. Figure 7.1 displays a few selected characteristics of the infinite number that exist within the time covered by the figure. Recognize that at time zero, when the disturbance first occurs, there are zero characteristics from it and the number of C^+ and C^- increase in number with time starting at the lower right hand corner of Figure 7.1.

The form of the St. Venant equations that will be selected are given as Equations 6.3 and 6.4 in Chapter 6. These are as follows:

$$A\frac{\partial V}{\partial x} + VT\frac{\partial Y}{\partial x} + V\frac{\partial A}{\partial x}\bigg|_{Yt} + T\frac{\partial Y}{\partial t} = q^* \tag{7.1}$$

and

$$V\frac{\partial V}{\partial x} + g\frac{\partial Y}{\partial x} + \frac{\partial V}{\partial t} = g(S_o - S_f) - gF_q \tag{7.2}$$

Combining these equations in the characteristic form as was done to get Equation 6.28 by multiplying Equation 7.1 by $\{g/(AT)\}^{1/2}$ and then adding and subtracting the result from Equation 7.2 gives

$$(V \pm c)\frac{\partial V}{\partial x} + \frac{\partial V}{\partial t} \pm \sqrt{\frac{gt}{A}}\left((V \pm c)\frac{\partial Y}{\partial x} + \frac{\partial Y}{\partial t}\right) = R_L \text{ or } R_R \tag{7.3}$$

in which the right side of the equal sign is given by

$$R_L = g(S_o - S_f) - \sqrt{\frac{gT}{A}}\left(V\frac{\partial A}{\partial x}\bigg|_{Yt} - q^*\right) - gF_q$$

$$R_R = g(S_o - S_f) + \sqrt{\frac{gT}{A}}\left(V\frac{\partial A}{\partial x}\bigg|_{Yt} - q^*\right) - gF_q$$

R_L applies in Equation 7.3 associated with the + sign on the left side of the equal sign, and R_R applies when the – is used on the left of the equal sign. Note that if there is no lateral inflow or outflow that the terms involving F_q and q^* disappear from the definition of R_L and R_R. Likewise for prismatic channels the terms involving the partially derivative of A with respect to x are zero. Thus for many situations, R_L or R_R equals the difference between the slope of the channel bottom and the slope of the energy line multiplied by the acceleration of gravity, g, and furthermore $R_L = R_R$ for these situations.

If the stage function is defined as in Chapter 6 by $w = \int_e^Y \sqrt{gT/A}\,dy$, then the above equation becomes

$$\frac{\partial(V \pm w)}{\partial t} + (V \pm c)\frac{\partial(V + w)}{\partial x} = R_L \text{ or } R_R \tag{7.4}$$

The equation above with the plus sign is associated with the positive characteristics

$$\frac{dx}{dt} = V + c \tag{7.5}$$

$$\left[(V + c)\frac{\partial}{\partial x} + \frac{\partial}{\partial t}\right]\left[V + w\right] = \frac{d(V + w)}{dt} = R_L \tag{7.6}$$

and the equation with the negative sign is associated with the negative characteristics, or

$$\frac{dx}{dt} = V - c \tag{7.7}$$

$$\left[V - c\right]\frac{\partial}{\partial x} + \frac{\partial}{\partial t}\left[V - w\right] = \frac{d(V - w)}{dt} = R \tag{7.8}$$

For rectangular channels w = 2c. The interpretation of Equations 7.5 through 7.8 is as follows: A special ordinary differential equation, Equation 7.6, applies along the curve defined by Equation 7.5 in the xt plane. Likewise Equation 7.8 represents a special ordinary differential equation that applies along the curve defined by Equation 7.7 in the xt plane. The solution can be obtained by solving the ordinary differential Equations 7.6 and 7.8 along these curves. These curves are also defined by ordinary differential equations (ODEs) Equations 7.5 and 7.7. The solutions to these ODEs will be denoted as integrations below.

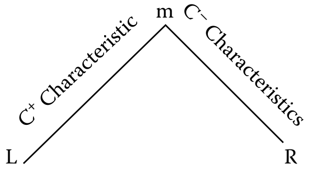


FIGURE 7.2 Characteristics uses to solve values at new grid m from values at known grids L and R.

Consider any three points with the relationships shown in Figure 7.2 with L and m defining a small segment of a C^+ characteristic and R and m defining a small segment of a C^- characteristic.

The positions x and t as well as the dependent variables V, c, w, and Y are known at L and R from the initial condition at t = 0 and subsequently from previous solutions as the solution advances through each new time step. These values (V, c, w and Y) are all sought at m. Integrating Equations 7.5 through 7.8 give

$$x_m = x_L + \int_{t_L}^{t_m} (V + c)dt \tag{7.9}$$

$$V_m + w_m = V_L + w_L + \int_{t_R}^{t_m} R_L dt \tag{7.10}$$

$$x_m = x_R + \int_{t_R}^{t_m} (V - c)dt \tag{7.11}$$

$$V_m - w_m = V_L - w_L + \int_{t_R}^{t_m} R_R dt \tag{7.12}$$

The following five unknowns occur in Equations 7.9 through 7.12: x_m, t_m, V_m, w_m, Y_m. One might make this six unknowns by including c_m since it must be computed to advance ahead. However, there is always an algebraic equation that relates c to Y. The fifth equation needed to solve for the five unknowns is the equation that defines the stage function w as a function of the depth of flow and the geometry of the channel, $w = \int_e^Y \sqrt{gT/A}dy$. For a rectangular channel w can be replaced by 2c and then the fifth equation is $Y = c^2/g$, or alternatively w_m might be removed from the list of unknowns with c defined from Y whenever needed by the equation $c = (gY)^{1/2}$.

The integrations in Equations 7.9 through 7.12 are commonly carried out using the trapezoidal rule. (Or if you wish to think of the integration as solving an ordinary differential equation, then only the Euler corrector is applied.) If higher order numerical methods were used it would be necessary to involve previous values along the characteristics making the numerical procedures much more complex. Using the trapezoidal rule results in the following four equations in place of Equations 7.9 through 7.12:

$$F_1 = x_m - x_L - \frac{1}{2}(t_m - t_L)(V_m + c_m + V_L + c_L) = 0 \tag{7.13}$$

$$F_2 = V_m + w_m - V_L - w_L - \frac{1}{2}(t_m - t_L)(R_{Lm} + R_{LL}) = 0 \tag{7.14}$$

$$F_3 = x_m - x_R - \frac{1}{2}(t_m - t_R)(V_m - c_m + V_R - c_R) = 0 \tag{7.15}$$

$$F_4 = V_m - w_m - V_R + w_R - \frac{1}{2}(t_m - t_R)(R_{Rm} + R_{RR}) = 0 \tag{7.16}$$

The fifth equation needed in general can be written as

$$F_5 = w_m - \left(w = \int_{\varepsilon}^{Y} \sqrt{\frac{gT}{A}} dY \right) = 0 \tag{7.17}$$

which can be implemented by utilizing a "table look-up" technique to define the relationship between the depth Y and geometric properties of the channel to the stage variable w. For a rectangular channel this fifth equation is

$$F_5 = Y - \frac{c^2}{g} = 0 \tag{7.17a}$$

Equation 7.13 through Equation 7.17 represent a system of 5 nonlinear equations. The Newton method will be used to solve these. If $\{\mathbf{x}\}$ is the unknown vector $\{\mathbf{x}\} = [x\ t\ V\ w\ Y]_m^T$ then the iterative equation

$$\{x\}^{(j+1)} = \{x\}^{(j)} - \{z\}^{(j)}$$

in which the vector $\{\mathbf{z}\} = [z_1\ z_2\ z_3\ z_4\ z_5]^T$ is the solution vector from solving the linear system of equations

$$[D]\{z\} = \{F\} = [F_1\ F_2\ F_3\ F_4\ F_5]^T$$

in which D is the Jacobian. Its elements can be evaluated numerically, or algebraically. The rows in the matrix correspond to the equations and the columns are the derivatives associated with the unknowns.

The implementation of the above procedure for solving for the unknowns x_m, t_m, V_m, w_m, Y_m at the advanced intersection of C^+ and C^- characteristics from known values at the point L and R in the xt plane is illustrated in the FORTRAN subroutine REGUL (named for regular point) below.

Subroutine to solve for unknowns to advance solution to new point in xt plane.

```
      SUBROUTINE REGUL(IL,IR,I)
      PARAMETER (NEQS=5)
      COMMON X(200),T(200),V(200),W(200),Y(200),C(200),SF(200),
     &Z(5),Va(5),F(5),F1(5),D(5,5),FMS,FM2,CN,FN,Q,SO2,B,FM,G,FL,
     &H,EK1,IGIV,MGIV,ITYPE,GIVDWS(10),GIVTIM(10),DELT
      NCT=0
      Va(1)=.5*(X(IL)+X(IR))
      Va(2)=.5*(T(IL)+T(IR))+DELT
      Va(3)=.5*(C(IL)+C(IR))
      Va(4)=.5*(V(IL)+V(IR))
      Va(5)=.5*(Y(IL)+Y(IR))
15    CALL FUNCT(0,IL,IR,DV)
      DO 20 J=1,NEQS
20    F1(J)=F(J)
      DO 22 J=1,NEQS
      CALL FUNCT(J,IL,IR,DV)
      DO 22 J1=1,NEQS
22    D(J1,J)=(F(J1)-F1(J1))/DV
      DO 30 L=1,NEQS-1
      DO 30 M=NEQS,L+1,-1
```

```
      IF(ABS(D(M,L)).LT. 1.E-15) GO TO 30
      FAC=D(M,L)/D(L,L)
      F1(M)=F1(M)-FAC*F1(L)
      DO 28 J=L+1,NEQS
28    D(M,J)=D(M,J)-FAC*D(L,J)
30    CONTINUE
      M=NEQS
      Z(M)=F1(M)/D(M,M)
      DIF=ABS(Z(M))
      Va(NEQS)=Va(NEQS)-Z(M)
31    M1=M-1
      SUM=0.
      DO 32 J=M,NEQS
32    SUM=SUM+Z(J)*D(M1,J)
      Z(M1)=(F1(M1)-SUM)/D(M1,M1)
      M=M1
      Va(M)=Va(M)-Z(M)
      DIF=DIF+ABS(Z(M))
      IF(M.GT.1) GO TO 31
      NCT=NCT+1
      IF(NCT.LT.20 .AND. DIF.GT. .00001) GO TO 15
      X(I)=Va(1)
      T(I)=Va(2)
      C(I)=Va(3)
      V(I)=Va(4)
      Y(I)=Va(5)
      W(I)=2.*C(I)
      SF(I)=(CN*Va(4)*((B+FMS*Va(5))/((B+FM*Va(5))*Va(5)))**
     &.6666667)**2
      RETURN
      END
      SUBROUTINE FUNCT(J,IL,IR,DV)
      COMMON X(200),T(200),V(200),W(200),Y(200),C(200),SF(200),
     &Z(5),Va(5),F(5),F1(5),D(5,5),FMS,FM2,CN,FN,Q,SO2,B,FM,G,FL,
     &H,EK1,IGIV,MGIV,ITYPE,GIVDWS(10),GIVTIM(10),DELT
      IF(J.GT.0) THEN
      DV=.001*Va(J)
      Va(J)=Va(J)+DV
      ENDIF
      AM=(B+FM*Va(5))*Va(5)
      F(1)=Va(1)-X(IL)-.5*(Va(2)-T(IL))*(Va(4)+Va(3)+V(IL)+C(IL))
      SFM=(CN*Va(4)*((B+FMS*Va(5))/AM)**.66666667)**2
      F(2)=Va(4)+2.*Va(3)-V(IL)-W(IL)-.5*(Va(2)-T(IL))*(G*
     &(SO2-SFM-SF(IL)))
      F(3)=Va(1)-X(IR)-.5*(Va(2)-T(IR))*(Va(4)-Va(3)+V(IR)-C(IR))
      F(4)=Va(4)-2.*Va(3)-V(IR)+W(IR)-.5*(Va(2)-T(IR))*(G*
     &(SO2-SFM-SF(IR)))
      F(5)=Va(5)-Va(3)**2/G
      IF(J.GT.0) Va(J)=Va(J)-DV
      RETURN
      END
```

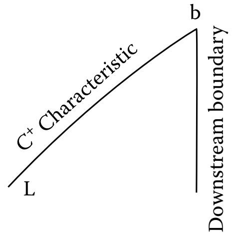
The arguments of this subroutine call are as follows: IL = the number of the point in the xt plane denoted by the subscript L in the above equations, IR = the number of the point in the xt plane denoted by the subscript R in the above equations, and I = the number of the point in the xt plane denoted by the subscript m in the above equations where the solution to the variables is sought. The subroutine REGUL calls a subroutine FUNCT that defines the equations that are to be solved simultaneously. Therefore the subroutine FUNCT implements the evaluation of the five equations (Equations 7.13 through 7.17). It is designed so that if the first argument J that is passed to it is different from 0, then the unknown identified by the value of J (1–5) has an increment DV added to its value consisting of a fraction of the value of this variable (0.001 × the variable, Va(J)) before the equations are evaluated. This allows the derivatives to be evaluated numerically in the subroutine REGUL by calling on FUNCT twice, once without incrementing the unknown variable, and once with its value incremented. The unknown variables are in the array Va. The array F1 contains the equation vector, with the array F used to temporarily store the values of the equation vector for the incremented values of the unknowns. The first few statements in REGUL provide starting values to the five unknowns for the Newton method. Next by calling on FUNCT both the equation vector {**F**}, and the Jacobian matrix, [D] are evaluated. Thereafter the linear system of equations [D]{**z**} = {**F**] is solved for {**z**} using Gaussian elimination. An alternative to solving this system of equations would be to call on another subroutine to perform this task. During the back substitution process of the solution the Newton method is implemented simultaneously.

Before discussing the structure of a driver program that call on REGUL to provide a solution, it is necessary to appropriately handle the intersections of the C^+ and C^- characteristics with the boundary at the beginning and ending of the channel. At these grid points, special consideration must be given to solving for the unknowns.

7.3 BOUNDARY CONDITIONS

7.3.1 Downstream End

In the discussion that follows it will be assumed that the initial condition consists of a steady-state flow throughout the length of the channel, and that the change from this steady-state condition is initiated by making some change at the downstream end of the channel. This change can be one of the following: (1) changing the depth as a function of time, (2) changing the velocity as a function of time, or (3) changing the flow rate as a function of time. These are the same conditions that were described in Chapter 6. There are other possibilities, but the above three are most common. A C^+ characteristic will intersect with the end of the channel at a point b as shown in the sketch below.



At this boundary the value of x is known, and equals the length of the channel from its beginning to its end. Thus when the subscript b replaces m in the above equations $x_b = L$ is known. The C^+ characteristics intersects with the boundary, and therefore it is possible to integrate only along these characteristics and not the C^- characteristics. Thus only Equations 7.9 and 7.10 are available from the characteristics, but Equations 7.11 and 7.12 are not available. The boundary condition gives an additional equation, however. Therefore there are generally three equations that are available to determine values of three variables (or unknowns) at a downstream boundary such as point b in the above sketch. These three unknowns are typically t_b, V_b, and Y_b. Other variables, such as c_b, w_b, and S_{ob} are computed from these.

The three equations available at a downstream boundary point b are as follows:

$$F_1 = L - x_L - \frac{1}{2}(t_b - t_L)(V_b + c_b + V_L + c_L) \tag{7.18}$$

$$F_2 = V_b + w_b - V_L - w_L - \frac{1}{2}(t_b - t_L)(R_{Lb} + R_{LL}) \tag{7.19}$$

$$F_3 = Y_b - f_Y(t) = 0 \quad \text{(When Y is given at the downstream end)} \tag{7.20a}$$

or

$$F_3 = Y_b - f_V(t) = 0 \quad \text{(When V is given at the downstream end)} \tag{7.20b}$$

or

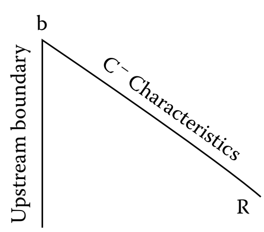
$$F_3 = Y_b - f_Q(t) = 0 \quad \text{(When Q is given at the downstream end)} \tag{7.20c}$$

The last of these equations (7.20a, 7.20b or 7.20c) are interpreted to means that the sought after time, t_b, velocity, V_b, and depth, Y_b will satisfy the relationship that is given that specifies how: (1) the depth varies with time, (2) the velocity varies with time, or (3) how the flow rate varies with time, respectively. A common method for giving these relationships is to read in a table of values giving Y, V or Q at the downstream end and the corresponding times. When using this method then the functions $f_y(t)$, $f_v(t)$ and $f_Q(t)$ are defined by these discrete data, and Equations 7.20a, 7.20b, or 7.20c is satisfied by appropriate interpolation in the table of these values.

7.3.2 Upstream End

When the effect of any change at the downstream end has propagated to the upstream end of the channel the conditions at this end of the channel will affect the unsteady flow conditions. A common boundary condition at the upstream end of the channel is a reservoir whose water surface remains constant regardless of the flow rate into the channel.

To obtain equations that allow for unknown variables to be solved at the beginning of the channel consider the intersection of a C^- characteristic at this boundary at a point b in the xt plane as shown in the sketch below.



The value of x_b equals zero, or whatever x is assigned to the beginning of the channel. Zero will be used in the equations below. The time t_b, velocity, V_b and depth Y_b are unknown. The three equations available to solve for these three unknowns are as follows:

$$F_1 = X_R + \frac{1}{2}(t_b - t_R)(V_b - c_b + V_R - c_R) = 0 \tag{7.21}$$

$$F_2 = X_R - w_b - V_R + w_R - \frac{1}{2}(t_b - t_R)(R_{Rb} + R_{RR}) = 0 \tag{7.22}$$

$$F_3 = H - Y_b - (1 + K_e)\frac{V_b^2}{2g} = 0 \tag{7.23}$$

in which H is the constant water surface elevation above the bottom of the channel at its entrance, and K_e is the minor losses coefficient at this entrance. Obviously depending upon what controls the flow into the channel, a different type boundary condition equation will replace Equation 7.23.

7.3.3 Solution to Boundary Unknowns

The Newton method can be effectively used to solve Equations 7.21 through 7.23. Likewise the Equations 7.18 through 7.20 for the downstream boundary condition can be effectively solved by the Newton method. At the regular points defined at the intersection of the C^+ and C^- characteristics, five equations need to be solved for the five unknowns. Thus from the viewpoint of the number of equations involved, it is easier to handle the boundary points than the regular points.

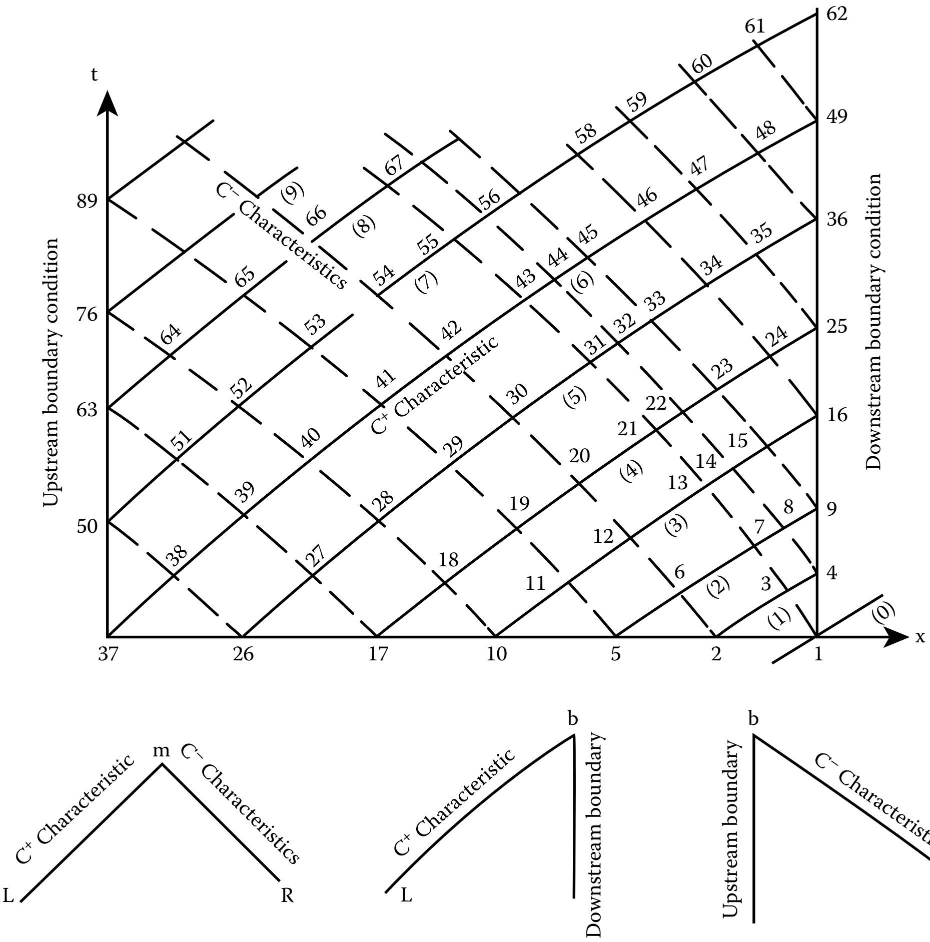
The implementation of solving for the upstream and downstream boundary values is given in the FORTRAN subroutine BOUNDY whose listing is given below.

Subroutine to solve for unknowns at upstream and downstream boundary points in xt plane.

```
      SUBROUTINE BOUNDY(II,I)
      PARAMETER (NEQS=3)
      COMMON
      X(200),T(200),V(200),W(200),Y(200),C(200),SF(200),Z(5),Va(5),
     &F(5),F1(5),D(5,5),FMS,FM2,CN,FN,Q,SO2,B,FM,G,FL,H,EK1,IGIV,
     &MGIV,ITYPE,GIVDWS(10),GIVTIM(10),DELT
      LOGICAL*2 UPSTRB
      IF(II.EQ.I-1) THEN
      UPSTRB=.FALSE.
      ELSE
      UPSTRB=.TRUE.
      ENDIF
      NCT=0
      Va(1)=T(II)+DELT
      Va(2)=V(II)
      Va(3)=Y(II)
15    IF(UPSTRB) THEN
      CALL FUNUPS(0,II,DV)
```

```
      ELSE
      CALL FUNDNS(0,II,DV)
      ENDIF
      DO 20 J=1,NEQS
20    F1(J)=F(J)
      DO 22 J=1,NEQS
      IF(UPSTRB) THEN
      CALL FUNUPS(J,II,DV)
      ELSE
      CALL FUNDNS(J,II,DV)
      ENDIF
      DO 22 J1=1,NEQS
22    D(J1,J)=(F(J1)-F1(J1))/DV
      DO 30 L=1,NEQS-1
      DO 30 M=NEQS,L+1,-1
      IF(ABS(D(M,L)).LT. 1.E-15) GO TO 30
      FAC=D(M,L)/D(L,L)
      F1(M)=F1(M)-FAC*F1(L)
      DO 28 J=L+1,NEQS
28    D(M,J)=D(M,J)-FAC*D(L,J)
30    CONTINUE
      M=NEQS
      Z(M)=F1(M)/D(M,M)
      DIF=ABS(Z(M))
      Va(NEQS)=Va(NEQS)-Z(M)
31    M1=M-1
      SUM=0.
      DO 32 J=M,NEQS
32    SUM=SUM+Z(J)*D(M1,J)
      Z(M1)=(F1(M1)-SUM)/D(M1,M1)
      M=M1
      Va(M)=Va(M)-Z(M)
      DIF=DIF+ABS(Z(M))
      IF(M.GT.1) GO TO 31
      NCT=NCT+1
      IF(NCT.LT.20.AND.DIF.GT. .00001) GO TO 15
      IF(UPSTRB) THEN
      X(I)=0.
      ELSE
      X(I)=FL
      ENDIF
      T(I)=Va(1)
      V(I)=Va(2)
      Y(I)=Va(3)C(I)=SQRT(G*((B+FM*Y(I))*Y(I))/(B+FM2*Y(I)))
      W(I)=2.*C(I)
      SF(I)=(CN*V(I)*((B+FMS*Y(I))/((B+FM*Y(I))*Y(I)))**
     &.6666667)**2
      RETURN
      END
```

```
      SUBROUTINE FUNUPS(J,IR,DV)
      COMMON X(200),T(200),V(200),W(200),Y(200),C(200),SF(200),
     &Z(5),Va(5),F(5),F1(5),D(5,5),FMS,FM2,CN,FN,Q,SO2,B,FM,G,
     &FL,H,EK1,IGIV,MGIV,ITYPE,GIVDWS(10),GIVTIM(10),DELT
      IF(J.GT.0) THEN
      DV=.001*Va(J)
      Va(J)=Va(J)+DV
      ENDIF
      AM=(B+FM*Va(3))*Va(3)
      CB=SQRT(G*Va(3))
      F(1)=X(IR)+.5*(Va(1)-T(IR))*(Va(2)-CB+V(IR)-C(IR))
      SFM=(CN*Va(2)*((B+FMS*Va(3))/AM)**.66666667)**2
      F(2)=Va(2)-2.*CB-V(IR)+W(IR)-.5*(Va(1)-T(IR))
     &*(G*(SO2-SFM-SF(IR)))
      F(3)=H-Va(3)-EK1*Va(2)**2
      IF(J.GT.0) Va(J)=Va(J)-DV
      RETURN
      END
      SUBROUTINE FUNDNS(J,IL,DV)
      COMMON X(200),T(200),V(200),W(200),Y(200),C(200),SF(200),
     &Z(5),Va(5),F(5),F1(5),D(5,5),FMS,FM2,CN,FN,Q,SO2,B,FM,G,FL,
     &H,EK1,IGIV,MGIV,ITYPE,GIVDWS(10),GIVTIM(10),DELT
      IF(J.GT.0) THEN
      DV=.001*Va(J)
      Va(J)=Va(J)+DV
      ENDIF
      AM=(B+FM*Va(3))*Va(3)
      CB=SQRT(G*Va(3))
      F(1)=FL-X(IL)-.5*(Va(1)-T(IL))*(Va(2)+CB+V(IL)+C(IL))
      SFM=(CN*Va(2)*((B+FMS*Va(3))/AM)**.66666667)**2
      F(2)=Va(2)+2.*CB-V(IL)-W(IL)-.5*(Va(1)-T(IL))*(G*(SO2-SFM-SF
     &(IL)))
10    IF(Va(1).LE.GIVTIM(IGIV) .OR. IGIV.EQ.MGIV) GO TO 20
      IGIV=IGIV+1
      GO TO 10
20    IF(Va(1).GE.GIVTIM(IGIV-1) .OR. IGIV.EQ.2) GO TO 30
      IGIV=IGIV-1
      GO TO 20
30    GIVB=(Va(1)-GIVTIM(IGIV-1))/(GIVTIM(IGIV)-GIVTIM(IGIV-1))
     &*(GIVDWS(IGIV)-GIVDWS(IGIV-1))+GIVDWS(IGIV-1)
      GO TO (40,50,60),ITYPE
40    F(3)=Va(3)-GIVB
      GO TO 70
50    F(3)=Va(2)-GIVB
      GO TO 70
60    F(3)=Va(2)*AM-GIVB
70    IF(J.GT.0) Va(J)=Va(J)-DV
      RETURN
      END
```



The argument II in the call to this subroutine BOUNDY is the number of the point on the characteristic adjacent to the downstream boundary where values of all variables are known, the point denoted by subscript L in the above equations if this is a downstream boundary, or the point R if this is an upstream boundary. The argument I is the number of the point where values are being solved, the point denoted by subscript b in the above equations. The subroutine BOUNDY calls on either subroutine FUNUPS or FUNDNS to define the equations that are to be solved simultaneously depending upon whether the point b is at the beginning or end of the channel, respectively. The way BOUNDY makes the decision whether the beginning or end of the channel is being handled is by whether the numbering of II and I are consecutive. If the point is a downstream boundary, then I is one larger than II, but these values are not consecutive at an upstream boundary. The main program that calls on BOUNDY must define the variables involved in the COMMON statement. This means that values must be in the arrays X, T, V, W, Y, C, and SF (for the position x, time t, velocity v, stage variable w, depth Y, and slope of energy line S_f) with the subscript values of L and R. The downstream boundary condition is handled by having the calling program provide values in the arrays GIVTIM and GIVDWS. GIVTIM contains the values of time and GIVDWS contains the corresponding values of depth, velocity, or flow rate, depending upon what the boundary condition consist of. The integer MGIV equals the number of entries in these arrays, and IGIV is an integer that gives the array index of the larger of the two values used for the interpolation. Linear interpolation is used in subroutine FUNDNS to implement giving values to Equations 7.20. The integer variable ITYPE, which is in the COMMON statement, communicates to FUNDNS whether the downstream boundary has (a) the depth (ITYPE = 1), (b) the velocity (ITYPE = 2), or (c) the flow rate

(ITYPE = 3) given, and the computed GO TO (40,50,60), ITYPE selects to implement Equations 7.20a, 7.20b, or 7.20c, respectively.

In studying over this subroutine BOUNDY, you will note that much of the code is identical to that contained in subroutine REGUL. A more efficient program would combine these identical portions into a single subroutine. As with REGUL a few statements limit the use of the program to rectangular channels, the added code necessary to implement the stage variable w for a trapezoidal or circular channel is not included. With minor additions and changes to the program other channels, in addition to just rectangular channels, can be accommodated. Combining regular points with boundary points and the initial condition the solution over any portion of the xt plane will be obtained by appropriately solving for the unknowns at a regular point defined by the intersection of the C^+ and C^- characteristics, or solving for the unknowns at a boundary where either a C^+ or C^- characteristic intersects with it. The numbering of the grid points is a vital part of being able to carry out such a solution. In the following discussion, and the numbering algorithm, the number of grid points between the upstream and downstream boundaries (including the two boundary points) will be denoted by N. The sketch below has seven such grid points, so N = 7. Whenever a solution is sought at a new point m (the subscript in the previous equations) or new point b, the values of all variables must have either been computed at points L and R or be known at these points from the initial condition. When solving a problem whose unsteady flow is caused by changes at the downstream boundary, a convenient numbering of grid points is shown in Figure 7.3, which is duplicated below for reference, with an added number in parenthesis to identify the C^+ characteristics. When a C^+ characteristic intersects with the boundary at the end of the channel the numbering moves to the next C^+ characteristic. As long as the C^+ characteristics intersects with the x-axis, then the initial condition provides the needed known values at points L and R. The grid node where the C^+ characteristic intersects the x-axis or t-axis is the first number of the consecutive sequence of numbers along that characteristic. It is therefore useful to store these first numbers in an integer indexing array, that will be called IBOT below. In Figure 7.3 these are 1, 2, 5, 10, 17, 26, 37, 50, 60, etc. Therefore IBOT should be given the following values: IBOT(0) = 1, IBOT(1) = 2, IBOT(2) = 5, IBOT(3) = 10,… IBOT(6) = 37, IBOT(7) = 50,… It can be seen by studying Figure 7.3 that each new C^+ characteristic line has 2 more grid points along it than the previous one does, if this line

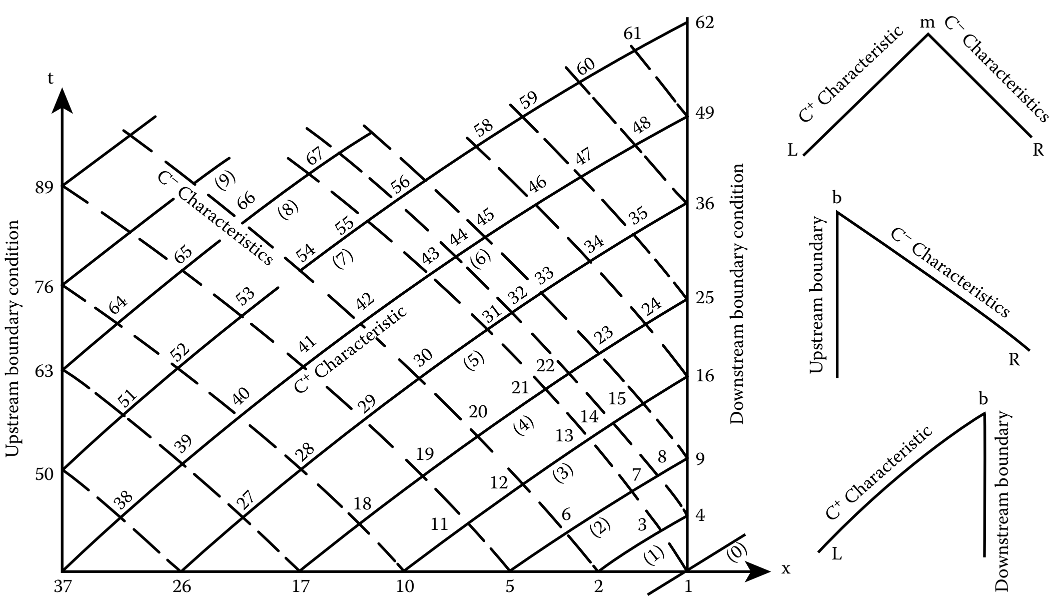


FIGURE 7.3 Sketch of C^+ and C^- characteristics in the xt plane with the numbering starting at downstream end.

intersects the x-axis. When the grid line intersects the t-axis the number of grid points along the line is constant. If we consider the first C^+ character line, with the number (0) passes through point 1, it has one grid point on it. Characteristic line (1) has 3 points along it; characteristic line (2) has 5 points, etc. as long as it intersects the x-axis. Since the subsequent C^+ characteristics that intersect the t-axis have the same number of grid points, the numbering can easily be generated by keeping track of the number of points on the last characteristic line that intersects the x-axis. Since this line passes through the origin, it also intersects the t-axis. Thus an easily implemented algorithm for generating values of index array IBOT consists of the following statements:

FORTRAN statements

```
      II=1
      J=1
      IBOT(0)=J
      DO 10 I=1,N
      J=J+II
      IBOT(I)=J
10    II=II+2
      II=II-2
      DO 20 I=N+1,NUM
20    IBOT(I)=IBOT(I-1)+II
```

C statements

```
j=1; ii=1;
jbot[0]=1;
for(i=1;i<=n;i++){j+=ii,ibot[i]=j;ii+=2;}
ii-=2;
for(i=n+1;i<=n;i++) ibot[i]=ibot[i-1]+ii;
```

(Note in the program below this algorithm is implemented within the DO loop in which I = N,1,–1 (the reverse order of that above) that assigns the initial values along the x-axis when t = 0, so the argument of IBOT is N + 1 – I, rather than I.)

Once this array, IBOT, contains numbers for the grid point that begins each new C^+ characteristic line, it can be used to define the loop, or repetitive, statements, etc. in a program that calls on the boundary, or regular grid point subroutines to solve for the unknowns at the next grid points. As long as the C^+ line intersects the x-axis, the initial values provide the knowns at L and R. When the C^+ line intersects the t-axis, then the subroutine that solves for unknowns at the upstream boundary is called upon. When calling on this subroutine, the number of the grid point denoted by subscript R is the second value on the previous grid line. So its value is obtained as IBOT(I – 1) + 1. Regular points occur from grid points IBOT(I) + 1 to IBOT(I + 1) – 2, with point L equal to one less than the number for which the unknowns are being computed. Grid point R equals IBOT(I – 1) + J – (IBOT(I) + 1), where J equals the values IBOT(I) + 1 to IBOT(I + 1) – 2. Finally the end grid point along each C^+ line consists of a boundary at the end of the channel.

The FORTRAN program below implements the approach described above. The first line of input called for by this program consists of: N = No. of grid points along the x-axis. NPTS = No. of grid points in the xt plane at which the solution is to be obtained, Q = flow rate (a negative value for the numbering scheme used in Figure 7.3.), FN = Manning's n, SO = bottom slope, B = bottom width; G = acceleration of gravity, EK1 = entrance minor loss coefficient, H = water surface elevation in upstream reservoir above the channel bottom. The second line of input contains the x and Y (depth) values at the N points along the x-axis. The third line of input provides the tabular values that define the downstream boundary condition. MGIV = No. of pairs of these values that are given. ITYPE = 1 if the depth is specified as a function of time, ITYPE = 2 if the velocity is specified as a function of time, and ITYPE = 3 if the flow rate is specified as a function of time.

Main Program that implements a solution using Characteristics, UNSCHA.FOR (UNSCHG.FOR—graphics also).

```
      COMMON X(200),T(200),V(200),W(200),Y(200),C(200),SF(200),
     &Z(5),Va(5),F(5),F1(5),D(5,5),FMS,FM2,CN,FN,Q,SO2,B,FM,G,
     &FL,H,EK1,IGIV,MGIV,ITYPE,GIVDWS(10),GIVTIM(10),DELT
      REAL XI(15),YI(15)
      INTEGER*2 IBOT(0:30)
      LOGICAL*2 LOWBOU
      DATA IN,IOUT/2,3/
      READ(IN,*) N,NPTS,Q,FN,SO,B,G,FL,EK1,H
      READ(IN,*)(XI(I),YI(I),I=N,1,-1)
      READ(IN,*) ITYPE,MGIV,(GIVTIM(I),GIVDWS(I),I=1,MGIV)
      IGIV=MGIV/2
      NPTM=NPTS-1
      CN=FN
      IF(G.GT.30.) CN=CN/1.486
      FM=0.
      FM2=2.*FM
      FMS=2.*SQRT(FM*FM+1.)
      SO2=2.*SO
      EK1=(1.+EK1)/(2.*G)
      II=1
      J=1
      IBOT(0)=1
      DO 10 I=N,1,-1
      A=(B+FM*YI(I))*YI(I)
      X(J)=XI(I)
      Y(J)=YI(I)
      V(J)=Q/A
      C(J)=SQRT(G*A/(B+FM2*YI(I)))
      W(J)=2.*C(J)
      T(J)=0.
      SF(J)=(CN*V(J)*((B+FMS*YI(I))/A)**.66666667)**2
      J=J+II
      IBOT(N+1-I)=J
10    II=II+2
      II=II-2
      JBOT=IBOT(N-1)
      DELT=(XI(N)-XI(N-1))/(V(1)+C(1))
      DO 20 I=N+1,30
      IBOT(I)=IBOT(I-1)+II
      IF(IBOT(I).GT.NPTS) GO TO 30
20    NT=I
      IBOT(NT+1)=NPTS+1
30    LOWBOU=.TRUE.
      III=0
      DO 50 I=1,NT
```

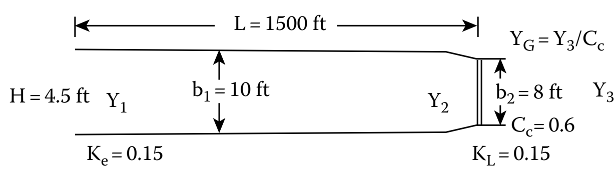
```
      II1=IBOT(I)
      IIP=II1+1
      II2=IBOT(I+1)-1
      IF(II2.GT.NPTM) THEN
      II2=NPTS+1
      LOWBOU=.FALSE.
      ENDIF
      IF(II1.GT.JBOT) THEN
      CALL BOUNDY(IBOT(I-1)+1,II1)
      III=2
      ENDIF
      DO 40 J=IIP,II2-1
40    CALL REGUL(J-1,IBOT(I-1)+J-IIP+III,J)
      IF(LOWBOU) CALL BOUNDY(II2-1,II2)
50    CONTINUE
      WRITE(IOUT,100)
100   FORMAT(' Pt.',6X,'x',8X,'t',8X,'Y',8X,'V',8X,'Q',8X,'c',8X,
     &'w',7X,'Sf',/,1X,76('-'))
      DO 60 I=1,NPTS
      QQ=V(I)*(B+FM*Y(I))*Y(I)
60    WRITE(IOUT,110) I,X(I),T(I),Y(I),V(I),QQ,C(I),W(I),SF(I)
110   FORMAT(I4,7F9.3,F9.6)
      END
```

EXAMPLE PROBLEM 7.1

Water enters a rectangular channel with b = 10 ft, n = 0.012, S_o = 0.0015 with an entrance loss coefficient of K_e = 0.15 from a reservoir whose water surface is 4.5 ft above the channel bottom. Downstream at a distance of L = 1500 ft a sluice gate controls the flow in a width of 8 ft. The gate is initially set at 2.5 ft above the channel bottom. The loss coefficient into the contraction at the gate is K_L = 0.15. At time zero the gate is raised so as to cause the depth at the end of the 10 ft wide channel to fall according to the values in the table below.

t (s)	0	20	60	100	140	180	300
Y (ft)	5.968	5.9	5.7	5.5	5.3	5.1	5.1

Solve the problem numerically using the method of characteristics and N = 6 grid points along the x-axis.



Solution

First the steady state problem, as defined above, must be solved to provide the initial condition to the unsteady problem. The solution to the steady state problem is: Q = 206.72 cfs, Y_1 = 4.030 ft, Y_2 = 5.968 ft, and Y_3 = 5.800 ft. (These depths are at the beginning of the 10 ft channel, at its end, and just upstream from the gate, respectively.) The depths at six equally spaced intervals over this length of channel are as follows:

x (ft)	0	300	600	900	1200	1500
Y (ft)	4.030	4.386	4.763	5.155	5.557	5.968

The input to the above FORTRAN program that solves this problem consists of:

```
6 85 206.72 .012 .0015 10 32.2 1500 .15 4.5
1500.000 5.968
1200.000 5.557
900.000 5.155
600.000 4.763
300.000 4.386
0.000 4.030
1 7 0 5.968 20 5.9 60 5.7 100 5.5 140 5.3 180 5.1 300 5.1
```

The solution to the problem is given below.

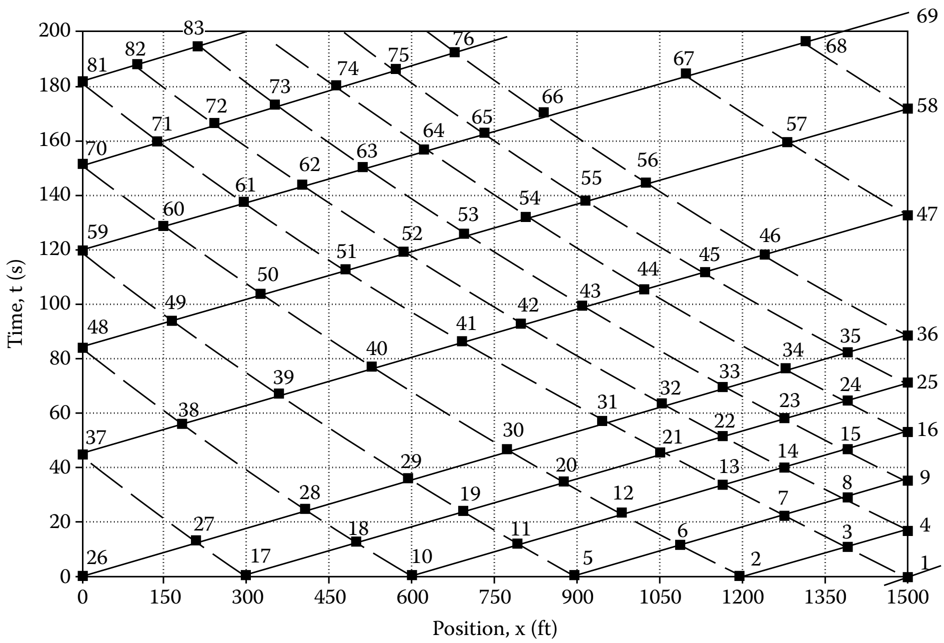
pt.	x	t	Y	V	Q	c	w	Sf
1	1500.000	0.000	5.968	3.464	206.720	13.863	27.725	0.000206
2	1200.000	0.000	5.557	3.720	206.720	13.377	26.753	0.000248
3	1387.757	10.937	5.813	3.555	206.693	13.682	27.364	0.000221
4	1500.000	17.420	5.909	3.599	212.635	13.794	27.587	0.000224
5	900.000	0.000	5.155	4.010	206.720	12.884	25.767	0.000303
6	1092.147	11.331	5.411	3.821	206.744	13.200	26.400	0.000266
7	1276.222	22.104	5.661	3.652	206.717	13.501	27.003	0.000237
8	1387.768	28.579	5.755	3.692	212.477	13.612	27.225	0.000239
9	1500.000	35.031	5.825	3.788	220.618	13.695	27.391	0.000250
10	600.000	0.000	4.763	4.340	206.720	12.384	24.768	0.000374
11	797.291	11.759	5.019	4.118	206.692	12.713	25.426	0.000325
12	985.324	22.892	5.268	3.924	206.715	13.024	26.048	0.000286
13	1165.639	33.493	5.511	3.751	206.690	13.321	26.641	0.000254
14	1276.471	39.957	5.603	3.789	212.263	13.431	26.863	0.000256
15	1388.584	46.435	5.672	3.881	220.141	13.514	27.028	0.000267
16	1500.000	52.807	5.736	3.981	228.335	13.590	27.181	0.000279
17	300.000	0.000	4.386	4.713	206.720	11.884	23.768	0.000467
18	503.364	12.223	4.640	4.455	206.698	12.223	24.446	0.000401
19	695.989	23.740	4.887	4.229	206.672	12.544	25.089	0.000349
20	879.839	34.666	5.128	4.031	206.694	12.850	25.699	0.000307
21	1056.329	45.086	5.363	3.854	206.669	13.141	26.283	0.000273
22	1166.431	51.537	5.453	3.889	212.052	13.251	26.502	0.000275
23	1278.417	58.038	5.522	3.978	219.662	13.334	26.668	0.000285
24	1389.806	64.440	5.585	4.075	227.575	13.410	26.821	0.000297
25	1500.000	70.712	5.646	4.169	235.423	13.484	26.968	0.000309
26	0.000	0.000	4.030	5.130	206.720	11.391	22.783	0.000588
27	210.479	12.722	4.277	4.833	206.705	11.736	23.471	0.000500
28	408.539	24.651	4.520	4.572	206.685	12.065	24.129	0.000430
29	596.440	35.917	4.758	4.343	206.659	12.378	24.756	0.000375
30	776.048	46.627	4.991	4.141	206.680	12.677	25.354	0.000330
31	948.659	56.859	5.219	3.960	206.657	12.964	25.927	0.000293
32	1058.015	63.294	5.307	3.992	211.846	13.073	26.145	0.000294
33	1169.865	69.816	5.375	4.078	219.183	13.155	26.311	0.000305
34	1281.218	76.246	5.437	4.171	226.812	13.232	26.464	0.000317
35	1391.383	82.546	5.498	4.263	234.377	13.306	26.611	0.000328
36	1500.000	88.700	5.556	4.353	241.869	13.376	26.752	0.000340
37	0.000	44.696	4.030	5.129	206.715	11.392	22.784	0.000588

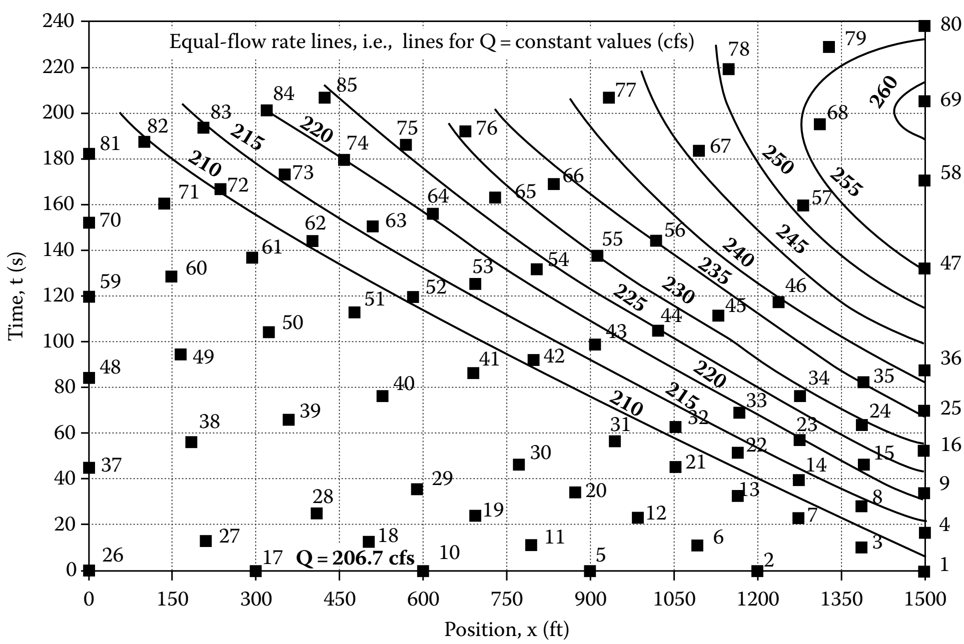
(*continued*)

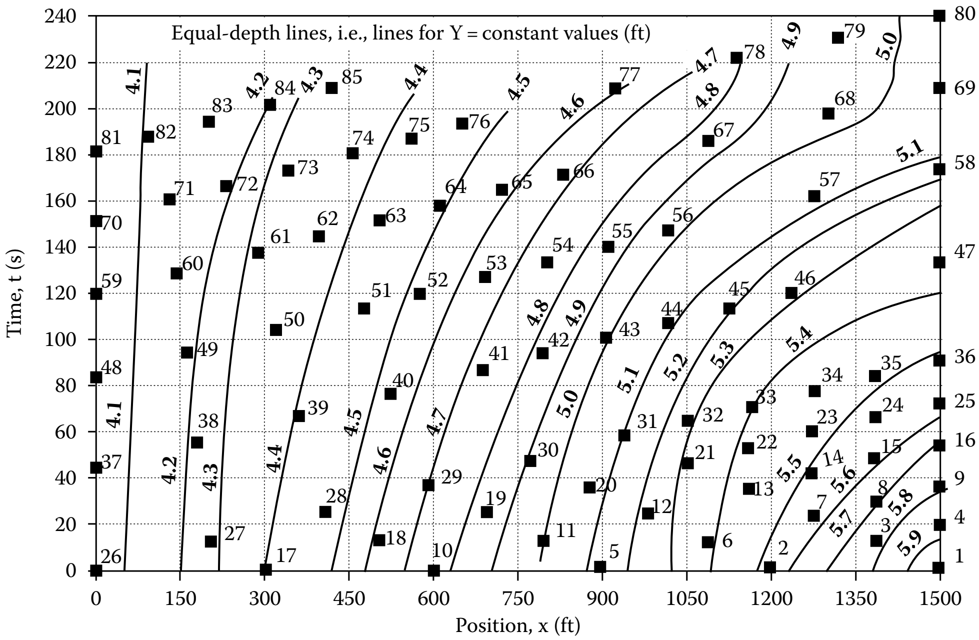
(continued)

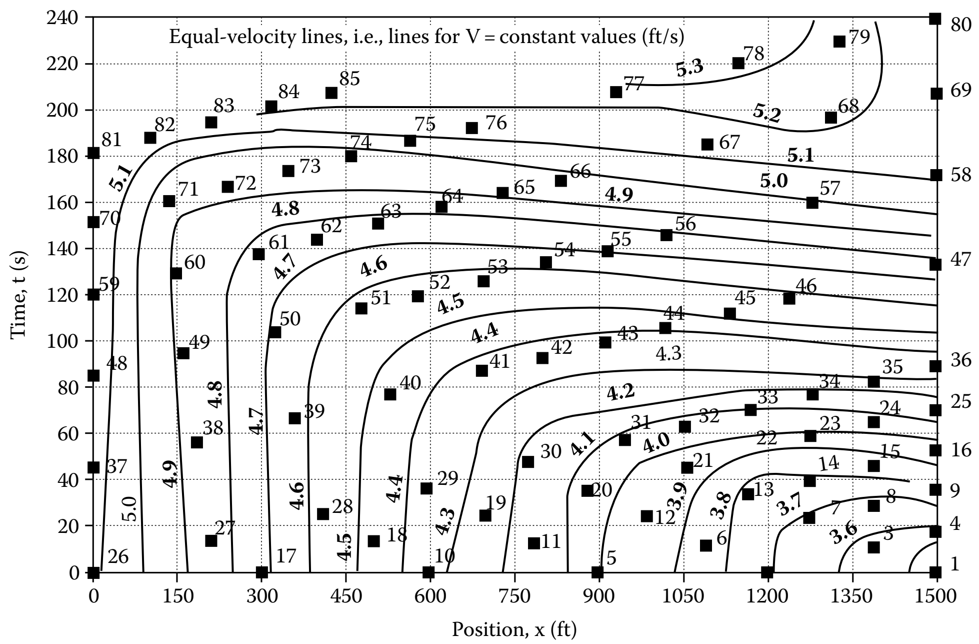
pt.	x	t	Y	V	Q	c	w	Sf
38	184.965	55.878	4.247	4.867	206.697	11.694	23.388	0.000510
39	361.132	66.497	4.462	4.632	206.675	11.986	23.973	0.000446
40	530.167	76.647	4.674	4.422	206.694	12.268	24.535	0.000393
41	693.089	86.385	4.883	4.232	206.673	12.540	25.079	0.000350
42	800.553	92.767	4.966	4.256	211.378	12.646	25.292	0.000350
43	912.019	99.331	5.031	4.334	218.034	12.728	25.456	0.000359
44	1023.234	105.819	5.092	4.418	224.954	12.805	25.609	0.000371
45	1133.274	112.179	5.151	4.501	231.817	12.878	25.756	0.000382
46	1241.795	118.397	5.207	4.582	238.612	12.949	25.898	0.000393
47	1500.000	132.977	5.335	4.779	254.989	13.107	26.214	0.000421
48	0.000	84.137	4.030	5.128	206.687	11.392	22.784	0.000588
49	165.766	94.160	4.224	4.892	206.668	11.663	23.326	0.000517
50	325.365	103.786	4.417	4.679	206.685	11.926	23.853	0.000458
51	479.601	113.057	4.610	4.483	206.667	12.184	24.367	0.000408
52	585.313	119.378	4.688	4.499	210.938	12.287	24.573	0.000406
53	696.360	125.964	4.751	4.568	216.986	12.368	24.736	0.000415
54	807.370	132.489	4.809	4.643	223.276	12.444	24.888	0.000425
55	917.215	138.891	4.866	4.717	229.516	12.517	25.034	0.000436
56	1025.567	145.152	4.920	4.790	235.697	12.587	25.174	0.000446
57	1283.561	159.856	5.045	4.968	250.600	12.745	25.490	0.000472
58	1500.000	171.981	5.140	5.124	263.395	12.865	25.730	0.000496
59	0.000	119.593	4.031	5.127	206.656	11.392	22.785	0.000588
60	151.092	128.731	4.206	4.913	206.672	11.638	23.277	0.000523
61	297.453	137.562	4.383	4.715	206.656	11.880	23.761	0.000468
62	401.515	143.812	4.457	4.724	210.534	11.980	23.960	0.000464
63	512.100	150.405	4.517	4.783	216.033	12.060	24.120	0.000471
64	622.839	156.953	4.573	4.849	221.758	12.135	24.269	0.000480
65	732.425	163.379	4.628	4.915	227.440	12.207	24.414	0.000489
66	840.540	169.669	4.680	4.980	233.072	12.276	24.553	0.000498
67	1098.139	184.462	4.800	5.138	246.666	12.433	24.866	0.000522
68	1314.536	196.686	4.893	5.279	258.348	12.553	25.105	0.000543
69	1500.000	207.068	5.100	5.084	259.273	12.815	25.630	0.000490
70	0.000	151.996	4.030	5.128	206.685	11.392	22.784	0.000588
71	139.156	160.412	4.193	4.929	206.670	11.619	23.238	0.000528
72	241.643	166.587	4.262	4.932	210.190	11.715	23.430	0.000522
73	351.724	173.174	4.319	4.982	215.191	11.793	23.586	0.000528
74	462.132	179.730	4.373	5.040	220.401	11.866	23.732	0.000535
75	571.398	186.169	4.425	5.097	225.579	11.937	23.874	0.000543
76	679.216	192.475	4.476	5.154	230.716	12.005	24.011	0.000551
77	936.261	207.327	4.592	5.295	243.133	12.160	24.320	0.000571
78	1152.453	219.626	4.682	5.421	253.821	12.279	24.558	0.000590
79	1330.238	229.655	4.874	5.227	254.741	12.527	25.054	0.000534
80	1500.000	239.212	5.100	4.959	252.922	12.815	25.630	0.000467
81	0.000	181.895	4.031	5.127	206.661	11.392	22.784	0.000588
82	100.977	187.992	4.096	5.124	209.854	11.484	22.968	0.000580
83	210.510	194.564	4.150	5.166	214.400	11.560	23.120	0.000584
84	320.530	201.118	4.201	5.216	219.144	11.631	23.263	0.000590
85	429.421	207.558	4.252	5.265	223.864	11.701	23.401	0.000596

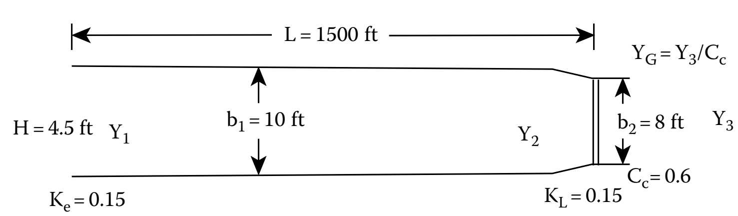
A plot of the grid points in the xt plane as determined by this solution is given below. Notice that this plot is obtained by locating the positions of the points using the x and y values from columns two and four from the above table, and labeling the individual points with the number from the first column. The three plots thereafter consist of contour type plots that contain iso-lines for constant flow rate, constant depth and constant velocity, respectively.






Equal-depth lines, i.e., lines for Y = constant values (ft)
Time, t (s)
Position, x (ft)
0 150 300 450 600 750 900 1050 1200 1350 1500
0 20 40 60 80 100 120 140 160 180 200 220 240


Equal-velocity lines, i.e., lines for V = constant values (ft/s)
Time, t (s)
Position, x (ft)
0 150 300 450 600 750 900 1050 1200 1350 1500
0 20 40 60 80 100 120 140 160 180 200 220 240


L = 1500 ft
$Y_G = Y_3/C_c$
H = 4.5 ft
Y_1
$b_1 = 10$ ft
Y_2
$b_2 = 8$ ft
Y_3
$C_c = 0.6$
$K_e = 0.15$
$K_L = 0.15$

EXAMPLE PROBLEM 7.2

Everything is the same as in Example Problem 7.1 except the channel is trapezoidal with b_1 = 10 ft, m_1 = 1.5, n = 0.013, S_o = 0.0015. The gate is 2.5 ft above the channel bottom in a rectangle with b_2 = 8 ft. The contraction coefficient is C_c = 0.6, and the reservoir supplying the channel at the upstream head has H = 4.5 ft with minor loss coefficient equal to .015 at both the entrance and through the transition from the trapezoidal to the rectangular channels. The depth at the end of the trapezoidal channel varies according to:

t (s)	0	20	60	100	150	180	300
Y (ft)	6.56	6.4	6.2	6.0	5.8	5.6	5.6

Solution

The solution to the steady-state problem produces: Q = 216.6 cfs, Y_1 = 4.34 ft, Y_2 = 6.56 ft, and Y_3 = 6.27 ft (in rectangular channel upstream from gate). The program listed for solving problems using the characteristics has been modified so it evaluates the stage function w using the methods in Chapter 6. Statements in the listed program that assign W(I) = 2*C(I) have been modified to call on a FUNCTION subprogram that interpolates in the table of values of y′ versus w′, and the equations have been modified so that 2.*cb, etc. have been changed to provide the value obtained from this function subprogram. You should make this modifications to the listed program, and see if you duplicate the results below. The input to this program is identical the program whose listing is given with the exception that the side slope m is read immediately after the bottom width on the first line of input. This input data consists of the following:

```
6 85 216.60 .013 .0015 10 1.5 32.2 1500 .15 4.5
1500.000 6.559
1200.000 6.111
900.000 5.664
600.000 5.219
300.000 4.776
0.000 4.336
7 0 6.56 20 6.4 60 6.2 100 6.0 150 5.8 180 5.6 300 5.6
```

The solution table consists of:

pt.	x	t	Y	V	Q	c	w	Sf
1	1500.000	0.000	6.559	1.665	216.600	11.882	31.020	0.000035
2	1200.000	0.000	6.111	1.849	216.600	11.537	29.788	0.000046
3	1368.756	11.485	6.367	1.681	209.284	14.318	30.496	0.000037
4	1500.000	20.304	6.398	2.007	251.714	11.760	30.583	0.000052
5	900.000	0.000	5.664	2.068	216.600	11.179	28.521	0.000063
6	1071.703	11.868	5.924	1.878	210.111	13.811	29.262	0.000050
7	1234.297	22.208	6.175	1.658	197.257	14.101	29.967	0.000037
8	1379.475	31.312	6.208	1.995	239.128	14.138	30.057	0.000053
9	1500.000	39.367	6.303	2.105	258.183	11.687	30.321	0.000058
10	600.000	0.000	5.219	2.328	216.600	10.806	27.218	0.000087
11	775.333	12.290	5.483	2.111	210.991	13.288	27.997	0.000068
12	940.276	22.979	5.738	1.869	199.619	13.593	28.735	0.000051
13	1098.660	33.200	5.986	1.646	187.007	13.883	29.437	0.000038
14	1244.822	42.498	6.019	1.989	227.746	13.921	29.530	0.000055
15	1377.392	50.771	6.113	2.108	247.065	14.030	29.795	0.000060

(*continued*)

(continued)

pt.	x	t	Y	V	Q	c	w	Sf
16	1500.000	58.950	6.205	2.231	267.281	11.611	30.051	0.000067
17	300.000	0.000	4.776	2.642	216.600	10.416	25.874	0.000123
18	479.841	12.758	5.046	2.388	211.729	12.747	26.699	0.000095
19	647.685	23.827	5.305	2.119	201.875	13.070	27.473	0.000071
20	808.051	34.365	5.556	1.873	190.767	13.375	28.207	0.000053
21	962.184	44.453	5.799	1.646	178.427	13.664	28.907	0.000039
22	1109.435	53.951	5.832	1.994	218.032	13.704	29.003	0.000057
23	1241.333	62.291	5.926	2.115	236.776	13.814	29.268	0.000063
24	1376.300	70.702	6.017	2.247	257.233	13.919	29.525	0.000070
25	1500.000	78.927	6.105	2.377	278.064	11.533	29.773	0.000077
26	0.000	0.000	4.336	3.027	216.600	10.008	24.486	0.000178
27	185.530	13.279	4.613	2.721	212.386	12.188	25.366	0.000135
28	356.942	24.762	4.877	2.415	203.905	12.531	26.184	0.000100
29	519.755	35.638	5.130	2.141	194.336	12.853	26.952	0.000075
30	675.505	46.008	5.376	1.890	183.518	13.157	27.682	0.000056
31	825.354	55.948	5.615	1.658	171.476	13.446	28.379	0.000041
32	973.809	65.650	5.649	2.012	209.952	13.487	28.478	0.000060
33	1105.086	74.057	5.742	2.135	228.124	13.597	28.744	0.000066
34	1239.538	82.539	5.832	2.268	247.955	13.703	29.001	0.000073
35	1375.838	91.009	5.919	2.407	269.025	13.806	29.250	0.000081
36	1500.000	99.229	6.004	2.543	290.229	11.453	29.488	0.000090
37	0.000	35.602	4.353	2.868	206.389	10.024	24.541	0.000160
38	145.581	46.152	4.605	2.531	197.041	12.177	25.341	0.000117
39	299.793	56.625	4.849	2.245	188.057	12.495	26.098	0.000087
40	447.677	66.644	5.086	1.983	177.841	12.797	26.819	0.000065
41	590.252	76.275	5.317	1.741	166.416	13.085	27.509	0.000048
42	741.248	86.322	5.353	2.103	202.963	13.128	27.614	0.000069
43	871.729	94.823	5.443	2.227	220.195	13.239	27.881	0.000076
44	1005.566	103.410	5.532	2.361	238.986	13.347	28.139	0.000084
45	1141.394	111.990	5.618	2.501	258.929	13.450	28.389	0.000093
46	1285.202	120.932	5.701	2.665	281.870	13.549	28.629	0.000104
47	1500.000	135.065	5.860	2.846	313.305	11.338	29.081	0.000115
48	0.000	63.157	4.382	2.576	187.044	10.051	24.632	0.000128
49	134.081	73.055	4.617	2.272	177.549	12.193	25.379	0.000094
50	275.418	82.815	4.847	2.004	167.730	12.493	26.093	0.000070
51	411.920	92.219	5.072	1.755	156.714	12.780	26.778	0.000051
52	564.649	102.579	5.109	2.125	191.729	12.826	26.887	0.000074
53	694.259	111.180	5.198	2.250	208.192	12.938	27.156	0.000082
54	827.373	119.874	5.286	2.386	226.136	13.046	27.416	0.000090
55	962.598	128.567	5.371	2.528	245.177	13.151	27.667	0.000099
56	1106.069	137.641	5.453	2.694	267.064	13.251	27.909	0.000111
57	1338.330	152.038	5.609	2.882	297.602	13.439	28.362	0.000123
58	1500.000	162.611	5.716	3.040	322.755	11.222	28.670	0.000135
59	0.000	88.200	4.406	2.289	167.531	10.075	24.712	0.000100
60	123.523	97.495	4.630	2.005	157.293	12.210	25.419	0.000073
61	254.389	106.691	4.849	1.750	146.625	12.496	26.099	0.000053
62	408.619	117.357	4.887	2.129	180.262	12.544	26.214	0.000078
63	537.363	126.061	4.975	2.257	196.032	12.657	26.483	0.000086
64	669.747	134.864	5.061	2.395	213.216	12.766	26.745	0.000095
65	804.354	143.671	5.145	2.539	231.440	12.872	26.998	0.000105

(continued)

pt.	x	t	Y	V	Q	c	w	Sf
66	947.447	152.876	5.227	2.706	252.382	12.973	27.241	0.000117
67	1178.203	167.417	5.381	2.896	281.575	13.163	27.697	0.000130
68	1355.236	178.339	5.486	3.065	306.594	13.291	28.006	0.000143
69	1500.000	187.791	5.600	3.148	324.346	11.127	28.336	0.000148
70	0.000	110.986	4.429	1.990	146.711	10.096	24.785	0.000075
71	114.941	119.816	4.643	1.723	135.712	12.227	25.460	0.000054
72	270.401	130.788	4.681	2.110	168.107	12.278	25.580	0.000080
73	398.246	139.599	4.769	2.240	183.245	12.392	25.851	0.000088
74	529.859	148.517	4.854	2.381	199.739	12.502	26.114	0.000098
75	663.801	157.441	4.937	2.528	217.226	12.609	26.369	0.000109
76	806.447	166.780	5.018	2.698	237.315	12.712	26.615	0.000122
77	1035.672	181.473	5.170	2.890	265.293	12.903	27.072	0.000135
78	1211.942	192.529	5.275	3.062	289.267	13.033	27.384	0.000149
79	1369.768	202.266	5.387	3.154	307.182	13.170	27.714	0.000154
80	1500.000	210.846	5.600	2.907	299.578	11.127	28.336	0.000126
81	0.000	131.952	4.450	1.678	124.505	10.115	24.850	0.000053
82	145.565	143.197	4.489	2.072	155.647	12.023	24.975	0.000081
83	272.482	152.123	4.575	2.206	170.203	12.138	25.248	0.000090
84	403.286	161.162	4.660	2.350	186.060	12.249	25.513	0.000100
85	536.520	170.208	4.742	2.500	202.872	12.357	25.769	0.000111

7.4 USING CHARACTERISTICS WITH SPECIFIED TIME INCREMENTS

7.4.1 Based on Second-Order Approximations

The disadvantage of not having the solution throughout the channel's length at the same time has prompted development of other methods that utilize the characteristic equations. The method described in this section, which has been attributed to Hartree, interpolates the velocity and depth between the x grid points so that the C^+ and C^- characteristics intersect at the specified time at the position where the solution is desired as illustrated in the sketch below (which applies only for subcritical flows). The values of the dependent variables are known at all grid points at the kth time step (k will be used as a superscript of the variables) from the initial condition to start with, and from the past time step solution thereafter. What is sought are values of the dependent variables at the advance time step k + 1. To use the characteristic equations, the x values of points L and R (as well as the variables of the flow) on the (k + 1)th time line are needed. The positions L and R are obtained from solving the characteristics equations simultaneously with interpolation equations that give the values of depth and velocity at points L and R. The details describing how this is accomplished are given below.

The slope of the positive characteristic passing through grid point i, k + 1 in the xt plane is defined by Equation 7.5 or dx/dt = V + c. The x position at point L can be determined from grid position i, k + 1 by using the average slope of this characteristic at its two ends, or

$$x_L = x_i^{k+1} - \frac{1}{2}\Delta t\left(V_i^{k+1} + c_i^{k+1} + V_L + c_L\right)$$

A superscript k could be placed on V_L and c_L; however, it is understood that these are on the kth time line. The superscript k + 1 is not needed on x_i since the grid spacing Δx does not change with time, and therefore in subsequent equations, it will be omitted. This equation, as well as subsequent equations that constitute a system of simultaneous equations, will be written as a function of the unknowns equal zero in a form ready for solution by the Newton method. The above equation then becomes:

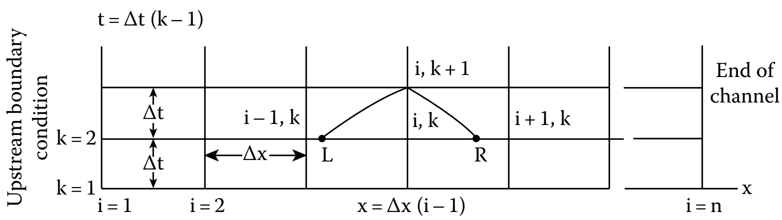
(*Note*: The superscript k + 1 on x can be dropped since x_i is established by the grid spacing and is not dependent upon time.)

$$F_1 = x_L - x_i^{k+1} - \frac{1}{2}\Delta t\left(V_i^{k+1} + c_i^{k+1} + V_L + c_L\right) = 0 \tag{7.24}$$

(Note the superscript k + 1 on x can be dropped since x_i is established by the grid spacing and is not dependent upon time.) Equation 7.24 is a second-order numerical approximation for solving Equation 7.5 from point L to point i, k + 1 in the xt plane. Along this C^+ characteristic, Equation 7.6 applies. Using a second order numerical approximation to integrate Equation 7.6 between points L and i, k + 1 results in

$$F_2 = V_i^{k+1} + w_i^{k+1} - V_L - w_L - \frac{1}{2}\Delta t\left(R_L^{k+1} + R_{LL}\right) = 0 \tag{7.25}$$

in which R_L is as defined earlier below Equation 7.3 with the added sub and/or superscripts to denote that it is to be evaluated at these points in the xt plane.



Sketch of portion of xt plane showing C^+ and C^- characteristics at point I, k + 1

Likewise using Equations 7.7 and 7.8 along the C^- characteristics in the above sketch produces the following two equations:

$$F_3 = x_i - x_R - \frac{1}{2}\Delta t\left(V_i^{k+1} - c_i^{k+1} + V_R - c_R\right) = 0 \tag{7.26}$$

$$F_4 = V_i^{k+1} - w_i^{k+1} - V_R - w_R - \frac{1}{2}\Delta t\left(R_R^{k+1} + R_{RR}\right) = 0 \tag{7.27}$$

Variables of the problem on the kth time line are known, as well as x_i and therefore the following eight variables are unknown in the above four equations, Equations 7.24 through 7.27: V_i^{k+1}, Y_i^{k+1}, x_L, x_R, V_L, V_R, Y_L, and Y_R. The w's and c's could also be added to this list of unknowns, but since they are a function of Y (and other known channel parameters) they will not be considered part of the list of unknown variables. To solve for these eight unknowns, four additional equations are needed. These will be interpolation equations to determine variables at the points L and R. If linear interpolation is used then these equations are

$$F_5 = V_L - V_{i-1}^k - \frac{(x_L - x_{i-1})}{\Delta x}\left(V_i^k - V_{i-1}^k\right) = 0 \tag{7.28}$$

$$F_6 = V_R - V_i^k - \frac{(x_R - x_i)}{\Delta x}\left(V_{i+1}^k - V_i^k\right) = 0 \tag{7.29}$$

$$F_7 = Y_L - Y_{i-1}^k - \frac{(x_L - x_{i-1})}{\Delta x}\left(Y_i^k - Y_{i-1}^k\right) = 0 \tag{7.30}$$

$$F_8 = Y_R - Y_i^k - \frac{(x_R - x_i)}{\Delta x}\left(Y_{i+1}^k - Y_i^k\right) = 0 \tag{7.31}$$

An alternative to considering x_L and x_R unknowns is to let the fraction of the regular grid space to points L and R, or $h_3 = (x_L - x_{i-1})/\Delta x$ and $h_4 = (x_R - x_i)/\Delta x$ be considered the unknowns. Equations 7.24 through 7.31 are eight equations that can be solved by the Newton method to determine the eight unknown variables.

The last four of these equations are first-order approximations, and might be replaced by second-order interpolation equations to be consistent with the approximation order of Equations 7.24 through 7.27. These second-order interpolation equation can be obtained from Lagrange's formula discussed in Appendix B, or more directly from an interpolation formula based on equal increments in Δx. These second-order equations that replace Equations 7.28 through 7.31 are

$$F_5 = V_L - V_{i-2}^k - \frac{1}{2}\left(3h_+ - h_+^2\right)\left(V_{i-1}^k - V_{i-2}^k\right) - \frac{1}{2}\left(h_+^2 - h_+\right)\left(V_i^k - V_{i-1}^k\right) = 0 \tag{7.28a}$$

$$F_6 = V_R - V_i^k - \frac{1}{2}\left(3h_4 - h_4^2\right)\left(V_{i+1}^k - V_i^k\right) - \frac{1}{2}\left(h_4^2 - h_4\right)\left(V_{i+2}^k - V_{i+1}^k\right) = 0 \tag{7.29a}$$

$$F_7 = Y_L - Y_{i-2}^k - \frac{1}{2}\left(3h_+ - h_+^2\right)\left(Y_{i-1}^k - Y_{i-2}^k\right) - \frac{1}{2}\left(h_+^2 - h_+\right)\left(Y_i^k - Y_{i-1}^k\right) = 0 \tag{7.30a}$$

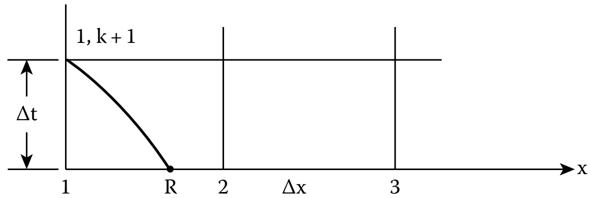
$$F_8 = Y_R - Y_i^k - \frac{1}{2}\left(3h_4 - h_4^2\right)\left(Y_{i+1}^k - Y_i^k\right) - \frac{1}{2}\left(h_4^2 - h_4\right)\left(Y_{i+2}^k - Y_{i+1}^k\right) = 0 \tag{7.31a}$$

in which $h_+ = h_3 + 1$. Another advantage in using these latter quadratic (second-order) interpolation equations is that should points L and/or R be outside the first Δx interval from position i, then the 3 grid points used encompass L and R. When this occurs, the indexes of the linear Equation 7.28 through 7.31 should be changed so that extrapolation is not used rather than interpolation. If no lateral inflow or outflow is taking place, then the terms involving R_L and R_R after being evaluated at the proper points are given by the following: $\Delta t(R_{Li}^{k+1} + R_{LL})/2 = g\Delta t[S_o - (S_{fi}^{k+1} + S_{fL})/2]$ and $\Delta t(R_{Ri}^{k+1} + R_{RR})/2 = g\Delta t[S_o - (S_{fi}^{k+1} + S_{fR})/2]$, in which the values of the slope of the energy line are evaluated using Manning's equation, and the depth at the designated point, e.g., to evaluate S_{fi}^{k+1} the depth Y_i^{k+1} is used, to evaluate S_{fR} the depth Y_R^k is used and to evaluate S_{fL} depth Y_L^k is used.

The above system of equations apply at all grid positions for i varying from 2 to n − 1. At the beginning of the channel where i = 1 and at the end of the channel where i = n special boundary condition equations must be used that properly reflect the occurrences here.

7.4.2 Upstream Boundary Condition Equations

To illustrate what must be done to add an appropriate equation at the upstream end of the channel where i = 1, it will be assumed that the channel is being supply water from a reservoir whose water surface elevation above the channel bottom is a known function of time, H(t). Then from the energy equation $H(t) = Y + (1 + K_e)V^2/(2g)$. A C^- characteristic at the k + 1 time line to the kth time line from the upstream boundary will appear as shown in the sketch below.



Use of Equations 7.7 and 7.8 along this characteristic produce the following two numerical equations:

$$F_1 = x_1 - x_R - \frac{1}{2}\Delta t\left(V_1^{k+1} - c_1^{k+1} + V_R - c_R\right) = 0 \tag{7.32}$$

$$F_2 = V_1^{k+1} - w_i^{k+1} + V_R - w_R - \frac{1}{2}\Delta t(R_{R1} + R_{RR}) = 0 \tag{7.33}$$

As a third equation the energy equation from the reservoir to the channel is written as

$$F_3 = H - Y_1^{k+1} - (1+K_e)\frac{\left(V_i^{k+1}\right)^2}{2g} = 0 \tag{7.34}$$

In addition, the following two interpolation equations are available if linear interpolation is used:

$$F_4 = V_R - V_1^k - \frac{(x_R - x_1)}{\Delta x}\left(V_2^k - V_1^k\right) = 0 \tag{7.35}$$

$$F_5 = Y_R - Y_1^k - \frac{(x_R - x_1)}{\Delta x}\left(Y_2^k - Y_1^k\right) = 0 \tag{7.36}$$

and if quadratic interpolation is used, then

$$F_4 = V_R - V_1^k - \frac{1}{2}\left(3h_4 - h_4^2\right)\left(V_2^k - V_1^k\right) - \frac{1}{2}\left(h_4^2 - h_4\right)\left(V_3^k - V_2^k\right) \tag{7.35a}$$

$$F_5 = Y_R - Y_1^k - \frac{1}{2}\left(3h_4 - h_4^2\right)\left(Y_2^k - Y_1^k\right) - \frac{1}{2}\left(h_4^2 - h_4\right)\left(Y_3^k - Y_2^k\right) \tag{7.36a}$$

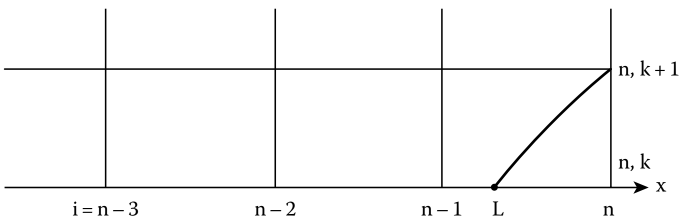
The five unknown in these five equations are: V_1^{k+1}, Y_1^{k+1}, x_R, V_R, and Y_R.

7.4.3 Downstream Boundary Condition Equations

The following three possible conditions will be discussed for the downstream boundary condition: (1) the depth of flow $Y_n(t)$ will be given as a function of time, (2) the velocity $V_n(t)$ will be given as a function of time, and (3) the flow rate $Q_n(t)$ will be given as a function of time. In the case of the downstream boundary condition, a C^+ characteristic will be used as shown in the sketch below. Equations 7.5 and 7.6 applied along this characteristic and provide the first two of the following equations. The specified boundary condition will be used for the third equation and the final two equations come from interpolation formula as done above. (While Equations 7.39a and 7.39b could be omitted if $Y_n(t)$, or $V_n(t)$ is specified, these equations are used so five equations can be solved for all three cases as well as being consists with solving five equations for the upstream boundary condition.)

$$F_1 = x_n - x_L - \frac{1}{2}\Delta t\left(V_n^{k+1} + c_n^{k+1} + V_L + c_L\right) = 0 \tag{7.37}$$

$$F_2 = V_n^{k+1} + w_n^{k+1} - V_L - w_L - \frac{1}{2}\Delta t\left(R_{Ln}^{k+1} + R_{LL}\right) = 0 \tag{7.38}$$



If $Y_n(t)$ is specified, then

$$F_3 = Y_n^{k+1} - Y_n(t) = 0 \tag{7.39a}$$

If $V_n(t)$ is specified, then

$$F_3 = V_n^{k+1} - V_n(t) = 0 \tag{7.39b}$$

If $Q_n(t)$ is specified, then

$$F_3 = A_n^{k+1}V_n^{k+1} - Q_n(t) = 0 \tag{7.39c}$$

$$F_4 = V_L - V_n^k - \frac{(x_n - x_L)}{\Delta x}\left(V_n^k - V_{n-1}^k\right) = 0 \tag{7.40}$$

$$F_5 = Y_L - Y_n^k - \frac{(x_n - x_L)}{\Delta x}\left(Y_n^k - Y_{n-1}^k\right) = 0 \tag{7.41}$$

Equations 7.40 and 7.41 are based on linear interpolation, and if quadratic interpolation is used the last two equations become

$$F_4 = V_L - V_{n-2}^k - \frac{1}{2}\left(3h_+ - h_+^2\right)\left(V_{n-1}^k - V_{n-2}^2\right) - \frac{1}{2}\left(h_+^2 - h_+\right)\left(V_n^k - V_{n-1}^k\right) \tag{7.40a}$$

$$F_5 = Y_L - Y_{n-2}^k - \frac{1}{2}\left(3h_+ - h_+^2\right)\left(Y_{n-1}^k - Y_{n-2}^2\right) - \frac{1}{2}\left(h_+^2 - h_+\right)\left(Y_n^k - Y_{n-1}^k\right) \tag{7.41a}$$

Again for this downstream boundary the five equations, Equation 7.37 through 7.41, allow for the following five unknown variables to be solved: V_n^{k+1}, Y_n^{k+1}, x_L, V_L, and Y_L.

The above boundary conditions that specify the head at the upstream end and allow for the depth, velocity, or the flow rate to be specified as a function of time at the downstream end are the most elementary types. Other boundary conditions that account for what may exist in a given channel include gates, and other controls. For these more complete boundary conditions, equation F_3 can be replaced by what might more accurately reflect what occurs, and/or additional equations added. See homework problems for a few of these more complex upstream and/or downstream boundary conditions.

The computer program given below implements the method using the specified time increments (as well as specified x-increments). In this program, there are two arrays for the velocity, V(51) and VK(51), and depth, Y(51) and YK(51), for the current time step and the advanced time step, respectively. (VK and YK are for the current or kth time step, and V and Y are for the k + 1st advanced time step for which the solution is sought.) The input to the program consists of the following three lines of data (which may spill over unto more lines as necessary):

- First line: NT = No. of time steps, N = No. of space increments, LINEAR = 1 to use linear interpolation or = 0 for quadratic interpolation, DELT = the time increment Δt, Q = the flow rate in the channel, FN = Manning's roughness coefficient, n, SO = the slope of the channel bottom, B = the bottom width of the channel, FM = the side slope of the channel, FL = the length of channel, H = head of reservoir, which is assumed to be constant in this program, and KE = the entrance minor loss coefficient.
- Second line: (YK(I),I = 1,N) = the array of depth values at each x position along the channel at a spacing of Δx that represent the initial condition, or the depths at time t = 0.
- Third line: ITYPE denotes the type of downstream boundary condition, e.g., ITYPE = 1 then $Y_n(t)$ is given, if ITYPE = 2, then $V_n(t)$ is given, and if ITYPE = 3 then $Q_n(t)$ is given. MGIV is the number of pairs of time and downstream boundary condition values that are given. (GIVTIM(I),GIVDWS(I),I = 1,MGIV) = pairs of time with Y_n, V_n or Q_n that specify the downstream boundary condition. The times GIVTIM(I) do not need to correspond with the times at which a solution is to be obtained.

Subroutine SOLVE provides guesses for the unknowns, defines the equation vector and Jacobian matrix for the Newton method, and solves the nonlinear system of equations using the Newton method. To evaluate the elements in the equation vector and the Jacobian, it calls on Subroutine FUNCT, which evaluates the equations F_1 through F_8 for regular points and F_1 through F_5 for the upstream and downstream boundaries when i = 1 and i = n, respectively. The FUNCTION subprogram WSTAGE uses interpolation of the arrays YST (for depth) and W (for stage variable) to determine w corresponding to the depth YYY. The relationship between depth Y and stage variable w is stored in the arrays W and YST and these are generated in subroutine STAGE which numerically integrates the equation that define this relationship.

The array Va(8) is used to hold values of the unknowns. For the regular grid points the correspondence is as follows: $Va(1) = V_i^{k+1}$, $Va(2) = Y_i^{k+1}$, $Va(3) = h_3 = (x_L - x_{i-1})/\Delta x$, $Va(4) = h_4 = (x_R - x_i)/\Delta x$, $Va(5) = V_L$, $Va(6) = V_R$, $Va(7) = Y_L$, $Va(6) = Y_R$. For the upstream boundary condition, the correspondence is: $Va(1) = V_1^{k+1}$, $Va(2) = Y_1^{k+1}$, $Va(3) = h_4 = (x_R - x_1)/\Delta x$, $Va(4) = V_R$, $Va(5) = Y_R$. For the downstream boundary condition the correspondence is: $Va(1) = V_n^{k+1}$, $Va(2) = Y_n^{k+1}$, $Va(3) = h_3 = (x_n - x_L)/\Delta x$, $Va(4) = V_L$, $Va(5) = Y_L$. The elements of the Jacobian matrix are generated using a numerical approximation for the derivatives and thus the subroutine FUNCT is called to re-evaluate the functions after each of the eight unknown variables have been incremented by a small amount. The solution of the linear system of equations that implements the Newton method is build into the program. For this solution method to work proper, it is necessary that the equations be ordered so that the Jacobian will not have a zero element on the diagonal. Once the solution of the new k + 1st time step is complete, and the results are written to the output file, then the values in the arrays V and Y are transferred to the arrays VK and YK, and the solution process repeated for the next time step, etc.

In using a program such as this that utilizes characteristics, the specifications for Δt should be such that points L and R lies within one Δx of the point i,k. This means that the sum of largest the celerity and magnitude of velocity divided into the increment Δx should not exceed Δt.

Listing of program to solve characteristic equation with a specified time increment, HARTREE. FOR

```
      COMMON V(51),Y(51),VK(51),YK(51),Va(8),F(8),F1(8),
     &D(8,8),Z(8),GBS,DELTDX,FMS,DTG2,FM2,CN,DX,DT2,SO,B,FM,G,DV,
     &DELT,H,FEK,VaN,N,II,IOUT,LINEAR,ITYPE
      COMMON /CSTAGE/ YST(400),W(400),ITAB
      REAL GIVTIM(40),GIVDWS(40),KE
      WRITE(*,*)' Give input and output units: IN & IOUT'
      READ(*,*) IN,IOUT
      READ(IN,*) NT,N,LINEAR,DELT,NPRT,Q,FN,SO,B,FM,G,FL,H,KE
      READ(IN,*) (YK(I),I=N,1,-1)
C     READ(IN,*) (YK(I),I=1,N)
      READ(IN,*) ITYPE,MGIV,(GIVTIM(I),GIVDWS(I),I=1,MGIV)
C ITYPE=1 then Y(N)-given; ITYPE=2 then V(N)-given;
C ITYPE=3 then Q(N)-given
C Unknowns: Va(1)=V, Va(2)=Y, Va(3)=XL, Va(4)=XR, Va(5)=VL, Va(6)=VR,
C Va(7)=YL, Va(8)=YR.
C For upstream BC Va(1)=V, Va(2)=Y, Va(3)=XR, Va(4)=VR, Va(5)=YR
C For downstream BC Va(1)=V, Va(2)=Y, Va(3)=XL, Va(4)=VL, Va(5)=YL
      GBS=SQRT(G*B/FM)
      DT2=DELT/2.
      DTG2=G*DELT
      DX=FL/FLOAT(N-1)
      DELTDX=DELT/DX
      FEK=(KE+1.)/(2.*G)
      CN=FN
      IF(G.GT.30.) CN=CN/1.486
      FM2=2.*FM
      FMS=2.*SQRT(FM*FM+1.)
      ITAB=100
      IF(FM.GT. 0.) CALL STAGE
      DO 5 I=1,N
      Y(I)=YK(I)
      VK(I)=Q/((B+FM*Y(I))*Y(I))
5     V(I)=VK(I)
      WRITE(IOUT,100) 0,0.,GIVDWS(1)
      WRITE(IOUT,101) (Y(I),I=1,N)
      WRITE(IOUT,101)(V(I),I=1,N)
      WRITE(IOUT,102)(Q,I=1,N)
      IGIV=2
      WRITE(*,*)' Give frequency of plot data'
      READ(*,*) NPRNT
      DO 50 K=1,NT
      TIME=DELT*FLOAT(K)
10    IF(GIVTIM(IGIV).GE.TIME .OR. IGIV.EQ.MGIV) GO TO 20
      IGIV=IGIV+1
      GO TO 10
```

```
20    IG=IGIV-1
      VaN=GIVDWS(IG)+(TIME-GIVTIM(IG))/(GIVTIM(IGIV)-GIVTIM(IG))*
     &(GIVDWS(IGIV)-GIVDWS(IG))
      CALL SOLVE
      IF(MOD(K,NPRT).EQ.0) THEN
      WRITE(IOUT,100) K,TIME,VaN
100   FORMAT(/,' K=',I3,', Depth, Velocity & Flow rate for
     &Time=',F8.1,' sec. (Given=',F8.2,')')
      WRITE(IOUT,101) (Y(I),I=1,N)
101   FORMAT(1X,16F7.2)
      WRITE(IOUT,101)(V(I),I=1,N)
      WRITE(IOUT,102)(V(I)*(B+FM*Y(I))*Y(I),I=1,N)
102   FORMAT(1X,16F7.1)
      ENDIF
      IF(MOD(K,NPRNT).EQ.0) write(4,*)' '
      DO 40 I=1,N
      YA=V(I)
      IF(MOD(K,NPRNT).EQ.0) write(4,333) dx*float(i-1),Y(I),YA,
     &V(I)*(B+FM*Y(I))*Y(I)
333   format(f6.1,2f8.3,f9.2)
      V(I)=2.*YA-VK(I)
      VK(I)=YA
      YA=Y(I)
      Y(I)=2.*YA-YK(I)
40    YK(I)=YA
50    CONTINUE
      END
      SUBROUTINE SOLVE
      COMMON V(51),Y(51),VK(51),YK(51),Va(8),F(8),F1(8),D(8,8),Z(8),
     &GBS,DELTDX,FMS,DTG2,FM2,CN,DX,DT2,SO,B,FM,G,DV,DELT,H,
     &FEK,VaN,N,II,IOUT,LINEAR,ITYPE
      DO 50 I=1,N
      II=I
      IP=I+1
      IM=I-1
      NCT=0
      Va(1)=V(I)
      Va(2)=Y(I)
C  Va(3) IS (XL-Xi-1)/Dx, or XL=X(I-1)+Dx*Va(3)
C  Va(4) IS (XR-Xi)/Dx, or XR=X(I)+Dx*Va(4)
      IF(I.EQ.1) THEN
      NEQ=5
      Va(3)=(SQRT(G*(B+FM*Y(1))*Y(1)/(B+FM2*Y(1)))-V(1))*DELTDX
      Va(4)=V(1)+Va(3)*(V(2)-V(1))
      Va(5)=Y(1)+Va(3)*(Y(2)-Y(1))
      ELSE IF(I.EQ.N) THEN
      NEQ=5
      Va(3)=1.-(V(N)+SQRT(G*(B+FM*Y(N))*Y(N)/(B+FM2*Y(N))))*DELTDX
      Va(4)=V(N-1)+(1.-Va(3))*(V(N)-V(N-1))
      Va(5)=Y(N-1)+(1.-Va(3))*(Y(N)-Y(N-1))
      ELSE
```

```
      NEQ=8
      IF(I.EQ.2) THEN
      Va(3)=1.-(V(2)+SQRT(G*(B+FM*Y(2))*Y(2)/(B+FM2*Y(2))))
     &*DELTDX
      Va(4)=(SQRT(G*(B+FM*Y(2))*Y(2)/(B+FM2*Y(2)))-V(2))*DELTDX
      ENDIF
      Va(5)=VK(IM)+Va(3)*(VK(I)-VK(IM))
      Va(7)=YK(IM)+Va(3)*(YK(I)-YK(IM))
      Va(6)=VK(I)+Va(4)*(VK(IP)-VK(I))
      Va(8)=YK(I)+Va(4)*(YK(IP)-YK(I))
      ENDIF
15    CALL FUNCT(0)
      DO 20 J=1,NEQ
20    F1(J)=F(J)
      DO 22 J=1,NEQ
      CALL FUNCT(J)
      DO 22 J1=1,NEQ
22    D(J1,J)=(F(J1)-F1(J1))/DV
      DO 30 L=1,NEQ-1
      DO 30 M=NEQ,L+1,-1
      IF(ABS(D(M,L)).LT. 1.E-15) GO TO 30
      FAC=D(M,L)/D(L,L)
      F1(M)=F1(M)-FAC*F1(L)
      DO 28 J=L+1,NEQ
28    D(M,J)=D(M,J)-FAC*D(L,J)
30    CONTINUE
      M=NEQ
      Z(M)=F1(M)/D(M,M)
      DIF=ABS(Z(M))
      Va(M)=Va(M)-Z(M)
31    M1=M-1
      SUM=0.
      DO 32 J=M,NEQ
32    SUM=SUM+Z(J)*D(M1,J)
      IF(ABS(D(M1,M1)).LT.1.E-10) THEN
      Z(M1)=0.
      ELSE
      Z(M1)=(F1(M1)-SUM)/D(M1,M1)
      ENDIF
      M=M1
      Va(M)=Va(M)-Z(M)
      DIF=DIF+ABS(Z(M))
      IF(M.GT.1) GO TO 31
      NCT=NCT+1
      IF(NCT.LT.20 .AND. DIF.GT. .00001) GO TO 15
      IF(NCT.EQ.20) WRITE(*,*)' Failed to converge',I,DIF
      V(I)=Va(1)
50    Y(I)=Va(2)
      RETURN
      END
      SUBROUTINE FUNCT(J)
```

```
      PARAMETER (EX=.66666667)
      COMMON V(51),Y(51),VK(51),YK(51),Va(8),F(8),F1(8),
     &D(8,8),Z(8),GBS,DELTDX,FMS,DTG2,FM2,CN,DX,DT2,SO,B,FM,G,DV,
     &DELT,H,FEK,VaN,N,II,IOUT,LINEAR,ITYPE
      IM=I-1
      IP=I+1
      IF(J.GT.0) THEN
      DV=.001*Va(J)
      IF(ABS(DV).LT. 1.E-5) DV=SIGN(1.E-5,DV)
      Va(J)=Va(J)+DV
      ENDIF
      A=(B+FM*Va(2))*Va(2)
      C=SQRT(G*A/(B+FM2*Va(2)))
      SF=(CN*Va(1)*((B+FMS*Va(2))/A)**EX)**2
      IF(I.EQ.1) THEN
C Upstream BC H=Y+(1+Ke)V**2/(2g)
      AR=(B+FM*Va(5))*Va(5)
      CR=SQRT(G*AR/(B+FM2*Va(5)))
      SFR=(CN*Va(4)*((B+FMS*Va(5))/AR)**EX)**2
      F(1)=DT2*(C-Va(1)+CR-Va(4))-DX*Va(3)
      F(2)=Va(1)-WSTAGE(Va(2))-Va(4)+WSTAGE(Va(5))-DTG2*
     &(SO-.5*(SF+SFR))
      F(3)=H-Va(2)-FEK*Va(1)**2
      IF(LINEAR.EQ.1) THEN
      F(4)=Va(4)-VK(1)-Va(3)*(VK(2)-VK(1))
      F(5)=Va(5)-YK(1)-Va(3)*(YK(2)-YK(1))
      ELSE
      F(4)=Va(4)-VK(1)-.5*Va(3)*((3.-Va(3))*(VK(2)-VK(1))+
     &(Va(3)-1.)*(VK(3)-VK(2)))
      F(5)=Va(5)-YK(1)-.5*Va(3)*((3.-Va(3))*(YK(2)-YK(1))+
     &(Va(3)-1.)*(YK(3)-YK(2)))
      ENDIF
      ELSE IF(I.EQ.N) THEN
C Downstream BC
      AL=(B+FM*Va(5))*Va(5)
      CL=SQRT(G*AL/(B+FM2*Va(5)))
      SFL=(CN*Va(4)*((B+FMS*Va(5))/AL)**EX)**2
      F(1)=DX*(1.-Va(3))-DT2*(Va(1)+C+Va(4)+CL)
      F(2)=Va(1)+WSTAGE(Va(2))-Va(4)-WSTAGE(Va(5))-DTG2*(SO-.5*
     &(SF+SFL))
      IF(ITYPE-2) 1,2,3
1     F(3)=Va(2)-VaN
      GO TO 4
2     F(3)=Va(1)-VaN
      GO TO 4
3     F(3)=VaN-A*Va(1)
4     IF(LINEAR.EQ.1) THEN
      F(4)=Va(4)-VK(IM)-Va(3)*(VK(I)-VK(IM))
      F(5)=Va(5)-YK(IM)-Va(3)*(YK(I)-YK(IM))
      ELSE
      F(4)=Va(4)-VK(I-2)-.5*(Va(3)+1.)*((2.-Va(3))*(VK(IM)-
```

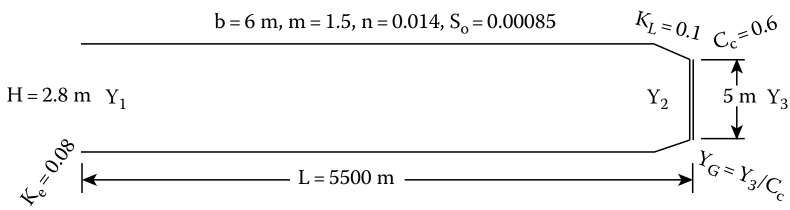
```
     &VK(I-2))+Va(3)*(VK(I)-VK(IM)))
      F(5)=Va(5)-YK(I-2)-.5*(Va(3)+1.)*((2.-Va(3))*(YK(IM)-
     &YK(I-2))+Va(3)*(YK(I)-YK(IM)))
      ENDIF
      ELSE
C Regular points
      AL=(B+FM*Va(7))*Va(7)
      CL=SQRT(G*AL/(B+FM2*Va(7)))
      SFL=(CN*Va(5)*((B+FMS*Va(7))/AL)**EX)**2
      AR=(B+FM*Va(8))*Va(8)
      CR=SQRT(G*AR/(B+FM2*Va(8)))
      SFR=(CN*Va(6)*((B+FMS*Va(8))/AR)**EX)**2
      F(1)=DX*(1.-Va(3))-DT2*(Va(1)+C+Va(5)+CL)
      WS=WSTAGE(Va(2))
      F(2)=Va(1)+WS-Va(5)-WSTAGE(Va(7))-DTG2*(SO-.5*(SF+SFL))
      F(3)=DT2*(C-Va(1)+CR-Va(6))-DX*Va(4)
      F(4)=Va(1)-WS-Va(6)+WSTAGE(Va(8))-DTG2*(SO-.5*(SF+SFR))
      IF(LINEAR.EQ.1) THEN
      F(5)=Va(5)-VK(IM)-Va(3)*(VK(I)-VK(IM))
      F(7)=Va(7)-YK(IM)-Va(3)*(YK(I)-YK(IM))
      ELSE
      IF(I.EQ.2) THEN
      F(5)=Va(5)-VK(1)-.5*Va(3)*((3.-Va(3))*(VK(2)-VK(1))+
     &(Va(3)-1.)*(VK(3)-VK(2)))
      F(7)=Va(7)-YK(1)-.5*Va(3)*((3.-Va(3))*(YK(2)-YK(1))+
     &(Va(3)-1.)*(YK(3)-YK(2)))
      ELSE
      F(5)=Va(5)-VK(I-2)-.5*(Va(3)+1.)*((2.-Va(3))*(VK(IM)-
     &VK(I-2))+Va(3)*(VK(I)-VK(IM)))
      F(7)=Va(7)-YK(I-2)-.5*(Va(3)+1.)*((2.-Va(3))*(YK(IM)-YK
     &(I-2))+Va(3)*(YK(I)-YK(IM)))
      ENDIF
      ENDIF
      IF(LINEAR.EQ.1) THEN
      F(6)=Va(6)-VK(I)-Va(4)*(VK(IP)-VK(I))
      F(8)=Va(8)-YK(I)-Va(4)*(YK(IP)-YK(I))
      ELSE
      IF(IP.EQ.N) THEN
      F(6)=Va(6)-VK(IM)-.5*(Va(4)+1.)*((2.-Va(4))*(VK(I)-
     &VK(IM))+Va(4)*(VK(N)-VK(I)))
      F(8)=Va(8)-YK(IM)-.5*(Va(4)+1.)*((2.-Va(4))*(YK(I)-
     &YK(IM))+Va(4)*(YK(N)-YK(I)))
      ELSE
      F(6)=Va(6)-VK(I)-.5*Va(4)*((3.-Va(4))*(VK(IP)-VK(I))+(Va(4)
     &-1.)*(VK(I+2)-VK(IP)))
      F(8)=Va(8)-YK(I)-.5*Va(4)*((3.-Va(4))*(YK(IP)-YK(I))+(Va(4)
     &-1.)*(YK(I+2)-YK(IP)))
      ENDIF
      ENDIF
      ENDIF
      IF(J.GT.0) Va(J)=Va(J)-DV
```

```
      RETURN
      END
      FUNCTION WSTAGE(YY)
      COMMON V(51),Y(51),VK(51),YK(51),Va(8),F(8),F1(8),D(8,8),
     &Z(8),GBS,DELTDX,FMS,DTG2,FM2,CN,DX,DT2,SO,B,FM,G,DV,DELT,
     &H,FEK,VaN,N,II,IOUT,LINEAR,ITYPE
      COMMON /CSTAGE/ YST(400),W(400),ITAB
      IF(FM.LT.1.E-5) THEN
      WSTAGE=2.*SQRT(G*YY)
      ELSE
      YYY=FM*YY/B
      IF(ITAB.LT.398) THEN
      I1=ITAB+1
      ELSE
      ITAB=398
      I1=399
      ENDIF
      IF(YYY.LT.YST(I1)) THEN
10    IF(YYY.LT.YST(ITAB)) THEN
      ITAB=ITAB-1
      IF(ITAB.GT.1) GO TO 10
      ENDIF
      I1=ITAB+1
      ELSE
20    IF(YYY.GT. YST(I1)) THEN
      I1=I1+1
      IF(I1.LT.400) GO TO 20
      ENDIF
      ITAB=I1-1
      ENDIF
      WSTAGE=GBS*(W(ITAB)+(YYY-YST(ITAB))/(YST(I1)-YST(ITAB))
     &*(W(I1)-W(ITAB)))
      ENDIF
      RETURN
      END
      SUBROUTINE STAGE
      COMMON /CSTAGE/ YST(400),W(400),ITAB
      FC(YP)=SQRT((1.+2.*YP)/(YP+YP*YP))
      DY=.005
      DY1=DY/10.
      DYH=DY1/2.
      N=(2.-.01)/DY+1
      YST(1)=.01
      RC=FC(.01)
      W(1)=.138
      DO 10 I=2,N
      SUM=0.
      DO 5 J=1,10
      YP=YST(I-1)+DY1*FLOAT(J)
      RC1=FC(YP)
      SUM=SUM+DYH*(RC+RC1)
```

```
5      RC=RC1
       YST(I)=YP
10     W(I)=W(I-1)+SUM
       RETURN
       END
```

EXAMPLE PROBLEM 7.3

A reservoir with a constant head of H = 2.8 m supplies a trapezoidal channel with b = 6 m, m = 1.5, n = 0.014 and S_o = 0.00085. At a position 5500 m downstream from the reservoir there is a gate in the channel set 1.00 m above the channel bottom. At this gate position the channel is rectangular with a width of 5 m. The gates contraction coefficient is C_c = 0.6, the entrance loss coefficient is K_e = 0.08, and the loss coefficient at the gate is K_L = 0.10. Determine the flow rate past the gate and the depth upstream from the gate under these conditions. Determine what positions the gate should have if the flow rate is to be increased as follows: t = 40 s, Q = 65 m^3/s; t = 80 s, Q = 110 m^3/s; t = 120 s, Q = 150 m^3/s; t = 160 s, Q = 175 m^3/s; t = 200 s, Q = 180 m^3/s and the flow rate decreases linearly to 50 m^3/s at 800 s, thereafter. What are the depths, velocities, and flow rates throughout the channel during these 800 s? Carry out the simulation in 20 s increments of time and compute the depth, velocity and flow rate at 125 m increments along the channel.



Solution

First a steady-state solution is needed to establish the initial conditions for the unsteady problem. This solution solves the energy equation from downstream of the gate to the trapezoidal channel immediately upstream there from, the energy equation at the entrance of the channel, and the GVF-profile through the length of the channel using the methods described in Chapter 4. The solution is: Q = 34.82 m_3/s, Y_1 = 2.71 m, and Y_2 = 7.30 m (with the depths along the channel given in the **input data** given below). Solving the upstream energy and Manning's equations gives the normal flow rate and depth as Q_o = 62.81 m^3/s and Y_o = 2.391 m (with E_o = 2.76 m and F_{ro} = 0.67). Thus the gate is causing a considerable increase in the depth above normal conditions and decreasing the flow rate below normal. The requested peak flow rate of 180 m^3/s is approximately three times the normal capacity, and therefore to satisfy this request water will need to be taken from channel storage. The input for the above **Hartree** program consists of the following:

Input to solve unsteady problem

```
40 45 1 20 34.82 .014 .00085 6 1.5 9.81 5500 2.8 .08
   7.301  7.195  7.089  6.982  6.876  6.770  6.664  6.558  6.452
   6.346  6.240  6.134  6.028  5.922  5.816  5.711  5.605  5.499
   5.393  5.288  5.182  5.076  4.971  4.865  4.760  4.655  4.550
   4.445  4.340  4.235  4.130  4.026  3.922  3.818  3.714  3.611
   3.508  3.406  3.304  3.202  3.102  3.003  2.904  2.806  2.710
3 6 40 65 80 110 120 150 160 175 200 180 800 50
```

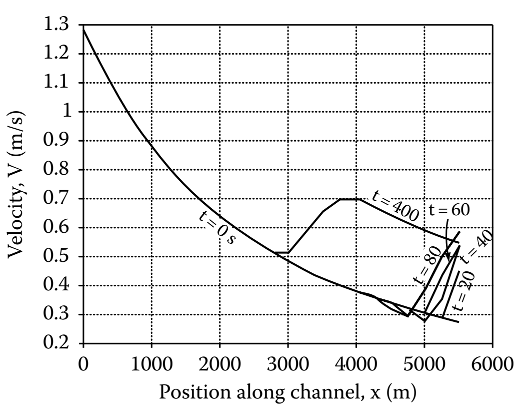
The flow rates, depths, and velocities throughout the channel with time are shown in the three graphs below. The separate lines on these graphs are at 20 s intervals. Note that as the flow rate decreases at the end of the channel after 200 s that the downstream depths again begin to rise at the end of the channel, etc. To determine the gate settings the depths from the unsteady solution at the end of the trapezoidal channel are used in the energy equation:

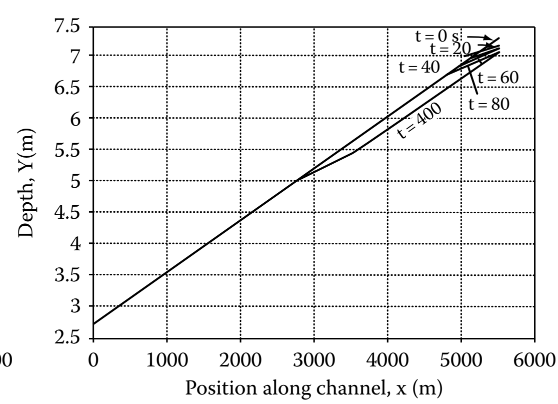
$$Y_2 + \frac{Q^2}{2gA^2} = Y_3 + \frac{(1+K_L)q_3^2}{2gY_3^2} \quad \text{and} \quad Y_G = \frac{Y_3}{C_c}$$

as shown in the table below.

Time (s)	0	40	80	120	160	200	240	280	320
Q_{out} (m³/s)	34.8	65.0	110.0	150.0	175.0	180.0	171.0	163.0	154.0
Depth, Y_2 (m)	7.30	7.18	6.90	6.56	6.32	6.18	6.19	6.26	6.29
Depth, Y_3 (m)	0.64	1.26	2.46	n	n	n	n	n	n
Depth, Y_G (m)	1.07	2.10	4.10						
Energy, E_2 (m)	7.30	7.19	6.95	6.69	6.48	6.37	6.36	6.40	6.42
Crit.D, Y_c (m)	1.70	2.58	3.67	4.51	5.00	5.09	4.92	4.77	4.59
Crit.E, E_c (m)	2.55	3.87	5.50	6.76	7.50	7.64	7.38	7.15	6.89
	360	400	440	480	520	560	640	720	800
	145.0	137.0	128.0	119.0	111.0	102.0	84.7	67.3	50.0
	6.32	6.37	6.42	6.46	6.50	6.55	6.64	6.73	6.82
	n	n	3.50	3.00	2.66	2.34	1.82	1.37	0.98
			5.83	5.00	4.43	3.90	3.03	2.28	1.63
	6.43	6.47	6.50	6.53	6.56	6.60	6.67	6.75	6.83
	4.41	4.27	4.06	3.87	3.69	3.49	3.08	2.64	2.17
	6.61	6.37	6.09	5.79	5.54	5.23	4.62	3.97	3.25

The unsteady solution is based on the trapezoidal channel upstream from the gate with the flow rates specified. There is a problem, however, in getting these flow rates past the gate in the smaller rectangular channel. When it is not possible to pass these flow rates, an n is given in the above table. For these entries, the critical specific energy exceeds that available at the end of the trapezoidal channel (minus the loss through the transition). To illustrate when the upstream unsteady solution is not possible, the last three columns in the table have been added. This problem illustrates that an entire channel system must be examined, and not just portions of it.





From these graphs that display the solution, it can be noted that during the 800 s of the simulation that the effects of changing the flow rate at the downstream end of the channel have propagated to a position just beyond x = 1000 m. It is interesting to numerically determine the position where the wave has propagated to at different times by numerically solving the position of the C^- characteristic that is defined by dx/dt = V − c. The time associated with the wave arriving at any position x is given by $t = L - \int dx/(c - V) = 5500 - \int dx/(c - V)$. Using the trapezoidal rule to carry out the integration to various positions of x provides the times in the last row of the table below. By examining the graphs that provide the solutions the times when the effects have propagated to varies positions can be verified.

x (m)	5500.	5375.	5250.	5125.	5000.	4875.	4750.	4625.
Y (m)	7.301	7.195	7.089	6.982	6.876	6.770	6.664	6.558
V (m/s)	0.281	0.288	0.295	0.303	0.310	0.318	0.327	0.335
c (m/s)	6.596	6.555	6.513	6.471	6.429	6.386	6.343	6.300
t (s)	0.0	19.9	39.9	60.1	80.4	100.9	121.6	142.5
	4500.	4375.	4250.	4125.	4000.	3875.	3750.	3625.
	6.452	6.346	6.240	6.134	6.028	5.922	5.816	5.711
	0.344	0.354	0.363	0.373	0.384	0.395	0.407	0.419
	6.256	6.212	6.167	6.123	6.077	6.032	5.986	5.940
	163.5	184.8	206.2	227.9	249.7	271.8	294.1	316.6
	3500.	3375.	3250.	3125.	3000.	2875.	2750.	2625.
	5.605	5.499	5.393	5.288	5.182	5.076	4.971	4.865
	0.431	0.444	0.458	0.473	0.488	0.504	0.521	0.538
	5.893	5.845	5.797	5.749	5.701	5.651	5.602	5.551
	339.4	362.4	385.7	409.2	433.0	457.2	481.6	506.4
	2500.	2375.	2250.	2125.	2000.	1875.	1750.	1625.
	4.760	4.655	4.550	4.445	4.340	4.235	4.130	4.026
	0.557	0.576	0.597	0.618	0.641	0.666	0.691	0.718
	5.500	5.449	5.397	5.345	5.292	5.238	5.183	5.129
	531.5	556.9	582.8	609.0	635.7	662.8	690.4	718.5
	1500.	1375.	1250.	1125.	1000.	875.	750.	625.
	3.922	3.818	3.714	3.611	3.508	3.406	3.304	3.202
	0.747	0.778	0.810	0.845	0.881	0.920	0.962	1.007
	5.073	5.016	4.959	4.902	4.843	4.784	4.724	4.663
	747.1	776.3	806.1	836.5	867.7	899.7	932.5	966.1
	500.	375.	250.	125.	0.			
	3.102	3.003	2.904	2.806	2.710			
	1.054	1.104	1.158	1.216	1.277			
	4.602	4.541	4.478	4.415	4.352			
	1000.8	1036.6	1073.6	1112.0	1151.8			

7.5 ITERATIVE SOLUTION TECHNIQUE

Rather than using the Newton method to solve eight equations simultaneously, several alternative approaches are available that involve using first-order approximations, and other simplifying assumptions. The procedure described in this section uses an iterative approach, similar to a predictor–corrector method for solving ordinary differential equations, to first establish positions x_L and x_R and then solve the characteristic equations. This approach is more in line with the Hartree method as it was initially proposed. In the next section explicit equations are obtained to evaluate V_L, c_L, V_R, and c_R, after which the characteristic equations are solved.

When these methods were first developed in the pre- or merging-computer era, it was vital to seek means for keeping the amount of computations as minimum as possible. With computer speed (and large memories) even on home desktops, or notebook, computers the concern to minimize computations no longer exists. More important is the accuracy of the results. The simplifications, and resulting reductions in computations, of the methods in this and the next section trade accuracy for amount of computations. This trade-off manifests itself as solutions proceed through many time steps by solution results that do not make physical sense. Therefore, the presentation of the methods occurs to document historic development rather than methods recommended for current computer implementations.

The following steps obtain a solution for V_i^{k+1} and Y_i^{k+1} using an iterative approach:

1. Start by assuming the positions x_R and x_L which can be determined using the slope of the C^+ and C^- characteristics at point i,k, respectively, then (leaving superscript k off x_i)

$$x_L = x_i - \left(V_i^k + c_i^k\right)\Delta t \tag{7.42}$$

$$x_R = x_i - \left(V_i^k - c_i^k\right)\Delta t \tag{7.43}$$

2. Using linear interpolation evaluate the velocity and celerity (or depth Y) at points L and R with the following equations:

$$V_L = V_{i-1}^k + \frac{x_L - x_{i-1}}{\Delta x}\left(V_i^k - V_{i-1}^k\right) \tag{7.44}$$

$$c_L = c_{i-1}^k + \frac{x_L - x_{i-1}}{\Delta x}\left(c_i^k - c_{i-1}^k\right) \tag{7.45}$$

$$V_R = V_{i+1}^k + \frac{x_{i+1} - x_R}{\Delta x}\left(V_i^k - V_{i+1}^k\right) \tag{7.46}$$

$$c_R = c_{i+1}^k + \frac{x_{i+1} - x_R}{\Delta x}\left(c_i^k - c_{i+1}^k\right) \tag{7.47}$$

If $x_L < x_{i-1}$ and/or $x_R > x_{i+1}$, then this interpolation should use positions i – 2 instead of i – 1 and i + 2 instead of i + 1, respectively; Also for supercritical flows ($x_R < x_i$) this interpolation must be adjusted.

3. Integrate Equations 7.6 and 7.8 evaluating the R's in these equations with the Y's at L and R, respectively, or,

$$V_i^{k+1} + w_i^{k+1} = V_L + w_L + \Delta t g(S_o - S_{fL}) \tag{7.48}$$

$$V_i^{k+1} - w_i^{k+1} = V_R - w_R + \Delta t g(S_o - S_{fR}) \tag{7.49}$$

Note that the sum of the left sides of Equations 7.48 and 7.49 is twice V_i^{k+1}, and the difference of Equation 7.48 minus Equation 7.49 is twice w_i^{k+1}.

4. More accurately locate points L and R using the average slopes of the characteristics at the end points using the values determined in step 3, or

$$x_L = x_i - \frac{1}{2}\left(V_i^k + V_L + c_i^k + c_L\right)\Delta t \tag{7.50}$$

$$x_R = x_i - \frac{1}{2}\left(V_i^k + V_R - c_i^k - c_R\right)\Delta t \tag{7.51}$$

5. Repeat steps 2 through 4, except that average values are used to evaluate the integrals of Equations 7.48 and 7.49, e.g., use $\Delta tg[(S_o - .5(S_{fL} + S_{fi}^{k+1})]$ and $\Delta tg[(S_o - .5(S_{fR} + S_{fi}^{k+1})]$, until the change in the positions x_L and x_R is less than some desired amount. This process is analogous to using a corrector in solving ordinary differential equations and typically stops after the first or second iteration.

7.6 EXPLICIT EVALUATION OF VARIABLES AT POINTS L AND R

The values of the variables, V and Y (or c) at the points L and R, can be evaluated explicitly by making some simplifying assumptions. These in turn can be used to solve the characteristic equations. First assume that the slope of the C^+ characteristic is constant over the interval Δt and can be evaluated at point L, then from Equation 7.5 $(x_i - x_L) = (V_L + c_L)\Delta t$. Next assume changes in velocity from point i,k to point L divided by the change in velocity from point i,k to point i − 1,k is proportional to the length ratio $(x_i - x_L)/\Delta x$, or

$$\frac{V_i^k - V_L}{V_i^k - V_{i-1}^k} = \frac{x_i - x_L}{x_i - x_{i-1}} = \frac{(V_L + c_L)\Delta t}{\Delta x}$$

Letting $\Delta V = V_i^k - V_{i-1}^k$ and $\lambda = \Delta t/\Delta x$ this equation becomes

$$V_i^k - V_L = \lambda(V_L + c_L)\Delta V \tag{7.52}$$

Using the same assumption for the change of celerity (with $\Delta c = c_i^k - c_{i-1}^k$) gives

$$\frac{c_i^k - c_L}{\Delta c} = \lambda(V_L + c_L) \tag{7.53}$$

Solving Equation 7.53 for c_L gives

$$c_L = \frac{(c_i^k/\Delta c - \lambda V_L)}{(\lambda + 1/\Delta c)} = \frac{c_i^k - \lambda \Delta c V_L}{1 + \lambda \Delta c} \tag{7.54}$$

Substituting Equation 7.54 in Equation 7.52 gives the following expression that allows V_L to be evaluated from known values at the kth time line

$$\left[\frac{1 + \lambda\Delta V - \lambda^2 \Delta V \Delta c}{(1 + \lambda\Delta c)}\right] V_L = V_i^k - \frac{\lambda \Delta V c_i^k}{(1 + \lambda \Delta c)}$$

or

$$V_L = \frac{[V_i^k - \lambda\Delta V c_i^k/(1+\lambda\Delta c)]}{[1+\lambda\Delta V - \lambda^2\Delta V\Delta c/(1+\lambda\Delta c)]} \tag{7.55}$$

Likewise approximations for the slope of C^- characteristics through point R from Equation 7.7 and assuming this slope is constant to point i, k + 1 then $x_i - x_R = (V_R - c_R)\Delta t$, and thus

$$V_R - V_i^k = \lambda(c_R - V_R)\Delta V \tag{7.56}$$

and

$$c_R - c_i^k = \lambda(c_R - V_R)\Delta c \quad \text{or} \quad c_R = \frac{\left(c_i^k - \lambda\Delta c V_R\right)}{(1-\lambda\Delta c)} \tag{7.57}$$

in which ΔV and Δc are now defined respectively as, $\Delta V = V_{i+1}^k - V_i^k$ and $\Delta c = c_{i+1}^k - c_i^k$. Substituting Equation 7.57 into Equation 7.56 gives the following expression to evaluate V_R explicitly:

$$\left[\frac{1+\lambda\Delta V + \lambda^2\Delta V\Delta c}{(1-\lambda\Delta c)}\right]V_R = V_i^k - \frac{\lambda\Delta V c_i^k}{(1-\lambda\Delta c)}$$

or

$$V_R = \frac{\left[V_i^k - \lambda\Delta V c_i^k/(1-\lambda\Delta c)\right]}{\left[1+\lambda\Delta V + \lambda^2\Delta V\Delta c/(1-\lambda\Delta c)\right]} \tag{7.58}$$

With V_L, c_L, V_R and c_R determined, it is possible to evaluate x_L and x_R from the proportionalities, leading to Equation 7.53 (or Equation 7.28) and to Equation 7.56 (or Equation 7.30). However, since x_L and x_R are not used in computing the other variables, there is no need to compute these values. Thereafter Equations 7.48 and 7.49 need to be solved for V_i^{k+1} and c_i^{k+1} (or Y_i^{k+1}), with the sum and difference of these providing twice the desired values. Since R_{Li}^{k+1} and R_{Ri}^{k+1} in these latter two equations require unknown Y_i^{k+1} they must be solved iteratively. However, such an iterative solution is hardly justified in light of the other assumptions. Using only R_{LL} and R_{RR} i.e., evaluating the slope of the energy line S_f at the current time step k from the velocity and depths known here, the equations are as follows:

$$V_i^{k+1} + w_i^{k+1} = V_L + w_L + \Delta t g(S_o - S_{fL}) \tag{7.59}$$

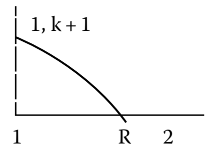
and

$$V_i^{k+1} - w_i^{k+1} = V_R - w_R + \Delta t g(S_o - S_{fR}) \tag{7.60}$$

At the upstream end of the channel where i = 1, it will be assumed again that a reservoir with a known head H(t) supplies the water. Here, Equations 7.57 and 7.58 apply, allowing V_R and c_R to be solved for explicitly with

$$V_R = \frac{\left[V_i^k + \lambda\Delta V c_i^k/(1-\lambda\Delta c)\right]}{\left[1+\lambda\Delta V + \lambda^2\Delta V\Delta c/(1-\lambda\Delta c)\right]} \tag{7.58a}$$

$$c_R = \frac{\left(c_i^k - \lambda\Delta c V_R\right)}{(1-\lambda\Delta c)} \tag{7.57a}$$



Equation 7.60 applies, giving

$$V_1^{k+1} = w_1^{k+1} + V_R - w_R + \Delta t g(S_o - S_{fR}) \tag{7.60a}$$

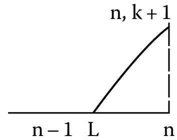
which is substituted in the energy equation for V_1^{k+1} to give the following implicit equation for depth Y_1^{k+1}:

$$F_1 = H(t) - Y_1^{k+1} - (1+K_e)\frac{\left(V_1^{k+1}\right)^2}{2g} = 0$$

At the downstream end of the channel where i = n, Equations 7.55 and 7.54 apply, giving

$$V_L = \frac{\left[V_n^k - \lambda\Delta V c_n^k/(1+\lambda\Delta c)\right]}{\left[1+\lambda\Delta V - \lambda^2\Delta V\Delta c/((1+\lambda\Delta c))\right]} \tag{7.55a}$$

$$c_L = \frac{c_n^k - \lambda\Delta c V_L}{1+\lambda\Delta c} \tag{7.54a}$$



If Y_n^{k+1} is given (specified) at the downstream end, then Equation 7.59 solves for V_n^{k+1} or

$$V_n^{k+1} = V_L + w_L - w_n^{k+1} + \Delta t g(S_o - S_{fL}) \tag{7.59a}$$

If V_n^{k+1} is given (specified) at the downstream end, then Equation 7.59 solves for w_n^{k+1} and there from Y_n^{k+1} is evaluated, or

$$w_n^{k+1} = V_L + w_L - V_n^{k+1} + \Delta tg(S_o - S_{fL}) \tag{7.59b}$$

If Q_n^{k+1} is given (specified) at the downstream end, then Equation 7.59 $V_n^{k+1} = Q(t)/A_n^{k+1}$ and the following implicit equation is solved:

$$F_l\left(Y_n^{k+1}\right) = \frac{Q_n(t)}{A_n^{k+1}} + w_n^{k+1} - V_L - w_L - g\Delta t(S_o - S_{fL}) = 0 \tag{7.59c}$$

and after Y_n^{k+1} is determined V_n^{k+1} is evaluated from Equation 7.59a.

The listing below for the SUBROUTINE SOLVE, that replaces the previous subroutine by this same name and SUBROUTINE FUNCT, obtains solutions to unsteady channel flow problems based on these simplifying assumptions. (Note that arrays F(8), F1(8), D(8,8) and Z(8) in the previous program are not used in this SOLVE and could be deleted from COMMON.)

Replacement subroutine for SOLVE and FUNCT

```
      SUBROUTINE SOLVE
      COMMON V(51),Y(51),VK(51),YK(51),Va(8),F(8),F1(8),D(8,8),
     &Z(8),DELTDX,FMS,DTG2,FM2,CN,DX,DT2,SO,B,FM,G,DV,DELT,H,FEK,
     &VaN,N,II,NEQS,IOUT,LINEAR,ITYPE
      COMMON /CSTAGE/ YST(400),W(400),ITAB
      AMB=DELTDX
      AMB2=AMB*AMB
      Va(1)=VK(1)
      Va(2)=YK(1)
      DO 50 I=1,N
      IP=I+1
      IM=I-1
      C=SQRT(G*(B+FM*Y(I))*Y(I)/(B+FM2*Y(I)))
      IF(I.NE.1) THEN
      DV1=VK(I)-VK(IM)
      DC1=C-SQRT(G*(B+FM*Y(IM))*Y(IM)/(B+FM2*Y(IM)))
      AM1=1.+AMB*DC1
      Va(5)=(VK(I)-AMB*DV1*C/AM1)/(1.+AMB*DV1-AMB2*DV1*DC1/AM1)
      CL=(C-AMB*DC1*Va(5))/AM1
      Va(7)=CL*CL/G
      IF(FM.GT.0.) THEN
      BB=Va(7)-.5*B/FM
      Va(7)=BB+SQRT(BB*BB+B*Va(7)/FM)
      ENDIF
      AL=(B+FM*Va(7))*Va(7)
      R1=Va(5)+WSTAGE(Va(7))+DTG2*(SO-(CN*Va(5)*((B+FMS*Va(7))/
     &AL)**.66666667)**2)
      ENDIF
      IF(I.NE.N) THEN
      DV2=VK(IP)-VK(I)
```

```
      DC2=SQRT(G*(B+FM*Y(IP))*Y(IP)/(B+FM2*Y(IP)))-C
      AM1=1.-AMB*DC2
      Va(6)=(VK(I)+AMB*DV2*C/AM1)/(1.+AMB*DV2+AMB2*DV2*DC2/AM1)
      CR=(C-AMB*DC2*Va(6))/AM1
      Va(8)=CR*CR/G
      IF(FM.GT.0.) THEN
      BB=Va(8)-.5*B/FM
      Va(8)=BB+SQRT(BB*BB+B*Va(8)/FM)
      ENDIF
      AR=(B+FM*Va(8))*Va(8)
      R2=Va(6)-WSTAGE(Va(8))+DTG2*(SO-(CN*Va(6)*((B+FMS*Va(8))/
     &AR)**.66666667)**2)
      ENDIF
      IF(I.EQ.1) THEN
C  UPSTREAM B.C.
      NCT=0
2     Va(1)=R2+WSTAGE(Va(2))
      F1=H-Va(2)-FEK*Va(1)**2
      DV=.005*Va(2)
      F11=H-Va(2)-DV-FEK*(R2+WSTAGE(Va(2)+DV))**2
      DIF=DV*F1/(F11-F1)
      Va(2)=Va(2)-DIF
      NCT=NCT+1
      IF(NCT.LT.20.AND.ABS(DIF).GT..000001)GO TO 2
      ELSE IF(I.EQ.N) THEN
C DOWNSTREAM B.C.
      IF(ITYPE.EQ.1) THEN
      Va(1)=R1-WSTAGE(VaN)
      Va(2)=VaN
      ELSE IF(ITYPE.EQ.2) THEN
      WK1=R1-VaN
      IF(FM.GT.0.) THEN
      Va(2)=B*YSTAGE(WK1)/FM
      ELSE
      Va(2)=(.5*WK1)**2/G
      ENDIF
      Va(1)=VaN
      ELSE
      NCT=0
4     F1=VaN/((B+FM*Va(2))*Va(2))+WSTAGE(Va(2))-R1
      DV=.005*Va(2)
      VA2=Va(2)+DV
      F11=VaN/((B+FM*VA2)*VA2)+WSTAGE(VA2)-R1
      DIF=DV*F1/(F11-F1)
      Va(2)=Va(2)-DIF
      NCT=NCT+1
      IF(NCT.LT.20 .AND. ABS(DIF).GT. .000001) GO TO 4
      Va(1)=R1-WSTAGE(Va(2))
      ENDIF
      ELSE
```

```
C REGULAR POINTS
      Va(1)=.5*(R1+R2)
      WK1=.5*(R1-R2)
      IF(FM.GT.0.) THEN
      Va(2)=B*YSTAGE(WK1)/FM
      ELSE
      Va(2)=(.5*WK1)**2/G
      ENDIF
      ENDIF
      V(I)=Va(1)
50    Y(I)=Va(2)
      RETURN
      END
```

7.7 ACCURACY OF NUMERICAL SOLUTIONS

An important question associated with numerical approximations is: How close is the solution to the true solution? Since the "true" solution cannot be obtained generally there is no way of answering this question precisely as a percentage, or a maximum deviation, of depth velocity or flow rate. However, methods can be employed that provide indications of whether the solution is reasonably close to the true solution that it might be used for practical applications. After all the St. Venant equations are only a mathematical model of the open channel's real behavior anyway. A commonly used method is to obtain several numerical solutions with a variety of spacings Δx and time increments Δt and when the solutions are "essential" the same one often is willing to accept the numerical solution as sufficiently accurate. Another means for assessing the accuracy is to formulate a problem that requires essentially as much sophistication from the numerical solution, but for which the answer is known, and then study the deviations that occur based on varying parameters, etc. that are used in the numerical solution. This section illustrates such an approach. However, with space limitations only the first Hartree method that is based on second-order approximations will be examined. The reader can carry out similar studies using the other methods.

The table below gives the solution obtained using the HARTREE program compared with a characteristic solution obtained using the methods of Chapter 6 in which uniform flow with Y_o = 6 ft, and V_o = 4 fps was given as the initial condition, and the depth at the downstream end was specified to decrease at a rate dY/dt = −.01 ft/s. In obtaining the solution using the HARTREE program the following input data were used, which specifies Δx = 50 ft, and Δt = 2.5 s:

```
40 31 1 2.5 4 24 0. 0. 1. 0. 32.2 1500. 6.2484472 0.
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
1 11 0 6 10 5.9 20 5.8 30 5.7 40 5.6 50 5.5 60 5.4 70 5.3 80 5.2 90 5.1
  100 5.
```

Note both S_o = 0 and n = 0, so it solves the idealized inviscid unsteady flow accommodated by the methods of Chapter 6. The two solutions are identical to at least two significant digits. The HARTREE solution does not give the position of the wave where the disturbance has propagated to, so the last column in this table has been obtained by noting where depths, etc. first begin to change so positions are only given to within Δx = 50 ft.

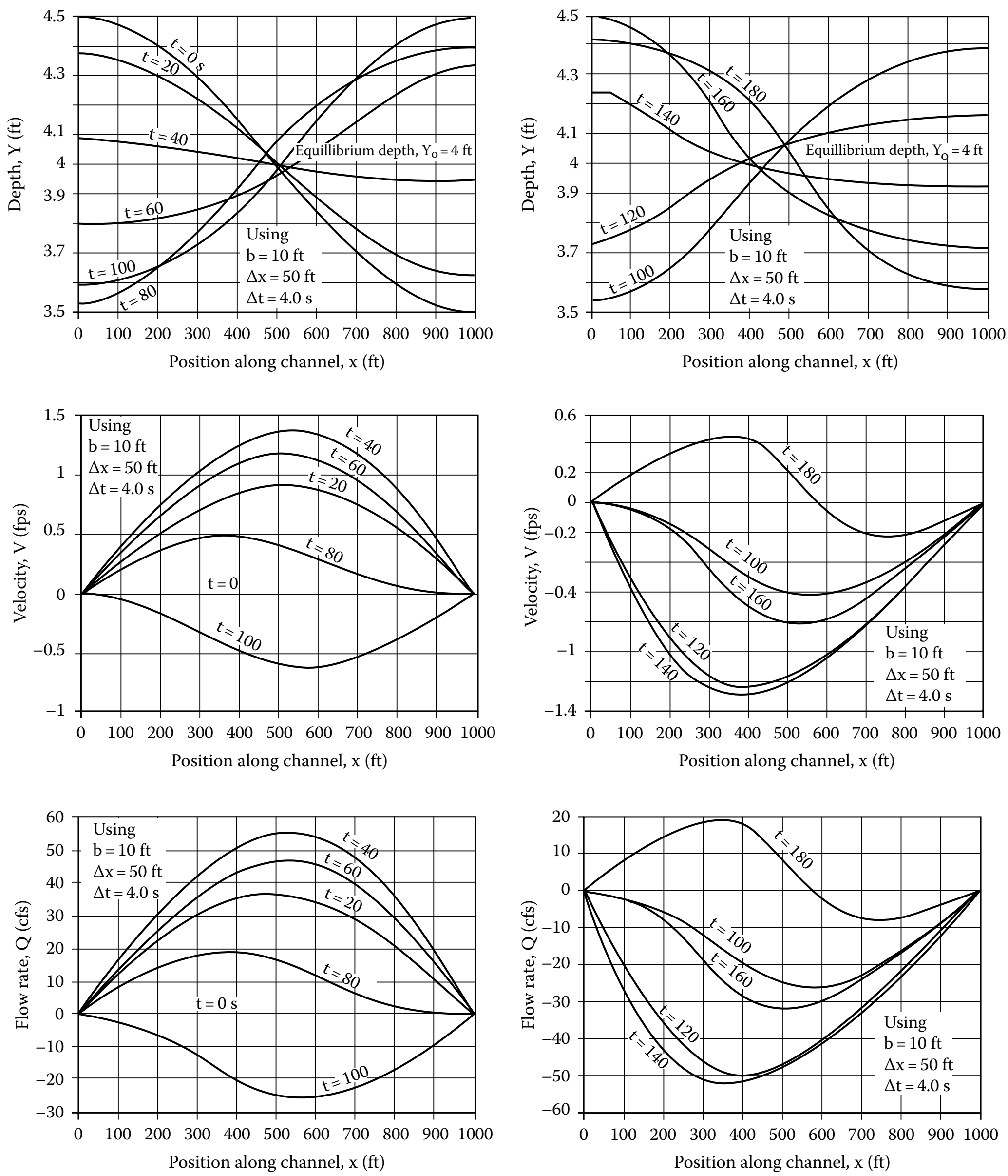
The table comparing numerical solution obtained by the HARTREE program with a solution obtained using the methods of Chapter 6 is given below. Channel length = 1500 ft, S_o = 0, n = 0,

$Y_o = 6$ ft, $V_o = 4$ fps, Depth Change at downstream end = dY/dt = 0.01 ft/s, Head at upstream reservoir H = 6.2484472 ft.

Time t	Chapter 6 Methods				HARTREE Program		
	Downstream				Downstream		
(sec)	Depth (ft)	Vel. (fp)	q (cfs/ft)	Pos. x (ft)	Vel. (fps)	q (cfs/ft)	Pos. x (ft)
0	6.0	4.00	24.0	0	4.0	24.0	0
10	5.9	4.23	25.0	99	4.23	25.0	100
20	5.8	4.47	25.9	198	4.47	25.9	200
30	5.7	4.70	26.8	296	4.70	26.8	300
40	5.6	4.94	27.7	396	4.94	27.7	400
50	5.5	5.18	28.5	495	5.18	28.5	500
60	5.4	5.43	29.3	594	5.43	29.3	600
70	5.3	5.67	30.1	693	5.67	30.1	700
80	5.2	5.92	30.8	792	5.92	30.8	800
90	5.1	6.17	31.5	891	6.17	31.5	900
100	5.0	6.42	32.1	990	6.42	32.1	1000

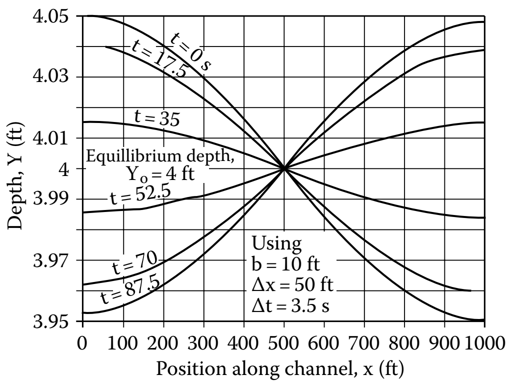
The above comparison is for a simple case with only a disturbance at the downstream end. To examine the accuracy of the Hartree methods for a more complex problem, consider the following idealized channel problem: Frictionless water confined between closed gates within a given length of horizontal channel (or between two vertical wall across a channel) is initially displaced by some known function from its equilibrium constant depth. As the displacement function the cosine over two quadrants, in which the angle varies from 0 to π will be used. Thus at one end the depth will be displaced upward from the equilibrium depth Y_o by an amount ΔY while the other end the depth will be $Y = Y_o - \Delta Y$, and at the center the depth will be $Y = Y_o$, or the equilibrium value, and the intermediate depths vary as the cosine (see figure below that shows the depth for t = 0 s). Initially the velocity throughout the channel is zero. The fluid at the end with the raised deep will accelerate downward while the end with the suppressed depth will accelerate upward, causing a velocity within the channel from the high end to the low end. However, because the celerity is larger at the end with the greater depth both ends will not arrive at the equilibrium depth Y_o at the same time. In addition to the effects of varying c's, the velocity will develop nonsymmetrically in the direction from the falling depth to the rising depth, and vice versa, that further influences the rates of propagation of the characteristics. Because of the fluid velocities when the depth passes through the equilibrium depth, the end with the initially raised depth will have its depth drop an amount close to ΔY while the other end will tend toward a depth $Y = Y_o + \Delta Y$, and this oscillatory wave will continue with further distortions due to the different celerities throughout the channel. These movements are all based on the assumption that no fluid friction exists, so $S_f = 0$ and the channel is horizontal. As the ratio $\Delta Y/Y_o$ approaches zero, the distortion due to differences in c will become smaller and the cosine wave will tend to more closely duplicate itself with a given period of oscillation.

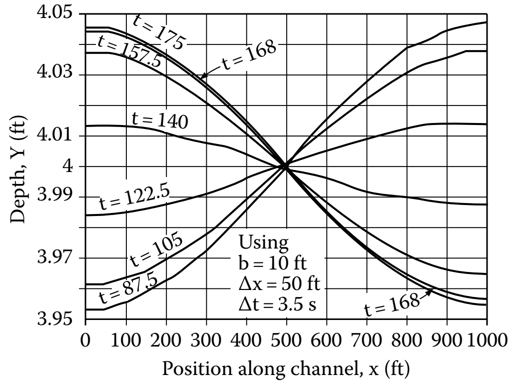
To solve this problem the program HARTREE must be modified so that both the upstream and downstream boundary conditions specify that the velocity is zero. Making these modifications to the HARTREE program is left as a homework problem. The depths, velocities, and flow rates from a solution obtained in a rectangular channel that was specified to be 1000 ft long, with a bottom width b = 10 ft, $S_o = 0$, and n = 0, with an equilibrium depth of $Y_o = 4$ ft, and $\Delta Y = 0.5$ ft and solving the problem with $\Delta x = 50$ ft, and $\Delta t = 3.5$ s are plotted in group of figures below.

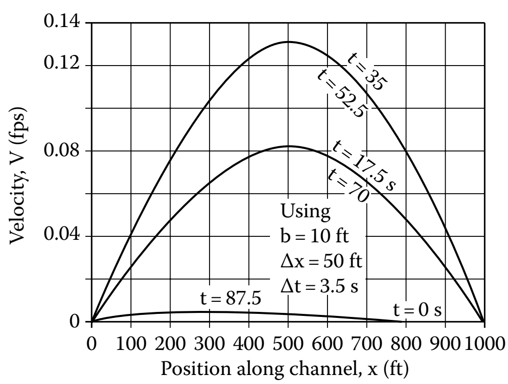


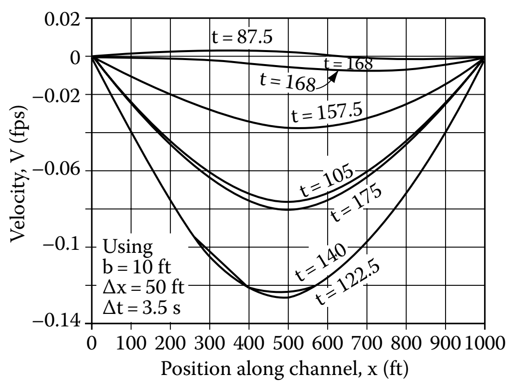
The same problem was solved using Δx = 25 ft, and Δt = 1.75 s (1/2 of the above two increments), with results that cannot be distinguished from this solution when plotted on a similar figure. Using Δt = 3.5 s with Δx = 50 ft resulted in no violations of the Courant condition, i.e., Δt was always less than Δx/(V + c) or Δx/(c − V) so that the graphs are not too cluttered. Two graphs are shown for each quantity; the first for the cycle in which the initially raised side falls to near the lowest position, and the second graph for the return to the near high position. Since the period of oscillation is part of the solution, the exact time of these extremes is not given, but only the results from every fifth solution, i.e., separate curve on figures are on a time increment of 17.5 s. This solution was obtained using linear interpolation (i.e., LINEAR = 1) to determine variables at points R and L. If quadratic interpolation is used this attenuation almost disappears. (This is left as a homework problem.) Note there is some attenuation, i.e., the depth initially at 4.5 ft at the beginning of the channel return to a depth about 0.05 ft less than this, with similar result from the end of the channel. Note also that the depth at the mid-position x = 500 ft does not remain constant, rather the position of Y_o for increasing times moves downstream from 500 ft during the first half cycle (the first plots), and upstream during the second half cycle (the second plots). These effects are expected because c is larger as the depth are larger and smaller as the depths are smaller.

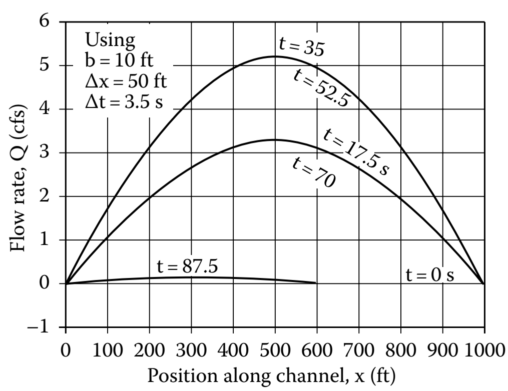
In this idealized example, the oscillations will tend toward reproducing the initial cosine function repeatedly as either ΔY approaches zero or Y_o approaches infinity; in other words as $\Delta Y/Y_o$ becomes very small the celerity c will be constant throughout the entire channel, and the influence of the velocity V is insignificant. The group of figures below show the results of reducing ΔY in the previous idealized problem from 0.5 to 0.05 ft. Note now that the depth at the midpoint is close to the equilibrium depth Y_o = 4 ft at all times, and the velocities and flow rates are nearly symmetric functions around this mid point. In this example the effect of varying celerity and the velocity are much smaller and a nearly repetitive oscillation is given as the solution.

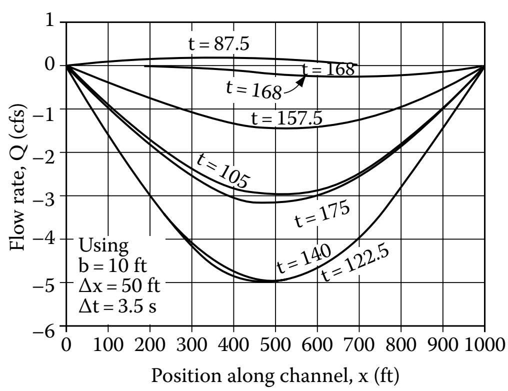




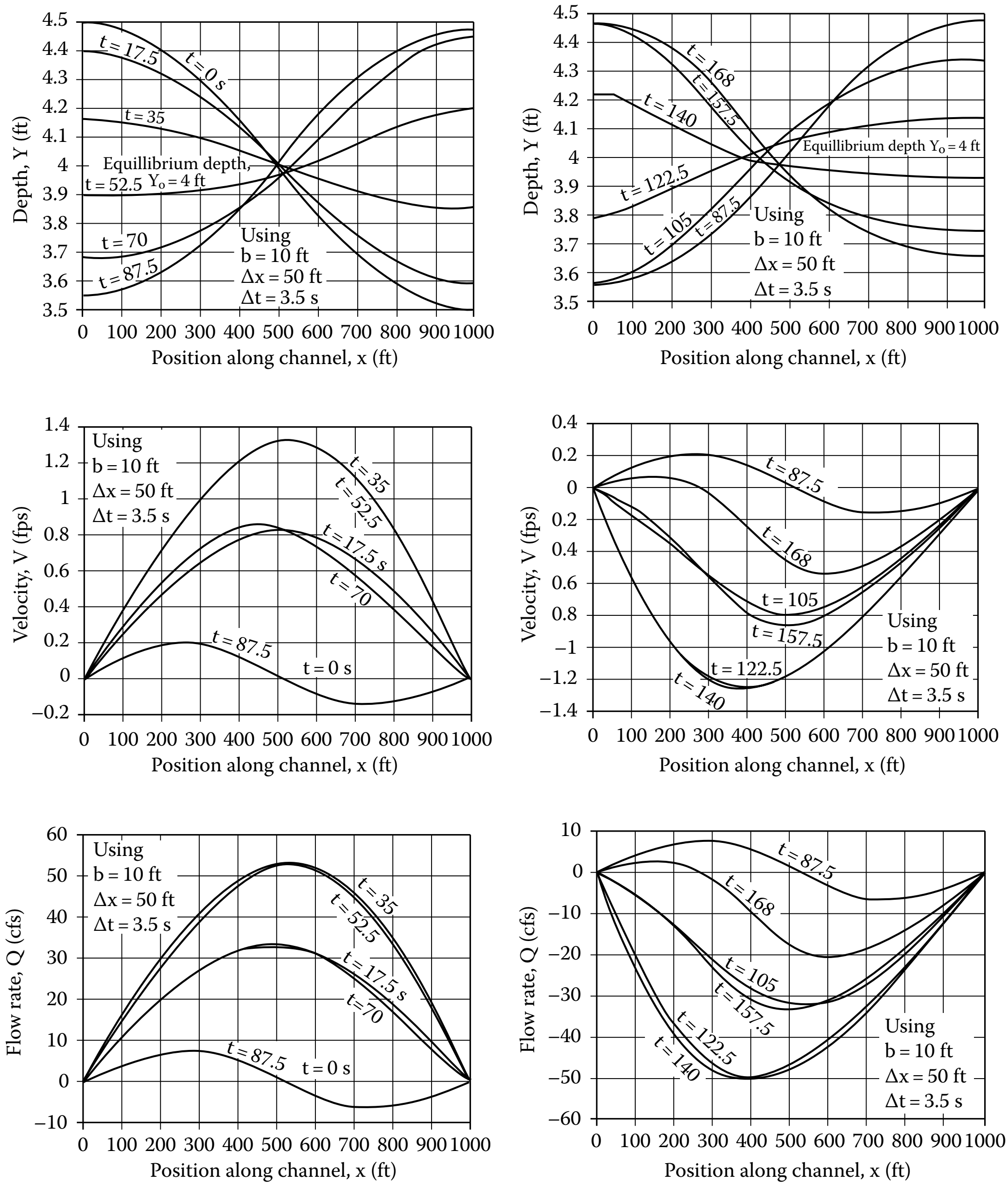








Finally the group of figures below show a similar solution with $\Delta x = 50$ ft and $\Delta t = 4$ s. If the velocities are ignored the 4 s time increment would appear to satisfy the Courant condition, i.e., $\Delta x/c = 50/(32.2 \times 4.5)^5 = 50/12.037 = 4.154$; however because of the velocities that develop this condition was violated at 11.7% of the computational points throughout a simulation period of 340 s. The violation was only modest however, with the smallest Δt required equal to 3.929 s. The effect of this violation is only slightly apparent in the solution displayed below since it only gives the result for one cycle.



However, during the next cycle the results from this solution deviate considerably from the solution obtained with $\Delta t = 3.5$ s.

The above solutions were obtained assuming that the fluid is inviscid by assigning Manning's n a value of zero. Including frictional loss will damp out the oscillations, but generally since the velocities developed from the oscillating water surface are small the amount of damping is not large. The damping effect is left as a homework problem for you. The above solutions also use one-dimensional hydraulics, i.e., the velocity, etc. are constant through the vertical at any position, x. In reality the velocities will be smaller near the bottom than at and near the surface, and the pressures will deviate slightly from hydrostatic. Accounting for these effects would require a two-dimensional (or possibly even three-dimensional) formulation of the problem, which is beyond the scope of this text.

7.8 IMPLICIT METHODS

A disadvantage to using the method of characteristics is that the variables are computed at irregular points throughout the xt plane. Thus interpolation is needed when one wishes to describe what the depth, flow rate, velocity, etc. are throughout the channel at any given time, or how these variables change at any position as a function of time. Consequently numerical methods that use fixed grids have been preferred by most users. There are two basic types of fixed grid finite difference methods for solving the St. Venant equations, **explicit** and **implicit**. In an explicit method, such as the Hartree method, the finite difference equations are developed in such a manner so that they produce values for the unknowns at one point at a time, very similar to how the method of characteristics produces values at a new point based on known values at the other points involved in writing the equations. With the grid points along the channel fixed, x_m and t_m are unknown. An explicit method of solution advances the solution through the next time step (which is at a known constant time) by solving unknowns one point at a time. In other words, values at the point are found independently of values at other points at the advanced time step. Consequently solutions based on explicit methods limit the region that can influence the solution to a pyramid-shaped area below the point in the xt plane. The effect is that the increment Δt must be limited in size in relationship with the space increment Δx or else the solution will be meaningless. In most problems this limiting criteria is along the C^+ characteristics and is that $\Delta t \le \Delta x/(V + c)$ anywhere throughout the length of channel. If the C^- characteristics limit the size of the time increment then $\Delta t \le \Delta x/(V - c)$. The most restrictive is required, and the user must be aware of this limitation in defining Δx and Δt as input to a program such as HARTREE. The program could check this criteria and either terminate or reduce Δt when it is not satisfied.

Implicit methods solve all the unknowns along the advanced time step simultaneously, on the other hand. In other words, the finite differencing of the St. Venant equations (in one of their several forms) occurs in such a manner as to link the unknowns in the advanced time line together and thereby produce a system of simultaneous algebraic equations. Implicit methods are not restricted to using very small Δt, as are explicit methods, but removal of this restriction is at the expense of more complicated calculations and procedures because a full system of equations must be solve simultaneously. The system of equations involves what might be considered regular equations, that come from interior grid points, and boundary equations that provide a mathematical description to define what occurs at the beginning and end of the channel. The equations from interior grid points are of the same form but different coefficients that come from known values at different positions along the channel. The boundary equations are generally different than those at interior points, and must be obtained specifically to define what is specified at the given boundary.

To assist in learning how implicit methods can be utilized, a so-called direct implicit method will be described first. This direct method allows for the solution by only using linear algebra to obtain the solution to the system of equations. Since the St. Venant equations are nonlinear, this linearization must be at the expense of accuracy. Later implicit methods will be described that require the solution to nonlinear systems of equations.

7.8.1 Direct Implicit Finite Differencing

In this section, an implicit method of differencing the St. Venant equations proposed by Stelkoff 1970 will be discussed. Strelkoff showed that if the slope of the energy line, S_f, is evaluated on the advanced time step that a solution to the resulting system of equations will supply stable values regardless of how large Δt is. The only restriction is to use Δt and Δx small enough so that the numerical solution is sufficiently accurate. This method will use a rectangular grid network in the xt plane, as shown below in Figure 7.4. The vertical grid lines will be spaced at a constant Δx, and will be constant across the entire x distance for any time step, and generally Δt will be constant through time, but could vary if the rate of changes suggests this be done to improve the accuracy of the solution.

The Q Y form of the St. Venant equations will be differenced to add some variety to the equations used since the V Y equations were used in the last section to obtain the characteristic equations. These are Equations 6.1 and 6.2 from Chapter 6 and are repeated below:

1. Continuity

$$\frac{\partial Q}{\partial x} - q* + \frac{\partial A}{\partial t} = \frac{\partial Q}{\partial x} - q* + T\frac{\partial Y}{\partial t} = 0 \tag{7.61}$$

and

2. Equation of motion

$$\frac{2Q}{gA^2}\frac{\partial Q}{\partial x} + (1-F_r^2)\frac{\partial Y}{\partial x} - \frac{Q^2}{gA^3}\left.\frac{\partial A}{\partial x}\right|_{Y,t} + S_f - S_0 + F_q + \frac{Qq*}{gA^2} + \frac{1}{gA}\frac{\partial Q}{\partial t} = 0 \tag{7.62}$$

in which $F_q = 0$ for bulk lateral outflow, $F_q = (Vq*)/(2gA) = (Qq*)/(2gA^2)$ for seepage flow, and $F_q = (V - U_q)q*/(gA) + (h_c/A)(\partial A/\partial x)|_{Y,t}$ for lateral inflow.

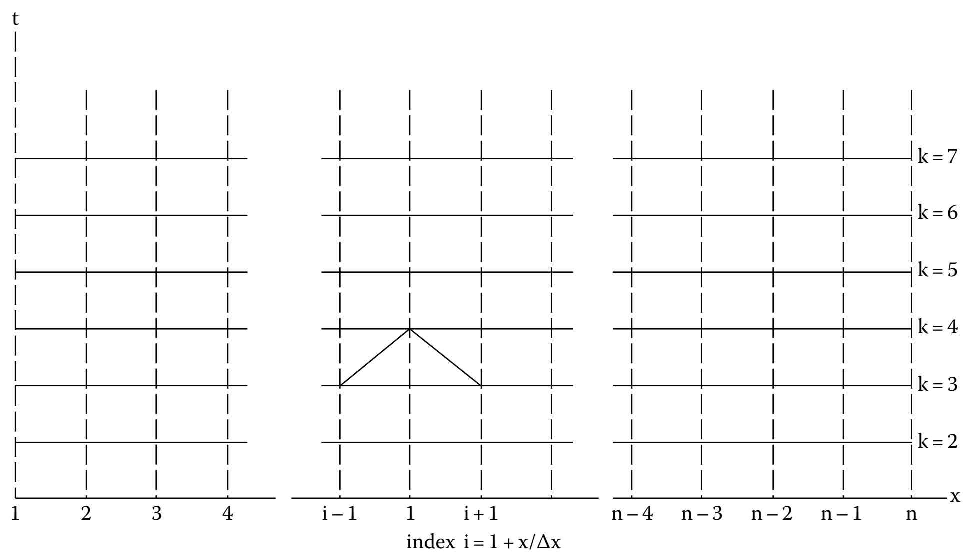


FIGURE 7.4 Finite difference grid in the xt plane at which values for the dependent variables Y and Q will be solved.

Multiplying Equation 7.62 by gA and replacing Q/A by the velocity V results in

$$2V\frac{\partial Q}{\partial x}+T(c^2-V^2)\frac{\partial Y}{\partial x}-V^2\left.\frac{\partial A}{\partial x}\right|_{Y,t}+gA(S_f-S_0+F_q)-Vq^*+\frac{\partial Q}{\partial t}=0 \qquad (7.62a)$$

To simplify the writing of equations the conveyance from Manning's equation will be defined as

$$K=\frac{C_uA^{5/3}}{nP^{2/3}} \qquad (7.63)$$

in which $C_u = 1.486$ for ES units and $C_u = 1.0$ for SI units. The slope of the energy line can then be written as

$$S_f=\frac{n^2Q\,|Q|\,P^{4/3}}{C_u^2A^{10/3}}=\frac{Q\,|Q|}{K^2} \qquad (7.64)$$

By taking |Q|Q rather than Q^2 accounts properly for the sign, so if Q is negative S_f is also negative. Using only first-order terms in a Taylor series allows S_f at the advanced time step to be approximated by

$$(S_f)_i^{k+1}=(S_f)_i^k+\left(\frac{\partial S_f}{\partial Q}\right)_i^k\left(Q_i^{k+1}-Q_i^k\right)+\left(\frac{\partial S_f}{\partial K}\right)_i^k\left(\frac{\partial K}{\partial Y}\right)_i^k\left(Y_i^{k+1}-Y_i^k\right) \qquad (7.65)$$

in which the derivatives in this equation are

$$\frac{\partial S_f}{\partial Q}=\frac{2S_f}{Q} \qquad (7.66)$$

$$\frac{\partial S_f}{\partial K}=-\frac{2S_f}{K} \qquad (7.67)$$

$$\frac{\partial K}{\partial Y}=\frac{K}{A}\left(\frac{5}{3}T-\frac{2}{3}\frac{A}{P}\frac{\partial P}{\partial Y}\right) \qquad (7.68)$$

The finite difference equations are obtained by replacing the space derivatives with respect to x by second-order central differences, such that, for example, $\partial Q/\partial x \approx (Q_{i+1} - Q_{i-1})/(2\Delta x)$ with a superscript k or k + 1 depending upon whether it is to be evaluated on the current or next time step. The St. Venant equations are then approximated by the following: (these equations are written with the unknowns with a superscript k + 1 on the left of the equal sign and Y and Q with a k superscript (known) on the right of the equal sign.)

$$-0.5Q_{i-1}^{k+1}+\frac{\Delta xT_i^k}{\Delta t}Y_i^{k+1}+0.5Q_{i+1}^{k+1}=\frac{\Delta xT_i^k}{\Delta t}Y_i^k+\Delta xq_i^{k+1} \qquad (7.69)$$

and

$$
\begin{aligned}
& -\frac{T_i^k(c^2-V^2)_i^k}{2\Delta x}Y_{i-1}^{k+1}-\frac{V_i^k}{\Delta x}Q_{i-1}^{k+1}-2g\left[S_f\left(\frac{5}{3}T-\frac{2}{3}\frac{A}{P}\frac{\partial P}{\partial Y}\right)\right]_i^k Y_i^{k+1} \\
& +\left[\frac{1}{\Delta t}+2g\frac{AS_f}{Q}\right]_i^k Q_i^{k+1}+\frac{T_i^k(c^2-V^2)_i^k}{2\Delta x}Y_{i+1}^{k+1}+\frac{V_i^k}{\Delta x}Q_{i+1}^{k+1} \\
& =\frac{Q_i^k}{\Delta t}+\left[V^2\frac{\partial A}{\partial x}\bigg|_{Y,t}\right]_i^k+g\left\{A\left[S_o-F_{q_i}^{k,k+1}-(S_f)_i^k+\left(\frac{2S_f}{Q}\right)_i^k Q_i^k\right]\right\} \\
& \qquad -g\left\{2(S_f)_i^k\left(\frac{5}{3}T-\frac{2}{3}\frac{A}{P}\frac{\partial P}{\partial Y}\right)_i^k Y_i^k\right\}+(Vq)_i^k
\end{aligned}
\tag{7.70}
$$

in which $F_{q_i}^{k,k+1}$ is computed according to what F_q represents at grid point i with V evaluated on the k-th time step, and q and U_q at the k + 1 time step.

To simplify writing of Equations 7.69 and 7.70 the following symbols will be defined:

$$
d_1=\frac{T(c^2-V^2)}{(2\Delta x)},\quad d_2=\frac{V}{\Delta x},\quad d_3=2gS_t\left(\left[\frac{2A}{(3P)}\right]\left[\frac{\partial P}{\partial Y}\right]-\frac{5T}{3}\right),\quad d_4=\frac{1}{\Delta t}+\frac{2gAS_t}{Q}
$$

and

$$
e_1=\frac{\Delta xT}{\Delta t}
$$

All of these new variables are evaluated at position i and at time step k. Then Equations 7.70 and 7.69 can be written, respectively, as follows for trapezoidal cross sections:

$$
\begin{aligned}
& -d_1Y_{i-1}^{k+1}-d_2Q_{i-1}^{k+1}+d_3Y_i^{k+1}+d_4Q_i^{k+1}+d_1Y_{i+1}^{k+1}+d_2Q_{i+1}^{k+1} \\
& =\frac{Q_i^k}{\Delta t}+\left[F_y+\left(\frac{V^2+H_c}{2\Delta x}\right)\left\{\left(b_{i+1}-b_{i-1}\right)+Y_i^k(m_{i+1}-m_{i-1})\right\}+gA_i^k\left(S_0-D_L+S_{f_i}^k\right)\right]Y_i^k+\frac{V_i^kq*}{\Delta x}
\end{aligned}
\tag{7.70a}
$$

in which $D_L=V_i^kq*/(g\Delta xA_i^k)$ for inflow and 1/2 this value for outflow and $F_y=2gA_{fi}^k(1.33333/\sqrt{m^2+1}\cdot A/P-1.666667T)_i^k$, $H_c=gh_c$ for lateral inflow (with $U_q=0$) and $H_c=0$ otherwise, and

$$
-0.5Q_{i-1}^{k+1}+e_1Y_i^{k+1}+0.5Q_{i+1}^{k+1}=e_1Y_i^k+q_i^k \tag{7.69a}
$$

In examining the above equations, it can be seen that the method used to difference the partial differential equations has produced linear algebraic equations in the unknown flow rate Q^{k+1} and unknown depth Y^{k+1}. Thus the implementation of advancing the solution through a time increment does not require an iterative solution, but can be accomplished through the use of linear algebra only. This approach eliminates the need to solve a nonlinear system of equations at the expense of less accuracy. Clearly, a better approach would be to evaluate the coefficients for the Y's and Q's at the k + 1 and the k time steps depending upon whether the term it multiplies is at the k + 1 or k time step, respectively. However, doing the latter results in a system of nonlinear algebraic equations. Such a system will be dealt with later. Using coefficients evaluated at the kth time step is an improvement over the techniques of Chapter 6 in which $S_o - S_f$ was assumed equal to 0, taking the coefficient from an initial uniform flow condition.

When characteristics were used the stage variable was introduced to accommodate nonrectangular channels. Now with the implicit method of differencing, this is not necessary because the characteristic form of the St. Venant equations do not need to be used. When using the characteristics, it was necessary to have these defined by equations and also have the special ordinary differential equations which apply along these characteristics defined. Now the original St. Venant equations can be used in either their Q Y form, or in the V Y form. Thus an additional advantage of using a regular grid in the xt plane is that channels of any geometric shape can be handled without the need for introducing another variable, which adds another equation to the mathematical problem that must be solved. With the implicit method, the only difference between solving unsteady flows in rectangular and nonrectangular channels is that a slightly larger amount of arithmetic is required to compute quantities such as the area and wetted perimeter to evaluate S_f, etc.

7.8.1.1 Boundary Conditions

At the boundaries i = 1 and i = n, difference equations must be obtained from boundary conditions which appropriately define the actual flow conditions at these ends. These conditions must reflect what is known at the boundary and use an appropriate relationship between the known(s) and unknown(s). The continuity equation

$$\frac{\partial Q}{\partial x} - q^* = -T\frac{\partial Y}{\partial T}$$

expressed in some difference form is generally available. If either Y or Q at the boundary is known as a function of time, it is all that is needed since only the other variable Q or Y is unknown at the boundary. When both Y and Q are unknown at the boundary, then some other relationship is needed. For instance, consider the problem in which the depth at the downstream end is varied as a function of time by raising or lowering a gate, then when i = n, $Y(L, t) = Y_b(t)$ (a known function of time), and the condition for Q(L, t) must satisfy the continuity equation. Using second order differences to evaluate $\partial Q/\partial x$ leads to

$$\tfrac{1}{2}Q_{n-2} - 2Q_{n-1} + \tfrac{3}{2}Q_n = \Delta x\left(q^* - T\left\{\frac{\partial Y_b}{\partial t}\right\}\right)$$

all evaluated at the k + 1 time step as the boundary condition operator for Q at i = n, in which $\partial Y_b/\partial t$ is the rate at which the depth changes as a function of time and is a known value.

If the flow rate is known at the downstream boundary as a function of time, then $Q(L, t) = Q_b(t)$, and the depth Y_n is unknown. For this case, the finite difference form of the continuity equation can be written as

$$\tfrac{1}{2}Q_{n-2}^{k+1} - 2Q_{n-1}^{k+1} + \tfrac{3}{2}Q_n^{k+1} - \Delta x(q^*)^{k+\frac{1}{2}} + \frac{\Delta x T_n^k}{\Delta t}\left(Y_n^{k+1} - Y_n^k\right) = 0$$

Since Q_n^{k+1} is known, the equation can be written as below in which the knowns have been moved to the right of the equal sign:

$$\tfrac{1}{2}Q_{n-2}^{k+1} - 2Q_{n-1}^{k+1} + \frac{\Delta x T_n^k}{\Delta t}Y_n^{k+1} = \Delta x(q^*)^{k+\frac{1}{2}} - \tfrac{3}{2}Q_n^{k+1} + \frac{\Delta x T_n^k}{\Delta t}Y_n^k \tag{7.71}$$

For the upstream end of the channel, a common condition occurs when the channel is supplied water from a constant head reservoir. In this case neither the depth Y_1 or the flow rate Q_1 is known, and therefore two equations are needed at this point to solve for the two unknowns. One of these equations is the continuity equation, and the other is the entrance energy equation,

$$H = Y + (1 + k_e)\frac{Q^2}{2gA^2}$$

To be consistent with establishing a linear system of finite difference equations, a linear relationship between Q_1 and Y_1 is sought that replaces the above nonlinear entrance energy equation. Such an equation can be written as

$$Y_1^{k+1} = Y_1^k + \frac{dY}{dQ}\left(Q_1^{k+1} - Q_1^k\right) = Y_1^k + c_1\left(Q_1^{k+1} - Q_1^k\right) \tag{7.72}$$

in which the slope $dY/dQ = c_1$ is determined from the entrance energy equation. Writing this energy equation as

$$\frac{Q^2}{2g} = A^2(H - Y)$$

and then taking the differentials of both sides gives

$$\frac{QdQ}{g} = 2ATdY(H - Y) - A^2dY = A[2(H - Y)T - A]dY \quad \text{or} \quad c_1 = \frac{Q}{(gA[2(H - Y)T - A)} \tag{7.73}$$

Thus one finite difference equation for the beginning of the channel is

$$Y_1^{k+1} - c_1Q_1^{k+1} = Y_1^k - c_1Q_1^k \tag{7.74}$$

The second equation needed to solve for the two unknowns Y_1^{k+1} and Q_1^{k+1} comes from differencing the continuity equation. There are a number of methods for doing this. Some of the methods will cause the solution process to diverge, or be unstable. For example using a second-order approximation for $\partial Q/\partial x$ at time step k + 1 such as the following:

$$-\frac{3}{2}Q_1^{k+1} + 2Q_2^{k+1} - \frac{1}{2}Q_3^{k+1} - \Delta xq + \frac{\Delta xT}{\Delta t}\left(Y_1^{k+1} - Y_1^k\right) = 0$$

is unstable. A stable approach takes the average of first-order derivative approximations of $\partial Q/\partial x$ at the k and k + 1 time steps resulting in

$$\frac{\Delta xT}{\Delta t}Y_1^{k+1} - \frac{1}{2}Q_1^{k+1} + \frac{1}{2}Q_2^{k+1} = \frac{\Delta xT}{\Delta t}Y_1^k + \frac{1}{2}Q_1^k - \frac{1}{2}Q_2^k - \Delta xq_1^{k+\frac{1}{2}} = 0 \tag{7.75}$$

A more general boundary condition would be to have the stage–discharge relationship specified by input data. A stage–discharge relationship specifies the flow rate as a function of depth. This stage–discharge relationship could be generated by solving the entrance energy equation, and providing such values would essentially duplicate the use of Equation 7.74, when properly implemented into computer code. However, a stage–discharge relationship can handle other possible problems, such as how the flow rate past a gate may be effected by a reduction in the water depth downstream of the gate, etc.

Other boundary conditions are possible. For problems with gate control upstream either the depth $Y(0,t) = Y_b(t)$ or the flow rate $Q(0,t) = Q_b(t)$ might be specified. The above conditions, however, illustrate how the same number of equations can be obtained as there are unknowns.

7.8.1.2 Solving the Difference Equations

When the appropriate finite difference equations are written simultaneously for all grid points on any time line k + 1, including the boundary points, if the variables are unknown on the boundary, a system of simultaneous equations results equal in number to the number of unknowns Q_i^{k+1} and Y_i^{k+1}. Since two unknowns exist at each point the number of equations are twice as many as there are grid points, and these equations come from the two equations, the continuity and the equation of motion (or a special equation from a boundary condition), applied at each point. A solution of this system advances the solution to the problem through one time step. After obtaining that solution, the k + 1 time line becomes the k-th time line and the process is repeated. The initial condition Q(x,0) and Y(x,0) is used to start the solution for the first time line above the x-axis as with the method that uses the characteristics. This initial condition is generally the solution to a steady state GVF problem, and is obtained using the methods described in Chapter 4 for solving the ODE that governs these problems.

Using the difference scheme described above and matrix notation the system of equations for any time line can be written as

$$[\mathbf{A}]\{\mathbf{z}\} = \{\mathbf{b}\}$$

in which $\{\mathbf{z}\}$ is the vector of unknowns Y_i^{k+1} and Q_i^{k+1}, $\{\mathbf{b}\}$ is the vector of knowns on the right of the equal signs of Equations 7.69 through 7.75, and $[\mathbf{A}]$ is the matrix of the coefficients of Y^{k+1} and Q^{k+1} on the left of the equal sign in these equations.

To illustrate the form of this system of equations consider the problem for which the flow rate $Q(L, t) = Q_b(t)$ is specified at the downstream end of a channel and its upstream end is supplied by a constant head reservoir. The system of equations that results when the appropriate finite difference equations are applied across an entire time line is shown in Figure 7.5. The order of writing these is to give the energy Equation 7.70a first and the continuity Equation 7.69a second as pairs. The unknowns are give with Y first and Q second. The nonzero elements in this coefficient matrix are concentrated near the diagonal. If the first and last equation that come from boundary conditions are ignored the pattern of these non zero coefficients is as follows: Since there are six unknown (three pairs of Y's and Q's from the three consecutive grid points, i − 1, i and i + 1) each line contains a maximum of six nonzero elements. In the case of the continuity Equation 7.69, however, three of these are zero. Using the above order of writing the equations and unknowns this results in two non zero elements before the diagonal element, and three after it for lines from Equation 7.70. Accounting for the zero elements from Equation 7.69 there are also two nonzero elements before the diagonal and only one after it for the lines coming from this equation. It should be noted that the diagonal element on lines coming from Equation 7.69 are zero. The equation number is defined by the row of the matrix. While Figure 7.5 uses the same symbols repeatedly in separate pairs of lines the values of these elements are different, in general, since they come from different positions, i along the channel. Rows and columns that are left bank in the coefficient matrix in Figure 7.5 have values equal to zero.

Exceptions to the above described pattern of nonzero elements are the first and final rows, which represent equations from the boundary conditions. The first row contains only two non zero elements, and is Equation 7.74. The last three rows are different because of the downstream boundary

$$
\begin{bmatrix}
1 & -c_1 & 0 & & & & & & & & & & \\
c_2 & -0.5 & 0 & 0.5 & & & & & & & & & \\
-d_1 & -d_2 & d_3 & d_4 & d_1 & d_2 & & & & & & & \\
0 & -0.5 & e_1 & 0 & 0 & 0.5 & & & & & & & \\
 & & -d_1 & -d_2 & d_3 & d_4 & d_1 & d_2 & & & & & \\
 & & 0 & -0.5 & e_1 & 0 & 0 & 0.5 & & & & & \\
 & & & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & & \\
 & & & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & & \\
 & & & & -d_1 & -d_2 & d_3 & d_4 & d_1 & d_2 & & & \\
 & & & & 0 & -0.5 & e_1 & 0 & 0 & 0.5 & & & \\
 & & & & & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \\
 & & & & & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \\
 & & & & & & & -d_1 & -d_2 & d_3 & d_4 & d_1 & d_2 \\
 & & & & & & & 0 & -0.5 & e_1 & 0 & 0 & 0.5 \\
 & & & & & & & & & -d_1 & -d_2 & d_3 & d_4 \; d_1 \\
 & & & & & & & & & 0 & -0.5 & e_1 & 0 \; 0 \\
 & & & & & & & & & & 0.5 & 0 & -2 \; e_1
\end{bmatrix}
\begin{Bmatrix}
Y_1 \\ Q_1 \\ Y_2 \\ Q_2 \\ Y_3 \\ Q_3 \\ \\ \\ Y_i \\ Q_i \\ \\ \\ Y_{n-2} \\ Q_{n-2} \\ Y_{i-1} \\ Q_{i-1} \\ Y_n
\end{Bmatrix}
=
\begin{Bmatrix}
b_{Y1} \\ b_{Q1} \\ b_{Y2} \\ b_{Q2} \\ b_{Y3} \\ b_{Q3} \\ \\ \\ b_{Yi} \\ b_{Qi} \\ \\ \\ b_{Yn-2} \\ b_{Qn-2} \\ b_{Yn-1} \\ b_{Qn-1} \\ b_{Yn}
\end{Bmatrix}
$$

FIGURE 7.5 Linear system of finite difference equations represented in matrix notation for constant upstream reservoir supplying the channel and the flow rate given as a function of time at the downstream end of the channel.

condition. The second and third from the last row contain only five elements, rather than the regular six. This is due to Q_n being specified. Thus the sixth elements from these two row have been transferred to the known vector side of the equation and are included in b_{Yn-1} and b_{Qn-1}. The last row in the coefficient matrix comes from Equation 71. It contains a coefficient 0.5 in the position 3 elements before the diagonal.

An efficient method for solving this special system of equations utilizes elimination methods specifically designed for the nature of the nonzero elements. This process starts by using a single Gaussian elimination operation on the last row to make the leading element (0.5) in this row zero by utilizing the second from the last row. After this operation the solution to the system is readily accomplished by two passes through each row to eliminate first the second and then the first elements before the diagonal with a couple of special eliminations in between these two passes, followed by back substitution in obtaining the values of the unknowns, Y_i and Q_i. This exact process of solution is defined in the FORTRAN listing given subsequently. Also storage requirements for matrix A are only $(2n - 1) \times 6$ instead of $(2n - 1) \times (2n - 1)$ if standard linear algebra subroutines were to be used.

7.8.1.3 Computer Code to Implement Direct Implicit Method

Below is the listing of a FORTRAN program that implements the method described above to obtain the solution to unsteady flows in channels with a constant head reservoir at their upstream ends, and with the flow rate specified as a function of time at the downstream end of the channel. The input to this program consists of the following:

First line:

IOUT = the logical FORTRAN unit that the output will be written to,

NFORW = an indicator of the order of initial condition depths; if this order is downstream first and upstream last then give NFORW=1, otherwise NFORW=0,

NX = number of stations along x-axis where data will be supplied, and solution variables will be written,

NT = number of time step through which the solution is to be obtained,

NTBETW = the number of computation increments between each of the increments specified by NT; to achieve greater accuracy,

DELT = the time step in seconds between printed solutions, i.e. the total time of simulation will equal NT × DELT,

DELX = the increment Δx along the x-axis,

G = acceleration of gravity, (The value determines whether ES or SI units will be used.)

Second line: The flow rates at the NX input stations. The difference between these values will be assumed to be lateral inflow/outflow uniformly distributed between the two stations.

Third line: The bottom widths b at the NX input stations.

Fourth line: The side slopes m at the NX input stations.

Fifth line: The value of Manning's roughness coefficient at the NX input stations.

Sixth line: The slope of the channel bottom at the NX input stations.

Seventh line: The initial depths at the NX input stations. If NFORW equals 1 these are listed in order of the downstream depth first and the upstream depth last; otherwise these depths are listed in the same order as the other data from the upstream end to the downstream end.

Eighth line: Values for the downstream flow rate for each of the NT time steps to be used for the solution including the initial flow rate at the downstream station. In other words NT + 1 values can be read.

With the exception of the seventh line that contains the initial depths any of the above lines can be terminated with a /. Furthermore as many lines as needed can be used for the input required under any of the lines identified by first, second, etc. If the list is terminated with a /, then the remaining values will be taken equal to the last value. The exception is the eighth line in which the remaining values will be taken so as to have the same increment between them as the last two given values did. Thus if an input does not change across the channel only one value needs to be given for the line of values followed by a /.

The program is divided into three components: The main program that reads in the problem specifications, etc. as described above; the subroutine IMPLIC that defines the linear algebra problem by generating the coefficient matrix, and the known vector for the linear system of equations, and prints out the solution when it has been obtain from these equations; the subroutine BAND which implements a special Gaussian elimination to solve the linear system of equations.

The program allows for only one type of boundary condition at the downstream end, and one at the upstream end of the channel. At the downstream end the flow rate must be given as a function of time by reading in the values of Q(L, t) for each time step involved in the solution. The upstream boundary assumes that a constant head reservoir supplies the channel. The head H of the water surface in this reservoir is computed based on the initial depth and flow rate given for the upstream end of the channel.

Program IMPLICIT.FOR

```
      COMMON /SOLV/ A(81,6),V(81)
      COMMON Y(41),AA(41),Q(41),PP(41),TOP(41),V1(41),X(41),SQ(41),
     &B(41),FM(41),SO(41),FN(41),SMS(41),SF(41),QGIV(0:100),DXG,
     &G,G2,DELX,DEL2,DELT,DXT,RDT,H,CMAN,NT,NX,NXM,IOUT,NTBETW
      DATA ONE/-1./
      READ(2,*) IOUT,NFORW,NX,NT,NTBETW,DELT,DELX,G
      DELT=.1*DELT
      G2=2.*G
      CMAN=1.
      IF(G.GT.30.) CMAN=1.486
```

```
      DEL2=2.*DELX
      DXT=DELX/DELT
      DXG=G*DELX
      RDT=1./DELT
      NXM=NX-1
      DO 10 I=1,NX
      Q(I)=ONE
      B(I)=ONE
      FM(I)=ONE
      FN(I)=ONE
10    SO(I)=ONE
      READ(2,*) (Q(I),I=1,NX)
      READ(2,*) (B(I),I=1,NX)
      READ(2,*) (FM(I),I=1,NX)
      READ(2,*) (FN(I),I=1,NX)
      READ(2,*) (SO(I),I=1,NX)
      DO 20 I=2,NX
      IM=I-1
      IF(Q(I).EQ.ONE) Q(I)=Q(IM)
      IF(B(I).EQ.ONE) B(I)=B(IM)
      IF(FM(I).EQ.ONE) FM(I)=FM(IM)
      IF(FN(I).EQ.ONE) FN(I)=FN(IM)
      IF(SO(I).EQ.ONE) SO(I)=SO(IM)
      SQ(I)=Q(I)-Q(IM)
20    SMS(I)=1.33333333*SQRT(FM(I)**2+1.)
      SMS(1)=1.33333333*SQRT(FM(1)**2+1.)
      IF(NFORW.EQ.1) THEN
      READ(2,*)(Y(I),I=1,NX)
      ELSE
      READ(2,*)(Y(I),I=NX,1,-1)
      ENDIF
      DO 30 I=1,NT
30    QGIV(I)=ONE
      READ(2,*)(QGIV(I),I=0,NT)
      IF(ABS(Q(NX)-QGIV(0)) .GT. .1) THEN
      WRITE(*,100) Q(NX),QGIV(0)
100   FORMAT(' Flowrate',F8.2,' at end must be same as 1st Q',
     &' at t=0',F8.2)
      STOP
      ENDIF
      DO 40 I=3,NT
      IF(QGIV(I).EQ.ONE) QGIV(I)=2.*QGIV(I-1)-QGIV(I-2)
40    CONTINUE
      H=Y(1)+(Q(1)/((B(1)+FM(1)*Y(1)) *Y(1)))**2/G2
      CALL IMPLIC
      END
      SUBROUTINE IMPLIC
      COMMON /SOLV/ A(81,6),V(81)
      COMMON Y(41),AA(41),Q(41),PP(41),TOP(41),V1(41),X(41),SQ(41),
     &B(41),FM(41),SO(41),FN(41),SMS(41),SF(41),QGIV(0:100),DXG,
     &G,G2,DELX,DEL2,DELT,DXT,RDT,H,CMAN,NT,NX,NXM,IOUT,NTBETW
```

```
      TIME=0.
      K=0
110   K=K+1
      QGIV1=QGIV(K-1)
      DQGIV=(QGIV(K)-QGIV1)/FLOAT(NTBETW)
      DO 290 K10=1,NTBETW
      QGIVI=QGIV1+DQGIV
      DO 120 I=1,NX
      AA(I)=(B(I)+FM(I)*Y(I)) *Y(I)
      V1(I)=Q(I)/AA(I)
      PP(I)=B(I)+1.5*SMS(I)*Y(I)
      TOP(I)=B(I)+2.*FM(I)*Y(I)
120   SF(I)=(FN(I)*Q(I)/(CMAN*AA(I)))**2*(PP(I)/AA(I))
     &**1.33333333
      IF(K10.NE.1) GO TO 145
      WRITE(IOUT,130) TIME
      IF(IOUT.NE.6) WRITE(*,130) TIME
130   FORMAT(' Solution for time =',F10.2,/,1X,60('-'),/,
     &' DEPTH ',, VELOCITY EL_SLOPE PERIM. FLOWRATE',
     &'AREA TOP_W.',/,1X,60('-'))
      DO 135 I=1,NX
      IF(IOUT.NE.6) WRITE(*,140) I,Y(I),V1(I),SF(I),PP(I),Q(I),
     &AA(I),TOP(I)
135   WRITE(IOUT,140) I,Y(I),V1(I),SF(I),PP(I),Q(I),AA(I),TOP(I)
140   FORMAT(I3,2F8.3,F8.5,4F8.3)
145   IF(K.GT.NT) STOP
      TIME=TIME+DELT
      I=1
C  Setting up equations
270   DO 280 II=2,NXM
      I=I+2
      IP=I+1
      C2=G*AA(II)/TOP(II)
      FYP=TOP(II)*(C2-V1(II)**2)/DEL2
      FQP=V1(II)/DELX
      A(I,1)=-FYP
      A(I,2)=-FQP
      A(IP,1)=0.
      A(IP,2)=-.5
      SC64=G2*SF(II)
      FY=SC64*(SMS(II)*AA(II)/PP(II)-1.6666667*TOP(II))
      A(I,3)=FY
      A(I,4)=RDT+SC64*AA(II)/Q(II)
      A(I,5)=FYP
      A(I,6)=FQP
      A(IP,3)=TOP(II)*DXT
      A(IP,4)=0.
      A(IP,5)=0.
      A(IP,6)=.5
      DL=V1(II)*SQ(II)/(DXG*AA(II))
      IF(SQ(II) .LT. 0.) DL=.5*DL
```

```
      SOD=SO(II)-DL
      V(I)=Q(II)/DELT+Y(II)*(FY+V1(II)**2*(B(II+1)-B(II-1)+
     &Y(II)*(FM(II+1)-FM(II-1)))/DEL2)+G*(AA(II)*
     &(SOD+SF(II)))+V1(II)*SQ(II)/DELX
280   V(IP)=A(IP,3)*Y(II)+SQ(II)
C  Upstream Boundary Condition
      C1=Q(1)/G/(AA(1)*(2.*(H-Y(1)) *TOP(1)-AA(1)))
      V(1)=Y(1)-C1*Q(1)
      A(1,4)=1.
      A(1,5)=-C1
      A(1,6)=0.
      A(2,3)=DELX*RDT*TOP(1)
      A(2,4)=-.5
      A(2,5)=0.
      A(2,6)=.5
      V(2)=A(2,3)*Y(1)+.5*(Q(1)-Q(2))
C  Downstream Boundary Condition, Q(t) = known
      IP=IP+1
      V(IP)=DELX*SQ(NX)-1.5*QGIV1+DELX*RDT*TOP(NX)*Y(NX)
      V(IP-1)=V(IP-1)-A(IP-1,6)*QGIVI
      V(IP-2)=V(IP-2)-A(IP-2,6)*QGIVI
      A(IP,2)=-2.
      A(IP,3)=DELX*RDT*TOP(NX)
      Q(NX)=QGIVI
C  Solving Equations
      CALL BAND(IP)
290   QGIV1=QGIVI
      GO TO 110
      END
      SUBROUTINE BAND(N)
      COMMON /SOLV/ A(81,6),V(81)
      COMMON Y(41),AA(41),Q(41),PP(41),TOP(41),V1(41),X(41),SQ(41),
     &B(41),FM(41),SO(41),FN(41),SMS(41),SF(41),QGIV(0:100),DXG,
     &G,G2,DELX,DEL2,DELT,DXT,RDT,H,CMAN,NT,NX,NXM,IOUT,NTBETW
      FAC=.5/A(N-1,2)
      A(N,1)=-FAC*A(N-1,3)
      A(N,3)=A(N,3)-FAC*A(N-1,5)
      V(N)=V(N)-FAC*V(N-1)
      FAC=A(3,1)/A(2,3)
      A(3,2)=A(3,2)-FAC*A(2,4)
      A(3,4)=A(3,4)-FAC*A(2,6)
      V(3)=V(3)-FAC*V(2)
      DO 40 K=0,1
      DO 30 I=4,N
      IM=I-1
      IF(MOD(I,2).EQ.0) THEN
      KK=K+2
      KM=KK
      K6=6
      ELSE
```

```
      KK=K+1
      KM=K+3
      K6=4
      ENDIF
      FAC=A(I,KK)/A(IM,KM)
      V(I)=V(I)-FAC*V(IM)
      DO 30 J=KK+1,K6
30    A(I,J)=A(I,J)-FAC*A(IM,KM-KK+J)
      IF(K.EQ.1) GO TO 40
      FAC=A(2,3)/A(1,4)
      A(2,4)=A(2,4)-FAC*A(1,5)
      V(2)=V(2)-FAC*V(1)
      FAC=A(3,2)/A(2,4)
      A(3,4)=A(3,4)-FAC*A(2,6)
      V(3)=V(3)-FAC*V(2)
40    CONTINUE
      I=NXM
      IN=N-1
      Y(NX)=V(N)/A(N,3)
      Q(I)=(V(IN)-Y(NX)*A(IN,5)) /A(IN,4)
      IN=IN-1
      Y(I)=(V(IN)-Q(I)*A(IN,4)-A(IN,5)*Y(NX)) /A(IN,3
50    IP=I
      I=I-1
      IN=IN-1
      Q(I)=(V(IN)-Y(IP)*A(IN,5)-Q(IP)*A(IN,6))/A(IN,4)
      IN=IN-1
      Y(I)=(V(IN)-Q(I)*A(IN,4)-Y(IP)*A(IN,5)-Q(IP)*A(IN,6)) /
     &A(IN,3)
      IF(I.GT.2) GO TO 50
      Q(1)=(V(2)-A(2,6)*Q(2)) /A(2,4)
      Y(1)=(V(1)-A(1,5)*Q(1)) /A(1,4)
      RETURN
      END
```

EXAMPLE PROBLEM 7.4

Flow enters a trapezoidal channel with b = 5 m, m = 1.3, n = 0.0135, and S_o = 0.0012. At the downstream end of this channel, which is 1000 m long, there is a sluice gate in a rectangular section that is 4.5 m wide. The transition to the rectangular section has a minor loss coefficient K_L = 0.10. The gate has a contraction coefficient C_c = 0.6. For a long time the gate has been at a height of 0.8 m above the channel bottom. At time t = 0 s, the gate is slowly raised so that the flow rate past it varies as given in the table below.

Time (s)	0	20	40	60	80	100	120	140	160
Flow (m^3/s)	17.05	18.0	20.0	22.0	24.0	26.0	28.0	30.0	30.0

Solve the unsteady problem for some time after the flow rate is maintained constant at the downstream end of the channel at 30 m^3/s.

Solution

The steady state problem must be solved as the initial condition to the unsteady problem. This solution is given as part of the input data. The input data to the above listed FORTRAN program consists of

```
3 0 21 20 10 20 50 9.81
17.05/
5./
1.3/
.0135/
.0012/
3.649 3.589 3.529 3.470 3.410 3.350 3.290 3.231 3.171 3.112
3.052 2.993 2.933 2.874 2.814 2.755 2.696 2.637 2.578 2.519 2.460
17.05 18 20 22 24 26 28 30 30/
```

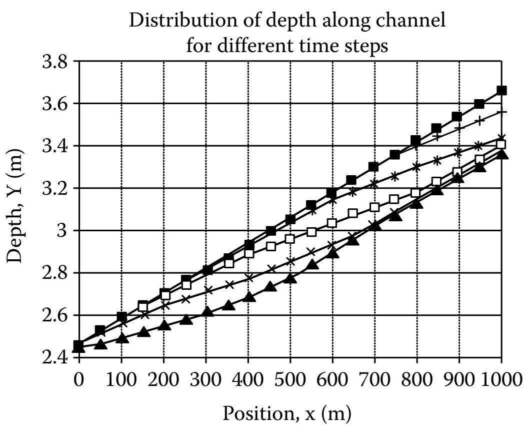
The tables that were obtained from the above input data for only two selected time increments are given below, and three figures are given that plot: (a) the depth, (b) the velocity, and (c) the flow rate for time steps 0 through 300 s. Note from these figure how the effect from increasing the flow rate at the downstream end of the channel propagates upstream through the channel with time. The effect reaches the upstream end of the channel between times 180 and 240 s. Thereafter the flow rate into the channel increases, and an examination of the solution table for 400 s shows that at this time the flow rate $Q_1 = 31.34\,m^3/s$ into the channel exceeds the $Q_n = 30\,m^3/s$ that leaves the end of the channel by a small amount. This added inflow is beginning to make ups the losses of channel storage that occurred during earlier times of the unsteady flow.

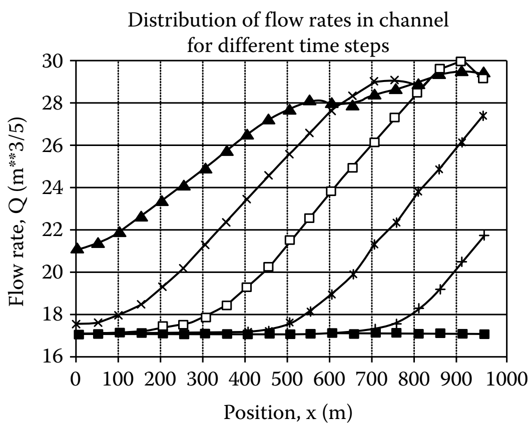
Solution for time = 80.00

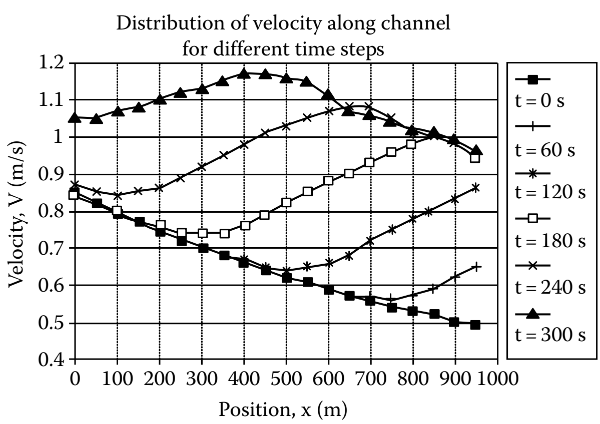
	DEPTH	VEL.	EL_SLOPE	PERIM.	FLOWR.	AREA	TOP_W.
1	2.460	0.846	0.00007	13.069	17.070	20.166	11.396
2	2.519	0.819	0.00007	13.264	17.070	20.846	11.550
3	2.578	0.792	0.00006	13.455	17.057	21.526	11.702
4	2.637	0.767	0.00006	13.650	17.057	22.226	11.856
5	2.696	0.743	0.00005	13.843	17.040	22.928	12.009
6	2.755	0.721	0.00005	14.038	17.052	23.646	12.164
7	2.814	0.700	0.00004	14.231	17.045	24.366	12.317
8	2.874	0.679	0.00004	14.427	17.055	25.105	12.472
9	2.933	0.660	0.00004	14.621	17.060	25.850	12.626
10	2.992	0.641	0.00003	14.816	17.063	26.603	12.780
11	3.052	0.624	0.00003	15.011	17.082	27.369	12.935
12	3.111	0.607	0.00003	15.204	17.068	28.133	13.088
13	3.170	0.595	0.00003	15.397	17.199	28.908	13.241
14	3.225	0.582	0.00003	15.580	17.253	29.650	13.386
15	3.281	0.582	0.00003	15.761	17.699	30.394	13.530
16	3.329	0.584	0.00003	15.920	18.124	31.053	13.656
17	3.375	0.602	0.00003	16.072	19.087	31.686	13.776
18	3.415	0.624	0.00003	16.202	20.124	32.235	13.879
19	3.455	0.656	0.00003	16.332	21.501	32.789	13.982
20	3.492	0.676	0.00003	16.454	22.525	33.311	14.079
21	3.534	0.708	0.00004	16.593	24.000	33.909	14.189

Solution for time = 400.00

	DEPTH	VEL.	EL_SLOPE	PERIM.	FLOWR.	AREA	TOP_W.
1	2.378	1.527	0.00025	12.800	29.381	19.241	11.183
2	2.402	1.526	0.00024	12.879	29.780	19.509	11.245
3	2.430	1.517	0.00024	12.970	30.071	19.824	11.317
4	2.460	1.505	0.00023	13.068	30.343	20.162	11.395
5	2.494	1.485	0.00022	13.180	30.511	20.552	11.484
6	2.532	1.452	0.00021	13.307	30.500	20.999	11.584
7	2.578	1.411	0.00019	13.455	30.376	21.524	11.702
8	2.627	1.360	0.00018	13.617	30.058	22.104	11.830
9	2.679	1.306	0.00016	13.787	29.666	22.722	11.965
10	2.729	1.262	0.00015	13.952	29.430	23.326	12.095
11	2.778	1.221	0.00013	14.114	29.217	23.927	12.224
12	2.825	1.191	0.00013	14.267	29.186	24.502	12.345
13	2.877	1.156	0.00012	14.437	29.077	25.146	12.480
14	2.929	1.126	0.00011	14.606	29.030	25.792	12.614
15	2.983	1.094	0.00010	14.784	28.973	26.479	12.755
16	3.035	1.071	0.00009	14.956	29.083	27.150	12.891
17	3.089	1.049	0.00009	15.133	29.217	27.851	13.032
18	3.144	1.029	0.00008	15.312	29.389	28.568	13.174
19	3.199	1.008	0.00008	15.495	29.536	29.305	13.319
20	3.256	0.991	0.00008	15.679	29.785	30.057	13.465
21	3.313	0.973	0.00007	15.868	30.000	30.837	13.614







7.8.2 Alternative Approach to Boundary Condition Equations

In implementing a solution using the **direct implicit finite differencing** method the continuity equation was used to obtain another finite difference equation beyond that given by the known boundary condition equation. An alternative to this is to use the appropriate characteristic form of the St. Venant equations. For the upstream boundary the equation that applies along the C^- negative characteristics is used, and for the downstream boundary condition the equation that applies along positive C^+ characteristics is used.

Since the Q-Y St. Venant equations are used in this section this form of the characteristic equations is needed. These equations can be obtained by adding and subtracting Equations 7.61 and 7.62 appropriately, or these equations can be obtained from the V-Y form of the characteristic equations, Equation 7.3. Using the latter approach we want to replace V by Q in Equation 7.3. Taking partial derivatives of the continuity equation Q = VA with respect to t and x give

$$\frac{\partial Q}{\partial t} = V\frac{\partial A}{\partial t} + A\frac{\partial V}{\partial t} \quad \text{and} \quad \frac{\partial Q}{\partial x} = V\frac{\partial A}{\partial x} + A\frac{\partial V}{\partial x}$$

and solving for the derivatives of the velocity gives

$$\frac{\partial V}{\partial t} = \frac{1}{A}\left(\frac{\partial Q}{\partial t} - VT\frac{\partial Y}{\partial t}\right) \quad \text{and} \quad \frac{\partial V}{\partial x} = \frac{1}{A}\left(\frac{\partial Q}{\partial x} - VT\frac{\partial Y}{\partial x}\right)$$

Substituting these into Equation 7.3 and then multiplying by A and rearranging terms gives

$$\frac{\partial Q}{\partial t} + (V \pm c)\frac{\partial Q}{\partial x} + T(-V \pm c)\left\{\frac{\partial Y}{\partial t} + (V \pm c)\frac{\partial Y}{\partial x}\right\} = gA(S_0 - S_f) = Tc^2 S_0 - \frac{gAQ|Q|}{K^2} \qquad (7.76a)$$

At the downstream end of the channel the boundary condition may consist of the following: (1) the depth specified as a function of time, (2) the flow rate specified as a function of time, (3) a known stage-discharge relationship, i.e., a relationship between Q and Y, or (4) the velocity specified as a function of time. These boundary conditions and their finite difference expressions are:

$Y(L,t) = \text{known} \quad Y_n^{k+1} = \text{known}$

$Q(L,t) = \text{known} \quad Q_n^{k+1} = \text{known}$

$Y(L,t) = f\{Q(L,t)\} \quad Y_n^{k+1} = f(Q_n^{k+1})$

$V(L,t) = \text{known} \quad (Q/A)_n^{k+1} = V_n^{k+1} = \text{known}$

The second needed equation comes from Equation 7.76a with the + sign (for along the C^+ characteristics) by replacing the derivatives by appropriate differences. One such equation that is consistent with the direct implicit finite differencing method that produces a linear system of equation for Y and Q at the advanced time step is

$$\frac{Q_n^{k+1} - Q_n^k}{\Delta t} + (v + c)_n^k \frac{Q_n^{k+1} - Q_{n-1}^{k+1}}{\Delta x} + T_n^k(c - V)_n^k\left[\frac{Y_n^{k+1} - Y_n^k}{\Delta t} + (V + c)_n^k \frac{Y_n^{k+1} - Y_{n-1}^k}{\Delta x}\right]$$

$$= gA\left(S_o - (S_f)_n^{k+1}\right) = (Tc^2)_n^k S_o - gA_n^k (S_f)_n^{k+1} \qquad (7.76b)$$

in which $(S_f)_n^{k+1}$ is evaluated using Equation 7.65. If the flow rate Q(L, t) is given then this equation is written with the Y's with the superscript k + 1 on the left of the equal sign and the knowns on the right of the equal sign. This boundary condition equation is

$$b_1 Y_{n-1}^{k+1} + b_2 Q_{n-1}^{k+1} + b_3 Y_n^{k+1} = f_5 - b_4 Q_n^{k+1} \tag{7.77}$$

in which

$$b_1 = \frac{T_n^k (V-c)_n^k (V+c)_n^k}{\Delta x} = \frac{T_n^k (V^2 - c^2)_n^k}{\Delta x}, \quad b_2 = \frac{-(V+c)_n^k}{\Delta x},$$

$$b_3 = -T_n^k (V-c)_n^k \left[\frac{1}{\Delta t} + \frac{(V+c)_n^k}{\Delta x} \right] + 2g \left[S_t \left\{ \frac{5T}{3} - \frac{2A}{3P} \frac{\partial P}{\partial Y} \right\} \right]_n^k,$$

$$b_4 = \frac{1}{\Delta t} + \frac{(V+c)_n^k}{\Delta x} - 2g \left(\frac{AS_f}{Q} \right)_n^k,$$

and

$$b_5 = S_o (Tc^2)_n^k + g(AS_t)_n^k - 2g \left[S_t \left\{ \frac{5T}{3} - \frac{2A}{3P} \frac{\partial P}{\partial Y} \right\} Y \right]_n^k + \frac{Q_n^k}{\Delta t} + \frac{[T(c-V)Y]_n^k}{\Delta t}$$

If the depth is specified at the downstream boundary then Equation 7.77 is written as

$$b_1 Y_{n-1}^{k+1} + b_2 Q_{n-1}^{k+1} + b_4 Q_n^{k+1} = f_5 - b_3 Y_n^{k+1}, \quad \text{and} \tag{7.77a}$$

For the case in which the stage–discharge relationship is given two equations exist, namely, the stage-discharge relationship written as a linear equation between the given values of Y that bracket the current depth

$$Q_n^{k+1} = Qs_j + \frac{Qs_{j+1} - Qs_j}{Ys_{j+1} - Ys_j} \left(Y_n^{k+1} - Ys_j \right) = Qs_j + \frac{dQ}{dY_s} \left(Y_n^{k+1} - Ys_j \right)$$

in which the Ys and Qs are the depth and flow rate values from the given stage discharge relationship, and the subscript j represents the value just smaller than Y_n^{k+1} and j + 1 just larger than or equal to Y_n^{k+1}. Likewise if H(t), the reservoir water surface depth above the channel bottom, is given, then two equations provide the boundary condition that allow both Y_n^{k+1} and Q_n^{k+1} to be solved. For the velocity V_n^{k+1} given the equation from the characteristics can be written as

$$Y_n^{k+1} = \frac{f_5 - b_1 Y_{n-1}^{k+1} - b_2 Q_{n-1}^{k+1}}{b_3 + b_4 V_n^{k+1} \left(b_n + m_n Y_n^{k+1} \right)}$$

At the upstream boundary any of the conditions given at the downstream end might exist and in addition if the channel is supplied by a reservoir (whose water surface elevation may vary with time), we have the energy equation at the entrance of the channel available as a fourth possibility. Thus any of the following four be known:

$Y(0,t) = \text{known}\ Y_1^{k+1} = \text{known}$
$Q(0,t) = \text{known}\ Q_1^{k+1} = \text{known}$
$Y(0,t) = f\{Q(0,t)\}\ Y_1^{k+1} = f\{Q_1^{k+1})$
$Y_1 + (1+K_e)Q_1^2/(2gA_1^2) = H(t) = \text{known}$ (with the differences given by Equation 7.74)

The second needed equation comes from appropriately differencing Equation 7.76a along C⁻ negative characteristics, i.e., using the negative sign in this equation to give

$$\frac{Q_1^{k+1}-Q_1^k}{\Delta t}+(V-c)_1^k\frac{Q_2^{k+1}-Q_1^{k+1}}{\Delta x}-T_1^k(V+c)_1^k\left[\frac{Y_1^{k+1}-Y_1^k}{\Delta t}+(V-c)_1^k\frac{Y_2^{k+1}-Y_1^{k+1}}{\Delta x}\right]$$
$$= gA_1^k\left(S_0-(S_f)_1^k\right) \tag{7.78}$$

After writing Equation 7.78 as a linear equation with Y and Q with a superscript k + 1 on the left of the equal sign, and the knowns with superscript k on the right of the equal sign it becomes

$$a_1Y_1^{k+1}+a_2Q_1^{k+1}+a_3Y_2^{k+1}+a_4Q_1^{k+1}=f_4 \tag{7.79}$$

in which

$$a_1=-T_1^k(V+c)_1^k\left[\frac{1}{\Delta t}-\frac{(V-c)_1^k}{\Delta x}\right]-2g\left[S_f\left\{\frac{5}{3}T-\frac{2}{3}\frac{A}{P}\frac{\partial P}{\partial Y}\right\}\right]_1^k$$

$$a_2=\frac{1}{\Delta t}-\frac{(V-c)_1^k}{\Delta x}+2g\left(\frac{AS_f}{Q}\right)_1^k$$

$$a_3=-T_1^k(V+c)_1^k\frac{(V-c)_1^k}{\Delta x}=-T_1^k\frac{(V^2-c^2)_1^k}{\Delta x};\quad a_4=\frac{(V-c)_1^k}{\Delta x}$$

$$f_4=S_0(Tc^2)_1^k+g(AS_f)_1^k-2g\left[S_f\left\{\frac{5}{3}T-\frac{2}{3}\frac{A}{P}\frac{\partial P}{\partial Y}\right\}Y\right]_1^k+\frac{Q_1^k}{\Delta t}-\left[T(V+c)Y\right]_1^k$$

Depending upon the boundary condition the known term in Equation 7.79 is transferred to the right side of the equal sign and Y_1^{k+1} or Q_1^{k+1} is solved depending, respectively whether Q_1^{k+1} or Y_1^{k+1} is known. If the boundary condition is a given stage-discharge relationship, i.e., pair of Qs's that correspond to Ys's, or H(t) is given, then two equations exist, Equation 7.79 and the given boundary condition equation, or the stage–discharge relationship.

7.9 GAUSS–SEIDEL OR SUCCESSIVE-OVER-RELAXATION (SOR) ITERATIVE SOLUTION TECHNIQUES

An alternative to using linear algebra to solve the system of equations at the (k + 1)th time step is to solve these equations with an iterative techniques such as the Gauss–Seidel, or the successive over-relaxation (SOR) method. These iterative methods are often used to solve the finite difference equations that result from solving elliptic partial differential equations boundary value problems such as those described by Laplace's equation. Such iterative methods solve one equation at a time and in doing this select one of the terms in that equation to place on the left of the equal sign with all other terms on the right of the equal sign. Basically in doing this the assumption is made that all variables are temporarily known except the one that is being solve for by that equation, but since only an estimate of their values are used on the right of the equal sign, these solutions must be repeated iteratively, until all values are correct to the desired accuracy. For such iterative methods to converge it is necessary that the term place on the left of the equal sign be dominant in the equation, i.e., that its coefficient have a magnitude equal to or larger than the sum of absolute values of all the coefficients of the variables whose values are temporarily assumed to be known. The rate of convergence is better the more dominant the term is that is being solved. Arranging the variables that are being solved in this manner is referred to as "diagonal dominance." See a book dealing with numerical methods for greater detail.

To decide which term in Equations 7.69a and 7.70a to place on the left of the equal sign typical values, as following, will be used to compute the magnitude of coefficients: $\Delta x = 50$ ft, $\Delta t = 0.2$ s, $Y = 10$ ft, $A = 100$ ft^2, $T = 15$ ft, $P = 15$ ft, $c = 12$ fps, $S_f = 0.001$, $Q = 300$ cfs and $dP/dY = 4$. Then $e_1 = 150 > 0.5 + 0.5$ (coef. of other terms) in Equation 7.69a so that $e_1 Y_i^{k+1}$ should be placed on the left side of the equal sign with the other terms temporarily assumed known. In Equation 7.70a $d_1 \approx 1.5$, $d_2 \approx 0.1$, $d_3 \approx -0.1$ and $d_4 \approx 4$, i.e., $|d_4| > |d_1| + |d_2| + |d_3| + |d_1| + |d_2|$, so that $d_4 Q_i^{k+1}$ should be used on the left of the equal sign. Similarly for the characteristic boundary condition Equations 7.77 and 7.79, $|b_3| > |b_1| + |b_2| + |b_4|$ and $|a_1| > |a_2| + |a_3| + |a_4|$ so in these equations the terms with coefficients b_3 and a_1 should be used on the left of the equal sign. Note that the magnitudes of d_4 and e_1 are inversely proportional to the size of the time step Δt, and therefore the SOR method will converge more rapidly as the time step is reduced using the above terms on the left of the equal sign, and if the time step is taken too large convergence will not occur. Thus in using the SOR method computational effort may actually be reduced by using smaller time steps. Using smaller time steps also improves the accuracy of the solution.

If the upstream boundary condition is a known reservoir elevation ($H(t)$ = known) then it will be necessary to solve for Q_1^{k+1} from Equation 7.74 if it is used. However, the magnitude of c_1 can becomes smaller than the sum of coef. of other terms, and therefore the Gauss–Seidel method would diverge when this occurs. It might be possible to solve for the flow rate Q_1^{k+1} from $Q = A\{2g(H - Y)\}^{1/2}$ under certain conditions but in general the cases that involve two simultaneous boundary condition equations (i.e., reservoir head known, and stage–discharge relationship known) are best solved using the Newton method in combination with the SOR method. After all the energy equation is nonlinear and using techniques designed to solve linear equations are successful only under special conditions. In apply the Newton method for these boundary equations the two equations will be solved based on assumed known values for Y_{n-1}^{k+1}, Q_{n-1}^{k+1} and Y_2^{k+1}, Q_2^{k+1}, and the Newton method will be repeatedly used to solve the boundary condition equations as part of each iteration through the rest of the interior node equations, Equations 7.69a and 7.70a.

Let us examine how the Newton method can be combined with the SOR method in solving the system of equations with Y_i^{k+1} and Q_i^{k+1} for $i = 1, 2, \ldots, n$ as the unknowns resulting if a stage–discharge relationship is known at the upstream boundary. Then the first two equations of the system are as follows:

$$F_1 = a_1 Y_1^{k+1} + a_2 Q_1^{k-1} + a_3 Q_2^{k+1} - f_4 = 0$$

and

$$F_2 = Qs_j + \left(\frac{dQs}{dYs}\right)_j \left(Y_1^{k-1} - Y_{s_j}\right) - Q_1^{k+1} = 0$$

in which $(dQs/dYs)_j = \{Qs_{j+1} - Qs_j\}/\{Ys_{j+1} - Ys_j\}$.

Thus the system of equations that needs to be solved for each Newton iterations is

$$\begin{bmatrix} a_1 & a_2 \\ \dfrac{dQs}{dYs} & -1 \end{bmatrix} \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix}$$

Note that this system of equations is readily solved by Gaussian elimination.

After solving for the {z} vector, the Newton method updates the first two unknowns with $\left(Y_1^{k+1}\right)^{m+1} = \left(Y_1^{k+1}\right)^m - z_1$ and $\left(Q_1^{k+1}\right)^{m+1} = \left(Q_1^{k+1}\right)^m - z_2$. Since the rate of convergence of the Newton method will generally exceed the rate at which the SOR method converges it is computational more efficient to use only one Newton iteration for each pass through the total system of equations being solved.

The condition for a known reservoir head H(t) is almost identical with F_2 above, or is

$$F_2 = H - Y_1^{k+1} - (1 + k_e)\frac{(Q^2)_1^{k+1}}{2gA_1^2} = 0$$

and the system of equations that needs to be solved to get the correction vector {z} is

$$\begin{bmatrix} a_1 & a_2 \\ \dfrac{Q_1^2 T}{gA_1^3} - 1 & -\dfrac{Q_1}{gA_1^2} \end{bmatrix} \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix}$$

The FORTRAN program IMPLICBC.FOR implements the SOR method for solving the "direct implicit finite difference" Equations 7.69a and 7.70a, along with allowing any of the above described boundary conditions to occur at either the upstream or downstream boundaries. You should study this program listing to understand the detail of implementing the SOR method, as well as how various types of boundary conditions can be accommodated. Much of the input to this program is the same as Program IMPLICIT. On the first line of input the additional variables are

W = over-relaxation factor minus 1, i.e., W = ω − 1.

ERR = residual to terminate SOR iteration. A typical value is ERR = 0.0001.

MAX = the maximum SOR iterations allowed.

IUBC = an integer whose value determines the type of upstream boundary condition to use according to

1. The depth Y_1^{k+1} will be given for each new time step in the solution. These depths are given in the input array QGIU.
2. The flow rate Q_1^{k+1} will be given for each new time step in the solution. These flow rates are given in the input array QGIU.
3. A stage–discharge relationship will be given for the upstream boundary condition, and these values will be given as pairs (Ys and Qs) in the input array QGIU.
4. The reservoir head H(t) is given for each new time step in the solution.

IDBC = an integer whose value determines the type of downstream boundary condition to use according to

1. The depth Y_n^{k+1} will be given for each new time step in the solution. These depths are given in the input array QGID.
2. The flow rate Q_n^{k+1} will be given for each new time step in the solution. These flow rates are given in the input array QGID.
3. A stage–discharge relationship will be given for the downstream boundary condition, and these values will be given as pairs (Ys and Qs) in the input array QGID.
4. The velocity $V_n^{k+1} = (Q/A)_n^{k+1}$ is given for each new time step in the solution. These velocities are given in the input array QGID.

As with the input to IMPLICIT only one value for the array for: (a) the flow rate, Q(I), (b) the bottom width, B(I), (c) the side slope, FM(I), (d) Manning's n, FN(I), and (e) the bottom slope SO(I) are needed if this value is constant throughout the channel, and the value is terminated by /. The arrays QGIU and QGID contain the values that defined the boundary conditions as described above. If IUBC or IDBC = 3 (for a stage–discharge boundary condition) it is necessary that an even number of values be given (pairs of Ys and Qs), and that this list be terminated with a /. If the values read into array QGIU or QGID are depths, flow rates, or velocities, then the list can be terminated with a/if all subsequent values are equal to the last value given in the list, otherwise as many values are required as time steps through which the solution is to be obtained. The READ statements after statement 22 reads in the initial condition of water depths. These values for Y(I) can be read starting with the downstream depth, or the upstream depth depending upon whether NFORW is given a 0 or 1, respectively.

The subroutine SOR in program IMPLICBC.FOR implements the SOR iterative method of solution. The subroutine STAGE carries out both the interpolation in the table of Ys and Qs that define the stage–discharge relationship and then implements the Newton method in solving the two needed boundary condition equations as described above. Note this subroutine thus returns the new iterative values of Y_1^{k+1} and Q_1^{k+1} for the upstream boundary, or Y_n^{k+1} and Q_n^{k+1} for the downstream boundary, i.e., it solves both the upstream and downstream stage–discharge boundary condition equations. If H(t) is given then the Newton method is implemented within subroutine SOR to solve the two upstream boundary condition equations. The input to program IMPLICBC is illustrated by Example Problem 7.5.

EXAMPLE PROBLEM 7.5

Solve Example Problem 7.4 for the combination of boundary conditions given in the table below.

Case	Upstream BC	Downstream BC
1	Y(0, t) = given	Y(L, t) = given
2	Y(0, t) = given	Q(L, t) = given
3	Y(0, t) = given	V(L, t) = given
4	Y(0, t) = given	Stage–Discharge
5	Q(0, t) = given	Y(L, t) = given
6	Q(0, t) = given	Q(L, t) = given
7	Q(0, t) = given	V(L, t) = given
8	Q(0, t) = given	Stage–Discharge
9	H(t) = given	Y(L, t) = given
10	H(t) = given	Q(L, t) = given
11	H(t) = given	V(L, t) = given
12	H(t) = given	Stage–Discharge
13	Stage–Discharge	Y(L, t) = given
14	Stage–Discharge	Q(L, t) = given

Downstr. Depth Change	
Second	**Y(L, t)**
0	3.649
20	3.6
40	3.5
60	3.4
80	3.3
100	3.2
120	3.1
140	3.0
160	2.9
180	2.8
200	2.6
220	2.5
240	″

Upstream Depth Change	
Second	**Y(0, t)**
0	2.46
20	2.44
40	2.42
60	2.40
80	2.38
100	″

Flow Rate Change	
Second	**Q(t)**
0	17.05
20	18.0
40	20.0
60	22.0
80	24.0
100	26.0
120	28.0
140	30.0
160	″

Downstream Stage–Discharge Relationship Files					
Depth, Y	3.4	3.5	3.6	3.649	3.7
Flow rate, Q	14.0	16.0	17.0	17.05	20.0

Other relationships, etc. given in input data given under Solution.

Solution

The input data files for Program IMPLICBC to solve these problems are given below. The solution results from Case 9 and Case 10 are plotted. Note for Case 9 that the depth is decreased at the downstream end of the channel in increments of 0.2 m, starting with 3.6 m at 0 second to 2.5 m at 240 s and held constant thereafter. As this depth decreases the flow rate increases just over 50 m^3/s when the depth first drops down to 2.5 m, but then decreases thereafter to 42 m^3/s at 400 s. By about 260 s all the channel storage under the M_1-GVF profile has been removed, and during subsequent times the depths fall below the normal depth (Y_o = 2.01 m and Q_o = 47 m^3/s), and for latter times of the simulation the depths increase at the end of the channel. Since the downstream depth is held above the normal depth, the flow rate will approach a value less than Q_o.

For Case 10 the flow rate is increased from 17 m^3/s to 30 m^3/s at the downstream end. For this case the depth decreases to a minimum of 3.3 m about 240 s but then increases again at subsequent time steps. The simulation was successful for only 320 s.

```
Case 1
3 0 21 20 10 20 50 9.81 .2 .000001 100 1 1
17.05/
5./
1.3/
.0135/
.0012/
2.46 2.44 2.42 2.40 2.38 2.38/
3.649 3.6 3.5 3.4 3.3 3.2 3.1 3. 2.9 2.8 2.7 2.6 2.5 2.5/
3.649 3.589,3.529,3.470,3.410,3.350,3.290,3.231,3.171,
3.112,3.052,2.993,2.993,2.874,2.814,2.755,
2.696,2.637,2.578,2.519,2.460
```

```
Case 2
3 0 21 20 10 20 50 9.81 .2 .000001 100 1 2
17.05/
5./
1.3/
.0135/
.0012/
2.46 2.44 2.42 2.40 2.38 2.38/
17.05 18 20 22 24 26 28 30 30/
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,
3.171,3.112,3.052,2.993,2.993,2.874,2.814,
2.755,2.696,2.637,2.578,2.519,2.460

Case 3
3 0 21 20 10 20 50 9.81 .2 .000001 100 1 4
17.05/
5./
1.3/
.0135/
.0012/
2.46 2.44 2.42 2.40 2.38 2.38/
.48 .5 .55 .6 .65 .7 .75 .8 .85 .9 .95 1. 1./
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,
3.171,3.112,3.052,2.993,2.993,2.874,2.814,
2.755,2.696,2.637,2.578,2.519,2.460

Case 4
3 0 21 20 20 20 50 9.81 .2 .000001 100 1 3
17.05/
5./
1.3/
.0135/
.0012/
2.46 2.48 2.50 2.52 2.54 2.54/
3.4 14 3.5 16 3.6 17 3.649 17.05 3.7 20/
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,
3.171,3.112,3.052,2.993,2.993,2.874,2.814,
2.755,2.696,2.637,2.578,2.519,2.460

Case 5
3 0 21 20 10 20 50 9.81 .2 .000001 100 2 1
17.05/
5./
1.3/
.0135/
.0012/
17.05 18 20 22 24 26 28 30 30/
3.649 3.6 3.5 3.4 3.3 3.2 3.1 3. 2.9 2.8 2.7 2.6 2.5 2.5/
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,
3.171,3.112,3.052,2.993,2.993,2.874,2.814,
2.755,2.696,2.637,2.578,2.519,2.460
```

```
Case 6
3 0 21 20 10 20 50 9.81 .2 .000001 100 2 2
17.05/
5./
1.3/
.0135/
.0012/
17.05 18 20 22 24 26 28 30 30/
17.05 18 20 22 24 26 28 30 30/
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,
3.171,3.112,3.052,2.993,2.993,2.874,2.814,
2.755,2.696,2.637,2.578,2.519,2.460

Case 7
3 0 21 20 10 20 50 9.81 .2 .000001 100 2 4
17.05/
5./
1.3/
.0135/
.0012/
17.05 18 20 22 24 26 28 30 30/
.48 .5 .55 .6 .65 .7 .75 .8 .85 .9 .95 1. 1./
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,
3.171,3.112,3.052,2.993,2.993,2.874,
2.814,2.755,2.696,2.637,2.578,2.519,2.460

Case 8
3 0 21 20 20 20 50 9.81 .2 .000001 100 2 3
17.05/
5./
1.3/
.0135/
.0012/
17.05 18 20 22 24 26 28 30 30/
3.4 14 3.5 16 3.6 17 3.649 17.05 3.7 20/
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,
3.171,3.112,3.052,2.993,2.993,2.874,
2.814,2.755,2.696,2.637,2.578,2.519,2.460

Case 9
3 0 21 20 10 20 50 9.81 .2 .000001 100 4 1
17.05/
5./
1.3/
.0135/
.0012/
2.49643 2.49643/
3.649 3.6 3.5 3.4 3.3 3.2 3.1 3. 2.9 2.8 2.7 2.6 2.5 2.5/
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,3.171,
3.112,3.052,2.993,2.993,2.874,2.814,
2.755,2.696,2.637,2.578,2.519,2.460
```

```
Case 10
3 0 21 20 30 20 50 9.81 .2 .00005 100 4 2
17.05/
5./
1.3/
.0135/
.0012/
2.49643 2.49643/
17.05 18 20 22 24 26 28 30 30/
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,3.171,3.112,
3.052,2.993,2.993,2.874,2.814,2.755,2.696,2.637,2.578,2.519,
2.460
etc.

Case 12
3 0 21 20 10 20 50 9.81 .2 .00001 100 4 3
17.05/
5./
1.3/
.0135/
.0012/
2.49643 2.49643/
3.4 40 3.5 30. 3.6 20. 3.649 17.05 3.7 15/
3.649,3.589,3.529,3.470,3.410,3.350,3.290,3.231,
3.171,3.112,3.052,2.993,2.993,2.874,
2.814,2.755,2.696,2.637,2.578,2.519,2.460
etc.

IMPLICBC.FOR
      COMMON Y(41),Q(41),B(41),FM(41),FN(41),SO(41),SMS(41),
     &QGIU(0:100),QGID(0:100),DELX,DELT,RDT,G,H,W,ERR,G2,
     &CMAN,DXT,NX,NT,NXM,NTBETW,MAX,IOUT,IUBC,IDBC,NUSD,NDSD
      DATA ONE/-1./
      READ(2,*) IOUT,NFORW,NX,NT,NTBETW,DELT,DELX,G,W,ERR,MAX,
     &IUBC,IDBC
C IUBC & IDBE are Upst. & Downst B.C. according to:
C 1-DEPTH known(given in QGIU or QGID);
C 2-FLOWRATE known(given in QGIU or QGDD);
C 3-STAGE-DISCHARGE given as pairs Y and Q
C  in QGIU or QGID;
C 4-H(t)=y+Q*Q/(2gA*A) given in QGIU and
C  VELOCITY known for downstr. B.C. & given in QGID.
      DELT=DELT/FLOAT(NTBETW)
      G2=2.*G
      CMAN=1.
      IF(G.GT.30.) CMAN=1.486
      DXT=DELX/DELT
      RDT=1./DELT
      NXM=NX-1
      DO 10 I=2,NX
      Q(I)=ONE
```

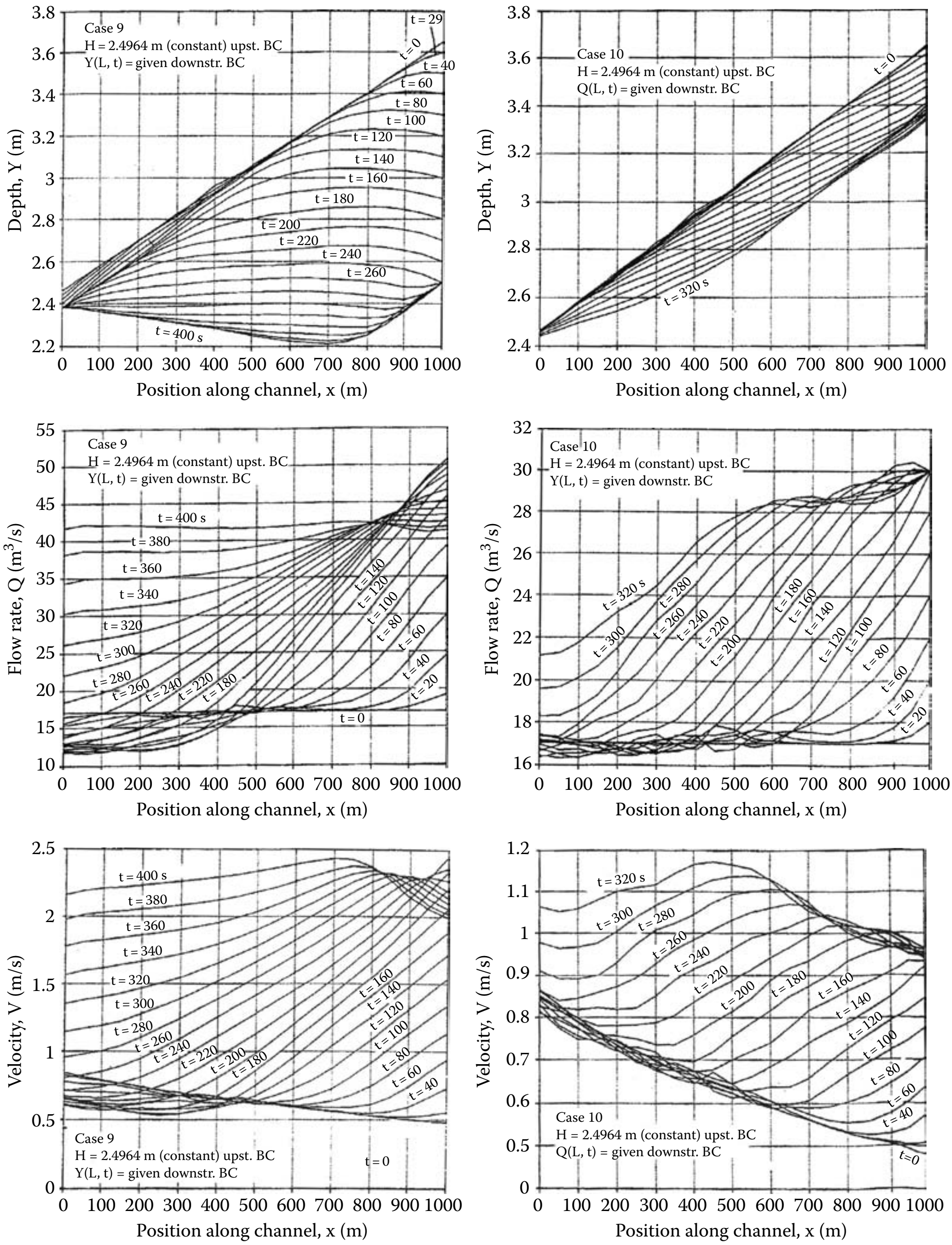
```
      B(I)=ONE
      FM(I)=ONE
      FN(I)=ONE
10    SO(I)=ONE
      DO 12 I=1,NT
      QGIU(I)=ONE
12    QGID(I)=ONE
      READ(2,*) (Q(I),I=1,NX)
      READ(2,*) (B(I),I=1,NX)
      READ(2,*) (FM(I),I=1,NX)
      READ(2,*) (FN(I),I=1,NX)
      READ(2,*) (SO(I),I=1,NX)
      READ(2,*) (QGIU(I),I=0,NT)
      READ(2,*) (QGID(I),I=0,NT)
      DO 20 I=2,NX
      IM=I-1
      IF(Q(I).EQ.ONE) Q(I)=Q(IM)
      IF(B(I).EQ.ONE) B(I)=B(IM)
      IF(FM(I).EQ.ONE) FM(I)=FM(IM)
      IF(FN(I).EQ.ONE) FN(I)=FN(IM)
      IF(SO(I).EQ.ONE) SO(I)=SO(IM)
20    SMS(I)=1.33333333*SQRT(FM(I)**2+1.)
      SMS(1)=1.33333333*SQRT(FM(1)**2+1.)
      NUSD=0
      NDSD=0
      DO 22 I=1,NT
      IF(QGIU(I).EQ.ONE) THEN
      IF(IUBC.NE.3) THEN
      QGIU(I)=QGIU(I-1)
      ELSE IF(NUSD.EQ.0) THEN
      NUSD=I-2
      ELSE
      GO TO 18
      ENDIF
      ENDIF
18    IF(QGID(I).EQ.ONE) THEN
      IF(IDBC.NE.3) THEN
      QGID(I)=QGID(I-1)
      ELSE IF(NDSD.EQ.0) THEN
      NDSD=I-2
      ELSE
      GO TO 22
      ENDIF
      ENDIF
22    CONTINUE
      IF(NFORW.EQ.1) THEN
      READ(2,*)(Y(I),I=1,NX)
      ELSE
      READ(2,*)(Y(I),I=NX,1,-1)
      ENDIF
      H=Y(1)+(Q(1)/((B(1)+FM(1)*Y(1))*Y(1)))**2/G2
```

```
      CALL SOR
      END
      SUBROUTINE SOR
      COMMON Y(41),Q(41),B(41),FM(41),FN(41),SO(41),SMS(41),
     &QGIU(0:100),QGID(0:100),DELX,DELT,RDT,G,H,W,ERR,G2,CMAN,
     &DXT,NX,NT,NXM,NTBETW,MAX,IOUT,IUBC,IDBC,NUSD,NDSD
      REAL D1(41),D2(41),D3(41),D4(41),E1(41),F1(41),F2(41),
     &A(41),P(41),T(41),V(41),SF(41)
      TIME=0.0
      JDSD=0
      JUSD=0
      DO 5 I=1,NX
      A(I)=(B(I)+FM(I)*Y(I))*Y(I)
      V(I)=Q(I)/A(I)
      P(I)=B(I)+1.5*SMS(I)*Y(I)
      T(I)=B(I)+2.*FM(I)*Y(I)
5     SF(I)=(FN(I)*Q(I)/(CMAN*A(I)))**2*(P(I)/A(I))**
     &1.33333333
      WRITE(IOUT,110) 0.
110   FORMAT(///1X,'Solution for time=',F8.2/1X,65('-')/,4X,
     &'x',7X,'Y',7X,'V',7X,'Sf',7X,'P',7X,'Q',7X,'A',5X,'TOP'/
     &1X,65('-'))
      WRITE(IOUT,120) (DELX*FLOAT(I-1),Y(I),V(I),SF(I),P(I),
     &Q(I),A(I),T(I),I=1,NX)
120   FORMAT(F8.1,2F8.3,F8.5,4F8.3)
      K=0
10    K=K+1
      IF(IUBC.NE.3) THEN
      QGIU1=QGIU(K-1)
      DQGIU=(QGIU(K)-QGIU(K-1))/FLOAT(NTBETW)
      ENDIF
      IF(IDBC.NE.3) THEN
      QGID1=QGID(K-1)
      DQGID=(QGID(K)-QGID(K-1))/FLOAT(NTBETW)
      ENDIF
      DO 80 K10=1,NTBETW
      IF(IUBC.NE.3) QGIUI=QGIU1+DQGIU
      IF(IDBC.NE.3) QGIDI=QGID1+DQGID
      IF(IUBC.EQ.1) Y(1)=QGIUI
      IF(IUBC.EQ.2) Q(1)=QGIUI
      IF(IDBC.EQ.1) Y(NX)=QGIDI
      IF(IDBC.EQ.2) Q(NX)=QGIDI
      TIME=TIME+DELT
      DO 50 I=2,NXM
      C2=G*A(I)/T(I)
      E1(I)=T(I)*DXT
      F1(I)=E1(I)*Y(I)
      D1(I)=T(I)*(C2-V(I)**2)/(2.*DELX)
      D2(I)=V(I)/DELX
      D3(I)=G2*SF(I)*(SMS(I)*A(I)/P(I)-1.6666667*T(I))
      D4(I)=RDT+G2*SF(I)*A(I)/Q(I)
```

```
50    F2(I)=Q(I)/DELT+Y(I)*(D3(I)+V(I)**2*(B(I+1)-B(I-1)+
     &Y(I)*(FM(I+1)-FM(I-1)))/(2.*DELX))+G*A(I)*(SO(I)+SF(I))
      C=SQRT(G*A(1)/T(1))
      VMC=V(1)-C
      VPC=V(1)+C
      FK=CMAN*A(1)**1.6666667/(FN(1)*P(1)**0.6666667)
      GAQK=G*A(1)*ABS(Q(1))/FK**2
      PKY=FK/A(1)*(1.6666667*T(1)-SMS(1)*A(1)/P(1))
      A1=-T(1)*VPC*(RDT-VMC/DELX)+2.*GAQK*Q(1)*PKY/FK
      A2=RDT-VMC/DELX-2.*GAQK
      A3=-T(1)*VPC*VMC/DELX
      A4=VMC/DELX
      F4=T(1)*SO(1)*C**2+(A(2)-A(1))/DELX*V(1)**2-GAQK*Q(1)+
     &2.*GAQK*(-Q(1)+Q(1)*PKY*Y(1)/FK)+(Q(1)-T(1)*VPC*Y(1))/
     &DELT
      C=SQRT(G*A(NX)/T(NX))
      VMC=V(NX)-C
      VPC=V(NX)+C
      FK=CMAN*A(NX)**1.6666667/(FN(NX)*P(NX)**0.6666667)
      GAQK=G*A(NX)*ABS(Q(NX))/FK**2
      PKY=FK/A(NX)*(1.6666667*T(NX)-SMS(NX)*A(NX)/P(NX))
      B1=T(NX)*VMC*VPC/DELX
      B2=-VPC/DELX
      B3=-T(NX)*VMC*(RDT+VPC/DELX)+2.*GAQK*Q(NX)*PKY/FK
      B4=RDT+VPC/DELX-2.*GAQK
      F5=T(NX)*SO(NX)*C**2+(A(NX)-A(NXM))/DELX*V(NX)
     &**2-GAQK*Q(NX)+2.*GAQK*(-Q(NX)+Q(NX)*PKY*Y(NX)/
     &FK)+(Q(NX)-T(NX)*VMC*Y(NX))/DELT
      M=0
60    SUM=0.0
      GO TO (61,62,65,64),IDBC
61    QT=(F5-B1*Y(NXM)-B2*Q(NXM)-B3*Y(NX))/B4
      DIF=QT-Q(NX)
      Q(NX)=QT+W*DIF
      SUM=SUM+DIF*DIF
      GO TO 66
62    YT=(F5-B1*Y(NXM)-B2*Q(NXM)-B4*Q(NX))/B3
      GO TO 63
64    QT=QGIDI*(B(NX)+FM(NX)*Y(NX))
      YT=(F5-B1*Y(NXM)-B2*Q(NXM))/(B3+QT*B4)
      Q(NX)=QT*YT
63    DIF=YT-Y(NX)
      Y(NX)=YT+W*DIF
      SUM=SUM+DIF*DIF
      GO TO 66
65    CALL STAGE(NX,Y,Q,B3,B4,B1,B2,F5,SUM,JDSD,NDSD,QGID)
66    DO 70 I=NXM,2,-1
      YT=(F1(I)+0.5*Q(I-1)-0.5*Q(I+1))/E1(I)
      DIF=YT-Y(I)
```

```
      Y(I)=YT+W*DIF
      SUM=SUM+DIF*DIF
      QT=(F2(I)+D1(I)*Y(I-1)+D2(I)*Q(I-1)-D3(I)*Y(I)-
     &D1(I)*Y(I+1)-D2(I)*Q(I+1))/D4(I)
      DIF=QT-Q(I)
      Q(I)=QT+W*DIF
70    SUM=SUM+DIF*DIF
C Newton method to solve Energy & C+ characteristic
c    eqs. simultaneously
      IF(IUBC.EQ.4) THEN
      FF1=A1*Y(1)+A2*Q(1)+A3*Y(2)+A4*Q(2)-F4
      AR=(B(1)+FM(1)*Y(1))*Y(1)
      FF2=QGIUI-Y(1)-(Q(1)/AR)**2/G2
      FAC=((Q(1)/AR)**2/(G*AR)-1.)/A1
      D22=-Q(1)/(G*AR**2)-FAC*A2
      FF2=FF2-FF1*FAC
      DIF=FF2/D22
      Q(1)=Q(1)-DIF
      SUM=SUM+DIF*DIF
      DIF=(FF1-A2*DIF)/A1
      Y(1)=Y(1)-DIF
      SUM=SUM+DIF*DIF
      GO TO 75
      ENDIF
      IF(IUBC.EQ.1) GO TO 72
      IF(IUBC.EQ.3) THEN
      CALL STAGE(1,Y,Q,A1,A2,A3,A4,F4,SUM,JUSD,NUSD,QGIU)
      GO TO 75
      ENDIF
71    YT=(F4-A2*Q(1)-A3*Y(2)-A4*Q(2))/A1
      DIF=YT-Y(1)
      Y(1)=YT+W*DIF
      SUM=SUM+DIF*DIF
      IF(IUBC.EQ.4) THEN
      IF(Y(1).GE.QGIUI) THEN
      Q(1)=0.
      ELSE
      Q(1)=(B(1)+FM(1)*Y(1))*Y(1)*SQRT(G2*(QGIUI-Y(1)))
      ENDIF
      ENDIF
      GO TO 75
72    QT=(F4-A1*Y(1)-A3*Y(2)-A4*Q(2))/A2
      DIF=QT-Q(1)
      Q(1)=QT+W*DIF
      SUM=SUM+DIF*DIF
75    M=M+1
      IF(SUM.GT.ERR.AND.M.LE.MAX) GO TO 60
      IF(M.GE.MAX) GO TO 90
      DO 78 I=1,NX
```

```
      A(I)=(B(I)+FM(I)*Y(I))*Y(I)
      V(I)=Q(I)/A(I)
      P(I)=B(I)+1.5*SMS(I)*Y(I)
      T(I)=B(I)+2.*FM(I)*Y(I)
78    SF(I)=(FN(I)*Q(I)/(CMAN*A(I)))**2*(P(I)/A(I))
     &**1.33333333
      IF(K10.EQ.NTBETW) THEN
      WRITE(IOUT,110) TIME
      WRITE(IOUT,120) (DELX*FLOAT(I-1),Y(I),V(I),SF(I),P(I),
     &Q(I),A(I),T(I),I=1,NX)
      ENDIF
      QGIU1=QGIUI
80    QGID1=QGIDI
      IF(K.GE.NT) STOP
      GO TO 10
90    WRITE(*,130) TIME,SUM
130   FORMAT(//' Failed to converge. Time=',F8.2,
     &'SUM=',E12.4)
      IF(SUM.LT. .001) GO TO 10
      END
      SUBROUTINE STAGE(I,Y,Q,A1,A2,A3,A4,F4,SUM,JJ,M,QGIV)
      REAL QGIV(0:100),Y(41),Q(41)
      IF(I.EQ.1) THEN
      I1=2
      ELSE
      I1=I-1
      ENDIF
10    IF(QGIV(JJ).GT.Y(I) .AND. JJ.GT.1) THEN
      JJ=JJ-2
      GO TO 10
      ENDIF
20    IF(QGIV(JJ+2).LE.Y(I) .AND. JJ.LT.M) THEN
      JJ=JJ+2
      GO TO 20
      ENDIF
      DQ=(QGIV(JJ+3)-QGIV(JJ+1))/(QGIV(JJ+2)-QGIV(JJ))
30    FF1=A1*Y(I)+A2*Q(I)+A3*Y(I1)+A4*Q(I1)-F4
      FF2=QGIV(JJ+1)+DQ*(Y(I)-QGIV(JJ))-Q(I)
      FAC=DQ/A1
      D22=-1.-FAC*A2
      FF2=FF2-FAC*FF1
      DIF=FF2/D22
      Q(I)=Q(I)-DIF
      SUM=SUM+DIF*DIF
      DIF=(FF1-A2*DIF)/A1
      Y(I)=Y(I)-DIF
      SUM=SUM+DIF*DIF
      RETURN
      END
```



7.10 CRANK–NICOLSON NEWTON ITERATIVE IMPLICIT METHOD

In the last section, the St. Venant equations were differenced so that a linear system of algebraic equations was produced for the advanced time step. The solution of this system of equations advances the solution of the unsteady channel flow through one time increment. To produce this linear system of equations it was necessary that several terms were evaluated at the kth time step, and by so doing the order of the approximation of derivatives was only first order. A method that allows second-order approximations of both space and time derivatives is the Crank–Nicolson method. The Crank–Nicolson method evaluates all terms in the St. Venant equations at k + ½ time step, or midway between the current time step k and the advanced time step k + 1 at which the new time step solution is being sought. Since the St. Venant equations are nonlinear the resulting system of algebraic equations is nonlinear, and must be solved by an iterative technique such as the Newton method. This Crank–Nicolson Newton iterative implicit method has wide application to a variety of problems in many different fields, and is widely used to solve problems governed by Parabolic partial differential equations. Because this method has such wide application to a large variety of problems it will be developed in this section, even though variations of it with special differences to take advantage of the way the characteristics propagate information from the disturbed to the "zone of quite" have been used more widely in computer programs that solve unsteady open channel flows because of stability considerations.

The two St. Venant equations that will be used in the Crank–Nicolson method are as follows:

$$2V\frac{\partial Q}{\partial x}+T(c^2-V^2)\frac{\partial Y}{\partial x}-V^2\left.\frac{\partial A}{\partial x}\right|_{Y,t}+gA(S_f-S_o+F_q)-Vq^*+\frac{\partial Q}{\partial t}=0 \tag{7.25a}$$

and

$$\frac{\partial Q}{\partial x}-q^*+T\frac{\partial Y}{\partial t}=0 \tag{7.24a}$$

Using second-order differences in place of all derivatives in these equations, and evaluating all terms at grid position i = 1 + x/Δx and at time step k + ½ results in the following two equations:

$$\begin{aligned}F_{i1}=\;&2V_i^{k+1}\left(Q_{i+1}^{k+1}-Q_{i-1}^{k+1}\right)+\left[T\left(c^2-V^2\right)\right]_i^{k+1}\left(Y_{i+1}^{k+1}-Y_{i-1}^{k+1}\right)-(V^2)_i^{k+1}\left[\left.\frac{\partial A}{\partial x}\right|_{Y,t}\right]_i^{k+1}\\&+2g\Delta xA_i^{k+1}\left\{S_{fi}^{\,k+1}-S_o+F_{qi}^{\,k+1}\right\}-2\Delta x\left[Vq^*\right]_i^{k+1}+\frac{4\Delta x}{\Delta t}Q_i^{k+1}+2V_i^k\left(Q_{i+1}^k-Q_{i-1}^k\right)\\&+\left[T(c^2-V^2)\right]_i^k\left(Y_{i+1}^k-Y_{i-1}^k\right)-\left(V^2\right)_i^k\left[\left.\frac{\partial A}{\partial x}\right|_{Y,t}\right]_i^k+2g\Delta xA_i^k\left\{S_{fi}^{\,k}-S_o+F_{qi}^{\,k}\right\}-2\Delta x\left[Vq^*\right]_i^k\\&-\frac{4\Delta x}{\Delta t}Q_i^k=0\end{aligned} \tag{7.80}$$

and

$$F_{i2}=Q_{i+1}^{k+1}-Q_{i-1}^{k+1}-2\Delta x(q^*)_i^{k+1}+\frac{2\Delta x}{\Delta t}\left(T_i^{k+1}+T_i^k\right)\left(Y_i^{k+1}-Y_i^k\right)+Q_{i+1}^k-Q_{i-1}^k-2\Delta x(q^*)_i^k=0 \tag{7.81}$$

The space derivatives with respect to x in the above equations have been evaluated midway between the time step k and the time step k + 1 or at time (k + ½) by taking the average of the difference at the kth and the (k + 1)th time step. Second-order differences of the time derivatives, evaluated at the (k + (½) time position, are: $\partial Y/\partial t = (Y^{k+1} - Y^k)/\Delta t$ and $\partial Q/\partial t = (Q^{k+1} - Q^k)/\Delta t$. These are second order difference approximations when the derivative is evaluated at k + ½ based on an increment Δt/2, i.e., a second degree polynomial (parabola) is used to define the variation of Y and Q as functions of time. In Equations 7.80 and 7.81 the quantities with a super k are known and those with a super k + 1 are unknown, unless specified as part of a boundary condition. The initial condition provides values to Y^k and Q^k (as well as V^k, c^k, etc.) when t = 0 (or when k = 1 with k defined by $k = 1 + t/\Delta t$) and for subsequent time steps these values are known from the past time step solution. The Crank–Nicolson method calls for equal weighting of space derivatives at the k and k + 1 time steps, or the average of these two values, and only then is a second order approximation of the time derivative used. However, often a weightings w and $w_1 = 1 - w$ are used to current and new time derivatives.

The difference Equations 7.80 and 7.81 have been denoted by F_{i1} and F_{i2} to distinguish that there are a pair of equations that result from each new grid point $i = 1 + x/\Delta x$. The first of this pair is the finite difference equation obtained from the equation of motion, Equation 7.80, and the second of the pair comes from the continuity equation, Equation 7.81. Thus the above system of equations represented by Equations 7.80 and 7.81 constitute a system of approximately 2n equations, where n is the number of grid point, or stations, taken along the length of the channel. The word approximately is used because the boundary conditions that are discussed below will have some effect on the exact number of equations.

An appropriate initial condition, that is a solution to a steady problem, will satisfy Equations 7.80 and 7.81 with the terms that come from $\partial Q/\partial t$ and $\partial Y/\partial t$ are omitted. In other words, advancing through a time step by solving Equations 7.80 and 7.81, with boundary conditions that change nothing should reproduce the "initial condition," e.g., the flow remains steady state. Exact reproduction of the steady state variables may not occur, however, because the initial condition is generally obtained using a much smaller (and variable) space increment Δx in solving the ODE than is practical when solving the unsteady problem. When lateral inflow or outflow ($q^* \neq 0$) occurs, or the channel is nonprismatic, it is important to interpret the terms in Equations 7.80 and 7.81 involving q^* and $\partial A/\partial x$ in a consist manner with the differencing scheme used. For example, assume in obtaining the steady state solution for the initial condition for an unsteady state problem that has outflow q^* occurring over a length L_{out}, that the increment Δx used in the unsteady solution is L_{out} ($\Delta x = L_{out}$) and outflow occurs between i = 10 and i = 11. If the upstream flow rate is Q_{up} then after the lateral outflow the flow rate is $Q_{dow} = Q_{up} - \Delta x q^*$. Thus Q_i for i = 1 to 10 will equal Q_{up} while Q_i for i = 11 to N will equal Q_{dow}. To satisfy Equation 7.81 the quantity $2\Delta x q^*$ must equal $q^* L_{out}$ at both grid point 10 and also 11, or 1/2 of $2\Delta x q^*$ is applied at each of two grid points rather than the total amount at one grid point. Why this is necessary can be seen by noting if Equation 7.81 is to satisfy the initial steady state conditions then $Q_{i+1} - Q_{i-1} - 2\Delta x q^*$ must equal zero at all grid points. When spanning intervals from subscript i + 1 to i − 1, which is $2\Delta x$ long, where there is no lateral outflow q^* is zero and $Q_{i+1} = Q_{i-1}$. These intervals would be defined for i = 2 to i = 9 and i = 12 to i = n − 1 (where n is the total count of x grid lines used in the finite difference solution). When i = 10 then $Q_{11} - Q_9$ will equal the total outflow $\Delta x q^*$. Likewise when i = 11 $Q_{12} - Q_{10} = \Delta x q^*$. The same is necessary to satisfy Equation 7.80.

Various types of boundary conditions have been discussed previously. For illustrative purposes the finite difference equations that need to be added to Equations 7.80 and 7.81 from boundary conditions, we will assume that the channel is supplied by a reservoir at its upstream end whose water surface is a height H above the bottom of the channel, and that at the downstream end of the channel either the flow rate Q(L, t), or the depth Y(L, t) is specified at a known function of time. In general height H may also be a known function of time. At the upstream end of the channel the energy equation

$$F_1 = H - Y - (1 + K_e)\frac{Q^2}{2gA^2} = 0$$

must be satisfied. The finite differences of this equation at the advanced time step becomes part of the system of equation defined by Equations 7.80 and 7.81. This equation is

$$F_{11} = H^{k+1} - Y_1^{k+1} - (1 + K_e)\left(\frac{Q^2}{2gA^2}\right)_1^{k+1} = 0 \tag{7.82}$$

Since there are two unknowns at the upstream end of the channel, Q_1^{k+1} and Y_1^{k+1} a second finite difference equation is needed at the upstream end of the channel. This second equation comes from satisfying the continuity equation, or,

$$F_{12} = Q_2^{k+1} - Q_1^{k+1} + \frac{2\Delta x}{\Delta t} T_1^{k+\frac{1}{2}} \left(Y_i^{k+1} - Y_i^k \right) - 2\Delta x (q^*)_1^{k+\frac{1}{2}} + Q_2^k - Q_1^k = 0 \tag{7.83}$$

Generally q* will equal zero at the upstream end of the channel since the inflow is coming from the reservoir, so the term involving q* can be dropped from Equation 7.83. Note that in evaluating the partial derivative of Q with respect to x in Equation 7.61 that only a first-order approximation is used, i.e., $\partial Q/\partial x = (Q_2 - Q_1)/\Delta x$ and to evaluate this derivative at the k + ½ time step the average is taken at the k + 1 and k time steps.

If the flow rate is known as a function of time at the downstream end of the channel then $Q_n^{k+1} = Q(L,t)$ is specified and only one unknown exists, namely, Y_n^{k+1}. Thus only one finite difference equation is required for the downstream boundary condition to complement the system of equations from Equations 7.80 through 7.83. This additional finite difference equation comes from the continuity Equation 7.61 and is

$$F_{n1} = F_{2n-1} = \frac{1}{4} Q_{n-2}^{k+1} - Q_{n-1}^{k+1} + \frac{3}{4} Q_n^{k+1} - \Delta x (q^*)_n^{k+\frac{1}{2}} + \frac{\Delta x}{\Delta t} T_n^{k+\frac{1}{2}} \left(Y_n^{k+1} - Y_n^k \right) + \frac{1}{4} Q_{n-2}^k - Q_{n-1}^k + \frac{3}{4} Q_n^k = 0 \tag{7.84}$$

In obtaining Equation 7.84 a second-order difference has been used to obtain the partial derivative of Q with respect to x, namely, $\partial Q/\partial x = (Q_{n-2}/2 - 2Q_{n-1} + 3Q_n/2)/\Delta x$, and to evaluate the terms midway between the time step k and k + 1 the average of this approximation to the derivative at k and k + 1 has been used. The Δx that multiplies q* comes about because the resulting difference equation has been multiplied by this quantity. The notation F_{n1} indicates this is the first of the nth pair of equations, but there is no second equation F_{n2}. This is actually the 2n − 1, and last, equation in the system of equations. An alternative subscript notation would be to use a single subscript rather then the double subscript with subscript (i,1) replaced by (2i − 1) and (i,2) replaced by (2i). Seldom does lateral inflow or outflow occur at the end of the channel besides that specified by $Q_n^{k+1} = Q(L,t)$ and the term involving q* can be dropped from Equation 7.84.

For the downstream boundary condition when the depth is specified as a function of time, or $Y(L,t)$ is known. It will be assumed that the time rate of this change is also known, or $(\partial Y/\partial t)_n$ is known. Again the continuity equation produces the one additional finite difference equation needed to solve for the unknown flow rate Q_n^{k+1}, and this equation is

$$F_{n1} = F_{2n-1} = \frac{1}{4} Q_{n-2}^{k+1} - Q_{n-1}^{k+1} + \frac{3}{4} Q_n^{k+1} - \Delta x (q^*)_n^{k+\frac{1}{2}} + \frac{\Delta x}{\Delta t} \left(T \frac{\partial Y}{\partial t} \right)_n^{k+\frac{1}{2}} + \frac{1}{4} Q_{n-2}^k - Q_{n-1}^k + \frac{3}{4} Q_n^k = 0 \tag{7.85}$$

There are 2n − 1 equations in the system regardless of whether Equation 7.84 or Equation 7.85 (the flow rate, or the depth is known as a function of time) is used. In either case one of the two variables Y_n^{k+1} or Q_n^{k+1} is known. The difference is that if Q is specified at the downstream end of the channel as a function of time then the last unknown is Y_n^{k+1}, whereas if Y is specified at the downstream end of the channel as a function of time, then the last unknown is Q_n^{k+1}. If this vector of unknowns is denoted by {X} then for Q(L, t) known

$$\{X\}^T = \left[Y_1 \; Q_1 \; Y_2 \; Q_2 \; Y_3 \; Q_3 \ldots Y_{i-1} \; Q_{i-1} \; Y_i \; Q_i \; Y_{i+1} \; Q_{i+1} \ldots Y_{n-1} \; Q_{n-1} \; Y_n \right]$$

whereas for Y(L, t) known

$$\{X\}^T = \left[Y_1\ Q_1\ Y_2\ Q_2\ Y_3\ Q_3 \ldots Y_{i-1}\ Q_{i-1}\ Y_i\ Q_i\ Y_{i+1}\ Q_{i+1} \ldots Y_{n-1}\ Q_{n-1}\ Q_n\right]$$

The Newton method obtains the solution to this unknown vector {X} from the iterative equation

$$\{X\}^{m+1} = \{X\}^m - \{Z\}^m \tag{7.86}$$

in which the correction vector $\{Z\}^{(m)}$ is obtained by solving the linear system of equations defined by

$$[D]^m\{Z\}^m = \{F\}^m \tag{7.87}$$

in which {F} is the equation vector consisting of the system of equations defined by Equations 7.80, 7.81, and the appropriate boundary condition equations

$$\{F\}^T = \left[F_{11}\ F_{12}\ F_{21}\ F_{22} \ldots F_{i1}\ F_{i2} \ldots F_{n-2,1}\ F_{n-2,2}\ F_{n-1,1}\ F_{n-1,2}\ F_{n1}\right]$$

and [D] is the Jacobian matrix consisting of the derivatives of the equations with respect to the unknowns. The first row of this matrix consist of the derivative of F_{11} with respect to all the unknowns, the second row consist of the derivatives of F_{12} with respect to all the unknowns, etc. until the last row consist of the derivatives of $F_{n1} = F_{2n-1}$ with respect to all the unknowns. Since most of these derivatives will be zero, the Jacobian [D] is a sparse matrix with non zero element clustered about the diagonal as shown below in which the diagonal elements are shown in bold type. Thus the solution to the linear system of equations, represented by Equation 7.87, should consist of a special algorithm that takes advantage of the zero elements, much as was done with the previously described implicit solution method, that used "Gaussian elimination" followed by "back substitution." In the case of the system, Equation 7.87, it is necessary to first eliminate the first element in the last row of [D]. Thereafter, starting with the third row there are only two elements in front of each diagonal to eliminate, to reduce the coefficient matrix to an upper triangular matrix. The solution to this linear system of equations is implemented in the SUBROUTINE BAND given as part of the FORTRAN listing under the next example problem. In this program the two dimensional array A(101,6) represents the Jacobian matrix, [D]. For each pair of equations F_{i1} and F_{i2} the second subscript of A(I,J) holds up to six nonzero derivatives, with $\partial F_{i1}/\partial Y_i$ in A(2*I − 1,3) and $\partial F_{i2}/\partial Q_i$ in A(2*I,4). Since these elements of the Jacobian matrix multiply the elements of the unknown vector that will subtracted from Y_i and Q_i (the depth and flow rate at the grid point i) these can be considered the diagonal elements of the Jacobian matrix. Thus in general two nonzero elements exist in front of the diagonal in each row of the Jacobian matrix. A process that solves this linear algebraic problem using this subscripting arrangement for the Jacobian is defined by the program statements in SUBROUTINE BAND and a means for defining the derivatives is defined by the program statements in SUBROUTINE IMPLNE. Numerical evaluation of the derivatives is used for the diagonal elements since these are more complex, and the off-diagonal elements are evaluated by the equations giving these derivatives. The program listing uses the Crank–Nicolson method only if WT (a weighting of the advance time step variables, as described in the next section) is given a value equal to 0.5. The array FB contains the portion of the equations at the current time step super k, which does not change during consecutive Newton iterations. The equation vector is held in the array V before entering SUBROUTINE BAND, and thereafter V contains the solution vector that is subtracted from the unknown vector in implementing the Newton method.

$$
[D] = \begin{bmatrix}
\frac{\partial F_{11}}{\partial Y_1} & \frac{\partial F_{11}}{\partial Q_1} & 0 & & & & & & & & & \\
\frac{\partial F_{12}}{\partial Y_1} & \frac{\partial F_{12}}{\partial Q_1} & 0 & \frac{\partial F_{12}}{\partial Q_2} & 0 & & & & & & & \\
\frac{\partial F_{21}}{\partial Y_1} & \frac{\partial F_{21}}{\partial Q_1} & \frac{\partial F_{21}}{\partial Y_2} & \frac{\partial F_{21}}{\partial Q_2} & \frac{\partial F_{21}}{\partial Y_3} & \frac{\partial F_{21}}{\partial Q_3} & 0 & & & & & \\
0 & \frac{\partial F_{22}}{\partial Q_1} & \frac{\partial F_{22}}{\partial Y_2} & 0 & 0 & \frac{\partial F_{22}}{\partial Q_3} & 0 & & & & & \\
0 & 0 & \frac{\partial F_{31}}{\partial Y_2} & \frac{\partial F_{31}}{\partial Q_2} & \frac{\partial F_{31}}{\partial Y_3} & \frac{\partial F_{31}}{\partial Q_3} & \frac{\partial F_{31}}{\partial Y_4} & \frac{\partial F_{31}}{\partial Q_4} & 0 & & & \\
0 & 0 & 0 & \frac{\partial F_{32}}{\partial Q_2} & \frac{\partial F_{32}}{\partial Y_3} & 0 & 0 & \frac{\partial F_{32}}{\partial Q_4} & 0 & & & \\
 & & & & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \\
 & & & & & \frac{\partial F_{i1}}{\partial Y_{i-1}} & \frac{\partial F_{i1}}{\partial Q_{i-1}} & \frac{\partial F_{i1}}{\partial Y_i} & \frac{\partial F_{i1}}{\partial Q_i} & \frac{\partial F_{i1}}{\partial Y_{i+1}} & \frac{\partial F_{i1}}{\partial Q_{i+1}} & 0 \\
 & & & & & 0 & \frac{\partial F_{i2}}{\partial Q_{i-1}} & \frac{\partial F_{i2}}{\partial Y_i} & 0 & 0 & \frac{\partial F_{i2}}{\partial Q_{i+1}} & 0 \\
 & & & & & & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 & & & & & & & \frac{\partial F_{n-11}}{\partial Y_{n-2}} & \frac{\partial F_{n-11}}{\partial Q_{n-2}} & \frac{\partial F_{n-11}}{\partial Y_{n-1}} & \frac{\partial F_{n-11}}{\partial Q_{n-1}} & \frac{\partial F_{n-11}}{\partial Q_n} \\
 & & & \text{if Y (t) given at end} & & & & 0 & \frac{\partial F_{n-12}}{\partial Q_{n-2}} & \frac{\partial F_{n-12}}{\partial Y_{n-1}} & 0 & \frac{\partial F_{n-12}}{\partial Q_n} \\
 & & & & & & & & \frac{\partial F_n}{\partial Q_{n-2}} & 0 & \frac{\partial F_n}{\partial Q_{n-1}} & \frac{\partial F_n}{\partial Q_n} \\
 & & & & & & & \frac{\partial F_{n-11}}{\partial Y_{n-2}} & \frac{\partial F_{n-11}}{\partial Q_{n-2}} & \frac{\partial F_{n-11}}{\partial Y_{n-1}} & \frac{\partial F_{n-11}}{\partial Q_{n-1}} & \frac{\partial F_{n-11}}{\partial Y_n} \\
 & & & \text{if Q (t) given at end} & & & & 0 & \frac{\partial F_{n-12}}{\partial Q_{n-2}} & \frac{\partial F_{n-12}}{\partial Y_{n-1}} & 0 & \frac{\partial F_{n-12}}{\partial Y_n} \\
 & & & & & & & & \frac{\partial F_n}{\partial Q_{n-2}} & 0 & \frac{\partial F_n}{\partial Q_{n-1}} & \frac{\partial F_n}{\partial Y_n}
\end{bmatrix}
$$

EXAMPLE PROBLEM 7.6

A trapezoidal channel with b = 5 m, m = 1.3, n = 0.013, and $S_o = 0.0008$ is supplied water from a reservoir with a water depth of 3.5 m above the channel bottom. At the downstream end of this 1500 m long channel, there is a gate whose initial setting causes the flow depth downstream from the gate to be 1.7 m. The gate exist in a rectangular section with a width of 4 m. Steady-state flow has existed in the channel for some time, but then the gate is opened so as to increase the flow rate passing it at a rate of 0.25 m³/s for each 5 s until the flow rate equals 55 m³/s, and then is held constant thereafter. Obtain the solution for depths, velocities, and flow rates throughout the channel for some time after the flow rate at the gate is 55 m³/s. Assume the entrance loss coefficient and loss coefficient at the gate are both 0.05.

Solution

The first step in the solution process consists of determining the flow rate in the channel and the depths throughout the length of the channel under steady-state conditions. This solution is needed as the "initial condition" for the unsteady problem. This steady solution is based on using the methods described in Chapter 4 to solve algebraic and ordinary differential equations simultaneously. The equations that need to be solved for this problem are: (1) the upstream energy equation $H = Y_1 + (1 + K_e)(Q/A_1)^2/(2g)$, (2) the energy equation across the gate $Y_2 + (Q/A_2)^2/(2g) = Y_3 + (1 + K_L)(Q/A_3)^2/(2g)$, and (3) the ODE for GVF $dY/dx = (S_o - S_f)/(1 - F_r^2)$. In these equations, Y_1 represents the depth at the beginning of the channel, Y_2 the depth at the end of the channel just upstream from the gate, and $Y_3 = 1.7$ m is the specified depth downstream from the gate. The solution to these three equations produce the following values for the unknowns: Q = 49.49 m³/s, $Y_1 = 3.37$ m, and $Y_2 = 4.48$ m. Also the depth at 50 ft increments along the channel are obtained and consist of the following:

x	1500	1450	1400	1350	1300	1250	1200	1150
Y	4.4818	4.4433	4.4048	4.3664	4.3280	4.2898	4.2516	4.2134
	1100	1050	1000	950	900	850	800	750
	4.1754	4.1374	4.0995	4.0618	4.0241	3.9865	3.9490	3.9116
	700	650	600	550	500	450	400	350
	3.8743	3.8372	3.8001	3.7633	3.7265	3.6899	3.6535	3.6172
	300	250	200	150	100	50	0 m	
	3.5811	3.5452	3.5095	3.4739	3.4386	3.4035	3.3687	

The unsteady solution is implemented by solving the system of equations described above using the Newton method. The FORTRAN listing below is a program designed to accomplish this solution:

Program IMPLICAL.FOR

```
      COMMON /SOLVNE/ A(101,6),V(101),FB(101)
      LOGICAL LBC
      COMMON Y(51),YB(51),AA(51),Q(51),PP(51),TOP(51),
     &TB(51),V1(51),X(51),SQ(51),B(51),FM(51),SO(51),
     &FN(51),SMS(51),SF(51),WT,WT1,QGIV(0:200),DXG,G,G2,
     &DELX,GDELX,DXDT2,DEL2,DELT,DXT,RDT,H,CMAN,EK,
     &UQ,ERR,NT,NX,NXM,NX2,IOUT,IYOUT,LBC,IFREQ
      DATA ONE/-1./
      READ(2,*) IOUT,IFREQ,NFORW,NX,NT,DELT,DELX,G,IBC2,IYOUT,
     &UQ,EK,ERR,WT
      WT1=1.-WT
      LBC=.FALSE.
```

```
      IF(IBC2.EQ.2) LBC=.TRUE.
      G2=2.*G
      CMAN=1.
      IF(G.GT.30.) CMAN=1.486
      DEL2=2.*DELX
      DXT=DELX/DELT
      GDELX=DELX*G2
      DXDT2=2.*DXT
      EK=(1.+EK)/G2
      DXG=G*DELX
      RDT=1./DELT
      NXM=NX-1
      NX2=NX-2
      DO 10 I=1,NX
      Q(I)=ONE
      B(I)=ONE
      FM(I)=ONE
      FN(I)=ONE
10    SO(I)=ONE
      READ(2,*) (Q(I),I=1,NX)
      READ(2,*) (B(I),I=1,NX)
      READ(2,*) (FM(I),I=1,NX)
      READ(2,*) (FN(I),I=1,NX)
      READ(2,*) (SO(I),I=1,NX)
      DO 20 I=2,NX
      IM=I-1
      IF(Q(I).EQ.ONE) Q(I)=Q(IM)
      IF(B(I).EQ.ONE) B(I)=B(IM)
      IF(FM(I).EQ.ONE) FM(I)=FM(IM)
      IF(FN(I).EQ.ONE) FN(I)=FN(IM)
      IF(SO(I).EQ.ONE) SO(I)=SO(IM)
20    SMS(I)=2.*SQRT(FM(I)**2+1.)
      DO 21 I=2,NXM
21    SQ(I)=Q(I+1)-Q(I-1)
      SMS(1)=2.*SQRT(FM(1)**2+1.)
      SQ(1)=2.*(Q(2)-Q(1))
      SQ(NX)=2.*(Q(NX)-Q(NXM))
      IF(NFORW.EQ.1) THEN
      READ(2,*)(Y(I),I=1,NX)
      ELSE
      READ(2,*)(Y(I),I=NX,1,-1)
      ENDIF
      DO 30 I=1,NT
30    QGIV(I)=ONE
      READ(2,*)(QGIV(I),I=0,NT)
      IF(LBC) THEN
      IF(ABS(Q(NX)-QGIV(0)).GT. .1) THEN
      WRITE(*,100) Q(NX),QGIV(0)
100   FORMAT(' Flowrate',F8.2,' at end must be same as',
     &' 1st Q at t=0',F8.2)
      STOP
      ENDIF
      ELSE
      IF(ABS(Y(NX)-QGIV(0)).GT. .1) THEN
      WRITE(*,101) Y(NX),QGIV(0)
101   FORMAT(' Depth',F8.2,' at end must be same as',
```

```
     &' 1st Y at t=0',F8.2)
      STOP
      ENDIF
      ENDIF
      DO 40 I=1,NT
      IF(QGIV(I).EQ.ONE) QGIV(I)=2.*QGIV(I-1)-QGIV(I-2)
40    CONTINUE
      H=Y(1)+EK*(Q(1)/((B(1)+FM(1)*Y(1))*Y(1)))**2
      CALL IMPLNE
      END
      SUBROUTINE IMPLNE
      LOGICAL LBC
      COMMON /SOLVNE/ A(101,6),V(101),FB(101)
      COMMON Y(51),YB(51),AA(51),Q(51),PP(51),TOP(51),TB(51),
     &V1(51),X(51),SQ(51),B(51),FM(51),SO(51),FN(51),SMS(51),
     &SF(51),WT,WT1,QGIV(0:200),DXG,G,G2,DELX,GDELX,DXDT2,DEL2,
     &DELT,DXT,RDT,H,CMAN,EK,UQ,ERR,NT,NX,NXM,NX2,IOUT,IYOUT,
     &LBC,IFREQ
      KNEXT=0
      DO 50 K=1,NT
      SQB1=SQ(1)
      SQBN=SQ(NX)
      DO 10 I=1,NX
      AA(I)=(B(I)+FM(I)*Y(I))*Y(I)
      V1(I)=Q(I)/AA(I)
      PP(I)=B(I)+SMS(I)*Y(I)
      TOP(I)=B(I)+2.*FM(I)*Y(I)
      TB(I)=TOP(I)
      SF(I)=V1(I)*ABS(V1(I)*(FN(I)/CMAN)**2*(PP(I)/
     &AA(I))**1.3333333)
      YB(I)=Y(I)
      IF(I.EQ.1) THEN
      FB(1)=0.
      FB(2)=WT1*(Q(2)-Q(1))
      ELSE IF(I.EQ.NX) THEN
      FB(2*NX-1)=WT1*(.75*Q(NX)-Q(NXM)+.25*Q(NX2))
      ELSE
      FB(2*I-1)=WT1*FUN(I)-DXDT2*Q(I)
      FB(2*I)=WT1*(Q(I+1)-Q(I-1)-SQ(I))
      ENDIF
10    CONTINUE
      IF(K.EQ.1) WRITE(IOUT,100) K-1,DELT*FLOAT(K-1),(I,Y(I),
     &Q(I),AA(I),V1(I),PP(I),TOP(I),SF(I),I=1,NX)
C  READ IN q* FOR THIS TIME STEP (No. of values,pair of
C     (position & q*)
      IF(K.GE.KNEXT) THEN
      READ(2,*) NSQ,(II,SQ(II),I=1,NSQ)
      READ(2,*,END=12) KNEXT
      GO TO 14
12    KNEXT=NT+10
      ENDIF
14    IF(LBC) THEN
      Q(NX)=QGIV(K)
      ELSE
      Y(NX)=QGIV(K)
      ENDIF
      NCT=0
```

```
15    V(1)=H-Y(1)-EK*V1(1)**2
      V(2)=DXT*(WT1*TB(1)+WT*TOP(1))*(Y(1)-YB(1))
      A(1,3)=2.*EK*V1(1)**2*TOP(1)/AA(1)-1.
      A(1,4)=-2.*EK*V1(1)/AA(1)
      A(1,5)=0.
      A(1,6)=0.
      YY=Y(1)
      Y(1)=1.01*Y(1)
      A(2,3)=(DXT*(WT1*TB(1)+WT*(B(1)+2.*FM(1)*Y(1)))
     &*(Y(1)-YB(1))-V(2))/(Y(1)-YY)
      Y(1)=YY
      V(2)=V(2)+WT*(Q(2)-Q(1)-.5*SQ(1))+FB(2)-.5*WT1*SQB1
      A(2,4)=-WT
      A(2,5)=0.
      A(2,6)=WT
      II=1
      DO 20 I=2,NXM
      II=II+2
      IP=II+1
      V(II)=WT*FUN(I)+DXDT2*Q(I)
      V(IP)=DXDT2*(WT1*TB(I)+WT*TOP(I))*(Y(I)-YB(I))
      A(II,5)=WT*TOP(I)*(G*AA(I)/TOP(I)-V1(I)**2)
      A(II,1)=-A(II,5)
      A(II,6)=2.*V1(I)*WT
      A(II,2)=-A(II,6)
      YY=Y(I)
      AAT=AA(I)
      V1T=V1(I)
      SFT=SF(I)
      PPT=PP(I)
      TOPT=TOP(I)
      Y(I)=1.01*Y(I)
      AA(I)=(B(I)+FM(I)*Y(I))*Y(I)
      V1(I)=Q(I)/AA(I)
      PP(I)=B(I)+SMS(I)*Y(I)
      TOP(I)=B(I)+2.*FM(I)*Y(I)
      SF(I)=V1(I)*ABS(V1(I)*(FN(I)/CMAN)**2*(PP(I)/
     &AA(I))**1.3333333)
      A(II,3)=(WT*FUN(I)+DXDT2*Q(I)-V(II))/(Y(I)-YY)
      A(IP,3)=(DXDT2*(WT1*TB(I)+WT*TOP(I))*(Y(I)-YB(I))
     &-V(IP))/(Y(I)-YY)
      Y(I)=YY
      AA(I)=AAT
      PP(I)=PPT
      TOP(I)=TOPT
      YY=Q(I)
      Q(I)=1.005*Q(I)
      V1(I)=Q(I)/AAT
      SF(I)=V1(I)*ABS(V1(I)*(FN(I)/CMAN)**2*(PPT/
     &AAT)**1.3333333)
      A(II,4)=(WT*FUN(I)+DXDT2*Q(I)-V(II))/(Q(I)-YY)
      Q(I)=YY
      V1(I)=V1T
      SF(I)=SFT
      V(II)=V(II)+FB(II)
```

```
      V(IP)=V(IP)+WT*(Q(I+1)-Q(I-1)-SQ(I))+FB(IP)
      A(IP,1)=0.
      A(IP,2)=-WT
      A(IP,4)=0.
      A(IP,5)=0.
20    A(IP,6)=WT
      IP=IP+1
      A(IP,1)=0.
      A(IP,2)=WT
      IF(LBC) THEN    ! Q(t) GIVEN
      V(IP)=DXT*(WT*TOP(NX)+WT1*TB(NX))*(Y(NX)-YB(NX))
      YY=Y(NX)
      TOPT=TOP(NX)
      Y(NX)=1.01*Y(NX)
      TOP(NX)=B(NX)+2.*FM(NX)*Y(NX)
      A(IP,3)=(DXT*(WT*TOP(NX)+WT1*TB(NX))*(Y(NX)-YB(NX))
     &-V(IP))/(Y(NX)-YY)
      V(IP)=V(IP)+FB(IP)+WT*(.25*Q(NX-2)-Q(NXM)+.75*QGIV(K))
      Y(NX)=YY
      TOP(NX)=TOPT
      ELSE      ! Y(t) GIVEN
      A(IP-1,5)=A(IP-1,6)
      A(IP-2,5)=A(IP-2,6)
      A(IP,3)=.75*WT
      V(IP)=WT*(.75*Q(NX)-Q(NXM)+.25*Q(NX2))+DXT*(WT*TOP(NX)+WT1
     &*TB(NX))*(QGIV(K)-QGIV(K-1))-.5*(WT1*SQBN+WT*SQ(NX))+FB(IP)
      ENDIF
      CALL BAND(IP)
      SUM=0.
      DO 30 I=1,NX
      IF(I.EQ.NX) THEN
      IF(LBC) THEN
      Y(I)=Y(I)-V(IP)
      ELSE
      Q(I)=Q(I)-V(IP)
      ENDIF
      SUM=SUM+ABS(V(IP))
      ELSE
      Y(I)=Y(I)-V(2*I-1)
      SUM=SUM+ABS(V(2*I-1))
      Q(I)=Q(I)-V(2*I)
      SUM=SUM+ABS(V(2*I))
      ENDIF
      AA(I)=(B(I)+FM(I)*Y(I))*Y(I)
      V1(I)=Q(I)/AA(I)
      PP(I)=B(I)+SMS(I)*Y(I)
      TOP(I)=B(I)+2.*FM(I)*Y(I)
30    SF(I)=V1(I)*ABS(V1(I)*(FN(I)/CMAN)**2*(PP(I)/
     &AA(I))**1.3333333)
      NCT=NCT+1
      WRITE(*,330) NCT,SUM
330   FORMAT(' Iteration=',I4,' Residual=',E10.4)
      IF(NCT.LT. 30 .AND. SUM.GT.ERR) GO TO 15
      IF(MOD(K,IFREQ).EQ.0) WRITE(IOUT,100) K,DELT*FLOAT(K),
     &(I,Y(I),Q(I),AA(I),V1(I),PP(I),TOP(I),SF(I),I=1,NX)
50    CONTINUE
```

```
100   FORMAT(/' Solution at K=',I4,' Time=',F8.1,/,1X,
     &' Posit. Depth Flowrate      Area Velocity Perimeter'
     &,' Top Width Slope, Sf',/,1X,76('-'),/,(I4,6F10.3,F10.6))
      END
      FUNCTION FUN(I)
      LOGICAL LBC
      COMMON Y(51),YB(51),AA(51),Q(51),PP(51),TOP(51),
     &TB(51),V1(51),X(51),SQ(51),B(51),FM(51),SO(51),
     &FN(51),SMS(51),SF(51),WT,WT1,QGIV(0:200),DXG,G,G2,
     &DELX,GDELX,DXDT2,DEL2,DELT,DXT,RDT,H,CMAN,EK,
     &UQ,ERR,NT,NX,NXM,NX2,IOUT,IYOUT,LBC,IFREQ
      IF(I.EQ.NX) THEN
      YAV=(Y(I)+Y(I-1))/2.
      DA=(B(I)-B(I-1)+(FM(I)-FM(I-1))*YAV)*YAV/DELX
      ELSE
      DA=(B(I+1)-B(I-1)+(FM(I+1)-FM(I-1))*Y(I))*Y(I)/DEL2
      ENDIF
      IF(IYOUT.EQ.0) THEN
      FQ=0.
      ELSE IF(IYOUT.EQ.1) THEN
      FQ=(V1(I)-UQ)*SQ(I)/DEL2/(G*AA(I))+(B(I)/2.+FM(I)
     &*Y(I)/3.)*(Y(I)/AA(I))**2*(AA(I)-AA(I-1))/DELX
      ELSE
      FQ=V1(I)*SQ(I)/DEL2/(G2*AA(I))
      ENDIF
      FUN=GDELX*(SF(I)-SO(I)+FQ)*AA(I)-V1(I)*SQ(I)
     &-DA*V1(I)**2
      FUN=FUN+2.*V1(I)*(Q(I+1)-Q(I-1))+(G*AA(I)-TOP(I)*
     &V1(I)**2)*(Y(I+1)-Y(I-1))
      RETURN
      END
      SUBROUTINE BAND(N)
      COMMON /SOLVNE/ A(101,6),V(101),FB(101)
      LOGICAL LBC
      COMMON Y(51),YB(51),AA(51),Q(51),PP(51),TOP(51),TB(51), V1
     &(51),X(51),SQ(51),B(51),FM(51),SO(51),FN(51),SMS(51),SF(51),
     &WT,WT1,QGIV(0:200),DXG,G,G2,DELX,GDELX,DXDT2,DEL2, DELT,DXT,
     &RDT,H,CMAN,EK,UQ,ERR,NT,NX,NXM,NX2,IOUT,IYOUT,LBC,IFREQ
      NM=N-1
      FAC=.25*WT/A(NM,2)
      A(N,1)=-FAC*A(NM,3)
      V(N)=V(N)-FAC*V(NM)
      A(N,3)=A(N,3)-FAC*A(NM,5)
      DO 30 K=1,2
      DO 30 I=4-K,N
      IM=I-1
      IF(MOD(I,2).EQ.0) THEN
      K2=0
      K3=K+1
      K4=K3
      K6=6
      KK=K+2
      ELSE
      K3=K
      K6=4
      KK=K+1
```

```
      K4=K+2
      K2=2
      ENDIF
      IF(I.EQ.NM) K6=5
      IF(I.EQ.N) K6=3
      IF(I.EQ.2) K6=4
      FAC=A(I,K3)/A(IM,K4)
      V(I)=V(I)-FAC*V(IM)
      DO 30 J=KK,K6
30    A(I,J)=A(I,J)-FAC*A(IM,J+K2)
      I=NM
      V(N)=V(N)/A(N,3)
      V(I)=(V(I)-V(N)*A(I,5))/A(I,4)
      I=I-1
      V(I)=(V(I)-V(I+1)*A(I,4)-V(I+2)*A(I,5))/A(I,3)
40    I=I-1
      V(I)=(V(I)-V(I+1)*A(I,5)-V(I+2)*A(I,6))/A(I,4)
      I=I-1
      V(I)=(V(I)-V(I+1)*A(I,4)-V(I+2)*A(I,5)-V(I+3)*
      &A(I,6))/A(I,3)
      IF(I.GT.2) GO TO 40
      RETURN
      END
```

The variables names that are read by the above program and what they represent are as follows:

First line of input

IOUT—the logical unit used to write the solution output. When this is given a value of 6 the output will be written to the terminal screen; otherwise the MS-FORTRAN will prompt for the file name.

IFREQ—the frequency at which solution will be written to the output file. If IFREQ = 2 every second time step solution will be written; if 3 every third, etc.

NFORW—if assign a 1 value depths are read in starting from the upstream end toward the downstream end; if 0 then the reverse is the case the depth are given from the downstream toward the upstream end of the channel.

NX—the number of station (i.e. numerical space increments plus 1) along the channel used in the numerical implicit finite difference method.

NT—the number of time steps through which the solution is to be carried. NT*DELT gives the total time of the solution.

DELT—the time increment in seconds to be used in obtaining the numerical solution.

DELX—the space increment to be used in the numerical solution. (NX-1)*DELX gives the length of channel.

G—the acceleration of gravity. If assigned 32.2 ES units will be assumed; if assigned 9.81 SI units will be used.

IBC2—denotes the type of boundary condition to be used at the downstream end of the channel. If IBC2 = 2 then the flowrate is given as a function of time; otherwise the depth is given as a function of time.

IYOUT—describes the lateral inflow/outflow. If IYOUT = 0 then no lateral inflow outflow occurs; if IYOUT = 1 then lateral inflow occurs and UQ is component of velocity in the direction of the channel flow; if IYOUT = 2 then lateral outflow is occurring and SQ is the magnitude thereof.

UQ—component of velocity of the lateral inflow in the direction of the many channel.

EK—the minor loss coefficient at the entrance of the channel.

ERR—the error criteria used in the Newton method in solving the system of simultaneous equations.

The following lines (i.e. series of lines if data does not fit on one line) can be truncated with a / with remaining values takes equal to the last value given.

Second line:
(Q(I),I=1,NX)—flowrates at the NX sections that represent the initial condition.

Third line:
(B(I),I=1,NX)—the bottom width at the NX sections.

Fourth line:
(FM(I),I=1,NX)—side slopes at the NX sections.

Fifth line:
(FN(I),I=1,NX)—values for Mannings n at the NX sections.

Sixth line:
(SO(I),I=1,NX)—slope of the channel bottom at the NX sections.

Seventh line (The remaining lines cannot be truncated with a /):
(Y(I),I=1,NX) or (Y(I),I=NX,1,-1) depending upon NFORW—the initial depths of flow at the NX sections.

Eight line:
(QGIV(I),I=0,NT)—the boundary flowrates, or water depths according to IBC2 for each time step starting with the initial condition t = 0.

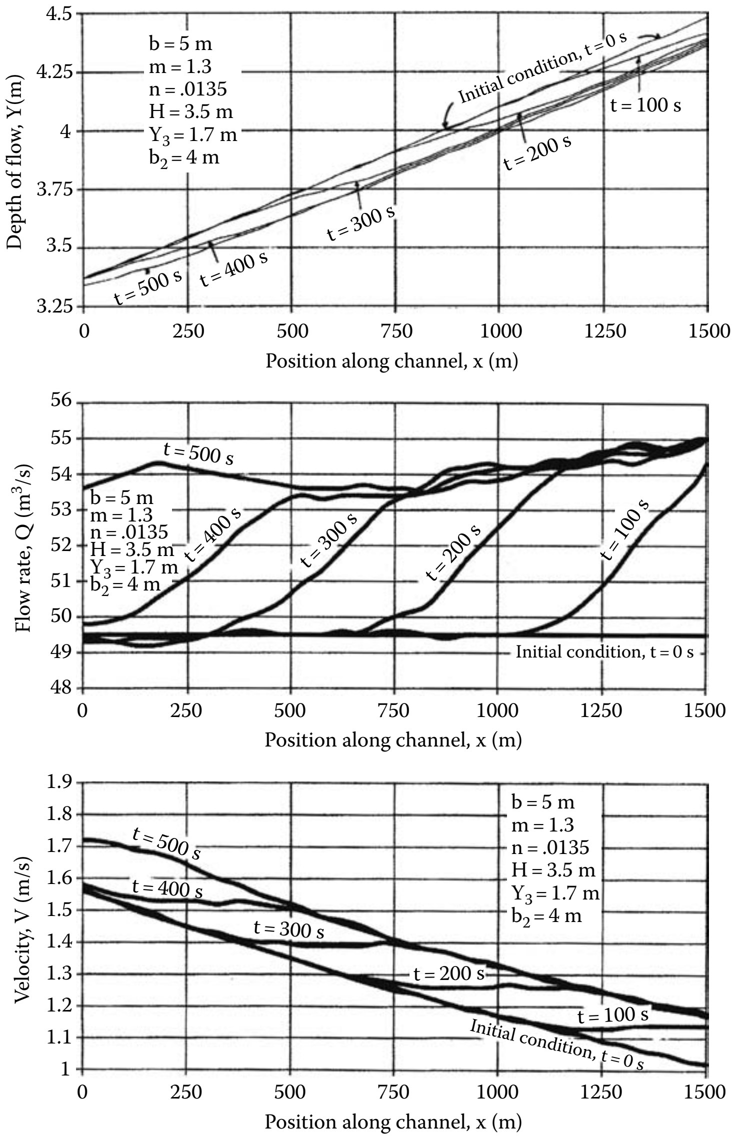
Ninth and tenth lines:—these lines described the lateral inflow or outflow. These values will be read in for the time steps designated as described below, i.e. they may be repeated any number of times up to NT. The first value NSQ indicates how many values of SQ will be read. Following NSQ on the same line are pairs of section number and value for the lateral inflow/outflow SQ(II), i.e. (II,SQ(II),I=1,NSQ). On line ten the next time step KNEXT at which new values for the lateral inflow/outflow will be read.

The input data to the above program to solve the problem consist of the following to carry out the solution for a time of 500 s in steps of 5 s intervals: (The results are printed out every fifth time step or on a 25 s interval.)

Input data to program to solve Example Problem 7.6

```
3 5 0 31 100 5. 50. 9.81 2 0 0. .08 .5
49.49/
5./
1.3/
.0135/
.0008/
 4.4818 4.4433 4.4048 4.3664 4.3280 4.2898 4.2516 4.2134 4.1754 4.1374
 4.0995 4.0618 4.0241 3.9865 3.9490 3.9116 3.8743 3.8372 3.8001 3.7633
 3.7265 3.6899 3.6535 3.6172 3.5811 3.5452 3.5095 3.4739 3.4386 3.4035
   3.3687
49.49 49.5 49.75 50. 50.25 50.5 50.75 51. 51.25 51.5 51.75 52 52.25 52.5
52.75 53. 53.25 53.5 53.75 54. 54.25 54.5 54.75 55. 55./
1 32 0./
110/
```

A graphical presentation of the solution is given in the following three graphs. These graphs have been plotted by using the average between each two interior grid points. If the averages are not used for plotting a saw-tooth-type pattern would be displayed because the effect of incrementally increasing the flow rate at the end of the channel causes this incremental change in flow rate to propagate throughout the channel.



EXAMPLE PROBLEM 7.7

At a distance $L_1 = 950$ m downstream from a reservoir a trapezoidal channel contains a side weir that discharges 1 m^3/s per meter of length, over a 50 m length. Over this lateral outflow length, the channel reduces in size from $b_1 = 5$ m, $m_1 = 1.3$ to $b_2 = 4$ m and $m_2 = 1$. At a distance $L_2 = 500$ m downstream from the end of the side weir a gate exists that controls the flow rate, etc. The depth of flow downstream from the gate under steady-state conditions is 1.7 m. The slope of the entire channel is 0.001 (i.e., $S_{o1} = S_{ot} = S_{o2} = 0.001$), and Manning's roughness coefficients are $n_1 = n_2 = 0.013$. The gate gradually opens, starting at t = 0 so as to increase the flow rate past the gate at a rate of 0.25 m^3/s every 2 s (dQ/dt = 0.125 m_2/s) until the flow rate becomes 32 m^3/s, and then after an additional 4 s the gate is closed so dQ/dt = −0.125 m^2/s until the flow rate is 30 m^3/s and then the flow rate is held constant thereafter for a long time. Determine the depths, flow rates, and velocities throughout the 1500 m long channel for a period of 200 s if the reservoir that supplies the channel has a water depth H = 3.5 m above the channel bottom and the lateral outflow does not change with time. Assume the entrance and the gate minor loss coefficients equal 0.05.

Solution

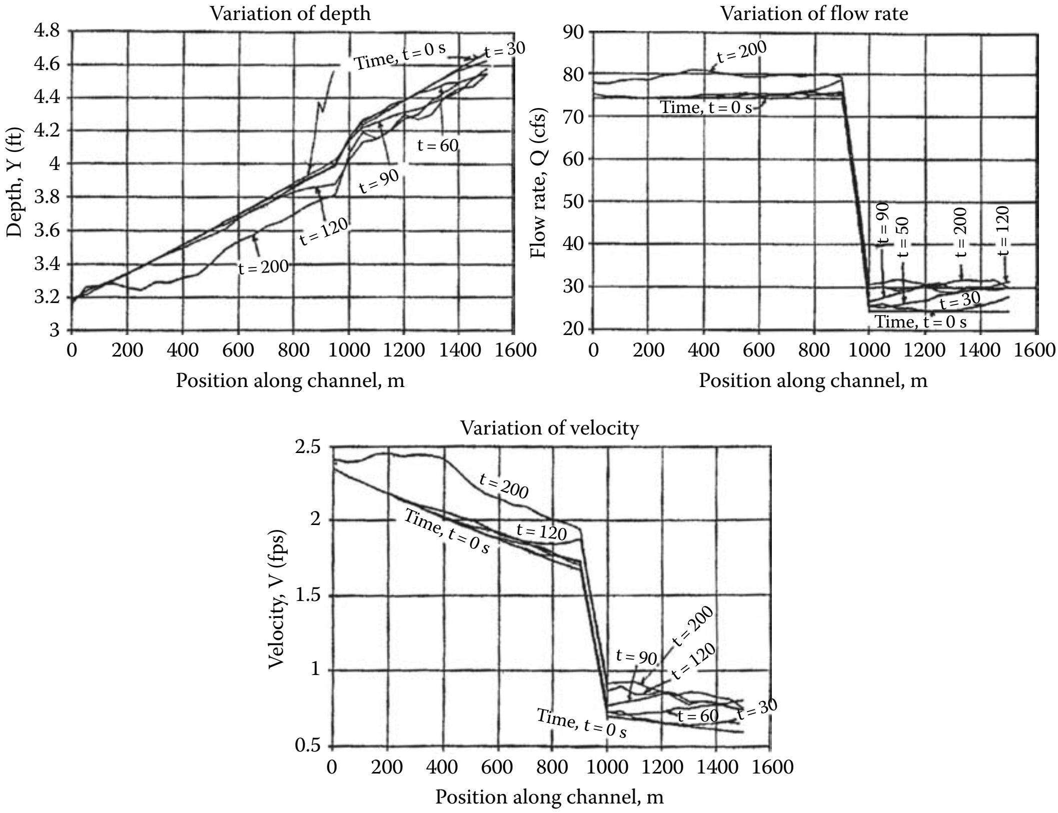
Before the unsteady problem can be solved, it is necessary to solve the steady-state problem, whose solution will be the "initial condition" for the unsteady problem. The solution to this steady-state problem requires that two algebraic energy, one at the entrance of the channel, and the other across the downstream gate, be solved simultaneously with the ODE for the gradually varied flow. The difference between this solution and that of the previous example problems is that the GVF profile solution will need to include both the lateral outflow term, and the term for the nonprismatic channel over the length of the outflow. Following methods described in Chapter 4, the solution gives a flow rate of $Q_1 = 74.28\,m^3/s$ upstream from the side weir, and $Q_2 = 24.28\,m^3/s$ downstream there from, and the depths through the channel are as follows:

x	1500	1450	1400	1350	1300	1250	1200	1150
Y	4.7060	4.6564	4.6068	4.5572	4.5076	4.4580	4.4085	4.3590
	1100	1050	1000	950	900	850	800	750
	4.3095	4.2601	4.2097	4.0497	4.0024	3.9553	3.9084	3.8616
	700	650	600	550	500	450	400	350
	3.8151	3.7688	3.7228	3.6770	3.6315	3.5863	3.5414	3.4969
	300	250	200	150	100	50	0 m	
	3.4528	3.4091	3.3659	3.3232	3.2811	3.2395	3.198	

Using the computer program whose listing is given under the previous example the input data required to solve this unsteady problem is as follows:

```
3 5 0 31 100 2 50. 9.81 2 0 0. .05 .2
74.28 74.28 74.28 74.28 74.28 74.28 74.28 74.28 74.28 74.28 74.28
74.28 74.28 74.28 74.28 74.28 74.28 74.28 74.28 74.28 24.28
24.28 24.28 24.28 24.28 24.28 24.28 24.28 24.28 24.28 24.28
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 4 4 4/
1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5
  1.5 1.5
1. 1. 1./
.013/
.001/
4.7060 4.6564 4.6068 4.5572 4.5076 4.4580 4.4085 4.3590 4.3095 4.2601
4.2097 4.0497 4.0024 3.9553 3.9084 3.8616 3.8151 3.7688 3.7228 3.6770
3.6315 3.5863 3.5414 3.4969 3.4528 3.4091 3.3659 3.3232 3.2811 3.2395
  3.198
24.28 24.35 24.5 24.75 25. 25.25 25.5 25.75 26 26.25 26.5 26.75 27 27.25
27.5 27.75 28 28.25 28.5 28.75 29 29.25 29.5 29.75 30 30.25 30.5 30.75 31
31.25 31.5 31.75 32 32. 31.75 31.5 31.25 31 30.75 30.5 30.25 30 30 30/
1 21 -50./
150/
```

The three graphs given below display the depths, flow rates, and velocities, respectively, throughout the channel for times: t = 0 s, t = 30 s, t = 60 s, t = 100 s, t = 150 s, and t = 200 s.



7.11 WEIGHTING CURRENT AND ADVANCED TIME STEPS DIFFERENTLY

When using the Crank–Nicolson finite differencing method described in the last section all the terms in the differential equation are evaluated by second order differences at a grid point i and at a time k + ½, which is midway between the kth and (k + 1)th time steps. The space derivatives (those with respect to x) are evaluated here by taking the average of second-order approximations at the kth and (k + 1)th time steps, and the time derivatives such as $\partial Y/\partial t \approx (Y_i^{k+1} - Y_i^k)/\Delta t$ is a second-order approximation when evaluated at k + ½ based on a time increment of Δt/2. If the same approximation is used to evaluate the derivative at k or k + 1, then the approximation is first order, or linear instead of quadratic. A disadvantage of the Crank–Nicolson method is that it is unstable, e.g., mathematical oscillations develop that invalidate the solution after many time steps. The more stable method can be developed by weight the advanced time step more than the current time step in evaluating the space derivatives. If the weighting of the current time step is zero then the method is referred to as a fully implicit method, and when the advance time step is weighted with zero then the method is referred to as an explicit method. When a different than equal weights is given to the current and advanced time steps, then the method can no longer be claimed to use second-order finite difference approximations because either the time derivatives cannot be interpreted at second-order approximations, or the interpretation is that different term in the PDE are evaluated at different points in the xt plane. In this section, the finite difference equations will be given that use a weighting δ for the advanced time step k + 1 and (1 − δ) for the current time step. When δ = 1 the finite difference equations are fully implicit; when δ = 0, they are explicit, and when δ = ½ they become the Crank–Nicolson equations.

Using the same approximation for the time derivative approximation of Equation 7.25a as in Equation 7.80 but multiplying the second-order approximation of space derivatives at the (k + 1)th time step by δ and those at the current time step by 1 − δ results in the following equation:

$$F_{i1} = \delta\left\{2V_i^{k+1}\left(Q_{i+1}^{k+1} - Q_{i-1}^{k+1}\right) + \left[T(c^2 - V^2)\right]_i^{k+1}\left(Y_{i+1}^{k+1} - Y_{i-1}^{k+1}\right) - (V^2)_i^{k+1}\left[\left.\frac{\partial A}{\partial x}\right|_{Y,t}\right]_i^{k+1}\right\}$$

$$+\delta\left\{2g\Delta x A_i^{k+1}\left\{S_{fi}^{k+1} - S_0 + F_{qi}^{k+1}\right\} - 2\Delta x\left[Vq*\right]_i^{k+1}\right\}$$

$$+(1-\delta)\left\{2V_i^{k}\left(Q_{i+1}^{k} - Q_{i-1}^{k}\right) + \left[T(c^2 - V^2)\right]_i^{k}\left(Y_{i+1}^{k} - Y_{i-1}^{k}\right) - (V^2)_i^{k}\left[\left.\frac{\partial A}{\partial x}\right|_{Y,t}\right]_i^{k}\right.$$

$$\left. + 2g\Delta x A_i^{k}\left\{S_{fi}^{k} - S_0 + F_{qi}^{k}\right\}\right\}$$

$$-(1-\delta)2\Delta x\left[Vq*\right]_i^{k} + \frac{2\Delta x}{\Delta t}\left(\delta Q_i^{k+1} - (1-\delta)Q_i^{k}\right) = 0 \tag{7.88}$$

Using a similar approximation of Equation 7.24a results in the following finite difference equation:

$$F_{i2} = \delta\left\{Q_{i+1}^{k+1} - Q_{i-1}^{k+1} - 2\Delta x(q*)_i^{k+1}\right\} + \frac{2\Delta x}{\Delta t}\left(\delta T_i^{k+1} + (1-\delta)T_i^{k}\right)\left(\delta Y_i^{k+1} - (1-\delta)Y_i^{k}\right)$$

$$+(1-\delta)\left\{Q_{i+1}^{k} - Q_{i-1}^{k} - 2\Delta x(q*)_i^{k}\right\} = 0 \tag{7.89}$$

You should verify that if δ = ½, then Equations 7.88 and 7.89 become identical to Equations 7.80 and 7.81. The latter equations were actually implement in the computer program listing given in the previous section. Even though assigning δ a value other than ½ result in not having a true second order approximation of the St. Venant equations, because of stability considerations it is recommended that δ be given values from .6 to .75.

EXAMPLE PROBLEM 7.8

A trapezoidal channel with b = 6 m, and m = 1.5 has a bottom slope $S_o = 0.00085$ and n = 0.014 is supplied by a reservoir at its upstream end with a constant water surface elevation 2.8 m above the bottom of the channel. At a distance 5500 m downstream gates control the flow in a rectangular section which is 5 m wide. Initially the gates (with $C_c = 0.6$) are at a height $Y_G = 1.0$ m above the channel bottom. The entrance loss coefficient is $K_e = 0.05$. It is desired to increase the flow rate to 180 m³/s in 200 s as follows:

Time (s)	40	80	120	160	200
Flow rate (m³/s)	65	110	130	175	180

Thereafter the flow rate is to be decreased linearly to 50 m³/s at time t = 800 s, and held constant thereafter. Determine depths, flow rates, and velocities throughout the channel and examine what gate opening might achieve the desired results, etc. There are two gates at the downstream end of the channel each 2.5 m wide.

Solution

The normal depth and flow rate are: $Y_o = 2.548\,m$ and $Q_o = 72.2\,m^3/s$, but with the gates set at 1 m above the channel a solution of the steady-state problem gives $Q = 34.82\,m^3/s$, and the depth varying from 7.301 m just upstream from the gates to 2.71 m at the entrance as given in the input data to the program below.

Input data to solve problem using previous program IMPLICAL.

```
3 1 0 45 80 20. 125. 9.81 2 0 0. .05 .0001 .75
34.82/
6/
1.5/
.014/
.00085/
   7.301 7.195 7.089 6.982 6.876 6.770 6.664 6.558 6.452
   6.346 6.240 6.134 6.028 5.922 5.816 5.711 5.605 5.499
   5.393 5.288 5.182 5.076 4.971 4.865 4.760 4.655 4.550
   4.445 4.340 4.235 4.130 4.026 3.922 3.818 3.714 3.611
   3.508 3.406 3.304 3.202 3.102 3.003 2.904 2.806 2.710/
34.82 49.91 65 87.5 110 130 150 162.5 175 177.5 180 175.67 171.33 167
162.67 158.34 154 149.67 145.33 141 136.67 132.34 128 123.67 119.33
115 110.67 106.34 102 97.67 93.33 89 84.67 80.34 76 71.67 67.33 63 58.67
54.34 50 50/
1 5 0./
100/
```

Plotting every fifth time step solution, or at 100 s intervals the three following graphs show the change in depth, flow rate, and velocity throughout the channel. Note from these graphs that condition remain unchanged at the upstream end of the channel for approximately 1000 s. Thereafter the constant head reservoir that supplies the water causes the flow rate and velocity to increase at the channel entrance and the depth to decrease in agreement with the upstream governing energy equation. Up to this time, the additional volume of water from the increased flow rates have come from channel storage. At the end of the simulation, which is to 1600 s the reservoir is supplying $58.3\,m^3/s$ whereas $50\,m^3/s$ is being withdrawn so that some of this depleted channel storage is now being replenished. A fourth graph plots the specified flow rate and the resulting change in depth at the downstream end of the channel. Note as the flow rate is increased that the depth is decreased, but after 800 s when the flow rate is held constant at $50\,m^3/s$ that the depth continues to decrease from 6.82 to 6.64 m at 1600 s.

The conditions at the downstream gate are the result what occurred by using a downstream boundary condition that specified the flow rate as a function of time, and satisfying the numerical approximation of the continuity equation, but does not reflect a given operation of the gates. This results in the specific energy (and velocity) at the downstream end of the channel varying as shown in the fifth graph below. Note that E_2 decreases from 7.31 to 6.35 m in 200 s, and thereafter it increases to 6.83 m at 800 s and decreased again thereafter to 6.65 m at 1600 s. The manner in which E_2 varies with the specified flow rate Q_2 is shown on the sixth graph below. As the flow rate is being increased the specific energy is decreased, and when the flow rate is decreased the specific energy again increases, but decreases again when the flow rate is held constant.

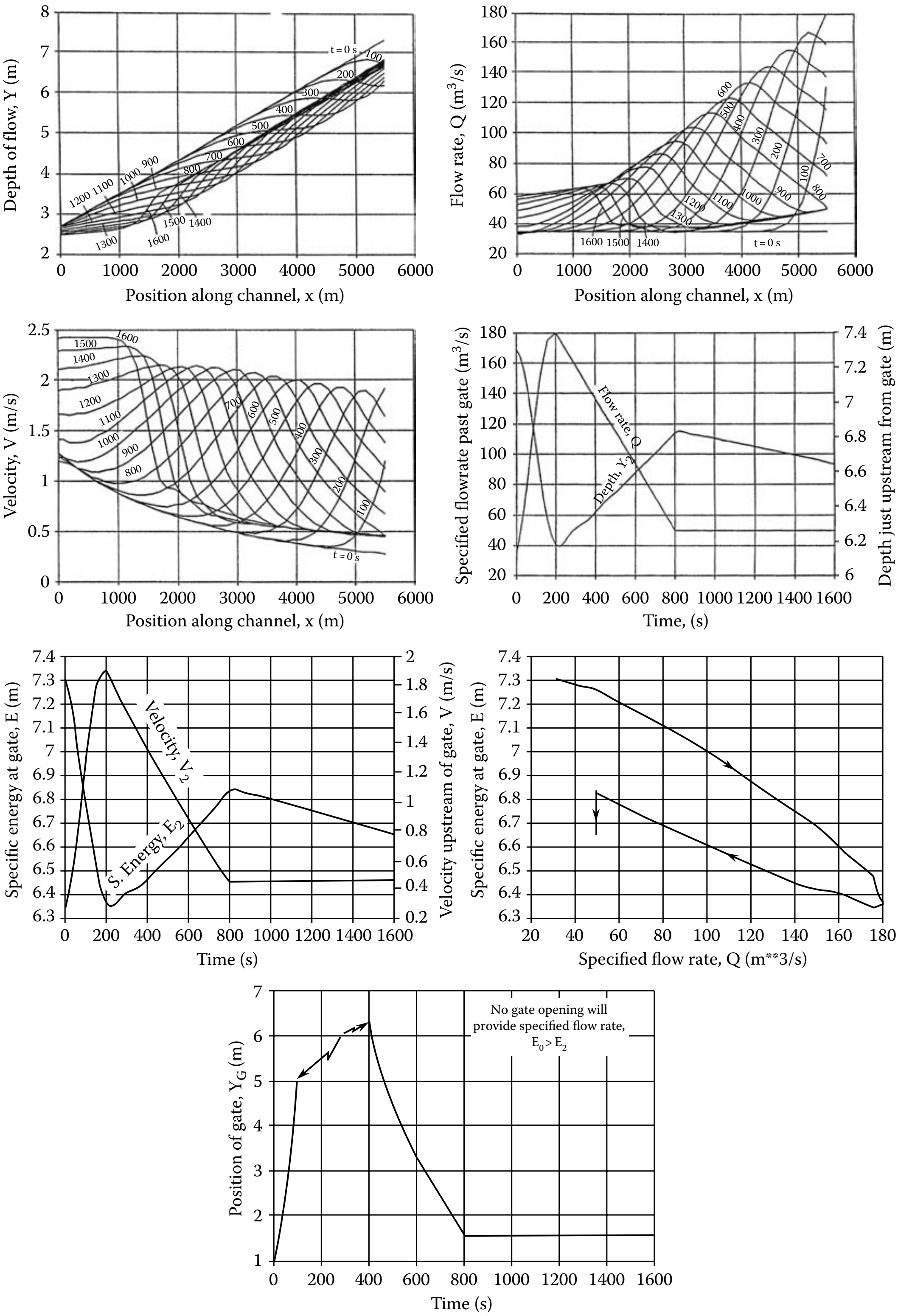
To achieve these given downstream conditions the position of the gates, if both were moved the same, would need to be changed with time as shown in the seventh graph below. Note from this graph that after 100 s until 400 s that it is not possible to achieve the specified flow rates since the critical specific energy E_c at the 5 m wide gate section is larger than the specific energy E_2 at the end of the channel. To achieve these flow rates the channel at the gate section would need to be wider.

In other words, the specified downstream boundary condition does not reflect specified changes in the positions of gates. In fact, with two gates set at different positions, it is possible to keep the specific energy constant, over limited ranges, with flow rates changed. To illustrate this, the table below examines what position the second gate should have if the position of the first gate is specified to give the flow rates from the above simulation, but keeping the specific energy constant at the initial 7.305 m by solving the three equations: $Q_1 = Q_2 + Q_3$; $E_2 = Y_3 + (Q_2/A_3)/(2g) = 7.305$ and $E_2 = Y_4 + (Q_4/A_4)/(2g) = 7.305$ (You should obtain the unsteady solution and carry out the needed computations to verify these values to fully understand the complexities that might arise with positioning two or more gate in controlling channel flows.) A time-dependent solution of flow in a channel whose gates have their positions changed according to a specified time schedule needs to have a more complex downstream boundary condition than built into program IMPLICAL.

Q_1	Y_3 (Given)	Y_4 (Computed)	Q_2	Q_3	Y_{G1}	Y_{G2}
34.82	0.6075	0.6075	17.41	17.41	1.013	1.013
65.0	0.6075	1.186	17.41	47.59	1.013	1.977
87.5	0.6075	1.838	17.41	70.09	1.013	3.063
110.0	0.6075	No solution	17.41		1.013	
87.5	1.000	2.445	27.81	59.69	1.667	4.083
110.0	1.000	4.239	27.81	82.19	1.667	7.065
130.0	1.000	No solution	27.81		1.667	
110.0	1.200	5.919	32.83	77.17	2.000	9.865
130.0	1.200	No solution	32.83		2.000	
110.0	2.000	2.407	51.01	58.99	3.333	4.012
130.0	2.000	3.802	51.01	78.99	3.333	6.337
150.0	2.000	No solution	51.01		3.333	
150.0	3.000	4.071	68.93	81.07	5.000	6.785

Table giving flow variables immediately upstream of gates as a function of time.

Time (s)	Depth Y_2 (m)	Flow Rate Q_2 (m³/s)	Velocity V_2 (m/s)	S. Energy E_2 (m)	D. Depth Y_3 (m)	Gate P. Y_G (m)	Critical E_{c3} (m)
0.0	7.301	34.820	0.281	7.305	0.608	1.013	2.555
20.0	7.260	49.910	0.407	7.268	0.893	1.488	3.248
40.0	7.178	65.000	0.540	7.193	1.199	1.998	3.874
60.0	7.055	87.500	0.748	7.084	1.703	2.839	4.723
80.0	6.904	110.000	0.974	6.952	2.304	3.840	5.502
100.0	6.745	130.000	1.196	6.818	3.007	5.011	6.150
120.0	6.585	150.000	1.435	6.690	No solution		6.765
.					. .		
380.0	6.343	141.000	1.433	6.448	No solution		6.492
400.0	6.368	136.670	1.380	6.465	3.735	6.224	6.358
.							
1560.0	6.654	50.000	0.470	6.665	0.944	1.573	3.252
1580.0	6.648	50.000	0.471	6.659	0.944	1.574	3.252
1600.0	6.643	50.000	0.472	6.654	0.945	1.575	3.252


Depth of flow, Y (m)
Position along channel, x (m)
t = 0 s
Flow rate, Q (m^3/s)
Position along channel, x (m)
Velocity, V (m/s)
Position along channel, x (m)
Specified flowrate past gate (m^3/s)
Depth just upstream from gate (m)
Flow rate, Q
Depth, Y_2
Time, (s)
Specific energy at gate, E (m)
Velocity upstream of gate, V (m/s)
Velocity, V_2
S. Energy, E_2
Time (s)
Specific energy at gate, E (m)
Specified flow rate, Q (m**3/s)
Position of gate, Y_G (m)
No gate opening will provide specified flow rate, $E_0 > E_2$
Time (s)

7.12 THE PREISSMANN IMPLICIT METHOD

In this section the Preissmann (Preissmann 1960, Preissmann and Cunge 1961) (or SOGREAH) implicit method will be discussed. Instead of evaluating space derivatives at one of the grid points as is done in the Crank–Nicolson method, the Preissmann method evaluates these derivatives midway between two consecutive grid points. In other words second-order differences with respect to x are evaluated by $\partial f/\partial x \approx (f_{i+1} - f_i)/\Delta x$ in which the derivative is being evaluated at i + ½ based on an increment of $\Delta x/2$. Another difference is that instead of solving for the variables of the problem, such as the depth Y and the flow rate Q, the Preissmann method solves for the difference in these variables between the current time step k and the new time step k + 1. As in the previous section a weighting factor δ for the advanced time step k + 1 and the current time step k is used so that variables, and their derivatives with respect to x and t are approximated by the following:

$$f(x,t) = \frac{1}{2}\left\{\delta\left(f_{i+1}^{k+1} + f_i^{k+1}\right) + (1-\delta)\left(f_{i+1}^{k} + f_i^{k}\right)\right\}$$

$$\frac{\partial f}{\partial x} = \delta\frac{f_{i+1}^{k+1} - f_i^{k+1}}{\Delta x} + (1-\delta)\frac{f_{i+1}^{k} - f_i^{k}}{\Delta x}$$

$$\frac{\partial f}{\partial t} = \frac{f_{i+1}^{k+1} - f_{i+1}^{k} + f_i^{k+1} - f_i^{k}}{2\Delta t}$$

Define the differences with respect to time by $\Delta f = f^{k+1} - f^k$ and dropping the superscript so that $f = f^k$, then the above three equations can be written as

$$f(x,t) \approx \frac{1}{2}\left\{\delta\left(\Delta f_{i+1} + \Delta f_i\right) + f_{i+1} + f_i\right\}$$

$$\frac{\partial f}{\partial x} \approx \delta\frac{\Delta f_{i+1} - \Delta f_i}{\Delta x} + \frac{f_{i+1}^{k} - f_i^{k}}{\Delta x}$$

$$\frac{\partial f}{\partial x} \approx \frac{\Delta f_{i+1} + \Delta f_i}{2\Delta t}$$

Using these approximations Equation 7.61 is discretized by the following equation:

$$\frac{2}{\delta(\Delta T_{i+1} + \Delta T_i) + T_{i+1} + T_i}\left\{\delta\frac{\Delta Q_{i+1} - \Delta Q_i}{\Delta x} + \frac{Q_{i+1} - Q_i}{\Delta x} - q^*\right\} + \frac{\Delta Y_{i+1} - \Delta Y_i}{2\Delta t} = 0 \tag{7.90}$$

In the remainder of this development, it will be assumed that no lateral inflow or outflow occurs so $q^* = 0$. To linearize Equation 7.90 the denominator of the first term will be rewritten in the form of $(1 + \varepsilon)^{-1}$ and then applying the binomial formula $(1 + x)^n = 1 + nx + n(n - 1)x^2/2! + n(n - 1)(n - 2)x^3/3! + \cdots$ results in the following using only the first term in the binomial formula since ε is small.

$$\frac{2}{(T_{i+1} + T_i)\left\{1 + \delta(\Delta T_{i+1} + \Delta T_i)/(T_{i+1} + T_i)\right\}} = \frac{2\left\{1 - \delta(\Delta T_{i+1} + \Delta T_i)/(T_{i+1} + T_i)\right\}}{T_{i+1} + T_i}$$

substituting this result in Equation 7.90 and dropping small terms of $\Delta f \Delta g \approx 0$ gives the following equation:

$$\frac{2}{T_{i+1} - T_i}\left\{\delta\frac{\Delta Q_{i+1} - \Delta Q_i}{\Delta x} + \frac{Q_{i+1} - Q_i}{\Delta x}\right\} - \frac{2\delta\left(\Delta T_{i+1} + \Delta T_i\right)}{\left(T_{i+1} + T_i\right)^2}\frac{Q_{i+1} - Q_i}{\Delta x} + \frac{\Delta Y_{i+1} + \Delta Y_i}{2\Delta t} = 0$$

ΔT can be evaluated by $\Delta T = (dT/dY)\Delta Y$, and then this equation becomes

$$H_i\Delta Y_{i+1} + B_i\Delta Q_{i+1} - C_i\Delta Y_i - B_i\Delta Q_i \tag{7.91}$$

in which

$$H_i = 1 - \frac{4\delta\Delta t}{\Delta x}\frac{(Q_{i+1} - Q_i)}{(T_{i+1} + T_i)^2}\left(\frac{dT}{dY}\right)_{i+1}, \quad B_i = \frac{4\delta\Delta t}{\Delta x(T_{i+1} + T_i)} = -D_i$$

$$C_i = \frac{4\delta\Delta t}{\Delta x}\frac{(Q_{i+1} - Q_i)}{(T_{i+1} + T_i)^2}\left(\frac{dT}{dY}\right)_i - 1, \quad G_i = \frac{4\Delta t}{\Delta x}\frac{(Q_{i+1} - Q_i)}{(T_{i+1} + T_i)}$$

If $D_i = -B_i$ then Equation 7.91 can be written as

$$H_i\Delta Y_{i+1} + B_i\Delta Q_{i+1} - C_i\Delta Y_i - D_i\Delta Q_i = G_i \tag{7.91a}$$

The equation of motion Equation 7.62 will be rearranged to be in accordance with the way the Preissman method was developed. The terms F_q and $Qq^*/(gA^2)$ associated with lateral inflow (outflow) and the term for non prismatic channels $(Q/A)^2\partial A/\partial x|_{Y,t}$ will be dropped. Then upon multiplying Equation 7.62 by gA it becomes

$$\frac{2Q}{A}\frac{\partial Q}{\partial x} + \left(gA - \frac{Q^2T}{A^2}\right)\frac{\partial Y}{\partial x} + gA(S_f - S_o) + \frac{\partial Q}{\partial t} = 0$$

Writing the first term twice in place of the 2 multiplier and replacing $\partial Q/\partial x = -T\partial Y/\partial t$ from the continuity equation in one of these terms, replacing $S_f = Q|Q|/K^2$ (where $K = CA^{5/3}/(nP^{2/3})$ is the conveyance), introducing the kinetic energy correction coefficient α, and noting that $T\partial Y/\partial x = \partial A/\partial x$ results in the following equation:

$$\frac{\alpha Q}{A}\frac{\partial Q}{\partial x} - \frac{Q^2T}{A^2}\frac{\partial Y}{\partial t} + gA\frac{\partial Y}{\partial x} - \frac{\alpha Q^2}{A^2}\frac{\partial A}{\partial x} + \frac{gAQ|Q|}{K^2} + \frac{Q^2}{2A}\frac{\partial \alpha}{\partial x} - gAS_0 + \frac{\partial Q}{\partial t} = 0 \tag{7.92}$$

Equation 7.92 can be discretized and linearized following the same process as was used for the continuity equation. In the linearization process it should be noted that for terms involving the reciprocal of a variable plus its increment, such as $(A_i + \Delta A_i)^{-1} = 1/\{A_i(1 + \Delta A_i/A_i)\} = (1 - \Delta A_i/A_i)/A_i$. After considerable algebra the following equation results:

$$H_i'\Delta Y_{i+1} + B_i'\Delta Q_{i+1} - C_i'\Delta Y_i - D_i'\Delta Q_i = G_i' \tag{7.93}$$

in which

$$H_i' = -\frac{1}{2}\left(\frac{Q_{i+1}T_{i+1}}{A_{i+1}} + \frac{Q_iT_i}{A_i}\right) + \frac{\Delta t\delta}{\Delta x}\left\{(Q_{i+1} - Q_i)\left\{\frac{Q_{i+1}}{A_{i+1}}\frac{d\alpha_{i+1}}{dY_{i+1}} - \frac{Q_{i+1}T_{i+1}\alpha_{i+1}}{A_{i+1}^2}\right\} - T_{i+1}\left(\frac{\alpha_{i+1}Q_{i+1}}{A_{i+1}^2} - \frac{\alpha_iQ_i}{A_i^2}\right)\right\}$$

$$-\frac{\Delta t\delta}{\Delta x}\left\{\frac{Q_{i+1}^2}{A_{i+1}^2}(A_{i+1} - A_i)\left(\frac{d\alpha_{i+1}}{dY_{i+1}} - \frac{2\alpha_{i+1}T_{i+1}}{A_{i+1}}\right) + gT_{i+1}(Y_{i+1} - Y_i) - \frac{Q_{i+1}T_{i+1}}{2A_{i+1}^2}(\alpha_{i+1} - \alpha_i)\right.$$

$$\left. + g(A_{i+1} + A_i)\right\}$$

$$+\frac{\Delta t\delta}{\Delta x}\left\{\frac{1}{2}\left(\frac{Q_{i+1}^2}{A_{i+1}} + \frac{Q_i^2}{A_i}\right)\frac{d\alpha_{i+1}}{dY_{i+1}}\right\} + g\delta\Delta t\frac{Q_{i+1}|Q_{i+1}|}{K_{i+1}^2}\left(T_{i+1} - \frac{2A_{i+1}}{K_{i+1}}\frac{dK_{i+1}}{dY_{i+1}}\right)$$

$$B'_i = 1 + \frac{\delta\Delta t}{\Delta x}\left\{\frac{\alpha_{i+1}Q_{i+1}}{A_{i+1}} + \frac{\alpha_i Q_i}{A_i} + (Q_{i+1} - Q_i)\frac{\alpha_{i+1}}{A_{i+1}} - (A_{i+1} - A_i)\frac{2\alpha_{i+1}Q_{i+1}^2}{A_{i+1}^2} + (\alpha_{i+1} - \alpha_i)\frac{Q_{i+1}}{A_{i+1}}\right\}$$
$$+ 2g\delta\Delta t\frac{A_{i+1}|Q_{i+1}|}{K_{i+1}^2}$$

$$C'_i = \frac{1}{2}\left(\frac{Q_{i+1}T_{i+1}}{A_{i+1}} + \frac{Q_i T_i}{A_i}\right) - \frac{\delta\Delta t}{\Delta x}\left\{(Q_{i+1} - Q_i)\left(\frac{Q_i}{A_i}\frac{d\alpha_i}{dY_i} - \frac{\alpha_i Q_i T_i}{A_i^2}\right) + T_i\left(\frac{\alpha_{i+1}Q_{i+1}^2}{A_{i+1}^2} + \frac{\alpha_i Q_i^2}{A_i^2}\right)\right\}$$
$$+ \frac{\delta\Delta t}{\Delta x}\left\{-\frac{Q_i^2}{A_i^2}(A_{i+1} - A_i)\left(\frac{d\alpha_i}{dY_i} - \frac{2\alpha_i T_i}{A_i}\right) + gT_i(Y_{i+1} - Y_i) - \frac{Q_i^2}{A_i^2}(\alpha_{i+1} - \alpha_i) - g(A_{i+1} + A_i)\right.$$
$$\left. - \frac{1}{2}\left(\frac{Q_{i+1}^2}{A_{i+1}} + \frac{Q_i^2}{A_i}\right)\frac{d\alpha_i}{dY_i}\right\} - g\delta\Delta t\frac{Q_i|Q_i|}{K_i^2}\left(T_i - \frac{2A_i}{K_i}\frac{dK_i}{dY_i}\right)$$

$$D'_i = -1 - \frac{\delta\Delta t}{\Delta x}\left\{-\frac{\alpha_{i+1}Q_{i+1}}{A_{i+1}} - \frac{\alpha_i Q_i}{A_i} + (Q_{i+1} - Q_i)\frac{\alpha_i}{A_i} - (A_{i+1} - A_i)\frac{2\alpha_i Q_i^2}{A_i^2} + (\alpha_{i+1} - \alpha_i)\frac{Q_i}{A_i}\right\}$$
$$- 2g\delta\Delta t\frac{A_i|Q_i|}{K_i^2}$$

$$G'_i = -\frac{\Delta t}{\Delta x}\left\{(Q_{i+1} - Q_i)\left(\frac{\alpha_{i+1}Q_{i+1}}{A_{i+1}} + \frac{\alpha_i Q_i}{A_i}\right) - (A_{i+1} - A_i)\left(\frac{\alpha_{i+1}Q_{i+1}^2}{A_{i+1}^2} + \frac{\alpha_i Q_i^2}{A_i^2}\right) + g(Y_{i+1} - Y_i)(A_{i+1} + A_i)\right\}$$
$$- \frac{\Delta t}{\Delta x}\left(\frac{Q_{i+1}^2}{2A_{i+1}} + \frac{Q_i^2}{2A_i}(\alpha_{i+1} + \alpha_i)\right) - g\Delta t\left(\frac{Q_{i+1}|Q_{i+1}|A_{i+1}}{K_{i+1}^2} + \frac{Q_i|Q_i|A_i}{K_i^2}\right) + g\Delta t S_0(A_{i+1} + A_i)$$

The coefficient H'_i, B'_i, C'_i, D'_i and G'_i can be computed from information known at the kth time step since values of Y_i, Q_i, Y_{i+1}, Q_{i+1} and the geometry, etc. of the channel are known. Equations 7.91 and 7.93 have the differences in depth and flow rates between the kth and the (k + 1)th time step, ΔY and ΔQ at the grid points i and i + 1 as the unknowns. The number of equations available at regular points is 2(n − 1), where n is the number of grid points along the channel. When appropriate boundary condition equations (2 in number, one for each end) are added to these equations the number of equations will equal the number of unknowns, i.e., 2n independent equations will occurs to solve for the 2n unknowns. Once these unknowns, $\Delta Y = Y^{k+1} - Y^k$ and $\Delta Q = Q^{k+1} - Q^k$, are obtained then the next time step values for depth and flow rate are obtained from $Y^{k+1} = Y^k + \Delta Y$ and $Q^{k+1} = Q^k + \Delta Q$, respectively, and the process is repeated again for the next advanced time step, etc. until the end of the period of simulation has been reached.

A question to comtemplate is how much accuracy is lost in using only the first term of the binomial expansion in linearizing the difference equations of the St. Venant equations. The answer of course depends upon how large the changes, i.e., the Δ's are compared with the variables. The table below compares how close the result of $1 - \epsilon$ (which is the linearized approximation value) compares with the true value of $(1 + \epsilon)^{-1}$, using 6 digits of accuracy. Notice that if the change is 1% or less that this linearization result in essentially 4 digits or better.

ϵ	$(1+\epsilon)^{-1}$	$1-\epsilon$	Difference
0.1	0.909091	0.900000	0.009091
0.05	0.952381	0.950000	0.002381
0.01	0.990099	0.990000	0.000099
0.005	0.995025	0.995000	0.000025
0.001	0.999001	0.999000	0.000001

7.12.1 Double Sweep Method of Solution

Since the resulting system of equations, for the changes in depth and flow rate ΔY and ΔQ is linear, the linear algebra methods used previously in the "direct implicit finite difference" section, that define a banded coefficient matrix, and known vector, could be used to obtain the solution vector. An alternative is to use what has become known in the literature as the **double sweep** solution method. Rather than viewing the problem as a linear algebra problem involving a coefficient matrix that is narrowly banded the double sweep method manipulates Equations 7.91 and 7.93 into a form that first allows ΔY and then ΔQ to be solved at each point, i, and then proceeds to the next point and repeats this process until values have been computed at all points. To accomplish this values of L_i, M_i and N_i are needed in the following equation:

$$\Delta Y_i = L_i \Delta Y_{i+1} + M_i \Delta Q_{i+1} + N_i \tag{7.94}$$

and E_i, and F_i in the following equation:

$$\Delta Q_i = E_i \Delta Y_i + F_i \tag{7.95}$$

so that Equation 7.94 can first be solved for ΔY_i and then Equation 7.95 can be solved for ΔQ_i. Equations 7.94 and 7.95 have been arranged assuming the control is downstream so that as the solution progresses to succeeding points ΔY_{i+1} and ΔQ_{i+1} are known starting with the downstream boundary values. Means for getting these beginning values from known boundary condition is described later. At point i + 1 Equation 7.95 becomes

$$\Delta Q_{i+1} = E_{i+1} \Delta Y_{i+1} + F_{i+1} \tag{7.95a}$$

The same definitions, etc. as used with Equation 7.91 above can be applied to Equation 7.93, except that E_i and F_i will be primed, or will be E_i' and F_i'.

To find the coefficients in Equations 7.94 and 7.95, first substitute Equation 7.95 into both Equations 7.91 and 7.93 and then solve for ΔY_i. This algebra produces

$$\Delta Y_i = \frac{H_i \Delta Y_{i+1}}{C_i + D_i E_i} + \frac{B_i \Delta Q_{i+1}}{C_i + D_i E_i} - \frac{G_i D_i F_i}{C_i + D_i E_i} \tag{7.96}$$

and from the same process, except using the primed definitions in Equation 7.93

$$\Delta Y_i = \frac{H_i' \Delta Y_{i+1}}{C_i' + D_i' E_i'} + \frac{B_i' \Delta Q_{i+1}}{C_i' + D_i' E_i'} - \frac{G_i' D_i' F_i'}{C_i' + D_i' E_i'} \tag{7.97}$$

Note that Equation 7.96 supplies the coefficients for Equation 7.94, namely,

$$L_i = \frac{H_i}{C_i - D_i E_i}, \quad M_i = \frac{B_i}{C_i - D_i E_i} \quad \text{and} \quad N_i = \frac{(G_i + D_i F_i)}{C_i + D_i E_i}$$

And likewise Equation 7.97 supplies the coefficients to the equivalent of Equation 7.94 except with primed quantities, L_i', M_i' and N_i'.

Next, equate the two ΔY_i's from Equations 7.96 and 7.97 and solve for ΔQ_{i+1}. This algebra provides terms with a least common denominator $(C_i + D_iE_i)(C_i' + D_i'E_i')$, that is identical for all terms. Multiplying this out of the equation, and then solving for ΔQ_{i+1} produces

$$\Delta Q_{i+1} = \frac{H_i(C_i' + D_i'E_i') - H_i'(C_i + D_iE_i)}{B_i'(C_i + D_iE_i) - B_i(C_i' + D_i'E_i')}\Delta Y_{i+1} + \frac{(G_i' + D_i'F_i')(C_i + D_iE_i) - (G_i + D_iF_i)(C_i' + D_i'E_i')}{B_i'(C_i + D_iE_i) - B_i(C_i' + D_i'E_i')}$$

This algebra has produces the coefficients E_{i+1} and F_{i+1} in Equation 7.95a (and by decrementing i by one the coefficients in Equation 7.95) so that

$$E_{i+1} = \frac{H_i\left(C_i' + D_i'E_i'\right) - H_i'(C_i + D_iE_i)}{B_i'(C_i + D_iE_i) - B_i(C_i' + D_i'E_i')} \quad \text{and} \quad F_{i+1} = \frac{(G_i' + D_i'F_i')(C_i + D_iE_i) - (G_i + D_iF_i)(C_i' + D_i'E_i')}{B_i'(C_i + D_iE_i) - B_i(C_i' + D_i'E_i')}$$

7.12.1.1 Boundary Conditions

The boundary conditions provide the means to start and end the recursive use of Equations 7.94 and 7.95. If the variable is not given directly by the boundary condition, then to be consistent with the double sweep method it must be expressed as a linear equation of the form of Equation 7.94 or Equation 7.95 that provides the change in either the depth Y, or the change in flow rate Q between the current time step k and the new time step k + 1 for which the solution is sought. The boundary conditions dealt with earlier, namely, (1) Y = given, (2) Q = given, (3) stage discharge relationship, and (4) V (velocity) = given, will be handled in this order, first for the downstream end of the channel, and then for the beginning of the channel. This treatment of the boundary conditions will be based on starting the recursive computations given by Equations 7.94 and 7.95 at the point n − 1, just upstream from the end of the channel. Thus ΔY_n and ΔQ_n must be obtained first.

7.12.1.2 Downstream Boundary Conditions

Case (1)—Depth Y(L, t) is given: With the depth Y^{+1} known the change in depth is given by $\Delta Y_n = Y_n^{k+1} - Y_n^k$. With ΔY_n now known, Equation 7.95 can be used to compute the change in flow rate over the time increment Δt, or

$$\Delta Q_n = E_n\Delta Y_n + F_n$$

Case (2)—Flow rate Q(L, t) is given: With the flow rate Q_n^{k+1} known the change in flow rate at the end of the channel is given by, $\Delta Q_n = Q_n^{k+1} - Q_n^k$. With ΔQ_n now known Equation 7.95 can be used to compute the change in depth over the time increment Δt, or

$$\Delta Y_n = \frac{\Delta Q_n - F_n}{E_n}$$

Case (3)—Stage Discharge relationship given or Q_s(L, t) = f[Y_s(L, t)]: A tabular stage discharge relationship generally gives flow rates related to specified depths. From this data the change in flow rate between any two consecutive entries of depth is given by $(dQ/dY)_s = (Q_{sj+1} - Q_{sj})/(Y_{sj+1} - Y_{sj})$, in which the subscript s denotes this tabular stage discharge data, and j is the entry number in the table. Using only through the first derivative term in Taylor's series $Q_n^{k+1} - Q_n^k + \Delta t(dQ/dt) = Q_{sj} + (dQ/dY)_s(\Delta Y/\Delta t)\Delta t = Q_{sj} + (dQ/dY)_s\Delta Y$, in which j is selected so Y_{sj}

and Y_{sj+1} bracket Y if data in the table allows this. From Equation 7.95 $Q_n^{k+1} = Q_n^k + E_n\Delta Y + F_n$. Equating these two Q_n^{k+1}'s and solving for ΔY gives

$$\Delta Y_n = (Qs_j - F_n - Q_n^k)\left(E_n - \left(\frac{dQ}{dY}\right)_s \right)$$

Generally the values Q_{sj} will be the entry in the stage discharge table of values corresponding to Y_{sj}, which is smaller than Y_n^k so that the entry Y_{sj+1} is larger than Y_n^k. The exception is if extrapolation is needed because Y_n^k is smaller than the first given depth Y_{s1}. If desired, then a quadratic approximation for $(dQ/dY)_s$ could be used by using three consecutive entries from the tabular stage discharge relationship. Next ΔQ_n can be computed from Equation 7.95, or

$$\Delta Q_n = E_n\Delta Y_n + F_n$$

Case (4)—Velocity V(L, t) is given: From the specified known velocity at the downstream end of the channel, the flow rate can be obtained from $Q_n^{k+1} = V_n^{k+1}A_n^{k+1}$. The area A_n^{k+1} can be obtained from $A_n^{k+1} = A_n^k + (dA/dY)\Delta Y$, in which dA/dY is the top width T, or for a trapezoidal channel dA/dY = b + 2mY. Evaluating dA/dY from the current time step, k, and getting $Q_n^{k+1} = Q_n^k + E_n\Delta Y_n + F_n$ from Equation 95 gives

$$V_n^{k-1}\left(A_n^k + T_n^k\Delta Y_n\right) = \left(Q_n^k + E_n\Delta Y_n + F_n\right) \quad \text{or} \quad \Delta Y_n = \frac{V_n^{k+1}A_n^k - Q_n^k - F_n}{E_n - V_n^{k+1}T_n^k}$$

Thereafter Equation 7.95 gives

$$\Delta Q_n = E_n\Delta Y_n + F_n$$

7.12.1.3 Upstream Boundary Conditions

The upstream boundary condition equations will be put in a form so that the recursive Equations 7.94 and 7.95 can be used to obtain ΔY_1 and ΔQ_1, just as these equations are used for points 2, 3, … n − 1. Thus E_1 and F_1 must be given values so this can be done.

Case (1)—Depth Y(0, t) is given: Since point 1 at the upstream end of the channel is the last point in using the recursive Equations 7.94 and 7.95, when Y^{+1} is known it is necessary to obtain a linear equation for ΔQ_1 of the form of Equation 7.95, namely, $\Delta Q_1 = E_1\Delta Y_1 + F_1$ in which ΔY_1 is known. Solving the equation for this known gives

$$\Delta Y_1 = \frac{\Delta Q_1}{E_n} - \frac{F_n}{E_n}$$

To allow ΔY_1 to be specified make E_n very large so that the first term after the equal sign is insignificant, say $E_n = 10^6$, and then solving for F_n gives

$$F_n = -10^6\Delta Y_1 = -10^6(Y(0, t + \Delta t) - Y(0, t)) = -10^6\left[Y_1^{k+1} - Y_1^k\right]$$

With these values for E_n and F_n, Equations 7.94 and 7.95 are used at point 1 to compute ΔY_1 and ΔQ_1.

Case (2)—Flow rate Q(0, t) is given: To use Equations 7.94 and 7.95 at the first point when ΔQ_1 is known, make $E_1 = 0$ and $F_1 = \Delta Q_1$ in Equation 7.95.

Case (3)—Stage Discharge relationship given or $Q_s(0, t) = f[Y_s(0, t)]$: Note that E_1 in Equation 7.95 must equal the slope of the stage discharge relationship if it is to be applicable. Therefore $E_1 = (dQ/dY)_{sj}$. Furthermore, then F_1 can be given by

$$F_1 = Q_s - Q_1^k$$

in which Q_s is the flow rate that corresponds to Y_1^k.

Case (4)—The head H in the reservoir given as a function of time, or from energy $H(t) = Y_1^{k+1} + (Q^2/A^2)/(2g)$ **is given**: This condition represents a special stage discharge relationship given by the energy equation at the channel entrance. Solving this energy equation for the flow rate gives $Q = A\{2g(H - Y)\}^{1/2}$ (or $V = \{2g(H - Y)\}^{1/2}$). E_1 must be the slope dQ/dY for Equation 7.95 to apply, and

$$\frac{dQ}{dY} = T\left\{2g(H-Y)\right\}^{\frac{1}{2}} - \frac{gA}{\sqrt{2g(H-Y)}} = TV - \frac{gA}{V}$$

and the coefficient F_1 is computed from

$$F_1 = A\sqrt{2g(H-Y)} - Q_1^k$$

The following FORTRAN program listing implements the Preissman method as described above in which the solution of the linearized equations is obtained using the double sweep method. The use of this Preissman method should be only for situations in which the changes in either the flow rate or depth are relatively small over any time increment Δt, because only first-order terms were used in linearizing the equations. This Preissman method does not actually use second-order approximations for the differential equations (before they are linearized) because in general the weighing factor δ must equal 1/2 if the derivatives with respect to time are to be second order approximations. An alternative would not linearize the finite difference equations and then solve for the ΔY's and ΔQ's using the Newton method. Another possibility would be to use the difference equations before they are manipulated into the forms that produces the differences in depths and flow rates between consecutive time steps and solve these nonlinear equations using the Newton method.

The input data required by program PREISDBS is very similar to that used in the previous programs, especially that used by program IMPLICBC. There are three differences in its input and that required by IMPLICBC. The fifth variable on the first line, IFREQ represents how many time step solutions should be obtained before printing the results instead of how many subincrements should be used, and this first line of input has variable DELTA (for δ) in place of the three variables: W,ERR,MAX. Also since this program implements the use of the kinetic energy correction coefficient α, the third difference is that the values for ALF (for α) for all points (station) along the channel are given after the slopes of the channel bottom. As with the other geometric data, etc. (Q, B, FM, FN, & SO) if values repeat the last one given the list for ALF can be terminated with a /.

FORTRAN listing of program that implements the Preissman double sweep method, PREISDBS.FOR

```
C  This program is based on the PREISSMANN method &
C     uses double Sweep to solve eqs.
      PARAMETER(ND=41,NDT=100) !ND=No. pts. NDT=No. time steps
```

```
      REAL Y(ND),Q(ND),Y1(ND),Q1(ND),B(ND),FM(ND),FN(ND),
     &SO(ND),A(ND),P(ND),T(ND),SMS(ND),QGIU(0:NDT),
     &QGID(0:NDT),V(ND),ALF(ND),FK(ND),DTDY(ND),DALFDY(ND),
     &DKDY(ND),H(ND),FB(ND),C(ND),D(ND),FG(ND),H1(ND),FB1(ND)
     &,C1(ND),D1(ND),FG1(ND),E(ND),F(ND),FL(ND),FMM(ND),FNN(ND)
      DATA ONE/-1./
      READ(2,*)
IOUT,NFORW,NX,NT,IFREQ,DELT,DELX,G,DELTA,IUBC,IDBC
C IUBC & IDBE are Upst. & Downst B.C. according to:
C   1-DEPTH known(given in QGIU or QGID;
C   2-FLOWRATE known(given in QGIU or QGDD);
C   3-STAGE-DISCHARGE given as pairs Y and Q in QGIU or QGID;
C   4-H(t)=y+Q*Q/(2gA*A) given in QGIU and
C     VELOCITY known for downstr. B.C. & given in QGID.
      G2=2.*G
      CMAN=1.
      IF(G.GT.30.) CMAN=1.486
      DXT=DELX/DELT
      GDELT=G*DELT
      GTDELT=GDELT*DELTA
      GEDELT=G2*DELTA*DELT
      NXM=NX-1
      DO 10 I=2,NX
      Q(I)=ONE
      B(I)=ONE
      FM(I)=ONE
      FN(I)=ONE
      ALF(I)=ONE
10    SO(I)=ONE
      DO 12 I=1,NT
      QGIU(I)=ONE
12    QGID(I)=ONE
      READ(2,*) (Q(I),I=1,NX)
      READ(2,*) (B(I),I=1,NX)
      READ(2,*) (FM(I),I=1,NX)
      READ(2,*) (FN(I),I=1,NX)
      READ(2,*) (SO(I),I=1,NX)
      READ(2,*) (ALF(I),I=1,NX)
      READ(2,*) (QGIU(I),I=0,NT)
      READ(2,*) (QGID(I),I=0,NT)
      DO 20 I=2,NX
      IM=I-1
      IF(Q(I).EQ.ONE) Q(I)=Q(IM)
      IF(B(I).EQ.ONE) B(I)=B(IM)
      IF(FM(I).EQ.ONE) FM(I)=FM(IM)
      IF(FN(I).EQ.ONE) FN(I)=FN(IM)
      IF(SO(I).EQ.ONE) SO(I)=SO(IM)
      IF(ALF(I).EQ.ONE) ALF(I)=ALF(IM)
20    SMS(I)=2.*SQRT(FM(I)**2+1.)
      SMS(1)=2.*SQRT(FM(1)**2+1.)
      NUSD=0
```

```
      NDSD=0
      DO 22 I=1,NT
      IF(QGIU(I).EQ.ONE) THEN
      IF(IUBC.NE.3) THEN
      QGIU(I)=QGIU(I-1)
      ELSE IF(NUSD.EQ.0) THEN
      NUSD=I-2
      ELSE
      GO TO 18
      ENDIF
      ENDIF
18    IF(QGID(I).EQ.ONE) THEN
      IF(IDBC.NE.3) THEN
      QGID(I)=QGID(I-1)
      ELSE IF(NDSD.EQ.0) THEN
      NDSD=I-2
      ELSE
      GO TO 22
      ENDIF
      ENDIF
22    CONTINUE
      JUBC=NUSD/2
      JDBC=NDSD/2
      IF(NFORW.EQ.1) THEN
      READ(2,*)(Y(I),I=1,NX)
      ELSE
      READ(2,*)(Y(I),I=NX,1,-1)
      ENDIF
      K=0
25    K=K+1
      DO 30 I=1,NX
      T(I)=B(I)+2.*FM(I)*Y(I)
      P(I)=B(I)+SMS(I)*Y(I)
      A(I)=(B(I)+FM(I)*Y(I))*Y(I)
      FK(I)=CMAN/FN(I)*A(I)*(A(I)/P(I))**.6666667
      DKDY(I)=FK(I)/A(I)*(5.*B(I)/3.-2.*A(I)/3./P(I)*SMS(I))
      DTDY(I)=2.*FM(I)
      V(I)=Q(I)/A(I)
30    DALFDY(I)=0.
      IF(MOD(K-1,IFREQ).EQ.0) THEN
      WRITE(IOUT,110) DELT*FLOAT(K-1)
110   FORMAT(///1X,'Solution for time=',F8.2/1X,57('-')/,4X,'x',7X,
     &'Y',7X,'V',7X,'P',7X,'Q',7X,'A',5X,'TOP'/1X,57('-'))
      WRITE(IOUT,120) (DELX*FLOAT(I-1),Y(I),V(I),P(I),Q(I),
     &A(I),T(I),I=1,NX)
120   FORMAT(F8.1,6F8.3)
      ENDIF
      IF(K.GT.NT) STOP
C  equation coefficients
      A0=4.*DELTA/DXT
      DO 40 I=1,NX-1
```

```
      TTT=T(I+1)+T(I)
      TTT2=TTT*TTT
      VV=Q(I)/A(I)
      VV2=VV*VV
      VVP=Q(I+1)/A(I+1)
      VVP2=VVP*VVP
      H(I)=1.-A0*(Q(I+1)-Q(I))/TTT2*DTDY(I+1)
      FB(I)=A0/TTT
      C(I)=A0*(Q(I+1)-Q(I))/TTT2*DTDY(I)-1.
      D(I)=FB(I)
      FG(I)=-4./DXT*(Q(I+1)-Q(I))/TTT
      A1=DELTA/DXT
      A3=.5*(VVP*T(I+1)+VV*T(I))
      A4=ALF(I+1)*VVP2+ALF(I)*VV2
      H1(I)=-A3+A1*((Q(I+1)-Q(I))*(VVP*DALFDY(I+1)-
     &Q(I+1)*T(I+1)*ALF(I+1)/A(I+1)**2)-T(I+1)*A4-VVP2*(A(I+1)-
     &A(I))*(DALFDY(I+1)-2.*ALF(I+1)*T(I+1)/A(I+1))+G*T(I+1)
     &*(Y(I+1)-Y(I))-.5*VVP2*T(I+1)*(ALF(I+1)-ALF(I))+G*(A(I+1)
     &+A(I))+.5*(Q(I+1)*VVP+Q(I)*VV)*DALFDY(I+1))+ GTDELT*Q(I+1)
     &*ABS(Q(I+1))/FK(I+1)**2*(T(I+1)-2.*A(I+1)/FK(I+1)*DKDY(I+1))
      FB1(I)=1.+A1*(ALF(I+1)*VVP+ALF(I)*VV+(Q(I+1)-Q(I))*ALF(I+1)/
     &A(I+1)-(A(I+1)-A(I))*2.*ALF(I+1)*VVP2+(ALF(I+1)-ALF(I))*VVP)+
     &GEDELT*A(I+1)*ABS(Q(I+1))/FK(I+1)**2
      C1(I)=A3-A1*((Q(I+1)-Q(I))*(VV*DALFDY(I)-Q(I)
     &*T(I)*ALF(I)/A(I)**2)+T(I)*A4-VV2*(A(I+1)-A(I))*
     &(DALFDY(I)-2.*ALF(I)*T(I)/A(I))+G*T(I)*(Y(I+1)-Y(I))
     &-.5*VV2*(ALF(I+1)-ALF(I))-G*(A(I+1)+A(I))-.5*(Q(I+1)*
     &VVP+Q(I)*VV)*DALFDY(I))-GTDELT*Q(I)*ABS(Q(I))/
     &FK(I)**2*(T(I)-2.*A(I)/FK(I)*DKDY(I))
      D1(I)=-1.-A1*(-ALF(I+1)*VVP-ALF(I)*VV+(Q(I+1)-Q(I))*
     &ALF(I)/A(I)-(A(I+1)-A(I))*2.*ALF(I)*VV2+(ALF(I+1)-
     &ALF(I))*VV)-GEDELT*A(I)*ABS(Q(I))/FK(I)**2
      FG1(I)=-((Q(I+1)-Q(I))*(ALF(I+1)*VVP+ALF(I)*VV)-
     &(A(I+1)-A(I))*(ALF(I+1)*VVP2+ALF(I)*VV2)+G*(Y(I+1)-
     &Y(I))*(A(I+1)+A(I))+(.5*Q(I+1)*VVP+.5*Q(I)*VV)*
     &(ALF(I+1)-ALF(I)))/DXT-GDELT*(Q(I+1)*ABS(Q(I+1))*
     &A(I+1)/FK(I+1)**2+Q(I)*ABS(Q(I))*A(I)/FK(I)**2)
     &+GDELT*SO(I)*(A(I)+A(I+1))
40    CONTINUE
C  compute E1, F1 from boundary condition at point I=1
      IF(IUBC.EQ.1) THEN      !Y(0,t) is given
      E(1)=1000000.
      F(1)=-1000000.*(QGIU(K)-Y(1))
      ELSE IF(IUBC.EQ.2) THEN !Q(0,T) is given
      E(1)=0.
      F(1)=QGIU(K)-Q(1)
      ELSE IF(IUBC.EQ.3)THEN !stage-discharge table
      CALL BOUNDY(NUSD,JUBC,QGIU,Y(1),f1,E(1))
      F(1)=f1-Q(1)
      ELSE                    ! Ho(t)=Y+V**2/2g is given
      TT=B(1)+2.*FM(1)*Y(1)
```

```
      AA=(B(1)+FM(1)*Y(1))*Y(1)
      f1=AA*SQRT(G2*(QGIU(K)-Y(1)))
      VV=f1/AA
      E(1)=TT*VV-AA*G/VV
      F(1)=f1-Q(1)
      ENDIF
C compute Li, Mi, Ni, and E(i+1), F(i+1)
      DO 50 I=1,NX-1
      CC1=C(I)+D(I)*E(I)
      FL(I)=H(I)/CC1
      FMM(I)=FB(I)/CC1
      FNN(I)=-(FG(I)+D(I)*F(I))/CC1
      CC2=FB1(I)*CC1-FB(I)*(C1(I)+D1(I)*E(I))
      E(I+1)=(H(I)*(C1(I)+D1(I)*E(I))-H1(I)*(C(I)+D(I)*E(I)))/CC2
      F(I+1)=((FG1(I)+D1(I)*F(I))*(C(I)+D(I)*E(I))-(FG(I)+D(I)*
     &F(I))*(C1(I)+D1(I)*E(I)))/CC2
50    CONTINUE
      IF(IDBC.EQ.1) THEN      !Y(L,t) is given
      DY=QGID(K)-Y(NX)
      DQ=E(NX)*DY+F(NX)
      ELSE IF(IDBC.EQ.2) THEN !Q(L,t) is given
      DQ=QGID(K)-Q(NX)
      DY=(DQ-F(NX))/E(NX)
      ELSE IF(IDBC.EQ.3) THEN !stage-discharge table given
      CALL BOUNDY(NDSD,JDBC,QGID,Y(NX),f2,df2dy)
      DY=(f2-F(NX)-Q(NX))/(E(NX)-df2dy)
      DQ=E(NX)*DY+F(NX)
      ELSE  !V(L,t) is given
      DY=(QGID(K)*A(NX)-Q(NX)-F(NX))/(E(NX)-QGID(K)*T(NX))
      DQ=E(NX)*DY+F(NX)
      ENDIF
      Y1(NX)=Y(NX)+DY
      Q1(NX)=Q(NX)+DQ
C Compute DQ, DY, and compute new Y, Q at new time.
      DO 60 I=NX-1,1,-1
      DY1=FL(I)*DY+FMM(I)*DQ+FNN(I)
      DQ1=E(I)*DY1+F(I)
      Y1(I)=Y(I)+DY1
      Q1(I)=Q(I)+DQ1
      DY=DY1
      DQ=DQ1
60    CONTINUE
C reinitialize the values for next time step
      DO 70 I=1,NX
      Q(I)=Q1(I)
      Y(I)=Y1(I)
70    CONTINUE
      GO TO 25
      END
      SUBROUTINE BOUNDY(N,JJ,QGIV,YY,f1,df1dy)
```

```
      REAL QGIV(0:100)
10    IF(QGIV(JJ+2).GT.YY .OR. JJ+2.GE.N) GO TO 20
      JJ=JJ+2
      GO TO 10
20    IF(QGIV(JJ).LE.YY .OR.JJ.LT.1) GO TO 30
      JJ=JJ-2
      GO TO 20
30    df1dy=(QGIV(JJ+3)-QGIV(JJ+1))/(QGIV(JJ+2)-QGIV(JJ))
      f1=QGIV(JJ+1)+df1dy*(YY-QGIV(JJ))
      RETURN
      END
```

7.13 SOLVING PREISSMANN DIFFERENCE EQUATIONS USING THE NEWTON METHOD

In this section the difference equation developed in the previous section will not be linearized, but solved using the Newton method. Recall that the second-order differences with respect of x were defined in the previous section by $\partial f/\partial x \approx (f_{i+1} - f_i)/\Delta x$ in which the derivative is being evaluated at i + ½ based on an increment of $\Delta x/2$. As discussed in the previous section where the equations are evaluated with respect to time can vary depending upon a weighting factor δ that can vary between 0 and 1. When $\delta = 0.5$ then this weighting is midway between the current and the advanced time steps and differences with respect to time are also second order.

Applying this differencing scheme to the Q-y form of the St. Venant equations as given in Equations 7.61 and 7.62 gives the following pair of equations for each grid point in the x directions (the equation of motion is written first and the continuity equation second, also note that the kinetic energy correction coefficient, α, used in the previous section has also been deleted but lateral inflow, or outflow has been included):

$$
\begin{aligned}
F_{i1} = {} & \delta\left\{2V^{k+1}_{i+\frac{1}{2}}\left(Q^{k+1}_{i+1} - Q^{k+1}_{i}\right) + \left[T(c^2 - V^2)\right]^{k+1}_{i+\frac{1}{2}}\left(Y^{k+1}_{i+1} - Y^{k+1}_{i}\right) + \Delta x (V^2)^{k+1}_{i+\frac{1}{2}}\left[\left.\frac{\partial A}{\partial x}\right|_{Y,t}\right]^{k+1}_{i+\frac{1}{2}}\right\} \\
& + \delta\left\{\Delta x A^{k+1}_{i+\frac{1}{2}}\left(S^{\,k+1}_{f_{i+\frac{1}{2}}} - S_0 + F^{\,k+1}_{q_{i+\frac{1}{2}}}\right) - \Delta x [Vq^*]^{k+1}_{i+\frac{1}{2}}\right\} + \frac{\Delta x}{2\Delta t}\left(Q^{k+1}_{i+1} + Q^{k+1}_{i} - Q^{k}_{i+1} - Q^{k}_{i}\right) \\
& + (1-\delta)\left\{2V^{k}_{i+\frac{1}{2}}\left(Q^{k}_{i+1} - Q^{k}_{i}\right) + \left[T(c^2 - V^2)\right]^{k}_{i+\frac{1}{2}}\left(Y^{k}_{i+1} - Y^{k}_{i}\right) - \Delta x (V^2)^{k}_{i+\frac{1}{2}}\left[\left.\frac{\partial A}{\partial x}\right|_{Y,t}\right]^{k}_{i+\frac{1}{2}}\right\} \\
& + (1-\delta)\left\{g\Delta x A^{k}_{i}\left(S^{\,k}_{f_{i+\frac{1}{2}}} - S_0 + F^{\,k}_{q_{i+\frac{1}{2}}} - \Delta x [Vq^*]^{k}_{i+\frac{1}{2}}\right)\right\} = 0 \qquad (7.98)
\end{aligned}
$$

and

$$
\begin{aligned}
F_{i2} = {} & \frac{\Delta x}{4\Delta t}\left[\delta\left(T^{k+1}_{i+1} + T^{k+1}_{i}\right) + (1-\delta)\left(T^{k}_{i+1} + T^{k}_{i}\right)\right]\left(Y^{k+1}_{i+1} + Y^{k+1}_{i} - Y^{k}_{i+1} - Y^{k}_{i}\right) \\
& + \delta\left\{Q^{k+1}_{i+1} - Q^{k+1}_{i} - \Delta x (q^*)^{k+1}_{i+\frac{1}{2}}\right\} + (1-\delta)\left\{Q^{k}_{i+1} - Q^{k}_{i} - \Delta x (q^*)^{k}_{i+\frac{1}{2}}\right\} = 0 \qquad (7.99)
\end{aligned}
$$

Equations 7.98 and 7.99 are the regular finite difference operators that apply from grid position i = 1/2 to i = n − 1/2, in which n is the number of the finally grid at the downstream end of the channel, and the grid at the upstream end of channel is numbered 1, e.g., i = 1 + x/Δx. When boundary condition equations are added to these regular equations and this system of equations is solved, then the solution is advanced through a new time step, at k + 1 similar to that used in the previously described implicit methods.

At the upstream end of the channel, if a reservoir exist then the energy principle gives the following equation:

$$F_{1,1} = H(t) - Y_1^{k+1} - (1+K_e)\left[\frac{Q^2}{2gA^2}\right]_1^{k+1} = 0 \tag{7.100}$$

A more general upstream boundary condition might specify a "stage discharge" relationship, that can be in the form of a table of values in which flow rates as a function of the depth Y at the entrance of the channel is given. The energy equation 100 could be used to develop such a stage discharge relationship, and if the head of the reservoir changes with time a different stage discharge relationship would apply for each succeeding time step. Using a stage discharge relationship for the upstream boundary, the equation is of the form

$$F_{1,1} = Q_1^{k+1} - Q^{k+1}\left(Y_1^{k+1}\right) = 0 \tag{7.101}$$

in which $Q^{k+1}(Y_1^{k+1})$ is this stage discharge relationship that gives the flow rate as a function of the depth at the entrance of the channel, and Y_1^{k+1} is used to evaluate this function. If tabular values define this relationship then $Q^{k+1}(Y_1^{k+1})$ is obtained by appropriate interpolation.

The second equations at the upstream boundary is the continuity equation in finite difference form, or,

$$\begin{aligned} F_{1,2} = {} & \frac{\Delta x}{4\Delta t}\left[\delta\left(T_{i+1}^{k+1} + T_i^{k+1}\right) + (1-\delta)\left(T_2^k + T_1^k\right)\right]\left(Y_2^{k+1} + Y_1^{k+1} - Y_2^k - Y_1^k\right) \\ & + \delta\left\{Q_2^{k+1} - Q_1^{k+1} - \Delta x(q^*)_{i+\frac{1}{2}}^{k+1}\right\} + (1-\delta)\left\{Q_2^k - Q_1^k - \Delta x(q^*)_{i+\frac{1}{2}}^{k}\right\} = 0 \end{aligned} \tag{7.102}$$

At the downstream end of the channel a stage discharge relationship might also be used. If so, then Equation 7.101 would apply with the subscript 1 replaced by the subscript n. If one of the three conditions used earlier is used, namely, (a) the flow rate given as a function of time, (b) the depth given as a function of time, or (c) the velocity given as a function of time, then

$$F_{n,1} = Q_n^{k-1} - Q(t) = 0 \quad \text{(flowrate specified)} \tag{7.103a}$$

$$F_{n,1} = Y_n^{k-1} - Y(t) = 0 \quad \text{(depth specified)} \tag{7.103b}$$

$$F_{n,1} = \frac{Q_n^{k-1}}{A_n^{k+1}} - V(t) = 0 \quad \text{(velocity specified)} \tag{7.103c}$$

in which Q(t), Y(t), or V(t) is the given flow rate, depth or velocity. Note using Equations 103a, 103b, or 103c is an alternative to the procedure used in implementing the Crank–Nicolson methods.

In that section since the flow rate, depth or velocity at the end of the channel is specified one equation was omitted from the pair. Using Equations 7.103a, 7.103b, or 7.103c has the disadvantage of adding one more equation to the system of equations, but it simplifies the logic in implementing a computer solution since now there are 2n equations in this system.

The second downstream equation comes from the continuity equation. If a second-order approximation is used and lateral outflow (inflow) is assumed zero, then this equation becomes

$$F_{n,2} = \delta\left\{1.5Q_n^{k+1} - 2Q_{n-1}^{k+1} - 0.5Q_{n-2}^{k+1}\right\} + \frac{\Delta x}{2\Delta t}\left[\left(\delta T_n^{k+1} + (1-\delta)T_n^k\right)\right]\left(Y_n^{k+1} + Y_n^k\right)$$
$$+ (1-\delta)\left\{1.5Q_n^k - 2Q_{n-1}^k - 0.5Q_{n-2}^k\right\} = 0 \tag{7.104}$$

In the original Preissmann method linearization of these equation is done, but if the Newton method is used to solve the resulting system it should be noted that in case of both the regular equations F_{i1} and F_{i2} that there are only four nonzero derivatives. If the unknowns are listed with the depth Y_i^{k+1} first and the flow rate Q_i^{k+1} second as pairs, then the resulting Jacobian matrix will have all zero elements in front of the diagonal with a nonzero element on the diagonal and the following three elements for the derivatives $\partial F_{i1}/\partial Y_i$, $\partial F_{i1}/\partial Q_i$, $\partial F_{i1}/\partial Y_{i+1}$ and $\partial F_{i1}/\partial Q_{i+1}$, respectively. On the second row of each pair in this Jacobian matrix, there will be one nonzero element in front of the diagonal, a nonzero element on the diagonal, followed by two nonzero elements consisting of the following four derivatives: $\partial F_{i2}/\partial Y_i$, $\partial F_{i2}/\partial Q_i$, $\partial F_{i2}/\partial Y_{i+1}$ and $\partial F_{i2}/\partial Q_{i+1}$.

The detail associated with define the equations and Jacobian matrix are illustrated in the following FORTRAN listing that implements the Preissmann method with the finite difference equations solved by the Newton method. The following explanations will help in following this program's logic, etc. Note now the array A(101,4), which is used for the Jacobian matrix has four elements across each row (the second subscript) for the four nonzero derivatives. The one dimensional array F is used for the equation vector before calling subroutine BAND, but after returning from this subroutine F hold the values of the correction vector {Z} used in the Newton method, or is the solution of

$$[D]\{Z\} = \{F\}$$

In solving this system of equations the subroutine BAND needs to only eliminate one element from every other row to reduce the banded matrix problem to upper triangular, after which back substitution solves the problem. A few extra elimination step are required in this solution process, however, to accommodate Equation 7.104 at the downstream boundary. Equation 7.104 is taken as the second from the last equation with Equation 7.103 taken as the last equation. Equation 7.103 produces only a 1 on the diagonal of the last row of the Jacobian matrix. The two unknowns at each point Y_i^{k+1} and Q_i^{k+1}, with i = 1 to n are stored in the one-dimensional array Y with the depth occupying odd positions and Q the even positions in this array. Therefore a depth corresponding to grid i is referenced in this program with II = 2*I − 1 and the flow rate is referenced by IP = II + 1 = 2*I. In this program all derivatives are evaluated by numerical approximations with the following statements in subroutine IMPLNE:

```
      DO 25 I=1,NX
      DO 20 J=1,N4
      JJ=2*I-2+J
      YY=Y(JJ)
      Y(JJ)=1.005*YY
      CALL EQUAT(I,.TRUE.)
      A(II,J)=(V(II)-F(II))/(Y(JJ)-YY)
      A(IP,J)=(V(IP)-F(IP))/(Y(JJ)-YY)
20    Y(JJ)=YY
```

```
      F(II)=F(II)+FB(II)
      F(IP)=F(IP)+FB(IP)
      IF(I.EQ.NXM) N4=2
25    CONTINUE
```

When the second argument in the CALL EQUAT is .TRUE., then the terms associated with the time derivatives are added to the equations F_{i1} and F_{i2}, and when this argument is .FALSE. these terms are not included. Therefore when EQUAT is called in the loop DO 15 I = 2,NX the portion of the equations with superscript k are evaluated, and when EQUAT is called in the loop DO 18 I = 1,NX the terms with superscript k + 1 from space derivatives are included in the equation vector F.

Listing of FORTRAN program PREISM6.FOR that implements Preissmann method and the Newton method for solving equations

```
C This program implements solving the Preissman equations using
C the Newton method. In defining the equations, the average of the
C values at the i & i+1 position is used. WT is the weighting
C between the current and the advanced time step. The number of
C equation written is 2*NX-1, i.e. the first eq. is the energy eq.
C at the beginning of channel with the reservior head H=const.
C At the downstream boundary either the depth may be specified,
C or the flow rate may be specified. This seem to be the most
C successful attemp at using this method, but the convergence
C still is very slow as if a linear convergant method is being used.
      LOGICAL LBC
      COMMON /SOLVNE/ A(101,4),F(101)
      COMMON Y(102),YB(51),QB(51),V(101),FB(101),TB(51),V1,X(51),
     &SQ(51),B(51),FM(51),SO(51),FN(51),SMS(51),SF,WT,WTP,
     &QGIV(0:200),DXG,G,G2,DELX,DXDT2,DXDT4,DEL2,DELT,DXT,RDT,H,
     &CMAN,EK,UQ,ERR,NT,NX,NXM,NX2,IOUT,IYOUT,LBC,IFREQ,IE,IEP,
     &IY,IQ,IYP,IQP,IA
      DATA ONE/-1./
      READ(2,*) IOUT,IFREQ,NFORW,NX,NT,DELT,DELX,G,IBC2,IYOUT,UQ,
     &EK,ERR,WT
      WTP=WT
      LBC=.FALSE.
      IF(IBC2.EQ.2) LBC=.TRUE.
      G2=2.*G
      CMAN=1.
      IF(G.GT.30.) CMAN=1.486
      DEL2=2.*DELX
      DXT=DELX/DELT
      DXDT2=DXT/2.
      DXDT4=DXT/4.
      EK=(1.+EK)/G2
      DXG=G*DELX/2.
      RDT=1./DELT
      NXM=NX-1
      NX2=NX-2
      DO 10 I=1,NX
      Y(2*I)=ONE
```

```
      B(I)=ONE
      FM(I)=ONE
      FN(I)=ONE
10    SO(I)=ONE
      READ(2,*) (Y(2*I),I=1,NX)
      READ(2,*) (B(I),I=1,NX)
      READ(2,*) (FM(I),I=1,NX)
      READ(2,*) (FN(I),I=1,NX)
      READ(2,*) (SO(I),I=1,NX)
      DO 20 I=2,NX
      IM=I-1
      IF(Y(2*I).EQ.ONE) Y(2*I)=Y(2*IM)
      IF(B(I).EQ.ONE) B(I)=B(IM)
      IF(FM(I).EQ.ONE) FM(I)=FM(IM)
      IF(FN(I).EQ.ONE) FN(I)=FN(IM)
      IF(SO(I).EQ.ONE) SO(I)=SO(IM)
20    SMS(I)=2.*SQRT(FM(I)**2+1.)
      DO 21 I=2,NXM
21    SQ(I)=Y(2*I+2)-Y(2*I-2)
      SMS(1)=2.*SQRT(FM(1)**2+1.)
      SQ(1)=2.*(Y(4)-Y(2))
      SQ(NX)=2.*(Y(2*NX)-Y(2*NXM))
      IF(NFORW.EQ.1) THEN
      READ(2,*)(Y(2*I-1),I=1,NX)
      ELSE
      READ(2,*)(Y(2*I-1),I=NX,1,-1)
      ENDIF
      DO 30 I=1,NT
30    QGIV(I)=ONE
      READ(2,*)(QGIV(I),I=0,NT)
      IF(LBC) THEN
      IF(ABS(Y(2*NX)-QGIV(0)).GT. .1) THEN
      WRITE(*,100) Y(2*NX),QGIV(0)
100   FORMAT(' Flowr.',F8.2,' at end must be same as 1st',
     &' Q at t=0',F8.2)
      STOP
      ENDIF
      ELSE
      IF(ABS(Y(2*NX-1)-QGIV(0)).GT. .1) THEN
      WRITE(*,101) Y(2*NX-1),QGIV(0)
101   FORMAT(' Depth',F8.2,' at end must be same as 1st',
     &' Y at t=0',F8.2)
      STOP
      ENDIF
      ENDIF
      DO 40 I=1,NT
      IF(QGIV(I).EQ.ONE) QGIV(I)=2.*QGIV(I-1)-QGIV(I-2)
40    CONTINUE
      H=Y(1)+EK*(Y(2)/((B(1)+FM(1)*Y(1))*Y(1)))**2
      CALL IMPLNE
      END
```

```
      SUBROUTINE IMPLNE
      LOGICAL LBC
      COMMON /SOLVNE/ A(101,4),F(101)
      COMMON Y(102),YB(51),QB(51),V(101),FB(101),TB(51),V1,X(51),
     &SQ(51),B(51),FM(51),SO(51),FN(51),SMS(51),SF,WT,WTP,
     &QGIV(0:200),DXG,G,G2,DELX,DXDT2,DXDT4,DEL2,DELT,DXT,RDT,H,
     &CMAN,EK,UQ,ERR,NT,NX,NXM,NX2,IOUT,IYOUT,LBC,IFREQ,IE,IEP,
     &IY,IQ,IYP,IQP,IA
      KNEXT=0
      DO 50 K=1,NT
      SQB1=SQ(1)
      SQBN=SQ(NX)
      IF(LBC) THEN
      Y(2*NX)=QGIV(K)   ! Q(t)=Given
      ELSE
      Y(2*NX-1)=QGIV(K)   ! Y(t)=Given
      ENDIF
      DO 10 I=1,NX
      YB(I)=Y(2*I-1)
      QB(I)=Y(2*I)
10    TB(I)=B(I)+2.*FM(I)*YB(I)
      WT=1.-WTP
      DO 15 I=1,NXM
      CALL EQUAT(I,.FALSE.)
      FB(IE)=V(IE)
15    FB(IEP)=V(IEP)
      IF(K.EQ.1) THEN
      WRITE(IOUT,100) K-1,DELT*FLOAT(K-1)
      DO 11 I=1,NX
      AA=(B(I)+FM(I)*Y(2*I-1))*Y(2*I-1)
      V1=Y(2*I)/AA
      E=Y(2*I-1)+V1**2/G2
      FR=SQRT(V1**2*(B(I)+2.*FM(I)*Y(2*I-1))/(G*AA))
11    WRITE(IOUT,101) IFIX(DELX*FLOAT(I-1)),Y(2*I-1),Y(2*I),
     &V1,AA,E,FR
      ENDIF
C READ IN q* FOR THIS TIME STEP (No. of values, pair of(position & q*)
      IF(IYOUT.GT.0 .AND. K.GE.KNEXT) THEN
      READ(2,*) NSQ,(II,SQ(II),I=1,NSQ)
      READ(2,*,END=12) KNEXT
      GO TO 14
12    KNEXT=NT+10
      ENDIF
14    WT=WTP
      NCT=0
16    DO 18 I=1,NXM
      CALL EQUAT(I,.TRUE.)
      F(IE)=V(IE)
18    F(IEP)=V(IEP)
      AA=(B(1)+FM(1)*Y(1))*Y(1)
```

```
      A(1,1)=2.*EK*Y(2)**2/AA**3*(B(1)+2.*FM(1)*Y(1))-1.
      A(1,2)=-2.*EK*Y(2)/AA**2
      DO 25 I=1,NX-2
      DO 20 J=1,4
      JJ=2*I-2+J
      YY=Y(JJ)
      Y(JJ)=1.005*YY
      CALL EQUAT(I,.TRUE.)
      A(IE,J)=(V(IE)-F(IE))/(Y(JJ)-YY)
      A(IEP,J)=(V(IEP)-F(IEP))/(Y(JJ)-YY)
20    Y(JJ)=YY
      F(IE)=F(IE)+FB(IE)
25    F(IEP)=F(IEP)+FB(IEP)
      JJ=2*(NX-2)
      DO 28 J=1,3
      JJ=JJ+1
      IF(LBC .AND. J.EQ.3) JJ=JJ+1
      YY=Y(JJ)
      Y(JJ)=1.005*YY
      CALL EQUAT(NXM,.TRUE.)
      A(IE,J)=(V(IE)-F(IE))/(Y(JJ)-YY)
      A(IEP,J)=(V(IEP)-F(IEP))/(Y(JJ)-YY)
28    Y(JJ)=YY
      CALL BAND(IEP)
      SUM=0.
      DO 30 I=1,IEP-1
      Y(I)=Y(I)-F(I)
30    SUM=SUM+ABS(F(I))
      IF(LBC) THEN
      Y(2*NX-1)=Y(2*NX-1)-F(IEP)
      ELSE
      Y(2*NX)=Y(2*NX)-F(IEP)
      ENDIF
      SUM=SUM+ABS(F(IEP))
      NCT=NCT+1
      WRITE(*,330) NCT,SUM
330   FORMAT(' Iteration=',I4,' Residual=',E10.4)
      IF(NCT.LT. 50 .AND. SUM.GT.ERR) GO TO 16
      IF(NCT.EQ.50) READ(*,*)
      IF(MOD(K,IFREQ).EQ.0) THEN
      WRITE(IOUT,100) K,DELT*FLOAT(K)
      DO 48 I=1,NX
      AA=(B(I)+FM(I)*Y(2*I-1))*Y(2*I-1)
      V1=Y(2*I)/AA
      E=Y(2*I-1)+V1**2/G2
      FR=SQRT(V1**2*(B(I)+2.*FM(I)*Y(2*I-1))/(G*AA))
48    WRITE(IOUT,101) IFIX(DELX*FLOAT(I-1)),Y(2*I-1),Y(2*I),
     &V1,AA,E,FR
      ENDIF
50    CONTINUE
```

```
100   FORMAT(/' Solution at K=',I4,' Time=',F8.1,/,1X,63('-')/,1X,
     &' Posit. Depth Flowrate Velocity    Area    E    Fr ',/,1X,
     &63('-'))
101   FORMAT(I4,6F10.3)
      END
      SUBROUTINE EQUAT(I,TIMDER)
      LOGICAL LBC,TIMDER
      COMMON Y(102),YB(51),QB(51),V(101),FB(101),TB(51),V1,
     &X(51),SQ(51),B(51),FM(51),SO(51),FN(51),SMS(51),SF,WT,
     &WTP,QGIV(0:200),DXG,G,G2,DELX,DXDT2,DXDT4,DEL2,DELT,DXT,
     &RDT,H,CMAN,EK,UQ,ERR,NT,NX,NXM,NX2,IOUT,IYOUT,LBC,
     &IFREQ,IE,IEP,IY,IQ,IYP,IQP,IA
      IA=I+1
      IE=2*I
      IEP=IE+1
      IY=2*I-1
      IQ=IY+1
      IYP=IQ+1
      IQP=IYP+1
      IF(TIMDER) THEN
      A1=(B(I)+FM(I)*Y(IY))*Y(IY)
      V1=Y(IQ)/A1
      P1=B(I)+SMS(I)*Y(IY)
      TOP1=B(I)+2.*FM(I)*Y(IY)
      A2=(B(IA)+FM(IA)*Y(IYP))*Y(IYP)
      V2=Y(IQP)/A2
      P2=B(IA)+SMS(IA)*Y(IYP)
      TOP2=B(IA)+2.*FM(IA)*Y(IYP)
      DA=(B(IA)-B(I)+(FM(IA)-FM(I))*Y(IY))*Y(IY)
      ELSE
      A1=(B(I)+FM(I)*YB(I))*YB(I)
      V1=QB(I)/A1
      TOP1=B(I)+2.*FM(I)*YB(I)
      P1=B(I)+SMS(I)*YB(I)
      A2=(B(IA)+FM(IA)*YB(IA))*YB(IA)
      V2=QB(IA)/A2
      TOP2=B(IA)+2.*FM(IA)*YB(IA)
      P2=B(IA)+SMS(IA)*YB(IA)
      DA=(B(IA)-B(I)+(FM(IA)-FM(I))*YB(I))*YB(I)
      ENDIF
      IF(I.EQ.1) THEN
      IF(TIMDER) THEN
      V(1)=H-Y(1)-EK*V1**2
      ELSE
      V(1)=0.
      ENDIF
      ENDIF
      SF1=V1*ABS(V1)*(FN(I)/CMAN)**2*(P1/A1)**1.3333333
      SF2=V2*ABS(V2)*(FN(IA)/CMAN)**2*(P2/A2)**1.3333333
      AA=.5*(A1+A2)
```

```
      VA=.5*(V1**2+V2**2)
      IF(IYOUT.EQ.0) THEN
      FQ=0.
      ELSE IF(IYOUT.EQ.1) THEN
      IF(TIMDER) THEN
      FQ=(V1-UQ)*SQ(I)/DEL2/(G*AA)+((B(I)+B(IA))/2.+(FM(I)+
     &FM(IA))*Y(IY)/3.)*(Y(IQ)/A1)**2*((B(IA)+FM(IA)*
     &Y(IYP))*Y(IYP)-(B(I)+FM(I)*Y(IY))*Y(IY))/DELX
      ELSE
      FQ=(V1-UQ)*SQ(I)/DEL2/(G*AA)+((B(I)+B(IA))/2.+(FM(I)+
     &FM(IA))*YB(I)/3.)*(QB(I)/A1)**2*((B(IA)+FM(IA)*
     &YB(IA))*YB(IA)-(B(I)+FM(I)*YB(I))*YB(I))/DELX
      ENDIF
      ELSE
      FQ=V1*SQ(I)/DEL2/(G2*A1)
      ENDIF
      Z1=DXG*(SF1+SF2-SO(I)-SO(IA)+FQ)*AA+.5*(V1+V2)*SQ(I)-DA*VA
      IF(TIMDER) THEN
      Z2=(V1+V2)*(Y(IQP)-Y(IQ))+(G*AA-.5*(TOP1+TOP2)*VA)*(Y(IYP)-
     &Y(IY))
      ELSE
      Z2=(V1+V2)*(QB(IA)-QB(I))+(G*AA-.5*(TOP1+TOP2)*VA)*(YB(IA)-
     &YB(I))
      ENDIF
      V(IE)=WT*(Z1+Z2)
      V(IEP)=WT*(Y(IQP)-Y(IQ)-DELX*SQ(I))
      IF(TIMDER) THEN
      V(IE)=V(IE)+DXDT2*(Y(IQP)+Y(IQ)-QB(I)-QB(IA))
      V(IEP)=V(IEP)+DXDT4*(WT*(TOP1+TOP2)+(1.-WT)*(TB(I)+TB(IA)))*
     &(Y(IY)+Y(IYP)-YB(I)-YB(IA))
      ENDIF
      RETURN
      END
      SUBROUTINE BAND(N)
      COMMON /SOLVNE/ A(101,4),F(101)
      NM=N-1
      N2=N-2
      DO 10 I=3,N,2
      IM=I-1
      FAC=A(I,1)/A(IM,1)
      F(I)=F(I)-FAC*F(IM)
      DO 10 J=2,4
10    A(I,J)=A(I,J)-FAC*A(IM,J)
      FAC=A(2,1)/A(1,1)
      F(2)=F(2)-FAC*F(1)
      A(2,2)=A(2,2)-FAC*A(1,2)
      DO 11 I=3,N
      IM=I-1
      IF(MOD(I,2).EQ.0) THEN
      FAC=A(I,1)/A(IM,3)
```

```
      A(I,2)=A(I,2)-FAC*A(IM,4)
      ELSE
      FAC=A(I,2)/A(IM,2)
      A(I,3)=A(I,3)-FAC*A(IM,3)
      A(I,4)=A(I,4)-FAC*A(IM,4)
      ENDIF
11    F(I)=F(I)-FAC*F(IM)
      F(N)=F(N)/A(N,3)
      F(NM)=(F(NM)-A(NM,3)*F(N))/A(NM,2)
      DO 15 I=N2,3,-2
      IM=I-1
      F(I)=(F(I)-A(I,4)*F(I+1))/A(I,3)
15    F(IM)=(F(IM)-A(IM,3)*F(I)-A(IM,4)*F(I+1))/A(IM,2)
      F(1)=(F(1)-A(1,2)*F(2))/A(1,1)
      RETURN
      END
```

7.14 TWO-DIMENSIONAL FREE SURFACE FLOWS

There are many occurrences of free surface flows in which the water movement has significant components in two directions that the one-dimensional flow assumptions in the previous part of this chapter are too much at variance with the real occurrence. A few examples of such flows are as follows: (1) water spreading over an irrigated field that enters from one or two spots, (2) the flood flow spreading out over the land area downstream from a dam that is failing, or has failed, (3) the flow of a river beyond its mouth into a relatively flat bay or estuary, (4) water entering a sewerage treatment lagoon, and (5) the overland flow caused by rainfall on a watershed that is moving to the streams and rivers that drain the area. These types of two-dimensional flow problems are now being solved numerically using some form of two-dimensional free surface flow equations. These equations are commonly called shallow water equations, or two-dimensional St. Venant equations. The exact form of these equations can vary to suit the specific application that they are used to solve.

To extend the St. Venant equations for two-dimensional flow, a third equation is needed to account for the motion in the second spatial direction. A forms of these equations is given below. A description of the terms in these equations is then given to provide you the reader with some intuitive feeling for them, and thereafter a more formal derivation of the equations is given. One form of the shallow water equations is as follows.

7.14.1 Two-Dimensional St. Venant Equations

I. Continuity

$$\frac{\partial(uY)}{\partial x}+\frac{\partial(vY)}{\partial y}+\frac{\partial Y}{\partial t}-V^*=0 \quad \left(\text{each term has dimensions of } \frac{L}{t}\right) \tag{7.105}$$

II. Equations of motion (Newton's second law) (equation with dimensions of L^2/t^2)
(a) x-direction

$$\frac{\partial(u^2Y)}{\partial x}+\frac{\partial(uvY)}{\partial y}+g\frac{\partial}{\partial x}\left(\frac{1}{2}Y^2\right)+\frac{\partial(uY)}{\partial t}-gY(S_{ox}-S_{fx})+m_x=0 \tag{7.106}$$

(b) y-direction

$$\frac{\partial(v^2Y)}{\partial y}+\frac{\partial(uvY)}{\partial x}+g\frac{\partial}{\partial y}\left(\frac{1}{2}Y^2\right)+\frac{\partial(vY)}{\partial t}-gY(S_{oy}-S_{fy})+m_y=0 \tag{7.107}$$

in which u is the average velocity through the depth of flow in the x-direction, v is the average velocity through the depth of flow in the y-direction, V* is the velocity of inflow (or outflow if negative) to the flow region, m_x and m_y are terms to account for the momentum flux transfer associated with the inflow (or outflow) to the flow region being considered and can be computed from $m_x = uV^*/2$ and $m_y = vV^*/2$, $S_{ox} = -\partial z/\partial x$, $S_{oy} = -\partial z/\partial y$, and z is the vertical distance from a horizontal plane to the bottom confining surface of the flow, and two-dimensional versions of Manning's equations are used to define the slopes of the energy lines as follows:

$$S_{fx}=\frac{n^2u\sqrt{u^2+v^2}}{C_u^2Y^{4/3}} \tag{7.108}$$

and

$$S_{fy}=\frac{n^2v\sqrt{u^2+v^2}}{C_u^2Y^{4/3}} \tag{7.109}$$

First let us examine the continuity, Equation 7.105, as an extension of the one-dimensional continuity, Equation 6.1. Note that the dimensions of the terms in Equation 6.1 are L^2/t and apply across the channel. An equivalent two-dimensional continuity equation would not include the channel width and therefore its dimensions would be L/t when used for two-dimensional flows. Visualize dividing Equation 6.1 by a width (i.e., T = b for a rectangular section) thus replacing Q by q and q* by V*. The flow rate per unit width in the x-direction is uY and that in the y-direction is vY. The first term $\partial Q/\partial x$ in Equation 6.1 become the two terms: $\partial(uY)/\partial x$ and $\partial(vY)/\partial y$, and the other two terms come directly from the remaining two terms in Equation 6.1.

The one-dimensional St. Venant equation of motion, Equation 6.2 is dimensionless when using either the Q-Y form of these equations or the V-Y form (Equation 6.4, Chapter 6). If these equations of motion are multiplied by the acceleration of gravity, which is the case when they are used in many applications, then their dimensions are L/t^2. The dimensionless Equations 6.2 and 6.4 come by dividing Newton's second law (with dimensions of force) by the fluid specific weight and a unit length in the direction of flow. Equations 7.106 and 7.107 are derived similarly but thereafter they have been multiplied by the acceleration of gravity and the depth of flow since no variation of the velocity components in the x- and y-directions is assumed to occur in the vertical direction. The dimensions of Equations 7.106 and 7.107 are thus L^2/t^2 or $(L/t)^2$. The terms involving the spatial derivatives with respect to x and y in Equations 7.106 and 7.107 come from the change in momentum flux terms ρQV in the direction of flow per unit width or the partial derivatives of: $g(\rho q_x u)/\gamma = u^2Y$, $g(\rho q_x v)/\gamma = uvY$, $g(\rho q_y u)/\gamma = vuY$, $g(\rho q_y v)/\gamma = v^2Y$. The terms $g\{\partial(Y^2/2)/\partial x\}$ and $g\{\partial(Y^2/2)/\partial y\}$ consist of the changes in hydrostatic pressure forces per unit width divided by the specific gravity, and finally multiplied by g. Finally, note that the terms $gY(S_{ox} - S_{fx})$ and $gY(S_{oy} - S_{fy})$ are equivalent to the term $gA(S_o - S_f)$ in the one-dimensional St. Venant equations where A is replaced by Y to apply per unit width rather than for the entire cross section of the channel flow.

Let us now intuitively justify the above two-dimensional Manning's equations. First we should recognize that Manning's equation empirically accounts for turbulent shears stresses (e.g., the dissipation of flow energy caused by fluid momentum transfer of the zero velocity fluid) on the channel wall to the moving fluid. For a very wide channel the Manning's equation is, $q = (C_u/n)Y^{5/3} S_f^{1/2}$, or $V = (C_u/n) Y^{2/3} S_f^{1/2}$, which is the equation applied per unit width. Solving for the slope of the energy line gives

$$S_f = \frac{n^2V^2}{C_u^2Y^{4/3}}$$

For the two-dimensional application of Manning's equation we want the component of the slope in the x and y directions which are: $S_{fx} = S_f\cos\Theta = S_f\, u/V$ and $S_{fy} = S_f\sin\Theta = S_f\, v/V$. Substitution of these last two relationships into the above equation for the slope of the energy line in the x and y directions results in Equations 7.108 and 7.109.

If Chezy's equation is to be used to describe the dissipation of energy then in place of Equations 7.108 and 7.109 to compute the components of the slope of the energy line in the x and y directions the following equations would apply:

$$S_{fx} = \frac{u\sqrt{u^2+v^2}}{C^2Y} \tag{7.110}$$

and

$$S_{fy} = \frac{v\sqrt{u^2+v^2}}{C^2Y} \tag{7.111}$$

in which C is now Chezy's coefficient defined in an earlier chapter rather than a different value depending upon whether SI or ES units are used.

The more general equations of motion are the Navier–Stokes equations that can be found in most fluid mechanics books. One would expect, therefore, that the above two dimensional St. Venant equation should be obtainable there from. The general continuity equation for an incompressible flow is

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0 \tag{7.112}$$

and the vector equation of motion is

$$\frac{DV}{Dt}=g-\frac{1}{\rho}\nabla p+\frac{\mu}{\rho}\nabla^2 V \tag{7.113}$$

Writing Equation 7.113 in the Cartesian xyz coordinate system with v substituted for μ/ρ results in the following three equations:

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}+\frac{\partial u}{\partial t}=g_x-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\nabla^2 u \tag{7.113a}$$

$$u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}+\frac{\partial v}{\partial t}=g_y-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\nabla^2 v \tag{7.113b}$$

$$u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}+\frac{\partial w}{\partial t}=g_z-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\nabla^2 w \tag{7.113c}$$

in which $\nabla^2 \equiv \frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}+\frac{\partial^2}{\partial z^2}$.

It should be noted that in Equations 7.105 through 7.111 that the velocity components u and v are average velocities through the depth, whereas the velocity components in the Navier–Stokes equations, Equations 7.112 and 7.113, that the velocities are point velocities. In other words the St. Venant equations do not account for any changes of velocity in the vertical direction z. Therefore if the Navier Stokes equations are to be simplified to obtain the shallow water equations, then first, Equation 7.113c will be discarded since we are not going to consider changes in the vertical, and second Equations 7.112, 7.113a, and 7.113b will be integrated through the depth of flow.

Starting with the continuity, Equation 7.105, the depth-averaged replaces the velocity component in the x and y directions in the Navier–Stokes equations with fluxes uY and vY in the x- and y-directions respectively. To account for the change in storage in this differential control volume with a base of (dx)(dy) and a height of Y, the change in depth with respect to time must be added. Also any lateral inflow per unit area (dx)(dy) needs to be accounted for and this is done through the term V*. For the equations of motion the shearing stresses involving the fluid viscosity and the pressure gradients are described by the slopes of the energy surfaces in the x and y directions. Also the above fluxes uY and vY replace the velocity components so that terms such as $\partial(u^2Y)/\partial x$ replace $u(\partial u/\partial x) + \partial u^2/\partial x^2$.

The above two-dimensional St. Venant equations, Equations 7.105 through 7.107 are the u-v-Y form of these equations. A form of these two-dimensional equations that is analogous to the one-dimensional Q-Y form of the equations can be obtained by substituting fluxes in the x- and y-directions for uY and vY, or by defining $q_x = uY$ and $q_y = vY$ the following q_x-q_y-Y set of equations can be obtained:

I. Continuity

$$\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial Y}{\partial t} + V^* = 0 \tag{7.114}$$

II. Equations of motion

(a) x-direction

$$\frac{\partial}{\partial x}\left(\frac{q_x^2}{Y}\right) + \frac{\partial}{\partial y}\left(\frac{q_x q_y}{Y}\right) + g\frac{\partial}{\partial x}\left(\frac{Y^2}{2}\right) + \frac{\partial q_x}{\partial t} = gn^2\frac{q_x\sqrt{q_x^2+q_y^2}}{C_u^2 Y^{7/3}} - \frac{q_x}{Y}\frac{V^*}{2} - gYS_{0x} = 0 \tag{7.115}$$

(b) y-direction

$$\frac{\partial}{\partial y}\left(\frac{q_y^2}{Y}\right) + \frac{\partial}{\partial x}\left(\frac{q_x q_y}{Y}\right) + g\frac{\partial}{\partial y}\left(\frac{Y^2}{2}\right) + \frac{\partial q_y}{\partial t} = gn^2\frac{q_x\sqrt{q_x^2+q_y^2}}{C_u^2 Y^{7/3}} - \frac{q_y}{Y}\frac{V^*}{2} - gYS_{0y} = 0 \tag{7.116}$$

PROBLEMS

7.1 Take the definition of the stage variable, $w = \int (gT/A)^{1/2}\, dy$ for any channel, and show that $w = 2c$ for a rectangular channel. Then demonstrate that Equation 7.4 becomes the characteristic equations in Chapter 6 that were used for unsteady flows in rectangular channels.

7.2 For a rectangular channel the stage variable w equals 2c (see Problem 7.1). Therefore only four simultaneous equations need to be solved for x, t, V, and Y at a regular grid point m defined at the intersection of C^+ and C^- negative characteristics. What are the four equations that replace Equations 7.13 through 7.17 for unsteady problems in rectangular channels.

7.3 Modify Subroutine REGUL so that it reduces the problem of solving regular points in a rectangular channel to dealing with only 4 simultaneous equations.

7.4 Take the modified equations from Problem 7.2 and show that if $g(S_o - S_f) = 0$, as was assumed in solving the problems in Chapter 6, that Equations 7.13 through 7.16 define the values of x_m and t_m for straight line characteristics with inverse slopes of $v + c$ and $v - c$, respectively. In other words with $g(S_o - S_f) = 0$, uniform flow occurs initially.

7.5 Given a rectangular channel with a bottom with of $b = 10$ ft, and Manning's $n = 0.014$, and a bottom slope $S_o = 0.0012$. Assume that the flow rate in this channel is $Q = 200$ cfs, and that at positions $x_L = 600$ ft, and $x_R = 900$ ft at time equal 0 that the depths at these two positions are: $Y_L = 4.4$ ft and $Y_R = 4.8$ ft, and solve the five equations to determine the following five variables at a regular point m, x_m, t_m, c_m, V_m, and Y_m.

7.6 Given a trapezoidal channel with a bottom width of $b = 10$ ft and a side slope $m = 1.8$, and Manning's $n = 0.014$ and a bottom slope $S_o = 0.0012$. Assume that the flow rate in this channel is $Q = 200$ cfs, and that at positions $x_L = 600$ ft, and $x_R = 900$ ft at time equal 0 that the depths at these two positions are: $Y_L = 4.4$ ft and $Y_R = 4.8$ ft, and solve the five equations to determine the following five variables at a regular point m, x_m, t_m, c_m, V_m, and Y_m (also obtain the corresponding value for the stage variable w at point m). Repeat the problem for a flow rate $Q = 300$ cfs.

7.7 Subroutine REGUL contains the array W(I) for the stage variable w, but it does not actually implement determining this variable except for the special case of a rectangular channel for which $w = 2c$. Modify this subroutine so that w is defined for a given trapezoidal channel by generating a table of values such as given in Chapter 6, and then evaluates w whenever needed by appropriately interpolating from this table.

7.8 Modify the boundary subroutine BOUNDY so that it also allows for the stage variable to be evaluated by the table look-up technique described in the previous problem.

7.9 Verify the initial condition used in Example Problem 7.1 by solving the appropriate steady-state problem.

7.10 Obtain the initial conditions for Example Problem 7.2.

7.11 In the solution to Example Problem 7.1 the depths reported at the length $L = 1500$ ft are at the end of the wider rectangular channel with a bottom width $b = 10$ ft. Compute the corresponding depths just upstream and downstream from the gate, where $b = 8$ ft, for each of the times for which one of the solution grid points falls on the downstream boundary with $x = L = 1500$ ft. Also make up a table using a 20 s time increment that gives the gate's height that will produce the t versus Y specified values for this downstream boundary condition. (Assume that the loss coefficient K_L applies for the transition, and that no local loss is due to the gate, i.e., the specific energy upstream and downstream from the gate are equal, $E_u = E_d$.)

7.12 Repeat the previous problem except do this for Example Problem 7.2, that has a trapezoidal with $b_1 = 10$ ft, and $m_1 = 1.5$.

7.13 Example Problem 7.1 deals with a situation in which the depth at the downstream end of the channel decreases with time. Solve this same problem in which the flow rate increases with time at the downstream end of the channel according to the values in the table below.

t (s)	0	20	60	100	140	180	300
Q (cfs)	206.72	210	230	240	250	260	270

7.14 In Example Problem 7.1 the depth at the downstream end of the 10 ft wide rectangular channel was specified to vary with time. Add a fourth possible boundary condition to the program that solves unsteady flow in rectangular channel in which the depth downstream from the gate (i.e., the position of the gate times its contraction coefficient) changes with time according to values given in a table. Then use this program to solve Example Problem 7.1 the depth of water downstream from the gate increases by 0.075 ft every 30 s, or according to the values in the table below.

t (s)	0	30	60	90	120	150	180	210	240	270	310
Y_d (ft)	1.5	1.575	1.65	1.725	1.8	1.875	1.95	2.025	2.05	2.125	2.2

7.15 As in the previous problem add the additional boundary condition to the downstream boundary that allows for the depth downstream from the gate to be specified for trapezoidal channels. Then solve Example Problem 7.2 with the same time dependent depth Y_d as given in the table of the previous problem.

7.16 A reservoir with a constant water surface of H = 5 ft above the channel bottom supplies water to a trapezoidal channel with b = 10 ft, m = 1.5, n = 0.013 and S_o = 0.0006. The loss coefficient at the channel entrance is K_e = 0.10. At a position 2000 ft downstream a gate exists to control the flow that is 10 ft wide in a rectangular channel. At the position of the gate the bottom of the channel drops 1 ft in elevation. Initially the gate is 2.5 ft above the bottom of the rectangular channel, and the gate has a contraction coefficient C_c = 0.6. What is the flow rate past the gate, what is the depth immediate upstream from the gate, and what is the depth at the beginning of the channel with this gate setting as the initial condition for the unsteady flow problem. At time zero the depth upstream of the gate before the transition is decreased at a rate dY_2/dt = 0.01 ft/s for 200 s and held constant thereafter. Obtain the intersection of the C^+ and C^- characteristics for 185 grid points.

7.17 Modify the program (and the subroutines it calls on) that solves unsteady problem based on the C^+ and C^- characteristics so that rather than having the celerity c as one of the unknowns being solved at a regular grid point the stage variable w is one of the five unknowns.

7.18 Since w and c are functions of the depth Y it is possible to reduce the number of unknowns at regular grid points to 4 rather than 5. Modify the program (and the subroutines it calls on) that solves unsteady problem based on the C^+ and C^- characteristics so that only four variables are considered unknown at regular grid points.

7.19 Modify the program (and the subroutines it calls on) that solves unsteady problems based on the C^+ and C^- characteristics so that it will solve unsteady problems in circular channels.

7.20 A 3 m diameter pipe takes water from a reservoir whose water surface is 2 m above the bottom of the pipe at its entrance. The entrance loss coefficient is K_e = 0.12. At a position 1000 m downstream the pipe discharges into a reservoir whose water surface elevation is 2.5 m above the bottom of the pipe, initially. The pipe has a roughness coefficient for Manning's equation of n = 0.012, and has a bottom slope S_o = 0.0006. At time zero the water surface in the downstream reservoir begins to drop at a rate dY/dt = 0.02 ft/s for 50 s and then remains constant again. Determine how the depth, flow rate, and velocity vary throughout the channel as a function of time for 300 s.

7.21 The numbering of the points of intersection of the C^+ and C^- characteristics on Figure 7.1 begins at the downstream end of the xt plane since the origin is at the beginning of the channel, and the x axis is in the direction of flow. This arrangement was used to provide compatibility with solutions of GVF problems as the initial conditions for unsteady problems. Place the origin of the xt plane at the downstream end of the channel with the x axis opposite to the direction of flow. Now what type of numbering system would you use.

7.22 A rectangular channel with a bottom width b = 8 ft, a Manning's n = 0.012 and a bottom slope S_0 = 0.001 is supplied its water from a reservoir with a water surface elevation of H = 3.5 ft. The entrance loss coefficient K_e = 0.10. The channel is L = 8000 ft long, and it discharges into a pond whose water surface is initially at H_2 = 4.5 ft above the channel bottom. Suddenly the water surface elevation in this pond begins to drop at a rate dH_2/dt = 0.01 ft/s for a time t = 300 s, and thereafter its water level remains constant. Using 195 grid points obtain a solution to the resulting unsteady flow. From this solution, determine the following volumes of water that are taken out of channel storage: (1) The amount before the flow rate from the upstream reservoir begins to increase, (2) the amount after this time until the ending grid point on the L = 8000 ft end of the channel is number 184, and (3) the total of these two amounts.

7.23 Assume that the specific energy is specified to change with time at the downstream end of the channel. Modify the program UNSCHAR.FOR for rectangular channels and UNSCHWN.FOR for trapezoidal channels so that they will handle having the specific energy specified at the downstream end of the channel as a function of time in a table of values. With these modified programs obtain solutions to Example Problems 7.1 and 7.2 with (1) the specific energy constant at the downstream end of the channel and equal to that from the initial gradually varied flow solution, (2) the specific energy decreases at a rate $dE/dt = -0.005$ ft/s, and (3) the specific energy increases at a rate $dE/dt = 0.002$ ft/s.

7.24 Assume that the initial condition to an unsteady problem consists of a steady state gradually varied flow situation, but that the time-dependent flow conditions in the channel is caused by changing the conditions at the upstream end of the channel. Develop the numbering scheme you would use in solving this problem in the xt plane. Modify the driver program that implements a solution based on this numbering of points. Using this program solve Example Problem 7.1 in which the reservoir head H varies with time according to:

Time (s)	0	20	60	100	140	180	300
H (ft)	4.5	4.48	4.46	4.44	4.42	4.40	4.40

7.25 Repeat the previous problem except make the program applicable to a trapezoidal channel by using the stage variable w. Solve Example Problem 7.2 with the upstream reservoir head varying according to the table given in the previous problem.

7.26 By solving the equation $dx/dt = V - c$ that defines the slope of negative characteristics determine the position of the wave (position where the effect of changing the depth) as a function of time in Example Problem 7.1 that is created by changing the depth at the downstream end of this channel that initially contains a uniform flow. In other words obtain a solution of values with time t in the first column that gives the corresponding values of x in the second column. If you want you can let x point upstream with is origin at the downstream end of this 1500 ft long channel.

7.27 Solve Problem 7.16 in which the flow rate from the reservoir Q varies according to:

Time (s)	0	10	20	40	100	200	300	400	1000
Flow rate (cfs)	253.88	260.0	270.0	280.0	290.0	300.0	310.0	310.0	310.0

7.28 Verify the four interpolation equations that use a second degree polynomial (second order) that are used to evaluate the velocities and depths at the intermediate points L and R on the kth time line. These equations (Equations 7.28a through 7.31a) are used to replace the linear interpolation Equations 7.28 through 7.31. To verify these equations you might start with Lagrange's formula, or some other interpolation formula such as the Newton forward, or backward formula.

7.29 The solution to Example Problem 7.3 is given by plots of Y, V, and Q versus the position x in the channel for each of the 20 s time steps used in obtaining the solution, but the actual tables of these values is not given. Obtain these tables of values that provide the time-dependent solution to this problem.

7.30 Initially a flow rate of Q = 400 cfs is passing through a gate that supplies water from a reservoir. At the gate the channel is rectangular with a width of $b_1 = 12$ ft, but immediately behind the gate the channel becomes trapezoidal with $b_2 = 12$ ft, m = 1.3, n = 0.013 and $S_0 = 0.0005$. This trapezoidal channel is 8000 ft long and discharges into a reservoir whose initial water surface is 6 ft. The gate is submerged so that the velocity head of the jet below the gate is dissipated, and as the channel discharges into the reservoir at it downstream end the velocity head is also dissipated. The unsteady flow conditions are caused by a downstream effect. Consider three downstream changes: (a) the downstream reservoir water surface increases from the initial 6 ft elevation to 6.5 ft in 300 s, and then to 7.0 ft in the next 300 s. (b) the downstream reservoir water surface decreases from the initial 6 ft elevation to 5.5 ft, in 300 s, and then to 5.0 ft in the next 300 s, and (c) the flow rate at the end of the channel in increased from 400 to 500 cfs in 300 s, and then to 600 cfs in the next 300 s. Use time steps of 10 s, and space increments of 200 ft, simulate the flow conditions throughout the channel for 480 s using the Hartree method.

7.31 In the previous problem, the head H that supplies water to the channel at its upstream end varies, so that H decreases by 0.2 ft each 300 s. Obtain solutions to the three cases of the previous problem with this additional source of unsteady flow at the channels upstream end.

7.32 Unsteady flow occurs in a trapezoidal channel that is L = 8000 ft long, has a bottom width b = 12 ft, a side slope m = 1.3, (as in Problem 7.30), and a bottom slope $S_o = 0.0002$. Initially a steady flow of Q = 400 cfs is occurring, and the channel discharges into a lake whose water surface elevation is 8 ft. The downstream lake level begins to drop at time zero so that the flow rate into the lake increases at a rate of dQ/dt = 100/300 = 0.333333 cfs/s. The channel is supplied by a constant head reservoir at the upstream end, and the water comes into the channel from a vertical (sluice) gate so that the jet below the gate has an initial depth of $Y_2 = C_cY_G = 2.67$ ft. The gate is submerged, and the width of the gate and the rectangular channel immediately downstream from the gate is $b_2 = 12$ ft. Assume there is no energy loss in the transition to the downstream trapezoidal channel. At time zero in addition to the flow rate into the downstream reservoir increasing, the gate is slowly lowered to that $dY_2/dt = -0.17/300 = -0.00056667$ ft/s. Using time steps of 10 s and a space increment of 200 ft simulate the unsteady flow using the Hartree method over a 480 s = 8 min time period. In solving the problem assume the special energy and momentum equations given in Chapter 5 apply across the submerged gate.

7.33 Modify Program HARTREE so that three gates with the same width exist at the downstream end of the channel each with a possible different setting. The positions of these gates can be changed with time to cause the unsteady flow in the channel. With this program solve Example Problem 7.3, assuming that all three gates are initially set at the same position so that they produce the same initial flow conditions as used in Example Problem 7.3. The combined width of the 3 gates equals the 5 m width of the downstream rectangular channel of Example Problem 7.3, so that each gate has a width $b_G = 5/3 = 1.6667$ m. Obtain two unsteady solutions: (1) in which the middle gate is raised so its free flowing downstream depth increases from the initial depth to 2.5 m in 400 s and remains at this setting for subsequent times, and the other two gates do not have their setting changed, and (2) both gates 2 and 3 have their setting changed so that the new depth downstream from them is 2.5 m in 400 s and remains at this depth for subsequent times, and gate one's setting does not change.

7.34 The programming technique used in Program HARTREE in generating the stage variable w for trapezoidal channel is to generate two arrays of values; one for the dimensionless depth Y′, or YST(400) and the other for w′ or W(400), and then to interpolate w′ by finding the two table entries of YST between which the given Y′ occurs. Since, however, the increments between consecutive YST values of the table are constant, the position in the table of w′ can be evaluated without storing the values of Y′ corresponding to w′. Modify Program HARTREE so that the array YST is eliminated.

7.35 With the modified HARTREE program of the previous problem solve an unsteady flow created in the trapezoidal channel of Example Problem 7.3 in which the depth at the downstream end of the channel is decreased at a rate of dY/dt = 0.01 ft/s for 400 s. Obtain the unsteady solution using a 10 s time steps, and carry out the solution for 400 s.

7.36 The Program HARTREE listed in the text contains statements within the subroutine SOLVE that solve the linear algebra problem by using Gaussian elimination in implementing the NEWTON method. Modify Program HARTREE (or the program from the Problem 7.34 that eliminates the array YST) so that rather than using built in statements to solve the linear system of equation at each grid point it calls on a linear algebra solver such as SOLVEQ.

7.37 Using the methods described under Section 7.5 for the Hartree solution method solve the unsteady flow in the channel of Example Problem 7.3 if the depth at the downstream end of this channel decreases by 2 ft in a time period of 200 s. In solving the problem use a time step of 10 s.

7.38 Repeat the previous problem except rather than using the iterative technique, use the technique described under Section 7.6.

7.39 Replace the depth Y by the celerity c as the variables in the list of unknowns that are solve at each grid point in the Hartree method. Using these unknowns develop a computer solution to unsteady problems in rectangular channels. (Note for a rectangular channel that the inverse slope of both the C^+ and C^- characteristics are given by $V \pm c$ and the derivatives that apply along these characteristics $d(V \pm 2c)/dt = g(S_o - S_f)$ involve c, and therefore it makes sense to consider the velocity V and the celerity c the basic dependent variables.) Use this computer solution capability to solve an unsteady flow in a 1500 ft long rectangular channel with b = 10 ft, n = 0.012, and $S_o = 0.0015$ which is supplies by a reservoir with a water surface elevation H = 4.5 ft and an entrance loss coefficient of $K_e = 0.15$. The initial flow rate Q = 195.53 cfs, and with a downstream gate controlling the initial depths at a increment of $\Delta x = 75$ ft, the depth from the downstream to the upstream end of the channel are (in ft):

6.081	5.976	5.872	5.768	5.664	5.561	5.458	5.356	5.254	5.153	5.052	4.952
4.853	4.755	4.657	4.560	4.464	4.369	4.276	4.183	4.092			

The unsteady condition is due to the downstream depth dropping to 5.9 ft in 20 s, to 5.7 ft in 60 s, to 5.5 ft in 100 s, to 5.3 ft in 140 s, to 5.1 ft in 180 s and remains constant at 5.1 ft thereafter. Use 5 s time steps in the solution and carry out the solution to 400 s.

7.40 Expand the capabilities of the computer solution you developed in the previous problem to also accommodate trapezoidal channels. This program should use c's to replace Y's as basic unknowns. Use this program to solve Example Problem 7.3.

7.41 In Example Problem 7.3 the solution of the specific energy equation across the downstream gate revealed that the specified flow rates exceeded the amount that could be passed by the gate if the depths upstream from the gate were taken at several time steps. Assume that a pipe is attached to the channel just upstream from the gate that discharges some of the water. What must be the flow rate in this pipe if the maximum discharge occurs past the gate that is possible for the depth given from the unsteady solution.

7.42 Modify the computer program HARTREE.FOR that was used to solve Example Problem 7.3 so that the unsteady solution will agree with what the gate is capable of passing.

7.43 Starting with Equation 7.62a derive the gradually varied flow equation for steady state conditions. To accomplish this first eliminate the time dependent terms, next assume that there is no lateral inflow (or outflow) and therefore Q is constant throughout the length of the channel.

7.44 Equation 7.70 involves the change in wetted perimeter with depth, $\partial P/\partial Y$. Evaluate this quantity for both a trapezoidal and a circular channel. Describe how this quantity would be evaluated for a irregular cross section in which a table of values of channel properties was used.

7.45 Verify the derivatives given in Equations 7.66, 7.67, and 7.68.

7.46 The depth is specified as a function of time ($Y_b(t)$ = given) as the downstream boundary to define an unsteady channel flow. One means for handling this condition is given in the second equation given before Equation 7.71 in which the value for the rate of change of depth with time $\partial Y_b/\partial Y$ is used in the continuity equation. Equation 7.72 on the other hand uses the known flow rate in the boundary finite difference equation. Develop a similar boundary finite difference equation to Equation 7.72 that applies if $Y_b(t)$ is known at the end of a channel.

7.47 Repeat the previous problem with the exception that $Y_b(t)$ is known at the upstream end of a channel.

7.48 A trapezoidal channel with b = 4 m, m = 1.5, n = 0.013, and $S_0 = 0.0012$ is supplied by a reservoir with a constant water surface that is 2.5 m above the channel bottom at it entrance. The entrance loss coefficient is $K_e = 0.14$. At a distance 600 m downstream from the entrance there are sluice gates in the channel to control the flow rate. A transition immediately upstream from the gates reduces the channel to 3.5 m wide rectangular section. Initially the gates are set at a position 0.5 m above the channel bottom. The contraction coefficient for the gates is .6, and their minor loss coefficient is $K_L = 0.09$. Determine the steady state flow rate in the channel with this gate setting, and the depth throughout the channel. The gate is slowly closed so as to increase the depth in the trapezoidal channel just upstream from it at a rate of dY/dt = 0.001 m/s for 50 s, and then is held constant to 100 s, and then raises again to drop the depth to the original depth at 200 s. Solve this unsteady problem.

7.49 Everything is the same as in Example Problem 7.3 except the channel at the position of the gate has a width of 6 ft instead of 5 ft, but is of rectangular shape at the gate. For the time and flow rates used in this example problem determine the positions that the gate should be set at for each time step.

7.50 Starting at time t = 0 the gates in the channel described in Problem 7.16 are slowly raised at a rate $dY_G/dt = 0.005$ m/s until the gates are 1.0 m above the channel bottom. Simulate the unsteady flow in the channel during this time period and for 100 s after the gates have stopped moving.

7.51 Modify the Hartree computer program so that it is designed to only solve unsteady problems in rectangular channels. Also add code into this program to write a warning message and reduce the time step, if the specified time step is larger than appropriate.

7.52 Obtain the solution to Example Problem 7.3 except increase the flow rate past the gate to only 70 m^3/s and obtain this solution for 800 s.

7.53 Obtain a solution to Example Problem 7.3 in which the flow rate at the end of the channel varies according to the following time schedule up to 400 s:

Time (s)	0	20	40	120	160	200	400
Flow rate (m^3/s)	34.8	65	110	150	175	180	0

and after 400 s the flow at the end of the channel is reversed so that negatives flow rates begins, e.g., water is pumped into the channel. The rate at which this negative flow rate increases at the downstream end of the channel is linear with time so that at 800 s the flow rate is −40 m^3/s. Plot the depth, velocity and flow rate throughout the channel for several time step.

7.54 Solve Example Problem 7.3 with the depth at the downstream end of the channel being specified as a function of time rather than the flow rate. Let the depth at the downstream end of the channel vary from the beginning 7.30 to 4.0 m over a 800 s time period. Plot the depth, velocity, and flow rate throughout the channel for several time step, and determine what the positions of the gate should be as a function of time to achieve these time dependent depths. Also determine the volume of water taken out of channel storage to satisfy the increased flow rate at the downstream end. Determine this volume by two different methods, and compare the results. First, integrate the flow rate out of the end of the channel with time and subtract from this the integrated flow rate entering at the upstream end of the channel. Second, integrate the change in depth across the entire channel as a function of time. (This latter method applies only until the effect of increasing the flow rate at the downstream channel propagates up to its entrance.)

7.55 Solve Example Problem 7.3 with the velocity at the downstream end of the channel being specified as a function of time rather than the flow rate. Let the velocity at the downstream end of the channel vary from V = 0.28 to 1.5 m/s over a 800 s time period. Plot the depth, velocity, and flow rate throughout the channel for several time step, and determine what the positions of the gate should be as a function of time to achieve these time dependent velocities.

7.56 A 12 ft wide rectangular channel with n = 0.013 and $S_o = 0.0005$ receives water from a reservoir whose water surface is 4.0 ft above the bottom of the channel and the entrance loss coefficient is $K_e = 0.08$. At a position 9000 ft downstream there is a gate in the channel that initially is set to cause the depth to be $Y_2 = 4.5$ ft. Determine the depths, velocities, and flow rates throughout the channel in 30 s time increments if the flow rate is increased to 250 cfs in 180 s and then linearly decrease to the initial flow rate at time 900 s.

7.57 Develop the downstream boundary conditions for the second-order "Hartree method" for a channel discharging into a downstream reservoir, and with an upstream boundary condition in which the flow rate is specified as a function of time.

7.58 In Section 7.7, an example problem was solved using a modified HARTREE program that specified both zero velocity at the beginning and end of the channel. Modify the HARTREE. FOR program so that it implements a zero velocity boundary condition at ends, and thereafter verify, one (or more) of the solution discussed in this section.

7.59 In Section 7.7, an example problem was solved in which the initial condition specified a cosine function for the depth, i.e., $Y = Y_o + \Delta Y \cos(\pi x/L)$. Modify this initial condition to be the sine function, $Y = Y_o - \Delta Y/2 + \Delta Y \sin(\pi x/L)$ so that the initial water depths at the beginnings and end of the channel are 1/2 an ΔY below the equilibrium depth and the depth is $\Delta Y/2$ above the equilibrium depth at the midpoint of the channel. This will require a slight modification of the program of the previous problem. Then obtain a solution for the unsteady frictional flow in a 10 ft wide rectangular channel with a zero bottom slope that has closed gates at a distance 1000 ft apart, and an equilibrium depth of $Y_o = 4$ ft. Solve the problem for $\Delta Y = 0.5$ and 0.05 ft.

7.60 In Section 7.7, an example problem was solved in which the initial condition specified a cosine function for the depth, i.e., $Y = Y_o + \Delta Y \cos(\pi x/L)$. Modify this initial condition to be the cosine function over a complete cycle of 0 to 2π radians. Then solve the same example problem. Explain any differences you see in the two solutions. (To obtain this solution will require that you make a slight modification to HARTREE program you modified to handle zero velocities at both ends of the channel.)

7.61 Solve the same problem given in Section 7.7 except do not assume that the fluid is frictionless, i.e., the channel has a Manning's roughness coefficient, n = 0.015. (If the program you modified still contains the statements, etc. that compute S_f you can use that program to solve this problem, otherwise you may need to modify it.)

7.62 Write out the system of equations in matrix form as given in Figure 7.5 for a problem in which $Y_b(t)$ is known at the downstream end of a channel as described in Problem 7.46.

7.63 Repeat the previous problem with the exception that $Y_b(t)$ is known at the upstream end of a channel.

7.64 Modify program IMPLICIT that implements the **direct implicit finite differencing** so that it will handle as the downstream boundary condition the depth being specified as a function of time. With this modified program solve for the unsteady flow for 400 s in a trapezoidal channel with b = 5 m, m = 1.5, n = 0.0135, $S_o = 0.0012$ and a length L = 1000 m. There is a sluice gate in a rectangular portion at the very end of the channel where it is 4.5 m wide. The gate has a contraction coefficient, $C_c = 0.65$. The channel is supplied by a reservoir with a water surface elevation H = 2.5 m with a loss coefficient of $K_e = 0.12$. At the transition and gate assume the loss coefficient is $K_L = 0.10$. For a long time the gate has been at a height of 0.8 m, and at t = 0 the gate is raised so that the depth in at the end of the trapezoidal channel varies as given in the table below:

Time (s)	0	20	40	60	80	100	120	140	160	180	200	220	240	260	etc.
Depth (m)	3.647	3.6	3.5	3.4	3.3	3.2	3.1	3.0	2.9	2.8	2.7	2.6	2.5	2.5	2.5

7.65 Write out the system of equations in matrix form as given in Figure 7.5 that describes the unsteady flow in a channel that discharges into a lake at its downstream end whose water surface elevation remains constant. At the upstream end of this channel the flow rate is controlled as a known function of time, e.g., $Q(0, t) = Q_b(t)$ is given.

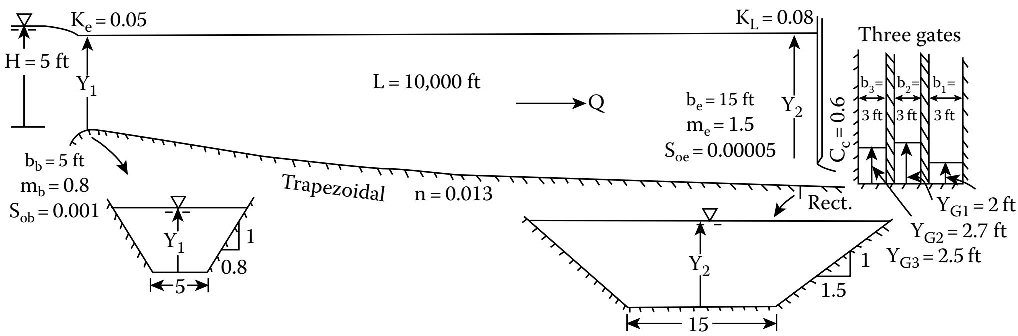
7.66 At the downstream end of the channel in Problem 7.64 the flow rate is increased as a function of time, rather than the depth decreased. This rate of change of flow rate is as follows: (1) at 20 s $Q_n = 33\,m^3/s$ (2) thereafter the flow rate increase by $2.5\,m^3/s$ every 50 s (or $\Delta Q = 0.05\,m^3/s^2$) until the flow rate is $Q_n = 72\,m^3/s$, and thereafter remains constant at $72\,m^3/s$. Use a 50 s time interval between consecutive solutions.

7.67 Solve Problem 7.64 except that instead of the depth at the downstream end of the channel decreasing with time, this depth increases with time. The depth increases is as follows: (1) it increases to 3.64 m from the steady-state downstream depth in 20 s, and (2) then increases by 0.01 m every 20 s thereafter until the depth is 3.7 m.

7.68 Resolve Problem 7.48 using the "direct implicit finite differencing method."

7.69 Solve Example Problem 7.3 using the "direct implicit finite differencing" method. First solve this problem using 20 s time steps as used in the example problem. Thereafter solve the problem using 40 s time steps, but in using 40 s as the time step assume that the changes in downstream flow rate also are extended so that it takes twice as long for changes in Q_n to occur, i.e., the flow rate increases from the initial value of $34.82\,m^3/s$ to $65\,m^3/s$ in t = 80 s, rather than in 40 s, etc.

7.70 A 10,000 ft long channel has a linearly varying bottom width b, side slope m, and bottom slope S_o. Manning's n = 0.013 is constant throughout the entire channel length. At the channel entrance $b_b = 5$ ft, $m_b = 0.8$, and $S_{ob} = 0.001$. At the end of the trapezoidal channel $b_e = 15$ ft, $m_e = 1.5$, and $S_{oe} = 0.00005$. There are three gates in a rectangular section at the end of the channel. Each gate is 3 ft wide, and initially their heights above the channel bottom are: $Y_{G1} = 2.0$ ft, $Y_{G2} = 2.7$ ft, and $Y_{G3} = 2.5$ ft. For all gates the contraction coefficient is $C_c = 0.6$, and the loss coefficient across the downstream transition and through the gates is $K_L = 0.08$. The water supply comes from a constant head reservoir with H = 5 ft, and the entrance loss coefficient is $K_e = 0.05$. Do the following: (1) Solve the steady-state problem. (2) Using the results from (1) for the initial condition, obtain an unsteady solution using the "direct implicit finite differencing method" in which the flow rate at the downstream end of the channel equal 240 cfs 30 s after the gates are first raised (at t = 0 s the steady-state flow rate for task 1 occurs), and increases by 10 cfs each 30 s thereafter until the flow rate equals 500 cfs, and remains constant thereafter. (3) Determine what positions the gates should have so that first gate 1 is raised until completely opened, then gate 2 is raised until completely opened, and then gate 3 is opened. At what time is it not possible to pass the above specified flow rate through the three gates? (4) Assume the gates are completely open, but that this downstream rectangular section of 9 ft width restricts the flow, but downstream from the gates there is nothing to limit the flow rate, and solve for the maximum steady-state flow rate. Why is this Q_{max} less than some of the unsteady flow rates accommodated by opening the gates as determined by task 2 and 3.



7.71 Repeat the previous problem except make all three gates 4 ft wide. Also initially the gates are set at the following heights: $Y_{G1} = 1.5$ ft, $Y_{G2} = 2.5$ ft, and $Y_{G3} = 2.5$ ft. For the unsteady solution the specified flow rate start with 250 cfs for the first 30 s, and then increase by 10 cfs for each 30 s thereafter, as in the previous problem, until the flow rate is 500 cfs, and thereafter Q is constant.

7.72 Less numerical noise would occur at the upstream end of a channel supplied by a constant head reservoir if the flow rate and depth were maintained constant until the effect of a downstream transient reached to the upstream end of the channel. Modify the computer program whose listing is given so that Y_1 and Q_1 are keep constant until just before the effects of the unsteady flow reach to the entrance of the channel.

7.73 A 2000 m long trapezoidal channel consists of two different sizes. The upstream 600 m length has a bottom width $b_1 = 8$ m, a side slope $m_1 = 1.7$, Manning's $n_1 = 0.013$ and a bottom slope $S_{o1} = 0.00015$, the downstream 1200 m length of channel is smaller and steep and has a bottom width $b_2 = 6$ m, a side slope $m_2 = 1.2$, Manning's $n_2 = 0.015$ and a bottom slope $S_{o2} = 0.001$. This channel system is feed by a constant head reservoir with H = 3 m, and $K_e = 0.05$. A rectangular vertical gate exists at the downstream end of the channel with a width of 6 m, and it produces a depth downstream of it of 0.5 m. The loss coefficient through the gate is $K_L = 0.06$. The channel has been flowing under steady-state conditions for some time, when suddenly the gate is slowly raised so that the flow rate is increased to 25 m^3/s at 30 s, and increase at a rate of 0.5 m^3/s each 30 s thereafter until the flow rate past the gate is 35 m^3/s, and then it is held constant. Using increments of 50 m across this channel system and 30 s time steps solve the unsteady for 900 s using the "direct implicit finite differencing method."

7.74 Assume that the initial depths in the previous problem consists of the depths that were determine as the steady-state values plus $0.5\sin(x\pi/L)$. In other words something has caused the depth at the center of the channel to be 0.5 m larger than that for the initial condition of the previous problem, but the depths at both ends of the channel are as in the previous problem at t = 0. Solve the problem using the same $\Delta = 50$ m and $\Delta t = 30$ s for a total time of 900 s as in the previous problem. Repeat the solution except that $0.5\sin(x\pi/L)$ is subtracted from the steady-state depths.

7.75 Modify the computer program listing that is under the name IMPLICIT.FOR so that it reads in a stage-discharge relationship as the upstream boundary condition rather than satisfying the entrance energy equation.

7.76 In obtaining the solution to Example Problem 7.4, the solution to a steady state gradually varied flow was used as the initial condition, and the flow rate and depths from this solution are read as input to the computer program used to solve the problem. Make up a table that contains a line for each station used in the numerical solution to this problem that shows that the magnitude of $(1 - F_r^2)(Y_{i+1} - Y_{i-1})$ equals the values of $2\Delta x(S_f - S_o)$, and therefore the initial condition satisfies the difference equations at t = 0 that are used to obtain the unsteady solution.

7.77 Using Taylor's series verify that $(Q_i^{k+1} - Q_i^k)/\Delta t$ is a second order approximation of $\partial Q/\partial t$ evaluated at k + ½. Note this approximation is based on an increment of $\Delta t/2$.

7.78 Equation 7.73, that evaluates c_1 for Equation 7.72 in the constant head (H = constant) upstream boundary condition for the "direct implicit finite differencing method," assumes that the entrance loss coefficient K_e is zero. What will c_1 be if K_e is not zero?

7.79 In the development of Equation 7.73 to obtain c_1 for Equation 7.72 the head H of the reservoir is assumed to be constant for the upstream boundary condition for the "direct implicit finite differencing method." Show that if H varies with time (H = f(t)) that the same approach, but that notes the differential dH = (dH/dY)dY = [(dH/dt)/(dY/dt)]dY results in a nonlinear equation. In other words c_1 contains $(Y_1^{k+1} - Y_1^k)$ so that Q_1^{k+1} and Y_1^{k+1} cannot be separated with coefficient multipliers.

7.80 Modify the Program IMPLICIT that uses the "direct implicit finite differencing method" to handle problems in which the upstream boundary condition may specify how the flow rate entering the channel varies with time, i.e., $Q_1(t)$ is given. Use this program to solve the channel in Problem 7.64 with the initial condition as given in that problem, and the entering flow rate starting with the steady-state flow rate and then increasing to 35 m³/s in 20 s, and increases of ΔQ = 5 m³/s each 20 s thereafter until the incoming flow rate is 60 m³/s and then held constant at this value thereafter. The flow rate at the downstream end of the channel is specified to start with the steady-state value and increase to 33 m³/s in the first 20 s, and increase by ΔQ = 2.5 m³/s for each 20 s thereafter until it equals 64 m³/s and remains constant thereafter.

7.81 Modify the Program IMPLICIT that uses the "direct implicit finite differencing method" to handle problems in which the upstream boundary condition may specify how the depth at the beginning of the channel varies with time, i.e., $Y_1(t)$ is given. Then solve the same problem as in the previous problem with the exception that the depth at the upstream end is specified to increase in time as follows: (1) for the first 20 s the depth increases to 2.5 m from the steady-state value, and (2) for each 20 s thereafter it increase by ΔY = 0.05 m until Y_1 = 2.8 m and then it remains constant at this value.

7.82 Use Program IMPLICBC that utilizes the SOR method to solve the channel in Problem 7.70. The same initial condition exist as in Problem 7.70, and the flow rate is increased at the downstream end of the channel to 500 cfs as specified in that problem. Solve the problem using two separate upstream conditions: (1) The depth remains constant at the steady-state value for the time of the simulation, and (2) the upstream depth increases to 4.3 ft in the first 30 s, and thereafter increases an additional 0.0025 ft every 30 s until the upstream depth equals 4.5 ft, and then it remains constant thereafter.

7.83 Solve the channel in Example Problem 7.3 using the SOR iterative implicit method of solution. For this solution the initial condition is the same as in the example problem, the depth of water at the beginning of the channel is the steady-state value of 2.71 ft, and at the downstream end of the channel the flow rate is increased according to the following: t = 30 s, Q_n = 38.7 m³/s, t = 60 s, Q_n = 60.0 m³/s, t = 90 s, Q_n = 68.3 m³/s, and: t = 120 s, Q_n = 75.0 m³/s. The flow rate at the end of the channel remains at 70 m³/s, thereafter. For a second case the flow rate is specified at the upstream end of the channel so that in 30 s it has increased from 34.82 to 35.0 m³/s, and thereafter it increases by 2.5 m³/s until Q_1 = 75 m³/s and remains at this amount thereafter. At the downstream end of the channel for this case the flow rate is the same as for case 1.

7.84 Solve the two different size channels of Problem 7.73 using the SOR iterative implicit method. Handle two cases. (1) the flow rate into the upstream end of the channel is identical to that at the downstream end of the channel, and these flow rates are as in Problem 73, i.e., are 25 m³/s in 30 s, and increase by 0.5 m³/s for each 30 s thereafter until it equals 35 m³/s. (2) the flow rate at the downstream end of the channel varies as in part (1) (and in Problem 7.73), but the upstream flow rate increases to 25.5 m^3/s in 30 s and thereafter increase by 1 m^3/s every 30 s until the flow rate is 45.5 m^3/s, i.e., twice as fast as the upstream Q.

7.85 In the ninth case solve in Example Problem 7.5 the reservoir head H = 2.49643 m is held constant through time while the depth at the downstream end of the channel is decreased to 2.5 m in 240 s. Solve another case for this channel in which the depth deceases the same at the downstream end, but the reservoir head at the upstream end of the channel increases to that after 20 s H = 2.5 m and it increases 0.05 m each 20 s thereafter until H = 3.0 m and is held constant thereafter.

7.86 The results from the solution to Example Problem 7.7 are only given in the text as plots showing how the depth, flow rate and velocity vary. Obtain the solution to this example problem.

7.87 Modify the program that you developed for Problem 7.64 that allows for either the depth as well as the flow rate to be specified at a function of time at the downstream end of the channel, but use the equations described in Section 7.8.2. This modified program should us the "direct implicit finite differencing method," or in other words set-up the linear system of equations, and then use linear algebra to solve this system. Test the program by resolving Problem 7.64 with the same downstream depths specified as in that problem. Then use this program to also solve the problem in which the flow rate is specified to increase at the downstream boundary such that at 20 s Q_n = 35 m^3/s and each 20 s thereafter increases an additional 5 m^3/s until the downstream flow rate is 60 m^3/s and remains constant thereafter.

7.88 Modify the program that you developed in the previous problem so that it will allow the following boundary conditions to be specified at the upstream end of the channel: (1) H = Constant (reservoir head, (2) $Y_1(t)$ is given, and (3) $Q_l(t)$ is given. In defining these upstream boundary conditions also use the equations given in Section 7.8.2, i.e., use Equation 7.79 with the four a's and f_4 defined below this equation. Then solve the channel of the previous problem for two additional cases in which the depths and flow rates change by the same amount at the upstream and downstream ends of the channel. The depth changes at the downstream end of the channel increases to 3.7 m (from the initial 3.637 m) in 20 s, and then increases by 0.05 m each 20 s thereafter until Y_n = 4.2 m and remains constant thereafter. At the upstream end of the channel the depth change is the same as at the downstream end of the channel. When solving the problem with both upstream and downstream flow rates changing by the same amount after 20 s it equals 35 m^3/s (from the initial 31.66 m^3/s), and increases by 5 m^3/s each 20 s thereafter until 60 m^3/s, and remains constant thereafter.

7.89 Modify the program (computer solution) that allowed either depths or flow rates to be specified at either the upstream or downstream boundary that you developed in the previous problem to also allow stage-discharge relationships to be allowed as either or both the upstream or downstream boundary conditions. To test out this program assume that a stage discharge relationship applies to the upstream boundary of the previous problem and that the flow rate into the channel will be according to the data in the table below. At the downstream end of the channel the depth increases from 2.477 to 4.2 m as in the previous problem.

Upstr. depth (m)	2.4733	2.55	2.65
Flow rate (m^3/s)	31.66	35	45

7.90 A trapezoidal channel initially contains a gate that is set so it produces a depth of 2.5 ft immediately downstream from the gate. This channel is 4000 ft long, and has the following: b_1 = 10 ft, m = 1.5, n = 0.014, S_o = 0.00085. Just before the gate there is a smooth transition to a rectangular channel with b_2 = 8.5 ft. The channel is supplied water from reservoir whose water surface elevation is H = 5 ft above the channel bottom. Assume the entrance and transition minor loss coefficients equal 0.05. Obtain the steady state solution to this problem. At time t = 0 s the gate is raised to increase the discharge past it by a rate of dQ/dt = 0.25 cfs/s for a time of 100 s, and is held constant thereafter. Obtain an unsteady solution to the problem for some time after the gate stops moving. This solution should give the depths, flow rates, velocities, etc. throughout the channel at a number of time increments.

7.91 The depth decreased at a rate of 0.2 ft every 20 s until the depth is 6.5 ft and then it remains constant in the previous problem. And the amount of flow entering the channel of the previous problem increases at a rate of 5 cfs every 20 s until it reaches a value of 397.77 cfs, and then it remains constant.

7.92 Evaluate each of the elements in the Jacobian matrix in Equation 7.87, and which is displayed on the following page that give [D].

7.93 The same channel as in Problem 7.91 has lateral outflow occurring at a constant rate of $q^* = 0.1$ cfs/ft over a 100 ft length 2000 ft downstream from its beginning. The gate raises as in the previous problem to cause dQ/dt = 0.25 cfs/s for a time of 100 s. Now obtain an unsteady solution for some time after the gate stops moving. Also solve the problem in which the gate continue to rise so that dQ/dt = 0.25 cfs/s continues for the entire 600 s of simulations.

7.94 A trapezoidal channel with b = 12 ft, m = 1.5, n = 0.013, and a bottom slope $S_o = 0.0008$ is supplied by a reservoir with a constant head H = 5 ft. The entrance loss coefficient is K_e = 0.08. At a distance 2000 ft downstream there is a gate, set so that initially it creates a depth of 1.5 ft downstream from it. Using the Crank–Nicolson method obtain the flow rates, velocity, and depth throughout the channel if the flow rate at its downstream end increases so that at 2.5 s the flow rate is 2.5 cfs larger than the steady-state flow rate, and the flow rate increases thereafter at a rate of 2.5 cfs each 2.5 s, or at a rate of |dQ/dt| = 1 cfs/s. (Solve the problem through 250 s of time using 2.5 s increments.)

7.95 Modified the program IMPLICAL so that it can simulate the flow in a channel whose flow is controlled by two gates at its downstream end such as in Example Problem 7.8. Each gate is rectangular with a width of b_g = 2.5 m, and has a contraction coefficient C_c = 0.6. With this modified program solve Example Problem 7.8, except that the gates have their positions changed according to the following time schedule:

Time (s)	0.0	40	80	120	160	200	300	1600
Gate # 1, Y_{G1} (m)	.6075	.6075	.8	1.0	1.2	1.4	1.5	1.5
Gate # 2, Y_{G2} (m)	.6075	.8	1.0	1.2	1.6	2.0	2.4	2.4

As a second part of this problem, assume that gate # 1 remains set at the initial height of 0.6075 m and gate # 2 is raised twice as fast as in the above table, until its position is 2.4 m, and then it is held constant. If the two gates supply different users, how much does the second user effect the first user by taking the more water? How much more water does user # 2 take? Speculate on how this effect is different with a channel supplying the two users versus if a pipe line were supplying the two users, and the major difference in head from the beginning to the end of the pipe line was equal to its frictional head loss.

7.96 Resolve the previous problem in which the two gates at the downstream end of the channel are raised as in that problem, but the upstream reservoir head also increases at a constant rate of 0.001 m/s = 0.06 m/min over the entire 1600 s of the simulation. Solve this problem, as the previous one, using the Crank–Nicolson–Newton method. To obtain this solution you will need to modify the program you developed for the previous problem so that the upstream reservoir head can be specified as a function of time.

7.97 The channel in Example Problem 7.8 has the same initial condition as given, and the gate at the downstream end of the channel is suddenly completely opened. Solve this revised problem in which the maximum flow rate at the downstream end of the channel is taking place over a time period of 1600 s. In other words modify the implicit finite difference Newton method so that the downstream boundary condition is the critical flow equation, and with this modified program solve Example Problem 7.8 with critical flow as the downstream boundary condition.

7.98 Assume that uniform flow with Q = 500 cfs is occurring initially throughout the 3000 ft length of a trapezoidal channel with b = 12 ft, m = 1.6, n = 0.014 and S_o = 0.00055. This channel is supplied by a constant head reservoir, and the entrance loss coefficient is zero. Determine what the maximum flow is that can be taken from the downstream end of this channel over a period of 1200 s = 20 min. How does time dependent maximum flow rate compare with that which could be taken directly from the reservoir if this channel had no length.

7.99 Verify the coefficients in the linearized Equations 7.91 and 7.93 that come from differing the St. Venant equations using the Preissmann method.

7.100 Obtain solutions to Example Problem 7.5, controlled by the various boundary condition cases specified in that problem using the Preismann method, and plot the results of the cases specified by your instructor showing how the flow rate, velocity, and depth vary with time throughout the length of the channel.

7.101 Solve the problem defined in Problem 7.94 using the Preissman double sweep method rather than the implicit method based on the Crank–Nicolson method.

7.102 Repeat the previous problem in using the Preissman double sweep method except have the flow rate increase as in the previous problem until it equals 420 cfs at the downstream end, and then hold it constant at this value for all subsequent time steps.

7.103 For use in the Preissmann method as implemented in the computer program PREISDBS.FOR the algorithm that will interpolate between three consecutive values in a stage discharge relationship table using a linear interpolation. Modify this algorithm and the SUBROUTINE BOUNDY so that quadratic interpolation is used (i.e., pass a second degree polynomial through three consecutive points). This algorithm should first determine which three point in the table of Y_s and Q_s encompass the value of Y or Y and thereafter fit these 3 points of data with a second degree polynomial with an interpolation formula, such as Lagrange's formula, and from this interpolation determine the value of the function (i.e., the flow rate) and its derivative. Use this program to solve the unsteady flow in a 3000 ft long trapezoidal channel with b = 12 ft, m = 1.6, n = 0.014 and S_o = 0.00055 that is supplied by an upstream reservoir with a head H = 5.5 ft. Initially at the downstream end a 10 ft wide rectangular gate has a depth of 2.0 ft downstream from it. The gate is suddenly opened and the stage-discharge relationship that occurs thereafter at the end of the channel is determined by a broadcrested weir that is 1.5 ft height (and 10 ft wide) just downstream from the gate. Both K_e and K_L = 0.15. Downstream from the weir the channel is steep.

7.104 The program PREISDBS.FOR for the Preismann method that solves the system of equations using the double sweep method allows for the geometry of the channel to vary from point to point. Modify this program so b, m, n, S_o, and α are constant throughout the channel. In other words remove the arrays B(ND), FM(ND),FN(ND,SO(ND) and ALF(ND) from this program, replacing them by real variables. With this program solve the Example Problem 7.5 for at least one of the boundary conditions.

REFERENCES

Preissman, A. 1960. Propagation des intumescences dans les canaux et rivierères, in *1st Congrès de l'Assoc. Francaise de Calcul*, Genoble, pp. 433–442.

Preissman, A. and J.A. Cunge, 1961. Calcul des intumescences sur machines èlectroniques, in *IX Meeting of the IAHR*, Dubrovnik.

Appendix A

The International (SI) System of Units

	Units	FLt	MLt
Fundamental quantities			
Length (L)	Meters (m)	L	L
Mass (m)	Kilogram (kg)	FT^2L^{-1}	M
Time (t)	Second (s)	t	t
Temperature (T)	Degrees Kelvin (°K)	T	T
Derived quantities			
Area (A)	Square meters (m^2)	L^2	L^2
Volume (V)	Cubic meters (m^3)	L^3	L^3
Velocity (V)	Meters/second (m/s)	Lt^{-1}	L^{-1}
Acceleration (a)	Meters/second^2 (m/s^2)	Lt^{-2}	Lt^{-2}
Angular velocity (ω)	Radian/second (rad/s)	t^{-1}	t^{-1}
Angular acceleration (α)	Radion/second^2 (rad/s^2)	t^{-2}	t^{-2}
Flow rate-volumetric (Q)	Meter^3/second (m^3/s)	L^3t^{-1}	L^3t^{-1}
Kinematic viscosity (ν)	Meter^2/second (m^2/s)	L^2t^{-1}	L^2t^{-1}
Force (F)	Newtons (N)	F	MLt^{-2}
Pressure (p)	Newtons/area (N/m^2)	FL^{-2}	$ML^{-1}t^{-2}$
Density (ρ)	Mass/volume (kg/m^3)	Ft^2L^{-4}	ML^{-3}
Energy, work or heat (E)	Joule (J = n.m)	FL	ML^2t^{-2}
power (P)	Watt (J/s)	FLt^{-1}	ML^2t
Absolute viscosity (μ)	(Ns/m^2)	FtL^{-2}	$ML^{-1}t^{-1}$
Flow rate-weight (W)	Newtons/second (N/s)	Ft^{-1}	ML^{-3}
Flow rate-mass (G)	kg/second (kg/s)	FtL^{-1}	Mt^{-1}

Prefixes

yotta	Y	10^{24}	deci	d	10^{-1}
zelta	Z	10^{21}	centi	c	10^{-2}
exa	E	10^{18}	milli	m	10^{-3}
pefa	P	10^{15}	micro	μ	10^{-6}
tera	T	10^{12}	nano	n	10^{-9}
giga	G	10^{9}	pico	p	10^{-12}
mega	M	10^{6}	femto	f	10^{-15}
kilo	k	10^{3}	atlo	a	10^{-18}
hector	h	10^{3}	zepto	z	10^{-21}
deka	da	10	yecto	y	10^{-24}

Units and Conversion Factors

	Basic Conversion Factors—ES to SI System	SI to ES
Length	1 meter (m) = 3.280,839,9 feet (ft)	1 ft = 0.3048 meters (m)
Force	1 Newton (N) = 0.224,809 pounds (lb)	1 lb = 4.448,22162 Newtons (N)
Mass	1 kilogram (kg) = 0.068,521,78 slugs	1 slug =14.5939 kilograms (kg)
Time	1 s = 1 s (1 h = 3600 s)	60 s = 1 min
Temperature	1 Kelvin = (9/5) °Rankine	1 °Rankine = (5/9) °Kelvin

Secondary Conversion Factors

Length	1 ft = 0.3048 m	1 m = 3.2808 ft
	1 ft = 30.48 cm	1 cm = 0.032808 ft
	1 in. = 2.54 cm	1 cm = 0.394 in.
	1 statute mile = 1.609 km	1 km = 0.622 statute miles
	1 nautical mile = 1.852 km	1 km = 0.540 nautical miles
Area	1 ft^2 = 0.09290304 m^2	1 m^2 = 10.7639104 ft^2
	1 in.^{2} = 6.4516 cm^2	1 cm^2 = 0.1550003 in.^{2}
	1 ac = 43,560 ft^2 = 0.4047 ha	1 ha = 2.471 ac
Volume	1 ft^3 = 0.02831685 m^3	1 m^3 = 35.31466672 ft^3
	1 in.^{3} = 16.387064 cm^3	1 cm^3 = 0.061023744 in.^{3}
	1 ft^3 = 28.32 L	1 L = 0.03531467 ft^3
	1 ft^3 = 7.481 gal	1 L = 1.057 quarts
	1 ac-ft = 1233.5 m^3	1 L = 0.264189 gal
	1 fluid oz = 29.57 cm^3	1 L = 1000 cm^3
Velocity	1 ft/s (fps) = 0.3048 m/s	1 m/s = 3.2808 ft/s
	1 mile/h (mph) = 0.4470 m/s	1 m/s = 2.237 mph
	1 knot = 1.69 ft/s = 0.514 m/s	1 km/h = 0.621 mph
Acceleration	Standard acceleration of gravity = 1 ft/s^2 = 0.3048 m/s^2	32.174,049 ft/s^2
		1 m/s^2 = 3.2808 ft/s^2
Mass	1 slug = 14.594 kg	1 kg = 0.0685 slugs
	1 lb$_m$ = 0.4536 kg	1 kg = 2.205 lb$_m$
	1 slug = 32.174 lb$_m$	
Density	1 slug/ft^3 = 515.363 kg/m^3	1 kg/m^3 = 0.001,94 slug/ft^3
	1 lb$_m$/ft^3 = 16.019 kg/m^3	1 kg/m^3 = 0.0624 lb$_m$/ft^3
	1 slug/ft^3 = 32.174 lb$_m$/ft^3	
Force	1 lb = 4.448 N	1 N = 0.2248 lb
	1 poundal = 0.138,255 N	1 N = 10^5 dynes
Specific weight	1 lb/ft^3 = 157.07 N/m^3	1 N/m^3 = 0.006,37 lb/ft^3
	1 lb/ft^3 = 0.0160 g/cm^3	1 g/cm^3 = 62.43 lb/ft^3
Pressure or stress	1 lb/ft^2 (psf) = 47.877 N/m^2	1 N/m^2 = 0.0209 lb/ft^2
	1 lb/in^2 (psi) = 6894.24 N/m^2	1 N/m^2 = 1.450 × 10^{-4} lb/in.^{2}
	1 in Hg = 70.73 lb/ft^2 = 0.491 psi = 3386 N/m^2	1 bar = 14.50 psi = 2099 lb/ft^2
	1 ft H_2O = 62.4 lb/ft^2 = 0.433 psi = 2989 N/m^2	
	1 standard atmosphere = 14.699 psi = 2116 psf	= 1.013 × 10^5 N/m^2
Energy or work or heat or torque	1 ft · lb = 1.356 N · m (J)	1 N · m = 1 J = 0.737 ft · lb
	1 ft · lb = 0.001,285 Btu = 0.324 cal	1 J = 1 N · m = 10^{7}ergs = 0/239 cal
	1 ft · lb = 3.7676768 × 10^{-7} kWh	
	1 Btu = 778 ft · lb/s = 1054 N · m = 252 cal	
Power	1 ft · lb = 1.356 N · m/s	1 N · m/s = 1 W = 0.737 ft · lb
	1 hp = 33,000 ft · lb/min = 550 ft · lb/s	1 kW = 1.341 hp = 738 ft · lb/s
	1 ft · lb = 746 W = 746 N · m/s	1 N · m/s = 0.0013405 ft · lb/s
	1 Btu/s = 1.0560 kW/s, 1 W = 0.738 ft · lb/s	1 kW = 0.947 Btu/s

Units and Conversion Factors (continued)

	Basic Conversion Factors—ES to SI System	SI to ES
Dynamic viscosity	1 slug/(ft · s) = 1 ft · s/ft^2 = 47.88 kg · (m · s)	1 kg · (m · s) = 1 N · s/m^2
	1 poise = 1 dyne · s/cm^2 = 0.1 N · s/m^2	
Kinematic viscosity	1 ft^2/s = 0.0929 m^2/s	1 m^2/s = 10.764 ft^2/s
	1 stoke = 1 cm^2/s = 0.00108 ft^2/s	
Flow rate	Volume: 1 ft^3/s (cfs) = 0.0283168 m^3/s	1 m^3/s = 35.3147 ft^3/s
	1 ft^3/s = 448.86 gal/min (gpm)	1 L/s = 0.264189 gal/s
	1 gal/min = 0.002227866 ft^3/s (cfs)	1 L/s = 4.228 gal/s
	Mass: 1 slug/s = 14.594 kg/s	1 kg/s = 0.0685 slugs/s
	Weight: 1 lb/s = 4.448 N/s	1 N/s = 0.2248 lb/s

Physical Properties of Water at 5° and 10° Intervals in Both the English (ES) and the International System (SI) Units

Temperature (°C)	Specific Weight, γ (kN/m^3)	Density, ρ (kg/m^3)	Viscosity, $\mu \times 10^3$ (N · s/m^3)	Kinematic Viscosity, $\nu \times 10^6$ (m^3/s)	Surface Tension, σ (N/m)	Abs. Vapor Pressure, p_v (kN/m^2)	Vapor Pres. Head, P_v/γ Abs	Bulk Modulus, $E_v \times 10^{-6}$ (kN/m^2)
0	9.805	999.8	1.781	1.785	0.0756	0.61	0.0622	2.02
5	9.807	1000.0	1.518	1.519	0.0749	0.87	0.0887	2.06
10	9.804	999.7	1.307	1.306	0.0742	1.23	0.1255	2.10
15	9.798	999.1	1.139	1.139	0.0735	1.70	0.1735	2.15
20	9.789	998.2	1.002	1.003	0.0728	2.34	0.239	2.18
25	9.777	997.0	0.890	0.983	0.0720	3.17	0.309	2.22
30	9.764	995.7	0.798	0.800	0.0712	4.24	0.434	2.25
40	9.730	992.2	0.653	0.658	0.0696	7.38	0.758	2.28
50	9.689	988.0	0.547	0.553	0.0679	12.33	1.273	2.29
60	9.642	983.2	0.466	0.474	0.0662	19.92	2.066	2.28
70	9.589	977.8	0.404	0.413	0.0644	31.16	3.250	2.25
80	9.530	971.8	0.354	0.364	0.0626	47.34	4.967	2.20
90	9.466	965.3	0.315	0.326	0.0608	70.10	7.405	2.14
100	9.399	958.4	0.282	0.294	0.0589	101.33	10.781	2.07

Temperature (°F)	γ (lb/ft^3)	ρ (slugs/ft^3)	$\mu \times 10^5$ (lb · s/ft^2)	$\nu \times 10^5$ (ft^2/s)	σ (lb/ft)	p_v (psia)	P_v/γ (ft)	$E_v \times 10^{-3}$ (psi)
32	62.42	1.940	3.746	1.931	0.518	0.09	0.20	293
40	62.43	1.940	3.229	1.664	0.514	0.12	0.28	294
50	62.41	1.940	2.735	1/410	0.509	0.18	0.41	305
60	62.37	1.938	2.359	1.217	0.504	0.26	0.59	311
70	62.30	1.936	2.050	1.059	0.500	0.36	0.84	320
80	62.22	1.934	1.799	0.930	0.492	0.51	1.17	322
90	62.11	1.931	1.595	0.826	0.486	0.70	1.61	323
100	62.00	1.927	1.424	0.739	0.480	0.95	2.19	327
110	61.86	1.923	1.284	0.667	0.473	1.27	2.95	331
120	61.71	1.918	1.168	0.609	0.465	1.69	3.91	333
130	61.55	1.913	1.069	0.558	0.460	2.22	5.13	334
140	61.38	1.908	0.981	0.514	0.454	2.89	6.67	330
150	61.20	1.902	0.905	0.476	0.447	3.72	8.58	328

(continued)

Physical Properties of Water at 5° and 10° Intervals in Both the English (ES) and the International System (SI) Units (continued)

Temperature (°F)	γ (lb/ft^3)	ρ (slugs/ft^3)	$\mu \times 10^5$ (lb·s/ft^2)	$\nu \times 10^5$ (ft^2/s)	σ (lb/ft)	p_v (psia)	P_v/γ (ft)	$E_v \times 10^{-3}$ (psi)
160	61.00	1.896	0.838	0.442	0.44	4.74	10.95	326
170	60.80	1.890	0.780	0.413	0.433	5.99	13.83	322
180	60.58	1.833	0.726	0.385	0.426	7.51	17.33	318
190	60.36	1.876	0.678	0.362	0.419	9.34	21.55	313
200	60.12	1.868	0.637	0.341	0.412	11.52	26.59	308
212	59.83	1.860	0.593	0.319	0.404	14.70	33.90	300

Physical Properties of Common Liquids at 20°C (68°) and 101.325 kPa abs (14.7 psia)

		Dynamic Viscosity		Surface Tension		Vapor[a] Pressure		Bulk Modulus	
Liquid	Specific Gravity S	N s/m^2 $\mu \times 10^5$	Lb s/ft^2 $\mu \times 10^5$	N/m σ	Lb/ft σ	kN/m^{2}abs p_v	psia p_v	N/m^2 $E_v \times 10^{-6}$	psi E_v
Benzene	0.90	6.5	1.4	0.029	0.002	10.0	1.48	1030	150,000
Carbon tetrachloride	1.59	9.7	2.0	0.026	0.0018	12.1	1.76	1100	160,000
Crude oil	0.87	51	11	0.03	0.002	—	—	—	—
Gasoline	0.68	3.2	0.62	—	—	55	8.0	—	—
Glycerin	1.26	14900	3100	0.063	0.004	0.000014	0.000002	4350	630,000
Hydrogen	0.072	0.21	0.043	0.003	0.0002	21.4	3.1	—	—
Kerosene	0.81	19.2	4.0	0.025	0.0017	3.29	0.46	—	—
Mercury	13.56	15.6	3.3	0.51	0.032	0.00017	0.000025	26200	3,800,000
Oxygen	1.21	2.8	0.58	0.015	0.001	21.4	3.1	—	—
SAE 10 oil	0.92	820	170	—	0.0025	—	—	—	—
SAE 30 oil	0.92	4400	920	—	0.0024	—	—	—	—
Water	1.00	10.02	2.1	0.0728	0.005	2.34	0.34	2180	316,000
Ethyl alcohol	0.79	13.0	2.51	0.022	0.0015	5.86	0.85	1210	175,000

[a] In contract with air.

Bulk Modulus of Water

Pressure		Temperature							
		0°C	32.2°F	20°C	68°F	48.9°C	120°F	93.3°C	200°F
Pa × 10^{-5}	psi	Pa × 10^{-9}	Psi × 10^{-5}	Pa × 10^{-9}	Psi × 10^{-5}	Pa × 10^{-9}	Psi × 10^{-5}	Pa × 10^{-9}	Psi × 10^{-5}
1.034	15	2.02	2.93	2.18	3.16	2.29	3.32	2.12	3.08
103.4	1,500	2.07	3.00	2.28	3.30	2.36	3.42	2.20	3.19
310.2	4,500	2.19	3.17	2.40	3.48	2.50	3.62	2.33	3.38
1034	15,000	2.62	3.80	2.83	4.10	2.94	4.26	2.79	4.05

A.1 OPEN CHANNEL GEOMETRY AND PROPERTIES

A.1.1 Classification of Channels

The subject of open channel flow includes fluid flow in **natural** and **man-made** structures. Rivers, stream, creeks, and gullies are common examples of natural channels. In these channels, the geometry, the slope of the channel bottom, and other physical features of the channel generally change from position to position along the channel. Man-made channels generally consist of some common geometric cross-sectional shape, such as a rectangle, a trapezoid, or a circle, and often have constant (or near constant) bottom slopes for large distances. Often, unlined man-made channels have properties closer to those of natural channels, than those of lined man-made channels. Another classification is to refer to a channel as a **natural** or an **artificial** channel.

A more technical classification of channels, which is utilized throughout this book, is the **nonprismatic versus the prismatic channel**.

A **prismatic** channel has an unvarying cross section, a constant bottom slope, and other properties such as the wall roughness that does not change with position. A **nonprismatic** channel has one or all of these properties changing with position along the channel. Thus, most man-made channels that are made from building materials are generally prismatic channels, but portions, such as channel transitions, will be nonprismatic. While rare over very long distances, a natural channel many be prismatic, but generally channels created by nature will be nonprismatic.

It is important to be able to obtain some common geometric properties of channels, such as the cross-sectional area, A, the wetted perimeter, P, and top width, T, as well as properties derived from these, from a description of the type of cross section and the depth of flow in the channel. The most commonly derived quantities are: (1) the hydraulic radius, R_h, which is defined as the area divided by the wetted perimeter or $R_h = A/P$, and (2) the hydraulic depth, H_d, that is commonly used in connection with critical-flow computations, which is defined as the area divided by the top width, or $H_d = A/T$. Another quantity often needed, especially in connection with the use of momentum, is the first moments of area, which for some geometries can be derived from basic variables, but for other geometries must be obtained by integration. This appendix first deals with computing these geometric properties for common geometries used in channels. Thereafter, techniques for obtaining these properties for natural and irregular geometries will be covered. The latter will be covered under the heading of **irregular cross sections** because it includes special shapes that may be man-made as well as those without any special geometric shape.

A.1.2 Geometric Properties of Common Prismatic Channels

A.1.2.1 Rectangle

A rectangular channel has vertical sides and a bottom width, b. The cross-sectional area is obtained from

$$A = bY \tag{A.1}$$

and the wetted perimeter, P is computed from

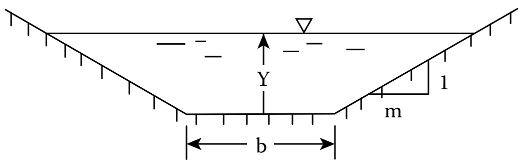
$$P = b + 2Y \tag{A.2}$$

The top width of a rectangular channel is the same as its bottom width, or T = b. For use with the momentum function in open channel flow, the first moment of area about the water surface will be denoted by Ah_c, and for a rectangle equal the area times, the distance from the water surface to the centroid of the rectangle, or one-half the depth and is given by

$$Ah_c = A\left(\frac{Y}{2}\right) = \frac{bY^2}{2} \tag{A.3}$$

A.1.2.2 Trapezoid

For use in defining a trapezoidal channel, the following symbols will be used as shown on the sketch below: b = the bottom width of the channel, Y = depth of flow in the channel, and m = the side slope. Note that, the side slope m is defined as it is in earth work engineering, e.g., the horizontal distance corresponding to a unit vertical distance, and as such is the reciprocal of the mathematical slope of a line on an x y plot that is defined as the vertical distance per unit horizontal distance.



The area of a trapezoidal channel can be obtained by adding the rectangular portion of the area, bY, to the triangular portions, $2(1/2)(mY^2)$, giving

$$A = bY + mY^2 = (b + mY)Y \tag{A.4}$$

The latter means for computing A by Equation A.4 is more efficient when implemented in computer code, since it involves only two multiplications, rather than three, as the former means does.

The wetted perimeter is given by

$$A = b + 2Y[m^2 + 1]^{1/2} \tag{A.5}$$

and the top width by

$$T = b + 2mY \tag{A.6}$$

It is worth noting that, if m = 0 in Equations A.4 through A.6, the results are identical to Equations A.1 through A.3, respectively, that apply for a rectangular channel. Thus, a rectangular channel is a special case of a trapezoidal channel for which m = 0. Likewise, a triangular channel is a special trapezoidal channel with the bottom width b = 0.

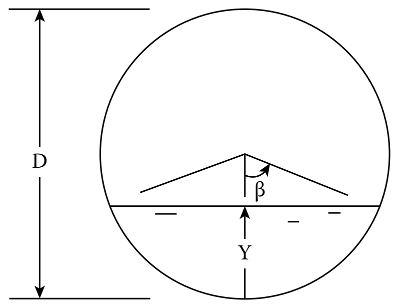
The first moment of area around the water surface is given by

$$Ah_c = \frac{bY^2}{2} + \frac{mY^3}{3} \tag{A.7}$$

This equation can be derived by considering the trapezoid as a composite geometry of a rectangle and two triangles. The first term in Equation A.7 is the area or the rectangle times the distance from the water surface to its centroid, and the second term is this same product for the triangles.

A.1.2.3 Circle

In computing the geometric quantities for circular channels, it will be convenient to introduce the angle, β, shown in the sketch below as the angle from a vertical through the center of the circle to a line from the center to the water surface on the sides of the circle. This angle is related to the depth of flow, Y, and the diameter, D, of the circle by the equation



$$\cos\beta = 1 - \frac{2Y}{D} \quad \text{or} \quad \cos\beta = 1 - \frac{Y}{R} \ (R = \text{radius})$$

or

$$\beta = \cos^{-1}\left(1 - \frac{2Y}{D}\right) \tag{A.8}$$

in which β is in radians (1 rad = (π/180) degrees).

The area can now be readily obtained by subtracting the two triangular areas from the portion of the circle's area below the radial lines from the center to the water surface on the sides of the circle, or $A = (\beta/\pi)(\pi D^2/4) - 2(1/2)[D/2 \cos(\beta)][D/2 \sin(\beta)]$ or,

$$A = \left(\frac{D^2}{4}\right)\left[\beta - \cos(\beta)\sin(\beta)\right] \tag{A.9}$$

Let θ = 2 β, e.g., be the angle between the two radial lines from the center of the circle to the positions of the water surface on the sides, and note that, the trigonometric identity $\sin(2\beta) = \sin(\theta) = [(\cos(\beta)\sin(\beta)]/2$ provides the following alternative formula for computing the area of flow in a circular channel:

$$A = \left(\frac{D^2}{8}\right)\left[\theta - \sin(\theta)\right] \tag{A.10}$$

The wetted perimeter is given by

$$P = \beta D \tag{A.11}$$

and the top width by

$$T = D\sin(\beta) \tag{A.12}$$

The top width can be obtained from depth Y (and D) without introducing angle β from the following equation:

$$T = 2\sqrt{Y(D-Y)} \tag{A.12a}$$

Equation A.12a can be obtained from the trig metric identity $\sin^2(\beta) + \cos^2(\beta) = 1$. From Equation A.8 $\cos^2(\beta) = 1 - 4Y/D + 4Y^2/D^2$, and $\sin^2(\beta) = 4Y/D - 4Y^2/D^2 = 4/D^2(YD - Y^2) = 4/D^2[Y(D - Y)]$. Therefore,

$$\sin(\beta) = \frac{2}{D}\sqrt{Y(D-Y)}$$

and Equation A.12a results by substituting this for the sin(β) into Equation A.12.

An alternative to using the above algebraic formula A.9 and A.10 to find the area is to numerically integrate the differential area shown below, or $A = \int dA = \int 2R\sin\alpha dy$ with limits from β to 0. Since, $y = R\cos(\alpha)$, $dy = -R\sin(\alpha)d\alpha$, and the area is evaluated by (note, the minus is eliminated by reversing the limits of integration)

$$A = 2R^2\int_0^\beta \sin^2(\alpha)\,d\alpha \tag{A.13}$$

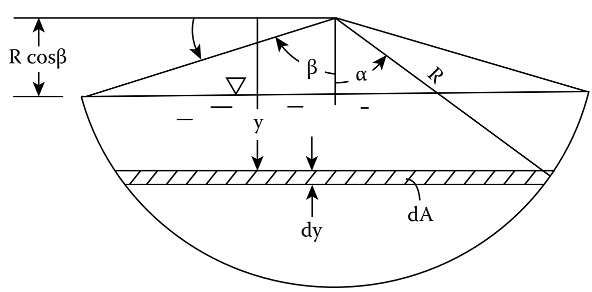
The integral $\int\sin^2\alpha d\alpha$ has a closed form solution equal to $[\beta\text{-}\sin(\beta)\cos(\beta)]/2 = \beta/2 - \sin(2\beta)/4$, which of course gives Equations A.9 and A.10 when multiplied by $2R^2$.

EXAMPLE PROBLEM A.1

Find the area of flow if the depth is Y = 1.2 m in a pipe with a 1.8 m diameter.

Solution

Let us obtain the solution by four different procedures. **First**, use Equations A.8 and A.9. From Equation A.8, β = 1.910623 rad. Then, substituting this β into Equation A.9 gives, $A = (.9)^2\{1.910633 - (\cos 1.910633)(\sin 1.910633)\} = 1.80217\ m^2$. **Second**, numerically evaluate Equation A.13 with your pocket calculator (or computer software). With an HP48G, press the SYMBOLIC key, select Integrate, enter the argument $\sin^2\alpha$, and integrating (numerically) from 0 to 1.910633 produces the value 1.11245. The area is $A = 2R^2(1.11245) = 2(.81)(1.11245) = 1.80217\ m^2$. Note, in order to have obtained this result, it was necessary to first evaluate β from Equation A.8. **Third**, let us use y as the variable of integration in place of α. Note $\sin\alpha = (1 - \cos^2\alpha)^{1/2} = \{1 - (1 - y/R)^2\}^{1/2} = \{2y/R - (y/R)^2\}^{1/2}$. Thus the area equals



Sketch of differential area, etc. for an arc of a circle.

$$A = D\int_0^{1.2}\sqrt{\frac{2y}{R}-\left(\frac{y}{r}\right)^2}\,dy$$

Numerically evaluating this integral gives 1.0012063 so, A = 1.8(1.0012063) = 1.80217 m^2. **Fourth**, let us replace y/R in the above equation from the third method by the dimensionless depth y′ = y/R. Thus, dy = R dy′, so the area is given by

$$A = RD\int_0^{4/3}\sqrt{(2y' - y'^2)}\,dy'$$

A numerical evaluation of this integral gives 1.1124515, so A = 0.9(1.8)(1.1124515) = 1.80217 m^2.

A.1.2.3.1 First Moments of Area

Several alternative formulae are available to compute the first moment of area around the water surface for a circular channel. The following equation requires the least amount of computation (see Table A.1 for a derivation of this equation):

$$Ah_c = \frac{D}{2}\left(\frac{D^2}{6}\sin^3\beta - A\cos\beta\right) \tag{A.14}$$

Again, an alternative to Equation A.14 is to integrate the differential area multiplied by the distance from the water surface down to this differential area. From the previous sketch, the distance from the water surface down to the differential area is R cosβ so that,

$$Ah_c = \int_Y^0 R(\cos\alpha - \cos\beta)D\sin\alpha dy = R^2D\int_0^\beta(\cos\alpha - \cos\beta)\sin^2\alpha d\alpha \tag{A.15}$$

Equation A.15 applies only for circular channels but if the function giving the width related to the depth is known, the above method of numerically integrating to find the area or the first moment of area around the water surface can be used for any shaped channel.

The determination of the location of hydrostatic forces on submerged surfaces involves the second moments of area. As an alternative to using equations that give the second moment of area around the centroid and then the transfer formula $I_o = I_c + Ad^2$, the above procedure for numerically integrating these second moments of area can be used. The difference is that the distance from the water surface to the differential area is squared.

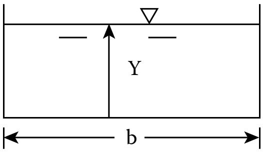
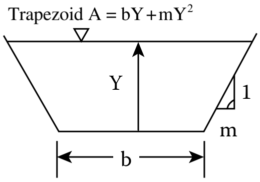
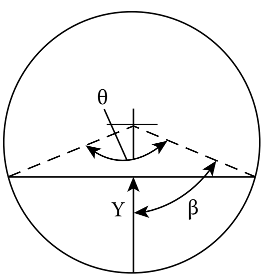
EXAMPLE PROBLEM A.2

Evaluate the first moment of area around the water surface for the previous problem in which the depth is 1.2 m in a pipe with a diameter of D = 1.8 m (R = 0.9 m).

Solution

Substituting D = 1.8 and β = 1.910633 rad in Equation A.14 gives, Ah_c = 0.94794 m^3. An alternative is to numerically integrate Equation A.15. Using this option, cosβ is first evaluated as −0.33333 (or −1/3) and the integral is numerically evaluated as 0.65017 so A = $(0.9)^2$1.8(0.65017) = 0.94794 m^3.

TABLE A.1
Geometric Properties of Cross Sections Often Used for Channels

	Formula for				
Type of Section	**Area**	**Perimeter**	**Top Width**	**Hydr. Radius**	**First Moment of Area**
Rectangle 	$A = bY$	$P = b + 2Y$	$T = b$	$R_h = \dfrac{bY}{b+2Y}$	$Ah_o = bY^2/2$
Trapezoid 	$A = bY + mY^2$	$P = b+2Y(m^2+1)^{1/2}$	$T = b+2mY$	$R_h = \dfrac{bY+mY^2}{b+2Y(m^2+1)^{1/2}}$	$Ah_c = bY^2/2 + mY^3/3$
Circular 	$A = \dfrac{D^2}{4}(\beta - \cos\beta\sin\beta)$ or $A = \dfrac{D^2}{8}(\theta - \sin\theta)$	$P = \beta D$ $P = \theta D/2$	$T = D\sin\beta$ $T = D\sin(\theta/2)$	$R_h = \dfrac{D}{4}(1 - \cos\beta\sin\beta/\beta)$ $R_h = \dfrac{D}{4}(1 - \sin\theta/\theta)$	$Ah_c = \dfrac{D}{2}\left(\dfrac{D^2}{6}\sin^3\beta - A\cos\beta\right)$ $Ah_c = \dfrac{D^3}{24}(3\sin\beta - \sin^3\beta - 3\beta\cos\beta)$

With the current availability of pocket calculators and desktop computers, the above formula will generally be used directly in computing the needed geometric properties of open channels. An understanding of how these quantities vary can be acquired by considering the dimensionless values of these properties. For example, a dimensionless area for a circular channel is the area of the flow divided by the total area of the circle or $A' = A/A_o = (1/2)(\theta - \sin\theta)$. A dimensionless wetted perimeter can be defined by dividing the wetted perimeter by the diameter of the circle, of $P' = P/D = \beta$. Thus, this dimensionless wetted perimeter is nothing other than the auxiliary angle used above. Likewise, a dimensionless top width defined as the top width divided by the diameter of the circle equals the sin of β, or $T' = T/D = \sin\beta$. These dimensionless properties, as well as a dimensionless hydraulic radius defined by dividing the hydraulic radius by D, and a dimensionless hydraulic depth are plotted on Figure A.1 as a function of the dimensionless depth $Y' = Y/D$. The top axis on this figure also provides a means for obtaining the value of the auxiliary angle β from the dimensionless depth Y', or vice versa. The Hydraulic radius $R_h = A/P$ and the Conveyance $K = (C_u/n)A^{4/3}/P^{2/3}$, which are the

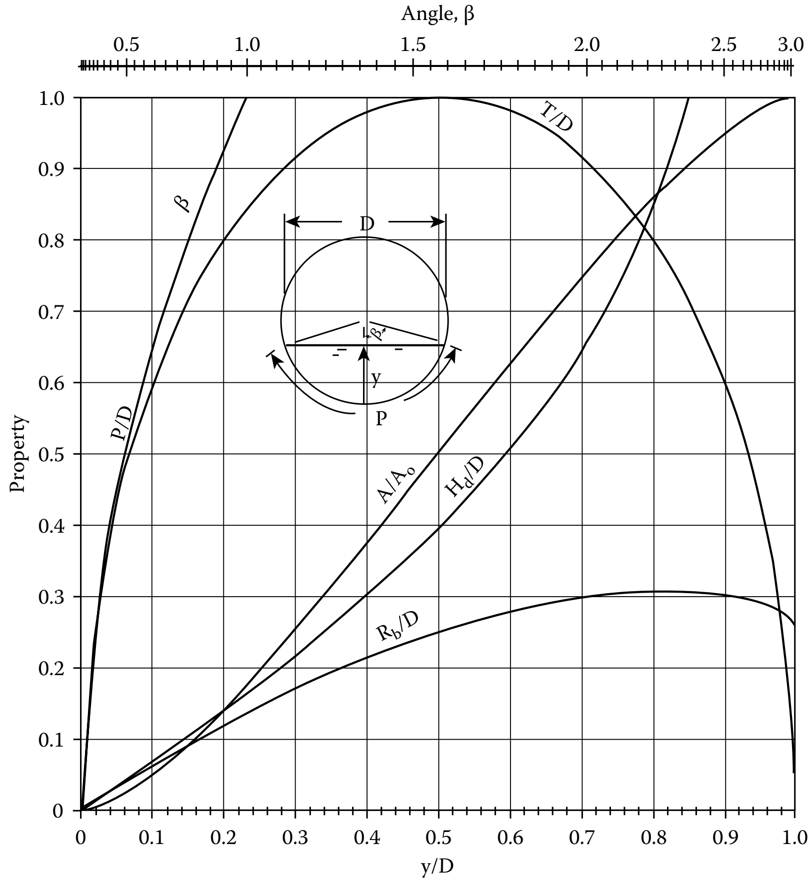


FIGURE A.1 Geometric properties of a circular section.

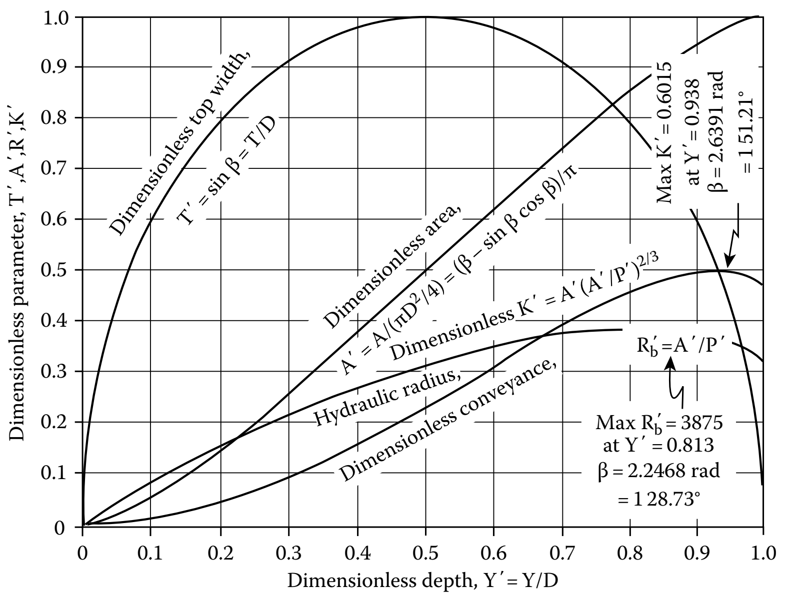


FIGURE A.2 Geometric properties of a circle, including the dimensionless conveyance, $K' = A'(A'/P')^{2/3}$.

channel properties used in Manning's equation, are important in hydraulic computations. Figure A.2 also included the dimensional relationship for these two parameters to the dimensionless depth.

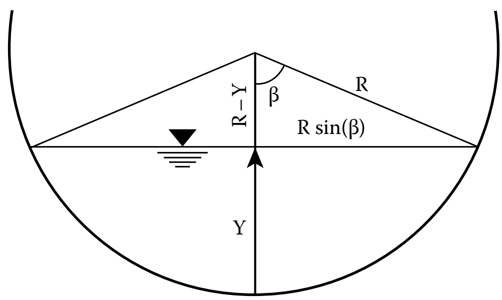
With computer software packages such as Mathcad, TK-Solver (and even your pocket calculator), etc., that allow implicit equations to be solved for any variable as the unknown, it is often advantageous not to introduce the additional variable β, or θ, but express A, T, and P in terms of the depth Y, or express the dimensionless form of these variables in terms of the dimensionless depth $Y' = Y/D$. Below, the equations relating the dimensionless variables are given. Using the sketch below, $\sin\beta$ and $\cos\beta$ can be related to the dimensionless depth $Y' = Y/D$ as shown below to eventually give A_d, $T' = T/D$ and $P' = P/D$.

$$R^2 = (R - Y)^2 + (R\sin\beta)^2$$

$$\sin^2\beta = \frac{2Y}{R} - \left(\frac{Y}{R}\right)^2 = \frac{4Y}{D} - \left(\frac{2Y}{D}\right)^2$$

$$\sin\beta = 2\left(\frac{Y}{D}\left(1 - \frac{Y}{D}\right)\right)^{1/2} = 2\sqrt{Y'(1-Y')}$$

$$\cos\beta = 1 - \frac{2Y}{D} = 1 - 2Y'$$



$$A_d = \frac{A}{D^2} = \frac{1}{4}\left\{\cos^{-1}(1-2Y') + 2(1-2Y')\left[Y'\left(1-Y'\right)\right]^{1/2}\right\} = \frac{1}{4}(\beta - \cos\beta\sin\beta) \qquad \text{(A.16)}$$

(Note dimensionless $A_d = (\pi/4)A'$)

$$T' = \frac{T}{D} = 2\sqrt{Y'(1-Y')} \qquad \text{(A.17)}$$

$$P' = \frac{P}{D} = \cos^{-1}(1-2Y') \qquad \text{(A.18)}$$

and defining a dimensionless first moment of area

$$Ah_c' = \frac{Ah_e}{D^s} = \frac{1}{2}\frac{\sin^3\beta}{6} - \frac{A_d\cos\beta}{2} \quad \text{or}$$

$$Ah_c' = \frac{2[Y'(1-Y')]^{15}}{3} - A_d(0.5 - Y') \qquad \text{(A.19a)}$$

$$Ah'_c = \frac{2[Y'(1-Y')]^{1.5}}{3} - (0.5-Y')\frac{\cos^{-1}(1-2Y') + 2(1-2Y')\sqrt{Y'(1-y')}}{4} \qquad \text{(A.19b)}$$

Another often used parameter is the Froude number, F_r. For any channel, the Froude number squared is $F_r^2 = Q^2T/(gA^3)$. For a circular section, the Froude number squared becomes

$$F_r^2 = \frac{2^7Q^2\left[(Y/D)(1-Y/D)\right]^{1/2}}{gD^5(\beta - \cos\beta\sin\beta)} = \frac{2Q'^2\left[Y'(1-Y')\right]^{1/2}}{A_d^3}, \quad \text{with } Q'^2 = \frac{Q^2}{gD^5}$$

The first moments of area about the water surface can be defined for a circular section by integration of the product of h and the differential area dA from the bottom of the section to the depth Y as shown in the sketch below, or $Ah_c = \int h dA$. Substituting $h = Y - z$ and $dA = D \sin\alpha\, dx$ results in

$$Ah_c = D\int (Y-x)\sin\alpha\, dx$$

Since, $\cos\alpha = 1 - 2x/D$, $dx = D\sin\alpha\, d\alpha/2$, and $Y = D(1-\cos\alpha)/2$, the above equation can be written as follows for easy integration.

$$Ah_c = \frac{D^3}{4}\int_0^\beta (\cos\alpha - \cos\beta)\sin^2\alpha\, d\alpha$$

Upon completing the integration and simplifying results in

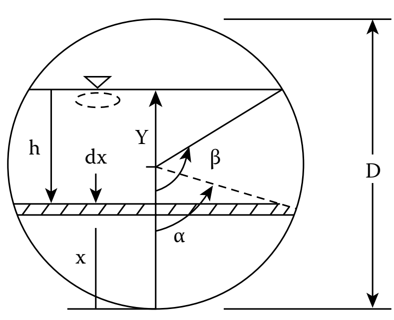
$$Ah_c = \frac{D^2}{24}(3\sin\beta - 3\beta\cos\beta - \sin^3\beta)$$

but since the area $A = (D^2/4)(\beta - \cos\beta\sin\beta)$, an alternative expression for the first moments of area about the water surface is

$$Ah_c = \frac{D}{2}\left(\frac{D^2}{6}\sin^3\beta - A\cos\beta\right)$$

Example

$D = 5$ ft, $Y = 3$ ft, so $\beta = \cos^{-1}(1 - 6/5) = 1.77215$ rad and $A = (D^2/4)(\beta - \cos\beta\sin\beta) = 12.3007\ \text{ft}^2$, so $Ah_c = (D^3/24)(3\sin\beta - 3\beta\cos\beta - \sin^3\beta) = 15.9483\ \text{ft}^2$, or $Ah_c = (D/2)\{(D^2/6)\sin^3\beta - A\cos\beta\} = 15.9483\ \text{ft}^2$.



The dimensionless values for the first moments of area Ah_c can be defined for trapezoidal and rectangular channels. A logical dimensionless first moment of area for a trapezoidal channel is $Ah'_c = Ah_c/(b^3/m^2) = (m^2/b^3)(bY^2/2 + mY^3/3) = Y'^2/2 + Y'^3/3$. Using this definition, the dimensionless momentum function $M' = (m^2/b^3)M = Ah'_c + Q'/(Y' + Y'^2)$ in which $Q' = m^2Q^2/(gb^5)$. Let $Y' = Y/b$ for a rectangular channel and $M' = M/b^3 = Y'^2/2 + Q'_r/Y'$ in which $Q'_r = Q^2/(gb^5)$. Thus, for a rectangular channel, the dimensionless first moment of area $Ah'_c = Y'^2/2$. Figure A.3 shows how the dimensionless first moments of area vary as functions of the dimensionless depths.

The above formula for computing the often used geometric properties of a prismatic channel are summarized in Table A.1.

EXAMPLE PROBLEMS

1. The depth of flow in a 1.5 m diameter channel is 0.95 m. Determine A, P, T, R_h, and H_d. Using Figure A.1, we enter the abscissa with Y/D = 0.63 and read $A/A_o = 0.67$, $P/D = \beta = 1.84$, T/D = 0.96, $H_d/D = 0.54$, and $R_d/D = 0.28$, and therefore: $A = 1.18\,m^2$, P = 2.76 m, T = 1.44 m, $H_d = 0.81$ m, and $R_d = 0.42$ m. Using the above equations to give three digits of accuracy beyond the decimal point gives: $\beta = 1.841$, $A = 1.180\,m^2$, T = 1.446 m, P = 2.761 m, $H_d = 0.816$, and $R_h = 0.427$.
2. The flow rate in a trapezoidal channel with b = 3 m and m = 1.5 is $Q = 17\,m^3/s$. The average velocity of the flow is determined as 1.5 m/s. Determine the depth of flow.
   The area $A = Q/V = 17/1.5 = 11.333\,m^2$, must equal $3Y + 1.5Y^2$. This resulting equation can be solved by the quadratic formula giving Y = 1.457 m.
3. The same flow rate of $17\,m^3/s$ at an average velocity of 1.5 m/s occurs in a circular channel with a diameter of 5 m. What is the depth of flow?

The solution consists of equating the formula for the area to $11.333\,m^2$. The area is obtained by dividing the flow rate by the velocity. For this problem, with a circular section, the resulting equation is implicit and is

$F(\beta) = 6.25(\beta - \cos\beta \sin\beta) - 11.333 = 0$, which must be solved by an iterative method, such as the Newton method (see Appendix B). From the Newton iterative formula,

$$\beta^{i+1} = \beta^i - (F(\beta)/dF/d\beta)^i$$

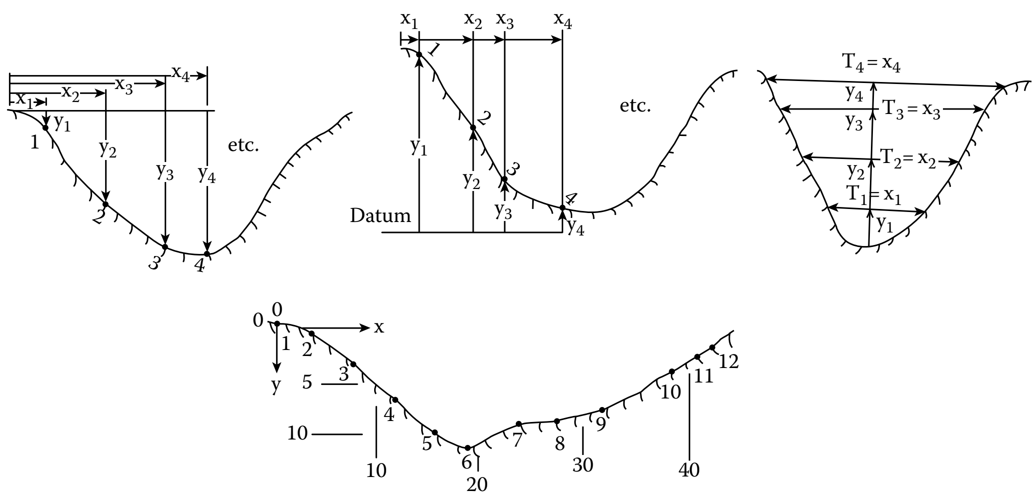
Implementation of this formula is accomplished using the following few lines of the BASIC code. A starting value for $\theta = 2\beta = 3$ rad provides an answer of $\beta = 1.693$ rad from which Y = 2.804 m.

Basic code to use Newton method

```
10 INPUT "ENTER: Q,V,D & est. TH ",Q,V,D,TH
20 AM=(Q/V)*8/(D*D)
30 NCT=0
40 DIF=(TH-SIN(TH)-AM)/(1–COS(TH))
50 TH=TH-DIF
60 NCT=NCT+1
70 IF (ABS(DIF)>.00001) AND (NCT<15) THEN GOTO 40
80 Y=D/2*(1–COS(TH/2))
90 PRINT Y,TH/2
100 END
```

A.1.2.4 Obtaining Geometric Properties of Irregular Cross Sections

A description of an irregular section is needed before such properties as the area, the perimeter, etc., can be computed. There are a number of methods used to describe these sections, such as a sketch of the cross section drawn to a known scale or field survey notes that give rod readings from a surveyor's level and corresponding chain readings. For use in open channel hydraulic computations, it is desirable to convert whatever the given description of the irregular channel is into tabular results giving the width across the channel, e.g., the top width T, as a function of the distance up from the lowest point in the channel, e.g., the position, y. From this tabular data of T versus y, it is relatively easy to compute the area, A and the wetted perimeter, P as a function of the depth of flow Y, and from these, other derived properties, such as the hydraulic radius, R_h and the hydraulic depth D_h. By using interpolation techniques on such tabular data, it is possible to determine any of these needed quantities for any depth, or the depth corresponding to a known cross-sectional area. The three most common methods of defining the irregular shape of a channel are (1) giving transect data, e.g., the horizontal distances from one bank as the x coordinate, and the distance below this beginning point to the channel side as the second value, or the y coordinate, (2) giving the same horizontal distances as in 1 for the x coordinate but the elevation of the points on the channel side as the y values, and (3) giving the widths across the channel as the x values (e.g., the top widths) and the depths corresponding thereto as the y coordinates. In the last definition, y corresponds with the depth Y. These three alternatives are illustrated in the sketch below in Figure A.2.

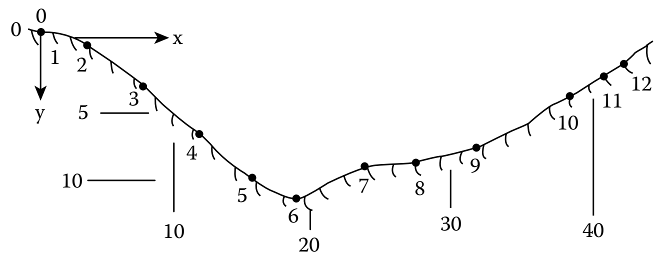


The first step in effectively utilizing any of these descriptions is to obtain, therefrom, a new tabular set of data in which the top width is given as a function of the depth of flow with the increment between consecutive depth values in this table equal to a constant. This is done by passing some type of curve through some number of consecutive points to give a continuous function for x related to y. Spline functions have many desirable features for this interpolation, but their use is more complex than using a polynomial of some selected degree. A second, or first degree polynomial is frequently used. Using a first degree polynomial provides for linear interpolation between two consecutive points in the original data. A second degree polynomial passes a parabola through three consecutive points, and is referred to as quadratic interpolation. Appendix B describes Lagrange's equation for carrying out such polynomial interpolations between unequal values of the dependent variable.

An example using the transect data, given below, illustrates the use of Lagrange's formula to create a new table of top widths as a function of depths.

x	y
0.0	0.0
3.8	0.5
6.7	3.0
11.5	6.3
15.7	9.9
19.1	11.1
24.0	8.9

x	y
27.5	8.4
31.5	7.5
34.9	5.2
38.0	3.1
40.8	1.1
43.5	0.0



In this new table, the areas, and wetted perimeters can be added as additional columns. The areas are obtained by numerically integrating the top width with respect to the position or

$$A = \frac{\Sigma(T_i + T_{i-1})}{2}(\Delta y) \tag{A.20}$$

An alternative is to accumulate the area by separating it into the rectangular center portion equal to Δy times T_{i-1} plus the two additional side portions, or

$$A_i = A_{i-1} + \Delta y T_{i-1} + \frac{\Delta y}{2}\left\{\text{chg in x(right side)} + \text{chg in x(left side)}\right\} \tag{A.21}$$

In this latter formula, $\Delta y/2$ can be changed to $2/3\Delta y$ if one wishes to assume that the sides of the channel are defined better by a parabola than a straight line. This latter assumption may be more applicable especially for the very bottom of the channel in obtaining A_2 from $A_1 = 0$.

The wetted perimeter is obtained by noting that each new increment of depth adds the incremental lengths $(\Delta x^2 + \Delta y^2)^{1/2}$ from both sides, or

$$P = \Sigma\left\{(\Delta x^2 + \Delta y^2)_1^{1/2} + (\Delta x^2 + \Delta y^2)_2^{1/2}\right\} \tag{A.22}$$

The transect data from the irregular section shown above will be used to illustrate the above procedures for creating a table of geometric channel properties with a constant increment of depth. The first step is to note that the position of the maximum value of y = 11.1 ft is number 6. This position, or close to it, represents the bottom of the channel, or the starting position where a second degree polynomial (or other interpolating function) is to pass through three consecutive points moving up both the right and left banks of the channel. The second degree polynomial used for the left bank will pass through points 6, 7, and 8 to begin with, and the second degree polynomial used for the right bank will initially pass through points 6, 5, and 4. After the depth of flow exceeds the difference between the y at the bottom point number 6 and the top point 8 or 4 of the triad of points, (or y is less than y_8 or y_4) then a new point is added, and the first point is dropped from the triad of points until the highest point is included in the triad.

In this problem, the geometric properties will be determined for 20 increments of depth. Therefore, $|\Delta y|$ equals the difference between the largest and smallest values of y divided by 20, or $\Delta y = |(11.1 - 0.0)/20| = 0.555$ ft will be used for the increment of depth in the table. Using LaGrange's interpolation formula (see Appendix B) for the left bank gives

$$x_{Li} = \frac{(y-8.9)(y-8.4)}{(11.1-8.9)(11.1-7.5)}19.1 + \frac{(y-11.1)(y-8.4)}{(8.9-11.1)(8.9-7.5)}24.0 + \frac{(y-11.1)(y-8.9)}{(8.4-11.1)(8.4-8.9)}27.5 \tag{A.23}$$

and for the right bank gives

$$x_{Li} = \frac{(y-9.9)(y-6.3)}{(11.1-9.9)(11.1-6.3)}19.1 + \frac{(y-11.1)(y-6.3)}{(9.9-11.1)(9.9-6.3)}15.7 + \frac{(y-11.1)(y-9.9)}{(6.3-11.1)(6.3-9.9)}115 \tag{A.24}$$

in which i = 2.

For the first entry in the table, $x_{R1} = x_{L1} = 19.1'$ and the second entry (i = 2) will have y = 11.10 − 0.555 = 10.545′, corresponding to a depth of y = 0.555′. Substituting this y into Equations A.23 and A.24 gives $x_{R2} = 17.403$ ft and $x_{L2} = 18.722$ ft. The top width is, therefore, $T_2 = 1.32$ ft. The area is

$$A_i = A_{i-1} + \Delta y(L_2 - R_2) + \tfrac{1}{2}\Delta y\left\{(x_{Ri-1} - x_{R_i})(x_{Li} - x_{L_{i-1}})\right\}$$

which for i = 2 gives $A_2 = 0.58$ sq ft. The wetted perimeter is computed from

$$P_i = P_{i-1} + \sqrt{\Delta y^2 + (x_{Ri-1} - x_{Ri})^2} + \sqrt{\Delta y^2 + (x_{Li} - x_{Li-1})^2}$$

or with $x_{R1} = x_{L1} = 19.1'$ and $x_{R2} = 17.403'$ and $x_{L2} = 18.722'$, $P_2 = 2.46$ ft.

Proceeding with similar computations for the remaining 19 increments of $|\Delta y| = \Delta y = 0.555$ ft produces the values for the geometric properties for this irregular section that are given in Table A.2.

The small Pascal program below implements these techniques in creating a new table in which the top width, the area and the wetted perimeter are given as tabular functions of the depth y on a constant increment. The method used for describing the irregular cross section is communicated to the computer program with the initial input, and this description can either be entered through the key board with the program prompting for the x and y values, or these values can be read from a file prepared in advance. Prompts given by the program inform the user how he can indicate what the meaning of the x and y coordinates are. He also can select between using linear, or quadratic interpolation in generating the table of geometric properties written by the program. For data that are not smooth, it is best to use linear interpolation since a parabola passing through three irregular points can have values far from those given between the points.

Listing of Pascal program GEONAT.PAS to generate geometric properties of an irregular channel

```
PROGRAM Geometry_Natural;
Const
 N2:integer=20; {Dimensions of array for properties = 20}
var
 X,Y: array [1..25] of real; { Up to 25 pairs of coord. allowed}
 Y1,T,A,P: array [1..20] of real;{T=top width,A=area,P=Perimeter}
 YM,DY,YMI,XR,XL,YY,AA,XR1,XL1,DY2,PP,DYS: real;
```

TABLE A.2
Geometric Properties for Irregular Channel Given Above

Depth	Top Width	Area	Perimeter
0.55	1.32	0.58	2.46
1.11	3.51	1.92	4.94
1.66	6.58	4.72	8.21
2.22	10.53	9.47	12.35
2.77	15.19	16.60	17.22
3.33	18.48	25.95	20.78
3.88	20.74	36.83	23.39
4.44	22.36	48.79	25.51
4.99	23.49	61.52	27.15
5.55	24.53	74.84	28.71
6.10	25.49	88.72	30.25
6.66	27.12	103.32	32.22
7.21	28.77	118.83	34.20
7.77	30.44	135.26	36.21
8.32	32.05	152.60	38.17
8.88	33.52	170.80	40.01
9.43	34.95	189.80	41.82
9.99	36.34	209.58	43.60
10.54	38.18	230.26	45.78
11.10	43.50	252.92	51.24

```
 N,I,IL,IR,NM,IU: integer;
 POSCOR,WIDCOR,LINEAR: Boolean;
 FILN:String[20]; INPUT:text; Label L5,L6,L7;
BEGIN POSCOR:=false; WIDCOR:=false; LINEAR:=false;
L7:Writeln('Give no. of pts, and then this many pairs of x &
  y coordinates');
Writeln('Precede no. with - to change type of interpolation-
  LINEAR=',LINEAR);
Writeln('Add 100 to no. if on file'); Writeln;
Writeln('Give 1000 if x & elev for y will be given, and');
Writeln(' 2000 if x=top width & y=depth will be given.');
 Read(N);ClrScr;If N=1000 then begin POSCOR:=true;ClrScr;
   GoTo L7 end;
 If N=2000 then begin WIDCOR:=true;ClrScr;GoTo L7 end;
 If N<0 Then begin N:=abs(N);LINEAR:=not LINEAR end;
 If N>100 Then Begin Writeln('Give file name for table ');
   Read(FILN);
  Assign(INPUT,FILN);Reset(INPUT);N:=N-100;
  For I:=1 to N do Read(INPUT,X[I],Y[I]); GoTo L5 End;
 If N>25 then begin GoToXY(1,10);Write('Dim.ed for only 25 -press
   return');
 If keypressed then GoTo L6 end;
 NM:=N-1;GoToXY(13,4);Writeln('x y');
 For I:=1 to N do Writeln(I:8);
```

```
 For I:=1 to N do Begin GoToXY(13,I+4); Read(X[I],Y[I]) End;
L5:If POSCOR then Begin YM:=Y[1]; If Y[N]>YM then YM:=Y[N];
 For I:=1 to N do Y[I]:=YM-Y[I] End;
If WIDCOR then Begin If Y[1]>0 then begin N:=N+1; For I:=N downto
  2 do begin
    X[I]:=X[I-1];Y[I]:=Y[I-1] end; Y[1]:=0; X[1]:=0 end;
   YM:=Y[N];YMI:=X[N]/2; For I:=1 to N do begin
     X[N+I-1]:=YMI+X[I]/2;
    Y[I+N-1]:=YM-Y[I] end;YMI:=2*YMI; For I:=1 to N-1 do begin
      Y[I]:=Y[2*N-I];
    X[I]:=YMI-X[2*N-I] end; N:=2*N-1 End;
 YM:=Y[1]; YMI:=YM; If Y[N]<YMI then YMI:=Y[N];
 For I:= 2 to N do If Y[I] > YM Then Begin YM:=Y[I];IL:=I End; :=
DY:=(YM-YMI)/N2;AA:=0;PP:=0;XR1:=X[IL];XL1:=XR1;DY2:=0.5*DY;
  DYS:=DY*DY;
  YY:=YM; IR:=IL+1; IL:=IL-1;
 For I:=1 to N2 do
 Begin
 YY:=YY-DY;
If LINEAR Then Begin
 While (YY<Y[IL]) and (IL>1) do IL:=IL-1;
 While (YY<Y[IR]) and (IR<N) do IR:=IR+1;
 XL:=X[IL+1]+(YY-Y[IL+1])/(Y[IL]-Y[IL+1])*(X[IL]-X[IL+1]);
 XR:=X[IR-1]+(YY-Y[IR-1])/(Y[IR]-Y[IR-1])*(X[IR]-X[IR-1]) End
else Begin {Uses Lagrange's formula for quadratic interpolation}
 While (YY<Y[IL-1]) and (IL>2) do IL:=IL-1;
 While (YY<Y[IR+1]) and (IR<NM) do IR:=IR+1;
 XL:=(YY-Y[IL])*(YY-Y[IL+1])*X[IL-1]/((Y[IL-1]-Y[IL])*(Y[IL-1]-Y
   [IL+1]))+
  (YY-Y[IL-1])*(YY-Y[IL+1])*X[IL]/((Y[IL]-Y[IL-1])*(Y[IL]-Y
    [IL+1]))+
  (YY-Y[IL-1])*(YY-Y[IL])*X[IL+1]/((Y[IL+1]-Y[IL-1])*(Y[IL+1]-Y
    [IL]));
 XR:=(YY-Y[IR])*(YY-Y[IR-1])*X[IR+1]/((Y[IR+1]-Y[IR])*(Y[IR+1]-Y
   [IR-1]))+
  (YY-Y[IR+1])*(YY-Y[IR-1])*X[IR]/
((Y[IR]-Y[IR+1])*(Y[IR]-Y[IR-1]))+
  (YY-Y[IR+1])*(YY-Y[IR])*X[IR-1]/((Y[IR-1]-Y[IR+1])*(Y[IR-1]-Y
    [IR]))
 End;
 Y1[I]:=YM-YY;
 T[I]:=abs(XR-XL);
 AA:=AA+DY*abs(XR1-XL1)+DY2*(abs(XR-XR1)+abs(XL1-XL)); A[I]:=AA;
 PP:=PP+sqrt(DYS+sqr(XR-XR1))+sqrt(DYS+sqr(XL1-XL));P[I]:=PP;
 XR1:=XR; XL1:=XL
End; ClrScr;
 Writeln(' Depth Top Width Area Perimeter');
 For I:=1 to N2 do Writeln(Y1[I]:10:2,T[I]:10:2,A[I]:10:2,
   P[I]:10:2);
L6: Writeln('When ready press return'); Read(IU); ClrScr;
END.
```

EXAMPLE PROBLEMS

1. A section of a river that contains a flow rate of Q = 225 cfs has the irregular cross section of previously given cross-section data. The average velocity here is 1.5 fps. Determine the depth of flow, the top width, and the wetted perimeter here.

   Dividing Q by V gives A = 225/1.5 = 150 sq ft. Using the geometric properties in Table A.2 and a linear interpolation allows the depth to be computed as

   Y = (150 − 135.26)/(152.60 − 135.26) × (8.32 − 7.77) + 7.77 = 0.85 × 0.55 + 7.77 = 8.24 ft; the top width is T = 0.85 × (32.05 − 30.44) + 30.44 = 31.81 ft and the wetted perimeter is P = 0.85 × (38.17 − 36.21) + 36.21 = 37.88 ft.
2. Use the same cross section data as given in previously given cross-section data, but use a linear interpolation to define the top width T, area A, and wetted perimeter, P as a function of y in 0.55 increments. The result from this problem gives the following table:

Depth	**Top Width**	**Area**	**Perimeter**
0.55	2.81	0.78	3.02
1.11	5.62	3.12	6.05
1.66	7.65	6.80	8.37
2.22	9.63	11.59	10.67
2.77	13.97	18.14	15.25
3.33	17.08	26.76	18.64
3.88	19.35	36.87	21.20
4.44	20.82	48.02	23.05
4.99	22.35	60.00	24.93
5.55	23.97	72.86	26.90
6.10	25.60	86.61	28.87
6.66	27.23	101.27	30.84
7.21	28.85	116.83	32.81
7.77	30.48	133.30	34.78
8.32	32.02	150.64	36.68
8.88	33.44	168.81	38.48
9.43	34.86	187.76	40.28
9.99	36.28	207.50	42.09
10.54	38.27	228.19	44.40
11.10	43.50	250.88	49.78

   You should note that larger quantities are produced using a quadratic interpolation near the bottom of the section, but smaller values occur near the top.
3. The following transect data define an irregular section of a stream.

x (m)	0.0	1.0	2.5	4.0	6.0	7.0	8.0	9.0	10.0	12.0
y (m)	0.0	0.5	1.2	2.0	2.9	3.2	2.7	1.6	0.7	−0.2

   Use this data to define the section by giving depths and corresponding top widths.

   Using 20 increments for y and a second degree polynomial interpolation as described above gives

Y (m)	0.17	0.34	0.51	0.68	0.85	1.02	1.19	1.36	1.53	1.70
T (m)	0.97	1.84	2.62	3.30	3.90	4.40	4.81	5.24	5.64	6.07
	1.87	2.04	2.21	2.38	2.55	2.72	2.89	3.06	3.23	3.40
	6.56	7.05	7.59	8.15	8.75	9.44	10.13	10.85	11.60	12.38

4. If the flow rate through the section of problem 2 is $Q = 18\,m^3/s$ and the depth is 2.4 m, determine the average velocity at this section and the hydraulic radius.

Numerically integrating to obtain the area and the wetted perimeter as described above, gives the following portion of a table of geometric properties for this irregular section.

y (m)	T (m)	A(m^2)	P(m)
2.38	8.15	10.89	9.62
2.55	8.75	12.32	10.32

From which $A = (0.02/.17) \times 1.43 + 10.89 = 11.06\,m^2$ giving $V = Q/A = 18/11.06 = 1.627\,m/s$; $P = (0.02/.17) \times 0.70 + 9.62 = 9.70\,m$, and $R_h = 11.06/9.70 = 1.14\,m$.

PROBLEMS (FOR YOU TO WORK)

A.1 A trapezoidal channel has a bottom width of $b = 3\,m$ and a side slope, $m = 1.3$. If the depth of flow is 2 m determine: (1) the area A, (2) the wetted perimeter P, (3) the top width T, (4) the hydraulic radius R_h, and (5) the hydraulic depth D_h.

A.2 The flow rate in a trapezoidal channel is $Q = 30\,m^3/s$ and the average velocity equals 0.8 m/s. If the bottom width is 8 m and the side slope is 1.5, determine the depth of flow, and the hydraulic radius.

A.3 A trapezoidal channel is to be designed with a side slope of 1.2 to convey a flow rate of $50\,m^3/s$. The velocity expected in the channel for this design flow rate is 2 m/s. If the ratio of depth to bottom width is to be 1/2 determine what size of channel is needed.

A.4 A flow rate of 400 cfs is occurs at a velocity of 2 fps in a trapezoidal channel with a bottom width of 12 ft and a side slope of 1. Determine the depth of flow.

A.5 A rectangular channel is to be designed with a width twice the depth of flow. If the area of flow is to be 50 sq ft, determine the size of the channel to be built.

A.6 A pipe with a 36 in. diameter is carrying a flow rate of 10.0 cfs at a velocity of 4 fps. What is the depth of flow, the wetted perimeter, the top width, and the hydraulic radius associated with this flow?

A.7 The depth of flow in a 3 m diameter pipe is 1.4 m. Compute the following quantities: A, P, T, R_h, and D_h.

A.8 If a circular culvert is to be designed to carry a flow rate of $100\,m^3/s$ with the depth equal to one half the diameter, determine the size of the pipe needed if the velocity is not to exceed 3 m/s.

A.9 The water elevation at a roadway culvert is expected to be 4500 ft. If a 12 ft diameter culvert is to be utilized, determine the elevation of its invert, e.g., its bottom if it should convey 100 cfs at a velocity of 3 fps.

A.10 Write a small computer program that will completely solve the geometry of a trapezoidal channel. This program should first determine whether SI or ES units are being used, and then be told what the unknown is, and finally read in the known values.

A.11 Write a small computer program that will completely solve the geometry of a circular section similar to that of Problem A.10. To solve some of the items, this program will need to use an iterative solution such as the Newton method.

A.12 The depth of flow in a symmetric river with the cross section defined by the following data is 12 ft. Determine the following: the top width T, the area A, the wetted perimeter, P, and the hydraulic radius R_h.

Depth, y′	0.5	1.2	3.0	6.0	9.4	14.3	17.1
Width, T′	3.2	9.3	15.8	22.4	31.8	50.9	104.7

A.13 Using a second degree polynomial interpolation and the data for the river cross section of problem 12 create a table of values for T, P, and A as a function of the depth y at a constant increment thereof of 0.5 ft.

A.14 Utilizing the data in the table produced in Problem A.13, determine the following at the two depths 6.4 and 10.2 ft: the top width T, the wetted perimeter, P, and the area A.

A.15 Using a spreadsheet such as LOTUS 123 enter the transect data for an irregular section as, given below, into the first two columns, and have it produce the following additional columns in this table: column 3—the depth of flow; column 4—the width of flow; column 5—the wetted perimeter; column 6—the area of flow, and column 7—the hydraulic radius.

x (m)	0.0	3.4	5.3	7.4	12.1	15.8	23.2	30.1	38.4
y (m)	0.0	1.1	2.8	4.1	6.0	5.1	4.3	2.1	0.1

A.16 Add graphical capabilities to the program listed in the previous section dealing with irregular sections so that it will plot the cross section of the channel on the PC's monitor. You might do this using the graphic capabilities in TURBO PASCAL, or utilize other graphic utilities.

A.17 Write a computer program that will accept as input data the output file produced by the program whose listing in the previous section deals with irregular sections that contain the geometric properties on an constant increment of the depth y, and accepts as input, any one of the following: Y, T, P, or A as the known and will computer the other three items as unknowns by using either a linear or a quadratic interpolation. Using Table A.2 as input to this program verify that it is working correctly by filling in the missing values in the table below.

Y	T	P	A
3.91	—	—	—
—	24.00	—	—
—	—	28.00	—
—	—	—	100.00

Appendix B Numerical Methods

B.1 NEWTON METHOD

There are numerous problems in engineering and the sciences that involve the solution of nonlinear, or implicit, algebraic, equations. Often only one equation is involved, whereas in other problems an entire system of nonlinear equations must be solved simultaneously. A method that is adapted to solve either a single equation or an entire system of nonlinear equations is the Newton method. (This method is sometimes called the Newton–Raphson method.) The Newton method converges rapidly to the solution provided a reasonable starting value is given. Mathematically, the Newton method has quadratic convergence whereas many other iterative methods have linear convergence characteristics. The Newton method is also very easy to implement into a computer program, and to utilize with a pocket calculator. Consequently, it is a method that you should become acquainted with as part of your training as an engineer, if you are not already familiar with it. In the discussion below, the following are covered: (1) why the Newton method works, (2) how the Newton method can be implemented in a computer solution for solving for a single variable from one equation and for solving for several unknowns from a system of this many equations, and (3) solution of example problems by the Newton method.

B.1.1 How the Newton Method Works

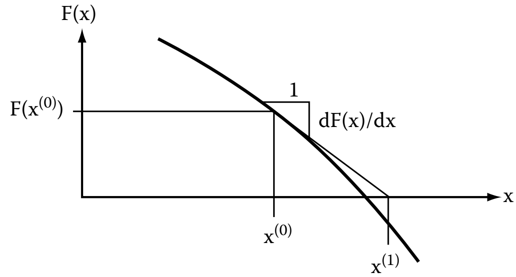
The first step in using the Newton method is to write the nonlinear equation(s) that is (are) to be solved in the form of a function(s) of the unknown(s) equal to zero. The equation(s) can now be represented as (a system of equations is described later)

$$F(x) = 0 \tag{B.1}$$

where F(x) represents the equation, and x represents the unknown, or the variable that you want a solution for.

The sketch below represents a graph of some general equation, F(x), in which the value that F(x) produces is plotted as the ordinate and the value of x as the abscissa. The solution, or the value of x that satisfies Equation B.1, is shown in this sketch where the curve intersects the x axis, i.e., produces a value of zero for F(x). Assume a starting value, or initial guess, for the solution denoted by $x^{(0)}$ is provided. In the notation used the superscript does not represent an exponent, but rather the iteration number. The superscript (0) indicates this is the 0th iteration. The value of x closer to the solution can be obtained by projecting a straight line that is tangent to the curve, down to the horizontal axis. This closer value of x is denoted by $x^{(1)}$. The slope of this straight line equals the value of the derivative of F(x), or dF(x)/dx evaluated at $x^{(0)}$. From the similar triangles shown in the sketch, one of which is the triangle formed by the straight line projecting to the x-axis and the other of which comes from the definition of the derivative, we see that

$$\frac{x^{(1)} - x^{(0)}}{F(x^{(0)})} = -\frac{1}{dF(x^{(0)})/dx}$$



The minus in front of the last term is needed because dF/dx is negative in the sketch. After solving for $x^{(1)}$ and replacing the zero by an m to indicate that the process will be repeated until the value of $x^{(m+1)}$ is close enough to the true solution to be accepted we have the Newton iterative formula

$$x^{(m+1)} = x^{(m)} - \frac{F(x^{(m)})}{dF(x^{(m)})/dx} \tag{B.2}$$

For some problems it may be convenient to differentiate the function F(x). If so, $dF(x^{(m)})/dx$ also becomes an equation that is evaluated in Equation B.2, just like $F(x^{(m)})$ is by substituting in the appropriate value of x as indicated by the iterative superscript. However, it is generally not necessary to actually differentiate the equation. Rather a numerical approximation of the derivative will be sufficient to cause Equation B.2 to converge to the correct solution, x. A numerical approximation is

$$\frac{dF(x^{(m)})}{dx} = \frac{F(x^{(m)}) - F(x^{(m)} - \Delta x)}{\Delta x}$$

When this numerical approximation is substituted into Equation B.2 the Newton iterative formula becomes

$$x^{(m+1)} = x^{(m)} - \frac{\Delta x F(x^{(m)})}{F(x^{(m)}) - F(x^{(m)} - \Delta x)} \tag{B.3}$$

The criteria to determine whether the iteration of the increments m should be discontinued because the answer is close enough to the desired zero of the function generally examines whether the absolute value of the quantity after the minus sign in Equations B.2 or B.3 is smaller than a preset error.

In a subsequent section the use of Equation B.3 in solving the Manning equation is discussed.

A more formal mathematical approach to developing the Newton iterative formula, Equation B.2 is to use Taylor series to evaluate the function F at $x^{(m+1)}$, where it is zero, from its value at $x^{(m)}$, which gives

$$F(x^{(m+1)}) = F(x^{(m)}) + (x^{(m+1)} - x^{(m)})\frac{dF(x^{(m)})}{dx} + \frac{(x^{(m+1)} - x^{(m)})^2}{2}\frac{d^2F(x^{(m)})}{dx^2} + O^3 = 0$$

where O^3 represents third-order and higher-order terms, i.e., those involving the difference in the two values of x cubed and higher. If the second-order and higher-order terms are discarded, and the above equation is solved for $x^{(m+1)}$, then Equation B.2 results. Since only the first-order term has been retained the function will not be zero here, and the Equation B.2 must be iteratively applied to drive the equation closer toward zero. It is worth noting that if the equation should be a linear equation such that second derivatives and all higher derivatives are zero then $F(x^{(m+1)})$ will be zero, and the use of Equation B.2 will give the exact solution to the equation. Therefore, should the Newton method be used to solve a linear equation, it will produce the solution after one iteration even though this may not be known until the second iteration is applied.

B.1.2 Solving Systems of Equations

The Newton iterative formula for solving a system of equations can be written as

$$\{x\}^{(m+1)} = \{x\}^{(m)} - [D]^{-1}\{F\}^{(m)} \tag{B.4}$$

You should note that Equation B.4 is similar to Equation B.2. The difference is that now {x} is an entire column vector of unknowns, {F} is an entire column vector of equations, and the division by the derivative has been replaced by the inverse of [D}. [D] is called the Jacobian and is a matrix of derivatives, i.e.,

$$[\mathbf{D}] = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_1}{\partial x_2} & \cdots & \frac{\partial F_1}{\partial x_n} \\ \frac{\partial F_2}{\partial x_1} & \frac{\partial F_2}{\partial x_2} & \cdots & \frac{\partial F_2}{\partial x_n} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial F_n}{\partial x_1} & \frac{\partial F_n}{\partial x_2} & \cdots & \frac{\partial F_n}{\partial x_n} \end{bmatrix}$$

Likewise, {x} and {F} are actually

$$\{\mathbf{x}\} = \left\{\begin{matrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{matrix}\right\}, \quad \{\mathbf{F}\} = \left\{\begin{matrix} F_1 \\ F_2 \\ \vdots \\ F_n \end{matrix}\right\}$$

Equation B.4 indicates that the Newton method solves a system of nonlinear equations by iteratively solving a system of linear equations because $[D]^{-1}\{F\}$ represents the solution of the linear system of equations

$$[\mathbf{D}]\{\mathbf{z}\} = \{\mathbf{F}\}$$

That is, the amount that is subtracted from the current estimate of the unknown vector is the solution {z} to the above linear system of equations. Thus, in practice the Newton method solves a system of equations by the iterative formula

$$\{\mathbf{x}\}^{(m+1)} = \{\mathbf{x}\}^{(m)} - \{\mathbf{z}\}$$

where $\{\mathbf{z}\}$ is the unknown vector that is obtained by solving $[\mathbf{D}]\{\mathbf{z}\} = \{\mathbf{F}\}$. Should the system actually contain only linear equations then the first iteration will produce the exact solution.

The actual development of Equation B.4 follows that given above, except that now since each equation depends upon a vector of x's, Taylor expansion to evaluate the equations where they are zero based on using x's, i.e., $\{x\}^{(m)}$, produces the following:

$$F_1^{(m+1)} = F_1^{(m)} + \Delta x_1 \frac{\partial F_1}{\partial x_1} + \Delta x_2 \frac{\partial F_1}{\partial x_2} + \cdots \Delta x_n \frac{\partial F_1}{\partial x_n} + O^2 = 0$$

$$F_2^{(m+1)} = F_2^{(m)} + \Delta x_1 \frac{\partial F_2}{\partial x_1} + \Delta x_2 \frac{\partial F_2}{\partial x_2} + \cdots \Delta x_n \frac{\partial F_2}{\partial x_n} + O^2 = 0$$

$$\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots$$

$$\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots$$

$$F_n^{(m+1)} = F_n^{(m)} + \Delta x_1 \frac{\partial F_n}{\partial x_1} + \Delta x_2 \frac{\partial F_n}{\partial x_2} + \cdots x_n \frac{\partial F_n}{\partial x_n} + O^2 = 0$$

When written using matrix notation, the above system of equations becomes

$$\begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ \vdots \\ \vdots \\ F_n \end{Bmatrix} + \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_1}{\partial x_2} & \cdots & \cdots & \frac{\partial F_1}{\partial x_n} \\ \frac{\partial F_2}{\partial x_1} & \frac{\partial F_2}{\partial x_2} & \cdots & \cdots & \frac{\partial F_2}{\partial x_1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial F_n}{\partial x_1} & \frac{\partial F_n}{\partial x_2} & \cdots & \cdots & \frac{\partial F_n}{\partial x_n} \end{bmatrix} \times \begin{Bmatrix} x_1^{(m+1)} - x_1^{(m)} \\ x_2^{(m+1)} - x_2^{(m)} \\ x_3^{(m+1)} - x_3^{(m)} \\ \vdots \\ \vdots \\ x_n^{(m+1)} - x_n^{(m)} \end{Bmatrix} = 0$$

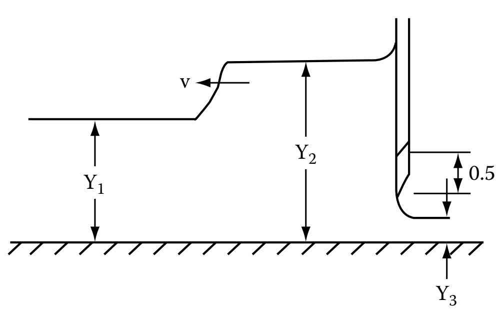
which can be denoted by

$$\{\mathbf{F}\} + [\mathbf{D}]^{(m)}(\{\mathbf{x}^{(m+1)} - \mathbf{x}^{(m)}\})$$

and this equation can be solved for $\{\mathbf{x}\}^{(m+1)}$ to produce Equation B.4.

Illustrative Example of Using the Newton Method to Solve a System of Equations:

As an example problem to apply the Newton method to solve a system of nonlinear equations, consider the problem of water flowing past a sluice gate. The gate is suddenly closed by 0.5 ft. Prior to dropping the gate, the velocity and depths in this channel just upstream from the gate were 5 fps and 4.5 ft, respectively. Find the depth, and velocity (and there from the discharge) just upstream from the gate as well as the speed at which the surge moves upstream when the gate is closed by the 0.5 ft. Assume the contraction coefficient of the gate does not change with the position of the gate.



To solve this problem, three simultaneous equations must be solved. The first equation comes from applying the energy principle (or Bernoulli's equation) across the gate. Prior to its closure, the depth upstream equals 4.5 ft and, therefore, to find the downstream depth, the following equation must be solved for Y_2:

$$Y_1 + \frac{q^2}{2gY_1^2} = Y_2 + \frac{q^2}{2gY_2^2}$$

Applying the Newton method for a single equation as described above produces as the solution $Y_2 = 1.530$ ft. Therefore, after the gate is dropped 0.5 ft, the new depth downstream from the gate must be 1.030 ft. If the energy equation is now written across the gate for the conditions after it has been dropped, the result is

$$F_1 = Y_2 + \frac{v_2^2}{2g} - Y_3 \frac{(Y_2V_2)^2}{2gY_g^2} = 0 \tag{B.5}$$

where $Y_3 = 1.030$ ft, the depth downstream after the gate was dropped. The remaining two equations that are needed for a solution come from moving upstream with the surge so that this moving observer sees a steady-state hydraulic jump in front of him. To this observer the continuity equation is

$$F_2 = (v + V_1)Y_1 - (v + V_2)Y_2 = 0 \tag{B.6}$$

and the momentum equation is

$$F_3 = \frac{1}{2}Y_1^2 - \frac{1}{2}Y_2^2 - \frac{\left(\left(V_1 + v\right)Y_1\right)^2}{g}\left(\frac{1}{Y_2} - \frac{1}{Y_1}\right) = 0 \tag{B.7}$$

The FORTRAN program listed below is designed to solve these three simultaneous equations with the Newton method. It calls on a matrix solver that has the coefficient matrix expanded by one column to contain the known vector and leaves the inverse in the additional columns. The first part of the FORTRAN listing is specific to this particular problem. However, the portion that numerically evaluates the derivatives in the Jacobian matrix is written more generally where N is used for the size of the matrix problem that is being solved. You should carefully study this listing to understand how the various tasks are performed. If input to this program is 4.5 1.03 5.5 2 6 8.327, then it produces the solution, $Y_2 = 5.146$ ft, $V_2 = 3.256$ fps, and $v = 8.327$ fps. Note that now the discharge past the gate is 16.76 cfs/ft of width, whereas before the gate was dropped, the discharge was 22.5 cfs/ft of width.

Program NEW3EQ.FOR

```
C Implements the NEWTON method in solving 3 eqs.
  REAL X(3),F(3),D(3,7)
  DATA N,N1/3,4/,DX/.001/,MAX/15/
 &,ERR/.0001/
  WRITE(5,*)' GIVE: V1,Y1,Y3,Y2,V2,V'
  READ(5,*)V1,Y1,Y3,X
  M=0
```

```
1      NT=0
5      F(1)=X(1)+X(2)**2/64.4-Y3-(X(1)*X(2))**2
     &/(64.4*Y3*Y3)
       F(2)=(X(3)+V1)*Y1-(X(3)+X(2))*X(1)
       F(3)=.5*(Y1*Y1-X(1)*X(1))-((V1+X(3))
     &*Y1)**2/32.2*(1./X(1)-1./Y1)
       IF(NT.NE.0) GO TO 15
       DO 10 I=1,N
10     D(I,N1)=F(I)
       X(1)=X(1)-DX
       NT=1
       GO TO 5
15     X(NT)=X(NT)+DX
       DO 20 I=1,N
20     D(I,NT)=(D(I,N1)-F(I))/DX
       NT=NT+1
       IF(NT.GT.N) GO TO 30
       X(NT)=X(NT)-DX
       GO TO 5
30     CALL INVERM(D,N)
       DIF=0.
       DO 40 I=1,N
       DIF=DIF+ABS(D(I,N1))
40     X(I)=X(I)-D(I,N1)
       M=M+1
       IF(DIF.GT. ERR .AND. M.LT.MAX) GO TO 1
       WRITE(5,*)' SOL IS',X
       END
       SUBROUTINE INVERM(A,N)
       REAL A(3,7)
       N1=N+1
       N2=N+2
       M=2*N+1
       DO 2 I=1,N
       DO 1 J=N2,M
       A(I,J)=0.
2      A(I,I+N1)=1.
       DO 10 I=1,N-1
       DIAG=A(I,I)
       A(I,I)=1.
       DO 5 J=I+1,M
5      A(I,J)=A(I,J)/DIAG
       DO 7 K=I+1,N
       FAC=A(K,I)
       A(K,I)=0.
       DO 7 J=I+1,M
7      A(K,J)=A(K,J)-FAC*A(I,J)
10     CONTINUE
       DIAG=A(N,N)
       A(N,N)=1.
       DO 12 J=N1,M
```

```
12     A(N,J)=A(N,J)/DIAG
       DO 20 I=N,2,-1
       DO 17 K=I-1,1,-1
       FAC=A(K,I)
       A(K,I)=0.
       DO 17 J=N1,M
17     A(K,J)=A(K,J)-FAC*A(I,J)
20     CONTINUE
       DO 40 I=1,N
       DO 40 J=1,N
40     A(I,J)=A(I,J+N1)
       RETURN
       END
```

Software packages are available for use with PCs as well as larger computers that make it easy to solve nonlinear as well as linear systems of equations. Mathcad is one such package that is designed to solve many if not most of the problems associated with engineering, as well as putting the results into engineering reports, etc. TK-Solve or MATLAB are similar, but not so all inclusive a software package. Spreadsheets such as EXCEL will solve linear systems of equations, and by using the Newton method you can get a solution to nonlinear systems. As you solve the problems in this book it will enhance your professional productivity to get acquainted with and utilize one or more of these software packages. If such software is available you will benefit by using it to solve the previous problem.

B.1.3 Implementing the Newton Method in Solving Manning's Equation

Manning's equation for a trapezoidal channel is

$$Q = \frac{C_u}{n} \frac{(bY + mY^2)^{5/3}\sqrt{S_0}}{\left(b + 2Y\sqrt{m^2+1}\right)^{2/3}} \tag{B.8}$$

where

Q is the flow rate
$C_u = 1.0$ for SI units and $C = 1.486$ for ES units
b is the bottom width, m is the side slope
Y is the depth of flow
S_0 is the slope of the channel bottom or energy line

As an exercise in which you implement some of the things described above, write a computer program which is capable of completely solving Manning's equation in a trapezoidal section. The program must carry out the correct computations to supply the correct unknown, and read-in the known quantities, depending upon the given problem. The following six cases exist:

Case	Known Quantities	Unknown Quantities
1	n,b,Y,m,S	Q
2	Q,b,Y,m,S	n
3	Q,n,Y,m,S	b
4	Q,n,b,m,S	Y
5	Q,n,b,Y,S	m
6	Q,n,b,Y,m	S

When Q, n, or S are unknown, Manning's equation can be solved explicitly. However, when b, Y, or m constitute the unknown, an implicit type solution is necessary. In solving Manning's equation by means of the Newton method, it is convenient to let ξ take on the value of the implicit unknown Y, m, or b so that Manning's equation becomes

$$F(\xi) = nQP^{2/3} - C_u A^{5/3} S^{1/2} = 0$$

For a trapezoidal channel the wetted perimeter P is

$$P = b + 2Y(m^2 + 1)^{1/2}$$

and the area A is given by

$$A = bY + mY^2 = (b + mY)Y$$

The derivatives of F(ξ) are given by

$$\frac{dF}{d\xi} = \frac{2}{3} nQP^{-3/5} \frac{\partial P}{\partial \xi} - \frac{5}{3} C_u S_0^{1/2} A^{2/3} \frac{\partial A}{\partial \xi}$$

where $\partial A/\partial \xi$ and $\partial P/\partial \xi$ will depend upon the unknown as given below:

Y-unknown	$\partial A/\partial Y = b + 2mY = T$	$\partial P/\partial Y = (m^2 + 1)^{1/2}$
m-unknown	$\partial A/\partial m = Y^2$	$\partial P/\partial m = 2mY/(m^2 + 1)^{1/2}$
b-unknown	$\partial A/\partial b = Y$	$\partial P/\partial b = 1$

B.1.3.1 Implementation of the Newton Method in Solving Manning's Equation in a Circular Channel

To provide you guidance in writing the computer program called for above, a Turbo Pascal program is listed below that solves Manning's equation in a circular section. In this program instead of evaluating the derivatives by mathematically differentiating the equation, they are evaluated numerically. You should study this listing over carefully to understand how the problem is solved.

Listing of a Turbo Pascal program to solve Manning's equation

```
Program Manning_circular;
Label L1,L2,L3;
Const
 UNK:array[1..5] of Char=('Q','n','S','Y','D');
 EX1:real=0.66666667;EX2:real=1.66666667;ERR:real=0.0001;
Var
 X: array[1..5] of Real;
 P,A,F,F1,DIF,C,BETA,ARG,TM1,TM2,DD,SX,XX: Real;
 I,IU,NCT: Integer;
 IFT: Boolean;
Function Expn(a,b:real):real;
 Begin
  If a>0 Then Expn:=exp(b*ln(a)) else if a=0 then Expn:=0 else
    if a<0 then begin writeln('Bad Expn argument'); Expn:=0; End;
 End;
```

```
Procedure Parmar;
  Begin
   ARG:=1-2*X[4]/X[5];If abs(ARG)<0.01 Then BETA:=1.57079633
else
   If ARG>0 Then BETA:=ArcTan(sqrt(1-ARG*ARG)/ARG) else
   BETA:=3.14159265-ArcTan(sqrt(abs(1-ARG*ARG))/abs(ARG));
   TM1:=Expn(X[5]*BETA,EX1);
   TM2:=C*Expn(0.25*X[5]*X[5]*(BETA-ARG*sin(BETA)),EX2);
  End;
Begin
L3:GoToXY(1,23);ClrEol;
  Write('Is Prob. in: 1-SI or 2-ES units?(Give 1 or 2) ');
   Readln(IU); If IU=1 Then C:=1 else C:=1.49;
   GoToXY(1,23); Write('Give number of unknown'); ClrEol;
   GoToXY(1,24); For I:=1 to 5 do Write(I,'-',UNK[I],', ');
     ClrEol;
   Repeat Readln(IU) until IU in [1..5];
   ClrScr; GoToXY(1,5); Write('Give value to knowns');
   For I:=1 to 5 do If I <> IU Then
    Begin
     GoToXY(1,I+6); Write(UNK[I],' = '); Readln(X[I]);
    End;
{ Solving or initializing unknown}
ClrScr;
Case IU of
 1: Begin Parmar; X[1]:=TM2*sqrt(X[3])/(TM1*X[2]); GoTo L1; End;
 2: Begin Parmar; X[2]:=TM2*sqrt(X[3])/(TM1*X[1]); GoTo L1; End;
 3: Begin Parmar; X[3]:=sqr(X[1]*X[2]*TM1/TM2); GoTo L1; End;
 4: Begin X[4]:=0.005*X[5]; DD:=0.05*X[5];XX:=X[1]*X[2];
   SX:=sqrt(X[3]);Parmar;
      F1:=XX*TM1-SX*TM2;
      repeat
        F:=F1; X[4]:=X[4]+DD; If X[4] > X[5] then X[4]:=X[5];
        Parmar;F1:=XX*TM1-SX*TM2;
      until (F*F1 <= 0) or (X[4]=X[5]);
      X[4]:=X[4]-DD/2;
    End;
 5: Begin X[5]:=1.01*X[4];DD:=0.05*X[4];XX:=X[1]*X[2];
   SX:=sqrt(X[3]); Parmar; F1:=XX*TM1-SX*TM2;
      repeat
        F:=F1; X[5]:=X[5]+DD;
        Parmar;F1:=XX*TM1-SX*TM2;
      until F*F1 <= 0;
      X[5]:=X[5]-DD/2;
    End;
End;
{ Implements Newton method}
NCT:=0;
repeat
  IFT:=true;
```

```
L2:Parmar;
   F:=XX*TM1-SX*TM2;
   If IFT then Begin F1:=F; IFT:=false; X[IU]:=X[IU]-0.01;
     GoTo L2; End;
   NCT:=NCT+1;
   DIF:=0.01*F1/(F1-F);
   X[IU]:=X[IU]+0.01-DIF;
until (ABS(DIF)<ERR) or (NCT>15);
If NCT>15 then Writeln('Did not converge - DIF= ',DIF);
{ Writes out solution}
L1: GoToXY(1,5); Writeln('Solution for ',UNK[IU],' = ',X[IU]);
   GoToXY(1,8); Writeln('Problem variables');
   For I:=1 to 5 do Begin GoToXY(1,I+8);
     Writeln(UNK[I],' = ',X[I]); End;
  GoToXY(1,23);
   Writeln('Do you want to solve another problem?(0-no,1-yes) ');
  Readln(IU); ClrScr;
  If IU = 1 Then GoTo L3;
End.
```

B.2 LAGRANGE'S INTERPOLATION FORMULA

Measurements provide discrete data relating two or more variables, one of which may be considered the independent variable, in general denoted by x, and the other the dependent variable, in general denoted by y. An example is the measurement of the cross sections of a river or channel (called transect data) in which the position across the channel is x and the distance down to the bottom from the highest bank is y. Such applications fall under the subject of the geometry of natural channel and are dealt with in Appendix A. Another example is the headloss produced by a control gate as a function of its setting. For this application the independent variable x is the gate setting, and the dependent variable y may be the headloss Δh, the flow rate Q, or some other quantity of interest. The operation of a channel pumping station is a similar example with the head difference Δh being positive instead of negative. For these and many other applications, the data available consist of a tabular list of the variables as shown below.

x_0	x_1	x_2	x_3	…	x_{m-1}	x_m
y_0	y_1	y_2	y_3	…	y_{m-1}	y_m

To effectively use this data it is desirable to conveniently obtain the value of y for any value of x between the tabular entries. The simplest approach is to linearly interpolate between the entries. A better fit of the data passes a polynomial of some higher degree through some selected number of values and uses it to interpolate within this range of values. A polynomial of degree n will pass through n + 1 pairs of data points. Thus, a straight line or polynomial of degree 1 passes through two pairs of points; a parabola, or polynomial, of degree two passes through three pairs of points, etc.

One method for obtaining this polynomial, and thereafter doing the interpolation, is to define a system of linear equations by substituting these n + 1 pairs of points into the general polynomial equation

$$y = P(x_n) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n$$

and then solve this system of n + 1 equations for the n + 1 unknown coefficients. Math packs for pocket calculators generally contain such linear equation solvers, or the subroutine INVERM in the above FORTRAN listing that utilizes the Newton method may also be used. A better alternative is to use an interpolation formula to either produce the polynomial equation or just do the interpolation. The development of Lagrange's interpolation formula is given below for this purpose.

Consider the polynomial

$$y = P(x_n) = b_0(x)y_0 + b_1(x)y_1 + b_2(x)y_2 + \cdots + b_n(x)y_n$$

where the b's are each polynomials of degree n defined as

$$b_0(x) = \frac{(x - x_1)(x - x_2)\cdots(x - x_n)}{(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_n)}$$

$$b_1(x) = \frac{(x - x_0)(x - x_2)\cdots(x - x_n)}{(x_1 - x_0)(x_1 - x_2)\cdots(x_1 - x_n)}$$

$$b_2(x) = \frac{(x - x_0)(x - x_1)(x - x_3)\cdots(x - x_n)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)\cdots(x_2 - x_n)}$$

$$\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots$$

$$\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots$$

$$b_n(x) = \frac{(x - x_0)(x - x_1)\cdots(x - x_{n-1})}{(x_n - x_0)(x_n - x_1)\cdots(x_n - x_{n-1})}$$

Note that the b's are polynomials of degree n because in carrying out the multiplication in each numerator, an x^n will occur that is multiplied by a constant determined by the values in the denominator consisting of known x's with a subscript. The desire is to have the polynomials defined by the b's equal zero at all tabular points except at the point corresponding to its subscript where it should equal one, or

$$b_s(x_k) = \begin{matrix} 0 & \text{for } k \neq s \\ 1 & \text{for } k = s \end{matrix}$$

Because, if this is the case the polynomial $y = P(x_n)$ above will equal the y's in the table when x takes the value of the corresponding x of the tabular data. An examination of the b's indicates they do in fact equal zero at all points of tabular data except at the point with the same subscript where $b_s = 1$.

Therefore, substituting the b's into the polynomial equation gives Lagrange's interpolation formula

$$\begin{aligned} y = P(x) = {} & \frac{(x - x_1)(x - x_2)\cdots(x - x_n)}{(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_n)} y_0 + \frac{(x - x_0)(x - x_2)\cdots(x - x_n)}{(x_1 - x_0)(x_1 - x_2)\cdots(x_1 - x_n)} y_1 \\ & + \frac{(x - x_0)(x - x_1)\cdots(x - x_{k-1})(x - x_{k+1})\cdots(x - x_n)}{(x_k - x_0)(x_k - x_1)\cdots(x_k - x_{k-1})(x_k - x_{k+1})\cdots(x_k - x_n)} y_k \\ & + \frac{(x - x_0)(x - x_1)\cdots(x - x_{n-1})}{(x_n - x_0)(x_n - x_1)\cdots(x_n - x_{n-1})} y_n + R_e \end{aligned} \quad \text{(B.9)}$$

where R_e is a remainder term that is defined in books dealing with numerical methods.

Example

As an example of the use of this interpolation formula, assume that it is desirable to define a pump curve around its normal capacity by a second-degree polynomial equation for use in a computer program so that the head the pump produces can readily be computed for any flow rate. In other words, the coefficients a, b, and c in the equation below are sought:

$$h_p = a + bQ + cQ^2$$

From the pump curve the following values are obtained (the middle point represents the normal capacity, i.e., the point of maximum efficient operation):

Q (cfs)	200	320	450
h_p (ft)	54.5	50.1	38.9

For this problem the flow rate Q is the independent variable, and h_p the dependent variable. Substituting the above three pairs of values into Equation B.9 gives

$$h_p = \frac{(Q-320)(Q-450)}{(200-320)(200-450)}54.5 + \frac{(Q-200)(Q-450)}{(320-200)(320-450)}50.1 + \frac{(Q-200)(Q-320)}{(450-200)(450-320)}38.9$$

which after some algebra can be rewritten as

$$h_p = 0.00622513\,Q^2 - 4.10873\,Q + 627.2415$$

B.2.1 Implementation of Lagrange's Formula in a Computer Program

It is relatively easy to program Equation B.9 into a computer program to carry out the desired interpolation. The use of the concept of divided differences, described in most books dealing with numerical methods, makes for a more efficient algorithm. Below is the listing of a Turbo Pascal program that prompts for the number of pairs of x y data that are to be interpolated or extrapolated, then asks for the degree of polynomial fit to use and the x value for which y is to be obtained, and finally writes out this solution:

Program LAGR.PAS

```
Program Lagrange_Interpolate;
Var
 x,y:Array[1..50] of Real; { x-independent var., y-dependent var.}
 BC,BD:Array[0..10] of Real; {work arrays used in interpolation}
 I,J,Deg,Io,Ie,Pts,S:Integer; Ans:Char;
   {Deg-degree of poly.,Pts-no. pairs}
 Xi,DX,Yi,BO,BP,BCC,DQ,RS:Real;
   {Xi & Yi are the interpolated values}
Label L1,L2;
BEGIN
 Repeat
  Writeln('Give no. pts followed by this many pairs of x y
    values');
  Readln(Pts);For I:=1 to Pts do Readln(x[I],y[I]);
  Write('Give Deg. of polyn. & Xi ');Readln(Deg,Xi);
```

```
 If Deg>=Pts then Writeln('Deg must be < pts'); until Deg<Pts;
L1: I:=0;
 Repeat I:=I+1 until (x[I] >= Xi) or (I=Pts);
  If x[I]=Xi then begin Yi:=y[I];RS:=0; GoTo L2 end;
 Io:=I−1−Deg div 2; Ie:=Io+Deg;
 If Io<1 then begin Ie:=Ie−Io+1;Io:=1 end;
 If Ie>Pts then begin Io:=Pts−Deg;Ie:=Pts end;DX:=1.E30;
 For I:=0 to Deg do
   Begin If Abs(Xi−x[Io+I])<DX then begin S:=Io+I;
     DX:=Abs(Xi−x[Io+I]) end;
   BC[I]:=y[Io+I];BD[I]:=y[Io+I] End;
 Yi:=y[S]; S:=S−1;
 For J:=1 to Deg do Begin
   For I:=0 to Deg−J do begin BO:=x[Io+I]−Xi;BP:=x[Io+I+J]−Xi;
    BCC:=BC[I+1]−BD[I];DQ:=BCC/(BO−BP);BC[I]:=BO*DQ;
      BD[I]:=BP*DQ end;
   If 2*(S−Io)<Deg−J then RS:=BC[S−Io+1] else begin RS:=BD[S−Io];
     S:=S end;
   Yi:=Yi+RS; End;
L2:Writeln('x= ',Xi:10:3,' y=',Yi:10:3,' Remainder=',RS:12:5);
   Write('Should another value be interpolated? (Y or N) ');
     Readln(Ans);
   If (Ans='Y') or (Ans='y') then begin Write('For x= ');
     Readln(Xi);
     GoTo L1 end;
END.
```

The use of this program can be illustrated by the following example:

Current meter measurements have been made at a stream-gaging site to determine the flow rate at different stages, i.e., water surface elevations, giving the data below. It is desirable to be able to determine the flow rate occurring in the river from a reading of the staff gage that gives the river stage.

Stage (ft)	1.97	2.75	3.16	3.95	5.10	5.90	7.06	7.85
Q (cfs)	42.6	82.7	109.2	173.1	300.0	412.8	615.5	782.4

You might type the above program, or an equivalent FORTRAN or BASIC adaptation therefrom, into your PC and run it with the following stages to get the corresponding flow rates given below, based on a third-degree polynomial interpolation.

Stage (ft)	**Discharge (cfs)**
3.25	115.6
4.15	192.2
8.25	1,035.9

The above program might be modified so that the above stage discharges were entered as constant values for the arrays x and y, and the degree of the interpolation fixed, so that all that is entered is a stage for which the discharge is desired.

Problems (for you to work)

1. A monitoring station telemeters data giving the stage of a river flow every 30 min. Write a computer program that (a) reads this stage from the data file where it exists, (b) computes the corresponding flow rate, and (c) integrates these flow rates over each day's period of time and determines the acre feet of volume passing this gaging station each day.
2. A 0.75 m diameter pipe is used to take water from a reservoir whose depth changes. The pipe has a bottom slope of 0.0008 for a long-distance downstream from the reservoir, and a Manning's n = 0.018. The entrance loss coefficient equals 0.12. Determine the discharge in increments of 0.05 m for reservoir water surface level from 0.1 to 0.7 m above the bottom of the pipe.

   For each reservoir elevation you could solve Manning's equation simultaneously with the energy equation. However, assume this has been done with the results in the following table.

Reserv. w.s. elev. above pipe bottom	0.1 m	0.4 m	0.6 m	0.69 m
Discharge into pipe, Q (m^3/s)	0.01	0.17	0.32	0.36

3. A gate is used to control the flow rate from a constant level reservoir into a canal. The four measurements below have been made giving the flow rate for given gate positions. Develop a pocket calculator program or computer program and from this develop a table that tells what the gate setting should be for any desired flow rate.

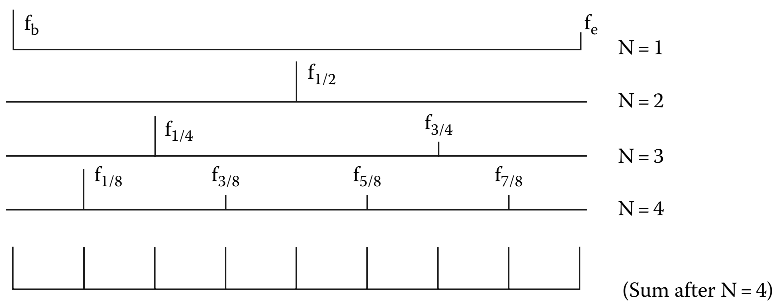
Gate setting (ft)	1.00	2.50	3.50	5.00
Flow rate (cfs)	181.0	457.5	643.6	923.0

B.3 NUMERICAL INTEGRATION

B.3.1 Trapezoidal Rule

The trapezoidal rule states that an approximation of the integral of a function f(x) from x_b (beginning value of the independent variable) to x_e (ending value of the independent variable) equals the average of the function evaluated at these two limits time the interval, or $\Delta F = \int f(x)dx = (x_e - x_b)\{f(x_b) + f(x_e)\}/2$. The accuracy of the numerical integration depends upon the size of Δx. In other words, to satisfy an error requirement the Δx used in the process needs to be made small enough. How can it be determined what small enough is? A good means for determining whether Δx is small enough is when the integration obtained using an increment $\Delta x/2$ produces the same final answer as when Δx was used. In other words, the numerical integration can be repeated by reducing the Δx (generally by 1/2) and the results compared with those previously obtained, until the difference is less than an error criterion. If this process is implemented as stated above, without considering how the number of computations can be minimized, a lot more arithmetic will be done than is required. Algorithms are discussed in the following paragraphs that will allow reducing the interval size without losing the benefit of previous arithmetic. To facilitate this discussion a first-order approximation, i.e., the trapezoidal rule will be applied.

Applying this equation repeatedly over consecutive intervals Δx where the total interval of the integration has been divided into N intervals, i.e., $\Delta x = (x_e - x_b)/N$ results in the following: $F(x_e) - F(x_b) = \Delta x\{f_0/2 + f_1 + f_2 + f_3 + \cdots + f_{N-1} + f_N/2\}$ or the functions at all intermediate points are added except the first and the last, which are divided by 2 before being added. Note $f_1 = f_b$ (the function being integrated evaluated at the beginning) and $f_N = f_e$ (the function being evaluated at the end.) The question we want to answer is how can this extended trapezoidal rule equation be implemented repeatedly with new Δxs equal to one-half their previous values, and not lose previous evaluations of the function, i.e., previous arithmetic. To visualize how such an algorithm can be developed consider first the coarsest implementation of the trapezoidal rule as the average of the function at the two end points x_b and x_e multiplied by $(x_e - x_b)$ as shown in the diagram below in which N = 1. When the range of integration is divided into two intervals, $\Delta x = (x_e - x_b)/2$, the function must be evaluated at one additional point, the midpoint as shown for N = 2 in the sketch below. The application of the extended trapezoidal rule will multiply the previous end values by one-half because Δx is now one-half as large as previously and this result added to the value of the function at the midpoint times Δx. Upon dividing the range of integration into four intervals, so $\Delta x = (x_e - x_b)/4$, the function needs to be evaluated at two additional points, i.e., at $x = x_b + (\Delta x)_{i-1}/2$ and at $x = x_b + (\Delta x)_{i-1}/2 + (\Delta x)_{i-1}$, where $(\Delta x)_{i-1}$ is the previous increment. These two additional points are shown on the line associated with N = 3. Associated with N = 4, the function needs to be evaluated at four additional points, and this process continues. As shown in the sketch the sum of all the evaluations provide all the values needed to implement the extended trapezoidal rule.



$$\frac{(x_e - x_b)\{f_b + f_e\}}{2} = \frac{\Delta x_1\{f_b + f_e\}}{2} = \text{Value}_1$$

$$\frac{(x_e - x_b)}{2}\left\{\frac{f_b}{2} + f_{1/2} + \frac{f_e}{2}\right\} = \Delta x_1 \frac{\{f_b/2 + f_{1/2} + f_e/2\}}{2}$$

$$= \frac{(\text{Value}_1 + \Delta x_1 f_{1/2})}{2} = \text{Value}_2 \text{ with } \Delta x_2 = \frac{(x_e - x_b)}{2}$$

$$\frac{(x_e - x_b)}{4}\left\{\frac{f_b}{2} + f_{1/4} + f_{1/2} + f_{3/4} + \frac{f_e}{2}\right\} = \frac{\{\text{Value}_2 + \Delta x_2(f_{1/4} + f_{3/4})\}}{2}$$

$$= \text{Value}_3 \text{ with } \Delta x_3 = \frac{(x_e - x_b)}{4}, \text{etc.}$$

Implementation of trapezoidal rule with every decreasing increments Δx equal to one-half the previous increment so that only functions at the new points are evaluated as Δx is decreased.

This process could continue for Ns as large as needed until the evaluations of the integral between consecutive increases of N produce the same value within the error limit selected.

The FORTRAN and C listings below implement this algorithm, and as such constitute a subroutine that numerically evaluates an integral using the trapezoidal rule. You should carefully study these statements and relate them to the discussion of the extended trapezoidal rule formula and the sketch above the listing, so you fully appreciate how each step of the algorithm is carried out.

Listing of program TRAPR.FOR It integrates to the selected error criteria by repeatedly reducing Δx.

Listing of FORTRAN program, TRAPR.FOR

```
      SUBROUTINE TRAPR(EQUAT,XB,XE,VALUE,ERR,MAX)
      EXTERNAL EQUAT
      EV=-1.E30
      VALUE=.5*(XE-XB)*(EQUAT(XB)+EQUAT(XE))
      M=1
      I=1
10    I=I+1
      DELX=(XE-XB)/FLOAT(M)
      X=XB+.5*DELX
      SUM=0.
      DO 20 J=1,M
      SUM=SUM+EQUAT(X)
20    X=X+DELX
      VALUE=.5*(VALUE+(XE-XB)*SUM/FLOAT(M))
      M=2*M
      IF(ABS(VALUE-EV).LT.ERR*ABS(EV)) RETURN
      EV=VALUE
      IF(I.LT.MAX) GO TO 10
      WRITE(*,*)' Failed to satisfy error req.'
     &, VALUE-EV
      RETURN
      END
```

Listing for C-program, TRAPR.C

```
#include <stdlib.h>
#include <math.h>
extern equat(float x);
float trapr(float xb,float xe,float err,int max){
 float ev=-1.e30,delx,sum,x,value;
 int m=1,i=1,j;
 value=.5*(xe-xb)*(equat(xb)+equat(xe));
 do{ i++;
  delx=(xe-xb)/(float)m;
  x=xb+.5*delx; sum=0;
  for(j=0;j<m;j++){sum+=equat(x); x+=delx;}
  value=.5*(value+(xe-xb)*sum/(float)m); m*=2;
  if(fabs(value-ev)<err*fabs(ev)) return value;
  ev=value;
 }while(i<max);
 printf("Failed to satisfy error req. %f\n",value-ev);
 return value;
}
```

To use this subroutine two other programs are needed. One is a FUNCTION subroutine (with the name EQUAT as the first argument in the call to TRAPR) that provides the value of the function being integrated evaluated at the argument X, i.e., a program that has the first statement as follows:

FUNCTION EQUAT(X) (or whatever name is used for the first argument in the main program when it calls TRAPR), and two, a main program that properly calls TRAPR and writes the solution, etc. The arguments for TRAPR are as follows:

1. EQUAT (EXTERNAL)—The name of the FUNCTION SUBPROGRAM that evaluates the function to be integrated at the given X value.
2. XB (REAL)—The beginning value of the independent variable.
3. XE (REAL)—The ending value of the independent variable, i.e., the integral is to be evaluated from XB to XE.
4. VALUE (REAL)—The value of the integral is returned in this real variable.
5. ERR (REAL)—The error criteria. The increment Δx will be repeatedly reduced by one-half until the absolute difference between the two values of the integral are less than ERR times the absolute value of the previously evaluated integral. A value for ERR equal to 1×10^{-6} is near the limit that can be used with 32-bit arithmetic and not have truncation error cause the algorithm to fail to meet the criteria.
6. MAX (INTEGER)—The maximum number of reductions in Δx that will be allowed.

Note: If the function can be defined as a single statement in the declaration portion of the main FORTRAN program, then a function statement can be used in place of the FUNCTION EQUAT. This approach is used in the first example problem below.

B.3.1.1 Simpson's Rule

Simpson's rule is a double-interval integration formula, i.e., it evaluates the integral over $2\Delta x$ and gives a second-order approximation by passing a second-degree polynomial through three consecutive evenly spaced points. Simpson's Rule is $\Delta F_{i-1}^{i+1} = \Delta x\left\{f_b + 4f_m + f_e\right\}/3$, where f_m is the value of the function being integrated at the midpoint of the interval $2\Delta x$, or at $x = \Delta x$. Simpson's rule can be implemented by using an algorithm that compares the result from a new interval size Δx with that obtained from $2\Delta x$, as with the trapezoidal rule. To show this, consider first using the entire range of the independent variable for the interval, or $\Delta x_o = x_e - x_b$. An approximate value for the integral is $VALU1_o = \Delta x_o(f_b + f_e)/2$. Dividing this interval by 2, so $\Delta x_1 = \Delta x_o/2$, and evaluating the function at the midpoint to get f_m, the approximate integral from x_b to x_e is $VALU1_1 = \Delta x_1(f_b + 2f_m + f_e)/2$. Now note that Simpson's rule results by multiplying this new value by 4, subtracting the old value and dividing the result by 3, or $VALUE = (4VALU1_1 - VALU1_o)/3 = \{4\Delta x_1(f_b + 2f_m + f_e)/2 - \Delta x_1(f_b + f_e)\}/3 = \Delta x_1\{2f_b + 4f_m + 2f_e - (f_b + f_e)\}/3 = \Delta x_1\{f_b + 4f_m + f_e\}/3$. Notice that $VALU1_o$ and $VALU1_1$ are obtained from the trapezoidal rule. This algorithm applies for the subsequent halving of Δx_i, i = 2, 3, etc., so the approximate value of the integral for Simpson's Rule, VALUE, for each of these new halved intervals is evaluated as $VALUE = (4VALU1_i - VALU1_{i-1})/3$ where VALU1s are obtained by the trapezoidal rule algorithm with $VALU1_i$ coming from the new Δx and $VALU1_{i-1}$ from the last Δx. The FORTRAN and C listings, SIMPR, implement the Simpson's Rule to numerically integrate.

Listing of program SIMPR.FOR. It integrates to the selected error criteria by repeatedly reducing Δx.

```
      SUBROUTINE SIMPR(EQUAT,XB,XE,VALUE,ERR,MAX)
      EXTERNAL EQUAT
      EV1=-1.E30
      EV=-1.E30
      VALU1=.5*(XE-XB)*(EQUAT(XB)+EQUAT(XE))
      M=1
      I=0
10    I=I+1
      DELX=(XE-XB)/FLOAT(M)
      X=XB+.5*DELX
```

```
      SUM=0.
      DO 20 J=1,M
      SUM=SUM+EQUAT(X)
20    X=X+DELX
      VALU1=.5*(VALU1+(XE-XB)*SUM/FLOAT(M))
      M=2*M
      VALUE=(4.*VALU1-EV1)/3.
      IF(ABS(VALUE-EV).LT.ERR*ABS(EV)) RETURN
      EV=VALUE
      EV1=VALU1
      IF(I.LT.MAX) GO TO 10
      WRITE(*,*)' Failed to satisfy error req.',
     & VALUE-EV
      RETURN
      END
```

```
#include <stdlib.h>
#include <math.h>
extern equat(float x);
float simpr(float xb,float xe,float err,int max){
 float ev=-1.e30,ev1=-1.e30,delx,sum,x,value,valu1;
 int m=1,i=1,j;
 valu1=.5*(xe-xb)*(equat(xb)+equat(xe));
 do{ i++;
  delx=(xe-xb)/(float)m;
  x=xb+.5*delx; sum=0;
  for(j=0;j<m;j++){sum+=equat(x); x+=delx;}
  valu1=.5*(valu1+(xe-xb)*sum/(float)m); m*=2;
  value=(4.*valu1-ev1)/3.;
  if(fabs(value-ev)<err*fabs(ev)) return value;
  ev=value; ev1=valu1;
 }while(i<max);
 printf("Failed to satisfy error req. %f\n",value-ev);
 return value;
}
```

If the function that is called to evaluate the argument being integrated is to have a name other than **`equat`**, then function simpr should read: `float simpr(float(*eq)(float x),float xb,float xe, float err, int max){ . .`

Its arguments are identical to those described above for the subroutine TRAPR. In fact what would be changed to use it with a program that previously called on TRAPR is the name change to SIMPR.

EXAMPLE PROBLEM B.1

Integrate the function $f(x) = x^2(x^2 - 2)\sin(x)$ between the limits of 0 and $\pi/2$ using first the trapezoidal rule and then Simpson's rule and compare the results with the exact integral.

Solution

The integral is $F(x) = 4x(x^2 - 7)\sin(x) - (x^4 - 14x^2 + 28)\cos(x)$. The following programs have been written to implement the solution to this problem:

FORTRAN MAIN program TRAP1.FOR and SUBROUTINE EQUAT to solve problem

```
      PARAMETER (NMAX=21,A=0,B=1.5707963,ERR=1.E-5)
      EXTERNAL EQUAT
      CLOINT(X)=4.*X*(X**2-7.)*SIN(X)-(X**4-14.*X**2+28.)
```

```
     &*COS(X)
      CALL TRAPR(EQUAT,A,B,VALUE,ERR,NMAX) ! The name is changed
     &to SIMPR to use Simpson's Rule
      WRITE(*,*) VALUE,CLOINT(B)-CLOINT(A)
      END
      FUNCTION EQUAT(X)
      EQUAT=X**2*(X**2-2.)*SIN(X)
      RETURN
      END
```

C program TRAP1.C to solve problem

```
#include <stdlib.h>
#include <math.h>
entern trapr(float x);
float equat(float x){
return x*x*(x*x-2.)*sin(x);}
float cloint(float x){
return 4.*x*(x*x-7.)*sin(x)-(pow(x,4)-14.*x*x+28.)*cos(x);}
void main(void){ int max=21;
 float xb=0.,xe=1.5707963,err=1.e-6;
 printf("%f %f\n",trapr(xb,xe,err,max),cloint(xe)-cloint(xb));
}
```

The results from executing the program are:
Using the trapezoidal rule

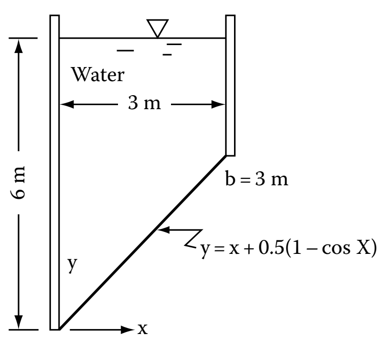
−4.791531E−1 (numerical value) −4.7915984E−1 (using known integral)

Using the Simpson's rule

−4.791583E−1 (numerical value) −4.7915984E−1 (using known integral)

EXAMPLE PROBLEM B.2

Find the vertical component of force (including its location) on the bottom surface that is 3 m long with water standing to a height of 6 m. The surface is defined by the equation y = x + 0.5(1 − cos x). The distance between vertical walls is 3 m.



Solution

The method of components allows the hydrostatic fluid force on the bottom of the tank to be computed by finding the volume, i.e., area times length, b = 3 m, and multiply this by the specific weight of the fluid, since the force equals the weight of water vertically above the surface. Since the bottom of the tank is given as a function of x the area will be determined by numerically integrating a differential area consisting of dA = (6 − y)dx or A = ∫(6 − x − 0.5(1 − cos x)dx. Thereafter, the centroid of this area will be determined from Ax_c = ∫x(6 − y)dx. The programs used to obtain the area and the first moment of area, are given below.

Program listing, SIMP1.FOR that integrates area

```
      PARAMETER (NMAX=21,A=0,B=3.,ERR=1.E-5)
      EXTERNAL EQUAT
      CALL SIMPR(EQUAT,A,B,VALUE,ERR,NMAX)
      WRITE(*,*) VALUE
      END
      FUNCTION EQUAT(X)
      EQUAT=6.-X-(1.-COS(X))
      RETURN
      END
```

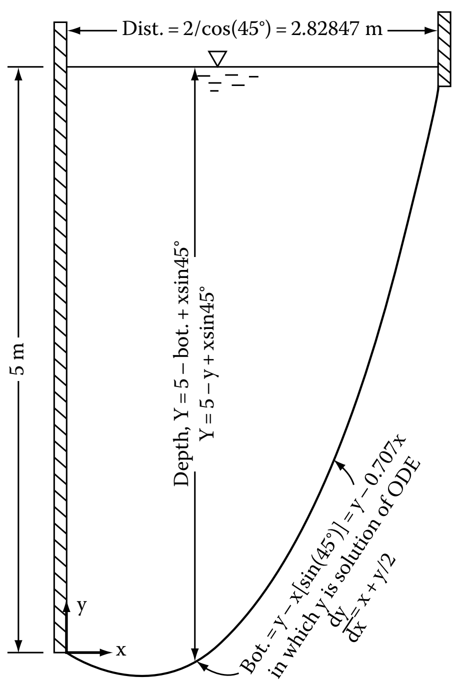
Program SIMP2.FOR that finds 1st moment of Area

```
      PARAMETER (NMAX=21,A=0,B=3.,ERR=1.E-5)
      EXTERNAL EQUAT
      CALL SIMPR(EQUAT,A,B,VALUE,ERR,NMAX)
      WRITE(*,*) VALUE
      END
      FUNCTION EQUAT(X)
      EQUAT=X*(6.-X-(1.-COS(X)))
      RETURN
      END
```

The solution for the area is 10.641 m^2 and the solution for the first moment of area is 11.9337 m^3. Therefore, the force on the bottom of the tank is $F = \gamma Vol = \gamma Ab = 9.806(10.641)3 = 313$ kN. It acts at a position $x_c = 11.9337/10.641 = 1.121$ m from the origin, the first vertical wall, and the force is in the downward vertical direction.

EXAMPLE PROBLEM B.3

The bottom of the tank is defined by the ODE dy/dx = x + y/2 minus x sin(45°). In other words, the depth of water at any position x is 5 + x sin(45°) − y where y is the solution to the above ODE. The gate exists between two vertical walls 2/cos(45°) = 2.828427 m apart as shown, and the tank is 5 m long and contains water that is 5 m deep at the left wall. Find the vertical component of force (and its location) at the bottom.



Solution

This problem requires that the ODE dy/dx = x + y/2 be solved to determined the shape of the bottom. After the shape of the bottom is known, the area can be determined by integrating from x = 0 to x = 2/cos45° the depth of water in the tank or $A = \int(5 + x\sin45° - y(x))dx$. This will be determined by writing a computer program that accomplishes this by calling on a subroutine that solves the ODE to the desired position x and another that accomplishes the numerical integration. A program listing that accomplishes this is given below. In this program the FUNCTION EQUAT which defines the function that is to be integrated must first have the value of y(x) in the above equation, and this value is obtained by calling on the subroutine ODESOL, which is a differential equation solver. The differential equation solver, ODESOL, in turn needs to have a subroutine that evaluates the derivative for any x that is passed to it, and therefore the program listing also contains the subroutine DYX to accomplish this task.

Listing for program PRESFOR.FOR to solve this example problem

```
      EXTERNAL EQUAT
      COMMON NGOOD,NBAD,KMAX,ICOUNT,DXSAVE
      COMMON /TRAS/ H,YO,X1
      WRITE(*,*)' Give: X1,X2,H,YO,GAMMA,ERR,MAX'
      READ(*,*) X1,X2,H,YO,GAMMA,ERR,MAX
      CALL SIMPR(EQUAT,X1,X2,VALUE,ERR,MAX)
      WRITE(*,"(' Area=',F10.3,/,' Force =',F10.3)") VALUE,
     &VALUE*GAMMA
      END
      FUNCTION EQUAT(X)
      EXTERNAL DYX
      REAL W(1,13),XP(1),YP(1,1),DY(1),Y(1)
      COMMON NGOOD,NBAD,KMAX,ICOUNT,DXSAVE
      COMMON /TRAS/ H,YO,X1
      SIN45=SIN(.7853982)
      TOL=.00001
      H1=.01
      HMIN=1.E-8
      Y(1)=YO
      IF(ABS(X-X1).GT.1.E-5) CALL ODESOL(Y,DY,1,X1,X,TOL,H1,HMIN,1,
     &XP, YP,W,DYX)
      EQUAT=H+SIN45*X-Y(1)! for 1st moments of area this
     &statement is
      RETURN              ! EQUAT=X*(H+SIN45*X-Y(1))
      END
      SUBROUTINE DYX(X,Y,DY)
      REAL Y(1),DY(1)
      DY(1)=X+.5*Y(1)
      RETURN
      END
```

(You may wish to also write a program that defines the bottom of the tank, and the depth, or in other words produce a table of y, dy/dx, bottom and depth Y as a function of x.)

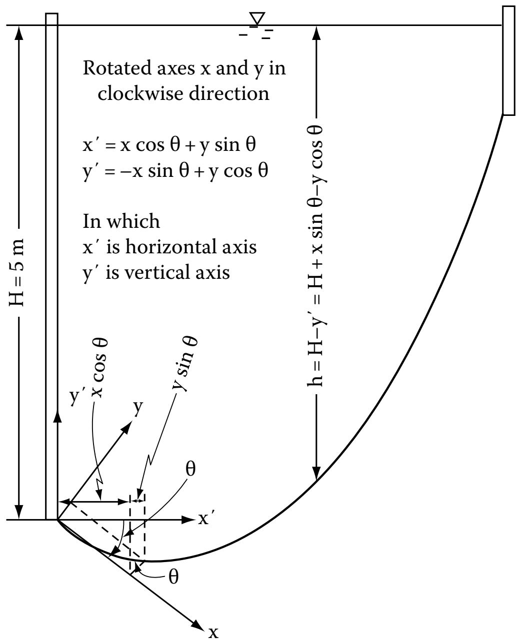
The input used to solve the problem consists of:

0 2.828427 5 0 9.806 1.e-6 21

The solution provided: Area = 11.378 m^2 and the vertical component of Force = 111.575 kN. To obtain the location of this force the first moment of area around the y axis must be evaluated ($Ax_c = \int x(5 + x*\sin45° - y)dx$). To accomplish this integration, the statement that evaluates the function needs to be changed, as given in the comment portion of this and the next line in the listing. The input might also eliminate the specific weight GAMMA since this is not needed, and only the results of the integration need to be written out. The value from this numerical integration is $Ax_c = 13.158\,m^3$ and, therefore, the vertical pressure force acts through the point $x_c = 13.158/11.378 = 1.156\,m$ from the first vertical wall.

EXAMPLE PROBLEM B.4

The bottom of the tank is defined by the ODE dy/dx = x + 0.5y from a coordinate system that is rotated downward 45° from the horizontal. The gate exists between two vertical walls 2 m apart as shown, and the tank is 5 m long and contains water that is 5 m deep at the left wall. Find the vertical component of force (and its location) on the bottom.



Solution

The program below obtains the solution. Because of the rotated coordinate system, relationships must be established between the movement in the x direction and in the horizontal direction, denoted by x′ in the above sketch. The differential area to integrate is dA = h dx′, where h is the distance from the bottom of the tank to the water surface, and from the above sketch is seen to be $h = H + x \sin\theta - y \cos\theta$, where H is the depth of water at the origin. Also from the sketch $x' = x \cos\theta + y \sin\theta$. Since for 45° the sine = cosine, one value will be used as the solution is programmed. While there are alternative approaches, for this problem the ODE will be solved first to obtain the shape of the bottom, and the results stored in an array, YY, and when needed the results from this array will be interpolated. The reason for doing this is that the numerical integration is in terms of x′, and the corresponding value of x (along the rotated axis) is needed, but the only way to determine this x is to note that x′/sin45° = x + y. For any x′, therefore, the table is searched for the entry where x + y is just larger than x′/sin45°, and then interpolated (linearly) to find x. This is done by noting that between two entries in the table $y = y_o + (\Delta y/\Delta x)(x - x_o)$, where subscript o denotes first entry. Thus, $x'/\sin 45° = y_o + (\Delta y/\Delta x)(x - x_o) + x$, or solving for x, $x = (x'/\sin 45° - y_o + (\Delta y/\Delta x)x_o)/(\Delta y/\Delta x + 1)$. Once x is determined, y is interpolated as $y = y_o + (x - x_o)/\Delta x(y_1 - y_o)$. Note in the program below that the ODE is now solved first in the MAIN program, and that the function EQUAT does the necessary interpolation to obtain h so that Simpson's Rule can properly evaluate the area. Giving as input 0 2 5 0. 9.801 1.e − 5 30, the solution gives Area = 9.437 m^2 per meter of basin length, resulting in a force = 92.491 kN/m, or $F_V = 5 \times 92.491 = 462.46$ kN. The first moment of area is determined by changing the statement as shown by the comment, and the result of the integration gives $x_cA = 8.824$, so the position of this vertical force is $x_c = 8.824/9.437 = 0.935$ m.

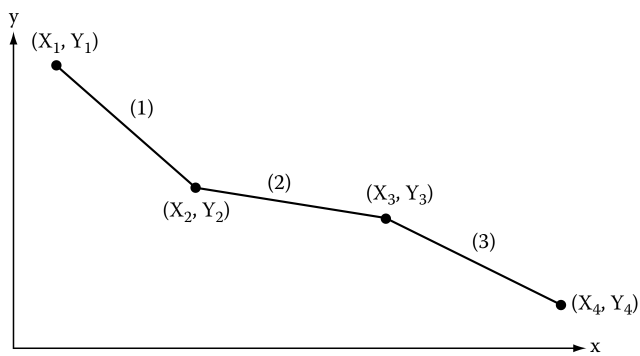
APPB4.FOR

```
      EXTERNAL EQUAT,DYX
      COMMON NGOOD,NBAD,KMAX,ICOUNT,DXSAVE
      COMMON /TRAS/XX(30),YY(30),H,SIN45,DELX,II
      REAL W(1,13),XP(1),YP(1,1),DY(1),Y(1)
      II=2
      WRITE(*,*)' Give: Xb,Xe,H,Yo,Gamma,ERR,MAX'
      READ(*,*) XB,XE,H,YO,GAMMA,ERR,MAX
      SIN45=SIN(.7853982)
      SXY=(XE-XB)/SIN45
      TOL=.000001
      H1=.01
      HMIN=1.E-8
      I=1
      DELX=(XE-XB)/50.
      XX(1)=XB
      YY(1)=YO
      Y(1)=YO
10    X=XX(I)+DELX
      CALL ODESOL(Y,DY,1,XB,X,TOL,H1,HMIN,1,XP,YP,W,DYX)
      I=I+1
      XX(I)=X
      YY(I)=Y(1)
      IF(X+Y(1).LT.SXY) GO TO 10
      CALL SIMPR(EQUAT,XB,XE,VALUE,ERR,MAX)
      WRITE(*,"(' Area=',F10.3,/,' Force=',F10.3)")
      &VALUE,VALUE*GAMMA
      END
      SUBROUTINE DYX(X,Y,DY)
      REAL Y(1),DY(1)
      DY(1)=X+.5*Y(1)
      RETURN
      END
      FUNCTION EQUAT(X)
      COMMON /TRAS/XX(30),YY(30),H,SIN45,DELX,II
      SX=X/SIN45
10    IF(XX(II)+YY(II).GT.SX) GO TO 20
      II=II+1
      GO TO 10
20    DYDX=(YY(II)-YY(II-1))/DELX
      XP=(SX-YY(II-1)+DYDX*XX(II-1))/(DYDX+1.)
      FAC=(XP-XX(II-1))/DELX
      YP=YY(II-1)+FAC*(YY(II)-YY(II-1))
      EQUAT=H+(XP-YP)*SIN45
C     EQUAT=(H+(XP-YP)*SIN45)*X
      RETURN
      END
```

B.4 SPLINE FUNCTIONS

B.4.1 Background

A common problem is to obtain values that represent a smooth curve through the original data, or table of X and Y values. These additional values may be needed to plot a smooth curve through the points, or allow any y to be determined for a specified x. In other words one wishes to interpolate (or possibly extrapolate) the original data. Spline functions are used for this purpose. The difference between splines and interpolation as described previously is that spline interpolation equations are obtained so that derivatives of y with respect to x (i.e., dy/dx, d^2y/dx^2, etc.) are also forced to be equal on both sides of the original (X_j, Y_j) pairs of values, and not just the values y. For example, consider the four pairs (X_j, Y_j), j = 1, 2, 3, and 4, shown on the graph below. If linear interpolation is used then straight lines will result between the data points. In other words, the interpolating function over the first segment is $y^{(1)} = a_1x + b_1$, where $a_1 = (Y_2 - Y_1)/(X_2 - X_1)$ and $b_1 = Y_1$ while x is in the interval X_1 to X_2, and while x is in the interval X_2 to X_3 $y^{(2)} = a_2x + b_2$, where $a_2 = (Y_3 - Y_2)/(X_3 - X_2)$ and $b_2 = Y_2$, etc. are the interpolation functions. Note that superscripts denote line segments and the subscripts on the X's and Y's denote points where pairs (X_j, Y_j) are given. Uppercase letters denote knowns and lower case letters denote unknowns. Obviously this linear interpolation function does not keep the derivatives on both sides of the points equal.



A means of maintaining first derivatives equal on both sides of the points is to use a second-order interpolation equation with this as the middle point. Lagrange's formula would be used for this interpolation if the spacing between the X values is not constant, or the Gregory Newton Forward or Backward, or the Stirling interpolation formula could be used if ΔX is constant. Using a second-order approximation would use three pairs of values that bracket the x for which the y is to be determined, and when x moves beyond the third X (or perhaps halfway between the second and third points) a new interpolation equation will be used by adding the next point and dropping the first point. At the position where the new interpolation equation is developed the first derivative is not generally continuous. Likewise, if a third-order interpolation equation is used, the second derivatives at the position x where the new interpolation equation is developed will not necessary be continuous, etc. Spline functions are obtained by making derivatives, one less than the order of the interpolation, continuous at the positions of the original X_i values. A second-order spline interpolation makes first derivative continuous and a third-order spline interpolation makes second derivatives continuous, etc. However, the interpolation equation applies only over the single interval X_j to X_{j+1}, and for the next interval another equation applies. In other words, second-order, or quadratic, splines are developed so that the first derivative of the two interpolation functions at X_i equal each other, i.e., $dy^{(1)}/dx = dy^{(2)}/dx$ at point 2, $dy^{(2)}dx = dy^{(3)}/dx$ at point 3, or in general $dy^{(i)}/dx = dy^{(i+1)}/dx$ at point j = i + 1. A third-order, or cubic spline, also makes second derivatives equal as one moves across the original X_is or $d^2y^{(i)}/dx^2 = d^2y^{(i+1)}/dx^2$ at point j = i + 1, etc.

B.4.2 Quadratic Splines

To illustrate how quadratic spline interpolation functions are developed let us consider a simple example in which four pairs (X_j, Y_j) are given as in the above graph. While within the interval X_1 to X_2 we want a second-degree polynomial $y^{(1)} = a_1x^2 + b_1x + c_1$ to apply, and when we move over the second X_2 and are not beyond X_3, we want $y^{(2)} = a_2x^2 + b_2x + c_2$ to apply, etc. The a's, b's and c's are the unknowns to solve so that for x between X_j and X_{j+1} the y within this interval $y^{(i)}$ $(i = j)$ can be determined. Thus, for our example there will be three unknown a's, three unknown b's, and three unknown c's, for a total of nine unknowns. In general, if there are n pairs (X_j, Y_j) then there will be $3(n - 1)$ unknowns, and that many independent equations must be used to solve for these unknowns. $2(n - 1)$ of these equations will be obtained by substituting Y_j and Y_{j+1} at both ends of each segment for $y^{(i)}$ as X_j and X_{j+1} replace x. In addition, $n - 2$ equations are obtained by making the first derivatives of the interpolating equations equal when evaluated at the $n - 2$ points X_2, X_3, …, X_{n-1}. Adding these up gives $2(n - 1) + (n - 2) = 3(n - 1) - 1$ equations; one short of the number needed. For our example these eight equations are

$$Y_1 = a_1X_1^2 + b_1X_1 + c_1 \tag{1}$$

$$Y_2 = a_1X_2^2 + b_1X_2 + c_1 \tag{2}$$

$$Y_2 = a_2X_2^2 + b_2X_2 + c_2 \tag{3}$$

$$Y_3 = a_2X_3^2 + b_2X_3 + c_3 \tag{4}$$

$$Y_3 = a_3X_3^2 + b_3X_3 + c_3 \tag{5}$$

$$Y_4 = a_3X_4^2 + b_3X_4 + c_3 \tag{6}$$

$$\left(\frac{dy^{(1)}}{dx}\right)_2 = \left(\frac{dy^{(2)}}{dx}\right)_2 \rightarrow 2a_1 + b_1 = 2a_2 + b_2 \tag{7}$$

$$\left(\frac{dy^{(2)}}{dx}\right)_3 = \left(\frac{dy^{(3)}}{dx}\right)_3 \rightarrow 2a_2 + b_2 = 2a_3 + b_3 \tag{8}$$

Note in these equations that the Y's and X's are known, i.e., these values are given as the pairs (X_j, Y_j). The last needed equation can be obtained in a number of ways. The end derivative $(dy^{(1)}/dx)_1$, or $(dy^{(n-1)}/dx)_n$ may be given. More commonly, the second derivative at the beginning or the end point is specified equal to zero. If $(d^2y^{(1)}/dx^2)_1 = 2a_1 = 0$, then $a_1 = 0$, which effectively reduces the number of unknowns by one, so that for our example the above eight equations will solve b_1, c_1, a_2, b_2, c_2, a_3, b_3, c_3. If the second derivative at the last point n is specified equal to zero then $(d^2y^{(n-1)}/dx^2)_n = 2a_n = 0$. When this is done the interpolating function for the first, or the last, interval is a straight line because either $a_1 = 0$ or $a_{n-1} = 0$, respectively. This fact represents a disadvantage of quadratic splines. Since the arithmetic involved in obtaining cubic splines is small, as seen in the next section, they are more widely used.

B.4.3 Cubic Splines

Cubic splines use a third-degree polynomial between each two consecutive points as the interpolating function, or

$$y^{(i)} = a_i x^3 + b_i x^3 + c_i x^3 + d_i$$

For the above example with four pairs of (X_j, Y_j), there will be three such interpolating equations. Now the number of unknown a's, b's, c's, and d's equals 4(n − 1), or for our example 4 × 3 = 12, unknowns. Thus 4(n − 1) equations must be obtained. 2(n − 1) of these equations come from substituting the known Y and X at the points j and j + 1 at the ends of each segment (i). (n − 2) equations come from equating the first derivatives of the two interpolating equations that apply at the data point and (n − 2) equations come from equating the second derivatives at these same points. The remaining two needed equations come from end boundary conditions. There are two commonly used boundary conditions. One sets second derivatives at the beginning and end to zero or $\left(d^2y/dx^2\right)_1 = y_1'' = 0$ and/or $\left(d^2y/dx^2\right)_n = y_n'' = 0$. These are called **natural cubic splines**. The other sets either y_1'' and or y_n'' to values calculated from giving values to the first derivatives.

For the four-point example above, the equations are

$$Y_1 = a_1X_1^3 + b_1X_1^2 + c_1X_1 + d_1 \tag{1}$$

$$Y_2 = a_1X_2^3 + b_1X_2^2 + c_1X_2 + d_1 \tag{2}$$

$$Y_2 = a_2X_2^3 + b_2X_2^2 + c_2X_2 + d_2 \tag{3}$$

$$Y_3 = a_2X_3^3 + b_2X_3^2 + c_2X_3 + d_2 \tag{4}$$

$$Y_3 = a_3X_3^3 + b_3X_3^2 + c_3X_3 + d_3 \tag{5}$$

$$Y_4 = a_3X_4^3 + b_3X_4^2 + c_3X_4 + d_3 \tag{6}$$

$$\left(\frac{dy^{(1)}}{dx}\right)_2 = \left(\frac{dy^{(2)}}{dx}\right)_2 \to 3a_1X_2^2 + 2b_1X_2 + c_1 = 3a_2X_2^2 + 2b_2X_2 + c_2 \tag{7}$$

$$\left(\frac{dy^{(2)}}{dx}\right)_3 = \left(\frac{dy^{(3)}}{dx}\right)_3 \to 3a_2X_3^2 + 2b_2X_3 + c_2 = 3a_3X_3^2 + 2b_3X_3 + c_3 \tag{8}$$

$$\left(\frac{d^2y^{(1)}}{dx^2}\right)_2 = \left(\frac{dy^{(2)}}{dx^2}\right)_2 \to 6a_1X_2 + 2b_1 = 6a_2X_2 + 2b_2 \tag{9}$$

$$\left(\frac{d^2y^{(2)}}{dx^2}\right)_3 = \left(\frac{d^2y^{(3)}}{dx^2}\right)_3 \to 6a_2X_3 + 2b_2 = 6a_3X_3 + 2b_3 \tag{10}$$

Boundary conditions

$$\left(\frac{d^2y}{dx^2}\right)_1 = y_1'' = 0 \tag{11}$$

$$\left(\frac{d^2y}{dx^2}\right)_n = y_4'' = 0 \tag{12}$$

or

$$\left(\frac{dy}{dx}\right)_1 = \text{specifed} = 3a_1X_1^2 + 2b_1X_1 + c_1 \tag{11a}$$

$$\left(\frac{dy}{dx}\right)_n = \text{specifed} = 3a_2X_4^2 + 2b_2X_4 + c_2 \tag{12b}$$

One obvious approach is to solve the above equations for the a's, b's, c's, and d's and then use the appropriate equation to compute a y for any given x. However, an alternative that results in less arithmetic uses the following interpolation equation:

$$y^{(i)} = a_iY_j + b_iY_{j+1} + c_iY_j'' + d_iY_{j+1}''$$

The coefficients a, b, c, and d are obviously different than above. The coefficients a and b are weighting functions to the dependent variable Y at point j and j + 1, and c and d are weighting functions of the second derivatives at these same points. In the finite element method a and b are commonly called the shape functions associated with the end points of a linear one-dimensional element. By using linear interpolation it can be shown easily that $a_i = (X_{j+1} - x)/(X_{j+1} - X_j)$ and $b_i = (x - X_j)/(X_{j+1} - X_j)$ and $a_i + b_i = 1$. Note a_i and b_i are linear functions of x. $a_i = 1$ at $x = X_j$ and $a_i = 0$ at $x = X_{j+1}$, whereas $b_i = 0$ at $x = X_j$ and $b_i = 1$ at $x = X_{j+1}$. As an expedient, in many of the following equations the subscripts and superscripts will be deleted. Just remember that the interpolating function provides values of y within the interval j to j + 1. Of importance is that c and d are functions of a and b; thus, the number of additional unknown produced for each new segment is 2 rather than 4. In other words, the number of equations needed is 2(n − 1) rather than 4(n − 1). Furthermore, since b = 1 − a, only one additional unknown is added for each new data point, so the number of equations needed is only n − 1. The relationships between c and d and a and b are

$$c = (1/6)(a^3 - a)(X_{j+1} - X_j)^2 \quad \text{and} \quad d = (1/6)(b^3 - b)(X_{j+1} - X_j)^2$$

The validity of these equations is proven subsequently. In other words the dependence on the independent variable x in the interpolating equation is entirely through the **linear** x-dependence of a and b. Yet because the derivatives are also weighted by c and d (which depend on a and b) a cubic interpolating polynomial results that applies between X_j and X_{j+1}. Let us prove these statements. First, since the definitions of c and d contain a^3 and b^3, and therefore c and d contain terms involving x^3 and x^2. Thus, the interpolating equation is a third-degree polynomial. Next, note that da/dx = $-1/(X_{j+1} - X_j)$ and db/dx = $1/(X_{j+1} - X_j)$ = −da/dx. Next, let us take the derivative of the interpolating function $y = aY_j + bY_{j+1} + cY_j'' + dY_{j+1}''$, or $y = aY_j + bY_{j+1} + cY_j'' + dY_{j+1}''$. After doing the algebra the following is obtained:

$$y' = \frac{dy}{dx} = \frac{Y_{j+1} - Y_j}{X_{j+1} - X_j} - \frac{3a^2 - 1}{6}(X_{j+1} - X_j)Y_j'' + \frac{3b^2 - 1}{6}(X_{j+1} - X_j)Y_{j+1}''$$

Taking the second derivative gives

$$y'' = aY_j'' + bY_j''$$

Since a = 1 at X_j and a = 0 at X_{j+1} and b = 0 at X_j and b = 1 at X_{j+1} we have just demonstrated that the relationship between c and a and d and b are valid, because $y'' = Y_j''$ at j and $y'' = Y_{j+1}''$ at j + 1.

The required equations can be obtained by evaluating the first derivatives given above at points 2, 3, … n − 2 and equating the two from adjacent segments to each other. We do not need to include the original equations, or the equations from equating second derivatives since these are already satisfied in the interpolating polynomial that was used to obtain the derivatives y′. Equating the first derivatives at the data points gives

$$\frac{Y_j - Y_{j-1}}{X_j - X_{j-1}} + \frac{X_j - X_{j-1}}{6}Y_{j+1}'' + \frac{X_j - X_{j-1}}{3}Y_j'' = \frac{Y_{j+1} - Y_j}{X_{j+1} - X_j} + \frac{X_{j+1} - X_j}{6}Y_j'' + \frac{X_{j+1} - X_j}{3}Y_{j+1}''$$

This equation has been obtained by evaluating a_i and b_i appropriately, i.e., at X_j, $a_{i-1} = 0$, $b_{i-1} = 1$, and $a_{i+1} = 1$ and $b_{i+1} = 0$ at X_{j+1}. This equation (e.g., these equations since j is incremented) needs to be rewritten so as to give a linear system of equations with the second derivatives Y_{j-1}'', Y_j'', and Y_{j+1}'' as the unknowns, and the known values on the right of the equal sign, or

$$(X_j - X_{j-1})Y_{j-1}'' + 2\left(X_{j+1} - X_{j-1}\right)Y_j'' + \left(X_{j+1} - X_j\right)Y_{j-1}'' = 6\left(\frac{Y_{j+1} - Y_j}{X_{j+1} - X_j} - \frac{Y_j - Y_{j-1}}{X_j - X_{j-1}}\right)$$

Written using matrix notation, the above equation consists of a coefficient matrix [A] multiplied by the unknown second-derivative vector {Y″} equal to a known vector {B}, or

$$[A]\{Y''\} = \{B\}$$

To make the system complete, boundary conditions must supply the first and last values. If the natural condition is used, then Y_1'' and Y_n'' are given zero values, which effectively starts the system at point 2 and ends the system at point n − 1. If first derivatives are given then these values provide the first and last equations in the system of equations. Note that three, and only three, consecutive values of the second derivatives are linked together in this system of equations, regardless of the boundary conditions used. Such a system, called a tridiagonal system of equations, is very common in many applications, and it can be solved readily by decomposition or elimination methods. Later algorithms for solving tridiagonal systems of equations will be covered in detail. Note now, however, that since only one element exists in front of the diagonal (and one after) that a single forward elimination pass through the rows of the matrix can make this an upper triangular matrix with only two nonzero elements. Thereafter, a back substitution pass can obtain the solution to the second derivatives.

In the above we have seen that this alternative to implementing a cubic spline consists of first solving the above tridiagonal system of linear algebraic equations for the values of the second derivatives at each of the points where the pairs of (X_j, Y_j) are given. Next, in each interval the values of a_i and $b_i = 1 - a_i$ are obtained by their definition that $a = (X_{j+1} - x)/(X_{j+1} - X_j)$, e.g., for the given x, a in the interval from X_j to X_{j+1} is evaluated, and finally the value of y corresponding to this x is obtained by the weighted interpolation equation

$$y = aY_j + bY_{j+1} + cY_j'' + dY_{j+1}''$$

where the appropriate values of the second derivatives are used for this interval, and c and d are defined as above in terms of a and b, or $c = (1/6)(a^3 - a)(X_{j+1} - X_j)^2$ and $d = (1/6)(b^3 - b)(X_{j+1} - X_j)^2$. Below listings of a FORTRAN, a C, and a PASCAL program are given that implement this process. These programs are designed to read-in N pairs of values for (X_j, Y_j) and then solve for y at M increments of x starting with X_1 and ending with X_n. The programs could easily be modified to allow for the user to provide a list of x values for which y's are to be computed, and this list could be provided from a file, or given individually from the keyboard. In studying the listing, in the computer language you are familiar with, you will note that after the data for (X_j, Y_j) are read-in, the main task of the program consists of solving the tridiagonal system of equations, and the final task is to provide the new table of M entries.

FORTRAN listing SPLINE3.FOR to implement cubic splines

```
      REAL X[ALLOCATABLE](:),Y[ALLOCATABLE](:),D[ALLOCATABLE](:),
     &D2Y[ALLOCATABLE](:)
      WRITE(*,*)' Give: INPUT,IOUT,N(No. pts),M(No. table),0=nat.
     &BC,1=give end derivatives'
      READ(*,*) INPUT,IOUT,N,M,ITY
      NM=N-1
      ALLOCATE(X(N),Y(N),D(NM),D2Y(N))
      IF(INPUT.EQ.0 .OR. INPUT.EQ.5) WRITE(*,*)' Give',N,' pairs
     &of x y'
      READ(INPUT,*) (X(I),Y(I),I=1,N)
      IF(ITY.EQ.0) THEN
      D2Y(1)=0.
      D(1)=0.
      ELSE
      IF(INPUT.EQ.0 .OR. INPUT.EQ.5) WRITE(*,*)' Give end
     &derivatives'
      READ(*,*) DY1,DYN
      D2Y(1)=-.5
      D(1)=3.*((Y(2)-Y(1))/(X(2)-X(1))-DY1)/(X(2)-X(1))
      ENDIF
      DO 10 J=2,NM
      JM=J-1
      JP=J+1
      FAC=(X(J)-X(JM))/(X(JP)-X(JM))
      FA1=FAC*D2Y(JM)+2.
      D2Y(J)=(FAC-1.)/FA1
10    D(J)=(6.*((Y(JP)-Y(J))/(X(JP)-X(J))-(Y(J)-Y(JM))/
     &(X(J)-X(JM)))/(X(JP)-X(JM))-FAC*D(JM))/FA1
      IF(ITY.EQ.0) THEN
      D2YN=0.
      DN=0.
      ELSE
      D2YN=.5
```

```
      DN=3.*(D2YN-(Y(N)-Y(NM)))/(X(N)-X(NM))**2
      ENDIF
      D2Y(N)=(DN-D2YN*D(NM))/(D2YN*D2Y(NM)+1.)
      DO 20 J=NM,1,-1
20    D2Y(J)=D2Y(J)*D2Y(J+1)+D(J)
      XX=X(1)
      DELX=(X(N)-XX)/FLOAT(M-1)
      WRITE(IOUT,100) XX,Y(1)
100   FORMAT(2F12.5)
      JP=2
      J=1
      DX=X(2)-X(1)
      DO 50 I=2,M
      XX=XX+DELX
30    IF(XX.LE.X(JP).OR.JP.EQ.N) GO TO 40
      J=JP
      JP=JP+1
      DX=X(JP)-X(J)
      GO TO 30
40    A=(X(JP)-XX)/DX
      B=1.-A
      YY=A*Y(J)+B*Y(JP)+((A*A-1.)*A*D2Y(J)+(B*B-1.)*B*D2Y(JP))*DX
     &*DX/6.
50    WRITE(IOUT,100) XX,YY
      END
```

C listing SPLINE3.C to implement cubic splines

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#define sqr(x) x*x;
void main(void){
float *x,*y,*d,*d2y,fac,fa1,dy1,dyn,dn,d2yn,dx,yy,xx,deltx,a,b;
int n,nm,m,ity,j,i,jm,jp; FILE *iout;
printf("Give: N(No. Pts),M(No. in table),0=nat. BC, 1=give\
  end der.\n");
scanf("%d %d %d",&n,&m,&ity); nm=n-1;
y=(float *)calloc(n,sizeof(float));
  x=(float *)calloc(n,sizeof(float));
d=(float *)calloc(nm,sizeof(float));
  d2y=(float *)calloc(n,sizeof(float));
printf("Give %3d pairs of X Y\n",n);
iout=fopen("spline.out","w");
  for(i=0;i<n;i++) scanf("%f %f",&x[i],&y[i]);
if (ity) {printf("Give end derivatives\n");
  scanf("%f %f",&dy1,&dyn);
 d2y[0]=-.5; d[0]=3.*((y[1]-y[0])/(x[1]-x[0])-dy1)/(x[1]-x[0]);}\
   else {
 d2y[0]=0.;d[0]=0.;}
for(j=1;j<nm;j++) {jm=j-1; jp=j+1;
```

```
 fac=(x[j]-x[jm])/(x[jp]-x[jm]); fa1=fac*d2y[jm]+2.;
   d2y[j]=(fac-1.)/fa1;
 d[j]=(6.*((y[jp]-y[j])/(x[jp]-x[j])-(y[j]-y[jm])/(x[j]-x[jm]))/\
  (x[jp]-x[jm])-fac*d[jm])/fa1;}
 if (ity) {d2yn=.5;dn=3.*(d2yn-(y[nm]-y[n-2]))/sqr(x[nm]-x[n-2]);}\
    else {d2yn=0.;dn=0.; }
d2y[nm]=(dn-d2yn*d[n-2])/(d2yn*d2y[n-2]+1.);
for(j=n-2;j>=0;j--) d2y[j]=d2y[j]*d2y[j+1]+d[j]; xx=x[0];
 deltx=(x[nm]-xx)/(float)(m-1);
fprintf(iout,"%12.5f %11.5f\n",xx,y[0]); jp=1; j=0; dx=x[1]-x[0];
for(i=1;i<m;i++){xx+=deltx;
  while ((xx>=x[jp])&&(jp<nm)) {j=jp;jp++;dx=x[jp]-x[j];}
a=(x[jp]-xx)/dx; b=1.-a;
yy=a*y[j]+b*y[jp]+((a*a-1.)*a*d2y[j]+(b*b-1.)*b*d2y[jp])\
  *dx*dx/6.;
fprintf(iout,"%12.5f %11.5f\n",xx,yy);} fclose(iout);
}
```

PASCAL listing SPLINE3.PAS to implement cubic splines

```
Program SPLINE3;
Type ary=array[1..50] of real;
Var X,Y,D,D2Y: ary;
 FAC,FA1,DY1,DYN,DN,D2YN,DX,YY,XX,DELX,A,B: Real;
 N,NM,M,ITY,J,I,JM,JP: Integer; IOUT: Text;
BEGIN
 Writeln('Give: N(No. Pts),M(No. in table),0=nat. BC, 1=give end
   der.');
 Readln(N,M,ITY); NM:=N-1;
 Writeln(' Give',N:3,' pairs of X Y');
 For I:=1 to N do Read(X[I],Y[I]);
 if ITY=0 then begin D2Y[1]:=0.0; D[1]:=0.0 end else begin
  Writeln(' Give end derivatives'); Readln(DY1,DYN);
  D2Y[1]:=-0.5;
    D[1]:=3.*((Y[2]-Y[1])/(X[2]-X[1])-DY1)/(X[2]-X[1]) end;
 For J:=2 to NM do begin JM:=J-1; JP:=J+1;
  FAC:=(X[J]-X[JM])/(X[JP]-X[JM]); FA1:=FAC*D2Y[JM]+2.0;
  D2Y[J]:=(FAC-1.0)/FA1;
  D[J]:=(6.0*((Y[JP]-Y[J])/(X[JP]-X[J])-(Y[J]-Y[JM])/
    (X[J]-X[JM]))/(X[JP]-X[JM])-FAC*D[JM])/FA1
 end;
 if ITY=0 then begin D2YN:=0.;DN:=0. end else begin
  D2YN:=0.5; DN:=3.*(D2YN-(Y[N]-Y[NM]))/sqr(X[N]-X[NM]) end;
 D2Y[N]:=(DN-D2YN*D[NM])/(D2YN*D2Y[NM]+1.0);
 For J:=NM downto 1 do D2Y[J]:=D2Y[J]*D2Y[J+1]+D[J]; XX:=X[1];
 DELX:=(X[N]-XX)/(M-1);
 Assign(IOUT, 'SPLINE.OUT'); Rewrite(IOUT);
   Writeln(IOUT,XX:12:5,Y[1]:12:5);
 JP:=2; J:=1; DX:=X[2]-X[1];
 For I:=2 to M do Begin XX:=XX+DELX;
```

```
 While (XX>X[JP]) and (JP<N) do begin J:=JP;inc(JP);
   DX:=X[JP]-X[J] end;
A:=(X[JP]-XX)/DX; B:=1.0-A;
YY:=A*Y[J]+B*Y[JP]+((A*A-1.0)*A*D2Y[J]+(B*B-1.0)*B*D2Y[JP])*sqr
  (DX)/6.0;
Writeln(IOUT,XX:12:5,YY:12:5) End; Close(IOUT);
END.
```

EXAMPLE PROBLEM B.5

We want to generate a smooth function for plotting x versus y using cubic splines through the following six pairs of (X_j, Y_j) values: (0, 0), (5, 2), (9, 9), (14, 15), (18, 17), and (22, 19).

Solution

To use the above programs the following input would be given: 6 21 0 (The FORTRAN program would need two values such as 5 and 3 to precede these three input values for the input and output logical units.); followed by the given (X,Y) values, or

0 0 5 2 9 9 14 15 18 17 22 19.

The results from this solution give the following table of (x, y) values and the graph that plots them:

x	y
0.00000	0.00000
1.10000	−0.00043
2.20000	0.13354
3.30000	0.53633
4.40000	1.34234
5.50000	2.67996
6.60000	4.50437
7.70000	6.57575
8.80000	8.64388
9.90000	10.48313
11.00000	12.03949
12.10000	13.33206
13.20000	14.38020
14.30000	15.20343
15.40000	15.83885
16.50000	16.35559
17.60000	16.82616
18.70000	17.31937
19.80000	17.85540
20.90000	18.42045
22.00000	19.00000

The above cubic spline program can readily be made into a SUBROUTINE (function). Then a main program can read-in the original pairs of values, and call on this subroutine to pass the array containing the second derivatives back to the main program. The following is a FORTRAN listing of such a subroutine (as an exercise you many want to do the same if you are a C or a PASCAL programmer):

Listing of SUBROUTINE SPLINESU.FOR that returns to its calling program the needed 2nd derivatives.

```
      SUBROUTINE SPLINESU(N,X,Y,D2Y,D,ITY)
      REAL X(N),Y(N),D(N),D2Y(N)
      NM=N-1
      IF(ITY.EQ.0) THEN
      D2Y(1)=0.
      D(1)=0.
      ELSE
      WRITE(*,*)' Give end derivatives'
      READ(*,*) DY1,DYN
      D2Y(1)=-.5
      D(1)=3.*((Y(2)-Y(1))/(X(2)-X(1))-DY1)/(X(2)-X(1))
      ENDIF
      DO 10 J=2,NM
      JM=J-1
      JP=J+1
      FAC=(X(J)-X(JM))/(X(JP)-X(JM))
      FA1=FAC*D2Y(JM)+2.
      D2Y(J)=(FAC-1.)/FA1
10    D(J)=(6.*((Y(JP)-Y(J))/(X(JP)-X(J))-(Y(J)-Y(JM))/
     &(X(J)-X(JM)))/(X(JP)-X(JM))-FAC*D(JM))/FA1
      IF(ITY.EQ.0) THEN
      D2YN=0.
      DN=0.
      ELSE
      D2YN=.5
      DN=3.*(D2YN-(Y(N)-Y(NM)))/(X(N)-X(NM))**2
      ENDIF
      D2Y(N)=(DN-D2YN*D(NM))/(D2YN*D2Y(NM)+1.)
      DO 20 J=NM,1,-1
20    D2Y(J)=D2Y(J)*D2Y(J+1)+D(J)
      RETURN
      END
```

The arguments to this subroutine call are as follows:

N is an integer that gives the number of pairs of (x, y) values that the cubic spline is to be used to fit.

X is a real array dimensioned to N that contains the values of the independent variable, i.e., the x of the above pairs.

Y is a real array dimensioned to N that contains the values of the dependent variable, i.e., the y values of the above pairs that are to be fit using the cubic spline.

D2Y is a real array dimensioned to N that will return the values of the second derivatives at the points. The calling program must use these values appropriately to do the interpolation.

D is a real array dimensioned to N that is used as workspace by the subroutine.

ITY is 0, or 1 depending upon the type of boundary conditions to be used at the first and last points. If ITY = 0, then the natural BC is used, otherwise the subroutine will read-in the values of the end first derivatives that the user must supply.

EXAMPLE PROBLEM B.6

Examine how well a cubic spline duplicates a circle. In making this examination generate the x and y coordinates of one-half of a circle on a 12° increment starting with 0° and ending with 180° (0°is at the bottom of the circle). Pass the cubic spline through these generated coordinates, and then on a 5° increment compare the interpolated x with the actual x. Also numerically integrate the cubic spline to determine the areas and wetted perimeters (i.e., the circumferences extending on both sides starting at the bottom of the circle) at each of the latter 5° increments and compare these with the actual areas and wetted perimeters. Use a radius of R = 10 in computing the values requested.

Solution

In solving this problem we note that the actual x and y coordinates are obtained from $x = R\sin(\beta)$ and $y = R(1 - \cos\beta)$. The actual area and wetted perimeter are obtained from $A = R^2(\beta - \cos\beta \sin\beta)$ and $P = 2R\beta = D\beta$. Simpson's one-third rule, as implemented in SIMPR, will be used to carry out numerical integrations by noting that the area A can be obtained from $A = 2\int x\,dy$ and the perimeter $P = 2\int\{dx^2 + dy^2\}^{1/2}dy$. Using the properties of a cubic spline as developed previously in this section,

$$dx = \left\{\frac{\Delta x}{\Delta y} + \frac{\Delta y}{6}\left[-(3a^2 - 1)X_j'' + (3b^2 - 1)X_j'' + 1\right]\right\}dy = Zdy$$

where a and b are the linear cubic spline weighting functions as given earlier, but since y is the independent variable they become $a = (Y_{j+1} - y)/(Y_{j+1} - Y_j)$ and $b = 1 - a$, and $\Delta x = X_{j+1} - X_j$ and $\Delta y = Y_{j+1} - Y_j$, and X_j'' and X_{j+1}'' are the second derivates d^2x/dy^2 at the two points j and j + 1 as generated by solving the tridiagonal system of equations as described in the cubic spline method with x considered the dependent variable and y the independent variable. Using the above to substitute for dx in the integral for the perimeter P gives

$$P = 2\int\left\{Z^2 + 1\right\}^{1/2}dy$$

where $Z = \{\Delta x/\Delta y + \Delta y/6[-(3a^2 - 1)X_j'' + (3b^2 - 1)X_{j+1}'']\}$.

Program CIRCSPLI.FOR obtains the numbers to compare the requested results from using a cubic spline for interpolation with the actual values. The table below the program listing is the output from running the program. Notice Program CIRCSPLI call on the subroutine SPLINESU to obtain the second derivatives needed to interpolate using a cubic spline, as well as SIMPR to carry out the numerical integrations. You should note the logical in function subprograms EQUAT and EQUAT1 (as well as the main program), that supply the arguments for the numerical integration of the area and wetted perimeters, respectively, in order to use the two original points that bracket the interval in which the interpolation is being done. In a sense using a cubic polynomial to approximate a circle raises some interesting questions since the derivative dx/dy is infinite at the bottom and top where $\beta = 0$ and $\beta = \pi$, and dy/dx is infinite at the sides where $\beta = \pi/2$ and $\beta = 3\pi/2$. The second derivatives are undefined at these points. Since the cubic spline equates first and second derivatives (in this problem dx/dy and d^2x/dy^2) at the given points it is clear that the cubic splines do not duplicate the circle. In the program the natural boundary condition at the ends, which sets $d^2x/dy^2 = 0$ at $\beta = 0$ and $\beta = \pi$ is, used.

Program CIRCSPLI.FOR

```
      EXTERNAL EQUAT,EQUAT1
      COMMON M,MP,N,NP,X(16),Y(16),D2X(16)
      REAL DD(16)
      M=1
      MP=2
      N=1
      NP=2
      CONV=.0174532925
```

```
      WRITE(*,*)' Give radius of circle & increment in degrees'
      READ(*,*) R,IDEG
      D=2.*R
      R2=R*R
      DO 10 I=1,16
      DEG=12*(I-1)
      RAD=CONV*DEG
      X(I)=R*SIN(RAD)
10    Y(I)=R*(1.-COS(RAD))
      CALL SPLINESU(16,Y,X,D2X,DD,0)
      Y1=0.
      AA=0.
      PP=0.
      WRITE(3,100)
100   FORMAT(1X,98('-'),/,25X,'Position,x',21X,'Area',21X,
     &'Perimeter',/ ' Deg radians y x(act) x(int) dif. A(act)
     &A(int) dif. P(act) P(int) dif.',/,1X,98('-'))
      WRITE(3,110) 0,(0.,I=1,11)
      DO 20 I=IDEG,180,IDEG
      RAD=CONV*FLOAT(I)
      Y2=R*(1.-COS(RAD))
      CALL SIMPR(EQUAT,Y1,Y2,AR,1.E-4,11)
      AA=AA+2.*AR
      CALL SIMPR(EQUAT1,Y1,Y2,PR,1.E-4,11)
      PP=PP+2.*PR
      X2=R*SIN(RAD)
      DY=Y(MP)-Y(M)
      DY6=DY*DY/6.
      A=(Y(MP)-Y2)/DY
      B=1.-A
      XX=A*X(M)+B*X(MP)+((A*A-1.)*A*D2X(M)+(B*B-1.)*B*D2X(MP))
     &*DY6
      AREA=R2*(RAD-COS(RAD)*SIN(RAD))
      PER=D*RAD
      WRITE(3,110) I,RAD,Y2,X2,XX,X2-XX,AREA,AA,
     &AREA-AA,PER,PP,PER-PP
110   FORMAT(I4,F7.4,4F8.3,2(2X,2F9.3,F8.3))
20    Y1=Y2
      END
      FUNCTION EQUAT(YY)
      COMMON M,MP,N,NP,X(16),Y(16),D2X(16)
5     IF(YY.LE.Y(MP) .OR. MP.EQ.16) GO TO 7
      M=MP
      MP=MP+1
      GO TO 5
7     IF(YY.GE.Y(M) .OR. M.EQ.1) GO TO 10
      MP=M
      M=M-1
      GO TO 7
10    DY=Y(MP)-Y(M)
      DY6=DY*DY/6.
      A=(Y(MP)-YY)/DY
      B=1.-A
      EQUAT=A*X(M)+B*X(MP)+((A*A-1.)*A*D2X(M)+(B*B-1.)*B*D2X(MP))
     &*DY6
      RETURN
```

```
      END
      FUNCTION EQUAT1(YY)
      COMMON M,MP,N,NP,X(16),Y(16),D2X(16)
5     IF(YY.LE.Y(NP) .OR. NP.EQ.16) GO TO 7
      N=NP
      NP=NP+1
      GO TO 5
7     IF(YY.GE.Y(N) .OR. N.EQ.1) GO TO 10
      NP=N
      N=N-1
      GO TO 7
10    DY=Y(NP)-Y(N)
      DX=X(NP)-X(N)
      A=(Y(NP)-YY)/DY
      B=1.-A
      Z=DX/DY-((3.*A*A-1.)*D2X(N)-(3.*B*B-1.)*D2X(NP))*DY/6.
      EQUAT1=SQRT(Z**2+1.)
      RETURN
      END
```

Output:

			Position, x			Area			Perimeter		
Deg	**radians**	**y**	**x(act)**	**x(int)**	**dif.**	**A(act)**	**A(int)**	**dif.**	**P(act)**	**P(int)**	**dif.**
0	.0000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
5	.0873	.038	.872	.393	.478	.044	.015	.029	1.745	.790	.955
10	.1745	.152	1.736	1.512	.225	.352	.234	.118	3.491	3.039	.452
15	.2618	.341	2.588	2.873	−.285	1.180	1.086	.094	5.236	5.789	−.553
20	.3491	.603	3.420	3.745	−.324	2.767	2.862	−.095	6.981	7.615	−.634
25	.4363	.937	4.226	4.153	.073	5.331	5.508	−.177	8.727	8.674	.053
30	.5236	1.340	5.000	4.804	.196	9.059	9.100	−.041	10.472	10.205	.266
35	.6109	1.808	5.736	5.698	.037	14.102	14.023	.078	12.217	12.226	−.008
40	.6981	2.340	6.428	6.508	−.080	20.573	20.534	.039	13.963	14.165	−.202
45	.7854	2.929	7.071	7.118	−.047	28.540	28.586	−.046	15.708	15.863	−.155
50	.8727	3.572	7.660	7.641	.020	38.026	38.083	−.057	17.453	17.521	−.068
55	.9599	4.264	8.192	8.167	.024	49.008	49.029	−.021	19.199	19.261	−.062
60	1.0472	5.000	8.660	8.660	.000	61.418	61.421	−.003	20.944	21.033	−.089
65	1.1345	5.774	9.063	9.073	−.009	75.144	75.157	−.013	22.689	22.787	−.098
70	1.2217	6.580	9.397	9.400	−.003	90.034	90.058	−.024	24.435	24.528	−.093
75	1.3090	7.412	9.659	9.657	.002	105.900	105.924	−.024	26.180	26.270	−.090
80	1.3963	8.264	9.848	9.846	.002	122.525	122.545	−.020	27.925	28.015	−.090
85	1.4835	9.128	9.962	9.962	.000	139.671	139.689	−.018	29.671	29.761	−.091
90	1.5708	10.000	10.000	10.002	−.002	157.080	157.100	−.020	31.416	31.507	−.091
95	1.6581	10.872	9.962	9.962	.000	174.489	174.511	−.022	33.161	33.252	−.091
100	1.7453	11.736	9.848	9.846	.002	191.634	191.655	−.021	34.907	34.998	−.091
105	1.8326	12.588	9.659	9.657	.002	208.260	208.277	−.017	36.652	36.743	−.092
110	1.9199	13.420	9.397	9.400	−.003	224.126	224.142	−.016	38.397	38.485	−.088
115	2.0071	14.226	9.063	9.073	−.009	239.015	239.043	−.028	40.143	40.226	−.083
120	2.0944	15.000	8.660	8.660	.000	252.741	252.779	−.038	41.888	41.980	−.093
125	2.1817	15.736	8.192	8.167	.024	265.151	265.171	−.020	43.633	43.752	−.119
130	2.2689	16.428	7.660	7.641	.020	276.133	276.117	.016	45.379	45.492	−.113
135	2.3562	17.071	7.071	7.118	−.047	285.619	285.614	.005	47.124	47.150	−.026
140	2.4435	17.660	6.428	6.508	−.080	293.586	293.667	−.080	48.869	48.848	.021

(continued)

			Position, x			Area			Perimeter		
Deg	radians	y	x(act)	x(int)	dif.	A(act)	A(int)	dif.	P(act)	P(int)	dif.
145	2.5307	18.192	5.736	5.698	.037	300.057	300.177	−.119	50.615	50.787	−.173
150	2.6180	18.660	5.000	4.804	.196	305.101	305.101	.000	52.360	52.808	−.448
155	2.7053	19.063	4.226	4.153	.073	308.828	308.693	.136	54.105	54.339	−.234
160	2.7925	19.397	3.420	3.745	−.324	311.392	311.338	.054	55.851	55.398	.452
165	2.8798	19.659	2.588	2.873	−.285	312.979	313.114	−.135	57.596	57.224	.372
170	2.9671	19.848	1.736	1.512	.225	313.807	313.966	−.159	59.341	59.974	−.633
175	3.0543	19.962	.872	.393	.478	314.115	314.185	−.070	61.087	62.223	−1.136
180	3.1416	20.000	.000	.000	.000	314.159	314.200	−.041	62.832	63.013	−.181

B.5 LINEAR ALGEBRA

Many problems require that a system of linear algebraic equations is solved. There are many problems in this course that require the solution of a system of nonlinear equations. The Newton method is a very effective method for solving nonlinear equations, and it iteratively solves a system of linear equations when used to solve a nonlinear system of equations. Some of the program listings given in this text refer to the linear algebra solver SOLVEQ. This section briefly describes how to use this subroutine. SOLVEQ implements methods that reduce the magnitude of truncation errors to a minimum. It will (a) solve a linear system of equations given the coefficient matrix and the known vector; (b) provide the inverse matrix of a square matrix; (c) evaluate the determinant and give its value to one of its arguments; (d) evaluate the determinant and produce the inverse matrix of a square matrix; and (e) evaluate the determinant, produce the inverse matrix, and solve the system of equations.

The use of SOLVEQ is documented first in this appendix. Thereafter, another subroutine SOLVEQL is described that does not allow all the options of SOLVEQ.

Use of FORTRAN subroutine SOLVEQ or function SOLVEQ:

SOLVEQ is available as a FORTRAN subroutine, or a C (or CPP) function. Both obtain the solution using the same method, but the details of how they are used vary slightly because of the differences in the programming languages. Only the use of the FORTRAN subroutine is described herein.

FORTRAN subroutine SOLVEQ:

Subroutine SOLVEQ must be called by a program that has completely defined the problem it is to solve by supplying the coefficient matrix in a two-dimensional array, and the known vector in a one-dimensional array, if it is called for. The matrix and vector beginning with subscript 1 and ending with subscript N. In other words, the default dimensioning of the two-dimensional matrix array and known vector should be used, i.e., in the main program the matrix and vector would be dimensioned with a statement such as REAL A(100,100),B(100). A call to subroutine SOLVEQ should consist of the following (the names of the arguments can be different, but the types must be as described below, and the dimensions of arrays must be as indicated):

```
CALL SOLVEQ(N,NPROB,NDIM,A,B,ITYPE,DET,INDX)
```

The meaning of each of these arguments in the call is as follows:

`N` is an integer that equals the number of equations that are to be solved, or the size of the matrix, if only the inverse is called for. The program that calls SOLVEQ must supply values for a square coefficient matrix with N rows and N columns.

`NPROB` is an integer that equals the number of problems that are to be solved by providing solution vectors, i.e., provides `NPROB` separate solutions from `NPROB` known vectors. A modified version of `SOLVEQ` may have this argument omitted because almost all applications have only one known vector. Make sure which version of SOLVEQ you are using. If only two integer arguments exit before the coefficient matrix A, then the vector (array) B must be one-dimensional, and it contains the known vector for the linear system of equations that is being solved.

NDIM is an integer equal to the dimensions of matrix array A(NDIM,NDIM) and vector B(NDIM). NDIM can be larger or equal to N. Its value is needed so that SOLVEQ can locate the proper positions of the elements within the two-dimensional coefficient array, A.

A is a two-dimensional real array containing the coefficient matrix and must be square with N rows and N columns. In other words, it is a two-dimensional array in the calling program. Upon entry into subroutine SOLVEQ the correct values for the problem that is to be solved must be within all the elements of this array. Upon returning from SOLVEQ this two-dimensional array will contain the inverse matrix, if this is called for. In any event the values of the coefficient matrix will be altered upon returning from SOLVEQ.

B is a real array containing the known vector {b} in the linear system of equations [A]{x} = {b}, and the correct values for this known vector must be in the elements of B when SOLVEQ is called. Upon returning from the call to SOLVEQ this array will contain the solution vector {x} for the linear algebra problem. Generally B will be dimensioned as a one-dimensional array. However, if NPROB is greater than one, i.e., more than one linear algebra problem is to be solved using the same coefficient matrix [A], then B can be a two-dimensional array with its second dimension equal to NPROB.

ITYPE is an integer variable that communicates to SOLVEQ what is to be done according to the value given from 1 through 6 as follows:

=1 Solves linear system of equations
=2 Produces inverse matrix (in A)
=3 Evaluates determinant, and gives value in DET
=4 Solves equations and produces inverse matrix
=5 Evaluates determinant and produces inverse matrix
=6 Evaluates determinant, produces inverse matrix, and solves equations

DET is a real variable that returns the value of determinant if it is ask for

INDX is an integer *2 one-dimensional array with the size NDIM used for work space. Upon entry to SOLVEQ it can be empty, or another integer array used subsequently in the calling program. The values in this array will be destroyed upon returning from SOLVEQ, so if it is an INTEGER*2 array used for some other purpose, this purpose must be after all calls to SOLVEQ have been completed.

Example programs in the body of the text can be used as examples of how to properly implement a call to SOLVEQ. It will be instructive, however, to include a simple example here, that you can use your calculator to check. Given the 4 × 4 matrix below that has the known vector attached as the fifth column, write a program that calls on SOLVEQ and solves the problem using the six options available with ITYPE.

Four coefficients for simultaneous equations; file: MAINSOL.DA1

```
 3 −2   1  0  4
−2  5  −1  1  7
 1 −1 −11 −2  2
 0  1  −2 12  3
```

Solution

The FORTRAN program might consist of

MAINSOL.FOR

```
      REAL A(5,5),B(5)
      INTEGER*2 INDX(5)
      WRITE(*,*)' Give: INPUT,IOUT,N,ITYPE'
```

```
      READ(*,*) INPUT,IOUT,N,ITYPE
      DO 10 I=1,N
10    READ(INPUT,*) (A(I,J),J=1,N),B(I)
      CALL SOLVEQ(N,1,5,A,B,ITYPE,DET,INDX)
      IF(ITYPE.EQ.1 .OR. ITYPE.EQ.3) GO TO 30
      DO 20 I=1,N
20    WRITE(IOUT,'(6E12.5)')(A(I,J),J=1,N),B(I)
30    IF(ITYPE.GT.2.AND.ITYPE.NE.4) WRITE(IOUT,'(' ' Det='
     &',E12.5)') DET
      IF(ITYPE.EQ.1.OR.ITYPE.EQ.6)WRITE(IOUT,'(' ' Solution:'
     &',/6E12.5)')(B(I),I=1,N)
      END
The solution using ITYPE=6 is:
       .44954E+00   .18725E+00  .25896E-01 -.11288E-01   .31268E+01
       .18725E+00   .27888E+00 -.39841E-02 -.23904E-01   .26215E+01
       .25896E-01  -.39841E-02 -.85657E-01 -.13944E-01  -.13745E+00
      -.11288E-01  -.23904E-01 -.13944E-01  .83001E-01   .86321E-02
      Det= -.15060E+04
      Solution:
      .31268E+01 .26215E+01 -.13745E+00 .86321E-02
```

Use of subroutine SOLVEQL:

The subroutine SOLVEQL is a limited version of SOLVEQ that does not allow the options of obtaining the inverse matrix, or evaluating the determinant. It only solves a linear system of equations, and only allows one known vector to be present.

A call to SOLVEQL should consist of the following (the names of the arguments can be different, but the types must be as described below, and the dimensions of the arrays must be as indicated):

```
CALL SOLVEQL(N,NDIM,A,B,INDX,VEC)
```

The meaning of each of these arguments in the call is as follows:

N is an integer that equals the number of equations that are to be solved. The program that calls SOLVEQL must supply values for a square coefficient matrix A with N rows and N columns.

NDIM is an integer equal to the dimensions of the matrix array A(NDIM,NDIM) and the vectors B(NDIM) and VEC(NDIM). NDIM can be larger than N, but not smaller. Its value is needed so that SOLVEQL can locate the proper positions of the elements within the two-dimensional coefficient matrix A.

A is a two-dimensional real array that contains the coefficient matrix when SOLVEQL is called.

B is a real array that contains the known vector {b} in the linear system of equations [A]{x} = {b}, and the correct values for this known vector must be in the elements of B when SOLVEQL is called. Upon returning from the call to SOLVEQL, this array will contain the solution vector {x} for the linear algebra problem.

INDX is an INTEGER*2 one-dimensional array with the size NDIM used for work space. Upon entry to SOLVEQL it can be empty, or another integer array used subsequently in the calling program. The values in this array will be destroyed upon returning from SOLVEQL, so if it is an INTEGER*2 array used for some other purpose, this purpose must be after all calls to SOLVEQL have been completed, or if used before, not be needed after calling SOLVEQL.

VEC is a REAL one-dimensional array with the size NDIM used for work space. The same restrictions apply for VEC that apply for INDX.

The program given above MAINSOL would look like the following if it called SOLVEQL instead of SOLVEQ, but now the program only has the capability to solve the given linear system of equations:

MAINSOL1.FOR

```
      REAL A(5,5),B(5),VEC(5)
      INTEGER*2 INDX(5)
      WRITE(*,*)' Give: INPUT,IOUT,N'
      READ(*,*) INPUT,IOUT,N
      DO 10 I=1,N
10    READ(INPUT,*) (A(I,J),J=1,N),B(I)
      CALL SOLVEQL(N,5,A,B,INDX,VEC)
      WRITE(IOUT,'(' ' Solution:' ',/6E12.5)')(B(I),I=1,N)
      END
```

Use of subroutine SOLVEQLD:

The subroutine SOLVEQLD is essential, the same as subroutine SOLVEQL, but has the added capability to return the value of the determinant. A call to SOLVEQLD is the same as a call to SOLVEQL with the exception that one additional argument is added to the call that returns the determinant of the coefficient matrix A, or

```
CALL SOLVEQLD(N,NDIM,A,B,INDX,VEC,DET)
```

where `DET` returns the value of the determinant.

B.5.1 Use of Subroutine LAGU That Implements Laguerre's Method

The subroutine LAGU is designed to extract a root from an n-degree polynomial whose coefficients are defined in the calling program. By calling LAGU n times, LAGU implements Laguerre's method in finding all roots of a polynomial, including imaginary roots. It requires the use of complex arithmetic, and uses an iterative approach to get ever closer to the root.

Laguerre's method first expresses the polynomial as

$P_n = (z - z_1)(z - z_2) \cdots (z - z_n)$, where $z_1, z_2, \ldots, z_n$ are its roots,

and then takes the log of the absolute value of this expression, i.e.,
$\mathrm{Ln}|P_n| = \mathrm{Ln}|z - z_1| + \mathrm{Ln}|z - z_2| + \cdots + \mathrm{Ln}|z - z_n|$ and then noting that the first and second derivatives are

$$\frac{d\mathrm{Ln}|P_n|}{dz} = \frac{1}{(z - z_1)} + \frac{1}{(z - z_2)} + \cdots + \frac{1}{(z - z_n)} = \frac{P_n'}{P_n} = A$$

and

$$\frac{d^2\mathrm{Ln}|P_n|}{dz} = \frac{1}{(z - z_1)^2} + \frac{1}{(z - z_2)^2} + \cdots + \frac{1}{(z - z_n)^2} = \left(\frac{P_n'}{P_n}\right)^2 - \frac{P_n''}{P_n} = B$$

Next, assume that root z_1, which is being sought, is a distance a from the guess z_1, and all other roots are at the distance b (or $a = z - z_1$ and $b = z - z_i$, $i = 2, 3 \ldots n$). Then from the above first and second derivatives,

$1/a + (n - 1)/b = A$ and $1/a^2 + (n - 1)/b^2 = B$ will allow a to be solved from

$a = n/(A \pm [(n-1)(nB - A^2)]^{1/2}$ with the sign taken so the largest magnitude occurs for the denominator. The process is repeated until the change in z_1 becomes small enough. Since the quantity within [] may be negative, complex arithmetic is used.

The subroutine LAGU in the file LAGUSUB.FOR implements this method, allowing up to 50 iterations (to change this allowable number of iterations change the 50 in DO 20 ITER=1,50). The call to LAGU(C,J,Z1,EPS) requires the following arguments:

C is a complex array that contains the ND+1 coefficients of the polynomial, $P_n = c_n z^n + c_{n-1}^{n-1} + \cdots + c_1 z + c_o = 0$, with C(1) = c_o, C(2) = c_1, …, C(ND + 1) = c_n. Since C is complex and the coefficients are generally real, the FORTRAN library function CMPLX will be used, i.e., C(1) = CMPLX(c_o,0.)

J = ND is the degree n of the polynomial.

Z1 is a guess for the root being sought, which for convenience may be selected as Z1 = CMPLX(0.,0.).

EPS is the desired fractional accuracy, i.e., Δroot<EPS*CABS(Z1).

If one wishes to extract all roots of a polynomial of degree n, then LAGU should be called n times with the degree of the polynomial decreased by 1 for each new call, with the coefficients in C modified to represent the new polynomial after the root that was just found is extracted.

LAGUSUB.FOR

```
      SUBROUTINE LAGU(C,ND,Z1,EPS)
      COMPLEX C(ND+1),Z1,DX,ZO,Z2,Z3,Z4,Z5,DZ,SS,Z6,Z7,Z8,ZERO,XX,FF
      ZERO=CMPLX(0.,0.)
      DO 20 ITER=1,50
      Z2=C(ND+1)
      Z3=ZERO
      Z4=ZERO
      DO 10 J=ND,1,-1
      Z4=Z1*Z4+Z3
      Z3=Z1*Z3+Z2
10    Z2=Z1*Z2+C(J)
      IF(CABS(Z2).LE.1.E-8) THEN
      DX=ZERO
      ELSEIF(CABS(Z3).LE.1.E-8.AND.CABS(Z4).LE.1.E-8)THEN
      DX=CMPLX(CABS(Z2/C(ND+1))**(1./FLOAT(ND)),0.)
      ELSE
      Z5=Z3/Z2
      Z8=Z5*Z5
      DZ=Z8-2.*Z4/Z2
      XX=(ND-1)*(ND*DZ-Z8)
      YY=ABS(REAL(XX))
      ZZ=ABS(AIMAG(XX))
      IF(YY.LT.1.E-12 .AND. ZZ .LT.1.E-12) THEN
      SS=ZERO
      ELSE IF (YY.GE.ZZ) THEN
      FF=(1./YY)*XX
      SS=SQRT(YY)*CSQRT(FF)
      ELSE
      FF=(1./ZZ)*XX
      SS=SQRT(ZZ)*CSQRT(FF)
```

```
      ENDIF
      Z6=Z5+SS
      Z7=Z5-SS
      IF(CABS(Z6).LT.CABS(Z7)) Z6=Z7
      DX=FLOAT(ND)/Z6
      ENDIF
      ZO=Z1-DX
      IF(Z1.EQ.ZO) RETURN
      Z1=ZO
      IF(CABS(DX).LE.EPS*CABS(Z1)) RETURN
20    CONTINUE
      WRITE(6,*)' FAILED TO CONVERGE'
      RETURN
      END
```

Appendix C ODESOL: Subroutine to Solve ODEs

C.1 BACKGROUND TO ALGORITHM

This description provides information to use the subroutine ODESOL, that is on the CD in the back cover of the book in the folder PROGRAM-TEXT, to obtain solutions to ordinary differential equations. The methods used in this subroutine obtain high-accuracy solutions to ordinary differential equations with minimal computational effort. The methods utilize an extrapolation, with a modified midpoint that is called the Bulirsh–Stoer method. The method is not well adapted for nonsmooth functions, such as often occurs with tabular data.

C.2 USING FORTRAN

The subroutine ODESOL is designed to solve a system of first-order (e.g., equations with first derivatives) ordinary differential equations or to solve a higher-order ordinary differential equation. If a higher-order equation, of order N, is to be solved, it must be reduced first to a system (or coupled set) of N first-order differential equations. Thus, for example, if the second-order equation,

$$\frac{d^2y}{dx^2} + f(x)\frac{dy}{dx} = g(x)$$

is to be solved, it is first rewritten as the following two first-order coupled equations:

$$\frac{dz}{dx} = g(x) - f(x)\cdot z(x) \quad \text{and} \quad \frac{dy}{dx} = z(x)$$

Subroutine ODESOL has been designed to provide users with considerable flexibility. A common method of use will involve calling ODESOL repeatedly with each new call over a new increment of the independent variable x, until the solution over the desired range is achieved. Another method of use will be to call ODESOL, but once with the beginning and ending values of the desired range of the independent variable given. In this latter method of use, intermediate values of the dependent variables (and the corresponding independent variable) can be stored and printed out. In fact, these intermediate values can also be stored and examined when the former method of use is employed with several calls between the beginning and ending values of the independent variable. The size of arrays used in ODESOL are established by integer values passed through arguments of the call, and therefore, the amount or memory required by ODESOL will correspond to the size of the problem being solved. For a single first-order equation, and very limited storage of intermediate values, a very small amount of memory is required by ODESOL, e.g., that of its code and variables, and the very small arrays passed as arguments. On the other hand, if a system of eight ordinary differential equations is being solved, the memory requirements for arrays will be larger.

C.3 HOW DO YOU USE ODESOL

The call in the driver program for ODESOL must consist of a statement such as

```
CALL ODESOL(YBEG,DYDX,NV,X1,X2,ERR,H1,HMIN,NSTOR,XP,YP,WK1,SLOPE)
```

in which

YBEG is a real array of dimensions NV, the elements of which represent the individual dependent variables for which a solution is being sought at the beginning of the interval, X1. On input, this array represents the starting values of the dependent variables, i.e., the values of the y's corresponding to the independent variable X1. Upon return from the call, the values in this array are the values of the dependent variables corresponding to the value of the independent variable X2 at the end of the interval of the solution.

DYDX is a real array of dimensions NV, the elements of which represent the individual derivatives of the dependent variables with respect to the independent variable, x. The subroutine SLOPE must define these derivatives. The main program need only dimension this array to size NV, or larger.

NV is an integer variable that defines the number of first-order equations that will be solved. In the example above, NV would equal 2.

X1 is the value of the independent variable at the beginning of the interval for which a solution is to be obtained.

X2 is the value of the independent variable at the end of the interval for which a solution is to be obtained; that is a solution will be obtained for x varying from X1 to X2. X2 can be less than X1, as well as larger than X1.

ERR is a real variable that defines the desired accuracy that ODESOL is to achieve in obtaining the solution. The step size used will be reduced (or enlarged), as necessary, to achieve this accuracy.

H1 is a starting increment for x that ODESOL will use in obtaining the solution. This value will be modified as needed to satisfy ERR. The usual procedure is to call ODESOL repeatedly to solve a complete problem over an extended range of the independent variable. When this is done H1 will be utilized only for the first call to ODESOL. Thereafter, the increment found appropriate from the previous call will be used and not H1. Therefore, it is best to provide H1 only for the first call to ODESOL in solving a given problem.

HMIN is a real value equal to the minimum step size that will be allowed in obtaining the solution. This value may be given as zero, and is positive even if X2 is less than X1.

NSTOR is an integer that agrees with the dimensions of XP and the second subscript dimension of YP. If the variable NBETN in the common statement is given a value of 0, then NSTOR can be 1.

NSTOR should never be assigned a value less than 1.

XP is a one-dimensional real array of size NSTOR that contains the values of the independent x upon return from the call to ODESOL, provided that NBETN is not zero. These values will not be on an equal interval of x, but will consist of values from X1 to X2.

YP is a two-dimensional real array of size NV (first subscript) and NSTOR (second subscript) that contains the values of the dependent variable y upon return from the call to ODESOL, provided that NBETW is not zero. The values in YP(I,J) with I between 1 and NV and J constant but between 1 and IBETW will be the values of the dependent variables y corresponding to XP(J).

WK1 is a two-dimensional real array of size NV (first subscript) and 13 as the second subscript that is used for the work space by ODESOL in obtaining the solution. That is WK1 should be dimensioned in the main program as WK1(NV,13).

SLOPE must be declared as EXTERNAL in the main or driver program and is the name given to the subroutine described below that defines the derivatives.

A COMMON statement must also be included in this main program that contains the five variables defined below. This COMMON statement can also be contained in the subroutine SLOPE, defined below, and if it is necessary to pass additional information from the main program to the subroutine SLOPE, this can be done by adding variables, or arrays to the end of the variables listed below. The common statement should consist of a statement such as

```
COMMON NGOOD,NBAD,NBETW,IBETW,DXBETW
```

in which

NGOOD is an integer variable that returns the number of steps used in the solution that equaled or exceeded the error condition established by the value of ERR above, e.g., good steps.

NBAD is an integer variable that returns the number of steps used in the solution that did not meet the error condition established by the value of ERR above, e.g., bad steps.

NBETW is an integer variable that defines the maximum number of intermediate values of x and of y's that will be returned in the arrays XP and YP above. If NBETW is zero, then no intermediate values will be returned. NBETW should not be greater than the dimensions of XP and YP as defined by the variable NSTOR, but the number of values returned in XP and YP will generally be less than NBETW, unless the needed step sizes are small, in which event the values will not be stored in XP and YP after NBETW values have been placed in these array. That is the value of IBETW, defined below will never exceed NBETW.

IBETW is an integer variable whose value from returning from a call to ODESOL equals the number of intermediate values of x and y's that have been stored in XP and YP above.

DXBETW is a real variable that defines the smallest increment of the independent variable for which intermediate values will be stored in the arrays XP and YP. Should the increment needed to solve the problem with the given error condition ERR become less than DXBETW, then some values actually obtained in the solution process will not be stored.

C.4 SUBROUTINE THAT YOU MUST SUPPLY

The subroutine SLOPE that defines the derivatives must consist of a beginning statement such as

```
SUBROUTINE SLOPE(X,Y,DYDX)
```

in which the name SLOPE is the EXTERNAL that is the last argument of the call to ODESOL.

X is a real variable that represents the independent variable x. Its value will be passed from ODESOL to SLOPE to use, as needed, to define the derivatives.

Y is a real array with the dimensions of NV above. Values of this array will be passed from ODESOL to SLOPE to use, as needed, to define the derivatives at X.

DYDX is a real array with dimensions of NV above. The subroutine SLOPE must contain the necessary statement that defines the elements of this array using X, and the elements of Y, appropriately to define the derivatives of the individual dependent variables with respective to the independent variable x. The name must correspond to the second argument of the call to ODESOL from the main program. Somewhere in the subroutine SLOPE, there must be a statement like DYDX(J) = … in which J must vary from 1 through NV.

SLOPE will be called repeatedly by ODESOL and must be written to provide the correct derivatives in the array DYDX that defines the system of ordinary differential equations that are being solved.

Example 1

As a very simple example, consider the gradually varied M_1 profile that develops in a trapezoidal channel. In this problem, the bottom width b of the channel is 10 ft and its side slope is 1. The bottom slope S_o of the channel equals 0.001, and its Manning's roughness coefficient n = 0.014. At station 10,000 ft, a gate causes the depth to be 6.3 ft.

The normal depth for this channel can be computed from Manning's equation, and is Y_o = 4.256 ft. Therefore, the GVF-profile will be from a depth of 6.3–4.3 ft (a small amount above the normal depth). This GVF-profile is governed by the differential equation,

$$\frac{dx}{dy} = \frac{1 - F_r^2}{S_0 - S_f}$$

in which S_f is the friction slope determined by Manning's equation. Note that x is the dependent variable and y is the independent variable, and since the right side of the equation depends only on the independent variable, this problem can be solved by numerically integrating the above equation rather than needing to solve an ODE. These types of problems can effectively be solved with a spread sheet such as LOTUS, using the trapezoidal rule to do the numerical integration.

An example of a driver program and a subroutine to use ODESOL is listed below.

APPEC1.FOR

```
      REAL X(1),XPRIME(1),XP(1),YP(1,1),WK1(1,13)
      EXTERNAL DXY
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE,B,FM,FN,SO,Q2,FNQ
      NV=1
      KMAX=0
      DXSAV=1.
      H1=-.01
      WRITE(6,* )'GIVE IOUT,TOL,DELY,XB,Q,FN,SO,B,FM,YBEG,YEND'
      READ(5,*) IOUT,TOL,DELY,XB,Q,FN,SO,B,FM,YBEG,YEND
      X(1)=XB
      FNQ=FN*Q/1.49
      Q2=Q*Q/32.2
      Y=YBEG
      WRITE(IOUT,100) Y,X
2     YZ=Y+DELY
      CALL ODESOL(X,XPRIME,NV,Y,YZ,TOL,H1,HMIN,1,XP,YP,WK1,DXY)
      Y=YZ
      WRITE(IOUT,100) Y,X
100   FORMAT(6X,2F10.3)
      IF(DELY .LT. 0.) GO TO 8
      IF(Y .LT. YEND) GO TO 2
      STOP
8     IF(Y .GT. YEND) GO TO 2
      STOP
      END
      SUBROUTINE DXY(Y,X,XPRIME)
      REAL X(1),XPRIME(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAV,B,FM,FN,SO,Q2,FNQ
20    A=(B+FM*Y)*Y
      T=B+2.*FM*Y
      P=B+2.*SQRT(FM*FM+1.)*Y
      SF=(FNQ*(P/A)**.66666667/A)**2
      A3=A**3
      FR2=Q2*T/A3
40    XPRIME(1)=(1.-FR2)/(SO-SF)
      RETURN
      END
```

The input to solve the above problem is: 0 .00001 −.1 10000 400 .014 .001 10 1 6.3 4.3 and the solution results are:

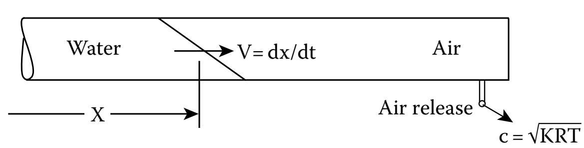
6.300	10000.000
6.200	9882.153
6.100	9762.763
6.000	9641.634
5.900	9518.536
5.800	9393.194
5.700	9265.275
5.600	9134.376
5.500	8999.996
5.400	8861.510
5.300	8718.119
5.200	8568.780
5.100	8412.104
5.000	8246.167
4.900	8068.219
4.800	7874.121
4.700	7657.220
4.600	7405.794
4.500	7096.100
4.400	6666.892
4.300	5836.147

Example 2

As a second example, consider solving the differential equation that describes the filling of a pipeline with water as air is released from an air-release valve at its downstream end, as shown in the sketch. The assumptions are that, the air flow is adiabatic, and does not offer frictional resistance, but must exhaust from the opening of the air-release valve. The limiting velocity of release is sonic velocity. The valve that allows the water into the pipeline is open instantly. Under these conditions, the differential equation that governs the water flow is

$$\frac{p_1 - p_2}{\gamma} - \frac{fx}{2gD}\left(\frac{dx}{dt}\right)^2 = \frac{x}{g}\frac{d^2x}{dt^2}$$

in which x is the distance from the reservoir to the leading edge of the advancing water column, p_1 is the pressure in the pipe at the reservoir, P_2 is the pressure in the air column, V = dx/dt is the velocity of the water flow, and f is the Darcy–Weisbach friction factor given by the Colebrook–White equation.



$$\frac{1}{\sqrt{f}} = \mathrm{Log}\left(\frac{e}{D} + \frac{9.35}{R_e\sqrt{f}}\right)$$

The above differential equation can be written as the coupled system of equations:

$$\frac{p_1 - p_2}{\gamma} - \frac{fx}{2gD}V^2 = \frac{x}{g}\frac{dV}{dt}$$

and

$$\frac{dx}{dt} = V$$

A listing of the main program and the subroutine needed to solve this problem are given below. The input consists of P1-the pressure p_1 in psi, P2-the air pressure in psi, T2-the air temperature in °F, X(1)-the distance from the reservoir to the valve in feet, i.e., the beginning position of the advancing water column, FL2-the remaining length of the pipe line in feet, NT-the number of time steps, DELT-the time increment in seconds, ANOZ-the area opening of the air release at its point of contraction in square feet, the equivalent sand roughness of the pipe wall in inches, and the diameter of the pipe in inches.

A possible line of input may consist of

```
80 0 60 100 4900 40 2 .1 .0004 8
```

Listing of FORTRAN program, APPEC2.FOR, to solve problem in Example 2

```
      EXTERNAL SLOPE
      REAL X(2),DXDT(2),WK1(2,13)
      COMMON NGOOD,NBAD,NBETW,IBETW,DXBETW,SF,ED,RE95,H12,D5
      NBETW=0
      READ (*,*) P1,P2,T2,X(1),FL2,NT,DELT,ANOZ,E,D
      P1=P1+14.7
      P2=P2+14.7
      T2=T2+460
      D12=D/12.
      ED=E/D
      D5=.5/D
      DXBETW=DELT
      SF=6.
      FL=X(1)+FL2
      A=.785398*D12**2
      RHOO=144.*P2/(1715.*T2)
      P20=P2
      TM=RHOO*A*FL
      X(2)=0.
      T20=T2
      TIME=0.
      DO 10 I=1,NT
      TIME1=DELT*FLOAT(I)
      IF(P2.GT.27.84) THEN
      PS=.528*P2
      TS=.833*T2
      ELSE
      PS=14.7
      TS=T2*(14.7/P2)**.2857
      ENDIF
      RHOS=.083965*PS/TS
      T2=T20*(P2/P20)**.2857
      X(2)=SQRT(12005.*(1.-TS/T2))
      G=ANOZ*X(2)*RHOS
```

```
      TM=TM-G*DELT*X(2)
      H12=2.3077*(P1-P2)
      IF(X(2).EQ. 0.) THEN
      RE95=.1
      ELSE
      RE95=1.122E-4/(D12*X(2))
      ENDIF
      CALL ODESOL(X,DXDT,2,TIME,TIME1,.00001,.2*DELT,TMIN,1,
     &XP,YP,WK1,SLOPE)
      TIME=TIME1
      FL2=FL-X(1)
      RHO=TM/(A*FL2)
      T2=T20*(RHO/RHOO)**.4
      P2=P20*(T2/T20)**3.5
10    WRITE(*,100)TIME,X,RHO,TM,PS-14.7,TS-460.,
     &G,T2-460.,P2-14.7
100   FORMAT(F8.1,2F8.2,F7.5,F8.3,2F8.2,F8.6,2F8.2)
      STOP
      END
      SUBROUTINE SLOPE(T,X,DXDT)
      COMMON NGOOD,NBAD,NBETW,IBETW,DXBETW,SF,ED,RE95,H12,D5
      REAL X(2),DXDT(2)
      DXDT(1)=X(2)
5     SF1=1.14-2.*ALOG10(ED+RE95*SF)
      IF(ABS(SF-SF1).LT.1.E-6) GO TO 10
      SF=SF1
      GO TO 5
10    F5=D5/(SF*SF)
      DXDT(2)=32.2/X(1)*H12-F5*X(2)*X(2)
      RETURN
      END
```

Example 3

As a third example, consider the flow from a reservoir whose water surface is 8 ft above the bottom of a trapezoidal channel with a bottom width, b = 10 ft, a side slope m = 1.5, a Manning's roughness coefficient, n = .013 and a bottom slope $S_o = 0.0008$. The entrance loss coefficient for the water entering the channel equals $K_e = 0.12$. At a distance 1200 ft downstream from the reservoir, a smooth transition occurs to a pipe with a diameter D = 10 ft. The pipe is laid on a steep slope. The bottom of the pipe and the trapezoidal channel are at the same level. Determine the discharge from the reservoir into the channel and the depth of flow throughout the length of the trapezoidal channel.

Solution

This problem involves the simultaneous solution of the ordinary differential equation for the gradually varied flow in the trapezoidal channel,

$$\frac{dY}{dx} = \frac{S_o - S_f}{1 - F_r^2}$$

and algebraic equations that govern the flow. At the entrance to the channel, the energy equation controls, e.g.,

$$H = 8 = y + \frac{Q^2}{2gA^2}$$

At the head of the steep pipe, the critical flow equation governs, e.g.,

$$\frac{Q^2T}{gA^3} = 1$$

and the specific energy at the beginning of the transition in the trapezoidal channel must equal the critical specific energy at the end of the transition, e.g.,

$$Y_2 + \frac{Q^2}{2gA_2^2} = Y_c + \frac{Q^2}{2gA_c^2} = E_c$$

Since ODEs cannot be solved simultaneously with algebraic equations except by trial, or iterative techniques, the procedure might be as follows:

First solve Manning's equation and the energy equation simultaneously at the entrance giving $Q_o = 1127.1$ cfs, and $Y_o = 6.845$ ft. With this flow rate, the critical depth in the pipe is $Y_c = 8.059$ ft and the associated critical specific energy is $E_c = 12.35$ ft. Solving the energy equation gives a depth of $Y_2 = 12.184$ ft at the end of the trapezoidal channel. This is the depth that can be used as a boundary condition in solving the ODE for the gradually varied flow in the trapezoidal channel. The main program and the subroutine for accomplishing such a solution are listed below.

Program APPC3.FOR

```
      REAL Y(1),DY(1),XP(1),YP(1,1),WK1(1,13)
      EXTERNAL DYX
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE,B,FM,FN,SO,Q2,FNQ
      WRITE(6,* )'GIVE IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND'
1     READ(5,*) IOUT,TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND
      H1=-.01
      Y(1)=YB
      FNQ=FN*Q/1.49
      Q2=Q*Q/32.2
      X=XBEG
      WRITE(IOUT,100) X,Y
2     XZ=X+DELX
      CALL ODESOL(Y,DY,1,X,XZ,TOL,H1,HMIN,1,XP,YP,WK1,DYX)
      X=XZ
      WRITE(IOUT,100) X,Y
100   FORMAT(6X,2F10.3)
      IF(DELX .LT. 0.) GO TO 8
      IF(X .LT. XEND) GO TO 2
      GO TO 1
8     IF(X .GT. XEND) GO TO 2
      STOP
      END
      SUBROUTINE DYX(X,Y,DY)
      REAL Y(1),DY(1)
      COMMON NGOOD,NBAD,KMAX,KOUNT,DXSAVE,B,FM,FN,SO,Q2,FNQ
20    A=(B+FM*Y(1))*Y(1)
      T=B+2.*FM*Y(1)
      P=B+2.*SQRT(FM*FM+1.)*Y(1)
```

```
      SF=(FNQ*(P/A)**.66666667/A)**2
      A3=A**3
      FR2=Q2*T/A3
40    DY(1)=(SO-SF+Q2*DA/A3)/(1.-FR2)
      RETURN
      END
```

Using the following line of input to the prompt of this program:

```
6 .0001 −200. 12.184 1127.1 .013 .0008 10 1.5 1200. 0.
```

(i.e., specifying Q = 1127.1 and the beginning depth at 1200 ft of, YB = 12.184 ft) gives the following solution:

Solution to GVF Problem	
1200.000	12.184
1000.000	12.031
800.000	11.879
600.000	11.727
400.000	11.575
200.000	11.424
.000	11.274

Since the depth of 11.274 ft is well above even the water surface elevation in the reservoir, it is clear that the flow rate will be considerably less than 1127.1 cfs that would exist under uniform flow conditions in a long trapezoidal channel of its size. Even though the pipe 1200 ft downstream is on a steep slope, it "chokes" the flow with the effect extending to the reservoir and reduces the flow rate. Trial flow rates must now be selected, the various algebraic equations resolved, and finally the ODE solved again. The table below shows trials until adequate agreement is achieved between column 2 and the last column.

Assumed Flow Rate	y_1 at Entr. Energy Eq.	Crit. Flow, Pipe		Depth y_2 Trap. Ch.	Depth y_2 GVF-Sol.
		y_c	E_c		
600 cfs	7.266 ft	5.870 ft	8.30 ft	8.128 ft	7.266 ft
700 in.	7.689 ft	6.361 ft	9.10 ft	8.925 ft	8.051 ft
650 in.	7.737 ft	6.120 ft	8.70 ft	8.526 ft	7.658 ft
660 in.	7.728 ft	6.169 ft	8.78 ft	8.606 ft	7.737 ft

The last flow rate of 660 cfs gives a close enough agreement to be acceptable. This last gradually varied flow for the resulting M_1 profile is

1200.000	8.606
1000.000	8.458
800.000	8.311
600.000	8.166
400.000	8.021
200.000	7.878
.000	7.737

C.5 ODESOLC-C-FUNCTION TO SOLVE ODE's

This description provides information to use the C-language function **odesolc**, to obtain solutions to ordinary differential equations. This C-function is a translation from FORTRAN of the subroutine described previously in this appendix. If you are a C-programmer read this section, and not the previous section.

The methods used in this procedure obtain high-accuracy solutions to ordinary differential equations with minimal computational effort. The methods utilize an extrapolation, with a modified midpoint that is called the Bulirsh–Stoer method. The method is not well adapted for nonsmooth functions, such as often occurs with tabular data.

The function **odesolc** is designed to solve a system of first-order (e.g., equations with first derivatives) ordinary differential equations or solve a higher-order ordinary differential equation. If a higher-order equation, of order N, is to be solved, it must be reduced first to a system (or coupled set) of N first-order differential equations. Thus, for example, if the second-order equation,

$$\frac{d^2y}{dx^2} + f(x)\frac{dy}{dx} = g(x)$$

is to be solved, it is first rewritten as the following two first-order coupled equations:

$$\frac{dz}{dx} = g(x) - f(x) \times z(x)$$

and

$$\frac{dy}{dx} = z(x)$$

Function **odesolc** has been designed to provide users with considerable flexibility. A common method of use will involve calling **odesolc** repeatedly with each new call over a new increment of the independent variable x until the solution over the desired range is achieved. Another method of use will be to execute **odesolc**, but once with the beginning and ending values of the desired range of the independent variable given. In this latter method of use, intermediate values of the dependent variables (and the corresponding independent variable) can be stored and printed out. In fact, these intermediate values can also be stored and examined when the former method of use is employed with several calls between the beginning and ending values of the independent variable. The size of pointers used in **odesolc** is established by integer values passed through arguments of the call, and therefore, the amount or the memory required by **odesolc** will correspond to the size of the problem being solved. For a single first-order equation, and very limited storage of intermediate values, a very small amount of memory is required by **odesolc**, e.g., that of its code and its variables, and the very small arrays passed as arguments. On the other hand, if a system of eight ordinary differential equations is being solved, the memory requirements for the arrays will be larger. A reduced version of odesolc, with the name odesolsc.c does not given any intermediate values, and does not add any values to xp or xp, and does not contain the int ngood, etc.

C.6 HOW DO YOU USE ODESOLC

The call in the driver program for **odesolc** must consist of a statement such as

```
Odesolc (float *y, float x1, float x2, float err, float h1, float
hmin, int nstor);
```

or

```
odesolsc (float *y, float *dydx, float x1, float x2, float err, float
h1, float hmin);
```

in which

y is a pointer with **nv** floating point values assigned to it, the elements of which represent the individual dependent variables for which a solution is being sought at the beginning of the interval, **x1**. On input, the pointer **y** represents the starting values of the dependent variables, i.e., the values of the y's corresponding to the independent variable **x1**. Upon return from the call, the values in this pointer are the values of the dependent variables corresponding to the value of the independent variable **x2** at the end of the interval of the solution.

x1 is the value of the independent variable at the beginning of the interval for which a solution is to be obtained.

x2 is the value of the independent variable at the end of the interval for which a solution is to be obtained; that is, a solution will be obtained for x varying from **x1** to **x2**. **x2** can be less than **x1**, as well as larger than **x1**.

err is a floating variable that defines the desired accuracy that **odesolc** is to achieve in obtaining the solution. The step size used will be reduced (or enlarged), as necessary, to achieve this accuracy.

h1 is a starting increment for x that **odesolc** will use in obtaining the solution. This value will be modified as needed to satisfy **err**. The usual procedure is to call **odesolc** repeatedly to solve a complete problem over an extended range of the independent variable. When this is done, **h1** from the previous call to **odesolc** will be used. **h1** can be given a negative or a positive value and regardless of the sign given to **h1**, its sign will be taken to be the same as that of the computed interval.

hmin is a real value equal to the minimum step size that will be allowed in obtaining the solution. This value may be given as zero, and is positive even if **x2** is less than **x1**.

nstor is an integer that establishes the size of array **xp** and the second subscript dimension of **yp** that might be used for additional information as described below. **nstor** can be 1, and should never be assigned a value less than 1.

Other arrays used in **odesolc** that may be accessed by modifying source.

xp is a floating pointer of size **nstor** that contains the values of the independent x upon return from the call to **odesolc**, provided that **nbetw** is not zero. **xp** is allocated in **odesolc**. The values in **xp** will not be on an equal interval of x, but will consist of values from **x1** to **x2**.

yp is a floating pointer of size **nstor** x **nv** that contains the values of the dependent variable y upon return from the call to **odesolc**, provided that **nbetw** is not zero.

wk1 is a floating pointer of size 13 x **nv** that is used for the work space by **odesolc** in obtaining the solution. Memory for **wk1** is allocated in **odesolc**.

The main(void) program that calls on odesolc must contain the #include “odesol.h” statement or declarations similar to those in this header file so that information is properly passed to the function slope that is described below. This header file consists of

```
int odesolc(float *y,float x1,float x2,float err,float h1,
  float hmin);
int ngood=0,nbad=0,nbetw=0,ibetw=0,nstor=1,nv=1;
float b,fm,so,q2,fnq,fms,dxbetw=0.000001;
```

extern variables are needed to pass information to the function **slope** from the main program and from the function **odesolc**, so that it can adequately define the derivatives. The example given later shows how this can be accomplished. Global variables defined in **odesolc** are: **ngood, nbad, nbetw, ibetw, dxbetw**

in which

ngood is an integer variable that returns the number of steps used in the solution that equaled or exceeded the error condition established by the value of **err** above, e.g., good steps.

nbad is an integer variable that returns the number of steps used in the solution that did not meet the error condition established by the value of **err** above, e.g., bad steps.

nbetw is an integer variable that defines the maximum number of intermediate values of x and of y’s that will be returned in the pointer **xp** and **yp** above. If **nbetw** is zero, then no intermediate values will be returned. **nbetw** should not be greater than the dimensions of **xp** and **yp** as defined by the variable **nstor**, but the number of values returned in **xp** and **yp** will generally be less than **nbetw** unless the needed step sizes are small, in which event the values will not be stored in **xp** and **yp** after **nbetw** values have been placed in these array. That is, the value of **ibetw**, defined below, will never exceed **nbetw**.

ibetw is an integer variable whose value from returning from a call to **odesolc** equals the number of intermediate values of x and y’s that have been stored in **xp** and **yp** above.

dxbetw is a floating variable that defines the smallest increment of the independent variable for which intermediate values will be stored in the arrays **xp** and **yp**. Should the increment needed to solve the problem with the given error condition **err** become less than **dxbetw**, then some values actually obtained in the solution process will not be stored.

C.7 PROCEDURE THAT YOU MUST SUPPLY

The function **void slope(x,*y,*dydx)** defines the derivatives and has the following arguments:

x is a floating variable that represents the independent variable x. Its value will be passed from **odesolc** to **slope** to use as needed to define the derivatives.

***y** is a floating pointer with **nv** variables of memory allocated. Values of this array will be passed from **odesolc** to **slope** to use as needed to define the derivatives at x.

***dydx** is a floating pointer with **nv** variables of memory allocated. The function **slope** must contain the necessary statement that defines the elements of this array using **x** and the elements of **y** appropriately to define the derivatives of the individual dependent variables with respective to the independent variable x. Somewhere in the function **slope**, there must be a statement like **dydx[0] =** ….

slope will be called repeatedly by **odesolc** and must be written to provide the correct derivatives in the array **dydx[]** that defines the system of ordinary differential equations that are being solved.

The header file **odesol.h** should be included in the main program to properly define **odesolc**, and the extern variables, as described above.

Example

As an example, consider the flow from a reservoir whose water surface is 8 ft above the bottom of a trapezoidal channel with a bottom width b = 10 ft, a side slope m = 1.5, a Manning's roughness n = .013 and a bottom slope $S_o = 0.0008$. The entrance loss coefficient for the water entering the channel equals $K_e = 0.12$. At a distance 1200 ft downstream from the reservoir, a smooth transition occurs to a pipe with a diameter D = 10 ft. The pipe is laid on a steep slope. The bottom of the pipe and the trapezoidal channel are at the same level. Determine the discharge from the reservoir into the channel and the depth of flow throughout the length of the trapezoidal channel.

Solution

This problem involves the simultaneous solution of the ordinary differential equation for the gradually varied flow in the trapezoidal channel,

$$\frac{dy}{dx} = \frac{S_o - S_f}{1 - F_r^2}$$

and the algebraic equations that govern the flow. At the entrance to the channel, the energy equation controls, e.g.,

$$H = 8 = y + \frac{Q^2}{(2gA^2)}$$

At the head of the steep pipe, the critical flow equation governs, e.g.,

$$\frac{Q^2T}{(gA^3)} + 1$$

and the specific energy at the beginning of the transition in the trapezoidal channel must equal the critical specific energy at the end of the transition, e.g.,

$$y_2 + \frac{Q^2}{2gA_2^2} = E_c$$

Since ODEs cannot be solved simultaneously with algebraic equations except by trial, or iterative techniques, the procedure might be as follows:

First, solve Manning's equation and the energy equation simultaneously at the entrance giving $Q_o = 1127.1$ cfs and $y_o = 6.845$ ft. With this flow rate, the critical depth in the pipe is $y_c = 8.059$ ft and the associated critical specific energy is $E_c = 12.35$ ft. Solving the energy equation gives a depth of $y_2 = 12.184$ ft at the end of the trapezoidal channel. This is the depth that can be used as a boundary condition in solving the ODE for the gradually varied flow in the trapezoidal channel. The main program and the procedure for accomplishing such a solution are listed below:

Program CHANRE2.C

```
#include <conio.h>
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "odesols.h"
void main(void){
int rep;
float yb,fn,c,tol,q,g,x,xbeg,xend,delx,xz,a,fmom,h1,hmin=.001;
float *y,*dydx;
y=(float *)calloc(nv,sizeof(float));
dydx=(float *)calloc(nv,sizeof(float));
wk1=(float *)calloc(13*nv,sizeof(float));
h1=.01;
L1:printf("Give: TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND,g\n");
scanf("%f %f %f %f %f %f %f %f %f %f %f",&tol,&delx,&yb,&q,&fn,\
  &so,&b,&fm,&xbeg,&xend,&g);
y[0]=yb; fms=2.*sqrt(fm*fm+1); c=1; if(g>30) c=1.486; fnq=fn*q/c;
  q2=q*q/g;
x=xbeg; a=(b+fm*y[0])*y[0]; fmom=(b/2+fm*y[0]/3)*y[0]*y[0]+q2/a;
printf("%10.3f %10.3f %10.3f\n",x,y[0],fmom);
do {xz=x+delx;
 odesolsc(y,dydx,x,xz,tol,h1,hmin);
 x=xz; a=(b+fm*y[0])*y[0];fmom=(b/2+fm*y[0]/3)*y[0]*y[0]+q2/a;
 printf("%10.3f %10.3f %10.3f\n",x,y[0],fmom);
 if(delx>0) {if(x<xend) rep=1; else rep=0;}\
   else {if(x>xend) rep=1; else rep=0;}
}while (rep);
printf("Solve another problem? 1=yes, 0=no ");scanf("%d",&rep);
  if(rep) goto L1;
  free(y);free(dydx);free(wk1);
}
void slope(float x,float *y,float *dydx){
extern float b,fm,so,q2,fnq,fms;
float a,t,p,sf,fr2;
a=(b+fm*y[0])*y[0]; t=b+2*fm*y[0]; p=b+fms*y[0];
sf=fnq*pow(p/a,0.66666667)/a; sf=sf*sf;
fr2=q2*t/(a*a*a);
dydx[0]=(so-sf)/(1-fr2); return;
}
```

```
Listing of header file odesols.h
int odesolsc(float *y,float *dydx,float x1,float x2,float\
  err,float h1,float hmin);
int nv=1;
float b,fm,so,q2,fnq,fms,dxbetw=0.000001,*wk1;
```

Notice that the above program listing calls on the smaller version of the ODE solver that does not allow intermediate values to be stored. To use the larger program you will need to include declarations for these added variables. The following program uses the complete ODE solver **odesolc.c**; with the header file odesol.h needed with it. A listing of these programs, and the header files, are on the CD.

CHANRE2O.C

```
#include <conio.h>
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "odesol.h"
void main(void){
int rep;
float yb,fn,c,tol,q,g,x,xbeg,xend,delx,xz,a,fmom,h1,hmin=.001;
float *y,*dydx;
y=(float *)calloc(nv,sizeof(float));
dydx=(float *)calloc(nv,sizeof(float));
wk1=(float *)calloc(13*nv,sizeof(float));
xp=(float *)calloc(nstor,sizeof(float));
yp=(float *)calloc(nv*nstor+12,sizeof(float));h1=.01;
L1: printf("Give: TOL,DELX,YB,Q,FN,SO,B,FM,XBEG,XEND,g\n");
scanf("%f %f %f %f %f %f %f %f %f %f %f",&tol,&delx,&yb,&q,&fn,\
  &so,&b,&fm,&xbeg,&xend,&g);
y[0]=yb; fms=2.*sqrt(fm*fm+1); c=1; if(g>30) c=1.486; fnq=fn*q/c;
  q2=q*q/g;
x=xbeg; a=(b+fm*y[0])*y[0]; fmom=(b/2+fm*y[0]/3)*y[0]*y[0]+q2/a;
printf("%10.3f %10.3f %10.3f\n",x,y[0],fmom);
do {xz=x+delx;
 odesolc(y,dydx,x,xz,tol,h1,hmin);
 x=xz; a=(b+fm*y[0])*y[0];fmom=(b/2+fm*y[0]/3)*y[0]*y[0]+q2/a;
 printf("%10.3f %10.3f %10.3f\n",x,y[0],fmom);
 if(delx>0) {if(x<xend) rep=1; else rep=0;} else {
   if(x>xend) rep=1; else rep=0;}
}while (rep);
printf("Solve another problem? 1=yes, 0=no ");scanf("%d",&rep);
  if(rep) goto L1;
 free(y);free(dydx);free(wk1);free(xp);free(yp);
}
void slope(float x,float *y, float *dydx){
extern float b,fm,so,q2,fnq,fms;
float a,t,p,sf,fr2;
a=(b+fm*y[0])*y[0]; t=b+2*fm*y[0]; p=b+fms*y[0];
sf=fnq*pow(p/a,0.66666667)/a; sf=sf*sf;
fr2=q2*t/(a*a*a);
dydx[0]=(so-sf)/(1-fr2);
}
```

Using the following line of input to the prompt of this program:

```
.0001 −200. 12.184 1127.1 .013 .0008 10 1.5 1200. 0. 32.2
```

(i.e. specifying Q = 1127.1 and the beginning depth at 1200 ft of, YB = 12.184 ft) gives the following solution:

Solution to GVF Problem	
1200.000	12.184
1000.000	12.031
800.000	11.879
600.000	11.727
400.000	11.575
200.000	11.424
.000	11.274

Since the depth of 11.274 ft is well above even the water surface elevation in the reservoir, it is clear that the flow rate will be considerably less than 1127.1 cfs that would exist under uniform flow conditions in a long trapezoidal channel of its size. Even though the pipe 1200 ft downstream is on a steep slope, it "chokes" the flow with the effect extending to the reservoir and reduces the flow rate. Trial flow rates must now be selected, the various algebraic equations resolved, and finally the ODE solved again. The table below shows trials until an adequate agreement is achieved between column 2 and the last column.

Assumed Discharge	y1 at Entrance (Energy Eq.)	Crit. Flow, Pipe		Depth y2 end Trap. Chan.	Depth y1 GVF-Sol.
		yc	Ec		
700 in	7.689 ft	6.361 ft	9.10 ft	8.925 ft	8.051 ft
660 in	7.728 ft	6.169 ft	8.78 ft	8.606 ft	7.737 ft
650 in	7.737 ft	6.120 ft	8.70 ft	8.526 ft	7.658 ft
600 cfs	7.266 ft	5.870 ft	8.30 ft	8.128 ft	7.266 ft

The last flow rate of 600 cfs gives a close enough agreement to be acceptable. This last gradually varied flow for the resulting M_1 profile is:

1200.000	8.128
1000.000	7.981
800.000	7.835
600.000	7.690
400.000	7.547
200.000	7.406
.000	7.267

C.8 DVERK: ODE SOLVER FROM THE INTERNATIONAL STATISTICAL MATHEMATICAL LIBRARIES, ISML

The differential equation solver DVERK from the ISML provides an alternative among many to the use of ODESOL. Most larger computer installations have the ISML available in the compiled form for use, and the ISML is also available for PCs. With relatively minor changes to the source code, a program that utilizes ODESOL can call on DVERK to accomplish the same solution and vice versus. The differential equation solver DVERK uses a Runga–Kutta–Verner fifth- and sixth-order method. A brief description of what is needed to utilize DVERK is given below. More details can be found in the ISML documentation.

To Use DVERK from ISML a main program needs to contain a callsuch as

`CALL DVERK(N,FCN,X,Y,XEND,TOL,IND,C,NW,W,IER)` **{IER is omitted in the PC version}**

in which the meaning of the arguments is as follows:

`N` is the number of equations (integer and input).

`FCN` —is the name of a subroutine for evaluating the derivatives such as,

```
SUBROUTINE FCN(N,X,Y,YPRIME)
REAL Y(N),YPRIME(N)
.    .
.    .
YPRIME(N)= . .
RETURN
END
```

that receives `X` and `Y` as input and must return the derivative(s) in `YPRIME`.
`FCN` must appear in an `EXTERNAL` statement in the calling or the main program that calls on `DVERK`.

`X` is the independent variable (real and input)

`Y` is the dependent variable and is a real array of at least N elements. On input, `Y(1)`...`Y(N)` supply initial values. On output, `Y(1)`...`Y(N)` are replaced with an approximate solution at XEND unless an error condition is detected.

`XEND` is the independent variable X at which the solution is desired (real).

`TOL` is a tolerance for error control.

`IND` is an indicator (integer both input and output). On initial entry, `IND` must be 1 or 2. `IND` = 1 causes all default options to be used. `IND` = 2 allows for options to be specified in C below. (See ISML documentation for details.)

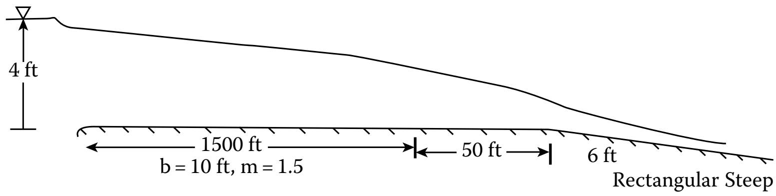
`C` is a communications real array of length 24. This array is used to select options and retain information between calls.

`NW` is the row dimensions of the matrix `W` (integer). `NW` must be greater or equal to `N`.

`W` is a workspace two-dimensional array (real). The first dimension of `W` must be `NW` and the second must be greater than or equal to 9. W must remain unchanged between consecutive calls.

`IER` is an error parameter that, if different from 0, indicates an error has been found (integer). See ISML documentation for error codes. (This parameter does not exist in the PC version of DVERK, i.e., the error messages are written to the screen, and it must be omitted from the argument list call to `DVERK`.)

To illustrate the use of DVERK of the IMSL, assume the problem of determining the flow rate into the channel sketched below, as well as the profile of water depths throughout it, to be solved. The reservoir is at an elevation 4 ft above the bottom of the channel. The upstream channel is trapezoidal with b = 10 ft and m = 1.5, and a bottom slope of $S_o = 0.0015$ and n = .013. At a position 1500 ft downstream, a smooth transition changes b and m linearly over a 50 ft length to 6 ft and m = 0, respectively. The rectangular channel thereafter, is on a steep slope.



The solution might begin by assuming normal depth at the entrance. Under this assumption, a simultaneous solution of the energy and Manning's equation give $Y_o = 3.016$ ft and $Q = 318.3$ cfs. Using this flow rate, it is possible to compute the depth of the water at the end of the transition by the critical flow equation, as $Y_2 = 4.438$ ft. This depth provides the boundary condition needed to solve the GVF-profile based on the assumption that the above flow rate is correct. The listing of the FORTRAN solution for accomplishing this solution is given below.

The solution of the GVF-profile based on the above indicates that the depth at the channel entrance is 3.125 or 0.107 ft above the normal depth; i.e., the M_1 GVF-profile reaches the entrance. Therefore, the flow rate must be reduced and computations repeated until the GVF-profile equals the depth at the entrance for the given flow rate.

Listing of FORTRAN program APPDVERK.FOR that uses DVERK

```
      REAL Y(1),C(24),W(2,9)
      EXTERNAL DYX
      COMMON X1,B1,B2,FM1,FM2,FN,SO,DB,DFM,FNQ,Q2,
     &DIFB,DIFM,B,FM,XBEG
      DATA IER,NN,IND,NW/0,1,1,2/
      WRITE(6,*)'GIVE IOUT,TOL,DELX1,DELX2,YB,Q,
     &FN,SO,B1,FM1,B2,FM2,XBEG,XEND,X1'
      READ(5,*) IOUT,TOL,DELX1,DELX2,YB,Q,FN,SO,
     &B1,FM1,B2,FM2,XBEG,XEND,X1
      Y(1)=YB
      DB=B2-B1
      DFM=FM2-FM1
      FNQ=FN*Q/1.49
      Q2=Q*Q/32.2
      DIFB=DB/(X1-XBEG)
      DIFM=DFM/(X1-XBEG)
      DELX=DELX1
      X=XBEG
      A=(B1+YB*FM1)*YB
      E=YB+Q2/(2.*A*A)
      WRITE(IOUT,100) X,Y,E
2     IF(X .LE. X1) DELX=DELX2
      XZ=X+DELX
      CALL DVERK(NN,DYX,X,Y,XZ,TOL,IND,C,NW,W,IER)
      IF(IND .LT. 1 .OR. IER .GT. 0) GO TO 10
      A=(B+Y(1)*FM)*Y(1)
      E=Y(1)+Q2/(2.*A*A)
      WRITE(IOUT,100) X,Y,E
100   FORMAT(6X,3F10.3)
      IF(X .GT. XEND) GO TO 2
      GO TO 12
```

```
10    WRITE(6,*)' ERROR TERMINATION',IER,IND
12    STOP
      END
      SUBROUTINE DYX(N,X,Y,YPRIME)
      REAL Y(N),YPRIME(N)
      COMMON X1,B1,B2,FM1,FM2,FN,SO,DB,DFM,FNQ,Q2,
     &DIFB,DIFM,B,FM,XBEG
      IF(X .LE. X1) GO TO 10
      XX=1.-(X-X1)/(XBEG-X1)
      B=B1+DB*XX
      FM=FM1+DFM*XX
      GO TO 20
10    B=B2
      FM=FM2
20    A=(B+FM*Y(1))*Y(1)
      T=B+2.*FM*Y(1)
      P=B+2.*SQRT(FM*FM+1.)*Y(1)
      SF=(FNQ*(P/A)**.66666667/A)**2
      A3=A**3
      FR2=Q2*T/A3
      IF(X .LE. X1) GO TO 30
      DA=(DIFB+Y(1)*DIFM)*Y(1)
      GO TO 40
30    DA=0.
40    YPRIME(1)=(SO-SF+Q2*DA/A3)/(1.-FR2)
      RETURN
      END
```

Input data for the above problem consists of:

```
6 .01 -5 -50 4.6 318.3 .013 .0015 6 0 10 1.5 1550 0 1500
```

C.9 RUNGE–KUTTA METHOD

C.9.1 Background

The Runge–Kutta method is very widely used to solve ordinary differential equations, and is the method used in the IMSL subroutine DVERK. A brief explanation of this method is given here. More extensive treatments of the method can be found in books dealing with numerical analysis. The Runge–Kutta method obtains the solution over each new increment Δx as a sequence of sub-steps. In other words, it is not necessary to obtain the start of the solution by the Euler predictor and the corrector as with other methods. The fourth-order Runge–Kutta method is widely used because it is a good balance between the accuracy and the amount of arithmetic.

C.9.2 Description of Method

To describe the Runge–Kutta method, the value of the dependent variable y at the next increment will be obtained from $y_{i+1} = y_i + \Delta y$. The Euler predictor obtains Δy by multiplying the increment in the independent variable Δx by the derivative of the function $dy/dx = y'$ evaluated at x_i, or $\Delta y = \Delta x y'(x_i, y_i)$. Consider a trial step to the midpoint of the increment. Use the values of x and y here to computer Δy, or $\Delta y = \Delta x y'(x_i + \Delta x/2, y_i + \Delta y_m)$, in which Δy_m is the Δy obtained from the Euler predictor for the mid point. This means of obtaining Δy is a second-order approximation, since the first-order terms cancel out. Use of this means for evaluating Δy is called the second-order Runge–Kutta, or the midpoint method.

The derivative y' can be evaluated using different combinations of independent and dependent variables, and from these, different values of Δy by multiplying by Δx. If the following are defined

$$\Delta y_1 = \Delta x y'(x_i, y_i) \quad \text{(The Euler predictor)}$$

$$\Delta y_2 = \Delta y_m = \Delta x y'\left(x_i + \frac{\Delta x}{2},\ y_i + \Delta y_m\right) = \Delta x y'\left(x_i + \frac{\Delta x}{2},\ y_i + \frac{\Delta y_1}{2}\right) \quad \text{(Runge–Kutta second order)}$$

$$\Delta y_3 = \Delta x y'\left(x_i + \frac{\Delta x}{2},\ y_i + \frac{\Delta y_2}{2}\right)$$

$$\Delta y_4 = \Delta x y'(x_i + \Delta x,\ y_i + \Delta y_3)$$

then the fourth-order Runge–Kutta formula is obtained as follows:

$$y_{i+1} = y_i + \frac{\Delta y_1 + \Delta y_4}{6} + \frac{\Delta y_2 + \Delta y_3}{3}$$

This formula requires that the derivative be evaluated four times in order to advance each new time step Δx, and an analysis of the terms truncated would show that terms involving Δx^5 are truncated, and therefore it provided a fourth-order approximation. A computer code to implement this fourth order Runge–Kutta approximate can consist of the following FORTRAN statement made into a subroutine.

APPERUK.FOR

```
      SUBROUTINE RUKU4(X,DX,Y)
      XH=X+.5*DX
      DY1=DX*SLOPE(X,Y)
      DY2=DX*SLOPE(XH,Y+.5*DY1)
      DY3=DX*SLOPE(XH,Y+.5*DY2)
      Y=Y+(DY1+DX*SLOPE(X+DX,Y+DY3))/6.+(DY2+DY3)/3.
      RETURN
      END
```

or

```
      SUBROUTINE RUKU4A(X,DX,Y)
      DX5=.5*DX
      XH=X+DX5
      DY1=SLOPE(X,Y) ! 1st sub-step
      DY2=SLOPE(XH,Y+DX5*DY1) ! 2nd sub-step
      Y3=SLOPE(XH,Y+DX5*DY2) ! 3rd sub-step
      Y=Y+DX*((DY1+SLOPE(X+DX,Y+DX*DY3))/6.+(DY2+DY3)/3.
      RETURN
      END
```

To use this subroutine requires a main program that calls it appropriately, and a FUNCTION subprogram SLOPE to evaluate the derivatives, much like the use of ODESOL or DVERK requires these.

The above listings are designed to solve a single ODE. If a system of ODEs is to be solved, as accommodated by ODESOL and DVERK, then arrays for Y and its derivatives are needed. Let SLOPE be a subroutine that returns the N derivatives for the N ODEs in its last array argument, evaluated at X and Y, its first two arguments. (Y must also be an array.) The solver could consist of the following subroutine:

APPCRKT.FOR

```
      SUBROUTINE RUKU4S(N,X,DX,Y)
      PARAMETER (NM=5)
      REAL Y(N),YT(NM),DY1(NM),DYT(NM),DYM(NM)
      DX5=.5*DX
      XH=X+DX5
      CALL SLOPE(X,Y,DY1) ! 1st sub-step
      DO 10 I=1,N
10    YT(I)=Y(I)+DX5*DY1(I)
      CALL SLOPE(XH,YT,DYT) ! 2nd sub-step
      DO 20 I=1,N
20    YT(I)=Y(I)+DX5*DYT(I)
      CALL SLOPE(XH,YT,DYM) ! 3rd sub-step
      DO 30 I=1,N
      YT(I)=Y(I)+DX*DYM(I)
30    DYM(I)=DYM(I)+DYT(I)
      CALL SLOPE(X+DX,YT,DYT) !4th sub-step
      DO 40 I=1,N
40    Y(I)=Y(I)+DX*((DY1(I)+DYT(I))/6.+DYM(I)/3.)
      RETURN
      END
```

The deficiency in using RUKU4 (or RUNK4S) is that the accuracy of the solution will be dependent upon the step size Δx used. One way to proceed would be to solve the ODE twice, once with some Δx and then with $\Delta x/2$, and if the solution agrees within an allowable error, accept the solution; otherwise reduce Δx again by one-half, etc. Rather than putting this type of burden on the user, it is much better to adjust the step size to satisfy some error criteria. The step sizes may be decreased, or increased, as suggested by the accuracy of the solution being obtained. To do this, an estimate of the error is needed. A means for obtaining this estimate is a "step doubling," i.e., each step is repeated; once using the full Δx and then, independently as two half steps $\Delta x/2$. Each of the three separate Runge–Kutta steps required using this function require four evaluates of y', but the single and the double computations share common arguments of x and y, initially, so the total number of evaluations of y' required is 11. Let the exact solution be denoted by y (without a subscript), the solution based on Δx by y_1, and the solution based on $\Delta x/2$ by y_2. Using the fourth-order Runge–Kutta method, the exact and the two numerical solutions are related by

$$y(x+\Delta x) = y_1 + C(\Delta x)^5$$

$$y\left(x+\Delta x\right) = y_2 + 2C\left(\frac{\Delta x}{2}\right)^5$$

in which C should remain constant over the step, i.e., from Taylor's series $C = (d^5y/dx^5)/5!$. Since y_1 involves $C\Delta x^5$ and y_2 involves $C\Delta x^5/16$, the difference between the two solutions provides a convenient estimate of the error, or

$$ERR = y_2 - y_1$$

In other words, the exact solution is

$$y(x + \Delta x) = y^2 + \frac{ERR}{15} + O^6$$

To develop a criteria to decide whether Δx should be changed to satisfy an accuracy requirement, let ERR_1 be the error from using Δx_1. Then, the step size Δx_o to produce an error of ERR_o is estimated by

$$\Delta x_0 = \Delta x_1 \left|\frac{ERR_0}{ERR_1}\right|^{1/5}$$

Let ERR_o be the error associated with the desired accuracy. If $|ERR_1|$ is larger than ERR_o, then the above equation gives $\Delta x = \Delta x_o$ to use to recompute the solution over the failed increment to satisfy the error condition ERR_o. In other words, if $|ERR_1|$ is larger than ERR_o, then the computations over the Δx_1 used did not satisfy the error requirement and need to be repeated with a smaller increment given by $\Delta x = \Delta x_1\{ERR_o/|ERR_1|\}^2$. If $|ERR_1|$ is less than ERR_o, then the above equation provides the Δx to use for the solution over the next step. In other words, the solution just obtained exceeds the accuracy needed and will be accepted, but the next increment will be increased so as to not do more arithmetic than necessary to satisfy the error ERR_o, as given by $\Delta x = \Delta x_1\{ERR_o/|ERR_1|\}^2$. For a system of ODEs, the errors (ERR_1) are an array of values, and the largest in magnitude should be used in the above formula. The listing below provides the logic needed to redetermine the step size to use to satisfy the error condition associated with the magnitude of ERROR.

RUKUST.FOR

```
      SUBROUTINE RUKUST(N,DXS,XBEG,XEND,ERROR,Y,YTT)
      PARAMETER (NM=5)
      REAL Y(N),YTT(N),YORI(NM)
      X1=XBEG
      DX=DXS
1     DO 10 I=1,N
      YTT(I)=Y(I)
10    YORI(I)=Y(I)
      X=X1
      IF(ABS(X+DX).GT.ABS(XEND)) DX=XEND-X
20    DX5=.5*DX
      CALL RUKU4S(N,X,DX5,Y) !Solve with 1/2inc.
      CALL RUKU4S(N,X+DX5,DX5,Y)
      X1=X+DX
      IF(ABS(X1).GT.ABS(XEND)-1.E-8) RETURN
      CALL RUKU4S(N,X,DX,YTT) !Solve with full inc.
      ERRM=0.
      X1=X+DX
```

```
      DO 30 I=1,N
      YTT(I)=Y(I)-YTT(I)
30    ERRM=MAX(ERRM,ABS(YTT(I)/Y(I)))
      IF(ERRM.EQ.0.) THEN
      DX=5.*DX
      DXS=DX
      GO TO 1
      ELSE
      ERRM=ERRM/ERROR
      DX=DX/ERRM**.2
      DXS=DX
      IF(ERRM.GT. 1.) THEN
      DO 40 I=1,N
      YTT(I)=YORI(I)
40    Y(I)=YORI(I)
      GO TO 20
      ENDIF
      ENDIF
      DO 50 I=1,N
50    Y(I)=Y(I)+YTT(I)/15.!Accounts for truncation error
      GO TO 1
      END
```

The arguments for RUKUST now have the following meanings:

N—number of ODEs that are to be solved, and for which derivatives will be given.

DXS—a starting value for Δx to use in solving the problem upon entry (this value will be decreased or increased in magnitude depending upon what is needed to satisfy the error criteria, ERROR. Note, in the previous Runge–Kutta subroutines, the DX is the increment over which the problem is to be solved. Now, DXS generally will be smaller than this increment. Upon returning from this subroutine, DXS represents the Δx that was found to be satisfactory at the end of the solution, and it can be used for the subsequent call to RUKUST as the starting increment.

XBEG—the beginning value for the independent variable.

XEND—the ending value for the independent variable. The difference between XEND and XBEG serves the same role as DX in the previous subroutine arguments.

ERROR—the error criterion that is to be meet in obtaining the numerical solution, as described above.

Y—is a real array of N values that upon entry to the subroutine provide the initial conditions to the dependent variable, and upon return from the subroutine, represent the solution of the dependent variable corresponding to x = XEND.

YTT—is a real array of N values that is used by RUKUST as the work space.

You should observe that this subroutine calls on RUKU4S three times; the first two times are to complete the solution over the increment Δx in two steps using Δx/2, and the third time using the increment Δx, i.e., using the four substeps involved in the Runge–Kutta method. The difference between these two solutions, (the first using Δx/2 and the second with Δx), are used to determine the error ERR (or ERR_1) above and then based on the above formula, the Δx that would supply the accuracy desired is computed. If the accuracy is not good enough, the solution is repeated using the computed Δx (the statement GO TO 20 does this). Otherwise, tests occur to see if the solution has proceeded to XEND. If not, then the solution proceeds over the next increment using the just computed Δx by going to statement one. There is some logic required to ensure that the solution ends at XEND, and this is done by making the last Δx equal to the current value of x and XEND.

C.9.2.1 Illustrative Use of Routines

If you followed through with the logic in developing the above routines, you should be able to write a main program and function, or subroutine that will solve ODEs for you. The following are given as examples if you find it helpful to use an example as a guide. The problem we will attempt to solve will be one in which the depth passes through critical, and therefore of course, a solution doesn't exist. Below, a program is given that calls on the fixed interval Runge–Kutta routine RUKU4A.

APPCR4A.FOR

```
      COMMON BO,FMO,FN,SO,QO,QS,UQ,DB,DM,G,C
      WRITE(6,*)'Give:IOUT,QO,QS,UQ,b,db,m,dm,So,n,Xbeg,Xend,DX,Y
     &beg,g'
      READ(5,*) IOUT,QO,QS,UQ,BO,DB,FMO,DM,SO,FN,XBEG,XEND,DX,YBE
     &G,G
      C=1.
      IF(G.GT.30.) C=1.486
      N=ABS(XEND-XBEG)/ABS(DX)
      Y=YBEG
      Q=QO+QS*XBEG
      B=BO+DB*XBEG
      FM=FMO+DM*XBEG
      A=(B+FM*Y)*Y
      WRITE(IOUT,100)XBEG,Y,Q,SQRT(Q*Q*(B+2*FM*Y)/(G*A**3))
100   FORMAT(F10.1,F10.3,F10.2,F10.3)
      DO 10 I=1,N
      CALL RUKU4A(X,DX,Y)
      X=XBEG+DX*FLOAT(I)
      Q=QO+QS*X
      B=BO+DB*X
      FM=FMO+DM*X
      A=(B+FM*Y)*Y
10    WRITE(IOUT,100)X,Y,Q,SQRT(Q*Q*(B+2*FM*Y)/(G*A**3))
      END
      FUNCTION SLOPE(X,Y)
      COMMON BO,FMO,FN,SO,QO,QS,UQ,DB,DM,G,C
      B=BO+DB*X
      FM=FMO+DM*X
      A=(B+FM*Y)*Y
      A2=A*A
      A3=A*A2
      Q=QO+X*QS
      QSG=QS/G
      Q2=Q*Q
      FNQ=FN*Q/C
      QG2=Q2/G
```

```
      QGA3=QG2/A3
      DA=Y*(DB+Y*DM)
      TA=Q2*DA/(G*A3)
      HC=(Y*Y*(B/2.+FM*Y/3.))/A
      FQ=((Q/A-UQ)*QSG+HC*DA)/A
      TQ=Q*QSG/(A*A)+FQ
      FR2=QGA3*(B+2.*FM*Y)
      SF=(FNQ*((B+2.*Y*SQRT(FM*FM+1.))/A)**.66666667/A)**2
      SLOPE=(SO-SF+TA-TQ)/(1.-FR2)
      RETURN
      END
```

The FUNCTION SLOPE is designed to solve a problem with lateral inflow (see Chapter 4).

The input that is used is as follows in response to the prompt:

Give: IOUT,QO,QS,UQ,b,db,m,dm,So,n,Xbeg,Xend,DX,Ybeg,g

```
3 400 1 0 10 0 1 0 .0008 .013 0 50 5 4 32.2
```

and the solution is as follows:

.0	4.000	400.00	.714
5.0	3.908	405.00	.752
10.0	3.789	410.00	.802
15.0	3.606	415.00	.883
20.0	3.760	420.00	.832
25.0	3.455	425.00	.972
30.0	3.130	430.00	1.160
35.0	3.580	435.00	.937
40.0	3.861	440.00	.834
45.0	3.574	445.00	.961
50.0	3.415	450.00	1.049

The last column is the Froude number. Notice that since the Froude number has changed from being less than unity to larger than unity twice, the solution has passed through critical conditions where the derivative dY/dx is infinite. Using an increment DX of 1 ft (but only displaying the solution on a 5 ft increment) results in the following:

.0	4.000	400.00	.714
5.0	3.908	405.00	.752
10.0	3.789	410.00	.802
15.0	3.606	415.00	.883
20.0	2.868	420.00	1.310
25.0	3.049	425.00	1.197
30.0	3.491	430.00	.966
35.0	4.070	435.00	.754
40.0	3.955	440.00	.800
45.0	3.787	445.00	.871
50.0	2.993	450.00	1.307

Note, these two numerical solutions are quite different. Obviously, good accuracy is not being achieved using a 5 ft increment (or even a 1 ft increment) since the flow is passing through a critical depth.

The main program and the subroutine to use the Runge–Kutta routine RUKUST above that finds an appropriate increment to meet the specified error criteria is listed below.

APPCRKT.FOR

```
      COMMON BO,FMO,FN,SO,QO,QS,UQ,DB,DM,G,C
      WRITE(6,*)'Give:IOUT,QO,QS,UQ,b,db,m,dm,So,n,Xbeg,Xend,DX,
     &Ybeg,g'
      READ(5,*) IOUT,QO,QS,UQ,BO,DB,FMO,DM,SO,FN,XBEG,XEND,DX,
     &YBEG,G
      C=1.
      FNQ=FN*Q
      IF(G.GT.30.) C=1.486
      N=ABS(XEND-XBEG)/ABS(DX)
      Y=YBEG
      Q=QO+QS*XBEG
      B=BO+DB*XBEG
      FM=FMO+DM*XBEG
      A=(B+FM*Y)*Y
      WRITE(IOUT,100)XBEG,Y,Q,SQRT(Q*Q*(B+2*FM*Y)/(G*A**3))
100   FORMAT(F10.1,F10.3,F10.2,F10.3)
      DXS=1.
      DO 10 I=1,N
      CALL RUKUST(1,DXS,DX*FLOAT(I-1),DX*FLOAT(I),1.E-6,Y,YTT)
      X=XBEG+DX*FLOAT(I)
      Q=QO+QS*X
      B=BO+DB*X
      FM=FMO+DM*X
      A=(B+FM*Y)*Y
10    WRITE(IOUT,100)X,Y,Q,SQRT(Q*Q*(B+2*FM*Y)/(G*A**3))
      END
      SUBROUTINE SLOPE(X,Y,DYX)
      REAL Y(1),DYX(1)
      COMMON BO,FMO,FN,SO,QO,QS,UQ,DB,DM,G,C
      B=BO+DB*X
      FM=FMO+DM*X
      A=(B+FM*Y(1))*Y(1)
      A2=A*A
      A3=A*A2
      Q=QO+X*QS
      QSG=QS/G
      Q2=Q*Q
      FNQ=FN*Q/C
      QG2=Q2/G
      QGA3=QG2/A3
      DA=Y(1)*(DB+Y(1)*DM)
      TA=Q2*DA/(G*A3)
      HC=(Y(1)**2*(B/2.+FM*Y(1)/3.))/A
      FQ=((Q/A-UQ)*QSG+HC*DA)/A
      TQ=Q*QSG/(A*A)+FQ
      FR2=QGA3*(B+2.*FM*Y(1))
```

```
      SF=(FNQ*((B+2.*Y(1)*SQRT(FM*FM+1.))/A)**.66666667/A)**2
      DYX(1)=(SO-SF+TA-TQ)/(1.-FR2)
      RETURN
      END
```

The solution failed with a division by zero, but the portion obtained is as follows: (This attempt at a solution used DX = 5 ft.)

.0	4.000	400.00	.714
5.0	3.908	405.00	.752
10.0	3.792	410.00	.801
15.0	3.616	415.00	.879

C.10 RUNGE–KUTTA USING C

If you are a C (or CPP)-programmer then you will want to read this section that implements the same algorithms, etc., as described using FORTRAN as the language, but that uses C as the programming language. If you use CPP, you might use its capabilities of passing arrays as arguments to function calls instead of pointers, making the program more identical to the previous FORTRAN listings. The following is the equivalent of the RUKUST subroutine as a void C function.

RUKUST.C

```
#include <math.h>
#include <stdlib.h>
extern void slope(float x,float *y,float *dy);
float yt[5],dy1[5],dyt[5],dym[5];
void ruku4s(int nm,float x,float dx, float *y){
/* nm = No. of ODE's, or order of ODE; x=indep. var. dx=inc. ind.
   var; y=dep. var. (array, i.e. pointer, allocated with nm
   elements) yt,dy1,dyt & dym must be global arrays (pointers)
   dimensioned to nm */
 float dx5,xh; int i;
 dx5=.5*dx; xh=x+dx5;
 slope(x,y,dy1); /* 1st sub-step */
 for(i=0;i<nm;i++) yt[i]=y[i]+dx5*dy1[i];
 slope(xh,yt,dyt); /* 2nd sub-step */
 for(i=0;i<nm;i++) yt[i]=y[i]+dx5*dyt[i];
 slope (xh,yt,dym); /* 3rd sub-step */
 for(i=0;i<nm;i++){yt[i]=y[i]+dx*dym[i];dym[i]+=dyt[i];}
 slope (x+dx,yt,dyt); /* 4th sub-step */
 for(i=0;i<nm;i++) y[i]+=dx*((dy1[i]+dyt[i])/6.+dym[i]/3.);
}   /* End of function ruku4s */
void rukust(int neq,float *dxs,float xbeg,float xend,\
  float error,float *y,float *ytt){
int i; float errm,dx,x,dx5,x1,yori[5];
 x1=xbeg; dx=*dxs;
L1: for(i=0;i<neq;i++){ytt[i]=y[i]; yori[i]=y[i];} x=x1;
  if(fabs(x+dx)>fabs(xend)) dx=xend-x;
L2: dx5=.5*dx;
 ruku4s(neq,x,dx5,y); /* Solve with 1/2inc. */
```

```
 ruku4s(neq,x+dx5,dx5,y);
 x1=x+dx; if(fabs(x1)>fabs(xend)-1.e-8) return;
 ruku4s(neq,x,dx,ytt); /* Solve with full inc. */
 errm=0.; x1=x+dx;
 for(i=0;i<neq;i++){ytt[i]=y[i]-ytt[i];
   errm=max(errm,fabs(ytt[i]/y[i]));}
 if(errm==0.){dx*=5.;*dxs=dx;goto L1;} else {errm/=error;
   dx/=pow(errm,.2);*dxs=dx;
  if(errm > 1.){for(i=0;i<neq;i++){ytt[i]=yori[i];y[i]=yori[i];}
  goto L2;}}
 for(i=0;i<neq;i++) y[i]+=ytt[i]/15.;
 goto L1;
} /* End of function rukust */
```

As was the case with the FORTRAN RUKUST subroutine, this C program is designed to accommodate up to five simultaneous ordinary differential equations. The program that calls on this function must have an extern declaration at its beginning as a prototype and the arguments are the same as those described for the FORTRAN subroutine RUKUST. Note that ruku4s is included as part of the listing RUKUST.C, and therefore only RUKUST must be linked with the calling program in creating an executable element. The listing below is an example of a C program that calls on the function rukust to solve a gradually varied flow profile in a trapezoidal channel.

Program RUKUSTS.C that call on RUKUST to solve a GVF in a trapezoidal channel

```
#include <conio.h>
#include <math.h>
#include <stdlib.h>
#include <stdio.h>
extern void rukust(int neq,float *dxs,float xbeg,float xend,\
  float error,float *y, float *ytt);
float b,m,n,so,q,c=1,q2g,fnq,fms;
void slope(float x,float *y,float *dy){ float a,fr2,sf;
 a=(b+m*y[0])*y[0];
 fr2=q2g*(b+2.*m*y[0])/(a*a*a);
 sf=pow(fnq*pow((b+fms*y[0])/a,.66666667)/a,2.);
 dy[0]=(so-sf)/(1.-fr2);
} /* End of function slope */
void main(void){float g,xbeg,xend,dx,ybeg,x,*y,*ytt,*dxs;\
  int i,nm; char fnam[20];
FILE *filo; printf("Give output file\n"); scanf("%s",fnam);
if((filo=fopen(fnam,"w"))==NULL){printf("Can not open output file\
  %s",fnam);exit(0);}
cprintf("Give: Q,b,m,So,n,Xbeg,Xend,DX,Ybeg,g\r\n");
scanf("%f %f %f %f %f %f %f %f %f %f",&q,&b,&m,&so,&n,&xbeg,&xend,\
  &dx,&ybeg,&g);
y=(float *)calloc(1,sizeof(float));
  ytt=(float *)calloc(1,sizeof(float));
dxs=(float *)calloc(1,sizeof(float));
if(g>30.) c=1.486; q2g=q*q/g; fnq=n*q/c;fms=2.*sqrt(m*m+1.);
nm=fabs(xend-xbeg)/fabs(dx);
y[0]=ybeg;x=xbeg;
fprintf(filo,"%10.1f %10.3f\n",xbeg,ybeg);
```

```
cprintf("%10.1f %10.3f\r\n",xbeg,ybeg);*dxs=.1*dx;
for(i=1;i<=nm;i++){rukust(1,dxs,x,x+dx,1.e-4,y,ytt); x+=dx;
fprintf(filo,"%10.1f %10.3f\n",x,*y);
  cprintf("%10.1f %10.3f\r\n",x,*y);}
}
```

EXAMPLE PROBLEM C.1

Use the above C program to solve the M_3-GVF downstream from a gate that causes the depth in a trapezoidal channel to be 1.5 ft immediately downstream from a gate. There is a flow rate Q = 400 cfs passing the gate, the channel has b = 10 ft, m = 1, n = 0.013, and S_o = 0.0004 downstream for the gate. Solve this GVF for a 100 ft downstream from the gate using 5 ft increments.

Solution

The input to the above program is: 400 10 1 .0004 .013 0 100 5 1.5 32.2
The output file is

X (ft)	Y (ft)	X (ft)	Y (ft)
0.0	1.500	55.0	1.650
5.0	1.514	60.0	1.663
10.0	1.527	65.0	1.677
15.0	1.541	70.0	1.691
20.0	1.554	75.0	1.705
25.0	1.568	80.0	1.718
30.0	1.582	85.0	1.732
35.0	1.595	90.0	1.746
40.0	1.609	95.0	1.760
45.0	1.622	100.0	1.774
50.0	1.636		

C.11 RUNGE–KUTTA–FEHLBERG METHOD

An alternative to "step doubling" to ascertain whether an error criteria is being satisfied is to use two Runge–Kutta methods of different orders. The Runge–Kutta–Fehlberg method uses one fourth order and one fifth order method simultaneously to move ahead one increment from x_i to x_{i+1}. When the details are worked out, only six function evaluations are required and the difference between $(y_{i+1})_5$ (fifth-order) and $(y_{i+1})_4$ (fourth-order) provides an estimate of the error that can be used to reduce the step size. The Runge–Kutta–Fehlberg algorithm consists of the following six evaluations of increments of the dependent variable, Δy.

$$\Delta y_1 = \Delta x y'\left(x_i, y_i\right)$$

$$\Delta y_2 = \Delta x y'\left(x_i + \frac{1}{4}\Delta x, y_i + \frac{1}{4}\Delta y_1\right)$$

$$\Delta y_3 = \Delta x y'\left(x_i + \frac{3}{8}\Delta x, y_i + \frac{1}{32}\left(3\Delta y_1 + 9\Delta y_2\right)\right)$$

$$\Delta y_4 = \Delta x y'\left(x_i + \frac{12}{13}\Delta x, y_i + \frac{1}{2197}\left(1932\Delta y_1 - 7200\Delta y_2 + 7296\Delta y_3\right)\right)$$

$$\Delta y_5 = \Delta x y'\left(x_i + \Delta x, y_i + \frac{439}{216}\Delta y_1 - 8\Delta y_2 + \frac{3680}{513}\Delta y_3 - \frac{845}{4104}\Delta y_4\right)$$

$$\Delta y_6 = \Delta x y'\left(x_i + \frac{1}{2}\Delta x, y_i + \frac{8}{27}\Delta y_1 + 2\Delta y_2 - \frac{3544}{2565}\Delta y_3 + \frac{1859}{4104}\Delta y_4 - \frac{11}{40}\Delta y_5\right)$$

The fourth-order approximation consists of

$$(y_{i+1})_4 = y_i + \frac{25}{216}\Delta y_1 + \frac{1408}{2565}\Delta y_3 + \frac{2197}{4104}\Delta y_4 - \frac{1}{5}\Delta y_5$$

and the fifth-order approximation consists of

$$(y_{i+1})_5 = y_i + \frac{16}{135}\Delta y_1 + \frac{6{,}656}{12{,}825}\Delta y_3 + \frac{2{,}856}{56{,}430}\Delta y_4 - \frac{9}{50}\Delta y_5 + \frac{2}{55}\Delta y_6$$

Taking the difference between $(y_{i+1})_5 - (y_{i+1})_4$,

$$E = \frac{1}{36}\Delta y_1 - \frac{128}{4275}\Delta y_3 - \frac{2{,}197}{75{,}240}\Delta y_4 + \frac{1}{50}\Delta y_5 + \frac{2}{55}\Delta y_6$$

that can be used to compare with the ERROR criteria for the accuracy desired. If $|E| < \text{ERROR}$, then computations continue using the same Δx (or Δx can be doubled if $|E|$ is much smaller than ERROR). If $|E| > \text{ERROR}$ then Δx is halved, etc., until $|E| < \text{ERROR}$.

Notice in implementing the Runge–Kutta–Fehlberg method $(y_{i+1})_5$ or E can be evaluated, but it is redundant to evaluate both since $E = (y_{i+1})_5 - (y_{i+1})_4$.

The subroutine RKFEHLF implements this method using the given fixed step size Δx. Its use is similar to RUKU4 (or RUKU4A), with the exception that the first argument is the name of the EXTERNAL that provides the derivative function that is to be solved, which must be a FUNCTION subprogram, and not a subroutine.

```
      SUBROUTINE RKFEHLF(DYX,X,DX,Y)
      EXTERNAL DYX
      PARAMETER(F2=.25,F31=.375,F32=.09375,F33=.28125,
     &F41=.92307692,F42=.87938097,F43=-3.27719618,F44=3.3208921,
     &F51=2.0324074,F52=-8.,F53=7.1734893,F54=-.2058967,
     &F61=-.2962963,F63=-1.3816764,F64=.4529727,F65=-.275,
     &Y1=.1185185,Y3=.51898635,Y4=.50613149,Y5=-.18,Y6=.036363636)
      DY1=DX*DYX(X,Y)
      DY2=DX*DYX(X+F2*DX,Y+DY1/4.)
      DY3=DX*DYX(X+F31*DX,Y+F32*DY1+F33*DY2)
      DY4=DX*DYX(X+F41*DX,Y+F42*DY1+F43*DY2+F44*DY3)
      DY5=DX*DYX(X+DX,Y+F51*DY1+F52*DY2+F53*DY3+F54*DY4)
      DY6=DX*DYX(X+.5*DX,Y+F61*DY1+2.*DY2+F63*DY3+F64*DY4+F65*DY5)
      Y=Y+Y1*DY1+Y3*DY3+Y4*DY4+Y5*DY5+Y6*DY6
      RETURN
      END
```

```
RKFEHLF.C
#include <stdlib.h>
#include <math.h>
void rkfehlf(float (*dyx)(float x,float y),float x,float dx,\
  float *y){
const float F2=.25,F31=.375,F32=.09375,F33=.28125,F41=.92307692,\
  F42=.87938097,F43=-3.27719618,F44=3.3208921,F51=2.0324074,\
  F52=-8.,F53=7.1734893,F54=-.2058967,F61=-.2962963,F63=-1.3816764,\
```

```
  F64=.4529727,F65=-.275,Y1=.1185185,Y3=.51898635,Y4=.50613149,\
  Y5=-.18,Y6=.036363636;
float yy,dy1,dy2,dy3,dy4,dy5,dy6;
yy=*y;dy1=dx*dyx(x,yy); yy=*y+.25*dy1; dy2=dx*dyx(x+F2*dx,yy);
yy=*y+F32*dy1+F33*dy2; dy3=dx*dyx(x+F31*dx,yy);
yy=*y+F42*dy1+F43*dy2+F44*dy3; dy4=dx*dyx(x+F41*dx,yy);
yy=*y+F51*dy1+F52*dy2+F53*dy3+F54*dy4; dy5=dx*dyx(x+dx,yy);
yy=*y+F61*dy1+2.*dy2+F63*dy3+F64*dy4+F65*dy5;
dy6=dx*dyx(x+.5*dx,yy);
*y+=Y1*dy1+Y3*dy3+Y4*dy4+Y5*dy5+Y6*dy6;
} /* End of function rkfehlf */
```

Note, subroutine RKFEHLF only uses the fifth-order equation to evaluate y_{i+1}, and it does not use the Error because it does not change the step size. As an example, let us use RKFEHLF to solve the differential equation. $dy/dx = e^x/Ln(y)$ with the initial condition that when x = 0.1, y = 1.4924887, i.e., y(.1) = 1.4924887. The exact solution to this ODE is the implicit equation $y(Ln(y) - 1) = e^x - 2$. The program below compares the solution with the exact solution. Notice that a function subprogram YEXACT is added to solve the implicit equation that gives the exact solution. (One would not always anticipate that the numerical solution obtained without an adaptive step size solver would duplicate the exact solution as is the case for this ODE and a step size of $\Delta x = 0.1$.)

```
      EXTERNAL DYX
      X=.1
      Y=1.4924887
      WRITE(3,100) X,Y,YEXACT(X,Y)
100   FORMAT(3F10.5)
      DO 10 I=1,19
      CALL RKFEHLF(DYX,X,.1,Y)
      X=X+.1
10    WRITE(3,100) X,Y,YEXACT(X,Y)
      END
      FUNCTION DYX(X,Y)
      DYX=EXP(X)/ALOG(Y)
      RETURN
      END
      FUNCTION YEXACT(X,YGUESS)
C YGUESS is a guess for Y
      XP=EXP(X)-2.
      M=0
      YEXACT=YGUESS
1     F=YEXACT*(ALOG(YEXACT)-1.)-XP
      M=M+1
      IF(MOD(M,2).GT.0) THEN
      F1=F
      YEXACT=YEXACT+.01
      GO TO 1
      ENDIF
      DIF=.01*F1/(F-F1)
      YEXACT=YEXACT-DIF-.01
      IF(ABS(DIF).GT. 1.E-6 .AND. M.LT.30) GO TO 1
      RETURN
      END
```

Output

x	y	y(exact)
.10000	1.49249	1.49249
.20000	1.73574	1.73573
.30000	1.94639	1.94639
.40000	2.14486	2.14486
.50000	2.33926	2.33926
.60000	2.53401	2.53401
.70000	2.73200	2.73200
.80000	2.93538	2.93538
.90000	3.14589	3.14589
1.00000	3.36508	3.36508
1.10000	3.59438	3.59438
1.20000	3.83517	3.83517
1.30000	4.08883	4.08883
1.40000	4.35675	4.35675
1.50000	4.64040	4.64040
1.60000	4.94128	4.94128
1.70000	5.26100	5.26100
1.80000	5.60125	5.60125
1.90000	5.96385	5.96385
2.00000	6.35074	6.35074

```
#include <stdlib.h>
#include <math.h>
#include <stdio.h>
float dyx(float x,float y) {return exp(x)/log(y);}
void rkfehlf(float (*dyx)(float x,float y),float x,float dx,\
  float *y);
float yexact(float x, float yguess){ int m;float xp,y,f,f1,dif;
 xp=exp(x)-2.; m=0; y=yguess;
 do{L1: f=y*(log(y)-1.)-xp; if((++m%2)){f1=f;y+=.01; goto L1;}
 dif=.01*f1/(f-f1); y-=(dif+.01);} while(fabs(dif)>1.e-6 ||\
   (m<30));
return y;}
void main(void){ int i; float x=.1,y=1.4924887,*yp; *yp=y;
printf("%10.5f %9.5f %9.5f\n",x,y,yexact(x,y));
for(i=0;i<19;i++){rkfehlf(dyx,x,.1,yp);x+=.1;y=*yp;
 printf("%10.5f %9.5f %9.5f\n",x,y,yexact(x,y));}
}
```

The subroutine RKFEHL, whose listing is given below, does use the Error equation and divides the increment Δx by one-half repeatedly until the computed error ERR is less in magnitude than the error ERROR given as its third argument. When using RKFEHL, it is necessary that it's calling program notes whether Δx has changed if the subroutine is called repeatedly within a DO loop. For example, the main program may consist of that listed after RKFEHL.

```
      SUBROUTINE RKFEHL(DYX,DX,ERROR,X,Y)
      EXTERNAL DYX
      PARAMETER(F2=.25,F31=.375,F32=.09375,F33=.28125,
```

```
     &F41=.92307692,F42=.87938097,F43=-3.27719618,F44=3.3208921,
     &F51=2.0324074,F52=-8.,F53=7.1734893,F54=-.2058967,
     &F61=-.2962963,F63=-1.3816764,F64=.4529727,
     &F65=-.275,Y1=.1185185,Y3=.51898635,Y4=.50613149,Y5=-.18,
     &Y6=.036363636,E3=-.02994152,E4=-.029199894)
      DX1=DX
      YY1=Y
      N=1
1     DO 10 I=1,N
      DY1=DX*DYX(X,Y)
      DY2=DX*DYX(X+F2*DX,Y+DY1/4.)
      DY3=DX*DYX(X+F31*DX,Y+F32*DY1+F33*DY2)
      DY4=DX*DYX(X+F41*DX,Y+F42*DY1+F43*DY2+F44*DY3)
      DY5=DX*DYX(X+DX,Y+F51*DY1+F52*DY2+F53*DY3+F54*DY4)
      DY6=DX*DYX(X+.5*DX,Y+F61*DY1+2.*DY2+F63*DY3+F64*DY4+
     &F65*DY5)
      Y=Y+Y1*DY1+Y3*DY3+Y4*DY4+Y5*DY5+Y6*DY6
      ERR=DY1/360.+E3*DY3+E4*DY4+DY5/50.+Y6*DY6
      IF(ABS(ERR).GT.ERROR) THEN
      DX=DX/2.
      Y=YY1
      N=DX1/DX
      GO TO 1
      ENDIF
10    CONTINUE
      END
C Main program (and Function DYX) that calls
C on RKFEHL M-1 additional times if DX was reduced
C to meet the error criteria; M = DSX/DX
      EXTERNAL DYX
      WRITE(*,*)' Give: Xo,Yo,DX,ERROR,N'
      READ(*,*) X,Y,DX,ERROR,N
      DXS=DX
      WRITE(3,100) X,Y
100   FORMAT(F8.2,F10.5)
      DO 20 I=1,N
      CALL RKFEHL(DYX,DX,ERROR,X,Y)
      IF(DX.LT.DXS) THEN
      M=DXS/DX+.5
      DO 10 J=1,M-1
      X1=X+DX*FLOAT(J-1)
10    CALL RKFEHL(DYX,DX,ERROR,X1,Y)
      ENDIF
      X=X+DXS
20    WRITE(3,100) X,Y
      END
      FUNCTION DYX(X,Y)
      DYX=X+Y
      RETURN
      END
```

```
#include <stdlib.h>
#include <math.h>
float dyx(float x,float y){return x+y;}
void rkfehl(float (*dyx)(float x,float y),float *dx,float error,\
  float x,float *y){
const float F2=.25,F31=.375,F32=.09375,F33=.28125,F41=.92307692,\
  F42=.87938097,F43=-3.27719618,F44=3.3208921,F51=2.0324074,\
  F52=-8.,F53=7.1734893,F54=-.2058967,F61=-.2962963,F63=-1.3816764,\
  F64=.4529727,F65=-.275,Y1=.1185185,Y3=.51898635,Y4=.50613149,\
  Y5=-.18,Y6=.036363636,E3=-.02994152,E4=.029199894;
float yy,dy1,dy2,dy3,dy4,dy5,dy6,dx1,yy1,err; int i,n;
 dx1=*dx;yy1=*y; n=1;
L1:for(i=0;i<n;i++){
yy=*y;dy1=dx1*dyx(x,yy); yy=*y+.25*dy1; dy2=dx1*dyx(x+F2*dx1,yy);
yy=*y+F32*dy1+F33*dy2; dy3=dx1*dyx(x+F31*dx1,yy);
yy=*y+F42*dy1+F43*dy2+F44*dy3; dy4=dx1*dyx(x+F41*dx1,yy);
yy=*y+F51*dy1+F52*dy2+F53*dy3+F54*dy4; dy5=dx1*dyx(x+dx1,yy);
yy=*y+F61*dy1+2.*dy2+F63*dy3+F64*dy4+F65*dy5;
  dy6=dx1*dyx(x+.5*dx1,yy);
*y+=Y1*dy1+Y3*dy3+Y4*dy4+Y5*dy5+Y6*dy6;
err=dy1/360.+E3*dy3+E4*dy4+dy5/50.+Y6*dy6;
  if(fabs(err)>error){*dx/=2.;*y=yy1;n=dx1/(*dx);goto L1;}}
} /* End of function rkfehl */
void main(void){ int i,j,n,m; float x,*y,yo,*dx,error,dxs,x1;
 printf("Give: Xo,Yo,DX,ERROR,N\n");
   scanf("%f %f %f %f %d",&x,&yo,&dxs,&error,&n);
 *dx=dxs; *y=yo; printf("%8.2f %9.5f\n",x,yo);
 for(i=0;i<n;i++) {
 rkfehl(dyx,dx,error,x,y);
 if((*dx) < dxs) { m=dxs/(*dx)+.5; for(j=1;j<(m-1);j++)\
   {x1=x+(*dx)*(j-1);rkfehl(dyx,dx,error,x1,y);}}
 x+=dxs; printf("%8.2f %9.5f\n",x,*y);}
}
```

Notice that subroutine RKFEHL also does not use the fourth order equation to compute $(y_{i+1})_4$, since the Error term is the difference between $(y_{i+1})_5$ and $(y_{i+1})_4$. Alternatively, $(y_{i+1})_4$ could be evaluated, and the Error obtained as the difference between the two values. A main program that uses RKFEHL to solve the ODE dy/dx = x + y is given below. Notice that it needs to contain logic so that if Δx is smaller upon returning from RKFEHL, multiple calls are needed to provide the solution over the original increment Δx; thus the additional DO 10 for J.

MRKFEHL.FOR

```
      EXTERNAL DYX
      WRITE(*,*)' Give: Xo,Yo,DX,ERROR,N'
      READ(*,*) X,Y,DX,ERROR,N
      DXS=DX
      WRITE(3,100) X,Y
100   FORMAT(F8.2,F10.5)
      DO 20 I=1,N
      CALL RKFEHL(DYX,DX,ERROR,X,Y)
      IF(DX.LT.DXS) THEN
```

```
      M=DXS/DX+.5
      DO 10 J=1,M-1
      X1=X+DX*FLOAT(J-1)
10    CALL RKFEHL(DYX,DX,ERROR,X1,Y)
      ENDIF
      X=X+DXS
20    WRITE(3,100) X,Y
      END
      FUNCTION DYX(X,Y)
      DYX=X+Y
      RETURN
      END
```

A better approach to making a user-friendlier Runge–Kutta–Fehlberg ODE solver is to pass it the beginning and ending values of the independent variable, and an estimate of an appropriate increment to use as a starter, as is done with RUKUST. The subroutine RKFEHLSG is a listing of such a subroutine. The use of RKFEHLSG is identical to RUKUST to solve ODEs, with the exception that the first argument is the name of the EXTERNAL, e.g., the subroutine that defines the derivatives for which solutions are sought. This EXTERNAL called by RKFEHLSG is a subroutine rather than a function subprogram since RKFEHLSG is designed to solve a system of first-order ODEs, and therefore an array of derivatives must be returned rather than a single value. This argument allows the user to use a name other than SLOPE for this subroutine if one so desires, but the name must also be declared EXTERNAL in the program that calls RKFEHLSG, otherwise the compiler would think this name is a REAL (or INTEGER) variable. You might want to add this additional argument to RUKUST so that other subroutine names, other than just SLOPE, could be used to evaluate the ODEs.

```
      SUBROUTINE RKFEHLSG(SLOPE,N,DXS,X1,X2,ERROR,Y,YYT)
      PARAMETER(F2=.25,F31=.375,F32=.09375,F33=.28125,F41=.92307692,
     &F42=.87938097,F43=-3.27719618,F44=3.3208921,F51=2.0324074,
     &F52=-8.,F53=7.1734893,F54=-.2058967,F61=-.2962963,
     &F63=-1.3816764,F64=.4529727,F65=-.275,Y1=.1185185,Y3=.51898635,
     &Y4=.50613149,Y5=-.18,Y6=.036363636,E3=-.02994152,
     &E4=-.029199894,M=4)
      REAL Y(N),YYT(N),DY1(M),DY2(M),DY3(M),DY4(M),DY5(M),DY6(M),
     &YT(M)
      IF(DXS.EQ.0.) STOP ' DXS cannot be 0'
      IF(X2.LT.X1 .AND. DXS.GT.0.) DXS=-DXS
      ERROR1=1.
      DX=DXS
      X=X1
1     DO 2 J=1,N
2     YYT(J)=Y(J)
3     IF(ABS(X+DX).GT.ABS(X2)) DX=X2-X
      CALL SLOPE(X,Y,DY1)
      DO 4 J=1,N
4     YT(J)=Y(J)+.25*DX*DY1(J)
      CALL SLOPE(X+F2*DX,YT,DY2)
      DO 5 J=1,N
5     YT(J)=Y(J)+DX*(F32*DY1(J)+F33*DY2(J))
      CALL SLOPE(X+F31*DX,YT,DY3)
      DO 6 J=1,N
```

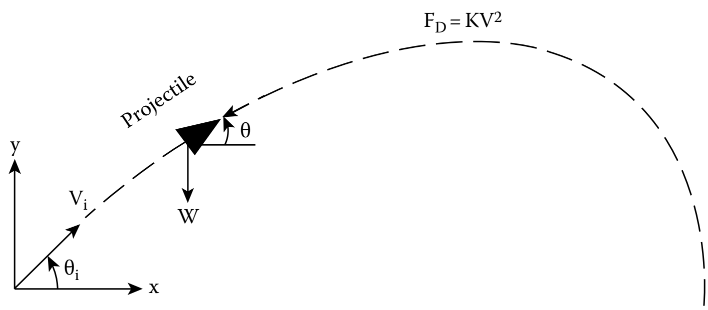
```
6     YT(J)=Y(J)+DX*(F42*DY1(J)+F43*DY2(J)+F44*DY3(J))
      CALL SLOPE(X+F41*DX,YT,DY4)
      DO 7 J=1,N
7     YT(J)=Y(J)+DX*(F51*DY1(J)+F52*DY2(J)+F53*DY3(J)+F54*DY4(J))
      CALL SLOPE(X+DX,YT,DY5)
      DO 8 J=1,N
8     YT(J)=Y(J)+DX*(F61*DY1(J)+2.*DY2(J)+F63*DY3(J)+F64*DY4(J)+
     &F65*DY5(J))
      CALL SLOPE(X+.5*DX,YT,DY6)
      ERR=0.
      DO 9 J=1,N
      Y(J)=Y(J)+DX*(Y1*DY1(J)+Y3*DY3(J)+Y4*DY4(J)+Y5*DY5(J)+
     &Y6*DY6(J))
9     ERR=ERR+DX*(DY1(J)/360.+E3*DY3(J)+E4*DY4(J)+DY5(J)/50.+
     &Y6*DY6(J))
      IF(ABS(ERR).GT.ERROR) THEN
      DX=DX/2.
      DXS=DX
      DO 10 J=1,N
10    Y(J)=YYT(J)
      GO TO 3
      ENDIF
      X=X+DX
      IF(ABS(X-X2).LT.1.E-7) RETURN
      IF(ABS(ERR+ERROR1).LT. .001*ERROR) THEN
      DX=2.*DX
      DXS=DX
      ENDIF
      ERROR1=ERR
      GO TO 1
      END
```

```
#include <stdlib.h>
#include <math.h>
void rkfehlsg(int (*slope)(float x,float *y, float *dxt),int n,\
  float *dxs,float x1,float x2,float error,float *y,float *ytt){
const float F2=.25,F31=.375,F32=.09375,F33=.28125,F41=.92307692,\
  F42=.87938097,F43=-3.27719618,F44=3.3208921,F51=2.0324074,\
  F52=-8.,F53=7.1734893,F54=-.2058967,F61=-.2962963,F63=-1.3816764,\
  F64=.4529727,F65=-.275,Y1=.1185185,Y3=.51898635,Y4=.50613149,\
  Y5=-.18,Y6=.036363636,E3=-.02994152,E4=-.029199894;
int j,irep; float err,error1,dx,x,dy1[4],dy2[4],dy3[4],dy4[4],\
  dy5[4],dy6[4],yt[4];
 error1=1.;x=x1; dx=*dxs;
do{
do{ for(j=0;j<n;j++) ytt[j]=y[j]; irep=0;
L2:slope(x,y,dy1); for(j=0;j<n;j++) yt[j]=y[j]+.25*dx*dy1[j];
  slope(x+F2*dx,yt,dy2);
    for(j=0;j<n;j++) yt[j]=y[j]+dx*(F32*dy1[j]+F33*dy2[j]);
  slope(x+F31*dx,yt,dy3); for(j=0;j<n;j++) yt[j]=y[j]+dx*(F42*dy1\
    [j]+F43*dy2[j]+F44*dy3[j]);
```

```
  slope(x+F41*dx,yt,dy4); for(j=0;j<n;j++) yt[j]=y[j]+dx*(F51*dy1\
    [j]+F52*dy2[j]+F53*dy3[j]+F54*dy4[j]);
  slope(x+dx,y,dy5); for(j=0;j<n;j++) yt[j]=y[j]+\
    dx*(F61*dy1[j]+2.*dy2[j]+F63*dy3[j]+F64*dy4[j]+F65*dy5[j]);
  slope(x+.5*dx,yt,dy6); err=0; for(j=0;j<n;j++){
   y[j]+=dx*(Y1*dy1[j]+Y3*dy3[j]+Y4*dy4[j]+Y5*dy5[j]+Y6*dy6[j]);
   err+=dx*(dy1[j]/360.+E3*dy3[j]+E4*dy4[j]+dy5[j]/50.+Y6*dy\
    6[j]);}
  if(fabs(err) > error) {dx/=2.;irep=1;*dxs=dx;
    for(j=0;j<n;j++)y[j]=ytt[j];}
}while (irep); if(err+error1 < .001*error){dx*=2.;*dxs=dx;}
error1=err; x+=dx; if(fabs(x+dx)>fabs(x2)) dx=x2-x;
}while (fabs(x-x2)>1.e-7); return;
} /* End of function rkfehlsg */
```

EXAMPLE PROBLEM C.2

A ship fires an explosive projectile with a weight W, from an initial angle θ_i from the horizontal and the initial velocity V_i, to hit a target. You are to write a program that will provide the solution of the motion of the projectile, that gives a table of values of x, y, V_x, V_y, and $V = \{V_x^2 + V_y^2\}^{1/2}$ at a number of even-spaced time increments, for any selected values of θ_i and V_i. Assume that, the air drag on the projectile will vary as the square of the total velocity, i.e., $F_{drag} = KV_2$. The ODEs that describe the projectile motion are given below. Give the input your program would need to solve the motion if the weight of the projectile W = 400 lbs, V_i = 800 ft/sec, $\theta_i = 45°$, and K = 0.0022.



Solution

To solve this problem, Newton's Second Law of motion must be used in both the x and y directions giving the two simultaneous second-order ODEs given below.

Newton's Second Law

$$\sum F_x = ma_x \quad \sum F_y = ma_y$$

$$-KV^2 \cos\theta = \frac{W}{g}\frac{dV_x}{dt} \quad -W - KV^2 \sin\theta = \frac{W}{g}\frac{dV_y}{dt}$$

$$\frac{d^2x}{dt^2} = \frac{dV_x}{dt} = -\frac{gKVV_x}{W} \quad \frac{d^2y}{dt^2} = \frac{dV_y}{dt} = -g - \frac{gKVV_y}{W}$$

These second-order equations need to be written as two first-order equations, or the first becomes V_x = dx/dt, with dV_x/dt as given above, and the second produces the two equations V_y = dy/dt and dV_y/dt, as given above. Thus, in solving these four simultaneous ODEs, a solution is sought for x, V_x, y, and V_y. These unknowns will be contained in an array X of four elements, so that X(1) = x, X(2) = V_x, X(3) = y, and X(4) = V_y. The derivatives that must be returned by the subroutine it calls on will also consist of four values, e.g., DXT(1) = dx/dt = V_x, DXT(2) = dV_x/dt, DXT(3) = dy/dt = V_y, and DXT(4) = dV_y/dt. The two listings below solve this problem by calling on RUKUST and RKFEHLSG, respectively. The input to these programs should consist of: 400 .0022 32.2 800 45 .1

PJECTILE.FOR (calls on RUKUST)

```
      REAL X(4),WX(4)
      COMMON G,GKW
      WRITE(*,*)' Give Projectile W,drag K,g',
     &',Initial V,angle,& DELt'
      READ(*,*) W,FK,G,V,ANGLE,DELT
      GKW=G*FK/W
      X(1)=0.
      X(3)=0.
      X(2)=V*COS(.0174532925*ANGLE)
      X(4)=V*SIN(.0174532925*ANGLE)
      DTS=.1*DELT
      T1=0.
10    T2=T1+DELT
      CALL RUKUST(4,DTS,T1,T2,1.E-5,X,WX)
      WRITE(3,100) T2,X,SQRT(X(2)**2+X(4)**2)
100   FORMAT(F7.2,5F9.3)
      IF(X(3).LT. 0.) STOP
      T1=T2
      GO TO 10
      END
      SUBROUTINE SLOPE(T,X,DXT)
      REAL X(4),DXT(4)
      COMMON G,GKW
      DXT(1)=X(2)
      DXT(3)=X(4)
      V=SQRT(X(2)**2+X(4)**2)
      DXT(2)=-GKW*X(2)*V
      DXT(4)=-G-GKW*X(4)*V
      RETURN
      END
```

PJECTILF.FOR (call on RKFEHLSG)

```
      EXTERNAL DERIV
      REAL X(4),WX(4)
      COMMON G,GKW
      WRITE(*,*)'Give Projectile W,drag K,g',
     &',Initial V,angle,& DELt'
      READ(*,*) W,FK,G,V,ANGLE,DELT
      GKW=G*FK/W
      X(1)=0.
```

```
      X(3)=0.
      X(2)=V*COS(.0174532925*ANGLE)
      X(4)=V*SIN(.0174532925*ANGLE)
      DTS=.1*DELT
      T1=0.
10    T2=T1+DELT
      CALL RKFEHLSG(DERIV,4,DTS,T1,T2,1.E-5,X,WX)
      WRITE(3,100) T2,X,SQRT(X(2)**2+X(4)**2)
100   FORMAT(F7.2,5F9.3)
      IF(X(3).LT. 0.) STOP
      T1=T2
      GO TO 10
      END
      SUBROUTINE DERIV(T,X,DXT)
      REAL X(4),DXT(4)
      COMMON G,GKW
      DXT(1)=X(2)
      DXT(3)=X(4)
      V=SQRT(X(2)**2+X(4)**2)
      DXT(2)=-GKW*X(2)*V
      DXT(4)=-G-GKW*X(4)*V
      RETURN
      END
```

Input data: `400 .0022 32.2 800 45 .1`

PJECTILE.C

```
#include <stdlib.h>
#include <math.h>
#include <stdio.h>
float g,gkw;
void rukust(int n,float *dxs,float x1,float x2,float error,\
  float *y,float *ytt);
int slope(float t,float *x,float *dxt){ float v;
 dxt[0]=x[1]; dxt[2]=x[3]; v=sqrt(x[1]*x[1]+x[3]*x[3]);
 dxt[1]=-gkw*x[1]*v; dxt[3]=-g-gkw*x[3]*v; return 0;}
void main(void){ float x[4],wx[4],*dts,delt,w,fk,v,angle,t1,t2;
  char fname[20];
FILE *fil;
 printf("Give Projectile W,drag K,g,initial V, angle & DELT\n");
 scanf("%f %f %f %f %f %f",&w,&fk,&g,&v,&angle,&delt);
   *dts=.1*delt;
 printf("Give output file name\n");scanf("%s",fname);
   fil=fopen(fname,"w");
 gkw=g*fk/w; x[0]=0.;x[2]=0.; x[1]=v*cos(.0174532925*angle);
   x[3]=v*sin(.0174532925*angle);
 t1=0;
do{ t2=t1+delt; rukust(4,dts,t1,t2,1.e-5,x,wx);
```

```
fprintf(fil,"%7.2f %9.3f %9.3f %9.3f %9.3f %9.3f\n",t2,x[0],x[1],\
  x[2],x[3],sqrt(x[1]*x[1]+x[3]*x[3]));
t1=t2;
} while (x[2]>0.); fclose(fil);
}
```

PJECTILF.C

```
#include <stdlib.h>
#include <math.h>
#include <stdio.h>
float g,gkw;
void rkfehlsg(int (*slope)(float x,float *y, float *dxt),int n,\
  float *dxs,float x1,float x2,float error,float *y,float *ytt);
int deriv(float t,float *x,float *dxt){ float v;
 dxt[0]=x[1]; dxt[2]=x[3]; v=sqrt(x[1]*x[1]+x[3]*x[3]);
 dxt[1]=-gkw*x[1]*v; dxt[3]=-g-gkw*x[3]*v; return 0;}
void main(void){ float x[4],wx[4],*dts,delt,w,fk,v,angle,t1,t2;
  char fname[20];
FILE *fil;
 printf("Give Projectile W,drag K,g,initial V, angle & DELT\n");
 scanf("%f %f %f %f %f %f",&w,&fk,&g,&v,&angle,&delt);
   *dts=.1*delt;
 printf("Give output file name\n");scanf("%s",fname);
   fil=fopen(fname,"w");
 gkw=g*fk/w; x[0]=0.;x[2]=0.; x[1]=v*cos(.0174532925*angle);
   x[3]=v*sin(.0174532925*angle);
 t1=0;
do{ t2=t1+delt; rkfehlsg(deriv,4,dts,t1,t2,1.e-5,x,wx);
fprintf(fil,"%7.2f %9.3f %9.3f %9.3f %9.3f %9.3f\n",t2,x[0],x[1],\
   x[2],x[3],sqrt(x[1]*x[1]+x[3]*x[3]));
t1=t2;
} while (x[2]>0.); fclose(fil);
}
```

Solution:

t(s)	x(ft)	V_x(ft/s)	y(ft)	V_y(ft/s)	V(ft/s)
.10	56.172	557.794	56.012	554.596	786.582
.20	111.567	550.141	110.929	543.790	773.539
.30	166.208	542.717	164.779	533.253	760.856
.40	220.118	535.512	217.588	522.975	748.516
.50	273.317	528.517	269.382	512.944	736.506
.	.	.	.	.	.
6.00	2493.005	320.257	2007.822	167.374	361.357
6.10	2524.929	318.225	2024.345	163.102	357.588
6.20	2556.651	316.226	2040.444	158.868	353.890
.	.	.	.	.	.
16.00	5004.523	199.335	1931.240	−155.942	253.086
16.10	5024.412	198.442	1915.520	−158.456	253.944

(continued)

t(s)	x(ft)	V_x(ft/s)	y(ft)	V_y(ft/s)	V(ft/s)
16.20	5044.212	197.550	1899.549	−160.957	254.820
.	.	.	.	.	.
23.80	6294.819	132.788	62.467	−308.287	335.669
23.90	6308.058	132.000	31.569	−309.667	336.627
24.00	6321.219	131.214	.534	−311.035	337.579
24.10	6334.301	130.431	−30.638	−312.388	338.525

WATER SCIENCE, TECHNOLOGY, AND ENGINEERING

OPEN CHANNEL FLOW

Numerical Methods and Computer Applications

A comprehensive treatment of open channel flow, ***Open Channel Flow: Numerical Methods and Computer Applications*** starts with basic principles and gradually advances to complete problems involving systems of channels with branches, controls, and outflows/inflows that require the simultaneous solutions of systems of nonlinear algebraic equations coupled with differential equations. The book includes a CD that contains a program that solves all types of simple open channel flow problems, the source programs described in the text, the executable elements of these programs, the TK-Solver and MathCad programs, and the equivalent MATLAB® scripts and functions.

The book provides applied numerical methods in an appendix and also incorporates them as an integral component of the methodology in setting up and solving the governing equations. Packed with examples, the book includes problems at the end of each chapter that give readers experience in applying the principles and often expand upon the methodologies used in the text. The author uses Fortran as the software to supply the computer instruction but covers math software packages such as MathCad, TK-Solver, MATLAB, and spreadsheets so that readers can use the instruments with which they are the most familiar. He emphasizes the basic principles of conservation of mass, energy, and momentum, helping readers achieve true mastery of this important subject, rather than just learn routine techniques.

With the enhanced understanding of the fundamental principles of fluid mechanics provided by this book, readers can then apply these principles to the solution of complex real-world problems. The book supplies the knowledge tools necessary to analyze and design economical and properly performing conveyance systems. Thus not only is the book useful for graduate students, but it also provides professional engineers the expertise and knowledge to design well performing and economical channel systems.

K11831

CRC Press
Taylor & Francis Group
an **informa** business
www.crcpress.com

6000 Broken Sound Parkway, NW
Suite 300, Boca Raton, FL 33487
270 Madison Avenue
New York, NY 10016
2 Park Square, Milton Park
Abingdon, Oxon OX14 4RN, UK

ISBN: 978-1-4398-3975-1
90000
9 781439 839751