

MASCARET

Theory guide

Version v8p5
December 1, 2023



Authors and contributions

This document was first created in French by Nicole GOUTAL (EDF) and Fabrice ZAOUI (EDF)

The translation was done by HR Wallingford

This version which comply with the official template of TELEMAT SYSTEM was created by Christophe Coulet (ARTELIA).

Work under construction

We would like to warn the reader that this document is still under construction and could be improved in many ways. Any remarks, suggestions, corrections, etc. are welcome and can be submitted to the TELEMAT SYSTEM development team through the forum in the Documentation category: <http://www.opentelemat.org/index.php/kunena/10-documentation>.

Abstract

Developed for more than 20 years by Electricité De France (EDF) and the Centre d'Etudes et d'Expertise sur les Risques, l'Environnement, la Mobilité et l'Aménagement (CEREMA), MASCARET is a hydraulic modelling software dedicated to one-dimensional free-surface flow, based on the Saint-Venant equations. MASCARET includes three computational engines, all of which can be coupled to the CASIER module in order to represent quasi-2D floodplain storage. The three engines can respectively represent the following flows:

- Subcritical and Transcritical steady;
- Subcritical unsteady;
- Transcritical unsteady.

The CASIER module represents water storage and movement on the floodplain with storage areas that can be connected to the river channel and connected to each other. Various relations can be used to represent these connections: weir, channel, siphon, orifice. MASCARET is suited for the following types of project:

- River and floodplain inundation modelling;
- Floodwave propagation after a hydraulic structure failure;
- Discharge regulation in rivers and canals;
- Wave propagation in canals (intumescence, lockage water, priming).

This report is the theoretical note for the software MASCARET and its algorithms. The principles of the mathematical modelling of one-dimensional free-surface flow used by the computational engines of MASCARET are presented here, as well as the numerical methods used to solve the equations for subcritical and transcritical flow. The principles of the storage areas module CASIER and the methods for coupling it with MASCARET are also described, as well as the principles of the automatic calibration module.

Contents

1	Introduction and general hypothesis	7
1.1	Introduction	7
1.2	General hypothesis	8
1.3	Definitions and notations	8
1.3.1	Definitions	8
1.3.2	Notations	10
2	The steady SARAP and unsteady REZO Finite Differences engines	12
2.1	The modelling principles	12
2.1.1	Definitions and notations	12
2.1.2	Basic modelling of a reach with low flow	12
2.1.3	Modelling a reach with high flow: compound channel	13
2.1.4	Treatment of the storage zones	18
2.1.5	Treatment of inflows	19
2.1.6	Treatment of junctions	19
2.1.7	Singular head losses	21
2.1.8	Singularities	21
2.1.9	Surcharged flow	24
2.2	Solution methods	24
2.2.1	Steady flow in a reach	24
2.2.2	Unsteady flow in a reach	29
2.2.3	Treatment of junctions in a steady subcritical regime	35
2.2.4	Computation of junctions in unsteady subcritical flow	41
2.2.5	Treatment of the singularities	42
2.2.6	Dealing with the transcritical flows	43
3	The Finite Volumes unsteady transcritical engine MASCARET	46
3.1	Introduction	46
3.2	Equations of Saint-Venant in a conservative form	47
3.2.1	General form of the equations (a reminder)	47
3.2.2	Equations written in a conservative form	48
3.2.3	Riemann invariants	50

3.2.4	The hydraulic jump	52
3.3	Numerical solution of the St-Venant equations	55
3.3.1	The explicit scheme	55
3.3.2	Use of the implicit scheme	60
3.3.3	Treatment of the source terms	65
3.3.4	Boundary conditions	71
3.3.5	Dry areas	76
3.4	Representation of more complex physical phenomena	77
3.4.1	Introduction to the modelling of compound channels	77
3.4.2	Confluence	79
3.4.3	Singularities	85
3.4.4	Eroding dams	87
3.4.5	Broad-crested and sharp-crested weirs	88
3.4.6	Summary of the treatment of singularities	90
3.4.7	Non-hydrostatic terms	91
4	The CASIER solver	96
4.1	Introduction	96
4.2	General principle	96
4.3	Derivation of the exchange flows	97
4.3.1	Weir link	98
4.3.2	Channel link	98
4.3.3	Siphon link	100
4.3.4	Orifice link	101
4.3.5	Drowning coefficient or correction coefficient	103
4.3.6	Contribution of rain	104
4.4	Coupling with the subcritical engine	104
4.5	Data pre-processing	105
5	Automatic calibration of the Strickler coefficient	109
5.1	Introduction	109
5.2	Definition of the problem	109
5.3	Algorithm of estimation of the Strickler coefficients	111
5.3.1	Reminder on the Saint-Venant equations	111
5.3.2	Detailed presentation of the minimisation algorithm	112
5.4	Method of calibration for compound channels	113
5.5	Validation and conclusions	113
	Bibliography	114

1. Introduction and general hypothesis

1.1 Introduction

This report sets out the mathematical modelling of one-dimensional subcritical and supercritical flows on which the three different hydraulic engines of the code system MASCARET are based, as well as the methods of solving equations resulting from these models. We present below the structure of the MASCARET system. This report only covers the *hydraulic* part of the system.

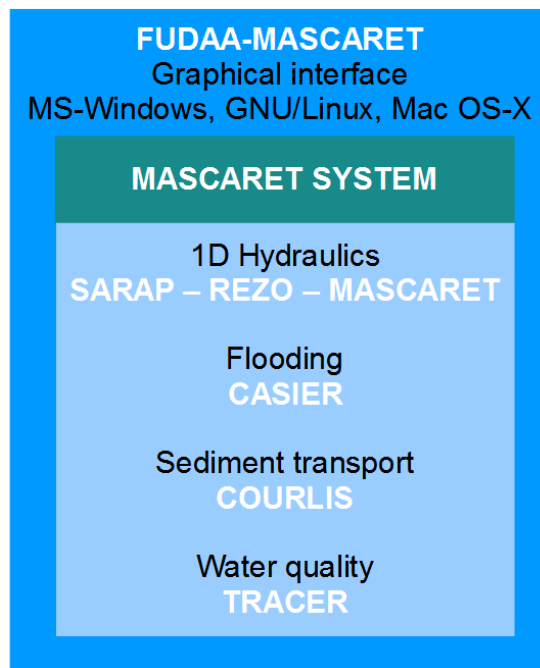


Figure 1.1: The MASCARET software

The numerical methods used vary depending on whether the flows are steady or unsteady, subcritical or supercritical and also depending on the type of hydraulic network : branched network (subcritical steady flow engine) or meshed network (subcritical unsteady flow engine). It is recalled that any branch arrangement are allowed in a meshed network, whilst in a branched network there is only one single path connecting two points of the network.

1.2 General hypothesis

The principal hypotheses justifying the modelling approach (1-D) used in the 3 hydraulic engines are stated below:

- each reach has a preferred axis of flow, the velocity vectors are always considered parallel to this axis ;
- the flow has a small curvature in the horizontal plane ; the vertical accelerations are negligible, and the distribution of pressures is quasi-hydrostatic ;
- the average slope of the flow is small (the cosine of the angle between the horizontal and the bottom is close to 1) ;
- the viscosity shear stress on the bottom and the banks are taken into account using empirical friction laws (Strickler's relation).

Therefore, in each plane perpendicular to its axis, the flow is entirely determined by the values of the average velocity (or of the total flow), and of the elevation of the free surface.

A few other hypotheses are also considered in order to simplify the data entry process and consider only the most usual cases :

- the influence of the wind is neglected;
- when a confluent is not modelled using a specific reach, it is assumed perpendicular to the main axis of flow: it is represented by a lateral flow input that does not add any momentum.

Other codes developed at EDF can be used when the hypotheses associated with MASCARET are too limiting, and complement the engines provided by MASCARET.

- Two-dimensional flows can be treated with the TELEMAC-2D system[20];
- The geometry of the reaches is not evolving during the simulations. The COURLIS code, which uses the hydraulic engine of MASCARET, is dedicated to the treatment of solid transport and its consequences ;
- Extended inundation plains, where the flows is not one-dimensional, can be modelled with a system of inter-connected storage cells. The CASIER code can be linked to the unsteady subcritical flow engine to widens the field of application of the MASCARET system. It represents a network of reaches and associated storage cells (equivalent to quasi-2d modelling)

1.3 Definitions and notations

1.3.1 Definitions

In the introduction we have outlined the basic characteristics of the modelling. We consider subcritical and supercritical flows in networks of reaches, each reach having a principal axis of flow. The calculated values are always relative to a section of flow perpendicular to this axis, each section is identified by its abscissa along this axis.

The flow sections are considered to be the union of the three sub-sections:

- The low flow channel, normal channel outside of flood periods. This channel can include islands but, if so, the elevation of the free surface is supposed to be identical on each side of the island. If this is not the case, one should resort to use a meshed network model ;
- The high flow channel, additional sections of flow active in times of flood, when the water level rises above the crest of the right or left bank. These sections are represented on each side of the channel, the elevation of the two banks being generally distinct ;
- The storage zones, treated like reservoirs, fill up as the flood rise and empty as water levels decrease. They act as water storage but, unlike the high flow channel, they do not convey any flow (the velocities in the direction of the axis of flow are supposed nil).

If the low flow channel is generally easy to identify, the limit between the high flow channel and storage zones is however much less well defined and can vary depending on the importance of the flood.

When the storage zones are likely to generate transverse flows in the floodplain, the present modeling approach becomes insufficient ; it is therefore necessary to couple MASCARET with a storage cell system.

It can be necessary to use a two-dimensional model instead of or in complement to the one-dimensional model, when the one-dimensional approach reaches its limits.

The estimation of the energy dissipated by friction is a fundamental element of the modelling. This is usually done with the Strickler roughness coefficient (see the following paragraph), it should be estimated separately in the low flow or high flow channel. The role of the calibration is to estimate the friction coefficients as well as possible by using the known natural level of the water. This calibration is often done by trial and errors. However, an optimisation algorithm has been specially developed for the steady flow engine, and this allows an automatic calibration of the friction coefficients.

Figure 1.2 and its notations summarise the key elements of the modelling approach in a flow section.

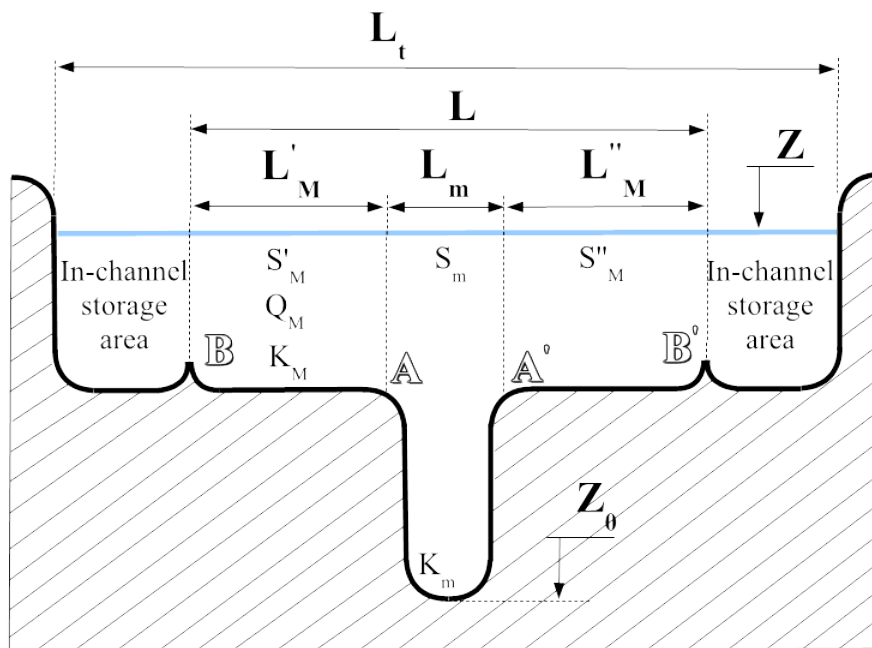


Figure 1.2: Representation of a profile - Notations

Comment :

it is important to distinguish between the “data” sections (or profiles) where the geometry is calculated directly from topographic information (survey, digital terrain model), and the “computation” sections (chosen by the user) where the values will be obtained by interpolation between “data” profiles. *In the computational sections, the flow variables are interpolated between data sections, not the bathymetry/geometry*

1.3.2 Notations

Unless otherwise stated, the following variables are used throughout the document (see figure 1.2) :

- t the time (s);
- x the abscissa along the principal axis of flow in the river (m);
- Q_m the discharge in the low flow channel ($m^3.s^{-1}$);
- Q_M the discharge in the high flow channel, left and right bank ($m^3.s^{-1}$);
- Q the total discharge in the active channel ($m^3.s^{-1}$) : $Q = Q_m + Q_M$;
- q_l the lateral inflow contribution per meter length ($m^2.s^{-1}$);
- L_m the surface width of the low flow channel (m);
- L_M the surface width of the high flow channel (m) : $L_M = L'_M + L''_M$;
- L the surface width of the active channel (m) : $L = L_m + L_M$;
- L_s the width of the storage zones (m);
- L_t the surface width of the total channel with storage (m);
- S_m the area of the low flow channel or wetted cross section (m^2);
- S_M the area of the high flow channel (m^2) : $S_M = S'_M + S''_M$;
- S the area of the active bed (m^2) : $S = S_m + S_M$;
- S_s the area of the active channel (m^2);
- S_t the area of the total channel with storage (m^2);
- K_m the Strickler coefficient of the low flow channel;
- K_M the Strickler coefficient of the high flow channel;
- P_m the wetted perimeter of the low flow channel (m) : $P_m = AA'$;

- P_M the wetted perimeter of the high flow channel (m) : $P_M = A'B' + AB$;
- P the wetted perimeter of the active channel (m) : $P = P_m + P_M$;
- R_m the hydraulic radius of the low flow channel (m) : $R_m = S_m/P_m$;
- R_M the hydraulic radius of the high flow channel (m) : $R_M = S_M/P_M$;
- R the hydraulic radius of the active channel (m) : $R = S/P$;
- V_m the velocity of water in the low flow channel ($m.s^{-1}$) : $V_m = Q_m/S_m$;
- V_M the velocity of water in the high flow channel ($m.s^{-1}$) : $V_M = Q_M/S_M$;
- V the mean velocity of the flow ($m.s^{-1}$) : $V = Q/S$;

2. The steady SARAP and unsteady REZO Finite Differences engines

2.1 The modelling principles

2.1.1 Definitions and notations

The hypotheses and notations are presented in sections 1.3.1 and 1.3.2 respectively.

2.1.2 Basic modelling of a reach with low flow

We use the well-established Saint-Venant equations :

- the continuity equation :

$$\frac{\partial Q}{\partial x} + \frac{\partial S}{\partial t} = q_l \quad (2.1)$$

with :

- $Q(x, t)$ the flow ($m^3.s^{-1}$);
- $Z(x, t)$ the elevation of the free surface (m).

- and the momentum equation :

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\beta \frac{Q^2}{S} \right) + gS \left(\frac{\partial Z}{\partial x} + J \right) = \gamma_l \quad (2.2)$$

with g the acceleration of gravity ($m.s^{-2}$).

The continuity equation relates to the conservation of the flow, the momentum equation corresponds to the fundamental equation of dynamics: the first member represents the acceleration of a *volume of water*; the second member represents the sum of the forces applied to it.

When there is no lateral inflow : $q_l = \gamma_l = 0$. In the case of a lateral inflow, the version 7.1.4 introduces two possible choices :

- the inflow is considered perpendicular to the flow and does not bring any momentum : $\gamma_l = 0$
- the inflow modifies the momentum equation : $\gamma_l = \frac{Q}{S} q_l$

The term gSJ relates to the effect of friction : J is a dimensionless number, representing the average rate of the energy dissipation. It depends on the flow, the hydraulic characteristics of the river and of course, on the roughness coefficient. It is calculated with the Strickler relation:

$$J = \frac{Q^2}{K_m^2 S^2 R^{4/3}} \quad (2.3)$$

where

$$J = \frac{Q^2}{D^2} \quad (2.4)$$

with :

- K_m the Strickler's roughness coefficient;
- R the hydraulic radius;
- D the conveyance.

The dimensionless coefficient β results from the variations of the actual flow velocity in a section. These variations would otherwise be ignored as only the average velocity is considered in the one-dimensional equations. The definition of β is :

$$\beta = \frac{S}{Q^2} \int V^2 dS \quad (2.5)$$

Taking into account the hypotheses described previously (see section 1.3.1), in a single channel we consider: $\beta = 1$, i.e. we ignore the variations of velocity within a section. However, that is not the case for a compound channel.

The boundary conditions in subcritical regime usually are:

- a discharge imposed at the upstream boundary;
- an elevation or, for unsteady flow a stage-discharge relationship, imposed at the downstream boundary.

In unsteady flow, it is possible (from a numeric point of view) to impose any types of boundary conditions (imposed elevation, discharge or a relationship between elevation and discharge) for both upstream and downstream boundaries. However, they have to be coherent to ensure that the algorithm runs successfully.

2.1.3 Modelling a reach with high flow: compound channel

Problem to be solved

The problem that arises in this case is due to the fact that we cannot directly write a system of equations analog to (2.1)(2.2). The energy losses due to friction are not identical in the two parts of the compound channel, and therefore:

- the hypothesis of a uniform velocity in a section and, in particular, a velocity identical in both the low flow and high flow channels, is not acceptable;
- we cannot define a priori a *global* headloss coefficient J .

It is therefore necessary to write a system (2.1)(2.2) for the low flow channel (index m) and for the high flow channel (index M), which gives :

$$\begin{cases} \frac{\partial S_m}{\partial t} + \frac{\partial Q_m}{\partial x} = 0 \\ \frac{\partial Q_m}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q_m^2}{S_m} \right) = -gS_m \frac{\partial Z}{\partial x} - gS_m J_m + \gamma_{mM} \end{cases} \quad (2.6)$$

and

$$\begin{cases} \frac{\partial S_M}{\partial t} + \frac{\partial Q_M}{\partial x} = 0 \\ \frac{\partial Q_M}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q_M^2}{S_M} \right) = -gS_M \frac{\partial Z}{\partial x} - gS_M J_M + \gamma_{mM} \end{cases} \quad (2.7)$$

with : $\gamma_{Mm} = -\gamma_{mM}$

With this formulation, we preserve the fundamental hypothesis: a free surface Z identical in both the low flow channel and the high flow channel.

The term γ_{mM} represents the interaction, initially unknown, between the two channels.

By adding the two systems, with the notation: $Q = Q_m + Q_M$ the total discharge, and : $S = S_m + S_M$ the total section, we can write:

$$\begin{cases} \frac{\partial S}{\partial t} + \frac{\partial Q}{\partial x} = 0 \\ \frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q_m^2}{S_m} + \frac{Q_M^2}{S_M} \right) = -gS \frac{\partial Z}{\partial x} - g(S_m J_m + S_M J_M) \end{cases} \quad (2.8)$$

with : $\sqrt{J_m} = \frac{Q_m}{D_m}$ et $\sqrt{J_M} = \frac{Q_M}{D_M}$

The low flow and high flow conveyances depend on the free surface elevation Z :

$$\begin{cases} D_m = K_m S_m R_m^{2/3} \\ D_M = K_M S_M R_M^{2/3} \end{cases} \quad (2.9)$$

The system (2.8) is made similar to the system (2.1)(2.2) by defining a *global* slope J with the following relationship :

$$SJ = S_m J_m + S_M J_M \quad (2.10)$$

and by choosing a coefficient β , different from 1 but satisfying the relationship :

$$\beta \frac{Q^2}{S} = \frac{Q_m^2}{S_m} + \frac{Q_M^2}{S_M} \quad (2.11)$$

which gives:

$$\beta = \frac{S}{Q^2} \left(\frac{Q_m^2}{S_m} + \frac{Q_M^2}{S_M} \right) \quad (2.12)$$

We have now two additional unknowns, Q_m and Q_M , so there are four unknowns in total with the two usual unknowns Q and Z .

However we have the simple relationship:

$$Q_m + Q_M = Q \quad (2.13)$$

To solve the system (2.8), it is necessary to write an additional differential equation or, more simply, to write an additional relationship linking the two unknowns. This relationship can be obtained by expressing the total conveyance D according to the low flow and high flow conveyances D_m, D_M . First the slope J is written as:

$$S \frac{Q^2}{D^2} = S_m \frac{Q_m^2}{D_m^2} + S_M \frac{Q_M^2}{D_M^2} \quad (2.14)$$

A simple assumption can be made, the slopes of the energy lines are identical in each channel :

$$\frac{Q_m}{D_m} = \frac{Q_M}{D_M} \quad (2.15)$$

Considering the definition of J , this is equivalent to set the global conveyance as :

$$D = D_m + D_M \quad (2.16)$$

Due to its simplicity, this solution has the advantage of being easily generalised to a generic compound channel, i.e. comprised of more than two channels (low flow and high flow), each channel having a specific roughness. The considerations above remain valid, and it is sufficient to take the global conveyance as the sum of the conveyances of each channel.

However, this approach neglects the interactions between the channels, these interactions result in the energy lines being different. The following approach is developed to take into account this phenomenon.

First of all we define the generalised low flow and high flow conveyances D_{mg} , D_{Mg} with the following relationships:

$$\begin{cases} Q_m = D_{mg} \sqrt{J} \\ Q_M = D_{Mg} \sqrt{J} \end{cases} \quad (2.17)$$

We consider the variable: $\eta = \frac{Q_m}{Q_M}$ and we use again the expression defining J , we can write:

$$\frac{D_{mg}}{D_{Mg}} = \eta \quad (2.18)$$

and also:

$$D_{mg}^2 = \frac{S}{\frac{S_m}{D_m^2} + \frac{1}{\eta^2} \frac{S_M}{D_M^2}} \quad (2.19)$$

We therefore establish that we only need to know the value of one of the following variables: η , D_{mg} , or D_{Mg} . Once one of these three variables is known, the others two can be calculated.

The system (2.8) is therefore rewritten as:

$$\begin{cases} \frac{\partial S}{\partial t} + \frac{\partial Q}{\partial x} = 0 \\ \frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\beta \frac{Q^2}{S} \right) = -gS \frac{\partial Z}{\partial x} - gSJ \end{cases} \quad (2.20)$$

with :

$$\begin{cases} \beta = \left(\frac{\eta^2}{S_m} + \frac{1}{S_M} \right) \frac{S}{(1+\eta)^2} \\ \sqrt{J} = \frac{Q}{D_{mg} + D_{Mg}} \end{cases} \quad (2.21)$$

If either η , D_{mg} or D_{Mg} is known, the variables β and J are also known, and the system (2.20) is perfectly analog to the system (2.1)(2.2) established for a single channel. The solution gives the value of Q , and the following relationships provide the values of Q_m and Q_M :

$$\eta = \frac{Q_m}{Q_M} = \frac{D_{mg}}{D_{Mg}} \quad (2.22)$$

and :

$$Q = Q_m + Q_M \quad (2.23)$$

In this context, the *simple* solution: $J_m = J_M = J$ (see above), is equivalent to:

$$D_{mg} = D_m \quad (2.24)$$

and :

$$D_{Mg} = D_M \quad (2.25)$$

Debord Modelling

This approach allows to differentiate the roughness in the low flow channel and in the high flow channel. It has been established experimentally, by studying the uniform regime and by identifying a law of discharge distribution of the type: $\frac{Q_m}{Q_M}$ (see [37],[12]).

The extrapolation to the gradually varied regime is appropriate, since this regime is defined as a succession of steady states.

The approach is based on the following consideration: in a compound channel, under the conditions described above, the discharge in the low flow channel is equal to the discharge in an identical single channel multiplied by a characteristic value A with : $A \in]0, 1[$.

A is expressed as a function of the roughness of each bed. Its expression is given later, as the variables D_{mg} and D_{Mg} need to be calculated first.

With the assumption above we can write: $D_{mg} = AD_m$, or : $Q_m^2 = A^2 D_m^2 J$.

We then derive D_{Mg} , by again using the expression of J :

$$SJ = S_m \frac{Q_m^2}{D_m^2} + S_M \frac{Q_M^2}{D_M^2} = A^2 S_m J + S_M \frac{Q_M^2}{D_M^2} \quad (2.26)$$

This is equivalent to:

$$J(S - A^2 S_m) = Q_M^2 \frac{S_M}{D_M^2} \quad (2.27)$$

And we finally obtain:

$$D_{Mg}^2 = \frac{Q_M^2}{J} = D_M^2 \left(\frac{S - A^2 S_m}{S_M} \right) = D_M^2 \left(1 + \frac{S_m}{S_M} (1 - A^2) \right) \quad (2.28)$$

The generalised high flow channel conveyance is therefore equal to the high flow channel conveyance multiplied by the value :

$$\sqrt{1 + \frac{S_m}{S_M} (1 - A^2)}$$

When $A = 1$, we obtain the *simple* model previously introduced. The ratio $\eta = \frac{Q_m}{Q_M}$ can be rewritten as:

$$\eta = \frac{A}{\sqrt{1 + \frac{S_m}{S_M} (1 - A^2)}} \frac{D_m}{D_M} \quad (2.29)$$

A is the constant parameter of the *Debord* model. It is evaluated with the relation:

$$A = \frac{1 - A_0}{2} \cos\left(\frac{\pi r}{0.3}\right) + \frac{1 + A_0}{2} \quad \text{for } r = \frac{R_M}{R_m} \in [0, 0.3] \quad (2.30)$$

and :

$$A = A_0 = 0.9 \left(\frac{K_m}{K_M}\right)^{-1/6} \quad \text{for } r > 0.3 \quad (2.31)$$

This equation has been derived from experiments on physical models.

It should be noted that A varies with R_m and R_M , so that $A = 1$ when $R_M = 0$ (no overflow).

Bottom-Bank Modelling

This approach is introduced to deal with low flow channels where it is important to distinguish the roughness of the banks from the roughness of the bottom. We can however follow the method used in the previous paragraph: here the compound channel results from the association of the section above the bottom (in place of the low flow channel) and the section above the banks (in place of the high flow channel).

The theoretical framework is the Mulloffer-Einstein hypothesis [12] for the composition of the roughness. This assumes that:

- the slope of the energy lines are the same for the two flows;
- the velocities are also the same.

The first hypothesis leads to the simple method of the conveyances composition described in the previous paragraph. This would be enough to solve the problem as it has so far been described, but then the second hypothesis (equality of velocities) cannot be guaranteed to be true. To impose this, we have to consider that the wetted sections in each channel are also unknown, i.e. they are not derived from the free surface elevation. This is equivalent to considering that the limit between the flows is not given by the two verticals above the bottom on the right and left bank.

We use the index 1 for the *bottom* flow and the index 2 for the *banks* flow, and we introduce two additional unknowns S_1 and S_2 . The two additional relationships are therefore :

- the total surface : $S_1 + S_2 = S$;
- the equality of velocity : $\frac{Q_1}{S_1} = \frac{Q_2}{S_2}$, which we aim to impose.

The total conveyance D is calculated by introducing an equivalent roughness k defined by :

$$D = \frac{Q}{\sqrt{J}} = kSR^{2/3} \quad (2.32)$$

The equality of the energy lines is accounted with the relationship:

$$D = D_1 + D_2 = k_1 S_1 R_1^{2/3} + k_2 S_2 R_2^{2/3} \quad (2.33)$$

and the equality of the velocities is accounted with the following relationship (2.34) where Strickler's friction relation is applied to each flow section:

$$k_1 R_1^{2/3} = k_2 R_2^{2/3} \quad (2.34)$$

By combining the relationships (2.32)(2.33)(2.34) we immediately obtain :

$$kR^{2/3} = k_1R_1^{2/3} = k_2R_2^{2/3} \quad (2.35)$$

$$R_1 = \frac{S_1}{P_1} \quad (2.36)$$

$$R_2 = \frac{S_2}{P_2} \quad (2.37)$$

$$R = \frac{S}{P} \quad (2.38)$$

from which we can deduce the expression of k characterising the composition of roughness :

$$k^{3/2} = \frac{P}{\frac{P_1}{k_1^{3/2}} + \frac{P_2}{k_2^{3/2}}} \quad (2.39)$$

Eventually, this approach is equivalent to considering a single bed whose roughness is the function of water depth, and is given by the formula above.

2.1.4 Treatment of the storage zones

The storage zones¹ act on the flow by abstracting or returning some water, but this water is not included in the energy balance. Consequently, in the continuity equation the wetted section is the total section (including the storage area), while in the dynamic equation the wetted section is only the area of the active bed (low flow and high flow only). In the flow equations, the wetted section often intervenes through its derivative with respect to the elevation i.e. the surface width L . Similarly, it is therefore the total surface width or the active bed width that must be used, depending on the equation considered. Storage areas are ignored in steady flow simulations.

The continuity equation can therefore be written as:

$$\frac{\partial(S + S_s)}{\partial t} + \frac{\partial Q}{\partial x} = q_l \quad (2.40)$$

which is equivalent to:

$$(L + L_s) \frac{\partial Z}{\partial t} + \frac{\partial Q}{\partial x} = q_l \quad (2.41)$$

and we finally have:

$$L \frac{\partial Z}{\partial t} + \frac{\partial Q}{\partial x} = q_l - L_s \frac{\partial Z}{\partial t} = q_l + q_s \quad (2.42)$$

the index s indicating the variables relating to storage.

The storage is therefore equivalent to introducing a lateral discharge q_s (assumed perpendicular). The influence of the sinuosity of the bed could be accounted by introducing an additional parameter s in the following form :

$$q_l = -sL_s \frac{\partial Z}{\partial t} \quad (2.43)$$

However this is not done because of the difficulty of accurately evaluating L_s .

It is recalled that the water surface elevation used for the storage zones is identical to that used in the low and high flow channels, unlike with the storage cells (see section 4).

¹Note that these are part of the cross-sections. Storage areas can also be described with storage cells. This modelling approach is described in section 4

2.1.5 Treatment of inflows

We consider a lateral inflow q_l , expressed in $m^2.s^{-1}$ (discharge per unit length).

The flow equations are obtained from the system (2.1)(2.2) by adding q_l to the second member of the continuity equation, and kq_lV_l to the second member of the dynamic equation (indicating respectively the contribution of mass and momentum).

The coefficient k results from the fact that the contribution of momentum depends on the respective axes of the main flow and the inflow; only the velocity in the direction of the main flow needs to be taken into account. Therefore, k is nil for an inflow perpendicular to the axis of the main flow, and k is theoretically equal to 1 for an inflow with the same direction as the main flow.

The standard version of MASCARET imposes $k = 0$, this deals with the most useful case and avoids the user to enter an additional parameter.

This modelling of inflows is therefore indicated for cases of *diffuse* inflows where the inflow momentum is usually negligible compared to that of the main flow. In other cases it is preferable to explicitly create a reach representing the inflow (branched network), which would be equivalent to have $k = 1$.

2.1.6 Treatment of junctions

The equations of Saint-Venant apply only between two sections of the same reach (see also the treatment of the singularities 2.2.5). At each junction or node in a network, it is necessary to use alternative equations. These are derived from the equality of the elevations and the conservation of the discharges at the junction. The method is described below (see figure 2.1) on a simple example (no head losses considered), but can be generalised to a more complex situation.

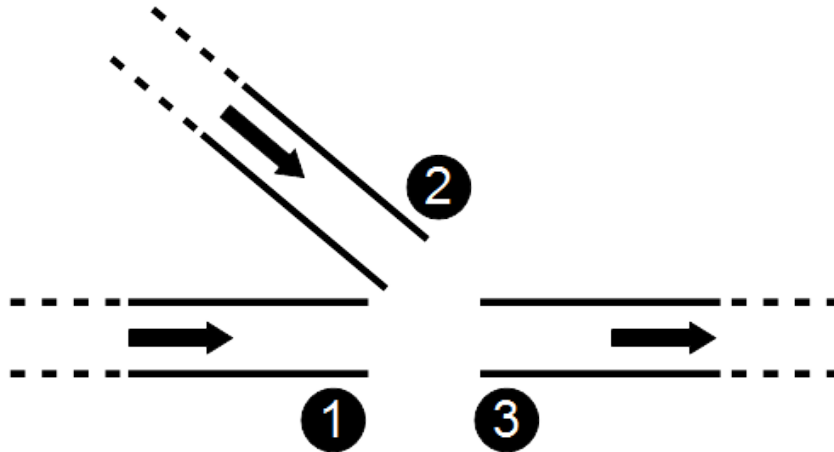


Figure 2.1: Diagram of a confluence with 3 reaches

The junction is bounded by three sections (numbered 1, 2 and 3) located at each end of the associated reaches. These sections are characterized by the hydraulic variables (Q_1, Z_1) , (Q_2, Z_2) et (Q_3, Z_3) .

We can establish the equations used at the junction by considering the following simple case :

- single channel;
- steady flow;
- velocities along the axis of each branch are identical in the 3 considered sections

- no friction in the junction

We can write the Saint-Venant equations along the reach defined by section no. 1 and section no. 3 (the reach corresponding to section no. 2 being treated as lateral contribution - see the previous section 2.1.5)

The continuity equation immediately gives:

$$Q_3 - Q_1 = Q_2 \quad (2.44)$$

The dynamic equation becomes:

$$V \frac{\partial Q}{\partial x} + gS \frac{\partial}{\partial x} \left(Z + \beta \frac{V^2}{2g} \right) = kq_l V_l \quad (2.45)$$

where k is the coefficient of momentum as indicated in section 2.1.5.

The key hypothesis is the conservation of velocities along the axis. It means that:

$$V = kV_l \quad (2.46)$$

and:

$$\frac{\partial}{\partial x} \left(\beta \frac{V^2}{2g} \right) = 0 \quad (2.47)$$

We can also write:

$$\frac{\partial Q}{\partial x} = q_l \quad (2.48)$$

The dynamic equation can therefore be reduced to:

$$\frac{\partial Z}{\partial x} = 0 \quad (2.49)$$

which is equivalent to:

$$Z_3 = Z_1 \quad (2.50)$$

Using a similar reasoning, we consider now the reach defined by sections no. 2 and no. 3, the reach of section no. 1 being treated as inflow, therefore we obtain : $Z_3 = Z_2$.

In summary, the continuity equation gives the conservation of discharges at the junction, and the dynamic equation gives the equality of the water levels.

In a general case, the simple hypotheses used previously are not verified, in particular:

- the velocities in each branch are rarely similar;
- although friction can often be neglected, singular head losses need to be considered (see the following section).

However, the simple relationships obtained here are applied in the subcritical engines. They are the only equations that can be easily implemented in an algorithm.

In practice, the conservation of discharges is always respected, however, the equality of water levels poses a problem. To overcome this difficulty it is usually better to:

- place the sections 1-2-3 as close as possible to the junction (to create a *point* junction);
- introduce singular head losses on the branches upstream of the junction to account for the additional dissipation of energy (see the report on confluents in the bibliography and the following section).

2.1.7 Singular head losses

In the dynamic equation, the term J represents the head losses known as linear, which result from the friction on the bottom and the banks. Localised head losses, known as singular head losses, can be produced in the presence of obstacles, rapid variations of sections or junctions. They are modeled by the mean of a term J_s , in addition to J :

- for a widening : $J_s = \xi_1 \frac{1}{2g} (\beta_j V_j - \beta_i V_i)^2$ with :
 - the indexes j and i denote respectively the upstream section and the downstream section;
 - with: $\beta_j V_j < \beta_i V_i$.
- for an obstacle placed immediately downstream of the section j :
 $J_s = \xi_2 \frac{1}{2g} \beta_j V_j^2$.

where ξ_1 and ξ_2 are the head losses coefficients.

In the current version of MASCARET, the value of ξ_1 is a fixed constant equal to 0.3 (based on the litterature), this is valid for gradual widenings (these are automatically taken into account).

The value of ξ_2 is set by the user and would often be chosen via a calibration. A singular head loss modelled with ξ_2 needs to be introduced whenever the head loss does not result from a progressive widening² (sudden narrowing or widening like bridges, etc.).

2.1.8 Singularities

General principle

The most common singularities are weirs or control dams. However, we will treat this problem more generally by referring to a singularity for any section of the river where the equations of Saint-Venant cannot be applied.

Therefore, in place of the Saint-Venant equations, it is necessary to use specific equations (transfer relationships) that define the influence of the singularity in order to carryout the water line calculations.

We assume that the singularity is located between two computational sections of indices i and j with : $i = j + 1$ (see figure 2.2).

²This head loss is to be placed at the section $i = j + 1$: in subcritical flow, with the downstream influencing the upstream, a head loss placed at the section i will increase the head calculated at the section $j = i - 1$, and not reduce the head calculated at the section i .

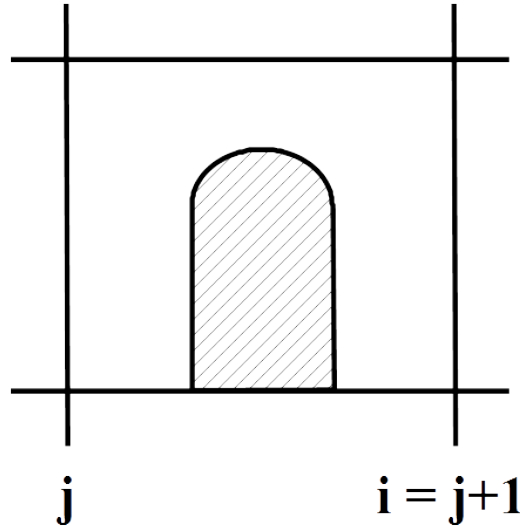


Figure 2.2: Diagram of a singularity

The continuity equation is reduced to $Q_i = Q_j$, expressing the equality of discharge on each side of the singularity. In unsteady flow, this introduces a slight discrepancy, the volume of water being no longer rigorously conserved. However, this bias is negligible if the distance separating the two calculation sections is *reasonably small*.

The dynamic equation is specific to each type of singularity. Its general expression is : $f(Q, Z_{us}, Z_{ds}) = 0$. The function f is only used in its discretised form in the solution algorithm (see section 2.2). The equations used for each type of singularity are indicated in the following sections. A summary of the different singularities available in MASCARET is shown in section 3.4.6.

Weirs

In the most general form, the equation for weirs (see the figures hereafter) is of the form : $Q = f(Z_{us}, Z_{ds})$, the type of flow (drowned or free flow) is directly integrated in the expression of the function f . f is given in a discretised form with a sufficient series of triplets (Q, Z_{us}, Z_{ds}) .

- Free flow weir : the discharge Q_d depends only on the upstream water level (see figure 2.3);
with $Q = Q_d = mL\sqrt{2g}(Z_{us} - Z_0)^{3/2}$ and :
 - m : the discharge coefficient;
 - L : the width of the weir.
- Drowned weir : the discharge Q_n is influenced by the downstream water level (see figure 2.4);
with $Q = Q_n = CQ_d$ and :
 - C : the drowned/free flow coefficient : $C = k \frac{Z_{ds} - Z_0}{Z_{us} - Z_0}$;
 - k : a function to be defined.

The discharge law for the weir can also be defined only for free flow conditions (the downstream level does not influence the upstream level), in the form : $Q = f(Z_{us})$, i.e. by indicating a series of pairs (Q, Z_{us}) .

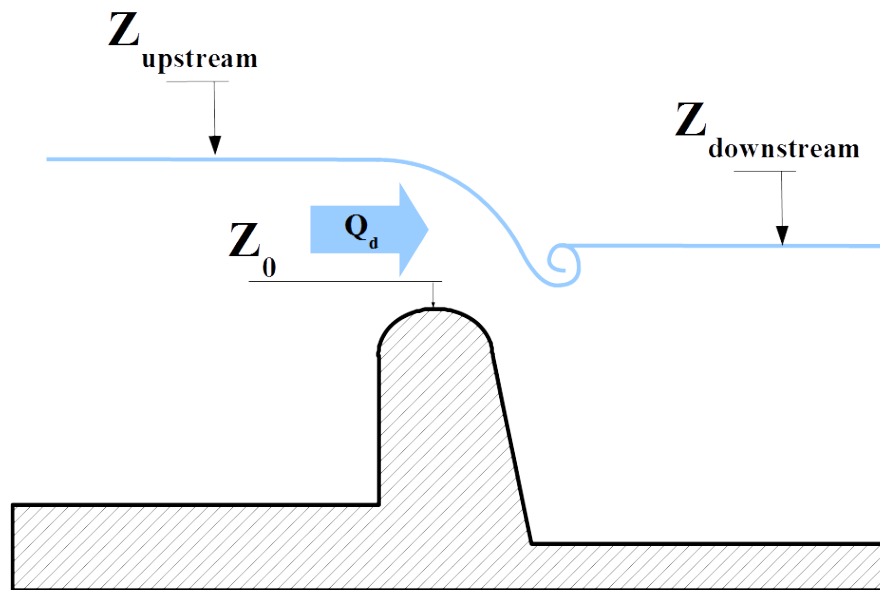


Figure 2.3: Free flow weir

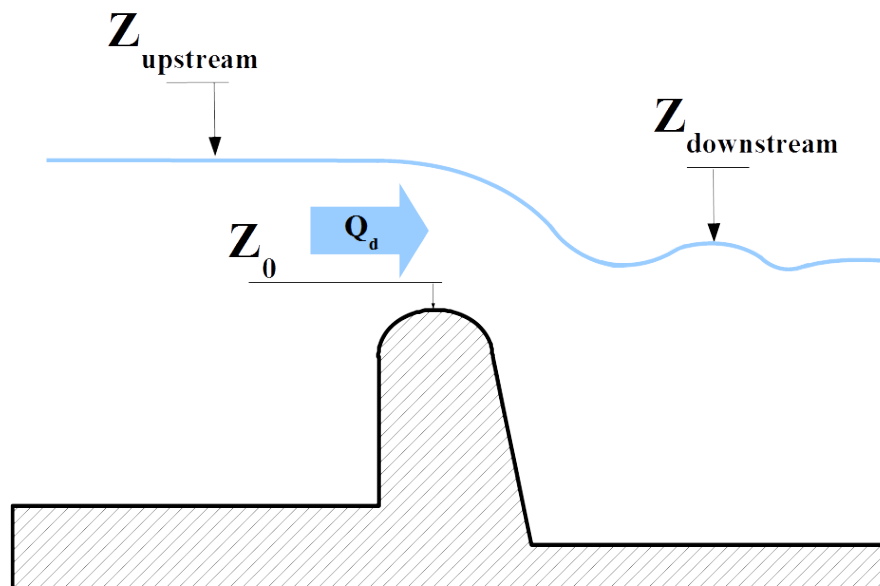


Figure 2.4: Drowned Weir

The flow regime is considered as drowned when : $R = \frac{Z_{ds} - Z_0}{Z_{us} - Z_0}$ is greater than a threshold value R_0 (the crest elevation Z_0 is a parameter of the weir). An automatic correction is therefore applied : $Q = k(R)f(Z_{us})$. In MASCARET the value of the constant R_0 , and the form of the function k , are imposed.

In the current version of MASCARET, two types of drowned weirs are distinguished with one relation for sharp-crested weirs and another one for broad-crested weirs (see section 3.4.5). For a broad-crested weir, R_0 is equal to 0.8, and k is a parabolic function defined by :

$$k(R) = -25R^2 + 40R - 15 \quad (2.51)$$

with :

- $k(R_0) = 1$;
- and $k(1) = 0$.

The weir can also be defined by the geometry of the crest, in this case the relation applied for free flow is a spill equation :

$$Q = m\sqrt{2g} \sum_k L_k (Z_{us} - Z_k)^{3/2} \quad (2.52)$$

where :

- L_k is the length of an element of the crest with an elevation Z_k ;
- m is the discharge coefficient (a parameter of the weir).

In drowned flow, the discharge correction is identical to that defined above for a broad-crested weir.

Control Structures

The equation for the singularity is simply the imposed upstream water level (varying with time).

Control Sections

The equation for the singularity is a generic relation of the type : $Q = f(Z_{us})$.

A relation : $Q = f(Z_{ds})$ can also be specified but it exists only to enable some specific tests because it does not make physical sense in a subcritical flow regime.

2.1.9 Surcharged flow

Sections where the flow is surcharged (e.g. under a bridge) do not require a specific treatment in the code. This is because the cross-sections incorporate a vertical slot with negligible width (the Preissmann slot) at the top of the section. When the cross-section is surcharged, the calculated water level Z is higher than the soffit (top) level of the section, and the pressure P is calculated by means of the relationship : $P = \rho gZ$.

2.2 Solution methods

2.2.1 Steady flow in a reach

We consider the typical case of a compound bed.

As we have noted previously, storage cells can be defined but they are not taken into account in the calculation. In this section we consider the case of a single reach; the computation of branched networks is possible for steady subcritical flow and is considered in section 2.2.3.

Simplification of the equations

The continuity equation is reduced to : $\frac{\partial Q}{\partial x} = q_l$, the discharge is constant along the river, except at the inflows.

The dynamic equation, also called the equation of the water line, takes the form :

$$\frac{\partial \beta Q V}{x} + g S \left(\frac{Z}{x} + J + J_s \right) \quad (2.53)$$

It can also be written as :

$$\frac{\beta V}{g S} \frac{\partial Q}{\partial x} + \frac{V^2}{2g} \left(\frac{\partial \beta}{\partial x} \right) + \frac{\partial}{\partial x} \left(\beta \frac{V^2}{2g} + Z \right) + J + J_s = 0 \quad (2.54)$$

or using the continuity equation :

$$\frac{\beta V}{g S} q_l + \frac{V^2}{2g} \left(\frac{\partial \beta}{\partial x} \right) + \frac{\partial}{\partial x} \left(\beta \frac{V^2}{2g} + Z \right) + J + J_s = 0 \quad (2.55)$$

This equation is discretised along the axis of flow : the discretisation step Δx (separating two consecutive calculation sections) is variable. Its order of magnitude is the width of the river, but in practice it will be related to variability in the sections geometry.

Solving principle

First of all, the continuity equation is solved from the upstream (where the discharge is imposed by the boundary condition) to the downstream, by adding or subtracting the inflows and outflows. Then the calculation of the water level is made, this time going step-by-step from downstream to upstream. For the calculation of a step, we make use of the upstream discharge Q_1 and the downstream elevation Z_2 , and we seek the value of the upstream elevation Z_1 (see figure 2.5).

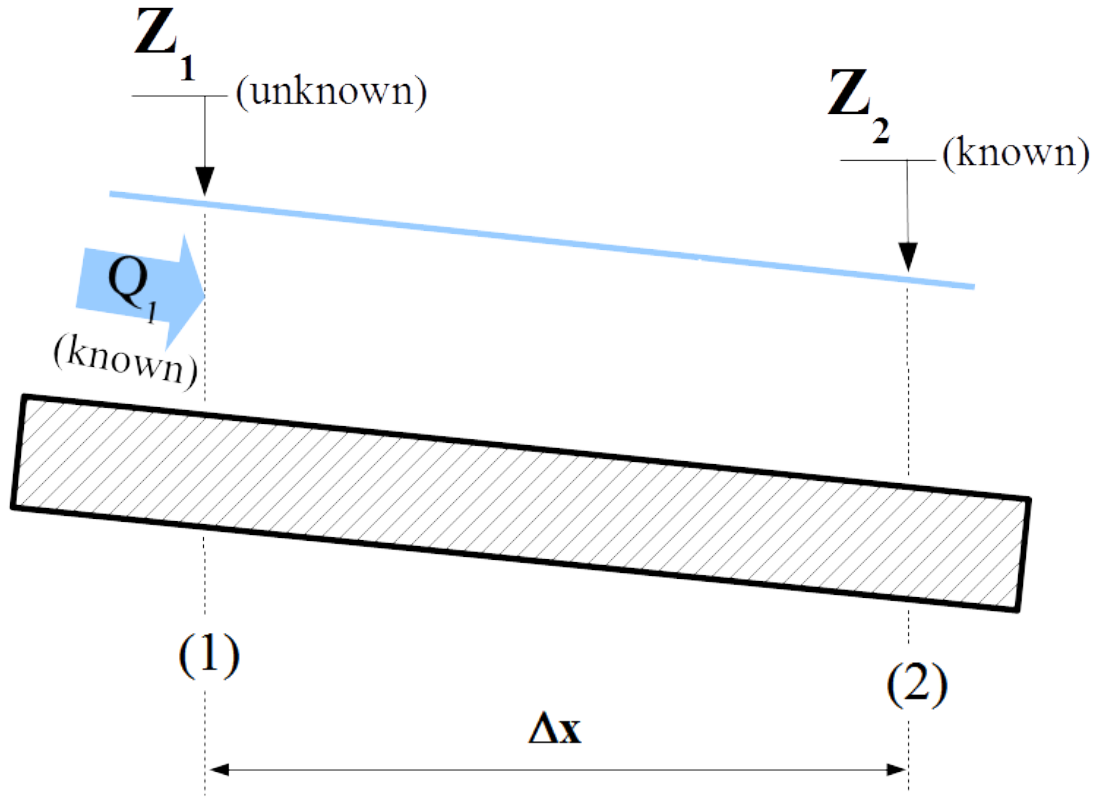


Figure 2.5: Principle of the steady flow solution

The equation of the water line is discretised between sections 1 and 2 in the following form :

$$\begin{aligned} \frac{q_l}{2g} \left(\beta_1 \frac{V_1}{S_1} + \beta_2 \frac{V_2}{S_2} \right) + \frac{V_1^2 + V_2^2}{4g} \frac{\beta_2 - \beta_1}{\Delta x} + \frac{Z_2 - Z_1}{\Delta x} + \\ \frac{1}{\Delta x} \left(\beta_2 \frac{V_2^2}{2g} - \beta_1 \frac{V_1^2}{2g} \right) + \frac{2}{\frac{1}{J_1} + \frac{1}{J_2}} + \\ \frac{\zeta_1}{\Delta x} \frac{(\beta_1 V_1 - \beta_2 V_2)^2}{2g} + \frac{\zeta_2}{\Delta x} \beta_1 \frac{V_1^2}{2g} = 0 \end{aligned} \quad (2.56)$$

It can also be written :

$$Z_1 = Z_2 + JDX + JSDX + DXBETA + DXQ + DXV \quad (2.57)$$

where :

$$JDX = \frac{2}{\frac{1}{J_1} + \frac{1}{J_2}} \Delta x \quad (2.58)$$

$$JSDX = \frac{1}{2g} \left(\zeta_1 (B_1 V_1 - B_2 V_2)^2 + \zeta_2 B_1 V_1^2 \right) \quad (2.59)$$

$$DXBETA = \frac{1}{2g} (\beta_2 - \beta_1) \left(\frac{V_1^2 + V_2^2}{2} \right) \quad (2.60)$$

$$DXQ = \frac{1}{2g}(Q_2 - Q_1) \left(\frac{\beta_1 V_1}{S_1} + \frac{\beta_2 V_2}{S_2} \right) \quad (2.61)$$

and :

$$DXV = \frac{1}{2g}(\beta_2 V_2^2 - \beta_1 V_1^2) \quad (2.62)$$

which is equivalent to :

$$Z_1 = f(Z_1) \quad (2.63)$$

We must therefore solve the equation (2.63) with the unknown Z_1 knowing that there are *a priori* several solutions ; we seek the subcritical solution.

The method involves determining the roots of the function : $\varepsilon = Z_1 - f(Z_1)$.

We start with the approximated solution : $Z_1(0) = Z_2 + J_2 \Delta x$.

We calculate two other approximated solutions with the help of (2.63) :

$Z_1(1) = f(Z_1(0))$ hence : $\varepsilon_0 = Z_1(0) - Z_1(1)$

$Z_1(2) = f(Z_1(1))$ hence : $\varepsilon_1 = Z_1(1) - Z_1(2)$

We therefore know two points, A et B, of the function ε (see figure 2.6) :

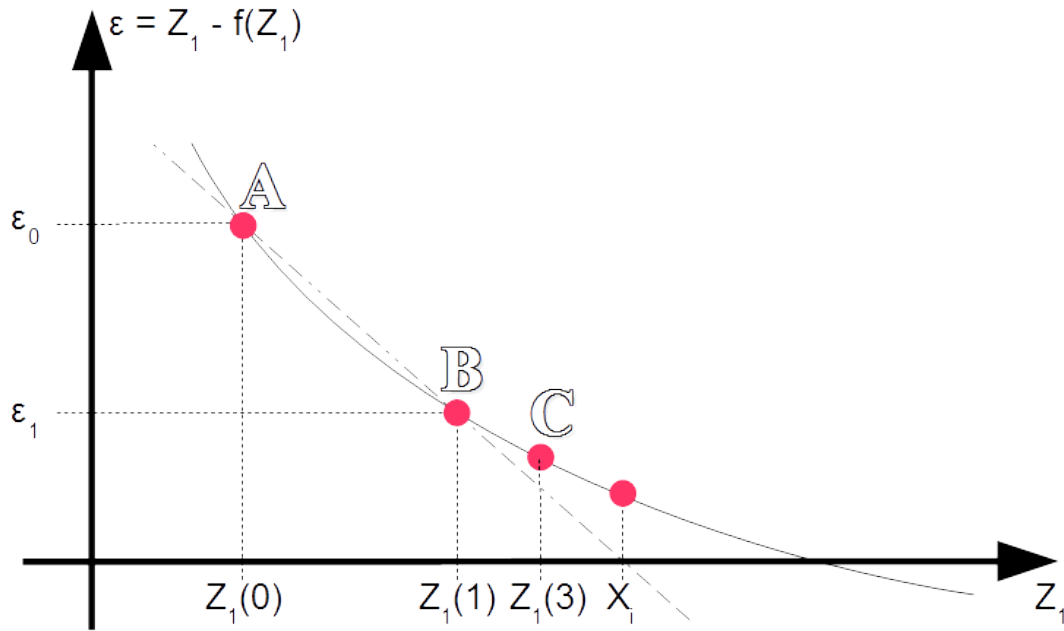


Figure 2.6: Calculation principle for the approximate solution

Knowing these two points, we can use the *secant method* to calculate another approximated solution X_i . Experience shows that it is not desirable to use this value, but rather the half-sum of the solutions closest to the axis of Z_1 , i.e. in figure 2.6 : $Z_1(3) = \frac{X_i + Z_1(1)}{2}$.

We then calculate $\varepsilon(Z_1(3))$, which will give another point on the graph $\varepsilon(Z_1)$, the point C. The process resumes from the two points closest to the axis of Z_1 (in our case : B et C).

We therefore proceed in an iterative manner.

In the case where $\varepsilon(Z_1)$ always keeps the same sign, we stop the calculation when the criteria $\varepsilon(Z) \leq \Delta$ is satisfied by : $\Delta = 10^{-6}$.

In the case where $\varepsilon(Z)$ changes sign, we initially eliminate, from the two points that are on the same side of the axis of the Z_1 , the point that is furthest away from it. We then determine a new approximation from the half-sums of the remaining points.

The calculation is repeated until the stopping criterion is satisfied.

Comment on the solution obtained :

- in general, the calculation converges within 10 iterations;
- the solution obtained is usually subcritical; however, the following paragraph explains the procedure adopted when this is not the case.

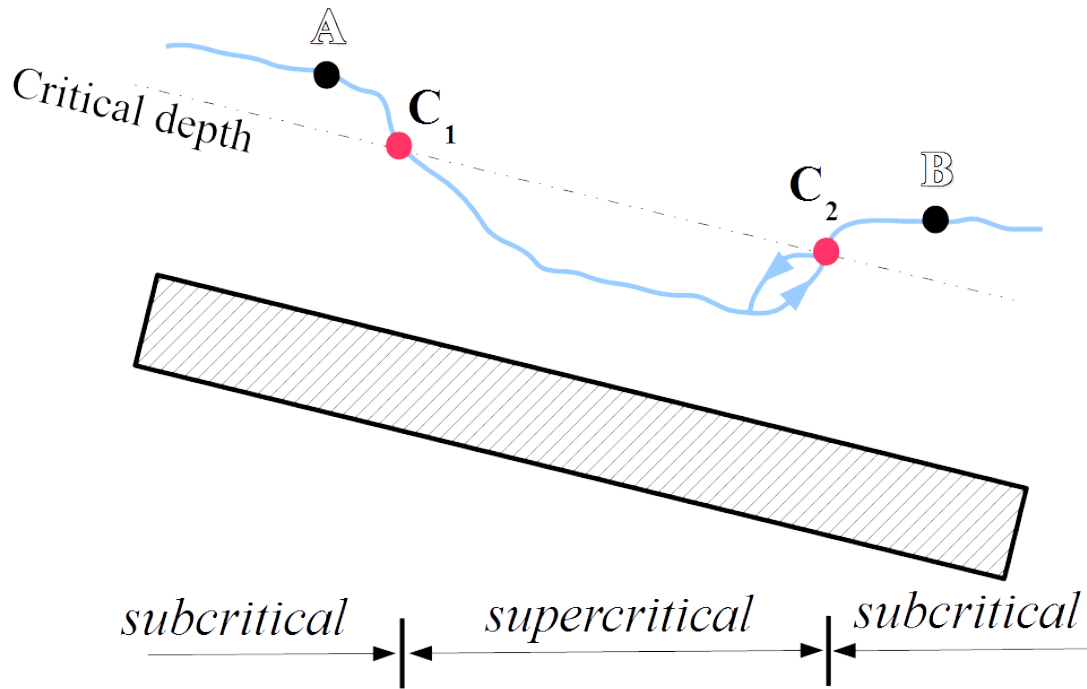


Figure 2.7: Diagram of a local passage in supercritical flow

Solving of supercritical zones

When scanning up and down to calculate the subcritical zones, the supercritical zones are also detected (Figure 2.7). In this case the domain will be swept from upstream to downstream and, for each supercritical zone, a finite-differences discretisation scheme will be applied.

The continuity equation has already been solved at the time of sweeping downstream to upstream. The discharge is known in all points of the discretisation. The equation of the water line, discretised between the section 1 and 2, is similar to equation (2.56). As opposed to the treatment of subcritical zones, Z_1 is known and we look for Z_2 .

It can also be written :

$$Z_2 = Z_1 - JDX - JSDX - DXBETA - DXQ - DXV \quad (2.64)$$

As previously, we eventually have to solve the equation :

$$Z_2 = G(Z_2) \quad (2.65)$$

We must therefore solve the equation (2.65) of unknown Z_2 admitting *a priori* several solutions ; we look for the supercritical solution.

The method involves determining the roots of the function : $\varepsilon = Z_2 - f(Z_2)$. To do that, we do not use the tangent method because the convergence to the supercritical solution would not be assured. The method used is a dichotomy method from the critical level calculated at each point during the calculation of the subcritical zones. The step of the dichotomy is the lesser of :

- a *one – hundredth* of the difference between the elevation of the bottom and the critical water level
- and $0.01m$.

The solution is supercritical by construction.

Validity of the hydraulic jump using impulse functions

Once the supercritical zone has been covered, we check that the hydraulic jump is possible from an energy point of view by considering the direction of the increase of the impulse function at the hydraulic jump. It is reminded that impulse functions are maintained through the passage of the hydraulic jump.

$$F_{imp} = \frac{Q^2}{S} + gSh_G \quad \text{with : } h_G = \frac{1}{S} \int_0^h (h-y)dy \quad (2.66)$$

If the impulse function at the upstream of the hydraulic jump is higher than the impulse function downstream of the hydraulic jump, the hydraulic jump is moved backward and the flow remains supercritical.

2.2.2 Unsteady flow in a reach

Principle

We consider the most general case of a compound channel consisting of : low flow channel, high flow channel and storage areas. We consider first the resolution for an unsteady flow for a single reach; the case of branched and meshed networks (possible with the unsteady flow engine) is considered in section 2.2.4.

We recall the equations governing the water line :

- Continuity equation :

$$\frac{\partial Q}{\partial x} + L_t \frac{\partial Z}{\partial t} = q_l \quad (2.67)$$

where L_t is the total surface width, including the width of the low flow and high flow channels and the storage areas.

- Dynamic equation :

$$\frac{\partial Q}{\partial t} + \frac{\partial \beta QV}{\partial x} + gS \left(\frac{\partial Z}{\partial x} + J + J_s \right) = 0 \quad (2.68)$$

This equation (2.68) can also be written by involving the conveyance D :

$$Q|Q| + D^2 \left(\frac{\partial Z}{\partial x} + J_s + \frac{1}{Sg} \left(\frac{\partial Q}{\partial t} + \frac{\partial \beta QV}{\partial x} \right) \right) = 0 \quad (2.69)$$

These equations are solved with a finite-difference method (see figure 2.8), using an implicit scheme. The discretisation is of the Wendroff type, θ is the coefficient of implicitation. In a subcritical regime, the scheme is stable when : $\theta > 0.5$ (see [13]).

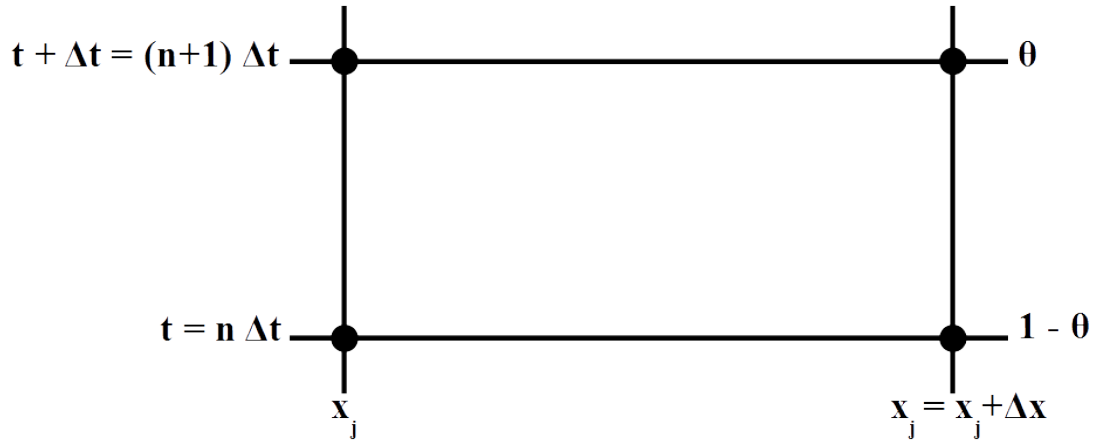


Figure 2.8: Finite differences discretisation

We recall the convention : $i = j + 1$. To be rigorous, Δx should be written as Δx_j , the spatial step varying *a priori* in the considered reach.

The derivatives with respect to x and t are calculated (in the first order) in the following way :

$$\frac{\partial f}{\partial x} = \theta \frac{f_i^{n+1} - f_j^{n+1}}{\Delta x} + (1 - \theta) \frac{f_i^n - f_j^n}{\Delta x} \quad (2.70)$$

$$\frac{\partial f}{\partial t} = \frac{1}{2} \left(\frac{f_i^{n+1} - f_i^n}{\Delta t} + \frac{f_j^{n+1} - f_j^n}{\Delta t} \right) \quad (2.71)$$

The functions that occur without a derivative are evaluated by :

$$f = \theta \frac{f_i^{n+1} + f_j^{n+1}}{2} + (1 - \theta) \frac{f_i^n + f_j^n}{2} \quad (2.72)$$

Discretised equations

The discretised equations are written as follows :

- Continuity equation :

$$\begin{aligned} (1 - \theta) \frac{Q_i^n - Q_j^n}{\Delta x} + \theta \frac{Q_i^{n+1} - Q_j^{n+1}}{\Delta x} + \frac{1}{2} (L_{t_i}^n + L_{t_j}^n) \\ \times \frac{1}{2} \left(\frac{Z_i^{n+1} - Z_i^n}{\Delta t} + \frac{Z_j^{n+1} - Z_j^n}{\Delta t} \right) = q_{l_j}^{n+1} \end{aligned} \quad (2.73)$$

- Dynamic equation :

$$\begin{aligned}
& \left(\frac{1-\theta}{2} \right) (Q_j^{n^2} + Q_i^{n^2}) + \frac{\theta}{2} (Q_j^{n+1^2} + Q_i^{n+1^2}) \\
& + \left(\frac{1-\theta}{2} (D_i^{n^2} + D_j^{n^2}) + \frac{\theta}{2} (D_i^{n+1^2} + D_j^{n+1^2}) \right) \\
& \times \left\{ (1-\theta) \frac{Z_i^n - Z_j^n}{\Delta x} + \theta \frac{Z_i^{n+1} - Z_j^{n+1}}{\Delta x} + J_s(i) + J_s(j) \right. \\
& + \left(\frac{1-\theta}{2g} \left(\left(\frac{1}{S} \right)_i^n + \left(\frac{1}{S} \right)_j^n \right) + \frac{\theta}{2g} \left(\left(\frac{1}{S} \right)_i^{n+1} + \left(\frac{1}{S} \right)_j^{n+1} \right) \right) \\
& \times \left[\frac{1}{2\Delta t} (Q_i^{n+1} - Q_i^n + Q_j^{n+1} - Q_j^n) + \frac{\theta}{\Delta x} \left(\beta_i^n \frac{Q_i^{n+1^2}}{S_i^{n+1}} - \beta_j^n \frac{Q_j^{n+1^2}}{S_j^{n+1}} \right) \right. \\
& \left. \left. + \frac{1-\theta}{\Delta x} \left(\beta_i^n \frac{Q_i^{n^2}}{S_i^n} - \beta_j^n \frac{Q_j^{n^2}}{S_j^n} \right) \right] \right\} = 0 \quad (2.74)
\end{aligned}$$

We introduce the following variables :

$$\begin{cases} \Delta Q_j = Q_j^{n+1} - Q_j^n \\ \Delta Q_i = Q_i^{n+1} - Q_i^n \\ \Delta Z_j = Z_j^{n+1} - Z_j^n \\ \Delta Z_i = Z_i^{n+1} - Z_i^n \end{cases} \quad (2.75)$$

and, after linearisation (i.e. preserving only the first order terms in ΔZ and ΔQ) we get :

- for the continuity equation :

$$\boxed{G\Delta Q_i + H\Delta Z_i = I\Delta Q_j + J\Delta Z_j + K} \quad (2.76)$$

with :

$$G = \theta \quad (2.77)$$

$$I = G \quad (2.78)$$

$$H = \frac{\Delta x}{4\Delta t} (L_{t_j}^n + L_{t_i}^n) \quad (2.79)$$

$$J = -H \quad (2.80)$$

$$K = Q_j^n - Q_i^n + q_{l_i}^{n+1} \quad (2.81)$$

- and for the dynamic equation :

$$\boxed{L\Delta Q_i + M\Delta Z_i = N\Delta Q_j + O\Delta Z_j + P} \quad (2.82)$$

with :

$$L = B_2 + C_1 H_1 (E_2 + F_2) \quad (2.83)$$

$$M = C_1 (D_4 + H_1 F_4 + H_4 F_1) + C_4 (D_1 + H_1 F_1) \quad (2.84)$$

$$N = -B_3 - C_1 H_1 (E_3 + F_3) \quad (2.85)$$

$$O = -C_1 (D_5 + H_1 F_5 + H_5 F_1) - C_5 (D_1 + H_1 F_1) \quad (2.86)$$

$$P = -B_1 - C_1 (D_1 + H_1 F_1) \quad (2.87)$$

knowing that the various terms of the dynamic equation have been linearised in the following way (the non-indexed variables correspond to the time steps n) :

$$Q|Q| = B_1 + B_2\Delta Q_i + B_3\Delta Q_j \quad (2.88)$$

$$B_1 = \frac{1}{2}(Q_i^2 \text{sign}(Q_i) + Q_j^2 \text{sign}(Q_j)) \quad (2.89)$$

$$B_2 = \theta|Q_i| \quad (2.90)$$

$$B_3 = \theta|Q_j| \quad (2.91)$$

*

$$D^2 = C_1 + C_4\Delta Z_i + C_5\Delta Z_j \quad (2.92)$$

$$C_1 = \frac{1}{2}(D_i^2 + D_j^2) \quad (2.93)$$

$$C_4 = \theta D_i(\alpha_{m_i} + \alpha_{M_i}) \quad (2.94)$$

$$C_5 = \theta D_j(\alpha_{m_j} + \alpha_{M_j}) \quad (2.95)$$

$$\alpha_{m_i} = \frac{K_m R_m^{2/3}}{3} \left(5L_m - 2R_m \frac{\partial P_m}{\partial z} \right)_i \quad (2.96)$$

$$\alpha_{m_j} = \frac{K_m R_m^{2/3}}{3} \left(5L_m - 2R_m \frac{\partial P_m}{\partial z} \right)_j \quad (2.97)$$

$$\alpha_{M_i} = \frac{K_M R_M^{2/3}}{3} \left(5L_M - 2R_M \frac{\partial P_M}{\partial z} \right)_i \quad (2.98)$$

$$\alpha_{M_j} = \frac{K_M R_M^{2/3}}{3} \left(5L_M - 2R_M \frac{\partial P_M}{\partial z} \right)_j \quad (2.99)$$

$$K'_m = AK_m \quad (2.100)$$

$$K'_M = \sqrt{1 + \frac{S_m}{S_M}(1 - A^2)} K_M \quad (2.101)$$

*

$$\frac{\partial Z}{\partial x} + J_s(i) + J_s(j) = D_1 + D_4\Delta Z_i + D_5\Delta Z_j \quad (2.102)$$

$$D_1 = \frac{Z_i - Z_j}{\Delta x} + J_s(i) + J_s(j) \quad (2.103)$$

$$D_4 = \frac{\theta}{\Delta x} \quad (2.104)$$

$$D_5 = \frac{\theta}{\Delta x} \quad (2.105)$$

*

$$\frac{\partial Q}{\partial t} = E_2\Delta Q_i + E_3\Delta Q_j \quad (2.106)$$

$$E_2 = \frac{1}{2\Delta t} \quad (2.107)$$

$$E_3 = \frac{1}{2\Delta t} \quad (2.108)$$

*

$$\frac{\partial}{\partial x} \left(\beta \frac{Q^2}{S} \right) = F_1 + F_2 \Delta Q_i + F_3 \Delta Q_j + F_4 \Delta Z_i + F_5 \Delta Z_j \quad (2.109)$$

$$F_1 = \frac{1}{\Delta x} (\beta_i V_i Q_i - \beta_j V_j Q_j) \quad (2.110)$$

$$F_2 = \frac{2\theta}{\Delta x} \beta_i V_i \quad (2.111)$$

$$F_3 = -\frac{2\theta}{\Delta x} \beta_j V_j \quad (2.112)$$

$$F_4 = -\frac{\theta}{\Delta x} \beta_i V_i^2 L_{t_i} \quad (2.113)$$

$$F_5 = \frac{\theta}{\Delta x} \beta_j V_j^2 L_{t_j} \quad (2.114)$$

*

$$\frac{1}{Sg} = H_1 + H_4 \Delta Z_i + H_5 \Delta Z_j \quad (2.115)$$

$$H_1 = \frac{1}{2g} \left(\frac{1}{S_i} + \frac{1}{S_j} \right) \quad (2.116)$$

$$H_4 = -\frac{\theta}{2g} \frac{L_{t_i}}{S_i^2} \quad (2.117)$$

$$H_5 = -\frac{\theta}{2g} \frac{L_{t_j}}{S_j^2} \quad (2.118)$$

Solving the discretised equations

The equations (2.76) and (2.82) lead directly to the establishment of a matrix system :

$$\begin{pmatrix} G & H \\ L & M \end{pmatrix} \begin{pmatrix} \Delta Q_i \\ \Delta Z_i \end{pmatrix} = \begin{pmatrix} I & J \\ N & O \end{pmatrix} \begin{pmatrix} \Delta Q_j \\ \Delta Z_j \end{pmatrix} + \begin{pmatrix} K \\ P \end{pmatrix} \quad (2.119)$$

where $G, H, I, J, K, L, M, N, O$ et P are macro-coefficients, their expression depend on the control variables, the state variables and the parameters of the model (2.77-2.81) (2.83-2.87).

In the case of a single reach, the boundary conditions at the upstream and downstream ends mean that ΔQ_1 and ΔZ_n are no longer unknowns, n being the number of calculation sections. This corresponds to an imposed discharge at the upstream end and an imposed water level at the downstream end. The system (2.119) therefore comprises $2n - 2$ equations and as many unknowns. It is expressed in the general form :

$$A.x = b \quad (2.120)$$

where :

- the vector x of the unknowns, is the solution of the variations of elevation and discharge at each one of the n calculation sections in the network of reaches :

$$x = \begin{pmatrix} \Delta Z_1 \\ \Delta Q_2 \\ \Delta Z_2 \\ \Delta Q_3 \\ \Delta Z_3 \\ \vdots \\ \Delta Z_{n-1} \\ \Delta Q_n \end{pmatrix} \quad (2.121)$$

- in the simple case of a single reach, the matrix A is an asymmetric band matrix :

$$A = \begin{pmatrix} -J & G & H & & & & & & & \\ -O & L & M & & & & & & & \\ & -I & -J & G & H & & & & & \\ & -N & -O & L & M & & & 0 & & \\ & & & -I & -J & G & H & & & \\ & & & -N & -O & L & M & & & \\ & & & & & & \dots & & & \\ & & & & & & & \dots & & \\ & 0 & & & & & & & \dots & \\ & & & & & & & & & \dots \\ & & & & & & & & & -I & -J & G \\ & & & & & & & & & -N & -O & L \end{pmatrix} \quad (2.122)$$

- b is the vector of the second member of the equation :

$$b = \begin{pmatrix} K \\ P \\ K \\ P \\ \vdots \\ K \\ P \end{pmatrix} \quad (2.123)$$

The various macro-coefficients $G, H, I, J, K, L, M, N, O$ et P are calculated at each time step. The matrix A presents two important characteristics : it is a diagonally-dominant band matrix and is very sparse. It is then possible to use a specialised solver to solve (2.120). Two linear solvers are implemented in the code REZO.

- the first one is the band solver DGBSV of the LAPACK³ numerical library. This solver is the default one when the problem deals with only one reach with no storage cell;
- the second is a sparse solver written in
 - the call of a single routine and a reduced number of sources;
 - the internal management of the renumbering to preserve the sparseness (a choice of the pivot, with a criterion of fill minimization);
 - the matrix chaining is filled using the coordinates format (i, j) for each non-zero element that is present in the line i and column j .

The solving of a sub-critical and unsteady problem by the REZO code follows the algorithmic stages described in Figure 2.9.

³<http://www.netlib.org/lapack/>

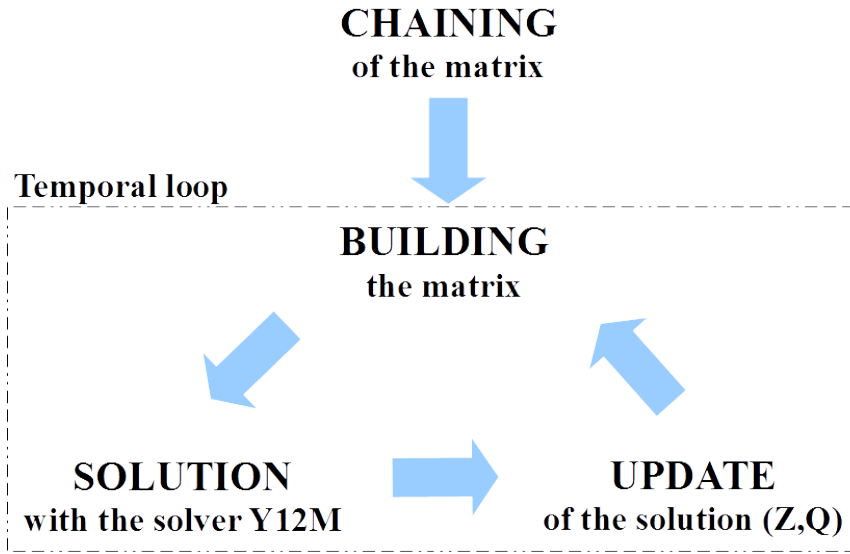


Figure 2.9: REZO algorithm

Before starting the time-based simulation, the chaining of the matrix must be determined, starting with the schematic of the network. This calculation is carried out only once because the connectivity of the network does not change with time.

The time loop consists of three phases :

- a calculation of the macro-coefficients $G, H, I, J, K, L, M, N, O$ and P and their assembly in the matrix A and the second member of the equation b ;
- the use of the direct solver Y12M or DGBSV for solving the system (2.120) and determine the solution of the elevation and discharge variations $(\Delta Z_i, \Delta Q_i)$ in each section i ;
- an update of the water line for the current time t_k :

$$\forall i \in 1 \dots n \begin{cases} Q_i^k = Q_i^{k-1} + \Delta Q_i \\ Z_i^k = Z_i^{k-1} + \Delta Z_i \end{cases} \quad (2.124)$$

2.2.3 Treatment of junctions in a steady subcritical regime

Reminder

The equations solved at the junctions are :

- the continuity of discharges;
- the equality of elevations in each branch (see the justification in section 2.1.6).

Therefore a junction does not add any additional unknown : the unknowns to be calculated are the unknowns (Z, Q) of the computational sections forming this junction. Therefore, in the example below (figure 2.10):

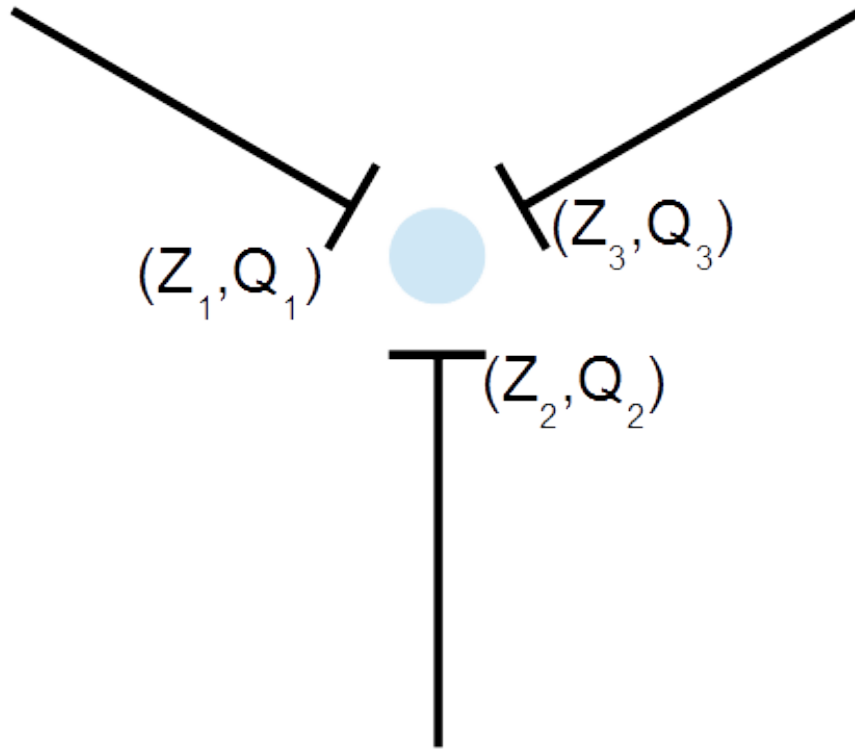


Figure 2.10: Example of a junction with three branches

The unknowns are (Z_1, Q_1) , (Z_2, Q_2) and (Z_3, Q_3) and the equations of the junction are :

- the distribution of discharges : $Q_1 + Q_2 + Q_3 = 0$;
- the equality of the water levels : $Z_1 = Z_2 = Z_3$.

We can easily verify that these relationships are sufficient to determine the values of (Z, Q) throughout the considered network.

If we call :

- the total number of computational sections to be processed (i.e. the sum of the calculation sections in all of the branches) : im ;
- the number of reaches : $nbreach$;
- the number of free ends : $nblimi$.

then for the whole network we can write $(im - nbreach)$ continuity equations and as many discrete dynamic equations of Saint-Venant.

The unknowns are the $(\Delta Q, \Delta Z)$ of the computation sections, and so there are $2im$ unknowns. The boundary conditions are known at the extremities, which gives $nblimi$ additional relationships.

For all of the nodes we can write :

- * $nbnode$ discharge conservation equations;
- * $(2nbreach - nblimi) - nbnode$ equal elevation equations.

This is because $(2nbreach - nblimi)$ represents the number of branch extremities connecting to a junction (each reach should be counted twice except those that have a free end). At each junction, if k branches are connected, there are only $k - 1$ equations that come from the equality of elevations. It is therefore necessary to subtract $nbnode$ from $2nbreach - nblimi$.

Finally, we have :

$2(im - nbreach) + nblimi + nbnode + (2nbreach - nblimi) - nbnode = 2im$ equations for $2im$ unknowns.

Principle

The algorithm presented here is valid regardless of the network (branched or meshed) provided the following conditions are met :

- the direction of flow is known *a priori* in each reach;
- for all extremities, the boundary condition is:
 - the discharge, if it is an upstream end;
 - the elevation, if it is a downstream end.

In practice, there is a risk that this algorithm becomes very time consuming if the network is complex (i.e. interconnected distributaries, see below). In that case, it would be better to use the unsteady subcritical engine (see the following section), starting with a simple initial water line, and choosing the boundary conditions so as to converge towards a stationary state corresponding to the water line being sought.

Before presenting the algorithm, it is useful to make the following observation. The problem is very simple if the network is only comprised of confluences because the discharge is calculated at every point of the domain descending from the upstream ends ; therefore it suffices to calculate the elevation of the water line by following the reverse path (i.e. gradually moving towards the upstream ends), using the same procedure as for a reach. At each node, the equality of the elevations can be imposed without difficulty.

The critical problem is therefore to calculate the distribution of discharge at a junction. The solution used here is an iterative method that is explained in the following paragraph. This means that the general algorithm is itself iterative : the iterations are interlinked in the same way as the distributaries (see figures 2.11 and 2.12). We can further specify that :

- *to descend a reach* means to calculate the discharge at each successive computational section, working from upstream to downstream;
- *to ascend a reach* means to calculate the elevation at each successive computational section, working from downstream to upstream.

Figure 2.11 presents the solution algorithm, while figure 2.12 illustrates the example of a meshed network including two interlinked distributaries. This example could be considered as representative of the more complex cases that can be computed within a reasonable time by this algorithm. Beyond that, as indicated previously, it is recommended that the REZO code for unsteady flows is used.

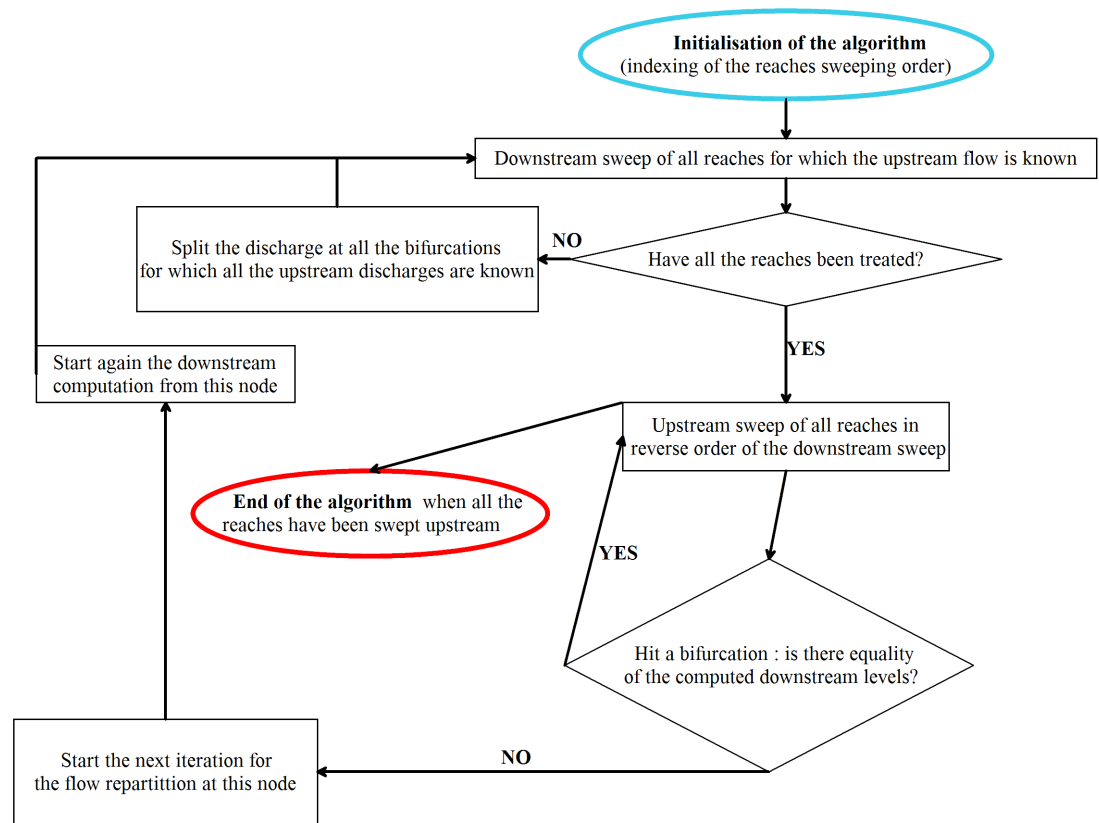


Figure 2.11: Solution algorithm for a steady state network

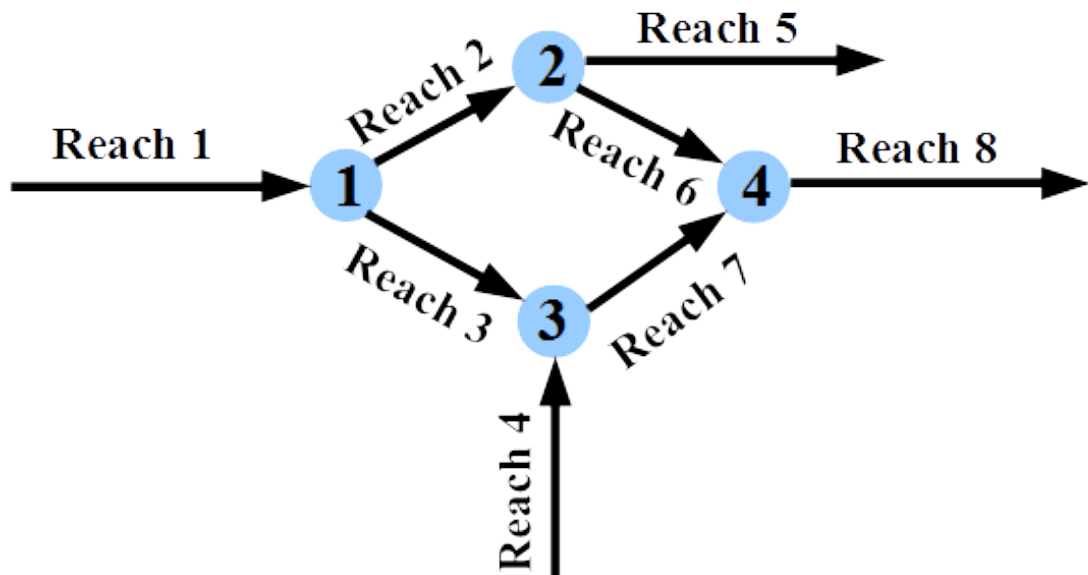


Figure 2.12: Example of a network for a steady state calculation

ALGORITHM (SEE FIGURE 2.12) :

- * Descend Reach 1
- * Descend Reach 4
- * $n = 1$
- * (1) Spread the discharge at node 1 (iteration n)
 - Descend Reach 2
 - Descend Reach 3
 - Descend Reach 7
 - $p = 1$
 - (2) Spread the discharge at node 2 (iteration $n.p$)
 - * Descend Reach 5
 - * Descend Reach 6
 - * Descend Reach 8

 - * Ascend Reach 8
 - * Ascend Reach 6
 - * Ascend Reach 5
 - * Equality of the downstream elevations at node 2?
 - if no : $p = p + 1$, go back to step (2)
 - if yes : continue
 - Ascend Reach 7
 - Ascend Reach 3
 - Ascend Reach 2
 - Equality of the downstream elevations at node 1?
 - * if no : $n = n + 1$, go back to step (1)
 - * if yes : continue
- Ascend Reach 4
- Ascend Reach 1

Discharge distribution in a diffluence (bifurcation)

First Iteration

The discharge distribution needs to verify :

$$Q_{us} + \sum_{i=1}^{n_{ds}} Q_i \quad (2.125)$$

where n_{ds} is the number of downstream branches.

$$Q_i = D_i J_i^{1/2} \quad (2.126)$$

The solution used here is the equality of the slopes of the total head line :

$$J_i = J = \left(\frac{Q_i^2}{D_i} \right) \quad (2.127)$$

or :

$$\frac{Q_i}{D_i} = \frac{Q_j}{D_j} = ct = J^{1/2} \quad (2.128)$$

Substituting this into the continuity equations leads to:

$$\frac{Q_{us}}{J^{1/2}} = \sum_{i=1}^{n_{ds}} D_i \quad (2.129)$$

The sum of the conveyances depends only on the geometry and the roughness of the downstream reaches and is a function of the elevation that, in general, increases monotonously.

For a given value of J , it is possible to calculate an elevation satisfying this relationship, and then calculate the D_j and then the Q_j for that elevation.

In the current version of MASCARET, the approach used for the first iteration is to take J as the average of the slopes of the bottom of the downstream branches.

Subsequent iterations

The problem to solve is determining the best corrections of discharge to make (compared to the distribution chosen for the previous iteration), in order to come as close as possible to the equality of elevations at the first sections of the downstream reaches.

The approach is to determine the correction of discharge by assuming the slopes of the energy lines J (equal to those of the preceding iteration) in order to tend towards the equality of the elevations (while still respecting the continuity of the discharges).

For each of the first sections of the downstream branches, we can write (for a given elevation) :

$$Q_i^2 = J_i D_i^2 \quad (2.130)$$

or :

$$Q_i = J_i^{1/2} D_i \quad (2.131)$$

the index i corresponds to the number of the downstream branch.

By differentiating between the iterations n and $n+1$, we get :

$$\delta Q_i = (J_i^{1/2})_n \left(\frac{\partial D_i}{\partial Z_i} \right)_n \delta Z_i \quad (2.132)$$

where n denotes the number of the current iteration,

with :

$$(J_i^{1/2})_n = \left(\frac{Q_i}{D_i} \right)_n \quad (2.133)$$

where : $\delta Q_i = Q_i^{n+1} - Q_i^n$ and : $\delta Z_i = Z_i^{n+1} - Z_i^n$

If n_{ds} is the number of downstream branches then the continuity of the discharges is written :

$$Q_{us} = \sum_{i=1}^{n_{ds}} Q_i \quad (2.134)$$

where :

$$\sum_{i=1}^{n_{ds}} \delta Q_i = 0 \quad (2.135)$$

The correction of discharge is done to tend towards the equality of the elevations at the junction.

We therefore want :

$$Z_i^{n+1} = ct = Z \quad (2.136)$$

which gives :

$$\delta Z_i = Z - Z_i^n \quad (2.137)$$

where the Z_i^n are the results of the previous iteration.

So we have :

- n_{ds} relationships (2.132);
- 1 relationship (2.135);
- n_{ds} relationships (2.137).

and so there are $2n_{ds} + 1$ relationships for $2n_{ds} + 1$ unknowns ($\delta Q_i, \delta Z_i$ and Z).

$$\begin{cases} \delta Z_i = Z - Z_i^n \\ \sum_{i=1}^{n_{ds}} \delta Q_i = 0 \\ \delta Q_i = \left(\frac{Q_i}{D_i} \right)_n \left(\frac{\partial D_i}{\partial Z_i} \right)_n \delta Z_i \end{cases} \quad (2.138)$$

from which it can be derived :

$$Z = \frac{\sum_{i=1}^{n_{ds}} \left(\frac{Q_i}{D_i} \right)_n \left(\frac{\partial D_i}{\partial Z_i} \right)_n Z_i}{\sum_{i=1}^{n_{ds}} \left(\frac{Q_i}{D_i} \right)_n \left(\frac{\partial D_i}{\partial Z_i} \right)_n} \quad (2.139)$$

then :

$$\delta Z_i = Z - Z_i^n \quad (2.140)$$

and finally :

$$\delta Q_i = \sum_{i=1}^{n_{ds}} \left(\frac{Q_i}{D_i} \right)_n \left(\frac{\partial D_i}{\partial Z_i} \right)_n \delta Z_i \quad (2.141)$$

2.2.4 Computation of junctions in unsteady subcritical flow

The matrix A (2.122) describes a solution to the 1D Saint-Venant equations in the case of a single reach. If several reaches exist, the representation of the confluents or diffuents requires additional equations. There are no dynamic equations (2.2) or continuity equations (2.1) between the sections relating to the same junction (Figure 2.1). For each subcritical junction of the network, it is necessary to take into account the two following constraints :

- equality of the elevations of each section of the reach extremities that are connected to the same node. In the case of a node of three branches (Figure 2.1), it is necessary to add two equations involving the equality of the elevations from one time step to the next :

$$\begin{cases} Z_1 + \Delta Z_1 = Z_2 + \Delta Z_2 \\ Z_1 + \Delta Z_1 = Z_3 + \Delta Z_3 \end{cases} \quad (2.142)$$

which is rewritten as :

$$\begin{cases} \Delta Z_1 - \Delta Z_2 = Z_2 - Z_1 \\ \Delta Z_1 - \Delta Z_3 = Z_3 - Z_1 \end{cases} \quad (2.143)$$

It is therefore necessary to add to the matrix A (2.122) two rows of coefficients $+1$ and -1 on the level variations. The second member vector b (2.123) is supplemented with two differences of elevation.

- conservation of the discharges, the total volume of water arriving in a node is equal to the volume of all the water leaving the node :

$$Q_1 + \Delta Q_1 + Q_2 + \Delta Q_2 = Q_3 + \Delta Q_3 \quad (2.144)$$

that is to say :

$$\Delta Q_1 + \Delta Q_2 - \Delta Q_3 = Q_3 - Q_2 - Q_1 \quad (2.145)$$

In the same way as above, the matrix A is supplemented with a line of $+1$ and -1 coefficients. We add to the second member an algebraic sum of the discharges of the attached sections.

An exploration of the network for the solution of the discretised equations is no longer appropriate. The solution algorithm is particularly complex and a linear solver system is used, in this case Y12M [38].

2.2.5 Treatment of the singularities

In section 2.1.8 we have defined the modelling principle for the singularities : they are supposed to be located between two sections of calculation j and $i = j + 1$. The continuity equation is replaced by the equality of the discharges for these sections, and the dynamic equation is replaced by the law of singularity of the following form after being discretised :

$$A_{sing}\Delta Q + B_{sing}\Delta Z_{us} + C_{sing}\Delta Z_{ds} = D_{sing} \quad (2.146)$$

with :

$$\text{si } Q > 0 \quad \begin{cases} Z_{us} = Z_j \\ Z_{ds} = Z_i \end{cases} \quad (2.147)$$

and :

$$\text{si } Q < 0 \quad \begin{cases} Z_{us} = Z_i \\ Z_{ds} = Z_j \end{cases} \quad (2.148)$$

We note that A_{sing} can be zero, but that B_{sing} and C_{sing} cannot be zero at the same time without the system being undefined.

Examples :

- Weir following the general relation : $Q = f(Z_{us}, Z_{ds})$;

$$A_{sing} = -1 \quad ; \quad B_{sing} = + \frac{\partial f}{\partial Z_{us}} \quad ;$$

$$C_{sing} = + \frac{\partial f}{\partial Z_{ds}} \quad ; \quad D_{sing} = 0$$

If this weir is under free flow, then : $C_{sing} = 0$

- Dam for which the retention equation : $Z_{us} = f(t)$ is known *a priori*;

$$A_{sing} = 0 \quad ; \quad B_{sing} = 1 \quad ;$$

$$C_{sing} = 0 \quad ; \quad D_{sing} = f(t + \Delta t) - f(t)$$

Solving for steady flow

We must determine the water elevation upstream of the singularity, knowing the discharge and the elevation downstream. The solution is therefore always very simple. In several cases (e.g. weirs of general law, a regulation, a control section), the discretisation of the law of the singularity gives the result directly. If not, the only difficulty lies with the drowned/free flow coefficient, the value of which supposes that the level upstream of the calculated object is already known. In that case, the upstream elevation is initially calculated by assuming a free flow regime, then, if this is not the case with the solution obtained, the upstream elevation is progressively increased until the free flow regime is verified. Note that for a weir, the head is the height above the weir for the upstream section.

Solution for unsteady flow

The continuity equation (2.76) is reduced to the equality of the discharges upstream and downstream of the singularity, which implies :

$$\begin{cases} H = 0 \\ J = 0 \\ G = 1 \\ I = 1 \end{cases} \quad (2.149)$$

The dynamic equation (2.82) is specific to each type of singularity. Nevertheless, this equation is modified as follows :

$$\begin{cases} L = 0 \\ N = -A_{sing} \\ O = -B_{sing} \\ M = C_{sing} \\ P = D_{sing} \end{cases} \quad (2.150)$$

where the coefficients A_{sing} , B_{sing} , C_{sing} and D_{sing} depend mainly on the nature of the weir and the type of regime (drowned or free flow). They are calculated *a priori* for each time step by external functions and are used for the creation of the matrix of coefficients A (2.122). The discharges calculated over the weirs by solving the equation (2.120) are corrected later by these same functions.

2.2.6 Dealing with the transcritical flows

Introduction

Until the version 7.1.3, flows with a Froude number greater than 1 stop the computation of the unsteady subcritical kernel.

Since the version 7.1.4, critical and supercritical flows can be avoided so that the computation does not stop. This possibility is based on a simplification in the governing momentum equations. This simplification is often used in engineering shallow water flow modelling softwares using a Preissmann scheme.

It makes impossible to reproduce the supercritical flow conditions and wave speed estimates are incorrect for large Froude numbers. Even if the robustness is improved (no computation error) the user must aware of the problem of a simplified model.

This modification is a trick that remains a numerical option since the version 7.1.4 of MAS-CARET

Subcritical modelling

Initially, the Preissmann scheme cannot deal with the shallow water equations when a critical condition occurs because of a ill-posed problem [29].

At the subcritical-supercritical transition, the system is locally underdetermined because of only one forward characteristic on the corresponding cross-section. On the contrary at the supercritical-subcritical transition, the system is overdetermined because of three forward characteristics instead of two.

Some authors have proposed modifications of the Preissmann scheme in order to deal with supercritical flows with a small Froude number [21][35].

These modifications are not straightforward and cannot be competitive in comparison with more modern schemes based on the finite volume approach [36].

Modifying the momentum equation

The goal of the modification (and that only concerns the code REZO) is to avoid to deal with a critical condition when the water flow is speeding up.

Thereby this not exactly the equation (2.2) that is solved with a consistent discretisation scheme but an altered form where the convection term is multiplied by a weighting coefficient α .

$$\frac{\partial}{\partial x} \left(\beta \frac{Q^2}{S} \right) \rightarrow \alpha \times \frac{\partial}{\partial x} \left(\beta \frac{Q^2}{S} \right) \quad (2.151)$$

with : $\alpha \in [0, 1]$

The selection of the value for the weighting coefficient α is based on a heuristic depending on the local Froude number. For small values of the Froude number, α is close to 1 in order to exactly solve the momentum equation.

On the contrary, for a Froude number close to 1, α tends to 0 in order to gradually decrease the speed-up.

The chosen formula est a regular function [19] :

$$\alpha = \max(0, 1 - F_r^2) \quad (2.152)$$

The attenuation of the convection term begins early with this formula.

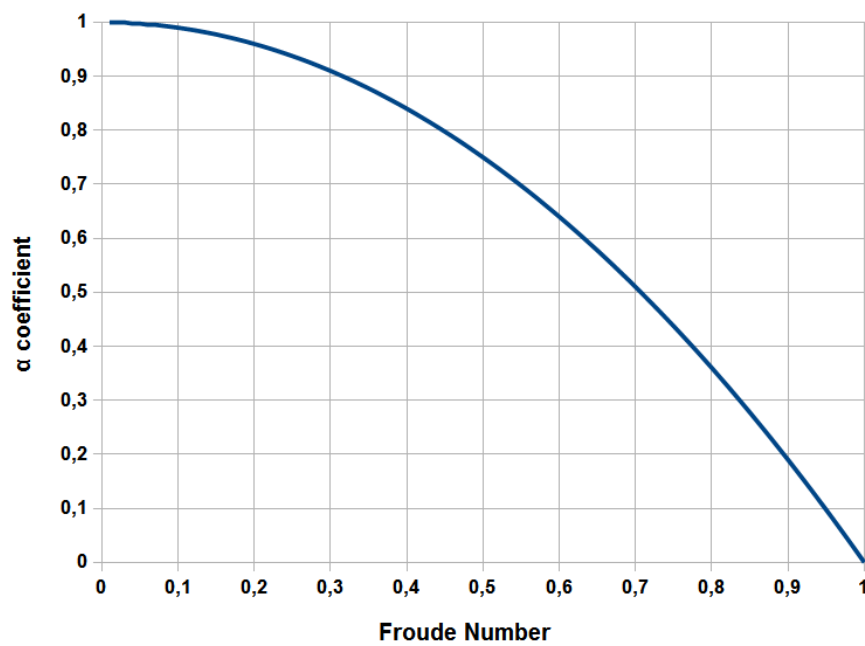


Figure 2.13: Coefficient α as a function of the Froude number

With this simplification, the code REZO is limited in its field of application: the subcritical flows. This improves the robustness in various situations. In return, the fast phenomena are not represented. They are replaced by the calculation of a dummy water level that has no physical reality. Therefore, the user must keep in mind the error committed by the code in rapid flow areas.

3. The Finite Volumes unsteady trans-critical engine MASCARET

3.1 Introduction

The development of the transcritical engine originated from the need to have a code to carry out studies of flood waves, that also met the associated criteria of quality and performance.

The initial objective of the transcritical engine was, therefore, to allow the calculation of :

- flow in valleys with bottom slopes possibly reaching 10% over zones of limited size, and of strong variation of width (plunging gorges, rocky constrictions, etc...);
- very fast flows with the presence of discontinuities ;
- wave propagation over dry zones ;
- water lines in valleys several hundreds of km in length and with the presence of tributaries;
- water lines with supercritical reaches and the treatment of steady flow, for such applications as the flushing of reservoirs.

The various constraints listed above resulted, at a numeric level, in the adoption of a conservative scheme (the Roe scheme with biased treatment of the source terms) allowing for discontinuous solutions.

In the first part 3.2, we recall the one-dimensional equations of Saint-Venant, written in a conservative form.

The second part is devoted to the numerical solution of the homogeneous problem and the source terms for a single reach. Presented in section 3.3 is the solution method used in the transcritical engine of MASCARET, with a *finite volume* scheme explicit or implicit in time.

The explicit scheme can address highly unstationary problems such as flood waves. The numerical results are satisfactory but the constraint on the time step, due to the explicit scheme, is very restrictive for slightly unstationary flows and subcritical flows. This led to implicit the initial scheme linearly.

In the third part 3.4, the more complex physical developments are presented such as the modelling of the compound channels, singularities (downstream dams, weirs, etc.) and confluences.

3.2 Equations of Saint-Venant in a conservative form

3.2.1 General form of the equations (a reminder)

The flow in a river can generally be considered as one-dimensional, i.e. the function of only one variable of space, the curvilinear abscissa x of the river channel.

The variables of the flow are therefore the discharge Q and the wetted section S , functions of x and of time t , and defined in each of the vertical sections of the river.

To obtain the equations of Saint-Venant starting from the equations of Navier-Stokes, it is supposed that the angle between the bottom and the horizontal is always small. Nevertheless, in practical application, we also need to compute zones with very strong slopes, but of limited extent.

In the hypothesis where the pressure within the fluid is hydrostatic and where the internal viscosity effects are negligible compared to the friction on the walls of the channel, Q and S are solutions of the equations of Saint-Venant and are written as [15] [1] :

$$\begin{cases} \frac{\partial S}{\partial t} + \frac{\partial Q}{\partial x} = q_a \\ \frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\frac{\beta(x,S)Q^2}{S} \right) + gS \frac{\partial Z}{\partial x} = gSJ \end{cases} \quad (3.1)$$

where:

- Z is the elevation of the free surface (m);
- S is the wetted section (m^2);
- g is the acceleration of gravity ($m.s^{-2}$);
- J is the linear head loss from friction. J is evaluated with the Strickler formula :

$$J = \frac{Q^2}{K^2 S^2 R_h^{4/3}} \quad (3.2)$$

with R_h the hydraulic radius and K the Strickler friction coefficient.

It is reminded that the Strickler friction coefficient K is equal to $1/n$ where n is the Manning friction coefficient.

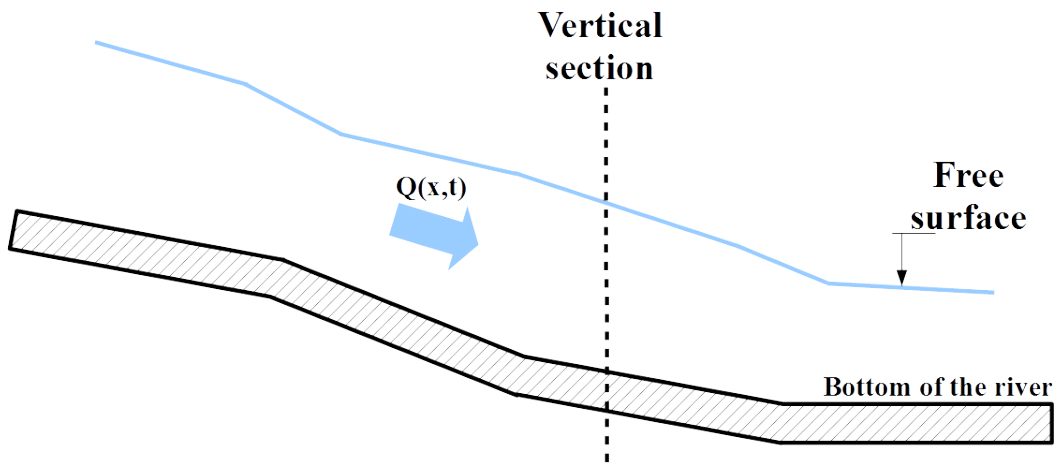


Figure 3.1: Longitudinal view

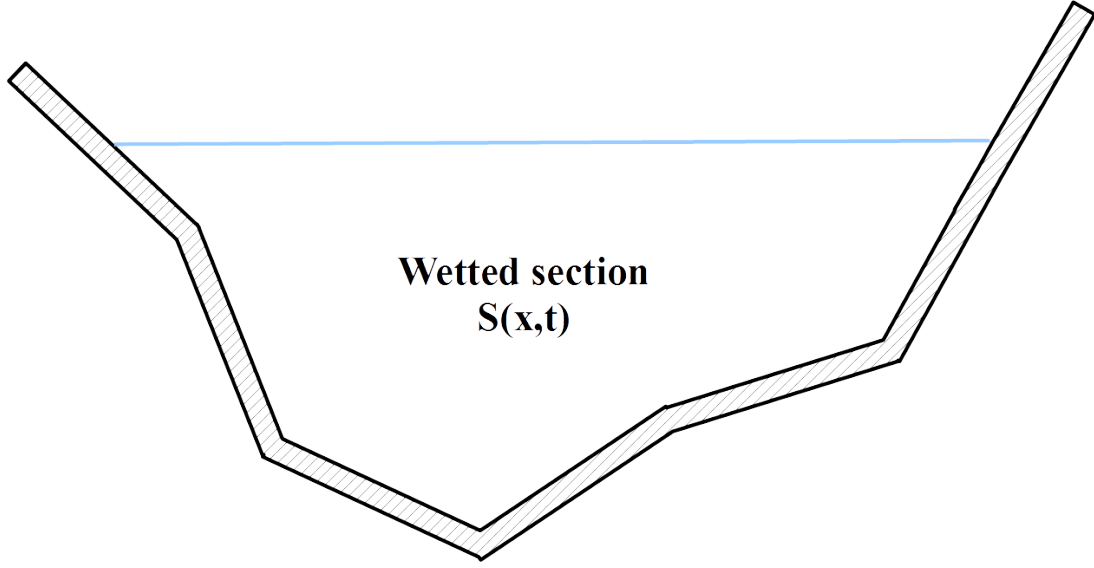


Figure 3.2: Lateral view

The dimensionless coefficient β , results from the variations of the actual velocity of the flow in a section :

$$\beta(x,S) = \frac{S}{Q^2} \int V^2 dS \quad (3.3)$$

In the case of a simple river channel, taking into account the hypothesis used to establish the equations of Saint-Venant, we take β equal to 1, i.e. the variations of velocity within a section are neglected. This will not be true for a compound channel.

The system of equations (3.1) is obtained by expressing the conservation of mass for the first equation and the conservation of momentum for the second.

3.2.2 Equations written in a conservative form

A conservative system of equations can be expressed as :

$$\frac{\partial W}{\partial t} + \text{div} F(W) = B(W) \quad (3.4)$$

where:

- W is the vector of state;
- $F(W)$ is the flux;
- $B(W)$ is the source term.

Or, if there is only one dimension of space :

$$\frac{\partial W}{\partial t} + \frac{\partial F}{\partial x} = B \quad (3.5)$$

This way of writing the equation is interesting because it contains, in terms of distributions, the jump relationships in the case of a shock.

To write the conservative form of these equations from (3.1), we must transform the term : $gS \frac{\partial Z}{\partial x}$ in the dynamic equation. The first two terms of the dynamic equation : $\frac{\partial Q}{\partial t}$ and $\frac{\partial}{\partial x} \left(\frac{Q^2}{S} \right)$

, correspond to the variation in momentum. This variation is equal to the action of external forces, i.e. the friction on the bottom represented by : $-gSJ$ and the forces of gravity and pressure, which are taken into account by : $gS\frac{\partial Z}{\partial x}$.

The term $gS\frac{\partial Z}{\partial x}$ can be transformed by using the term for pressure : $P = g \int_0^y S dy$, where : $y = Z - Z_f$, Z being the free surface elevation and Z_f the bottom elevation.

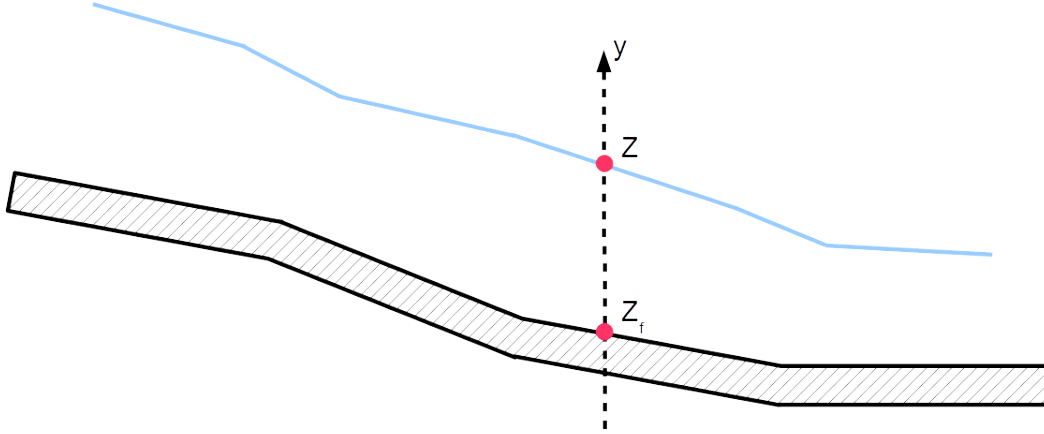


Figure 3.3: Longitudinal view : the elevations

We can write :

$$\frac{\partial P}{\partial x} = gS \frac{\partial y}{\partial x} + g \int_0^y \left(\frac{\partial S}{\partial x} \right) dy \quad (3.6)$$

which is equivalent to :

$$\frac{\partial P}{\partial x} = gS \frac{\partial Z}{\partial x} + g \int_0^y \left(\frac{\partial S}{\partial x} \right) dy - gS \frac{\partial Z_f}{\partial x} \quad (3.7)$$

The dynamic equation can be re-written as :

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q^2}{S} + P \right) = g \int_0^y \left(\frac{\partial S}{\partial x} \right) dy - gS \frac{\partial Z_f}{\partial x} - gSJ \quad (3.8)$$

It is noted that the source term related to the geometry of the dynamic equation can equally be written : $\left(\frac{\partial P}{\partial x} \right)_{z=\text{constant}}$.

The source term of this new equation depends only on the variables Q and S of the geometry of the river, it does not depend on the derivative in respect to x or t of Q and S . In effect, the derivative : $\left(\frac{\partial S}{\partial x} \right)_y$ is purely geometrical (here S indicates the geometric section and not the wetted section). This equation is written in a conservative form.

So, the conservative formulation of the equations of the system (3.1) is (with : $q_a = 0$) :

$$\begin{cases} \frac{\partial S}{\partial t} + \frac{\partial Q}{\partial x} = 0 \\ \frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q^2}{S} + P \right) = g \int_0^y \left(\frac{\partial S}{\partial x} \right)_y dy - gS \frac{\partial Z_f}{\partial x} - gSJ \end{cases} \quad (3.9)$$

with : $P = g \int_0^y S dy$

This equation is actually of the form :

$$\frac{\partial W}{\partial t} + \frac{\partial F}{\partial x} = B \quad (3.10)$$

with :

$$W = \left(\frac{S}{Q} \right) \quad (3.11)$$

$$F = \left(\begin{array}{c} Q \\ \frac{Q^2}{S} + P \end{array} \right) \quad (3.12)$$

and :

$$B = \left(\begin{array}{c} 0 \\ g \int_0^y \left(\frac{S}{x} \right)_y dy - gS \frac{dZ_f}{dx} - gSJ \end{array} \right) \quad (3.13)$$

Comment :

- The flux depends not only on W but also on x . Knowing only W is not sufficient to calculate F . In effect, the term for pressure P appearing in F can only be evaluated if the wetted section and the variation of the section based on the flow depth are known explicitly. However the variation of the section depends on the geometry of the river. Therefore it can only be known if the x axis is given. We will see the importance of this statement in the search for the Riemann invariants of the system of equations (3.9);
- The variables Q and S can be discontinuous (e.g. in the case of a hydraulic jump). The derivatives appearing in (3.9) therefore need to be understood in the sense of distributions. Actually, in this case, the system (3.9) is the only correct equation system, because in (3.1) the term : $gS \frac{\partial Z}{\partial x}$ has no meaning, both in a classical sense and in the sense of the distributions.

3.2.3 Riemann invariants

The system (3.9) is hyperbolic. The jacobian matrix of $F : \left(\frac{\partial F}{\partial W} \right)_x$ admits two real and distinct eigenvalues : λ^+ et λ^- :

$$\left(\frac{\partial F}{\partial W} \right)_x = \left(\begin{array}{cc} 0 & 1 \\ -\frac{Q^2}{S^2} + \left(\frac{\partial P}{\partial S} \right)_x & \frac{2Q}{S} \end{array} \right) \quad (3.14)$$

$$\left\{ \begin{array}{l} \lambda^+ = \frac{Q}{S} + C \\ \lambda^- = \frac{Q}{S} - C \end{array} \right. \quad (3.15)$$

with :

$$C = \sqrt{\left(\frac{\partial P}{\partial S} \right)_x} \quad (3.16)$$

The celerity C is only a function of S and x .

In using the properties of the hyperbolic systems [14][1], (3.9) can be rewritten in the form of two convection equations of the Reiman invariants.

We do not give the details of the calculations here (see [14] and [1]).

We finally obtain the system (3.17) :

$$\begin{cases} \frac{df^-}{dt} = -\lambda^- \int_0^S \frac{1}{S} \left(\frac{\partial C}{\partial x} \right)_S dS - g \left(\left(\frac{\partial y}{\partial x} \right)_S + \frac{dZ_f}{dx} + J \right) \\ \frac{df^+}{dt} = +\lambda^+ \int_0^S \frac{1}{S} \left(\frac{\partial C}{\partial x} \right)_S dS - g \left(\left(\frac{\partial y}{\partial x} \right)_S + \frac{dZ_f}{dx} + J \right) \end{cases} \quad (3.17)$$

where :

- $\frac{df^-}{dt} = \frac{\partial f^-}{\partial t} + \lambda^- \frac{\partial f^-}{\partial x}$ is the derivative of the Reimann invariant : $f^- = \frac{Q}{S} - K$ with $K = \int_0^S \frac{C}{S} dS$ along the characteristic curve C^- of slope : $\frac{dx}{dt} = \lambda^-$;
- $\frac{df^+}{dt} = \frac{\partial f^+}{\partial t} + \lambda^+ \frac{\partial f^+}{\partial x}$ is the derivative of the Reimann invariant : $f^+ = \frac{Q}{S} + K$ along the characteristic curve C^+ of slope : $\frac{dx}{dt} = \lambda^+$.

We refer to the subcritical characteristic as the curve C^+ , and to the supercritical characteristic as the curve C^- .

Expressing (3.9) in the form of equations along the characteristic curves is equivalent to considering two mobile markers moving along the characteristics. The discharge and the wetted section at a point in the channel at a given time, are determined by the information provided by the characteristics C^+ and C^- which intersect at this point.

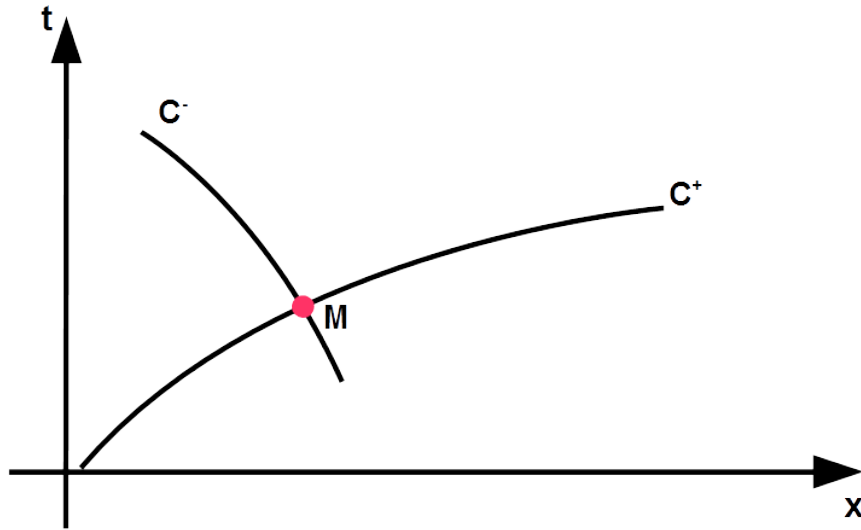


Figure 3.4: The two characteristic curves

Knowing f^+ and f^- (calculated with the help of the information conveyed by the characteristics in M), we can deduce Q and S .

Formulating the system (3.9) in the form of characteristics also allows it to remain *related* to the physical problem during the computational process. We will see an interesting application of this during the processing of the boundary conditions (see section 3.3.4).

However, the system (3.17) is only equivalent to the system (3.9) when the flow has no discontinuities. The development of the derivative $\frac{\partial f}{\partial x}$ used to obtain the system (3.17) is not possible in the presence of a discontinuity.

3.2.4 The hydraulic jump

Note :

we use the terms *hydraulic jump* and *shock* indifferently when describing a discontinuity.

A shock appears in a flow when two characteristics of the same family intersect. Two types of shock exist :

- the one due to the intersection of characteristics C^- : shock C^- ;
- the other due to the meeting of characteristics C^+ : shock C^+ .

At the point where the characteristics meet, two values of f^- (for a C^- shock) or two values of f^+ (for a C^+ shock) are introduced, in addition to the information derived from the other family of characteristics. This overload of information, mathematically unacceptable, results in a discontinuity at the intersection. From a physical point of view it results in the establishment of a dissipative process : the hydraulic jump.

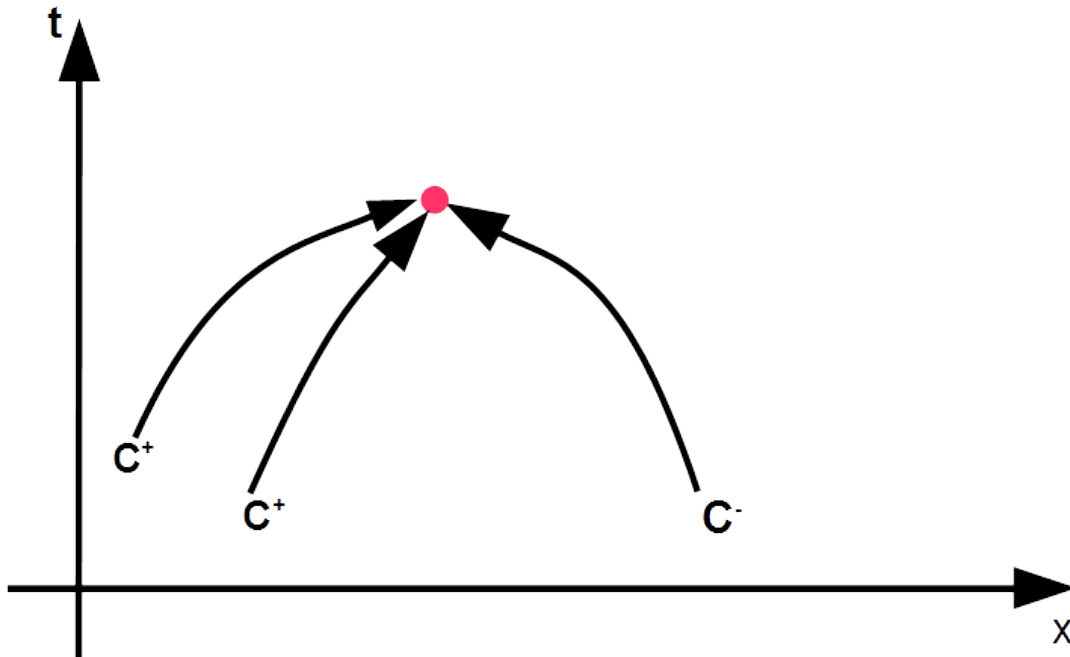
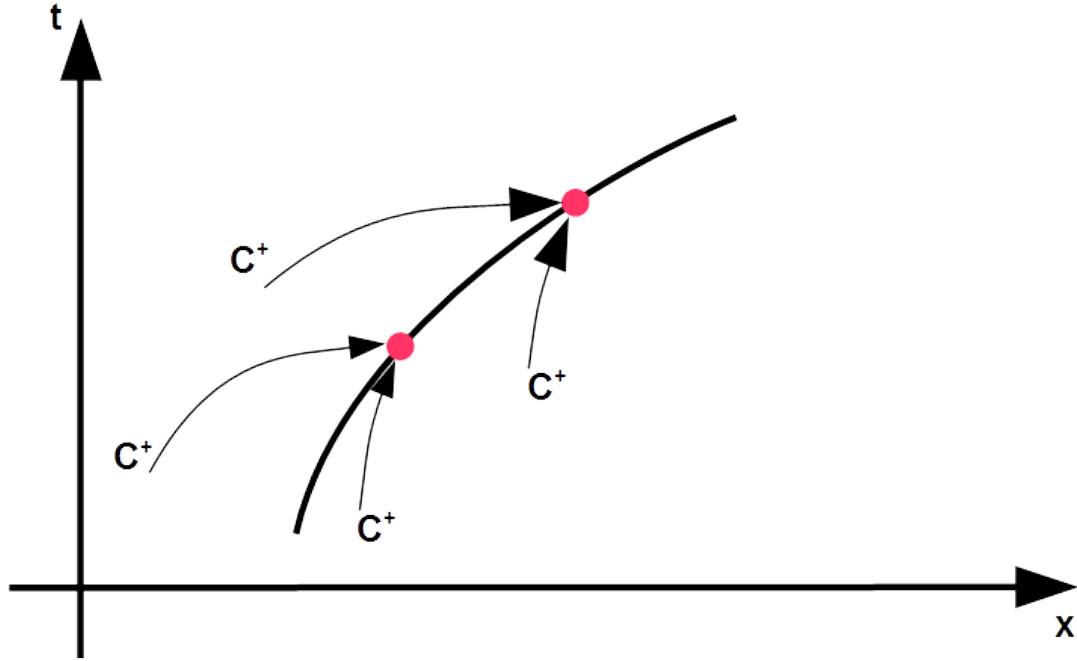


Figure 3.5: Example of a C^+ shock

The trajectory of the shock is made of all the points where characteristics of the same family intersect.

Figure 3.6: Trajectory of a C^+ shock

To the left and to the right of the shock, the matrix $\left(\frac{\partial F}{\partial W}\right)_x$ is well defined, it is similarly true for its eigenvalues which are $(\lambda_g^+, \lambda_g^-)$ to the left and $(\lambda_d^+, \lambda_d^-)$ to the right.

The speed s^- of a C^- shock verifies :

$$\begin{cases} \lambda_d^- < s^- < \lambda_g^- (< \lambda_g^+) \\ s^- < \lambda_d^+ \end{cases} \quad (3.18)$$

and the speed s^+ of a C^+ shock verifies :

$$\begin{cases} (\lambda_d^- <) \lambda_d^+ < s^+ < \lambda_g^+ \\ \lambda_g^- < s^+ \end{cases} \quad (3.19)$$

Indeed, the characteristics that create the shock catch this shock up from upstream (here to the left) and are caught up by it on the downstream (here on the right). This is the meaning of the first inequalities for the C^- shocks and the C^+ shocks : the speed of the shock is intermediate between those of the characteristics that create it. The meaning of the second inequalities is that there cannot be a C^- and a C^+ at the same point : the velocity of a shock cannot be between the velocities of characteristics that did not take part in its creation.

Therefore, the scheme shown in figure 3.7 will never be found.

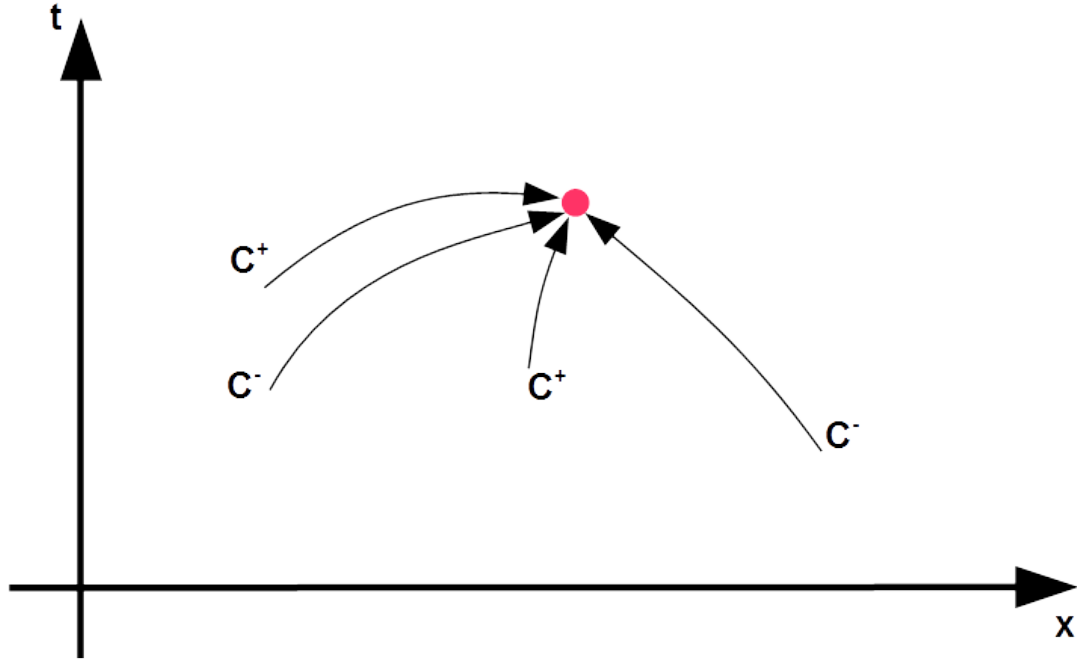


Figure 3.7: Example of a characteristics scheme that is not possible

These inequalities are all the information that the characteristics provide about the shock. Originating from a non-conservative formulation of the equations, they do not lead to a quantitative description of shocks. The states to the left and the right of a shock can not be determined by the Riemann invariants provided by the characteristics intersecting at the shock. However, for weak shocks, the speed of a shock is the average of the speed λ_g and λ_d of the characteristics that create it [34][22]).

The only way to describe shocks correctly is to come back to the equations written in a conservative form. When the flow contains a hydraulic jump, the homogeneous hyperbolic system, interpreted in the sense of distributions, allows to connect the states to the left and right of the hydraulic jump [14].

The relations thereby obtained are the jump conditions (these are also sometimes called Rankine-Hugoniot conditions). They are written as, with s is the celerity of the hydraulic jump :

$$\begin{cases} s(S_d - S_g) = Q_d - Q_g \\ s(Q_d - Q_g) = \left(\frac{Q_d^2}{S_d} + P_d \right) - \left(\frac{Q_g^2}{S_g} + P_g \right) \end{cases} \quad (3.20)$$

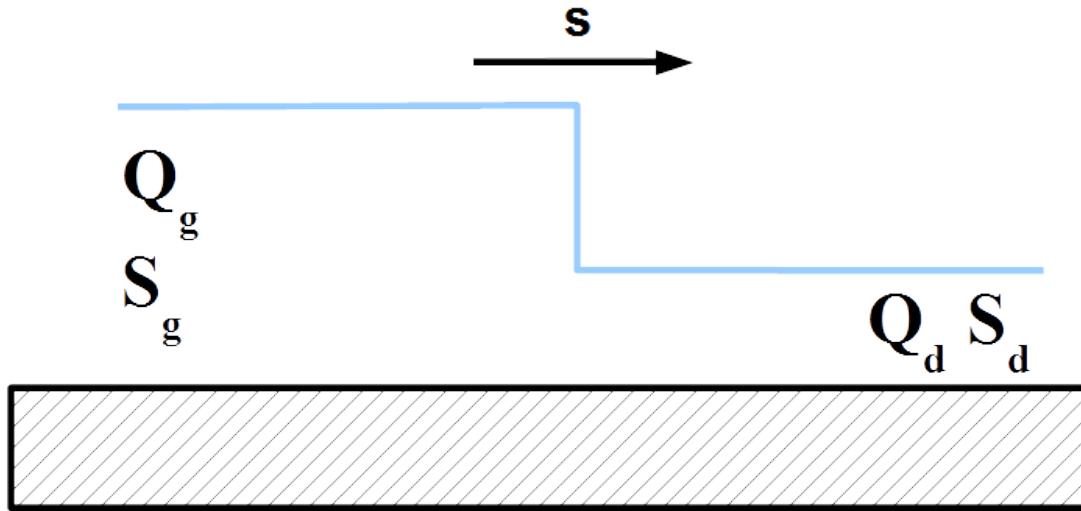


Figure 3.8: Celerity of the hydraulic jump

Comments :

- The Rankine-Hugoniot conditions are only a particular form of the system of equations (3.9). This can be demonstrated by doing a balance of the mass and the momentum over a slice of fluid including the hydraulic jump. In the equation of momentum, the source term disappears when the width of the slice tends towards zero.
- In the coordinate system of the hydraulic jump (in translation at speed s), the discontinuity always separates a supercritical flow from a subcritical flow.
In the frame of reference of the hydraulic jump : $s = 0$, so :
 $Qd = Qg = Q_{rel}$ relative discharge.
The function : $f(S, x) = \frac{Q_{rel}^2}{S} + P$ has the same value on both sides of the jump.
Moreover f admits a minimum for S such that : $\left(\frac{\partial f}{\partial S}\right)_x = 0$, i.e. :
 $Q_{rel} = CS$. The value of : $S = \frac{Q_{rel}}{C}$ corresponds to a critical flow (Froude number equal to 1). Two values of S giving the same value of f (Rankine-Hugoniot condition) are therefore located on each side of the critical section.

3.3 Numerical solution of the St-Venant equations

3.3.1 The explicit scheme

In this section, we describe the numerical methods used in MASCARET to solve the system of equations (3.9). A explicit scheme was developed first, then the transcritical engine was made implicit. This leaves the choice of using one of the two schemes to the user.

Description of the problem

The system to solve is as follows :

$$\begin{cases} W = (S, Q) \\ \frac{\partial W}{\partial t} + \frac{\partial F(W, x)}{\partial x} = B(x, W) \end{cases} \quad (x, t) \in [a, b] \times [0, T] \quad (3.21)$$

with :

$$F(x, W) = \begin{pmatrix} Q \\ \frac{Q^2}{S} + P(x, S) \end{pmatrix} \quad (3.22)$$

and :

$$B(x, W) = \begin{pmatrix} 0 \\ g \int_0^y \left(\frac{\partial S}{\partial x} \right)_y dy - g S \frac{dZ_f}{dx} - g S J \end{pmatrix} \quad (3.23)$$

So that this problem is well presented, it is necessary to add an initial condition and boundary conditions.

The problem (3.21) to be solved, is a strictly hyperbolic system where the source term is dominant in the calculation of the flow (see the previous section). Considering that the scope of application of the MASCARET code is primarily the calculation of flood waves, this implies :

- geometry with very steep slopes (locally reaching 10% possibly) and large variations of width;
- very fast flows (speeds above 10m/s);
- wave propagation over dry areas;
- very long computational domains.

In addition to the flood wave calculations, the intention is to be able to model any flow presenting a supercritical regime such as the flushing of reservoirs and the steady flows in torrents. This adds an additional constraint, which is that the source term must be correctly calculated in order to obtain a correct convergence towards steady flows. The objective was to find a scheme that complies with these various constraints (which can sometimes be contradictory) as closely as possible.

For now we are interested only in the numerical solution of the homogeneous problem. A following section, 3.3.3 is devoted to the treatment of the source terms.

Solution of the homogeneous problem

The homogeneous problem to solve is written :

$$\frac{\partial W(x, t)}{\partial t} + \frac{\partial F(W)}{\partial x} = 0 \quad (3.24)$$

This homogeneous problem is very similar to the isentropic Euler equations. The principle difference resides in the fact that the flux does not depend solely on the state variables (which are the wetted section and the discharge) but also depends on the variable of space x . Moreover, the pressure term is not known explicitly but is a tabulated variable.

The time interval $[0, T]$ is discretised in a series of intervals $[t_n, t_{n+1}]$ with : $t_{n+1} = t_n + \Delta t$ where Δt is the time step.

The computational domain will be discretised in N intervals $[x_i, x_{i+1}]$ with : $\Delta x_i = x_{i+1} - x_i$.

We denote $x_{i+1/2}$ the midpoint of the interval $[x_i, x_{i+1}]$.

The scheme used is a finite volume scheme based on a integral formulation of the mass and momentum balance over a cell $[x_{i-1/2}, x_{i+1/2}]$ (see figure 3.9).

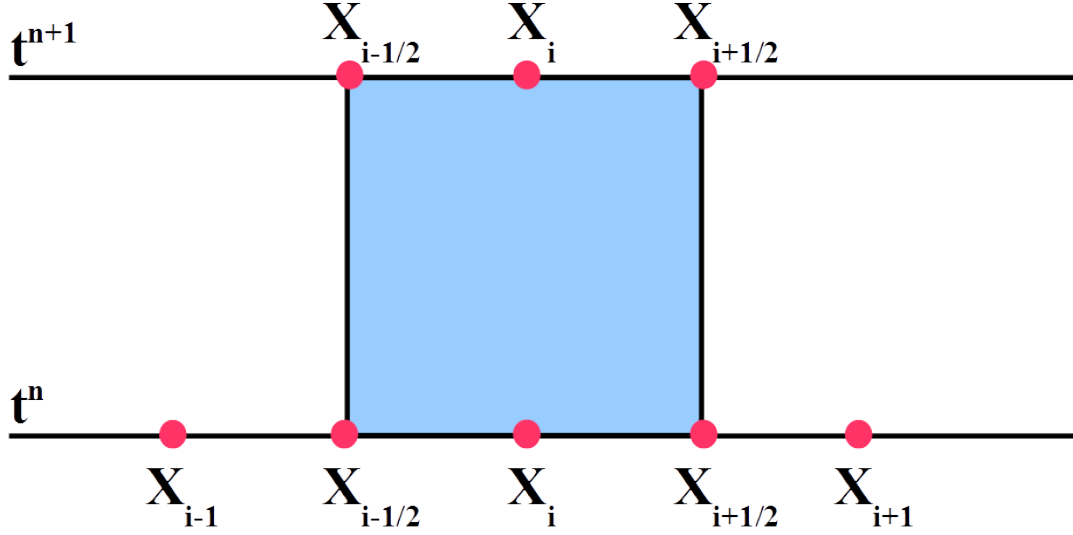


Figure 3.9: Finite volume cell

Note :

$$\underline{W}_h = \left\{ \Psi_h \in L^2(\Omega), \Psi_h|_{[x_{i-1/2}, x_{i+1/2}]} = \text{constant}, i = 1..N \right\}$$

A variational description of the problem (3.21) can be formulated as :

find $W_h(S_h, Q_h) \in \underline{W}_h$ such as :

$$\int_{t_n}^{t_{n+1}} \int_a^b \left(\frac{\partial W_h}{\partial t} + \frac{\partial F(x, W_h)}{\partial x} \right) \Psi_h dx dt = 0 \quad \forall \Psi_h \in \underline{W}_h \quad (3.25)$$

Each function Ψ_h of \underline{W}_h is determined in a unique way by :

$$\Psi_h = \sum_{i=1}^n \Psi(a_i) \phi_i \quad (3.26)$$

with $\{\phi_i\}_{i=1..N}$ which forms a base of \underline{W}_h . In fact, the functions of base ϕ_i are the characteristic functions of the interval $[x_{i-1/2}, x_{i+1/2}]$.

By writing the variational equation (3.25) for each base function and by using the Green formula, we obtain :

$$\begin{aligned} \int_{x_{i-1/2}}^{x_{i+1/2}} W_h^{n+1} dx &= \int_{x_{i-1/2}}^{x_{i+1/2}} W_h^n dx + \int_{t_n}^{t_{n+1}} F(x_{i+1/2}, W_{i+1/2}) dt \\ &\quad - \int_{t_n}^{t_{n+1}} F(x_{i-1/2}, W_{i-1/2}) dt \end{aligned} \quad (3.27)$$

W_h^{n+1} being sought in the space of the functions that are constant per cell, the previous equation becomes :

$$W_i^{n+1} = W_i^n + \frac{1}{x_{i+1/2} - x_{i-1/2}} \times \left(\int_{t^n}^{t^{n+1}} F(x_{i+1/2}, W_{i+1/2}) dt - \int_{t^n}^{t^{n+1}} F(x_{i-1/2}, W_{i-1/2}) dt \right) \quad (3.28)$$

The problem can therefore be reduced to the evaluation of the numerical flux at each interface. Moreover, W_{n+1} being a function constant per cell, the calculation of the numerical flux will require the solution of a Riemann problem at each interface.

Therefore, at each time step and each cell interface, the following problem needs to be solved :

$$\begin{cases} \frac{\partial W_h}{\partial t} + \frac{\partial F(x_{i+1/2}, W_h)}{\partial x} = 0 \\ W_h^n = \begin{cases} W_i^n & \text{si } x < x_{i+1/2} \\ W_{i+1}^n & \text{si } x > x_{i+1/2} \end{cases} \end{cases} \quad (3.29)$$

To solve this problem, it is natural to consider a Godunov scheme, based on the exact solving of the Riemann problem at each interface. However, the interest of this exact solution is counterbalanced by the fact that firstly the projection on each cell can cause a loss of quality from the exact solving, and secondly an exact Riemann solver is computationally expensive. For this reason an approximate Riemann solver has been chosen. More specifically, we have retained the Roe scheme which will be presented in the following paragraph. Moreover, another reason to choose the Roe scheme is that it is widely used for solving Euler and Saint-Venant equations [31][6][2][30].

Roe linearisation

The aim here is to define a Riemann problem *close* to (3.29) but for which the solution is simpler to calculate. To do this, we apply the Roe linearisation method. This method is well known and widely used. We recall the main references where the reader can find the details of the calculation of the Roe linearisation [32][4].

The principle of the Roe linearisation method rely on the fact that there exists a matrix A (called 'Roe linearised') verifying :

- * $F(W_g) - F(W_d) = A(W_g, W_d)(W_g - W_d)$;
- * $A(W, W) = DF_w(W)$;
- * the matrix $A(W, W)$ has real eigenvalues and the eigenvectors generate the complete space.

Comment:

The Roe linearisation method defined here is only valid for cases where the flux F depends solely on the state variables. Nevertheless, this method has been applied in the present case.

The Roe scheme cannot be used if it is not possible to find a matrix satisfying these three properties. In this context, the difficulty lies with the fact that the flux depends on x and the state variables.

To calculate the Roe matrix, the method is similar to that used for the isentropic Euler equations i.e. an average state is sought, so that the Roe matrix is the jacobian matrix of the flux taken in that average state.

In our case, the average of Roe \tilde{W} is given by :

$$\tilde{W} = \frac{W_g \sqrt{S_g} + W_d \sqrt{S_d}}{\sqrt{S_g} + \sqrt{S_d}} \quad (3.30)$$

where :

- $W_g \begin{pmatrix} S_g \\ V_g \end{pmatrix}$ indicates the state to the left;
- and $W_d \begin{pmatrix} S_d \\ V_d \end{pmatrix}$ the state to the right;

The Roe matrix $\tilde{A}(x, W_g, W_d)$ is the jacobian $A(x, W)$ calculated in this state.

$$A(x, \tilde{W}) = \begin{pmatrix} 0 & 1 \\ \frac{-\tilde{Q}^2}{\tilde{S}^2} + \frac{\partial P(x, \tilde{S})}{\partial \tilde{S}} & 2 \frac{\tilde{Q}}{\tilde{S}} \end{pmatrix} \quad \text{with } \tilde{W}(\tilde{S}, \tilde{Q}) \quad (3.31)$$

The Riemann problem is therefore equivalent to the following linearised problem at each interface :

$$\begin{cases} \frac{\partial W(x, t)}{\partial t} + \tilde{A}(x_{i+1/2}, W_g, W_d) \frac{\partial W(x, t)}{\partial x} = 0 \\ W(x, t^n) = \begin{cases} W_g & \text{si } x < x_{i+1/2} \\ W_d & \text{si } x > x_{i+1/2} \end{cases} \end{cases} \quad (3.32)$$

with : $\tilde{A}(x_{i+1/2}, W_g, W_d) = A(x_{i+1/2}, \tilde{W})$

The solution of the linear problem (3.32) will allow the Roe scheme to be fully explicited.

Comment :

The Roe matrix is evaluated at the interface $x_{i+1/2}$.

The Roe solver

The solution of the problem (3.32) is simple. It consists of 3 constant states separated by jumps through the characteristics defined by :

$$\frac{x}{t} = \lambda_{1,2} \quad (3.33)$$

with :

$$\lambda_1(x, S, Q) = \frac{Q}{S} - C(x, S) \quad (3.34)$$

$$\lambda_2(x, S, Q) = \frac{Q}{S} + C(x, S) \quad (3.35)$$

and :

$$C(x, S) = \frac{\partial P(x, S)}{\partial S} \quad (3.36)$$

We can note that :

$$\begin{cases} W_g = \sum_{i=1,2} \alpha_{ig}(W_g, W_d) r_i(W_g, W_d) \\ W_d = \sum_{i=1,2} \alpha_{id}(W_g, W_d) r_i(W_g, W_d) \end{cases} \quad (3.37)$$

where r_i are the right side eigenvectors associated with the eigenvalue λ_i .

By supposing $\lambda_1 < \lambda_2$, the solution is written:

$$W\left(\frac{x}{t}, W_g, W_d\right) = \begin{cases} W_g & \text{if : } \frac{x}{t} < \lambda_1(x, W_g, W_d) \\ W_m & \text{if : } \lambda_1 < \frac{x}{t} < \lambda_2 \\ W_d & \text{if : } \frac{x}{t} > \lambda_2(x, W_g, W_d) \end{cases} \quad (3.38)$$

with : $W_m = \alpha_{1d} r_1(W_d, W_g) + \alpha_{2g} r_2(W_d, W_g)$

A is the jacobian matrix of F which is diagonalisable in the basis of the eigenvectors X . A is therefore equal to : $X\Lambda X^{-1}$

$|\Lambda|$ is the diagonal matrix of the generic term $|\Lambda_i|$, Λ^+ and Λ^- defined respectively by :

$$\begin{cases} \Lambda_{i,j}^+ = \lambda_i^+ \delta_{i,j} \\ \Lambda_{i,j}^- = \lambda_i^- \delta_{i,j} \end{cases} \quad (3.39)$$

In the same way A^+ , A^- et $|A|$ are defined by :

$$\begin{cases} A^+ = X\Lambda^+X^{-1} \\ A^- = X\Lambda^-X^{-1} \\ |A| = X|\Lambda|X^{-1} \end{cases} \quad (3.40)$$

The Roe flux therefore takes the following form :

$$\begin{aligned} F(x_{i+1/2}, W_g, W_d) &= \frac{1}{2} (F(x_{i+1/2}, W_g) + F(x_{i+1/2}, W_d)) \\ &\quad + \frac{1}{2} |\tilde{A}(x_{i+1/2}, W_g, W_d)| (W_d - W_g) \\ &= F(x_{i+1/2}, W_d) - \tilde{A}^+(x_{i+1/2}, W_d, W_g)(W_d - W_g) \\ &= F(x_{i+1/2}, W_g) + \tilde{A}^-(x_{i+1/2}, W_d, W_g)(W_d - W_g) \end{aligned} \quad (3.41)$$

From the definition of the flux, the scheme is completely defined. This scheme is first-order in space and time. Moreover, this scheme is stable if the Courant number is less than 1.

Entropic correction

The main drawback of the Roe scheme is that it is not entropic and can therefore admit non-entropic stationary discontinuities in the vicinity of the sonic points, i.e. where an eigenvalue associated with a nonlinear field of jacobian matrix changes sign on both sides of an interface. To avoid this problem, it is therefore necessary to modify the calculation of the flux in the vicinity of the points where an eigenvalue $\lambda_{m=1,2}(\tilde{W})$ is close to zero.

It is possible to consider several modifications of the eigenvalue $\lambda_m(\tilde{W})$. We have retained the entropic correction defined by Leveque [26]. $\lambda_m(\tilde{W})$ is replaced by : $\lambda_m(W_d) \left(\frac{\lambda_m(\tilde{W}) - \lambda_m(W_g)}{\lambda_m(W_d) - \lambda_m(W_g)} \right)$

Comment :

A second-order on time scheme would remove the need for the entropic correction.

3.3.2 Use of the implicit scheme

Introduction

This paragraph describes the implicit scheme developed in the MASCARET code. The aim of this development was to lift the time step constraint coming from the CFL condition for an explicit scheme (see section 3.3.1). The applications targeted by this development are essentially

subcritical situations (propagation of medium floods) for which the numerical constraint on the time step is very detrimental in terms of computing time.

This scheme is based on a linear implicitation of the Roe-type finite-volumes scheme. The linear scheme is solved by a direct method that avoids the convergence problems of an iterative method when the matrix is poorly conditioned. The direct method is a method of decomposition of type *LU* adapted to tri-diagonal block matrices.

The results obtained with the implicit scheme on validation cases are detailed in the note [16]. The study cases include both analytical tests and more complex tests representative of real applications : propagation of floods, flushing of reservoirs and flood waves.

Reminder and interest of the implicitation

We recall that the problem to solve, detailed in section 3.2 (the St-Venant equations written in their conservative form), is a strictly hyperbolic system with a prominent source term :

$$\begin{cases} W = (S, Q) \\ \frac{\partial W}{\partial t} + \frac{\partial F(W, x)}{\partial x} = B(x, W) \quad (x, t) \in [a, b] \times [0, T] \end{cases} \quad (3.42)$$

with :

$$F(W, x) = \begin{pmatrix} Q \\ \frac{Q^2}{S} + P(x, S) \end{pmatrix} \quad (3.43)$$

and :

$$B(x, W) = \begin{pmatrix} 0 \\ g \int_0^y \left(\frac{\partial S}{\partial x} \right)_y dy - gS \frac{\partial Z_f}{\partial x} - gSJ \end{pmatrix} \quad (3.44)$$

The numerical scheme used to solve this problem is detailed in the previous section : the scheme used is a finite volume scheme based on the integral formulation of the balance of mass and of momentum on a cell, $[x_{i-1/2}, x_{i+1/2}]$ (see section 3.3.1).

After a discretisation of the scheme in time and space, and if the discrete solution is sought in the space of the functions constant over a cell, the problem (3.42) is written :

$$\begin{aligned} W_i^{n+1} &= W_i^n - \frac{1}{x_{i+1/2} - x_{i-1/2}} \\ &\times \left(\int_{t^n}^{t^{n+1}} F(x_{i+1/2}, W_{i+1/2}) dt - \int_{t^n}^{t^{n+1}} F(x_{i-1/2}, W_{i-1/2}) dt \right) \\ &+ B_{i+1/2}^n - B_{i-1/2}^n \end{aligned} \quad (3.45)$$

with :

- Δt the time step and : $t_{n+1} = t_n + \Delta t$;
- $\Delta x = x_{i+1} - x_i$;
- $x_{i+1/2}$ the midpoint of the interval $[x_i, x_{i+1}]$.

The problem is then reduced to the evaluation of the numerical flux F at each interface ; as done for the Roe scheme (see 3.3.1) and recalled below. The source terms are evaluated with a mixed treatment, centered and decentered, which is detailed in the following section.

The jacobian matrix A of the function F is diagonalisable in the basis of the eigenvectors X . A is therefore equal to $X\Lambda X^{-1}$.

\tilde{A} denote the Roe Matrix : $\tilde{A} = \tilde{X} \tilde{\Lambda} \tilde{X}^{-1}$

$|\Lambda|$ is the matrix of generic term $|\Lambda_i|$, Λ^+ and Λ^- are respectively defined by : $\Lambda_{i,j}^+ = \lambda_i^+ \delta_{i,j}$ and $\Lambda_{i,j}^- = \lambda_i^- \delta_{i,j}$.

In the same way \tilde{A}^+ , \tilde{A}^- and $|\tilde{A}|$ are defined by : $\tilde{A}^+ = \tilde{X} \tilde{\Lambda}^+ \tilde{X}^{-1}$, $\tilde{A}^- = \tilde{X} \tilde{\Lambda}^- \tilde{X}^{-1}$ and $|\tilde{A}| = \tilde{X} |\tilde{\Lambda}| \tilde{X}^{-1}$.

Therefore, the Roe flux takes the following form :

$$\begin{aligned} F(x_{i+1/2}, W_g, W_d) &= \frac{1}{2} (F(x_{i+1/2}, W_g) + F(x_{i+1/2}, W_d)) \\ &\quad + \frac{1}{2} |\tilde{A}(x_{i+1/2}, W_g, W_d)| (W_d - W_g) \\ &= F(x_{i+1/2}, W_d) - \tilde{A}^+(x_{i+1/2}, W_d, W_g) (W_d - W_g) \\ &= F(x_{i+1/2}, W_g) + \tilde{A}^-(x_{i+1/2}, W_d, W_g) (W_d - W_g) \end{aligned} \quad (3.46)$$

This is a first-order scheme in space and in time. It is stable under the following CFL condition :

$$\max(|\lambda_1|, |\lambda_2|) \frac{\Delta t}{\min_{i=2, N-1} (x_{i+1} - x_i)} < 0.5 \quad (3.47)$$

In theory, the Courant number is limited to 0.5 but in practice it is limited to 0.9.

This condition implies that the computation time step over the whole domain is constrained in part by the largest of the two eigenvalues and by the zone with the finest meshing. This means that, if the spatial step is locally decreased, the global time step is also reduced.

The advantages of the implicitation are that on the one hand it is possible to make calculations with Courant numbers well above 1 and on the other hand it is possible to avoid the local effect of a refinement of the mesh, even if the Courant number is not very high overall.

Description of the implicit scheme

In the previous section, the explicit scheme with a Roe solver has given satisfaction in terms of the quality of the results for both highly non-stationary and subcritical flows. However, in the case of subcritical flows, the constraint on the time step is very restrictive because the Courant number is determined by the largest of the eigenvalues which is in turn directly linked to the wave celerity, i.e. for an extreme case such as a dam at rest, the time step is constrained by the water depth.

To remove this CFL condition, the *Finite Volume* scheme was linearly implicit.

In this situation, the implicitation concerns only the mass and momentum flux terms. On the other hand, the source terms flux is not modified (see section 3.3.3). The implicitation is therefore presented on a homogenous system to simplify its presentation. In a general way, the implicit *Finite Volume* scheme is written :

$$h_i(W_i^{n+1} - W_i^n) + \Delta t (F_{i+1/2}(W_i^{n+1}, W_{i+1}^{n+1}) - F_{i-1/2}(W_{i-1}^{n+1}, W_i^{n+1})) = 0 \quad (3.48)$$

The function $F_{i+1/2}$ being not linear, it is linearised using a limited development :

$$\begin{aligned} F_{i+1/2}(W_i^{n+1}, W_{i+1}^{n+1}) &= F_{i+1/2}(W_i^n, W_{i+1}^n) \\ &\quad + \Delta t \frac{\partial F_{i+1/2}^n}{\partial W_i} \delta W_i \\ &\quad + \Delta t \frac{\partial F_{i+1/2}^n}{\partial W_{i+1}} \delta W_{i+1} + O(\Delta t) \end{aligned} \quad (3.49)$$

where : $\delta W_{i+1} = W_{i+1}^{n+1} - W_{i+1}^n$

In the case of a Roe scheme, the flux function F is written as :

$$\begin{aligned} F_{i+1/2}^{n+1}(W_i^{n+1}, W_{i+1}^{n+1}) &= \frac{1}{2}(F(W_i^{n+1}) + F(W_{i+1}^{n+1})) \\ &\quad - \frac{1}{2}|\tilde{A}(x_{i+1/2}, W_i, W_{i+1})|(W_{i+1} - W_i) \end{aligned} \quad (3.50)$$

With this definition of the numerical flux function, the limited development of $F_{i+1/2}$ is written :

$$\begin{aligned} F_{i+1/2}(W_i^{n+1}, W_{i+1}^{n+1}) &\approx F_{i+1/2}(W_i^n, W_{i+1}^n) \\ &\quad \frac{1}{2} \left(\frac{\partial F(W_i)}{\partial W_i} + |\tilde{A}(x_{i+1/2}, W_i, W_{i+1})| \delta W_i \right) \\ &\quad \frac{1}{2} \left(\frac{\partial F(W_{i+1})}{\partial W_{i+1}} - |\tilde{A}(x_{i+1/2}, W_i, W_{i+1})| \delta W_{i+1} \right) \end{aligned} \quad (3.51)$$

The complete scheme therefore takes the following form :

$$\begin{aligned} h_i(W_i^{n+1} - W_i^n) + \Delta t (F_{i+1/2}(W_i^n, W_{i+1}^n) - F_{i-1/2}(W_i^n, W_{i-1}^n)) \\ + \frac{\Delta t}{2} \left(\frac{\partial F(W_{i+1})}{\partial W_{i+1}} - |\tilde{A}(x_{i+1/2}, W_i, W_{i+1})| \delta W_{i+1} \right) \\ - \frac{\Delta t}{2} \left(\frac{\partial F(W_{i-1})}{\partial W_{i-1}} + |\tilde{A}(x_{i-1/2}, W_i, W_{i-1})| \delta W_{i-1} \right) \\ + \frac{\Delta t}{2} (|\tilde{A}(x_{i+1/2}, W_i, W_{i+1})| + |\tilde{A}(x_{i-1/2}, W_i, W_{i-1})|) \delta W_i = 0 \end{aligned} \quad (3.52)$$

We obtain :

$$\begin{aligned} &\frac{1}{2} \left[\frac{\partial F(W_{i-1})}{\partial W_{i-1}} + |\tilde{A}(x_{i-1/2}, W_i, W_{i-1})| \right] \delta W_{i-1} \\ &+ \left[\frac{h_i}{\Delta t} + \frac{1}{2} |\tilde{A}(x_{i+1/2}, W_i, W_{i+1})| + \frac{1}{2} |\tilde{A}(x_{i-1/2}, W_i, W_{i-1})| \right] \delta W_i \\ &\quad + \frac{1}{2} \left[\frac{\partial F(W_{i+1})}{\partial W_{i+1}} - |\tilde{A}(x_{i+1/2}, W_i, W_{i+1})| \right] \delta W_{i+1} \\ &= - \frac{F_{i+1/2}(W_i^n, W_{i+1}^n) - F_{i-1/2}(W_i^n, W_{i-1}^n)}{\Delta t} \end{aligned} \quad (3.53)$$

That results in the solving of a linear system written as :

$$P_{i,i-1} \delta W_{i-1} + Q_{i,i} \delta W_i + R_{i,i+1} \delta W_{i+1} = S_i \quad \forall i = 2..N-1 \quad (3.54)$$

with :

$$\begin{cases} P_{i,i-1} = -\frac{1}{2}(A(W_{i-1}) + |\tilde{A}(x_{i-1/2}, W_i, W_{i-1})|) \\ Q_{i,i} = \frac{h_i}{\Delta t} \frac{1}{2} (|\tilde{A}(x_{i+1/2}, W_i, W_{i+1})| + |\tilde{A}(x_{i-1/2}, W_i, W_{i-1})|) \\ R_{i,i+1} = \frac{1}{2}(A(W_{i+1}) - |\tilde{A}(x_{i+1/2}, W_i, W_{i+1})|) \end{cases} \quad (3.55)$$

The numeric system to be solved is therefore a tridiagonal block system of the second-dimension (the size of the physical system). More precisely, the system is written :

$$\begin{pmatrix} I_d & 0 & & & \\ 0 & P_{2,1} & Q_{2,2} & R_{2,3} & \\ & & \dots & & \\ & & & P_{i,i-1} & Q_{i,i} & R_{i,i+1} & \\ & & & & \dots & & \\ & & & & & 0 & I_d \end{pmatrix} \begin{pmatrix} W_1 \\ W_2 \\ \\ W_{i-1} \\ W_i \\ W_{i+1} \\ \\ W_n \end{pmatrix} = \begin{pmatrix} W_1 \\ S_2 \\ \\ S_i \\ S_{n-1} \\ W_n \end{pmatrix} \quad (3.56)$$

where : $W_i = (\delta S_i, \delta Q_i)^T$

The second member S takes into account the explicit part of the momentum and mass fluxes, and the source terms fluxes.

The *boundary condition states* W_1 and W_n are calculated with the Riemann invariants or by solving explicitly the continuity equation. It is important to note that the calculation made with the continuity equation is explicit and is only applicable for Courant numbers equal to or less than 1.

To obtain a better conditioned matrix, the first and last equations of the preceding system are removed and the second members S_2 and S_{n-1} are modified as a consequence.

The solution of the linear system was initially calculated with an iterative method of a normal equation type. This is a robust method, appropriate for all linear systems which do not present any particular properties ; which is the case for this matrix.

The first tests with an iterative method showed on one hand that the results obtained with the implicit scheme were correct but that negligible computational time was saved because the greater the Courant number the more poorly the matrix was conditioned. Consequently, the saving in computational time that was hoped for with Courant numbers greater than 1, was lost due to the bad convergence of the iterative method.

To mitigate this problem, a direct method is substituted in place of the iterative method. The direct method is a *LU* decomposition applied to a tridiagonal block matrix.

As a result, the calculation time for a time step is completely independent from the conditioning of the matrix and therefore from the Courant number.

Conclusion

The simulations with the implicit scheme can be carried out with either a constant time step or with a variable time step corresponding to a Courant number greater than 1. It must be noted that the explicit scheme does not allow for a simulation with a constant time step because that would mean the time step satisfying the CFL condition at each temporal iteration needs to be known *a priori*.

The linear implicitation of the Finite Volume scheme in MASCARET allows therefore to lift the constraint on the time steps while keeping the results quality similar to those of the explicit scheme (except for the extreme cases of dam break flow on a frictionless dry bottom).

The validation of the implicit scheme has shown that :

- for fluvial applications such as an average-sized flood on a reach of the river Rhone (France), a reduction of the computation time by a factor of 20 can be obtained between the explicit scheme and the implicit scheme;
- for studies such as flood waves (that were not initially targeted by this development), a reduction of the computation time by a factor of 3 to 4 can be achieved.

Nevertheless, the implicit scheme is not unconditionally stable because :

- the fluxes of the source terms are explicit;
- the treatment of the boundary conditions is also explicit.

3.3.3 Treatment of the source terms

Overview

Source terms have a predominant role in the Saint-Venant equations : they largely control the evolution of the flow. It was therefore considered important to give much attention to the discretisation of these terms in the development of the MASCARET solver. Inappropriate treatment can adversely affect the quality of the model results. For example, a body of water initially at rest (horizontal level) in a channel with variations in bottom elevation and/or width could be artificially put in motion.

The Saint-Venant equations have three different source terms (refer to section 3.2) :

- **the input flows** : q_a ;
- **the source term related to the geometry** : this term encompasses the variations in bottom elevation and width (specific to 1D equations). It can be written in a compact and generic form as:

$$\left(\frac{\partial P(x, S)}{\partial x} \right)_{z=cst} = \left(\frac{\partial P(x, S)}{\partial x} \right)_{S=cst} + \frac{\partial P(x, S)}{\partial S} \left(\frac{\partial S}{\partial x} \right)_{z=cst} \quad (3.57)$$

It is important to note that this term is made of two parts : the first relates to pressure variations as a function of the geometry (and not as a function of state variables), the second is the product of the square of the speed (thus function of state variables) by the variations in the wet cross-sectional area (thus geometry);

- **the friction** : $-gSJ$ where J is represented by the Strickler formulation.

The discretisation of the source terms is key in the overall scheme. It must conform to the following criteria :

- maintain a stable horizontal water body at rest;
- converge rapidly toward a steady state for a river with constant flow rate, without oscillations at the interface when friction is included;
- represent friction adequately for shallow water depth flows (front of the wave in flood wave computations).

Note :

The treatment of the source terms is the same for explicit and implicit schemes.

The non-homogeneous problem to solve can be written as :

$$\frac{\partial W(x,t)}{\partial t} + \frac{\partial F(W)}{\partial x} = B(x, W) \quad (3.58)$$

Using a classical finite volume discretisation, the problem can then be written as :

$$W_j^{n+1} = W_j^n - \frac{\Delta t}{\Delta x} (F_{j+1/2}^n - F_{j-1/2}^n) + \int_{\Delta t} \int_{\Delta x} B(x, W^n) dx dt \quad (3.59)$$

The problem becomes: how to discretise the term

$$\int_{\Delta t} \int_{\Delta x} B(x, W^n) dx dt$$

?

A centred discretisation could be used whereby the source term $B(x, W)$ is approximated at the centre of the cell and then integrated in time and space, but this discretisation does not verify the first two criteria above. In fact, an effective way to solve this problem is to off-centre the source terms (upwinding), in which case : $\int_{\Delta t} \int_{\Delta x} B(x, W^n) dx dt$ is approximated by : $\frac{\Delta t}{2} (B_{j+1/2}^n + B_{j-1/2}^n)$ with :

$$\begin{cases} B_{j+1/2}^n = \int_{x_j}^{x_{j+1}} \Psi_d(x_j, x_{j+1}, W_j^n, W_{j+1}^n) \\ B_{j-1/2}^n = \int_{x_j}^{x_{j+1}} \Psi_g(x_j, x_{j-1}, W_j^n, W_{j-1}^n) \end{cases} \quad (3.60)$$

It became apparent during the development of the solver that only the source terms containing the state variables to the first order needed to be off-centred. The other terms, purely geometrical, are treated in a centred way.

Upwinding the source terms

There are few articles in the literature covering the specific problem of source terms in hyperbolic systems. One of the only methods that is adequate when the homogeneous problem is solved using a Roe scheme is that proposed by M. E. Vazquez Cendon [6]. This method was initially applied to the source term for bottom elevation variations to solve the two-dimensional Saint-Venant equations [10] with a Roe scheme. The results obtained were found satisfactory. The same method was therefore applied to some of the source terms in the one-dimensional Saint-Venant equations. This method is presented below in detail.

The numerical basis presented by Vazquez Cendon is described below. The purpose is to solve the equation with source term defined in the previous section.

With the notations introduced earlier, A is the Jacobian of F , which can be made diagonal in the basis of eigenvectors on the right X . A is therefore equal to : $X \Lambda X^{-1}$.

Given $|\Lambda|$ the diagonal matrix with generic term $|\Lambda_i|$.

Λ^+ and Λ^- are respectively defined as : $\Lambda_{i,j}^+ = \lambda_i^+ \delta_{i,j}$ et $\Lambda_{i,j}^- = \lambda_i^- \delta_{i,j}$.

Initially, in the continuous case, it is assumed that all the eigenvalues for A are non-null. In this case A^{-1} and Λ^{-1} exist.

The source term $B(x, W)$ is projected on the eigenvectors for A :

The components of the source term in this basis are noted $\sigma(W)$.

$$\begin{aligned}
B(x, W) &= X(W) \cdot \sigma(W) \\
&= X(W) \cdot (\Lambda \cdot \Lambda^{-1}) \cdot \sigma(W) \\
&= X(W) \cdot (\Lambda^+ + \Lambda^-) \cdot \Lambda^{-1} \cdot \sigma(W) \\
&= X(W) \cdot \Lambda^+ \cdot \Lambda^{-1} \cdot \sigma(W) + X(W) \cdot \Lambda^- \cdot \Lambda^{-1} \cdot \sigma(W)
\end{aligned} \tag{3.61}$$

Given the relationships between Λ^+ , Λ^- and $|\Lambda|$, the following expression can be written as :

$$\begin{aligned}
X(W) \cdot \Lambda^+ \cdot \Lambda^{-1} \cdot \sigma(W) &= \frac{1}{2} X(W) \cdot (\Lambda \cdot \Lambda^{-1} + |\Lambda| \cdot \Lambda^{-1}) \cdot \sigma(W) \\
&= \frac{1}{2} (B(x, W) + |A| \cdot A^{-1} \cdot X(W) \cdot \sigma(W)) \\
&= \frac{1}{2} (I + |A| \cdot A^{-1}) \cdot B(x, W) \\
&= X(W) \cdot \left[\frac{1}{2} (I + \Lambda \cdot \Lambda^{-1}) \right] \cdot X^{-1}(W) \cdot B(x, W) \\
X(W) \cdot \Lambda^- \cdot \Lambda^{-1} \cdot \sigma(W) &= X(W) \cdot \left[\frac{1}{2} (I - \Lambda \cdot \Lambda^{-1}) \right] \cdot X^{-1}(W) \cdot B(x, W)
\end{aligned} \tag{3.62}$$

In the discontinuous case, two functions Ψ_l and Ψ_r can be defined by analogy on either side of the considered node.

$$\begin{cases} \Psi_l(x, y, U, V) = \gamma \cdot X(W(U, V)) \times \\ \quad \left(\frac{1}{2} (I + |\Lambda(U, V)| \Lambda^{-1}(U, V)) X^{-1}(W(U, V)) \tilde{B}(x, y, u, V) \right) \\ \Psi_r(x, y, U, V) = \gamma \cdot X(W(U, V)) \times \\ \quad \left(\frac{1}{2} (I - |\Lambda(U, V)| \Lambda^{-1}(U, V)) X^{-1}(W(U, V)) \tilde{B}(x, y, u, V) \right) \end{cases} \tag{3.63}$$

with:

$$\tilde{B}(x, y, u, V) = B\left(\frac{x+y}{2}, \tilde{W}(U, V)\right) \tag{3.64}$$

where $\tilde{W}(U, V)$ is the Roe average.

In cases where an eigenvalue is null, the projection on the associated eigenvector is considered in centred form, giving :

$$\begin{cases} \left(\frac{\gamma}{2} (I + |\Lambda(U, V)| \Lambda^{-1}(U, V)) \right)_i = \frac{\gamma}{2} \\ \left(\frac{\gamma}{2} (I - |\Lambda(U, V)| \Lambda^{-1}(U, V)) \right)_i = \frac{\gamma}{2} \end{cases} \tag{3.65}$$

The value of γ is obtained using the following equation [6].

$$\frac{\Psi_l(x, x, W, W) + \Psi_r(x, x, W, W)}{2} = B(x, W) \tag{3.66}$$

which gives : $\gamma = 2$.

The contribution of the source terms will therefore be determined in each half cell by the relations :

- to the left of the interface, in the downstream half of cell i :
 $\int_{x_i}^{x_{i+1/2}} \Psi_d(i, i+1) dx;$

- to the right of the interface, in the upstream half of cell $i + 1$:

$$\int_{x_{i+1/2}}^{x_{i+1}} \Psi_g(i, i+1) dx.$$

It is useful to explicitly compute the contribution of a source term noted TS in the momentum equation.

Given \tilde{U} the Roe average of the speed and \tilde{c} the Roe celerity. The Froude number of Roe is defined at the interface by the relation :

$$\tilde{F}_r = \frac{|\tilde{u}|}{\tilde{c}} \quad (3.67)$$

Similarly, the functions Ψ_l and Ψ_r take the following values depending on the values of \tilde{F}_r and on the sign of \tilde{u} (assumed > 0 for simplicity) :

* $\tilde{F}_r < 1$ (subcritical flow)

$$\Psi_l = \begin{pmatrix} \frac{\tilde{T}S}{\tilde{c}} \\ (1 - \tilde{F}_r).TS \end{pmatrix} \quad \text{and} \quad \Psi_r = \begin{pmatrix} -\frac{\tilde{T}S}{\tilde{c}} \\ (1 + \tilde{F}_r).TS \end{pmatrix} \quad (3.68)$$

* $\tilde{F}_r > 1$ (supercritical flow)

$$\Psi_l = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \text{and} \quad \Psi_r = \begin{pmatrix} 0 \\ 2.\tilde{T}S \end{pmatrix} \quad (3.69)$$

with : $\tilde{T}S = TS(\frac{x+y}{2}, \tilde{W}(U, V))$

This treatment of the source terms therefore leads to a non-null contribution in the continuity equation only for subcritical flows. In this case, the upwinding of the momentum is proportional to the Froude number.

When \tilde{F}_r increases, the contribution of the source term increases in the upstream half of the cell (to the right of the interface, Ψ_r term) and decreases in the downstream half (to the left of the interface, Ψ_l term). This is expected since information in the cell tends to come preferentially from upstream. The case with supercritical flows is the extreme case where the upwinding is complete: in a cell, the only source term which matters is that from the upstream interface; this is actually where all the information comes from.

Note :

Upwinding a source term requires that it is evaluated at every interface and for the Roe average.

Treatment of the various source terms

It became apparent during the development of the solver that only the source terms containing the state variables to the first order needed to be upwinded. The other terms, purely geometrical, are treated in a centred way.

A) Source terms related to the geometry

This term can be written $\left(\frac{\partial P}{\partial x}\right)_z$ and takes into account the variations in bottom elevation and width. It can be decomposed as indicated by (3.57).

The treatment of these two terms will be examined separately.

\Rightarrow Treatment of $\frac{\partial P(x,S)}{\partial S} \left(\frac{\partial S}{\partial x}\right)_{z=cst}$:

This term can also be written as : $C^2(x, S) \left(\frac{\partial S}{\partial x} \right)_{x=cstt}$ where C is the celerity. This term depends on the state variables. It can be approximated at the interface, and then upwinded using the method presented in the previous section in an effort to be consistent with the flux formulation. Approximation at the interface between cells i and $i + 1$:

To approximate the purely geometrical part, $\left(\frac{\partial S}{\partial x} \right)_{z=cstt}$, at the elevation z_i , the wetted cross-section is S_i to the left, and S_{i+1}^* to the right. Similarly at the elevation z_{i+1} the wetted cross-section is S_i^* to the left, and S_{i+1} to the right.

The discretisation of $\left(\frac{\partial S}{\partial x} \right)_{z=cstt}$ can be written as :

$$\frac{1}{2\Delta x} (S_{i+1}^* - S_i + S_{i+1} - S_i^*)$$

The $\frac{\partial P(x, S)}{\partial S} \left(\frac{\partial S}{\partial x} \right)_{z=cstt}$ term can therefore be written as :

$$\frac{C_{i+1/2}^2(\tilde{S})}{2\Delta x} (S_{i+1}^* - S_i + S_{i+1} - S_i^*)$$

\Rightarrow Treatment of the term $\left(\frac{\partial P(x, S)}{\partial x} \right)_{S=cstt}$

This source term does not depend on the state variables but only on the geometry. It represents variations in pressure due to the geometry. Furthermore, the pressure is only defined at the interfaces in x (see discretisation of the momentum). The derivative of the term $P(x, S)$ with constant wetted cross-section is therefore only meaningful inside the cell and this term will be treated in a centred way.

In the centre of a cell, the wetted cross-section and the free surface elevation are constant, function $\left(\frac{\partial P}{\partial x} \right)_z$ is approximated by the relation :

$$\left(\frac{\partial P}{\partial x} \right)_z = \frac{P_{i+1/2}(S_i) - P_{i-1/2}(S_i)}{\Delta x}$$

The discretisation introduced above preserves a horizontal water level at rest.

Note :

The term discretised in the centre of the cell is non-null only when the geometry has variations in width. In the case of a rectangular channel with bottom elevation variations only, this term will therefore be equal to zero.

B) Friction term

As introduced in previous sections, this term can be expressed in the form $-gSJ$ where J is given by the Strickler formulation :

$$J = \frac{Q|Q|}{K^2 S^2 R_h^{4/3}} \quad (3.70)$$

\Rightarrow Explicit computation of the friction

Initially the representation of friction is upwinded following the method described earlier. This term is therefore by design discretised at the interface; each variable is then computed as the average of the two cells to which the interface belong.

This term is then upwinded using the method described in section 3.3.3. It will contribute to the source terms for the downstream half of cell i and for the upstream half of cell $i + 1$. This method allows a treatment of the friction term consistent with that of the bottom elevation gradient term.

In particular, convergence towards a flow rate constant in space is obtained when a steady state is required. The second criterion is therefore achieved.

However, this treatment poses problem for computations in shallow water depths and with small Strickler coefficients. The explicit treatment of friction can cause non-physical changes in direction during a time step.

An implicit treatment of friction is thus essential for the particular application of wetting and drying (e.g. flood wave propagation over a dry ground). Furthermore the implicitation of the friction term has the advantage that it stabilises the solution for fast flows such as flood waves. This implicit method, however, did not give convergence towards a steady state; it was therefore decided to offer the two alternatives (explicit or implicit types of discretisation) to the users. The implicit treatment of friction is described in the following paragraph.

⇒ Implicit computation of friction

The implicit contribution of friction can be computed with a split step method [31]. In a first step, the $U^{n+1/2}$ state is computed from all the terms in the equations but the friction term. In a second step, friction is taken into account.

It is important to note that only the flow rate will be modified at this stage, since friction is not in the continuity equation. With $S^{n+1} = S^{n+1/2}$ the equation to be solved for the flow rate can be written as :

$$\frac{Q_i^{n+1} - Q_i^{n+1/2}}{\Delta t} = -g \frac{Q_i^{n+1} |Q_i^{n+1}|}{K_i^2 S_i^{n+1} R_i^{n+1/4/3}} \quad (3.71)$$

This is a second order equation in Q_i^{n+1} , which form varies depending on its sign. Its resolution yields a single solution of the same sign as $Q_i^{n+1/2}$:

$$Q_i^{n+1} = \begin{cases} \frac{-1 + \sqrt{1 + 4aQ_i^{n+1/2}}}{2a} & \text{if } Q_i^{n+1/2} > 0 \\ \frac{-1 - \sqrt{1 - 4aQ_i^{n+1/2}}}{2a} & \text{if } Q_i^{n+1/2} < 0 \end{cases} \quad (3.72)$$

with :

$$a = \frac{g\Delta t}{K_i^2 S_i^{n+1} R_i^{n+1/4/3}} \quad (3.73)$$

Note :

The implicit treatment of friction could be carried out in an off-centred way. This solution was not adopted here because it meant solving a linear system, which had a negative impact on the performance of the solver (in terms of runtime).

C) Inflows

Inflows are linear contributions in the continuity equation. In actual facts, these contributions generally represent tributaries, and therefore an input of flow regarded as punctual. The following methodology is adopted: it is assumed that the input of flow is applied at the interface $i + 1/2$ between two cells. A linear inflow, assumed constant in the downstream half of cell i and the upstream half of cell $i + 1$ is estimated by the relation :

$$q_{a_{i+1/2}} = \frac{q_a}{x_{i+1/2} - x_{i-1/2}} \quad (3.74)$$

This term is then upwinded according to the method introduced in section 3.3.3.

3.3.4 Boundary conditions

Analysis using the characteristics method

The characteristics method applied to the Saint-Venant equations makes it possible to determine the precise number of boundary conditions required such that the initial problem is well set out. The following four cases are possible depending on the nature of the input and output flows.

A) Upstream subcritical flow

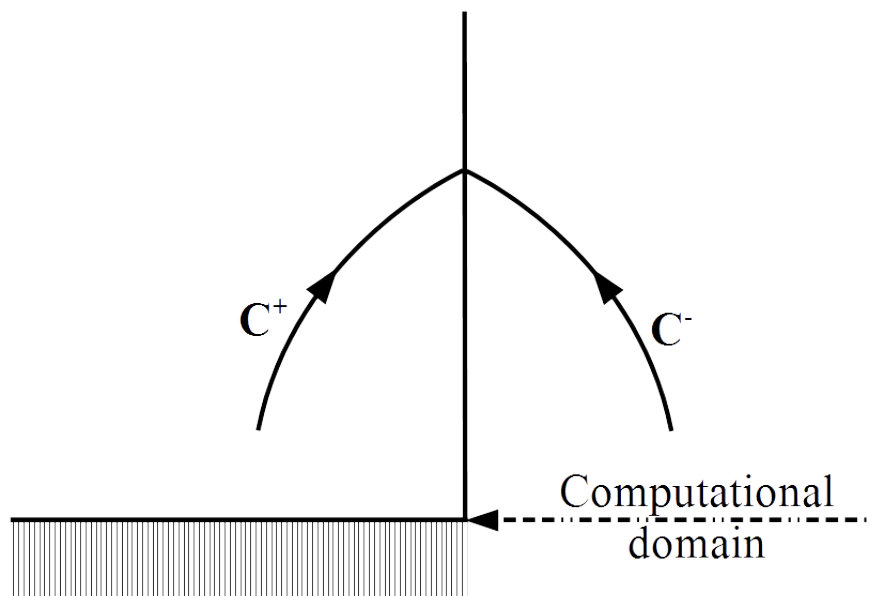


Figure 3.10: Upstream subcritical flow

Only one of the characteristics comes from outside the domain. The other characteristic (characteristic C^- in this case) comes from inside the domain and carries information to the boundary of the domain. It is therefore necessary to impose the elevation or the flow but not both variables.

B) Upstream supercritical flow

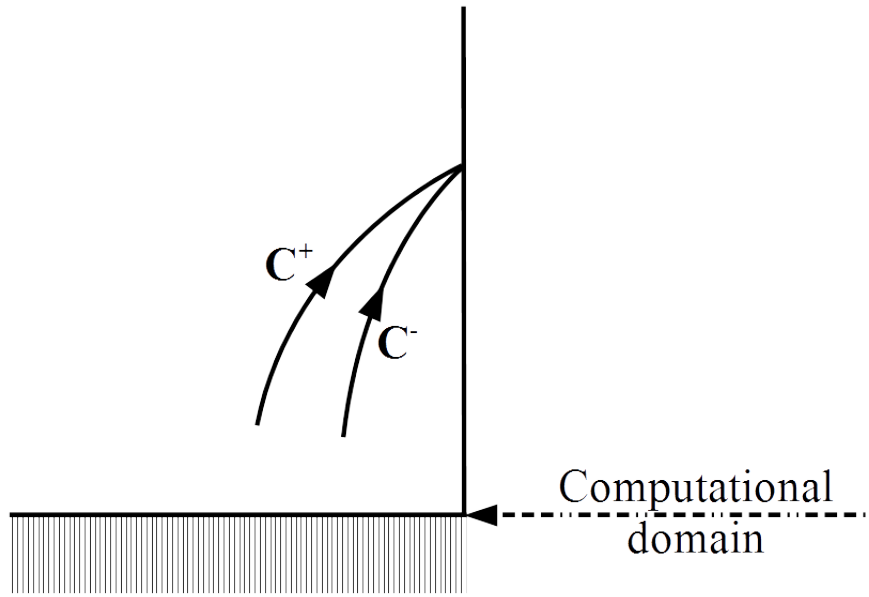


Figure 3.11: Upstream supercritical flow

Both characteristics come from outside the domain. In this case it is necessary to impose the elevation and the flow.

C) Downstream subcritical flow

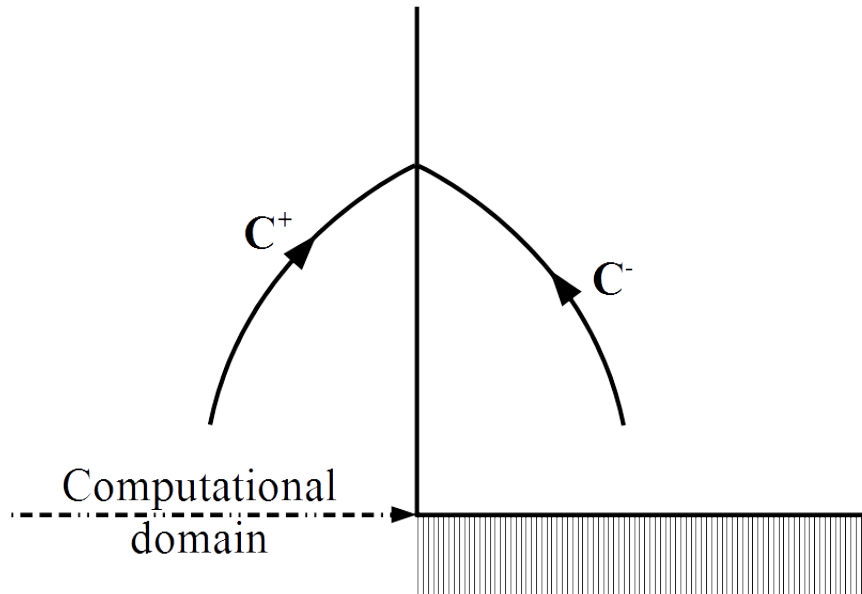


Figure 3.12: Downstream subcritical flow

These are the same conditions as for upstream subcritical flow. Only one of the variables therefore has to be imposed (either the elevation or the flow). In practice, the elevation is often chosen.

D) Downstream supercritical flow

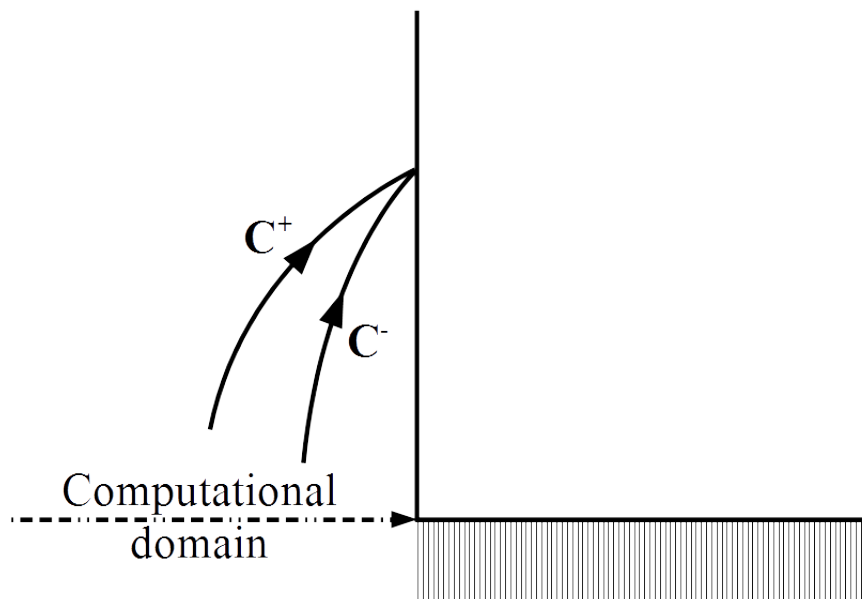


Figure 3.13: Downstream supercritical flow

Both characteristics come from inside the domain. No extra information is required.

The following table summarises the scenarios described above.

Table 3.1: Type of flow and upstream/downstream boundary conditions

	Subcritical	Supercritical
Upstream	Elevation or Flow	Elevation and Flow
Downstream	Elevation or Flow	nothing

Modelling the boundary conditions

The numerical scheme defined in section 3.3.1 assumes that the values for the two variables are known at every time step at the boundary of the domain. This, however, can be conflicting with the scenarios introduced in the previous section and following from the characteristics method.

Initially, it is assumed that the boundary conditions are entirely known. The following will focus on how they are taken into account in the overall scheme.

Let us consider a given flow rate applied upstream of the domain.

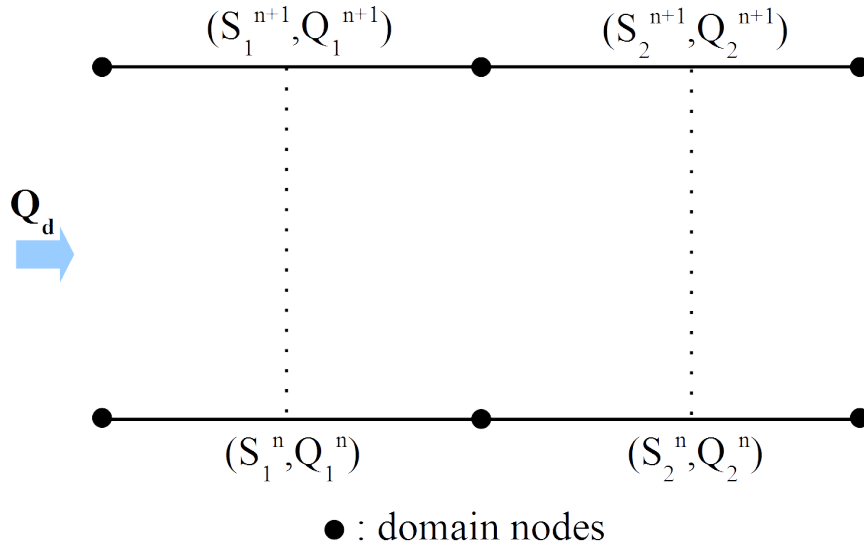


Figure 3.14: Downstream supercritical flow

Let us suppose that all the variables are known at the time step n .

All the variables at all the nodes in the domain, except those at the boundary, can be evaluated at the time step $n + 1$ with the Roe scheme. There are two options to compute (S_1^{n+1}, Q_1^{n+1}) :

- (S_1^{n+1}, Q_1^{n+1}) is computed using the Roe scheme, where the conditions to the left are given by the boundary conditions (Q_r, S_r) . It is important to note that the wetted cross-section S_r is not provided by the user but is computed. The following section will introduce how the missing information can be computed;
- the first half of the cell is not considered as a cell in the finite volume definition. (S_1^{n+1}, Q_1^{n+1}) is set to be equal to the boundary condition (Q_r, S_r) . As was the case before, the wetted cross-section will be computed separately.

The second method was selected to treat the boundary conditions since it guarantees that the values imposed by the user are found at the first mesh node. This is not guaranteed with the first method.

The various methods available in MASCARET to calculate the boundary condition information, independently from the finite volume scheme, are presented in the following sections.

A) Computation of the boundary conditions using the Riemann invariants

The principle of this method is very simple and based on the non-conservative form of the Saint-Venant equations. It is based on the assumption that the solution is continuous in the immediate vicinity of the boundary conditions. Four possible scenarios can be identified (refer to table 3.1) depending on the nature of the flow and on whether considering the upstream or downstream sections.

⇒ UPSTREAM BOUNDARY OF THE DOMAIN

→ Supercritical flow

The two characteristics enter the domain: it is therefore necessary to provide both the value of the wetted cross-section and that of the flow.

→ Subcritical flow

The characteristic C^- leaves the domain and carries information from inside the domain towards the upstream boundary condition. This information, the Riemann invariant f^- , will be used to supplement the data (wetted cross-section or flow) that the user will have to provide.

This computation is detailed as follows.

Q_1^{n+1} or S_1^{n+1} , which are known (provided as input), give a first relationship.

The Riemann invariant f_1^+ is constant along the characteristic curve C^- resulting from the upstream boundary of the model. This yields a second relationship.

The following system is to be solved :

$$\begin{cases} Q_1^{n+1} \text{ or } S_1^{n+1} \text{ known} \\ \frac{Q_1^{n+1}}{S_1^{n+1}} + K(S_1^{n+1}) = f_1^+ \end{cases} \quad (3.75)$$

- Imposed flow : the second equation is multiplied by S_1^{n+1} to determine S_1^{n+1} , giving :

$$S_1^{n+1} K(S_1^{n+1}) - f_1^+ S_1^{n+1} + Q_1^{n+1} = 0 \quad (3.76)$$

This last equation is solved using a Newton method.

- Imposed elevation : the computation is much simpler.

$$Q_1^{n+1} = -S_1^{n+1} K(S_1^{n+1}) + f_1^+ S_1^{n+1} \quad (3.77)$$

⇒ DOWNSTREAM BOUNDARY OF THE DOMAIN

→ Supercritical flow

Both characteristics leave the domain. No additional information is therefore necessary. The following system is solved :

$$\begin{cases} \frac{Q_N^{n+1}}{S_N^{n+1}} + K(S_N^{n+1}) = f_N^+ & \text{with } f_N^+ \text{ known} \\ \frac{Q_N^{n+1}}{S_N^{n+1}} - K(S_N^{n+1}) = f_N^- & \text{with } f_N^- \text{ known} \end{cases} \quad (3.78)$$

→ Subcritical flow

This is similar to the upstream subcritical flow scenario.

Note :

The computation of the boundary conditions using the Riemann invariants is satisfactory and respects the theory of the characteristics. Studies have, however, indicated that this method gives significant errors in cases of small Froude number flows. This is attributed to the fact that the dominant term in the equation is then $SK(S)$ where $K(S)$ is a non-linear function which had to be tabulated (vertical discretisation). To mitigate this major shortcoming for subcritical flows, other methods have been introduced, which are based on a finite difference discretisation of the continuity or momentum equations.

B) Computation of the boundary conditions using the Saint-Venant equations

This is only relevant for subcritical flows.

⇒ UPSTREAM BOUNDARY OF THE DOMAIN

- Imposed flow : the continuity equation is discretised in the simplest possible way using finite differences, to give :

$$Q_1^{n+1} = Q_r \quad (3.79)$$

and :

$$\frac{S_1^{n+1} - S_1^n}{\Delta t} + \frac{Q_2^n - Q_1^n}{\Delta x} = 0 \quad (3.80)$$

S_1^{n+1} is computed using the previous equation.

- Imposed elevation : the momentum equation is discretised in a similar way to that used to discretise the continuity equation.

⇒ DOWNSTREAM BOUNDARY OF THE DOMAIN

There is no added difficulty in this case. The approach is similar.

A parameter, called *limit Froude*, which is imposed by the user (refer to user manual) determines the choice of the method to compute the boundary conditions. If the Froude number, upstream or downstream of the domain, is greater than this parameter, the missing information for the boundary conditions will be computed using the Riemann invariants.

C) Representation of a stage-discharge curve

So far only the simplest cases have been considered where the elevation or the flow is imposed. For real applications, however, a very useful boundary condition is a *stage-discharge curve* : the elevation and the flow are not known explicitly but are interlinked with a relationship $Z = f(Q)$. The system with two equations and two unknowns is solved to obtain the value of the flow and of the elevation at the domain boundaries. The only difference with the previous cases is that the first equation is not straightforward any more.

D) Representation of a free downstream flow

The methods introduced in the previous sections make it possible to compute missing information in the case of subcritical flows but it is still necessary to provide at least the elevation or the flow. This is difficult for flood wave computations where it is impossible for the user to know the elevation or the flow at the downstream boundary of the domain. To address this, it was tried to represent a *free downstream flow*. In this case the user does not provide any boundary condition, these are computed using values from the previous time step only. It is clear that this representation is not correct in consideration of the characteristics method applied to the Saint-Venant equations. The purpose of this representation is not to provide a correct flow for the boundary condition but to let the flood wave leave the domain without influencing too much the flow within the domain.

To obtain a *free downstream flow*, the two variables wetted cross-section and flow rate are moved along at the flow speed.

For flood wave computations it is recommended that the downstream boundary condition for the computational domain is located at least 30 km downstream of the last computational point in order not to disrupt the characteristics at this point. It is strongly advised not to use this type of boundary condition for any other type of computation.

3.3.5 Dry areas

For applications such as flood waves, it is necessary to be able to model dry areas and the propagation of a front on dry areas. From a numerical viewpoint, the following problems arise :

- definition of a flow regime in the dry area because speed is not defined;
- propagation of a front on dry areas.

The main objective with this problem is to calculate a good propagation velocity of the front since this information is paramount to flood wave computations.

To treat this problem with the Roe scheme, a threshold had to be introduced for the water depth. Using this approach the correct propagation velocity was found on simple test cases where an analytical solution exists [11]. The drawback with this approach is that the perfect

conservativity inherent to the finite volumes is then lost. For flood wave computations, the Roe scheme was retained but this entails a strict control of the conservativity.

It should be noted that, within the IAHR work group on numerical methods for flood waves on dry ground, all the teams face the same difficulties. One of the only schemes that does not have this problem is the so-called equilibrium scheme, developed at the university of Bordeaux by Professor Leroux's team [7][3].

It will be necessary to consider other approaches than that of the threshold depth in the future, but it is a satisfactory solution in the meantime.

3.4 Representation of more complex physical phenomena

3.4.1 Introduction to the modelling of compound channels

The transcritical engine of MASCARET treats flows in compound channels made up of a low flow channel (river bed) and a high flow channel (floodplain), but storage zones are not considered.

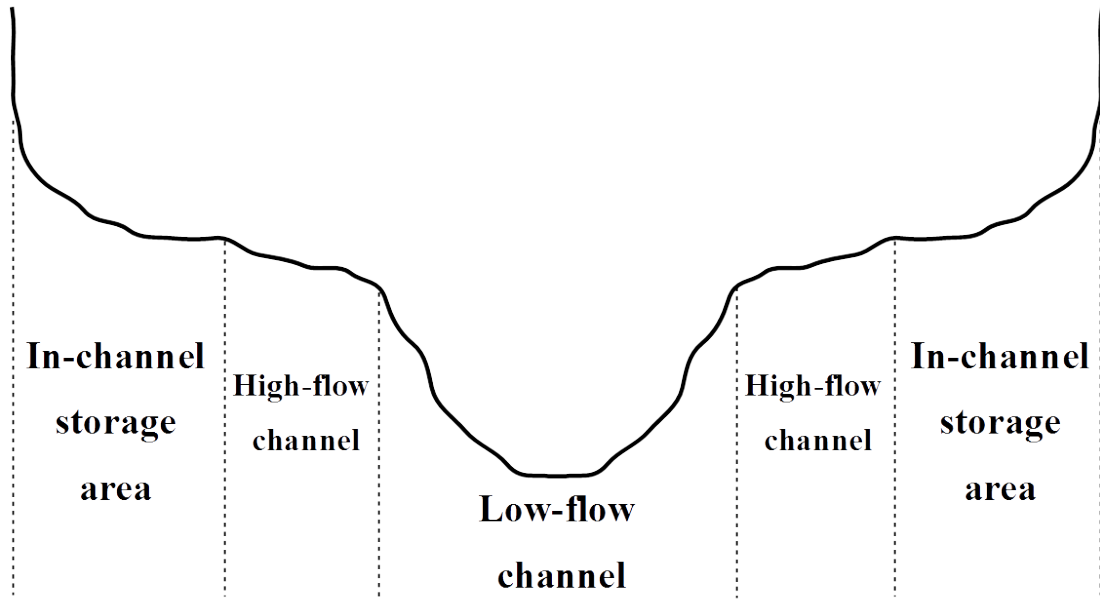


Figure 3.15: Channels and storage sections

The *Debord* formulation was selected to model the low flow and high flow channels such that the subcritical and transcritical engines are as much compatible as possible.

Reminder

The system of equations to solve can now be written as :

$$\left\{ \begin{array}{l} \frac{\partial S}{\partial t} + \frac{\partial Q}{\partial x} = q_l - \frac{\partial S_s}{\partial t} \\ \frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q_m^2}{S_m} + \frac{Q_M^2}{S_M} + P \right) \\ = g \int_0^y \left(\frac{\partial S}{\partial x} \right)_y dy - g S \frac{dZ_f}{dx} - g (S_m J_m + S_M J_M) \end{array} \right. \quad (3.81)$$

with :

- q_I the inflow and S_s the storage section;
- $S = S_m + S_M$ and $Q = Q_m + Q_M$;
- $\sqrt{J_m} = \frac{Q_m}{D_m}$ and $\sqrt{J_M} = \frac{Q_M}{D_M}$;
- $D_m = K_m S_m R_m^{2/3}$ and $D_M = K_M S_M R_M^{2/3}$ are the conveyances in the river bed and in the floodplain respectively, and depend on the free surface elevation.

This system is similar to the Saint-Venant equations in the case of a single channel with the overall slope J defined by means of the relation :

$$SJ = S_m J_m + S_M J_M \quad (3.82)$$

and with the β coefficient no longer equal to 1 (as in a river bed), but satisfying the relation :

$$\beta \frac{Q^2}{S} = \frac{Q_m^2}{S_m} + \frac{Q_M^2}{S_M} \quad (3.83)$$

Note :

the β coefficient is greater than or equal to 1.

The *Debord* formulation

The system of equations (3.81) is not complete and a closing relation is necessary. Various solutions have been proposed to represent the composition of the two channels. The *Debord* formulation was eventually retained to be compatible with the subcritical engine (refer to section 2.1.3) and following feedback from various applications of this formulation.

Given :

$$\eta = \frac{Q_m}{Q_M} \quad (3.84)$$

With the *Debord* formulation :

$$\eta = \frac{A}{\sqrt{1 + \frac{S_m}{S_M}(1 - A^2)}} \frac{D_m}{D_M} \quad (3.85)$$

where A is a constant of the *Debord* formulation, computed with :

$$\begin{cases} A = \frac{1-A_0}{2} \cos\left(\frac{\pi r}{0.3}\right) + \frac{1+A_0}{2} & \text{for } r = \frac{R_M}{R_m} \in [0, 0.3] \\ A = A_0 = 0.9 \left(\frac{K_m}{K_M}\right)^{-1/6} & \text{for } r > 0.3 \end{cases} \quad (3.86)$$

Implementation in the transcritical engine

The equations for compound channels only differ from those for single channels by the β term and the form of the source term related to friction. The resolution of this system is therefore formally identical to that treated previously.

The *Debord* formulation makes it possible to write β in the form $\beta(S, x)$. This gives an additional unknown function.

For simplicity, the β function will be taken into account explicitly. The momentum equation to be solved at time $n + 1$ is therefore :

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\beta^n \frac{Q^2}{S} + P \right) = g \int_0^y \left(\frac{\partial S}{\partial x} \right)_y dy - g S \frac{dZ_f}{dx} - g S J \quad (3.87)$$

It is a partial derivative equation for which the β coefficient is variable in time.

The advection system is unconditionally hyperbolic because β is greater than or equal to 1. The Riemann problem is solved in a similar way to that related to the Saint-Venant equations for a single channel; the eigenvalues for the system are modified. They become : $\beta u + c'$ and $\beta u - c'$ with : $c' = \sqrt{\frac{\partial P}{\partial S} + u^2(\beta^2 - \beta)}$. The Roe variables remain unchanged.

The representation of friction in this case is exactly identical to that in the case of a single channel; the generalised conveyance takes into account both the river bed and the floodplain, and can be discretised along the vertical in a similar way to the conveyance of the low flow channel.

Note 1 :

The fact that the β term is considered as a forcing function, varying with each time step, simplifies considerably the implementation of the method. This assumption is partly justified because flows in compound channels are generally evolving slowly.

Note 2 :

Conservation properties are lost when modelling compound channels. These properties made it possible to reach a steady state with constant flow. In all the cases studied to date, variations in the flow of the order of 2 to 3% of the total flow have been observed when a stationary state is reached.

3.4.2 Confluence

A new method has been developed to model confluences. A detailed description can be found in document [28] together with validation test cases. The following paragraph summarises the method.

It can be assumed that the 2D Saint-Venant equations replace the 1D equations to represent the flow locally, at and around the junction. It can then be considered that the total domain (main valley + tributary) is the overlap of two sub-domains, where the first sub-domain consists in three 1D reaches (the one-dimensional Saint-Venant equations are supposed to be valid in this case) and where the second sub-domain is the zone around the junction (the two-dimensional equations are supposed to be valid in that case). Modelling the confluence therefore amounts to performing a 1D-2D coupling between the two sub-domains. The 2D Saint-Venant equations are solved using a finite volume module equivalent to that used in 1D (Roe scheme). This module has numerical schemes similar to those used in MASCARET. This last point together with the fact that such schemes are explicit has the advantage of facilitating the coupling procedure. A two-dimensional model usually requires a fine discretisation of the geometry. For flood wave studies, it would not be practical to create a local 2D model for every tributary in a valley, with detailed representation of the geometry and a fine mesh. A simple tool that automate this process is considered as the best approach. This is why it was decided that the one-dimensional software generates automatically the 2D domain representative of the confluence and the associated mesh. A 12 cells mesh is created by default.

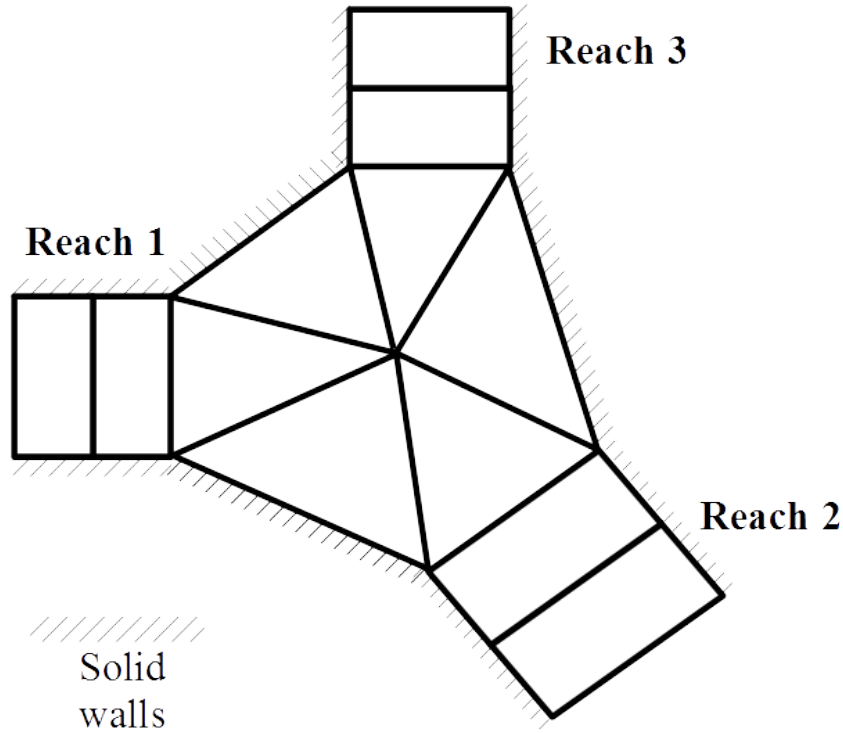


Figure 3.16: 2D confluence

1D-2D coupling

In the one-dimensional model, the aim is to compute the value of (S_1, Q_1) at the boundary of each 1D reach linked to the confluence. This is equivalent to defining the 1D-2D coupling.

The coupling is done by overlapping domains, i.e. the boundary points of the 2D model are located inside the 1D model. For illustrative purposes, the following figure shows the overlap between the 2D model and the three 1D reaches.

It is assumed that the boundaries of the 1D model, where reaches B_1, B_2, B_3 are located, are represented in the 2D model by cells A, B, C respectively. Cells a, b, c , at the boundaries of the 2D model, represent the cells close to the ends of each 1D reach. The 1D-2D coupling is then done by exchange of boundary conditions.

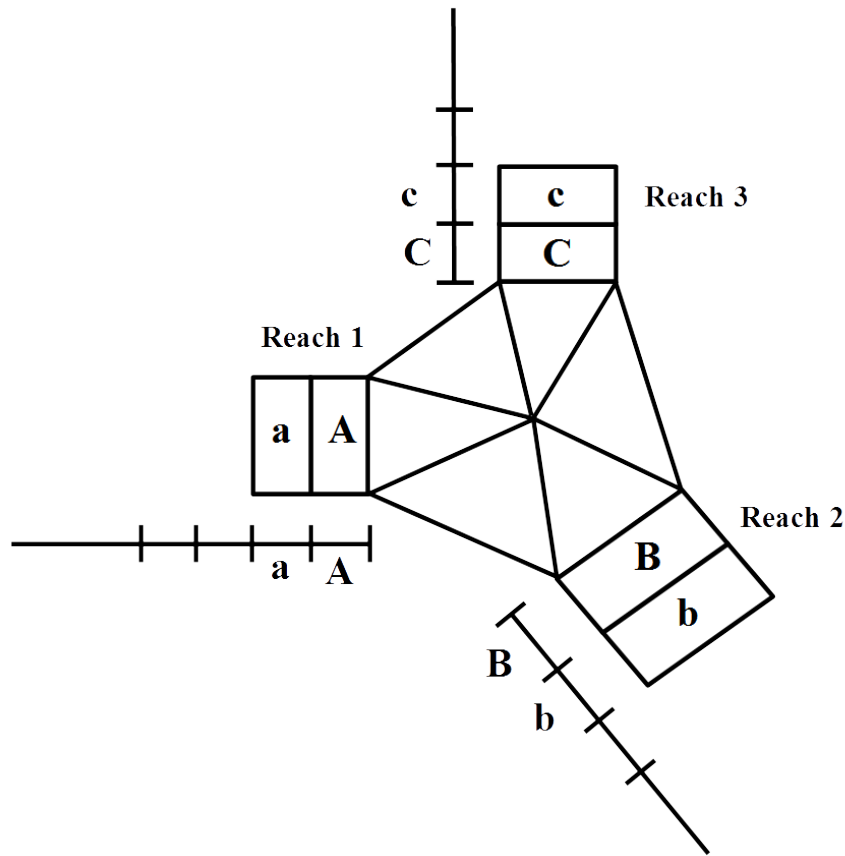


Figure 3.17: 1D-2D overlap

A computation can be described as follows :

The state in each 1D reach: B_1, B_2, B_3 and in the 2D confluence is assumed to be known at time n . The state at time $n + 1$ is determined in four steps :

- **Step 1** : computation of the state at the confluence (except for the three boundary cells a, b and c). This stage is carried out with a 2D Roe scheme. The scheme is explicit; the boundary conditions at time $n + 1$ are not required, only the conditions for state n everywhere;
- **Step 2** : computation of the state inside each reach (except for boundary cells A, B, C). This stage is carried out with a 1D Roe scheme for the same reasons and in the same way as previously discussed;
- **Step 3** : computation of S_1, Q_1 in each reach. These are the values of S and Q in A, B, C . These values have been computed at step 1. The free surface level is maintained from one model to the other. The computed flow values, however, are two-dimensional. The flow vector is therefore projected along the normal to the free boundary in order to obtain a one-dimensional value.
- **Step 4** : computation of the boundary conditions for the 2D model. These were determined at step 2 in the cells adjacent to the ends of each reach. Similarly to the previous step, the free surface level is maintained from the 1D model to the 2D model. Furthermore, the flow direction is assumed to be that of the normal to the free boundary in order to have a two-dimensional flow.

Finally, the complete state of the one-dimensional model (reaches B_1, B_2, B_3) and of the two-dimensional model (the confluence) is computed at time $n + 1$.

The use of a finite volume scheme in the 2D model makes modelling of the confluence a priori conservative. This is not entirely true because a projection of the flow is used to go from the 2D state to the 1D state and this introduces a source of non-conservatism to the momentum.

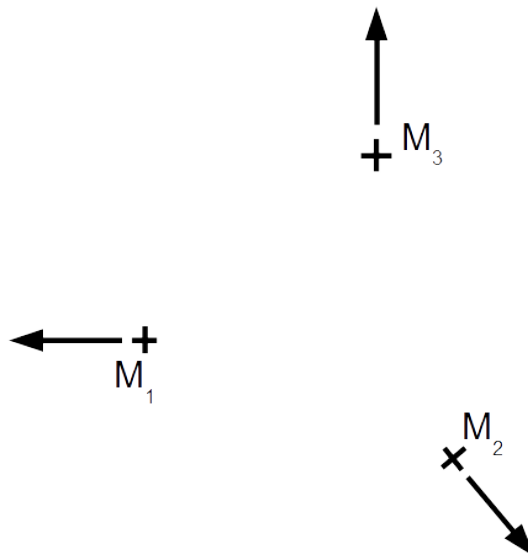
Adapting the 1D geometry to the 2D model

Now that the coupling method has been defined, all that remains is to describe the 2D model selected to represent the junction. Initially, the simple case of rectangular channels is considered. Then the method is extended to real geometries.

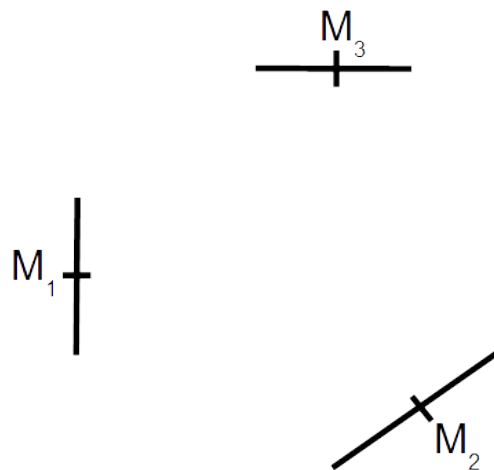
⇒ Rectangular channels

The geometry of the junction is discretised using twelve cells such that it is reasonable close to the real geometry.

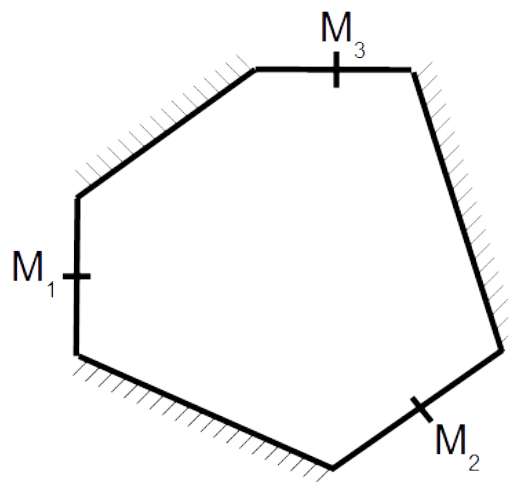
- * For each tributary, the coordinates of the boundary point on the hydraulic axis are provided, as well as the angle this 1D reach forms with a set direction. These are the only external data specific to the confluence. They are determined only from geometrical criteria;



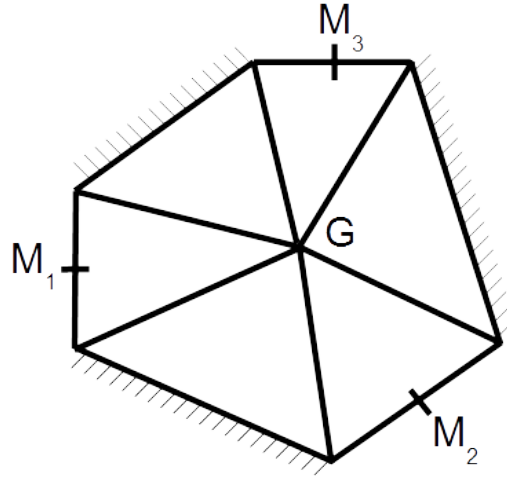
- * for each tributary i , a segment can be traced with length L_i (the width of the channel, independent from the definition of the confluence, and coming directly from the geometry of the 1D model). This segment is normal to the angle previously set;



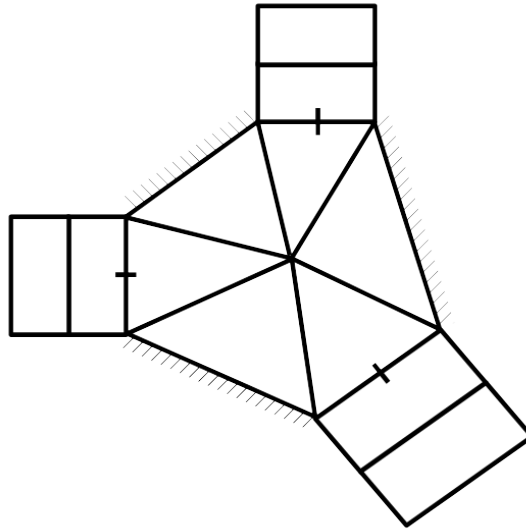
- * these segments are then joined up by walls. The three segments previously defined together with the walls form a hexagon at the junction;



- * G is the centre of gravity of the hexagon. Inside the hexagon, six triangles are formed with G as a common vertex;



- * the cells in the 1D-2D overlapping area are rectangles with sides L_i and DX_i , the size of the 1D cell in each reach;



- * last but not least, the bottom elevation in each cell is set as follows :
- for the six overlapping cells, it is the elevation of the 1D cell, Z_{f1D} ;
 - inside the hexagon, the elevation of G (Z_{fG}) is taken to be the arithmetic mean of those in the three neighbouring 1D-2D overlapping cells. The three cells that have an edge in common with the overlapping cells (thus no walls) have an elevation Z_{f2D} defined by : $Z_{f2D} = \frac{2Z_{f1D} + Z_{fG}}{3}$ where Z_{f1D} is the elevation of the adjacent 1D cell;
 - for each of the other three cells, the bottom elevation is assumed to be the average of the elevation at the two adjacent cells.

This parameterisation has the advantage that it only requires few specific parameters: the coordinates of points M_i and the relative orientation of each reach. These are purely geometrical

parameters, which are easy to determine. The remainder of the information comes directly from the 1D parameterisation (geometry of the cross-reach profiles) and from the state of each reach. The geometry thus defined takes into account :

- the importance of the extent of the zone of confluence, through the relative position of the three points M_i ;
- the orientation of each reach, through the angles provided for each M_i ;
- the width of each channel, since it is transferred in the 2D model from the 1D state;
- the bottom variations, since the elevation of the overlapping cells is that of the 1D cells. Inside the hexagon, the bottom elevations vary smoothly.

⇒ Real geometry channels

A difficulty arises in the case of a real geometry. A cross-section will be discretised using only one cell in the 2D model. This 2D cell represents a rectangular profile since the bottom of a cell is flat with the P_0 discretisation used here. This point was obviously trivial in the previous case, but must now be addressed specifically.

The problem is to find the rectangular geometry equivalent to the real geometry, in the sense that the flows represented in each model must be as close as possible. A 1D-2D transformation must therefore be defined and the following was retained :

- the bottom elevation Z_f is preserved;
- the free surface elevation Z is preserved (thus Y , the water depth, is also preserved);
- the discharge Q is preserved;
- the velocity V is preserved.

It is thus necessary to preserve the wetted cross-section S . The depth being fixed by the first two points, the width can be deduced from :

$$L_{2D} = \frac{S_{1D}}{Y} \quad (3.88)$$

This cell width is likely to change when the depth varies. The grid must therefore be updated in a regular way at each time step.

3.4.3 Singularities

When a flood wave propagates in a real valley, dams located downstream of the main dam need to be taken into account. These structures can respond in several possible ways: instantaneous failure when the wave arrives or spilling over the crest of the structure. In the second case, the dam can be modelled in two different ways :

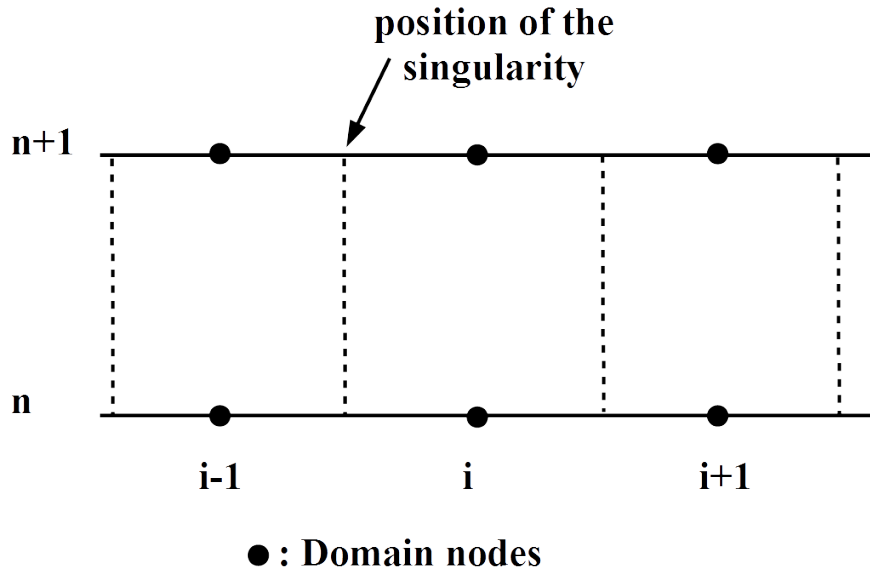
- direct representation in the geometry. This solution, however, would require a very fine grid in the immediate proximity of the structure since the slopes of the faces are steep upstream and downstream of the structure. This would strongly constrain the time step for the whole computational domain;
- representation as a singularity. With this option the dam is not treated as a regular node in the numerical scheme. Instead, a specific relationship relates the water upstream and downstream of the dam.

The method used to model a singularity in the case of a finite volume scheme will be introduced in the remainder of this section.

It is considered that a singularity is defined by a relation linking the conveying flow Q_s through this singularity to the upstream and downstream water levels. This relation depends on the physical and hydraulic characteristics of the structure.

A part of the domain is represented in the figure below.

A singularity has to be located at the interface between 2 cells. The flow through this interface is defined by : $Q_s = f(Z_{i-1}, Z_i)$.



With a finite volume scheme, the hydraulic state in cells i and $i - 1$ at time $n + 1$ depends on the fluxes at the interface of both cells.

For cell i (respectively $i - 1$), the flux to the right (respectively to the left) will be that computed by the scheme defined in section n . The mass and momentum fluxes at the interface between cell i and cell $i - 1$, however, will not be that of Roe, but the analytical flux explicitly computed using the relation defining the singularity.

For cell i

Mass flux to the left : $Q_s = f(Z_{i-1}^n, Z_i^n)$

Momentum flux to the left : $\frac{Q_s^2}{S_i^n} + P(S_i^n)$

For cell $i - 1$

Mass flux to the right : $Q_s = f(Z_{i-1}^n, Z_i^n)$

Momentum flux to the right : $\frac{Q_x^2}{S_{i-1}^n} + P(S_{i-1}^n)$

Note :

- mass is therefore conserved. Momentum, however, is no longer conserved;
- at the very beginning of the spill, the wetted cross-section downstream of the singularity is very small. This implies very strong flows and consequently numerical problems (instabilities). In this situation, only the pressure term is kept in the momentum flux equation;
- this representation allows all types of singularities to be taken into account. All that is required is the function linking the discharge to the upstream and downstream elevations.

3.4.4 Eroding dams

The transcritical engine of MASCARET makes it possible to model erosion, by overflow, of the gravity dams located downstream of the principal dam.

The selected method is largely inspired from that implemented in the EROSIF solver [33]. EROSIF computes the evolution of a breach in an embankment as well as the downstream hydrograph resulting from the progressive failure.

Within MASCARET, a dam eroding by overflow is treated as a spilling dam with a crest elevation variable in time.

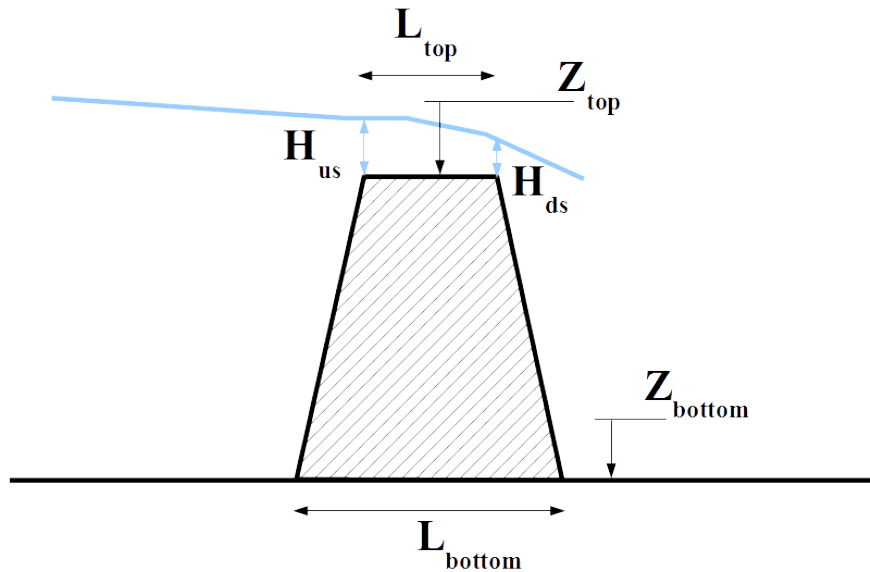


Figure 3.18: Modelling notations

Upstream of the dam, the hydraulic conditions are assumed to be identical to those computed by the transcritical engine in the cell immediately upstream of the dam at the previous time step. Downstream of the dam, a critical condition is imposed. The flow across the whole structure is assumed to be equal to the upstream flow.

The evolution of the elevation of the dam crest is given by the equation :

$$(1 - p) \frac{dZ_{top}}{dt} = - \frac{dQ_s}{dx} = - \frac{Q_{s_{d/s}} - Q_{s_{u/s}}}{L_{top}} \quad (3.89)$$

where p is the porosity of the material (usually taken as 0.4), $Q_{s_{u/s}}$ et $Q_{s_{d/s}}$ are the sediment transport rate respectively upstream and downstream of the erosion channel, and calculated using the of Meyer Peter Muller formulation, i.e. :

$$Q_s = \frac{8}{g\sqrt{\rho}(\rho_s - \rho)} (\tau - \tau_c)^{1.5} \quad (3.90)$$

with :

- τ the global hydrodynamic stress (Manning Strickler formulation);
- τ_c the critical shear stress : $0.05 \cdot (\rho_s - \rho) \cdot g \cdot d_{50}$;
- ρ_s the density of the sediment;
- ρ the density of water;
- d_{50} the median diameter of the material.

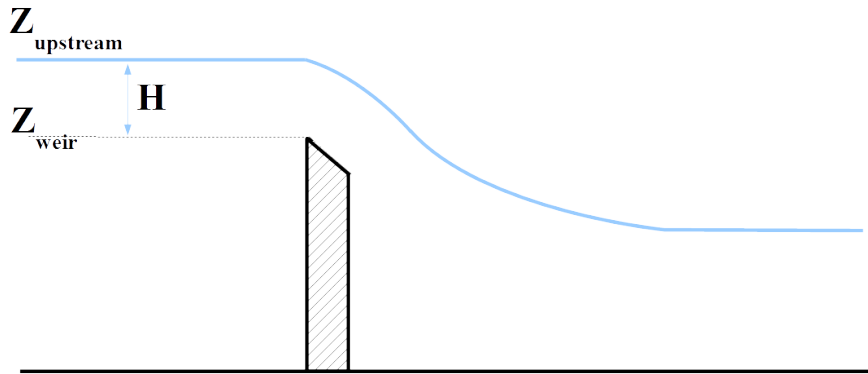
All these equations make it possible to compute Z_{top} , the crest elevation. Erosion is limited at the bottom by the elevation Z_{low} .

3.4.5 Broad-crested and sharp-crested weirs

This section presents the methodology for modelling singularities of the type broad-crested or sharp-crested weirs, as implemented in the transcritical and subcritical engines. For the other singularities, the pair (flow, elevation) is directly determined by interpolation from a discretised table, provided by the user.

The laws for weirs implemented in the transcritical and subcritical engines give the user a choice between two types of weirs: broad-crested or sharp-crested weirs. These two laws differ only when the weir is drowned, in particular with the flow conditions and the drowning coefficient. The validation tests carried out in the case of a sharp-crested weir are detailed in [17].

Broad-crested or sharp-crested free flow weir



The flow over a broad-crested or sharp-crested free flow weir can be expressed as :

$$Q = CLH \sqrt{2gH} \quad (3.91)$$

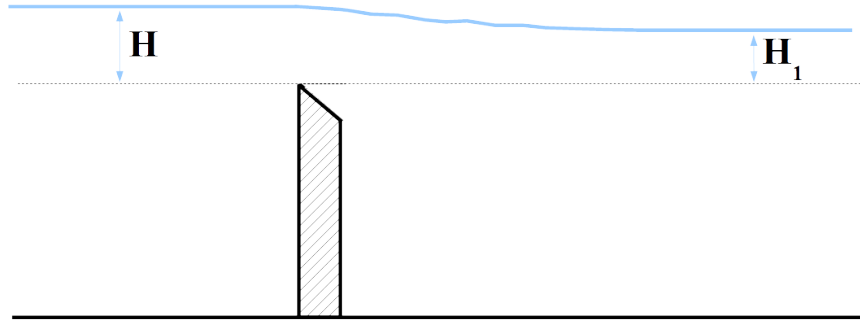
where:

- H is the water head upstream of the weir (the approach velocity is ignored) : $H = Z_{u/s} - Z_{weir}$;
- g is the gravity;
- L is the width of the weir;
- C is a discharge coefficient which depends on contraction effects and on approximations made when writing the formula (3.91) : for example the fact that the approach velocity is not accounted for. This coefficient, C , is determined by the user and can result from a calibration of the model.

More details on how the previous formula was established are given in [5].

The laws are identical when the weirs are not drowned. When they are drowned, however, the laws differ with the type of weir (conditions and drowning coefficient).

Drowned weir



A weir is considered drowned when the upstream elevation is influenced by the downstream elevation. The discharge for a broad-crested or sharp-crested drowned weir can be expressed in a similar way to that for a drowned weir. Only the discharge coefficient, C , is modified to take into account the influence of the downstream reach on the discharge over the weir. A drowning coefficient, K , is defined as :

$$K = \frac{Q_{\text{drowned flow}}}{Q_{\text{free flow}}} \quad (3.92)$$

The discharge over the drowned weir is therefore expressed as :

$$Q = KCLH \sqrt{2gH} \quad (3.93)$$

where :

- K is a coefficient which depends on the ratio $\frac{H_1}{H}$ and the type of weir;
- $H = Z_{u/s} - Z_{weir}$ et $H_1 = Z_{d/s} - Z_{weir}$;
- C is the discharge coefficient defined above for the drowned weir.

⇒ Broad-crested drowned weir

For a broad-crested weir, the downstream elevation does not influence the discharge over the sill when H_1 is lower than $0.8H$ (even though the elevation of the downstream reach may be

higher than the weir crest elevation). This is linked to the presence of a critical section over the weir.

The drowning coefficient K in the formula (3.93) is that typically used for broad-crested weirs. It can be expressed as :

$$\begin{cases} K = -25 \left(\frac{H_1}{H}\right)^2 + 40 \left(\frac{H_1}{H}\right) - 15 & \text{if } \frac{H_1}{H} > 0.8 \\ K = 1 & \text{if } \frac{H_1}{H} < 0.8 \end{cases} \quad (3.94)$$

⇒ Sharp-crested drowned weir

For a sharp-crested weir :

- if $H_1 < 0$, $K = 1$: the downstream elevation does not influence the discharge over the weir;
- if $H_1 > 0$, i.e. as soon as the downstream elevation is higher than that of the weir crest, we have :

$$K = \left(\left(1 - \frac{H_1}{H} \right)^{1.5} \right)^{0.385} \quad (3.95)$$

Figure 3.19 gives the shape of the drowning coefficients as a function of $\frac{H_1}{H}$, for both a broad-crested and a sharp-crested weir.

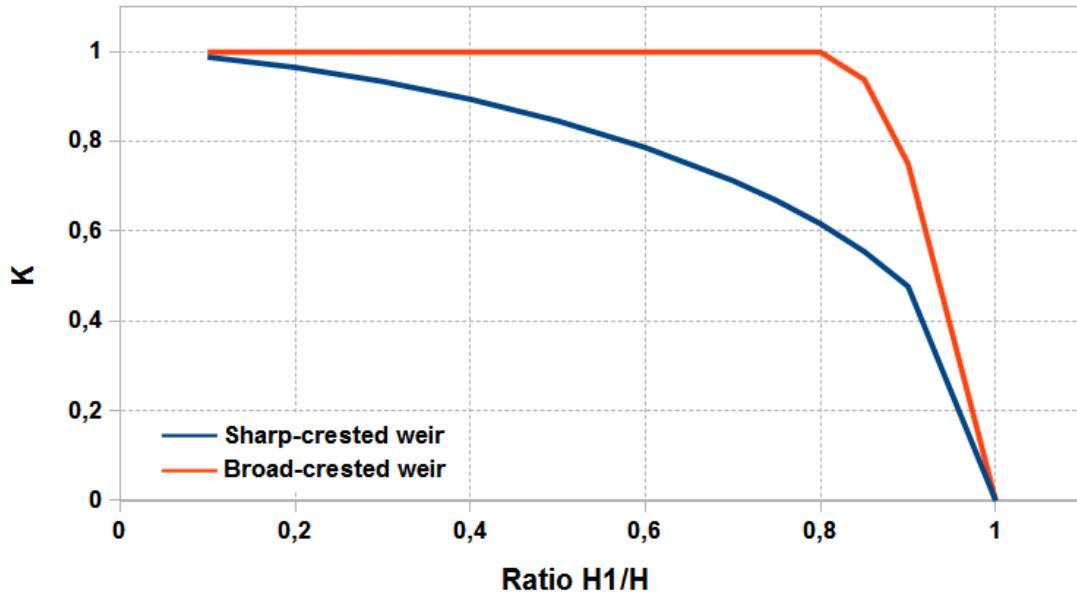


Figure 3.19: Drowning coefficient

3.4.6 Summary of the treatment of singularities

There may exist in a river some regions where the Saint-Venant equations do not apply because the assumptions (for example, uniform distribution of the speeds along the vertical) are no longer valid. Flows over weirs, valves etc... are some examples.

The relationships between the elevation and the discharge in the close vicinity of the singularities are then described by explicit laws. The subcritical engines of MASCARET can model

singularities such as weirs and valves in a river when those are explicitly modelled by hydraulic laws relating the discharge through the singularity, the upstream and downstream elevations and the characteristics of the singularity. Although a number of singularities are implemented in MASCARET, they are not all available for the transcritical engine. The following table lists the various singularities and their availability in the subcritical and transcritical engines.

Table 3.2: Singularities modelled in MASCARET

Type of singularities	Subcritical engine	Supercritical engine
$Q = f(Z_{d/s})$	yes	no
$Q = f(Z_{u/s})$	yes	yes
Law for broad-crested weir	yes	yes
Law for sharp-crested weir	yes	yes
$Z_{u/s} = f(Q)$	yes	no
$Z_{u/s} = f(Z_{d/s}, Q)$	yes	no
Weir with crest defined by multiple points	yes	no
$Z_{u/s} = f(t)$	yes	no
Law for valves	yes	no

3.4.7 Non-hydrostatic terms

This section presents an extension of the Saint-Venant equations, which considers some of the terms introduced when non-hydrostatic effects are taken into account. The aim is to model physical phenomena such as Favre waves (or undular bores). It should be noted that these developments are only valid for subcritical, unsteady flows with a shallow bed slope. This is derived from work by BRISTEAU et al. [27].

The following paragraphs describe the additional terms and their representation within the framework of the transcritical engine in MASCARET (finite volume scheme of Roe type). These developments can also be applied without much difficulty to the REZO engine dedicated to subcritical flows. These were the subject of a publication where the reader will find all the necessary explanations [27].

The Saint-Venant equations are obtained from the Navier-Stokes equations by making the following assumptions :

- long waves (shallow-water);
- hydrostatic pressure;
- uniform distribution of speeds in the vertical.

Let us start from the equations of Euler with 2 dimensions (x, z) to eliminate the constraint of hydrostatic pressure, where z represents the vertical axis and x the horizontal axis. For the purpose of simplicity, friction is neglected.

$$\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} = 0 \quad (3.96)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + w \frac{\partial u}{\partial z} + \frac{\partial p}{\partial x} = 0 \quad (3.97)$$

$$\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + w \frac{\partial w}{\partial z} + \frac{\partial p}{\partial z} = -g \quad (3.98)$$

with : $t > t_0$, $x \in \mathbb{R}$ and $Z_b(x) \leq z \leq Z(x, t)$

$Z(x, t)$ represents the free surface elevation, $\mathbf{u} = (u, w)^T$ is the velocity vector and p the pressure. The water depth is : $h = Z - Z_b$.

The bed elevation Z_b varies with the curvilinear coordinate x of the river.

The system of equations (3.96) to (3.98) is supplemented by the following boundary conditions.

If n_s is the normal to the free surface and n_b the normal to the bed surface :

$$n_s = \frac{1}{\sqrt{1 + \left(\frac{\partial Z}{\partial x}\right)^2}} \begin{pmatrix} -\frac{\partial Z}{\partial x} \\ 1 \end{pmatrix} \quad (3.99)$$

and

$$n_b = \frac{1}{\sqrt{1 + \left(\frac{\partial Z_b}{\partial x}\right)^2}} \begin{pmatrix} -\frac{\partial Z_b}{\partial x} \\ 1 \end{pmatrix} \quad (3.100)$$

At the free surface

The boundary condition at the free surface is a traditional kinematic condition :

$$\frac{\partial Z}{\partial t} + u_s \frac{\partial Z}{\partial x} - w_s = 0 \quad (3.101)$$

where subscript s indicates the value of the variable at the free surface.

At the bottom

$$u_f \frac{\partial Z_b}{\partial x} - w_f = 0 \quad (3.102)$$

where subscript b indicates that the value of the variable is taken at the bottom.

The notations here are the same as those introduced in the previous section about the transcritical engine of MASCARET.

$$S(x, t) = \int_{l_1}^{l_2} h(x, y, t) dy \quad (3.103)$$

$$\bar{u}(x, t) = \frac{1}{S(x, t)} \int_{l_1}^{l_2} \int_{Z_b}^{Z(x, t)} u(x, y, z, t) dz dy \quad (3.104)$$

$$Q(x, t) = S(x, t) \bar{u}(x, t) \quad (3.105)$$

Integration along the vertical of the continuity equation (3.96), with the boundary conditions defined by (3.101) and (3.102), gives :

$$\frac{\partial S}{\partial t} + \frac{\partial Q}{\partial x} = 0 \quad (3.106)$$

Integration of equation (3.98) gives the expression of the pressure.

$$p(x, z, t) = p^a + g(Z(x, t) - z(x, t)) + \int_z^Z \frac{\partial w}{\partial t} dz \quad (3.107)$$

With the assumption that the pressure is hydrostatic, the term which depends on the vertical velocity component is not taken into account. It is also noted that only the term corresponding to the temporal derivative of the vertical velocity component is considered. The advective terms

are here neglected. Integration along the vertical of the momentum equation (3.97) gives the following equation :

$$\frac{\partial Q}{\partial t} + \frac{\partial Q^2}{\partial x} + g \int_{l_1}^{l_2} h \frac{\partial Z}{\partial x} dy + D = 0 \quad (3.108)$$

The friction terms are neglected. D represents part of the non-hydrostatic terms, which are developed in the following section.

Development of the non-hydrostatic terms

$$D = \int_{l_1}^{l_2} \int_{Z_b}^{Z(t)} \frac{\partial}{\partial x} \left(\int_z^{Z(t)} \frac{\partial w}{\partial t} dz_1 \right) dz \quad (3.109)$$

The velocity is such that its divergence equals zero (3.96). Combined with the impermeability conditions at the bottom, the following expressions are obtained :

$$w(z) = -\frac{\partial}{\partial x} \int_{Z_b}^z u dz_1 \quad (3.110)$$

$$= -\frac{\partial}{\partial x} ((z - Z_b) \bar{u}) \quad (3.111)$$

$$= -(z - Z_b) \frac{\partial \bar{u}}{\partial x} + \frac{\partial Z_b}{\partial x} \bar{u} \quad (3.112)$$

If the expression for the vertical velocity component is incorporated in the term D , representing part of the non-hydrostatic terms, and if the Leibnitz rule is applied to (3.109), the following relation is obtained :

$$D = \int_{l_1}^{l_2} \int_{Z_b}^{Z(t)} \left(\int_z^{Z(t)} \frac{\partial^2 w}{\partial x \partial t} dz + \frac{\partial Z(t)}{\partial x} \frac{\partial w}{\partial t} \Big|_s \right) dz dy \quad (3.113)$$

This relation is further developed when replacing the vertical velocity component by its expression (3.112).

$$\begin{aligned} D &= -\frac{1}{2} \int_{l_1}^{l_2} \int_{Z_b}^{Z(t)} \left[h^2 - (z - Z_f)^2 \right] dz dy \frac{\partial^3 \bar{u}}{\partial x^2 \partial t} \\ &+ 2 \int_{l_1}^{l_2} \int_{Z_b}^{Z(t)} (Z - z) \frac{\partial Z_f}{\partial x} dz dy \frac{\partial^2 \bar{u}}{\partial x \partial t} \\ &+ \int_{l_1}^{l_2} \int_{Z_b}^{Z(t)} (Z - z) \frac{\partial^2 Z_f}{\partial x^2} dz dy \frac{\partial \bar{u}}{\partial t} \\ &+ \int_{l_1}^{l_2} \int_{Z_b}^{Z(t)} h \frac{\partial Z}{\partial x} dz dy \frac{\partial^2 \bar{u}}{\partial x \partial t} \\ &+ \int_{l_1}^{l_2} \int_{Z_b}^{Z(t)} \frac{\partial Z}{\partial x} \frac{\partial Z_f}{\partial x} dz dy \frac{\partial \bar{u}}{\partial t} \end{aligned} \quad (3.114)$$

Not all the above terms are taken into account in the transcritical engine of MASCARET. Only the dominant term that does not contain bottom gradient terms is kept:

$$-\frac{1}{2} \int_{l_1}^{l_2} \int_{Z_b}^{Z(t)} \left[h^2 - (z - z_f)^2 \right] dz dy \frac{\partial^3 \bar{u}}{\partial x^2 \partial t} \quad (3.115)$$

The double integral is expressed as follows :

$$= \int_{Z_b}^{Z(t)} \left(\int_{l_1(z)}^{l_2(z)} [h^2 - (z - Z_f)^2] dy \right) dz \quad (3.116)$$

$$= h^2 \int_{Z_b}^{Z(t)} L(z) dz - \int_{Z_b}^{Z(t)} ((z - Z_f)^2) L(z) dz \quad (3.117)$$

$$= h^2 S(x, t) - \int_{Z_b}^{Z(t)} ((z - Z_f)^2) L(z) dz \quad (3.118)$$

In the case of a channel with rectangular cross-section, the previous term is equal to $\frac{2}{3} L h^3$.

The following term is added to the momentum equation of the Saint-Venant system to model some of the hydrostatic terms :

$$\left(h^2 S(x, t) - \int_{Z_b}^{Z(t)} ((z - Z_f)^2) L(z) dz \right) \frac{\partial^3 \bar{u}}{\partial x^2 \partial t} \quad (3.119)$$

The system of the Saint-Venant equations in 1D, for a real geometry, and accounting for the principal hydrostatic terms, can be written as below, when the advective terms are neglected in the expression of the pressure derivative with respect to z (this assumption is justified with an asymptotic development, see [27]).

$$\frac{\partial S}{\partial t} + \frac{\partial Q}{\partial x} = 0 \quad (3.120)$$

$$\frac{\partial Q}{\partial t} + \frac{\partial Q^2}{\partial x} + g \int_{l_1}^{l_2} h \frac{\partial Z}{\partial x} dy + D(x, t, \frac{\partial}{\partial t}, \frac{\partial}{\partial x}) = 0 \quad (3.121)$$

$$\begin{aligned} D(x, t, \frac{\partial}{\partial t}, \frac{\partial}{\partial x}) &= \\ \left(h^2 S(x, t) - \int_{Z_b}^{Z(t)} ((z - Z_f)^2) L(z) dz \right) \frac{\partial^3 \bar{u}}{\partial x^2 \partial t} &= \\ &= \alpha(x, t) \frac{\partial^3 \bar{u}}{\partial x^2 \partial t} \end{aligned} \quad (3.122)$$

Discretisation of the non-hydrostatic terms

This section presents the discretisation of the Saint-Venant system with the addition of the term D defined by equation (3.122).

It shows how the non-hydrostatic terms with third order temporal and spatial derivatives of the velocity can be integrated relatively easily, starting with a numerical scheme of the finite volume type, and using a scheme of the predictor-corrector type.

The prediction step corresponds to the traditional Saint-Venant equations and provides the wetted cross-section at time t^{n+1} . This is then used with the flow in the correction step, taking the non-hydrostatic term into account. The finite volume scheme developed in this section is used for the Saint-Venant part of the equations.

It is assumed that the computational domain is discretised with I nodes x_i .

C_i is the cell of length $\Delta x_i = x_{i+1/2} - x_{i-1/2}$ with $x_{i+1/2} = (x_i + x_{i+1})/2$. X_i represents the average value of X in cell i .

For the temporal discretisation, $t^n = \sum_{k < n} \Delta t^k$ where the time step Δt^k is specified later on with a CFL condition. The conservative part of the equations is computed with the finite volume numerical scheme used for transcritical flows. This conservative part also includes the source terms (bottom gradient and friction).

For clarity, the general structure of an explicit finite volume scheme is reminded below :

$$\tilde{X}_i^{n+1} + D_i^{n+1} - (X_i^n + D_i^n) + \frac{\Delta t^n}{\Delta x_i} (F_{i+1/2}^n - F_{i-1/2}^n) = \Delta t^n S_b(X_i^n) \quad (3.123)$$

where $\tilde{X} = (S, Q)$, the numerical flux $F_{i+1/2}^n$ represents the flux as computed by the Roe scheme for the Saint-Venant equations and S_b represents the source terms including friction and bottom gradients. The term $\left(\alpha(x, t) \frac{\partial^3 \bar{u}}{\partial x^2 \partial t}\right)$ is taken into account at the next step.

The spatial derivatives of the velocity are expressed using traditional finite differences at time t^{n+1} (respectively t^n).

$$\left(\alpha(x, t) \frac{\partial^2 \bar{u}}{\partial x^2}\right)_i^{n+1} = \alpha(x_i, t^{n+1}) \left(\frac{\left(\frac{\partial u}{\partial x}\right)_{i+\frac{1}{2}} - \left(\frac{\partial u}{\partial x}\right)_{i-\frac{1}{2}}}{\Delta x_i} \right)^{n+1} \quad (3.124)$$

where $u_i = \frac{Q_i}{S_i} \cdot \left(\frac{\partial u}{\partial x}\right)_{i \mp \frac{1}{2}}$ is still to be evaluated. With a finite difference discretisation this gives in $i + \frac{1}{2}$:

$$\frac{\frac{Q_{i+1}}{S_{i+1}} - \frac{Q_i}{S_i}}{\frac{\Delta x_i + \Delta x_{i+1}}{2}} \quad (3.125)$$

All the terms in $n + 1$ and those in n are grouped, this gives a tridiagonal matrix to be solved. The unknown of the system is the vector $(Q, S)^T$, given by the prediction step.

Boundary conditions

In a non-hydrostatic situation, boundary conditions are required for terms such as :

$$\frac{\partial^2 \bar{u}}{\partial x^2} \quad \text{and} \quad \frac{\partial \bar{u}}{\partial x}$$

Since there are no natural conditions, it is assumed that :

$$\frac{\partial \bar{u}}{\partial \underline{n}} = 0$$

\underline{n} is the outgoing normal vector at the upstream and downstream boundaries of the domain.

4. The CASIER solver

4.1 Introduction

This section details the methodology used to represent the role of the floodplain with storage cells in the MASCARET system.

MASCARET allows to model the floodplain along a river by the mean of *storage cells* or *basins*, with the assumptions of uniform elevation water and zero speed. These storage cells are connected together and to the river by various exchange laws, function of the connection type. This allows to model a floodplain that does not actively contribute to the flow in the main channel. It is therefore an enhancement over channel-only 1D modelling, although it does not replace the accuracy of 2D modelling.

The following sections present the methodology used in MASCARET, the equations governing the various exchange flows through the links storage cell – storage cell and river – storage cell, the numerical coupling between the CASIER solver and the unsteady subcritical engine, the solution to the storage cell system and finally the vertical discretisation of the elevation/volume curve of the storage cell.

4.2 General principle

The system to be solved can be schematised as follows :

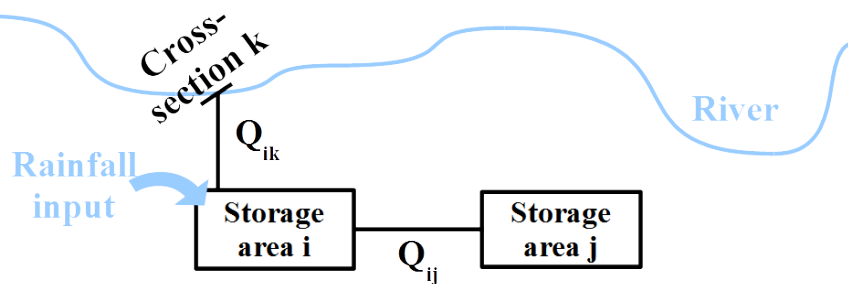


Figure 4.1: Principle of Storage Cell – River interaction

The flow in the river is determined by solving the one dimensional Saint-Venant equations. In the MASCARET system, three computational engines can be used to solve the equations: a

steady transcritical engine, an unsteady subcritical engine and an unsteady transcritical engine. This chapter only covers the coupling between the storage cell system and the unsteady subcritical engine. The algorithm used in the unsteady subcritical engine is a finite difference method with an implicit scheme, the details of which are given in section 4.4.

Regarding the storage cell system, the following assumptions are made :

- the free surface elevation in each storage cell is horizontal;
- the flow speed in the storage cells is null;
- the discharge in the exchange links only depends on the free surface elevation in the storage cells and/or that in the river.

Each storage cell is therefore modelled by a continuity equation, which can be written at each time step t :

$$\frac{dV_i(Z_i)}{dt} = \sum_j Q_{ij}(Z_i, Z_j) + \sum_k Q_{ik}(Z_i, Z_k) + Q_{input,i} \quad (4.1)$$

where :

- i and j are indices for the storage cells;
- k is the index for a computation section in a reach;
- V_i is the volume of storage cell i ;
- Z_i is the free surface elevation in storage cell i , unknown;
- Z_j is the free surface elevation in storage cell j , known at the previous time step;
- Z_k is the river level elevation in the computation section k , as computed by the subcritical engine;
- Q_{ij} is the discharge flowing from storage cell j towards storage cell i , unknown;
- Q_{ik} is the discharge flowing from the computation section k in the river towards storage cell i , unknown;
- $Q_{input,i}$ is the discharge resulting from potential contributions (rain for example) in storage cell i , given by the user.

The water elevation in storage cell i is therefore function of the exchange flows and potential input flows.

The exchange flows Q_{ij} and Q_{ik} are derived by solving the equations defining the exchange links (see section 4.3).

The elevation Z_k in the river is computed by the subcritical engine. The coupling between the storage cell system and the subcritical engine is described in section 4.4. The elevation in storage cell i is then computed by solving the storage cell continuity equation, detailed in section ??.

4.3 Derivation of the exchange flows

The exchange flow through the river - storage cell links, Q_{ik} , or storage cell - storage cell links, Q_{ij} , depends on the type of connection. The solver accepts four connection types : weir link, channel link, siphon link and orifice link.

4.3.1 Weir link

The discharge is computed from the weir formulation :

$$Q = Cml\sqrt{2g}h^{\frac{3}{2}} \quad (4.2)$$

with :

- Q discharge in the link;
- C correction coefficient (allows distinction between drowned or free flow);
- m discharge coefficient;
- l width of the weir;
- $h = Z_{u/s} - Z_{sill}$ the water depth above the weir ($Z_{u/s}$ is the elevation upstream of the weir, in the river or the storage cell, and Z_{sill} is the elevation of the weir crest). The elevation upstream of the weir, $Z_{u/s}$, is provided by the flow engine.

It is therefore necessary to know the width of the weir, its elevation (representative average height) and its discharge coefficient.

It is good practice to calibrate only the discharge coefficient, the other parameters matching their associated real physical values.

Note :

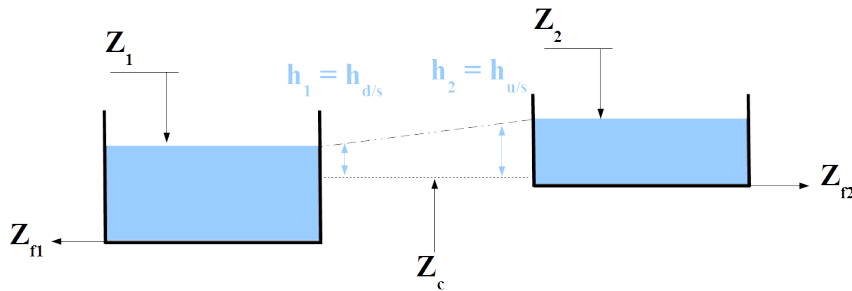
The correction coefficient, C , makes it possible to represent the physics of the flow by taking account of the drowned or free flow conditions on the weir. It also allows to stabilise the numerical computation when the difference in elevation between the storage cells tends towards 0. This is why it is called correction coefficient.

4.3.2 Channel link

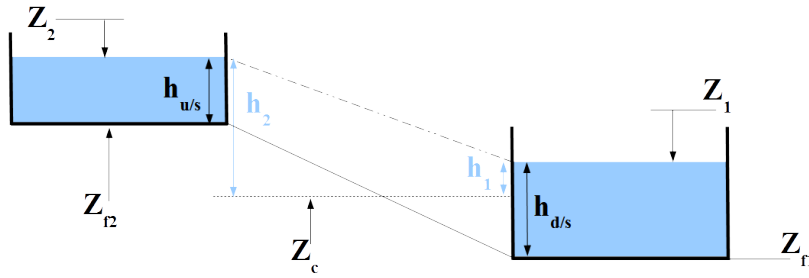
The discharge is computed with the Manning-Strickler formulation, commonly used for free surface flows, with a check of the water depth upstream of the connection.

Two distinct cases are considered :

- **Case 1** : case where the average bottom elevation of the channel is higher than the bottom elevation in the storage cells or the river (horizontal channel for example);



- **Case 2** : case where the average bottom elevation of the channel is lower than the bottom elevation in one of the storage cells (channel with uniform slope for example).



where :

- Z_1 and Z_2 are the free surface elevations in the storage cells or the river;
- Z_{b1} and Z_{b2} are the bottom elevations of the storage cells or the river;
- Z_c is the average bottom elevation of the channel;
- h_1 and h_2 correspond to the water depths reduced to the average bottom elevation of the channel ($h_1 = Z_1 - Z_c$ and $h_2 = Z_2 - Z_c$). They allow to determine the direction of the flow and to distinguish upstream from downstream (index 2 refers to upstream);
- $h_{u/s}$ and $h_{d/s}$ correspond to the real water depths upstream and downstream of the connection. In case 1, they are computed relative to the average bottom elevation of the channel and are therefore equal to h_1 and h_2 respectively. In case 2, they are computed relative to the bottom elevations of the storage cells or the river ($h_{u/s} = Z_2 - Z_{b2}$ and $h_{d/s} = Z_1 - Z_{b1}$). It is necessary to calculate these values to test the presence of water upstream of the connection. If $h_{u/s} = 0$ (whereas $h_2 > 0$), the exchange flow will be null.

The exchange flow is function of $\Delta Z = |h_1 - h_2|$ and is calculated with the Manning-Strickler formula, common for free surface flows :

$$Q = KSR_h^{\frac{2}{3}} \sqrt{\frac{\Delta Z}{L}} \quad (4.3)$$

with :

- Q discharge in the link;
- S wetted cross-section of the link;
- R_h hydraulic radius of the link;
- L length of the link;
- K Strickler coefficient for the link ($m^{1/3}.s^{-1}$);
- ΔZ difference in elevation between the two storage cells or between the storage cell and the river.

The water depth in the channel h is the average of $h_{u/s}$ and $h_{d/s}$. It is assumed that the water depth, h , is much smaller than the width, l , of the connection and that the hydraulic radius R_h can be approximated by the water depth, in other words : $h \ll l$ and $R_h \simeq h$.

With these assumptions, the previous equation becomes :

$$Q = K l h^{\frac{5}{3}} \sqrt{\frac{\Delta Z}{L}} \quad (4.4)$$

However, when the elevation difference ΔZ tends towards 0, this formulation is difficult to deal with, because the derivative of the flow relative to the difference in elevation then tends towards infinity. Numerically, this problem is addressed by introducing a correction coefficient C . This coefficient is automatically computed by the solver. The discharge then becomes :

$$Q = C K l h^{\frac{5}{3}} \sqrt{\frac{\Delta Z}{L}} \quad (4.5)$$

Therefore, to model this type of connection, it is necessary to know the average bottom elevation of the channel, its width, its length as well as its roughness coefficient. These parameters are supplied via the *.cas file or via the user interface.

It is good practice to try and calibrate only the roughness coefficient, the other parameters matching their associated real physical values.

4.3.3 Siphon link

The discharge is computed with the general equation for pressurised flows :

$$Q = \sqrt{\frac{2g}{\lambda L} \Delta Z} \times S^{\frac{5}{4}} \quad (4.6)$$

with :

- Q discharge in the link;
- S section of the siphon;
- L length of the siphon;
- ΔZ the difference in elevation between the two storage cells or between the storage cell and the river, computed by the solver;
- λ head loss coefficient with respect to the walls of the tunnel (from the Moody chart).

The siphon is considered pressurised when one of its extremities is covered with water. If the siphon is not pressurised (free surface flow), the siphon link is treated like a channel link, with the width calculated as the square root of the section S , and the Strickler coefficient calculated as :

$$K = \sqrt{\frac{2g}{\lambda} S^{\frac{1}{6}}} \quad (4.7)$$

When the elevation difference ΔZ tends towards 0, the discharge formulation above becomes unstable [9]. As for the channel link, it is necessary to introduce a correction coefficient C . The flow formulation can then be rewritten :

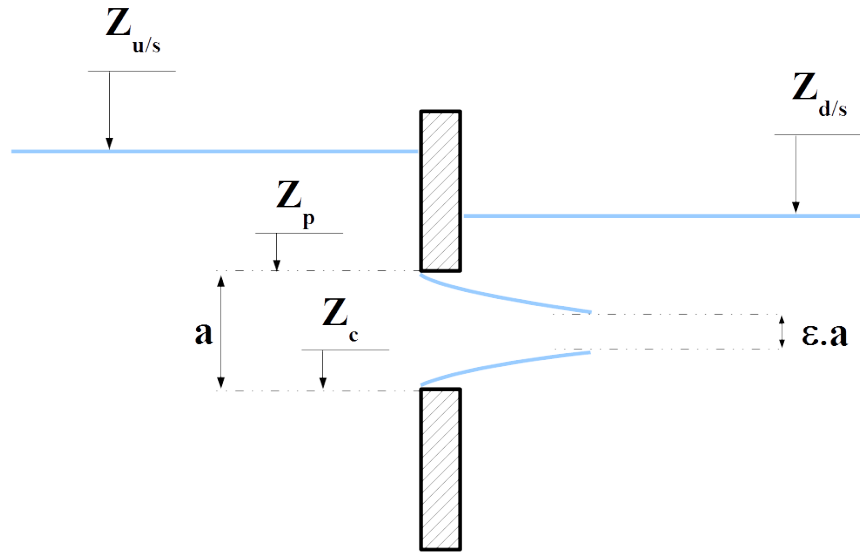
$$Q = C \sqrt{\frac{2g}{\lambda L} \Delta Z} \times S^{\frac{5}{4}} \quad (4.8)$$

Therefore, to model this type of connection, it is necessary to know the average bottom elevation of the siphon, its length and its section as well as the head loss coefficient, λ . These parameters are supplied via the *.cas file or via the user interface.

It is good practice to calibrate only the λ coefficient, the other parameters matching their associated real physical values.

4.3.4 Orifice link

The figure below presents the methodology diagram for this type of connection :



where :

- Z_c is the invert level;
- Z_p is the soffit level;
- $a = Z_p - Z_c$ is the height of the opening;
- $Z_{u/s}$ is the water elevation upstream of the connection, and $h_{u/s}$ the associated water depth : $h_{u/s} = Z_{u/s} - Z_c$;
- $Z_{d/s}$ is the water elevation downstream of the connection, and $h_{d/s}$ the associated water depth : $h_{d/s} = Z_{d/s} - Z_c$;
- ϵ is the contraction coefficient (along the vertical).

The connection is also defined by :

- B width of the opening;
- S section of the opening : $S = aB$;
- m discharge coefficient;
- The direction of flow:
 - type 1 : the flow is possible in the 2 directions;

- type 2 : the flow is only possible in the direction upstream storage cell → downstream storage cell or storage cell → river;
- type 3 : the flow is only possible in the direction downstream storage cell → upstream storage cell or river → storage cell.

The values for Z_c , B , S and m as well as the type of valve are specified by the user in the *.cas file or via the user interface. Parameters a and Z_p are calculated by the solver : $a = \frac{S}{B}$ and $Z_p = Z_c + a$.

The coefficient of vertical contraction, ε , is calculated by the solver according to the height of the opening and the load ($h_{u/s}$) :

- * if $0 \leq \frac{a}{h_{u/s}} < 0.55$: $\varepsilon = 0.65$;
- * if $0.55 \leq \frac{a}{h_{u/s}} < 0.9$: $\varepsilon = 0.5 + 0.268 \frac{a}{h_{u/s}}$;
- * if $0.9 \leq \frac{a}{h_{u/s}} \leq 1$: $\varepsilon = 0.745 + 0.255 \left(\frac{a}{h_{u/s}} - 0.9 \right)$.

The discharge calculation then depends on the type of flow :

- **free surface flow** : $Z_{u/s} < Z_p$

The orifice behaves like a weir. The discharge is calculated as for the weir link, the modular limit α is set to 0.2 (see section 4.3.5). The elevation of the weir crest is taken equal to Z_c , its width to B . The discharge coefficient is taken from the *.cas file. The user must therefore specify two values for the discharge coefficient: one for a weir formulation and one for an orifice formulation;

- **drowned surcharged flow** : $Z_{u/s} > Z_p$ et $h_{d/s} > \frac{a}{2}$

The discharge is calculated using the following formulation :

$$Q = m\varepsilon S \sqrt{2g(h_{u/s} - h_{d/s})} \quad (4.9)$$

- **free surcharged flow** : $Z_{u/s} > Z_p$ et $h_{d/s} < \frac{a}{2}$

The discharge is computed using the following formulation :

$$Q = m\varepsilon S \sqrt{2g(h_{u/s} - \frac{a}{2})} \quad (4.10)$$

Finally, the calculation of the exchange flow velocity depends whether the orifice is surcharged or not :

- **if the orifice is surcharged** :

$$V_{exch} = \frac{Q_{exch}}{S} \quad (4.11)$$

- **if the orifice is not surcharged** :

$$V_{exch} = \frac{Q_{exch}}{S_m} \quad (4.12)$$

where $S_m = B \cdot (Z_{avg} - Z_c)$ is the average wet surface above the connection. The average water level, Z_{avg} , is computed by the following formulations:

- if $Z_{d/s} < Z_c$ then $Z_{avg} = \frac{Z_{u/s} + Z_c}{2}$;
- if $Z_{d/s} > Z_c$ then $Z_{avg} = \frac{Z_{u/s} + Z_{d/s}}{2}$.

4.3.5 Drowning coefficient or correction coefficient

Need for a correction coefficient

In the case of a weir link, when the flow becomes drowned, it is necessary to use the downstream elevation in the calculation of the discharge. The downstream elevation is introduced through the drowning coefficient, C .

Validation of the solver [9] has shown that numerical instabilities occur, irrespective of the type of connection, when the difference in elevation between the storage cells or between a storage cell and the river tends towards 0. The drowning coefficient then also serves to stabilise the computation. This is why it is also called correction coefficient.

In the case of a weir link, the drowning coefficient is also used as correction coefficient. In the case of channel and siphon links, no drowning coefficient is required and a correction coefficient alone is applied. The definition of the correction coefficient matches that used for the weir link.

Definition of the C function

In the unsteady subcritical engine, drowned weirs are treated using a drowning coefficient, C , which is function of the variable, R :

$$R = \frac{Z_{d/s} - Z_{crest}}{Z_{u/s} - Z_{crest}} \quad (4.13)$$

where Z_{crest} is the elevation of the weir crest. Function C has a parabolic shape; its derivative is not continuous around 1. However, when solving the storage cells system, the derivative of the discharge appears explicitly in the computations [8]. It is therefore necessary that the C function and its derivative are continuous.

On the other hand, the C function depends on a parameter α , which defines the activation period of the C function. C is thus defined only for : $R > \alpha$. In the case of a drowned weir link, the α coefficient corresponds to the drowned flow criteria.

The polynomial function $C(R, \alpha)$ has to satisfy the following conditions :

$$\begin{aligned} C(R = \alpha) &= 1 \quad \text{and} \quad C(R = 1) = 0 \\ \frac{dC}{dR}(R = \alpha) &= 0 \quad \text{and} \quad \frac{dC}{dR}(R = 1) = 0 \\ \frac{d^2C}{dR^2}(R = \frac{1+\alpha}{2}) &= 0 \end{aligned} \quad (4.14)$$

This results in a third order function instead of the parabolic function used in the subcritical engine for weirs in rivers. In addition to the continuity of its derivatives, this formulation yields a faster reduction of the discharge when the levels of the two connected storage cells become closer.

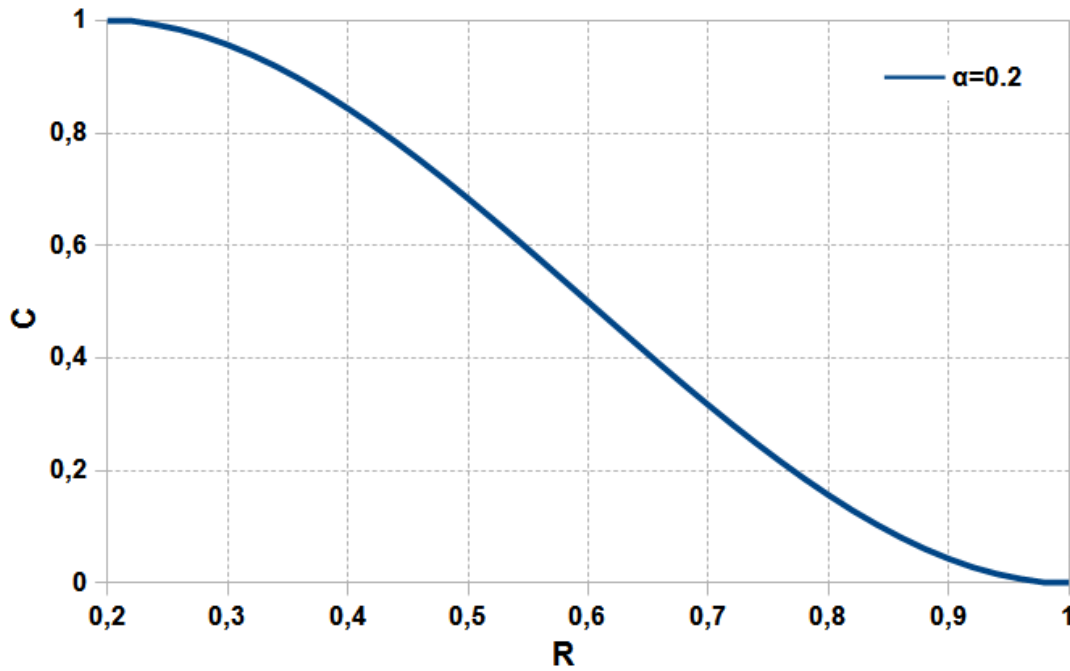
The selected function is defined as follows :

- in free flow mode and without correction : $R < \alpha$ and $C = 1$;
- in drowned mode and/or with correction : $R \geq \alpha$ ($R < 1$ in all the cases) and :

$$C = -2 \left(\frac{1-R}{1-\alpha} \right)^3 + 3 \left(\frac{1-R}{1-\alpha} \right)^2 \quad (4.15)$$

The α parameter is set to 0.95 for the channel and siphon links and varies between 0.2 and 0.8 for the weir links. It is set by the user according to the type of weir.

The variable R is calculated relative to the Z_{crest} elevation, which corresponds to the elevation of the weir crest, or to the average bottom level of the channel or that of the siphon.

Figure 4.2: Function $C = f(R)$

4.3.6 Contribution of rain

The user specifies the flows due to rainfall through the definition of a hydrograph $Q(t)$. The discharge at time t is taken as an average of the provided discharges at times t and $t - dt$. The interpolation of the flows from the hydrograph is linear.

4.4 Coupling with the subcritical engine

The storage cells and the river are implicitly coupled with the help of a state-transition matrix 2.120. Two new equations are introduced to reflect :

- the exchange flow in the links;

$$A_{link}\Delta Q_{link} + B_{link}\Delta Z_{u/s} + C_{link}\Delta Z_{d/s} = D_{link} \quad (4.16)$$

- and the change of the water level in the storage cells;

$$S_{cell} \frac{\Delta Z_{cell}}{\Delta t} = \Sigma Q_{link} \quad (4.17)$$

where S_{cell} is the surface of a storage cell and :

$$A_{link} = -1 ; B_{link} = \frac{\partial Q}{\partial Z_{u/s}} ; C_{link} = \frac{\partial Q}{\partial Z_{d/s}} ; D_{link} = 0$$

For each storage cell and each connection, the size of system 2.120 is increased with one variable and one equation. In the case of a river - storage cell link, the shallow water equations on the corresponding river cross-sections are replaced. The continuity equation becomes an equality between the flow discharges and the momentum equation is an equality between the water levels.

For the simple case of a reach with only 7 cross-sections, the initial shallow water system of equations is :

$$\begin{pmatrix} -J & G & H & & & & \\ -O & L & M & & & & \\ & -I & -J & G & H & & \\ & -N & -O & L & M & & \\ & & -I & -J & G & H & \\ & & -N & -O & L & M & \\ & & & -I & -J & G & H \\ & & & -N & -O & L & M \\ & & & & -I & -J & G & H \\ & & & & -N & -O & L & M \\ & & & & & -I & -J & G \\ & & & & & -N & -O & L \end{pmatrix} \begin{pmatrix} \Delta Z_1 \\ \Delta Q_2 \\ \Delta Z_2 \\ \Delta Q_3 \\ \Delta Z_3 \\ \Delta Q_4 \\ \Delta Z_4 \\ \Delta Q_5 \\ \Delta Z_5 \\ \Delta Q_6 \\ \Delta Z_6 \\ \Delta Q_7 \end{pmatrix} = \begin{pmatrix} K \\ P \\ K \\ P \\ K \\ P \\ K \\ P \\ K \\ P \\ K \\ P \end{pmatrix} \quad (4.18)$$

A storage cell with a link in between the river cross-sections 4 and 5 gives the following system :

$$\begin{pmatrix} -J & G & H & & & & & & & & & \\ -O & L & M & & & & & & & & & \\ & -I & -J & G & H & & & & & & & \\ & -N & -O & L & M & & & & & & & \\ & & -I & -J & G & H & & & & & & \\ & & -N & -O & L & M & & & & & & \\ & & & 1 & & -1 & & & & & & \\ & & & & 1 & & -1 & & & & & \\ & & & & & -I & -J & G & H & & & \\ & & & & & -N & -O & L & M & & & \\ & & & & & & -I & -J & G & & & \\ & & & & & & -N & -O & L & & & \\ & & & & & & & & & B & C & A \\ & & & & & & & & & & & -1 & \frac{S}{\Delta t} \end{pmatrix} \begin{pmatrix} \Delta Z_1 \\ \Delta Q_2 \\ \Delta Z_2 \\ \Delta Q_3 \\ \Delta Z_3 \\ \Delta Q_4 \\ \Delta Z_4 \\ \Delta Q_5 \\ \Delta Z_5 \\ \Delta Q_6 \\ \Delta Z_6 \\ \Delta Q_7 \\ \Delta Q_l \\ \Delta Z_c \end{pmatrix} = \begin{pmatrix} K \\ P \\ K \\ P \\ K \\ P \\ \Sigma Q \\ \Sigma Z \\ K \\ P \\ K \\ P \\ D \\ Q_l \end{pmatrix} \quad (4.19)$$

4.5 Data pre-processing

The engine for the storage cell system requires that the area at the free surface $S(Z)$ and the associated volume $V(Z)$ are known for each free surface elevation Z in a storage cell. It is therefore necessary to define the laws $S(Z)$ and $V(Z)$, which link an area and a volume to a given elevation. These laws need to be discretised, they are not continuous functions. They have a well-defined number of pairs (S, Z) and (V, Z) and the increment between elevations is constant (vertical discretisation).

The user can enter these discretised laws $S(Z)$ and $V(Z)$ *manually*, i.e. by editing directly the file holding the storage cell geometry. However, these laws are often difficult to determine. Here the user can ask that they are computed automatically by the solver, after having defined the border points and the interior points characteristic of the storage cell.

The principle of automatic vertical discretisation is as follows.

⇒ Computation of the elevations

The elevations of the interior points are sorted in ascending order in an array called Z_{INT} .

By definition Z_{INT} is such that :

$Z_{INT}(i-1) \leq Z_{INT}(i)$ for $i = 2..$ number of interior points.

Z is the array containing the discretised elevations.

The first discretised elevation is the lowest elevation from the interior points :

$Z(1) = \text{Min}(Z(i), i = 1.. \text{number of interior points}) = Z_{INT}(1)$.

The elevations are then sorted in ascending order with :

$Z(i+1) - Z(i) =$ vertical discretisation step.

The number of elevations is equal to the chosen number of vertical discretisation steps N_{planim} .

The topmost elevation is defined as follows :

$$\begin{aligned} Z(N_{planim}) &= Z(1) \\ &+ (N_{planim} - 1) \\ &\times \text{vertical discretisation step} \end{aligned}$$

⇒ Computation of the areas

The border points define the maximum area of the storage cell (S_{max}) : this area is computed from the definition of the area of a polygon. If the user has chosen the computation option whereby the area can be defined independently of the elevation, the area of the storage cell is equal to S_{max} irrespective of the elevation in the storage cell.

Otherwise, the first area is computed as follows :

$S(1) = S_{max} \times (\text{number of interior points with reference elevation } Z(1)) / (\text{number of interior points})$

If Z_{max} is the highest elevation from the interior points :

$$\begin{aligned} Z_{max} &= \text{Max}(Z(i), i = 1.. \text{number of interior points}) \\ &= Z_{INT}(\text{number of interior points}) \end{aligned}$$

As long as this elevation is not reached, $S(i)$ is computed as follows :

given k such that : $Z_{int}(k-1) \leq Z(i) < Z_{INT}(k)$

$$S_1 = \frac{k-1}{\text{size}(Z_{INT})} S_{max} \quad (4.20)$$

where $\text{size}(Z_{INT})$ is the size of array Z_{INT} .

$$S_2 = \frac{k}{\text{size}(Z_{INT})} S_{max} \quad (4.21)$$

$$\alpha = \frac{Z(i) - Z_{INT}(k-1)}{Z_{INT}(k) - Z_{INT}(k-1)} \quad (4.22)$$

$$S(i) = (1 - \alpha) \times S_1 + \alpha \times S_2 \quad (4.23)$$

When Z_{max} is reached or exceeded, the area is equal to the maximum area of the storage cell :

$S(i) = S_{max}$.

⇒ Computation of the volumes

With regard to volumes, the first volume is null : $V(1) = 0$ since it is associated to the bottom elevation of the storage cell, i.e. the storage cell is empty. The other volumes are defined by the relationship :

$V(i) = V(i-1) + [(S(i-1) + S(i))/2] \times \text{vertical discretisation step}$

EXAMPLE :

Table 4.1: Summary table of the laws $S(Z)$ and $V(Z)$

Elevation, Z	Area, S	Volume, V
$Z(1) = Z_{INT}(1)$	$S(1) = S_{max} \times$ (number of interior points with reference elevation $Z(1)) /$ (number of interior points)	$V(1) = 0$
$Z(i) = Z(i-1) +$ vertical discretisation step	$S(i) = S(i-1) + \Delta S$ $\Delta S = (1 - \alpha)S_1 + \alpha S_2$	$V(i) = V(i-1) +$ $(S(i-1) + S(i))/2 \times$ vertical discretisation step
$Z(i) \geq Z_{INT}$ (number of interior points)	$S(i) = S_{max}$	$V(i) = V(i-1) +$ $(S(i-1) + S(i))/2 \times$ vertical discretisation step
Z (number of steps of vertical discretisation) = $Z(1)$ + (number of steps of vertical discretisation - 1) \times vertical discretisation step	S (number of steps of vertical discretisation) = S_{max}	$V(i) = V(i-1) +$ $(S(i-1) + S(i))/2 \times$ vertical discretisation steps

In this example, figure 4.3) shows an plan view of a storage cell and identifies the location of the interior points with associated elevations. The points are located by their coordinates (X, Y) .

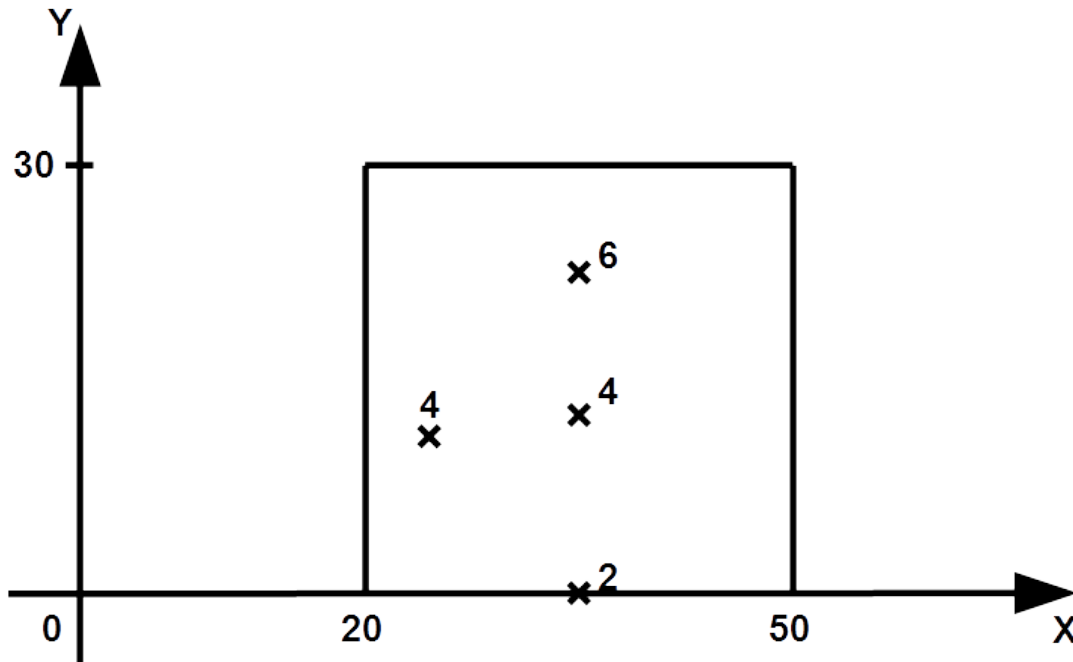


Figure 4.3: Example of vertical discretisation of a storage cell

The vertical discretisation step is 1 m, the number of vertical discretisation steps is 10. In this example, the solver determines the laws $S(Z)$ and $V(Z)$ indicated in table 4.2.

Table 4.2: Example of vertical discretisation result

Elevation, Z (<i>m</i>)	Area, S (<i>m</i>²)	Volume, V (<i>m</i>³)
2	22.5	0
3	33.8	28.1
4	67.5	78.8
5	78.8	151.9
6	90	236.3
7	90	326.3
8	90	416.3
9	90	506.3
10	90	596.3
11	90	686.3

5. Automatic calibration of the Strickler coefficient

Warning:

The automatic calibration is only possible in steady state (engine SARAP) and with only one reach with MASCARET.

5.1 Introduction

For 1D and 2D hydraulic modelling based on the Saint-Venant equations, it is necessary to define a number of data including the geometry, the initial conditions in non-stationary mode, the boundary conditions and the Strickler coefficient. All these data but the Strickler coefficient are well defined by the parameters of the study (the geometry data are provided by topographical surveys; the boundary conditions and the initial conditions are defined by the study case).

The Strickler coefficient models the linear head losses attributed to friction on the bottom and the river banks. With one-dimensional modelling, however, it also accounts for dissipation phenomena which are not directly modelled. The determination of this coefficient therefore requires some calibration. This is carried out from water levels measured for flows of the same order of magnitude as those of the study.

This calibration step, which is critical for the quality of the study, is one of the most time consuming tasks in a study.

This section presents the method integrated in the MASCARET system and based on the solution of an adjoint problem, by which the Strickler coefficients are identified. This method shortens one of the longest tasks in applied studies and allows better control of the uncertainties related to the Strickler coefficients.

The first part presents in general terms the problem of parameters identification. The complete algorithm is then presented, applied to the case of the Strickler coefficient.

The validation of the method on several simplified cases and a real case is detailed in [18].

5.2 Definition of the problem

The problem of the estimation of the Strickler coefficient is briefly presented in this section. Parameter estimation is equivalent to finding the Strickler coefficients that minimise a cost function J equal to the standard deviation between the measured values and those predicted by a numerical simulation. The general principle of the existing methods to minimise a problem with several variables is reminded below.

Let us consider a steady flow in a stretch of river. A number of measurement are available that link flows to associated elevations. The purpose is to estimate, from these measurements, the Strickler coefficient (or Strickler coefficients in different areas of the model domain) which will minimise the following function J :

$$J(K) = \sum_{i=1..m} \alpha_i (Z_i^m - Z_i^c(K))^2 \quad (5.1)$$

where :

- m is the number of measurement points;
- α_i is a weighting factor for each measurement (this coefficient can be equal to zero if no measurement is available);
- Z_i^m is the elevation measured at point i for the flow considered;
- Z_i^c is the elevation computed at point i when solving the Saint-Venant equations with a set of Strickler coefficients K_i .

Note :

- J is a cost function quantifying the variance between the elevation measured and that computed for a given set of Strickler coefficients;
- J is not known analytically. To determine J , it is necessary to evaluate the elevations computed for a given set of Strickler coefficients using the Saint-Venant equations.

Many methods of optimisation are presented in the literature to minimise the J function. Currently, the method implemented in MASCARET is that used in the old version of the CASTOR solver [23][24][25]. It is a method of gradient descent.

The algorithm of gradient descent has the following general form :

- 1) estimation of the starting point;
- 2) for : $k = 0, 1, 2, \dots$ until convergence is reached :
 - test of convergence of x_k ;
 - computation of the new direction of descent d_k ;
 - computation of the descent step λ_k : problem of non-linear minimisation with only one variable;
 - update in the direction of descent : $x_{k+1} = x_k + \lambda_k d_k$.

The various algorithms differ by the choice of the direction of descent and by the method used to compute the optimum step.

In the case of the method of gradient descent, the direction of descent is opposite to the gradient of the function. The complexity is in the estimation of the gradient of this function when the function itself is not known analytically. Minimisation is carried out by dichotomy and parabolic approximation when only one variable is used to find the parameter. This method of minimisation does not ensure that the minimum found is a global minimum (as opposed to a local minimum). Besides, convergence becomes very slow as the local minimum is reached.

5.3 Algorithm of estimation of the Strickler coefficients

This section presents in detail the algorithm used to estimate the Strickler coefficients in the case of a one-dimensional steady flow in a single reach. The gradient method used to minimise the cost function will also be presented.

5.3.1 Reminder on the Saint-Venant equations

The one-dimensional flow in a river is represented by a system of partial derivative equations where friction is modelled by the Strickler relation. These equations are obtained from the Navier-Stokes equations after several simplifying assumptions including :

- a uniform velocity direction;
- a constant velocity profile in a section of the river.

In steady state, the boundary conditions are a discharge at the upstream end of the model domain and an elevation or a stage-discharge relationship at the downstream end of the domain.

The unknown factors are the elevations at every node of the computational domain : $[x_0, x_1]$. The flow is determined by the upstream boundary condition and the flow contributions along the computational domain $[x_0, x_1]$.

$$\frac{\partial Q}{\partial x} = q_a \quad (5.2)$$

$$\frac{\partial}{\partial x} \left(\frac{Q^2}{S(Z)} \right) + gS(Z) \frac{\partial Z}{\partial x} + \frac{gQ^2}{K^2 S(Z) R^{\frac{4}{3}}} = 0 \quad (5.3)$$

$$\begin{cases} Q(x_0) = Q_{u/s} \\ Z(x_1) = Z_{d/s} \end{cases} \quad (5.4)$$

with:

- x the curvilinear x-coordinate along the river;
- Q the discharge;
- $S(Z)$ the wetted cross-section of the river, dependent upon the elevation;
- R the hydraulic radius, g the gravity and K the Strickler coefficient, as a function of x . In fact, this coefficient will be considered constant by zone.

The flow can be computed at all points of the computational domain from the continuity equation (5.2) given the flow imposed at the upstream boundary condition.

The momentum equation (5.3) is a differential equation which is discretised using a finite difference scheme. After discretisation, the equation to solve is of the form : $Z = F(Z)$.

This is equivalent to solving for the roots of function : $Z - F(Z)$. Each time the Saint-Venant equations are to be solved, the steady state subcritical is called. For more details, the reader is referred to the section introducing the principles of the steady state subcritical engine.

The solution to this differential equation, i.e. the elevations computed in each discrete point of the domain, depends on the Strickler coefficient. This coefficient is not known a priori but results from the calibration of the model.

5.3.2 Detailed presentation of the minimisation algorithm

In the presentation of the algorithm, the step where the discrete Saint-Venant equations are solved in steady state is called *solution of the Saint-Venant equations in steady state for a discrete set of Strickler coefficients defined by zone*.

The minimisation problem presented in section 5.2 is considered. The algorithm for parameter estimation can be explained as follows :

- **Step 1** : Initialisation

A spatially constant Strickler coefficient is selected over the whole domain, i.e. a vector : $K_i \quad i = 1..m$, m being the number of zones for the Strickler coefficients;

- **Step 2** : Computation of the direction of descent, equal to the gradient of function J with respect to vector K .

The gradient of J is not immediately known because J is not known analytically. J is a function of the various values taken by the Strickler coefficient in each zone. Each partial derivative with respect to a component will be approximated by a Taylor expansion.

$$\frac{\partial J(K(K_i))}{\partial K_i} \simeq \frac{J(K(K_i + dK_i)) - J(K)}{dK_i} \quad (5.5)$$

The Saint-Venant equations are solved for each component K_i of the vector representing *the constant Strickler coefficient by zones*, incremented by the discretisation step dK_i . There are as many Saint-Venant problems to solve as there are different Strickler zones. This gradient corresponds to the gradient of the discrete problem and not that of the continuous problem. Notes : In transient state, the computation of the gradient $\frac{\partial J}{\partial S_k}$ could not be carried out by this method : For n time steps varying between 1 and M , and m

measurement points, the gradient can be expressed as follows :

$$\frac{\partial J}{\partial S_k} = 2 \sum_{i,n} U_i^n (Z_i^n - m_i^n) \quad (5.6)$$

with : $U_i^n = \frac{\partial Z_i^n}{\partial S_k}$ For each simulation time step and for each measurement point, the

Saint-Venant equations must be solved to evaluate the gradient.

- **Step 3** : Minimisation in the direction of descent

The problem is as follows: to minimise the function J in the direction opposite that of the gradient, i.e. to find ρ minimising :

$J\left(K_k - \rho \frac{\partial J(K_k)}{\partial K}\right)$ where J is a non-linear function. The idea behind the estimation of ρ is to find by dichotomy a configuration such that for 3 points ρ_1, ρ_2 and ρ_3 : $\rho_1 < \rho_2 < \rho_3$, $J(\rho_2) < J(\rho_1)$ and $J(\rho_2) < J(\rho_3)$. The function J can then be locally approximated by a parabola. The optimal ρ parameter is the minimum of this parabola, which can be computed analytically. The determination of the 3 points (ρ_1, ρ_2, ρ_3) is done by dichotomy. For each estimate of function J , a Saint-Venant problem is solved.

- **Step 4** : Increment in the direction of descent

$$K_{k+1} = K_k - \rho \nabla J_k$$

- **Step 5** : Testing convergence

$$\|J(K_{k+1})\| < \varepsilon$$

Note :

This method only finds local minima. Although this minimisation technique is relatively old, it has the advantage of being very simple to implement.

5.4 Method of calibration for compound channels

The Strickler coefficients for the river bed (low flow channel) and floodplains (high flow channel) are defined by zones to reproduce measured water levels corresponding to bankfull and overtopping flows. As a first step, the Strickler coefficients for the river bed are estimated from the water levels measured for the bankfull flow using the automatic calibration algorithm. In a second step, the Strickler coefficients for the floodplain are then estimated using the calibrated Strickler coefficients for the river bed and the water levels measured for the overtopping flow, using again the automatic calibration algorithm.

5.5 Validation and conclusions

The algorithm for parameter estimation was initially validated on an idealised test case, such that the results obtained could be checked. The algorithm was then applied to a real study.

In steady state, automatic calibration of the Strickler coefficient is carried out using a method of parameter estimation and the computation of the cost function gradient. This approach gives satisfactory results for both idealised and real test cases. It is, however, noted that convergence is relatively slow, especially near a local minimum. In practice, the accuracy obtained with this approach is sufficient: it is indeed not realistic to expect a better accuracy than 1 cm away from the measured levels.

Future developments will focus on the calibration for the unsteady flow state (introducing the adjoint problem to evaluate the gradient of the cost function) and on the improvement of the minimisation method (introducing more robust and more recent methods such as the conjugate gradient method or Quasi-Newton methods).

- [1] M. AFIF. *Analyse de quelques problèmes hyperboliques issus de la modélisation des crues de rivières*. PhD thesis, Université de Saint-Etienne, Juin 1986.
- [2] D. AMBROSI. Approximate of shallow water equations by roe's riemann solver. *International Journal for Numerical Methods in Fluids*, 20:157–168, 1995.
- [3] C. BON. *Modélisation et simulation numérique d'écoulements hydrauliques et de ruissellement en topographie quelconque*. PhD thesis, Université Bordeaux I, Talence, 1997.
- [4] T. BUFFARD. *Analyse de quelques méthodes de volumes finis non structurés pour la résolution des équations d'Euler*. PhD thesis, Université Paris VI, France, décembre 1993.
- [5] M. CARLIER. *Hydraulique générale et appliquée*. Collection de la Direction des Etudes et Recherches d'Electricité de France, Eyrolles, Paris, 1987.
- [6] M.E. VAZQUEZ CENDON. *Estudio de esquemas descentrados para su aplicación a las leyes de conservación hiperbólicas con términos fuente*. PhD thesis, Universidad Santiago de Compostella, Spain, 1994.
- [7] O. LE COZ. *Un schéma équilibre pour les écoulements à surface libre instationnaires avec bathymétrie*. PhD thesis, Université Bordeaux I, Talence, 1996.
- [8] N. GOUTAL et C. RISSOAN. MASCARET 5.1 note d'utilisation. Technical Report HP-76/02/013, EDF, 2002.
- [9] C. RISSOAN et E. MONIER. Système MASCARET: validation du couplage d'un système de casiers avec le noyau fluvial non permanent. Technical Report HP-76/02/043, EDF, 2002.
- [10] N. GOUTAL et F. MAUREL. Note de principe de la version 4.1 du code MASCARET. Technical Report HE-43/96/075/B, EDF, 1996.
- [11] N. GOUTAL et F. MAUREL. Proceedings of the 2nd workshop on the dam-break wave simulation. Technical Report HE-43/97/016/A, EDF, 1997.
- [12] G. NICOLLET et M. UAN. Ecoulements permanents à surface libre en lits composés. *La Houille Blanche*, 1:21–30, janvier 1979.
- [13] J. A. CUNGE et M. WEGNER. Intégration numérique des équations d'écoulement de barré de Saint-Venant par un schéma implicite de différences finies. *La Houille Blanche*, 1:33–39, janvier 1964.

- [14] E. GODLEWSKI and P.A. RAVIART. *Hyperbolic systems of conservation laws*. Mathématiques et applications, Ellipses, 1991.
- [15] N. GOUTAL. Note de principe du code RUPTUR. Technical Report HE-43/91/038, EDF, 1991.
- [16] N. GOUTAL. Implication du noyau transcritique de MASCARET. Technical Report H-P76/2002/008/B, EDF, 2002.
- [17] N. GOUTAL. Système MASCARET v5p2 : ajout de nouvelles singularités. Technical Report HP-76/03/020/A, EDF, 2003.
- [18] N. GOUTAL. Calage automatique du coefficient de strickler en régime permanent sur un bief unique. Technical Report HP-75/05/021/A, EDF, 2005.
- [19] V. GUINOT. Uncertainty in modelling from an engineering perspective. In *SimHydro 2010 : Hydraulic modeling and uncertainty*, Sophia Antipolis, 2-4 June 2010.
- [20] J.-M. Hervouet. *Hydrodynamics of free surface flows, modelling with the finite element method*. John Wiley & Sons Ltd., 2007.
- [21] V. KUTIJA and C. J. M. HEWETT. Modelling of supercritical flow conditions revisited; NewC Scheme. *Journal of Hydraulic Research*, 40, 2002.
- [22] P.D. LAX. Hyperbolic systems of conservation laws and the mathematical theory of stock wales. In Regional Conference Series SIAM, editor, *Lectures in Applied Math.*, Philadelphia, 1972.
- [23] A. LEBOSSE. Code CASTOR – Calage du coefficient de Strickler Optimisé en Rivière – note de principe. Technical Report HE-43/89/40, EDF, 1989.
- [24] A. LEBOSSE. Code CASTOR – Calage du coefficient de Strickler Optimisé en Rivière – note de validation. Technical Report HE-43/89/57, EDF, 1989.
- [25] A. LEBOSSE. Estimation of the manning-strickler-roughness coefficient in st-venant equations. Technical Report HE-43/89/58, EDF, 1989.
- [26] R. J. LEVEQUE. Numerical methods for conservation laws, 1990.
- [27] N. GOUTAL M.-O. BRISTEAU and J. SAINTE-MARIE. Numerical simulations of a non-hydrostatic shallow water model. *Computers and Fluids*, 47:51–64, 2011.
- [28] F. MAUREL. Traitement des confluent dans le logiciel MASCARET 4.0 - principes de la méthode et éléments de validation. Technical Report HE-43/96/067/A, EDF, 1996.
- [29] E. A. MESELHE and F. M. HOLLY Jr. Invalidity of the Preissmann scheme for transcritical flow. *Journal of Hydraulic Engineering*, pages 652–655, 1997.
- [30] L.A. MONTHE. *Etude des Equations aux Dérivées Partielles Hyperboliques Raides. Application aux équations de Saint-Venant*. PhD thesis, Université de Rouen, France, Juillet 1997.
- [31] A. PAQUIER. *Modélisation et simulation de la progression de l'onde de rupture de barrage*. PhD thesis, Université Jean Monnet, Octobre 1995.

- [32] P.L. ROE. Approximate riemann solvers, parameters vectors, and difference schemes. *Journal of Computational Physics*, 43, Issue 2:357–372, October 1981.
- [33] E. BEN SLAMA. Code EROSIF - note de principe, note d'utilisation et éléments de validation. Technical Report HE-43/95/073/A, EDF, 1995.
- [34] J. SMOLLER. *Shock waves and reaction-diffusion equations*. Springer-Verlag, Berlin, 1983.
- [35] M. J. BAINES T. C. JOHNSON and P. K. SWEBY. A box scheme for transcritical flow. *International Journal for Numerical Methods in Engineering*, 55:895–912, 2002.
- [36] E. F. TORO. *Shock-capturing methods for free-surface shallow flows*. Ed. Wiley and Sons, 2001.
- [37] M. UAN. Etablissement des équations régissant les écoulements non permanents à surface libre avec champ d'inondation. Technical Report C43-75-66, EDF/LNH, 1975.
- [38] J. WASNIEWSKI Z. ZLATEV and K. SCHAUMBURG. *Y12M Solution of large and sparse systems of linear algebraic equations: documentation of subroutines*. Springer-Verlag Gmbh & Co. Kg, Berlin and Heidelberg, october 1981.