

O. Kolditz

Computational
Methods
in Environmental
Fluid Mechanics

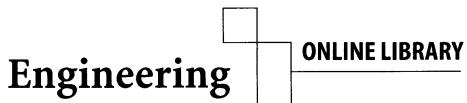


Springer

Olaf Kolditz

Computational Methods in Environmental Fluid Mechanics

Springer-Verlag Berlin Heidelberg GmbH



<http://www.springer.de/engine/>

Olaf Kolditz

Computational Methods in Environmental Fluid Mechanics



Springer

Professor Dr.-Ing. Olaf Kolditz
University of Tübingen
Center for Applied Geosciences
Sigwartstraße 10
D-72076 Tübingen
Germany
e-mail: kolditz@uni-tuebingen.de

Library of Congress Cataloging-in-Publication-Data applied for

Die Deutsche Bibliothek – CIP-Einheitsaufnahme
Kolditz, Olaf:
Computational methods in environmental fluid mechanics / Olaf Kolditz. -
Berlin ; Heidelberg ; New York ; Barcelona ; Hong Kong ; London ; Milan ;
Paris ; Tokyo : Springer 2002

ISBN 978-3-642-07683-1
DOI 10.1007/978-3-662-04761-3

ISBN 978-3-662-04761-3 (eBook)

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitations, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

<http://www.springer.de>

© Springer-Verlag Berlin Heidelberg 2002
Originally published by Springer-Verlag Berlin Heidelberg New York in 2002
Softcover reprint of the hardcover 1st edition 2002

The use of general descriptive names, registered names trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Typesetting: Data delivered by author
Cover Design: de'blick Konzept & Gestaltung, Berlin
Printed on acid free paper SPIN: 10856102 62/3020/M – 5 4 3 2 1 0

Preface

Fluids play an important role in environmental systems appearing as surface water in rivers, lakes, and coastal regions or in the subsurface as well as in the atmosphere. Mechanics of environmental fluids is concerned with fluid motion as well as associated mass and heat transport. In subsurface systems we have additionally to deal with deformation processes in soil and rock systems.

This textbook is organized in four sections. The first part gives the background of continuum mechanics. We consider the general balance equations of mass, momentum and energy, averaging concepts for turbulence and the porous medium approach. The second part is dealing with numerical methods for solving partial differential equations. The basic concepts of approximation theory and afterwards finite differences, finite elements as well as finite volumes are explained and illustrated with basic equations for diffusion, advection and transport processes. In the third part of this book aspects of implementation numerical methods in an object-oriented framework are discussed. Object-oriented programming (OOP) has become exceedingly popular in the past few years. OOP is more than rewriting programs in modern languages, OOP is a new way of thinking about designing and realizing software projects.

The first three parts of this book were used for a master course in computational fluid mechanics. In the fourth section several topics about recent research in fractured-porous media modeling are presented (e.g. non-linear flow, heat transport, density-dependent flow, multiphase flow and deformation processes). These topics, in particular on non-linear problems, might be useful for PhD courses in computational mechanics.

The material of this textbook I prepared during my work at the Institute of Fluid Mechanics at the University of Hannover from 1994-2001. My lectures on computational fluid mechanics (Strömungsmechanik V/VI) were part of a more general lecture circle on fluid mechanics by Prof. W. Zielke, J. Strybny, R. Ratke (Strömungsmechanik I/II), Prof. M. Markofsky (Strömungsmechanik III/IV) for civil engineering students (Roman numbers indicate semesters). The idea to prepare a textbook in English was to offer this material also to Master and PhD courses in civil and environmental engineering. The lectures were part of the teaching program in civil engineering at the International Centre of Computational Engineering Science (ICCES) in Hannover (www.icces.de)

and now in environmental engineering at the Centre of Applied Geosciences in Tübingen (ZAG) (www.uni-tuebingen.de/geo/zag/).

I am indebted to many of my colleagues and students at the Institute of Fluid Mechanics in Hannover. In particular, I thank Prof. Werner Zielke for continuously supporting my work and my PhD students Carsten Thorenz, René Kaiser, Abderrahmane Habbar, Thomas Rother, Martin Kohlmeier and Sylvia Moenckes (in the order of appearance) for their enthusiastic work concerning the RockFlow project (www.rockflow.de). The assistance by Martin Beinhorn and Jöelle de Jonge to improve the English is also appreciated. A special thank is extended to Prof. Hans-Jörg Diersch (Berlin, www.wasy.de) for his useful comments on this book. I appreciate the valuable discussions with Profs. T. Taniguchi (Okayama), B. Berkowitz (Rehovot), G. Starke (Hannover), T. Schanz (Weimar), M. Sauter (Jena) and G. Teutsch (Tübingen) on theoretical and practical aspects of fluids in the environment. I would also like to thank the Springer-Verlag for realizing this publishing project and, especially, Dr. Ditzinger for encouraging me to spend the time this summer to transform the lecture script into a textbook.

The heaviest burden involved in writing this book was borne by my wife, Barbara, who had to put up with the many inconveniences that are unavoidable when one is engaged in writing a book. I am very grateful to her for the continuous encouragement. Finally, I would like to dedicate this book to my family without whom it might be never have been completed.

Hannover / Tübingen, August 2001

Olaf Kolditz

Contents

Part I - Continuum Mechanics	1
1 Balance Equations of Fluid Mechanics	3
1.1 General Conservation Law	3
1.1.1 Basic Equations of Fluid Dynamics	3
1.1.2 Conservation Quantities	5
1.1.3 Lagrangian Description of Motion	5
1.1.4 Eulerian Description of Motion	7
1.1.5 Reynolds Transport Theorem	9
1.1.6 Fluxes	9
1.1.7 General Balance Equation	11
1.2 Mass Conservation	12
1.3 Momentum Conservation	13
1.3.1 General Momentum Equation	13
1.3.2 Stress Tensor - σ	14
1.3.3 Kinematics of a Fluid Element	17
1.3.4 Viscous Stress Tensor - τ	19
1.3.5 Euler Equations	21
1.3.6 Navier-Stokes Equations	21
1.3.7 Stokes Equations	21
1.3.8 Darcy Equations	22
1.4 Energy Conservation	22
1.4.1 Energy Balance	22
1.4.2 Energy Fluxes	23

1.4.3	Energy Sources	23
1.4.4	Integral Energy Balance Equation	23
1.4.5	Differential Energy Balance Equation	23
1.4.6	Balance Equation of Kinetic Energy	25
1.4.7	Balance Equation of Internal Energy	25
1.5	Problems	27
2	Turbulence	29
2.1	Physics of Turbulence	29
2.2	Reynolds Equations	34
2.3	Turbulence Models	36
2.3.1	Mixing Length Model	39
2.3.2	$k - \varepsilon$ Model	40
3	Porous Media	45
3.1	Multiphase Media	45
3.2	Macroscopic Equations	47
3.3	Isothermal Consolidation of Porous Media	50
3.3.1	Mass Conservation	50
3.3.2	Momentum conservation	53
3.4	Mass Transport in Porous Media	58
3.5	Heat Transport in Porous Media	60
4	Problem Classification	63
4.1	Mathematical Classification	63
4.2	Physical Classification	64
4.3	Elliptic Equations	66
4.4	Parabolic Equations	67
4.5	Hyperbolic Equations	68
4.6	Equation Types	70
4.7	Boundary Conditions	70
4.7.1	General Remarks	70
4.7.2	Mass Fluxes	71
4.7.3	Momentum Fluxes	72

4.7.4	Energy Transfer	75
4.8	Problems	76
Bibliography		77
Part II - Numerical Methods		79
5	Numerical Methods	81
5.1	Solution Procedure	81
5.2	Theory of Discrete Approximation	83
5.2.1	Terminology	83
5.2.2	Errors and Accuracy	84
5.2.3	Convergence	85
5.2.4	Consistency	86
5.2.5	Stability	86
5.3	Solution Process	86
5.3.1	Linear Solver	87
5.3.2	Non-Linear Solver	91
5.4	Problems	96
6	Finite Difference Method	97
6.1	Approximation of Derivatives	97
6.1.1	Taylor Series Expansion (TSE)	97
6.1.2	First-Order Derivatives	98
6.1.3	Second-Order Derivatives	99
6.2	Diffusion Equation	100
6.2.1	Explicit and Implicit Schemes	100
6.2.2	Explicit FTCS Scheme	101
6.2.3	Fully Implicit Scheme	105
6.2.4	Crank-Nicolson Scheme (CNS)	107
6.2.5	Generalized Scheme	108
6.2.6	Initial and Boundary Conditions	108
6.3	Advection Equation	110
6.3.1	Explicit FTCS Scheme	110

6.3.2	Upwind Difference Representation	111
6.3.3	Leapfrog and Lax-Wendroff Schemes	113
6.3.4	Crank-Nicolson Scheme (CNS)	114
6.3.5	Euler-Taylor Scheme	115
6.3.6	Numerical Dispersion and Dissipation	115
6.4	Transport Equation	116
6.4.1	Steady Transport Equation	117
6.4.2	Explicit Schemes for the Linear Transport Equation . . .	121
6.4.3	Implicit Schemes for the Linear Transport Equation . .	121
6.5	Burgers Equation	122
6.5.1	Explicit Schemes	124
6.5.2	Implicit Schemes	125
6.6	Problems	126
7	Finite Element Method	129
7.1	Domain Discretization	130
7.2	Equation Discretization	132
7.2.1	Variational Principles	133
7.2.2	Method of Weighted Residuals	134
7.2.3	Galerkin Method	135
7.3	Interpolation and Shape Functions	137
7.3.1	General Method for Shape Function Derivation	137
7.3.2	Shape Function Conditions	138
7.3.3	Isoparametric Mapping	139
7.4	Element Shape Functions	140
7.4.1	1-D Linear Bar Elements	140
7.4.2	1-D Quadratic Elements	144
7.4.3	2-D Triangular Elements	145
7.4.4	2-D Bilinear Quadrilateral Elements	148
7.4.5	3-D Tetrahedral Elements	150
7.4.6	3-D Triangular Prismatic Elements	152
7.4.7	3-D Hexahedral Element	154
7.5	Network Simulation	157

7.5.1	Pipe Networks - 1-D Elements in \mathcal{R}^3	157
7.5.2	Fracture Networks - 2-D Triangular Elements in \mathcal{R}^3	159
7.6	Diffusion Equation	161
7.6.1	Steady Diffusion Equation	161
7.6.2	Unsteady Diffusion Equation	163
7.7	Transport Equation	166
7.7.1	Unsteady Transport Equation	166
7.8	Problems	170
8	Finite Volume Method	173
8.1	Discretization Process	174
8.2	Evaluation of Time Derivatives	176
8.3	Evaluation of Cell Area/Volume	176
8.4	Evaluation of Fluxes	178
8.4.1	Two-Dimensional Schemes	178
8.4.2	Central Scheme on a Cartesian Mesh	179
8.4.3	Upwind Scheme on a Cartesian Mesh	181
8.4.4	Integration Formulas	182
8.5	Diffusion Equation	183
8.6	Advection Equation	185
8.7	Problems	187
Bibliography		189
Part III - Software-Engineering		191
9	Object-Oriented Methods for Hydrosystem Modeling	193
9.1	Introduction	193
9.2	Investigation Objects	194
9.2.1	Geometric modeling	194
9.2.2	Processes	196
9.3	Mathematical Objects	196
9.4	Numeric Objects	199
9.4.1	Discretization Process	199

9.4.2	Solution Process	200
9.4.3	Common Methods	200
9.5	Informatic Objects	201
9.5.1	Data Structures	202
9.6	Software Objects	203
10	Object-Oriented Programming Techniques	207
10.1	Object Orientation	207
10.2	Object Design	208
10.2.1	Mathematical Objects	208
10.2.2	Numerical Objects	209
10.2.3	Algebraic Equations - Solver	211
10.2.4	Model-Kernel Concept	213
10.3	Object Implementation	214
10.3.1	Object	214
10.3.2	Nodes and Elements	215
10.3.3	Node Related Objects	217
10.3.4	Element Related Objects	219
10.3.5	Equation Systems and Solver	222
10.3.6	Models	224
10.3.7	FEM – Application	227
10.4	Graphical User Interface	228
10.4.1	Shell – Objects	228
10.4.2	Application Embedding	229
10.4.3	Object - GUI - Interface	230
11	Element Implementation	233
11.1	2-D Linear Triangular Elements	233
11.1.1	Data Input	233
11.1.2	Element Geometry	234
11.1.3	Element Matrices	235
11.1.4	Equation System	235
11.1.5	Element Resultants	235
11.2	Problems	236

Bibliography	237
Part IV - Selected Topics	239
12 Non-Linear Flow in Fractured Media	241
12.1 Introduction and Background	241
12.1.1 Theoretical Issues	241
12.1.2 Experimental Issues	244
12.2 Governing Equations	249
12.3 Numerical Scheme	251
12.3.1 Galerkin Method	251
12.3.2 Finite Element Approach	253
12.3.3 Evaluation of Element Matrices in Local Coordinates	253
12.3.4 Resolution of the Non-Linearities	254
12.4 Non-Linear Flow in Single Fractures	255
12.4.1 Numerical Properties	256
12.4.2 Effects of Non-Linearity	259
12.4.3 Fracture Roughness	260
12.5 Non-linear Flow in Fracture Systems	261
12.6 Concluding Remarks	266
12.7 Appendix	266
Bibliography	269
13 Heat Transport in Fractured-Porous Media	271
13.1 Introduction	271
13.2 Governing Equations	275
13.3 Numerical Procedure	277
13.4 Example - Rosemanowes Hot Dry Rock Site	277
13.4.1 Data	277
13.4.2 Simulation Results	280
13.4.3 Discussion	289
13.5 Case Study: Soultz-sous-Forêts	293
Bibliography	296

14 Density Dependent Flow in Porous Media	301
14.1 Introduction	302
14.1.1 Background	302
14.1.2 Prior work concerning variable-density flow and transport	303
14.2 Governing Equations	305
14.2.1 Macroscopic balance and constitutive equations	306
14.2.2 Boussinesq approximation of the mass balance of the fluid	309
14.2.3 Different formulations of the balance equation of solute mass	309
14.3 Finite Element Formulations	311
14.4 Examples	314
14.4.1 The Henry problem	315
14.4.2 The Elder problem	318
14.4.3 The salt dome problem	323
Bibliography	329
15 Multiphase Flow in Deformable Porous Media	333
15.1 Macroscopic Balance Equations	333
15.1.1 Mass Conservation in Static Porous Media	333
15.1.2 Mass Conservation in Deformable Porous Media	335
15.1.3 Momentum Conservation	336
15.2 Constitutive Relationships	337
15.2.1 Saturation	337
15.2.2 Density	338
15.2.3 Capillary Pressure	339
15.2.4 Relative Permeability	343
15.2.5 Total and Effective Stress	345
15.3 Governing Equations	347
15.3.1 Static Porous Medium - Two Phase Flow	348
15.3.2 Deformable Porous Medium - Richards Approximation . .	349
15.3.3 Deformable Porous Medium - Two Phase Flow	350
15.4 Finite Element Formulations	352
15.4.1 Mass Balance of Porous Medium	352

15.4.2 Equilibrium Problem	354
15.5 Implementation	356
15.5.1 Model	358
15.5.2 Loop	360
15.5.3 Element Matrices and Resultants	361
15.5.4 Equation System	361
15.5.5 Solver	361
15.5.6 Materials	363
15.5.7 Data Input	364
15.6 Examples	364
15.6.1 Benchmark Problem - Liakopoulos Experiment	364
15.6.2 Moisture Swelling of Bentonites	367
Bibliography	371

Index	373
--------------	------------

Part I

Continuum Mechanics

Chapter 1

Balance Equations of Fluid Mechanics

1.1 General Conservation Law

1.1.1 Basic Equations of Fluid Dynamics

The basic idea of continuum mechanics is that the evolution of a physical system is completely determined by conservation laws, i.e. basic properties such as mass, momentum, and energy are conserved during the considered process at all times. Any physical system can be completely determined by this conservation properties. In contrast, other quantities such as pressure or entropy do not obey conservation laws. The only additional information concerns the consistence of the material (e.g. fluids, solids, porous medium) in form of constitutive laws.

The concept of conservation means that the variation of a conservation quantity within a given control volume is due to the net effect of internal *sources* and of the amount of the quantity which is crossing the boundary surface of the considered volume - *fluxes*. Sources and fluxes are, in general, dependent of space-time coordinates as well as on mechanical and thermodynamic factors. Fluxes result from two contributions: first due to advective transport by fluid motion and second due to diffusion/dispersion processes. Diffusion is always present even when the fluid is at rest. Diffusion is the tendency towards equilibrium or homogeneity of a physical system.

Kinematics of Continua

In continuum mechanics we distinguish between two methods concerning the derivation of balance equations. In the **Lagrangian formulation** we follow the quantity along a pathline, i.e. following particles (Fig. 1.1). In the **Eulerian**

formulation of motion we consider variations of the quantity with respect to a fixed control volume at fixed places (Fig. 1.2).

A **pathline** is a curve along which a fixed particle of a continuum moves during a sequence of time. Pathline is Lagrangian concept of motion. A **streamline** is a curve along which a sequence of particles moves at a given time. By definition, the tangent to a streamline coincides with the velocity vector at that point. Streamline is Eulerian concept of motion. Note, for unsteady flow the streamline may vary from one instant to the next, whereas for steady flow streamlines remain unchanged with time. For steady motion both pathlines and streamlines coincide. Any particle will remain on a given streamline as time proceeds. Additional terms associated with kinematics of continua are the following (see also section 1.3.3).

- particle
- material coordinates (Lagrangian coordinates) (Fig. 1.1)

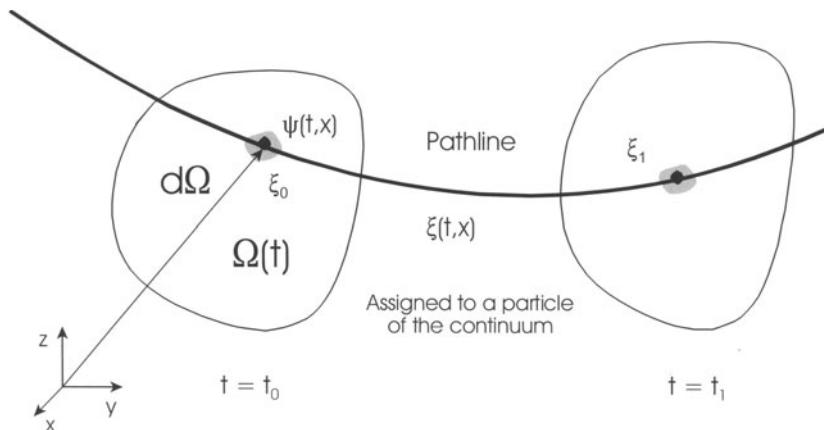


Figure 1.1: Lagrangian description of motion

- spatial coordinates (Eulerian coordinates) (Fig. 1.2)
- displacement \mathbf{w} and volumetric dilatation (relative growth of volume with respect to the original one)

$$\varepsilon = \nabla \cdot \mathbf{w} = \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}$$

- material derivative (see section 1.1.5)

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla \quad (1.1)$$

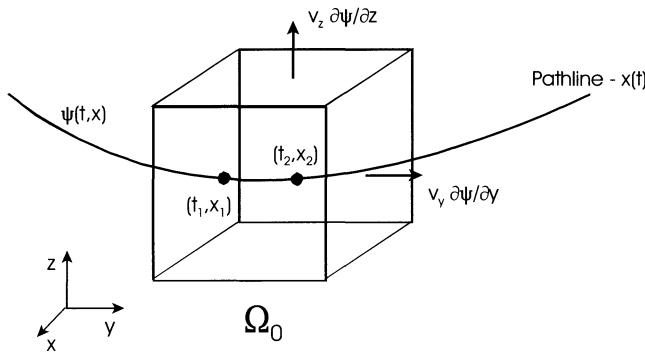


Figure 1.2: Eulerian description of motion

1.1.2 Conservation Quantities

The amount of a quantity in a defined volume Ω is given by

$$\Psi = \int_{\Omega} \psi d\Omega(t) \quad (1.2)$$

where Ψ is an extensive conservation quantity (i.e. mass, momentum, energy) and ψ is the corresponding intensive conservation quantity such as mass density ρ , momentum density $\rho\mathbf{v}$ or energy density e (see Table 1.1).

Table 1.1: Conservation quantities

Extensive quantity	Symbol	Intensive quantity	Symbol
Mass	M	Mass density	ρ
Linear momentum	\mathbf{m}	Linear momentum density	$\rho\mathbf{v}$
Energy	E	Energy density	$e = \rho i + \frac{1}{2}\rho v^2$

1.1.3 Lagrangian Description of Motion

From the Lagrangian view point, the general conservation law postulates that the quantity remains conserved during the physical process within Ω .

$$\frac{d}{dt} \int_{\Omega} \psi d\Omega = \int_{\Omega} q^\psi d\Omega \quad (1.3)$$

where d/dt is the material derivative (eqn 1.1). From the definition of the derivative we have (Fig. 1.3)

$$\begin{aligned}
 & \frac{d}{dt} \int_{\Omega} \psi d\Omega = \\
 &= \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \left\{ \int_{\Omega(t+\Delta t)} \psi(t + \Delta t) d\Omega - \int_{\Omega(t)} \psi(t) d\Omega \right\} \\
 &= \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \left\{ \int_{\Omega_2} [\psi(t + \Delta t) - \psi(t)] d\Omega + \int_{\Omega_3} \psi(t + \Delta t) d\Omega - \int_{\Omega_1} \psi(t) d\Omega \right\} \\
 &= \int_{\Omega_2} \frac{\partial \psi}{\partial t} d\Omega + \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \left\{ \int_{\Omega_3} \psi(t + \Delta t) d\Omega - \int_{\Omega_1} \psi(t) d\Omega \right\} \tag{1.4}
 \end{aligned}$$

According to Fig. 1.3, the area elements can be defined as (note the time direction)

$$\begin{aligned}
 d\Omega_1 &= (\mathbf{v} \cdot d\mathbf{S})(-\Delta t) & , & \quad d\Omega_3 = (\mathbf{v} \cdot d\mathbf{S})\Delta t \\
 t + \Delta t \rightarrow t & & , & \quad t \rightarrow t + \Delta t \tag{1.5}
 \end{aligned}$$

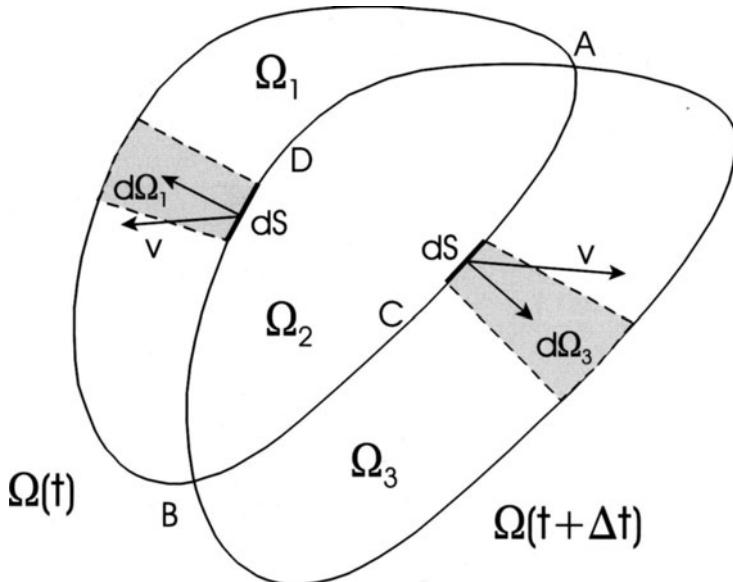


Figure 1.3: Lagrangian description of motion - proof

Therefore, we have

$$\begin{aligned}
 & \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \left\{ \int_{\Omega_3} \psi(t + \Delta t) d\Omega - \int_{\Omega_1} \psi(t) d\Omega \right\} = \\
 &= \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \left\{ \oint_{BCA} \psi(t + \Delta t) (\mathbf{v} \cdot d\mathbf{S}) \Delta t + \oint_{ADB} \psi(t) (\mathbf{v} \cdot d\mathbf{S}) \Delta t \right\} \\
 &= \oint_{\partial\Omega} \psi(t) \mathbf{v} \cdot d\mathbf{S}
 \end{aligned} \tag{1.6}$$

where $\partial\Omega$ is the domain boundary. Finally, we have following expression for the general conservation law

$$\frac{d}{dt} \int_{\Omega} \psi d\Omega = \int_{\Omega} \frac{\partial \psi}{\partial t} d\Omega + \oint_{\partial\Omega} \psi(t) \mathbf{v} \cdot d\mathbf{S} = \int_{\Omega} q^{\psi} d\Omega \tag{1.7}$$

1.1.4 Eulerian Description of Motion

On the other hand, in the Eulerian sense, the variation of the quantity results only from internal sources and from fluxes through the surface of a fixed control volume $\partial\Omega$. Variation of the considered quantity per unit time is

$$\frac{\partial}{\partial t} \int_{\Omega} \psi d\Omega \tag{1.8}$$

Incoming flux through the fixed control volume surface is given by

$$- \oint_{\partial\Omega} \Phi^{\psi} \cdot d\mathbf{S} \tag{1.9}$$

where Φ^{ψ} is the total flux vector of quantity ψ . Internal sources of the considered quantity can be written as

$$\int_{\Omega} q^{\psi} d\Omega \tag{1.10}$$

where q^{ψ} is the internal sources density of ψ . Therefore, conservation of quantity ψ in the Eulerian description can be written in following form

$$\underbrace{\frac{\partial}{\partial t} \int_{\Omega} \psi d\Omega}_1 = \underbrace{- \oint_{\partial\Omega} \Phi^{\psi} \cdot d\mathbf{S}}_2 + \underbrace{\int_{\Omega} q^{\psi} d\Omega}_3 \tag{1.11}$$

with:

1. Rate of change of total amount of quantity ψ in the control volume,
2. Net rate of increase / decrease of ψ due to fluxes,
3. Rate of increase / decrease of ψ due to sources.

Note equations (1.7) and (1.11) are equivalent for $\Phi = \mathbf{v}\psi$

The **Gauss-Ostrogradskian integral theorem** (Gaussian divergence theorem) can be used to transform surface into volume integrals for the flux term.

$$\oint_{\partial\Omega} \Phi^\psi \cdot d\mathbf{S} = \int_{\Omega} \nabla \cdot \Phi^\psi d\Omega \quad (1.12)$$

We can rewrite the general conservation law now in following form

$$\int_{\Omega} \frac{\partial \psi}{\partial t} d\Omega = - \int_{\Omega} \nabla \cdot \Phi^\psi d\Omega + \int_{\Omega} q^\psi d\Omega \quad (1.13)$$

where additionally we used the fact that the control volume is fixed for the moment $\partial\Omega/\partial t = 0$

We consider a parallelepiped control volume in a Cartesian coordinate system (Fig. 1.4). The limes of net efflux through the control volume surface $\partial\Omega$ as $\Omega \rightarrow 0$ will be

$$\begin{aligned} & \lim_{\Omega \rightarrow 0} \frac{1}{\Omega} \oint_{\partial\Omega} \Phi \cdot d\mathbf{S} = \\ &= \lim_{\Delta x \Delta y \Delta z \rightarrow 0} \frac{1}{\Delta x \Delta y \Delta z} \left\{ (\Phi_x |_{x+\frac{\Delta x}{2}, y, z} - \Phi_x |_{x-\frac{\Delta x}{2}, y, z}) \Delta y \Delta z + \right. \\ & \quad (\Phi_y |_{x, y+\frac{\Delta y}{2}, z} - \Phi_y |_{x, y-\frac{\Delta y}{2}, z}) \Delta x \Delta z + \\ & \quad \left. (\Phi_z |_{x, y, z+\frac{\Delta z}{2}} - \Phi_z |_{x, y, z-\frac{\Delta z}{2}}) \Delta x \Delta y \right\} \\ &= \frac{\partial \Phi}{\partial x} + \frac{\partial \Phi}{\partial y} + \frac{\partial \Phi}{\partial z} = \nabla \cdot \Phi \end{aligned} \quad (1.14)$$

The above expression provides a physical interpretation of the divergence at a point as the excess of efflux over influx of quantity ψ through a closed surface surrounding the point of interest, per unit volume.

$$\nabla \cdot \Phi = \lim_{\Omega \rightarrow 0} \frac{1}{\Omega} \oint_{\partial\Omega} \Phi \cdot d\mathbf{S} \quad (1.15)$$

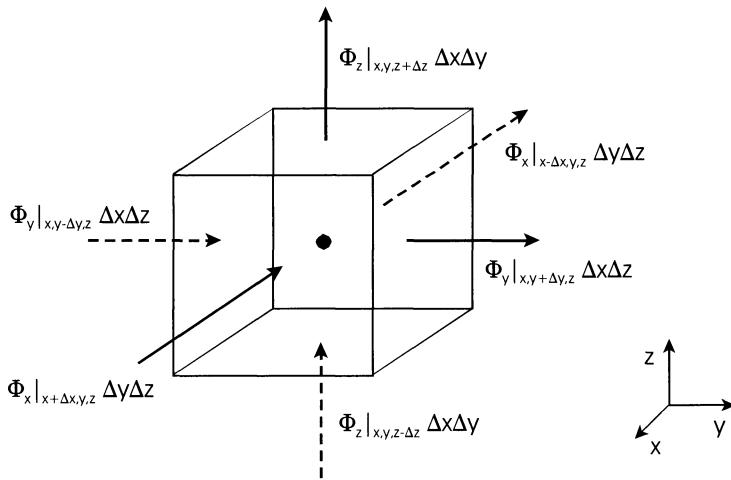


Figure 1.4: Eulerian description of motion - fluxes

1.1.5 Reynolds Transport Theorem

Of course, conservation does not depend on the specific method for derivation. Both the Lagrangian (1.3) and the Eulerian point of view (1.13) must result in identical expressions of conservation laws.

$$\frac{d}{dt} \int_{\Omega} \psi d\Omega = \int_{\Omega} \left(\frac{\partial \psi}{\partial t} + \nabla \cdot \Phi^{\psi} \right) d\Omega = \int_{\Omega} q^{\psi} d\Omega \quad (1.16)$$

The above equation is known as the **Reynolds transport theorem** which can be understood as the relationship between the Lagrangian and the Eulerian descriptions. To derive the corresponding differential equation of the above integral equation, we make use of the fact that the integral expression must be valid for all (in particular infinitesimal) volumes Ω .

1.1.6 Fluxes

The total flux Φ^{ψ} of a quantity ψ is defined as

$$\Phi^{\psi} = \mathbf{v}^E \psi \quad (1.17)$$

where \mathbf{v}^E is a mean particle velocity. Physically Φ^{ψ} represents the quantity of ψ passing through a unit area of the continuum, normal to \mathbf{v}^E , per unit time with respect to a fixed coordinate system.

For the case of a multi-component continuum let \mathbf{v} denote the mass-weighted velocity describing a more ordered motion of the particles of a fluid element. The total flux can be written as

$$\Phi^\psi = \mathbf{v}^E \psi = \underbrace{\mathbf{v} \psi}_{\Phi_A^\psi} + \underbrace{(\mathbf{v}^E - \mathbf{v}) \psi}_{\Phi_D^\psi} \quad (1.18)$$

and, therefore, decomposed into two parts: an advective flux Φ_A^ψ and a diffusive flux Φ_D^ψ relative to the mass-weighted velocity:

- advective flux of quantity ψ

$$\Phi_A^\psi = \mathbf{v} \psi \quad (1.19)$$

- diffusive flux of quantity ψ (Fick's law)

$$\Phi_D^\psi = -\alpha \nabla \psi \quad (1.20)$$

This means, diffusive flux is positive in the direction of negative gradient.

If the conservation quantity is a vector (e.g. linear momentum) then the flux becomes a tensor and the source term a vector (e.g. body forces):

- advective flux of vector quantity ψ

$$\Phi_A^\psi = \mathbf{v} : \psi = [v_x \ v_y \ v_z] \begin{bmatrix} \psi_x \\ \psi_y \\ \psi_z \end{bmatrix} = \begin{vmatrix} v_x \psi_x & v_x \psi_y & v_x \psi_z \\ v_y \psi_x & v_y \psi_y & v_y \psi_z \\ v_z \psi_x & v_z \psi_y & v_z \psi_z \end{vmatrix} \quad (1.21)$$

- diffusive flux of vector quantity ψ

$$\Phi_D^\psi = -\rho \nabla : \psi = -\alpha \begin{vmatrix} \frac{\partial \psi_x}{\partial x} & \frac{\partial \psi_y}{\partial y} & \frac{\partial \psi_z}{\partial z} \\ \frac{\partial \psi_x}{\partial x} & \frac{\partial \psi_y}{\partial y} & \frac{\partial \psi_z}{\partial z} \\ \frac{\partial \psi_x}{\partial y} & \frac{\partial \psi_y}{\partial z} & \frac{\partial \psi_z}{\partial z} \end{vmatrix} \quad (1.22)$$

Using Fick's law the total flux can be written as

$$\mathbf{v} : \psi = \mathbf{v} \psi - \alpha \nabla \psi \quad (1.23)$$

When substituting this equation into the general balance equation, we yield the so-called transport equation

$$\underbrace{\int_{\Omega} \frac{\partial \psi}{\partial t}}_{1} = - \underbrace{\int_{\Omega} \nabla \cdot (\mathbf{v} \psi)}_{2} + \underbrace{\int_{\Omega} \nabla \cdot (\alpha \nabla \psi)}_{3} + \underbrace{\int_{\Omega} q^\psi}_{4} \quad (1.24)$$

with:

1. Rate of increase of ψ within a fluid element
2. Net rate of ψ due to flux out of the fluid element
3. Rate of increase / decrease of ψ due to diffusion
4. Rate of increase / decrease of ψ due to sources

1.1.7 General Balance Equation

Now a differential form of the general conservation law can be derived, because the conservation law in integral form must be valid for an arbitrary small volume. We obtain

$$\frac{\partial \psi}{\partial t} = -\nabla \cdot \Phi^\psi + q^\psi = -\nabla \cdot (\mathbf{v}^E \psi) + q^\psi \quad (1.25)$$

which is denoted as the general balance equation for quantity ψ in differential form. Using partial differentiation we can introduce the convective form of general balance equation.

$$\begin{aligned} \frac{\partial \psi}{\partial t} &= -\mathbf{v}^E \nabla \psi - \psi \nabla \cdot \mathbf{v}^E + q^\psi \\ \frac{d\psi}{dt} &= \frac{\partial \psi}{\partial t} + \mathbf{v}^E \nabla \psi = -\psi \nabla \cdot \mathbf{v}^E + q^\psi \end{aligned} \quad (1.26)$$

While eqn (1.25) is the Eulerian form of the general balance equation the above equation is the corresponding Lagrangian form. The term $d\psi/dt$ [see eqn (1.1) for definition of material derivative] represents the rate of change of ψ as a particle moves along a pathline, while the right-hand-side represents the sources that cause this change.

Integral forms are more general formulations of conservation laws, since they remain valid also in the presence of discontinuous variation of the quantities (e.g. such as inviscid shock waves or contact discontinuities). Only if the thermodynamical properties remain continuous, integral and differential forms are fully equivalent.

1.2 Mass Conservation

The conservation law of mass states the empirical fact that in the absence of sources mass cannot disappear from a system nor be created.

$$\frac{dM}{dt} = \frac{d}{dt} \int_{\Omega} \rho d\Omega = 0 \quad (1.27)$$

where M is mass and ρ is the mass density (specific mass). This means, no diffusive fluxes exist for mass transport in a homogeneous, single phase medium. Mass can only be transported through advection. Applying Reynolds transport theorem we have for $\psi = \rho$

$$\frac{d}{dt} \int_{\Omega} \rho d\Omega = \int_{\Omega} \left[\frac{\partial \rho}{\partial t} + \nabla \cdot (\mathbf{v}\rho) \right] d\Omega = 0 \quad (1.28)$$

The differential equation of mass conservation in divergence form becomes

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\mathbf{v}\rho) = 0 \quad (1.29)$$

Partial differentiation of the above equation gives

$$\frac{\partial \rho}{\partial t} + \mathbf{v} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{v} = 0 \quad (1.30)$$

Using the material (or convective) derivative (1.1) the mass conservation equation can be rewritten as

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v} \quad (1.31)$$

Note, above convective form of mass conservation equation becomes zero only for incompressible flows, i.e.

$$\frac{d\rho}{dt} = 0 \quad (1.32)$$

requires divergence-free flow.

$$\nabla \cdot \mathbf{v} = 0 \quad (1.33)$$

From eqn. (1.30) results that the above expression is the continuity equation for a homogeneous fluid ($\rho = \text{const}$).

1.3 Momentum Conservation

1.3.1 General Momentum Equation

As momentum is a vector quantity we need the general conservation law in vector form to state the balance equation for momentum conservation. Starting from the Eulerian general balance equation (1.11) for momentum density $\psi = \rho\mathbf{v}$ we have

$$\frac{\partial}{\partial t} \int_{\Omega} \rho \mathbf{v} d\Omega + \oint_{\partial\Omega} \Phi^m \cdot d\mathbf{S} = \int_{\Omega} \rho \mathbf{f} d\Omega \quad (1.34)$$

Flux term: The advective momentum flux is defined as

$$\Phi^m = (\rho\mathbf{v}) \otimes \mathbf{v} = (\rho\mathbf{v})\mathbf{v} \quad (1.35)$$

Note that momentum flux is a tensor $\Phi_{ij}^m = \rho v_i v_j$.

Source term: From Newton's second law it is known, that variation of momentum result from forces acting on the physical system on consideration. We distinguish between external forces \mathbf{F}^e and internal forces \mathbf{F}^i .

$$\mathbf{F} = \int_{\Omega} \rho \mathbf{f} d\Omega = \int_{\Omega} \rho(\mathbf{f}^e + \mathbf{f}^i) d\Omega = \underbrace{\int_{\Omega} \rho \mathbf{f}^e d\Omega}_{\text{External forces}} + \underbrace{\oint_{\partial\Omega} \boldsymbol{\sigma} : d\mathbf{S}}_{\text{Internal forces}} = \mathbf{F}^e + \mathbf{F}^i \quad (1.36)$$

The latter are dependent on the nature of the fluid. Internal forces (stresses) express the resistance of the fluid due to internal deformation. Internal forces within the volume Ω are those acting on points of the boundary surface. Since they have no opposite counterparts within the considered volume internal forces act as surface forces.

Substituting now flux and source terms of momentum we obtain

$$\frac{\partial}{\partial t} \int_{\Omega} \rho \mathbf{v} d\Omega + \oint_{\partial\Omega} \rho \mathbf{v} (\mathbf{v} \cdot d\mathbf{S}) = \int_{\Omega} \rho \mathbf{f}^e d\Omega + \oint_{\partial\Omega} \boldsymbol{\sigma} : d\mathbf{S} \quad (1.37)$$

Applying the Gauss-Ostrogradskian theorem to the surface integrals

$$\begin{aligned} \oint_{\partial\Omega} \rho \mathbf{v} (\mathbf{v} \cdot d\mathbf{S}) &= \int_{\Omega} \nabla \cdot (\rho \mathbf{v} \mathbf{v}) d\Omega \\ \oint_{\partial\Omega} \boldsymbol{\sigma} : d\mathbf{S} &= \int_{\Omega} \nabla \cdot \boldsymbol{\sigma} d\Omega \end{aligned} \quad (1.38)$$

the momentum balance equation becomes

$$\frac{\partial}{\partial t} \int_{\Omega} \rho v d\Omega + \int_{\Omega} \nabla \cdot (\rho \mathbf{v} \mathbf{v}) d\Omega = \int_{\Omega} \rho \mathbf{f}^e d\Omega + \int_{\Omega} \nabla \cdot \boldsymbol{\sigma} d\Omega \quad (1.39)$$

The differential form of the momentum conservation law is then

$$\frac{\partial}{\partial t} (\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v} \mathbf{v}) = \rho \mathbf{f}^e + \nabla \cdot \boldsymbol{\sigma} \quad (1.40)$$

The above equation is now extended by partial integration

$$\begin{aligned} \rho \frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \frac{\partial \rho}{\partial t} + (\rho \mathbf{v}) \cdot \nabla \mathbf{v} + \mathbf{v} \nabla \cdot (\rho \mathbf{v}) &= \rho \left[\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right] + \mathbf{v} \left[\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) \right] \\ &= \rho \mathbf{f}^e + \nabla \cdot \boldsymbol{\sigma} \end{aligned} \quad (1.41)$$

Using the mass conservation equation (1.29) and dividing by ρ we obtain

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = \mathbf{f}^e + \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma} \quad (1.42)$$

In index notation the above vector equation is written as

$$\begin{aligned} \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} &= \frac{1}{\rho} \left(\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} \right) \\ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} &= \frac{1}{\rho} \left(\frac{\partial \sigma_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z} \right) \\ \frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} &= g + \frac{1}{\rho} \left(\frac{\partial \sigma_{zx}}{\partial x} + \frac{\partial \sigma_{zy}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} \right) \end{aligned} \quad (1.43)$$

with $u = v_x, v = v_y, w = v_z$ and $\mathbf{f}^e = \mathbf{g}$.

1.3.2 Stress Tensor - $\boldsymbol{\sigma}$

Surface forces on a fluid element result from pressure and viscous forces. The stress tensor can be, therefore, defined as

$$\boldsymbol{\sigma} = -p \mathbf{I} + \boldsymbol{\tau} \quad (1.44)$$

Pressure is the normal stress on the fluid element denoted by p . Fig. 1.5 shows the nine viscous stress components acting on a fluid element in motion. Suffix notation is applied to indicate the direction of the stress components. Suffices i and j in τ_{ij} mean, that stress acts in j -direction on a surface normal to the i -direction.

As an example we consider the x -components of the force due to pressure and viscous stress. The magnitude of the force resulting from acting stresses is

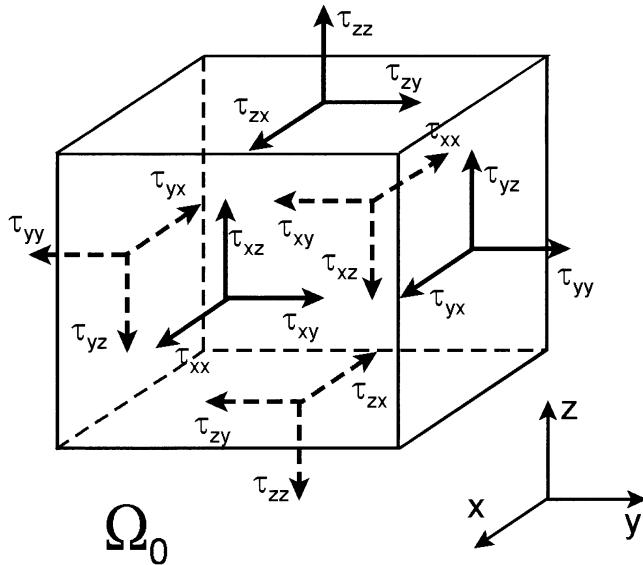


Figure 1.5: Components of the viscous stress tensor τ_{ij}

the product of these stresses and the corresponding area. On each pair faces of a fluid element with volume $dV = dx dy dz$ we have the following net force components in y -direction (Fig. 1.6).

(E,W) - pair faces

$$\begin{aligned} \rho f_y^i dV &= \left[(p - \frac{\partial p}{\partial y} \frac{1}{2} dy) - (p + \frac{\partial p}{\partial y} \frac{1}{2} dy) \right] dx dz \\ &+ \left[-\tau_{yy} + \frac{\tau_{yy}}{\partial y} \frac{1}{2} dy + \tau_{yy} + \frac{\tau_{yy}}{\partial y} \frac{1}{2} dy \right] dx dz \\ &= \left(-\frac{\partial p}{\partial y} + \frac{\tau_{yy}}{\partial y} \right) dx dy dz \end{aligned} \quad (1.45)$$

(N,S) - pair faces

$$\begin{aligned} \rho f_y^i dV &= - \left(\tau_{xy} - \frac{\tau_{xy}}{\partial x} \frac{1}{2} dx \right) dy dz + \left(\tau_{xy} + \frac{\tau_{xy}}{\partial x} \frac{1}{2} dx \right) dy dz \\ &= \frac{\partial \tau_{xy}}{\partial x} dx dy dz \end{aligned} \quad (1.46)$$

(T,B) - pair faces

$$\rho f_y^i dV = - \left(\tau_{zy} - \frac{\tau_{zy}}{\partial z} \frac{1}{2} dz \right) dx dy + \left(\tau_{zy} + \frac{\tau_{zy}}{\partial z} \frac{1}{2} dz \right) dx dy$$

$$= \frac{\partial \tau_{zy}}{\partial z} dx dy dz \quad (1.47)$$

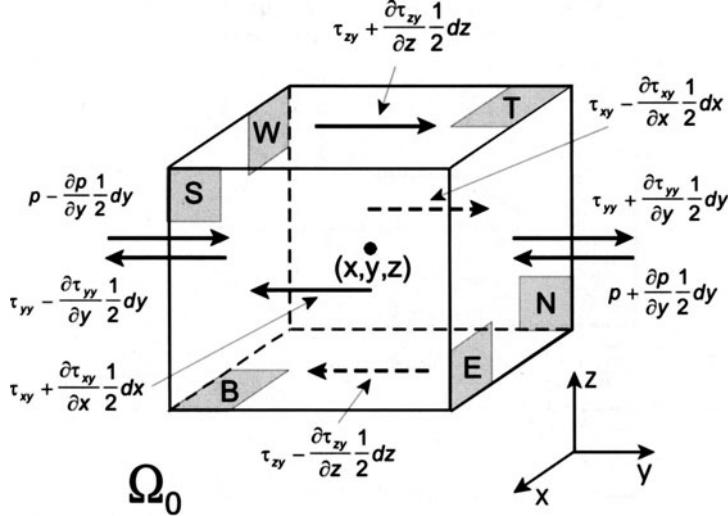


Figure 1.6: Forces due to stress

The usual sign conventions are: forces aligned with the coordinate direction get a positive sign and those in opposite direction a negative one. The total force per unit volume is then equal to the sum of all directional components divided by the volume of the fluid element. The corresponding components in x - and z -directions can be obtained in the same way.

$$\begin{aligned} \rho f_x^i &= \frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\tau_{yx}}{\partial y} + \frac{\tau_{zx}}{\partial z} \\ \rho f_y^i &= \frac{\tau_{xy}}{\partial x} + \frac{\partial(-p + \tau_{yy})}{\partial y} + \frac{\tau_{zy}}{\partial z} \\ \rho f_z^i &= \frac{\partial(-p + \tau_{xz})}{\partial x} + \frac{\tau_{yz}}{\partial y} + \frac{\partial(-p + \tau_{zz})}{\partial z} \end{aligned} \quad (1.48)$$

or in compact vector notation we have

$$\rho \mathbf{f}^i = -\nabla p + \nabla \cdot \boldsymbol{\tau} \quad (1.49)$$

The momentum balance equation (1.41) can be rewritten now as

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = \mathbf{f}_e - \frac{1}{\rho} \nabla p + \frac{1}{\rho} \nabla \cdot \boldsymbol{\tau} \quad (1.50)$$

1.3.3 Kinematics of a Fluid Element

Velocity components of a deformation process are given by (Fig. 1.7)

$$\begin{aligned} u(\mathbf{x} + d\mathbf{x}) &= u(\mathbf{x}) + \frac{\partial u}{\partial x} dx + \frac{\partial u}{\partial y} dy + \frac{\partial u}{\partial z} dz \\ v(\mathbf{x} + d\mathbf{x}) &= v(\mathbf{x}) + \frac{\partial v}{\partial x} dx + \frac{\partial v}{\partial y} dy + \frac{\partial v}{\partial z} dz \\ w(\mathbf{x} + d\mathbf{x}) &= w(\mathbf{x}) + \frac{\partial w}{\partial x} dx + \frac{\partial w}{\partial y} dy + \frac{\partial w}{\partial z} dz \end{aligned} \quad (1.51)$$

where $\mathbf{w} = d\mathbf{x}$ is an infinitesimal displacement vector.

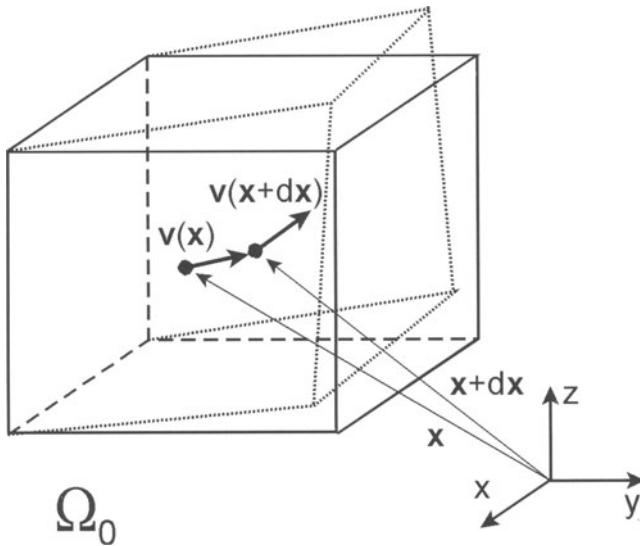


Figure 1.7: Deformation of a fluid element

The above equation can be given in a more compact form as

$$v_i(\mathbf{x} + d\mathbf{x}) = v_i(\mathbf{x}) + \frac{\partial v_i}{\partial x_j} dx_j \quad (1.52)$$

In vector notation the above relations can be expressed by

$$\mathbf{v}(\mathbf{x} + d\mathbf{x}) = \mathbf{v}(\mathbf{x}) + (\nabla \otimes \mathbf{v}) \cdot d\mathbf{x} \quad (1.53)$$

The tensor components of spatial velocity variations can be divided into following parts

$$\frac{\partial v_i}{\partial x_j} = \underbrace{\frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right)}_{\text{Symmetric}} + \underbrace{\frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i} \right)}_{\text{Skew-symmetric}} = \underbrace{D_{ij}}_{\text{Deformation}} + \underbrace{R_{ij}}_{\text{Rotation}} \quad (1.54)$$

In compact tensor notation we have

$$\nabla \otimes \mathbf{v} = \nabla \mathbf{v} = \mathbf{D} + \mathbf{R} \quad (1.55)$$

The first term denotes deformation and the second term rotation of a fluid element. Now the deformation velocity of a fluid element can be expressed by

$$\mathbf{v}(\mathbf{x} + d\mathbf{x}) = \underbrace{\mathbf{v}(\mathbf{x})}_{\text{Translation}} + \underbrace{\mathbf{D} \cdot d\mathbf{x}}_{\text{Deformation}} + \underbrace{\mathbf{R} \cdot d\mathbf{x}}_{\text{Rotation}} \quad (1.56)$$

Any movement of a fluid element consists basically of translation, deformation, and rotation (Fig. 1.8). These basic movements of fluid elements are shortly described in the following.

Translation

- rigid movement

Rotation

- rotation angle

$$d\gamma \approx \sin(d\gamma) = \frac{\partial v}{\partial x} dt = -\frac{\partial u}{\partial y} dt \quad (1.57)$$

- displacements

$$\begin{aligned} d(\Delta x) &= \Delta x \sin(d\gamma) \approx \Delta x \frac{\partial v}{\partial x} dt \\ d(\Delta y) &= \Delta y \sin(d\gamma) \approx \Delta y \frac{\partial u}{\partial y} dt \end{aligned} \quad (1.58)$$

- angular velocity

$$\omega = \frac{d\gamma}{dt} = \frac{\partial v}{\partial x} = -\frac{\partial u}{\partial y} \quad (1.59)$$

$$\omega = \frac{1}{2} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) \quad (1.60)$$

Deformation - Elongation

- displacements

$$d(\Delta x) = \frac{\partial u}{\partial x} \Delta x dt \quad , \quad d(\Delta y) = \frac{\partial v}{\partial y} \Delta y dt \quad (1.61)$$

- area change

$$\begin{aligned} d(\Delta A) &= (\Delta x + \frac{\partial u}{\partial x} \Delta x dt + \Delta y + \frac{\partial v}{\partial y} \Delta y dt) - \Delta x \Delta y \\ &= \frac{\partial u}{\partial x} \Delta x \Delta y dt + \frac{\partial v}{\partial y} \Delta x \Delta y dt + \frac{\partial u}{\partial x} \frac{\partial v}{\partial y} \Delta x \Delta y dt^2 \end{aligned} \quad (1.62)$$

- velocity of area change

$$\frac{1}{\Delta A} \frac{d(\Delta V)}{dt} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \mathcal{O}(dt) \quad (1.63)$$

- divergence of deformation velocity / dilatation velocity / 3-D

$$\lim_{\Delta x \Delta y \Delta z \rightarrow 0} \left[\frac{1}{\Delta V} \frac{d(\Delta V)}{dt} \right] = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = \nabla \cdot \mathbf{v} \quad (1.64)$$

Deformation - Shearing

- shear angles

$$\begin{aligned} d\gamma_1 &\approx \sin(d\gamma_1) = \frac{\partial v}{\partial x} dt \\ d\gamma_2 &\approx \sin(d\gamma_2) = \frac{\partial u}{\partial y} dt \end{aligned} \quad (1.65)$$

- displacements

$$\begin{aligned} d(\Delta x) &= \Delta x \sin(d\gamma_1) \approx \Delta x \frac{\partial v}{\partial x} dt \\ d(\Delta y) &= \Delta y \sin(d\gamma_2) \approx \Delta y \frac{\partial u}{\partial y} dt \end{aligned} \quad (1.66)$$

- shear velocity

$$\vartheta = \frac{1}{2} \frac{d\gamma}{dt} = \frac{d\gamma_1}{dt} + \frac{d\gamma_2}{dt} = \frac{1}{2} \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) \quad (1.67)$$

1.3.4 Viscous Stress Tensor - τ

For an non-rotational flow, i.e. $\nabla \times \mathbf{v} = 0$, the expression for deformation velocities can be further reduced to

$$\mathbf{v}(\mathbf{x} + d\mathbf{x}) = \mathbf{v}(\mathbf{x}) + \mathbf{D} \cdot d\mathbf{x} = \mathbf{v}(\mathbf{x}) + \frac{d\mathbf{v}}{d\mathbf{x}} \cdot d\mathbf{x} \quad (1.68)$$

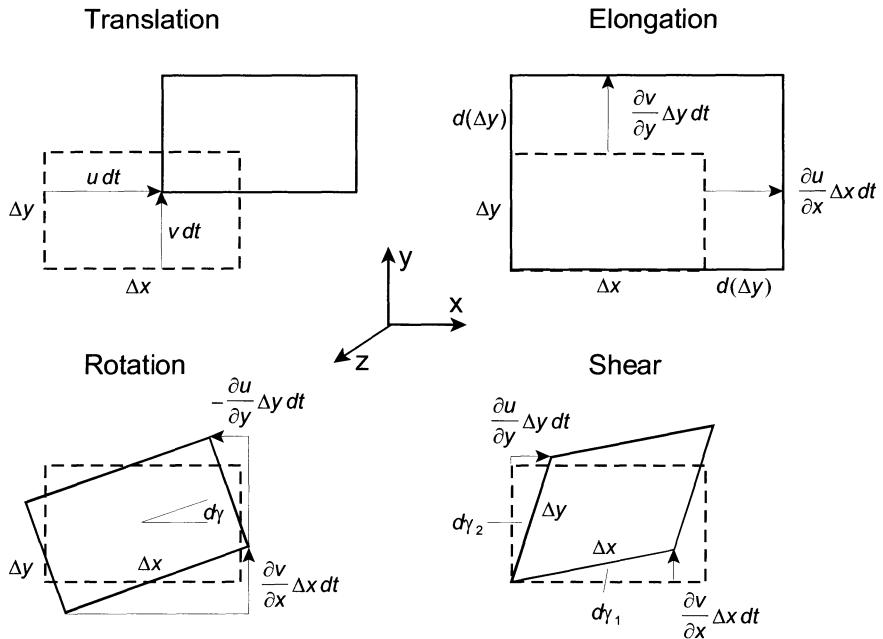


Figure 1.8: Kinematics of a fluid element

According to Newton's law for viscous stresses ($\tau = \mu dv/dn$), the viscous stress tensor is defined by

$$\tau = 2\mu \mathbf{D} \quad (1.69)$$

Therefore, internal forces (1.50) within Newtonian fluids are given by

$$\rho \mathbf{f}^i = -\nabla p + 2\mu \nabla \cdot \mathbf{D} \quad (1.70)$$

The divergence of the deformation tensor is equal to

$$\nabla \cdot \mathbf{D} = \nabla \cdot (\nabla \mathbf{v}) = \frac{1}{2} \Delta \mathbf{v} \quad (1.71)$$

Finally we obtain the following expressions for internal viscous forces

$$\rho \mathbf{f}^i = -\nabla p + \mu \Delta \mathbf{v} \quad (1.72)$$

or

$$\mathbf{f}^i = -\frac{\nabla p}{\rho} + \nu \Delta \mathbf{v} \quad (1.73)$$

where μ is the dynamic viscosity in [Pas] and ν is the kinematic viscosity [$m^2 s^{-1}$]. From the above expressions for internal forces acting on fluid elements, balance equations for different fluids can be obtained, e.g. the Euler and the Navier-Stokes equations (see next chapters).

1.3.5 Euler Equations

For an ideal fluid without internal shear stresses (inviscid fluid),

$$\sigma = -p\mathbf{I} \quad (1.74)$$

the momentum conservation law (1.50) reduces to the **Euler equations**

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = \mathbf{f}^e - \frac{1}{\rho} \nabla p \quad (1.75)$$

1.3.6 Navier-Stokes Equations

For a Newtonian viscous fluid with stress tensor

$$\sigma = -p\mathbf{I} + \tau \quad (1.76)$$

the momentum conservation law (1.50) is

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = \mathbf{f}^e - \frac{1}{\rho} \nabla p + \nu \Delta \mathbf{v}$$

(1.77)

which are called the **Navier-Stokes equations**. In general fluid mass continuity equation (1.29) has to be added to obtain a closed set of equations for determination of all unknown functions ρ, \mathbf{v} .

Dimensional analysis of the Navier-Stokes equation yields the non-dimensionalized form

$$\frac{v^* t^*}{L^*} \frac{\partial \mathbf{v}_D}{\partial t_D} + (\mathbf{v}_D \cdot \nabla) \mathbf{v}_D = -\nabla p_D + \frac{1}{Re} \Delta \mathbf{v}_D \quad (1.78)$$

with $Re = v^* L^* / \nu^*$ the Reynolds number. Asterix * indicates characteristic values of the quantity and D denotes dimensionless quantity, i.e. $\mathbf{v}_D = \mathbf{v}/v^*$.

1.3.7 Stokes Equations

For strongly viscous dominated flows (i.e. small Reynolds numbers) convective acceleration can be neglected with respect to the viscous term. As the result we obtain the Stokes equation.

$$\frac{\partial \mathbf{v}}{\partial t} = \mathbf{f}^e - \frac{1}{\rho} \nabla p + \nu \Delta \mathbf{v} \quad (1.79)$$

In non-dimensionalized form we have

$$\frac{v^*^2 t^*}{\nu^*} \frac{\partial \mathbf{v}_D}{\partial t_D} = \mathbf{f}_D^e - Re \nabla p_D + \Delta \mathbf{v}_D \quad (1.80)$$

1.3.8 Darcy Equations

For slow motion (e.g. in porous media) frequently we can neglect inertial forces and turbulence effects. Then we obtain the Darcy equations.

$$0 = \mathbf{f}^e - \frac{1}{\rho} \nabla p + \nu \Delta \mathbf{v} \quad (1.81)$$

The friction term in porous media is discussed in chapter 3.

1.4 Energy Conservation

1.4.1 Energy Balance

The equation of energy conservation is derived from the first law of thermodynamics which states that the variation of total energy of a system is due to the work of acting forces and heat transmitted to the system.

Rate of change of energy	=	Rate of of heat change	+	Rate of of done work on the system
-----------------------------	---	---------------------------	---	---------------------------------------

The total energy per unit mass (specific energy) can be defined as the sum of internal (thermal) energy i and specific kinetic energy $v^2/2$. Internal energy is due to molecular movement. Gravitation is considered as an energy source term, i.e. a body force which does work on the fluid element as it moves through the gravity field. Starting again from the Eulerian general balance equation (1.11) now for total energy density

$$\psi = \rho e = \rho \left(i + \frac{v^2}{2} \right) \quad (1.82)$$

we have (in work per time).

$$\frac{\partial}{\partial t} \int_{\Omega} \rho \left(i + \frac{v^2}{2} \right) d\Omega + \oint_{\partial\Omega} \Phi^e \cdot d\mathbf{S} = \int_{\Omega} q_V^e d\Omega + \oint_{\partial\Omega} \mathbf{q}_S^e \cdot d\mathbf{S} \quad (1.83)$$

1.4.2 Energy Fluxes

Concerning energy fluxes we can distinguish between advective energy flux

$$\Phi_A^e = \mathbf{v} \rho e \quad (1.84)$$

and diffusive energy flux (heat conduction).

$$\Phi_D^e = -\frac{c_p}{c_V} \lambda \rho \nabla T = -\lambda \nabla T \quad (1.85)$$

1.4.3 Energy Sources

Concerning energy sources we can distinguish between volume and surface ones. The first one is work of volume forces \mathbf{f}^e and heat sources (radiation, chemical reactions).

$$q_V^e = \rho \mathbf{f}^e \cdot \mathbf{v} + q_H \quad (1.86)$$

The latter one which is the rate of work done by surface forces result from internal shear stresses (heat dissipation)

$$\mathbf{q}_S^e = \boldsymbol{\sigma} \cdot \mathbf{v} = -p \mathbf{v} + \boldsymbol{\tau} \cdot \mathbf{v} \quad (1.87)$$

1.4.4 Integral Energy Balance Equation

If equating the rate of change of energy of a fluid element to the sum of energy fluxes and sources, the energy conservation law becomes

$$\begin{aligned} \frac{\partial}{\partial t} \int_{\Omega} \rho e d\Omega &= - \underbrace{\oint_{\partial\Omega} \rho e (\mathbf{v} \cdot d\mathbf{S})}_{\text{Energy flux}} + \underbrace{\oint_{\partial\Omega} \lambda \nabla T \cdot d\mathbf{S}}_{\text{Diffusive heat flux}} + \underbrace{\oint_{\partial\Omega} (\boldsymbol{\sigma} \cdot \mathbf{v}) \cdot d\mathbf{S}}_{\text{Work of internal forces}} \\ &\quad + \underbrace{\int_{\Omega} \rho \mathbf{f}^e \cdot \mathbf{v} d\Omega}_{\text{Work of external forces}} + \underbrace{\int_{\Omega} q_H d\Omega}_{\text{Internal heat sources}} \end{aligned} \quad (1.88)$$

1.4.5 Differential Energy Balance Equation

Conducting now the usual procedure (Gauss-Ostrogradskian integral theorem) we obtain the following conservation equation for energy in a differential form

$$\underbrace{\frac{\partial \rho e}{\partial t}}_1 = - \underbrace{\nabla \cdot (\rho e \mathbf{v})}_2 + \underbrace{\nabla \cdot (\lambda \nabla T)}_3 + \underbrace{\rho \mathbf{f}^e \cdot \mathbf{v}}_4 + \underbrace{q_H}_5 + \underbrace{\nabla \cdot (\boldsymbol{\sigma} \cdot \mathbf{v})}_6 \quad (1.89)$$

1,2 - Rate of energy change and advective energy flux

The first terms of the above equation can be written in expanded form using partial integration

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot (\rho e \mathbf{v}) = e \underbrace{\left[\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) \right]}_{\text{Mass conservation}} + \rho \left[\frac{\partial e}{\partial t} + \mathbf{v} \cdot \nabla e \right] = \rho \frac{de}{dt} \quad (1.90)$$

The first term of the middle part is equal to zero due to mass conservation (1.29).

6 - Work of internal forces

Partial integration yields

$$\begin{aligned} \frac{\partial}{\partial x_i} (\sigma_{ij} v_j) &= \sigma_{ij} \frac{\partial v_j}{\partial x_i} + v_j \frac{\partial \sigma_{ij}}{\partial x_i} \\ \nabla \cdot (\boldsymbol{\sigma} \cdot \mathbf{v}) &= \boldsymbol{\sigma} \cdot (\nabla \mathbf{v}) + \mathbf{v} \cdot (\nabla \cdot \boldsymbol{\sigma}) \end{aligned} \quad (1.91)$$

The total rate of work per unit volume done on the fluid by surface stresses is given by

$$\nabla \cdot (\boldsymbol{\sigma} \cdot \mathbf{v}) = \nabla \cdot [(-p\mathbf{I} + \boldsymbol{\tau}) \cdot \mathbf{v}] = \underbrace{\nabla \cdot (-p\mathbf{I} \cdot \mathbf{v})}_{\text{Reversible work}} + \underbrace{\nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{v})}_{\text{Irreversible work}} \quad (1.92)$$

This equation describing the work done by internal forces (friction) can be splitted into a term of reversible work and a term of irreversible work.

$$\begin{aligned} \nabla \cdot (-p\mathbf{I} \cdot \mathbf{v}) &= -\frac{\partial}{\partial x_i} (p I_{ij} v_j) \\ &= -\frac{\partial}{\partial x_j} (p v_j) = -\nabla \cdot (p \mathbf{v}) = -\operatorname{div}(p \mathbf{v}) \end{aligned} \quad (1.93)$$

The dissipation term is acting as irreversible heat source

$$\varepsilon_V = \nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{v}) = \frac{\partial}{\partial x_i} (\tau_{ij} v_j) \quad (1.94)$$

Taking the above relationships into account we can rewrite the energy conservation equation in following form.

$$\rho \frac{de}{dt} = \nabla \cdot (\lambda \nabla T) + \rho \mathbf{f}^e \cdot \mathbf{v} + q_H - \nabla \cdot (p \mathbf{v}) + \nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{v}) \quad (1.95)$$

1.4.6 Balance Equation of Kinetic Energy

To separate contributions of kinetic and internal energy to the total energy balance, we consider first the conservation of kinetic energy. The balance equation can be obtained by multiplication of momentum balance equation (1.41) by momentum vector.

$$\rho \frac{d\mathbf{v}}{dt} \cdot \mathbf{v} = \rho \frac{\partial \mathbf{v}}{\partial t} \cdot \mathbf{v} + \rho[(\mathbf{v} \cdot \nabla)\mathbf{v}] \cdot \mathbf{v} = \rho \mathbf{f}_e \cdot \mathbf{v} + (\nabla \cdot \boldsymbol{\sigma}) \cdot \mathbf{v} = 0 \quad (1.96)$$

The relation to kinetic energy $\rho \mathbf{v}^2 / 2$ becomes obvious if the left side of the above equation is transformed to following form

$$\frac{\rho}{2} \frac{d\mathbf{v}^2}{dt} = \frac{\rho}{2} \frac{\partial \mathbf{v}^2}{\partial t} + \frac{\rho}{2} (\mathbf{v} \cdot \nabla) \mathbf{v}^2 \quad (1.97)$$

where $\mathbf{v} \cdot \mathbf{v} = \mathbf{v}^2 = u^2 + v^2 + w^2$. The kinetic energy balance equation is therefore

$$\frac{\rho}{2} \frac{d\mathbf{v}^2}{dt} = \rho \mathbf{f}_e \cdot \mathbf{v} + (\nabla \cdot \boldsymbol{\sigma}) \cdot \mathbf{v} \quad (1.98)$$

1.4.7 Balance Equation of Internal Energy

The balance equation for internal energy can be obtained by subtracting the kinetic energy equation from the total energy equation.

$$\rho \frac{di}{dt} = \nabla \cdot (\lambda \nabla T) + q^H + \boldsymbol{\sigma} \cdot (\nabla \mathbf{v}) \quad (1.99)$$

In the case of incompressible fluid we have $\nabla \cdot \mathbf{v} = 0$; $i = cT$, where c is specific heat. This assumption allows us to rewrite the internal energy equation as heat transport equation.

$$\rho c \frac{dT}{dt} = \nabla \cdot (\lambda \nabla T) + q^H + \boldsymbol{\tau} \cdot \nabla \mathbf{v} \quad (1.100)$$

Enthalpy and Entropy

Enthalpy and entropy are state quantities for a thermodynamic system. For compressible fluids internal energy is commonly written in terms of specific enthalpy

$$h_0 = i + \frac{p}{\rho} \quad (1.101)$$

or total enthalpy

$$h = i + \frac{p}{\rho} + \frac{v^2}{2} \quad (1.102)$$

Table 1.2: Example for state quantities

Quantity	Enthalpy
Steam	2677 kJ/kg
Water	419 kJ/kg

Specific entropy is introduced to separate the terms of irreversible heat production. It can be defined by the following equation

$$Tds = di + pd\left(\frac{1}{\rho}\right) = dh - \frac{dp}{\rho} \quad (1.103)$$

or considering time variation of the above differential terms

$$\rho T \frac{ds}{dt} = \frac{di}{dt} + p \frac{d}{dt}\left(\frac{1}{\rho}\right) \quad (1.104)$$

Another definition is given by

$$di = Tds - pdV \quad (1.105)$$

Using now the balance equation for internal energy we obtain

$$\begin{aligned} \rho T \frac{ds}{dt} &= \nabla \cdot (\lambda \nabla T) + q^H + \tau \cdot \nabla \mathbf{v} - p \underbrace{\left(\frac{d\rho}{dt} + \rho \nabla \cdot \mathbf{v} \right)}_{\text{Mass conservation}} \\ &= \underbrace{\nabla \cdot (\lambda \nabla T) + q^H}_{\text{Reversible heat addition}} + \underbrace{\tau \cdot \nabla \mathbf{v}}_{\text{Irreversible heat production}} \end{aligned} \quad (1.106)$$

The last term of the second part can be deleted due to mass conservation. Since irreversible heat production term is non-negative, from the above equation results that entropy will always increase in an closed (adiabatic) system where reversible heat addition is equal to zero. The above equation states the second principle of thermodynamics on entropy increase.

1.5 Problems

General Conservation Law

- 1 What are the two fundamental concepts for description of motion in mechanics ? Explain the conceptual difference between them ? [sec. 1.1.1]
- 2 What are the fundamental conservation quantities in mechanics ? [sec. 1.1.2]
- 3 Derive the general conservation law (1.7). [sec. 1.1.3]
- 4 Explain mathematical and physical meaning of the Gauss-Ostrogradskian integral theorem (1.7). Give a physical interpretation of the divergence of fluxes $\nabla \cdot \Phi$. Hint: Use eqn.(1.15). [sec. 1.1.4]
- 5 What is the meaning of Reynolds transport theorem concerning the Lagrangian and Eulerian concepts for motion. [sec. 1.1.5]
- 6 What are the two fundamental flux types ? Explain the physical reason for diffusion processes using the term mass-weighted velocity of quantity. [sec. 1.1.6]
- 7 Derive the general conservation law in differential form from its integral expression (1.24). [sec. 1.1.7]

Mass Conservation

- 8 Derive the integral balance equation of mass from the general conservation law (1.11). [sec. 1.2]
- 9 Derive the Lagrangian form of the mass conservation equation (1.29). Hint: Use chain rule and material derivative $d/dt = \partial/\partial t + \mathbf{v} \cdot \nabla$. What assumption is required for mass conservation $d\rho/dt = 0$. [sec. 1.2]

Momentum Conservation

- 10 Derive the integral balance equation of linear momentum from the general conservation law (1.11). [sec. 1.3.1]
- 11 Explain the physical background of the constitutive equation for the stress tensor of fluids (1.44). [sec. 1.3.2]
- 12 What are the four basic movements of fluid elements ? [sec. 1.3.3]
- 13 Explain the physical difference between the Euler (1.75) and Navier-Stokes (12.1) equations. [sec. 1.3.4-6]
- 14 What assumptions has to be applied to the Navier-Stokes equations (12.1) to obtain the Stokes equation (1.79) ? [sec. 1.3.7]

- 15** What assumptions has to be applied to the Navier-Stokes equations (12.1) to obtain the Darcy equation (1.81) ? [sec. 1.3.8]
- 16** Give some examples of flows which are described by Euler equations (1.75), Navier-Stokes equations (12.1), Stokes equation (1.79), and Darcy equation (1.81). [sec. 1.3.5-8]

Energy Conservation

- 17** What does the first law of thermodynamics postulate ? [sec. 1.4.1]
- 18** What are the two parts of total energy ? [sec. 1.4.1] What balance equations can be derived from each of them ? [sec. 1.4.5-6]
- 19** Derive the integral balance equation of total energy from the general conservation law (1.11). Explain the physical meaning of its terms. [sec. 1.4.4]
- 20** On which balance equation the Bernoulli equation is based on ? [sec. 1.4.6]

Chapter 2

Turbulence

2.1 Physics of Turbulence

The Reynolds number of flow

$$Re = \frac{v^* L^*}{\nu^*} \quad (2.1)$$

gives a measure of the importance of inertial related to viscous forces. Experiments show that all flows become unstable above a certain Reynolds number. Below values of the so-called critical Reynolds number Re_{crit} the flow is smooth and adjacent layers of fluid slide past each other in an orderly regime. This regime is called laminar flow. At Reynolds numbers larger than the critical value a complicated series of physical events takes place which eventually result in a radical change of the flow behavior. Finally, the flow becomes turbulent, i.e. velocity and other flow properties become chaotic and random. The flow is then unsteady even with constant boundary conditions. Turbulence is a kind of a chaotic and random state of motion. Nevertheless, velocity and pressure vary continuously with time within substantial regions of flow. Velocity fluctuations associated with turbulence give rise to additional stresses on the fluid - so-called Reynolds stresses. Examples of turbulent flows are: free turbulent flows (jet flow), turbulent boundary layer flows.

The random character of turbulent flow requires statistic techniques to represent flow properties (section 2.2). All properties ψ are decomposed into a steady mean value $\bar{\psi}$ and fluctuation components ψ' describing the random process

$$\psi = \bar{\psi} + \psi' \quad (2.2)$$

In general it is of great interest to describe turbulent flow by mean values $\bar{\psi}$ of flow properties.

Eddies of turbulent flows

Turbulent flow reveals rotational eddies of a large variety of length scales. The existence of turbulent eddies provides an intensive mixing process. As a consequence, mass, momentum, and heat are effectively exchanged. Such effective mixing mechanisms give rise to high values of corresponding diffusion coefficients. Large eddy Reynolds number is as high as the ordinary Reynolds number if the characteristic velocity and length of large eddies are of the same order as the flow velocity and length scale of the mean flow. Following these large eddies are dominated by inertia effects and viscous effects are small. The smallest scale of motion (smallest eddies) which can occur in a turbulent flow is determined by the fluids viscosity. Viscous effects are important for small Reynolds numbers near to unity. For $\nu = 10^{-6} m^2 s^{-1}$ and eddy frequencies of about $f = 10$ kHz typical length scales are in the order of $L = 0.01 - 0.1 m$.

Structure of turbulent flows: Typical structures of turbulent flows are the mean flow itself, large and small eddies. Large eddies are highly anisotropic formatted, they are strongly influenced by the mean flow, whereas small eddies behave more isotropic (see Fig. 2.1 and Fig. 2.2).

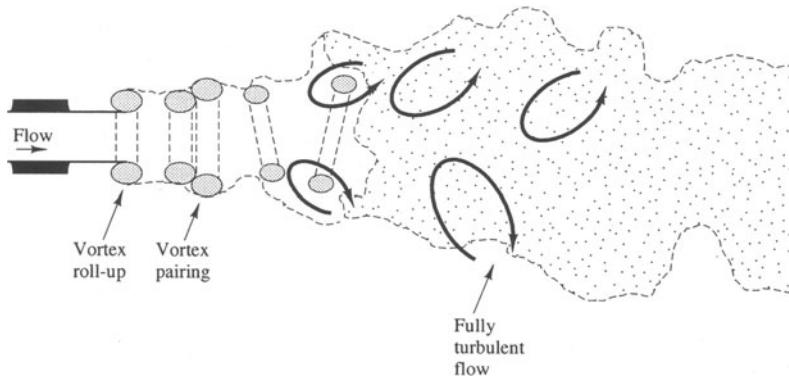


Figure 2.1: Structure of turbulent flow - Theory (Versteeg & Malalasekera 1995)



Figure 2.2: Structure of turbulent flow - Experiment (Versteeg & Malalasekera 1995)

Energy of turbulent flows

Fig. 2.3 shows the energy spectrum of turbulent flow and connected wave numbers. Mechanic work is performed against the work of viscous stresses. Therefore, the energy associated with the motion of eddies is dissipated and converted to thermal energy. This permanent dissipation process results in a consecutive kinetic energy loss. Stretching work on large eddies done by the mean flow provides the energy to maintain turbulence.

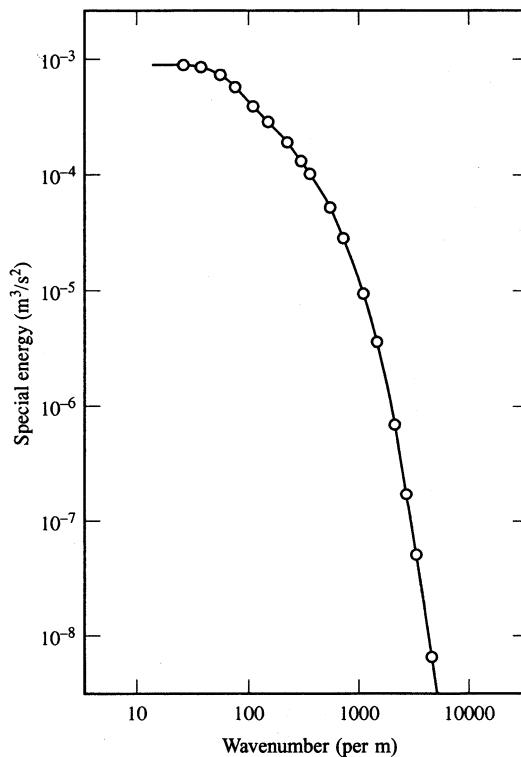


Figure 2.3: Energy spectrum of turbulence (Versteeg & Malalasekera 1995)

Hydrodynamic stability: Linear theory of hydrodynamic stability analyses the stability of laminar flow due to small disturbances, i.e. its decay or amplification.

Figures 2.4 and 2.5 illustrate the transition process from laminar to turbulent regimes. The characteristic steps are:

1. Amplification of small disturbances
2. Areas with rotational flow structures

3. Formation of intensive small motions (spots)
4. Growth of instabilities up to fully developed turbulent flow

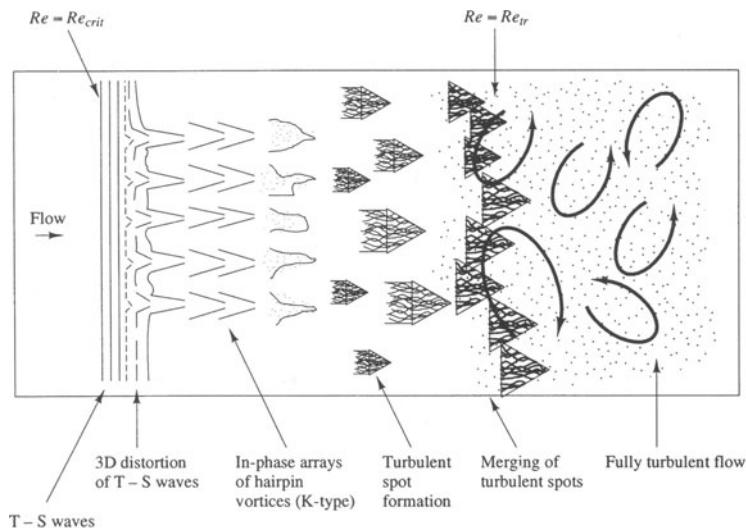


Figure 2.4: Transition process from laminar to turbulent flow - Theory

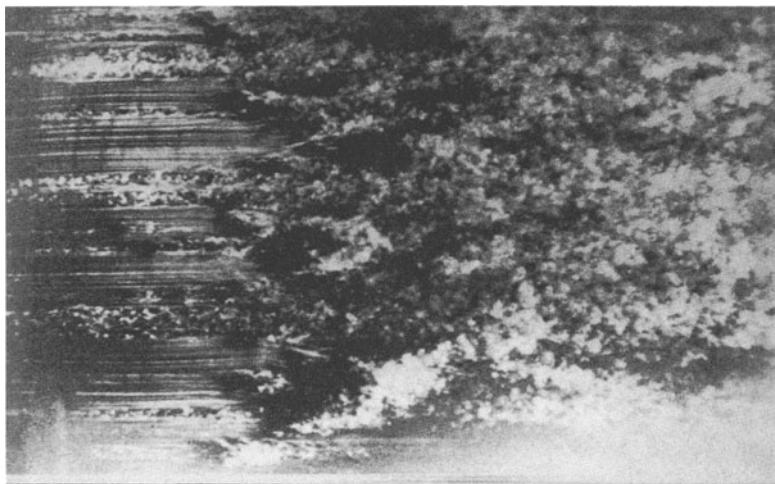


Figure 2.5: Transition process from laminar to turbulent flow - Experiment

2.2 Reynolds Equations

Engineers need computational procedures which can supply adequate information about turbulent flow processes, but which avoid the need to model all eddies and fluctuations in the flow. Therefore, we are interested in the mean flow. The procedure described in the following yields time-averaged flow properties (e.g. mean velocities) and the mean flow equation for turbulent flow. In performing the time-averaging procedure we neglect all details concerning the state of flow contained in the instantaneous fluctuations. As a result of the time-averaging procedure we obtain six additional unknowns - the Reynolds stresses for the Navier-Stokes equations and extra terms of turbulent transport.

We define the mean $\bar{\psi}$ of a flow property ψ as follows

$$\bar{\psi} = \frac{1}{\Delta t} \int_0^{\Delta t} \psi dt \quad (2.3)$$

Therefore, the flow property can be decomposed in a mean value and turbulent fluctuations

$$\psi = \bar{\psi} + \psi' \quad (2.4)$$

The averaging time step Δt must be large enough to capture the slowest variations of property ψ (due to the largest eddies). The derivation procedure starts with averaging the corresponding balance equation (1.25).

$$\frac{1}{\Delta t} \int_0^{\Delta t} \left\{ \frac{\partial \psi}{\partial t} + \nabla \cdot \Phi^\psi - q^\psi \right\} dt \quad (2.5)$$

Averaging rules

To derive mean flow equations we need means of several quantities such as combinations of fluid properties, derivatives, and integrals.

$$\begin{aligned} \overline{\psi'} &= 0 \\ \overline{\bar{\psi}} &= \bar{\psi} \\ \overline{\nabla \bar{\psi}} &= \nabla \bar{\psi} \\ \overline{\nabla \cdot \bar{\psi}} &= \nabla \cdot \bar{\psi} \\ \overline{\nabla \cdot \alpha \bar{\psi}} &= \nabla \cdot \overline{\alpha \bar{\psi}} = \nabla \cdot \alpha \bar{\psi} + \nabla \cdot \overline{\alpha' \psi'} \\ \overline{\nabla \cdot \nabla \bar{\psi}} &= \nabla \cdot \nabla \bar{\psi} \\ \overline{\int \psi dV} &= \int \bar{\psi} dV \end{aligned} \quad (2.6)$$

For flow quantities $\psi_i = \bar{\psi}_i + \psi'_i$ it is obvious that

$$\begin{aligned}\overline{\psi_1 + \psi_2} &= \bar{\psi}_1 + \bar{\psi}_2 \\ \overline{\psi_1 \psi_2} &= \bar{\psi}_1 \bar{\psi}_2 + \overline{\psi'_1 \psi'_2} \\ \overline{\psi_1 \bar{\psi}_2} &= \bar{\psi}_1 \bar{\psi}_2 \\ \overline{\psi'_1 \bar{\psi}_2} &= 0\end{aligned}\tag{2.7}$$

Applying the above averaging rules to continuity, Navier-Stokes, and transport equations we obtain the **Reynolds equations** of turbulent incompressible flow and transport.

Time-averaged continuity equation

$$\nabla \cdot \bar{\mathbf{v}} = 0\tag{2.8}$$

Time-averaged momentum equation

The derivation of the Reynolds equations starts with averaging the Navier-Stokes equation 12.1.

$$\frac{1}{\Delta t} \int_0^{\Delta t} \left\{ \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = \mathbf{f}^e - \frac{1}{\rho} \nabla p + \nu \Delta \mathbf{v} \right\} dt\tag{2.9}$$

Again the Boussinesq approximation is used: $\rho' = 0$. We consider correlation terms given by

$$\overline{(\mathbf{v}' \cdot \nabla) \mathbf{v}'} = \overline{v'_i \frac{\partial v'_j}{\partial x_i}} = \overline{\frac{\partial v'_i v'_j}{\partial x_i}} = \nabla \cdot \overline{(\mathbf{v}' \mathbf{v}')}, \quad j = 1, 2, 3\tag{2.10}$$

$$\begin{aligned}\overline{u' \frac{\partial u'}{\partial x}} + \overline{v' \frac{\partial u'}{\partial y}} + \overline{w' \frac{\partial u'}{\partial z}} &= \overline{\frac{\partial u' u'}{\partial x}} - \overline{u' \frac{\partial u'}{\partial x}} + \overline{\frac{\partial u' v'}{\partial y}} - \overline{u' \frac{\partial v'}{\partial y}} + \overline{\frac{\partial u' w'}{\partial z}} - \overline{u' \frac{\partial w'}{\partial z}} \\ &= \overline{\frac{\partial u' u'}{\partial x}} + \overline{\frac{\partial u' v'}{\partial y}} + \overline{\frac{\partial u' w'}{\partial z}} - \underbrace{u' \left(\frac{\partial u'}{\partial x} + \frac{\partial v'}{\partial y} + \frac{\partial w'}{\partial z} \right)}_{\text{Mass conservation}}\end{aligned}\tag{2.11}$$

We obtain the following time-averaged momentum equation.

$$\frac{\partial \bar{\mathbf{v}}}{\partial t} + (\bar{\mathbf{v}} \cdot \nabla) \bar{\mathbf{v}} = \bar{\mathbf{f}}^e - \frac{1}{\rho} \nabla \bar{p} + \underbrace{\nu \Delta \bar{\mathbf{v}}}_{2\nu \nabla \bar{\mathbf{D}}} - \nabla \cdot \overline{\mathbf{v}' \mathbf{v}'}$$

The new term involves products of velocity fluctuations. We have placed this term to the right-hand-side and we consider it as additional turbulent stresses. These stresses are called **Reynolds stresses**. They build a symmetric tensor of normal and shear stresses. Finally, we need additional equations to determine the six new unknown Reynolds stresses. This is the objective of turbulence modeling (section 2.3).

$$-\bar{\mathbf{v}'\mathbf{v}'} = -v'_i v'_j = - \begin{bmatrix} \bar{u'u'} & \bar{u'v'} & \bar{u'w'} \\ \bar{v'u'} & \bar{v'v'} & \bar{v'w'} \\ \bar{w'u'} & \bar{w'v'} & \bar{w'w'} \end{bmatrix} \quad (2.13)$$

Details of turbulence structure are very much dependent on the flow itself. The Reynolds stresses are also affected by the flow itself. Frequently, viscous and turbulent stresses are written in following form.

$$\nu \Delta \bar{\mathbf{v}} - \nabla \cdot \bar{\mathbf{v}'\mathbf{v}'} = \nabla \cdot (\nu \nabla \bar{\mathbf{v}} - \bar{\mathbf{v}'\mathbf{v}'}) = \nabla \cdot (\mathbf{D} - \mathbf{T}) \quad (2.14)$$

Time-averaged transport equation of a scalar quantity (mass or heat)

$$\frac{\partial \bar{\psi}}{\partial t} + (\bar{\mathbf{v}} \cdot \nabla) \bar{\psi} = \alpha \Delta \bar{\psi} - \alpha \cdot \bar{\mathbf{v}'\psi'} \quad (2.15)$$

We have three additional turbulent transport terms.

$$\bar{u'\psi'}, \bar{v'\psi'}, \bar{w'\psi'} \quad (2.16)$$

2.3 Turbulence Models

The problem of turbulence modeling consists in the wide range of length scales in turbulent flows (small eddies down to $10 \mu\text{m}$ size). For direct simulation of turbulence spatial discretizations up to 10^{12} grid points will be needed to describe all processes at all length scales. Moreover, there is also a wide range of time scales. Fastest events take place with a frequency of about 10 kHz, which results in computational time steps of 0.1 seconds. Direct turbulence simulation is beyond the capabilities of today computers. Therefore, averaging techniques are used to reduce the computational expense of turbulence simulation. Figure 2.6 shows most commonly turbulence models.

Unfortunately the complexity of turbulence precludes simple formulas for calculating the extra turbulence terms. It is therefore a main task of turbulence modeling to derive expressions and to develop computational schemes of sufficient accuracy to calculate the Reynolds stresses and the turbulence transport

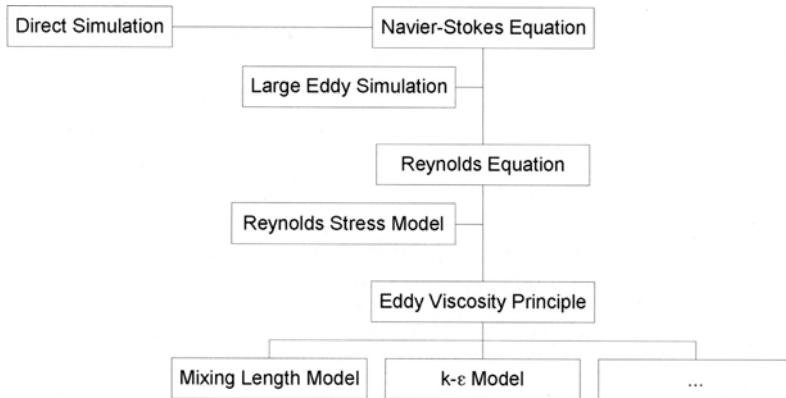


Figure 2.6: Overview on turbulence models

Table 2.1: Classification of turbulence models (Fig. 2.6)

Classical models	based on Reynolds equations time-averaged Navier-Stokes equations
	Mixing length model (zero equation model) $k - \varepsilon$ model (two equation model) Reynolds stress model Algebraic stress model
Large eddy simulation	based on space-filtered equations space-averaged Navier-Stokes equations
Direct simulation	based on Navier-Stokes equations

terms. For most engineering problems it is not necessary to know all details of turbulent flows. Only the effects of turbulence on the mean flow are sought.

Classical turbulence models are based on the time-averaged Reynolds equations. The mixing length model (section 2.3.1) and the $k - \varepsilon$ model (section 2.3.2) belong to the presently most widely used models in computational practice. It is observed in experiments that turbulent structures decay in isothermal flows as there is shear. Mixing length as well as $k - \varepsilon$ models are based on the assumption that there exists an analogy between the action form of viscous and

Reynolds stresses, i.e. Reynolds stresses are assumed to be proportional to the deformation rate of the fluid element. More than hundred years ago Boussinesq (1877) proposed the following expression for Reynolds stresses which is denoted as the **eddy viscosity principle**.

$$-\overline{v'_i v'_j} = \nu_t \left(\frac{\partial \bar{v}_i}{\partial x_j} + \frac{\partial \bar{v}_j}{\partial x_i} \right) = 2\nu_t \bar{\mathbf{D}} \quad (2.17)$$

$$\tau_{ij} = -\rho \overline{v'_i v'_j} = 2\mu_t \bar{\mathbf{D}} \quad (2.18)$$

where $\nu_t = \mu_t/\rho$ is the kinematic **turbulent viscosity** and μ_t is the dynamic turbulent viscosity. From equation (2.17) results that

$$2\nu_t \nabla \cdot \bar{\mathbf{v}} = 0 \quad (2.19)$$

However it must be

$$-\overline{v'_i v'_i} = -\overline{(u'^2 + v'^2 + w'^2)} \quad (2.20)$$

Therefore, equation (2.17) has to be extended in the following manner.

$$\nu_t \left(\frac{\partial \bar{v}_i}{\partial x_j} + \frac{\partial \bar{v}_j}{\partial x_i} \right) - \frac{1}{3} \overline{v'_i v'_j} I_{ij} = \nu_t \bar{D}_{ij} - \frac{2}{3} k' I_{ij} \quad (2.21)$$

where k' is the turbulent kinetic energy (section 6.3.2).

Note, the turbulent viscosity is not a constant. The eddy viscosity principle is based on the assumption that velocity fluctuations act as shear stresses and any deformation of fluid elements is volume-conserving. The mean momentum equation for turbulent flow can now be written as

$$\frac{\partial \bar{\mathbf{v}}}{\partial t} + (\bar{\mathbf{v}} \cdot \nabla) \bar{\mathbf{v}} = \rho \bar{\mathbf{f}}^e - \frac{1}{\rho} \nabla \bar{p} + \nabla \cdot [(\nu + \nu_t) \nabla \bar{\mathbf{v}}] \quad (2.22)$$

In analogy it is assumed that turbulence transport terms are proportional to the gradient of the mean value of a scalar transport quantity ψ .

$$-\rho \overline{v'_i \psi'} = \alpha_t \frac{\partial \bar{\psi}}{\partial x_i} \quad (2.23)$$

where α_t is the **turbulent diffusivity**. The transport equation for turbulent flow can now be written as

$$\frac{\partial \bar{\psi}}{\partial t} + (\bar{\mathbf{v}} \cdot \nabla) \bar{\psi} = \nabla \cdot [(\alpha + \alpha_t) \bar{\psi}] + q^\psi \quad (2.24)$$

The ratio of turbulent viscosity and turbulent diffusivity is denoted as the Prandtl-Schmidt number.

$$\sigma_t = \frac{\nu_t}{\alpha_t} \quad (2.25)$$

Numerous experiments have shown that this ratio is constant nearly to unity.

The **large eddy concept** is a turbulence model where the time-dependent Navier-Stokes equations are solved for the mean flow and the largest eddies. The effects of smaller eddies are modeled in a specific way. It is argued that the largest eddies (which contain the most of energy) strongly interact with the mean flow and, therefore, this approach is well suited to describe the main effects of turbulence. Large eddy simulations are based on space-filtered equations. This simulation technique is at present at the research stage and computations are very costly.

2.3.1 Mixing Length Model

As mentioned earlier simple turbulence models are based on the assumption that mean flow is affected only by the large eddies. Let us denote L^* the characteristic length scale of these eddies, containing most of the kinetic energy of the flow, interacting with the mean flow. The mixing length model (MLM) is based on dimensional analysis of turbulence. It is assumed, that turbulent viscosity can be expressed as the product of a turbulent velocity scale v^* and the above characteristic length scale L^* .

$$\nu_t = C_1 v^* L^* \quad \left[\frac{L^2}{T} \right] \quad (2.26)$$

The characteristic turbulent velocity v^* can be easily estimated for a 2-D flow if the only significant mean velocity gradient $\partial \bar{u} / \partial y$ is normal to the mean flow direction (e.g. boundary layer flow) and the characteristic eddy length is L^*

$$v^* = C_2 L^* \left| \frac{\partial \bar{u}}{\partial y} \right| \quad \left[\frac{L}{T} \right] \quad (2.27)$$

where C_1 and C_2 are dimensionless coefficients. Combining the above equations we obtain the following expression for turbulent viscosity.

$$\nu_t = \underbrace{C_1 C_2 (L^*)^2}_{l_m^2} \left| \frac{\partial \bar{u}}{\partial y} \right| \quad (2.28)$$

where l_m is the mixing length. This is denoted as **Prandtl's mixing length model**. Now Reynolds stresses can be written e.g. as

$$\tau_{xy} = \tau_{yx} = -\overline{u'v'} = l_m^2 \left| \frac{\partial \bar{u}}{\partial y} \right| \left| \frac{\partial \bar{u}}{\partial y} \right| \quad (2.29)$$

This means turbulent Reynolds stresses are functions of the flow. For some classes of turbulent flows (e.g. free turbulent flow, turbulent boundary layer flow) simple algebraic formulas can be achieved, as given in the below table from Rodi (1980).

Table 2.2: Mixing lengths for 2-D turbulent flows

Flow type	Mixing length l_m	Characteristic length L^*
Mixing layer	$0.07L^*$	Layer width
Jet	$0.09L^*$	Jet half width
Wake	$0.16L^*$	Wake half width

The mixing length model can be applied also to turbulent transport. The turbulent transport term is given by

$$-\overline{u'\psi'} = \alpha_t \frac{\partial \bar{\psi}}{\partial y} \quad (2.30)$$

The mixing length model is based on the assumption that turbulent properties develop in proportion to the mean flow length scale.

2.3.2 $k - \varepsilon$ Model

If advection and diffusion of turbulence properties becomes important we have to consider dynamics of turbulence. In this case (e.g. recirculating flows) a mixing length representation is no longer feasible. The $k - \varepsilon$ model was developed to represent mechanisms which are affected by kinetic energy. This approach is another way to determine the turbulent viscosity $\nu_t \approx v^* L^*$, where v^* and L^* are expressed by turbulent kinetic energy k' and dissipation ε .

k - Kinetic energy of turbulent flow

According to the statistic concept of turbulence, instantaneous kinetic energy is decomposed into a mean flow and turbulent ones.

$$k(t) = \bar{k} + k'(t) = \underbrace{\frac{1}{2}(\bar{\mathbf{v}} \cdot \bar{\mathbf{v}})}_{\text{Mean kinetic energy}} + \underbrace{\frac{1}{2}(\overline{\mathbf{v}' \cdot \mathbf{v}'})}_{\text{Turbulent kinetic energy}} \quad (2.31)$$

According to the **Kolmogorov-Prandtl equation**, turbulent viscosity and turbulent kinetic energy are connected by the following relationship

$$\nu_t = C_1 \sqrt{k' L^*} \quad (2.32)$$

ε - Dissipation of kinetic energy, rate of deformation

Mean and fluctuating components of dissipation are due to fluid deformation,

$$\varepsilon = 2\nu \mathbf{D} : \mathbf{D} \quad [L^2/T^3] \quad (2.33)$$

with overall the deformation tensor (1.54).

$$\mathbf{D} = D_{ij} = \bar{D}_{ij} + D'_{ij} = \frac{1}{2} \left(\frac{\partial \bar{v}_i}{\partial x_j} + \frac{\partial \bar{v}_j}{\partial x_i} \right) + \frac{1}{2} \left(\frac{\partial v'_i}{\partial x_j} + \frac{\partial v'_j}{\partial x_i} \right) \quad (2.34)$$

Equation for mean flow kinetic energy

As described in section 1.4 the governing equation for kinetic energy can be obtained by multiplying the momentum conservation equation (2.12) by velocity. Applying partial differentiation, we have for mean values of kinetic energy

$$\begin{aligned} \underbrace{\frac{\partial \rho \bar{k}}{\partial t}}_1 &= - \underbrace{\nabla \cdot (\rho \bar{k} \bar{\mathbf{v}})}_2 - \underbrace{\nabla \cdot (\bar{p} \bar{\mathbf{v}})}_3 + \underbrace{\nabla \cdot (2\mu \bar{\mathbf{D}} \cdot \bar{\mathbf{v}})}_4 - \underbrace{\nabla \cdot (\rho \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{v}})}_6 \\ &\quad - \underbrace{2\mu \bar{\mathbf{D}} \cdot \bar{\mathbf{D}}}_5 + \underbrace{\rho \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{D}}}_7 \\ &= \underbrace{\nabla \cdot (-\rho \bar{k} \bar{\mathbf{v}} - \bar{p} \bar{\mathbf{v}} + 2\mu \bar{\mathbf{D}} \cdot \bar{\mathbf{v}} - \rho \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{v}})}_{\text{Flux term}} - \underbrace{2\mu \bar{\mathbf{D}} \cdot \bar{\mathbf{D}} + \rho \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{D}}}_{\text{Dissipation term}} \end{aligned} \quad (2.35)$$

with:

1. Rate of change of \bar{k}
2. Transport of \bar{k} by advection
3. Transport of \bar{k} by pressure forces
4. Transport of \bar{k} by viscous stresses
5. Transport of \bar{k} by Reynolds stresses
6. Rate of dissipation of \bar{k} due to viscous stresses
7. Rate of dissipation of \bar{k} due to turbulent stresses

The total flux vector of mean turbulent energy is, therefore, given by

$$\Phi^{\bar{k}} = -\rho \bar{k} \bar{\mathbf{v}} - \bar{p} \bar{\mathbf{v}} + 2\mu \bar{\mathbf{D}} \cdot \bar{\mathbf{v}} - \rho \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{v}} \quad (2.36)$$

For incompressible flow with nearly constant fluid density, the equation for mean kinetic energy can be rewritten as.

$$\frac{\partial \bar{k}}{\partial t} + \bar{\mathbf{v}} \cdot \nabla \bar{k} = -\frac{\bar{\mathbf{v}}}{\rho} \cdot \bar{\nabla} p + \nabla \cdot (2\nu \bar{\mathbf{D}} \cdot \bar{\mathbf{v}} - \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{v}}) - 2\nu \bar{\mathbf{D}} \cdot \bar{\mathbf{D}} + \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{D}} \quad (2.37)$$

Equation for turbulent kinetic energy

The equation for turbulent kinetic energy is obtained from the Navier-Stokes and the Reynolds equations which are multiplied by the fluctuation velocity vector and then subtracted from each other, i.e. $k' = k - \bar{k}$

$$\frac{\partial \rho k'}{\partial t} = \nabla \cdot (-\rho k' \bar{v} - \bar{p} \bar{v} + 2\mu \bar{\mathbf{D}}' \cdot \mathbf{v}' - \rho \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{v}) - 2\mu \bar{\mathbf{D}}' \cdot \bar{\mathbf{D}}' + \rho \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{D}} \quad (2.38)$$

For incompressible flow with nearly constant fluid density, the equation for turbulent kinetic energy can be rewritten as

$$\frac{\partial k'}{\partial t} + \bar{v} \cdot \nabla k' = -\frac{\bar{\mathbf{v}}' \cdot \nabla p'}{\rho} + \nabla \cdot (\underbrace{2\nu \bar{\mathbf{D}}' \cdot \mathbf{v}'}_1 - \underbrace{\bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{v}}_2) - \underbrace{2\nu \bar{\mathbf{D}}' \cdot \bar{\mathbf{D}}'}_{3:\varepsilon'} + \underbrace{\bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{D}}}_4 \quad (2.39)$$

Terms 1 and 3 as well as 2 and 4 are derived from partial differentiation.

$$\begin{aligned} \nabla \cdot (\nu \bar{\mathbf{D}}' \cdot \mathbf{v}') &= (\nabla \cdot \nu \bar{\mathbf{D}}') \cdot \mathbf{v}' + \nu \bar{\mathbf{D}}' \cdot \bar{\mathbf{D}}' \\ \nabla(\bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{v}) &= \nabla(\bar{\mathbf{v}}' \bar{\mathbf{v}}') \cdot \bar{v} + \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{D}} \end{aligned} \quad (2.40)$$

The following relationships are used to simplify the $k - \varepsilon$ model

$$\begin{aligned} \varepsilon' &= 2\nu \bar{\mathbf{D}}' \cdot \bar{\mathbf{D}}' && \text{Dissipation} \\ \frac{\bar{\mathbf{v}}' \cdot \nabla p'}{\rho} + \nabla \cdot (2\nu \bar{\mathbf{D}}' \cdot \mathbf{v}' - \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{v}) &= \frac{\nu_t}{\sigma_k} \nabla k' && \text{Flux term} \\ \bar{\mathbf{v}}' \bar{\mathbf{v}}' \cdot \bar{\mathbf{D}} &= 2\nu_t \bar{\mathbf{D}} \cdot \bar{\mathbf{D}} && \text{Production} \end{aligned} \quad (2.17)$$

where ε' is the dissipation rate per unit mass which is caused by work done by small eddies against viscous stress. This term is the main destruction term of turbulence.

The idea behind the $k - \varepsilon$ model is to use turbulent kinetic energy and dissipation rate to define the characteristic velocity v^* and length scales L^* of turbulent flow as follows (Launder and Spalding 1974). From dimensional consideration we can argue

$$[k'] = [L^2/T^2] \implies v^* = k'^{1/2} \quad (2.41)$$

$$[\varepsilon'] = [L^2/T^3] \implies \varepsilon' = 2\nu \bar{\mathbf{D}} \cdot \bar{\mathbf{D}} = C_D \frac{k'^{3/2}}{L^*} \quad L^* = C_D \frac{k'^{3/2}}{\varepsilon'} \quad (2.42)$$

where C_D is a dimensionless factor. Physically, linking L^* and k' means that the rate at which large eddies extract energy from the mean flow is matched to the transfer of energy across the energy spectrum to small, dissipating eddies. Otherwise, the energy at some scales of turbulence could grow or diminish without limit. However, those artifacts are not observed in practice.

According to the mixing length concept, eddy viscosity can be defined now as

$$\nu_t = Cv^*L^* = C_\mu \frac{k'^2}{\varepsilon'} \quad (2.43)$$

where C_μ is a dimensionless constant.

Table 2.3: Coefficients of the $k - \varepsilon$ model (Rodi 1980)

$C_\mu = 0.09$	$\sigma_k = 1$	$\sigma_\varepsilon = 1.3$	$C_{1\varepsilon} = 1.44$	$C_{2\varepsilon} = 1.92$
----------------	----------------	----------------------------	---------------------------	---------------------------

Model equations for k and ε

Frequently k instead of k' for turbulent kinetic energy and ε instead of ε' for dissipation term are used to simplify the notation. Using the above relationships, the $k - \varepsilon$ model can be written as follows

$$\frac{\partial k}{\partial t} + \nabla \cdot (k \bar{v}) = \nabla \cdot \left(\frac{\nu_t}{\sigma_k} \nabla k \right) - \varepsilon + 2\nu_t \bar{\mathbf{D}} \cdot \bar{\mathbf{D}} \quad (2.44)$$

$$\frac{\partial \varepsilon}{\partial t} + \nabla \cdot (\varepsilon \bar{v}) = \nabla \cdot \left(\frac{\nu_t}{\sigma_\varepsilon} \nabla \varepsilon \right) + 2\nu_t C_{1\varepsilon} \frac{\varepsilon}{k} \bar{\mathbf{D}} \cdot \bar{\mathbf{D}} - C_{2\varepsilon} \frac{\varepsilon^2}{k} \quad (2.45)$$

The $k - \varepsilon$ model equations contain five adjustable constants $C_\mu, \sigma_k, \sigma_\varepsilon, C_{1\varepsilon}, C_{2\varepsilon}$. The values for these variables can be taken as constants for a wide range of turbulent flows (see Table 2.3).

Prandtl-Schmidt numbers σ_k and σ_ε connect the diffusivities of turbulent kinetic energy and dissipation to the eddy viscosity. It is assumed that production and destruction of dissipation are proportional to production and destruction terms of turbulent kinetic energy.

Chapter 3

Porous Media

3.1 Multiphase Media

Soil or rock can be considered as a multiphase medium consisting of a solid phase (solid matrix) and of one or more fluid phases (gas and liquids), which occupy the void space (Fig. 3.1). Fluids are immiscible, if a sharp interface is maintained between them. In general, a phase is defined as a part of a continuum, which is characterized by distinct material properties and by a well-defined set of thermodynamic state variables. State variables describe the physical behaviour at all points of the phase. They must vary continuously within the considered phase of a continuum. Phases are separated from each other by surfaces referred to as interphase boundaries. Transport of components may occur within a phase and/or across interphase boundaries. Those interphasic exchange processes between adjacent phases can result from diffusive and/or advective mechanisms.

In fact, it is impossible to describe the complex geometry of the solid matrix and the topology of the void space at the microscopic level, i.e. the topology of the pore space will never be known in detail. As a consequence, boundary conditions for a mathematical model cannot be stated at this scale, since they are not known at the microscopic level. Moreover, it will be extremely difficult to measure values of state variables at each point within a phase in order to observe processes, to calibrate and to verify any model. Finally, the complete formulation and resolution of balance equations at the microscopic level is impossible and may not be reasonable. Therefore, it is necessary to transform the problem from a microscopic scale to a macroscopic level. Starting from the microscopic balance equations for extensive quantities (masses, momentum, energy), this procedure is the subject of the theory of the porous medium (Bear 1972, Diersch 1985, Bear & Bachmat 1990, de Boer & Ehlers 1986, Lewis & Schrefler 1998, de Boer 2000). The entire problem is rewritten in terms of averages of microscopic quantities, which have measurable values. The resulting

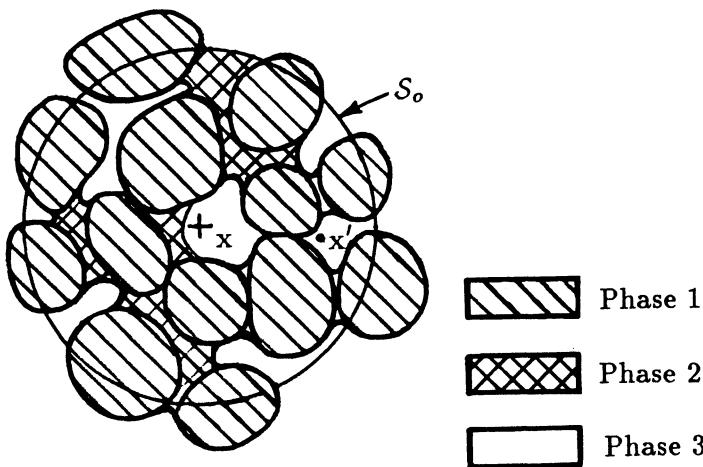


Figure 3.1: Conceptual model of a porous medium (Bear and Bachmat, 1990)

macroscopic model is referred to as the continuum approach. This conceptual model implies that a real system is replaced by a number of overlapping continua representing the corresponding phases. It is assumed that each phase, occupying a certain part of the porous domain, is regarded as a continuum. These individual phases interact with each other at any place within the entire domain, because they are present at each point within the porous medium, i.e. all phases are completely mixed.

$$\Omega_0 = \sum_{\alpha} \Omega^{\alpha} \quad (3.1)$$

In addition to the porous medium approach, there exist different types of structural models for fractured rocks, which characterize the degree of inhomogeneity: the fractured medium and the fractured porous medium. The term fractured medium means that only the fractures are important for the considered process, so blocks surrounded by the fractures may be neglected in the model. The term fractured porous medium means that both fracture system and porous matrix are significant for the considered process. The domain of a fractured porous medium consists of two subdomains, representing heterogeneities at different scales, i.e. the diameter of pores in the matrix and the characteristic length of fractures.

Appropriate averaging rules must be defined in order to realize the above described transformation from a microscopic to a macroscopic level. For this purpose, a well-defined sample size of an averaging volume must be found, which is referred to as the representative elementary volume (abbreviated REV). On the one hand, this averaging unit has to be sufficiently large, so that the in-

fluence of microscopic inhomogeneity on the values of averaged (macroscopic) quantities will vanish, i.e. they become independent of size, shape, and orientation of the REV. On the other hand, the REV must be small enough to reflect the macroscopic heterogeneity. In particular, the REV must be much smaller than the domain of interest, which may vary in size for a flow or a transport problem, respectively. From the mathematical point of view, the macroscopic (averaged) quantities must be continuous and differentiable functions (in space and in time), so that solutions of the governing differential balance equations can be determined. Finally, the continuum approach cannot be employed unless a common range of a REV can be selected for all material properties (e.g. porosity, permeability, dispersivity) as well as for all relevant state variables. This requirement is important with respect to the different conceptual models for fractured rock, which are introduced in the following.

We consider two-phase systems (one fluid phase, one solid phase) such as water-saturated porous media, the groundwater zone (section 3.3).

3.2 Macroscopic Equations

Statistic approach

In fact, it is impossible to describe the complex geometry of the solid matrix and the topology of the void space at the microscopic level, i.e. the topology of the pore space will never be known in detail. Therefore, a statistical approach is used for the derivation of balance equations at a macroscopic level. The physical property of a porous medium is decomposed in phase-related mean values $\overline{\psi^\alpha}$ and local fluctuations ψ'^α .

$$\psi^\alpha = \overline{\psi^\alpha} + \psi'^\alpha \quad (3.2)$$

Averaging volume

An appropriate averaging volume is called the Representative Elementary Volume (REV) (Fig. 3.2)

Averaging operator

Several averaging procedures can be defined (Bear & Bachmat 1990). As an example we consider volumetric averaging which is also denoted as the concept of volume fractions. The volumetric averaging operator is given by

$$\overline{\psi^\alpha}^\alpha = \frac{1}{\epsilon^\alpha \Omega_0} \int_{\Omega_0} f^\alpha \psi^\alpha d\Omega \quad (3.3)$$

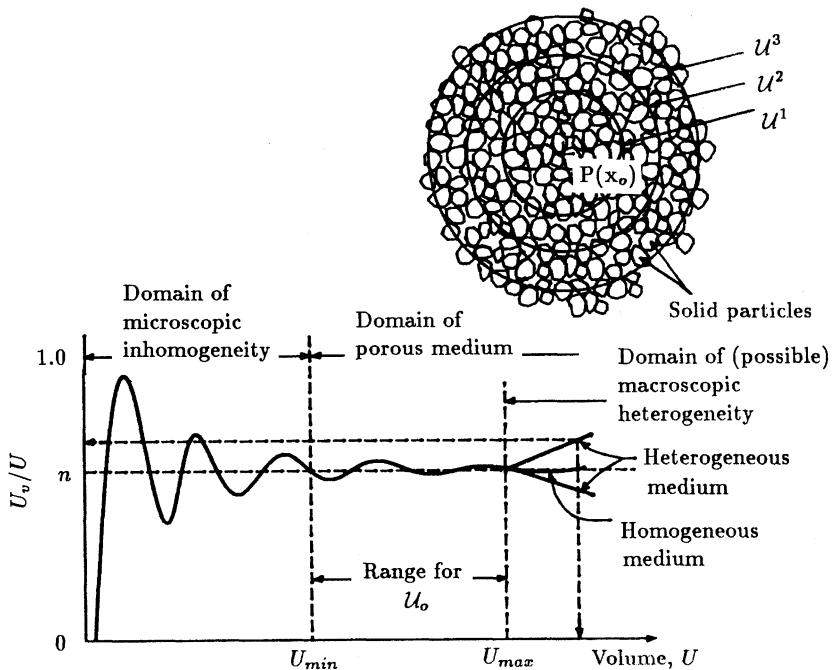


Figure 3.2: Representative average volume (REV) (Bear & Bachmat 1990)

where α is the phase indicator, $\epsilon^\alpha = \Omega^\alpha / \Omega_0$ is the volumetric fraction of the α phase, Ω_0 is the averaging volume (corresponding to the representative elementary volume), $f^\alpha = 1/0$ (inside or outside α phase) is the phase distribution function.

Averaging rules

Due to the above definition the following averaging rules can be derived.

- Sum

$$\overline{\psi_1^\alpha + \psi_2^\alpha} = \overline{\psi_1^\alpha} + \overline{\psi_2^\alpha} \quad (3.4)$$

- Product

$$\overline{\psi_1^\alpha \psi_2^\alpha} = \overline{\psi_1^\alpha} \overline{\psi_2^\alpha} + \overline{\psi_1'^\alpha \psi_2'^\alpha} \quad (3.5)$$

- Time derivative

$$\epsilon^\alpha \frac{\partial \overline{\psi^\alpha}}{\partial t} = \frac{\partial \epsilon^\alpha \overline{\psi^\alpha}}{\partial t} - \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \psi^\alpha \mathbf{w} \cdot d\mathbf{S} \quad (3.6)$$

- Spatial derivative

$$\epsilon^\alpha \overline{\nabla \psi^\alpha} = \nabla(\epsilon^\alpha \overline{\psi^\alpha}) + \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \psi^\alpha \cdot d\mathbf{S} \quad (3.7)$$

where \mathbf{w} is the velocity of the $\alpha\beta$ -phase interface.

Proof

$$\overline{\nabla \psi^\alpha} = \frac{1}{\epsilon^\alpha \Omega_0} \int_{\Omega_0^\alpha} \nabla \psi^\alpha d\Omega = \frac{1}{\Omega_0} \int_{\partial \Omega_0^\alpha} \psi^\alpha \cdot d\mathbf{S} = \frac{1}{\Omega_0} \left(\int_{S^{\alpha\alpha}} \psi^\alpha \cdot d\mathbf{S} + \int_{S^{\alpha\beta}} \psi^\alpha \cdot d\mathbf{S} \right) \quad (3.8)$$

We argue vice versa, beginning with the expected result $\nabla \overline{\psi^\alpha}$.

$$\begin{aligned} \frac{\partial}{\partial x_i} \int_{\Omega^\alpha} \psi^\alpha d\Omega &= \lim_{\Delta x_i \rightarrow 0} \frac{1}{\Delta x_i} \left(\int_{\Omega^\alpha(x+\Delta x_i)} \psi^\alpha d\Omega - \int_{\Omega^\alpha(x)} \psi^\alpha d\Omega \right) \\ &= \lim_{\Delta x_i \rightarrow 0} \frac{1}{\Delta x_i} \left(\int_{\Omega_2^\alpha + \Omega_3^\alpha} \psi^\alpha d\Omega - \int_{\Omega_1^\alpha + \Omega_2^\alpha} \psi^\alpha d\Omega \right) \\ &= \lim_{\Delta x_i \rightarrow 0} \frac{1}{\Delta x_i} \left(\int_{\Omega_3^\alpha} \psi^\alpha d\Omega - \int_{\Omega_1^\alpha} \psi^\alpha d\Omega \right) \\ &= \lim_{\Delta x_i \rightarrow 0} \frac{1}{\Delta x_i} \left(\int_{\Omega_3^\alpha} \psi^\alpha \Delta x_i dS + \int_{\Omega_1^\alpha} \psi^\alpha \Delta x_i dS \right) \\ &= \int_{BDA} \psi^\alpha dS + \int_{ACB} \psi^\alpha dS \\ &= \int_{\partial \Omega^\alpha} \psi^\alpha dS = \int_{S^{\alpha\alpha}} \psi^\alpha dS \end{aligned} \quad (3.9)$$

General macroscopic balance equation

To derive a phase related macroscopic balance equation, we have to average the balance equation in differential form for a certain phase (1.25). By use of the above averaging operators and rules the following general macroscopic balance

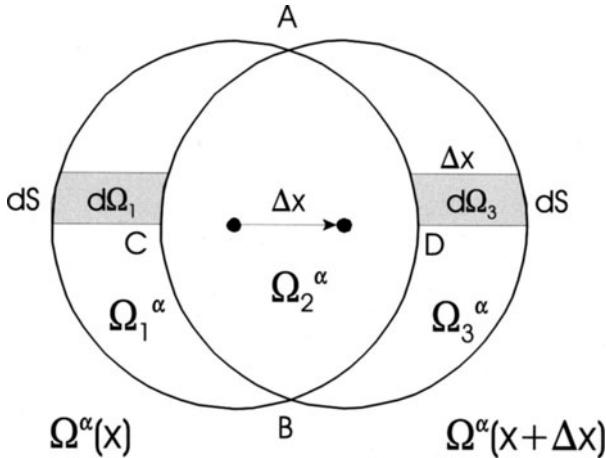


Figure 3.3: Proof of general macroscopic balance equation

equation can be obtained.

$$\frac{\partial \epsilon^\alpha \bar{\psi}^\alpha}{\partial t} = -\nabla \cdot (\epsilon^\alpha \bar{\psi}^\alpha \bar{\mathbf{v}}^\alpha + \epsilon^\alpha \bar{\psi}'^\alpha \bar{\mathbf{v}}'^\alpha + \epsilon^\alpha \bar{\Phi}_{\text{Diff}}^{\psi^\alpha}) - \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \bar{\Phi}_{\text{Diff}}^{\psi^\alpha} \cdot d\mathbf{S} - \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \psi^\alpha (\mathbf{v} - \mathbf{w}) \cdot d\mathbf{S} + q^{\psi^\alpha} \quad (3.10)$$

with the dispersive flux

$$\bar{\Phi}_{\text{Disp}}^{\psi^\alpha} = \epsilon^\alpha \bar{\psi}'^\alpha \bar{\mathbf{v}}'^\alpha \quad (3.11)$$

3.3 Isothermal Consolidation of Porous Media

The simplest case of fluid-solid interaction in porous media is isothermal consolidation in small strain situations. Consolidation models describe slow deformation of solid phase accompanied by flow of pore fluids (single or multiphase flows). In general it is assumed that inertial forces are negligible. In this section we give a brief overview of macroscopic balance equations of mass and momentum conservation in multiphase porous media. To keep mathematical formulations shortly, vector notation is used throughout this part.

3.3.1 Mass Conservation

The Boussinesq approximation is used, where density fluctuations are neglected $\rho'^\alpha = 0$. In the following we omit overlining for mean values to simplify the notation.

Mass conservation of each phase

For mass conservation of each α -phase we have

$$\frac{\partial \epsilon^\alpha \rho^\alpha}{\partial t} + \nabla \cdot (\epsilon^\alpha \rho^\alpha \mathbf{v}^\alpha) = 0 \quad , \quad \alpha = w, s \quad (3.12)$$

with n the porosity of a water-saturated porous medium

$$\epsilon^w = \frac{\Omega^w}{\Omega_0} = \phi \quad , \quad \epsilon^s = \frac{\Omega^s}{\Omega_0} = 1 - \epsilon^w = 1 - \phi \quad (3.13)$$

Mass conservation of the solid phase

For solid phase we can write

$$\frac{\partial (1-n)\rho^s}{\partial t} + \nabla \cdot [(1-n)\rho^s \mathbf{v}^s] = 0 \quad (3.14)$$

Partial differentiation gives

$$-\rho^s \frac{d^s \phi}{dt} + (1-\phi) \left(\frac{d^s \rho^s}{dt} + \rho^s \nabla \cdot \mathbf{v}^s \right) = 0 \quad (3.15)$$

where

$$\frac{d^\alpha}{dt} = \frac{\partial}{\partial t} + \mathbf{v}^\alpha \cdot \nabla \quad (3.16)$$

is the material or total derivative with respect to α -phase. We obtain the equation for porosity change

$$\frac{d^s \phi}{dt} = \frac{1-\phi}{\rho^s} \frac{d^s \rho^s}{dt} + (1-\phi) \nabla \cdot \mathbf{v}^s \quad (3.17)$$

Frequently it is assumed that grains are incompressible.

$$\frac{d^s \rho^s}{dt} = 0 \quad (3.18)$$

Mass conservation of the fluid phase

Mass conservation of the fluid phase gives

$$\frac{\partial \phi \rho^w}{\partial t} + \nabla \cdot (\phi \rho^w \mathbf{v}^w) = 0 \quad (3.19)$$

with specific discharge

$$\mathbf{q} = \phi(\mathbf{v}^w - \mathbf{v}^s) \quad (3.20)$$

Therefore we have

$$\frac{\partial \phi \rho^w}{\partial t} + \nabla \cdot (\rho^w \mathbf{q} + \phi \rho^w \mathbf{v}^s) = 0 \quad (3.21)$$

Partial differentiation gives the following expression

$$\phi \frac{d^s \rho^w}{dt} + \underbrace{\rho^w \frac{d^s \phi}{dt}}_{\text{see 3.17}} + \nabla \cdot (\rho^w \mathbf{q}) + \phi \rho^w \nabla \cdot \mathbf{v}^s = 0 \quad (3.22)$$

Mass conservation within a porous medium

The continuity equation for the water-saturated porous medium can be obtained by combination of phase equations (thermodynamic equilibrium, $p^w = p^s$). We use the expression for porosity changes in the fluid phase conservation equation.

$$\begin{aligned} \phi \frac{d^s \rho^w}{dt} + \rho^w (1 - \phi) \nabla \cdot \mathbf{v}^s + \nabla \cdot (\rho^w \mathbf{q}) + \phi \rho^w \nabla \cdot \mathbf{v}^s &= \\ \phi \frac{d^s \rho^w}{dt} + \rho^w \nabla \cdot \mathbf{v}^s + \nabla \cdot (\rho^w \mathbf{q}) &= 0 \end{aligned} \quad (3.23)$$

Constitutive relationships

Primary variables are fluid pressure p^w and solid strain ε^s .

- Dilatation of solid phase (definition of ε)

$$\frac{d^s \varepsilon^s}{dt} = \frac{\partial \varepsilon^s}{\partial t} + \mathbf{v}^s \cdot \nabla \varepsilon^s = \nabla \cdot \mathbf{v}^s \quad (3.24)$$

$$\Rightarrow \phi \frac{d^s \rho^w}{dt} + \rho^w \frac{d^s \varepsilon^s}{dt} + \nabla \cdot (\rho^w \mathbf{q}) = 0 \quad (3.25)$$

- Fluid density

$$\frac{1}{\rho^\alpha} \frac{d \rho^\alpha}{dt} = \beta_p^\alpha \frac{dp^\alpha}{dt} \quad , \quad \alpha = w \quad (3.26)$$

$$\Rightarrow \phi \rho^w \beta_p^w \frac{d^s p^w}{dt} + \rho^w \frac{d^s \varepsilon^s}{dt} + \nabla \cdot (\rho^w \mathbf{q}) = 0 \quad (3.27)$$

- Assumptions

1. Solid grains are incompressible $d^s \rho^s / dt = 0$
2. Local density variations are neglected $\nabla \rho^w$
3. Local pressure and dilatation variations are small compared to temporal ones

The above assumptions result in the following mass balance equation for a fluid-saturated porous medium.

$$\boxed{\phi \beta_p^w \frac{\partial p^w}{\partial t} + \frac{\partial \varepsilon^s}{\partial t} + \nabla \cdot \mathbf{q} = 0} \quad (3.28)$$

Therefore, we have five unknown functions to determine in 3-D case: $p^w, \varepsilon_1, \varepsilon_2, \varepsilon_3, n$.

3.3.2 Momentum conservation

Phase momentum conservation

The linear momentum of a phase is defined by

$$\overline{\psi^\alpha} = \overline{\rho^\alpha \mathbf{v}^\alpha} = \overline{\rho^\alpha} \overline{\mathbf{v}^\alpha} + \underbrace{\overline{\rho'^\alpha \mathbf{v}'^\alpha}}_{=0} \quad , \quad \alpha = w, s \quad (3.29)$$

Again the Boussinesq approximation is employed. In the following we omit overlining for mean quantities to simplify the notation. The macroscopic momentum balance equation for a specified phase is given by (divergence form)

$$\begin{aligned} \frac{\partial \epsilon^\alpha \rho^\alpha \mathbf{v}^\alpha}{\partial t} &= \epsilon^\alpha \rho^\alpha \mathbf{f}^\alpha - \nabla \cdot [\epsilon^\alpha (\rho^\alpha \mathbf{v}^\alpha \mathbf{v}^\alpha + \rho^\alpha \overline{\mathbf{v}'^\alpha \mathbf{v}'^\alpha} - \sigma^\alpha)] \\ &- \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \rho \mathbf{v} (\mathbf{v} - \mathbf{w}) \cdot d\mathbf{S} + \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \sigma \cdot d\mathbf{S} \end{aligned} \quad (3.30)$$

In the following we consider Newtonian fluids and we assume no exchange between phases.

Fluid phase momentum conservation

The following assumptions are employed:

- Inertial forces: can be neglected $d\mathbf{v}/dt \approx 0$
- Body forces: gravity $\mathbf{f}^e = \mathbf{g}$
- Pressure forces: $\epsilon^\alpha \overline{\nabla p^\alpha} = \nabla(\epsilon^\alpha \overline{p^\alpha}) + \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} p \mathbf{I} \cdot d\mathbf{S}$

- Viscous forces: internal fluid friction is small in comparison to friction on the fluid-solid interface $|\nabla \cdot (\epsilon^\alpha \tau^\alpha)| << |1/\Omega_0 \int_{S^{\alpha\beta}} \tau \cdot d\mathbf{S}|$

$$\epsilon^\alpha \nabla \cdot \tau^\alpha \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \tau \cdot d\mathbf{S} \quad (3.31)$$

- Turbulence: Reynolds stresses can be neglected

Darcy law

Using the above assumptions we derive the following fluid phase momentum equation

$$\phi \rho^w \mathbf{g} - \nabla(\phi p^w) + \frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \tau \cdot d\mathbf{S} = 0 \quad (3.32)$$

Using the constitutive relationship for fluid friction on pore surfaces

$$\frac{1}{\Omega_0} \int_{S^{\alpha\beta}} \tau \cdot d\mathbf{S} = -\frac{\mu}{k} \phi \mathbf{q} \quad (3.33)$$

we finally derive the Darcy law for fluid flow in a porous medium.

$$\mathbf{q} = -\frac{k}{\mu} (\nabla p^w - \rho^w \mathbf{g})$$

(3.34)

A mathematical description of fluid flow in a consolidating porous media can be obtained by combining the mass conservation equations for fluid and solid with Darcy's law.

Solid phase momentum conservation

The following assumptions are employed:

- Inertial forces: can be neglected $d\mathbf{v}/dt \approx 0$
- Body forces: gravity $g\rho$
- Stress symmetry in 3-D: $\boldsymbol{\sigma} = \{\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \tau_{xy}, \tau_{xz}, \tau_{yz}\}$
- Strain symmetry in 3-D: $\boldsymbol{\epsilon}^{in} = \{\epsilon_{xx}, \epsilon_{yy}, \epsilon_{zz}, \gamma_{xy}, \gamma_{xz}, \gamma_{yz}\}$

The equilibrium condition for the solid phase is then

$$(1 - \phi)\rho^s \mathbf{g} + \nabla \cdot [(1 - \phi)\boldsymbol{\sigma}^s] = 0 \quad (3.35)$$

Total stress and effective stress are defined as follows

$$\begin{aligned} \boldsymbol{\sigma} &= (1 - \phi)\boldsymbol{\sigma}^s + \phi\boldsymbol{\sigma}^w = (1 - \phi)\boldsymbol{\sigma}^s - \phi p^w \mathbf{1} \\ &= (1 - \phi)(\boldsymbol{\sigma}^s + p^w \mathbf{1}) - (1 - \phi)p^w \mathbf{1} - \phi p^w \mathbf{1} = \underbrace{(1 - \phi)(\boldsymbol{\sigma}^s + p^w \mathbf{1})}_{\text{Effective stress}} - p^w \mathbf{1} \\ &= \boldsymbol{\sigma}' - p^w \mathbf{1} \end{aligned} \quad (3.36)$$

Effective stress is the difference between total stress and pore fluid pressure related to the volume fraction of the solid phase.

Equilibrium equation

We consider fluid at rest. Summing the momentum balance equations over all phases we obtain the equilibrium condition (due to absence of inertial forces).

$$\begin{aligned} \phi\rho^w \mathbf{g} &- \nabla \cdot (\phi\boldsymbol{\sigma}^w + (1 - \phi)\rho^s \mathbf{g} - \nabla \cdot [(1 - \phi)\boldsymbol{\sigma}^s]) \\ &= [\phi\rho^w + (1 - \phi)\rho^s] \mathbf{g} - \nabla \cdot [\phi\boldsymbol{\sigma}^w + (1 - \phi)\boldsymbol{\sigma}^s] \\ &= \rho \mathbf{g} + \nabla \cdot \boldsymbol{\sigma} \\ \implies &= \boxed{\rho \mathbf{g} + \nabla \cdot \boldsymbol{\sigma}' - \nabla p^w = 0} \end{aligned} \quad (3.37)$$

Example

As an example we consider consolidation of soil due to loading forces (e.g. building structure). This example serves as benchmarks for testing numerical consolidation models (Lewis & Schrefler 1998). The following figures show displacements, σ_{xx} stresses, pore water pressure as well as velocities.

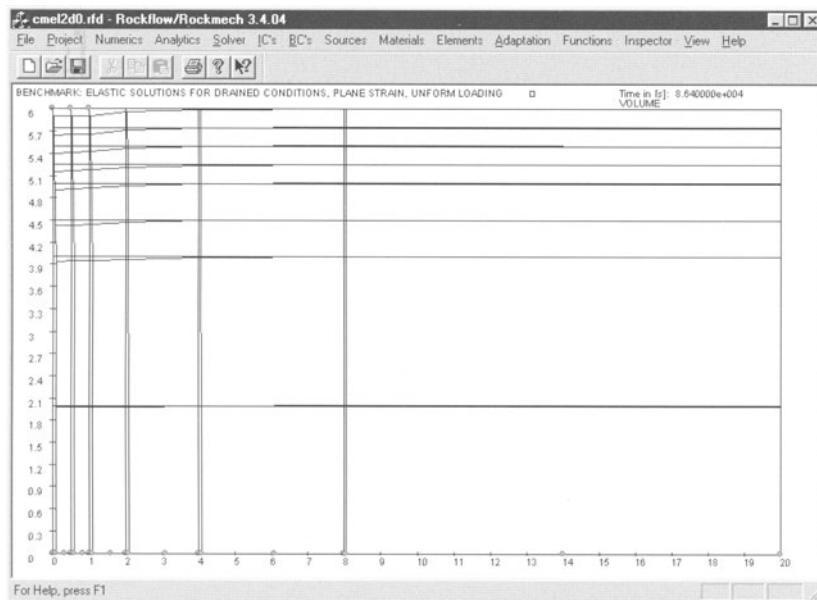


Figure 3.4: Consolidation example: Vertical displacements \mathbf{u}_y

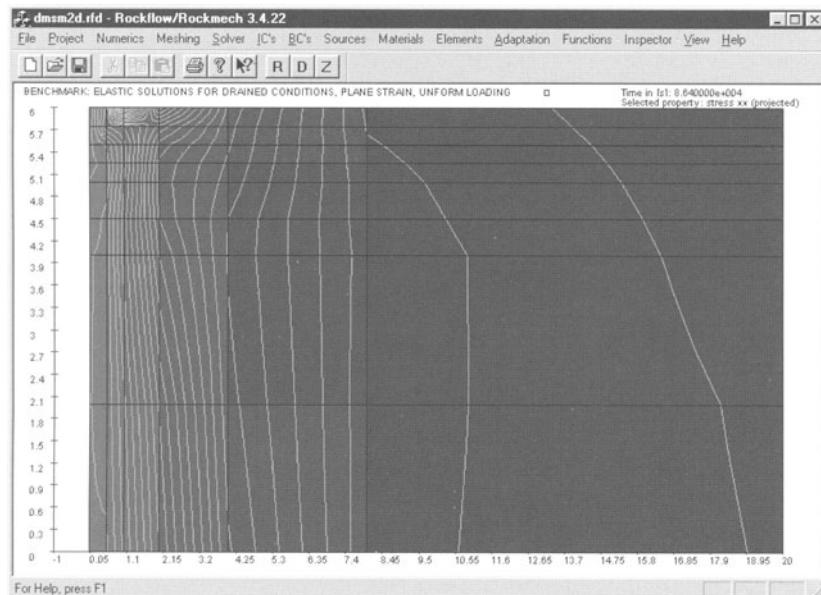


Figure 3.5: Consolidation example: Stresses σ_{xx}

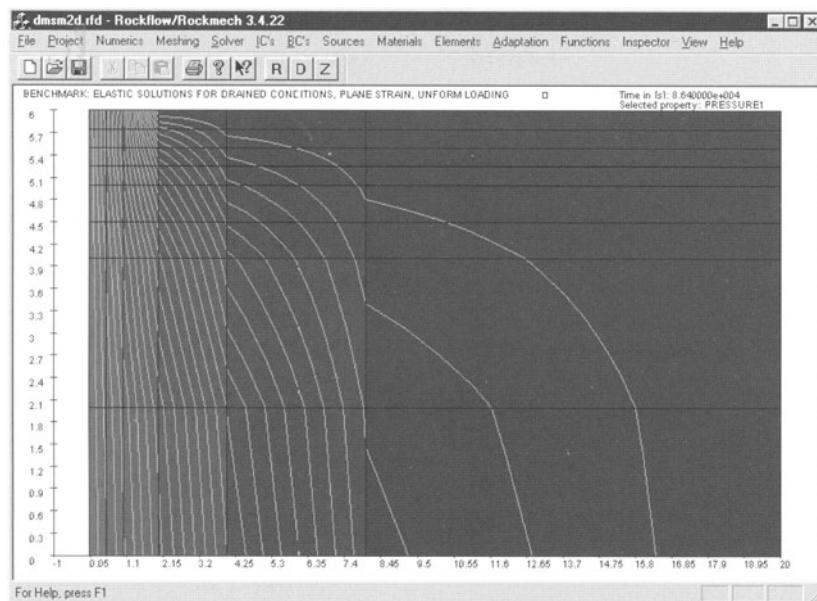


Figure 3.6: Consolidation example: Pore water pressure p^w

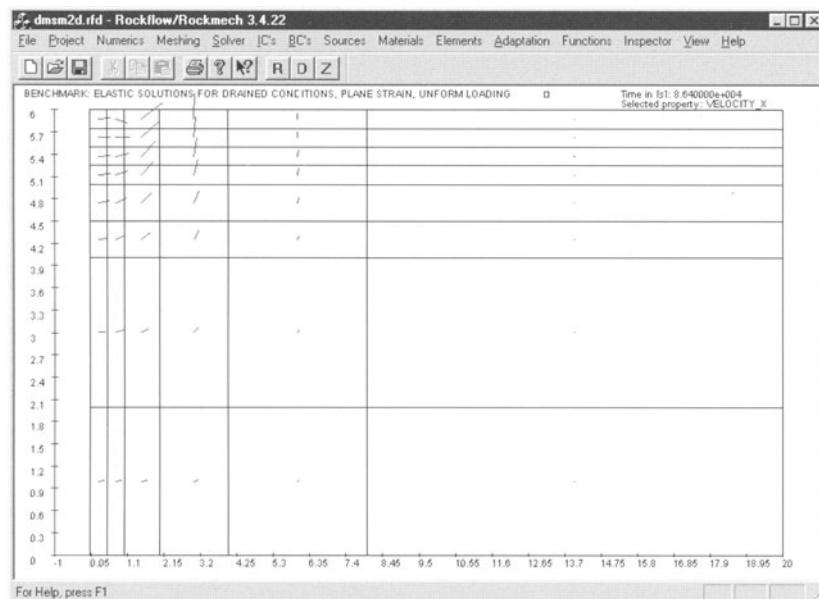


Figure 3.7: Consolidation example: Pore water velocities v^w

3.4 Mass Transport in Porous Media

To simulate transport of mass we must have an appropriate representation of the velocity field. The pressure field and the resulting velocity distribution is available from the groundwater flow simulation described in section 3.3.

A number of assumptions for solute transport are invoked:

- transport of a single component,
- considered tracer are non-reacting chemicals
- negligible variation of gas density due to variation of mass concentration of the tracer,
- single phase flow
- no transfer of the considered species from other phases into the gas phase via phase change,
- no transfer of the considered species from other phases into the gas phase by diffusion across a phase boundary,
- non-deformable porous medium.

Balance equation of the mass of the solute component is

$$\frac{\partial}{\partial t}(n\rho_k) + \frac{\partial}{\partial x_\alpha}(q_\alpha \rho_k) + \frac{\partial j_\alpha}{\partial x_\alpha} = \frac{\partial}{\partial t}(n\rho_k) + \nabla \cdot (\mathbf{q}\rho_k + \mathbf{j}) = n\rho Q_{\rho_k} \quad \alpha, \beta = 1, 2, 3 \quad (3.38)$$

with the Darcy velocity of solute components q . Mass concentration (per unit volume of fluid phase) and mass fraction of the solute component are defined as follows.

$$\rho_k = C \quad , \quad \omega := \omega_k = \frac{C}{\rho_k} \quad (3.39)$$

Dependent on which formulation for dispersive fluxes is chosen a corresponding equation of solute mass conservation can be derived, written either in terms of mass concentrations

$$\frac{\partial}{\partial t}(nC) + \frac{\partial}{\partial x_\alpha}(q_\alpha C) - \frac{\partial}{\partial x_\alpha}(nD_{\alpha\beta} \frac{\partial C}{\partial x_\beta}) = nQ_C \quad (3.40)$$

or in terms of mass fractions

$$\frac{\partial}{\partial t}(n\rho\omega) + \frac{\partial}{\partial x_\alpha}(q_\alpha \rho\omega) - \frac{\partial}{\partial x_\alpha}(n\rho D_{\alpha\beta} \frac{\partial \omega}{\partial x_\beta}) = n\rho Q_\omega \quad (3.41)$$

Different Formulations of the Mass Balance Equation

By use of the continuity equation, the divergent form of the equation of solute mass conservation can be transformed to a convective formulation

$$n\rho \frac{\partial \omega}{\partial t} + \rho q_\alpha \frac{\partial \omega}{\partial x_\alpha} - \frac{\partial}{\partial x_\alpha} (n\rho D_{\alpha\beta} \frac{\partial \omega}{\partial x_\beta}) - \omega n \rho Q_\rho = n\rho Q_\omega \quad (3.42)$$

Developing the equation of solute mass conservation in terms of mass concentration we obtain the following relation

$$n \frac{\partial C}{\partial t} + C \frac{\partial n}{\partial t} + q_\alpha \frac{\partial C}{\partial x_\alpha} + C \frac{\partial q_\alpha}{\partial x_\alpha} - \frac{\partial}{\partial x_\alpha} (n D_{\alpha\beta} \frac{\partial C}{\partial x_\beta}) = n Q_C \quad (3.43)$$

For saturated groundwater flow we assume that (non-deformable porous medium, potential flow),

$$\frac{\partial n}{\partial t} + \frac{\partial q_\alpha}{\partial x_\alpha} = 0 \quad (3.44)$$

therefore, the balance equation of the solute mass conservation can be written in a simplified form

$$n \frac{\partial C}{\partial t} + q_\alpha \frac{\partial C}{\partial x_\alpha} - \frac{\partial}{\partial x_\alpha} (n D_{\alpha\beta} \frac{\partial C}{\partial x_\beta}) = n Q_C \quad (3.45)$$

In multiphase flow, pore space is filled by several non-miscible fluids and we use the following expression for effective porosity $n = nS$, where S is the water saturation and n is total porosity. The mass conservation equation for multiphase system is therefore

$$nS \frac{\partial C}{\partial t} + q_\alpha \frac{\partial C}{\partial x_\alpha} - \frac{\partial}{\partial x_\alpha} (n S D_{\alpha\beta} \frac{\partial C}{\partial x_\beta}) = n Q_C \quad (3.46)$$

where, additionally, saturation variations have to be considered.

Constitutive Relationships

Hydrodynamic dispersion in porous media

We use the classic dispersion model by Scheidegger (1961). The coefficient of dispersion (including molecular diffusion) is expressed as

$$n D_{\alpha\beta} = n \tau D_m + \alpha_T |q| \delta_{\alpha\beta} + (\alpha_L - \alpha_T) \frac{q_\alpha q_\beta}{|q|} \quad (3.47)$$

Boundary Conditions

1. Dirichlet (first-type) boundary condition

$$\omega(t, \mathbf{x}) = \omega|_{B1} \quad (3.48)$$

$$C(t, \mathbf{x}) = C|_{B1} \quad (3.49)$$

2. Neumann (second-type) boundary condition

$$n\rho D_{\alpha\beta} \frac{\partial \omega(t, \mathbf{x})}{\partial x_\beta} n_\alpha = n\rho Q_\omega|_{B2} \quad (3.50)$$

$$nD_{\alpha\beta} \frac{\partial C(t, \mathbf{x})}{\partial x_\beta} n_\alpha = nQ_C|_{B2} \quad (3.51)$$

3.5 Heat Transport in Porous Media

In this chapter we consider heat transport in porous and fractured media. To simulate transfer of heat, an appropriate representation of macroscopic velocities must be available which may be obtained from the pressure field (section 3.3).

The governing equation for heat transport is derived from the macroscopic energy balance (see e.g. Kolditz 1997 pp 17ff, 215ff). We assume for the liquid phase that the change of internal energy may be expressed as a function of temperature and pressure. Furthermore, we assume locally thermal equilibrium between the phases, i.e. temperature of solid and fluid are identical. The well known Fourier's law is used to describe heat flux. Moreover, the following physical effects are neglected: macroscopic viscous dissipation effects, i.e. the term containing the stress tensor, divergence of velocity in the energy balance, and grain velocity. We consider the case of water-saturated soil or rock. The tensor of hydrodynamic thermal dispersion for a porous medium is written in an analogous manner to this for solute by Scheidegger (1961):

$$E_{\alpha\beta} = \beta_T |q| \delta_{\alpha\beta} + (\beta_L - \beta_T) \frac{q_\alpha q_\beta}{|q|} \quad (3.52)$$

Employing the assumptions and simplifications above, the heat transport equation becomes

$$c\rho \frac{\partial T}{\partial t} + c^w \rho^w q_\alpha \frac{\partial T}{\partial x_\alpha} - \frac{\partial}{\partial x_\alpha} (D_{\alpha\beta}^T \frac{\partial T}{\partial x_\beta}) = \rho Q_T \quad (3.53)$$

with:

heat capacity of the porous medium in $[Jm^{-3}K^{-1}]$

$$c\rho = nc^w \rho^w + (1-n)c^s \rho^s \quad (3.54)$$

heat diffusion-dispersion tensor of the porous medium in [$Jm^{-1}s^{-1}K^{-1}$]

$$D_{\alpha\beta}^T = n\lambda^w \delta_{\alpha\beta} + (1-n)\lambda_{\alpha\beta}^s + c^w \rho^w E_{\alpha\beta} \quad (3.55)$$

heat source term of the porous medium in [$Jm^{-3}K^{-1}$]

$$\rho Q_T = n\rho^w Q_T^w + (1-n)\rho^s Q_T^s \quad (3.56)$$

where $Q_T[Jkg^{-1}s^{-1}]$ is the heat production rate per unit mass of porous medium.

Chapter 4

Problem Classification

The governing equations for fluid flow and related transport processes are partial differential equations (PDE) containing first and second order derivatives in spatial coordinates and first order derivatives in time. Whereas time derivatives appear linearly, spatial derivatives often have a non-linear form. Frequently, systems of partial differential equations occur rather than a single equation.

Table 4.1: Mathematical classification of PDE's

PDE type	Discriminant	Eigenvalues	Canonical form	Example
Elliptic 4.3	$B^2 - 4AC < 0$ complex characteristics	$\forall \lambda > 0$ equal signs	$\frac{\partial^2 \psi}{\partial \xi^2} + \frac{\partial^2 \psi}{\partial \eta^2} = 0$	Laplace equation
Parabolic 4.4	$B^2 - 4AC = 0$	$\exists \lambda = 0$	$\frac{\partial^2 \psi}{\partial \eta^2} = G$	Diffusion, Burgers equations
Hyperbolic 4.5	$B^2 - 4AC > 0$ real characteristics	$\exists \lambda < 0$ different signs	$\frac{\partial^2 \psi}{\partial \xi^2} - \frac{\partial^2 \psi}{\partial \eta^2} = 0$	Wave equation

4.1 Mathematical Classification

The mathematical classification of PDEs should be tempered by a knowledge of the physical processes which are governed by the specific type of the PDE. Sev-

eral procedures are available for classifying partial differential equations, such as algebraic methods (discriminant and eigenvalue evaluation), characteristics as well as Fourier analysis (Tab. 4.1).

A common formulation of a PDE in \mathcal{R}^3 is

$$L(\psi) = F(t, x_i, \psi, \frac{\partial\psi}{\partial x_i}, \dots, \frac{\partial^n\psi}{\partial x_i^n}) = 0 \quad , \quad i = 3 \quad (4.1)$$

where L is a differential operator. Second-order PDE with two independent variables are given by

$$A \frac{\partial^2\psi}{\partial x^2} + B \frac{\partial^2\psi}{\partial x \partial y} + C \frac{\partial^2\psi}{\partial y^2} + D \frac{\partial\psi}{\partial x} + E \frac{\partial\psi}{\partial y} + F\psi + G = 0 \quad (4.2)$$

It is apparent that the classification depends only on the highest-order derivatives in each independent variable. Second-order PDEs with more independent variables can be classified by examination of the eigenvalues of the matrix a_{ij} .

$$\sum_i \sum_j a_{ij} \frac{\partial\psi^2}{\partial x_i \partial x_j} + G = 0 \quad (4.3)$$

4.2 Physical Classification

Now we examine different types of PDEs from a physical point of view and indicate the flow categories for which they occur (Tab. 4.2).

Elliptic PDEs are associated with steady-state problems (equilibrium problems). For example, potential flow is described by an elliptic equation. Parabolic PDEs are usually related to propagation problems with dissipation, e.g. from viscous or heat diffusion effects. Examples are many of the reduced forms of the Navier-Stokes equation, transport equation and Burgers equation. Hyperbolic PDEs are usually connected with propagation problems without dissipation (see section 6.5), e.g. if a wave-like motion remains unattenuated.

Example: Subsonic/supersonic flow

Change of PDE type in different areas of the computational domain may be illustrated by considering 2-D subsonic/supersonic flow. The governing equation for steady, compressible, potential flow can be written as

$$(1 - M) \frac{\partial\psi^2}{\partial x^2} + \frac{\partial\psi^2}{\partial y^2} = 0 \quad (4.4)$$

where ψ is the potential function and M is the Mach number. Therefore, the equation may change its type in different parts of the flow domain depending on the local Mach number. Also for non-linear equations the PDE type can change locally.

Table 4.2: Physical classification of PDEs

Physical problem	Math. problem	Examples
Equilibrium problems	Elliptic equations	Irrational incompressible flow Inviscid incompressible flow Steady state heat conduction
Propagation problems (infinite propagation speed)	Parabolic equations	Unsteady viscous flow Transient heat transfer
Propagation problems (finite propagation speed)	Hyperbolic equations	Wave propagation (vibration) Inviscid supersonic flow

Example: Forms of the Navier-Stokes equation

As shown in section 1.3.6 momentum conservation for incompressible, viscous fluid is given by

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\frac{1}{\rho} \nabla p + \nu \Delta \mathbf{v} \quad (4.5)$$

where external forces are neglected.

If using $(p^* = \rho^*(v^*)^2)$ the non-dimensional form of the Navier-Stokes equation is then

$$\frac{v^* t^*}{L^*} \frac{\partial \mathbf{v}_D}{\partial t_D} + (\mathbf{v}_D \cdot \nabla) \mathbf{v}_D = -\nabla p_D + \frac{1}{Re} \Delta \mathbf{v}_D \quad (4.6)$$

In general, the Navier-Stokes equation has a parabolic-hyperbolic structure for unsteady flow problems and it has an elliptic-hyperbolic structure for steady flow problems. We see the equation type depends strongly on the Reynolds number.

(1) Strongly viscous dominated flow: $Re \ll 1$ - Stokes equation

In this case the advection term can be neglected with respect to the viscous one.

$$\frac{(v^*)^2 t^*}{\nu^*} \frac{\partial \mathbf{v}_D}{\partial t_D} = -Re \nabla p_D + \Delta \mathbf{v}_D \quad (4.7)$$

The above Stokes equation is elliptic in the steady-state case and parabolic in the transient case. Dissipation is isotropic, it works in all directions.

(2) Inviscid flow - Euler equation

In the case of large Reynolds numbers, viscous terms are small.

$$\frac{v^* t^*}{L^*} \frac{\partial \mathbf{v}_D}{\partial t_D} + (\mathbf{v}_D \cdot \nabla) \mathbf{v}_D = -\nabla p_D \quad (4.8)$$

The above Euler equation is hyperbolic in space and time. It describes a propagation problem with negligible dissipation effects. There is an essential direction of propagation.

4.3 Elliptic Equations

Elliptic PDEs are related to equilibrium and steady-state problems, e.g. steady state temperature distribution in a rod of solid material, equilibrium stress of a solid under a given load or irrotational flow of an incompressible fluid as well as potential flow.

Example: Steady state heat conduction (1-D)

A very simple example of an equilibrium problem is steady state heat conduction ($\psi = T$) in an insulated rod whose ends are kept at different constant temperatures. This problem is governed by the following equation

$$\frac{d^2 \psi}{dx^2} = 0 \quad (4.9)$$

The solution will be a linear temperature distribution (Fig. 4.1).

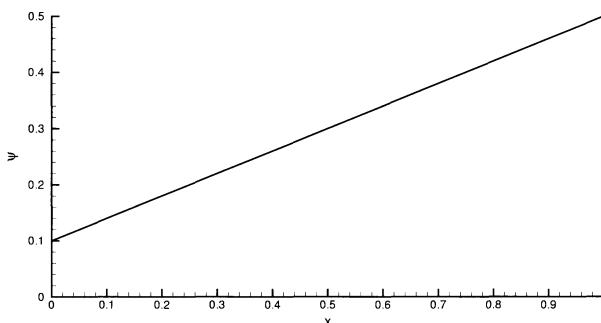


Figure 4.1: Solution of an elliptic equation

Example: Laplace equation (2-D potential flow)

The prototype of an elliptic equation is the Laplace equation.

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = 0 \quad (4.10)$$

By substitution it can be easily verified that the exact solution of the Laplace equation is

$$\psi = \sin(\pi x) \exp(-\pi y) \quad (4.11)$$

A unique solution of the Laplace equation and in general to all elliptic problems can be obtained by specifying values of the dependent variable ψ on all boundaries of the solution domain.

An important feature of elliptic problems is, that local disturbances in the interior changes the values of the dependent variable everywhere else in the solution domain, i.e. the signal propagates in all directions. Consequently, solutions of elliptic problems are always smooth even for discontinuous boundary conditions.

4.4 Parabolic Equations

Parabolic PDEs are usually related to (time-dependent) propagation problems with dissipation processes (e.g. from viscous shear or heat diffusion effects). A prototype of a parabolic equation is the heat conduction equation.

Example: Heat conduction in a rod (1-D problem)

$$\frac{\partial \psi}{\partial t} = \alpha \frac{\partial^2 \psi}{\partial x^2} \quad (4.12)$$

It can be easily shown, that the general solution of the linear diffusion equation can be expressed as

$$\psi(t, x) = \sin(\pi x) \exp(-\pi t^2) \quad (4.13)$$

If the rod cools down after an initially uniform temperature (at $t = 0$), where the ends of the rod are kept at equal constant temperature, the transient temperature distribution is given in Fig. 4.2. It can be seen, that the solution march forward in time but diffuse in space. At the beginning, temperature distributions have parabolic forms. The final, steady state consists of a uniform distribution ($\psi = 0$).

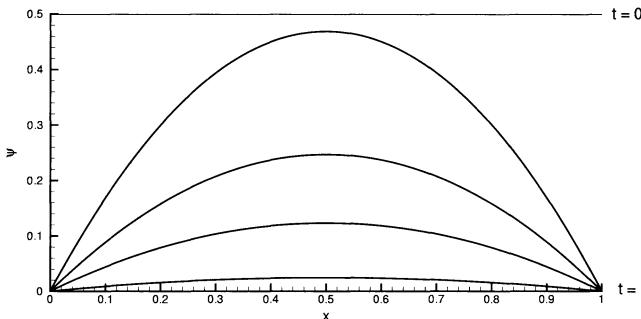


Figure 4.2: Solution of a parabolic equation

For the above parabolic problem appropriate initial as well as boundary conditions are needed. This type of problem is, therefore, termed an initial-boundary-value problem. The parabolic equation converts into an elliptic one for steady state conditions.

Note, the linear diffusion equation has a single characteristic direction dt/dx normal to the diffusion direction.

4.5 Hyperbolic Equations

Examples of problems leading to hyperbolic equations are wave propagation or steady supersonic inviscid flow. Hyperbolic PDEs are usually related to propagation problems with negligible dissipation effects (e.g. a wave-like motion which remains unattenuated).

Example: First-order hyperbolic equation

$$A \frac{\partial \psi}{\partial t} - B \frac{\partial \psi}{\partial x} = 0 \quad (4.14)$$

The characteristic direction is $dx/dt = B/A$. Along the characteristics the PDE becomes ordinary and it is possible to integrate this easily. If A, B, C are constants, the characteristics are straight lines (Fig. 4.3).

$$\frac{d\psi}{dt} = \frac{C}{A} \quad , \quad \frac{d\psi}{dx} = \frac{C}{B} \quad (4.15)$$

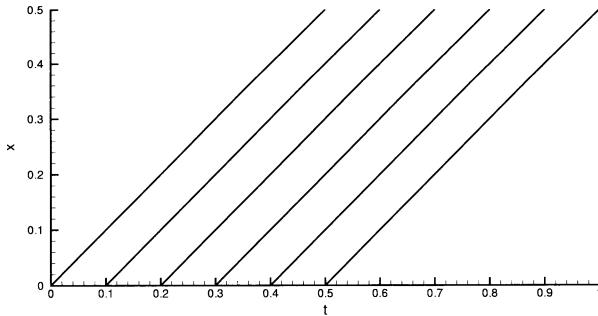


Figure 4.3: Characteristics of a hyperbolic equation, slope B/A

Example: Wave equation

Second-order hyperbolic equations describe e.g. wave propagation.

$$\frac{\partial^2 \psi}{\partial t^2} - c^2 \frac{\partial^2 \psi}{\partial x^2} = 0 \quad (4.16)$$

where c is the wave speed. Introducing the substitutions $\xi = x + ct, \eta = x - ct$ we obtain the canonical form of the wave equation.

$$\frac{\partial^2 \psi}{\partial \xi \partial \eta} = 0 \quad (4.17)$$

It can be easily shown, that the general solution of the wave equation can be expressed as

$$\psi(\xi, \eta) = f(\xi) + g(\eta) \quad (4.18)$$

The wave equation describes for example vibration of a string. The solution for the above wave equation can be given in a Fourier series.

$$\psi(t, x) = a \cos\left(\frac{\pi c t}{L}\right) \sin\left(\frac{\pi x}{L}\right) \quad (4.19)$$

where L is the length of the string. The constant amplitude a shows the absence of any dissipation (damping) process.

Domains of influence and dependence for hyperbolic equations are defined by their characteristics (Fig. 4.4).

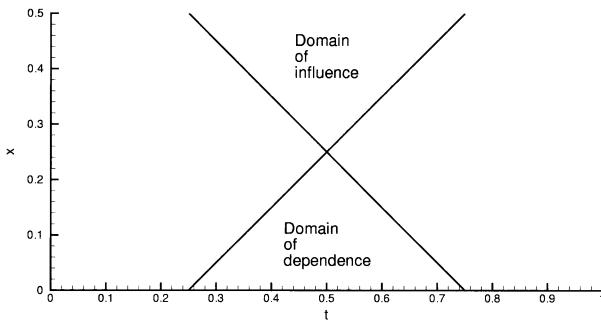


Figure 4.4: Zones of dependence and influence for hyperbolic problems

4.6 Equation Types

The following table gives typical examples of balance equations for the denoted quantities and their PDE types.

Physical meaning	Equation structure	Examples
Continuity	$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = 0$	Laplace equation
Mass/energy	$\frac{\partial \psi}{\partial t} + u \frac{\partial \psi}{\partial x} - \alpha \frac{\partial^2 \psi}{\partial x^2} = 0$	Fokker-Planck equation
Momentum	$\frac{\partial \psi}{\partial t} + \psi \frac{\partial \psi}{\partial x} - \frac{\partial}{\partial x} [\alpha(\psi) \frac{\partial \psi}{\partial x}] = 0$	Navier-Stokes equation

4.7 Boundary Conditions

4.7.1 General Remarks

The solution of an initial-boundary-value-problem (e.g. partial differential equations for flow and transport problems) requires the specification of initial and boundary conditions. In this section we describe the most common boundary conditions and their physical meaning. Methods for implementation of boundary conditions in discrete equations are discussed in the part of numerical methods. Examples of common flow boundary conditions are e.g. inlet, outlet, wall, prescribed pressure values.

Constant pressure boundary condition: The constant pressure boundary condition is used in situations where exact details of the flow distribution are unknown but the boundary values of pressure are known.

No-slip boundary condition at a wall: This boundary condition is most common encountered in confined flow problems. The no-slip condition is the appropriate one for the velocity components at solid walls. The normal velocity component can simply be set to zero at this boundary.

The following table gives an overview on common boundary condition types and its mathematical representation.

Table 4.3: Boundary conditions types

Type of BC	Mathematical Meaning	Physical Meaning
Dirichlet type	ψ	prescribed value potential surface
Neumann type	$\nabla\psi$	prescribed flux stream surface
Cauchy type	$\psi + A\nabla\psi$	resistance between potential and stream surface

To describe conditions at boundaries we can use flux expressions of conservation quantities.

Table 4.4: Fluxes through surface boundaries

Quantity	Flux term
Mass	$\rho \mathbf{v}$
Momentum	$\rho \mathbf{v} \mathbf{v} - \sigma$
Energy	$\rho e \mathbf{v} - \lambda \nabla \mathbf{T}$

4.7.2 Mass Fluxes

Figure 4.5 shows a typical situation for the placement of boundary conditions for fluid flow. First, we have a boundary between the fluid and a solid (e.g. soil for river flow). Second, there is a boundary between fluids, e.g. between water and air. Finally, a finite domain must be defined by cutting off the area of interest (modeling domain). Therefore, we have boundaries which are closed to flow (e.g. water/soil and water/air interfaces) and boundaries which are open for flow (specified domain boundaries).

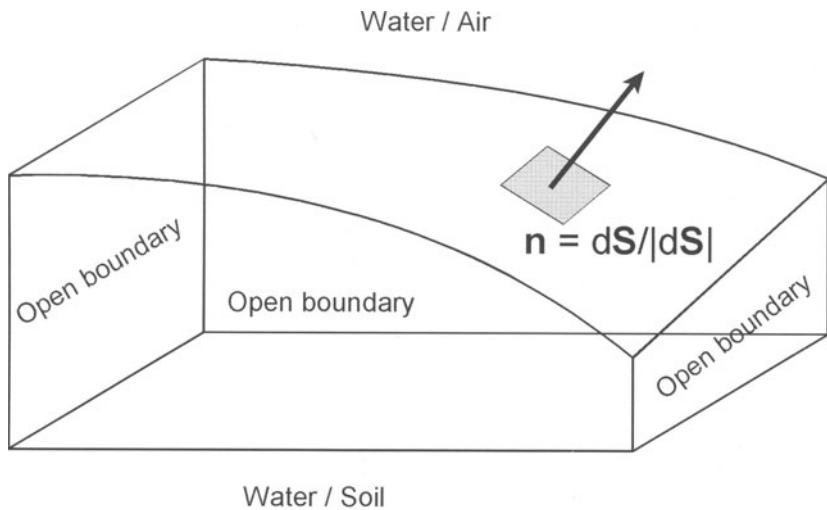


Figure 4.5: Boundary conditions for mass flow

No-flux boundary: At boundaries which are closed to flow, the flux of fluid mass will be equal to zero.

$$\Phi^m \cdot \mathbf{n} = \rho \mathbf{v} \cdot \mathbf{n} = 0 \quad \Rightarrow \quad \mathbf{v} \cdot \mathbf{n} = 0 \quad (4.20)$$

For a curved boundary given by $z(x, y)$ the outward unit surface vector will be

$$\mathbf{n}(x, y) = \frac{d\mathbf{S}}{|d\mathbf{S}|} = \frac{1}{\sqrt{1 + \frac{\partial z}{\partial x}^2 + \frac{\partial z}{\partial y}^2}} \begin{pmatrix} \frac{\partial z}{\partial x} \\ \frac{\partial z}{\partial y} \\ -1 \end{pmatrix} \quad (4.21)$$

Therefore, the no-flux boundary condition can be written as

$$\mathbf{v} \cdot \mathbf{n} = u \frac{\partial z}{\partial x} + v \frac{\partial z}{\partial y} - w = 0 \quad \Rightarrow \quad w = u \frac{\partial z}{\partial x} + v \frac{\partial z}{\partial y}. \quad (4.22)$$

For a time-depending free water surface we can write

$$w = \frac{\partial z}{\partial t} + u \frac{\partial z}{\partial x} + v \frac{\partial z}{\partial y} = \frac{dz}{dt}. \quad (4.23)$$

4.7.3 Momentum Fluxes

The total momentum flux for viscous flow is given by the tensor (see eqn. 1.40).

$$\Phi^m = \rho \mathbf{v} \mathbf{v} - \boldsymbol{\sigma} = \rho \mathbf{v} \mathbf{v} + p \mathbf{I} - \mu \nabla \mathbf{v} \quad (4.24)$$

The momentum flux normal to a surface element $d\mathbf{S}$ can be written as

$$\Phi^m \cdot \mathbf{n} = \rho \mathbf{v} \cdot \mathbf{n} + p \cdot \mathbf{n} - \mu \nabla \mathbf{v} \cdot \mathbf{n} \quad (4.25)$$

where the first term on the left-hand-side vanishes for a closed boundary as $\mathbf{v} = 0$.

Water/Soil Interaction

At a fluid-solid interface we can distinguish three types of boundary conditions (Fig. 4.6).

- A: If there is no friction on the interface we have a slipping condition. From the relation

$$\mu \nabla \mathbf{v} \cdot \mathbf{n} = 0 \quad (4.26)$$

it results that

$$\frac{\partial \mathbf{v}}{\partial \mathbf{n}} = 0 \quad (4.27)$$

- B: In hydraulic practice, finite shear stresses act on the fluid/solid interface. Frequently, these shear stresses are defined by empirical formulas. For shear stresses the tangential components of the viscous stress tensor are of particular interest. For special case that the surface element is in the x-y plane (Fig. 4.7) we have

$$\nu \frac{\partial u}{\partial z} = \frac{u \sqrt{u^2 + v^2}}{C^2} \quad , \quad \nu \frac{\partial v}{\partial z} = \frac{v \sqrt{u^2 + v^2}}{C^2} \quad (4.28)$$

where C is a roughness coefficient.

- C: If the fluid adheres on the solid surface (no-slipping condition), the velocity vector is zero at this interface.

$$\mathbf{v} = 0 \quad (4.29)$$

Water/Air Interaction

At a free surface of a water body (water/air interface) motion can be induced by blowing wind. Air pressure and wind shear stresses are acting on those surfaces. The momentum flux due to shear stresses in the tangential area of the free surface is given by

$$\mu \nabla \mathbf{v} \cdot \mathbf{n} = \frac{\mu}{\sqrt{1 + \frac{\partial z}{\partial x}^2 + \frac{\partial z}{\partial y}^2}} \left(\begin{array}{c} \frac{\partial u}{\partial x} \frac{\partial z}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial z}{\partial y} + \frac{\partial u}{\partial z} \\ \frac{\partial v}{\partial x} \frac{\partial z}{\partial x} + \frac{\partial v}{\partial y} \frac{\partial z}{\partial y} + \frac{\partial v}{\partial z} \\ \frac{\partial w}{\partial x} \frac{\partial z}{\partial x} + \frac{\partial w}{\partial y} \frac{\partial z}{\partial y} + \frac{\partial w}{\partial z} \end{array} \right) \quad (4.30)$$

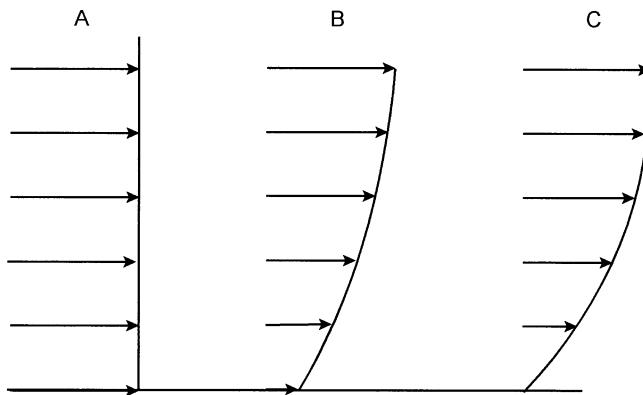


Figure 4.6: Boundary conditions for momentum flow (1-D)

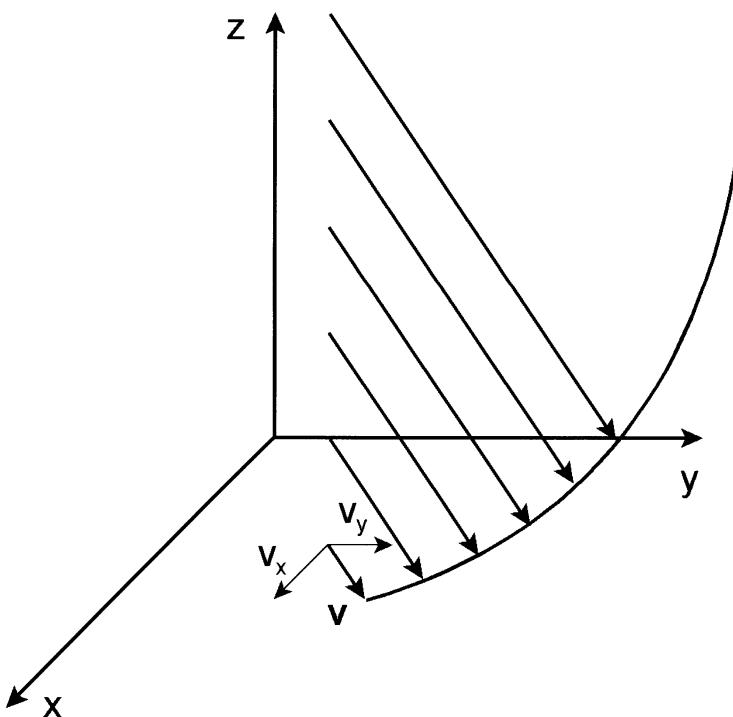


Figure 4.7: Boundary conditions for momentum flow (2-D)

For the case of a horizontal interface, i.e. $z = \text{const}$ we have

$$\mu \nabla \mathbf{v} \cdot \mathbf{n} = \frac{\mu}{\sqrt{1 + \frac{\partial z^2}{\partial x} + \frac{\partial z^2}{\partial y}}} \begin{pmatrix} \frac{\partial u}{\partial z} \\ \frac{\partial v}{\partial z} \\ \frac{\partial w}{\partial z} \end{pmatrix} \quad (4.31)$$

Due to the continuity of momentum flux across the free surface the following relationship holds at the interface

$$\mu^w \frac{\partial \mathbf{v}}{\partial z} |_{\text{Water}} = \mu^a \frac{\partial \mathbf{v}}{\partial z} |_{\text{Air}} = C_D(z) \mathbf{v}^a |\mathbf{v}^a| \quad (4.32)$$

where C_D is the wind shear coefficient. Several values of this parameter can be found in Malcherek (2001).

4.7.4 Energy Transfer

As described in section 1.4 the advective energy flux is given by

$$\Phi^e = \rho(i + \frac{v^2}{2})\mathbf{v} \quad (4.33)$$

Advective heat flux is expressed by

$$\Phi^H = \rho c T \mathbf{v} \quad (4.34)$$

4.8 Problems

PDE Classification

- 1** What are the basic types of PDEs ?
- 2** What mathematical methods exist for classification of PDEs ?
- 3** Prove that the following system of PDEs (Riemann equations) is elliptic.

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad , \quad \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} = 0 \quad (4.35)$$

- 4** Prove that eqn (4.14) and (4.16) are hyperbolic equations using the discriminant method.
- 5** Give examples for PDEs and corresponding physical problems.

Elliptic PDEs

- 6** Prove that eqn (4.11) is a special solution of the Laplace equation (4.10).
- 7** Prove that the expressions $u = x/(x^2 + y^2)$, $v = y/(x^2 + y^2)$ are solutions of eqn (4.35).

Parabolic PDEs

- 8** Prove that eqn (4.13) is a special solution of the diffusion equation (4.12).

Hyperbolic PDEs

- 9** Prove that eqn (4.19) is a special solution of the wave equation (4.16).
- 10** Derive characteristics and characteristic directions of eqn (4.14).

Boundary Conditions

- 11** What are boundary conditions needed for with respect to solving PDEs ?
- 12** What are the basic types of boundary conditions ? Explain their physical meaning.

Bibliography

- [1] Bear J (1972): *Dynamics of fluids in porous media*, American Elsevier.
- [2] Bear J & Bachmat Y (1990): *Introduction to modeling of transport phenomena in porous media*, Kluwer Academic Publishers.
- [3] de Boer R & Ehlers W (1986): *Theorie der Mehrkomponentenkontinua mit Anwendung auf bodenmechanische Probleme*, Forschungsberichte aus dem Fachbereich Bauwesen der Universität-GH-Essen, vol 40, Essen.
- [4] de Boer R (2000): *Theory of porous media*, Springer-Verlag.
- [5] Diersch H-J G (1985): *Modellierung und numerische Simulation geohydrodynamischer Transportprozesse*, Habilitationsschrift, Akademie der Wissenschaften der DDR, Berlin.
- [6] Helmig R (1997): *Multiphase flow and transport in the subsurface*, Springer-Verlag.
- [7] Kinzelbach W (1992): *Numerische Methoden zur Modellierung des Transport von Schadstoffen im Grundwasser*, Oldenbourg-Verlag.
- [8] Kolditz O (1997): *Strömung, Stoff- und Wärmetransport im Kluftgestein*, Gebrüder-Borntraeger Verlag, Berlin-Stuttgart.
- [9] Launder B E & Spalding D B (1974): *The numerical computation of turbulent flows*, Comp. Methods Apl. Mech. Eng., 3:269-289.
- [10] Lewis R W & Schrefler B A (1998): *The finite element method in the static and dynamic deformation and consolidation of porous media*, Wiley & Sons.
- [11] Malcherek A (2001): *Hydrodynamik der Fließgewässer*, Bericht Nr. 61, Institut für Strömungsmechanik und Elektr. Rechnen im Bauwesen, Universität Hannover.
- [12] Scheidegger A E (1961): *General theory of dispersion in porous media*, J. Geophys. Res. (66): 3273-3278.
- [13] Strybny J (2002): *Keine Panik vor Strömungsmechanik - Ein alternatives Lern- und Übungsbuch*, Institut für Strömungsmechanik und Elektr. Rechnen im Bauwesen, Universität Hannover.
- [14] Rodi W (1980): *Turbulence models and their application in hydraulics - A state of the art review*, IAHR, Delft, The Netherlands.
- [15] Truckenbrodt E (1968): *Strömungsmechanik*, Springer-Verlag.
- [16] Versteeg H K & Malalasekera W (1995): *An introduction to computational fluid dynamics: The finite volume method*, Longman Scientific & Technical.
- [17] Zielke W (1998): *Strömungsmechanik für Bauingenieure*, Vorlesungsskript, Institut für Strömungsmechanik und Elektr. Rechnen im Bauwesen, Universität Hannover.

Part II

Numerical Methods

Chapter 5

Numerical Methods

There are many alternative methods to solve initial-boundary-value problems arising from flow and transport processes in subsurface systems. In general these methods can be classified into analytical and numerical ones. Analytical solutions can be obtained for a number of problems involving linear or quasi-linear equations and calculation domains of simple geometry. For non-linear equations or problems with complex geometry or boundary conditions, exact solutions usually do not exist, and approximate solutions must be obtained. For such problems the use of numerical methods is advantageous. In this chapter we use the Finite Difference Method to approximate time derivatives. The Finite Element Method as well as the Finite Volume Method are employed for spatial discretization of the region. The Galerkin weighted residual approach is used to provide a weak formulation of the PDEs. This methodology is more general in application than variational methods. The Galerkin approach works also for problems which cannot be casted in variational form.

Fig. 5.1 shows an overview on approximation methods to solve partial differential equations together with the associated boundary and initial conditions. There are many alternative methods for solving boundary and initial value problems. In general, these method can be classified as discrete (numerical) and analytical ones.

5.1 Solution Procedure

For a specified mechanical problem the governing equations as well as initial and boundary conditions will be known. Numerical methods are used to obtain an approximate solution of the governing equations with the corresponding initial and boundary conditions. The procedure of obtaining the approximate solution consists of two steps that are shown schematically in Fig. 5.2. The first

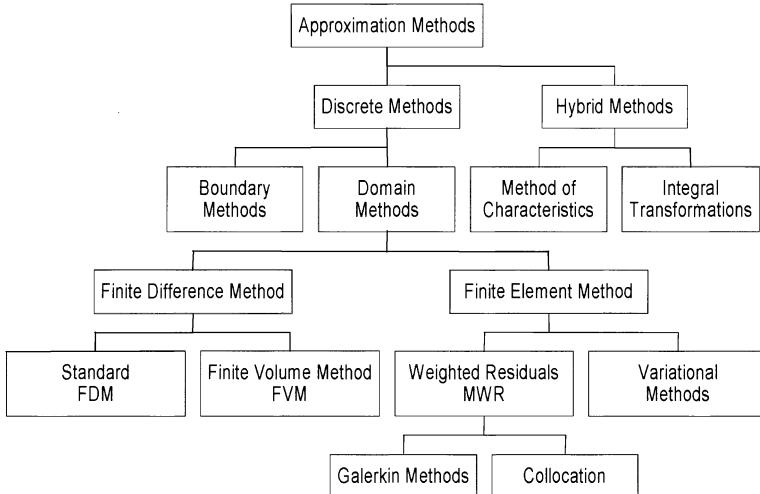


Figure 5.1: Overview of approximation methods and related sections for discussion

step converts the continuous partial differential equations and auxiliary conditions (IC and BC) into a discrete system of algebraic equations. This first step is called discretization. The process of discretization is easily identified if the finite difference method is used but it is slightly less obvious with more complicated methods as the finite element method (FEM), the finite volume method (FVM), and combined Lagrangian-Eulerian methods (method of characteristics, operator split methods).

The replacement of partial differential equations (PDE) by algebraic expressions introduces a defined truncation error. Of course it is of great interest to chose algebraic expressions in a way that only small errors occur to obtain accuracy. Equally important as the error in representing the differentiated terms in the governing equation is the error in the solution. Those errors can be examined as shown in section (5.2).

The second step of the solution procedure, shown in Fig. 5.2, requires the solution of the resulting algebraic equations. This process can also introduce an error but this is usually small compared with those involved in the above mentioned discretization step, unless the solver scheme is unstable. Appropriate methods to solve systems of algebraic equations are discussed in Habbar (1995). The approximate solution of the PDE is exact solution of the corresponding system of algebraic equations (SAE).

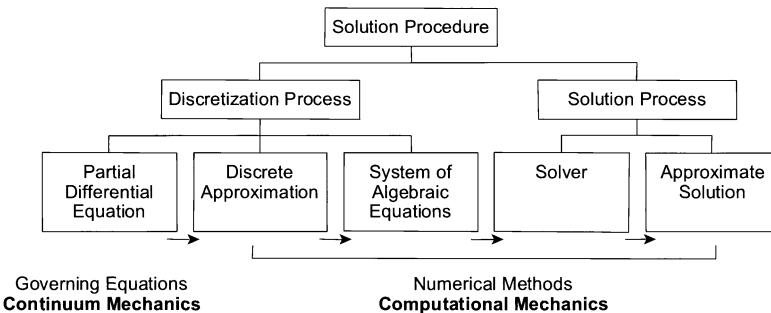


Figure 5.2: Steps of the overall solution procedure

5.2 Theory of Discrete Approximation

5.2.1 Terminology

In the first part of this script we developed the governing equations for fluid flow, heat and mass transfer from basic conservation principles. We have seen that hydromechanical field problems (as well as mechanical equilibrium problems) have to be described by partial differential equations (PDEs). The process of translating the PDEs to systems of algebraic equations is called - discretization (Fig. 5.3). This discretization process is necessary to convert PDEs into an equivalent system of algebraic equations that can be solved using computational methods.

$$L(u) = \tilde{L}(\tilde{u}) = 0 \quad (5.1)$$

In the following, we have to deal with discrete equations \tilde{L} and with discrete solutions \tilde{u} .

An important question concerning the overall solution procedure for discrete methods is what guarantee can be given that the approximate solution will be close to the exact one of the PDE. From truncation error analysis, it is expected that more accurate solutions could be obtained on refined grids. The approximate solution should converge to the exact one as time step sizes and grid spacing shrink to zero. However, convergence is very difficult to obtain directly, so that usually two steps are required to achieve convergence:

$$\text{Consistency} + \text{Stability} = \text{Convergence}$$

This formula is known as the **Lax equivalence axiom**. That means, the system of algebraic equations resulting from the discretization process should be consistent with the corresponding PDE. Consistency guarantees that the PDE

is represented by the algebraic equations. Additionally, the solution process, i.e. solving the system of algebraic equations, must be stable.

Fig. 5.3 presents a graphic to illustrate the relationship between the above introduced basic terms of discrete approximation theory: convergence, stability, truncation, and consistency. These fundamental terms of discrete mathematics are explained further and illustrated by examples in the following.

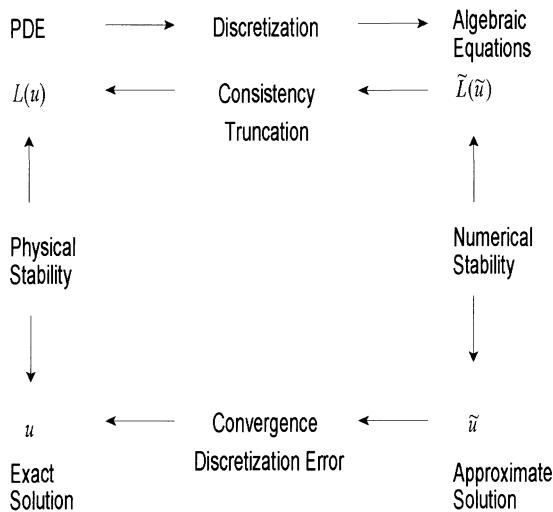


Figure 5.3: Discrete approximation of a PDE and its solution (after Fletcher 1990)

5.2.2 Errors and Accuracy

The following discussion of convergence, consistency, and stability is concerned with the behavior of the approximate solution if discretization sizes ($\Delta t, \Delta x$) tends to zero. In practice, approximate solutions have to be obtained on finite grids which must be calculable on available computers. Therefore, errors and achievable accuracy are of great interest.

If we want to represent continuous systems with the help of discrete methods, of course, we introduce a number of errors. Following types of errors may occur: solution error, discretization error, truncation error, and round-off errors. Round-off errors may result from solving equation systems. Truncation errors

are omitted from finite difference approximations. This means, the representation of differentiated terms by algebraic expressions connecting nodal values on a finite grid introduces a certain error. It is desirable to choose the algebraic terms in a way that only errors as small as possible are invoked. The accuracy of replaced differentiated terms by algebraic ones can be evaluated by considering the so-called truncation error. Truncation error analysis can be conducted by Taylor series expansion (TSE). However, the evaluation of this terms in the TSE relies on the exact solution being known. The truncation error is likely to be a progressively more accurate indicator of the solution error as the discretization grid is refined.

There exist two techniques to evaluate accuracy of numerical schemes. At first, the algorithm can be applied to a related but simplified problem, which possesses an analytical solution (e.g. Burgers' equation which models convective and diffusive momentum transport). The second method is to obtain solutions on progressively refined grids and to proof convergence. In general, accuracy can be improved by use of higher-order schemes or grid refinement.

5.2.3 Convergence

Definition: A solution of the algebraic equations which approximate a given PDE is said to be convergent if the approximate solution approaches the exact solution of the PDE for each value of the independent variable as the grid spacing tends to zero. Thus we require

$$\lim_{\Delta t, \Delta x \rightarrow 0} | u_j^n - u(t_n, x_j) | = 0 \quad (5.2)$$

Or in other words, the approximate solution converges to the exact one as the grid sizes becomes infinitely small. The difference between exact and approximate solution is the solution error, denoted by

$$\varepsilon_j^n = | u_j^n - u(t_n, x_j) | \quad (5.3)$$

The magnitude of the solution error typically depends on grid spacing and approximations to the derivatives in the PDE.

Theoretical proof of convergence is generally difficult, except very simple cases. As an example, proof of convergence for the approximate solution to the diffusion equation is given by Noye (1984, pp. 117-119) who used a FTCS scheme (forward differences in time - central difference in space, see section 6.2.2). For indication of convergence, comparison of approximate solutions on progressively refined grids is used in practice. For PDEs which possesses an analytical solution, like the 1-D advection-diffusion problem, it is possible to test convergence by comparison of numerical solutions on progressively refine grids with the exact solution of the PDE.

5.2.4 Consistency

Definition: The system of algebraic equations (SAE) generated by the discretization process is said to be consistent with the original partial differential equation (PDE) if, in the limit that the grid spacing tends to zero, the SAE is equivalent to the PDE at each grid point. Thus we require

$$\lim_{\Delta t, \Delta x \rightarrow 0} |\tilde{L}(u_j^n) - L(u[t_n, x_j])| = 0 \quad (5.4)$$

Or in other words, the SAE converges to the PDE as the grid size becomes zero. Obviously, consistency is necessary for convergence of the approximate solution. However, consistency is not sufficient to guarantee convergence. Although the SAE might be consistent, it does not follow that the approximate solution converges to the exact one, e.g. for unstable schemes. As an example, solutions of the FTCS algorithm diverge rapidly if the scheme is weighted backwards ($\theta > 0.5$). This example emphasizes that, as indicated by the Lax-Equivalence-Axiom, both consistency and stability are necessary for convergence. Consistency analysis can be conducted by substitution of the exact solution into the algebraic equations resulting from the discretization process. The exact solution is represented as a TSE. Finally, we obtain an equation which consists the original PDE plus a reminder. For consistency the reminder should vanish as the grid size tends to zero. In section 6 we present several examples for consistency analysis.

5.2.5 Stability

Frequently, the matrix method and the von Neumann method are used for stability analysis. In both cases possible growth of the error between approximate and exact solution will be examined. It is generally accepted that the matrix method is less reliable as the von Neumann method. Using the von Neumann method, error at one time level is expanded as a finite Fourier series. For this purpose, initial conditions are represented by a Fourier series. Each mode of the series will grow or decay depending on the discretization. If a particular mode grows without bounds, then an unstable solution exists for this discretization.

5.3 Solution Process

We recall, that the overall solution procedure for PDEs consists of the two major steps: discretization and solution processes (Fig. 5.2). In this section we give a brief introduction to the solution process for equation systems, resulting from discretization methods such as finite difference (FDM), finite element (FEM) and finite volume methods (FVM) (see chapters 6-8). More details on the solution of equation systems can be found e.g. in Meyer (1990), Hackbusch

(1991), Habbar (1995), Schwetlick & Kretschmar (1991), Knabner & Angermann (2000), Wriggers (2001).

Several problems in environmental fluid dynamics lead to non-linear PDEs such as non-linear flow (chapter 12), density-dependent flow (chapter 14), multi-phase flow (chapters 15,16). The resulting algebraic equation system can be written in a general way, indicating the dependency of system matrix \mathbf{A} and right-hand-side vector \mathbf{b} on the solution vector \mathbf{x} . Consequently, it is necessary to employ iterative methods to obtain a solution.

$$\mathbf{A}(\mathbf{x}) \mathbf{x} - \mathbf{b}(\mathbf{x}) = \mathbf{0} \quad (5.5)$$

In the following we consider methods for solving linear equation systems (section 5.3.1) and non-linear equation systems (section 5.3.2). Aspects of the implementation of solvers in an object-oriented way are discussed in chapter 10.

5.3.1 Linear Solver

The linear version of equation (5.5) is given by

$$\mathbf{A} \mathbf{x} - \mathbf{b} = \mathbf{0} \quad (5.6)$$

In general there are two types of methods: direct and iterative algorithms. Direct methods may be advantageous for some non-linear problems. A solution will be produced even for systems with ill-conditioned matrices. On the other hand, direct schemes are very memory consuming. The required memory is in the order of $O(nb^2)$, with n the number of unknowns and b the bandwidth of the system matrix. Therefore, it is always useful to apply algorithms for bandwidth reduction (Cuthill & McKee (1969), Gibbs et al. (1976)). Iterative methods have certain advantages in particular for large systems with sparse matrices. They can be very efficient in combination with non-linear solver.

The following list reveals an overview on existing methods for solving linear algebraic equation systems.

- Direct methods
 - Gaussian elimination
 - Block elimination (to reduce memory requirements for large problems)
 - Cholesky decomposition
 - Frontal solver
- Iterative methods
 - Linear steady methods (Jacobian, Gauss-Seidel, Richardson and block iteration methods)
 - Gradient methods (CG) (also denoted as Krylov subspace methods)

Direct Methods

Application of direct methods to determine the solution of equation (5.6)

$$\mathbf{x} = \mathbf{A}^{-1} \mathbf{b} \quad (5.7)$$

requires an efficient techniques to invert the system matrix.

As a first example we consider the Gaussian elimination technique. If matrix \mathbf{A} is not singular (i.e. $\det \mathbf{A} \neq 0$), can be composed in following way.

$$\mathbf{P} \mathbf{A} = \mathbf{L} \mathbf{U} \quad (5.8)$$

with a permutation matrix \mathbf{P} and the lower \mathbf{L} as well as the upper matrices \mathbf{U} in triangle forms.

$$\mathbf{L} = \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ l_{n1} & \cdots & 1 \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} u_{11} & \cdots & u_{1n} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & u_{nn} \end{bmatrix} \quad (5.9)$$

If $\mathbf{P} = \mathbf{I}$ the matrix \mathbf{A} has a so-called LU-decomposition: $\mathbf{A} = \mathbf{L} \mathbf{U}$. The task reduces then to calculate the lower and upper matrices and invert them. Once \mathbf{L} and \mathbf{U} are determined, the inversion of \mathbf{A} is trivial due to the triangle structure of \mathbf{L} and \mathbf{U} .

Assuming that beside non-singularity and existing LU-decomposition, \mathbf{A} is symmetrical additionally, we have $\mathbf{U} = \mathbf{D} \mathbf{L}^T$ with $\mathbf{D} = \text{diag}(d_i)$. Now we can conduct the following transformations.

$$\mathbf{A} = \mathbf{L} \mathbf{U} = \mathbf{L} \mathbf{D} \mathbf{L}^T = \underbrace{\mathbf{L}}_{\tilde{\mathbf{L}}} \underbrace{\sqrt{\mathbf{D}}}_{\tilde{\mathbf{L}}^T} \underbrace{\sqrt{\mathbf{D}}}_{\tilde{\mathbf{L}}^T} \mathbf{L}^T \quad (5.10)$$

The splitting of \mathbf{D} requires that \mathbf{A} is positive definite thus that $\forall d_i > 0$. The expression

$$\mathbf{A} = \tilde{\mathbf{L}} \tilde{\mathbf{L}}^T \quad (5.11)$$

is denoted as Cholesky decomposition. Therefore, the lower triangle matrices of both the Cholesky and the Gaussian method are connected via

$$\tilde{\mathbf{L}} = \mathbf{L}^T \sqrt{\mathbf{D}} \quad (5.12)$$

Iterative Methods

High resolution FEM leads to large equation systems with sparse system matrices. For this type of problems iterative equation solver are much more efficient than direct solvers. Concerning the application of iterative solver we have to distinguish between symmetrical and non-symmetrical system matrices with

Symmetric Matrices	Non-symmetric Matrices
CG	BiCG
Lanczos	CGStab
Gauss-Seidel, Jacobian, Richards	GMRES
SOR and block-iteration	CGNR

different solution methods. The efficiency of iterative algorithms, i.e. the reduction of iteration numbers, can be improved by the use of pre-conditioning techniques).

The last two rows of solver for symmetric problems belong to the linear steady iteration methods. The algorithms for solving non-symmetrical systems are also denoted as Krylov subspace methods.

Gradient Methods (CG)

As an example of an iterative solver for linear equations systems, we consider the conjugate gradient (CG) method. We assume that the system matrix \mathbf{A} is symmetric and positive definite. The basic idea of CG methods is, that the equation system

$$\mathbf{Ax} = \mathbf{b} \quad (5.13)$$

is equivalent to the task of minimizing the functional

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Ax} - \mathbf{b}^T \mathbf{x} \quad (5.14)$$

i.e. if \mathbf{x} is a solution of equation (5.14) it will be a solution of equation (5.13), too. This can be immediately proofed by differentiating the functional.

$$\nabla f(\mathbf{x}) = \mathbf{Ax} - \mathbf{b} = 0 \quad (5.15)$$

This means, we are solving a conjugate problem, instead of the original equation system. Now an iterative scheme can be constructed to determine the solution of the equations system (5.13), which is at the same time the pivoting element of equation (5.15).

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \quad (5.16)$$

where k is the iteration level, α is a scalar and \mathbf{d} is the search direction. The search direction is selected to be direction of steepest descend, i.e. $\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$, giving the algorithm its name - gradient method. The minimization condition

$$\frac{\partial f(\mathbf{x}_{k+1})}{\partial \alpha} = \nabla f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)^T \mathbf{d}_k = 0 \quad (5.17)$$

yields an equation for determination of α

$$\alpha_k = -\frac{\nabla f(\mathbf{x}_k)^T \mathbf{d}_k}{\mathbf{d}_k^T \mathbf{A} \mathbf{d}_k} \quad (5.18)$$

According to equation (5.15) and having in mind that \mathbf{x}_k is not yet the exact solution, we yield

$$\nabla f(\mathbf{x}_k - \mathbf{x}) = \mathbf{A}(\mathbf{x}_k - \mathbf{x}) = -\mathbf{r}_k \quad (5.19)$$

Therefore, we obtain the following expression for the determination of the scalar parameter.

$$\alpha_k = \frac{\mathbf{r}_k^T \mathbf{d}_k}{\mathbf{d}_k^T \mathbf{A} \mathbf{d}_k} \quad (5.20)$$

The algorithm to find the solution vector \mathbf{x} of the equation system (5.13) can now written as:

- determine the search direction \mathbf{d}_k
- minimize $f(\alpha)$ along direction \mathbf{d}_k to determine α_k
- update the solution vector $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$
- restart the procedure

Convergence of CG methods can be improved, if the search direction is determined from a linear combination

$$\mathbf{p}_k = \mathbf{d}_k + \beta_k \mathbf{p}_{k-1} \quad (5.21)$$

The additional parameter β can be found from the relationship (conjugate vectors)

$$\mathbf{p}_k^T \mathbf{A} \mathbf{p}_{k-1} = (\mathbf{d}_k + \beta_k \mathbf{p}_{k-1})^T \mathbf{A} \mathbf{p}_{k-1} = 0 \quad (5.22)$$

Pre-conditioning Methods

The conditioning number κ of a matrix \mathbf{A} is defined as

$$\kappa = \| \mathbf{A} \| \| \mathbf{A}^{-1} \| \quad (5.23)$$

This characteristic number plays an important role for solving equations systems concerning accuracy and convergence of iterative methods. The smaller κ is the better the iteration scheme proceeds. The convergence is limited by the factor $\sqrt{\kappa} - 1/\sqrt{\kappa} + 1$. The idea of pre-conditioning techniques is the transformation of the equation system in a way that the condition number will be reduced ($\kappa \rightarrow 1$).

Two principle pre-conditioning techniques exists. Pre-conditioning from the left side results in a transformation of the equation system.

$$\mathbf{C}^{-1} \mathbf{A} \mathbf{x} = \mathbf{C}^{-1} \mathbf{b} \quad (5.24)$$

Pre-conditioning from the right side results in a transformation of the solution vector.

$$\mathbf{A} \mathbf{C}^{-1} \mathbf{y} = \mathbf{b} , \quad \text{with } \mathbf{x} = \mathbf{C}^{-1} \mathbf{y} \quad (5.25)$$

Assuming the pre-conditioning matrix \mathbf{C} is symmetric and positive definite, then it can be represented as

$$\mathbf{C} = \mathbf{H} \mathbf{H}^T \quad (5.26)$$

where \mathbf{H} is not singular. Then the equation system (5.13) can be transformed to

$$\underbrace{\mathbf{H}^{-1} \mathbf{A} \mathbf{H}^{-T}}_{\tilde{\mathbf{A}}} \underbrace{\mathbf{H}^T \mathbf{x}}_{\tilde{\mathbf{x}}} = \underbrace{\mathbf{H}^{-1} \mathbf{b}}_{\tilde{\mathbf{b}}} \quad (5.27)$$

The application of the CG method to the above equation is denoted as pre-conditioned conjugate gradient (PCG) scheme. The original solution vector \mathbf{x} is simply obtained from

$$\mathbf{x} = \mathbf{H}^{-T} \tilde{\mathbf{x}} \quad (5.28)$$

The pre-conditioning matrix \mathbf{C} has to be selected in a way that the condition number improves: $\kappa(\tilde{\mathbf{A}}) < \kappa(\mathbf{A})$. The sparse system matrix \mathbf{A} can be decomposed into its lower and upper triangle matrices \mathbf{L} , \mathbf{U} and its diagonal matrix \mathbf{D} .

$$\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U} \quad (5.29)$$

Axelsson proposed the following approach for the conditioning matrix.

$$\begin{aligned} \mathbf{C} &= (\mathbf{D} + \omega \mathbf{L}) \mathbf{D}^{-1} (\mathbf{D} + \omega \mathbf{U}) \\ &= \mathbf{D} + \omega \mathbf{L} + \omega \mathbf{U} + \omega^2 \mathbf{L} \mathbf{D}^{-1} \mathbf{U} \end{aligned} \quad (5.30)$$

with ω as relaxation parameter.

The following list gives some commonly used pre-conditioning techniques:

- Jacobian method (diagonal pre-conditioning): $\mathbf{C} = \mathbf{D}$, i.e. $\omega = 0$
- over-relaxation methods (JOR and SSOR): $\mathbf{D} = \mathbf{I}$,
- polynomial pre-conditioning
- incomplete Cholesky pre-conditioning (incomplete factorization): $\mathbf{C} \approx \mathbf{L} \mathbf{L}^T$ (see section 5.3.1)

5.3.2 Non-Linear Solver

In this section we present a description of selected iterative methods that are commonly applied to solve non-linear problems.

- Picard method (fixpoint iteration) (section 5.3.2)
- Newton methods (section 5.3.2)
- Cord slope method

- Dynamic relaxation method

All methods call for an initial guess of the solution to start but each algorithm uses a different scheme to produce a new (and hopefully closer) estimate to the exact solution. The general idea is to construct a sequence of linear sub-problems which can be solved with ordinary linear solver (see section 5.3.1).

Picard Method

The general algorithm of the Picard method can be described as follows. We consider a non-linear equation written in the form

$$\mathbf{A}(\mathbf{x}) \mathbf{x} - \mathbf{b}(\mathbf{x}) = 0 \quad (5.31)$$

We start the iteration by assuming an initial guess \mathbf{x}_0 and we use this to evaluate the system matrix $\mathbf{A}(\mathbf{x}_0)$ as well as the right-hand-side vector $\mathbf{b}(\mathbf{x}_0)$. Thus this equation becomes linear and it can be solved for the next set of \mathbf{x} values.

$$\begin{aligned} \mathbf{A}(\mathbf{x}_{k-1}) \mathbf{x}_k - \mathbf{b}(\mathbf{x}_{k-1}) &= 0 \\ \mathbf{x}_k &= \mathbf{A}^{-1}(\mathbf{x}_{k-1}) \mathbf{b}(\mathbf{x}_{k-1}) \end{aligned} \quad (5.32)$$

Repeating this procedure we obtain a sequence of successive solutions for \mathbf{x}_k . During each iteration loop the system matrix and the right-hand-side vector must be updated with the previous solution. The iteration is performed until satisfactory convergence is achieved. A typical criterion is e.g.

$$\varepsilon \geq \frac{\|\mathbf{x}_k - \mathbf{x}_{k-1}\|}{\|\mathbf{x}_k\|} \quad (5.33)$$

where ε is a user-defined tolerance criterion. For the simple case of a non-linear equation $\mathbf{x} = \mathbf{b}(\mathbf{x})$ (i.e. $\mathbf{A} = \mathbf{I}$), the iteration procedure is graphically illustrated in Fig. 5.4. To achieve convergence of the scheme it has to be guaranteed that the iteration error

$$e_k = \|\mathbf{x}_k - \mathbf{x}\| < C \|\mathbf{x}_{k-1} - \mathbf{x}\|^p = e_{k-1} \quad (5.34)$$

or, alternatively, the distance between successive solutions will reduce

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| < \|\mathbf{x}_k - \mathbf{x}_{k-1}\|^p \quad (5.35)$$

where p denotes the convergence order of the iteration scheme. It can be shown that the iteration error of the Picard method decreases linearly with the error at the previous iteration step. Therefore, the Picard method is a first-order convergence scheme.

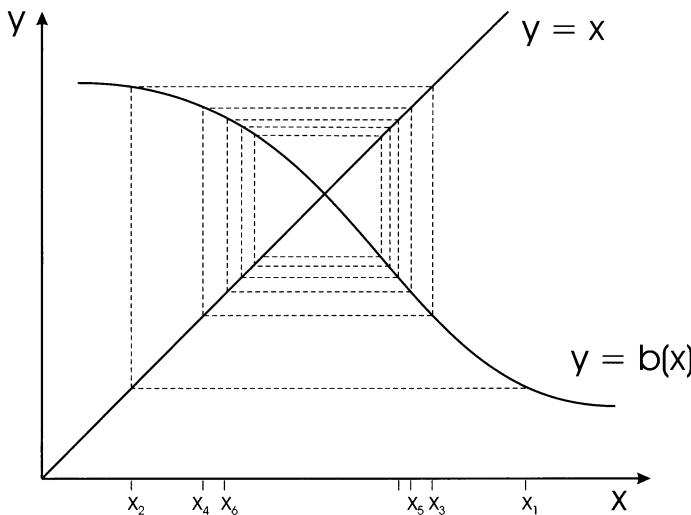


Figure 5.4: Graphical illustration of the Picard iteration method

Newton Method

In order to improve the convergence order of non-linear iteration methods, i.e. derive higher-order schemes, the Newton-Raphson method can be employed. To describe this approach, we consider once again the non-linear equation (5.5).

$$\mathbf{R}(\mathbf{x}) = \mathbf{A}(\mathbf{x})\mathbf{x} - \mathbf{b}(\mathbf{x}) = 0 \quad (5.36)$$

Assuming that the residuum $\mathbf{R}(\mathbf{x})$ is a continuous function, we can develop a Taylor series expansion about any known approximate solution \mathbf{x}_k .

$$\mathbf{R}_{k+1} = \mathbf{R}_k + \left[\frac{\partial \mathbf{R}}{\partial \mathbf{x}} \right]_k \Delta \mathbf{x}_{k+1} + o(\Delta \mathbf{x}_{k+1}^2) \quad (5.37)$$

Second- and higher-order terms are truncated in the following. The term $\partial \mathbf{R} / \partial \mathbf{x}$ represents tangential slopes of \mathbf{R} with respect to the solution vector and it is denoted as the Jacobian matrix \mathbf{J} . As a first approximation we can assume $\mathbf{R}_{k+1} = 0$. Then the solution increment can be immediately calculated from the remaining terms in equation (5.37).

$$\Delta \mathbf{x}_{k+1} = -\mathbf{J}_k^{-1} \mathbf{R}_k \quad (5.38)$$

where we have to cope with the inverse of the Jacobian. The iterative approximation of the solution vector can be computed now from the increment.

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta \mathbf{x}_{k+1} \quad (5.39)$$

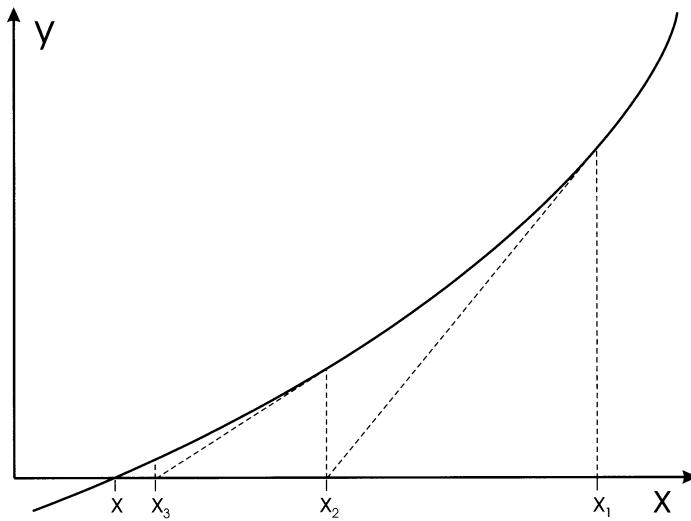


Figure 5.5: Graphical illustration of the Newton-Raphson iteration method

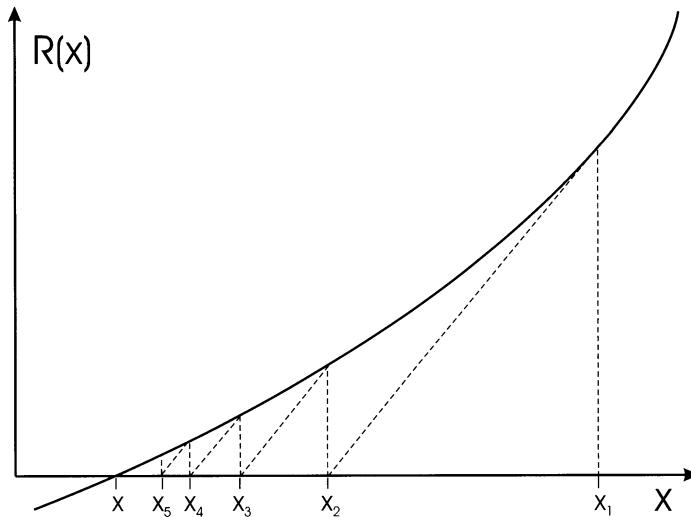


Figure 5.6: Graphical illustration of the modified Newton-Raphson iteration method

Once an initial guess is provided, successive solutions of \mathbf{x}_{k+1} can be determined using equations (5.38) and (5.39) (Fig. 5.5). The Jacobian has to re-evaluated and inverted at every iteration step, which is a very time-consuming procedure in fact. At the expense of slower convergence, the initial Jacobian \mathbf{J}_0 may be kept and used in the subsequent iterations. Alternatively, the Jacobian can be

updated in certain iteration intervals. This procedure is denoted as modified or 'initial slope' Newton method (Fig. 5.6).

The convergence velocity of the Newton-Raphson method is second-order. It is characterized by the expression.

$$\| \mathbf{x}_{k+1} - \mathbf{x} \| \leq C \| \mathbf{x}_k - \mathbf{x} \|^2 \quad (5.40)$$

5.4 Problems

Solution Procedure

- 1 Give examples of approximation methods for solving differential equations.
- 2 What are the two basic steps of the solution procedure for discrete approximation methods ?

Theory of Discrete Approximation

- 3 Explain the term convergence of an approximate solution for a PDE. Give a mathematical definition for that.
- 4 Explain the term consistency of an approximation scheme for a PDE. Give a mathematical definition for that.
- 5 Explain the term stability of an approximate solution for a PDE. What general methods for stability analysis do you know ?
- 6 Explain the relationships between the terms convergence, consistency, and stability using Fig. (5.3).
- 7 What does the Lax equivalence theorem postulate ?
- 8 What are the three analysis steps for discrete approximation schemes ?

Solution Process

- 9 Using the Newton-Raphson method solve the following set of non-linear equations:

$$\begin{aligned}f_1(x_1, x_2) &= x_1^2 + x_2^2 - 5 = 0 \\f_2(x_1, x_2) &= x_1 + x_2 - 1 = 0\end{aligned}$$

Chapter 6

Finite Difference Method

The basic steps in order to set up a finite difference scheme are:

- definition of a space discretization by which the mesh points are distributed along families of non-intersecting lines,
- development of the unknown functions by means of Taylor series expansion (TSE) around grid points
- replacement of derivative terms in the partial differential equations (PDE) with equivalent finite difference expressions.

6.1 Approximation of Derivatives

In this section we present basic techniques to discretize derivatives. Algebraic formulas are constructed first by Taylor series expansion, to evaluate the truncation error and second by a more general technique.

6.1.1 Taylor Series Expansion (TSE)

A well-known technique to represent the value of a function $u(t, x)$ at a node (j, n) in terms of values of the function at nearby points is the Taylor series expansion

in time

$$u_j^{n+1} = \sum_{m=0}^{\infty} \frac{\Delta t^m}{m!} \left[\frac{\partial^m u}{\partial t^m} \right]_j^n \quad (6.1)$$

in space

$$u_{j+1}^n = \sum_{m=0}^{\infty} \frac{\Delta x^m}{m!} \left[\frac{\partial^m u}{\partial x^m} \right]_j^n \quad (6.2)$$

These series may be truncated after any number of terms. The resulting **truncation error** is then defined by the next term being left in the series expansion. The above expressions can be rewritten in following forms.

$$u_j^{n+1} = u_j^n + \Delta t \left[\frac{\partial u}{\partial t} \right]_j^n + \frac{\Delta t^2}{2} \left[\frac{\partial^2 u}{\partial t^2} \right]_j^n + 0(\Delta t^3) \quad (6.3)$$

$$u_{j+1}^n = u_j^n + \Delta x \left[\frac{\partial u}{\partial x} \right]_j^n + \frac{\Delta x^2}{2} \left[\frac{\partial^2 u}{\partial x^2} \right]_j^n + 0(\Delta x^3) \quad (6.4)$$

As can be seen, the truncation error depends mainly on the step size Δt or Δx , respectively. The error introduced in this approximation rapidly reduces as the step size decreases.

6.1.2 First-Order Derivatives

By rearranging the above equations, a finite difference expression for first-order derivatives can be directly obtained.

$$\left[\frac{\partial u}{\partial t} \right]_j^n = \frac{u_j^{n+1} - u_j^n}{\Delta t} - \frac{\Delta t}{2} \left[\frac{\partial^2 u}{\partial t^2} \right]_j^n + 0(\Delta t^2) \quad (6.5)$$

$$\left[\frac{\partial u}{\partial x} \right]_j^n = \frac{u_{j+1}^n - u_j^n}{\Delta x} - \frac{\Delta x}{2} \left[\frac{\partial^2 u}{\partial x^2} \right]_j^n + 0(\Delta x^2) \quad (6.6)$$

Clearly, the following standard finite difference expressions for first-order derivatives are accurate at first order:

Forward difference approximation

$$\left[\frac{\partial u}{\partial x} \right]_j^n = \frac{u_{j+1}^n - u_j^n}{\Delta x} + 0(\Delta x) \quad (6.7)$$

Backward difference approximation

$$\left[\frac{\partial u}{\partial x} \right]_j^n = \frac{u_j^n - u_{j-1}^n}{\Delta x} + 0(\Delta x) \quad (6.8)$$

A second-order scheme is obtained from central differences. To this purpose, we subtract the following TSE from each other.

$$\begin{aligned} u_{j+1}^n &= u_j^n + \Delta x \left[\frac{\partial u}{\partial x} \right]_j^n + \frac{\Delta x^2}{2} \left[\frac{\partial^2 u}{\partial x^2} \right]_j^n + O(\Delta x^3) \\ u_{j-1}^n &= u_j^n - \Delta x \left[\frac{\partial u}{\partial x} \right]_j^n + \frac{\Delta x^2}{2} \left[\frac{\partial^2 u}{\partial x^2} \right]_j^n - O(\Delta x^3) \end{aligned} \quad (6.9)$$

Central difference approximation

$$\left[\frac{\partial u}{\partial x} \right]_j^n = \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} + O(\Delta x^2) \quad (6.10)$$

A geometric interpretation (i.e. slopes) of forward and backward difference formulas is given in following graphic. Note, forward and backward differences only use information from one side of the corresponding grid point, whereas central differences use information from both sides.

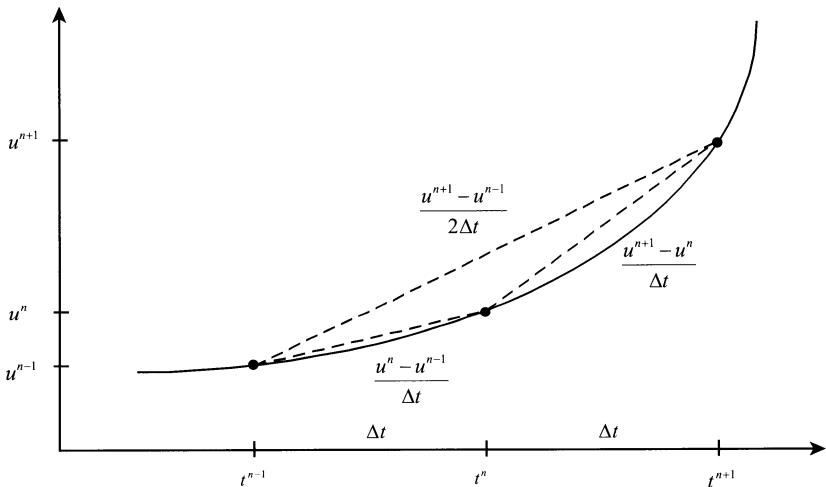


Figure 6.1: Geometric interpretation of finite difference approximations

6.1.3 Second-Order Derivatives

Finite difference schemes for higher-order derivatives can be obtained by repeated application of first-order approximations.

$$\left[\frac{\partial^2 u}{\partial x^2} \right]_j^n \approx \frac{1}{\Delta x} \left(\left[\frac{\partial u}{\partial x} \right]_{j+1}^n - \left[\frac{\partial u}{\partial x} \right]_j^n \right)$$

$$\approx \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2} \quad (6.11)$$

The FD scheme for a second-order derivative can be obtained by adding the first-order schemes given in eqn (6.9). From the TSE can be seen that the symmetrical, central differences scheme is of second-order accuracy.

$$\left[\frac{\partial^2 u}{\partial x^2} \right]_j^n = \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2} + \frac{\Delta x^2}{12} \left[\frac{\partial^4 u}{\partial x^4} \right]_j^n + \dots \quad (6.12)$$

Note, that first-order schemes for n -th derivatives will introduce higher-order $n + 1$ -th derivative terms, which result in large numerical diffusion/dispersion terms. Therefore, we may infer the general guidance, higher-order schemes for derivatives should be preferred.

6.2 Diffusion Equation

Typical diffusion problems are mass diffusion, heat conduction, and viscous fluid flow. The diffusion equation is a parabolic PDE.

$$\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial x^2} = 0 \quad (6.13)$$

The unknown field variable u may be interpreted as velocity, vorticity, temperature or concentration depending on whether the diffusion of momentum, vorticity, heat or mass is being considered. Boundary as well as initial conditions must be specified to obtain an unique solution of such a PDE.

In this chapter we introduce several (explicit and implicit) finite difference schemes for the numerical solution of the diffusion equation (Fig. 6.2) Attention will be paid to stability and accuracy of the several algorithms. Moreover, the implementation of initial as well as boundary conditions will also be considered.

Analysis of approximation schemes consists of three steps:

- Develop the **algebraic scheme**,
- Check **consistency** of the algebraic approximate equation,
- Investigate **stability** behavior of the scheme.

6.2.1 Explicit and Implicit Schemes

A discrete scheme gives the solution at a new time level in terms of the known solution at earlier time levels. An algebraic scheme is denoted as an explicit one if the unknown at new time level u_j^{n+1} depends only on values from old time

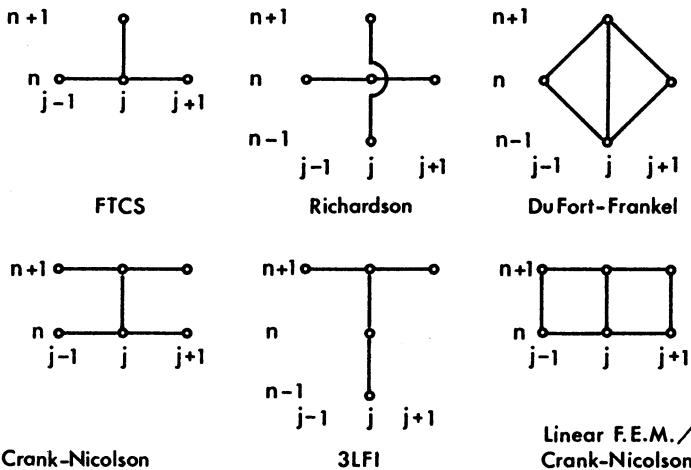


Figure 6.2: Overview on finite difference schemes

level u_j^n . The FTCS, Richardson, and DuFort-Frankel algorithms are examples of explicit schemes. For implicit schemes spatial derivatives are evaluated also at the new time level. As a result equations are coupled for each node at new time level and, therefore, a system of algebraic equations has to be solved.

Typically implicit schemes are unconditionally stable. However the use of implicit schemes leads to a system of coupled algebraic equations at new time level. As a result, perturbations introduced at one node affects the solution to all other nodes at the next time level. Those perturbations are propagated immediately through entire domain. Physically this behavior corresponds to a diffusion process. Therefore, this type of error propagation is denoted as numerical diffusion or dispersion.

6.2.2 Explicit FTCS Scheme

Algebraic Scheme

One of the simplest finite difference representations of the diffusion equation is the forward time, centered space scheme (FTCS). To obtain a discretized form of the PDE we replace the time derivatives by a two-point forward difference formula

$$\left[\frac{\partial u}{\partial t} \right]_j^n \approx \frac{u_j^{n+1} - u_j^n}{\Delta t} \quad (6.14)$$

and the spatial derivative by a three-point centered difference formula.

$$\left[\frac{\partial^2 u}{\partial x^2} \right]_j^n \approx \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} \quad (6.15)$$

The resulting algebraic equation is denoted as FTCS scheme.

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} - \alpha \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} = 0 \quad (6.16)$$

The discretization process implies, that the original problem (i.e. PDE for the exact, continuous solution) $u(t, x)$ has been replaced with the problem of finding discrete values u_j^n at node j at time n . Rearranging the above formula we obtain the expression for the unknown value u_j^{n+1} at new time level in terms of known values at old time level.

$$u_j^{n+1} = u_j^n + \frac{\alpha \Delta t}{\Delta x^2} (u_{j-1}^n - 2u_j^n + u_{j+1}^n) \quad (6.17)$$

where

$$Ne = \boxed{\frac{\alpha \Delta t}{\Delta x^2}} \quad (6.18)$$

is the **Neumann number**. For the above developed explicit scheme only a single unknown appears on the left hand side of the algebraic equation.

Consistency Analysis

Substitution of the exact solutions (i.e. its Taylor series expansion) for the differential terms (eqns 6.5, 6.12) into the algebraic scheme yields

$$\left[\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial x^2} \right]_j^n + E_j^n = 0 \quad (6.19)$$

with truncation error

$$E_j^n = \left[\frac{\Delta t}{2} \frac{\partial^2 u}{\partial t^2} - \alpha \frac{\Delta x^2}{12} \frac{\partial^4 u}{\partial x^4} \right]_j^n + O(\Delta t^2, \Delta x^4) = 0 \quad (6.20)$$

The leading term of the truncation error indicates that the FTCS scheme is first-order accurate in time and second-order accurate in space. The FTCS scheme is consistent with the PDE. As the step size tends to zero both equations (SAE and PDE) will coincide.

Stability Analysis

Stability is the tendency for any perturbations in the approximate solution to decay. As mentioned in the previous section, several methods exist for stability analysis such as the matrix method and the von Neumann method.

(1) Matrix Method

Using the matrix method, the system of algebraic equations is expressed in matrix form first. For stability consideration then the eigenvalues are examined. The algebraic scheme (6.17) can be written as

$$\mathbf{u}^{n+1} = \mathbf{A}\mathbf{u}^n \quad , \quad n = 0, 1, 2, \dots \quad (6.21)$$

where \mathbf{A} is a $(np - 2)$ (np = number of grid points) square matrix transforming the result vector \mathbf{u} from old to new time level

$$\mathbf{A} = \begin{bmatrix} 1 - 2\frac{\alpha\Delta t}{\Delta x^2} & \frac{\alpha\Delta t}{\Delta x^2} & & & \\ \frac{\alpha\Delta t}{\Delta x^2} & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \frac{\alpha\Delta t}{\Delta x^2} \\ & & & \frac{\alpha\Delta t}{\Delta x^2} & 1 - 2\frac{\alpha\Delta t}{\Delta x^2} \end{bmatrix} \quad (6.22)$$

$$\mathbf{u}^n = \begin{bmatrix} u_2^n \\ u_3^n \\ \dots \\ u_{np-2}^n \\ u_{np-1}^n \end{bmatrix} \quad (6.23)$$

Note, it is assumed that the boundary values u_1^n , u_{np}^n are known. It can be shown that the result vector is bounded for propagating time if the eigenvalues λ_j of matrix \mathbf{A} are defined and have absolute values less or equal to one, i.e.

$$|\lambda_j| \leq 1 \quad , \quad \forall j \quad (6.24)$$

The eigenvalues of the above tridiagonal matrix are

$$\lambda_j = 1 - 4\frac{\alpha\Delta t}{\Delta x^2} \sin^2 \left[\frac{j\pi}{2(np-1)} \right] \quad , \quad j = 1, \dots, np-2 \quad (6.25)$$

Then stability condition (6.24) can be written as

$$-1 \leq 1 - 4\frac{\alpha\Delta t}{\Delta x^2} \sin^2 \left[\frac{j\pi}{2(np-1)} \right] \leq +1 \quad (6.26)$$

Whereas the right-hand side of the inequality is always satisfied, the left-hand side yields the stability condition for the FTCS scheme.

$$\boxed{Ne = \frac{\alpha \Delta t}{\Delta x^2} \leq \frac{1}{2}} \quad (6.27)$$

The matrix method is based on Fourier series representation of the result vector. Therefore, this method yields growth or decay of individual modes in a Fourier series representation of the initial conditions.

Incorporation of Neumann boundary conditions result in a slightly different scheme. Suppose at point ($j = 1$)

$$\frac{\partial u(t, x_1)}{\partial x} = g(t) \quad (6.28)$$

the algebraic equation (6.16) for this point can be written as

$$\begin{aligned} \frac{u_1^{n+1} - u_1^n}{\Delta t} - \alpha \left(\frac{u_2^n - u_1^n}{\Delta x^2} - \frac{1}{\Delta x} \left[\frac{\partial u}{\partial x} \right]_1^n \right) &= \\ = \frac{u_1^{n+1} - u_1^n}{\Delta t} - \alpha \left(\frac{u_2^n - u_1^n}{\Delta x^2} - \frac{1}{\Delta x} g(t) \right) &= 0 \end{aligned} \quad (6.29)$$

Rearranging the above equation, we obtain the following explicit scheme for the unknown value at new time level.

$$u_1^{n+1} = u_1^n + \frac{\alpha \Delta t}{\Delta x^2} [u_2^n - u_1^n - \Delta x g(t)] \quad (6.30)$$

The matrix form for the system of algebraic equations is now

$$\mathbf{u}^{n+1} = \mathbf{A}\mathbf{u}^n + \mathbf{b} \quad , \quad n = 0, 1, 2, \dots \quad (6.31)$$

with a modified system matrix and an additional vector resulting from Neumann type boundary conditions.

$$\mathbf{A} = \begin{bmatrix} 1 - \frac{\alpha \Delta t}{\Delta x^2} & \frac{\alpha \Delta t}{\Delta x^2} & & & \\ \frac{\alpha \Delta t}{\Delta x^2} & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \frac{\alpha \Delta t}{\Delta x^2} \\ & & & \frac{\alpha \Delta t}{\Delta x^2} & 1 - \frac{\alpha \Delta t}{\Delta x^2} \end{bmatrix} \quad (6.32)$$

$$\mathbf{b} = \begin{bmatrix} -\frac{\alpha \Delta t}{\Delta x} g(t^n) \\ 0 \\ \dots \\ 0 \\ 0 \end{bmatrix} \quad (6.33)$$

It can be seen, that one eigenvalue will be slightly modified. The maximum eigenvalues can be determined by use of the power method. In general, the implementation of Neumann boundary conditions slightly reduces numerical stability (Fletcher 1990).

(2) Von Neumann Method

The von Neumann method is most commonly used, but it is restricted to linear initial value problems with constant coefficients. For more sophisticated problems including variable coefficients, non-linearities, and complicated boundary conditions, this method is useful to determine necessary conditions. But it will fail to define sufficient conditions for stability. The von Neumann method is based on Fourier analysis. Stability or instability of the algebraic scheme is investigated by considering whether defined Fourier components of the error decay or amplify in progressing to the next time level.

6.2.3 Fully Implicit Scheme

Algebraic Scheme

For the fully implicit scheme, the spatial derivative (diffusion term) is evaluated completely at the unknown time level.

$$\left[\frac{\partial^2 u}{\partial x^2} \right]_j^{n+1} \approx \frac{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}{\Delta x^2} \quad (6.34)$$

The algebraic scheme for the fully implicit discretization of the diffusion equation is then

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} - \alpha \frac{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}{\Delta x^2} = 0 \quad (6.35)$$

Rearranging the above formula we obtain an expression for the unknown values at new time level in terms of known values at old time level.

$$\frac{\alpha \Delta t}{\Delta x^2} (-u_{j-1}^{n+1} + 2u_j^{n+1} - u_{j+1}^{n+1}) + u_j^{n+1} = u_j^n \quad (6.36)$$

Consistency Analysis

To analyze consistency we start from equation (6.35) which will be applied to the exact solution. First, outside terms are expanded as Taylor series about the central node j at new time level and substituted into equation (6.35).

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} - \alpha \left\{ \left[\frac{\partial^2 u}{\partial x^2} \right]_j^{n+1} + \frac{\Delta x^2}{12} \left[\frac{\partial^4 u}{\partial x^4} \right]_j^{n+1} + \frac{\Delta x^4}{360} \left[\frac{\partial^6 u}{\partial x^6} \right]_j^{n+1} + \dots \right\} = 0 \quad (6.37)$$

Second, central terms at new time level are expanded about the (j, n) -th node and then substituted.

$$\begin{aligned} & \left[\frac{\partial u}{\partial t} \right]_j^n + \frac{\Delta t}{2} \left[\frac{\partial^2 u}{\partial t^2} \right]_j^n + \frac{\Delta t^2}{6} \left[\frac{\partial^3 u}{\partial t^3} \right]_j^n + \dots \\ & -\alpha \left[\frac{\partial^2 u}{\partial x^2} \right]_j^n + \Delta t \left[\frac{\partial^3 u}{\partial x^2 \partial t} \right]_j^n + \frac{\Delta t^2}{2} \left[\frac{\partial^4 u}{\partial x^2 \partial t^2} \right]_j^n + \dots \\ & + \frac{\Delta x^2}{12} \left(\left[\frac{\partial^4 u}{\partial x^4} \right]_j^n + \Delta t \left[\frac{\partial^5 u}{\partial x^4 \partial t} \right]_j^n + \dots \right) \\ & + \frac{\Delta x^4}{360} \left(\left[\frac{\partial^6 u}{\partial x^6} \right]_j^n + \dots \right) = 0 \end{aligned} \quad (6.38)$$

We use the original diffusion equation and related time derivatives to substitute all spatial derivatives.

$$\left[\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial x^2} \right]_j^n = 0 \quad , \quad \left[\frac{\partial^2 u}{\partial t^2} - \alpha^2 \frac{\partial^4 u}{\partial x^4} \right]_j^n = 0 \quad , \quad \left[\frac{\partial^3 u}{\partial t^3} - \alpha^3 \frac{\partial^6 u}{\partial x^6} \right]_j^n = 0 \quad (6.39)$$

By this step it would be possible to represent the truncation error in terms of derivatives and spatial grid size only. Now we can simply the above equation to

$$\left[\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial x^2} \right]_j^n + E_j^n = 0 \quad (6.40)$$

and, finally, we determine the truncation error of the fully implicit scheme.

$$\begin{aligned} E_j^n = & -\frac{\Delta t}{2} \left(1 + \frac{1}{6} \frac{\Delta x^2}{\alpha \Delta t} \right) \left[\frac{\partial^2 u}{\partial t^2} \right]_j^n - \frac{\Delta t^2}{3} \left(1 + \frac{1}{4} \frac{\Delta x^2}{\alpha \Delta t} + \frac{1}{120} \frac{\Delta x^4}{\alpha^2 \Delta t^2} \right) \left[\frac{\partial^3 u}{\partial t^3} \right]_j^n + \\ & + 0(\Delta t^3, \Delta x^4) \end{aligned} \quad (6.41)$$

The truncation error vanishes as time step size tends to zero, i.e. algebraic and partial differential equations coincide in the limit. Consequently, the algebraic equation is consistent with the PDE. In contrast to the explicit FTCS scheme (6.20), there is no specific choice of time and spatial grid sizes to reduce the truncation error further. Consistency order of the fully implicit scheme is therefore: $O(\Delta t, \Delta x^2)$.

Stability Analysis

We use the matrix method for stability analysis. In analogy to the procedure for the explicit FTCS scheme (section 6.2.2) we write the system of algebraic

equations (6.36) in matrix form. If neglecting boundary conditions, the system matrix for the fully implicit scheme is

$$\mathbf{A} = \begin{bmatrix} 1 + 2\frac{\alpha\Delta t}{\Delta x^2} & -\frac{\alpha\Delta t}{\Delta x^2} & & & \\ -\frac{\alpha\Delta t}{\Delta x^2} & \dots & \dots & & \\ & \dots & \dots & \dots & \\ & & \dots & \dots & -\frac{\alpha\Delta t}{\Delta x^2} \\ & & & -\frac{\alpha\Delta t}{\Delta x^2} & 1 + 2\frac{\alpha\Delta t}{\Delta x^2} \end{bmatrix}^{-1} \quad (6.42)$$

and the corresponding eigenvalues are

$$\lambda_j = \left[1 + 4\frac{\alpha\Delta t}{\Delta x^2} \sin^2 \left(\frac{j\pi}{2(np-1)} \right) \right]^{-1}, \quad j = 1, \dots, np-2 \quad (6.43)$$

For any choice of time and spatial step sizes the value of eigenvalues will be less or equal to one.

$$|\lambda_j| \leq 1, \quad \forall \Delta t, \Delta x \quad (6.44)$$

Consequently, the fully implicit scheme is unconditionally stable. This is clearly an advantage in comparison to the conditionally stable explicit FTCS scheme. However, for implicit schemes we have to solve the complete equation system, because unknown values at new time level in several nodes are connected. It is apparent that the system of equations to be solved has tridiagonal structure.

6.2.4 Crank-Nicolson Scheme (CNS)

Algebraic Scheme

A specific implicit algorithm with improved consistency order is the Crank-Nicolson scheme (CNS). Spatial derivatives are evaluated at old n -th as well as at new $(n+1)$ -th time level, i.e. at intermediate $(n+1/2)$ -th level. The CNS, which is also called a semi-implicit algorithm, is given by

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} - \frac{\alpha}{2} \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} - \frac{\alpha}{2} \frac{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}{\Delta x^2} = 0 \quad (6.45)$$

Rearranging the above formula we obtain an expression for the unknown values at new time level in terms of known values at old time level. Again the implicit three-point scheme produces a tridiagonal SAE.

$$\frac{\alpha\Delta t}{\Delta x^2} \left(-\frac{1}{2}u_{j-1}^{n+1} + u_j^{n+1} - \frac{1}{2}u_{j+1}^{n+1} \right) + u_j^{n+1} = u_j^n \quad (6.46)$$

Consistency - Applying the procedure based on TSE about node j at $n+1/2$ -th level we found that the scheme is consistent with a truncation error of $O(\Delta t^2, \Delta x^2)$.

Stability - Stability analysis yields that the CNS is unconditionally stable. However, this scheme is on the limit of unconditionally stability. The CNS can produce oscillatory solutions for unfavourable time step and grid sizes.

Mitchell & Griffits (1980, pp 47-53) investigated the application of the CNS to the diffusion equation in detail. Additionally, they considered various boundary conditions. In connection with Dirichlet boundary conditions the CNS is unconditionally stable, whereas Neumann boundary conditions introduce eigenvalues equal to unity, which produces oscillatory solutions. For Cauchy boundary conditions additional restrictions are necessary to ensure stability.

6.2.5 Generalized Scheme

The above discussed discretization schemes can be generalized by writing

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} - \alpha \left(1 - \theta \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} + \theta \frac{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}{\Delta x^2} \right) = 0 \quad (6.47)$$

where $0 \leq \theta \leq 1$ is a weighting (collocation) factor. The following table summarizes some properties of several specified schemes. Generalized three-level schemes and higher-order schemes are discussed e.g. by Fletcher (1990).

Collocation factor	Scheme	Consistency order	Stability condition
$\theta = 0$	explicit	$O(\Delta t, \Delta x^2)$	
$\theta = 0$	improved FTCS	$O(\Delta t^2, \Delta x^4)$	$\Delta t \leq \frac{\Delta x^2}{2\alpha(1-2\theta)}$
$0 < \theta < 0.5$	implicit	$O(\Delta t, \Delta x^2)$	
$\theta = 0.5$	CNS	$O(\Delta t^2, \Delta x^2)$	
$0.5 < \theta < 1$	implicit	$O(\Delta t, \Delta x^2)$	
$\theta = 1$	fully implicit	$O(\Delta t, \Delta x^2)$	unconditionally stable

6.2.6 Initial and Boundary Conditions

Initial Conditions

In general, the implementation of initial conditions does not cause any difficulty. Problems rise if initial and boundary conditions are not compatible, i.e. if the value at a boundary node specified by an initial condition is different from the value specified by the boundary condition. In this case a suitable strategy is to average the values for the first time step, and use the boundary condition for

subsequent time. Sometimes (e.g. for multi-level schemes), integration in time should be used to improve accuracy.

Dirichlet Boundary Conditions

Up to now we developed algebraic schemes for PDEs by approximation of the corresponding derivatives. We discussed the numerical properties (consistency, stability) of this approximation schemes resulting from the discretization process. Beside the approximation of derivatives, additional errors can be introduced from the implementation of boundary and initial conditions. Implementation of Dirichlet boundary conditions is trivial, because the value of unknown is explicitly given. The treatment of Neumann boundary conditions is more expensive.

Neumann Boundary Conditions

Use of centered differences for approximation of Neumann boundary conditions would require information from outside the domain. A simple way to represent the boundary condition is the use of one-sided differences.

$$\left[\frac{\partial u}{\partial x} \right]_1^{n+1} = g^{n+1} = \frac{u_2^{n+1} - u_1^{n+1}}{\Delta x} \quad (6.48)$$

The value of unknown at boundary can be calculated now by

$$u_2^{n+1} = u_1^{n+1} + g^{n+1} \Delta x \quad (6.49)$$

The problem here is that this boundary condition representation has a first-order truncation error. Therefore using this formula, the accuracy of higher-order schemes (e.g. FTCS scheme) will be disturbed. In particular for diffusion problems (parabolic PDE) lower-order accuracy at the boundary will affect the accuracy of the solution within the whole domain for all later time. Therefore, we should design a scheme for nodes with Neumann boundary conditions that has the same order of accuracy as the scheme for interior nodes.

To achieve that, we introduce a fictitious node outside the domain at old time level. Now a second-order accurate formula in space can be applied to fit the boundary condition (see Chapter 6.1).

$$\left[\frac{\partial u}{\partial x} \right]_1^n = g^n = \frac{u_2^n - u_0^n}{2\Delta x} \quad (6.50)$$

Due to the temporary expansion of the computational domain we can apply an appropriate scheme for internal nodes (e.g. explicit FTCS) to obtain the value of the 'boundary' node at new time level.

$$u_1^{n+1} = u_1^n + \frac{\alpha \Delta t}{\Delta x^2} (u_0^n - 2u_1^n + u_2^n) \quad (6.51)$$

Now the above boundary condition representation can be used to eliminate the fictitious value u_0^n again. Finally we obtain

$$u_1^{n+1} = -2\frac{\alpha\Delta t}{\Delta x}g^n + \left(1 - \frac{2\alpha\Delta t}{\Delta x^2}\right)u_1^n - \frac{\alpha\Delta t}{\Delta x^2}u_2^n \quad (6.52)$$

and the truncation error will be of second order in space everywhere in the computational domain.

If an implicit scheme is being used for discretization of the PDE, the combination of boundary condition representation and the scheme for interior nodes yields.

$$\left(1 + \frac{2\alpha\Delta t}{\Delta x^2}\right)u_1^{n+1} - \frac{2\alpha\Delta t}{\Delta x^2}u_2^{n+1} = u_1^n - \frac{2\alpha\Delta t}{\Delta x}g^n \quad (6.53)$$

The accuracy of Neumann boundary condition implementation is discussed in detail by Fletcher (1990, pp. 238-241).

6.3 Advection Equation

In this section we examine the following linear, hyperbolic partial differential equation describing advection processes

$$\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x} = 0 \quad (6.54)$$

where $u(t, x)$ is an unknown scalar function (e.g. mass concentration, temperature) and v is the known propagation velocity of quantity u . The above equation is the advective part of the transport equation we discuss later (see 6.4).

6.3.1 Explicit FTCS Scheme

Algebraic Scheme

Applying the FTCS scheme (see section 6.2.2) to the linear advection equation we obtain the following algebraic equation

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + v\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = 0 \quad (6.55)$$

The resulting expression for the unknown value in j -th node at $n + 1$ -th time level is

$$u_j^{n+1} = u_j^n - \frac{1}{2}\frac{v\Delta t}{\Delta x}(u_{j+1}^n - u_{j-1}^n) \quad (6.56)$$

Introducing now the **Courant number**

$$\boxed{Cr = \frac{v\Delta t}{\Delta x}} \quad (6.57)$$

which is an important characteristic number for numerical analysis, the above algebraic expression can be written as

$$u_j^{n+1} = u_j^n - \frac{Cr}{2}(u_{j+1}^n - u_{j-1}^n) \quad (6.58)$$

Stability Analysis

We proof stability of the FD scheme by an inductive manner. According to the von Neumann method we use a Fourier series mode representation of the solution.

$$u_j^n = (G)^n e^{i\theta j} = (G)^n [\cos(\theta j) + i \sin(\theta j)] \quad (6.59)$$

Substitution of the above Fourier mode representation into the algebraic equation (6.56) and then dividing through $(G)^n e^{i\theta j}$ yields

$$\frac{1}{\Delta t} G - 1 + \frac{1}{2\Delta x} (e^{i\theta} - e^{-i\theta}) = 0 \quad (6.60)$$

Finally, we obtain the following expression for the amplification factor

$$G = 1 - \frac{v\Delta t}{\Delta x} \frac{e^{i\theta} - e^{-i\theta}}{2} = 1 - iCr \sin \theta \quad (6.61)$$

and therefore

$$G^2 = 1 + Cr^2 \sin^2 \theta \rightarrow |G| \geq 1 \quad , \quad \forall \theta \quad (6.62)$$

Consequently, the FTCS scheme will fail for the advection equation because it is unconditionally unstable.

Consistency - The FTCS scheme is consistent with the advection equation with a truncation error of $O(\Delta t, \Delta x^2)$.

6.3.2 Upwind Difference Representation

Algebraic Scheme

To construct a stable scheme we combine one-side difference formulas (backward for positive and forward for negative velocities) for the spatial derivative

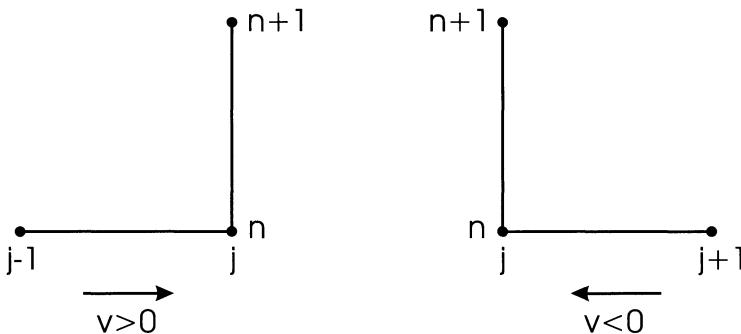


Figure 6.3: Finite difference pattern for upwind schemes

with the forward difference formula for the time derivative. By this way, the unknown value is determined by information upwind of node j . The corresponding algebraic equations, which are referred to as upwind schemes, are

for $v \geq 0$

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + v \frac{u_j^n - u_{j-1}^n}{\Delta x} = 0 \quad (6.63)$$

$$u_j^{n+1} = (1 - \frac{v\Delta t}{\Delta x})u_j^n + \frac{v\Delta t}{\Delta x}u_{j-1}^n = (1 - Cr)u_j^n + Cr u_{j-1}^n \quad (6.64)$$

and for $v \leq 0$

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + v \frac{u_{j+1}^n - u_j^n}{\Delta x} = 0 \quad (6.65)$$

$$u_j^{n+1} = (1 - \frac{v\Delta t}{\Delta x})u_{j+1}^n + \frac{v\Delta t}{\Delta x}u_j^n = (1 - |Cr|)u_{j+1}^n + |Cr|u_j^n \quad (6.66)$$

Consistency Analysis

Substitution of Taylor series expansion (6.5, 6.6) about node j at time level n into the above algebraic equation yields

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} + \frac{\Delta t}{2} \frac{\partial^2 u}{\partial t^2} - \frac{v\Delta x}{2\Delta t} \frac{\partial^2 u}{\partial x^2} + O(\Delta t^2, \Delta x^2) = 0 \quad (6.67)$$

This expression is consistent with the PDE and the leading term of truncation error has the order of $O(\Delta t, \Delta x)$. Substituting the second-order time derivative by

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} = 0 \rightarrow \frac{\partial^2 u}{\partial t^2} - v^2 \frac{\partial^2 u}{\partial x^2} = 0 \quad (6.68)$$

we can rewrite the above algebraic equation as

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} - \underbrace{\frac{v\Delta x}{2}(1 - Cr) \frac{\partial^2 u}{\partial x^2}}_{\text{Numerical diffusion}} + 0(\Delta t^2, \Delta x^2) = 0 \quad (6.69)$$

Therefore, a kind of artificial diffusion is introduced by this scheme.

$$\alpha^* = \frac{v\Delta x}{2}(1 - Cr) \quad (6.70)$$

This term can be reduced using small spatial grid steps and it becomes zero when the Courant number is equal to unity. Moreover, keeping of $Cr = 1$ provides the exact solution of the linear advection equation. For Courant numbers $Cr > 1$, the scheme becomes unstable due to 'negative' diffusion resulting in an infinite accumulation.

Stability

The von Neumann analysis reveals the following stability condition

$$Cr = \frac{v\Delta t}{\Delta x} \leq 1 \quad (6.71)$$

which is denoted as the **Courant-Friedrichs-Lowy condition** (CFL). The CFL condition is generally valid for explicit schemes for hyperbolic PDEs. Physically, the CFL condition means that a tracer (mass or heat) should not travel more than one grid step during one time step.

6.3.3 Leapfrog and Lax-Wendroff Schemes

Except the particular case of $Cr = 1$ which provides the exact solution for the linear pure advection equation, the above schemes suffer from instability or low (only first-order) accuracy. To improve accuracy we use centered differences in both time and space.

Algebraic Schemes

The Leapfrog scheme (the name comes from the typical shape - leapfrog - of used time and space node) gives

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\Delta t} + v \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = 0 \quad (6.72)$$

For determination of the unknown value we obtain

$$u_j^{n+1} = u_j^n - Cr(u_{j+1}^n - u_{j-1}^n) \quad (6.73)$$

To improve accuracy for the Lax-Wendroff scheme a Taylor series expansion is used for the time derivative.

$$\frac{\partial u}{\partial t} \approx \frac{u_j^{n+1} - u_j^n}{\Delta t} - \frac{\Delta t}{2} \frac{\partial^2 u}{\partial t^2} = \frac{u_j^{n+1} - u_j^n}{\Delta t} - \frac{v^2 \Delta t}{2} \frac{\partial^2 u}{\partial x^2} \quad (6.74)$$

Additionally we made use of eqn (6.68). The Lax-Wendroff scheme is developed based on a second-order formula for the time derivative and on the centered difference formula for spatial derivatives.

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} - \frac{v^2 \Delta t}{2} \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2} + v \frac{u_{j+1}^n - u_{j-1}^n}{2 \Delta x} = 0 \quad (6.75)$$

For determination of the unknown value we obtain the algebraic expression

$$u_j^{n+1} = u_j^n - Cr(u_{j+1}^n - u_{j-1}^n) + \frac{Cr^2}{2}(u_{j+1}^n - 2u_j^n + u_{j-1}^n) \quad (6.76)$$

Numerical Properties - Leapfrog as well as Lax-Wendroff schemes are consistent with the PDE and have a truncation error of $O(\Delta t^2, \Delta x^2)$. Both schemes are stable if the CFL condition is satisfied.

The Leapfrog scheme implies no damping of the solution which corresponds to the physical advection processes. However, the Leapfrog scheme is memory expensive because the results of prior two time steps have to be kept during the solution process. In general, problems arise for centered differences if boundaries have to be approximated. To overcome this problem, whether fictional points have to be introduced or lower-order one-side schemes have to be applied.

6.3.4 Crank-Nicolson Scheme (CNS)

Algebraic Scheme

Applying the CNS (see Fig. 6.2 and eqn 6.2.4) to the linear advection equation we obtain

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + \frac{v}{2} \frac{u_{j+1}^n - u_{j-1}^n}{2 \Delta x} + \frac{v}{2} \frac{u_{j+1}^{n+1} - u_{j-1}^{n+1}}{2 \Delta x} = 0 \quad (6.77)$$

The implicit scheme produces a tridiagonal system of algebraic equations to determine the unknown values at new time

$$-\frac{1}{4}u_{j-1}^{n+1} + u_j^{n+1} - \frac{1}{4}u_{j+1}^{n+1} = -\frac{1}{4}u_{j-1}^n + u_j^n - \frac{1}{4}u_{j+1}^n \quad (6.78)$$

Numerical Properties: The CNS is consistent with the PDE and has a truncation error of $O(\Delta t, \Delta x^2)$. The CNS is unconditionally stable.

6.3.5 Euler-Taylor Scheme

Donea (1984) proposed a generalized Lax-Wendroff scheme (6.3.3) for the construction of higher-order numerical solutions for advection-dominated problems. In this method a more accurate representation of the time derivative is used based on a Taylor series expansion.

$$\left[\frac{\partial u}{\partial t} \right]_i^n = \frac{u^{n+1} - u^n}{\Delta t} - \frac{\Delta t}{2} \left[\frac{\partial^2 u}{\partial t^2} \right]_i^n - \frac{\Delta t^2}{6} \left[\frac{\partial^3 u}{\partial t^3} \right]_i^n + O(\Delta t^3) \quad (6.79)$$

Using the above auxiliary relationship (6.68) the following semi-discrete scheme for the advection equation can be derived.

$$(1 - \frac{u^2 \Delta t^2}{6} \frac{\partial^2}{\partial x^2}) \frac{u^{n+1} - u^n}{\Delta t} + v \frac{\partial u}{\partial x} - \frac{v^2 \Delta t}{2} \frac{\partial^2 u}{\partial x^2} = 0 \quad (6.80)$$

The truncation error of the above scheme is $O(\Delta t^3, \Delta x^3)$. For stability the CFL condition has to be satisfied for this explicit scheme. The Euler-Taylor scheme can be also applied to finite elements.

6.3.6 Numerical Dispersion and Dissipation

As reported in section 4 hyperbolic PDEs play an important role in fluid dynamics. The underlying physic is connected with propagation processes. Pure advection processes, as described by eqn (6.54), contain no dissipation terms. Appropriate numerical algorithms must be able to conserve amplitudes of transported quantities and the speed of propagation. However, many schemes suffer from artificial (non-physical) dispersion (second order spatial derivatives) and/or dissipation (third order spatial derivatives) and from disturbance of propagation speeds. Those artifacts are denoted as numerical diffusion/dispersion and numerical dissipation.

As an example we discuss wave propagation with constant speed and without any damping (dissipation). This process can be described by the advection equation. To investigate numerical dispersion and dissipation we compare Fourier representations for the exact and the approximate solutions. The Fourier series representation for the exact solution of the PDE is

$$u(t, x) = \sum_{k=-\infty}^{\infty} u_k e^{ik(x-vt)} \quad (6.81)$$

where k is the wave number and v is the propagation speed of the wave. Note, the Fourier representation is a kind of separation of variables. In general, the approximate solution will suffer from numerical dispersion or dissipation

$$\hat{u}(t, x) = \sum_{k=-\infty}^{\infty} u_k e^{-\hat{p}(k)t} e^{ik[x-\hat{v}(k)t]} \quad (6.82)$$

where $\hat{p}(k)$ is the artificial dissipation parameter that determines how rapidly the amplitude of the wave attenuates and $\hat{v}(k)$ is the erroneous propagation speed of the wave. Both parameters depend on the wave number. As a simplification, terms of the Fourier series representation can be considered independently since the governing partial and algebraic equations are linear. Moreover, all wave components will propagate with identical speed. Therefore, analysis can be conducted for each term separately.

To evaluate dissipation of the upwind scheme (6.64), we determine the amplification factor of the k -th wave length

$$\begin{aligned} G_k &= \frac{u_j^{n+1}}{u_j^n} = (1 - Cr) + Cr \frac{u_{j-1}^n}{u_j^n} \\ &= e^{-(\hat{p} + ik\hat{v})\Delta t} = 1 - Cr + Cr e^{-ik\Delta x} \\ &= 1 + Cr[\cos(k\Delta x) - 1] - iCr \sin(k\Delta x) \end{aligned} \quad (6.83)$$

and finally

$$|G_k| = e^{-\hat{p}\Delta t} = 1 - 4Cr(1 - Cr) \sin^2\left(\frac{k\Delta x}{2}\right) \quad (6.84)$$

Therefore, dissipation of the numerical scheme depends both on Courant number and on wave-number parameter. Largest dissipation results from $k\Delta x = \pi$, which corresponds to the shortest wave lengths. Note, we obtained the relationship to evaluate dissipation of the scheme in fact by a von Neumann stability analysis. A stable solution requires $\hat{p} \geq 0$.

Now we evaluate the wave propagation speed. From the above equation we have

$$-k\hat{v}\Delta t = \arctan \frac{-Cr \sin(k\Delta x)}{1 + Cr[\cos(k\Delta x) - 1]} \quad (6.85)$$

Therefore, error in propagation speed grows as $k\Delta x \rightarrow \pi$. Consistency analysis provides also inside to numerical dispersion and dissipation.

For the above investigation we used the basic formulas

$$\sin z = \frac{1}{2i}(e^{iz} - e^{-iz}) \quad , \quad \cos z = \frac{1}{2}(e^{iz} + e^{-iz}) \quad (6.86)$$

6.4 Transport Equation

In this section we examine linear transport problems where a scalar quantity (e.g. temperature or mass) is being advected with a known velocity and being diffused. The 1-D transport equation can be written as

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} - \alpha \frac{\partial^2 u}{\partial x^2} = 0 \quad (6.87)$$

The advection term in the above transport equation is linear if the velocity field is known. The transport equation has mixed character. In general it is a parabolic equation. However, when the ratio v/α becomes large the first two terms would dominate the equation and change its character to a hyperbolic one.

In the previous sections we considered already diffusion and advection equations and how they can be best handled, computationally. We will see that simultaneous appearance of advection and diffusion terms implies new difficulties. Care must be taken in particular to the advection term. Use of upwind schemes can result in spurious artificial higher-order derivative terms that may be of comparable magnitude to the physical important terms.

6.4.1 Steady Transport Equation

Exact Solution

Steady transport problems are e.g. dissipation processes in boundary layer flows. The linear steady transport equation is written as

$$v \frac{\partial u}{\partial x} - \alpha \frac{\partial^2 u}{\partial x^2} = 0 \quad (6.88)$$

In fact the above equation is an ordinary partial equation (OPE). For connected Dirichlet boundary conditions

$$u(0) = 0 \quad , \quad u(1) = 1 \quad (6.89)$$

the exact solution is given by

$$u(x) = \frac{e^{vx/\alpha} - 1}{e^{v/\alpha} - 1} \quad (6.90)$$

Algebraic Scheme

Using finite differences for the substitution of spatial derivatives (6.6, 6.12) the following algebraic equation can be derived

$$v \left[\gamma \frac{u_i - u_{i-1}}{\Delta x} + (1 - \gamma) \frac{u_{i+1} - u_i}{\Delta x} \right] - \alpha \frac{u_{i-1} - 2u_i + u_{i+1}}{\Delta x^2} = 0 \quad (6.91)$$

where γ is the upwind parameter which is a weighting parameter defining the specific finite difference scheme.

Upwind parameter	FD Scheme
$\gamma = 0$	Forward differences
$\gamma = 0.5$	Centered differences
$\gamma = 1$	Backward differences

By multiplying with Δx^2 and dividing by α the above algebraic equation can be rewritten as

$$[1 + \gamma Pg]u_{i-1} - [2 + (2\gamma - 1)Pg]u_i + [1 - (1 - \gamma)Pg]u_{i+1} = 0 \quad (6.92)$$

where Pg is the grid Peclet number (or cell Reynolds number R_{cell}).

$$P_g = \frac{v \Delta x}{\alpha} \quad (6.93)$$

In the case of centered differences we have

$$(1 + \frac{Pg}{2})u_{i-1} - 2u_i + (1 - \frac{Pg}{2})u_{i+1} = 0 \quad (6.94)$$

The above algebraic equation can be solved directly. For each node we obtain the following expression

$$u_i = A + B \left[\frac{1 + \frac{Pg}{2}}{1 - \frac{Pg}{2}} \right]^i \quad (6.95)$$

where A and B are chosen to satisfy the boundary conditions. In Fig. (6.4) the exact solution of the PDE as well as solutions of the algebraic equation above are compared. In particular we are interested in the discretization effect. As can be seen from the graphic, grid Peclet numbers smaller than 2 provide good agreement.

$$Pg < 2 \quad (6.96)$$

If $Pg = 2$ the solution is correct except adjacent to the boundary condition $x = 1$. For grid Peclet numbers larger than 2 the solution becomes oscillatory.

Consistency Analysis

Using Taylor series expansion for nodal values and substituting into the algebraic equation we obtain the following expression for the i -th node

$$v \left[\frac{\partial u}{\partial x} \right]_i - \alpha \left[\frac{\partial^2 u}{\partial x^2} \right]_i + E(x_i) = 0 \quad (6.97)$$

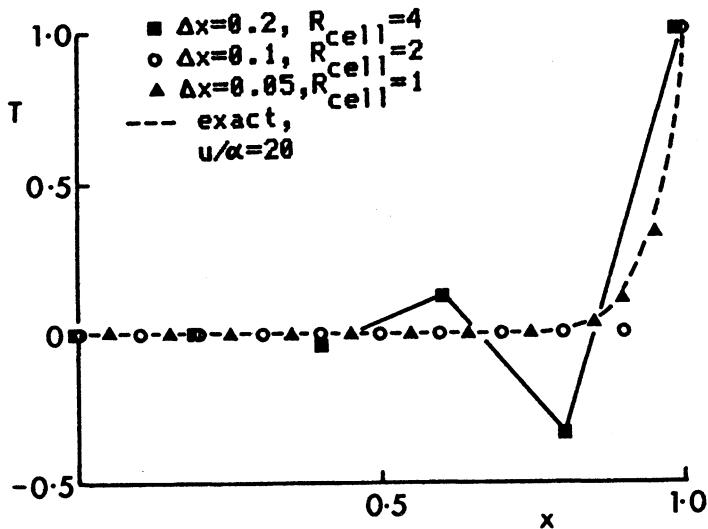


Figure 6.4: Cell Reynolds number influence on solution of steady transport equation (Fletcher 1990)

with a truncation error

$$E(x_i) = \left(\frac{1}{2} - \gamma\right)\Delta x v \left[\frac{\partial^2 u}{\partial x^2}\right]_i + \frac{\Delta x^2}{6} v \left[\frac{\partial^3 u}{\partial x^3}\right]_i + O(\Delta x^4) = 0 \quad (6.98)$$

From this relation we see that one-side differences lead to a first-order accurate scheme whereas central differences ($\gamma = 0.5$) provide second-order accuracy in space.

Using backward (upwind) differences for the first-order derivative, the algebraic equation can be rewritten in following form

$$v \left[\frac{\partial u}{\partial x}\right]_i - \alpha(1 + \frac{Pg}{2}) \left[\frac{\partial^2 u}{\partial x^2}\right]_i + O(\Delta x^2) = 0 \quad (6.99)$$

i.e. an artificial diffusivity of $\alpha Pg/2$ has been introduced by this scheme. Therefore, for accuracy it is desirable to keep Peclet numbers as small as possible.

Stability Analysis

As there is an exact solution of the algebraic scheme available, stability analysis can be conducted very conveniently. The exact solution for central differences (eqn 6.95) implies already that the solution becomes oscillatory if $Pg > 2$. By

we use of the matrix method we can write the SAE including the boundary conditions in the following form

$$\mathbf{A}\mathbf{u} = \mathbf{b} \quad (6.100)$$

where \mathbf{A} is again a tridiagonal matrix and vector \mathbf{b} contains the boundary conditions. For the case of central differences ($\gamma = 0.5$), eqn (6.94), we have

$$\mathbf{A} = \begin{bmatrix} -2 & 1 - \frac{Pg}{2} & & & \\ 1 + \frac{Pg}{2} & -2 & 1 - \frac{Pg}{2} & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 1 - \frac{Pg}{2} \\ & & & 1 + \frac{Pg}{2} & -2 \end{bmatrix} \quad (6.101)$$

$$\mathbf{b} = \begin{bmatrix} (-1 - \frac{Pg}{2})u_1 \\ 0 \\ \dots \\ 0 \\ (-1 + \frac{Pg}{2})u_{np} \end{bmatrix} \quad (6.102)$$

The eigenvalues λ of the above tridiagonal matrix are determined from matrix equation

$$|\mathbf{A} - \lambda \mathbf{E}| = 0 \quad (6.103)$$

The eigenvalues are

$$\lambda_j = -2 + 2\sqrt{(1 - Pg/2)(1 + Pg/2)} \cos \frac{j\pi}{np - 1} \quad , \quad j = 1, 2, \dots, np - 2 \quad (6.104)$$

It is note worthy that the condition for existence of real eigenvalues coincides with the stability condition (6.96).

$$(1 - Pg/2)(1 + Pg/2) \geq 0 \implies \boxed{Pg \leq 2} \quad (6.105)$$

The stability condition (6.96) is also fulfilled if matrix \mathbf{A} is diagonal dominant. In addition to matrix or von Neumann methods, investigation of spectral radii of eigenvalues can be used for stability analysis (Fletcher 1990, p 295).

Using upwind differences ($\gamma = 1$) for the first-order derivative, the algebraic equation becomes

$$(1 + Pg)u_{i-1} - (2 + Pg)u_i + u_{i+1} = 0 \quad (6.106)$$

The corresponding system matrix \mathbf{A} has real eigenvalues for all values of Peclet number. Therefore, the scheme is stable. However from consistency analysis we know that upwind differences introduce additional artificial diffusion terms which may result in not very accurate solutions in particular for large Peclet numbers.

6.4.2 Explicit Schemes for the Linear Transport Equation

Algebraic Scheme

Use of FTCS algorithm to the linear transport equation (6.87) yields the following algebraic scheme.

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + v \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} - \alpha \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} = 0 \quad (6.107)$$

For determination of the unknown values at new time level we obtain from the above equation

$$u_j^{n+1} = (Ne + \frac{Cr}{2})u_{j-1}^n + (1 - 2Ne)u_j^n + (Ne - \frac{Cr}{2})u_{j+1}^n \quad (6.108)$$

Consistency - The FTCS scheme is consistent with the PDE in the order of $O(\Delta t, \Delta x^2)$.

Stability - From the above studies we understand, on the one hand, FTCS schemes are conditionally stable for the diffusion equation (see 6.2), on the other hand, FTCS schemes are always unstable for the advection equation (see 6.3). Noye (1983) developed the following stability condition for the transport equation, i.e. a parabolic-hyperbolic equation.

$$0 \leq Cr^2 \leq 2Ne \leq 1 \quad (6.109)$$

6.4.3 Implicit Schemes for the Linear Transport Equation

Algebraic Scheme

As an example of an implicit scheme, we consider the Crank-Nicolson scheme, which yields the following algebraic equations.

$$\begin{aligned} \frac{u_j^{n+1} - u_j^n}{\Delta t} &+ \frac{v}{2} \left(\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} + \frac{u_{j+1}^{n+1} - u_{j-1}^{n+1}}{2\Delta x} \right) \\ &- \frac{\alpha}{2} \left(\frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} + \frac{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}{\Delta x^2} \right) = 0 \end{aligned} \quad (6.110)$$

By rearranging the above equation, we obtain the following expression for the unknown values at new time level.

$$\begin{aligned} & -(Ne + \frac{Pg}{2})u_{j-1}^{n+1} + 2(1 + 2Ne)u_j^{n+1} - (Ne - \frac{Pg}{2})u_{j+1}^{n+1} \\ & = (Ne + \frac{Pg}{2})u_{j-1}^n + 2(1 - 2Ne)u_j^n + (Ne - \frac{Pg}{2})u_{j+1}^n \end{aligned} \quad (6.111)$$

Consistency - The CNS is consistent with the PDE in order of $O(\Delta t^2, \Delta x^2)$. Therefore, we have no artificial diffusion arising from first-order truncation errors.

Stability - The CNS is unconditionally stable. To prevent spatial oscillations, the Peclet criterion $Pg \leq 2$ should be fulfilled.

6.5 Burgers Equation

In the previous section we considered linear transport processes. In contrast to the transport equations for mass or heat migration, the advection term in the momentum transport equations is non-linear. In the following we analyze the Burgers equation, which has a non-linear advection term. It is the same form of the non-linear advection term as the Navier-Stokes equation for incompressible flow. Burgers equation is our favor because there are exact solutions for several combinations of initial and boundary conditions. The one-dimensional Burgers equation is given by

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} - \nu \frac{\partial^2 u}{\partial x^2} = 0 \quad (6.112)$$

where u is the velocity and ν is kinematic viscosity (in contrast to the transport equation diffusivity now has been replaced by kinematic viscosity).

To examine the impact of non-linear terms we consider the inviscid Burgers equation

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0 \quad (6.113)$$

which is obtained if the viscous term is dropped. Note it is similar to the pure advection equation (6.54). From the non-linear term it is obvious that points with larger velocity advect faster than points with smaller velocities. This can result in the development of non-unique solutions. To have a unique solution it is necessary to assume a shock across which the velocity changes discontinuously. The viscous term in Burgers' equation (6.112) at first reduces the amplitude of the wave as time proceeds and at second prevents non-unique (non-physically) solutions (Fig. 6.5).

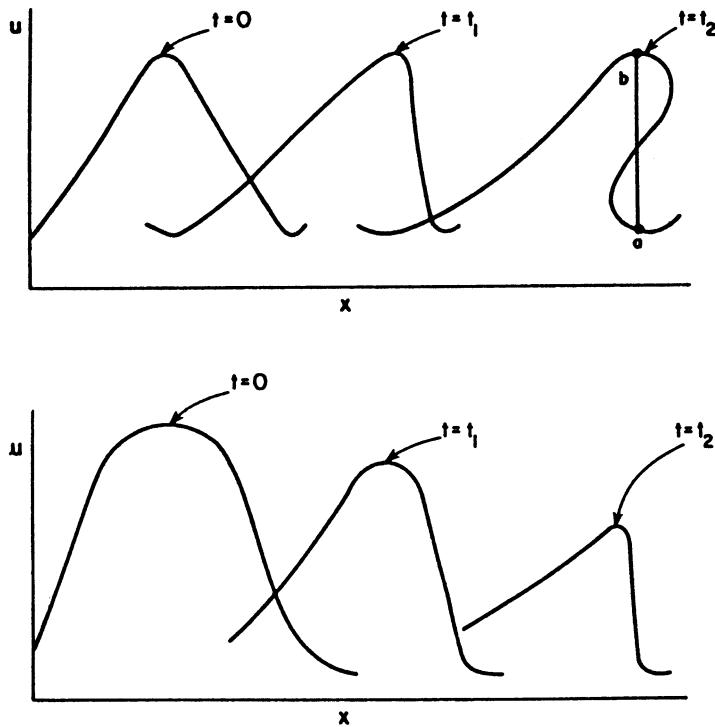


Figure 6.5: Formation of multi-valued solutions for the inviscid Burgers equation (Fletcher 1990)

Burgers equation is a very suitable test case for the numerical computation of non-linear advection-dominated problems. The Burgers equation contains the same form of convective non-linearity as in many of the fluid dynamics governing equations. The Cole-Hopf transformation to diffusion-like equation allows exact solutions for many combinations of initial and boundary conditions (Hopf 1950, Cole 1951).

The key difference in comparison to the ordinary transport equation discussed in (6.4) is the handling of the non-linear advection term. There is an alternative procedure, rewriting equation (6.112) in divergence or conservation form.

$$\frac{\partial u}{\partial t} + \frac{\partial F}{\partial x} - \nu \frac{\partial^2 u}{\partial x^2} = 0 \quad (6.114)$$

with $F = u^2/2$

6.5.1 Explicit Schemes

Algebraic Scheme

Applying the well-known FTCS scheme to Burgers equation we obtain

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + u_j^n \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} - \nu \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} = 0 \quad (6.115)$$

Resolving the above equation with respect to the unknown values at new time level we have

$$\begin{aligned} u_j^{n+1} &= (Ne + \frac{Cr_j^n}{2})u_{j-1}^n + (1 - 2Ne)u_j^n + (Ne - \frac{Cr_j^n}{2})u_{j+1}^n \\ &= (Ne + \frac{u_j^n \Delta t}{2\Delta x})u_{j-1}^n + (1 - 2Ne)u_j^n + (Ne - \frac{u_j^n \Delta t}{2\Delta x})u_{j+1}^n \end{aligned} \quad (6.116)$$

with $Cr_j^n = u_j^n \Delta t / \Delta x$. The FTCS representation of divergence form (6.114) of Burgers equation is obviously

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + \frac{F_{j+1}^n - F_{j-1}^n}{2\Delta x} - \nu \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} = 0 \quad (6.117)$$

and therefore

$$\begin{aligned} u_j^{n+1} &= Ne u_{j-1}^n + \frac{\Delta t}{2\Delta x} F_{j-1}^n - 2Ne u_j^n + Ne u_{j+1}^n - \frac{\Delta t}{2\Delta x} F_{j+1}^n \\ &= (Ne + \frac{\Delta t}{4\Delta x} u_{j-1}^n)u_{j-1}^n - 2Ne u_j^n + (Ne - \frac{\Delta t}{4\Delta x} u_{j+1}^n)u_{j+1}^n \end{aligned} \quad (6.118)$$

Obviously, both forms of Burgers equations result in different discrete representations.

The Lax-Wendroff scheme (6.3.3) can be applied to improve the consistency of the discrete formulation. To this purpose we replace the second time derivative with an equivalent spatial one.

$$\frac{\partial F}{\partial t} = \frac{\partial u}{\partial t} \frac{\partial F}{\partial u} = J \frac{\partial u}{\partial t} = -J \frac{\partial F}{\partial x} \quad (6.119)$$

$$\frac{\partial}{\partial t} \frac{\partial u}{\partial t} = -\frac{\partial}{\partial t} \frac{\partial F}{\partial x} = -\frac{\partial}{\partial x} \frac{\partial F}{\partial t} = \frac{\partial}{\partial x} \left(J \frac{\partial F}{\partial x} \right) \quad (6.120)$$

with the Jacobian $J = \partial F / \partial u$. As an example, the Lax-Wendroff scheme applied to the inviscid Burgers equation (6.113) gives

$$\begin{aligned} u_j^{n+1} = u_j^n - \frac{\Delta t}{2\Delta x} (F_{j+1}^n - F_{j-1}^n) &+ \frac{(\Delta t)^2}{2(\Delta x)^2} J_{j+1/2} (F_{j+1}^n - F_j^n) \\ &- J_{j-1/2} (F_j^n - F_{j-1}^n) \end{aligned} \quad (6.121)$$

with $J_{j+1/2} = u_{j+1/2} = 1/2(u_j + u_{j+1})$ and $J_{j-1/2} = u_{j-1/2} = 1/2(u_j + u_{j-1})$. Note, the appearance of the Jacobian at half steps makes the computational scheme less economically.

Numerical properties: The algebraic scheme is consistent with the PDE in order of $O(\Delta t^2, \Delta x^2)$. For stability the CFL condition $Cr \leq 1$ has to be fulfilled.

6.5.2 Implicit Schemes

Algebraic Scheme

The Crank-Nicolson scheme applied to the divergence form of Burgers equation yields the following algebraic expression.

$$\begin{aligned} \frac{u_j^{n+1} - u_j^n}{\Delta t} &+ \frac{1}{2} \frac{F_{j+1}^n - F_{j-1}^n}{2\Delta x} + \frac{1}{2} \frac{F_{j+1}^{n+1} - F_{j-1}^{n+1}}{2\Delta x} \\ &- \nu \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{2\Delta x^2} - \nu \frac{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}{2\Delta x^2} = 0 \end{aligned} \quad (6.122)$$

Appearance of F_j at new time level poses a problem for the solution process. We use the TSE to eliminate those expressions.

$$\begin{aligned} F_j^{n+1} &= F_j^n + \Delta t \left[\frac{\partial F}{\partial t} \right]_j^n + \frac{\Delta t^2}{2} \left[\frac{\partial^2 F}{\partial t^2} \right]_j^n + O(\Delta t^3) \\ &= F_j^n + \Delta t \left[\frac{\partial F}{\partial u} \right]_j^n \left[\frac{\partial u}{\partial t} \right]_j^n + O(\Delta t^2) \\ &= F_j^n + \Delta t J_j^n \frac{u_j^{n+1} - u_j^n}{\Delta t} + O(\Delta t^2) \end{aligned} \quad (6.123)$$

Substituting the above TSE into the algebraic CNS scheme (6.123) we yield the following linearized formulation.

$$\begin{aligned} &\left(\frac{1}{2} Cr_{j-1}^n - Ne \right) u_{j-1}^{n+1} + (2 + Ne) u_j^{n+1} - \left(\frac{1}{2} Cr_{j+1}^n - Ne \right) u_{j+1}^{n+1} \\ &= -\left(\frac{1}{4} Cr_{j-1}^n - Ne \right) u_{j-1}^n + 2(1 - Ne) u_j^n + \left(\frac{1}{4} Cr_{j+1}^n + Ne \right) u_{j+1}^n \end{aligned} \quad (6.124)$$

with the local, time depending Courant number $Cr_j^n = u_j^n \Delta t / \Delta x$. Note J_j^n is cancelled out, which improves the computational efficiency of this scheme substantially.

It is important to point out, the linearization process introduces new truncation error by the TSE.

Numerical properties: The algebraic scheme is consistent with the PDE in order of $O(\Delta t^2, \Delta x^2)$. The implicit scheme is unconditionally stable.

6.6 Problems

Taylor Series Expansion

- 1 Develop the function $u(t, x)$ in x at nearby node x_i using the Taylor series expansion.
- 2 Use the Taylor series expansion for first order derivatives in time and space of function $u(i, x)$.

Diffusion Equation

- 3 Write the diffusion equation in one dimension.
- 4 Examine the consistency of the explicit FTCS scheme (6.17) for the diffusion equation (6.13). Determine the truncation error (6.20) by substituting the Taylor series expansion into the PDE.
- 5 Determine the stability bounds (Neumann criterion) of the explicit FTCS scheme (6.17) for the diffusion equation (6.13) using the matrix method.
- 6 Determine the stability bounds of the explicit FTCS scheme (6.17) for the diffusion equation (6.13) using the von Neumann method.
- 7 Determine the stability bounds of the semi-implicit Crank-Nicolson scheme (6.46) for the diffusion equation (6.13) using the von Neumann method.
- 8 Extend the explicit FTCS scheme (6.17) for the diffusion equation (6.13) in two-dimensions.

Advection Equation

- 9 Write the advection equation in one dimension.
- 10 Determine the stability bounds of the explicit FTCS scheme (6.17) for the advection equation (6.54) using the von Neumann method.
- 11 Examine the consistency of the upwind scheme (6.64) for the advection equation (6.54). Determine the truncation error by substituting the Taylor series expansion into the PDE.
- 12 Derive eqn (6.68) from the advection equation (6.54).
- 13 Determine the stability bounds of the upwind scheme (6.64) for the advection equation (6.54) using the von Neumann method. What is the name of this stability criterion ?

- 14** Derive the Courant-Friedrich-Lowy (CFL) condition of the Crank-Nicolson-Scheme (CNS) for the advection equation (6.54) using the von Neumann method.
- 15** Extend the Crank-Nicolson-Scheme (CNS) (6.77) for the advection equation (6.54) in two-dimensions.

Transport Equation

- 16** Write the transport equation in one dimension.
- 17** Check the exact solution (6.90) by substituting into the steady transport equation (6.88).
- 18** Examine the consistency of the general scheme (6.91) for the steady transport equation (6.88). Determine the truncation error by substituting the Taylor series expansion into the PDE.
- 19** Determine the stability bounds (Peclet criterion) of the general scheme (6.91) for the steady transport equation (6.88) using the matrix method.
- 20** Explain the term nonlinearity of a PDE using the Burgers equation.
- 21** Explain the difference between explicit and implicit schemes.

Chapter 7

Finite Element Method

The Finite Element Method (FEM) was originated from the field of structural calculation (e.g. stress analysis) in the beginning of the fifties. The terminus - **finite elements** - was introduced by Turner et al. (1956). The concept of the finite element technique consists in a subdivision of a complex structure into small substructures assembling the elements. After its initial development as an engineering tool mathematicians worked out a rigorous formal framework of this method, e.g. concerning consistence of solutions, criteria for numerical stability and convergence behavior as well as accuracy and error bounds. The mathematical background of FEM is functional analysis. The FEM was introduced into the field of computational fluid dynamics (CFD) by Chung (1978), Baker (1983), Huyakorn & Pinder (1983), Diersch (1985) and others. The FEM is a more general approximation technique containing many finite difference schemes as special cases (Chapter 6).

The basic steps in the finite element approximation are:

- Space discretization by finite elements (meshing) (Chapter 7.1)
- Discretization of the integral (weak) formulations of the governing partial differential equations (Chapter 7.2)

$$L(u) \rightarrow \hat{L}(\hat{u}) \quad (7.1)$$

- Generation of interpolation functions to approximate the unknown values of field variables (Chapter 7.3)

$$\hat{u} = f(\hat{u}_1, \dots, \hat{u}_n) \quad (7.2)$$

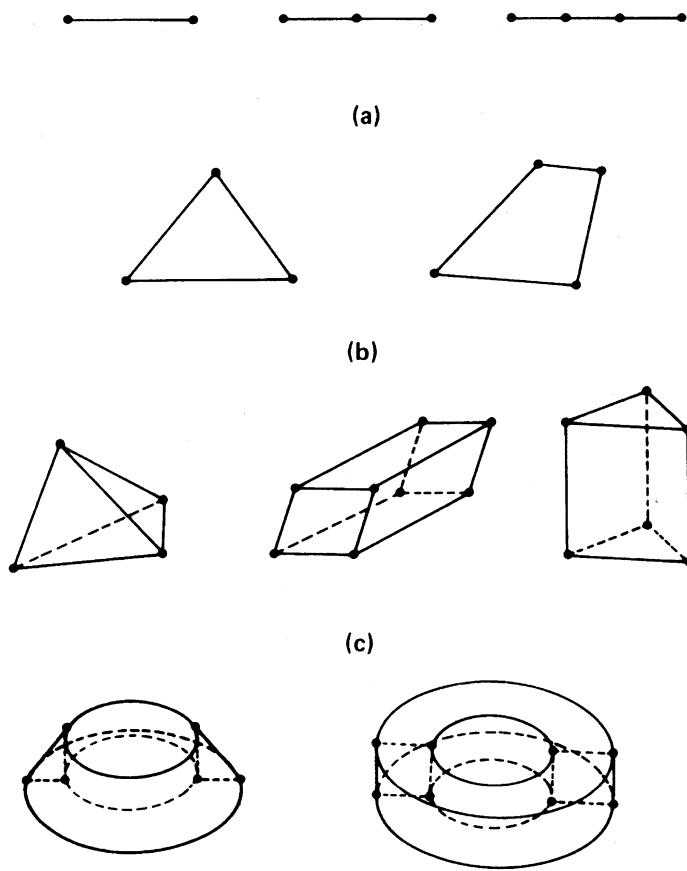


Figure 7.1: Basic element types (Huyakorn & Pinder, 1983)

7.1 Domain Discretization

The basic concept of the finite element method is the subdivision of the computational domain into elements of arbitrary shape and size. The only restriction is that the elements may not overlap and that they have to cover the complete computational domain. The geometric object is finally represented by a mesh of 1-D, 2-D and/or 3-D basic elements (geometric units) such as bars, triangles, quadrilaterals, tetrahedrons, hexahedra and pyramids (Fig. 7.1). Which type of element is most appropriate for a particular problem depends on several factors, such as domain geometry, required accuracy, computational costs etc.). The ability to handle non-uniform and distorted computational domains has always been an important feature of the finite element method. Figures 7.2 and 7.3 show finite element discretizations for fracture networks which are used in

practical computations for fluid flow and mass transport in crystalline rock. In particular, automatic grid adaptation is employed. Within each finite element a certain number of grid points is defined, which are located along the sides/faces of the element. At these points the unknown functions have to be determined. The total number of unknowns (function values, function derivatives) is denoted as the **degree of freedom** of the discrete system.

In general, finite elements are required to have convex shapes. This condition should avoid singularities of the Jacobian, to ensure unique transformation of coordinates.

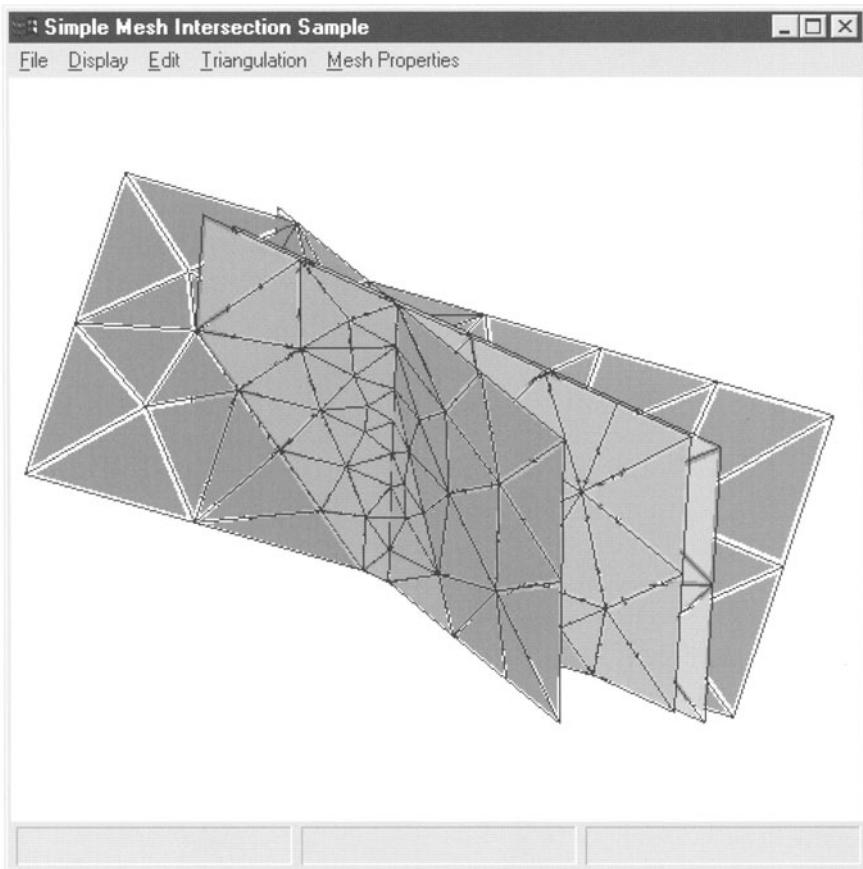


Figure 7.2: Finite element mesh for a fracture system (Kasper et al. 1998)

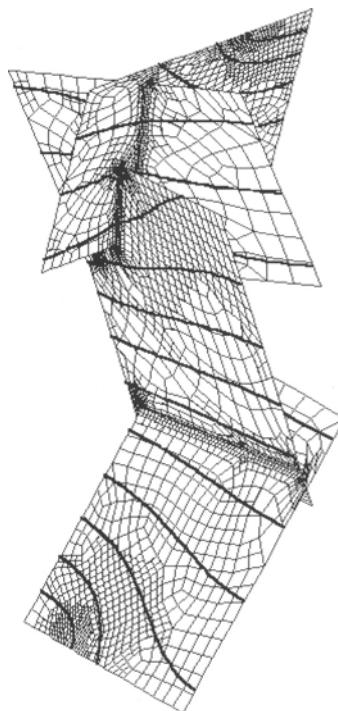


Figure 7.3: Adaptive finite element mesh for a fracture system (Kaiser et al. 1998)

7.2 Equation Discretization

The FEM requires the definition of an integral (weak) formulation of the continuum-mechanics field equations describing the physical problem. In particular, integral formulations are required in situations where discontinuous solutions are possible. In this case, derivatives of the discontinuous quantity are not defined, and, therefore, the integral formulation is the only physical meaningful problem description.

In general, there are two ways to derive integral formulations for continuum problems:

- Variational principles (Chapter 7.2.1)
- Method of weighted residuals (MWR) (Chapter 7.2.2)

A variational principle means that the physical problem can be expressed by the extremum of a functional. Although many physical problems can be described by variational equations (e.g. potential flow), there is no guarantee to find a corresponding variational principle for all problems (e.g. for flow governed by the Navier-Stokes equations). On the other hand, construction of an equivalent integral formulation, i.e. a weak formulation of the governing differential equations, based on the method of weighted residuals is always possible.

Note, that integral formulations have restricted continuity properties of the unknown functions under consideration. Therefore, it is particularly important to define appropriate criteria in order to derive meaningful convergence and error bounds.

7.2.1 Variational Principles

By use of a variational principle the PDE of the original problem is replaced by an equivalent variational problem, which consists in finding a function that maximizes a certain integral quantity, a functional - i.e. a function of an unknown function. A general expression for those functionals is given by the Euler-Lagrange equation

$$F = \int_{\Omega} f(x, u, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}, \dots) dx \quad (7.3)$$

The maximum condition of this functional must reveal again the governing PDE and the boundary condition of the original physical problem. To obtain the maximum condition the calculus of variation is used.

Rayleigh-Ritz method

To obtain an approximate solution for a variational problem the Rayleigh-Ritz method can be used. There it is assumed, the unknown function within the whole domain Ω can be represented by following trial solution

$$\hat{u}(x) = \sum_{i=1}^m \phi_i(x) C_i \quad (7.4)$$

where $\phi_i(x)$ are linearly **basis functions** satisfying the boundary conditions, C_i are unknown parameters, m is the number of finite the series terms (grid points).

Therefore, the Euler-Lagrange equation (7.3) becomes

$$F(C_1, \dots, C_m) = 0 \quad (7.5)$$

For maximization it has to be

$$\frac{\partial F}{\partial C_i} = 0 \quad , \quad i = 1, \dots, m \quad (7.6)$$

which yields a set of algebraic equations for the determination of C_i . Unfortunately for many practical problems, variational principles does not exist.

The main difference between the Rayleigh-Ritz method (RRM) and the finite element method lies in the definition of the basis functions. For FEM, these are element-related functions, whereas for RRM these are valid for the whole domain and have to fit the boundary conditions. The Rayleigh-Ritz method for homogeneous boundary conditions leads to the same discretized equations as the Galerkin method of weighted residuals.

7.2.2 Method of Weighted Residuals

In general the method of weighted residuals (MWR) can be divided into 3 steps:

1. Approximation of the unknown function by a trial solution (kind of approximation),
2. Definition of weighting functions,
3. Derivation of a system of algebraic equations (rules for determination of nodal values).

Approximate solution

Again we assume, the unknown function within the whole domain Ω can be represented by a trial solution \hat{u}

$$u \approx \hat{u}(x) = \sum_{i=1}^m \phi_i(x) C_i \quad (7.7)$$

where $\phi_i(x)$ are linearly **basis functions** or **interpolation functions** satisfying the boundary conditions, C_i are unknown parameters, m is the number of finite the series terms (grid points).

If an approximate solution \hat{u} is substituted into the original differential equation $L(u) = 0$ it will not identically fulfill this.

$$R(t, x) = L(u) - L(\hat{u}) = L(\hat{u}) \neq 0 \quad (7.8)$$

where R is the **residual**, L is the differential operator, u is the exact solution, and \hat{u} is the approximate solution. The residual is a measure of the accuracy of approximation or of the error introduced by the discretization method. Since the error in general can not be made to vanish simultaneously in all mesh points, an approximate solution can be derived by requiring that some weighted average of the nodal residuals should vanish over the computational domain.

$$\langle \omega, R \rangle = \int_{\Omega} \omega_i R(t, x) d\Omega = \int_{\Omega} \omega_i L(\hat{u}) d\Omega = 0 \quad (7.9)$$

where ω_i are **weighting functions** or test functions. Now, the unknown values C_i have to be defined in such a way, that the error (i.e. weighted residuals) is minimal.

From the above equations a system of equations can be obtained for determination of the unknown nodal values of the field variable u . For steady PDEs this will be a system of algebraic equations and for unsteady PDEs a system of ordinary differential equations will result. Note, the above equation is closely related to the integral form of the governing equations. The weak formulation allows discontinuities in the solution to appear. The MWR expresses the condition that the projection of the residual in the subspace of weighting functions is zero, i.e. the residual is orthogonal to the subspace. An approximate solution algorithm will converge if R tends to zero for increasing number of weighting functions and nodes (i.e. degree of freedom).

The MWR is a more general procedure to obtain approximate solutions. Different choices of the weighting functions and corresponding algorithms to minimize the residual result in several methods.

Table 7.1: Captured methods for several weighting functions

Method	Weighting function	Remarks
Subdomain method	$\omega_i(x) = \begin{cases} 1 & : x \in \Omega_i \\ 0 & : x \notin \Omega_i \end{cases}$	FVM (conservation)
Collocation method	$\omega_i(x) = \delta(x - x_i)$	FDM
Least-square method	$\omega_i(x) = \partial R / \partial C_i$	Minimization problem
Galerkin method	$\omega_i(x) = \phi_i(x)$	FEM

7.2.3 Galerkin Method

The most popular scheme of MWR is the Bubnov-Galerkin method in which weighting and interpolation functions are selected identically $\omega = \phi$.

Example: General conservation law

As a first example we consider the general conservation law in differential form (see section 1.1, equation (1.25)).

$$\frac{\partial u}{\partial t} + \nabla \cdot \Phi^u = \frac{\partial u}{\partial t} + \frac{\partial \Phi_\alpha^u}{\partial x_\alpha} = q^u \quad (7.10)$$

where Φ^u is the flux vector of conservation quantity u and q^u is a source term. The corresponding weak form of the conservation equation in analogy to (10.3) due to the Galerkin method is

$$\int_{\Omega} \phi_i \frac{\partial u}{\partial t} d\Omega + \int_{\Omega} \phi_i \nabla \cdot \Phi^u d\Omega = \int_{\Omega} \phi_i q^u d\Omega \quad (7.11)$$

The main steps are now: integration by parts of spatial derivation terms and use of Green's theorem in order to convert volume to surface integrals. After these steps we obtain

$$\int_{\Omega} \phi_i \frac{\partial u}{\partial t} d\Omega - \int_{\Omega} \Phi^u \cdot \nabla \phi_i d\Omega + \oint_{\partial\Omega} \phi_i \Phi^u \cdot d\mathbf{S} = \int_{\Omega} \phi_i q^u d\Omega \quad (7.12)$$

To simplify the notation, we omit the suffix u in the following. The finite element representations of conservation quantity and flux term are

$$\hat{u}(t, x) = \sum_j \phi_j(x) u_j(t) \quad , \quad \hat{\Phi} = \sum_j \phi_j(x) \Phi_j(t) \quad (7.13)$$

The discretized form of the general conservation equation is

$$\sum_i \underbrace{\frac{du_j}{dt} \int_{\Omega} \phi_i \phi_j d\Omega}_{M_{ij}} - \sum_i \underbrace{\Phi_j \cdot \int_{\Omega} \phi_j \nabla \phi_i d\Omega}_{K_{ij}} + \oint_{\partial\Omega} \phi_i \Phi^u \cdot d\mathbf{S} = \int_{\Omega} \phi_i q^u d\Omega \quad (7.14)$$

where Ω_j is the subdomain containing node j . The summations extends over all elements containing this node. By this means the original partial differential equation was transformed to an ordinary differential equation. In this representation we have the so-called **mass matrix**

$$M_{ij} = \int_{\Omega} \phi_i \phi_j d\Omega \quad (7.15)$$

which is a symmetric but not diagonal one and the so-called **stiffness matrix**

$$K_{ij} = \int_{\Omega} \phi_j \nabla \phi_i d\Omega \quad (7.16)$$

which is no longer symmetric. Note for linear problems, both mass and stiffness matrices are independent of the unknown function u . They will depend then only on the element geometry.

We have seen the application of the method of weighted residuals to differential balance equations leads to several matrix-integral expressions that must be evaluated for each finite element. To calculate these expressions we have to know the interpolation functions and their derivatives as functions of coordinate directions.

7.3 Interpolation and Shape Functions

The general idea is to approximate the unknown field variables by linear combinations of defined **interpolation functions** ϕ_i (also called basis or trial functions) and the unknown nodal values of the dependent variable u_i

$$u(t, x) \approx \hat{u}(t, x) = \sum_i u_i(t) \phi_i(x) \quad (7.17)$$

where $u_i(t) = \hat{u}(t, x_i)$ and the summation extends over all nodes of the finite element mesh. The interpolation functions are attached to each node and, therefore, corresponds to the degree of freedom of the discrete system.

As written above, here are several possibilities to define basis functions leading to different methods:

- Collocation or spectral methods: definition of interpolation functions on the whole computation domain (e.g. by trigonometric functions, Legendre or Chebyshev polynomials, splines)
- Standard FEM is based on local interpolation, i.e. interpolation functions are chosen to be locally defined polynomials within each element and being zero outside the considered element. Polynomials are popular because they are simple functions to manipulate mathematically.

7.3.1 General Method for Shape Function Derivation

The general method for derivation of shape functions consists of the following three steps:

1. Polynomial representation of the trial function

$$\hat{\mathbf{u}}(x, y, z) = \mathbf{P}^T(x, y, z) \mathbf{a} \quad (7.18)$$

with polynomial \mathbf{P} and general parameter vector \mathbf{a}

$$\mathbf{P}^T(x, y, z) = \{1, x, y, z, x^2, xy, xz, y^2, yz, z^2, \dots\} \quad , \quad \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_n \end{bmatrix} \quad (7.19)$$

For all grid points (i.e. nodal degrees of freedom) we have

$$\left\{ \begin{array}{c} u_1 \\ u_2 \\ \dots \\ u_m \end{array} \right\} = \left[\begin{array}{cccccccccc} 1 & x_1 & y_1 & z_1 & x_1^2 & x_1y_1 & x_1z_1 & y_1^2 & y_1z_1 & z_1^2 & \dots \\ 1 & x_2 & y_2 & z_2 & x_2^2 & x_2y_2 & x_2z_2 & y_2^2 & y_2z_2 & z_2^2 & \dots \\ \dots & \dots \\ 1 & x_m & y_m & z_m & x_m^2 & x_my_m & x_mz_m & y_m^2 & y_mz_m & z_m^2 & \dots \end{array} \right] \left\{ \begin{array}{c} a_1 \\ a_2 \\ \dots \\ a_n \end{array} \right\}$$

$$u_i = P_{ij} a_j \quad , \quad i = 1, \dots, m \quad j = 1, \dots, n$$

2. General parameter vector

Inversion of the above equation gives

$$a_i = P_{ij}^{-1} u_j \quad (7.20)$$

3. General shape function

Combining the above equations we obtain

$$\hat{\mathbf{u}}(x, y, z) = \underbrace{P_i^T(x, y, z) P_{ij}^{-1}}_{\mathbf{N}^T} u_j \quad (7.21)$$

7.3.2 Shape Function Conditions

The shape functions locally satisfy the following condition

$$\phi_i^{(e)}(x_i) = \begin{cases} 1 & : x = x_i \\ 0 & : x \in (e) \end{cases} \quad , \quad \phi_i^{(e)}(x_j) = \delta_{ij} \quad (7.22)$$

Additionally, the shape functions have to satisfy the following condition

$$\sum_i \phi_i^{(e)}(x) = 1, \quad \forall x \in (e) \quad (7.23)$$

The above equations define the properties of local polynomial interpolation functions for finite element approximations. Different classes of interpolation

functions can be constructed according to their inter-element continuity conditions. The most popular are Lagrangian and Hermitian elements. If nodal values of the dependent field variable are defined by the values of the unknown functions, C^0 -continuity at the inter-element boundary is sufficient. These elements, which are called **Lagrangian elements**, are appropriate to describe differential equations of order no higher than two. If continuity of first-order derivatives (i.e. C^1 -continuity) is required at the inter-element boundaries, we need so-called **Hermitian elements**. In this case, first-order derivatives of the unknown functions are to be considered as additional degrees of freedom of the discrete system.

In the following we show an approach to the construction of interpolation functions for various element types. Due to its specific properties for different element shapes those functions are denoted as **element shape functions**. For the construction of shape functions we have to consider factors such as function completeness, domain geometry, solution accuracy, and computation effort. In general, several function types are possible to this purpose (e.g. trigonometric functions). As noted above polynomials are our favor to represent unknown variables within finite elements.

7.3.3 Isoparametric Mapping

The ability to handle non-uniform and distorted computational domains is an important outline of the finite element method. This feature is based on specific mapping methods.

$$x(r, s, t) = \sum_i N_i(r, s, t) x_i \quad (7.24)$$

where (x, y, z) are physical (also denoted as global) coordinates, (r, s, t) are virtual (also denoted as natural or local) coordinates, and N_i are shape functions, describing the shape of the element. The summation extends over all nodes of the element. This mapping rule performs a geometric transformation of the element between physical and virtual spaces.

We distinguish between subparametric, isoparametric, and superparametric mapping in dependence of the degrees of shape and interpolation functions.

Subparametric mapping	Isoparametric mapping	Superparametric mapping
$\deg(N) < \deg(\phi)$	$\deg(N) = \deg(\phi)$	$\deg(N) > \deg(\phi)$

In next sections we see the use of parametric mapping has particular benefit for the evaluation of integral equations.

7.4 Element Shape Functions

7.4.1 1-D Linear Bar Elements

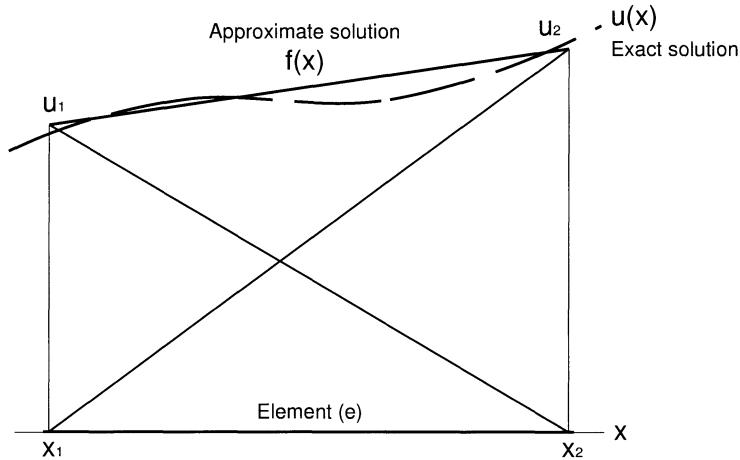


Figure 7.4: Approximate solution by use of 1-D linear shape functions

Shape functions

A simple approximation of the unknown function can be obtained by linear approximation.

$$\hat{u}(x) = a_1 + a_2 x \quad (7.25)$$

Due to the C^0 -continuity condition we have at the nodes

$$\begin{aligned} u_1 &= a_1 + a_2 x_1 \\ u_2 &= a_1 + a_2 x_2 \end{aligned} \quad (7.26)$$

or written compactly

$$\begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix} \quad (7.27)$$

By inversion of the above matrix equation we can derive relations for the coefficients

$$\begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix} = \frac{1}{x_2 - x_1} \begin{bmatrix} x_2 & -x_1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \quad (7.28)$$

Inserting the relations for the coefficients a_i into equation (7.25) this can be rearranged

$$\hat{u}(x) = \frac{x_2 - x}{x_2 - x_1} u_1 + \frac{x - x_1}{x_2 - x_1} u_2 = N_1(x)u_1 + N_2(x)u_2 \quad (7.29)$$

and rewritten as a trial solution. Therefore, we derived following element-related interpolation functions (i.e. shape functions) for a one-dimensional linear element fulfilling the conditions to shape functions.

$$N_1^{(e)}(x) = \frac{x_2 - x}{x_2 - x_1} \quad , \quad N_2^{(e)}(x) = \frac{x - x_1}{x_2 - x_1} \quad (7.30)$$

By this case we construct a piece-wise linear interpolation of the unknown function (Fig. 7.4).

In Fig. 7.5 we see two connected 1-D elements. Both elements have node i in common. The shape functions of these elements are defined in physical (global) coordinates by

$$\begin{aligned} N_1^{(1)}(x) &= \frac{x - x_1}{x_2 - x_1} \quad , \quad N_2^{(1)}(x) = \frac{x_2 - x}{x_2 - x_1} \\ N_1^{(2)}(x) &= \frac{x - x_2}{x_3 - x_2} \quad , \quad N_2^{(2)}(x) = \frac{x_3 - x}{x_3 - x_2} \end{aligned} \quad (7.31)$$

Note, the shape functions in local coordinates depend only on the structure of the element and the number of nodal points. The above shape functions define a normalized, universal element.

By use of linear Lagrangian elements, the unknown function $u(x)$ is approximated on element 1 by

$$\hat{u}(x) = N_1^{(1)}(x)u_{i-1} + N_2^{(1)}(x)u_i = u_{i-1} + \frac{x - x_{i-1}}{x_i - x_{i-1}}(u_i - u_{i-1}) \quad (7.32)$$

and on element 2, respectively

$$\hat{u}(x) = N_1^{(2)}(x)u_i + N_2^{(2)}(x)u_{i+1} = u_i + \frac{x - x_i}{x_{i+1} - x_i}(u_{i+1} - u_i) \quad (7.33)$$

Derivatives

The spatial derivative of u is then approximated by

$$\left[\frac{\partial \hat{u}(x)}{\partial x} \right]^{(e)} = \sum_{i=1}^2 \frac{\partial N_i^{(e)}}{\partial x} u_i = -\frac{u_i - u_{i-1}}{x_i - x_{i-1}} + \frac{u_{i+1} - u_i}{x_{i+1} - x_i} \quad (7.34)$$

and the nodal values at point x_i from the different elements are

$$\left[\frac{\partial \hat{u}(x)}{\partial x} \right]_i^{(1)} = \frac{u_i - u_{i-1}}{x_i - x_{i-1}} \quad , \quad \left[\frac{\partial \hat{u}(x)}{\partial x} \right]_i^{(2)} = \frac{u_{i+1} - u_i}{x_{i+1} - x_i} \quad (7.35)$$

which correspond to first-order backward and forward difference schemes, respectively. Note, the approximate derivatives are not continuous at element boundary, there exists only C^0 continuity. To define a local derivative, we average the individual contributions of the connected elements. If an arithmetic average is taken, we have

$$\left[\frac{\partial \hat{u}(x)}{\partial x} \right]_i^{(e)} = \frac{1}{2} \left(\frac{u_{i+1} - u_i}{x_{i+1} - x_i} - \frac{u_i - u_{i-1}}{x_i - x_{i-1}} \right) \quad (7.36)$$

It is note worthy, that the above local approximation is second-order accurate in space but only on equidistant meshes. However, if we use a weighted average according to

$$\left[\frac{\partial \hat{u}(x)}{\partial x} \right]_i^{(e)} = \frac{1}{x_{i+1} - x_{i-1}} \left((x_{i+1} - x_i) \left[\frac{\partial \hat{u}(x)}{\partial x} \right]_i^{(1)} + (x_i - x_{i-1}) \left[\frac{\partial \hat{u}(x)}{\partial x} \right]_i^{(2)} \right)$$

we obtain a central scheme

$$\left[\frac{\partial \hat{u}(x)}{\partial x} \right]_i^{(e)} = \frac{1}{x_{i+1} - x_{i-1}} \left(\frac{x_i - x_{i-1}}{x_{i+1} - x_i} (u_{i+1} - u_i) + \frac{x_{i+1} - x_i}{x_i - x_{i-1}} (u_i - u_{i-1}) \right) \quad (7.37)$$

which is second-order accurate in space on arbitrary meshes.

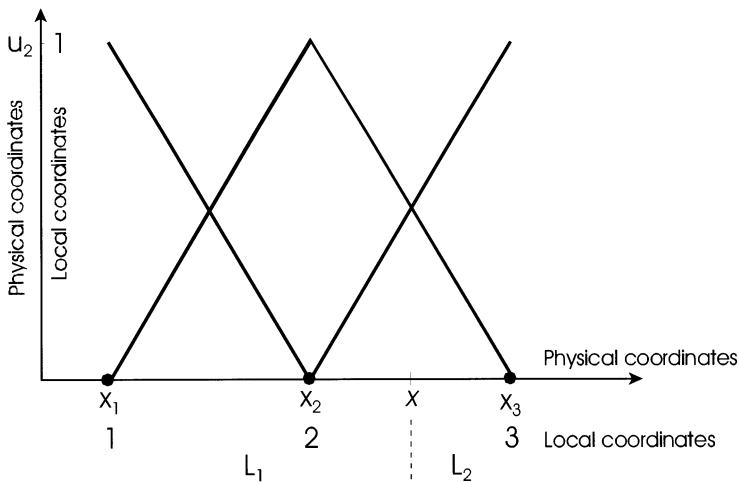


Figure 7.5: 1-D linear elements in local and physical coordinates

Isoparametric mapping

We assume that the coordinate transformation $(x) \rightarrow (r)$ can be conducted by a linear function.

$$x(r) = a_1 + a_2 r \quad (7.38)$$

Additional we require

$$x(r = -1) = x_1 \quad , \quad x(r = +1) = x_2 \quad (7.39)$$

i.e. we use an unified element in local (natural) coordinates as shown in Fig. 7.6. In other words, by this mapping function we convert the $[x_1, x_2]$ -space into normalized $[-1, 1]$ -space.

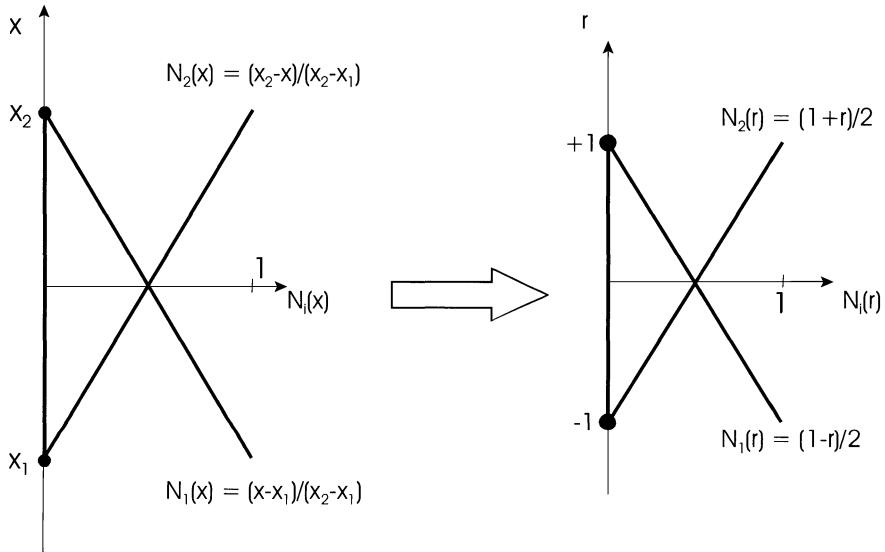


Figure 7.6: Isoparametric coordinate transformation between physical and local coordinates

Performing the same procedure as in Chapter 7.4.1 we obtain the following relations

$$x(r) = \frac{1-r}{2}x_1 + \frac{r-1}{2}x_2 = N_1(r)x_1 + N_2(r)x_2 \quad (7.40)$$

Therefore, we can write the coordinate transformation by use of shape functions.

$$N_1^{(e)}(r) = \frac{1-r}{2} \quad , \quad N_2^{(e)}(r) = \frac{r-1}{2} \quad (7.41)$$

This means we interpolate the coordinates in the same manner as the unknown function u . We see in local coordinates these shape functions take a simple form and they are independently on physical coordinates. This coordinate transformation allows us to write the trial solution for the unknown function u in terms of local coordinates.

7.4.2 1-D Quadratic Elements

In general we used Lagrangian polynomials to obtain shape functions. For an one-dimensional element consisting of m nodes we can write the following Lagrangian polynomial.

$$N_i = \prod_{i=1, j \neq i}^m \left[\frac{x - x_j}{x_i - x_j} \right] = \frac{(x - x_1)(x - x_2) \dots (x - x_{i-1})(x - x_{i+1}) \dots (x - x_m)}{(x_i - x_1)(x_i - x_2) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_m)}$$

The use of linear interpolation functions imposes a constraint on the approximate solution that it must vary linearly between the nodal points. Quadratic interpolation functions are required if linear interpolation failed to approximate the unknown function, e.g. if continuity of first-order derivatives is required. Moreover, it is expected that a smaller interpolation error will be occur if quadratic interpolation is used for the same grid instead of linear one.

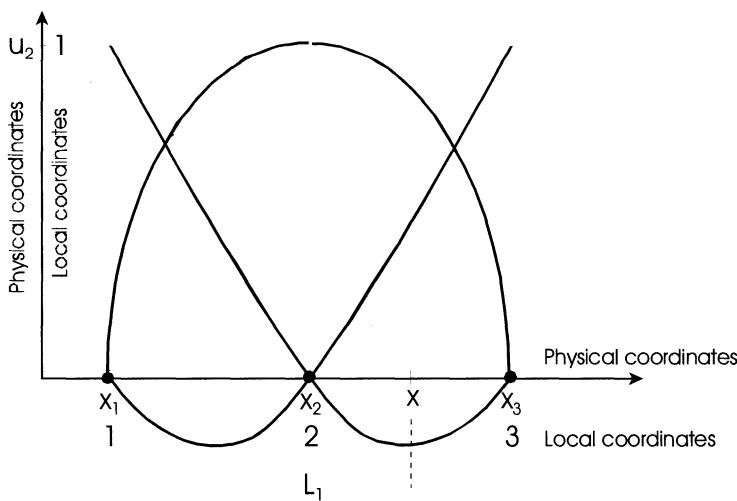


Figure 7.7: 1-D quadratic element in local and physical coordinates

Use of three-node elements leads to a quadratic shape functions

$$N_1^{(e)}(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)}$$

$$N_2^{(e)}(x) = \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)}$$

$$N_3^{(e)}(x) = \frac{(x - x_1)(x - x_2)}{(x_3 - x_2)(x_3 - x_1)} \quad (7.42)$$

7.4.3 2-D Triangular Elements

We consider a triangular element in physical space (x, y) as shown in Fig. 7.8.

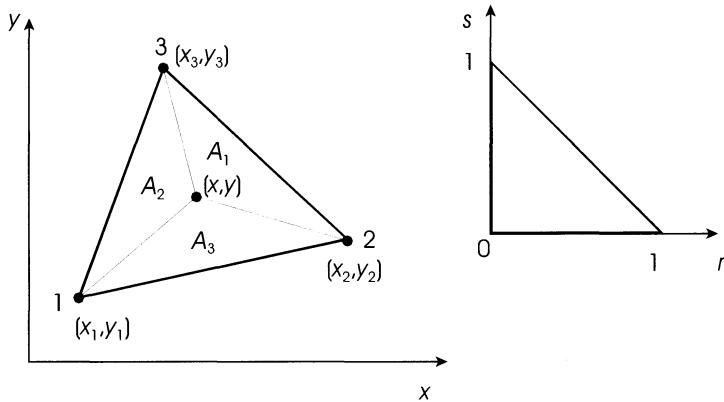


Figure 7.8: Linear triangular element

Local coordinates

Obviously, an arbitrary point in the triangle can be identified by use of the following local (area) coordinates (Fig. 7.8),

$$\begin{aligned} L_1 &= A_1/A \\ L_2 &= A_2/A \\ L_3 &= A_3/A \end{aligned} \quad (7.43)$$

where A is the area of triangular element

$$A = \frac{1}{2} \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \quad (7.44)$$

From geometrical reasons we have

$$1 = L_1 + L_2 + L_3 \quad (7.45)$$

Furthermore we can write

$$L_i = \begin{cases} 1 & : \text{ at node } i \\ 0 & : \text{ at remaining nodes} \end{cases} \quad (7.46)$$

From this condition it can be concluded that

$$\begin{aligned} x &= L_1 x_1 + L_2 x_2 + L_3 x_3 \\ y &= L_1 y_1 + L_2 y_2 + L_3 y_3 \end{aligned} \quad (7.47)$$

Therefore, L_1 , L_2 and L_3 can be used as a kind of local normalized coordinates. i.e. the physical coordinates of an arbitrary point within the triangle are given by use of the above local coordinates as above.

Now we write the above equations in following compact matrix form

$$\begin{Bmatrix} 1 \\ x \\ y \end{Bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} \begin{Bmatrix} L_1 \\ L_2 \\ L_3 \end{Bmatrix} \quad (7.48)$$

Inversion gives

$$\begin{Bmatrix} L_1 \\ L_2 \\ L_3 \end{Bmatrix} = \frac{1}{2A} \begin{bmatrix} x_2 y_3 - x_3 y_2 & y_2 - y_3 & x_3 - x_2 \\ x_3 y_1 - x_1 y_3 & y_3 - y_1 & x_1 - x_3 \\ x_1 y_2 - x_2 y_1 & y_1 - y_2 & x_2 - x_1 \end{bmatrix} \begin{Bmatrix} 1 \\ x \\ y \end{Bmatrix} \quad (7.49)$$

Shape functions

We approximate the unknown function $u(x, y)$ at the triangular element by a linear approximation.

$$\hat{u}(x, y) = a_1 + a_2 x + a_3 y \quad (7.50)$$

At the corner nodes we require that

$$\begin{aligned} u_1 &= a_1 + a_2 x_1 + a_3 y_1 \\ u_2 &= a_1 + a_2 x_2 + a_3 y_2 \\ u_3 &= a_1 + a_2 x_3 + a_3 y_3 \end{aligned} \quad (7.51)$$

or written compactly

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} \quad (7.52)$$

By inversion of the above matrix equation we can derive relations for the determination of the unknown interpolation coefficients

$$\begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} = \frac{1}{2A} \begin{bmatrix} x_2y_3 - x_3y_2 & x_3y_1 - x_1y_3 & x_1y_2 - x_2y_1 \\ y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_3 - x_2 & x_1 - x_3 & x_2 - x_1 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} \quad (7.53)$$

Inserting the above relations for the coefficients a_i into equation (7.50) this can be rearranged and rewritten as a trial solution for the element

$$\hat{u}(x, y) = N_1(x, y)u_1 + N_2(x, y)u_2 + N_3(x, y)u_3 \quad (7.54)$$

By this way, we derive following element-related interpolation functions (i.e. shape functions) for a linear triangular element.

$$\begin{aligned} N_1(x, y) &= \frac{1}{2A} [(x_2y_3 - x_3y_2) + (y_2 - y_3)x + (x_3 - x_2)y] \\ N_2(x, y) &= \frac{1}{2A} [(x_3y_1 - x_1y_3) + (y_3 - y_1)x + (x_1 - x_3)y] \\ N_3(x, y) &= \frac{1}{2A} [(x_1y_2 - x_2y_1) + (y_1 - y_2)x + (x_2 - x_1)y] \end{aligned} \quad (7.55)$$

or written in a compact matrix form

$$\begin{Bmatrix} N_1 \\ N_2 \\ N_3 \end{Bmatrix} = \frac{1}{2A} \begin{bmatrix} x_2y_3 - x_3y_2 & y_2 - y_3 & x_3 - x_2 \\ x_3y_1 - x_1y_3 & y_3 - y_1 & x_1 - x_3 \\ x_1y_2 - x_2y_1 & y_1 - y_2 & x_2 - x_1 \end{bmatrix} \begin{Bmatrix} 1 \\ x \\ y \end{Bmatrix} \quad (7.56)$$

Comparing equation (7.73) with the above one, we see the shape functions N_i are simply the area coordinates L_i , i.e. we have isoparametric elements.

Now the derivatives of the shape functions can be easily written down.

$$\frac{\partial \mathbf{N}}{\partial x} = \left\{ \begin{array}{l} \frac{\partial N_1}{\partial x} = \frac{y_2 - y_3}{2A} \\ \frac{\partial N_2}{\partial x} = \frac{y_3 - y_1}{2A} \\ \frac{\partial N_3}{\partial x} = \frac{y_1 - y_2}{2A} \end{array} \right\} \quad \frac{\partial \mathbf{N}}{\partial y} = \left\{ \begin{array}{l} \frac{\partial N_1}{\partial y} = \frac{x_3 - x_2}{2A} \\ \frac{\partial N_2}{\partial y} = \frac{x_1 - x_3}{2A} \\ \frac{\partial N_3}{\partial y} = \frac{x_2 - x_1}{2A} \end{array} \right\} \quad (7.57)$$

Integration

When element matrices have to be evaluated we are faced with integration of quantities defined in terms of area coordinates. There is the following exact integration expression for linear triangular elements.

$$\iint_{\Delta} L_1^a L_2^b L_3^c dx dy = 2\Delta \frac{a!b!c!}{(a+b+c+2)!} \quad (7.58)$$

7.4.4 2-D Bilinear Quadrilateral Elements

Local coordinates

Again we want to approximate the unknown function by a trial solution based on four nodes of a quadrilateral element (Fig. 7.9)

$$\hat{u}(x) = \sum_{i=1}^4 N_i(x, y) u_i \quad (7.59)$$

A typical situation, illustrating the advantage of isoparametric mapping is given in following figure. The above considered element is distorted in the physical space (x, y) . However, these elements can easily transformed into uniform elements in a virtual space (ξ, η) by using mapping functions according to a Lagrangian interpolation.

Shape functions

For the element interpolation we can use the product of first-degree Lagrangian polynomials

$$N_i(x, y) = L_i(x)L_i(y) = \frac{x - x_i}{x_i - x_{i+1}} \frac{y - y_i}{y_i - y_{i-1}} \quad (7.60)$$

It is most convenient to define shape functions in so-called natural (unified) coordinates (Fig. 7.9) and then transform the relationship into the physical coordinates. The set of shape functions becomes then

$$\begin{aligned} N_1(r, s) &= \frac{1}{4}(1+r)(1+s) \\ N_2(r, s) &= \frac{1}{4}(1-r)(1+s) \\ N_3(r, s) &= \frac{1}{4}(1-r)(1-s) \\ N_4(r, s) &= \frac{1}{4}(1+r)(1-s) \end{aligned} \quad (7.61)$$

To approximate derivatives of the unknown function we need to have the derivatives of the interpolation function. To this purpose we have to determine the inverse Jacobian matrix.

$$\nabla_{xy} \mathbf{N} = [J_{2D}]^{-1} \nabla_{rs} \mathbf{N} \quad (7.62)$$

with

$$[J_{2D}]^{-1} = \begin{bmatrix} \frac{\partial r}{\partial x} & \frac{\partial s}{\partial x} \\ \frac{\partial r}{\partial y} & \frac{\partial s}{\partial y} \end{bmatrix} \quad (7.63)$$

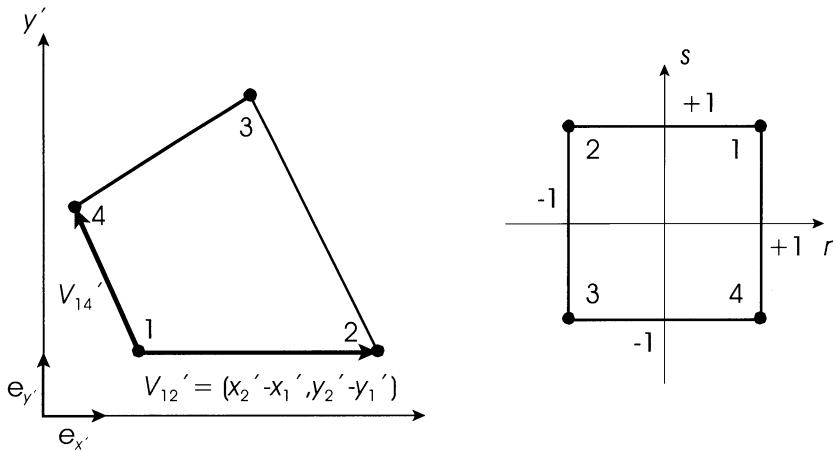


Figure 7.9: Isoparametric mapping of bilinear quadrilateral elements

$$\nabla_{xy} \mathbf{N} = \begin{bmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_1}{\partial y} \\ \frac{\partial N_2}{\partial x} & \frac{\partial N_2}{\partial y} \\ \frac{\partial N_3}{\partial x} & \frac{\partial N_3}{\partial y} \\ \frac{\partial N_4}{\partial x} & \frac{\partial N_4}{\partial y} \end{bmatrix}, \quad \nabla_{rs} \mathbf{N} = \begin{bmatrix} \frac{\partial N_1}{\partial r} & \frac{\partial N_1}{\partial s} \\ \frac{\partial N_2}{\partial r} & \frac{\partial N_2}{\partial s} \\ \frac{\partial N_3}{\partial r} & \frac{\partial N_3}{\partial s} \\ \frac{\partial N_4}{\partial r} & \frac{\partial N_4}{\partial s} \end{bmatrix} \quad (7.64)$$

In general we can use the above interpolation function (7.59) to transform coordinates too

$$\begin{aligned} x &= x(r, s) = \sum_{i=1}^4 N_i(r, s) x_i \\ y &= y(r, s) = \sum_{i=1}^4 N_i(r, s) y_i \end{aligned} \quad (7.65)$$

Then we speak about an isoparametric coordinate transformation. In this case the two-dimensional Jacobian matrix is given by

$$[J_{2D}] = \begin{bmatrix} \frac{\partial N_i}{\partial r} x_i & \frac{\partial N_i}{\partial r} y_i \\ \frac{\partial N_i}{\partial s} x_i & \frac{\partial N_i}{\partial s} y_i \end{bmatrix} = [\nabla_{rs} \mathbf{N}] [A_{2D}] \quad (7.66)$$

Finally we determine $[J_{2D}]^{-1}$ by inversion of the Jacobian.

7.4.5 3-D Tetrahedral Elements

We consider a tetrahedral element in 3-D physical space (x, y, z) as shown in Fig. 7.10. Faces of the tetrahedral element are triangles. In the following we will see, properties of a tetrahedron are very similar to that of triangular elements.

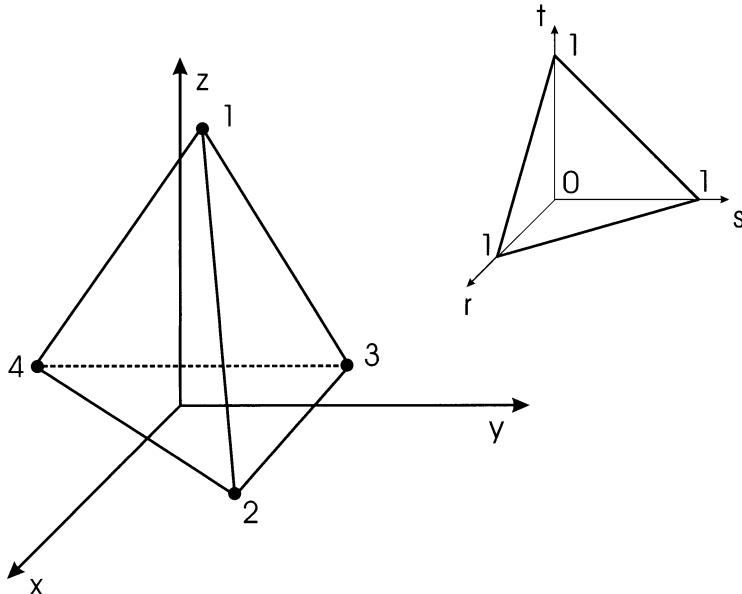


Figure 7.10: Linear tetrahedral element

Local coordinates

Obviously, an arbitrary point in the tetrahedron can be identified by use of the following local (volumetric) coordinates (Fig. 7.10),

$$\begin{aligned} L_1 &= V_1/V \\ L_2 &= V_2/V \\ L_3 &= V_3/V \\ L_4 &= V_4/V \end{aligned} \tag{7.67}$$

where V is the volume of tetrahedral element

$$V = \frac{1}{6} \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix} = \frac{1}{6} \begin{vmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{vmatrix} \tag{7.68}$$

For geometrical reasons we have

$$1 = L_1 + L_2 + L_3 + L_4 \quad (7.69)$$

Furthermore we can write

$$L_i = \begin{cases} 1 & : \text{ at node } i \\ 0 & : \text{ at remaining nodes} \end{cases} \quad (7.70)$$

From this condition it can be concluded that

$$\begin{aligned} x &= L_1 x_1 + L_2 x_2 + L_3 x_3 + L_4 x_4 \\ y &= L_1 y_1 + L_2 y_2 + L_3 y_3 + L_4 y_4 \\ z &= L_1 z_1 + L_2 z_2 + L_3 z_3 + L_4 z_4 \end{aligned} \quad (7.71)$$

Therefore, L_1 , L_2 , L_3 and L_4 can be used as a kind of local normalized coordinates. i.e. the physical coordinates of an arbitrary point within the tetrahedron are given by use of the above local coordinates as above.

Now we write the above equations in following compact matrix form

$$\begin{Bmatrix} 1 \\ x \\ y \\ z \end{Bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{bmatrix} \begin{Bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{Bmatrix} \quad (7.72)$$

Inversion gives

$$\begin{Bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{Bmatrix} = \frac{1}{6V} \begin{bmatrix} a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ a_3 & b_3 & c_3 & d_3 \\ a_4 & b_4 & c_4 & d_4 \end{bmatrix} \begin{Bmatrix} 1 \\ x \\ y \\ z \end{Bmatrix} \quad (7.73)$$

with

$$\begin{aligned} a_i &= \begin{vmatrix} x_{i+1} & y_{i+1} & z_{i+1} \\ x_{i+2} & y_{i+2} & z_{i+2} \\ x_{i+3} & y_{i+3} & z_{i+3} \end{vmatrix}, \quad -b_i = \begin{vmatrix} 1 & y_{i+1} & z_{i+1} \\ 1 & y_{i+2} & z_{i+2} \\ 1 & y_{i+3} & z_{i+3} \end{vmatrix} \\ c_i &= \begin{vmatrix} x_{i+1} & 1 & z_{i+1} \\ x_{i+2} & 1 & z_{i+2} \\ x_{i+3} & 1 & z_{i+3} \end{vmatrix}, \quad d_i = \begin{vmatrix} x_{i+1} & y_{i+1} & 1 \\ x_{i+2} & y_{i+2} & 1 \\ x_{i+2} & y_{i+3} & 1 \end{vmatrix} \end{aligned} \quad (7.74)$$

where i is running in a cyclic permutation $\{1, 2, 3, 4\}$.

Shape functions

Following the same procedure as described for the triangular element (sec 7.4.3), shape functions and local coordinates are identical.

$$\begin{Bmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \end{Bmatrix} \equiv \begin{Bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{Bmatrix} = \frac{1}{6V} \begin{bmatrix} a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ a_3 & b_3 & c_3 & d_3 \\ a_4 & b_4 & c_4 & d_4 \end{bmatrix} \begin{Bmatrix} 1 \\ x \\ y \\ z \end{Bmatrix} \quad (7.75)$$

Derivatives of shape functions are

$$\frac{\partial N_i}{\partial x} = \frac{b_i}{6V}, \quad \frac{\partial N_i}{\partial y} = \frac{c_i}{6V}, \quad \frac{\partial N_i}{\partial z} = \frac{d_i}{6V} \quad (7.76)$$

Integration

When element matrices have to be evaluated we are faced with integration of quantities defined in terms of area coordinates. There is the following exact integration expression for linear tetrahedral elements.

$$\iiint_V L_1^a L_2^b L_3^c L_4^d dx dy dz = 6V \frac{a! b! c! d!}{(a+b+c+d+3)!} \quad (7.77)$$

7.4.6 3-D Triangular Prismatic Elements

We consider a triangular prism element in 3-D space (x, y, z) as shown in Fig. 7.11. This element is a combination of triangles and quadrilaterals. Those elements have many practical applications, e.g. if 2-D discretizations have to be extended into three dimensions.

Local coordinates

We use local (area) coordinates for triangles.

$$\begin{aligned} L_1 &= N_1^\Delta = A_1/A \\ L_2 &= N_2^\Delta = A_2/A \\ L_3 &= N_3^\Delta = A_3/A \end{aligned} \quad (7.78)$$

where A is the area of triangular element basis.

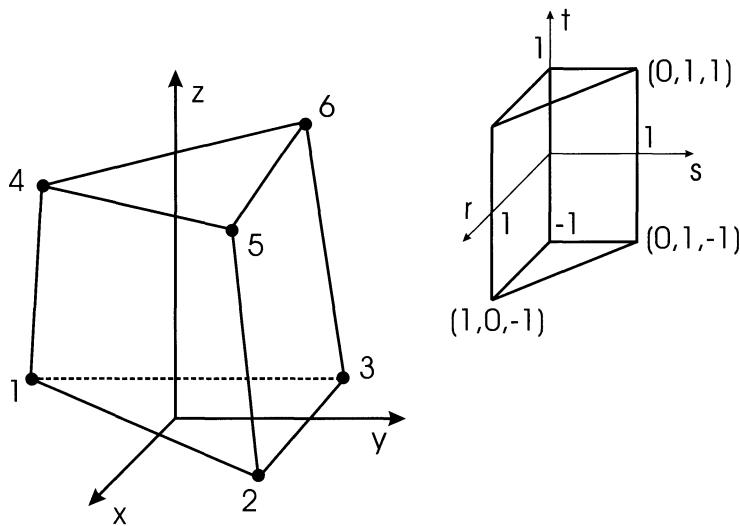


Figure 7.11: Triangular prismatic element

Shape functions

Shape functions of triangle based prism elements can be derived by superposition of shape functions for triangles and line elements.

$$\begin{aligned} N_1 &= N_1^\Delta(1-t) \quad , \quad N_2 = N_2^\Delta(1-t) \quad , \quad N_3 = N_3^\Delta(1-t) \\ N_4 &= N_1^\Delta(1+t) \quad , \quad N_5 = N_2^\Delta(1+t) \quad , \quad N_6 = N_3^\Delta(1+t) \end{aligned} \quad (7.79)$$

Derivatives of shape functions are

$$\nabla N = \left\{ \begin{array}{l} \frac{\partial N_1}{\partial x} \quad \frac{\partial N_1}{\partial y} \quad \frac{\partial N_1}{\partial z} \\ \frac{\partial N_2}{\partial x} \quad \frac{\partial N_2}{\partial y} \quad \frac{\partial N_2}{\partial z} \\ \frac{\partial N_3}{\partial x} \quad \frac{\partial N_3}{\partial y} \quad \frac{\partial N_3}{\partial z} \\ \frac{\partial N_4}{\partial x} \quad \frac{\partial N_4}{\partial y} \quad \frac{\partial N_4}{\partial z} \\ \frac{\partial N_5}{\partial x} \quad \frac{\partial N_5}{\partial y} \quad \frac{\partial N_5}{\partial z} \\ \frac{\partial N_6}{\partial x} \quad \frac{\partial N_6}{\partial y} \quad \frac{\partial N_6}{\partial z} \end{array} \right\} = \left[\begin{array}{ccc} \frac{\partial N_1^\Delta}{\partial x}(1-t) & \frac{\partial N_1^\Delta}{\partial y}(1-t) & -N_1^\Delta J_z \\ \frac{\partial N_2^\Delta}{\partial x}(1-t) & \frac{\partial N_2^\Delta}{\partial y}(1-t) & -N_2^\Delta J_z \\ \frac{\partial N_3^\Delta}{\partial x}(1-t) & \frac{\partial N_3^\Delta}{\partial y}(1-t) & -N_3^\Delta J_z \\ \frac{\partial N_1^\Delta}{\partial x}(1+t) & \frac{\partial N_1^\Delta}{\partial y}(1+t) & N_1^\Delta J_z \\ \frac{\partial N_2^\Delta}{\partial x}(1+t) & \frac{\partial N_2^\Delta}{\partial y}(1+t) & N_2^\Delta J_z \\ \frac{\partial N_3^\Delta}{\partial x}(1+t) & \frac{\partial N_3^\Delta}{\partial y}(1+t) & N_3^\Delta J_z \end{array} \right] \quad (7.80)$$

with N_i^Δ the 2-D shape functions of triangular elements (7.75) and the Jacobian

for line elements

$$J_z = \frac{dt}{dz} = \frac{2A}{A_1|\mathbf{z}_{14}| + A_2|\mathbf{z}_{25}| + A_3|\mathbf{z}_{36}|} \quad (7.81)$$

for prismatic elements.

Integration

When element matrices have to be evaluated we are faced with integration of quantities defined in terms of area coordinates. Ratke (1996) outlined the following procedure.

$$\begin{aligned} \iiint_V f_1(N_1^\Delta, N_2^\Delta, N_3^\Delta) f_2(N_z) dV &= \iint_{\Delta} f_1 \left(\int_{z_u}^{z_o} f_2 dz \right) dA \\ &= \iint_{\Delta} f_1 J_z^{-1} \left(\int_{-1}^{+1} f_2 dt \right) dA \\ &= \iint_{\Delta} f_1 J_z^{-1} dA \times \int_{-1}^{+1} f_2 dt \end{aligned} \quad (7.82)$$

Now we can use the integration formula (7.58) for triangular integrals.

7.4.7 3-D Hexahedral Element

We consider a hexahedral element in 3-D space (x, y, z) as shown in Fig. 7.12. It is a straightforward generalization of the 2-D bilinear quadrilateral element presented in section 7.4.4.

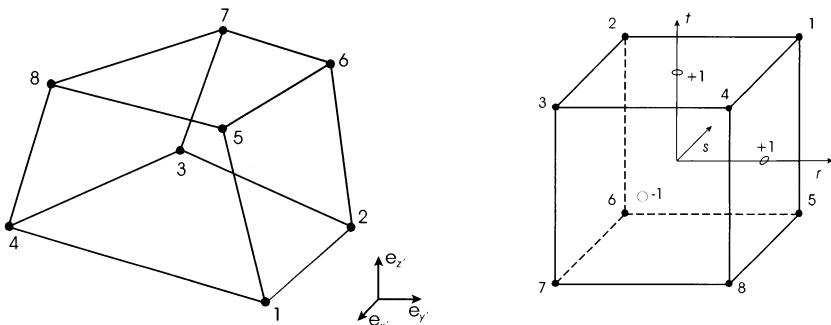


Figure 7.12: Hexahedron in global and local coordinates

Local coordinates

We use isoparametric mapping for coordinate transformation between global and local coordinates (Fig. 7.12).

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \sum_{i=1}^8 N_i(r, s, t) \begin{Bmatrix} x_i \\ y_i \\ z_i \end{Bmatrix} \quad (7.83)$$

Shape functions

For the element interpolation we can use the product of first-degree Lagrangian polynomials

$$N_i(x, y, z) = L_i(x) L_i(y) L_i(z) \quad (7.84)$$

Shape functions in local and global element coordinates. can be defined in following way.

$$\begin{aligned} N_1 &= \frac{1}{8}(1+r)(1+s)(1+t), & N_1 &= \frac{x-x_2}{x_1-x_2} \frac{y-y_4}{y_1-y_4} \frac{z-z_5}{z_1-z_5} \\ N_2 &= \frac{1}{8}(1-r)(1+s)(1+t), & N_2 &= \frac{x-x_3}{x_2-x_3} \frac{y-y_1}{y_2-y_1} \frac{z-z_6}{z_2-z_6} \\ N_3 &= \frac{1}{8}(1-r)(1-s)(1+t), & N_3 &= \frac{x-x_4}{x_3-x_4} \frac{y-y_2}{y_3-y_2} \frac{z-z_7}{z_3-z_7} \\ N_4 &= \frac{1}{8}(1+r)(1-s)(1+t), & N_4 &= \frac{x-x_1}{x_4-x_1} \frac{y-y_3}{y_4-y_3} \frac{z-z_8}{z_4-z_8} \\ N_5 &= \frac{1}{8}(1+r)(1+s)(1-t), & N_5 &= \frac{x-x_6}{x_5-x_6} \frac{y-y_8}{y_5-y_8} \frac{z-z_1}{z_5-z_1} \\ N_6 &= \frac{1}{8}(1-r)(1+s)(1-t), & N_6 &= \frac{x-x_7}{x_6-x_7} \frac{y-y_5}{y_6-y_5} \frac{z-z_2}{z_6-z_2} \\ N_7 &= \frac{1}{8}(1-r)(1-s)(1-t), & N_7 &= \frac{x-x_8}{x_7-x_8} \frac{y-y_6}{y_7-y_6} \frac{z-z_3}{z_7-z_3} \\ N_8 &= \frac{1}{8}(1+r)(1-s)(1-t), & N_8 &= \frac{x-x_5}{x_8-x_5} \frac{y-y_7}{y_8-y_7} \frac{z-z_4}{z_8-z_4} \end{aligned} \quad (7.85)$$

Implementation: MOmega3D(omega,r,s,t)

The matrix of shape function derivatives is therefore given as follows

$$\nabla N_{rst} = \begin{bmatrix} \frac{\partial N_1}{\partial r} & \frac{\partial N_1}{\partial s} & \frac{\partial N_1}{\partial t} \\ \frac{\partial N_2}{\partial r} & \frac{\partial N_2}{\partial s} & \frac{\partial N_2}{\partial t} \\ \frac{\partial N_3}{\partial r} & \frac{\partial N_3}{\partial s} & \frac{\partial N_3}{\partial t} \\ \frac{\partial N_4}{\partial r} & \frac{\partial N_4}{\partial s} & \frac{\partial N_4}{\partial t} \\ \frac{\partial N_5}{\partial r} & \frac{\partial N_5}{\partial s} & \frac{\partial N_5}{\partial t} \\ \frac{\partial N_6}{\partial r} & \frac{\partial N_6}{\partial s} & \frac{\partial N_6}{\partial t} \\ \frac{\partial N_7}{\partial r} & \frac{\partial N_7}{\partial s} & \frac{\partial N_7}{\partial t} \\ \frac{\partial N_8}{\partial r} & \frac{\partial N_8}{\partial s} & \frac{\partial N_8}{\partial t} \end{bmatrix} \quad (7.86)$$

$$= \frac{1}{8} \begin{bmatrix} (1+s)(1+t) & (1+r)(1+t) & (1+s)(1+t) \\ -(1+s)(1+t) & (1-r)(1+t) & (1+s)(1+t) \\ -(1-s)(1+t) & -(1-r)(1+t) & (1-s)(1+t) \\ (1-s)(1+t) & -(1+r)(1+t) & (1-s)(1+t) \\ (1+s)(1-t) & (1+r)(1-t) & (1+s)(1-t) \\ -(1+s)(1-t) & (1-r)(1-t) & (1+s)(1-t) \\ -(1-s)(1-t) & -(1-r)(1-t) & (1-s)(1-t) \\ (1-s)(1-t) & -(1+r)(1-t) & (1-s)(1-t) \end{bmatrix}$$

Implementation: MGradOmega3D(gradient,r,s,t)

Shape function derivatives in global coordinates ($x - y - z$) can be determined by use of the inverse Jacobian matrix $[J_{3D}]^{-1}$.

$$\nabla_{xyz} \mathbf{N} = [J_{3D}]^{-1} \nabla_{rst} \mathbf{N} \quad (7.87)$$

The Jacobian matrix in 3-D case is given by

$$[J_{3D}] = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} & \frac{\partial z}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} & \frac{\partial z}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} & \frac{\partial z}{\partial t} \end{bmatrix} = [\nabla_{rst} \mathbf{N}] [A_{3D}] \quad (7.88)$$

which can be rewritten, if using the isoparametric coordinate transformation eqn (7.83), as

$$[J_{3D}] = \begin{bmatrix} \frac{\partial N_i}{\partial r} x_i & \frac{\partial N_i}{\partial r} y_i & \frac{\partial N_i}{\partial r} z_i \\ \frac{\partial N_i}{\partial s} x_i & \frac{\partial N_i}{\partial s} y_i & \frac{\partial N_i}{\partial s} z_i \\ \frac{\partial N_i}{\partial t} x_i & \frac{\partial N_i}{\partial t} y_i & \frac{\partial N_i}{\partial t} z_i \end{bmatrix} \quad (7.89)$$

Implementation: Calc3DElementJacobianMatrix(element,r,s,t,invjac,detjac)

The required inverse Jacobian $[J_{3D}]^{-1}$ has to be calculated by formulas for matrix inversion.

Implementation: M3Invertiere(matrix)

7.5 Network Simulation

7.5.1 Pipe Networks - 1-D Elements in \mathcal{R}^3

We consider linear one-dimensional elements in three-dimensional space. Those elements may be used e.g. for pipe network modeling. The following figure shows the physical (x, y, z) and the virtual (or natural) systems of coordinates (r) .

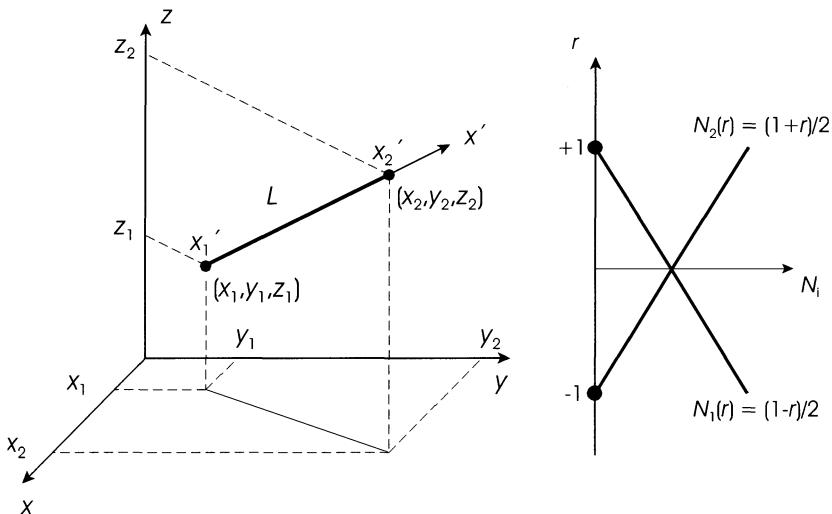


Figure 7.13: 1-D element in \mathcal{R}^3 and in local isoparametric coordinates

Derivatives of shape functions of isoparametric elements are

$$\begin{aligned}\frac{dN_1}{dx'} &= \frac{dr}{dx'} \frac{dN_1}{dr} = [J_{1D}^{-1}] \frac{dN_1}{dr} = -\frac{1}{L} \\ \frac{dN_2}{dx'} &= \frac{dr}{dx'} \frac{dN_2}{dr} = [J_{1D}^{-1}] \frac{dN_2}{dr} = \frac{1}{L}\end{aligned}\quad (7.90)$$

The Jacobian matrix of the 1-D element in local physical coordinates is given by

$$[J_{1D}] = \left[\frac{dx'}{dr} \right] = \left[\frac{dN_1}{dr} \frac{dN_2}{dr} \right] \begin{bmatrix} x'_1 \\ x'_2 \end{bmatrix} = \left[\frac{dN}{dr} \right]^T [A'_{1D}] \quad (7.91)$$

Nodal coordinates have to be transformed into the global coordinate system by use of following transformation matrices

$$\begin{aligned}[A'_{1D}] &= \begin{bmatrix} x'_1 \\ x'_2 \end{bmatrix} = [\cos(x', x) \cos(x', y) \cos(x', z)] \begin{bmatrix} x_1 & x_2 \\ y_1 & y_2 \\ z_1 & z_2 \end{bmatrix} \\ &= [{}_{1D}T_{3D}] [A_{3D}]^T\end{aligned}\quad (7.92)$$

The Jacobian matrix (1st rank tensor) of the 1-D element can be written in global physical coordinates as follows

$$\begin{aligned}[J_{1D}] &= \left[\frac{dN}{dr} \right]^T [{}_{1D}T_{3D}] [A_{3D}]^T \\ &= \frac{1}{2} \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} = \frac{L}{2}\end{aligned}\quad (7.93)$$

The determinant of the Jacobian matrix is simply

$$\det[J_{1D}] = \frac{L}{2} \quad (7.94)$$

and therefore the inverse of the Jacobian matrix is given by

$$[J_{1D}^{-1}] = \frac{2}{L} \quad (7.95)$$

Alternative approach

Alternatively, the derivatives of shape functions in global coordinates can be calculated by the following one-step transformation.

$$\left\{ \begin{array}{l} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \\ \frac{\partial N}{\partial z} \end{array} \right\} = \left[\begin{array}{ccc} \frac{\partial x'}{\partial x} & \frac{\partial x'}{\partial y} & \frac{\partial x'}{\partial z} \\ \frac{\partial y'}{\partial x} & \frac{\partial y'}{\partial y} & \frac{\partial y'}{\partial z} \\ \frac{\partial z'}{\partial x} & \frac{\partial z'}{\partial y} & \frac{\partial z'}{\partial z} \end{array} \right] \left\{ \begin{array}{l} \frac{\partial N}{\partial x'} \\ \frac{\partial N}{\partial y'} \\ \frac{\partial N}{\partial z'} \end{array} \right\} \quad (7.96)$$

$$\begin{aligned}
 &= \begin{bmatrix} \cos(x', x) & \cos(x', y) & \cos(x', z) \\ \cos(y', x) & \cos(y', y) & \cos(y', z) \\ \cos(z', x) & \cos(z', y) & \cos(z', z) \end{bmatrix} \begin{Bmatrix} \frac{\partial N}{\partial x'} \\ 0 \\ 0 \end{Bmatrix} \\
 &= \begin{Bmatrix} \cos(x', x) \\ \cos(y', x) \\ \cos(z', x) \end{Bmatrix} \frac{\partial N}{\partial x'}
 \end{aligned}$$

7.5.2 Fracture Networks - 2-D Triangular Elements in \mathcal{R}^3

We consider linear triangular elements in three-dimensional space. Those elements may be used e.g. for fracture network modeling. The following figure shows several coordinate systems: global 3-D coordinates (x, y, z) , 2-D element coordinates (x', y') , and 2-D element tensor coordinates (x'', y'') ,

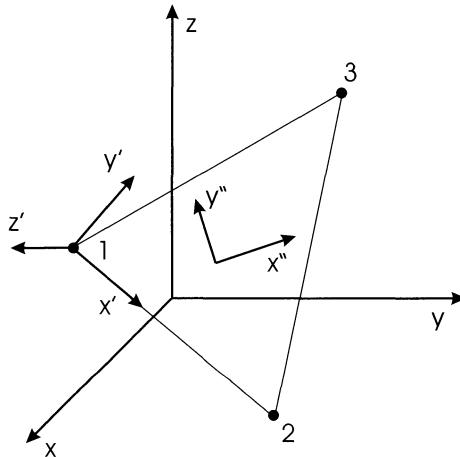


Figure 7.14: 2-D triangular element in \mathcal{R}^3 , several coordinate systems

Element Area

Calculation of triangle area is very simple using the following formula, which is well known from analytic geometry.

$$A = \frac{1}{2} |\mathbf{x}_{12} \times \mathbf{x}_{13}| = \begin{vmatrix} 1 & 1 & 1 \\ x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \end{vmatrix} \quad (7.97)$$

Implementation: `CalcElementVolumeTriangle()`

Derivatives of Shape Functions

Frequently, we need to have the derivatives of shape functions for several element matrices expressing fluxes of quantities (e.g. conductance, advection, diffusion, stiffness matrices).

$$\nabla_{x'y'} \mathbf{N} = \begin{bmatrix} \frac{\partial N_1}{\partial x'} & \frac{\partial N_1}{\partial y'} \\ \frac{\partial N_2}{\partial x'} & \frac{\partial N_2}{\partial y'} \\ \frac{\partial N_3}{\partial x'} & \frac{\partial N_3}{\partial y'} \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} y'_2 - y'_3 & x'_3 - x'_2 \\ y'_3 - y'_1 & x'_1 - x'_3 \\ y'_1 - y'_2 & x'_2 - x'_1 \end{bmatrix} \quad (7.98)$$

Note, we have written the derivatives in local element coordinates based on eqn (7.57). Now we have only to rewrite this expression in global coordinates ($x - y - z$) using the following coordinate transformation.

Coordinate Transformation

The transformation between global (3-D: $x - y - z$) and local (2-D: $x' - y'$) coordinates is given by

$$[A'_{2D}] = [A_{3D}] [{}_{2D}T_{3D}] \quad (7.99)$$

with:

- Local coordinate matrix

$$[A'_{2D}] = \begin{bmatrix} x'_1 & y'_1 \\ x'_2 & y'_2 \\ x'_3 & y'_3 \end{bmatrix} \quad (7.100)$$

- Coordinate transformation matrix

$$[{}_{2D}T_{3D}] = \begin{bmatrix} \cos(x', x) & \cos(y', x) \\ \cos(x', y) & \cos(y', y) \\ \cos(x', z) & \cos(y', z) \end{bmatrix} \quad (7.101)$$

- Global coordinate matrix

$$[A_{3D}] = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix} \quad (7.102)$$

Implementation: Calc2DElementCoordinatesTriangle()

Sometimes, material tensors are related to specific directions, i.e. they have anisotropic behavior. We assume that $x'' - y''$ are the principal directions (Fig.

7.14). In order to evaluate element matrices in local element coordinates, we have to transform those tensors into these coordinates.

$$\mathbf{k}' = \mathbf{T} \mathbf{k}'' \mathbf{T}^T \quad (7.103)$$

with coordinate transformation matrix:

$$\mathbf{T} = \begin{bmatrix} \cos(x'', x') & \cos(x'', y') \\ \cos(y'', x') & \cos(y'', y') \end{bmatrix} \quad (7.104)$$

7.6 Diffusion Equation

7.6.1 Steady Diffusion Equation

Example: 1-D steady diffusion equation

As an example we consider the one-dimensional diffusion equation

$$\frac{\partial}{\partial x} (\alpha \frac{\partial u}{\partial x}) = q \quad (7.105)$$

We develop a Galerkin weak formulation with linear elements. The discretized nodal equation is written as

$$-\sum_i u_i \int_{\Omega_j} \alpha \nabla N_i \cdot \nabla N_j d\Omega + \oint_{\partial\Omega_j} N_j \alpha \frac{\partial u}{\partial n} dS = \int_{\Omega_j} N_j q d\Omega \quad (7.106)$$

Using linear shape functions

$$N_1(x) = \frac{1-x}{x_2-x_1}, \quad N_2(x) = \frac{x}{x_2-x_1} \quad (7.107)$$

the weak formulation for the 1-D problem is

$$-\sum_{j=i-1} u_j \int_{i-1} \alpha \frac{\partial N_j}{\partial x} \frac{\partial N_i}{\partial x} dx - \int_{i-1} N_i q dx = 0 \quad (7.108)$$

The derivatives of the shape functions are given by

$$\frac{\partial N_1}{\partial x} = -\frac{1}{x_2-x_1}, \quad \frac{\partial N_2}{\partial x} = \frac{1}{x_2-x_1} \quad (7.109)$$

Performing the integration we obtain

$$\alpha_{i+\frac{1}{2}} \frac{u_{i+1} - u_i}{\Delta x^2} - \alpha_{i-\frac{1}{2}} \frac{u_i - u_{i-1}}{\Delta x^2} = \frac{q_{i-1} + 4q_i + q_{i+1}}{6} \quad (7.110)$$

For a linear variation of the diffusion coefficient we have

$$\alpha_{i+\frac{1}{2}} = \frac{1}{\Delta x} \int_{x_i} \alpha dx = \frac{\alpha_{i+1} + \alpha_i}{2} \quad (7.111)$$

Note the left-hand-side is identical to a central second-order finite difference scheme (6.11).

$$\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} \quad (7.112)$$

It is a general rule for the case of uniform meshes that elements of order p lead to $p+1$ -order accurate discretizations.

Example: 2-D steady diffusion equation

In order to demonstrate the method of weighted residuals we consider the 2-D steady diffusion equation

$$\frac{\partial}{\partial x} (\alpha \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y} (\alpha \frac{\partial u}{\partial y}) = q \quad (7.113)$$

The residual resulting from the approximation of the solution

$$R(\hat{u}) = \frac{\partial}{\partial x} (\alpha \frac{\partial \hat{u}}{\partial x}) + \frac{\partial}{\partial y} (\alpha \frac{\partial \hat{u}}{\partial y}) - q = \nabla \cdot (\alpha \nabla \hat{u}) - q \quad (7.114)$$

The method of weighted residuals yields

$$\int_{\Omega} \omega R(\hat{u}) d\Omega = \int_{\Omega} \omega (\nabla \cdot (\alpha \nabla \hat{u}) - q) d\Omega \quad (7.115)$$

Next we perform integration by part of the second order derivative terms.

$$\int_{\Omega} \omega \nabla \cdot (\alpha \nabla \hat{u}) d\Omega = - \int_{\Omega} \nabla \omega \cdot (\alpha \nabla \hat{u}) d\Omega + \int_{\Omega} \nabla (\omega \alpha \nabla \hat{u}) d\Omega \quad (7.116)$$

Using the Gauss-Ostrogradskian theorem for volume integrals we obtain

$$\int_{\Omega} \nabla (\omega \alpha \nabla \hat{u}) d\Omega = \oint_{\Omega} \omega \alpha \frac{\partial \hat{u}}{\partial n} dS \quad (7.117)$$

Finally we derived the following weak formulation of the 2-D diffusion equation.

$$- \int_{\Omega} \nabla \omega \cdot (\alpha \nabla \hat{u}) d\Omega + \oint_{\Omega} \omega \alpha \frac{\partial \hat{u}}{\partial n} dS = \int_{\Omega} \omega q d\Omega \quad (7.118)$$

7.6.2 Unsteady Diffusion Equation

As an example we consider isothermal flow of homogeneous liquids in a porous medium (e.g. saturated groundwater flow). Readers who are interested in more details of the derivation of the governing equations should refer to Pinder & Gray (1983) and Diersch (1985).

Governing Equations

We start from the differential form of the balance equation of the fluid mass (giving the exact solution for the unknown function - fluid pressure) which belongs to the parabolic PDEs (see Chapter 4.4).

$$S_0^p \frac{\partial p}{\partial t} - \frac{\partial}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}}{\mu} \left(\frac{\partial p}{\partial x_\beta} + \rho g \frac{\partial z}{\partial x_\beta} \right) \right) = Q_\rho \quad , \quad \alpha, \beta = 1, 2, 3 \quad (7.119)$$

Galerkin Method

The use of the method of weighted residuals (see 7.2.2) provides us an integral formulation of the flow equation to obtain the approximate solution.

$$\int_\Omega \omega_i \left[S_0^p \frac{\partial \hat{p}}{\partial t} - \frac{\partial}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}}{\mu} \left(\frac{\partial \hat{p}}{\partial x_\beta} + \rho g \frac{\partial \hat{z}}{\partial x_\beta} \right) \right) \right] d\Omega = \int_\Omega \omega_i Q_\rho d\Omega \quad (7.120)$$

The unknown function is approximated by a trial solution (7.7). Note in the following we use Einstein summation convention for repeating indices.

$$\hat{p}(t, x_\alpha) = \sum_{j=1}^m \phi_j(x_\alpha) p_j(t) \equiv \phi_j(x_\alpha) p_j(t) \quad (7.121)$$

According to the Galerkin method we use identical weighting ω and interpolation functions ϕ .

$$\int_\Omega \phi_i \left[S_0^p \phi_j \frac{dp_j}{dt} - \frac{\partial}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}}{\mu} \left(\frac{\partial \phi_j}{\partial x_\beta} p_j + \rho g \frac{\partial \phi_j}{\partial x_\beta} z_j \right) \right) \right] d\Omega = \int_\Omega \phi_i Q_\rho d\Omega \quad (7.122)$$

Partial integration $\phi \nabla A = \nabla(\phi A) - A \nabla \phi$ and the Gauss-Ostrogradskian integral theorem are used to reduce the order of the derivatives.

$$\begin{aligned} \int_\Omega \phi_i \left[S_0^p \phi_j \frac{dp_j}{dt} + \frac{\partial \phi_j}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}}{\mu} \left(\frac{\partial \phi_j}{\partial x_\beta} p_j + \rho g \frac{\partial \phi_j}{\partial x_\beta} z_j \right) \right) \right] d\Omega = \\ - \int_{\partial\Omega} \phi_i q_n dS + \int_\Omega \phi_i Q_\rho d\Omega \end{aligned} \quad (7.123)$$

with the outward flux vector

$$q_n = -\frac{k_{\alpha\beta}}{\mu} \left(\frac{\partial \hat{p}}{\partial x_\beta} + \rho g \frac{\partial z}{\partial x_\beta} \right) n_\alpha \quad (7.124)$$

We rearrange the above equation to put the unknowns terms only to the left hand side.

$$\begin{aligned} & \int_{\Omega} \phi_i \left[S_0^p \phi_j \frac{dp_j}{dt} + \frac{\partial \phi_j}{\partial x_\alpha} \frac{k_{\alpha\beta}}{\mu} \frac{\partial \phi_j}{\partial x_\beta} p_j \right] d\Omega = \\ & - \int_{\Omega} \phi_i \frac{\partial \phi_j}{\partial x_\alpha} \frac{k_{\alpha\beta}}{\mu} \rho g \frac{\partial \phi_j}{\partial x_\beta} z_j d\Omega - \int_{\partial\Omega} \phi_i q_n dS + \int_{\Omega} \phi_i Q_\rho d\Omega \end{aligned} \quad (7.125)$$

The above equation forms a global system of equations where the number of equations corresponds to the number of grid points.

$C_{ij} \frac{dp_j}{dt} + K_{ij} p_j = r_i \quad , \quad i, j = 1, \dots, np$

(7.126)

with

$$\begin{aligned} C_{ij} &= \int_{\Omega} \phi_i S_0^p \phi_j d\Omega \\ K_{ij} &= \int_{\Omega} \frac{\partial \phi_i}{\partial x_\alpha} \frac{k_{\alpha\beta}}{\mu} \frac{\partial \phi_j}{\partial x_\beta} d\Omega \\ r_i &= -\rho g K_{ij} z_i - \int_{\partial\Omega} \phi_i q_n dS + \int_{\Omega} \phi_i Q_\rho d\Omega \end{aligned} \quad (7.127)$$

Finite Element Approach

Decomposition of the computation domain into finite elements

$$\Omega = \bigcup_{e=1}^{ne} \Omega^e \quad , \quad \partial\Omega = \bigcup_{e=1}^{ne} \partial\Omega^e \quad (7.128)$$

means that the global matrices can be composed by its element contributions.

$$\begin{aligned} C_{ij} &= \sum_{e=1}^{ne} C_{ij}^e \\ K_{ij} &= \sum_{e=1}^{ne} K_{ij}^e \\ r_i &= \sum_{e=1}^{ne} r_i^e \end{aligned} \quad (7.129)$$

Interpolation functions correspond now to individual elements and shape functions are used for interpolation at the element level.

$$\hat{p}^e = N_j(x_\alpha)p_j(t) \quad , \quad N_j = \begin{cases} 1 & , \quad j = i \\ 0 & , \quad j \neq i \end{cases} \quad (7.130)$$

$$\begin{aligned} C_{ij}^e &= \int_{\Omega^e} N_i S_0^p N_j d\Omega \\ K_{ij}^e &= \int_{\Omega^e} \frac{\partial N_i}{\partial x_\alpha} \frac{k_{\alpha\beta}}{\mu} \frac{\partial N_j}{\partial x_\beta} d\Omega^e \\ r_i^e &= -\rho g K_{ij}^e z_i - \int_{\partial\Omega^e} N_i q_n dS + \int_{\Omega^e} N_i Q_\rho d\Omega \end{aligned} \quad (7.131)$$

Evaluation of Element Matrices in Local Coordinates

Transformation from physical (x, y, z) to local (r, s, t) coordinates, e.g. by isoparametric functions, allows to evaluate the element matrices in unit coordinates.

$$C_{ij}^e = \int_{\Omega^e} N_i S_0^p N_j d\Omega \quad (7.132)$$

$$\begin{aligned} \mathbf{C}^e &= \int_{\Omega^e} \mathbf{N} S_0^p \mathbf{N} d\Omega \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} S_0^p \mathbf{N} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \end{aligned}$$

$$K_{ij}^e = \int_{\Omega^e} \frac{\partial N_i}{\partial x_\alpha} \frac{k_{\alpha\beta}}{\mu} \frac{\partial N_j}{\partial x_\beta} d\Omega^e \quad (7.133)$$

$$\begin{aligned} \mathbf{K}^e &= \int_{\Omega^e} \nabla \mathbf{N} \frac{1}{\mu} \mathbf{k} \nabla \mathbf{N}^T d\Omega^e \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \underbrace{\nabla \mathbf{N} (\mathbf{J}^{-1})^T}_{\nabla \mathbf{N}(x,y,z)} \frac{1}{\mu} \underbrace{\mathbf{T}^T \mathbf{k} \mathbf{T}}_{\mathbf{k}(x,y,z)} \underbrace{\nabla \mathbf{N}^T \mathbf{J}^{-1}}_{\nabla \mathbf{N}^T(x,y,z)} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \end{aligned}$$

$$r_i^e = -\rho g K_{ij}^e z_i - \int_{\partial\Omega^e} N_i q_n dS + \int_{\Omega^e} N_i Q_\rho d\Omega \quad (7.134)$$

$$\mathbf{r}^e = -\rho g \mathbf{K}^e \mathbf{z} - \int_{\partial\Omega^e} \mathbf{N} q_n dS + \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} Q_\rho \det \mathbf{J} dr ds dt$$

with \mathbf{J} the Jacobian matrix and \mathbf{T} the ordinary coordinate transformation matrix.

More details of finite element formulations for subsurface problems, in particular evaluation of element matrices for different flow and transport elements, can be found in Kolditz et al. (2001).

7.7 Transport Equation

7.7.1 Unsteady Transport Equation

We start from the differential form of the balance equation of mass transport (3.40). The weighted residual method applied to this equation yields the following weak formulation for the unknown functions.

Galerkin Method

$$\int_{\Omega} \omega \left[\frac{\partial(n\hat{C})}{\partial t} + \frac{\partial(q_{\alpha}\hat{C})}{\partial x_{\alpha}} - \frac{\partial}{\partial x_{\alpha}} \left(nD_{\alpha\beta} \frac{\partial\hat{C}}{\partial x_{\beta}} \right) \right] d\Omega = \int_{\Omega} \omega(nQ_C) d\Omega \quad (7.135)$$

where ω are weighting functions, and Ω is the solution domain. Approximate solutions of the unknown functions are specified by a hat.

After applying partial differentiation, we use Gauss-Ostrogradskian integral theorem to reduce second-order derivatives

$$\begin{aligned} \int_{\Omega} \left[\omega \frac{\partial(n\hat{C})}{\partial t} + \omega \frac{\partial(q_{\alpha}\hat{C})}{\partial x_{\alpha}} + \frac{\partial\omega}{\partial x_{\alpha}} \left(nD_{\alpha\beta} \frac{\partial\hat{C}}{\partial x_{\beta}} \right) \right] d\Omega = \\ \int_{\Gamma} \omega q_n d\Gamma + \int_{\Omega} \omega(nQ_C) d\Omega \end{aligned} \quad (7.136)$$

with tracer flux vector

$$q_n = nD_{\alpha\beta} \frac{\partial\hat{C}}{\partial x_{\beta}} n_{\alpha} \quad (7.137)$$

where $\partial\Omega$ is the segment of the domain boundary at which Neumann or Cauchy (flux type) boundary conditions have to be specified.

$$\begin{aligned} \int_{\Omega} \left[\omega \frac{\partial(n\hat{C})}{\partial t} + \omega \frac{\partial(q_{\alpha}\hat{C})}{\partial x_{\alpha}} + \frac{\partial\omega}{\partial x_{\alpha}} \left(nD_{\alpha\beta} \frac{\partial\hat{C}}{\partial x_{\beta}} \right) \right] d\Omega = \\ \int_{\Gamma} \omega q_n d\Gamma + \int_{\Omega} \omega(nRd \frac{A}{V} (C_s - \hat{C})) d\Omega \end{aligned} \quad (7.138)$$

$$\int_{\Omega} \left[\omega n \frac{\partial \hat{C}}{\partial t} + \omega \hat{C} \frac{\partial n}{\partial t} + \omega \frac{\partial(q_{\alpha} \hat{C})}{\partial x_{\alpha}} + \frac{\partial \omega}{\partial x_{\alpha}} \left(n D_{\alpha\beta} \frac{\partial \hat{C}}{\partial x_{\beta}} \right) \right] d\Omega + \int_{\Omega} \omega (n R d \frac{A}{V} \hat{C}) d\Omega = \int_{\Gamma} \omega q_n d\Gamma + \int_{\Omega} \omega (n R d \frac{A}{V} C_s) d\Omega \quad (7.139)$$

The unknown function is approximated by a trial solution

$$\hat{C}(t, x_{\alpha}) = \sum_{i=1}^{np} \phi_i(x_{\alpha}) C_i(t) = \phi_i(x_{\alpha}) C_i(t) \quad (7.140)$$

where ϕ_i are interpolation functions, and np is the number of grid points. Indices denote values of the unknown functions at the grid points.

According to the Galerkin method we use identical weighting ω_i and interpolation functions ϕ_i .

$$\int_{\Omega} \left[\phi_i n \phi_j \frac{dC_j}{dt} + \phi_i \frac{\partial n}{\partial t} \phi_j C_j + \phi_i \frac{\partial(q_{\alpha} \phi_j)}{\partial x_{\alpha}} C_j + \frac{\partial \phi_i}{\partial x_{\alpha}} (n D_{\alpha\beta} \frac{\partial \phi_j}{\partial x_{\beta}} C_j) \right] d\Omega + \int_{\Omega} \phi_i (n R d \frac{A}{V}) \phi_j C_j d\Omega = \int_{\Gamma} \phi_i q_n d\Gamma + \int_{\Omega} \phi_i (n R d \frac{A}{V} C_s) d\Omega \quad (7.141)$$

Algebraic equations

If rearranging the above equations to put unknowns terms to the left hand side only, they form system of algebraic equations with respect to grid values of the unknown functions.

$$C_{ij} \frac{dC_j}{dt} + (B_{ij} + D_{ij} + E_{ij} + F_{ij}) C_j = r_j \quad i, j = 1, \dots, np \quad (7.142)$$

with

C _{ij} =	$\int_{\Omega} \phi_i n \phi_j d\Omega$	Tracer capacitance matrix
B _{ij} =	$\int_{\Omega} \phi_i \frac{\partial(q_{\alpha} \phi_j)}{\partial x_{\alpha}} d\Omega$	Tracer advection matrix
D _{ij} =	$\int_{\Omega} \frac{\partial \phi_i}{\partial x_{\alpha}} (n D_{\alpha\beta} \frac{\partial \phi_j}{\partial x_{\beta}}) d\Omega$	Tracer diffusion matrix
E _{ij} =	$\int_{\Omega} \phi_i \frac{\partial n}{\partial t} \phi_j d\Omega$	
F _{ij} =	$\int_{\Omega} \phi_i (n R d \frac{A}{V}) \phi_j d\Omega$	Tracer solution matrix
r _j =	$\int_{\Gamma} \phi_i q_n d\Gamma + \int_{\Omega} \phi_i (n R d \frac{A}{V} C_s) d\Omega$	RHS vector

(7.143)

Finite Element Approach

Decomposition of the computation domain into finite elements

$$\Omega = \bigcup_{e=1}^{ne} \Omega^e \quad (7.144)$$

means that the global matrices can be composed by its element contributions.

Interpolation functions correspond now to individual elements and shape functions are used for interpolation at the element level.

$$\hat{C}^e(x_\alpha, t) = \sum_{j=1}^n N_j(x_\alpha) C_j(t) \quad (7.145)$$

where n is the node number per element.

Element matrices:

$$\begin{aligned} C_{ij}^e &= \int_{\Omega^e} N_i n N_j d\Omega^e \\ B_{ij}^e &= \int_{\Omega^e} N_i q_\alpha \frac{\partial N_j}{\partial x_\alpha} d\Omega^e \\ D_{ij}^e &= \int_{\Omega^e} \frac{\partial N_i}{\partial x_\alpha} (n D_{\alpha\beta}) \frac{\partial N_j}{\partial x_\beta} d\Omega^e \\ E_{ij}^e &= \int_{\Omega^e} N_i \frac{dn}{dt} N_j d\Omega^e \\ F_{ij}^e &= \int_{\Omega^e} N_i (n R d \frac{A}{V}) N_j d\Omega^e \\ r_j^e &= \int_{\Gamma^e} N_i q_n d\Gamma^e + \int_{\Omega^e} N_i (n R d \frac{A}{V} C_s) d\Omega^e \end{aligned} \quad (7.146)$$

Transformation to Local Element Coordinates

Transformation from physical (x, y, z) to local (r, s, t) coordinates, e.g. by isoparametric functions, allows to evaluate the element matrices in unit coordinates.

$$\begin{aligned} C_{ij}^e &= \int_{\Omega^e} N_i n N_j d\Omega^e \\ \mathbf{C}^e &= \int_{\Omega^e} \mathbf{N} n \mathbf{N} d\Omega^e \end{aligned}$$

$$= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} n \mathbf{N} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \quad (7.147)$$

$$\begin{aligned} B_{ij}^e &= \int_{\Omega^e} N_i q_\alpha \frac{\partial N_j}{\partial x_\alpha} d\Omega^e \\ \mathbf{B}^e &= \int_{\Omega^e} \mathbf{N} \mathbf{q} \nabla \mathbf{N}^T d\Omega^e \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} \underbrace{\mathbf{T} \mathbf{q}}_{\mathbf{q}(x,y,z)} \underbrace{\nabla \mathbf{N}^T \mathbf{J}^{-1}}_{\nabla \mathbf{N}^T(x,y,z)} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \end{aligned} \quad (7.148)$$

$$\begin{aligned} D_{ij}^e &= \int_{\Omega^e} \frac{\partial N_i}{\partial x_\alpha} (n D_{\alpha\beta}) \frac{\partial N_j}{\partial x_\beta} d\Omega^e \\ \mathbf{D}^e &= \int_{\Omega^e} \nabla \mathbf{N} (n \mathbf{D}^T) \nabla \mathbf{N}^T d\Omega^e \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \underbrace{\nabla \mathbf{N} (\mathbf{J}^{-1})^T}_{\nabla \mathbf{N}(x,y,z)} \underbrace{\mathbf{T}^T (n \mathbf{D}^T) \mathbf{T}}_{n \mathbf{D}^T(x,y,z)} \underbrace{\nabla \mathbf{N}^T \mathbf{J}^{-1}}_{\nabla \mathbf{N}^T(x,y,z)} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \end{aligned} \quad (7.149)$$

$$\begin{aligned} E_{ij}^e &= \int_{\Omega^e} N_i \frac{dn}{dt} N_j d\Omega^e \\ \mathbf{E}^e &= \int_{\Omega^e} \mathbf{N} \frac{dn}{dt} \mathbf{N} d\Omega^e \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} \frac{dn}{dt} \mathbf{N} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \end{aligned} \quad (7.150)$$

$$\begin{aligned} F_{ij}^e &= \int_{\Omega^e} N_i (n R d \frac{A}{V}) N_j d\Omega^e \\ \mathbf{F}^e &= \int_{\Omega^e} \mathbf{N} (n R d \frac{A}{V}) \mathbf{N} d\Omega^e \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} (n R d \frac{A}{V}) \mathbf{N} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \end{aligned} \quad (7.151)$$

$$\begin{aligned} r_j^e &= \int_{\Gamma^e} N_i q_n d\Gamma^e + \int_{\Omega^e} N_i (n R d \frac{A}{V} C_s) d\Omega^e \\ \mathbf{r}^e &= \int_{\Gamma^e} \mathbf{N} q_n d\Gamma^e + \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} (n R d \frac{A}{V} C_s) \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \end{aligned} \quad (7.152)$$

with \mathbf{J} the Jacobian matrix and \mathbf{T} the ordinary coordinate transformation matrix.

7.8 Problems

Domain Discretization (Section 7.1)

- 1 What are basic elements for domain discretization in several dimensions ?

Equation Discretization (Section 7.2)

- 2 Explain the difference between variational methods and weighted residual methods for discretization of differential equations.
- 3 What is the role of weighting functions for with respect to different discretization methods. Hint: use Tab. 7.1.
- 4 Derive eqn (7.123) from eqn (7.122). Hint: Use the chain rule for partial differentiation.

Interpolation and Shape Functions (Section 7.3)

- 5 Explain the difference between interpolation and shape functions.
- 6 Explain the terms global (physical) and local (natural) coordinates of elements. What is isoparametric mapping of element coordinates ?

Element Shape Functions in Local Coordinates (Section 7.4)

- 7 Calculate shape functions \mathbf{N} and its derivatives $\nabla \mathbf{N}$ for a 1-D linear element in point $r = 1/3$
- 8 Calculate shape functions \mathbf{N} and its derivatives $\nabla \mathbf{N}$ for a 1-D quadratic element in point $r = 1/3$
- 9 Calculate shape functions \mathbf{N} and its derivatives $\nabla \mathbf{N}$ for a 2-D bilinear element in Gaussian point $r = 1/\sqrt{3}, s = -1/\sqrt{3}$

Coordinate Transformation using the Jacobian Matrix (Sections 7.4, 7.5)

- 10 Calculate the Jacobian J_{1D} as well as the inversion J_{1D}^{-1} for a 1-D linear element $P_1 = (0, 0, 0)$ and $P_2 = (1, 2, 3)$ in point $r = 1/3$.
- 11 Calculate the Jacobian J_{2D} as well as the inversion J_{2D}^{-1} for a 2-D linear element in Gaussian point $r = 1/\sqrt{3}, s = -1/\sqrt{3}$.

$$[A'_{2D}]^T = \begin{bmatrix} 0 & 2 & 2 & 0 \\ 0 & 0 & 2 & 1 \end{bmatrix} \quad (7.153)$$

- 12** Calculate the Jacobian J_{2D} as well as the inversion J_{2D}^{-1} for a 2-D bilinear element in Gaussian point $r = 0, s = 0$.

$$[A_{3D}]^T = \begin{bmatrix} +1 & +1 & -1 & -1 \\ -0.866 & +0.866 & +0.866 & -0.866 \\ -0.35 & +0.65 & +0.65 & -0.35 \end{bmatrix} \quad (7.154)$$

- 13** Calculate the Jacobian J_{3D} as well as the inversion J_{3D}^{-1} for a 3-D trilinear element in Gaussian point $r = 0, s = 0, t = 0$.

$$[A_{3D}]^T = \begin{bmatrix} +1 & +1 & -1 & -1 & +1 & +1 & -1 & -1 \\ -1 & +1 & +1 & -1 & -1 & +1 & +1 & -1 \\ -1 & -1 & -1 & -1 & +1 & +1 & +1 & +1 \end{bmatrix} \quad (7.155)$$

Element Matrices (Section 7.6)

- 14** Derive the flow capacitance and conductance matrices for 1-D linear elements.
- 15** Derive the flow capacitance and conductance matrices for 2-D linear triangular elements.
- 16** Derive the flow capacitance and conductance matrices for 2-D bilinear quadrilateral elements.
- 17** Derive the flow capacitance and conductance matrices for 3-D tetrahedral elements.
- 18** Derive the flow capacitance and conductance matrices for 3-D triangle based prismatic elements.
- 19** Derive the flow capacitance and conductance matrices for 3-D trilinear hexahedral elements.
- 20** Derive the mass and heat advection matrices for 1-D linear elements.
- 21** Derive the mass and heat advection matrices for 2-D linear triangular elements.
- 22** Derive the mass and heat advection matrices for 2-D bilinear quadrilateral elements.
- 23** Derive the mass and heat advection matrices for 3-D tetrahedral elements.
- 24** Derive the mass and heat advection matrices for 3-D triangle based prismatic elements.
- 25** Derive the mass and heat advection matrices for 3-D trilinear hexahedral elements.

Element Resultants

- 26** Derive the velocity vector (fluid mass flux) for 1-D linear elements.
- 27** Derive the velocity vector (fluid mass flux) for 2-D linear triangular elements.
- 28** Derive the velocity vector (fluid mass flux) for 2-D bilinear quadrilateral elements.
- 29** Derive the velocity vector (fluid mass flux) for 3-D tetrahedral elements.
- 30** Derive the velocity vector (fluid mass flux) for 3-D triangle based prismatic elements.
- 31** Derive the velocity vector (fluid mass flux) for 3-D trilinear hexahedral elements.

Solving Equation Systems

- 32** Conduct the numerical analysis of steady-state groundwater flow through a vertical column (solving the equation system manually).
- 33** Conduct the numerical analysis of steady-state groundwater flow through an inclined column.
- 34** Conduct the numerical analysis of transient groundwater flow through a vertical column (solving the equation system manually).
- 35** Conduct the stability analysis for 1-D saturated groundwater flow.

Chapter 8

Finite Volume Method

The Finite Volume Method (FVM) was introduced into the field of computational fluid dynamics in the beginning of the seventies (McDonald 1971, MacCormack and Paullay 1972). From the physical point of view the FVM is based on balancing fluxes through control volumes, i.e. the Eulerian concept is used (see section 1.1.4). The integral formulation of conservative laws are discretized directly in space. From the numerical point of view the FVM is a generalization of the FDM in a geometric and topological sense, i.e. simple finite volume schemes can be reduced to finite difference schemes. The FDM is based on nodal relations for differential equations, whereas the FVM is a discretization of the governing equations in integral form. The Finite Volume Method can be considered as specific subdomain method as well. FVM has two major advantages: First, it enforces conservation of quantities at discretized level, i.e. mass, momentum, energy remain conserved also at a local scale. Fluxes between adjacent control volumes are directly balanced. Second, finite volume schemes takes full advantage of arbitrary meshes to approximate complex geometries. Experience shows that non-conservative schemes are generally less accurate than conservative ones, particularly in the presence of strong gradients.

We recall the general conservation equation (1.13) which is the base for the following discussion.

$$\int_{\Omega} \frac{\partial \psi}{\partial t} d\Omega = - \int_{\Omega} \nabla \cdot \Phi^{\psi} d\Omega + \int_{\Omega} q^{\psi} d\Omega \quad (8.1)$$

This general balance equation states that the time variation of a scalar quantity t inside the volume Ω only depends on the surface values of fluxes. Therefore, the global conservation within Ω can be recovered by adding up subvolume conservation expressions for arbitrary subdivision of Ω . When summing the

surface integrals (i.e. fluxes) the contributions of internal surfaces always appear twice but with opposite signs and therefore will vanish.

The FVM easily allows the introduction of natural boundary conditions (Neumann type). If normal flux components are zero, i.e. $\Phi \cdot d\mathbf{S} = 0$, the corresponding contributions to the discrete equation would vanish.

8.1 Discretization Process

Now the general balance equation (8.1) is replaced by the discrete form

$$\frac{\partial \Omega_j \bar{u}_j}{\partial t} + \sum_{\partial\Omega_j} \bar{\Phi}^u \cdot \mathbf{S} = \Omega_j \bar{q}_j^u \quad (8.2)$$

where \bar{u}_j is an average value of cell, which is not necessarily attached to a fixed point within Ω_j . The first term denotes the time rate of change of the averaged variable over the control volume. Note that mesh coordinates appear only for determination of cell volume and boundary. The sum of flux terms corresponds to all external sides of the cell. The above finite volume formulation states that in absence of source terms the variation of the average variable \bar{u}_j over a time interval Δt is equal to the sum of fluxes exchanged between adjacent cells. Therefore, we have to evaluate the following quantities:

$\frac{\partial}{\partial t}$	Time derivatives	see section 8.2
$\Omega, \partial\Omega$	Cell area/volume	see section 8.3
$\Phi \cdot \mathbf{S}$	Flux terms	see section 8.4

The FVM can handle any type of mesh and has, therefore, the same flexibility as the FEM. It is applicable to structured as well as unstructured meshes. Structured meshes are finite difference type one which consist of two sets of lines. Unstructured meshes are arbitrary combinations of triangles, quadrilaterals in a 2-D case and tetrahedrons, hexahedra or pyramids in 3-D. Elements of those meshes can not be numbered subsequently but have to be numbered individually. Unstructured meshes offer a large flexibility for complex geometric configurations.

In general we distinguish between cell-centered and cell-vertex methods. In cell-centered FVM the variables are attached to a cell. They are averaged values over the cell which coincides with the control volume (Fig. 8.1). In the cell-vertex FVM the variables are associated with a mesh point (Fig. 8.2). Control volumes can be defined individually, different choices are possible.

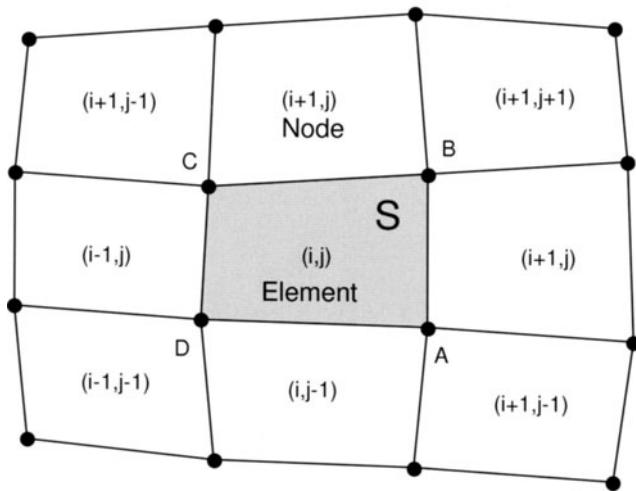


Figure 8.1: Mesh for cell-centered finite volumes

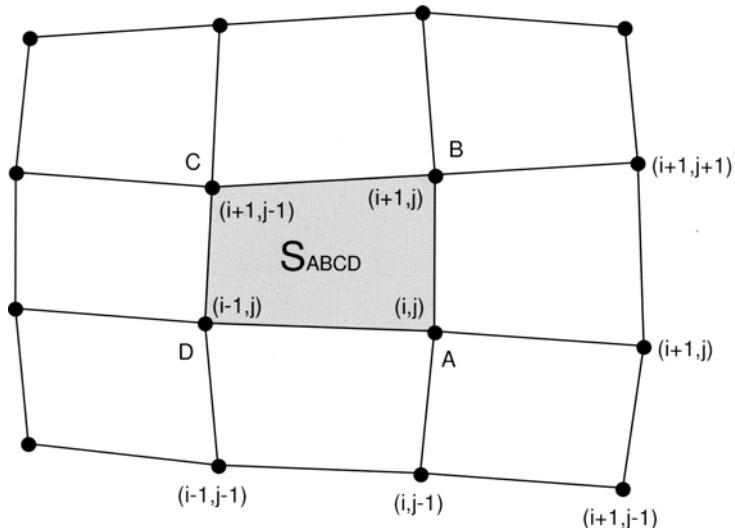


Figure 8.2: Mesh for cell-vertex finite volumes

8.2 Evaluation of Time Derivatives

Time derivatives can be easily approximated using finite differences.

$$\frac{\partial \Omega_j \bar{u}_j}{\partial t} \approx \Omega_j \frac{\bar{u}_j^{n+1} - \bar{u}_j^n}{\Delta t} \quad (8.3)$$

8.3 Evaluation of Cell Area/Volume

In this section we show several methods to calculate the area/volume of finite volume cells. In particular we take advantage of the Gauss-Ostrogradskian divergence theorem (1.12).

- (1) The area of a quadrilateral can be calculated from the vector product of its diagonals

$$\Omega_j = \frac{1}{2} | \mathbf{x}_{AC} \times \mathbf{x}_{BD} | = \frac{1}{2} (\Delta x_{AC} \Delta y_{BD} - \Delta x_{BD} \Delta y_{AC}) \quad (8.4)$$

where x_A is the position vector of point A and so on, $\mathbf{x}_{AB} = \mathbf{x}_B - \mathbf{x}_A$ is a difference vector, $\Delta x_{AB} = x_B - x_A$ is the difference of x -coordinates (Fig. 8.3). The area expression will be positive if the points A, B, C, D are located counterclockwise.

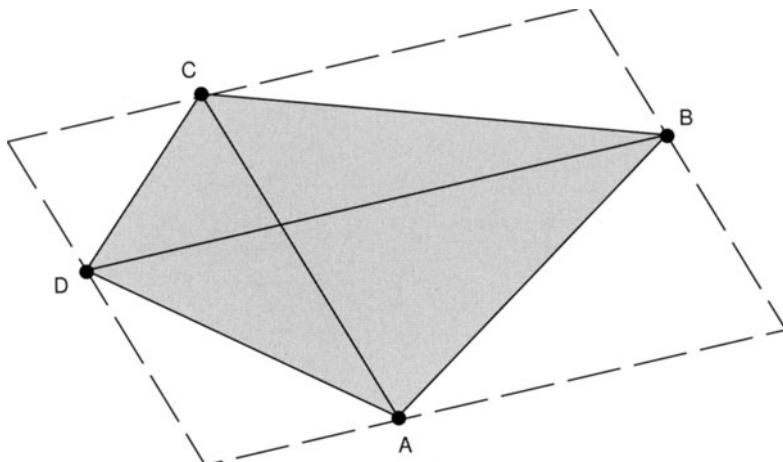


Figure 8.3: Area evaluation of finite volume cells

(2) The area of arbitrary 2-D cells can also be obtained by use of the divergence theorem. Setting for the unknown function $u = x$ we have for the x -component

$$\frac{1}{\Omega_j} \int_{\Omega_j} \frac{\partial x}{\partial x} d\Omega = \frac{1}{\Omega_j} \oint_{\partial\Omega_j} d\mathbf{e}_x d\mathbf{S} = \frac{1}{\Omega_j} \oint_{\partial\Omega_j} x dy = 1 \quad (8.5)$$

Therefore, the cell area can be calculated by

$$\Omega_j = \oint_{\partial\Omega_j} x dy = \sum_l \frac{x_l + x_{l+1}}{2} (y_{l+1} - y_l) \quad (8.6)$$

(3) Cell areas/volumes can be determined by use of the divergence theorem for vectors.

$$\int_{\Omega} \nabla \cdot \mathbf{A} d\Omega = \oint_{\partial\Omega} d\mathbf{A} \cdot d\mathbf{S} \quad (8.7)$$

Setting $\mathbf{A} = \mathbf{x} = (x, y)$ we have for the 2-D case

$$\begin{aligned} \int_{\Omega} \nabla \cdot \mathbf{x} d\Omega &= \int_{\Omega} \left(\frac{\partial x}{\partial x} + \frac{\partial y}{\partial y} \right) d\Omega = 2\Omega \\ &= \oint_{\partial\Omega} \mathbf{x} \cdot d\mathbf{S} = \oint_{\partial\Omega} (xdy - ydx) \end{aligned} \quad (8.8)$$

and therefore

$$\Omega = \frac{1}{2} \oint_{\partial\Omega} (xdy - ydx) \quad (8.9)$$

(4) Cell face area

In general, we can apply equation (8.4) to calculate the area of face $ABCD$ by using the auxiliary parallelogram shown in Fig. 8.3. An alternative of equation (8.4) is

$$\mathbf{S}_{ABCD} = \frac{1}{2} [(\mathbf{x}_{AB} \times \mathbf{x}_{BC}) + (\mathbf{x}_{CD} \times \mathbf{x}_{DA})] \quad (8.10)$$

or respectively

$$\mathbf{S}_{ABCD} = \frac{1}{2} [(\mathbf{x}_{BC} \times \mathbf{x}_{CD}) + (\mathbf{x}_{DA} \times \mathbf{x}_{AB})] \quad (8.11)$$

i.e. the surface vector is expressed as the average of the surface vectors of the two parallelograms constructed on the adjacent sides (AB, BC) and (CD, DA) or by opposite sides. The directions of the two normal vectors will not be identical since the triangles (ABC) and (CDA) are not in the same plane, i.e. if the four points are not forming a co-planar cell face. By averaging the above equations we obtain

$$\mathbf{S}_{ABCD} = \frac{1}{4}[(\mathbf{x}_{AB} + \mathbf{x}_{DC}) \times (\mathbf{x}_{BC} + \mathbf{x}_{AD})] \quad (8.12)$$

(5) Cell volume

The most current approach to evaluate hexahedral or more complex cell volumes consist in a subdivision into tetrahedrons or pyramids. The volume of a tetrahedron can be easily calculated from the 3-D version of equation (8.7), since $\nabla \cdot \mathbf{x} = 3$

$$\Omega = \frac{1}{3} \oint_{\partial\Omega} \mathbf{x} \cdot d\mathbf{S} = \frac{1}{3} \sum_{faces} \mathbf{x} \cdot \mathbf{S} = \frac{1}{3} \mathbf{x}_{PA} \cdot \mathbf{S}_{ABC} \quad (8.13)$$

Since \mathbf{x}_{PA} lies in a plane containing P , it is orthogonal to the connected surface vectors. The only remaining term of the above equation comes therefore from the plane ABC which is opposite to P .

$$\Omega = \frac{1}{6} \mathbf{x}_{PA} \cdot (\mathbf{x}_{AB} \times \mathbf{x}_{BC}) = \frac{1}{6} \begin{vmatrix} x_P & y_P & z_P & 1 \\ x_A & y_A & z_A & 1 \\ x_B & y_B & z_B & 1 \\ x_C & y_C & z_C & 1 \end{vmatrix} \quad (8.14)$$

8.4 Evaluation of Fluxes

The user has to define how to estimate cell volume and cell faces of the control volume and how to evaluate the fluxes through the cell faces. Flux through a face (except domain boundaries) contributes to 2 cells but with opposite signs. When calculating fluxes it is reasonable to add this contribution to flux balance of one cell and subtract it from the other one. This procedure guarantees global conservation.

Essentially, we can distinguish between central and upwind schemes. Central schemes corresponds to a local flux evaluation, whereas upwind schemes evaluate cell face flux according to the propagation direction of flux vector.

8.4.1 Two-Dimensional Schemes

We consider the cell-vertex mesh (Fig. 8.2) to express flux terms.

$$\sum_{\partial\Omega_j} \Phi^u \cdot \mathbf{S} \quad (8.15)$$

in Cartesian coordinates. To this purpose we need the Cartesian components of flux and surface vectors. The normal vector of side AB will be (Fig. 8.2)

$$\mathbf{n}_{AB} = \frac{\Delta y}{\sqrt{\Delta x^2 + \Delta y^2}} \mathbf{e}_x - \frac{\Delta x}{\sqrt{\Delta x^2 + \Delta y^2}} \mathbf{e}_y \quad (8.16)$$

Therefore, the surface vector is given by

$$\mathbf{S}_{AB} = \Delta y \mathbf{e}_x - \Delta x \mathbf{e}_y \quad (8.17)$$

Now we can write the two-dimensional general balance equation for cell Ω_j in following form

$$\int_{\Omega} \frac{\partial u}{\partial t} dt + \oint_{\partial\Omega} \Phi^u \cdot d\mathbf{S} = \frac{\partial \Omega_j \bar{u}_j}{\partial t} + \oint_{\partial\Omega} (\Phi_x^u dy - \Phi_y^u dx) = \int_{\Omega} q^u d\Omega = \Omega_j \bar{q}_j^u \quad (8.18)$$

We obtain the finite volume equation

$$\frac{\partial \Omega_j \bar{u}_j}{\partial t} + \sum_{\partial\Omega_j} \Phi^u \cdot \mathbf{S} = \frac{\partial \Omega_j \bar{u}_j}{\partial t} + \sum_{ABCD} (\Phi_x^u \Delta y - \Phi_y^u \Delta x) = \Omega_j \bar{q}_j^u \quad (8.19)$$

where the sum extends over the four sides of the quadrilateral $ABCD$.

8.4.2 Central Scheme on a Cartesian Mesh

We consider the cell-centered mesh shown in Fig. 8.4. By summing the flux contributions over the sides of the quadrilateral $ABCD$ the flux terms becomes

$$\sum_{ABCD} \Phi^u \cdot \mathbf{S} = \Phi_x |_{AB} \Delta y_{AB} + \Phi_y |_{BC} \Delta x_{BC} - \Phi_x |_{CD} \Delta y_{CD} - \Phi_y |_{DA} \Delta x_{DA} \quad (8.20)$$

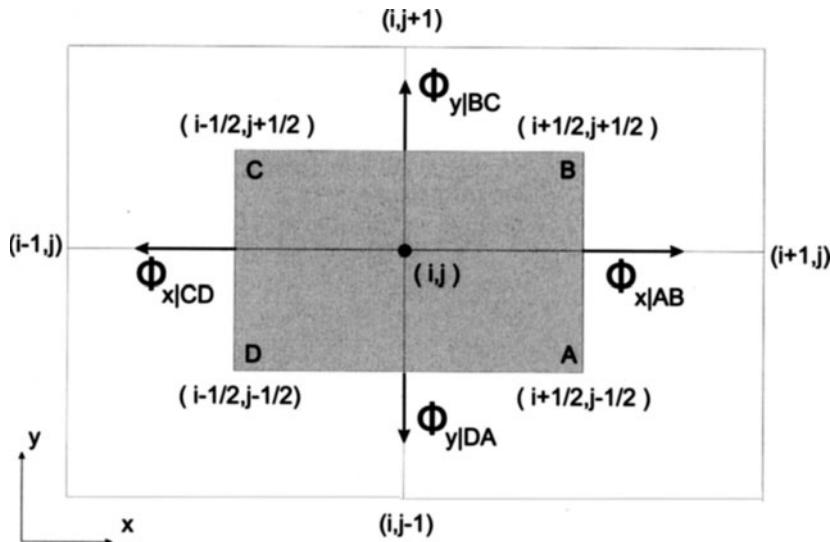


Figure 8.4: Flux evaluation for Cartesian cell-centered schemes

The points can be expressed by nodal indices of the Cartesian mesh (Fig. 8.4), e.g.

$$\Delta y_{AB} = y_B - y_A = y_{i+\frac{1}{2},j+\frac{1}{2}} - y_{i+\frac{1}{2},j-\frac{1}{2}} = \Delta y \quad (8.21)$$

If fluxes are taken at mid point, the flux term is given by

$$\begin{aligned} \sum_{ABCD} \Phi^u \cdot \mathbf{S} &= \Phi_x|_{i+\frac{1}{2},j} \Delta y + \Phi_y|_{i,j+\frac{1}{2}} \Delta x - \Phi_x|_{i-\frac{1}{2},j} \Delta y - \Phi_y|_{i,j-\frac{1}{2}} \Delta x \\ &= (\Phi_x|_{i+\frac{1}{2},j} - \Phi_x|_{i-\frac{1}{2},j}) \Delta y + (\Phi_y|_{i,j+\frac{1}{2}} - \Phi_y|_{i,j-\frac{1}{2}}) \Delta x \end{aligned} \quad (8.22)$$

Now we have to evaluate the flux components by nodal values. We use simple average expressions, e.g.

$$\Phi_x|_{i+\frac{1}{2},j} = \frac{1}{2}(\Phi_x|_{i+1,j} + \Phi_x|_{i-1,j}) \quad (8.23)$$

Dividing the expression by the area of the cell $\Omega_{ij} = \Delta x \Delta y$ we obtain a formula which is equivalent to a first-order central finite difference scheme (i.e. for rectangular meshes, see equation (6.10)).

$$\frac{\partial \bar{u}_{i,j}}{\partial t} + \frac{\Phi_x|_{i+1,j} - \Phi_x|_{i-1,j}}{2\Delta x} + \frac{\Phi_y|_{i,j+1} - \Phi_y|_{i,j-1}}{2\Delta y} = q_{i,j}^u \quad (8.24)$$

The advective flux vector is given by

$$\Phi = \begin{pmatrix} v_x u \\ v_y u \end{pmatrix} \quad (8.25)$$

Therefore, the corresponding finite volume scheme for the advection equation will be

$$\frac{\partial \bar{u}_{i,j}}{\partial t} + \frac{v_x|_{i+1,j} u_{i+1,j} - v_x|_{i-1,j} u_{i-1,j}}{2\Delta x} + \frac{v_y|_{i,j+1} u_{i,j+1} - v_y|_{i,j-1} u_{i,j-1}}{2\Delta y} = 0 \quad (8.26)$$

8.4.3 Upwind Scheme on a Cartesian Mesh

By using upwind schemes, we take the cell-side flux equal to the flux generated in the upstream cell ($\Omega_{i,j}$), i.e. the cell side flux is fully determined by this in the direction of advection.

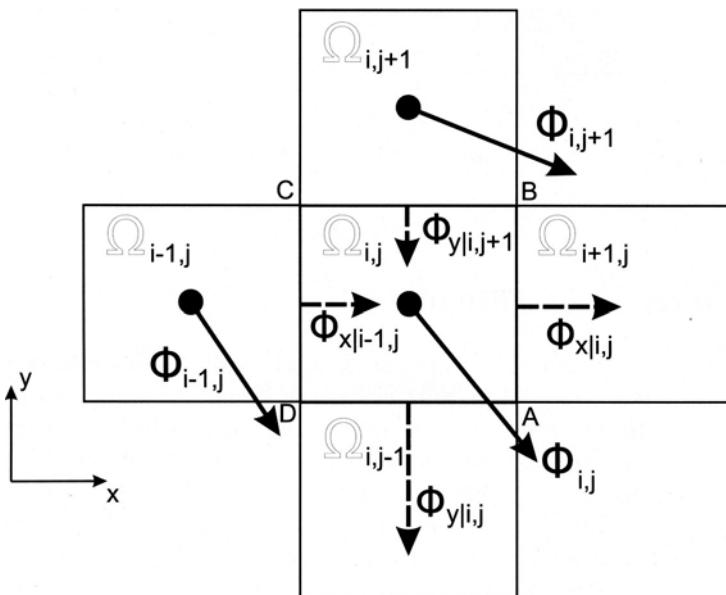


Figure 8.5: Upwind scheme for flux calculation

The individual flux components across the sides of element $\Omega_{i,j}$ for the cell-centered mesh (Fig. 8.5) will be

$$\begin{aligned} (\Phi \cdot \mathbf{S})|_{AB} &= \Phi_x|_{i,j} \Delta y \\ (\Phi \cdot \mathbf{S})|_{BC} &= -\Phi_y|_{i,j-1} \Delta x \end{aligned}$$

$$\begin{aligned} (\Phi \cdot \mathbf{S})|_{CD} &= -\Phi_x|_{i-1,j} \Delta y \\ (\Phi \cdot \mathbf{S})|_{DA} &= \Phi_y|_{i,j} \Delta x \end{aligned} \quad (8.27)$$

The resulting flux term is then

$$\begin{aligned} \sum_{ABCD} \Phi \cdot \mathbf{S} &= \Phi_x|_{i,j} \Delta y + \Phi_y|_{i,j-1} \Delta x - \Phi_x|_{i-1,j} \Delta y - \Phi_y|_{i,j} \Delta x \\ &= (\Phi_x|_{i,j} - \Phi_x|_{i-1,j}) \Delta y + (\Phi_y|_{i,j} - \Phi_y|_{i,j-1}) \Delta x \end{aligned} \quad (8.28)$$

Dividing the expression by the cell area $\Omega_{ij} = \Delta x \Delta y$ we obtain a formula which is equivalent to an upwind finite difference scheme which is first-order accurate in space.

$$\frac{\partial u_{ij}}{\partial t} + \frac{\Phi_x|_{i,j} - \Phi_x|_{i-1,j}}{\Delta x} + \frac{\Phi_y|_{i,j} - \Phi_y|_{i,j-1}}{\Delta y} = q_{i,j} \quad (8.29)$$

The corresponding finite volume scheme for the advection equation will be (see equation (6.63))

$$\frac{\partial u_{i,j}}{\partial t} + \frac{v_x|_{i,j} u_{i,j} - v_x|_{i-1,j} u_{i-1,j}}{\Delta x} + \frac{v_y|_{i,j} u_{i,j} - v_y|_{i,j-1} u_{i,j-1}}{\Delta y} = 0 \quad (8.30)$$

For a cell-vertex mesh the upwind scheme for flux evaluation is (Fig. 8.2)

$$(\Phi \cdot \mathbf{S})|_{AB} = \Phi|_{CD} \cdot \mathbf{S}|_{AB} \quad (8.31)$$

8.4.4 Integration Formulas

Frequently it is necessary to approximate numerically derivatives of unknown field variables (e.g. for diffusive or viscous flux components). In the case of the FVM, whether we need to define cell averages or gradients on cell faces. A general method to approximate spatial derivatives can be obtained again from the Gauss-Ostrogradskian divergence theorem (8.1). The divergence theorem can be used to define the average of the gradient of a scalar function u as a function of its values at the boundary of considered domain Ω . Averaged gradients can be defined as

$$\overline{\left[\frac{\partial u}{\partial x_i} \right]} = \frac{1}{\Omega} \int_{\Omega} \frac{\partial u}{\partial x_i} d\Omega = \frac{1}{\Omega} \oint_{\partial\Omega} u \mathbf{e}_i \cdot d\mathbf{S} \quad (8.32)$$

In two dimensions the above expression can be written as

$$\overline{\left[\frac{\partial u}{\partial x} \right]} = \frac{1}{\Omega} \int_{\Omega} \frac{\partial u}{\partial x} d\Omega = \frac{1}{\Omega} \oint_{\partial\Omega} u dy = -\frac{1}{\Omega} \oint_{\partial\Omega} y du \quad (8.33)$$

The last expression was obtained after partial differentiation. Considering cell-centered unstructured 2-D finite volume meshes (Fig. 8.1) we obtain the following equations

$$\overline{\left[\frac{\partial u}{\partial x} \right]} = \frac{1}{\Omega} \oint_{\partial\Omega} u dy = \frac{1}{\Omega} \sum_l \frac{u_l - u_{l+1}}{2} (y_{l+1} - y_l) = \frac{1}{2\Omega} \sum_l u_l (y_{l+1} - y_{l-1}) \quad (8.34)$$

where the trapezoidal rule was applied for integration. After rearranging the summation we have

$$\begin{aligned} \overline{\left[\frac{\partial u}{\partial x} \right]} &= \frac{1}{2\Omega} [u_A(y_B - y_D) + u_B(y_C - y_A) + u_C(y_D - y_B) + u_D(y_A - y_C)] \\ &= \frac{1}{2\Omega} [(u_A - u_C)(y_B - y_D) - (u_B - u_D)(y_A - y_C)] \end{aligned} \quad (8.35)$$

The area of quadrilateral $ABCD$ is given by equation (8.4).

$$\Omega_{ABCD} = \frac{1}{2} (\Delta x_{AC} \Delta y_{BD} - \Delta x_{BD} \Delta y_{AC}) \quad (8.36)$$

Finally, we have the following expression for the spatial derivation term

$$\overline{\left[\frac{\partial u}{\partial x} \right]} = \frac{(u_A - u_C)(y_B - y_D) - (u_B - u_D)(y_A - y_C)}{\Delta x_{AC} \Delta y_{BD} - \Delta x_{BD} \Delta y_{AC}} \quad (8.37)$$

8.5 Diffusion Equation

We consider the two-dimensional diffusion equation

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} (\alpha \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y} (\alpha \frac{\partial u}{\partial y}) = 0 \quad (8.38)$$

with the diffusive flux components

$$\Phi = \begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = \alpha \begin{pmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \end{pmatrix} \quad (8.39)$$

Using the general flux evaluation according to equation (8.20), the cell-centered scheme for a Cartesian mesh (Fig. 8.1) can written for cell (i, j) as follows.

$$\begin{aligned} \Delta x \Delta y \left[\frac{\partial u}{\partial t} \right]_{i,j} + \Phi_x|_{AB} \Delta y + \Phi_y|_{BC} \Delta x - \Phi_x|_{CD} \Delta y - \Phi_y|_{DA} \Delta x \\ = 0 \quad (8.40) \end{aligned}$$

A simple choice for the flux term can be obtained for a Cartesian mesh by using forward differences.

$$\Phi_x|_{AB} = \alpha \left[\frac{\partial u}{\partial x} \right]_{AB} = \alpha \frac{u_{i+1,j} - u_{i,j}}{\Delta x} \quad (8.41)$$

The resulting finite volume scheme

$$\left[\frac{\partial u}{\partial t} \right]_{i,j} + \alpha \frac{u_{i+1,j} + u_{i,j-1} + u_{i-1,j} + u_{i,j+1} - 4u_{i,j}}{\Delta x^2} = 0 \quad (8.42)$$

is identical to the five-point finite difference scheme.

Alternatively, derivatives of u for a point can be taken as averages over the adjacent cells, i.e. for point A the cells $(i,j), (i,j-1), (i+1,j-1), (i+1,j)$ respectively (Fig. 8.1).

$$\Phi_x|_A = \alpha \left[\frac{\partial u}{\partial x} \right]_A = \frac{\alpha}{2\Delta x} (u_{i+1,j} - u_{i,j} + u_{i+1,j-1} - u_{i,j-1}) \quad (8.43)$$

and for point B

$$\Phi_x|_B = \alpha \left[\frac{\partial u}{\partial x} \right]_B = \frac{\alpha}{2\Delta x} (u_{i+1,j} - u_{i,j} + u_{i+1,j+1} - u_{i,j+1}) \quad (8.44)$$

If flux terms over a cell face are evaluated by averaging nodal values, e.g. for AB

$$\Phi_{AB} = \frac{1}{2}(\Phi_A + \Phi_B) \quad (8.45)$$

we finally obtain the following finite volume scheme for the two-dimensional diffusion equation on an equidistant mesh

$$\left[\frac{\partial u}{\partial t} \right]_{i,j} + \alpha \frac{u_{i+1,j+1} + u_{i+1,j-1} + u_{i-1,j+1} + u_{i-1,j-1} - 4u_{i,j}}{4\Delta x^2} = 0 \quad (8.46)$$

This scheme corresponds to a five-point finite difference scheme (Laplace operator B).

The calculation of fluxes along the sides of control volume depends on the selected scheme as well as on the location of the unknown variables with respect to the mesh. Additionally, several schemes to average fluxes are available. For example, taking the average of the fluxes in points A and B for a cell-vertex mesh, we obtain a trapezium formula for the flux term.

$$\oint_{ABCD} \Phi^u \cdot d\mathbf{x} = \frac{\Phi_A^u + \Phi_B^u}{2} (x_B - x_A) \quad (8.47)$$

By summing the flux contributions over the sides of the quadrilateral the flux terms becomes

$$\oint_{ABCD} \Phi^u \cdot d\mathbf{S} = \frac{1}{2} \left([\Phi_x(A) - \Phi_x(C)](y_D - y_B) + [\Phi_x(B) - \Phi_x(D)](y_A - y_C) - [\Phi_y(A) - \Phi_y(C)](x_D - x_B) - [\Phi_y(B) - \Phi_y(D)](x_A - x_C) \right) \quad (8.48)$$

Note, the above cell-vertex finite volume scheme is equivalent with the Galerkin method for linear triangular or bilinear quadrilateral elements.

8.6 Advection Equation

We consider the two-dimensional advection equation in divergence (conservative) form.

$$\frac{\partial u}{\partial t} + \frac{\partial v_x u}{\partial x} + \frac{\partial v_y u}{\partial y} = 0 \quad (8.49)$$

If we select $u = \rho$ the above equation corresponds to the two-dimensional continuity equation. After applying the Green-Ostrogradskian theorem the integral formulation is given by

$$\int_A \frac{\partial u}{\partial t} dA + \oint_L \mathbf{u} \cdot \mathbf{n} dL = 0 \quad (8.50)$$

As an example we consider two-dimensional finite volume cells shown in Fig. 8.6

In a first step the time derivation term has to be discretized. We assume the area A of quadrilateral $ABCD$ is not time dependent, we can write

$$\int_A \frac{\partial u}{\partial t} dA = \frac{\partial}{\partial t} \int_A u dA \approx A \frac{\partial \bar{u}_{j,k}}{\partial t} \quad (8.51)$$

where $\bar{u}_{j,k}$ is an average value associated with node (j, k) . The time derivation term can be discretized e.g. by a standard finite difference scheme according to equation (6.5). Next, the flux term has to be discretized. An approximate evaluation for advective flux across line AB would be

$$\int_A \mathbf{u} \cdot \mathbf{n} dL \approx \frac{1}{2}(u_{j,k-1} + u_{j,k})v_x(y_B - y_A) - \frac{1}{2}(u_{j,k-1} + u_{j,k})v_y(x_B - x_A) \quad (8.52)$$

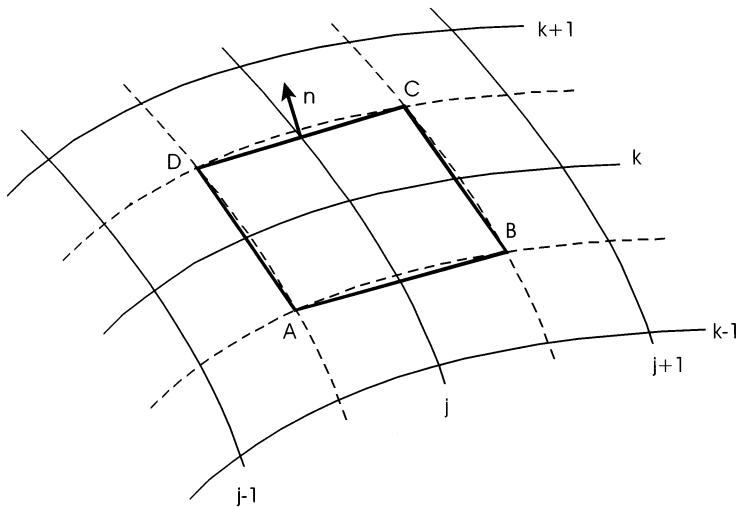


Figure 8.6: Finite volume discretization for cell-centered scheme

For the general case of an irregular grid the above finite volume equation provides a discretization in Cartesian coordinates.

$$\begin{aligned}
 A \frac{\partial u_{j,k}}{\partial t} + & \frac{1}{2}(u_{j,k-1} + u_{j,k})v_x(y_B - y_A) - \frac{1}{2}(u_{j,k-1} + u_{j,k})v_y(x_B - x_A) \\
 + & \frac{1}{2}(u_{j+1,k} + u_{j,k})v_x(y_C - y_B) - \frac{1}{2}(u_{j+1,k} + u_{j,k})v_y(x_C - x_B) \\
 + & \frac{1}{2}(u_{j,k+1} + u_{j,k})v_x(y_D - y_C) - \frac{1}{2}(u_{j,k+1} + u_{j,k})v_y(x_D - x_C) \\
 + & \frac{1}{2}(u_{j-1,k} + u_{j,k})v_x(y_A - y_D) - \frac{1}{2}(u_{j-1,k} + u_{j,k})v_y(x_A - x_D) \\
 = & 0
 \end{aligned} \tag{8.53}$$

If the grid is uniform (i.e. $\Delta x = \Delta y$) the above finite volume scheme coincides with a centered difference scheme for spatial derivatives.

$$\frac{\partial u_{j,k}}{\partial t} + v_x \frac{u_{j+1,k} - u_{j-1,k}}{2\Delta x} + v_y \frac{u_{j,k+1} - u_{j,k-1}}{2\Delta y} = 0 \tag{8.54}$$

Compare this expression with finite difference scheme (6.55).

8.7 Problems

- 1 Give the integral form of the general balance equation.
- 2 What are the basic cell types for finite volumes ?
- 3 Apply the Gauss-Ostrogradskian theorem (1.12) for determination of cell areas/volumes. Hint: Take $\psi = x$.
- 4 Derive the algebraic expression of eqn (8.6). Hint: Consider the flux vector onto a boundary edge in $x - y$ coordinates.
- 5 Explain the essential feature of upwind-schemes concerning fluxes into/from finite volume cells. Hint: Use Fig. 8.5 .
- 6 Apply the Gauss-Ostrogradskian theorem (1.12) for determination of cell-averaged derivatives:

$$\overline{\left[\frac{\partial u}{\partial x} \right]} = \frac{1}{\Omega} \int_{\Omega} \frac{\partial u}{\partial x} d\Omega = ?$$

- 7 Explain the conceptual difference between equations (6.63) and (8.30).

Bibliography

- [1] Baker (1983): *Finite element computational fluid mechanics*, McGraw-Hill, New York.
- [2] Chung T J (1978): *Finite element analysis in fluid dynamics*, McGraw-Hill, New York.
- [3] Cole J D (1951): *Q Appl Mech*, 1:171-219.
- [4] Cuthill E H & MacKee J (1969): Reducing the bandwidth of a sparse symmetric matrices. in *Proc 24th Nat Conf Assoc Comp Mech*, ACM Publ:157-172.
- [5] Diersch H-J G (1985): *Modellierung und numerische Simulation geohydrodynamischer Transportprozesse*, Habilitationsschrift, Akademie der Wissenschaften der DDR, Berlin.
- [6] Diersch H-J G (2001): *Feflow - Reference Manual*, WASY - Institute for Water Resources Planning and Systems Research, Berlin.
- [7] Donea J (1984): *A Taylor-Galerkin method for convective transport problems*, Int J Numer Meth Engng, 20:101-119.
- [8] Fletcher C A J (1990): *Computational techniques for fluid dynamics*, Springer series in Computational Physics (two volumes), Springer-Verlag.
- [9] Gibbs N E, Poole W G & Stockmeyer P K (1976): An algorithm for reducing the bandwidth and profile of a sparse matrix. *SIAM J Num Anal*, 13:236-250.
- [10] Habbar A (1995): *Vergleich verschiedener Krylov-Verfahren für allgemeine reguläre und sehr große lineare Gleichungssysteme*, Institut für Strömungsmechanik und Elektr. Rechnen im Bauwesen, Universität Hannover.
- [11] Hackbusch W (1991): *Iterative Lösung schwachbesetzter Gleichungssysteme*, Teubner-Verlag, Stuttgart.
- [12] Helmig R (1997): *Multiphase flow and transport in the subsurface*, Springer-Verlag.
- [13] Hopf E (1950): *Communications Pure Appl Mathematics*, 3:201-230.
- [14] Huyakorn P S & Pinder G F (1983): *Computational methods in subsurface flow*, Academic Press, New York - London.
- [15] Kaiser R, Kolditz O & Zielke W (1998): *Automatic grid adaption for subsurface fluid flow problems - Application to fractured-porous reservoirs*, In: Proc. XII international Conference on Computational Methods in Water Resources, pp 125-132, Crete, Greece, June 15-19, 1998.

- [16] Kasper H, Kosakowski G, Rother T, Thorenz C, Kolditz O & Taniguchi T (1998): *Development of a 3-D CAD system for numerical analysis of subsurface flow and transport*, In: Proc. 6th Int Conf on Num. Grid Generation in Comp. Field Sim., pp 683-694. Greenwich, UK, July 06-09, 1998.
- [17] Kinzelbach W (1992): *Numerische Methoden zur Modellierung des Transport von Schadstoffen im Grundwasser*, Oldenbourg-Verlag.
- [18] Knabner P & Angermann L (2000): *Numerik partieller Differentialgleichungen*, Springer-Verlag.
- [19] Kolditz O, Habbar A, Kaiser R, Rother T & Thorenz C (2001): *ROCKFLOW - Theory and Users Manual*, Release 3.5, Groundwater Modeling Group, Institute of Fluid Mechanics, University of Hannover, Internet publication, (www.rockflow.de).
- [20] Launder B E & Spalding D B (1974): *The numerical computation of turbulent flows*, Comp. Methods Appl. Mech. Eng., 3:269-289.
- [21] Lax PD & Wendroff B (1960): *Systems of conservation laws*, Comm Pure and Applied Mathematics, 13:217-237.
- [22] Lewis R W & Schrefler B A (1998): *The finite element method in the static and dynamic deformation and consolidation of porous media*, Wiley & Sons.
- [23] MacCormack R W & Paullay A J (1972): *Computational efficiency achieved by time splitting of finite difference operators*, AIAA paper 72-154, San Diego.
- [24] McDonald P W (1971): *The computation of transonic flow through two-dimensional gas turbine cascades*, ASME paper 71-GT-89.
- [25] Meyer A (1990): A parallel preconditioning conjugate gradient method using domain decomposition and inexact solvers on each subdomain. *Computing*, vol 45.
- [26] Mitchell A R & Griffiths D F (1980): *The finite difference method in partial differential equations*, Wiley, Chichester.
- [27] Noye J (1984): *Numerical solution of differential equations*, North-Holland, Amsterdam.
- [28] Pinder G F & Gray W G (1977): *Finite element simulation in surface and subsurface hydrology*, Academic Press, New York-London.
- [29] Schwetlick H & Kretschmar H (1991): *Numerische Verfahren für Naturwissenschaftler und Ingenieure*, Fachbuch-Verlag, Leipzig.
- [30] Turner M J, Clough R W, Martin H C & Topp L P (1956): *Stiffness and deflection analysis of complex structures*, J Aeron Soc, 23: 805.
- [31] Versteeg H K & Malalasekera W (1995): *An introduction to computational fluid dynamics: The finite volume method*, Longman Scientific & Technical.
- [32] Wriggers P (2001): *Nichtlineare Finite Elemente Methoden*, Springer-Verlag.

Part III

Software-Engineering

Chapter 9

Object-Oriented Methods for Hydrosystem Modeling

Object-oriented (OO) methods are a necessary tool in responding to many of the challenges in scientific computation, in particular, in managing modeling of complex systems such as multicomponent/multiphase processes in porous / fractured media. Use of OO methods can significantly reduce the effort to maintain and extend codes as requirements change. In addition, OO techniques offer means to increase code reuseability.

9.1 Introduction

Fluids play an important role in environmental systems appearing as surface water in rivers, lakes, and coastal regions or in the subsurface as well as in the atmosphere. Mechanics of environmental fluids is concerned with fluid motion as well as associated mass and heat transport (Zielke 1999).

Fig. 9.1 gives an overview on possible object orientation during system analysis process. Objects we understand as logic units characterized by properties (1), methods (2) applied to the object. Objects can communicate (3) with each other, e.g. transfer data or messages. On top of Fig. 9.1 we find the investigation object of interest here - subsurface environment. Properties of this object are the structure of the underlying geologic system and its inherent processes (flow, transport, deformation). Methods to investigate this object are experiments and mathematical modeling. Finally, objects are realized as data constructs (C implementation) or classes (C++ representation). These informatic objects are denoted by capital letters through the paper.

The paper is organized in that way to discuss the steps of system analysis (see

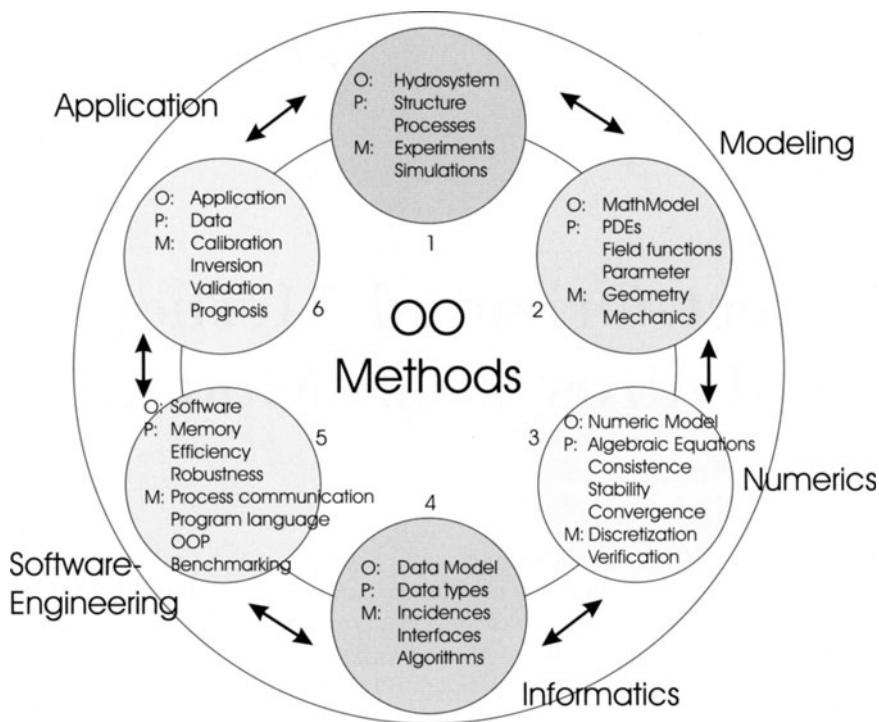


Figure 9.1: Object-oriented methods for system analysis

Fig. 9.1): modeling, numerics, informatics, software and application an object-oriented framework.

9.2 Investigation Objects

Concerning subsurface environments we have to consider both the geometric structure of the geologic system and physico-chemical processes within. In general exploration of geologic systems is conducted in an indirect manner, therefore, most data suffers from uncertainty.

9.2.1 Geometric modeling

The general idea of discrete geometric modeling is the hierarchical boundary representation concept (BREP). Any volumetric body (e.g. geologic formations) can be represented by its bounding surfaces, which can be described by

polylines. Polyline, finally, consists of points. Fundamental geometric objects are, therefore, points, polylines, surfaces and volumes. For discrete geometric modeling of surfaces the triangulated irregular network concept (TIN) is used. In this way planes as well as curved surfaces can be captured. This is important for flexibility to model complex aquifer systems (Fig. 9.2) as well as fractured hard rock (Fig. 9.3). The BREP allows efficient construction and modification of geometric models (Rother et al. 1998).

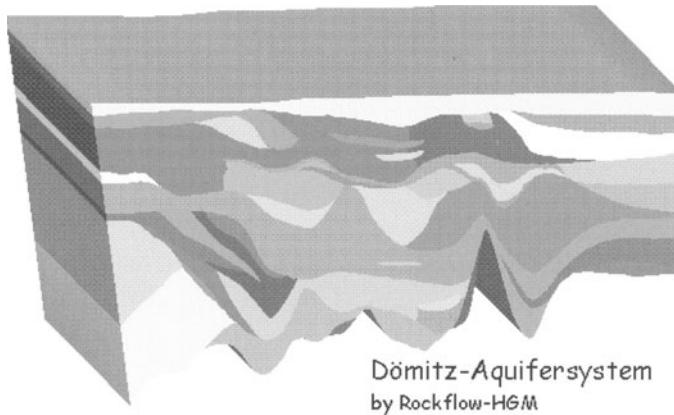


Figure 9.2: Geometric model of an aquifer system

Fracture systems are generated by consecutive intersection of individual fractures. Fracture orientation can be given by strike and dip angles. Additionally fracture extension must be specified. Otherwise, corresponding to the BREP a definition by bounding polygons is required. Data from in AutoCADs dxf-format can be used as input. For discrete geometric modeling TIN representations of each fracture are used. These TINs have to be intersected to generate the discrete fracture network model.

Two meshing methods are available: advancing front and Delaunay methods (Taniguchi et al. 1996). To improve mesh quality for finite element simulations remeshing and smoothing algorithms can be used (Fig. 9.3).

Complexity of multiphase systems and the marked heterogeneity is characteristic of subsurface environments. Heterogeneities often occur over multiple scales and result in a large set of problems that must be solved. Concerning the structure of subsurface systems we classify into porous, fractured or porous-fractured media.

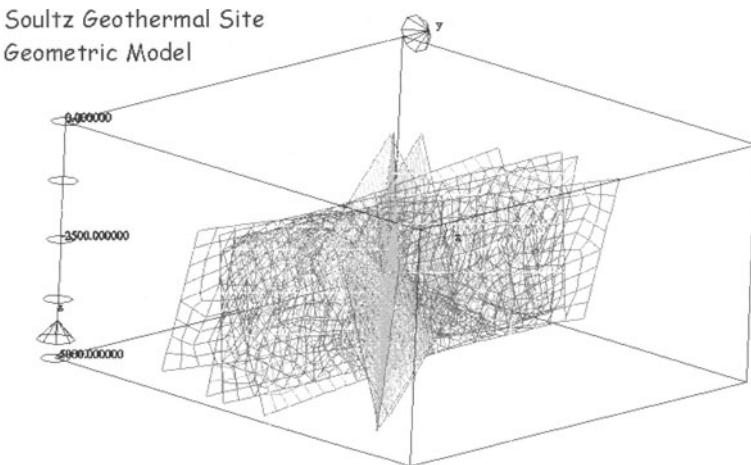


Figure 9.3: Fracture network model of a crystalline hard rock and finite element mesh

9.2.2 Processes

Processes of interest in subsurface systems are fluid flow, species and energy transport as well as deformation of the porous/fractured matrix. These processes can be more or less coupled (Fig. 9.4). Typical coupling phenomena are e.g. convection, consolidation, osmosis, and clogging.

In particular, the above mentioned disciplines have a special view point to the investigation object - subsurface system. In general a number of unknown functions (objects) have to be determined: fluid phase pressure, water saturation, solid matrix displacement, chemical component concentrations, temperature. All these objects are governed by conservation principles and corresponding material relationships with heterogeneities at different scales.

9.3 Mathematical Objects

The basic idea of continuum mechanics is that the evolution of a physical system is completely determined by conservation laws, i.e. basic properties such as mass, momentum, and energy are conserved during the considered process at all times. Any physical system can be completely determined by its conservation properties. In contrast, other quantities such as pressure or entropy do not obey conservation laws. The only additional information concerns the consistence of the material (e.g. fluids, solids, porous medium) in form of constitutive laws.

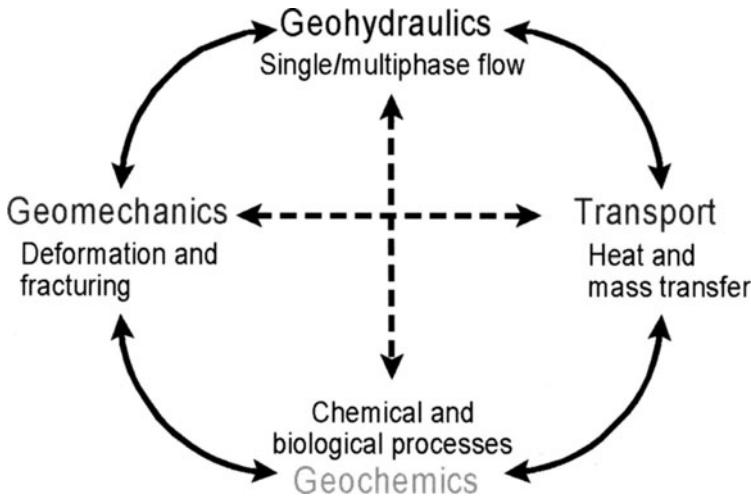


Figure 9.4: Coupled processes in subsurface systems

The concept of conservation means that the variation of a conservation quantity within a given control volume is due to the net effect of internal sources and of the amount of the quantity which is crossing the boundary surface of the considered volume - fluxes. Sources and fluxes are, in general, dependent of space-time coordinates as well as on mechanical and thermodynamic factors. Fluxes result from two contributions: first due to advective transport by fluid motion and second due to diffusion/dispersion processes. Diffusion is always present even when the fluid is at rest. Diffusion is the tendency towards equilibrium or homogeneity of a physical system.

The continuum-mechanical mathematical formulation (model development) relies on fundamental conservation principles and corresponding closure relations (equations of state (EOS) and constitutive relationships - material theory) (e.g. Bear & Bachmat 1990, Lewis & Schrefler 1998). From the viewpoint of object definition, we can say that each process can be described by a partial differential equation (PDE) which has to be specified with initial and boundary conditions. The general continuum balance in Eulerian form equation can be written as follows

$$\int_{\Omega} \frac{\partial u_i}{\partial t} d\Omega = \int_{\Omega} \nabla \cdot \Phi^{u_i} d\Omega + \int_{\Omega} q^{u_i} d\Omega \quad (9.1)$$

where u is an unknown function (i.e. conservation quantity - mass, momentum

or energy), Φ is flux term, and q is source term. All quantities are averaged over a represented elementary volume and describe macroscopic behaviour. At the same time they are related to specific phases. The first term on right hand side expresses macroscopic fluxes of the conservation quantities.

The following flux terms include equations of state (EOS) and constitutive relationships.

Fluid phase

- advective flux of multiple

$$u = \rho^\alpha \quad , \quad \Phi^u = \phi \rho^\alpha (\mathbf{v}^\alpha - \mathbf{v}^s) \quad (9.2)$$

$$\mathbf{q}^\alpha = -\frac{\mathbf{k}_0 k^{\alpha\beta}}{\mu^\alpha} (S^{\alpha\beta}) (\nabla p^\alpha - \rho^\alpha \mathbf{g}) \quad (9.3)$$

$$\mathbf{k}_0 = \frac{\phi^3}{C^\alpha \Sigma^{\alpha s}} \alpha^{-1} \mathbf{T}^* \quad (9.4)$$

- diffusive/dispersive mass fluxes

$$\mathbf{j}^\gamma = \mathbf{D}^\gamma \nabla C^\gamma \quad (9.5)$$

$$\mathbf{D}^\gamma = (\alpha_L^\gamma - \alpha_T^\gamma) \frac{\mathbf{v}\mathbf{v}}{v} + \alpha_T^\gamma v \mathbf{I} \quad (9.6)$$

- coupled fluxes (osmotic effects)

$$\mathbf{q}^\alpha = -\frac{\mathbf{k}_0 k^{\alpha\beta}}{\mu^\alpha} (S^{\alpha\beta}) (\nabla p^\alpha - \rho^\alpha \mathbf{g}) - \mathbf{D}^\gamma \nabla C^\gamma \quad (9.7)$$

Solid phase

$$\varepsilon = \varepsilon_{el} + \varepsilon_{in} \quad (9.8)$$

$$\sigma'^s = \mathbf{C} \varepsilon_{el} = 2G\varepsilon_{el} + \lambda \varepsilon_{el} \mathbf{I} \quad (9.9)$$

$$\varepsilon_{in} = k_s \frac{\partial S}{\partial \sigma'^s} \quad (9.10)$$

Equations of state express the equilibrium between pressure, density and temperature. These relations are typically based on thermodynamic models. Typical constitutive relations for multicomponent/multiphase systems concern capillary pressure / saturation / relative permeability, interphase mass transfer, chemical reaction, and stress-strain relationships. While the derivation of constitutive relationships from first principles is theoretically possible in some cases (Diersch 1985) they are usually empirically based. EOS and constitutive relationships increase both the number of equations and of unknown quantities. However, they supply sufficient information to close the system of equations by expressing unknown quantities in the governing equations in terms of known parameters.

From the object-oriented viewpoint we can separate following logic units which are required for each mathematical model: unknown functions, material properties (EOS and constitutive relationships), initial and boundary conditions.

9.4 Numeric Objects

There are many alternative methods to solve initial boundary value problems arising from flow and transport in subsurface systems. In general these methods can be classified into analytical and numerical ones. Analytical solutions can be obtained for a number of problems involving linear or quasi-linear equations for calculation domains of simple geometry. For non-linear equations or problems with complex geometry or boundary conditions, exact solutions usually do not exist, and approximate solutions must be obtained. For such problems the use of numerical methods is advantageous. The approximate solution procedure of initial-boundary-value-problems consists of two steps: discretization process and solution process.

9.4.1 Discretization Process

The finite element method (FEM) is employed for spatial discretization (Kolditz et al. 1998). Therefore, the finite element (ELEMENT) can be regarded as basic object of FEM. In addition, methods of characteristics and operator splitting techniques for advection dominated transport processes are developed (Thorenz et al. 1999). The Galerkin weighted residual approach is used to provide a weak formulation of the PDEs. This methodology is more general in application than variational methods. The Galerkin approach works also for problems which cannot be casted in variational form. Basic units of the numerical method are, therefore, FE-Kernels (KERNEL), which are concerned with the calculation of

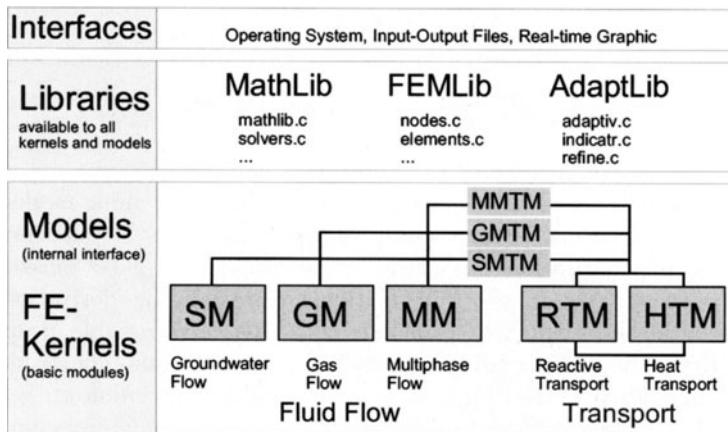


Figure 9.5: Model configuration by finite element kernels and common available methods

element matrices for flow, mass, and heat transport in several dimensions (Fig. 9.5). The global system of equations (EQS) is then assembled from element contributions. Elements of several geometric dimension can be coupled. Calculation of element matrices requires corresponding material properties (MATERIAL) and unknown vectors of iteration steps for non-linear problems. For simulation of coupled problems KERNELs can be combined to MODELS. MODELS have several tasks: to govern the communication between KERNELs, to store KERNELs result vectors, and to control time discretization.

9.4.2 Solution Process

The second step of the solution procedure requires the solution of the resulting algebraic equations (EQS). This process can also introduce an error but this is usually small compared with those involved in the above mentioned discretization step, unless the solver scheme is unstable. Appropriate methods to solve systems of algebraic equations are discussed by Habbar (1995). Beside direct methods a large variety of iterative solvers are available to all EQS.

9.4.3 Common Methods

The solution procedure requires many general mathematical methods, such as linear algebra (e.g. coordinate transformation and other matrix operations). Dynamic grid adaptation is available to several MODELS, in particular, heuristic methods (indicator) (Kaiser et al. 1999).

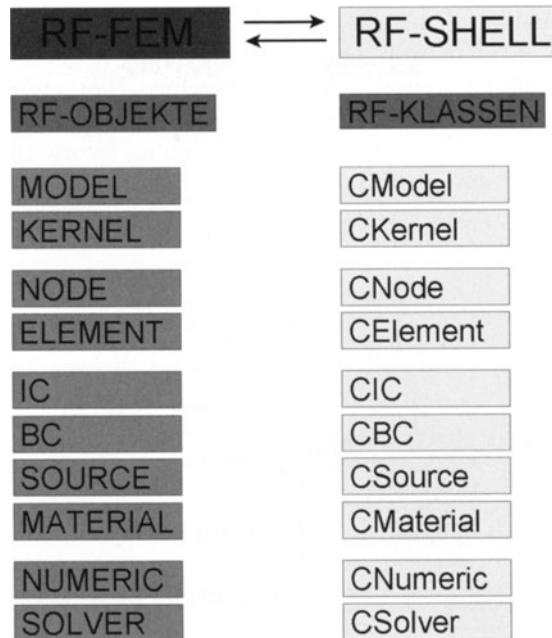


Figure 9.6: RF-objects and class representation

9.5 Informatic Objects

Fig. 9.6 shows the informatic objects of the finite element program and the corresponding representations as classes. Objects can have complexities at different levels. We distinguish between basic and combined objects. NODEs and ELEMENTs are basic objects for building geometry and topology. In addition to geometric information they contain also physical data. Values of unknown functions are connected to NODEs. MATERIAL data are associated with ELEMENTs. Topological relationships between neighbored elements are important for grid adaptation algorithms. Of course, ELEMENTs consist of NODEs. From a topological point of view, NODEs can be considered as zero-type ELEMENTs. Higher-order ELEMENTs are build in a hierarchical way: polylines are composed by nodes, plains by polylines, volumes by plains. This method of boundary representation (BREP) was already used for geometric modeling (see section 9.2). Efficient data structures are most important for dynamic grid adaptation. Moreover, elements of different dimension can be adapted during simulation. Next objects are associated with PDEs: IC - initial conditions, BC - boundary conditions, SOURCE - sink / source terms. It is reasonable to treat them as objects because they belong to any PDE (mathematical object). Moreover, these objects are basic units for the solution process (solving EQS).

MATERIALs are the coefficients in the PDE. They capture non-linearity of the

PDE. It is reasonable to consider the equation system EQS as an (combined) object because it is the discrete representation of the original PDE. Properties of KERNELs are finite element matrices, therefore, they are objects to FEM. KERNELs use MATERIALs for element matrix calculation. MODELs are more complex objects. Their objective is the approximate solution of sets of PDEs. They can consist of several KERNELs. Moreover, MODELs have to control storage and access to unknown vectors which are organized in tables.

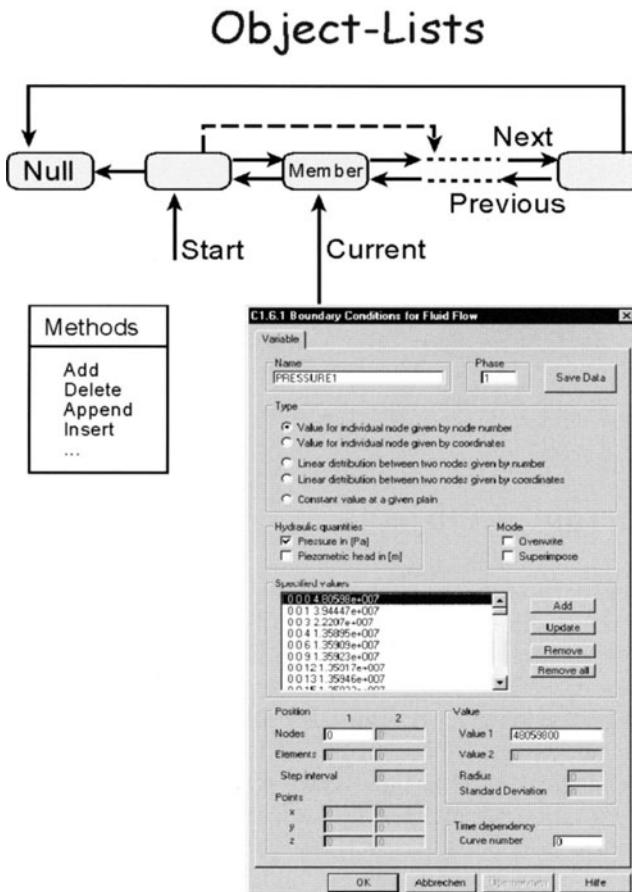


Figure 9.7: Object organization in lists

9.5.1 Data Structures

Data constructs (tydedef struct {}) are used for object implementation. As an example, Fig. 9.7 shows the graphical representation of the object BC (bound-

ary condition). To organize data objects we use pointer arrays and chained lists (Fig. 9.8) which have advantages as well as disadvantages. Main features of arrays are the following: arbitrary data types can be handled as elements, array size is incremented by a defined step, if deleting elements 'holes' appear, data access is fast. Lists are more flexible concerning dynamic construction / deconstruction of list elements. Data access is slower than for arrays. Sometimes it is convenient to apply both: lists and arrays. So MATERIAL lists for flexible access (RF-SHELL) and arrays for fast access during simulation (RF-FEM) to save computation time.

9.6 Software Objects

In general we talk about software components rather than software objects. Fig. 9.8 gives an overview of component development and cooperation with other institutes. Due to the complexity of environmental system analysis those co-operations are very important to integrate the expertise of several specialists.

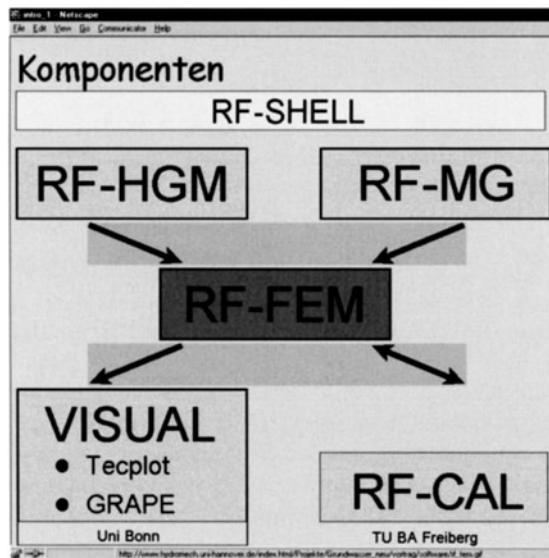


Figure 9.8: Software components for subsurface system analysis

RF-HGM is the geometry simulator for HydroGeologic Modeling (Rother et al. 1998) as described in section 9.2. This tool is suited for porous as well as for fractured aquifers. Now methods are extended to fractured-porous media. Mesh generation is developed in close cooperation with Prof. Taniguchi from the Okayama University (Taniguchi et al. 1996, Kasper et al. 1998). Process simulation is conducted with the finite element simulator RF-FEM (see also

section 9.4). Inverse modeling for automatic parameter estimation of density-dependent flow was developed by the team of Prof. Häfner from the Freiberg University (Häfner et al. 1998). Currently we extend inverse modeling to reactive processes (Habbar & Kolditz 1998). For visualization of geometric and simulation results we use the graphic system GRAPE (Prof. Kröner, Freiburg; Prof. Rumpf, Bonn) and Tecplot (AmTec). The software component concept emphasizes the necessity of inter-disciplinary co-operation for system analysis.

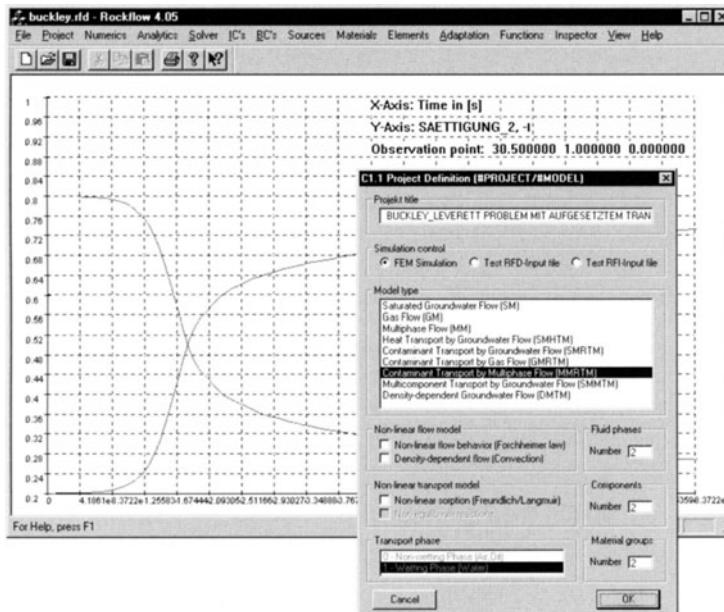


Figure 9.9: Graphical user interface of RF-SHELL

RF-SHELL is the graphical user interface of RF-FEM. The shell is based on the informatic objects described above. The primary aim is to support and test object-oriented methods. Additionally, the user interface is a helpful assistant for program application, including simple x-y plot graphics for data analysis (Fig. 9.9).

Object-oriented approaches are of essential importance for the development of large programs. Further code integration is easier. An alternative to avoid ever increasing code integration is process orientation (PO). The concept of PO is based on the server-client structure (Fig. 9.10). Communicating programs can exchange data via defined process interfaces. CORBA and COM are those tools for process communication which is also possible for distributed systems, i.e. processes running on different computers.

Software development in a team has numerous aspects (Kolditz et al. 2001), last not least a human factor. Important aspects of software engineering are version

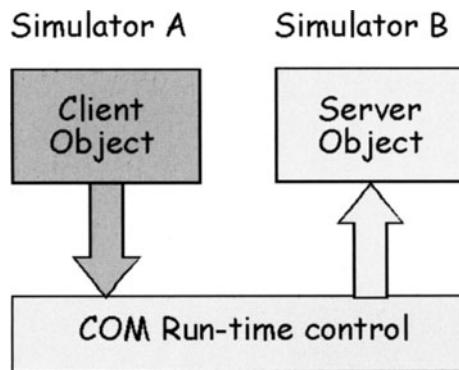


Figure 9.10: Server-client architecture for process orientation

management, permanent benchmarking. RF-FEM is developed and running on different platforms (Unix, Linux, OS/2 and Windows). In general we use GNU and VISUAL C++ software for code development.

Chapter 10

Object-Oriented Programming Techniques

10.1 Object Orientation

Object-oriented programming (OOP) has become exceedingly popular in the past few years. OOP is more than rewriting programs in modern languages, OOP is a new way of thinking about designing and realizing software projects. This requires a complete re-evaluation of existing programs (Budd 1996).

In this chapter we describe the development and realization of OO techniques for simulation of processes in environmental hydrosystems. There are two main parts about object design and object implementation. Design of objects starts from physical and mathematical principles, but there are no specifications for certain processes. Objects can be reused for very different applications. Physics is introduced only with the model-kernel concept. Kernels provide the discretized forms of process-specific partial differential equations. Models are the configuration and control unit for the arrangement of kernels to represent coupled multi-field problems. The implementation of objects is strongly guided by OOP principles such as encapsulation of user-defined data objects, protected data access via interfaces, virtual functions for polymorphism. Central units are algebraic equation systems, which are treated as combined objects. To show the advantages of OOP, such as code reuse, we present two applications based on the OO concept: finite element method for process simulation (FEM application) and a graphical user interface (GUI application). More than 10 physico-chemical models are implemented up to now, such as multiphase flow and multicomponental transport as well as deformation processes in porous media.

Object-oriented techniques

Object-oriented programming techniques are a necessary tool for realization of the concept (Stroustrup 1998). Important principles of OOP are:

- extraction of data units as well as the relationships between them (data model),
- object encapsulation (properties and methods) with private as well as public data and functions (classes),
- user-defined data constructs and abstract data types,
- virtual functions (prototypes) with defined interfaces but not yet implemented (polymorphism),
- class hierarchy and heredity (basic and derived classes),
- communication by message passing (method execution on request),
- software reuse.

10.2 Object Design

Design of objects means a clean separation of all essential information to represent the problem to be solved. Frequently, this requires a complete re-evaluation of existing codes.

10.2.1 Mathematical Objects

The continuum-mechanical, mathematical formulation (model development) relies on fundamental conservation principles of thermodynamics and corresponding closure relations (equations of state and constitutive relationships of material theory) (e.g. Bear & Bachmat 1990, Lewis & Schrefler 1998). From the viewpoint of object design, we may say that each process can be described by a partial differential equation (PDE) which has to be specified with initial and boundary conditions. The general continuum balance in Eulerian form equation can be written as follows

$$\int_{\Omega} \frac{\partial u_I}{\partial t} d\Omega = \int_{\Omega} \nabla \cdot \Phi^{u_I} d\Omega + \int_{\Omega} q^{u_I} d\Omega \quad (10.1)$$

where u_I is an unknown field function (i.e. conservation quantity - mass, momentum or energy), Φ^{u_I} is the flux term, q^{u_I} is the source term of the quantity, Ω is the model domain. Index I indicates that we have to deal with multi-field problems with more than one field function.

According to mixture theory for porous media, all quantities are averaged over a represented elementary volume and describe macroscopic behaviour. At the same time they are related to specific phases. The first term on the right hand side expresses macroscopic fluxes of the conservation quantities.

Equations of state express the equilibrium between pressure, density and temperature. These relations are typically based on thermodynamic models. Typical constitutive relations for multicomponent/multiphase systems concern capillary pressure / saturation / relative permeability, interphase mass transfer, chemical reaction, and stress-strain relationships. While derivations of constitutive relationships from first principles are theoretically possible in some cases (Diersch 1985) they are usually based on empirically determined relationships. Equations of state and constitutive relationships increase the number of equations and of unknown quantities. However, they supply sufficient information to close the system of equations by expressing unknown quantities in the governing equations in terms of known parameters.

From the viewpoint of object design we can separate following logic units which are required for each mathematical model: field functions u_I , material properties contained in flux terms Φ^{u_I} (equations of state and constitutive relationships), source-sink terms q^{u_I} , as well as initial conditions $u_I(t = 0)$ and boundary conditions $u_I(t) \in \Gamma$.

10.2.2 Numerical Objects

Numerical methods are used to obtain approximate solutions of the governing partial differential equations with the corresponding initial and boundary conditions. The procedure of deriving approximate solutions consists of two steps: the discretization and the solution process. The discretization step converts the continuous partial differential equations (PDE) into a discrete system of algebraic equations (EQS). There are several approximation methods to solve initial-boundary-problems numerically, such as finite difference method (FDM), finite volume method (FVM) or finite element method (FEM). In general, these approximation methods can be derived from the method of weighted residuals (MWR). Different choices of the weighting functions and corresponding algorithms to minimize the residual result in several methods (Tab. 10.1). Using those general principles is very important for object orientation. This allows the design of numerical objects which are independent of the specific approximation method applied.

PDE discretization

If an approximate solution \hat{u} is substituted into the original differential equation $L(u) = 0$, it will not yield identical results

$$R(t, x) = L(u) - L(\hat{u}) \neq 0 \quad (10.2)$$

Table 10.1: Approximation methods

Method	Weighting function	Remarks
Subdomain	$\omega_i(x) = \begin{cases} 1 & : x \in \Omega_i \\ 0 & : x \notin \Omega_i \end{cases}$	FVM
Collocation	$\omega_i(x) = \delta(x - x_i)$	FDM
Galerkin	$\omega_i(x) = \phi_i(x)$	FEM

where R is the residual, L is the differential operator, u is the exact solution, and \hat{u} is the approximate solution. The residual is a measure of the accuracy of approximation or of the error introduced by the discretization method. Since the error, in general, can not be made to vanish simultaneously in all discrete points, an approximate solution can be derived by requiring that some weighted average of the nodal residuals should vanish over the computational domain.

$$\langle \omega, R \rangle = \int_{\Omega} \omega_i R(t, x) d\Omega = \int_{\Omega} \omega_i L(\hat{u}) d\Omega = 0 \quad (10.3)$$

where ω_i are weighting or test functions. Now, the unknown values have to be defined in such a way, that the error (i.e. weighted residuals) is minimal.

Domain discretization

Approximate equations of the governing PDEs are always solved on finite grids. We obtain values of solution vectors \mathbf{x}_I in discrete points - nodes. Grid nodes and/or elements (consisting of nodes) are the basic topological units of approximation methods. Therefore, nodes (**NOD**) and elements (**ELE**) should be selected as topological objects in an object-oriented approach, in particular, for the finite element method. As a next step in object design, we have to classify node (Tab. 10.2) and element (Tab. 10.3) related units needed in the discrete approach.

Considering the mathematical problem (10.1), we can identify following units which depend on geometry, i.e. these can be related to grid nodes x_i : the unknown field function itself $u(t, x_i)$, initial conditions $u(t=0, x_i)$, boundary conditions $u(t, x_i), x_i \in \Gamma_1$, and source-sink terms $q(t, x_i) \in \Gamma_2$. Bold symbols in the following indicates nodal vectors $\mathbf{x} = \{u_1, u_2, \dots, u_{np}\}$.

In the same manner, we extract units, which can be related to elements: material properties which depend on the corresponding geologic unit (e.g. porosity) and on field variables as well (e.g. permeability varies with saturation in multiphase flow), resultants, which are not primary field variables but can be determined from these (e.g. fluxes).

Table 10.2: Node related objects

Quantity	Dependencies	Object
Unknown field vectors	$\mathbf{u}_I(t)$	MOD
Initial conditions	$\mathbf{x}_I(t = 0)$	IC
Boundary conditions	$\mathbf{x}_I(t) \in \Gamma_1$	BC
Source-sink terms	$\mathbf{q}_I(t) \in \Gamma_2$	SS

Table 10.3: Element related objects

Quantity	Dependencies	Object
Material properties	$\Omega^e, \mathbf{x}_I(t)$	MAT
Element resultants	Materials, $\mathbf{x}_I(t)$	MOD

From the physical point of view it is reasonable to introduce following subunits of material properties: fluid properties (FP), solid properties (SP), component properties (CP), and porous medium properties (MP). The first two are phase related properties of the composite porous medium. Actually, fluids must not be related to geometrically fixed elements as they can change their properties in different soil or rock types. In contrast, solid and porous medium properties obviously are geometrically related to elements (soil/rock types). Chemical components can belong to both fluid and solid phases, e.g. they can be dissolved chemicals in a fluid phase or sorpted onto a solid phase. Material properties are basic physical units. Therefore, it is necessary to treat them as an independent object (MAT).

10.2.3 Algebraic Equations - Solver

In general, discrete approximation methods result in systems of algebraic equations to be solved. These equations might be linear or still non-linear. Frequently, these equations are coupled for multifield problems. In this case, several equation systems have to be solved in an appropriate way. A general form of those algebraic problems can be written as

$$\mathbf{A}_I(t^{n+1}, x_i, \mathbf{x}_I, \mathbf{x}_J) \mathbf{u}_I = \mathbf{b}_i(t^{n+1}, x_i, \mathbf{u}_I, \mathbf{u}_J) \quad (10.4)$$

with: system matrix \mathbf{A} , right-hand-side (RHS) vector \mathbf{b} , new time level $n + 1$.

The algebraic equation system can be regarded as a basic unit of approximation methods. Moreover, establishment of the equation system is the final product for determination of the unknown field function. Therefore, we introduce equation

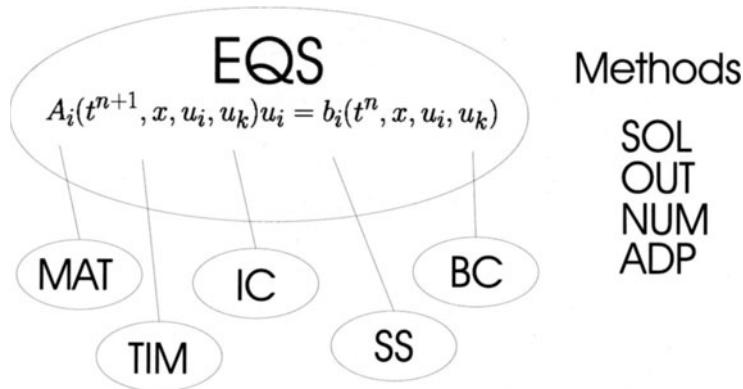


Figure 10.1: Components of equation system object (EQS)

system objects (**EQS**). Fig. (10.3.5) shows the contributions from other objects, we introduced before.

The system matrix **A** contains material data **MAT**, which may depend on field functions as for non-linear problems. Additionally, parts from the interpolation procedure (shape functions and its derivatives) as well as from time discretization belong to the system matrix. Time stepping schemes Δt^{n+1} are controlled by the time object (**TIM**). Field functions \mathbf{x}_I have to be initialized with initial conditions (**IC**). Boundary conditions (**BC**) as well as source-sink terms (**SS**) have to be incorporated at every time step. The above mentioned contributing objects can be regarded as properties of the **EQS** object. Methods which can be applied to the equation system are: explicit/implicit schemes for time depending problems resulting in modification of the system matrix and the RHS vector (**NUM**), grid adaptation for improving approximation quality (**ADP**) resulting in size varying matrices, and (linear) equation solver (**LS**) yielding the field function solution vector.

For solving non-linear (i.e. \mathbf{A}_I depends on field function \mathbf{x}_I) as well as coupled multifield problems (i.e. \mathbf{A}_I depends on field function \mathbf{x}_J) the solution process must be embedded in iteration loops. It is convenient to introduce non-linear solver objects (**NLS**) and iteration objects (**ITE**) to this purpose.

We see that equation systems can be designed in a very common way. In fact, physics are introduced by the material properties (**MAT**) only. It is important to note, that the above described **EQS** objects can be constructed independently of a specific physical problem.

10.2.4 Model-Kernel Concept

The general purpose of this OO concept is to model coupled processes in subsurface systems (Fig. 9.4), such as saturated groundwater flow, air flow, multiphase fluid flow, mass transport of multiple chemical components, chemical / biochemical processes, heat transport in porous / fractured media, as well as deformation processes.

The motivation and main advantage of OO methods is that further program development, in particular extensions of physico-chemical models, is easily possible. To this purpose the model-kernel concept based on OO techniques was developed which is discussed in following in more detail.

Kernels (KER) contain the specific physico-chemical information. They are responsible for the calculation of finite element matrices for the specific process. The following table lists the existing kernels.

Table 10.4: RF Kernels

Objective	Kernel
Saturated groundwater flow	RF-SM
Air flow	RF-GM
Multiphase fluid flow	RF-MMP RF-MMS
Reactive transport	RF-RTM
Multicomponent transport	RF-MTM
Heat transport	RF-HTM
Deformation processes	RF-DM

For multifield problems several kernels can be combined to so-called models. **Models (MOD)** as an object have properties and methods.

Properties: Model data consist of node and element related information (Tables 10.2 and 10.3). Models contain element matrices from the corresponding kernels and tables of specific field functions (node related) as well as of required element resultants (element related).

Methods: Model functions are basically concerned with configurations, e.g. which kernels are required for the specific problem. Models have to manage the data exchange between objects, such as kernels, materials, node data, element data, solver (see also section 10.3.6). A main task of those models is the configuration of data access and the assignment of function prototypes. Function prototyping allows polymorphical algorithms. Polymorphism is an important feature of object-oriented programming. Successive and/or iterative solving of the corresponding equation systems is conducted in the model time loop (see also section 10.3.6).

10.3 Object Implementation

In this part we discuss the implementation of objects which were introduced in the previous section. This will illustrate the features of object orientation, i.e. data encapsulation, protected data access, object-related methods.

10.3.1 Object

Important features of objects are their properties (data), methods (operations on data), and interfaces (data exchange).

Properties

We introduce user-defined data constructs for the definition of data objects. We can involve arbitrary data types for object properties and functions for object methods.

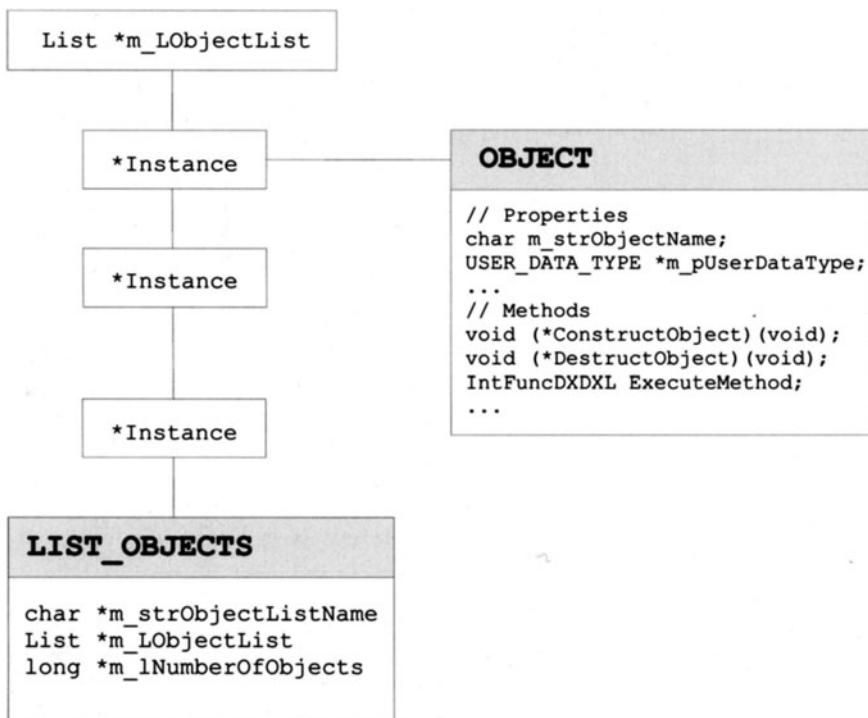


Figure 10.2: Object lists

Instances of data objects are organized in arrays or lists (Fig. 10.2). Lists are

very flexible for data manipulation. Arrays have the advantage of a faster data access. Object identification is realized by its name (strings).

Methods

Typical object methods are data construction / destruction, list operations (insert, delete, append, move ...). Examples of specific object-related methods are execute-object-functions, e.g.

```
ExecuteInitialConditions()  
ExecuteBoundaryConditions()
```

These methods assign initial and boundary conditions, e.g. for a defined area, to the corresponding nodal values of field functions. It is important to point out that these execute-object-functions work for all field functions. That means, calling an execute-object-function results in updating all IC and BC objects for all specified field functions. Frequently, some objects use data of other ones. Therefore, these objects must supply their data to others via interfaces.

Interface

The capability of objects to offer their data to others (communication) is realized by interfaces. On the other hand objects must be able to hide their data - as a protection against forbidden access. Private data are available only to methods of this object. Typical internal access functions to private object data look as follows, with object address as function argument.

```
get_object_property(m_pObject)  
set_object_property(m_pObject,property)
```

External data access (i.e. by other objects) is realized by following functions.

```
OBJECT *GetThisObject(m_strName)  
SetThisObject(m_pObject,m_strName)
```

Data access is protected by object name, which is more secure than direct access via object address.

10.3.2 Nodes and Elements

As described above (10.2.2), nodes and elements are fundamental topological units of discrete approximation procedures. Values of corresponding field functions are assigned to the grid nodes. Field functions are specified by the MODEL.

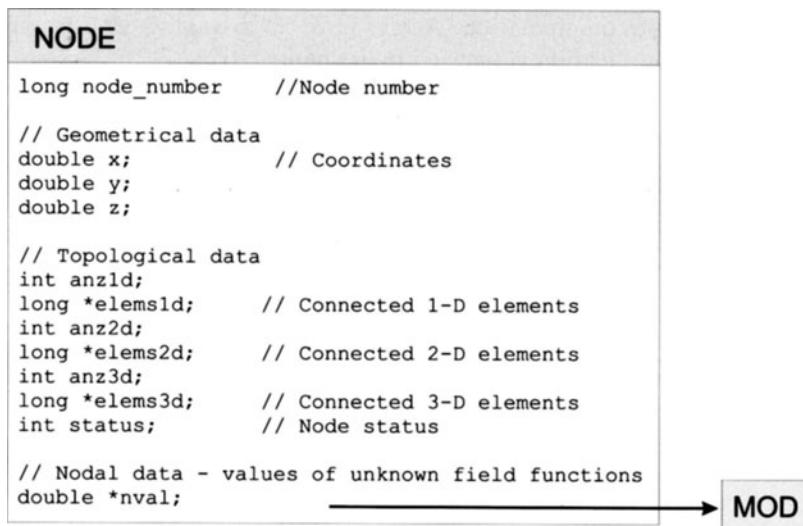


Figure 10.3: NODE data object

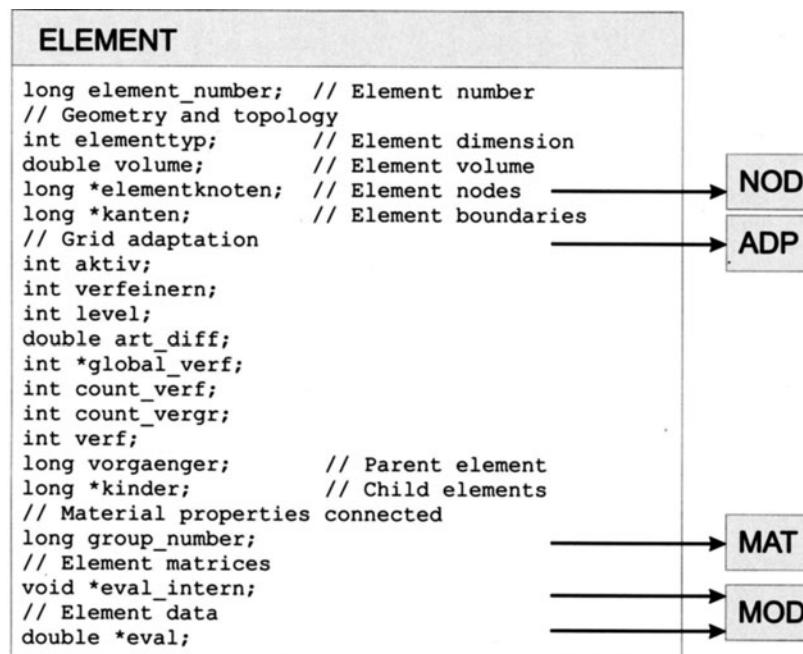


Figure 10.4: ELEMENT data object

Therefore, we embed nodal value vectors into the **NODE** object (Fig. 10.3). Node-related objects are discussed in section 10.3.3.

Elements are quite complex data constructs with several relationships to other objects. They capture topological information, parameter of grid adaptation control as well as element matrices and element resultants. The data construct reflects specifics of the finite element method (Fig. 10.4). Element-related objects are discussed in section 10.3.4.

10.3.3 Node Related Objects

Most data required for establishing the equation systems are connected with grid nodes as geometric unit of discrete approximation methods. Based on this we introduce node related objects with common and specific properties given in Tab. 10.5.

Table 10.5: Properties of node related objects

Common	Geometry Topology	\mathbf{x} i
Specific	Assignment to field function Object-ID	\mathbf{u} name

It is important to examine the relationships of node related objects to the equation system.

$$\underbrace{\mathbf{A} \underbrace{\mathbf{u}}_{\text{IC,BC}}}_{\text{EQS}} = \underbrace{\mathbf{b}}_{\text{SS}} \quad (10.5)$$

We distinguish between two kinds of node related objects.

I	Field functions	\mathbf{x}
II	Initial conditions Boundary conditions Source-sink terms	IC BC SS

The first group consists of data connected to all nodes. During the solution process, values of the unknown field functions (Tab. 10.6) have to be determined at all grid nodes. As a consequence, nodal values of field functions are directly connected with the **NODE** object and organized in the node list (see section 10.3.2). In fact, field functions are treated as properties of the **NODE** object.

Table 10.6: Field functions

Fluid phase pressure	p^γ
Fluid phase saturation	S^γ
Componental volumetric concentration	C_k^γ
Phase temperature of multiphase medium	T^γ
Solid phase displacement	\mathbf{u}^s
Phase identifier	γ
Component identifier	k

The second group consists of objects, which are not necessarily required for all nodes. Their individual status is reasonable, because their treatment is independent on the specific field function, e.g. boundary conditions of any field function are incorporated into the equation system (10.5) in the same way. Therefore, methods for boundary conditions must be available to all equation systems. The second group of node related objects are also organized in lists (see Fig. 10.2).

Example - BC

As an example we consider the boundary conditions BC object. Boundary conditions can be prescribed in nodes on the domain boundary: $u_i(t) \in \Gamma_i^1 \subset \Omega$. The BC object has to be usable for any field function.

Object properties

BC properties have to cover requirements of all field functions. Fig. 10.5 graphically illustrates the BC object with its properties (P) and methods (M). Common properties are e.g. positions, values, validity (i.e. life time). All boundary conditions are specified by an ID (Tab. 10.7).

Object methods

Typical BC methods are object construction / destruction, assignment to nodes, incorporation into the equation system (10.5).

Object organization

BC objects can be organized in lists (Fig. 10.2) and containers (Fig. 10.6). Each row is an array with fixed length. Row lengths may vary in the container. Lists are very flexible for data handling, but data access is slow. In contrast, containers allow faster data access.

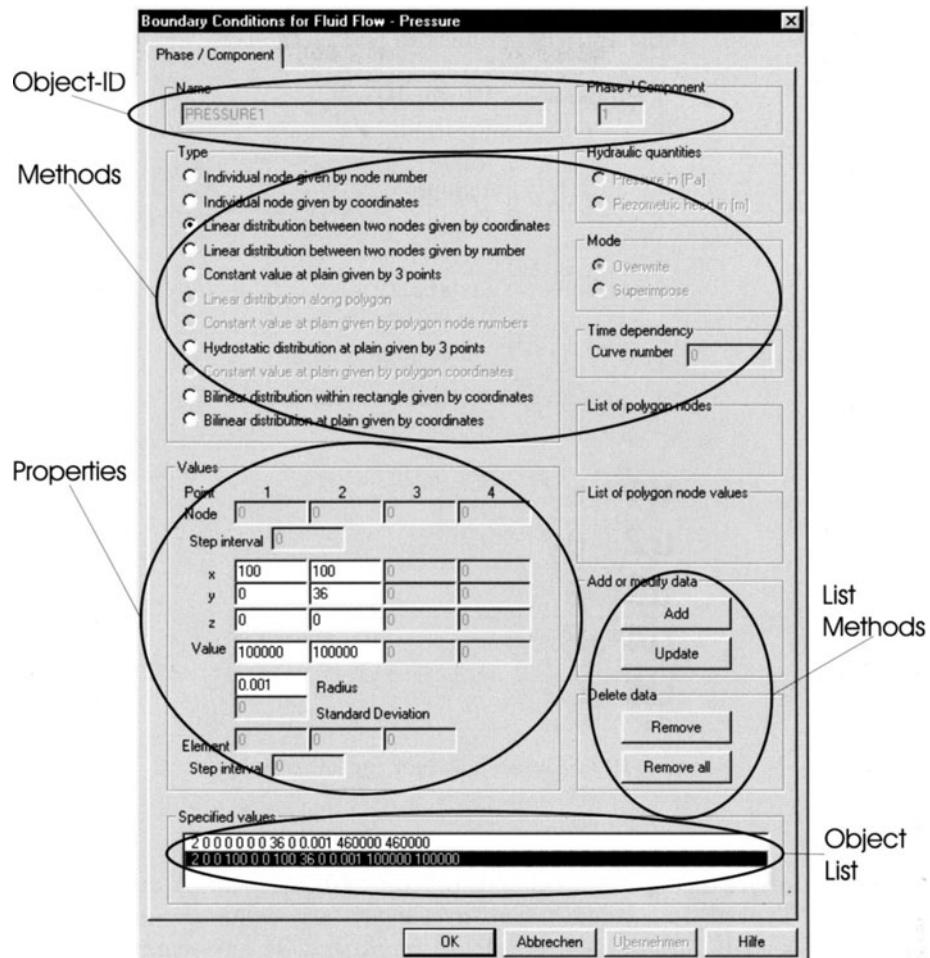


Figure 10.5: Graphical representation of BC object

10.3.4 Element Related Objects

Finite elements are fundamental units of the specific approximation method we selected. Therefore, the finite element itself is treated as an object (see section 10.3.2). Now we examine objects related to elements with respect to the equation system.

$$\underbrace{\mathbf{A}}_{\text{MAT}} \underbrace{\mathbf{u}}_{\text{EQS}} = \underbrace{\mathbf{b}}_{\text{MAT}} \quad (10.6)$$

Table 10.7: BC properties and methods

Properties	Object-ID Position Value Validity Mode ...
Methods	CreateBC() ExecuteBC() IncorporateBC() ...

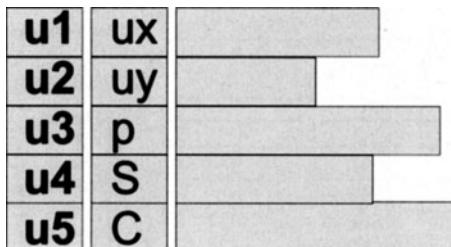


Figure 10.6: Data container for BC objects

Element related objects are associated with the system matrix **A** and the RHS vector **b**. Typical data connected with elements are geometric and material properties contributing to the finite element matrices. The system matrix then is assembled from the element matrices.

The basic step in object design is the analysis of object dependencies. At first, element related objects may vary locally: **A**(*x*), e.g. as soil or rock types change. Secondly, element related objects may depend on field functions: **A**(*u*) , as for non-linear problems. We see that element related objects provide complex dependencies.

Example - MAT

In section 10.2.1 we introduced a process describing PDE as an abstract mathematical object. The specific physical problem behind the PDE depends mainly on the corresponding material properties which are coefficients in the PDE. Due to the physical importance of material properties we treat them as objects (MAT), we consider in the following.

Object properties

Material properties cover possible non-linearities, i.e. they can depend on field functions (Tab. 10.8).

Table 10.8: Non-linear material behavior

Porosity	$n^e(C^s, \mathbf{u}^s)$
Permeability	$\mathbf{k}^e(S^\gamma, p^\gamma, C^s, \mathbf{u}^s)$

For porous media consisting of multiple phases we distinguish between fluid (FP) and solid properties (SP) as well as properties of the porous medium itself (MP). Additionally we have to consider the properties of chemical species within the phases (CP) (see Fig. 10.11).

Methods for MAT objects are calculation of material properties depending on field functions:

```
CalcPermeabilityPressureDependence(pha)
CalcPermeabilitySaturationDependence(pha)
CalcPermeabilityDissolutionDependence(pha, com)
CalcPermeabilityDilatanceDependence(pha)
```

Object organization

To organize MAT objects we use again lists and containers. To ensure fast data access we use container. In this case we order the data objects in a table with variable column lengths (Fig. 10.7). In fact, the MAT container is a 3-D data construct showing the basic physically dependencies on phases (fluids and solids), components (chemical species with phases) and material groups (i.e. soil/rock types).

Object interface

Materials are required for calculation of element matrices and element resultants. The data access to the permeability tensor is provided via material group (mat), phase (pha) and component number (com). The GetFunction contains all CalcFunctions required for non-linear permeabilities. If permeabilities do not depend on specific field functions the corresponding CalcFunction returns 1.

```
permeability =
GetPermeabilityFunction(mat, pha, com)

GetPermeabilityFunction(mat, pha, com)
```

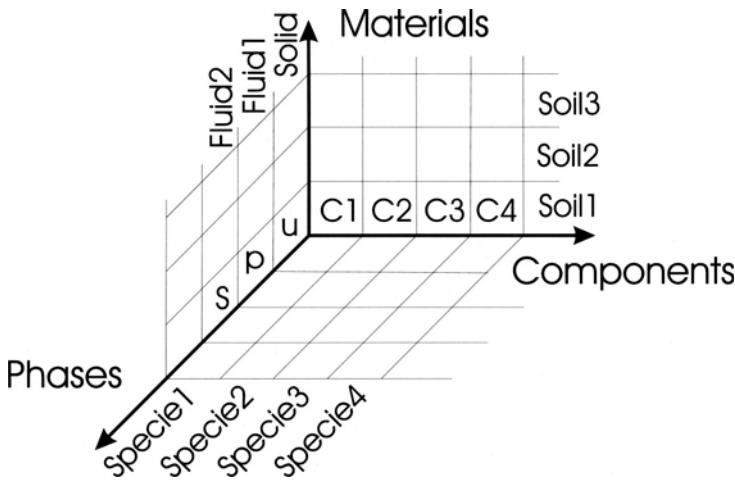


Figure 10.7: Material properties container

```
{
permeability =
GetPermeabilityTensor(mat) \
* CalcPermeabilityPressureDependence(pha)
* CalcPermeabilitySaturationDependence(pha)
* CalcPermeabilityDissolutionDependence(pha,com)
* CalcPermeabilityDilatanceDependence(pha)
return permeability;
}
```

The example given above shows the protected data access of the KERNEL to a MAT object (interface to permeabilities). Only the physically important information is needed, i.e. which element, which phase and which chemical component are active. This `GetFunction` for permeability access, of course, is available to all models.

10.3.5 Equation Systems and Solver

As described in the section about design of EQS objects, algebraic equation systems (10.4) are the central unit of approximation procedures. Fig. shows the dependencies on other objects. In this section we focus on the solving process of equation systems. From the concept of object orientation, it is important to develop methods which are independent of the specific, underlying physical problem (i.e. PDE). Equation systems can be linear or non-linear. Therefore, we divide EQS objects in two classes: linear equations systems (LES) and non-linear equations systems (NLES).

Linear Equation System - **LES**

Data properties of linear solver objects are: system matrix **A**, solution vector **x**, RHS-vector **b** and residuum **r** = **A** **x** – **b**

Table 10.9: **LES** properties and methods

LES - Property	LES - Method
System matrix A	<code>AssembleSystemMatrix()</code> <code>IncorporateBoundaryConditions()</code>
Solution vector x	<code>LinearSolver()</code> <code>IncorporateBoundaryConditions()</code>
RHS vector b	<code>IncorporateSourceTerms()</code>
Residuum vector r	<code>LinearSolver()</code>

Methods of linear solver objects are summarized in the Tab. 10.9 above. Additionally, we see the implemented functions. Assembling the system matrix is based on the individual element matrices, which are provided by the corresponding kernel. Incorporation of boundary conditions results in a modification of the system matrix and the RHS-vector as well. Source terms refer to the RHS-vector alone. Finally, an equation solver is employed to obtain the solution vector.

For flexible use of **LES** objects configuration methods are available concerning specification of element matrices for system matrix assemblage, boundary conditions as well as source terms belonging to the corresponding field function. The relationship between model and linear equation system is given by specifying the kernel used for element matrices calculation (Fig. 10.8).

Non-Linear Equation System - **NLES**

For non-linear problems the system matrix depends on the solution vector itself **A(x)**. As a consequence, the linear solution step has to be embedded in an iteration loop (Fig. 10.9). The general idea for design of (**NLES**) objects is, therefore, to embed linear equation objects (**LES**) (Fig. 10.8).

Methods of the **NLES** object are (i) updating the system matrix **A(x)** depending on the solution vector itself, (ii) repeatedly execution of the linear (equation) solver, and (iii) iteration error calculation. The non-linear solver has to stop if a certain tolerance criterion is fulfilled.

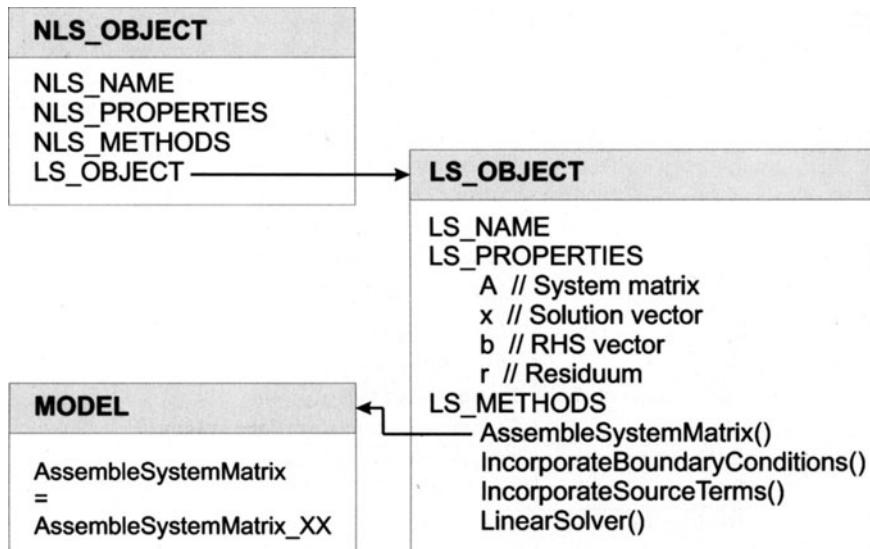


Figure 10.8: Non-linear and embedded linear solver objects

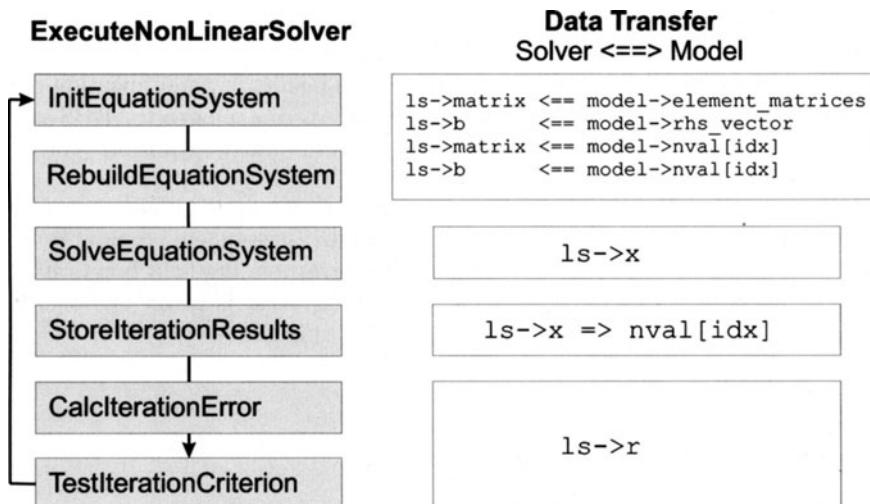


Figure 10.9: Algorithm for solving NLES

10.3.6 Models

Models are the configuration and control units of the program. They arrange several kernels required for solving the underlying physico-chemical problem.

(Fig. 10.10). The arrangement of several loops for the overall solution procedure is implemented in the model loop function. For rearrangement only this function has to be modified.

Model properties are: table of field functions (nodal values), table of element resultants (element values), and element matrices provided by kernels. These data are stored in the **NODE** and **ELEMENT** objects, respectively.

Most important **MODEL** methods are as follows:

- Creation and configuration of required equation system objects (EQS)
- Configuration of node and element data
- Configuration of output data
- Implementation of virtual functions

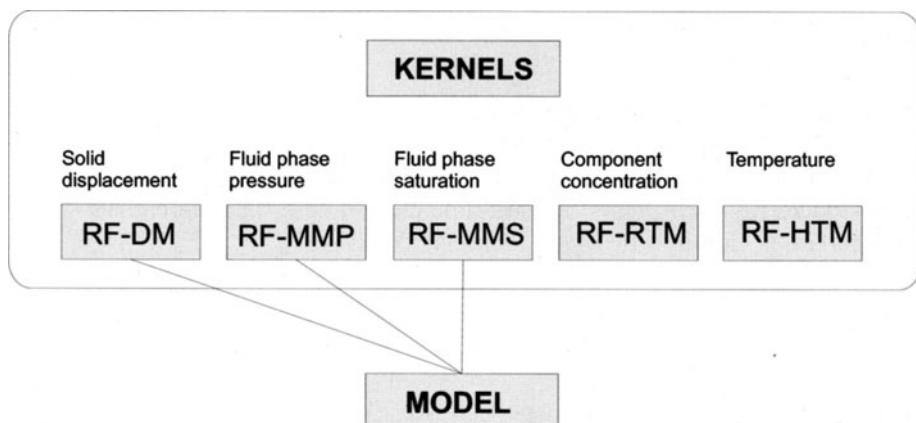


Figure 10.10: Model-Kernels

Kernels

As described in section 10.2.4 kernels represent the specific physico-chemical problem. They produce the corresponding finite element matrices for the set of PDEs. Kernels need two types of information: material data and geometric/topologic data (Fig. 10.11).

Kernels are based on templates providing the common frame to all of them, i.e. geometric operations (coordinate transformation, spatial derivatives) for several geometric element types (e.g. line, triangle, quadrilateral, tetrahedron, hexahedron, prism).

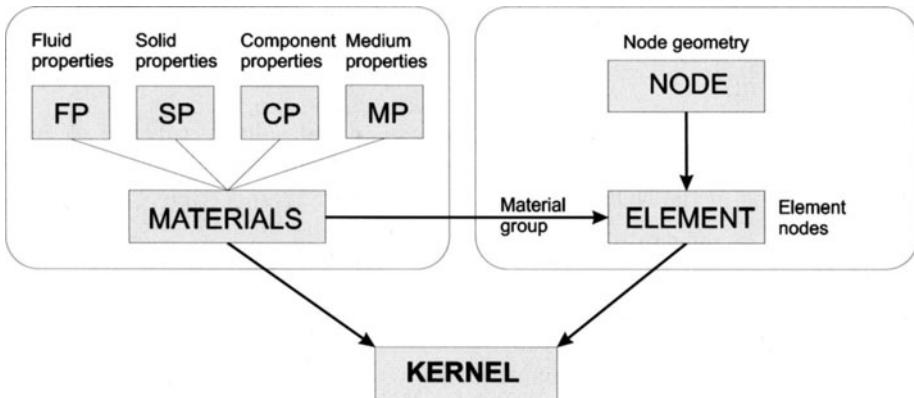


Figure 10.11: Kernel data

Loops

As we deal with initial-boundary-value-problems the unknown field functions $u(t, \mathbf{x})$ vary in time and space. During the discretization process we separate the independent variables t and \mathbf{x} . For example, in the finite element method, nodal values (at fixed points) remain time depending $u_i(t)$, whereas the spatial dependence is captured by shape functions $N_i(\mathbf{x})$.

$$u(t, \mathbf{x}) \approx \hat{u}(t, \mathbf{x}) = u_i(t)N_i(\mathbf{x})$$

This is important for the implementation. Fig. 10.12 shows several nested loops. The time discretization process builds the outer loop, whereas spatial discretization (grid adaptation, elements, Gaussian points) is in the inner loops. The position of the iteration loop for non-linear problems depends on the regularization method. As an example, permeability k could be a non-linear function, which can depend on saturation S , fluid pressure gradient ∇p , concentration of dissolved minerals C or grain displacements \mathbf{u} . If permeability is evaluated at element level, the iteration loop should cover the element loop.

Fig. 10.12 also displays the positions of kernels x, y and z for successive determination of multiple field functions within nested loops. The model specifies the used kernels x, y and z as well as manages data transfer between kernel and common objects. The time loop determines the order of kernels x, y and z as well as covers the iteration loop for resolution of non-linearities. The kernel supplies the required system matrices for the equation systems to be solved. In this OO implementation, problem-specific model-kernel source files can be kept at a minimum (Table 10.10).

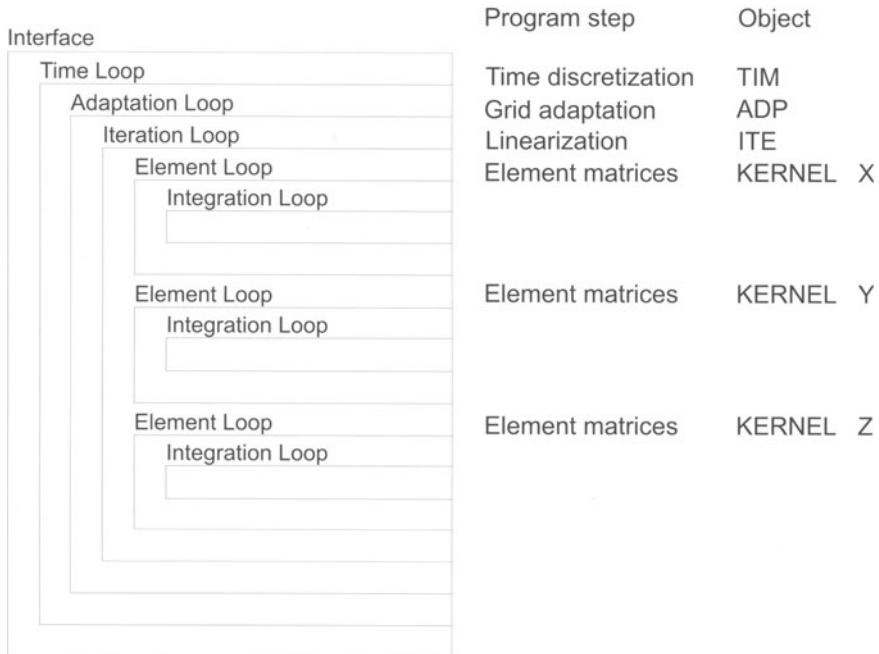


Figure 10.12: Nested loops

Table 10.10: Model-kernel sources

<code>mod_xyz.h/cpp</code>	Model configuration
<code>loop_xyz.h/cpp</code>	Time loop organization
<code>cgs_x.h/cpp</code>	System matrix assemblation
<code>cel_x.h/cpp</code>	Element matrix calculation

10.3.7 FEM – Application

Fig. (10.13) gives an overview on the general implementation concept. The finite element part (RF-FEM) is embedded as an application (APL) in the OO framework. Implementation of other applications based on different approximation methods (e.g. finite differences, finite volumes) is possible in the same way. Basic data structures (e.g. object lists or objects itself) and methods (e.g. equation solver) are not restricted to a certain numerical method. Specifics of the finite element method are limited to the kernel-model level.

All data structures are organized in object lists, which are constructed before execution of the application and destructed afterwards. There are two user interfaces of RF-FEM to the operation system: data files and a graphical user

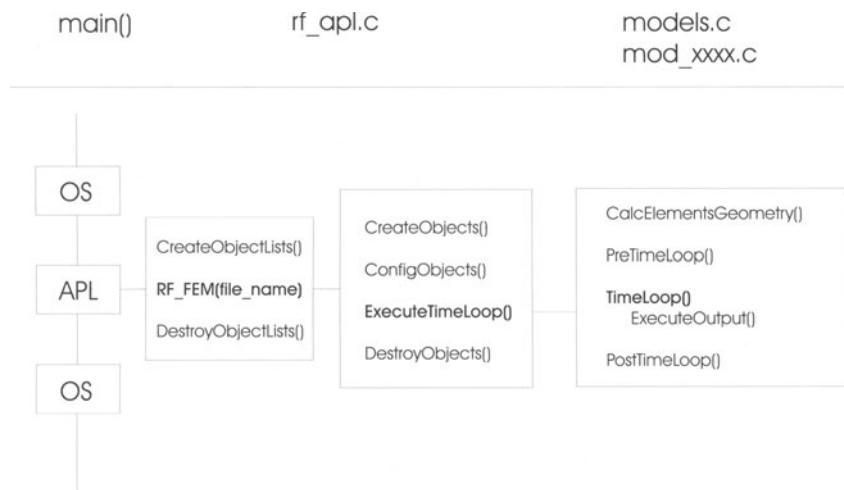


Figure 10.13: Application embedding

interface (RF-SHELL), which is introduced in the next section 10.4.

10.4 Graphical User Interface

RF-SHELL is the graphical user interface (GUI) of the finite element simulator RF-FEM. The shell is based on the above described data objects. It is primary aimed to support and test object-oriented methods. The main task of the shell is interactive handling of all objects, i.e. construction/destruction and filling objects with data. Additionally, the user interface is a helpful assistant for program applications, including simple graphics for data analysis.

10.4.1 Shell – Objects

The GUI design is strongly guided by the object-oriented approach. Fig. 10.14 shows the mainframe menu and the connected source files of the FEM application (below) as well of the GUI application (above).

Tab. 10.11 shows the classes based on the RF objects.

Data construction

Data construction is possible in two ways: first by reading data files and second by data creation using GUI dialogs (Tab. 10.12). Default data sets are created by selecting a specific physical model. Nonetheless, arbitrary data objects have to be available at any time. Therefore, constructor and destructor functions exists for each object. Again lists are used for dynamic data handling.

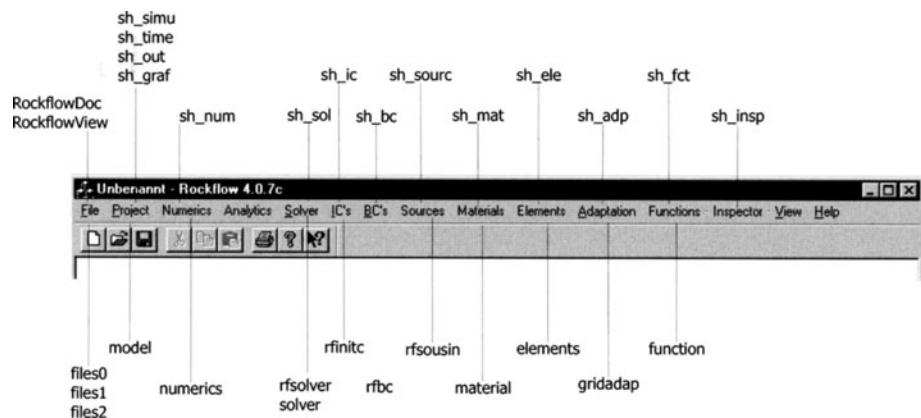


Figure 10.14: Mainframe menu of RF-GUI

Table 10.11: Shell classes

Shell classes	Object properties
CRockflowDoc	RF objects
CRockflowView	Object-dialog interface
CProject	Simulation control
CNumerics	NUM
CSolver	EQS
CInitialConditions	IC
CBoundaryConditions	BC
CSources	SS
CMaterials	MAT
CFuctions	Auxiliary functions
CGraphics	Graphical output

Table 10.12: General constructor functions

Functions	Purpose
OnFileNew()	Creating data of a new model
OnFileOpen()	Creating data from existing file

10.4.2 Application Embedding

Fig. 10.13 illustrates the general scheme of the FEM application. For interactive simulations these procedures must be splitted up. Fig. 10.15 shows three methods of the CRockflowView class necessary for embedding the FEM application

within the shell. Consequent encapsulation of data construction and destruction supports implementation of applications into graphical user interface.

OnFileOpen()	RunSimulator()	OnOK()
CreateObjectLists() CreateObjects() ConfigObjects()	ExecuteTimeLoop()	DestroyObjects() DestroyObjectLists()

Figure 10.15: Application implementation

10.4.3 Object - GUI - Interface

GUI dialogs represent the object properties. Data construction as well as data access are realized by object methods given in Tab. 10.13. Data construction/destruction has to be possible interactively with help of dialog menus.

Table 10.13: Methods for data construction and access

Data exchange	::GetObjectData() ::SetObjectData() ::AddObjectData() ::RemoveObjectData()
Construction Destruction	

Fig. 10.16 illustrates the data exchange between object lists (RF-FEM) and dialogs (RF-SHELL).



Figure 10.16: Data exchange between data objects and GUI dialogs

Chapter 11

Element Implementation

In this chapter we discuss the implementation of finite elements within the object-oriented framework presented in chapter 10. We consider all required steps for introduction of element types into the code. As an example we present the implementation of triangular elements. Theory of triangular elements is described in section 7.4.3. Based on this element template, new ones can be introduced very easily (see Problems).

11.1 2-D Linear Triangular Elements

11.1.1 Data Input

```
0  0.000000  0.000000  0.000000
1  3.000000  0.000000  0.000000
2  1.000000  2.000000  0.000000
0 0  tri  0  1  2
```

Read function

files2.c

```
int GeomReadRFInputFileASCII ( FILE *f, FILE *prot )

/* Zeile lesen */
if (fgets(s,MAX_ZEILE,f)==NULL) { ...
    if (sscanf(s," %ld %ld %ld %5s\n",&nr,&mat,&vorgaenger,etyps,&pos)!=4)
        { ... }

/* Elementtyp auswerten, Nummern weiterhin zulaessig */
```

```

if ((strcmp(etyps,"tri")==0) || (strcmp(etyps,"4")==0))
    etyp = 4;

/* Element erzeugen */
CreateElementTopology(etyp,vorgaenger,0,nr);
ElSetElementStartNumber(nr,start_number);
ElSetElementGroupNumber(nr,group_number);
switch (etyp) {
    case 4: anz_2D++; break;
}

/* Elementknoten lesen */
etyp--;
knoten = (long *) Malloc(sizeof(long)*ElNumberOfNodes[etyp]);
for (j=0;j<ElNumberOfNodes[etyp];j++) {
    if (sscanf(&s[p+=pos]," %ld\n",&(knoten[j]),&pos)!=1) {
        FilePrintString(prot,"* !!! Fehler:
            El-Knotennummer nicht gelesen; Einlesen beendet !");
        LineFeed(prot);
        knoten = (long *)Free(knoten);
        s = (char *)Free(s);
        return 0;
    }
    if (GetNode(knoten[j])==NULL) {
        FilePrintString(prot,"* !!! Fehler:
            El-Knotennummer existiert nicht; Einlesen beendet !");
        LineFeed(prot);
        knoten = (long *)Free(knoten);
        s = (char *)Free(s);
        return 0;
    }
}

/* Elementknoten eintragen */
ElSetElementNodes(nr,knoten);

```

11.1.2 Element Geometry

elements.c/h

```

int ElNumberOfNodes[4] = { 2, 4, 8, 3 };
/* Anzahl der Knoten eines Elements, siehe Header */
int ElNumberOfEdges[4] = { 1, 4, 0, 3 };
/* Anzahl der Kanten eines Elements, siehe Header */
int ElNumberOfPlains[4] = { 0, 1, 6, 1 };
/* Anzahl der Flaechen eines Elements (1D bei 3D-Netz: n) */
int ElNumberOfChilds[4] = { 2, 4, 8, 4 };

```

femlib.c/h

```
CalcElementsGeometry();
InitElementGeometry(i);
CalcElementJacobiMatrix(index, invjac, &dj);
SetElementJacobiMatrix(index, invjac, dj);
CalcElementVolume(index, &volume);
Calc2DTriangleElementVolume ( long index, double *volume )
ElSetElementVolume(index, volume);
```

11.1.3 Element Matrices

cel_asm.c/h

```
void CalcElementMatrix_ASM(long index)
void CalcEle2DTriangle_ASM(long index)
```

11.1.4 Equation System

cgs_asm.c/h

```
void MakeGS_ASM ( double *rechts, double *ergebnis,
case 4: MakeGS2DTriangles_ASM(index,rechts,ergebnis,aktuelle_zeit);
```

11.1.5 Element Resultants

cvel.c/h mod_00.c/h

```
int RefreshOutputData_ASM00()
void CalcVelo2DTriangleRS(int,long,double,double,double,double *);
```

11.2 Problems

- 1 Implement a linear tetrahedral element described in section 7.4.5. Nodes and element are given as follows.

```
0 0.000000 0.000000 0.000000  
1 1.000000 0.000000 0.000000  
2 0.000000 2.000000 0.000000  
3 0.000000 0.000000 2.000000  
0 0 tet 0 1 2 3
```

- 2 Implement a linear triangle-based prismatic element described in section 7.4.6. Nodes and element are given as follows.

```
0 0.000000 0.000000 0.000000  
1 1.000000 0.000000 0.000000  
2 0.000000 2.000000 0.000000  
3 0.000000 0.000000 2.000000  
4 1.000000 0.000000 2.000000  
5 0.000000 2.000000 2.000000  
0 0 prism 0 1 2 3 4 5
```

Bibliography

- [1] Bear J & Bachmat Y (1990): *Introduction to modeling of transport phenomena in porous media*, Kluwer Academic Publisher, Dordrecht.
- [2] Budd T A (1996): *An introduction to object-oriented programming*, Addison-Wesley.
- [3] Diersch H-J G (1985): *Modellierung und numerische Simulation geohydrodynamischer Transportprozesse*, Habilitationsschrift, Akademie der Wissenschaften der DDR, Berlin, 267S.
- [4] Habbar A (1995): *Vergleich verschiedener Krylov-Verfahren für allgemeine reguläre und sehr große lineare Gleichungssysteme*, Institut für Strömungsmechanik und Elektr. Rechnen im Bauwesen, Universität Hannover.
- [5] Habbar A & Kolditz O (1998): *Numerical modelling of reactive transport processes in fractured porous media*, In Proc. of the International Conference on Groundwater Quality GQ'98, Tuebingen, Germany, September 21-25, 1998.
- [6] Häfner F, Boy S, Behr A, Wagner S; Zielke W, Ratke R, Kolditz O (1998): *Entwicklung von Verfahren und Programmen zur Kalibrierung von Strömungs- und Transportmodellen*, Abschlußbericht zum BMBF-Vorhaben 02 C 0244 8.
- [7] Kaiser R, Rother T & Kolditz O (1999): *Modelling of flow in fractured aquifers using automatic grid adaptation*, In Proc. of the XXVIII IAHR Conference on Hydraulic Engineering for Sustainable Water Resources Management, 22-27.08.1999, Graz.
- [8] Kasper H, Kosakowski G, Rother T, Thorenz C, Kolditz O & Taniguchi T (1998): *Development of a 3-D CAD system for numerical analysis of subsurface flow and transport*, In Proc. 6th Int Conf on Num. Grid Generation in Comp. Field Sim., pp 683-694. Greenwich, UK, July 06-09, 1998.
- [9] Kolditz O, Habbar A, Kaiser R, Kasper H, Rother T, Schulze-Ruhfus M, Thorenz C & Zielke W (1998a): *Software concept of simulating coupled processes in subsurface hydroystems*, In Proc. Hydroinformatics 98, pp 613-618, Copenhagen, 24-26 August, Balkema, Rotterdam-Brookfield, ISBN 90 5410 984 X.
- [10] Kolditz O (1999): *Object-oriented methods for hydrosystem modeling*, In: Gekoppelte Prozesse der Fluid- und Festkörpermechanik, International Centre of Computational Engineering Science, ICCES, ISBN 3-9807259-0-1

- [11] Kolditz O, Habbar A, Kaiser R, Rother T & Thorenz C (2001): *ROCKFLOW - Theory and Users Manual*, Release 3.5, Groundwater Modeling Group, Institute of Fluid Mechanics, University of Hannover, Internet publication, (www.rockflow.de).
- [12] Lewis R W & Schrefler B A (1998): *The finite element method in the static and dynamic deformation and consolidation of porous media*, Wiley & Sons, Chichester.
- [13] Rother T, Jüttner U, Kasper H, Kolditz O & Zielke W (1998): *Geometric Modelling of Complex Hydrogeologic Systems*, Technical Report, Institute of Fluid Mechanics and Computer Applications in Civil Engineering, University of Hannover.
- [14] Rother T, Kolditz O, Zielke W & Taniguchi T (2000): *Geometric analysis of fractured-porous aquifers*. Rockflow-Preprint [2000-3], Institute of Fluid Mechanics, University of Hannover, accepted by J Environmental Geology.
- [15] Stroustrup B (1998): *Die C++ Programmiersprache*, Addison-Wesley.
- [16] Taniguchi T, Goda T, Kasper H, Zielke W & Kosakowski G (1996): *Hexahedral Mesh Generation of Complex Composite Domain - Groundwater Flow and Transport Analysis in Fractured Rock*, Beitrag zur Proc. 5th Int. Conf. on Num. Grid Generation in Comp. Field Simulations, April 1-5, 1996, Mississippi State University, Starkville, Mississippi.
- [17] Thorenz C, Kolditz O & Zielke W (1999): *A method of characteristics (MOC) concept for advective tracer transport in fracture networks*, In: Proc. International Conference on Calibration and Reliability in Groundwater Modeling - Model-CARE99, 20-23.09.1999, Zürich).
- [18] Zielke W (Hrsg.) (1999): *Numerische Modelle von Flüssen, Seen und Küstengewässern*, DVWK Schriften, **127**, Wirtschafts und Verlagsgesellschaft Gas und Wasser, Bonn.

Part IV

Selected Topics

Chapter 12

Non-Linear Flow in Fractured Media

12.1 Introduction and Background

This chapter deals with theory and computation of fluid flow in fractured rock. Non-Darcian flow behavior was observed in pumping tests at the geothermal research site at Soultz-sous-Forêts (France). Examples are examined to demonstrate the influence of fracture roughness and pressure-gradient dependent permeability on pressure build-up. A number of test examples based on classical models by Darcy (1856), Blasius (1913), Nikuradse (1930), Lomize (1951) and Louis (1967) are investigated, which may be suited as benchmarks for non-linear flow. This is a prelude of application of the non-linear flow model to real pumping test data. Frequently, conceptual models based on simplified geometric approaches are used. Here, a realistic fracture network model based on borehole data is applied for the numerical simulations. The obtained data fit of the pumping test shows the capability of fracture network models to explain observed hydraulic behavior of fractured rock systems.

12.1.1 Theoretical Issues

Dealing with flow in porous and fractured media we have to consider the mechanics of fluids in tubes and channels. The governing equation for flow of incompressible viscous fluids is the well-known Navier-Stokes equation

$$\underbrace{\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v}}_{\text{inertial terms}} = \mathbf{g} - \frac{1}{\rho} \nabla p + \underbrace{\frac{\mu}{\rho} \nabla^2 \mathbf{v}}_{\text{viscous term}} \quad (12.1)$$

where \mathbf{v} is the local velocity vector of a point, p is pressure, ρ is density, μ is dynamic viscosity of the fluid and \mathbf{g} are external volume forces per unit mass (we consider gravity only).

The Navier-Stokes equation can be integrated for laminar flow in straight, circular tubes when steady-state boundary conditions are applied. In this case convective acceleration term $(\mathbf{v} \cdot \nabla)\mathbf{v}$ becomes zero. This solution is called the Hagen-Poiseuille equation (Lamb 1932).

$$\Delta p = \frac{8\mu L}{\pi R^4} Q \quad (12.2)$$

where Q is the volumetric flow rate, which is velocity multiplied by tube cross-section area, L is tube length and R is tube radius. The linear relationship between pressure drop and flow rate Q breaks down if convective acceleration and/or transient effects become important. The first case is denoted **non-linear laminar flow**, when inertial effects become important, e.g. due to curvature of tubes or channels. The second case is related to **turbulent flow**, when flow pattern become transient due to velocity fluctuations.

Confusion between non-linear laminar flow and 'true' turbulent flow may arise from the fact that - concerning the relationship between pressure drop and the flow rate - inertia effects in laminar flow are expressed in the same fashion as in turbulent flows

$$\Delta p = A Q + B Q^2. \quad (12.3)$$

This means if inertia effects or turbulence effects become significant, the relationship between pressure drop and flow rate (12.2) is no longer linear. Therefore, we have to distinguish between three different flow regimes: linear laminar flow, non-linear laminar flow and 'true' turbulent flow. Equation (12.3) is known as the Forchheimer equation (Forchheimer 1914).

Darcy (1856) found that the volume of fluid percolating through a sand column is proportional to the applied pressure difference

$$Q = qA = A \frac{k}{\mu} \frac{\Delta p}{L} \rightarrow \Delta p = \frac{L}{A} \frac{\mu}{k} Q. \quad (12.4)$$

Comparing the structure of equations (12.4) and (12.2), the analogy between porous media flow and tube flow becomes obvious. Both equations are characterized by linear relationships between pressure drop and flow rate.

Darcy's law can be derived from the Navier-Stokes equations. To this purpose a spatial averaging procedure over a representative elementary volume (REV) has to be conducted, where microscopic quantities are transformed into macroscopic ones (Bear 1972, Diersch 1985)

$$\langle \psi \rangle = \frac{1}{\text{REV}} \int_{\text{REV}} \psi dV \quad (12.5)$$

where ψ is a local, microscopic quantity and $\langle \psi \rangle$ is a spatially averaged macroscopic quantity. For fractures the averaging procedure can be splitted into two steps

$$\langle \psi \rangle = \frac{1}{2b \text{REA}} \int_{-b}^{+b} \int_{\text{REA}} \psi dx dA \quad (12.6)$$

where b is half fracture aperture and REA is a representative elementary area. In the following we deal with quantities which are averaged over fracture thickness and, therefore, are representative for a certain area of fracture surface.

Darcy's law is based essentially on the assumption that fluid motion is inertialess, i.e. inertial terms can be neglected with regard to viscous forces

$$0 = \langle \mathbf{g} \rangle - \frac{1}{\langle \rho \rangle} \nabla \langle p \rangle + \frac{\langle \mu \rangle}{\langle \rho \rangle} \nabla^2 \langle \mathbf{v} \rangle. \quad (12.7)$$

Brackets indicate macroscopic quantities. Thus Darcian flow is a special case of creeping flow for which viscous forces prevail over inertial forces. A central topic in porous medium theory is the determination of the viscous drag term. This leads to the concept of permeability for characterization of the hydromechanical properties of porous media (Scheidegger 1974). Introducing permeability in the following manner

$$\nabla^2 \langle \mathbf{v} \rangle = -\frac{1}{k} \mathbf{q} \quad (12.8)$$

where k is the permeability of the porous medium and \mathbf{q} is the Darcy or seepage velocity, which are macroscopic quantities by definition. Substituting this expression into equation (12.7) we obtain

$$0 = \mathbf{g} - \frac{1}{\langle \rho \rangle} \nabla \langle p \rangle - \frac{\langle \mu \rangle}{\langle \rho \rangle} \frac{1}{k} \mathbf{q}. \quad (12.9)$$

Rearranging the terms, we yield the usual form of Darcy's law. We omit the averaging brackets in the following to keep the notation briefly

$$\mathbf{q} = -\frac{k}{\mu} (\nabla p - \rho \mathbf{g}). \quad (12.10)$$

We emphasize that quantities in the above Darcy equation are macroscopic ones related to a certain REV of a porous medium, whereas quantities in the Navier-Stokes equation (12.1) have local meaning.

Darcy's law has been accepted as fundamental relationship for porous medium hydraulics. However, its validity is restricted to a certain range of geometric and physical conditions. Deviations from linearity between seepage velocity and pressure drop are denoted as non-Darcian flow. Geometric issues are concerned with pore and fracture geometry.

As described above we used the analogy to flow in straight tubes for explanation of hydromechanical processes in porous media. Tube bundles model is one approach to hydromechanics of porous media. However in real geologic materials pores are curved, have varying cross-sections, may be sealed, and suffer from dead-end effects. Rock fractures are characterized by rough surfaces. Physical causes underlying non-linear effects can be high flow rates, molecular effects, ionic effects and non-Newtonian behavior of the fluid itself (Scheidegger 1974, Dullien 1979). This paper will be focused on effects of high flow rates (12.4.2) and fracture roughness (12.4.3) causing additional drags due to increasing inertial forces.

12.1.2 Experimental Issues

In order to evaluate flow regimes, we have to examine the Reynolds number, which relates inertial to viscous forces

$$Re = \frac{\rho q D_h}{\mu} \quad (12.11)$$

where q is Darcy or mean tube velocity, D_h is the hydraulic radius, which is a characteristic length scale of the flow process.

In pipe hydraulics the loss of mechanical flow energy due to skin friction (friction pressure loss) is given by the formula

$$\Delta p = \lambda \frac{L}{D_h} \frac{\rho}{2} q^2 \quad (12.12)$$

which is known as the Darcy-Weisbach relationship (Zielke 1990). A main topic of experimental tube hydraulics is the determination of friction factors λ . Friction or hydraulic resistance coefficient is defined as

$$\lambda = \frac{2D_h}{\rho q^2} \frac{\Delta p}{L} \quad (12.13)$$

where L is column or tube length, respectively.

Substituting the tube flow equation (12.2) into equation (12.13) using the Reynolds number (12.11) we yield the following expression for friction coefficient for tube flow.

$$\lambda(Re) = \frac{64}{Re}. \quad (12.14)$$

For parabolic flow profiles in plane channels we have

$$\lambda(Re) = \frac{96}{Re}. \quad (12.15)$$

The differences result from corresponding mean velocities for circular tubes and parallel plane channels. The above equations relate friction factor and Reynolds

number. The critical parameter for determination of Reynolds numbers and friction coefficients is the characteristic length scale for flow processes - the hydraulic radius D_h . Equation (12.14) is a well-known relationship for flow in straight circular tubes, when selecting $D_h = 2R$. The critical Reynolds number dividing laminar from turbulent flow regimes for that kind of problems is

$$Re_{\text{crit}} = 2300. \quad (12.16)$$

Porous Media

Using Darcy's law (12.4) we can easily derive relationships between Reynolds number, friction factor, seepage velocity and permeability.

$$\begin{aligned} \lambda &= \frac{1}{Re} \frac{2D_h^2}{k} \quad , \quad Re = \frac{1}{\lambda} \frac{2D_h^2}{k} \\ q &= \frac{2\mu D_h}{\rho k} \frac{1}{\lambda} = \frac{Re}{D_h} \quad , \quad k = \frac{1}{\lambda} \frac{2D_h^2}{Re} \end{aligned} \quad (12.17)$$

The characteristic length scale D_h for porous media is based on a mean value of grain diameters. The order of critical Reynolds numbers, when non-linear effects become evident, determined in experiments is about 1-10 (Barenblatt et al. 1990). Therefore, corresponding velocities are only in the range of millimeters or centimeters per second. It is obvious that critical Reynolds numbers for porous media are much less than critical Reynolds numbers of flow through straight tubes with circular cross section (12.11). Therefore, it is questionable that turbulence, i.e. additional drag terms due to Reynolds stresses, resulting from velocity fluctuations in time, is responsible for non-linear flow phenomena. Rather than turbulence, non-linear effects can result from the curvature of pores. That means convective accelerations become important. Transient phenomena related to 'true' turbulence in porous media were observed by Dybbs & Edwards (1984) at Reynolds numbers larger than 100. In general, linear and non-linear flow regimes in porous media do not change immediately. There is a transition zone between the two regimes because of the range of pore radii and the corresponding velocity variations. Couland et al. (1986) found those transition effects by numerical modeling of flow through cylinder arrays. They found that linear flow is valid up to $Re_{\text{crit}}^l < 1$ and a quadratic flow law is valid beginning with $Re_{\text{crit}}^t > 13$. In the intermediate case, there can coexist areas with both linear and non-linear flow regimes.

Fractured Media

Witherspoon et al. (1980) conducted experiments in rock fractures to study the hydraulic behavior. They proved the 'cubic law' for the transmissivity of a fracture consisting of parallel planar plates

$$Q = Cb^3 \Delta p \quad (12.18)$$

where b is the fracture aperture, and C is a constant depending on flow geometry and fluid properties. This relationship means, that fracture flow can be described by Darcy's law using an equivalent hydraulic permeability. When choosing the hydraulic radius for fractures equal to double fracture aperture $D_h = 2b$, we yield a friction factor $\lambda = 96/Re$ (eqn 12.15). Fracture permeability is then $k = b^2/12$ (see Tab. 12.1).

Beside hydraulic conditions, such as high flow rates, geometric properties such as fracture roughness can influence the flow behavior. One of the first comprehensive works on flow through fractures was given by Lomize (1951). He investigated the effect of fracture roughness using parallel glass plates with rough surfaces for his experiments. Lomize (1951) introduced the concept of defining fracture roughness $\epsilon = d/b$ in terms of mean asperities height d . He developed several empirical equations for friction factors as function of both Reynolds number and fracture roughness $\lambda(Re, \epsilon)$ (see Fig. 12.3). If the fracture roughness is larger than $\epsilon > 0.032$, permeability becomes significantly depending of it - $k(D_h, \epsilon)$. Commonly by this critical value of roughness, fracture flow is divided into parallel and non-parallel motion (i.e. velocity components normal to the fracture plane arise from asperities). Moreover, fracture roughness can cause channeling effects in fracture planes (Brown, 1987, Moreno et al., 1988, Tsang, 1992).

Fig. 12.2 shows simulated fracture aperture and resulting velocity distribution within a rough fracture (Kolditz 1997). Fracture aperture distribution was generated using a fractal model. The channelized velocity field gives an idea on the importance of local accelerations in fracture flow.

Table 12.1: Linear flow friction factors and permeabilities

Friction factor λ	Permeability k	
$\frac{96}{Re}$	$\frac{b^2}{12}$	[7]
$\frac{96}{Re} [1 + 8.8\epsilon^{3/2}]$	$\frac{b^2}{12} \frac{1}{1+8.8\epsilon^{3/2}}$	[21]
$\frac{96}{Re} [1 + 6\epsilon^{3/2}]$	$\frac{b^2}{12} \frac{1}{1+17\epsilon^{3/2}}$	[20]

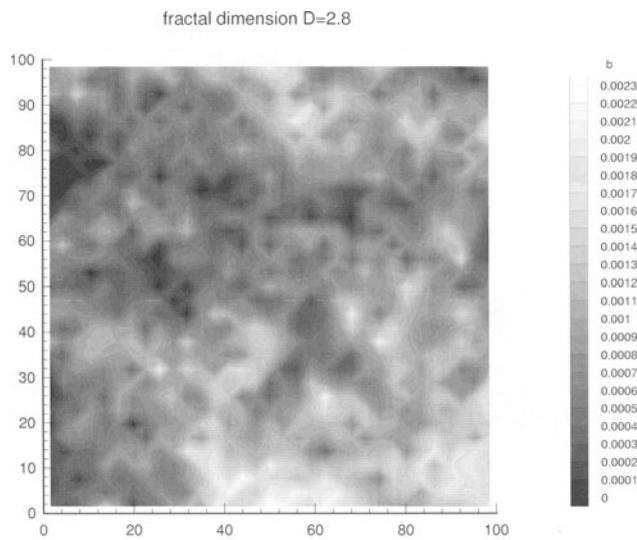


Figure 12.1: Fractal aperture distribution within a rough fracture

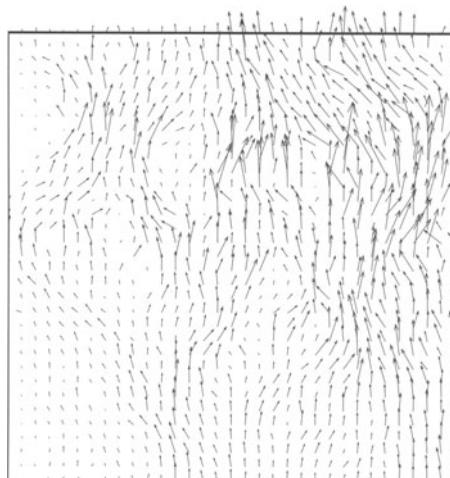


Figure 12.2: Simulated velocity distribution within a rough fracture

Table 12.2: Non-linear flow friction factors and permeabilities

Friction factor λ	Permeability k	
$\lambda = \frac{0.316}{Re^{1/4}}$	$\frac{\mu}{\rho} \left[C \frac{b^5 \rho^4}{\mu} \right]^{1/7} \left[\frac{\partial p}{\partial x} \right]^{-3/7}$	[2]
$\frac{1}{\sqrt{\lambda}} = 2 \log(\frac{\epsilon}{3.7}) $	$\frac{4g\mu\sqrt{b}}{\sqrt{\rho}} \log \frac{3.7}{\epsilon} \left[\frac{\partial p}{\partial x} \right]^{-1/2}$	[23]
$\frac{1}{\sqrt{\lambda}} = 2 \log(\frac{\epsilon}{1.9}) $	$\frac{4g\mu\sqrt{b}}{\sqrt{\rho}} \log \frac{1.9}{\epsilon} \left[\frac{\partial p}{\partial x} \right]^{-1/2}$	[21]
$\frac{1}{\sqrt{\lambda}} = 2.55 \log(\frac{\epsilon}{1.24}) $	$\frac{5.11g\mu\sqrt{b}}{\sqrt{\rho}} \log \frac{1.24}{\epsilon} \left[\frac{\partial p}{\partial x} \right]^{-1/2}$	[20]

Rearranging the quadratic term of the Forchheimer equation (12.3) in following way

$$\Delta p = BQ^2 \quad \Rightarrow \quad Q = \underbrace{\frac{\sqrt{L}}{\sqrt{B\Delta p/L}}}_{k/\mu} \frac{\Delta p}{L} \quad (12.19)$$

we see, that permeability becomes dependent on pressure gradients (Tab. 12.2).

$$k = \frac{\mu \sqrt{\rho D_h}}{\rho \sqrt{\lambda}} \left(\frac{\Delta p}{L} \right)^{-1/2} = \underbrace{\frac{\nu \sqrt{D_h}}{\lambda g}}_{k_0} \left(\frac{\Delta p / \rho g}{L} \right)^{-1/2} \quad (12.20)$$

In contrast to porous media flow, critical Reynolds numbers at the transition from laminar to turbulent flow were found to be in the order of 2300 for fractures (Louis 1967). Romm (1966) and Lomize (1951) reported on the critical value to be even 2400. This means, that critical Reynolds number for fracture flow are in the same order as those for tube flows. These values are about two orders of magnitudes larger than those for porous medium flow.

Analogy between flow through tubes and porous media is manifested by the laws of Darcy-Weisbach (12.12) and by Darcy (12.4). Louis (1967) considered the similarity between channel and fracture flows. He adapted friction factors to fracture flow for linear (Tab. 12.1) and non-linear regimes. (Tab. 12.2). Blasius (1913) originally determined the friction factor for smooth tubes in the range $2320 < Re < 10^5$. Nikuradse (1930) considered slightly rough tubes with non-circular cross-sections. However it is questionable, whether these critical Reynolds numbers are valid for natural fractures providing significant roughness.

As a summary of classic hydraulic approaches, Fig. 12.3 illustrates the above discussed models for linear and non-linear flow in smooth as well as rough fractures.

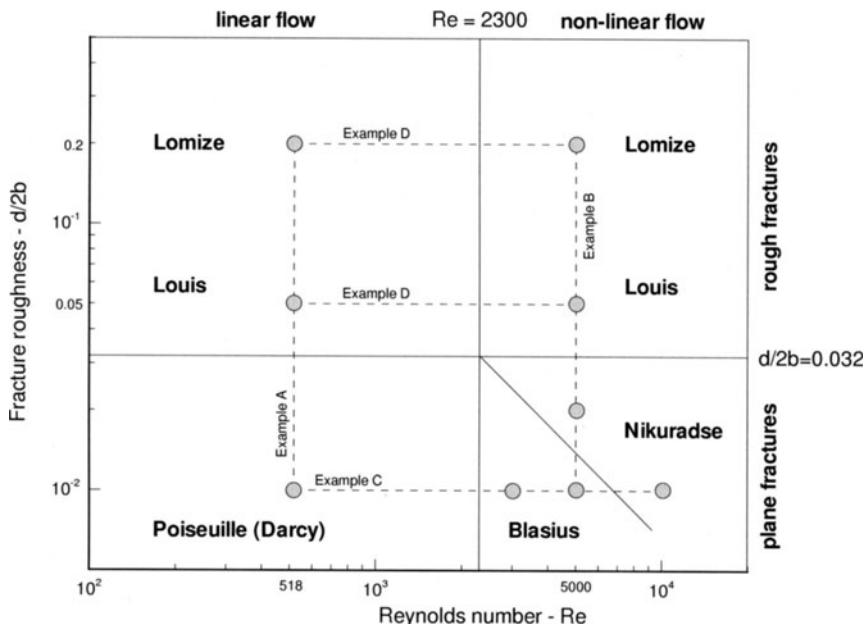


Figure 12.3: Linear and non-linear flow in plane and rough fractures (adapted from Louis (1967))

12.2 Governing Equations

In this section we give a brief summary of the governing equations for fluid flow in fractured-porous media. Readers who are interested in more details of the derivation of the governing equations should refer to Bear (1972), Scheidegger (1974), Pinder & Gray (1977), Diersch (1985), Barenblatt et al. (1990) or Lewis & Schrefler (1998). We are dealing with macroscopic quantities corresponding to the averaging procedure (12.6).

We consider isothermal flow of homogeneous liquids in a porous medium (e.g. saturated groundwater flow). We start from the differential form of the balance equation of the fluid mass. The unknown field function is fluid pressure.

$$S_0^p \frac{\partial p}{\partial t} - \nabla \cdot \mathbf{q} = Q_\rho^p \quad (12.21)$$

In groundwater hydraulics, frequently, piezometric head $h = p/g\rho + z$ (for homogeneous fluids) is used instead of fluid pressure p

$$S_0^h \frac{\partial h}{\partial t} - \nabla \cdot \mathbf{q} = Q_\rho^h \quad (12.22)$$

with: Q_ρ^h - fluid mass source term with respect to piezometric head, Q_ρ^p - fluid mass source term with respect to fluid pressure S_0^p - specific storativity with respect to fluid pressure, S_0^h - specific storativity with respect to piezometric head.

Based on the Forchheimer equation (12.3), we use the following generalization of Darcy's law for non-linear groundwater flow

$$\mathbf{q} = -\mathbf{K}(h, \nabla h) \nabla h. \quad (12.23)$$

where the hydraulic conductivity tensor \mathbf{K} is assumed to be pressure dependent. In general pressure as well as pressure gradient dependencies are assumed for compressible and/or multiphase flows. For groundwater flow (incompressible fluids) it is assumed, that the influence of pressure gradients dominates (e.g. Häfner, 1985, Busch et al., 1993). Then the hydraulic conductivity is given by:

$$\mathbf{K}(\nabla h) = \mathbf{K}_0 |\nabla h|^{\gamma-1} = \mathbf{K}_0 K_{\text{rel}} \quad (12.24)$$

$$\gamma = \begin{cases} > 1 & \text{pre-linear} \\ = 1 & \text{linear (Darcy law)} \\ < 1 & \text{post-linear} \\ & = 1/2 \text{ turbulent friction} \end{cases} \quad (12.25)$$

The non-linear flow parameter γ divides three flow regimes: pre-linear, linear, and post-linear ones. In the pre-linear modus, motion is restricted by electro-molecular forces attracting the fluid molecules to the solid surface. The pre-linear regime is characterized by small pressure gradients $|\nabla h| \ll 1$, whereas the post-linear regime is characterized by large gradients $|\nabla h| \gg 1$. In the post-linear mode, motion is restricted by non-linear friction effects. From the above equation it can be seen, the effective hydraulic resistance K_{rel} increases for smaller pressure gradients in pre-linear situations and for larger pressure gradients in post-linear situations as well.

Returning to fluid pressure p as primary variable we use permeability instead of hydraulic conductivity.

$$\mathbf{k} = \frac{\mu}{\rho g} \mathbf{K} \quad (12.26)$$

In analogy to multiphase flow theory for porous media, we introduce a relative permeability k_{rel} to separate the non-linear term of the permeability tensor

$$\begin{aligned} \mathbf{k} &= \mathbf{k}_0 k_{\text{rel}}(p) \\ k_{\text{rel}}(p) &= |\nabla(p/g\rho + z)|^{\gamma-1}. \end{aligned} \quad (12.27)$$

The equation of fluid motion (12.23) can be rewritten now as

$$\mathbf{q} = -\frac{\mathbf{k}_0}{\mu} k_{\text{rel}} \nabla(p + g\rho z). \quad (12.28)$$

Additionally, we assume homogeneity of the fluid (i.e. constant density and viscosity). Combining both the mass and momentum balance equations we obtain the flow equation

$$S_0^p \frac{\partial p}{\partial t} - \nabla \cdot \left(\frac{\mathbf{k}_0}{\mu} k_{\text{rel}} \nabla(p + g\rho z) \right) = Q_\rho^p. \quad (12.29)$$

As can be seen, a non-linear flow equation results if the non-linear flow parameter γ differs from unity. In the following we use index notation to make tensor operations more obvious.

$$S_0^p \frac{\partial p}{\partial t} - \frac{\partial}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}^0}{\mu} k_{\text{rel}} \frac{\partial}{\partial x_\beta} (p + g\rho z) \right) = Q_\rho^p \\ \alpha, \beta = 1, 2, 3 \quad (12.30)$$

12.3 Numerical Scheme

12.3.1 Galerkin Method

The use of the method of weighted residuals provides an integral formulation of the flow equation to obtain the approximate solution.

$$\int_{\Omega} \omega_i \left[S_0^p \frac{\partial \hat{p}}{\partial t} - \frac{\partial}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}^0}{\mu} k_{\text{rel}} \left(\frac{\partial \hat{p}}{\partial x_\beta} + \rho g \frac{\partial \hat{z}}{\partial x_\beta} \right) \right) \right] d\Omega \\ = \int_{\Omega} \omega_i Q_\rho^p d\Omega \quad (12.31)$$

with: $\omega = \{\omega_i\}$ - weighting functions, Ω - model domain.

The unknown function p is approximated by a trial solution \hat{p} . Note in the following we use Einstein summation convention for repeating indices.

$$p(t, x_\alpha) \approx \hat{p}(t, x_\alpha) \equiv \sum_{j=1}^m \phi_j(x_\alpha) p_j(t) \quad (12.32)$$

$$z(x_\alpha) \approx \hat{z}(x_\alpha) \equiv \sum_{j=1}^m \phi_j(x_\alpha) z_j \quad (12.33)$$

with: $\phi = \{\phi_i\}$ - interpolation functions, p_j - nodal values of pressure, z_j - nodal values of elevation, m - number of nodes.

According to the Bubnov-Galerkin method we use identical weighting ω and interpolation functions ϕ .

$$\begin{aligned} & \int_{\Omega} \phi_i \left[S_0^p \phi_j \frac{dp_j}{dt} \right. \\ & - \left. \frac{\partial}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}^0}{\mu} k_{\text{rel}} \left(\frac{\partial \phi_j}{\partial x_\beta} p_j + \rho g \frac{\partial \phi_j}{\partial x_\beta} z_j \right) \right) \right] d\Omega \\ & = \int_{\Omega} \phi_i Q_\rho^p d\Omega \end{aligned} \quad (12.34)$$

Partial integration is used to reduce the order of the derivatives.

$$\begin{aligned} & \int_{\Omega} \phi_i \left[S_0^p \phi_j \frac{dp_j}{dt} \right. \\ & + \left. \frac{\partial \phi_j}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}^0}{\mu} k_{\text{rel}} \left(\frac{\partial \phi_j}{\partial x_\beta} p_j + \rho g \frac{\partial \phi_j}{\partial x_\beta} z_j \right) \right) \right] d\Omega \\ & = - \int_{\partial\Omega} \phi_i q_n dS + \int_{\Omega} \phi_i Q_\rho d\Omega \end{aligned} \quad (12.35)$$

with the outward flux vector

$$q_n = - \frac{k_{\alpha\beta}^0}{\mu} k_{\text{rel}} \left(\frac{\partial \hat{p}}{\partial x_\beta} + \rho g \frac{\partial z}{\partial x_\beta} \right) n_\alpha \quad (12.36)$$

We rearrange the above equation to put the unknown terms to the left hand side only.

$$\begin{aligned} & \int_{\Omega} \phi_i \left[S_0^p \phi_j \frac{dp_j}{dt} + \frac{\partial \phi_j}{\partial x_\alpha} \frac{k_{\alpha\beta}^0}{\mu} k_{\text{rel}} \frac{\partial \phi_j}{\partial x_\beta} p_j \right] d\Omega = \\ & - \int_{\Omega} \phi_i \frac{\partial \phi_j}{\partial x_\alpha} \frac{k_{\alpha\beta}^0}{\mu} k_{\text{rel}} \rho g \frac{\partial \phi_j}{\partial x_\beta} z_j d\Omega \\ & - \int_{\partial\Omega} \phi_i q_n dS + \int_{\Omega} \phi_i Q_\rho d\Omega \end{aligned} \quad (12.37)$$

This equation forms a global system of equations where the number of equations corresponds to the number of grid points.

$$C_{ij} \frac{dp_j}{dt} + K_{ij}(\hat{p}) p_j = r_i \quad , \quad i, j = 1, \dots, np \quad (12.38)$$

with

$$\begin{aligned} C_{ij} &= \int_{\Omega} \phi_i S_0^p \phi_j d\Omega K_{ij}(\hat{p}) \\ &= \int_{\Omega} \frac{\partial \phi_i}{\partial x_\alpha} \frac{k_{\alpha\beta}^0}{\mu} k_{\text{rel}} \frac{\partial \phi_j}{\partial x_\beta} d\Omega r_i(\hat{p}) \\ &= -\rho g K_{ij}(\hat{p}) z_i - \int_{\partial\Omega} \phi_i q_n dS + \int_{\Omega} \phi_i Q_\rho d\Omega \end{aligned} \quad (12.39)$$

The resolution of non-linearities as well as time discretization are described in section 12.3.4.

12.3.2 Finite Element Approach

Decomposition of the computation domain into finite elements

$$\Omega = \bigcup_{e=1}^{ne} \Omega^e , \quad \partial\Omega = \bigcup_{e=1}^{ne} \partial\Omega^e \quad (12.40)$$

means that the global matrices can be composed by its element contributions.

$$C_{ij} = \sum_{e=1}^{ne} C_{ij}^e , \quad K_{ij} = \sum_{e=1}^{ne} K_{ij}^e , \quad r_i = \sum_{e=1}^{ne} r_i^e \quad (12.41)$$

Interpolation functions ϕ correspond now to individual elements and shape functions \mathbf{N} are used for interpolation at the element level.

$$\hat{p}^e = N_j(x_\alpha, x_\beta) p_j(t) , \quad N_j = \begin{cases} 1 & , j = i \\ 0 & , j \neq i \end{cases} \quad (12.42)$$

Element matrices (12.40) are written in terms of shape functions.

$$\begin{aligned} C_{ij}^e &= \int_{\Omega^e} N_i S_0^p N_j d\Omega \\ K_{ij}^e(\hat{p}) &= \int_{\Omega^e} \frac{\partial N_i}{\partial x_\alpha} \frac{k_{\alpha\beta}}{\mu} k_{\text{rel}} \frac{\partial N_j}{\partial x_\beta} d\Omega^e \\ r_i^e(\hat{p}) &= -\rho g K_{ij}^e(\hat{p}) z_i - \int_{\partial\Omega^e} N_i q_n dS + \int_{\Omega^e} N_i Q_\rho d\Omega \end{aligned} \quad (12.43)$$

12.3.3 Evaluation of Element Matrices in Local Coordinates

Transformation from physical (x, y, z) to local (r, s, t) coordinates, e.g. by isoparametric functions, allows the evaluation of element matrices in unit coordinates. We return at this point to vector notation to keep the equations compactly.

$$\begin{aligned} C_{ij}^e &= \int_{\Omega^e} N_i S_0^p N_j d\Omega \\ \mathbf{C}^e &= \int_{\Omega^e} \mathbf{N} S_0^p \mathbf{N} d\Omega \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} S_0^p \mathbf{N} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \end{aligned} \quad (12.44)$$

$$K_{ij}^e = \int_{\Omega^e} \frac{\partial N_i}{\partial x_\alpha} \frac{k_{\alpha\beta}}{\mu} k_{\text{rel}} \frac{\partial N_j}{\partial x_\beta} d\Omega^e \quad (12.45)$$

$$\begin{aligned} \mathbf{K}^e &= \int_{\Omega^e} \nabla \mathbf{N} \frac{1}{\mu} \mathbf{k}_0 k_{\text{rel}} \nabla \mathbf{N}^T d\Omega^e \\ &= \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \underbrace{\nabla \mathbf{N} (\mathbf{J}^{-1})^T}_{\nabla \mathbf{N}(x,y,z)} \frac{1}{\mu} \underbrace{\mathbf{T}^T \mathbf{k}_0 k_{\text{rel}} \mathbf{T}}_{\mathbf{k}(x,y,z)} \\ &\quad \underbrace{\nabla \mathbf{N}^T \mathbf{J}^{-1}}_{\nabla \mathbf{N}^T(x,y,z)} \underbrace{\det \mathbf{J} dr ds dt}_{d\Omega^e} \\ r_i^e &= -\rho g K_{ij}^e(\hat{p}) z_i - \int_{\partial \Omega^e} N_i q_n dS \\ &\quad + \int_{\Omega^e} N_i Q_\rho d\Omega \quad (12.46) \\ \mathbf{r}^e &= -\rho g \mathbf{K}^e(\hat{p}) \mathbf{z} - \int_{\partial \Omega^e} \mathbf{N} q_n dS \\ &\quad + \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{N} Q_\rho \det \mathbf{J} dr ds dt \end{aligned}$$

with \mathbf{J} the Jacobian matrix and \mathbf{T} the ordinary coordinate transformation matrix. This general finite element approach, described above, allows coupling of multi-dimensional elements (1-D/2-D/3-D) positioned arbitrarily in 3-D space. The specification of above finite element formulations for linear 1-D and bilinear 2-D element types is given in the Appendix.

12.3.4 Resolution of the Non-Linearities

As described before for large flow rates, effects of turbulent friction have to be taken into account. In this case, hydraulic permeability is no longer a constant but it depends on pressure gradient. Non-linear flow behavior results in a non-linear flow equation, which is resolved by use of the Picard (fixpoint) iteration procedure.

The flow equation (12.30) is rewritten to emphasize the solution process:

$$\begin{aligned} S_0^p \frac{\partial p^{[k]}}{\partial t} \\ - \frac{\partial}{\partial x_\alpha} \left(\frac{k_{\alpha\beta}^0 k_{\text{rel}}(p^{[k-1]})}{\mu} \left(\frac{\partial p^{[k]}}{\partial x_\beta} + \rho g \frac{\partial z}{\partial x_\beta} \right) \right) = Q_\rho \quad (12.47) \end{aligned}$$

where $[k]$ is the new iteration level and $[k-1]$ is the prior one. The pressure gradient dependent permeability is evaluated at old iteration level. Velocities can be calculated at each time level by:

$$q_\alpha = \frac{k_{\alpha\beta}^0 k_{\text{rel}}(p^{[k]})}{\mu} \left(\frac{\partial p^{[k]}}{\partial x_\beta} + \rho g \frac{\partial z}{\partial x_\beta} \right) \quad (12.48)$$

where $[k]$ is the final iteration step. For time discretization we return to matrix equations (12.40). All terms are evaluated at the intermediate time point $t + \theta_t \Delta t$.

$$C_{ij} \frac{dp_j}{dt} |_{t+\theta_t \Delta t} + K_{ij}(\hat{p}) p_j |_{t+\theta_t \Delta t} = r_i |_{t+\theta_t \Delta t} \quad (12.49)$$

Linear interpolation in time yields:

$$\begin{aligned} C_{ij} & \left[\frac{p_j(t + \Delta t) - p_j(t)}{\Delta t} \right] \\ & + K_{ij}(\hat{p}) [(1 - \theta_t)p_j(t) + \theta_t p_j(t + \Delta t)] \\ & = [(1 - \theta_t)r_i(t) + \theta_t r_i(t + \Delta t)] \end{aligned} \quad (12.50)$$

where the unknown function and the right-hand-side terms are weighted in time. This way, explicit and implicit schemes can be selected easily.

Rearranging the above terms, new time level to left and old time level to right hand side, we obtain

$$\begin{aligned} & \left[\frac{1}{\Delta t} C_{ij} + \theta_t K_{ij}^{[k-1][n]} \right] p_j^{[k][n]} \\ & = \left[\frac{1}{\Delta t} C_{ij} - (1 - \theta_t) K_{ij}^{[n-1]} \right] p_j^{[n-1]} \\ & + (1 - \theta_t) r_i(t)^{[n-1]} + \theta_t r_i^{[k-1][n]} \end{aligned} \quad (12.51)$$

where $[n], [n - 1]$ are new and old time level, respectively, and $[k], [k - 1]$ are new and old iteration level, respectively.

The numerical scheme is implemented in the object-oriented simulation system Rockflow (Kolditz et al., 1999). In the next two sections we apply the non-linear flow model to single fracture and fracture network simulations.

12.4 Non-Linear Flow in Single Fractures

In this section we consider flow through single fractures with rectangular shape (Fig. 12.4). In particular, we are interested in effects of non-linear flow behavior due to large seepage velocities (section 12.4.2) and effects of fracture roughness (section 12.4.3). Common simulation parameters for all case studies are given in Tab. 12.3. Fracture permeabilities are selected corresponding to the models by Darcy (1856), Blasius (1913), Nikuradse (1930), Lomize (1951) and Louis (1967) (Tab. 12.1 and Tab. 12.2). For large Reynolds numbers fracture permeability depends on both fracture roughness and pressure gradient. Tab. 12.4 presents a summary of specific model parameters.

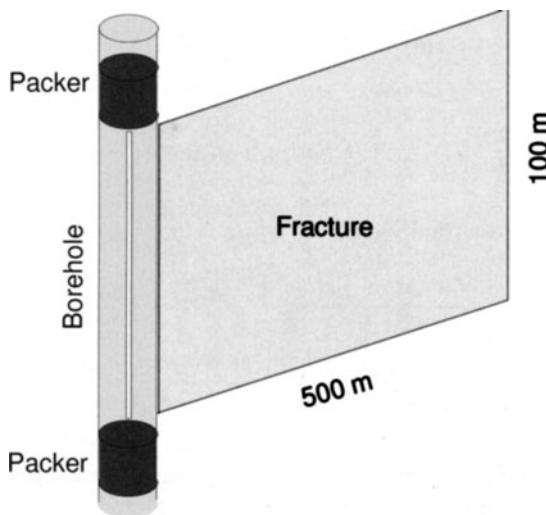


Figure 12.4: Single fracture model

Table 12.3: Common simulation parameters (Water at $T = 130^\circ C$, $p = 35 MPa$)

Symbol	Quantity	Value
L	Fracture length	500 m
H	Fracture height	100 m
b	Hydraulic fracture aperture	$4.1 \times 10^{-4} m$
$S_0^p = S_0^h / g\rho$	Specific storativity of the fracture	$10^{-3} - 10^{-7} Pa^{-1}$
Q	Injection rates	$(6 - 116) \times 10^{-3} m^3 s^{-1}$
μ	Dynamic viscosity of water	$2.2 \times 10^{-4} Pa s$
ρ	Density of water	$950 kg m^{-3}$

12.4.1 Numerical Properties

The first study concerns stability and accuracy of the numerical scheme. To this purpose we have to satisfy the von Neumann criterion for diffusion type equations

$$Ne = \frac{\Delta t D}{\Delta x^2} \leq \frac{1}{2}. \quad (12.52)$$

This stability criterion relates both temporal and spatial discretizations. Diffusivity coefficient for pressure equation (12.30) is defined as

$$D = \frac{\|\mathbf{k}\| k_{\text{rel}}}{\mu S_0^p}. \quad (12.53)$$

Table 12.4: Overview of single fracture case studies

	linear flow	non-linear flow	
very rough	Case A3 $Re = 518$ $d/2b = 0.2$ $K = 0.23543ms^{-1}$ Lomize	Case B4 $Re = 5000$ $d/2b = 0.2$ $K = 0.24803ms^{-1}$ Lomize	
rough	Case A2 $Re = 518$ $d/2b = 0.05$ $K = 0.54026ms^{-1}$ Louis	Case B3 $Re = 5000$ $d/2b = 0.05$ $K = 0.45191ms^{-1}$ Louis	
plane	Case A1/C1 $Re = 518$ $d/2b < 0.032$ $K = 0.59341ms^{-1}$ Darcy	Case C2 $Re = 1123$ $d/2b < 0.032$ $K = 0.59341ms^{-1}$ Darcy	Case B2 $Re = 5000$ $d/2b = 0.02$ $K = 0.59341ms^{-1}$ Nikuradse
$d/2b < 0.032$		Case B1 $Re = 5000$ $d/2b < 0.01$ $K = 0.59341ms^{-1}$ Blasius	Case C4 $Re = 10000$ $d/2b = 0.01$ $K = 0.59341ms^{-1}$ Blasius
	$Re < 2300$	$Re > 2300$	

From criterion (12.52) we derive the following restriction to time step length

$$\Delta t \leq \frac{1}{2} \frac{\Delta x^2 \mu S_0^p}{\|k\| k_{\text{rel}}} . \quad (12.54)$$

Violating the von Neumann criterion yields oscillating numerical solutions. Fig. 12.5 shows simulated pressure build-up curves for different storativity values. As a consequence of the von Neumann criterion we have to enlarge the time step length with increasing storativity values. As a result we obtain identical curves with respect to its shape. The curves are shifted in time corresponding to increased storativity values. This is an appropriate test for the time stepping scheme.

Another test is connected with steady state behavior. For this moment we assume linear flow regime. In this case the steady state pressure level directly corresponds to the Reynolds number. Fig. 12.6 depicts pressure build-up for different fracture velocities (Tab. 12.5). For this study we selected flow rates

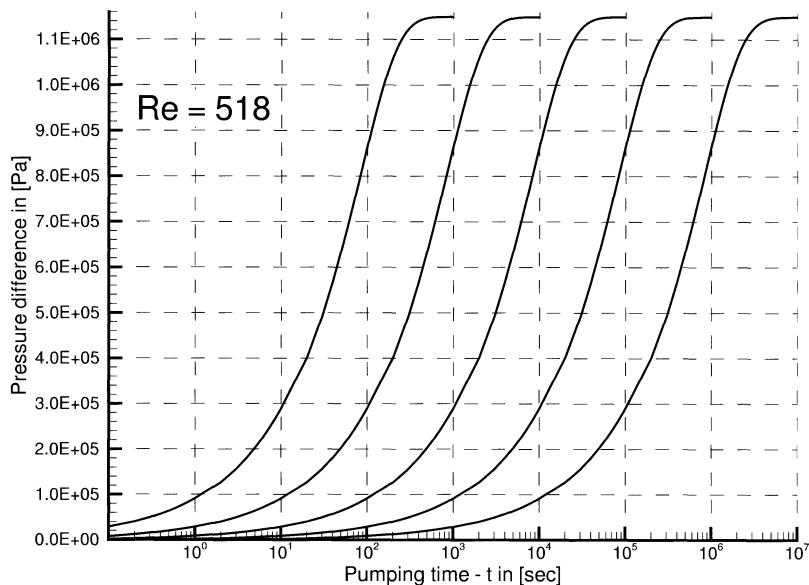


Figure 12.5: Effect of storativity for linear flow in smooth fractures

corresponding to pumping test conditions at the Soultz site and to empirical models of interest (Fig. 12.3). Doubling the Reynolds number (e.g. $Re = 5000 \rightarrow 10000$) has to result in doubled pressure difference (e.g. $p = 1.1 \times 10^7 \rightarrow 2.2 \times 10^7$) at steady state level.

Table 12.5: Flow rates and corresponding Reynolds numbers

Q in $[m^3 s^{-1}]$	Re
6×10^{-3}	518
1.3×10^{-2}	1123
1.9×10^{-2}	1641
2.6×10^{-2}	2245
3.47368×10^{-2}	3000
5.78947×10^{-2}	5000
1.15789×10^{-1}	10000

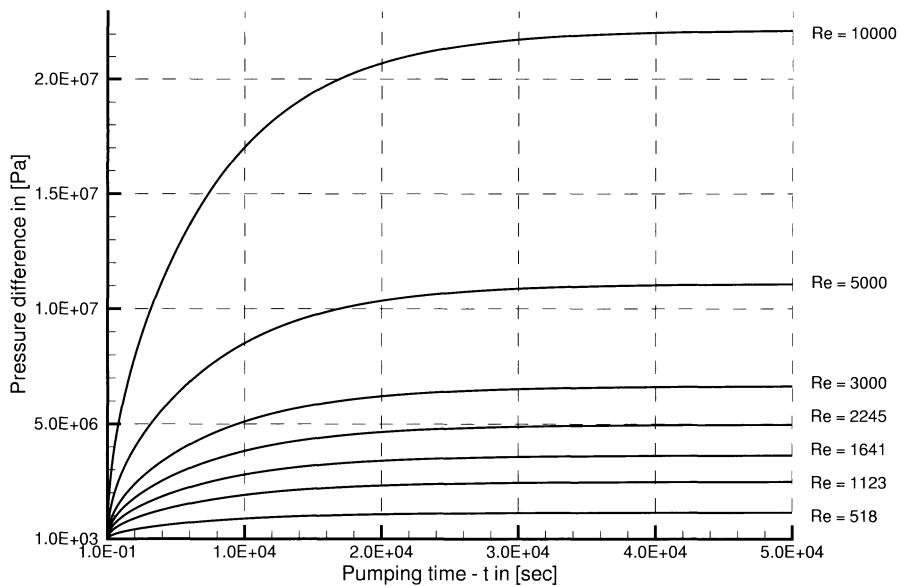


Figure 12.6: Pressure build-up during linear flow in smooth fractures

12.4.2 Effects of Non-Linearity

After getting more confidence in the numerical scheme we turn to effects of non-linear flow behavior. However, we have to keep in mind that the von Neumann criterion is strictly valid only for linear problems.

To investigate linear and non-linear flow regimes in plane fractures (e.g. $d/2b = 0.01$), different flow rates are forced through the fracture (Tab. 12.5). These examples correspond to models by Darcy (1856), Blasius (1913), and Nikuradse (1930) (Fig. 12.3). Fig. 12.7 presents the resulting pressure build-up curves for linear and non-linear flow behavior. We obtain following results:

- Non-linear effects become important for Reynolds numbers larger than $Re > 2245$. For $Re = 2245$ we observe a slight deviation from linear flow behavior in the steady state phase. This is in good agreement with the range of critical Reynolds numbers $Re_{\text{crit}} \in [2300 - 2400]$ found by Lomize (1951), Romm (1966) and Louis (1967) for smooth fractures. With increasing Reynolds numbers the deviation from linear flow behavior becomes more and more significant. For $Re = 10000$ the pressure required to force the 116 liters per second through the fracture is more than 4 times larger due to non-linear flow effects.

- Whereas the pressure increase during the transient stage is very similar (i.e. shapes of curves), non-linear effects have strong influence on the onset of steady state flow. The more non-linear the flow is, the later steady state is reached, i.e. for larger Reynolds numbers steady state flow will be reached later.

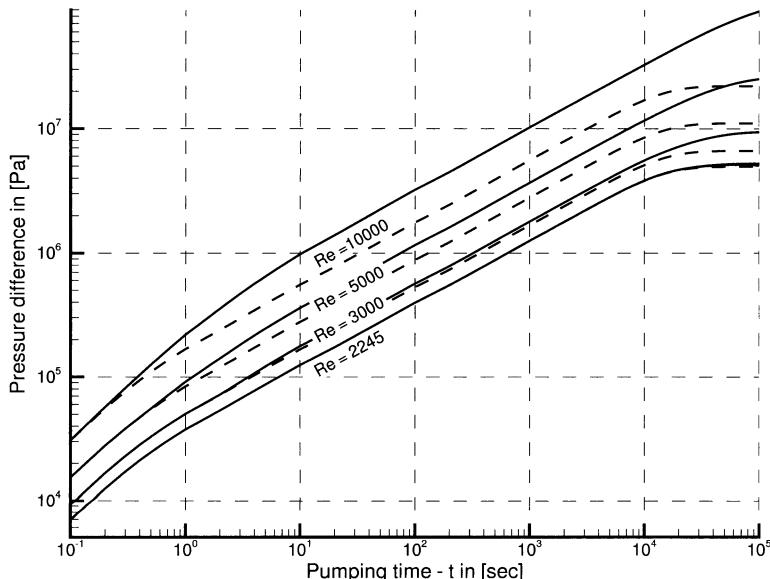


Figure 12.7: Effects of non-linear flow in smooth fractures (Examples C)

12.4.3 Fracture Roughness

Now we study the influence of fracture roughness for linear and non-linear flows.

Fig. 12.8 shows the computed pressure build-up curves for different values of fracture roughness under linear flow conditions (i.e. models by Darcy (1856), Louis (1967), and Lomize (1951)). In the parentheses of the figure captions the simulation cases (capital letters) are given according to Tab. 12.4 and Fig. 12.3.

The next simulations concern pressure transients during non-linear flow. This example was selected to capture the empirical models by Blasius (1913), Nikuradse (1930), Louis (1967), and Lomize (1951) displayed in Fig. 12.3. Reynolds number was chosen to be $Re = 5000$. Certainly, this is a large number for

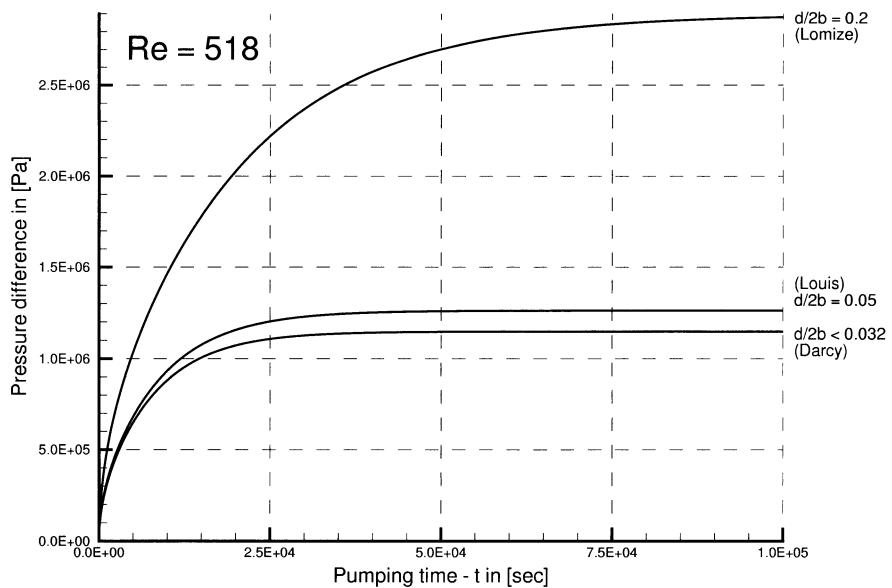


Figure 12.8: Pressure build-up during linear flow for different values of fracture roughness (Examples A1,A2,A3)

pumping test situations in natural fractures, and consequently resulting in large pressure values. Fig. 12.9 shows the computed pressure increase at the fracture entry for different values of fracture roughness. We draw the following conclusions from both examples:

- Reduced hydraulic conductivity due to larger fracture roughness requires larger pressure differences to inject a certain fluid volume into the fracture.
- Steady state flow is reached after considerably larger time than in plane or less rough fractures.

12.5 Non-linear Flow in Fracture Systems

This section deals with non-linear flow behavior in natural fracture systems. Data for this study are from the geothermal research program at Soultz-sous-Forêts in France. Data were provided by the Geoscientific Research Institute (GGA) (Pribnow & Clauser 1999). Non-linear flow behavior was observed from

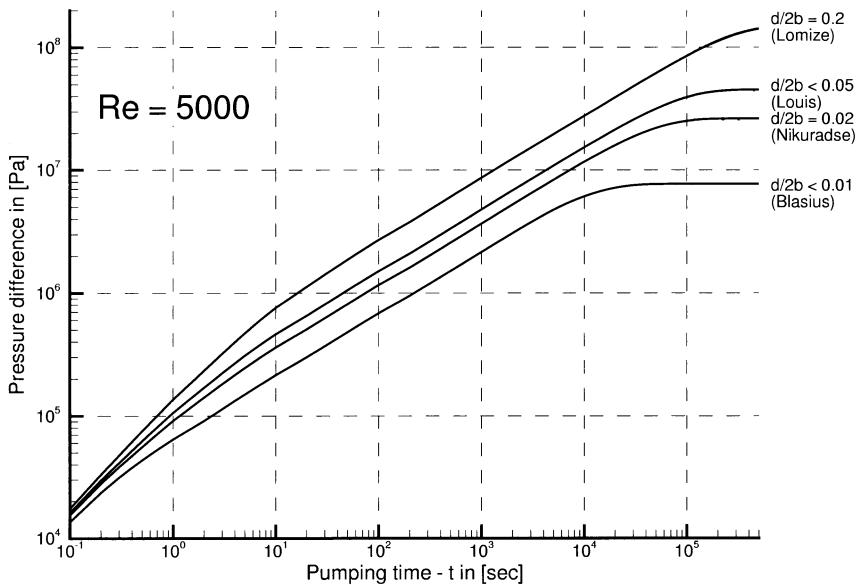


Figure 12.9: Pressure build-up during non-linear flow for different values of fracture roughness (Examples B1,B2,B3,B4)

high-rate pumping tests at the Soultz site (Jung et al. 1995). The subject of this research program is the utilization of geothermal energy from hot dry rock systems. The experimental area is located in deep crystalline rocks about 3500 m in depth. Recently, a new reservoir at 5000 m depth was created. The geometry of the fracture network as reconstructed from borehole logging as well as the generated finite element grid are shown in Fig. 12.10. The finite element discretization consists of 4829 nodes and 5296 quadrilateral elements (Rother et al. 2000). Numerical simulations of non-linear flow and, in particular, of transport processes require sufficiently good mesh qualities.

Kohl et al. (1997) analyzed two prior pumping tests (94JUN16, 94JUL04) conducted in the Soultz geothermal reservoir by using a finite element model. Their geometric model was a conceptual approach consisting of one fracture embedded in the rock matrix. Real geometry of the hydraulically active fracture system was neglected in their studies. The purpose of our study is to investigate non-linear flow behavior using data-based fracture network models (Fig. 12.10).

Fig. 12.11 shows simulations of pumping test 95JUL01 conducted at the Soultz site (Jung et al. 1995). Circles illustrate measured data and solid lines mark simulated pressure for both cases linear flow (Darcy) and non-linear flow (Forch-

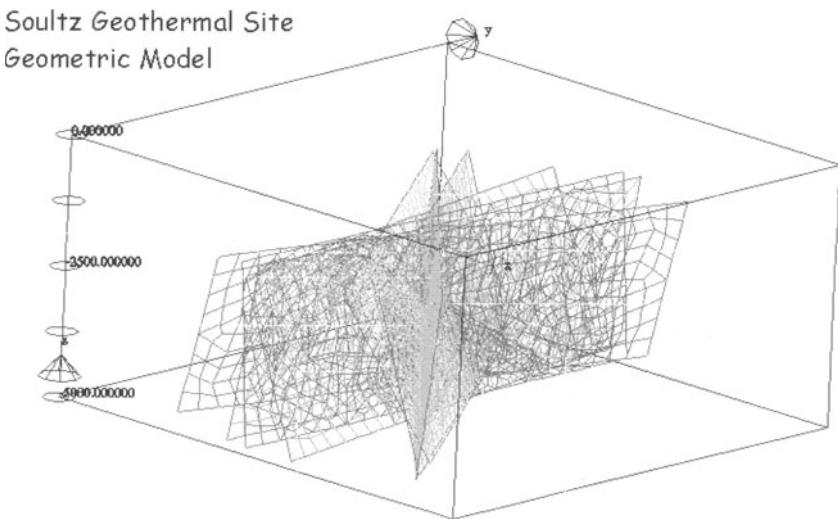


Figure 12.10: Fracture network model and finite element discretization (Rother et al. 2000)

heimer). Linear flow behavior shows a linear relationship between pumping rate increase and corresponding pressure increase to force the fluid volume through the system (lower solid curve in Fig. 12.11). The pumping test clearly indicates non-linear flow behavior. The flow rates of the four steps increase nearly in a linear stepwise way: 6 ls^{-1} , 13 ls^{-1} , 19 ls^{-1} and 26 ls^{-1} . As can be seen from Fig. 12.11 the pressure increase is nearly quadratical. Parameters required for the data fit are given in Tab. 12.6. The range of parameters found by Kohl et al. (1997) for the two prior pumping tests is also plotted in Tab. 12.6. It can be seen that permeability values are close but storativity values differ. Note, storativity values by Kohl et al. (1997) correspond to the rock matrix, whereas our values correspond to the fracture system. In fact, this is the conceptual difference between both models. Kohl et al. (1997) assumed that fluid can be stored in the rock matrix. We think that in the short time scale of the experiment fluid loss into the rock matrix must be very small and, therefore, we assume that the fluid is stored in the fracture system. If relating both storativity values by the rock porosity of about $n = 10^{-3}$, we see that the volume of storable fluid is comparable for both models.

From Fig. 12.11 it can be seen that the pumping test data are well matched by using the non-linear flow model except the shut-off period. During this period fluid pressure is decreasing down to the hydrostatic level. The overestimated pressure drawdown means that the storativity of the reservoir is underestimated in the shut-off period. This indicates storativity changes during the hydraulic

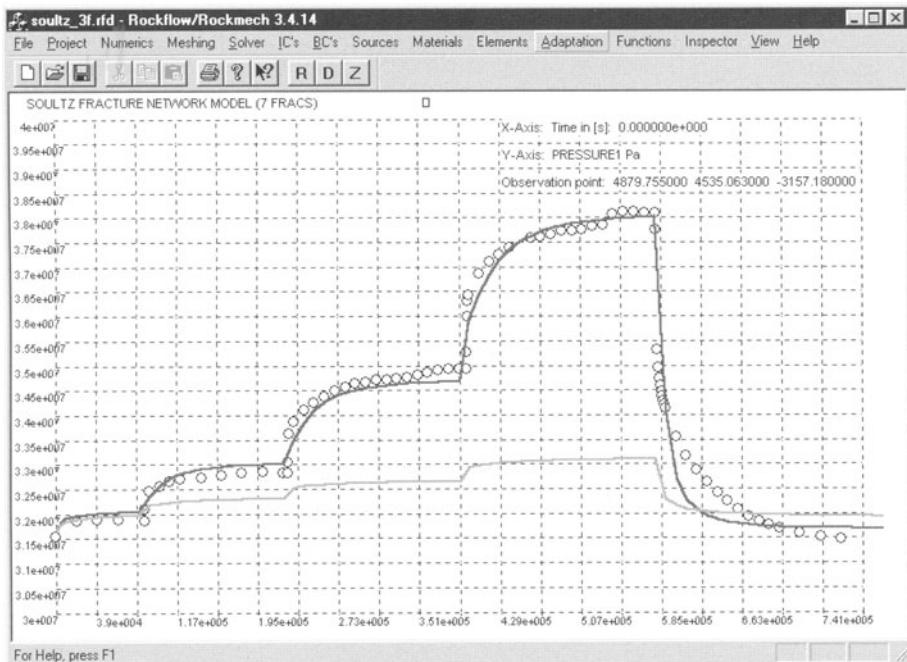


Figure 12.11: Analysis of pumping test data - Darcy versus Forchheimer models

tests. This can be explained with the increased volume of the stimulated fracture system. Because of small relative displacements of rough fracture surfaces during pressure increase they will not close perfectly after reducing the reservoir pressure again. Those effects of residual apertures after loading and unloading of rough fractures was observed e.g. by Brown (1987). We repeated the simulations with increased storativity coefficient in the shut-off period by factor 2 and 3. The corresponding curves in Fig. 12.12 show a better fit now in the shut-off period. However, we should be careful with changing parameters not to change the underlying physical model. If storativity values change dramatically, we have to consider mechanical effects of fracture deformation.

All simulations run with time step length according to the von Neumann criterion (12.52). About 5-10 iterations were needed to achieve satisfactory convergence during updating the pressure gradient dependent permeabilities according to equation (12.27).

These are first results of the pumping test evaluation and are attempted to show that fracture network models in combination with non-linear flow models are capable for analysis of pumping tests in fractured rock. Fracture network models are able to represent both real system geometry and hydromechanical processes. In the specific case of pumping test 95JUL01 the packer interval

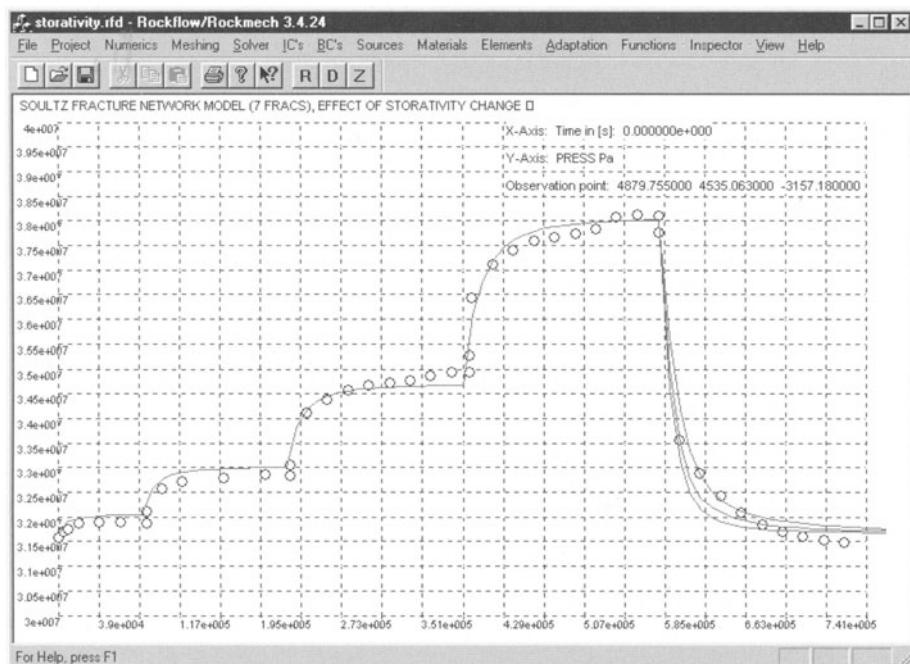


Figure 12.12: Analysis of pumping test data - effect of storativity change

covered two fracture intersections. Therefore, the hydraulic regime is clearly affected by radial flow effects. Accurate approximation of the real flow field is even more important for analysis of tracer transport through natural rock systems. For transport simulations with sharp concentration fronts the use of grid adaptation methods becomes necessary (Kaiser et al., 1999).

Table 12.6: Simulation parameter

Reference Permeability k_0	Specific Storativity S_0^p	Non-linear flow parameter γ
$3.1 \times 10^{-9} m^2$	5×10^{-6} fracture	0.48
$[0.53 - 12] \times 10^{-9} m^2$	$10^{-9} - 2 \times 10^{-11}$ rock matrix	0.5 [15]

12.6 Concluding Remarks

In this paper we have investigated non-linear flow behavior in fractured systems. Single fracture examples are examined to study the influence of non-linear flows effects and fracture roughness. Empirical models by Darcy (1856), Blasius (1913), Nikuradse (1930), Lomize (1951), Louis (1967) are used to calculate the corresponding values of fracture permeability. For smooth fractures we found good agreement between critical Reynolds numbers for onset of non-linear flow behavior known from experiments and from the numerical simulations. Non-linear flow phenomena and fracture roughness lead to reduction of effective permeabilities. Additionally, the onset of steady state flow is significantly delayed. These benchmark simulation for single fractures were a prelude for analysis of pumping test data from the Soultz site. A realistic fracture network model was set up which is based on geologic data. The pumping test data could be well reproduced applying the non-linear flow model. Questions remain with respect to the evaluation of critical Reynolds numbers for natural (rough) fractures, in particular, for more complex flow conditions. Another topic of further investigations is storativity of fracture systems. To this purpose mechanical properties (deformation) should be considered in detail.

12.7 Appendix

1-D Non-Linear Flow Element in \mathcal{R}^3

Pressure dependent permeability for non-linear flow along the 1-D element is given by:

$$\begin{aligned}
 k_{x'}(p) &= k_0 \left| \frac{\partial h}{\partial x'} \right|^{1-\gamma} = k_0 \left| \frac{\partial}{\partial x'} \left(\frac{p}{g\rho} + z \right) \right|^{1-\gamma} \\
 &= k_0 \left| [J_{1D}^{-1}] [\nabla_r N]^T \left(\frac{\hat{p}}{g\rho} + \hat{z} \right) \right|^{1-\gamma} \\
 &= k_0 \left| \frac{(p_2 - p_1)/g\rho + (z_2 - z_1)}{L} \right|^{1-\gamma} \\
 &= k_0 k_{\text{rel}}(p)
 \end{aligned} \tag{12.55}$$

The element conductivity matrix for non-linear flow can be developed explicitly for a 1-D linear finite element:

$$\begin{aligned}
 [K_{1D}^e] &= \int_{\Omega^e} \left[\begin{array}{c} \nabla N_1 \\ \nabla N_2 \end{array} \right] \frac{k_{x'}^e(h)}{\mu} \left[\begin{array}{cc} \nabla N_1 & \nabla N_2 \end{array} \right] d\Omega^e \\
 &= A^e \int_{L^e} \left[\begin{array}{c} \partial N_1 / \partial x' \\ \partial N_2 / \partial x' \end{array} \right] \frac{k_{x'}^e(h)}{\mu} \left[\begin{array}{cc} \partial N_1 / \partial x' & \partial N_2 / \partial x' \end{array} \right] dx'
 \end{aligned}$$

$$\begin{aligned}
&= \frac{k_0^e}{\mu} \left| \frac{p_2 - p_1}{g\rho} + (z_2 - z_1) \right|^{1-\gamma} \frac{A^e}{(L^e)^{1-\gamma}} \times \\
&\quad \int_{-1}^{+1} \left[\begin{array}{c} \partial N_1 / \partial r \\ \partial N_2 / \partial r \end{array} \right] [J_{1D}^{-1}]^T [J_{1D}^{-1}] \left[\begin{array}{cc} \partial N_1 / \partial r & \partial N_2 / \partial r \end{array} \right] \det[J_{1D}] dr \\
&= \frac{k_0^e}{\mu} \left| \frac{p_2 - p_1}{g\rho} + (z_2 - z_1) \right|^{1-\gamma} \frac{A^e}{(L^e)^{2-\gamma}} \left[\begin{array}{cc} 1 & -1 \\ -1 & 1 \end{array} \right] \\
&= k_{\text{rel}}(p) \frac{k_0^e}{\mu_0} \frac{A^e}{L^e} \left[\begin{array}{cc} 1 & -1 \\ -1 & 1 \end{array} \right]
\end{aligned} \tag{12.56}$$

A pressure based formulation is preferred, therefore, the gravity term has to be evaluated separately. This can be done easily using the above element conductivity matrix:

$$\begin{aligned}
\{g_{1D}^e\} &= [K_{1D}^e] \times g\rho\{z\} \\
&= \int_{\Omega^e} \left[\begin{array}{c} \nabla N_1 \\ \nabla N_2 \end{array} \right] \frac{k_{x'}^e(p)}{\mu} g\rho \left[\begin{array}{cc} \nabla N_1 & \nabla N_2 \end{array} \right] d\Omega^e \times \{z\} \\
&= k_{\text{rel}}(p) \frac{k_0^e}{\mu_0} g\rho(z_1 - z_2) \frac{A^e}{L^e} \left[\begin{array}{c} 1 \\ -1 \end{array} \right]
\end{aligned} \tag{12.57}$$

2D Non-Linear Flow Element in \mathcal{R}^3

Using a pressure based formulation we have to split the piezometric head gradient in the following way:

$$\left[\begin{array}{c} \frac{\partial h}{\partial x'} \\ \frac{\partial h}{\partial y'} \end{array} \right] = \{\nabla_{x'y'}h\} = \{\nabla_{x'y'}(\frac{p}{g\rho} + z)\} = [J_{2D}^{-1}][\nabla_{rs}N]^T \left(\frac{\{p\}}{g\rho} + \{z\} \right)
\tag{12.58}$$

For convenience we formulate the pressure gradient dependent permeability for following cases:

- General anisotropic case

$$\left[\begin{array}{cc} k_{x'x'}(p) & k_{x'y'}(p) \\ k_{y'y'}(p) & k_{y'x'}(p) \end{array} \right] = [\mathbf{k}'(p)] = [\mathbf{k}'_0] \left[[J_{2D}^{-1}][\nabla_{rs}N]^T \left(\frac{\{p\}}{g\rho} + \{z\} \right) \right]^{1-\gamma}
\tag{12.59}$$

- Orthotropic case ($k_{x'y'} = k_{y'x'} = 0$)

$$\begin{bmatrix} k_{x'x'}(p) \\ k_{y'y'}(p) \end{bmatrix} = k'_0 \left[[J_{2D}^{-1}] [\nabla_{rs} N]^T \left(\frac{\{p\}}{g\rho} + \{z\} \right) \right]^{1-\gamma} = k'_0 \begin{bmatrix} k_{x'x'}^{\text{rel}}(p) \\ k_{y'y'}^{\text{rel}}(p) \end{bmatrix} \quad (12.60)$$

- Isotropic case

$$k'(p) = k'_0 \left[\sqrt{\left(\frac{\partial(p/g\rho + z)}{\partial x'} \right)^2 + \left(\frac{\partial(p/g\rho + z)}{\partial y'} \right)^2} \right]^{1-\gamma} = k'_0 k_{\text{rel}}(p) \quad (12.61)$$

For the orthotropic case the 2-D element conductance matrix is given by:

$$\begin{aligned} [K_{2D}^e] &= \int_{\Omega^e} \begin{bmatrix} \nabla N_1 \\ \nabla N_2 \\ \nabla N_3 \\ \nabla N_4 \end{bmatrix} \frac{k'_0}{\mu} \begin{bmatrix} k_{x'x'}^{\text{rel}}(p) & 0 \\ 0 & k_{y'y'}^{\text{rel}}(p) \end{bmatrix} \begin{bmatrix} \nabla N_1 & \nabla N_2 & \nabla N_3 & \nabla N_4 \end{bmatrix} d\Omega^e \\ &= L^e \int_{-1}^{+1} \int_{-1}^{+1} [N] [J_{2D}^{-1}]^T \frac{k'_0}{\mu} \begin{bmatrix} k_{x'x'}^{\text{rel}}(p) & 0 \\ 0 & k_{y'y'}^{\text{rel}}(p) \end{bmatrix} [J_{2D}^{-1}] [N]^T \det[J_{2D}] dr ds \\ &= L^e \sum_{i=1}^{ngp} \sum_{j=1}^{ngp} g_i g_j [N] [J_{2D}^{-1}]^T \frac{k'_0}{\mu} \begin{bmatrix} k_{x'x'}^{\text{rel}}(p) & 0 \\ 0 & k_{y'y'}^{\text{rel}}(p) \end{bmatrix} [J_{2D}^{-1}] [N]^T \det[J_{2D}] \end{aligned} \quad (12.62)$$

The terms has to evaluated in the corresponding Gaussian points. The gravity term can be easily evaluated using the above element conductivity matrix:

$$\{g_{2D}^e\} = [K_{2D}^e(p)] \times g\rho\{z\} \quad (12.63)$$

Bibliography

- [1] Bear, J. (1972), *Dynamics of fluids in porous media*, American Elsevier.
- [2] Blasius, H. (1913), *Das Ähnlichkeitsgesetz bei Reibungsvorgängen in Flüssigkeiten*, Forsch. Ing.-Wesen, VDI-Heft 131.
- [3] Barenblatt, G.I., Entov, V.M. & Ryzhik, V.M. (1990), *Theory of fluid flows through natural rocks*, Kluwer Academic Publishers.
- [4] Brown, S.R. (1987), "Fluid flow through rock joints: The effect of surface roughness", *J Geophysical Research*, Vol. 92, No. B2, pp. 1337-1347.
- [5] Busch, K.-F., Luckner, L. & Thiemer, K. (1993), *Geohydraulik*, In: Matthess, G. (Hrsg.), Lehrbuch der Hydrogeologie, Band 3, Gebrüder Borntraeger Verlag, Berlin - Stuttgart.
- [6] Couland, O., Morel, P. & Caltagirone, J.-P. (1986), "Effects non lineaires dans les écoulements en milieu poreux", *C.R. Acad. Sci.*, ser. 2, vol 302, pp. 263-266.
- [7] Darcy, H. (1856), *Les fontaines publiques de la ville de Dijon*, Dalmont, Paris.
- [8] Diersch, H.-J. G. (1985), *Modellierung und numerische Simulation geohydrodynamischer Transportprozesse*, Habilitationsschrift, Akademie der Wissenschaften der DDR, Berlin.
- [9] Dullien, F.A.L. (1979), *Porous media, fluid transport and pore structure*, Academic Press, New York.
- [10] Dybbs, A. & Edwards, R.V. (1984), *A new look at porous media fluid mechanics - Darcy to turbulent*, in Bear & Corapcioglu (eds), *Fundamentals of transport phenomena in porous media*, pp. 201-256. Dordrecht Publisher.
- [11] Forchheimer, P. (1914), *Hydraulik*, Teubner Verlag, Berlin-Leipzig.
- [12] Häfner, F. (1985), *Geohydrodynamische Erkundung von Erdöl-, Erdgas- und Grundwasserlagerstätten*, Wissenschaftlich-Technischer Informationsdienst des Zentralen Geologischen Instituts, Vol. 26, Berlin.
- [13] Jung, R., Willis-Richards, J., Nicholls, J.D., Bertozzi, A. & Heinemann, B. (1995), "Evaluation of hydraulic test at Soultz - European HDR site", In: Proc. Worlds Geothermal Congress, pp. 2575-2580, Florenz.
- [14] Kaiser, R., Kolditz, O. & Rother, T. (1999), "Modelling of flow in fractured aquifers using automatic grid adaptation", In: Proc. XXVIII IAHR Conference on Hydraulic Engineering for Sustainable Water Resources Management, Graz.
- [15] Kohl, T., Evans, K.F., Hopkirk, R.J., Jung, R. & Rybach, L. (1997), "Observation and simulation of non-Darcian flow transients in fractured rock", *Water Res. Research*, Vol. 33 No. 3, pp. 407-418.
- [16] Kolditz, O. (1997), *Strömung, Stoff- und Wärmevertrag im Kluftgestein*, Borntraeger Verlag, Berlin-Stuttgart.
- [17] Kolditz, O. & Clauser, C. (1998): "Numerical simulation of flow and heat transfer in fractured crystalline rocks: Application to the hot dry rock site at Rosemanowes (UK)", *Geothermics*, 27(1): 1-23.
- [18] Kolditz, O., Habbar, A., Kaiser, R., Rother, T. & Thorenz, C. (1999), *ROCKFLOW - Theory and Users Manual*, Release 3.4, Groundwater Modeling Group, Institute of Fluid Mechanics, University of Hannover, (www.rockflow.de).

- [19] Lamb, H. (1932), *Hydrodynamics*, Cambridge University Press.
- [20] Lewis R W & Schrefler B A (1998): *The finite element method in the static and dynamic deformation and consolidation of porous media*, Wiley & Sons.
- [21] Lomize, G.M. (1951), *Seepage in fissured rocks*, State Press, Moscow-Leningrad.
- [22] Louis, C. (1967), *Strömungsvorgänge in klüftigen Medien und ihre Wirkung auf die Standsicherheit von Bauwerken und Böschungen im Fels*, Mitteilungen des Instituts für Boden- und Felsmechanik, Heft 23, Universität Karlsruhe.
- [23] Moreno, L., Tsang, Y.W., Tsang, C.F., Hale, F.V.& Neretnieks, I. (1988), "Flow and tracer transport in a single fracture: A stochastic model and its relation to some field observations", Water Res. Research, Vol. 24 No. 12, pp. 2033-2048.
- [24] Nikuradse, J. (1930), "Turbulente Strömungen in nicht-kreisförmigen Rohren", Ing. Arch., Vol. 1, pp. 306-332.
- [25] Pinder, G.F. & Gray, W.G. (1977), *Finite element simulation in surface and subsurface hydrology*, Academic Press, New York-London.
- [26] Pribnow, D. & Clauser, C. (1999), *Heat and fluid flow in the Rhein graben: Regional and local models for a HDR system*, Technical Report, Institut für Geowissenschaftliche Gemeinschaftsaufgaben (GGA),
- [27] Romm, E.S. (1966), *Seepage properties of fractured rock*, (in Russian), Nedra Publisher, Moscow.
- [28] Rother, T., Kolditz, O., Zielke, W. & Taniguchi, T. (2000), *Geometric analysis of fractured-porous aquifers*, Rockflow-Preprint [2000-3], Groundwater Modeling Group, Institute of Fluid Mechanics, University of Hannover, accepted for publication by J Environmental Geology.
- [29] Scheidegger A. E. (1974), *The physics of flow through porous media*, University of Toronto Press, 3rd edition.
- [30] Tsang, Y.W. (1992), "Usages of equivalent apertures for rock fractures as derived from hydraulic and tracer tests", Water Res. Research, Vol. 28 No. 5, pp. 1451-1455.
- [31] Witherspoon, P.A., Wang, J.S.Y., Iwai, K. & Gale, J.E. (1980), "Validity of cubic law for fluid flow in a deformable rock fracture", Water Res. Research, Vol. 16 No. 6, pp. 1016-1024.
- [32] Zielke, W. (1990), *Strömungsmechanik für Bauingenieure*, Eigenverlag, Institut für Strömungsmechanik, Universität Hannover.

Chapter 13

Heat Transport in Fractured-Porous Media

In this chapter we examine heat transfer during forced water circulation through fractured crystalline rock using a fully 3-D finite-element model. We propose an alternative to strongly simplified single or multiple parallel fracture models or porous media equivalents on the one hand, and to structurally complex stochastic fracture network models on the other hand. The applicability of this "deterministic fracture network approach" is demonstrated in an analysis of the 900-day circulation test for the Hot Dry Rock (HDR) site at Rosemanowes (UK). The model design in respect to structure, hydraulic and thermal behavior is strictly conditioned by measured data such as fracture network geometry, hydraulic and thermal boundary and initial conditions, hydraulic reservoir impedance, and thermal drawdown. Another novel feature of this model is that flow and heat transport in the fractured medium are simulated in a truly 3-D system of fully coupled discrete fractures and porous matrix. While an optimum model fit is not the prime target of this study, this approach permits to make realistic long-term predictions of the thermal performance of HDR systems.

13.1 Introduction

Geothermal Hot Dry Rock (HDR) systems have been studied since 1970. Various mathematical models were proposed to analyze the data obtained from field experiments in order to predict reservoir long-term performance. Several conceptual models were developed to simulate mechanical, hydraulic, and thermal processes in HDR systems. These can be classified in respect to the way in which the fractured reservoir geometry is simplified: single fracture, parallel fracture arrays, fracture networks, or equivalent porous media. Both deterministic and

stochastic approaches were used. Analytical methods are available for the calculation of the heat extracted from a geothermal reservoir by steady-state flow in a single fracture and in parallel fracture systems (Lauwerier, 1955; Gringarten & Sauty, 1975; Häfner et al., 1992). Rodemann (1979), Schulz (1987) and Heuer et al. (1991) extended these analytical tools for the case of finite fractures. Rowe et al. (1989) presented a semi-analytical solution for contaminant migration in a regular, orthogonally fractured system. However, the simulation of the coupled hydraulic, thermal and mechanical processes during heat extraction requires numerical methods (e.g. Sekine and Mura, 1980; Noorishad et al., 1982; Pine and Cundall, 1985; Vaughan, 1987; Verma and Pruess, 1988; Diersch et al. 1989; Nicol et al., 1990; Kohl, 1992; Kolditz and Diersch, 1993). These generic models for simple reservoir geometries are useful for parameter studies and for gaining an understanding of the physics of the involved highly coupled, non-linear processes.

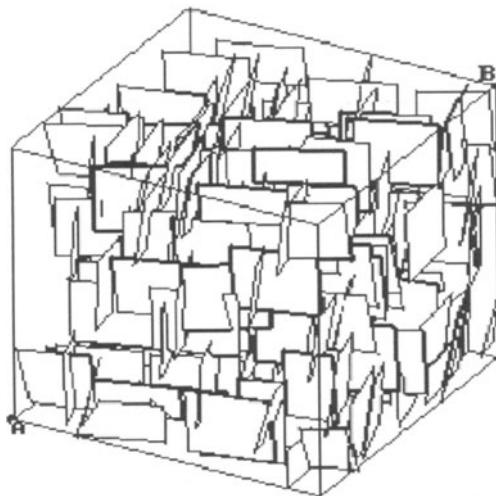


Figure 13.1: Stochastic fracture network designed on the base of data available for the Rosemanowes HDR site (details see text).

Information on fracture orientation, (apparent) apertures, dimensions, and fracture densities are available from borehole logs (e.g. formation-microscanner (FMS) or the borehole televIEWer (BHTV)), drill cores, and fracture traces in mines or quarries. This can be used for a realistically geometric description of fracture systems. This information is statistical in nature and especially suited as input for stochastic models. As an example, Fig. 13.1 shows a part of a synthesized fracture network established on the base of data for the HDR site

in Rosemanowes. In constructing this spatial fracture network it is assumed that 1-D data from borehole-logs may be extrapolated in three dimensions to a given model domain of interest. The volumetric dimension of the accessible HDR reservoir may be estimated by monitoring micro-seismic events during hydraulic stimulations. Typical fracture densities detected from BHTV- or FMS-logs are on the order of 10 m^{-1} . Depending on the extent of the fractures in question, model regions of practical interest (about 1 km^3) are therefore likely to contain several thousands of fractures. Commonly, Monte Carlo techniques are used in order to determine the average and variance of the system's behavior. To this end, a great number of geometric realizations is generated which consist of different, but statistically equivalent fracture networks. Stochastic fracture network modeling was introduced by several groups (e.g. Long 1983; Andersson et al., 1984; Dershowitz, 1985) investigating the migration of chemical species from waste disposals in the bedrock. In this case, the dominant transport mechanisms were shown to be advection and dispersion in the fissures, and migration is retarded by adsorption and decay. For the purpose of worst-case predictions, the exchange of contaminants between flow paths and the rock matrix is often neglected. This simplification permits the use of models with reduced dimension (e.g. 2.5-D), which do not account for the storativity of the rock matrix. However, the importance of matrix diffusion, especially for long-term forecasts, was frequently stressed (e.g. Neretnieks, 1980). In the case of heat transfer, diffusion is the dominating process transporting heat from the matrix to the flow system in fractured rocks. Bruel & Cacas (1992) and Lanyon & Batchelor (1992) introduced stochastic modeling of 3-D fracture networks for geothermal problems. However, stochastic models suffer from their high degree of freedom (number of free parameters). Therefore, fractal methods were applied to constrain the variability of the statistical parameters (Watanabe and Takahashi, 1993; Kosakowski, 1996). In order to limit the computational effort, heat exchange between fractures and rock matrix is often approximated by simplified transfer functions or 1-D diffusion. Kolditz (1995a) studied the dimensional effect of matrix heat diffusion, illustrating the need for a truly 3-D treatment of the matrix heat diffusion. In field studies, the reliable calibration of thermal models requires data from long-term circulation experiments, where a considerable amount of heat is removed from the reservoir. Birdsell and Robinson (1988) examined such circulation data from the Fenton Hill site (USA) using both a multiple flow-path model and an equivalent porous medium approach. Robinson and Brown (1990) applied a 2-D fracture network model to simulate the transient fluid flow behavior in a fractured rock mass. They included a pressure-dependent aperture submodel. Using a piston-flow transport model (heat sweep model), where heat is supplied to the fractures by diffusion from the rock matrix, Robinson and Kruger (1992) predicted temperature decline at the Fenton Hill site on the basis of data from the long-term flow test. DuTeau and Swenson (1994) used a 2-D fracture network model of the Fenton Hill HDR site to illustrate the importance of hydraulic, thermal, and mechanical coupling for the development of channeling effects. Nicol and Robinson (1990) analyzed the long-term circulation test at the Rosemanowes HDR site using a parallel frac-

ture model and obtained an acceptable model fit. The reservoir performance was shown to be sensitive to variations in the model's key parameters, such as flow rates, fracture areas, and reservoir volume. Recently, Bruel (1995) presented a stochastic model in a new analysis of hydraulic and thermal data from the long-term circulation test at the Rosemanowes site. The stochastic reservoir model (on the basis of 25 probabilistic realizations of the fracture network) had to be constrained by deterministic information in order to keep the computational effort in acceptable bounds. Bruel (1995) showed, that only a limited number of preferential flow paths determine the total hydraulic performance of the reservoir. Setting up large deterministic models or generating many realizations of stochastic models requires a great amount of time. Also, running both inverse and forward models of flow and transport in large networks is extremely time consuming. While analytical solutions are available only for extremely simple fracture configurations (single or parallel plane fractures), stochastic models require more experimental data for calibration than are generally available. In this study, we therefore propose to simplify the geometry of the fracture network by conditioning a deterministic fracture network model (Kolditz, 1995b) by measured data. The proposed deterministic fracture model introduces only as much complexity into the fracture network configuration as can be constrained by measured data. In this way it takes an intermediate position between oversimplified and over-parameterized approaches (Fig. 13.2). The specific merits of our model approach will be contrasted with results both from a parallel fracture model (Nicol and Robinson, 1990) and a stochastic fracture network model (Bruel, 1995).

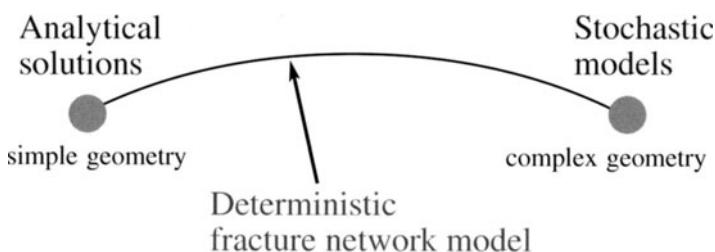


Figure 13.2: Geometric complexity of the deterministic fracture network model

Conceptual 3-D Deterministic Fracture Network Model

Many field experiments demonstrate the strong correlation between the tectonic stress field, fracture opening, and flow distribution in stimulated crystalline rocks (Rummel and Baumgärtner, 1991). From hydraulic tests within the HDR

site at Soultz-sous-Forêts (France) Jung (1991) concludes that during stimulation healed discontinuities normal to the actual least compressing stresses will open again. Thus, new permeability is created. Although individual re-opened fractures may be oriented differently, statistical analyses of the available fracture data often indicates fracture clustering, i.e. the existence of several sets of fractures with preferential directions. The individual fractures of such a set scatter around a mean strike direction. As seen above, fracture densities as high as 10 m^{-1} are not uncommon in crystalline rocks. However, many of these fractures apparent on the borehole wall are healed by mineral precipitation and are hydraulically inactive. The important quantity for the characterization of flow and transport in a reservoir is thus the spacing of those active fractures that actually can accommodate appreciable quantities of water. These hydrothermally relevant fractures which intersect the borehole can be identified, e.g. using the thermal flow meter method (Murphy, 1977; Schellschmidt and Schulz, 1991). Analyses of tracer breakthrough and thermal drawdown provide estimates for the characteristic spacing of fractures within the reservoir. Characteristic values are on the order of some tens of meters (Nicol and Robinson, 1990). In our approach the following assumptions and simplifications are therefore invoked, which reflect the experimental evidence about the orientation, distribution, and spacing of hydrothermally relevant fractures: (1) Fracture orientation - Flow and advective transport are restricted to discrete fracture systems, whose orientation is governed by the tectonic stress field. As a consequence, the main flow paths are developing orthogonal to the direction of minimum principal stresses. Secondary flow paths, not along strike with the maximum direction of principal stress, may be created by shear-stress induced fracture slipping. Within the deterministic approach all flow paths will be oriented at well-defined directions of average strike and dip for each fracture set; (2) Fracture distribution - Fracture clusters are represented by large fractures, which represent the average flow paths; (3) Fracture spacing - Fractures are assumed to be spaced equidistantly in hydrothermally relevant intervals (Kolditz, 1995b). If data are available, only hydraulically active fractures are considered which are identified by appropriate logs. The rock matrix between discrete fractures is treated as a porous medium with hydraulic and thermal properties assumed to represent the network of microfissures smaller than the scale of the flow paths.

13.2 Governing Equations

The equations governing fluid flow and heat transport in a fractured porous medium are derived from the conservation laws of fluid mass, momentum, and thermal energy. The steady-state conservation of mass for an incompressible fluid of pore velocity \mathbf{v} in a medium with an effective porosity n is expressed by,

$$\nabla \cdot (n\mathbf{v}) = 0. \quad (13.1)$$

The conservation of momentum is described by Darcy's law,

$$\mathbf{q} = n\mathbf{v} = -\nabla \cdot (\mathbf{K}h) \quad (13.2)$$

where \mathbf{q} denotes specific discharge, \mathbf{K} hydraulic conductivity, and h hydraulic head. A fully developed Hagen-Poiseuille flow is assumed for the fractures, leading to a Darcy-law analogy for the averaged fluid velocity. Thus an equivalent hydraulic permeability $k = b^2/12$ can be specified for fractures with hydraulic aperture b . From this the value for the corresponding hydraulic conductivity follows from $\mathbf{K} = \mathbf{kg}\rho_w/\mu$, where μ is the dynamic viscosity of water. Darcy's law (13.2) holds for laminar flow regimes with low Reynolds numbers. Recent studies at the Soultz HDR site in France showed indications of a non-linear flow behavior near injection wells (Kohl et al. 1996). This was not reported on the experiments in Rosemanowes. Therefore, Darcy's law is assumed to be valid for the present study. The conservation of thermal energy in a medium free of heat sources is expressed by

$$c\rho \frac{\partial T}{\partial t} + c^w \rho^w n\mathbf{v} \nabla T - \lambda \nabla^2 T = 0 \quad (13.3)$$

where c is thermal capacity, T temperature, λ thermal conductivity, and the superscript w refers to the mobile fluid phase. In this form all effects from thermal dissipation, radiation, and dispersion are neglected. Heat is transferred through the fractured porous medium by conduction and advection. The fluid and the rock matrix are assumed to be in thermal equilibrium. The parameters in equations (13.1)-(13.3) for the rock matrix (A^m) and the fracture domains (A^f) are:

$$\begin{aligned} \text{porosity: } n &= \begin{cases} n^m & \mathbf{x} \text{ in } A^m \\ 1 & \mathbf{x} \text{ in } A^f \end{cases} \\ \text{hydraulic conductivity: } \mathbf{K} &= \begin{cases} \mathbf{K}^m & \mathbf{x} \text{ in } A^m \\ \frac{g\rho^w}{\mu} \frac{b^2}{12} \delta & \mathbf{x} \text{ in } A^f \end{cases} \\ \text{thermal capacity: } c\rho &= \begin{cases} c^m \rho^m & \mathbf{x} \text{ in } A^m \\ c^w \rho^w & \mathbf{x} \text{ in } A^f \end{cases} \\ \text{thermal conductivity: } \lambda &= \begin{cases} \lambda^m & \mathbf{x} \text{ in } A^m \\ \lambda^w & \mathbf{x} \text{ in } A^f \end{cases} \end{aligned} \quad (13.4)$$

where the parameters for the rock matrix are weighted linearly according to matrix porosity n^m ,

$$\begin{aligned} c^m \rho^m &= n^m c^w \rho^w + (1 - n^m) c^r \rho^r \\ \lambda^m &= n^m \lambda^w + (1 - n^m) \lambda^r \end{aligned} \quad (13.5)$$

The superscripts w and r refer to water and rock; f and m correspond to fractures and rock matrix, respectively.

13.3 Numerical Procedure

The system of partial differential equations resulting from equations (13.1) - (13.5) is solved numerically with finite elements using the ROCKFLOW code (Kolditz et al. 1999). The finite element formulations for a transport problem are described in detail in section 7.7. This program is particularly well suited for the simulation of flow and transport in fractured media. A multi-dimensional element library permits to approximate conveniently typical geological structures such as 1-D boreholes, 2-D fractures and 3-D rock matrix blocks. The accuracy and performance of ROCKFLOW was verified in more than 20 benchmark tests (Kolditz and Zielke, 1996). Fig. 13.3 displays the FE-mesh used in the calculations, which accommodates the fracture network. The mesh consists of roughly 30000 coupled 2-D (linear quadrilateral) and 3-D (linear hexahedral) finite elements. We apply the standard Galerkin formulation to avoid numerical dispersion. Time discretization follows the Crank-Nicolson scheme. Time steps are on the order of days. The algebraic equations are handled by an iterative CGS solver (Lanczos-type Bi-Conjugate Gradient Squared solver). The calculations were performed on different platforms: a PC-486DX2, an IBM-RISC 6000, and a HP730. The required CPU time per time step ranges from about 2 minutes on the HP workstation to 5 minutes on the PC equipped with 32MB RAM.

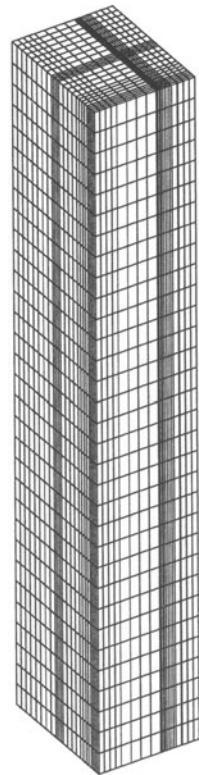


Figure 13.3:

13.4 Example - Rosemanowes Hot Dry Rock Site

13.4.1 Data

The Rosemanowes HDR site is located in Cornwall (SW England) within the Carnmenellis granite. Three wells (RH11, RH12, RH15) were drilled to depths between 2.0 and 2.6 km. Stimulation and circulation experiments had been conducted in these boreholes over a period of more than ten years (Parker, 1989; Richards et al., 1994). Two reservoirs were developed and tested: the RH11/RH12 and the RH12/RH15 system. The long-term circulation test was carried out within the hydraulically stimulated RH12/RH15 reservoir. This provides the data base for our modeling study.

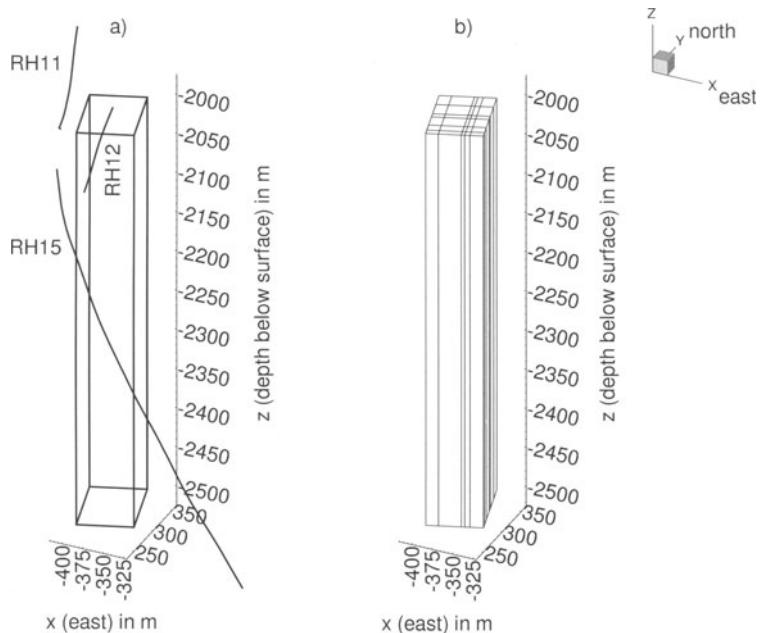


Figure 13.4: Perspective view of the stimulated reservoir and the boreholes RH11, RH12, and RH15 (a), perspective view of the synthesized fracture network (b)

The size and the shape of the accessible reservoir (see Fig. 13.4a) was estimated from monitoring the micro-seismicity during the stimulation of the wells (Green and Parker, 1992). Various methods were employed to detect the location and the orientation of the fracture system, such as geophysical surface and borehole methods. From statistical analysis of the data two sub-vertical sets of fractures were identified, which scatter around the two major strike directions of 165 degree N and 250 degree N. Therefore, a deterministic fracture network is established in this study according to these average orientations. Further, the number of fractures is restricted to those, which are hydraulically active as shown by well logs. Fig. 13.4b presents the resulting fracture network which forms the base of our simulations.

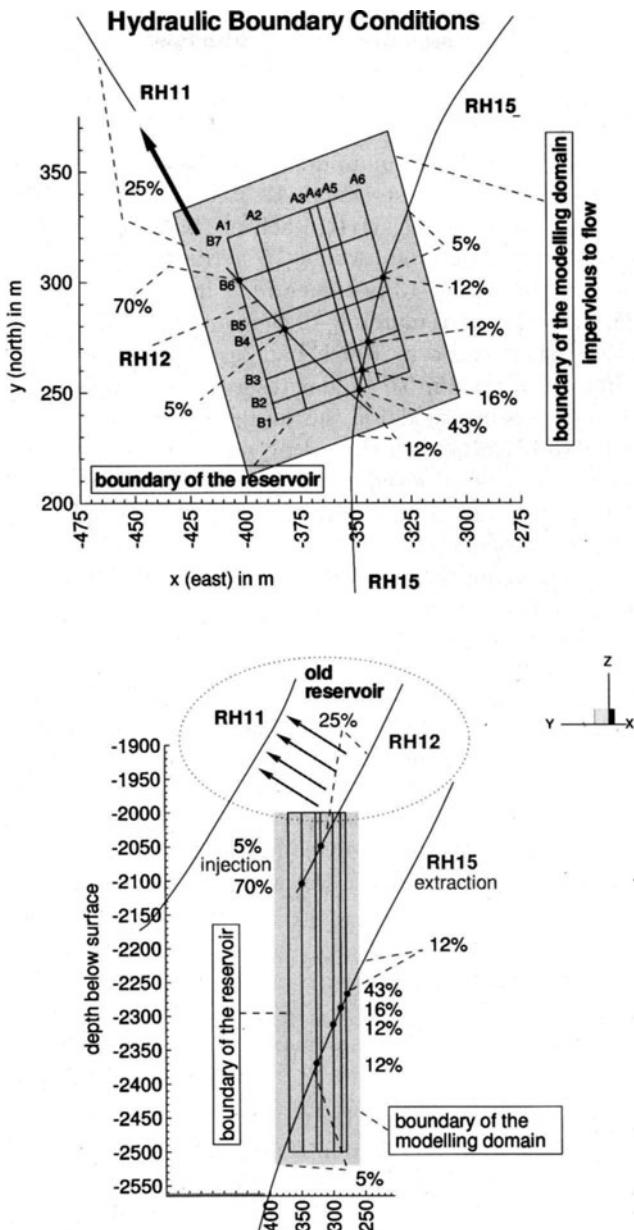


Figure 13.5: Perspective view of the stimulated reservoir and the boreholes RH11, RH12, and RH15 (a), perspective view of the synthesized fracture network (b)

Circulation in the RH12/RH15 system was maintained continuously for about three years, starting in August 1985. Figures 13.5a and 13.5b display horizontal and vertical projections of this network and the boreholes as well as the corresponding flow fractions, which were absorbed by the individual fractures. Most of the fluid leaves the borehole RH12 through the fracture at a depth of 2100 m. Injection logs show the main depth ranges of water loss from borehole RH12 during the circulation test (Figs. 13.5a,b). There are five main zones of fluid inflow into the borehole RH15, each of them producing more than 10 percent of the total fluid volume. About 75 percent of the injected water was recovered from this borehole. 13 percent of the injected water was flowing out of borehole RH11, and the remaining 12 percent (i.e. 3 l/s) was lost into the shallower RH11/RH12 reservoir, which had been tested before 1985. The injection and circulation tests in this older reservoir also have a strong influence on the initial temperature field for the long-term circulation test conducted in the RH12/RH15-reservoir, which will be discussed later. Optimum performance of the system was achieved at an injection pressure of 10 MPa above hydrostatic, producing a hydraulic impedance of 0.6 MPa sl⁻¹ and an average production rate of 14 kgs⁻¹. The hydraulic impedance of a reservoir is defined as the pressure drop along the flow paths divided by the production flow rate. Thus, the overall impedance is one of the key parameters for the calibration of a steady-state hydraulic model.

The thermal history of inlet and outlet temperatures at the wellheads and casing shoes of both boreholes is summarized in Fig. 13.6. The temperatures at the wellheads were recorded continuously. However, the downhole temperatures at the casing shoes are of primary interest for our simulation of the reservoir behavior. Thus, a mean injection temperature is estimated for the reservoir from the average temperature at the wellhead, which is heated up by about 4 degrees Celsius on its way down from the wellhead to the casing shoe.

13.4.2 Simulation Results

Hydraulic model

The continuity equation (13.1) and Darcy's law (13.1) yield the flow equation for the hydraulic model. Water flow into and out of reservoir is simulated by source and sink terms which correspond to the flow fractions measured within the boreholes (see Figs. 13.5a and 13.5b). The alternative to this, the use of pressure or hydraulic head boundary conditions, would require variation and adjustment of individual fracture apertures to simulate the measured flow fractions. We feel that this would introduce too many degrees of freedom to be sufficiently constrained by data. For instance, no vertical pressure profiles are available for the boreholes. In contrast, use of sources and sinks is based on the measured data. Boundary conditions applied in the solution of the flow equation are summarized in Figs. 13.5a and 13.5b. The flow rates specified for

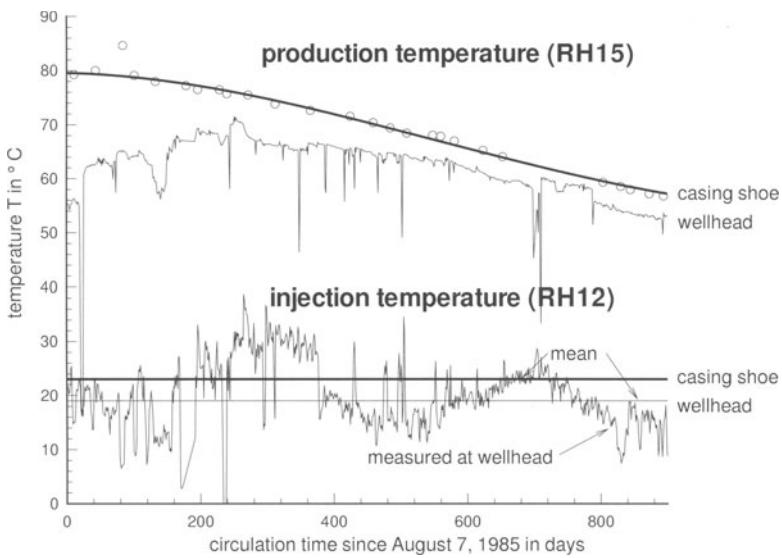


Figure 13.6: Measured temperatures in the boreholes RH12 and RH15 during the long-term circulation test. Circles are measured temperatures at the casing shoe. Full line is cubic spline interpolation

the injection and production boreholes correspond to the measured flow fractions. As mentioned earlier, 25 percent of the injected water was lost to the shallow RH11/RH12-reservoir. This means that only 75 percent of the injected water was circulated within the RH12/RH15-reservoir. Therefore, a constant average flow rate of 14 kgs^{-1} was used in our model. Because of the position of the boreholes (Fig. 4b) the flow between RH12 and RH15 will be mainly vertical. The extent of the artificially created reservoir was estimated from the distribution of microseismic hypocenter locations (Fig. 13.4a). This stimulated region is completely encapsulated by the model domain. To avoid boundary effects, the active hydraulic reservoir was embedded in a larger model domain with impermeable boundaries (Figs. 13.5a,b).

The fracture network model was used to study the dependence of the reservoir impedance i on different fracture permeabilities. It is defined as the pressure drop between the injection and the production borehole divided by the flow rate, i.e. the pressure difference which is necessary to circulate a certain fluid volume:

$$i = \frac{\bar{p}_{\text{inlet}} - \bar{p}_{\text{outlet}}}{Q} \quad (13.6)$$

where $\bar{p} = \bar{h}gp^w$ is the average pressure in the injection or production borehole and $Q = \dot{m}/\rho^w$ is the volumetric flow rate. The use of a uniform reservoir impedance in model calibration is justified by the computed pressure distributions within the wells since the calculated pressure differences within each borehole are small compared to the overall pressure drop between the wells: between 0.3 and 0.5 MPa within each borehole versus 9 MPa between the injection and the production well.

Since normal and shear forces are acting on the fracture surfaces, the hydraulic characteristics of a fracture network are strongly affected by the actual tectonic stress field. Anisotropy in the tectonic stress field, therefore, implies anisotropy as well in the hydraulic behavior of the fracture system. The maximum horizontal principle stress at the Rosemanowes site is in a direction of about 150 degree N (Pine et al., 1990). The first fracture set is nearly aligned with this direction of maximum principle stress. Therefore, we expect a better permeability for this set than for the second one. The second fracture set strikes 250 degree N, which is nearly the direction of minimum principle stress. In order to characterize the hydraulic anisotropy of the fracture network we define a hydraulic anisotropy factor as the ratio of the hydraulic fracture apertures $b1$ (set 1) and $b2$ (set 2).

Fig. 13.7 shows the effect of hydraulic anisotropy on the overall impedance. Three different hydraulic systems are studied: an isotropic one and two anisotropic cases with a hydraulic anisotropy factor of $b1/b2 = 2$ and $b1/b2 = 5$. This range of hydraulic anisotropy factors is motivated from mechanical considerations and from models of the hydraulic stimulation of the reservoir (Jupe, personal communication). For each case of hydraulic anisotropy those fracture apertures are identified, which correspond to the measured impedance of 0.6 MPa s l^{-1} . This range of 2-5 for hydraulic anisotropy is confirmed by previous findings of Hodgkinson (1984) and Jung (1991). A more complete characterization of the hydraulic reservoir can be achieved if transport data (chemical and thermal) are included in the analysis (see below).

The conductivity contrast between fractures and rock matrix needs to be estimated, too. According to a compilation of rock permeabilities by Clauser (1992), an upper limit of $K^m = 10^{-4} \text{ ms}^{-1}$ is used for the hydraulic conductivity of the rock matrix. During massive heat mining from the reservoir significant stresses can develop within the rock matrix, especially near the injection borehole. Thermal cracking caused by these stresses will improve the hydraulic conductivity of the rock matrix, particularly in respect to the long-term behavior. However, it must be realized that $K^m = 10^{-4} \text{ ms}^{-1}$ corresponds to an extremely high permeability for crystalline rocks.

Figures 13.8a and 13.8b illustrate the simulated flow distribution within the fracture network. Maximum Darcy velocities near the injection and extraction wells are on the order of 10^{-2} ms^{-1} . The corresponding positions of the individual fractures within the network are shown in Fig. 13.4a. Fractures A1 and B6 contain the main injection zone, which supplies about 70 percent of the injection flow rate.

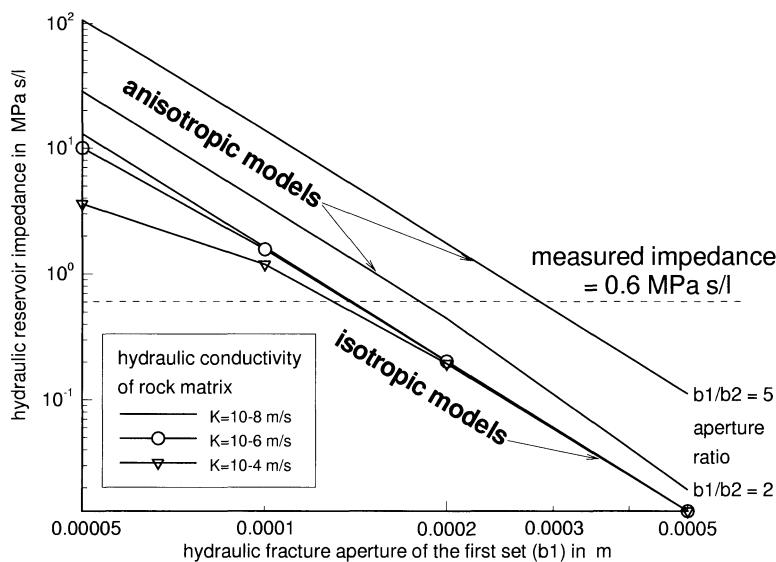


Figure 13.7: Dependence of the hydraulic reservoir performance depending on hydraulic anisotropy of the two fracture sets with apertures b_1 and b_2 . K is the hydraulic conductivity of the rock matrix.

Thermal model

The thermal model is calibrated on the base of the production (bottom-hole) temperatures in the borehole RH15. Variations in inlet temperature on the order of ± 20 K modify the production temperatures less than ± 2 K (Fig. 13.6). Therefore, a constant mean injection temperature of 23 degree C is assumed.

First simulations showed the strong dependence of the results on the initial temperature distribution. A review of experiments conducted at the Rosemanowes site before the start of the long-term circulation test (project phase 2C) shows that the temperature in the RH12/RH15 reservoir was affected by a number of these previous operations. The RH11/RH12 system was created during project phase 2A and circulated, with intermissions, from 1980 - 1983. This reservoir extends down to a depth of 2100 m. The temperature field was further disturbed when the RH12/RH15 reservoir was created, particularly within project phase 2B by drilling the new well RH15 and by massive stimulation of borehole RH12 in the years 1983 - 1985. Fig. 13.9c shows the situation of the last temperature logs in the boreholes RH12 and RH15 before the beginning of the long-term circulation test. Cooling in the upper part of the RH12/RH15 reservoir was caused by earlier operations in the RH11/RH12 system. The lower part of the

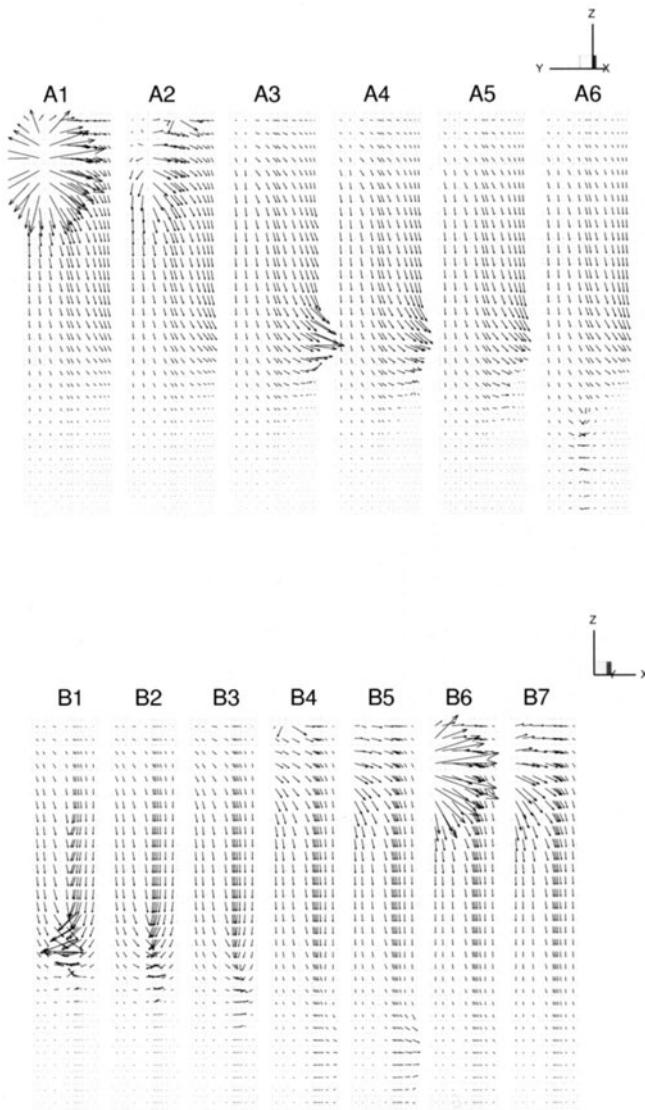


Figure 13.8: Linearly scaled Darcy velocities within the fracture network. A1-A6 (a) and B1-B7 (b) are the discrete fractures shown in Fig. 13.5a.

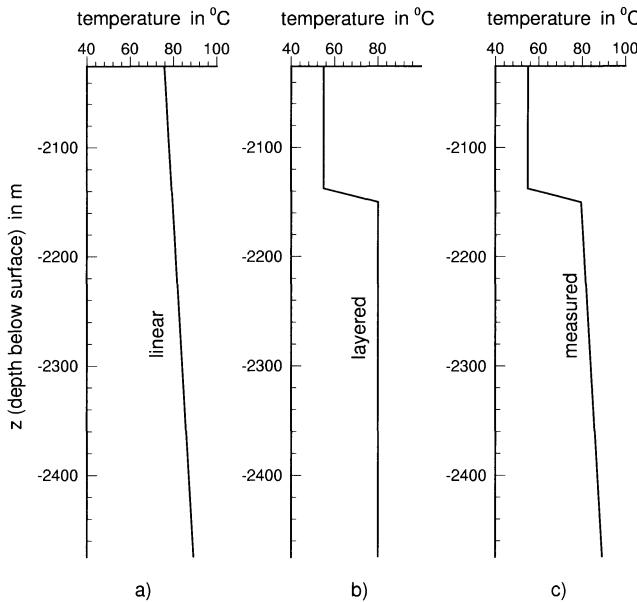


Figure 13.9: Initial temperature distributions used in modeling: linear (a), layered (b), and measured logs (c).

reservoir shows a nearly undisturbed, linear temperature distribution with a temperature gradient of about 30 mK m^{-1} .

Fig. 13.10 demonstrates the influence of different initial temperature conditions on the evolution of the mean production temperature. In this case, the isotropic model is used which assumes equal permeabilities of both fracture sets. Starting with an undisturbed thermal reservoir with constant temperature gradient (Fig. 13.9a) the corresponding model over-estimates the thermal energy stored in the reservoir. On the other hand, a vertically layered thermal regime (Fig. 13.9b) leads to a stronger thermal drawdown than observed. Finally, an initial temperature distribution corresponding to the measured logs (Fig. 13.9c) yields the best result. The strong influence of the initial condition on the thermal drawdown, illustrated in Fig. 13.10 emphasizes the importance of correct initial conditions for models of reservoirs with a complicated thermal history as HDR sites often are.

Fig. 13.11 compares the results of isotropic and anisotropic models. Two cases of hydraulic anisotropy are considered, corresponding to factors of 2 and 5 for hydraulic anisotropy, as suggested by the tectonic stress field at Rosemanowes. It turns out that the isotropic model (B in Fig. 13.11) yields a more rapid thermal drawdown than the anisotropic ones. For simple network geometries

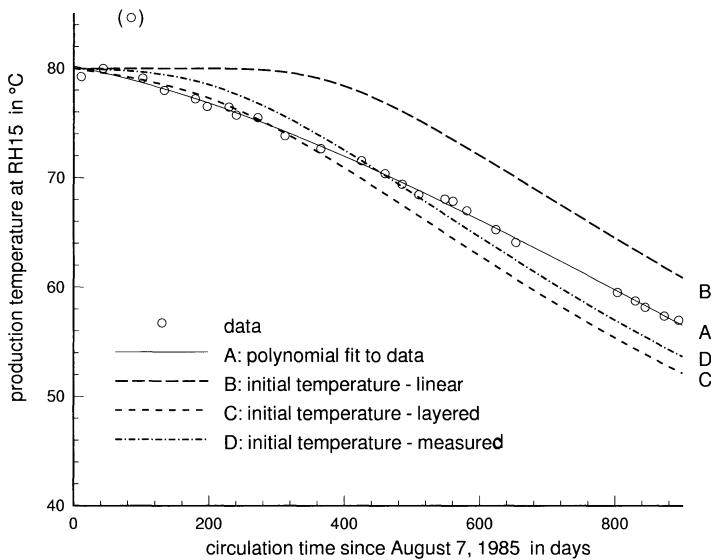


Figure 13.10: Dependence of the computed thermal drawdown in the production borehole RH15 on the initial temperature distributions shown in Fig. 13.9 for isotropic conditions.

and directly connected inlet and outlet points heat is removed more efficiently from isotropic systems because there is a well distributed flow within these networks. Since the amount of heat stored in a matrix block is limited, this leads to a smaller thermal drawdown at early times followed by a more rapid drawdown at later times. In contrast, anisotropic systems behave more like parallel fracture arrays. The advective contribution from secondary fractures is reduced. Therefore, the diffusive part of heat propagation in the reservoir becomes more important. This results in a more uniform decrease in production temperature. Compared to anisotropic fracture networks there is more thermal drawdown at early times and less at later times.

However, the difference in initial thermal drawdown between our isotropic and anisotropic networks (Fig. 13.11) is less than predicted by earlier studies (Kolditz, 1994). This effect can be understood by comparing the flow fields for the isotropic and anisotropic cases. In the specific 3-D configuration of the inlet and outlet points in the Rosemanowes HDR reservoir (Fig. 13.5), hydraulic anisotropy forces flow to penetrate deeper down into the reservoir than in the isotropic case (Fig. 13.12). This is due to the increased hydraulic impedance caused by the less permeable fractures of the second set (B1-B7). Maximum Darcy velocities near the injection and extraction wells are on the order of

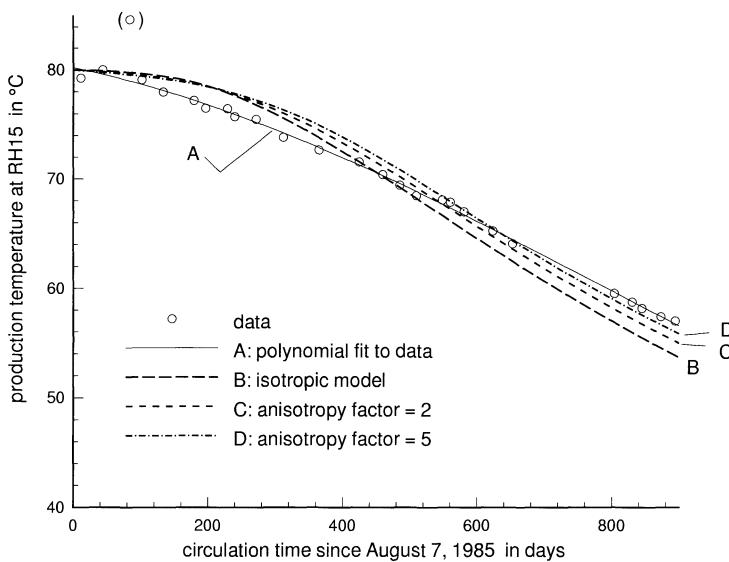


Figure 13.11: Dependence of the computed thermal drawdown in the production borehole RH15 on the hydraulic anisotropy of the two fracture sets.

10^{-2} ms^{-1} . In the isotropic system, in contrast, little flow penetrates to the lower part of the reservoir. Therefore, basically only heat from the upper part of the reservoir is removed, resulting in an accelerated thermal drawdown. This behavior is in contrast to the situation discussed above for a simpler configuration of inlet and outlet points. Thus, in this special case, hydraulic anisotropy leads to a more efficient removal of heat from the reservoir. These results emphasize the need for 3-D models in complex geometric and hydraulic conditions.

Next, the effect of matrix porosity is studied. Total porosity, which includes the volume fraction of stagnant fluid, is relevant for the thermal capacity as well as for the thermal conductivity of the saturated matrix. While thermal capacity increases with porosity (i.e. bulk water content) thermal conductivity of the saturated rock matrix is decreasing. Fig. 13.13 shows the evolution of the production temperature in borehole RH15 for an anisotropy factor of 5. An increase of matrix porosity reduces the thermal drawdown. There is no effect for matrix porosities lower than 1 percent. Porosities on the order of some percent appear quite high for crystalline rocks, even for altered granite. However, porosity for the rock matrix in the porous medium approach additionally includes contributions from the microfissure system and also from those fractures, which do not intersect any boreholes and thus remain undetected.

Finally, the model is fitted to the thermal drawdown recorded at the Rose-

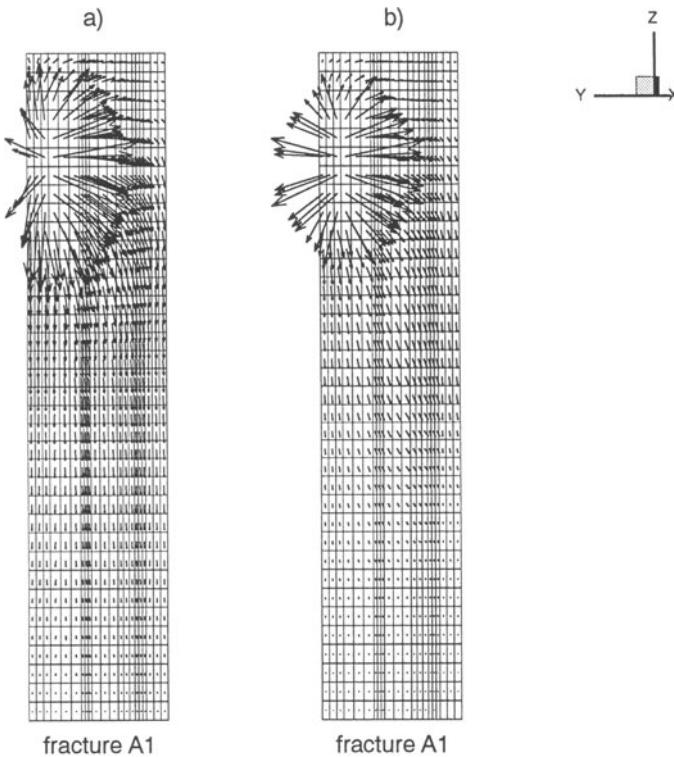


Figure 13.12: Comparison of flow distributions (linearly scaled Darcy velocities) within the fracture A1 (see Fig. 13.5a) for an anisotropy factor of 5 (a) and for isotropic conditions (b).

manowes HDR site by varying only two parameters, matrix porosity and the factor of hydraulic anisotropy (Fig. 13.14). The data point in parentheses at the beginning of the circulation was reported to be disturbed by a wrong measurement (Parker, 1989) and is therefore ignored. An anisotropy factor of $b_1/b_2=5$ and a matrix porosity of 1 percent provide the best fit to the data. Nicol and Robinson (1990) matched the thermal drawdown during the long-term circulation at Rosemanowes using a parallel fracture model. To constrain the choice of parameters they used information from tracer tests. In this way, measured tracer dispersion and multiple peak behavior of the tracer response are clear indications of the presence of multiple flow paths. Their data fit (Fig. 13.14) resulted from a model consisting of two independent flow paths with different heat exchange areas involving different flow fractions. One fracture represents the main flow path and the other one an assumed short circuit. A certain roughness in the drawdown curve calculated by Nicol and Robinson (1990) is caused by the varying injection flow rates. The long-wavelength variations are due to

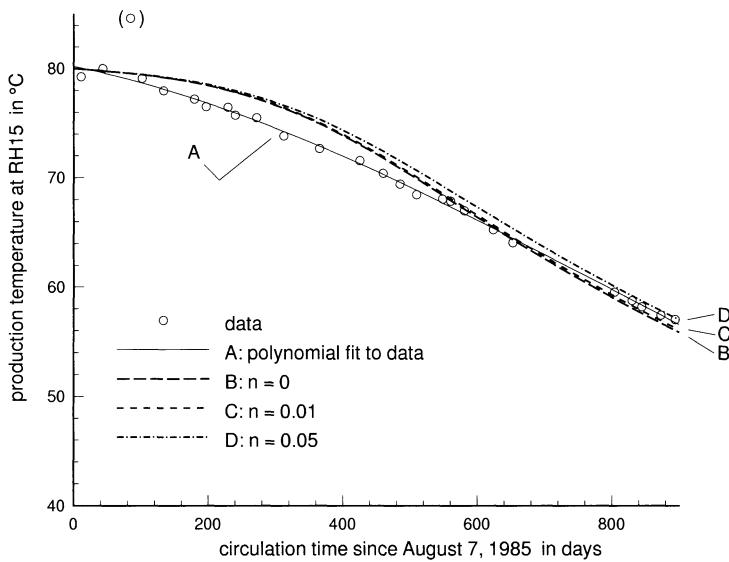


Figure 13.13: Dependence of the computed thermal drawdown in the production borehole RH15 on the total porosity of the rock matrix.

the seasonal changes of the injection temperatures included in this model, in contrast to ours. Bruel (1995) applied a stochastic approach to analyze the hydraulic and thermal data of the long-term circulation test at Rosemanowes. For this purpose, 25 statistically equivalent realizations of possible flow paths were generated and their thermal performance was evaluated. Fig. 13.14 shows the range of thermal performance obtained from these stochastic fracture network simulations. Bruel (1995) found that the thermal performance is determined by only a few, i.e. 2-3, preferential flow paths. The length and corresponding heat exchange area of these flow paths may be quite different for each fracture network realization. Therefore, the stochastic approach yields a large range of possible drawdown curves (Fig. 13.14). A mean drawdown curve with corresponding standard deviation is not available from Bruel (1995) for comparison with other results.

13.4.3 Discussion

In general, the fit of the measured thermal drawdown is reproduced well by our model. The maximum difference between measured and simulated production temperatures is about 2 K. This fit is slightly better than by other models (Fig. 13.14). However it is obvious, that the polynomial fit of the data and the

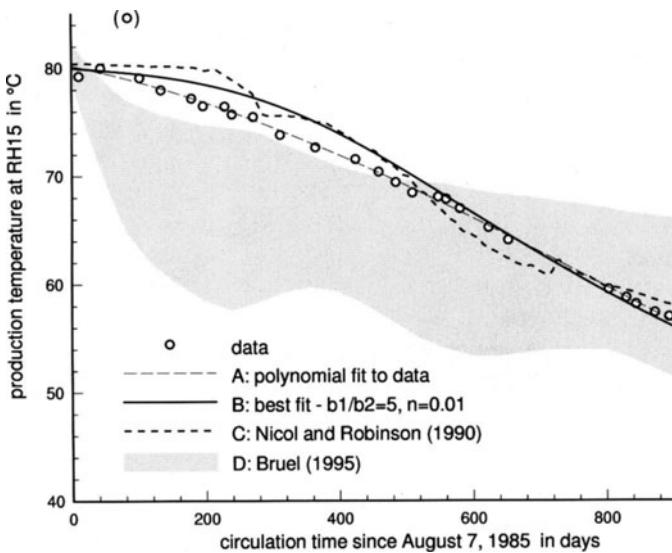


Figure 13.14: Comparison of the thermal drawdown computed by use of different conceptual models: a parallel fracture model by Nicol and Robinson (1990), a stochastic fracture network model by Bruel (1995), and the results obtained with the deterministic fracture network model in this study.

computed drawdown curves have a different behavior. The measured thermal drawdown is underestimated by the model during the early stages of the circulation test, and overestimated during the final stage. These different drawdown characteristics are similar to those of isotropic and anisotropic fracture networks (see above). However, if the anisotropy factor is increased further, the shape of thermal drawdown is not altered any more. Moreover, the apertures of the fracture sets become unrealistic for large factors of hydraulic anisotropy (Fig. 13.7). Thus, it may be speculated that the observed thermal drawdown might be due to a short circuit. This could be caused by channeling both within the fracture planes or the total network. Fracture channeling is due to fracture roughness, and network channeling can be caused by a variable finite extent of fractures. Change of flow path geometry during the circulation experiment may provide an alternative explanation of the measured thermal drawdown. Thus, to improve the model further, a number of new parameters must be introduced, such as fracture sizes and distributions of fracture apertures, which are difficult to determine. Other studies of the long-term thermal behavior of a HDR reservoir, e.g. the investigation of the dimensional effect of matrix heat diffusion (Kolditz, 1995a), confirm that thermal tracers require long calibration times.

Even a period of 3 years appears too short to discriminate clearly between different parameter choices on the basis of thermal data alone. Thus, apart from hydraulic and thermal data, chemical tracers, providing transport information on shorter time scales, are required for a more refined model. The essential difference between the three modeling approaches discussed (i.e. parallel fracture models, deterministic fracture network models, and stochastic fracture network models) is the kind and amount of structural information required by each of these models (Fig. 13.1). Nicol and Robinson (1990) call their approach an "inexact structure model" based on simplified flow paths with corresponding heat exchange areas. Stochastic fracture network models seem to have great potential for generating a wide variety of models for different subsurface structures. However, they are often difficult to be constrained by the available experimental data. Our fracture network model is placed in between these two model end-members: it incorporates substantial information on the structure, geometry, and hydrothermal characteristics of the real reservoir, such as the shape of the reservoir, borehole locations, geometry of the fracture network, hydrothermally active fractures, anisotropy, and initial temperature distribution (Tab. 13.1). This can only be accommodated in a fully 3-D model. Structurally, this approach is not as over-simplified as the parallel fracture model on the one hand, but has much less degrees of freedom than stochastic models on the other hand. Thus, it appears to be well adapted to the type, number, and quality of data typically available for HDR reservoirs in crystalline bedrock.

Conclusions

This study shows that a deterministic model based on structural information can be used successfully to simulate 3-D heat transfer processes in a HDR reservoir if two factors are combined: (1) a data base sufficient to construct a 3-D structural, hydraulic, and thermal model, and (2) a conceptual approach fit to handle the amount and kind of data. This strategy yields a satisfactory fit of simulation results with experimental data and involves only a minimum number of tuning parameters. Based on the data set from Rosemanowes Hot Dry Rock reservoir, it was possible for the first time to simulate the transient heat transfer processes between fractures and matrix fully in three dimensions. We feel this 3-D deterministic fracture network modeling strategy is a reasonable alternative to the overly simplified single or multiple parallel fracture models or porous media equivalents on the one hand, and to the structurally complex stochastic fracture network models on the other hand. In general, the deterministic approach benefits from the fact that the prevailing stress field prevents stimulated fractures from being oriented in a totally random way. When applied to a specific field case, the better the structure of this reservoir is known the more successfully this approach can be applied. The model basically contains only two free parameters: the hydraulic anisotropy factor of the fracture network, and the porosity of the surrounding rock matrix. Comparison of data and modeling results indicate hydraulic fracture apertures with an anisotropy

factor of 5 for the two sets of fractures. This anisotropy and a matrix porosity of about 1 percent provide the best history match of the observed thermal drawdown. The model misfit in this case is less than 2 K, which corresponds to about 5 percent of the total thermal drawdown after three years of circulation. In principle, even the identification of reservoir parameters is possible by parameter variation. Thus, the deterministic fracture network approach is well suited both for a long-term prediction of the reservoir performance and for an efficient estimation of reservoir parameters.

Table 13.1: Experimental data used to condition the deterministic fracture network model

Reservoir structure	<p>Reservoir size: size and shape of the stimulated reservoir estimated from microseismic hypocenter locations (Fig. 13.4a)</p> <p>Fracture network geometry: fractures oriented in two orthogonal sets, identified from statistical analysis (Fig. 13.1) model contains only hydraulically active fractures as detected in flow logs (Fig. 13.4b, Figs. 13.5a,b)</p> <p>Borehole geometry: approximated in vertical intervals (Fig. 13.4a, Figs. 13.5a,b)</p>
Hydraulic data	<p>hydraulically active fractures as modeled explicitly (Fig. 13.4b, Figs. 13.5a,b)</p> <p>measured depth-dependent flow fractions (Figs. 13.5a,b)</p> <p>measured hydraulic reservoir impedance (Fig. 13.7)</p>
Temperature data	<p>mean injection temperature (Fig. 13.6)</p> <p>initial temperature distribution (Fig. 13.9c)</p> <p>production temperature (Fig. 13.6)</p>
Material properties	<p>thermal capacity of water and rock matrix</p> <p>thermal conductivity of water and rock matrix</p> <p>density of water and rock matrix</p> <p>viscosity of water</p>

13.5 Case Study: Soultz-sous-Forêts

In this section we return to the case study of the European geothermal research site at Soultz-sous-Forêts in France (see Chapter 12). More details of the investigation of this research site can be found in Pribnow & Clauser (1999) and for modeling Kolditz (2001), Rother (2001), Kaiser (2001). Here we consider the modeling of heat transport in fractured porous media.

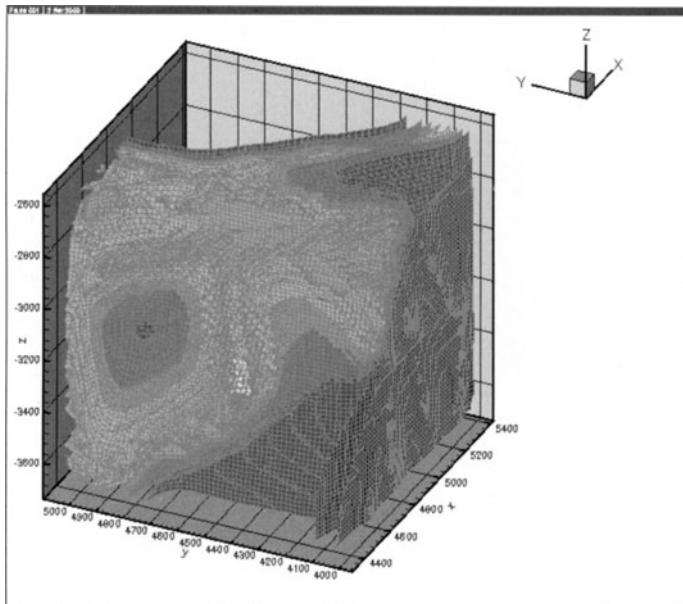


Figure 13.15: Cut through the reservoir model (Octree level 8): geometry and pressure distribution

This section deals with a subdomain of the Soultz model as described by Rother et al. (2000). A cube with approximate edge length 1km was extracted from the overall geometric model (Fig. 13.15). Based on this a coupled 2D/3D model is automatically generated using the Octree method (Rother 2001). Two discretizations based on different octree levels were generated for the numerical analysis.

Octree level	Nodes	3-D Elements	2-D Elements
7	81356	75852 hexahedra	6613 quadrilaterals
8	266240	232048 hexahedra	34192 quadrilaterals

Figure 13.16 shows the simulated pressure distribution within the reservoir from different view points. At the beginning the pressure field is assumed to be

hydrostatic. The pumping well is represented by a flux boundary condition. Steady state flow after switching the pump on is achieved after about 40000 seconds. The outside boundary conditions of the reservoir are hydrostatic. The distorted pressure isolines indicate the influence of the high permeable fractures. Fig. 13.17 shows the corresponding temperature field.

The results presented in Fig. 13.15 correspond to the finite element discretization generated by an octree level equal to 8. The graphic shows a section through the modeling domain. Additionally, the pressure distribution within the reservoir is plotted. This illustration provides an insight into the complexity of the system, i.e. the structural complexity of fractured rock and the processes which are strongly affected by medium heterogeneity.

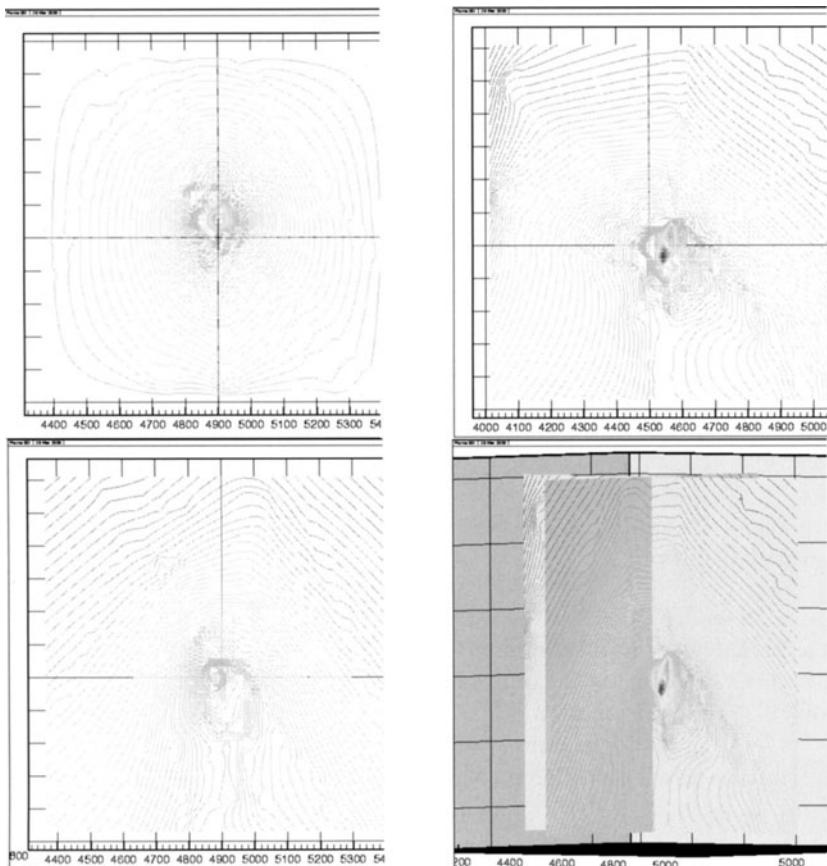


Figure 13.16: Simulated pressure field within the Soultz reservoir (3D view, cut-off in xy, xz and yz direction)

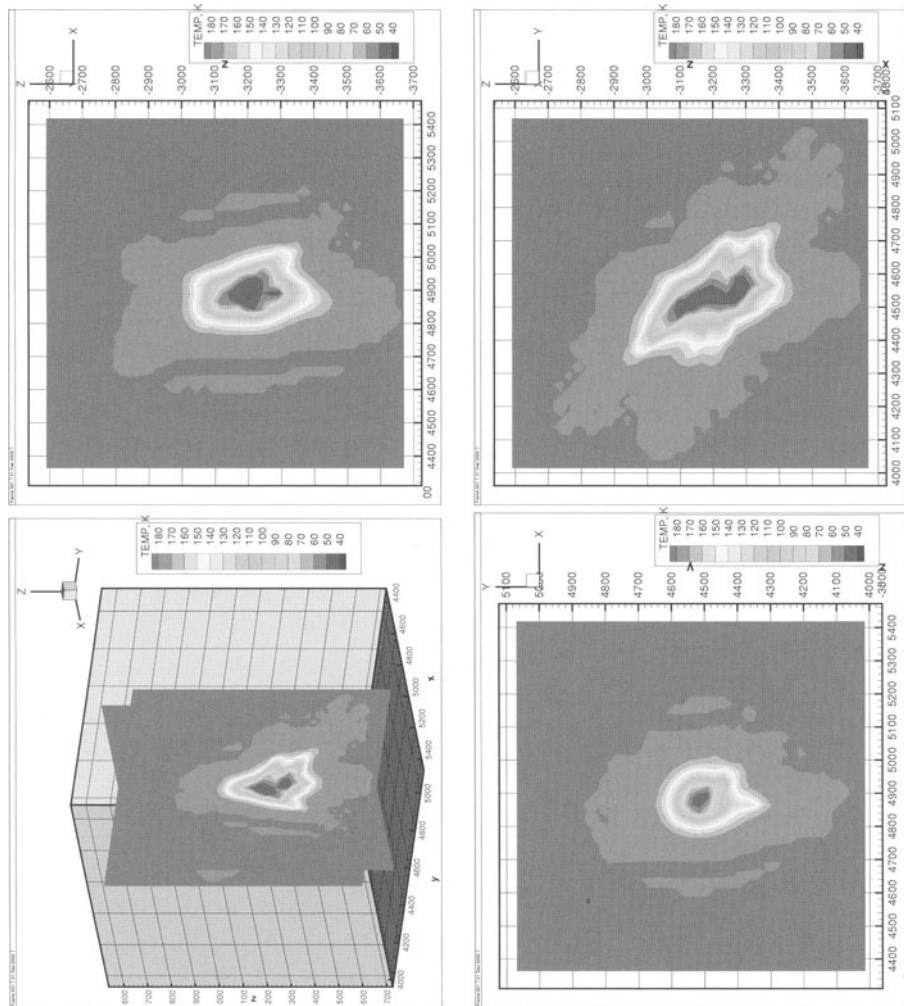


Figure 13.17: Simulated temperature field within the Soultz reservoir (3D view, cut-off in xy, xz and yz direction)

Bibliography

- [1] Andersson, J., Shapiro, A. M. and Bear, J. (1984) A stochastic model of fractured rock conditioned by measured information. *Water Resour. Res.*, 20(1), 79-88.
- [2] Birdsell, S. A. and Robinson, B. A. (1988) A three-dimensional model for fluid, heat, and tracer transport in the Fenton Hill hot dry rock reservoir. In *Proceedings of at the 13th Annual Workshop on Geothermal Reservoir Engineering*, Stanford, CA.
- [3] Bruel, D. and Cacas, M. C. (1992) Numerical modelling technique: contribution to the Soultz HDR project. In *Geothermal Energy in Europe - the Soultz Hot Dry Rock Project*, ed. Bresee, J. C., Gordon and Breach Science Publishers, 267-279.
- [4] Bruel, D. (1995) Heat extraction modelling from forced fluid flow through stimulated fractured rock masses: application to the Rosemanowes Hot Dry Rock reservoir. *Geothermics*, 24(3), 361-374.
- [5] Clauser, C. (1992) Permeability of crystalline rocks. *EOS Trans. Am. Geophys. Union*, 73(21), 233+237.
- [6] Dershowitz, W. S. (1985) Rock joint systems. Ph.D. thesis, Dep. of Civ. Eng., Massachusetts Institute of Technology, Cambridge.
- [7] Diersch, H.-J., Kolditz, O. and Jesse, J. (1989) Finite element analysis of geothermal circulation processes in hot dry rock fractures. *Z. Angew. Math. Mech. (ZAMM)*, 69, 139-153.
- [8] DuTeau, R. and Swenson, D. V. (1994) Modeling flow in a jointed geothermal reservoir. *Trans. Geothermal Resources Council*, 18, 465-472.
- [9] Green, A. S. P. and Parker, R. H. (1992) A multi-cell design of a HDR-reservoir. *Trans. Geothermal Resources Council*, 16, 457-464.
- [10] Gringarten, A. C. and Sauty, J. P. (1975) A Theoretical Study of Heat Extraction from Aquifers with Uniform Regional Flow. *J. Geophys. Res.*, 80(35), 4956-4962.
- [11] Häfner, F., Sames, D. and Voigt, H.-D. (1992) *Wärme- und Stofftransport - Mathematische Methoden*, Springer Verlag.
- [12] Hodgkinson, D. P. (1984) Analyses of steady state hydraulic tests in fractured rock. *Atomic Energy Research Harwell, Report R 11287*.
- [13] Heuer, N., Küpper T. and Windelberg, D. (1991) Mathematical model of a Hot Dry Rock system. *Geophys. J. Int.*, 105, 659-664.
- [14] Jung, R. (1991) Hydraulic fracturing and hydraulic testing in the granitic section of borehole GPK1 - Soultz- sous-Forêts. *Geotherm. Sci. & Tech.*, 3(1-4), 149-198.
- [15] Kaiser, R (2001) Gitteradaption für die Finite-Elemente-Modellierung gekoppelter Prozesse in geklüftet-porösen Medien. PhD thesis, in preparation, Institute of Fluid Mechanics, Groundwater Modeling Group, University of Hannover.
- [16] Kohl, T. (1992) Modellsimulationen gekoppelter Vorgänge beim Wärmeentzug aus heißem Tiefengestein. Doctoral Dissertation, No. 9802, ETH Zürich, (in German).
- [17] Kohl, T., Jung, R., Hopkirk, R. J. and Rybach, L. (1996) Non-linear flow transients in fractured rock masses - The 1995 injection experiment in Soultz. In *Proc. 21th Workshop on Geothermal Reservoir Engineering*, Stanford, Jan 22-24.

- [18] Kolditz, O. and Diersch, H.-J. (1993) Quasi steady-state strategy for numerical simulation of geothermal circulation processes in hot dry rock fracture. Int. J. Non-Linear Mechanics, 28 (4), 467-481.
- [19] Kolditz, O. (1994) Modellierung von Strömungs- und Transportprozessen im geklüfteten Festgestein: Anwendung auf die Wärmeextraktion aus heißen, trockenen Tiefengesteinen (hot dry rock). Report, No. 112281, Niedersächsisches Landesamt für Bodenforschung (NLfB), Hannover, (in German).
- [20] Kolditz, O. (1995a) Modelling flow and heat transfer in fractured rocks: dimensional effect of matrix-heat-diffusion. Geothermics, 24(3), 421-437. Kolditz, O. (1995b) Modelling flow and heat transfer in fractured rocks: conceptual model of a 3-D deterministic fracture network. Geothermics, 24(3), 451-470.
- [21] Kolditz, O. and Zielke, W. (1996) ROCKFLOW - benchmarks and examples in subsurface hydrology. Technical Report, Institut für Strömungsmechanik, Universität Hannover.
- [22] Kolditz, O. & Clauser C. (1998) Numerical simulation of flow and heat transfer in fractured crystalline rocks: Application to the hot dry rock site at Rosemanowes (UK). Geothermics, 27(1): 1-23.
- [23] Kolditz, O., Habbar, A., Kaiser, R., Rother, T. & Thorenz, C. (1999) *ROCKFLOW - Theory and Users Manual*, Release 3.4, Groundwater Modeling Group, Institute of Fluid Mechanics, University of Hannover, (www.rockflow.de).
- [24] Kolditz, O. (2001) Non-linear flow in fractured rock. accepted by Int. J. Numerical Methods in Heat & Fluid Flow.
- [25] Kosakowski, G. (1996) Modellierung von Strömungs- und Transportprozessen in geklüfteten Medien: Vom natürlichen Kluftsystem zum numerischen Gitterwerk. VDI-Fortschrittberichte, 7(304), Doctoral Dissertation, Mathematisch-Naturwissenschaftliche Fakultät, Rheinische Friedrich-Wilhelms-Universität, Bonn.
- [26] Kröhn, K.-P. and Zielke, W. (1991) FE-Simulation von Transportvorgängen im klüftigen Gestein. Deutsche Gewässerkundliche Mitteilungen, 35(3/4), 82-88.
- [27] Lanyon, G. W. and Batchelor, A. S. (1992) Modelling for the UK hot dry rock programme using the fracture network method: summary report. Geoscience Report CC59/R5/078.
- [28] Lauwerier, H. A. (1955) The Transport of Heat in an Oil Layer Caused by the Injection of Hot Fluid. Appl. Sci. Res. A5, 145-150.
- [29] Lege, T., Kolditz, O. and Zielke, W. (1996) Strömungs- und Transportmodellierung. Methoden- handbuch zur Erkundung des Untergrunds von Deponien und Altlasten, Band 2, Springer-Verlag.
- [30] Long, J. C. S. (1983) Investigation of equivalent porous medium permeability in networks of discontinuous fractures. Ph.D. thesis, Univ. of California, Berkeley.
- [31] Murphy, H. D. (1977) Fluid Injection Profiles - A Modern Analysis of Wellbore Temperature Surveys. Annual Fall Technical Conference of the Soc. of Petroleum Engineers of AIME, Denver, Col. (USA), 9-12 Oct, Soc. Pet. Eng. AIME Pap., 52. (Vol. SPE 6783).
- [32] Neretnieks, I. (1980) Diffusion in the rock matrix: An important factor in radionuclide retardation. J. Geophys. Res., 85(B8), 4379-4397.

- [33] Nicol, D. and Robinson, B. A. (1990) Modelling the heat extraction from the Rosemanowes HDR reservoir. *Geothermics*, 19 (3), 247-257. Nicol, D. A., Randall, M. M. and Hicks, T. W. (1990) Stimulation modelling. In Proc. of the Camborne School of Mines Int. Conference on Hot Dry Rock Geothermal Energy, ed. Baria, R., 289-298.
- [34] Noorishad, J., Ayatollahi, M. S., Witherspoon, P. A. (1982) A finite-element method for coupled stress and fluid flow analysis in fractured rock masses. *Int. J. Rock Mech. Min. Sci.*, 19, 185-193.
- [35] Parker, R. H. (ed) (1989) Hot dry rock geothermal energy - Phase 2B. Final Report of the Camborne School of Mines Project, Pergamon Press, ISBN 0-08-037929-X.
- [36] Pine, R. J. and Cundall, P. A. (1985) Application of the fluid-rock interaction programme (FRIP) to the modelling of hot dry rock geothermal systems. In *Int. Symp. on the Fundamentals of Rock Joints*, 293-302.
- [37] Pine, R. J., Jupe, A. and Tunbridge, L. W. (1990) An evaluation of the in situ measurements affecting different volumes of rock in the Carnmenellis granite. In *Scale effects in rock mechanics*, Pinto da Cunha (ed), Balkema, Rotterdam.
- [38] Pribnow, D. and Clauser, C. (1999), *Heat and fluid flow in the Rhein graben: Regional and local models for a HDR system*, Technical Report, Institut für Geowissenschaftliche Gemeinschaftsaufgaben (GGA),
- [39] Richards, H. G., Parker, R. H., Green, A. S. P., Jones, R. H., Nicholls, J. D., Nicol, D. A., Randall, M. M., Richards, S., Stewart, R. C. and Willis-Richards, J. (1994) The performance and characteristics of the experimental hot dry rock geothermal reservoir at Rosemanowes, Cornwall (1985-1988). *Geothermics*, 23(2), 73-109.
- [40] Robinson, B. A. and Brown, D. W. (1990) Modeling the hydraulic characteristics of the Fenton Hill, New Mexico hot dry rock reservoir. In *Trans. Geothermal Resources Council*, 14, 1333- 1337.
- [41] Robinson, B. A. and Kruger, P. (1992) Pre-test estimates of temperature decline for the LANL Fenton Hill long-term flow test. In *Trans. Geothermal Resources Council*, 16, 473-479.
- [42] Rodemann, H. (1979) Modellrechnungen zum Wärmeaustausch in einem Frac. Report, No. 81990, Niedersächsisches Landesamt für Bodenforschung (NLfB), Hannover.
- [43] Rother, T. (2001) Geometric modeling of Geo-Systems. PhD thesis, in preparation, Institute of Fluid Mechanics, Groundwater Modeling Group, University of Hannover.
- [44] Rother T, Kolditz O, Zielke W & Taniguchi T (2000): *Geometric analysis of fractured-porous aquifers*. Rockflow-Preprint [2000-3], Institute of Fluid Mechanics, University of Hannover, accepted by J Environmental Geology.
- [45] Rowe, R. K., Hammond, A. and Booker, J. R. (1989) The effect of multi-directional matrix diffusion on contaminant transport through fracture systems. In *Contaminant Transport in Groundwater*, Kobus and Kinzelbach (eds), Balkema, Rotterdam, ISBN 9061918790.
- [46] Rummel, F. and Baumgärtner, J. (1991) Hydraulic fracturing stress measurements in the GPK1 borehole, Soultz-sous-Forêts. *Geotherm. Sci. & Tech.*, 3(1-4), 119-148.

- [47] Schellschmidt, R. and Schulz, R. (1991) Hydrogeothermic studies in the hot dry rock project at Soultz-sous-Forêts. *Geotherm. Sci. & Tech.*, 3(1-4), 217-238.
- [48] Schulz, R. (1987) Analytical model calculations for heat exchange in a confined aquifer. *J. Geophysics*, 61, 12-20. Sekine, H. and Mura, T. (1980) Characterization of a penny-shaped reservoir in a hot dry rock. *J. Geophys. Res.*, 85(B7), 3811-3816.
- [49] Watanabe, K. and Takahashi, H. (1993) A model analysis of the long-term performance of hot dry rock geothermal energy extractions systems. In Proc. of the JSME-ASME International Conference on Power Engineering, Sep. 12-16, Tokyo, 453-458.
- [50] Wollrath, J. and Zielke, W. (1990) FE-Simulation von Strömungen im klüftigen Gestein. *Deutsche Gewässerkundliche Mitteilungen*, 34(1/2), 2-7.
- [51] Vaughan, P. J. (1987) Analysis of permeability reduction during flow of heated, aqueous fluid through westerly granite. In *Coupled Processes Associated with Nuclear Waste Repositories*, Tsang, C. F. (ed), San Diego, CA.
- [52] Verma, A. and Pruess, K. (1988) Thermohydrological conditions and silica redistribution near high-level nuclear waste emplaced in saturated geological formations. *J. Geophys. Res.*, 93(B2), 1159-1173.

Chapter 14

Density Dependent Flow in Porous Media

In this chapter we examine variable-density flow and corresponding solute transport in groundwater systems. Fluid dynamics of salty solutions with significant density variations are of increasing interest in many problems of subsurface hydrology. The mathematical model comprises a set of non-linear coupled partial differential equations to be solved for pressure/hydraulic head and mass fraction/concentration of the solute component. The governing equations and underlying assumptions are developed and discussed. The equation of solute mass conservation is formulated in terms of mass fraction and mass concentration. Different levels of the approximation of density variations in the mass balance equations are used for convection problems (e.g. the Boussinesq approximation and its extension, full density approximation). The impact of these simplifications is studied by use of numerical modeling. Numerical models for non-linear problems, such as density-driven convection, must be carefully verified in a particular series of tests. Standard benchmarks for proving variable-density flow models are the Henry, the Elder, and the salt dome problems. We studied these benchmarks using two finite element simulators - ROCKFLOW, which was developed at the Institute of Fluid Mechanics and Computer Applications in Civil Engineering, and FEFLOW, which was developed at the Institute for Water Resources Planning and Systems Research Ltd. Although both simulators are based on the Galerkin finite element method, they differ in many approximation details such as temporal discretization (Crank-Nicolson versus predictor-corrector schemes), spatial discretization (triangular and quadrilateral elements), finite element basis functions (linear, bilinear, biquadratic), iteration schemes (Newton, Picard), and solvers (direct, iterative). The numerical analysis illustrates discretization effects and defects arising from the different levels of the density approximation. We present results for the salt dome problem, for which inconsistent findings exist in literature (Kolditz et al. 1998).

14.1 Introduction

14.1.1 Background

A number of environmentally important problems require the analysis of brine or seawater dynamics in subsurface systems, e.g. disposal of hazardous (toxic and radioactive) waste in crystalline or salt rock formations, infiltration of leachates from landfills and industrial waste disposals, and saltwater intrusion in exploited coastal aquifers or aquifers overlaying salt formations. In the past, numerous codes have been developed to simulate groundwater systems including density effects (Tab. 14.1).

Table 14.1: Simulators for variable-density groundwater flow and transport problems

Simulator	References for code development and verification
FAST	Holzbecher (1991)
FEFLOW	Diersch (1988, 1994)
HEATFLOW	Frind (1982)
METROPOL	Leijnse and Hassanzadeh (1989), Leijnse (1992)
NAMMU	Herbert et al.(1988)
ROCKFLOW	Kröhn (1991), Kolditz et al. (1995), Ratke (1995)
SHEMAT	Clauser and Kiesner (1987)
SUTRA	Voss and Souza (1987)
SWIFT	Reeves et al. (1986)
TOUGH2	Oldenburg and Pruess (1995)
UG	Bastian (1992)
VapourT	Mendoza (1990)

Groundwater density may vary slightly as a result of either pressure or small temperature variations. However, the density can alter considerably in geothermal areas or concentrated brine pools. In most of such situations groundwater dynamics are strongly influenced by destabilizing density differences producing convective currents. Density gradients can introduce gravitational instabilities that give rise to recirculating groundwater systems. Concentrated salt solutions occur e.g. near salt domes (Schelkes and Vogel 1992), in deep crystalline rocks, and in bedded-salt formations. Commonly, saturated brine densities are higher than 1200 kg/m^3 . Usually, the coupling between flow and transport in groundwater systems is reduced to advective and velocity-dependent hydrodynamic dispersion mechanisms. This approximation holds for low-concentration situations if all material properties such as density and viscosity are nearly constant and the overall fluid motion is independent of the motion of individual species. Groundwater-brine systems are more strongly coupled by the buoyancy term of the momentum equation (i.e. generalized Darcy's law). Furthermore, osmotic

effects may become important under certain circumstances (Hassanizadeh and Leijnse (1988)). For high-concentration situations, Hassanizadeh (1986) developed a more general form of Darcy's and Fick's laws, which contribute to these cross-coupling effects. Recently, Vogel (1995) reported on artifacts arising from inconsistent formulations of the balance equations with respect to velocities of fluid mass or fluid volume. The reasons of hydrodynamic dispersion in porous media are twofold. First, kinematic effects such as streamline dividing result from the structure of a porous medium, whereas dynamic effects result from fluid velocity fluctuations. Interaction between dispersion and convection phenomena is quite complex. On the one hand, buoyancy effects produce additional fluid motion and, therefore, support macroscopic hydrodynamic dispersion, which depends on the microscopic velocity field. On the other hand, dispersion causes additional mixing and, therefore, reduces the potential for convective currents, i.e. density differences become smoothed. Recently, Hassanizadeh and Leijnse (1995) presented a non-linear theory of high-concentration-gradient dispersion in porous media, which is supported by their theoretical and experimental studies of hydrodynamically stable flow regimes in vertical columns.

14.1.2 Prior work concerning variable-density flow and transport

In the laboratory, density effects have been frequently studied in columns and cavities. Wooding (1962) performed experiments in saturated soil columns with aqueous solutions denser than water. Elders (1966) experimental studies concerning thermal free convection in a Hele-Shaw cell became famous and are frequently employed for the verification of numerical codes. The Hele-Shaw analogy is based on the similarity of flow between two plane walls to this in a porous medium having an equivalent permeability equal to (gap of the cell)²/12. The Hele-Shaw analogy is constrained to situations where advection and rapid diffusion of vorticity are negligible. Schincariol and Schwartz (1990), Oostrom et al. (1992) conducted experiments in flow containers to study mixed convective motions. They observed complex concentration distributions that develop as lobes of dense fluid move downward from the plume, while less dense fluid migrates upward into the plume. Field evidence of convective phenomena is difficult to prove. Only a few relevant have been undertaken. They concern investigation of plume behavior under landfills (e.g. van der Molen and van Ommen 1988).

Modeling of variable-density flow has been addressed by numerous authors. In the past, a number of codes were developed to simulate density-dependent groundwater systems. Tab. 14.1 summarizes some of these simulators. Internal natural convections in closed cells were treated already by Horton and Rogers (1945) and Lapwood (1948) who examined the linear stability providing a perturbation analysis. They found that instabilities appear if a certain Rayleigh number of $4\pi^2$ is exceeded. The Rayleigh number can be physically interpreted

as the ratio of the rate of thermal energy release to the rate of viscous dissipation of energy (Nield and Bejan (1992)). First numerical models were introduced e.g. by Desai and Contractor (1977), Elder (1967), Frind (1982), Pinder and Cooper (1970), and Segol et al. (1975). Diersch (1981) presented a finite element model based on primitive variable formulations. He used the Elder problem for verifying the numerical model. Voss and Souza (1987) and Herbert et al. (1988) discussed the problem of consistent approximation of terms involved in fluid velocity calculations. Herbert et al. (1988) preferred mixed finite element interpolations and computed continuous velocity distributions by a global smoothing technique for velocity components. They discussed also the question, how the viscous force between the matrix skeleton and a multi-component fluid has to be modified, if there are significantly different densities of the fluid mixture components. Moreover, Herbert et al. (1988) introduced a more general drag term for flow through a porous medium containing a dispersive contribution to the viscous force due to mass fraction gradients. They showed that this additional term can be neglected, if the salt concentration does not change appreciably over the dispersion length. Hassanizadeh and Leijnse (1988) considered extensions of Darcy's and Fick's laws in order to account for coupling effects. By means of numerical experiments, the potential significance e.g. of osmotic processes is illustrated. Furthermore, they discussed the issue of compatibility between balance equations and corresponding boundary conditions for variable-density problems. Senger and Fogg (1990), Evans and Raffensperger (1992) formulated variable-density flow in terms of stream functions and equivalent freshwater heads. They found that stream functions should be defined in terms of mass flux. Otherwise, the Boussinesq approximation is involved implicitly. Schincariol et al. (1994) studied mixed convective flows observed during their flow container experiments (Schincariol and Schwartz (1990)). They concluded that the classic Rayleigh number criterion alone is inappropriate to evaluate the stability of mixed convective flow systems. It was found that initial perturbations can grow or decay in dependence on the wavelength of the perturbing function (e.g. harmonic oscillations of the brine boundary condition). Recently, Oldenburg and Pruess (1995) studied the Henry problem as well as the Elder problem to verify the finite difference simulator TOUGH2. Additionally, they investigated the salt dome problem in detail and determined new results which will be discussed and compared in section 5.3.

In this chapter we present two numerical approaches for the simulation of variable-density flow and transport in groundwater systems: ROCKFLOW-DM2, developed at the Institute of Fluid Mechanics and Computer Applications in Civil Engineering, and FEFLOW (Release 4.5), developed at the Institute for Water Resources Planning and Systems Research Ltd. Three benchmark tests are studied: the Henry problem, the Elder problem, and the salt dome problem (HYDROCOIN level 1 case 5).

14.2 Governing Equations

In this section we will summarize the governing equations of variable-density groundwater flow and solute transport, which are described in detail e.g. by Bear and Bachmat (1990), Hassanzadeh and Gray (1979), Nield and Bejan (1992).

General balance equation

The general form of the microscopic differential balance equation of an extensive thermodynamic property (e.g. mass, linear momentum, energy) is given by

$$\frac{\partial e}{\partial t} + \nabla \cdot (e\mathbf{v}^E) = \rho f^E \quad (14.1)$$

Equation (14.1) is a classical balance law of continuum mechanics expressing the conservation principle of any thermodynamic property. Considering a continuum consisting of a number of particles: the particle velocity must be replaced by a representative velocity of the continuum, which will be a certain average velocity of a cloud of particles (e.g. mass- or volume-weighted velocities). Now, the total flux of an extensive quantity E may be divided into two parts, an advective flux corresponding to an average velocity, and a diffusive flux relative to this average velocity

$$e\mathbf{v}^E = e\mathbf{v}^c + e(\mathbf{v}^E - \mathbf{v}^c) = e\mathbf{v}^c + \mathbf{j} \quad (14.2)$$

Substituting eqn (14.2) into (14.1) the balance law takes the general form

$$\frac{\partial e}{\partial t} + \nabla \cdot (e\mathbf{v}^c) + \nabla \cdot \mathbf{j} = \rho f^E \quad (14.3)$$

Diffusive fluxes

Diffusive mass fluxes can be expressed by Fick's law. The use of a certain formulation of Fick's law in terms of mass fraction (Tab. 14.2a) or mass concentration (Tab. 14.2b) implies that diffusive mass fluxes are defined relative to mass- or volume-weighted velocities.

Subscript s is used to denote the solute component of the fluid. Quantities without subscripts are related to the fluid. The equations given above are written for a binary fluid consisting of two components, water and concentrated salt solution (solute). The mass- and volume-weighted velocities are related by the following expression (Bear and Bachmat 1990):

$$\mathbf{v}^* = \mathbf{v}' - D_m \frac{\nabla \rho}{\rho} \quad (14.4)$$

Table 14.2: Microscopic diffusive mass fluxes (Fick's law)

(a) Diffusive mass flux relative to the mass-weighted velocity (in terms of mass fraction) $\mathbf{j}^* = \rho^s(\mathbf{v}^s - \mathbf{v}^*)$	(b) Diffusive mass flux relative to the volume-weighted velocity (in terms of mass concentration) $\mathbf{j}' = \rho^s(\mathbf{v}^s - \mathbf{v}')$
Fick's law -	first order approximation of diffusive fluxes
$\mathbf{j}^* = -\rho D_m \nabla (\frac{\rho^s}{\rho}) = -\rho D_m \nabla \omega$	$\mathbf{j}' = -D_m \nabla \rho^s = -D_m \nabla C$

14.2.1 Macroscopic balance and constitutive equations

The derivation of balance equations for a porous medium requires an appropriate transformation of the microscopic relationships above to a macroscopic level. To this end, averaging rules are employed, which correspond to a representative elementary volume (REV). The following discussion is restricted to the macroscopic level. Therefore, no special symbols will be used to indicate that quantities are macroscopic ones.

Assumptions

A number of assumptions, common for groundwater flow in porous media, will be introduced:

- (1) The inertia and convective acceleration terms can be neglected (Stokes flow).
- (2) The viscous drag of the fluid on the skeleton surface is assumed to be proportional to the velocity, and stress is simply a pressure term.
- (3) Macroscopic dispersive mass fluxes in porous media may be written in analogy to Fick's law (Tab. 14.3).
- (4) Moreover, it is assumed that the advective flux by the bulk fluid flow is much larger than the sum of diffusive and dispersive ones, i.e. effects of solute dispersion to the mass flux of the whole fluid mixture are negligible.

$$|\rho \mathbf{v}| \gg |\mathbf{D} \cdot \nabla \rho| \quad (14.5)$$

Equation (14.5) can be derived from a dimensional analysis of the balance equation of the fluid mass if the following two conditions are introduced (Bear and Bachmat 1990): (i) the characteristic length of density variations is much larger than the maximum value of dispersivity $L^{(\rho)} \gg \alpha_L$ and (ii) the diffusive mass flux is much smaller than the advective one, which is expressed by large values of the Peclet number $Pe = (L^{(\rho)} v)/D_m \gg 1$. The assumption (14.5) implies that differences between volume-weighted and mass-weighted velocities vanish.

Table 14.3: Macroscopic dispersive fluxes

(a) Dispersive mass flux (in terms of mass fraction)	(b) Dispersive mass flux (in terms of mass concentration)
$\mathbf{J}^* = -\rho n \mathbf{D} \cdot \nabla \left(\frac{\rho^*}{\rho} \right) = -\rho n \mathbf{D} \cdot \nabla \omega$	$\mathbf{J}' = -n \mathbf{D} \cdot \nabla \rho^s = -n \mathbf{D} \cdot \nabla C$

Macroscopic balance equations

The mass balance equation of the fluid phase in a porous medium is given by

$$\frac{\partial n\rho}{\partial t} + \nabla \cdot (n\rho \mathbf{v}^w) = \rho Q_\rho \quad (14.6)$$

or

$$n \frac{d\rho}{dt} + \nabla \cdot (n\rho \mathbf{v}) + \frac{\rho}{1-n} \frac{dn}{dt} = \rho Q_\rho \quad (14.7)$$

where \mathbf{v}^w is the fluid velocity with respect to a fixed coordinate system, while \mathbf{v} is the fluid velocity relative to the solid; d/dt denotes the material derivative when moving with the solid phase. Depending on the formulation for dispersive fluxes chosen (Tab. 14.3), a corresponding equation of solute mass conservation can be derived, written either in terms of mass fractions

$$\frac{\partial n\rho\omega}{\partial t} + \nabla \cdot (n\mathbf{v}\rho\omega) - \nabla \cdot (n\rho \mathbf{D} \cdot \nabla \omega) = Q_\omega \quad (14.8)$$

or in terms of mass concentrations

$$\frac{\partial nC}{\partial t} + \nabla \cdot (n\mathbf{v}C) - \nabla \cdot (n \mathbf{D} \cdot \nabla C) = Q_C \quad (14.9)$$

The momentum balance equation for variable-density fluid flow in a porous medium leads to the generalized form of Darcy's law

$$\mathbf{q} = n\mathbf{v} = -\frac{\mathbf{k}}{\mu} \cdot (\nabla p - \rho \mathbf{g}) = -\mathbf{K} \cdot \left(\nabla h - \frac{\rho - \rho^w}{\rho^w} \frac{\mathbf{g}}{|g|} \right) \quad (14.10)$$

where $h = p/\rho^w - z$ is the hydraulic head related to the mass density of water ρ^w . The dimensional analysis of the momentum balance equation for fluid flow provides characteristic numbers that characterize the flow regime, e.g. the Rayleigh number of solute

$$Ra_S = \frac{KH\beta_\omega \Delta\omega}{nD_m} = \frac{KH\beta_C \Delta C}{nD_m} \quad (14.11)$$

where ΔC and $\Delta\omega$ represent characteristic differences of mass concentration and mass fraction.

Equations of state

The functional dependencies of the density on the variables of state are derived by using a total differential representation. For isothermal situations, the equations of state of the bulk fluid density and their linearized forms are given in Tab. 14.4. An equation of state for the bulk fluid viscosity of saline groundwater is e.g. presented by Lever and Jackson (1985).

Table 14.4: Equations of state of the bulk fluid density

(a) in terms of mass fraction	(b) in terms of mass concentration
$\rho = \rho^w e^{\beta_p(p - p_0) + \beta_\omega \omega}$	$\rho = \rho^w e^{\beta_h(h - h_0) + \beta_C C}$
linearized form	linearized form
$\rho = \rho^w [1 + \beta_p(p - p_0) + \beta_\omega \omega]$	$\rho = \rho^w [1 + \beta_h(h - h_0) + \beta_C C]$

It is assumed that changes of the porosity result from the compressibility of the skeleton of the porous medium due to fluid pressure variations.

$$dn = \frac{\partial n}{\partial p} dp = (1 - n) \left(\frac{1}{1 - n} \frac{\partial n}{\partial p} \right) dp = (1 - n) \alpha_p dp \quad (14.12)$$

$$dn = \frac{\partial n}{\partial h} dh = (1 - n) \left(\frac{1}{1 - n} \frac{\partial n}{\partial h} \right) dh = (1 - n) \alpha_h dh \quad (14.13)$$

Employing the equations of state for the bulk fluid density (Tab. 14.4) and the porosity, eqn (14.12) or eqn (14.13), the mass balance equation of the fluid phase can be written as

$$\begin{aligned} \rho S_0^p \frac{\partial p}{\partial t} + \nabla \cdot (\rho n \mathbf{v}) &= \rho Q_\rho - \rho n \beta_\omega \frac{\partial \omega}{\partial t} \\ S_0^p &= n \beta_p + \alpha_p \end{aligned} \quad (14.14)$$

in terms of pressure and mass fraction and

$$\begin{aligned} S_0^h \frac{\partial h}{\partial t} + \nabla \cdot (n \mathbf{v}) &= Q_\rho - n \beta_C \frac{\partial C}{\partial t} - n \mathbf{v} \cdot (\beta_C \nabla C) \\ S_0^h &= n \beta_h + \alpha_h \end{aligned} \quad (14.15)$$

in terms of hydraulic head and mass concentration, where S_0^p and S_0^h represent the specific storativity of the porous medium with respect to pressure or hydraulic head changes, respectively. Both eqns (14.14) and (14.15) are equivalent. Note, eqn (14.15) is fully developed by using the equation of state for the bulk fluid density, Tab. 14.4b. The formulations of the mass balance equations of the fluid phase given in eqns (14.14) and (14.15) are implemented in the simulators ROCKFLOW and FEFLOW, respectively.

14.2.2 Boussinesq approximation of the mass balance of the fluid

Commonly, the Boussinesq approximation, which neglects density variations within the mass balance equation of the fluid phase, is introduced. Density variations are included by the buoyancy term of the Darcy equation only. Accordingly, the equation of mass conservation of the fluid phase eqn (14.6) takes the simplified form

$$S_0^p \frac{\partial p}{\partial t} + \nabla \cdot (n\mathbf{v}) = Q_\rho \quad (14.16)$$

in terms of pressure and

$$S_0^h \frac{\partial h}{\partial t} + \nabla \cdot (n\mathbf{v}) = Q_\rho \quad (14.17)$$

in terms of hydraulic head. For the flow of a non-deformable medium this simplification substitutes the conservation of mass by the conservation of fluid volume. Analyzing eqn (14.6), it can be easily shown that this approach holds true only if the streamlines follow the density-isolines, i.e., for barocline flow. The Boussinesq approximation is inappropriate, if significant density changes arise from solute concentration variations. In this case, the contours of equipotential pressures are not orthogonal to the velocity vectors. Evans and Raffensperger (1992) studied the limitation of the Boussinesq approximation for a problem which is similar to the Elder problem. By comparing results of both formulations of the mass balance of the fluid phase by eqns (14.6) and (14.16), they found significant differences in the salinity distributions up to 9 percent.

14.2.3 Different formulations of the balance equation of solute mass

By use of the continuity equation (14.6), the divergent form of the equation of solute mass conservation, eqn (14.8), can be transformed to a convective formulation

$$n\rho \frac{\partial \omega}{\partial t} + n\rho \mathbf{v} \cdot \nabla \omega - \nabla \cdot (n\rho \mathbf{D} \cdot \nabla \omega) + \omega \rho Q_\rho = Q_\omega \quad (14.18)$$

Developing the equation of solute mass conservation in terms of mass concentration, eqn (14.9), we obtain the following relation

$$n \frac{\partial C}{\partial t} + n\mathbf{v} \cdot \nabla C - \nabla \cdot (n\mathbf{D} \cdot \nabla C) + CQ_\rho = Q_C \quad (14.19)$$

$$= Q_C + C \left((1-n)\alpha_h \frac{\partial h}{\partial t} + n\beta_C \frac{\partial C}{\partial t} + n\mathbf{v} \cdot (\beta_C \nabla C) \right) \quad (14.20)$$

If density and porosity variations are neglected for the balance equation of the solute mass conservation, eqn (14.20) can be written in a simplified form

$$n \frac{\partial C}{\partial t} + n \mathbf{v} \cdot \nabla C - \nabla \cdot (n \mathbf{D} \cdot \nabla C) + C Q_\rho = Q_C \quad (14.21)$$

Levels of the density approximation

As mentioned above, the Boussinesq approximation was employed by some authors in order to simplify the mathematical model of variable-density flow, i.e. to derive semi-analytical solutions (Henry 1964) or to introduce stream functions (Elder 1967). Moreover, different formulations of the balance equation of solute mass are used either in terms of mass fraction or in terms of mass concentration. These simplifications result in different approximation levels of density variations in the balance equations for fluid and solute masses as summarized in Tab. 14.5. The impact of these different approximation levels will be considered in the present study, especially in the discussion of the Elder problem of transient free convection.

Table 14.5: Approximation levels of density variations in the mass balance equations

- Level 1
 - mass balance equation for the fluid phase: eqn (14.17) Boussinesq approximation
 - mass balance equation for the solute component : eqn (14.21) reduced convective formulation
- Level 2
 - mass balance equation for the fluid phase: eqn (14.15) extended Boussinesq approximation
 - mass balance equation for the solute component: eqn (14.21) reduced convective formulation
- Level 3: full density approximations
 - mass balance equation for the fluid phase: eqn (14.15) extended Boussinesq approximation
 - mass balance equation for the solute component: eqn (14.9) divergent formulation
- Level 3: full density approximations
 - mass balance equation for the fluid phase: eqn (14.14)
 - mass balance equation for the solute component: eqn (14.18) convective formulation

14.3 Finite Element Formulations

Two finite element simulators, ROCKFLOW (Kolditz et al. 1999) and FEFLOW (Diersch 1994), were developed to model variable-density flow and transport in porous media. Both simulators differ in several points concerning the formulations of the basic equations as well as the numerical methods.

While ROCKFLOW is based on balance equations in terms of pressure and mass fraction, eqns (14.10), (14.14), (14.18), FEFLOW applies the adequate formulations in terms of hydraulic head and mass concentration, eqns (14.10), (14.15), (14.9).

To develop the finite element model, all variables are discretized as piece by piece continuous functions using either linear basis functions on triangular elements, or bilinear and biquadratic basis functions on quadrilateral elements in 2-D problems. To keep numerical dispersion as low as possible, no upwinding is used. In general, time derivatives are evaluated by finite difference schemes. Both simulators provide some specialities concerning the temporal approximation.

In particular, ROCKFLOW employs the explicit Euler-Taylor-Galerkin (ETG) method for the discretization of the advection-diffusion equation (Donea9). The basic idea is to expand the mass concentration in Taylor series with respect to time. Time derivatives of up to third order can be expressed by space derivatives obtained by the differentiation of the pure advection equation, which normally is the main cause of severe numerical diffusion. Diffusion, dispersion and source terms are added afterwards. The resulting differential equation, first order in time and second order in space, is handled with Galerkin's method in conjunction with the Green-Gauss theorem.

In combination with Euler time stepping, the ETG scheme leads to a truncation error of fourth order in time. The element matrices become symmetric and diagonally dominant. Highly efficient methods, e.g. PCG (pre-conditioned conjugate gradients) exist for solving this type of equations. The flow equations are set up with pressure as the only degree of freedom per node. The coefficients of the global matrix then have exactly the same pattern of storage, no index tables have to be recalculated. Crank-Nicolson time-stepping avoids fourth order error terms with the disadvantage of non-symmetric matrices where standard PCG cannot be applied (Ratke 1995).

For stability of this higher order formulation, time step size limitation according to the Courant criterion is generally required (Kröhn (1991)). The iterative coupling between the discretized flow and transport equations is realized by a fixpoint (Picard) iteration scheme, i.e. the actual densities of a new iteration step are calculated from the field variables (pressure and salt concentration) of the previous iteration step.

Truncation errors and resulting effective Rayleigh numbers

The spatial and temporal discretization can introduce spurious dispersion effects where the amount of (physical) hydrodynamic dispersion is enlarged by the numerical dispersion

$$D^{eff} = D^{phys} + D^{num} \quad (14.22)$$

The numerical dispersion results from the spatial and temporal discretization. To estimate the actual dispersion effective in the numerical approach, the truncation errors are determined

$$\begin{aligned} D_{spatial}^{num} &\approx \alpha \frac{v\Delta l}{2} + O(\Delta l^2) \\ D_{temporal}^{num} &\approx (\theta_t - \frac{1}{2})v^2 \Delta t_n + O(\Delta t_n^2) \end{aligned} \quad (14.23)$$

where l is the characteristic element length, t^n is the time step increment at time level n , α is the upwind parameter, and θ_t is the weighting factor of the time collocation point. Now, the numerical solutions can be characterized by an effective Rayleigh number

$$Ra_S^{eff} \approx \frac{Ra_S}{1 + \frac{\alpha}{2}Pg + (\theta_t - \frac{1}{2})CrPg} \quad (14.24)$$

The upwind parameter is zero for the Galerkin finite element method and unity for the streamline upwind method. The weighting factor of the time collocation point is 0.5 for the Crank-Nicolson as well as 2nd order Adams-Bashforth/Trapezoid-Rule schemes and is unity for fully implicit schemes. It is obvious that especially upwind and fully implicit techniques can change the physical conditions actually simulated by the numerical model. Therefore, appropriate schemes should be preferred to keep the numerical dispersion as low as possible, especially for the investigation of convection problems.

Substituting Darcy's law, eqn (14.10), into the balance equation for the overall fluid mass (14.13), a combined flow equation can be derived

$$\rho S_0 \frac{\partial p}{\partial t} - \nabla \cdot (\rho \frac{\mathbf{k}}{\mu} \cdot \nabla p) = \rho Q_\rho - n\rho\beta_\omega \frac{\partial \omega}{\partial t} - \nabla \cdot (\rho \frac{\hat{k}}{\mu} \cdot \mathbf{g}) \quad (14.25)$$

The weighted residual method applied to this flow equation yields the following weak formulation

$$\begin{aligned} \int_A N_i \left(S_0 N_j \rho_j \frac{N_k \Delta p_k}{\Delta t} - \frac{k_{\alpha\beta}}{\mu} \frac{\partial}{\partial x_\alpha} (N_j \rho_j \frac{\partial N_k}{\partial x_\beta})(p_k^{(n)} + \theta_t \Delta p_k) \right) dA = \\ \int_A N_i \left(N_j \rho_j (Q_\rho^{(\theta_t)} - n\beta_\omega \frac{N_j \Delta \omega_j}{\Delta t}) - \frac{k_{\alpha\beta}}{\mu} \frac{\partial}{\partial x_\alpha} (N_j \rho_j N_l \rho_l) g_\beta \right) dA \end{aligned} \quad (14.26)$$

where indices i, j, k denote nodal values, and indices α, β denote coordinates. The Einstein summation convention is used for repeated indices.

The unknown pressures $p_k^{(n+1)} = p_k^{(n)} + \Delta p_k$ are evaluated from the relationship at time level $t^{\theta_t} = t^n + \theta_t \Delta t$. Applying the Gauss-Green theorem to terms containing second order derivatives we obtain

$$\begin{aligned} & \left[\frac{S_0 \rho_j}{\Delta t} \int_A N_i N_j N_k dA + \theta_t \frac{k_{\alpha\beta}}{\mu} \rho_j \int_A \frac{\partial N_i}{\partial x_\alpha} N_j \frac{\partial N_k}{\partial x_\beta} dA \right] \Delta p_k \\ &= (\rho Q_\rho^{(\theta_t)})_j \int_A N_i N_j dA - \frac{n \rho_j \beta_\omega \Delta \omega_k}{\Delta t} \int_A N_i N_j N_k dA \\ &- \frac{k_{\alpha\beta}}{\mu} \rho_j p_k^{(n)} \int_A \frac{\partial N_i}{\partial x_\alpha} N_j \frac{\partial N_k}{\partial x_\beta} dA \\ &- \frac{k_{\alpha\beta}}{\mu} g_\beta \rho_j \rho_l \int_A \frac{\partial N_i}{\partial x_\alpha} N_j N_l dA + I_{\rho_i}^{(\theta_t)} \end{aligned} \quad (14.27)$$

The total flux of fluid mass through the element boundaries can be represented by concentrated nodal load (source) terms.

$$\begin{aligned} I_{\rho_i}^{(\theta_t)} &= \frac{k_{\alpha\beta}}{\mu} \rho_j (p_k^{(n)} + \theta_t \Delta p_k) \int_S N_i N_j \frac{\partial N_k}{\partial x_\beta} n_\alpha dS \\ &- \frac{k_{\alpha\beta}}{\mu} g_\beta \rho_j \rho_l \int_S N_i N_j N_l n_\alpha dS \end{aligned} \quad (14.28)$$

A given source of fluid mass at node i must be equal to the sum of the fluxes taken from all elements adjacent to that node. For nodes without sources, this sum vanishes obviously. The condition, eqn (14.27), sets up the global equation system for the unknown nodal pressures at the new time level.

Velocities for each finite element are computed afterwards by the relation

$$v_\alpha = -\frac{k_{\alpha\beta}}{n\mu} \cdot (p_k \frac{\partial N_k}{\partial x_\beta} - n_j \rho_j g_\beta) \quad (14.29)$$

The use of the Euler-Taylor-Galerkin method (Donea 1984) results in a modified equation for the salt mass conservation, which contains additional terms due to the higher order approximation in time.

$$\begin{aligned} n\rho \left[\frac{\partial \omega}{\partial t} - \frac{\Delta t^2}{6} v_\alpha v_\beta \frac{\partial}{\partial t} \left(\frac{\partial^2 \omega}{\partial x_\alpha \partial x_\beta} \right) - D_{\alpha\beta} \frac{\partial^2 \omega}{\partial x_\alpha \partial x_\beta} \right] &= \\ \rho Q_\omega \omega - n\rho \left[v_\alpha \frac{\partial \omega^{(n)}}{\partial x_\alpha} - \frac{\Delta t}{2} v_\alpha v_\beta \frac{\partial^2 \omega^{(n)}}{\partial x_\alpha \partial x_\beta} \right] \end{aligned} \quad (14.30)$$

The same procedure as for the equation of fluid mass conservation is applied to eqn (14.30), resulting in the following finite element formulation

$$n\rho_j \left[\frac{1}{\Delta t} \int_A N_i N_j N_k dA + (\theta_t D_{\alpha\beta} + \frac{\Delta t}{6} v_\alpha v_\beta) \int_A \frac{\partial N_i}{\partial x_\alpha} N_j \frac{\partial N_k}{\partial x_\beta} dA \right] \Delta \omega_k$$

$$\begin{aligned}
&= (\rho Q \rho \omega)_k^{(\theta_t)} \int_A N_i N_k dA \\
&- n \rho_j \left[v_\alpha \int_A N_i N_j \frac{\partial N_k}{\partial x_\alpha} dA + \left(\frac{\Delta t}{2} v_\alpha v_\beta + D_{\alpha\beta} \right) \int_A \frac{\partial N_i}{\partial x_\alpha} N_j \frac{\partial N_k}{\partial x_\beta} dA \right] \omega_k^{(n)} \\
&+ I_{\omega i}^{(\theta_t)} + J_{\omega i}^{(\theta_t)}
\end{aligned} \tag{14.31}$$

The boundary integrals $I_{\omega i}^{(\theta_t)}$ and $J_{\omega i}^{(\theta_t)}$ result from applying the Gauss-Green theorem to the discretized form of eqn (14.30)

$$I_{\omega i}^{(\theta_t)} = n \rho_j D_{\alpha\beta} (\omega_k^{(n)} + \theta_t \Delta \omega_k) \int_S N_i N_j \frac{\partial N_k}{\partial x_\alpha} n_\beta dS \tag{14.32}$$

$$J_{\omega i}^{(\theta_t)} = n \rho_j v_\alpha v_\beta \left(\frac{\Delta t}{2} \omega_k^{(n)} - \frac{\Delta t}{6} \Delta \omega_k \right) \int_S N_i N_j \frac{\partial N_k}{\partial x_\alpha} n_\beta dS \tag{14.33}$$

Both integrals vanish at the boundary of all complete element patches assembled around node i , because the test function N_i is zero on all patch outer edges. They are also neglected for nodes residing on the system boundary, which implies suppression of the diffusive-dispersive flux across the border. This is equivalent to the condition $\nabla \omega \cdot \mathbf{n} = 0$

$J_{\omega i}^{(\theta_t)}$ contains Euler-Taylor correction terms only. At inflow boundaries, the mass fraction must be prescribed, whereas it is swept out of the system at outlet nodes within the next time step. Therefore, no correction is needed for any boundary nodes. Nodal equations are obtained from assembling the element formulations (14.31) in the usual manner. If present, concentrated external loads (solute mass per unit time) must be added to the right hand side to maintain an overall mass balance at the nodes.

To avoid non-symmetrical terms within the discretized transport equation, the interpolation of the density function is introduced after the reduction of second order derivatives by the Green-Gauss formula. The setup condition for the global system is the same as for the flow equation.

14.4 Examples

The usual verification procedure by use of analytical solutions (Häfner et al. 1992) is complicated due to the non-linear nature of variable-density flow and transport. In general, there are no exact solutions of this type of problem for comparison, except a semi-analytical solution by Henry (1964) for a steady-state case. Therefore, a cross-verification of results obtained by different codes is required to test numerical models. A particular series of benchmarks will be carefully analyzed by use of the simulators ROCKFLOW and FEFLOW to show that the numerical models accurately represent the physics implied by

the governing differential equations and, finally, to build confidence in the numerical modeling of convection phenomena. We present results for three benchmark problems: (14.4.1) the classic seawater intrusion problem of Henry (1964), termed the Henry problem; (14.4.2) the free convection problem by Elder (1967) (fingering problem), called the Elder problem; and (14.4.3) the salt dome problem, which is the test case 5 of level 1, that was proposed by the international HYDROCOIN (1986) project for the verification of groundwater models. In addition to density variations, both simulators ROCKFLOW and FEFLOW can also account for viscosity variations. The considered standard benchmarks are stated for variable-density, but constant-viscosity flow. Therefore, to make our results comparable to prior works, we keep the viscosity constant through our simulations. Since viscosity variations can become important in brine systems, the standard benchmarks for saltwater intrusion should be extended to variable-viscosity flow in forthcoming cross-verification tests, accordingly.

14.4.1 The Henry problem

Definition of the problem

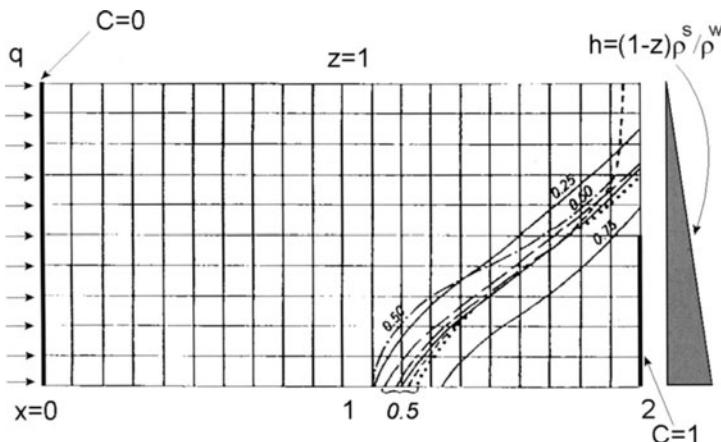


Figure 14.1: Definition of the Henry problem and prior results by Henry (1964) - dashed line, Pinder and Cooper (1970) - dashed-dotted line, Segol et al. (1975) - dotted line, Desai and Contractor (1977) - long-dashed line, Frind (1982) - short-long-dashed line, and Voss and Souza (1987) - solid line; positions of the 25, 50, and 75 percent isochlors of the steady-state solution (Fig. adapted from Voss and Souza (1987))

The Henry problem describes the advance of a saltwater front in a confined aquifer which was initially saturated with uncontaminated fresh water. Henry (1964) developed a semi-analytical solution technique for this problem. Based on

the Boussinesq approximation he derived analytical expressions for the stream function and the salt concentration in the form of Fourier series. The resulting algebraic equations for determining the coefficients of the Fourier series must be solved by numerical techniques. Using quite different approximation methods, a number of authors obtained similar results (Pinder and Cooper 1970, Segol et al. 1975, Desai and Contractor 1977, Frind 1982, Voss and Souza 1987, Kröhn and Zielke 1991, Oldenburg and Pruess 1995, Croucher and O'Sullivan 1995). However, the "mystery" of Henry's solution is, that no numerical model so far has been able to reproduce closely his semi-analytical results (Segol 1994). This may be due to some form of inaccuracy in the original findings by Henry (1994) (Croucher and O'Sullivan 1995). The Boussinesq approximation, which was assumed by Henry, requires perpendicular Darcy velocity vectors and density gradients. However, if inspecting the results obtained by Henry (1964), streamlines and density gradients are found to be rather colinear in the region of the outflow from the model domain. Nevertheless, as there exists no other non-numerical technique for this kind of non-linear problem, Henry's solution has become one of the standard tests of variable-density groundwater models. The idealized aquifer for the simulation of Henry's problem is shown in Fig. 14.1. The boundary conditions for flow consist of impermeable borders along the top and the bottom. Hydrostatic pressure is assumed along the vertical boundary of the sea side. The aquifer is charged with fresh water at a constant flux from the left side. At the inland side, the concentration is zero, which corresponds to a fresh water condition. At the coastal side the normalized concentration of brine is imposed (Fig. 14.1). Instead of velocity-dependent dispersion a correspondingly large diffusivity was used by Henry (1964) in order to allow a semi-analytical solution. The simulation parameters for the Henry problem are given in Tab. 14.6.

Table 14.6: Simulation parameters for the Henry problem

Symbol	Quantity	Value	Unit
D_m	coefficient of molecular diffusion	6.6×10^{-6}	$m^2 s^{-1}$
\mathbf{g}	vector of gravity acceleration	9.81	ms^{-2}
k	permeability of the porous medium	1.019368×10^{-9}	m^2
\mathbf{q}	specific discharge on the left side	6.6×10^{-2}	$kg m^{-1} s^{-1}$
Ra_S	Rayleigh number of solute	250	—
n	porosity	0.35	—
μ	dynamic viscosity	10^{-3}	$kg m^{-1} s^{-1}$
ρ^w, ρ^s	density of water and solute	$(1, 1.025) \times 10^3$	$kg m^{-3}$

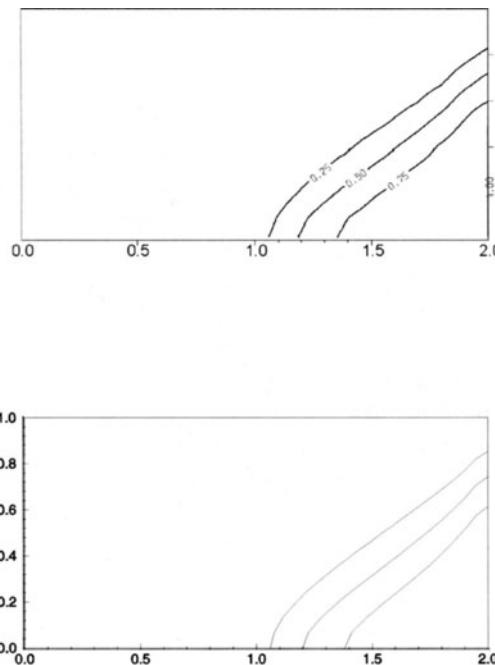


Figure 14.2: Positions of the 25, 50, and 75 percent isochlors of the steady-state solution for the Henry problem (a) FEFLOW simulation and (b) ROCKFLOW simulation

Results and discussion

Figure 14.1 summarizes some former findings for the Henry problem obtained by several authors, who used quite different calculation methods. Comparing these results, it has to be kept in mind that slightly different parameter values were chosen by the authors. There have been some discrepancies in the use of the diffusion coefficients. Note, the effective diffusivity is the product of the molecular diffusion coefficient and the porosity. We have checked the sensitivity of the model due to different values of the coefficients of molecular diffusion and hydrodynamic dispersion (Kolditz 1994, Ratke 1995). Seawater intrudes into the model domain through the lower right boundary. In this area, where the density is highest, pressure gradients are oriented almost vertically upward. In contrast, the gravitational force is directed vertically downward. These two driving forces cause a small lateral flow which in turn causes the observed seawater intrusion into the aquifer domain. As density differences decrease along the aquifer bottom, the potential-driven flow forced by the fresh water influx from the left hand side becomes more important. Finally, the velocity directions

are turned into the opposite direction and the less dense fluid leaves the domain through the upper right boundary section. As a result, a stable "cline" forms due to the situation that denser brine moves under the fresh water. Results of the ROCKFLOW and FEFLOW simulations for the Henry problem are presented in Figs 14.2a and 14.2b, where the steady-state salinity distributions are shown. We use varying spatial discretizations to prevent grid effects, i.e. (i) a regular grid of 200 bilinear finite elements with an element length of 0.1m and (ii) a grid of 581 linear triangular finite elements. Our results are in good agreement with prior works. Steady state is reached after about three hours of the convection process. The Henry problem is used as a benchmark for variable-density flow and transport although it has some deficiencies. At first, an unrealistically large amount of diffusion is introduced which results in a widely dispersed transition zone. Furthermore, the Henry problem is not appropriate for verifying purely density-driven flow. Additional benchmark tests are necessary to check the numerical models for free convection problems (the Elder problem) and situations with narrow transition zones (the salt dome problem).

14.4.2 The Elder problem

Definition of the problem

The Elder problem serves as an example of free convection phenomena, where the bulk fluid flow is driven purely by fluid density differences. Elder (1966,1967) presented both experimental and numerical studies concerning the thermal convection produced by heating a part of the base of a porous layer. The original experiment, which was performed in a Hele-Shaw cell, was called the "short-heater problem". Elder conducted these studies mainly to verify the finite difference model he used for the numerical analysis of thermally-driven convection. Furthermore, he suggested criteria for preventing numerical instabilities. The Elder problem can be used as a thermal analogue for the salt water intrusion by density-driven convection. Due to the characteristic salinity pattern developing it is also denoted as a fingering problem (Fig. 14.3). The simulation parameters and boundary conditions for the saline Elder problem are given in Tab. 14.7 and Fig. 14.3. In the second part of the paper we will present thermohaline as well as 3-D extensions of the Elder problem.

Numerical analysis

Due to the absence of an exact solution for the Elder problem a cross-verification of several numerical approaches is necessary. First, we aim at a possible reproduction of prior findings by other authors. Elder (1967) used a finite difference model to approximate the governing balance equations for vorticity, streamfunction, and heat. The equations are solved by repeated application of Leibmann's extrapolation method with alternating scanning directions. Voss and Souza (1987) developed a finite element model for the simulation of the saline

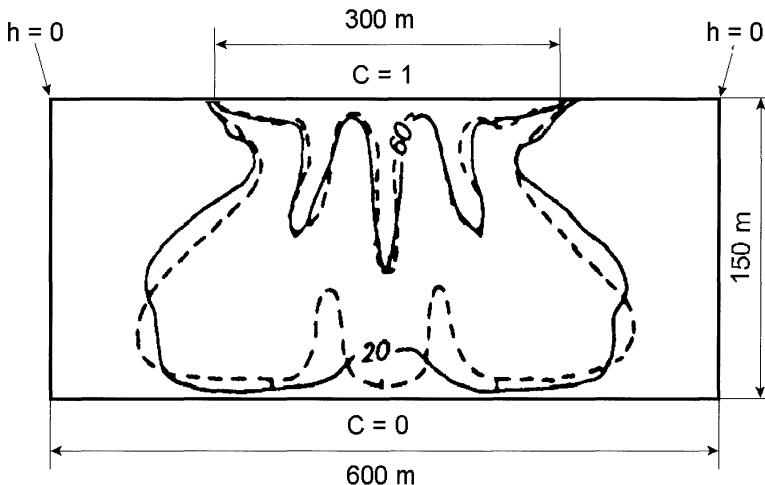


Figure 14.3: Definition of the Elder problem and prior results by Elder (1967) - solid line and Voss and Souza (1987) - dashed line; positions of the 20 and 60 percent isochlors at 10 years simulation time (Fig. adapted from Voss and Souza (1987))

Table 14.7: Simulation parameters for the Elder problem

Symbol	Quantity	Value	Unit
D_m	coefficient of molecular diffusion	3.565×10^{-6}	$m^2 s^{-1}$
\mathbf{g}	vector of gravity acceleration	9.81	ms^{-2}
k	permeability of the porous medium	4.845×10^{-13}	m^2
Ras	Rayleigh number of solute	400	—
n	porosity	0.1	—
β_C	volumetric concentration expansion	0.2	—
μ	dynamic viscosity	10^{-3}	$kgm^{-1}s^{-1}$
ρ^w, ρ^s	density of water and solute	$(1, 1.2) \times 10^3$	kgm^{-3}

Elder problem. The non-linear equations are solved applying an iteration procedure. Figure 14.3 shows results by Elder (1967), Voss and Souza (1987) for a calculation time of ten years. Using a similar spatial discretization, we can reproduce these salinity patterns very well (Fig. 14.4a). However, we observe strong discretization effects for these coarse meshes. At first, we investigated the influence of different temporal discretization schemes, where time marching schemes of different order of accuracy are applied: (i) a fully implicit, and Crank-Nicolson schemes in combination with the Newton-Raphson method, and (ii) one-step predictor-corrector methods of first and second order accuracy em-

bedded in a full Newton method. The discrepancies in the results indicate that a convergent approximate solution has not yet been reached (Kolditz 1994).

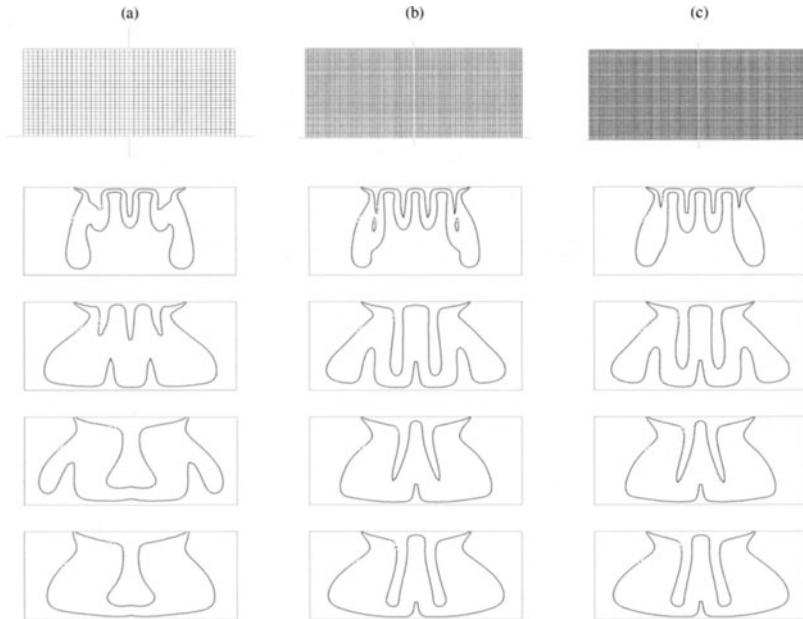


Figure 14.4: Elder problem - Effect of spatial discretization on the computed salinity evolution at 4, 10, 15, and 20 years simulation time; positions of the 20 and 60 percent isochlors: (a) coarse mesh (1170 grid points, 1100 finite elements), similar to the discretization used by Elder (1967) and Voss and Souza (1987), (b) fine mesh (4539 grid points, 4400 bilinear finite elements), (c) very fine mesh (10108 grid points, 9900 finite elements)

Therefore, we consecutively refine the meshes until grid convergence is achieved (Fig. 14.4). To this end, a spatial discretization consisting of at least 4400 bilinear finite elements (Fig. 14.4b, 88 elements in horizontal and 50 elements in vertical direction) will be required. Note, the number of convection cells is doubled for the fine mesh calculations and, furthermore, the resulting convection patterns are quite different. There is a central upward flow rather than a central downward flow, resulting from the coarse mesh calculations. Finally, our high-resolution results obtained with both simulators (Figs 14.4c and 14.6) compare very well with those given by Oldenburg and Pruess (1995). A mesh analysis is provided by use of the grid Peclet number, which is defined as the ratio of (grid) advective to (grid) dispersive fluxes. The vertical grid Peclet number can be written by

$$\begin{aligned}
 Pg_z = \frac{q}{nD_m} \Delta z &\approx \frac{K(\frac{\Delta h}{H} - \beta_C \Delta C)}{nD_m} \Delta z \\
 &= \left[\frac{K\Delta h}{nD_m} - \frac{KH\beta_C \Delta C}{nD_m} \right] \frac{\Delta z}{H} \\
 &= \left[\frac{k g \rho^w \Delta h}{\mu n D_m} - Ra_S \right] \frac{\Delta z}{H}
 \end{aligned} \tag{14.34}$$

where ΔC is the maximum difference of mass concentration, Δh is the maximum difference of hydraulic head, and Δz is the characteristic vertical element length. Note, a relative large diffusivity is used instead of hydrodynamic dispersion for the Elder problem. If the vertical velocity is estimated by the buoyancy term only, a maximum grid Peclet number can be determined by use of the Rayleigh number, which is equal to 400 for the Elder problem. The maximum difference of the calculated hydraulic head is equal to 15 m. Therefore, the following estimation of the vertical grid Peclet number is given for the Elder problem

$$Pg_z \approx 200 \frac{\Delta z}{H}. \tag{14.35}$$

Now, the corresponding grid Peclet numbers of the several meshes above can be characterized by the vertical discretization z/H and are equal to $Pg_z = 8$ for the coarse mesh (Fig. 14.4a), $Pg_z = 4$ for the fine mesh (Fig. 14.4b), and $Pg_z = 2.5$ for the very fine mesh (Fig. 14.4c). In fact, these grid Peclet numbers are still larger than the theoretical stability criterion of linear advection-diffusion problems (e.g $Pg < 2$ for bilinear finite elements, Huyakorn and Pinder (1983)). The grid convergence test (Fig. 14.4) may be the best proof of stability of the solution of this non-linear problem, where no exact discretization criteria exist. This study reveals, that a grid convergent numerical solution can be obtained for grid Peclet numbers smaller than 5 for the Elder problem.

Levels of the density approximation

According to Tab. 14.5, we distinguish three different levels of the density approximation:

- Level 1: Usually, the Boussinesq approximation implies, that density effects are considered only in the buoyancy term of the momentum equation. Density variations in the balance equations of mass of fluid and mass of solute are neglected. In this case, the set of mass balance equations is given by eqns (14.17) and (14.21).
- Level 2: The extended Boussinesq approximation accounts for density variations in the balance equation of the fluid mass, according to eqn (14.15).

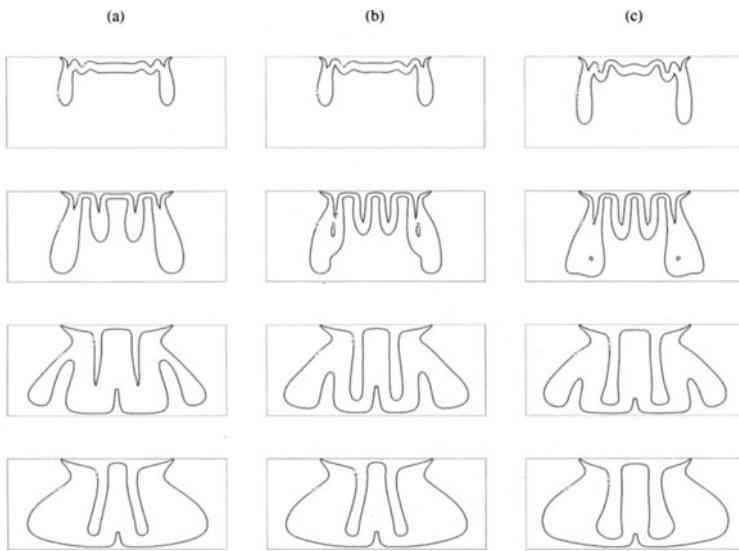


Figure 14.5: Elder problem - Computed salinity distributions (20 and 60 percent isochlors) at 2, 4, 10, and 20 years: (a) level 1, (b) level 2, and (c) level 3 of the density approximation (FEFLOW simulation)

- Level 3: The full density approximation accounts for density variations in all balance equations, according to eqns (14.14) and (14.18) in terms of mass fractions and according to eqns (14.15) and (14.9) in terms of mass concentrations.

In order to evaluate the impact of the Boussinesq approximation, we compare Figs 14.5a with 14.5b showing the salinity evolution corresponding to the level 1 and level 2 approximations. Differences are apparent at early stages, when large concentration contrasts are involved. During the first 10 years, considerable discrepancies in the solute distributions are found. Concerning the long-term behaviour, where density differences are already smoothed, these deviations decrease again. Finally, the concentration distributions become nearly identical at 20 years. Regarding the density variations in the equation describing conservation of the mass of solute, the results of the level 2 simulation and the full density approximation of level 3 are analyzed. If comparing the computed salinity evolution in Figs 14.5b and 14.5c, we find the isohalines propagate more rapidly into the depth of the cavity for the full density approximation, i.e. while neglecting density variations in the balance equation of solute mass, the salt water intrusion into the aquifer is apparently underestimated. Obviously, the differences between the simulation results of the level 2 and level 3 approximations are smaller than those of the level 1 and level 2 approximations. Finally,

both simulations using the full density approximations (level 3) are compared. We find a good agreement between the results of the FEFLOW (Fig. 14.5c) and the ROCKFLOW simulations (Fig. 14.6). Note, the balance equations of the solute mass are given in different forms, the divergent (FEFLOW) and the convective (ROCKFLOW). The divergent form of the mass balance equation (8) has the distinct advantage in conserving the total (dispersive and advective) mass flux through boundaries. The good agreement of the simulation results proves the reliability of both simulators.

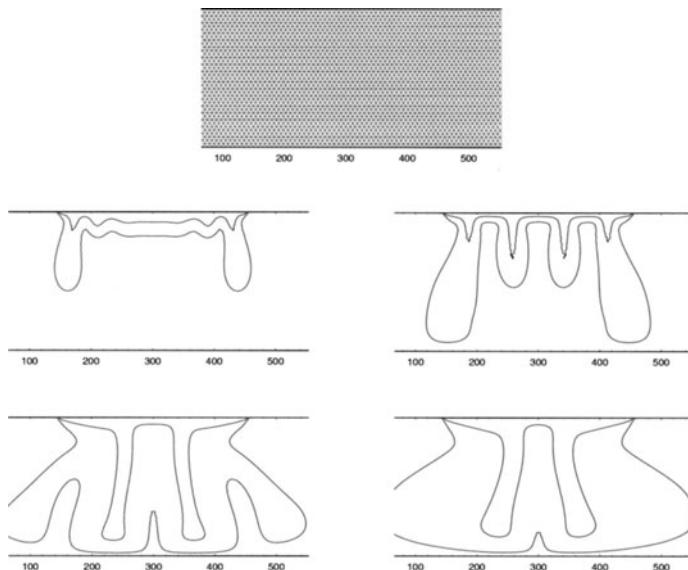


Figure 14.6: Elder problem - Finite element mesh and computed salinity distributions (20 and 60 percent isochlors) at 1, 2, 4, 10, and 20 years simulated time (from top to bottom): Level 3 of the density approximation (ROCKFLOW simulation)

14.4.3 The salt dome problem

Definition of the problem

In this section we discuss one of the test cases that was proposed by the participants of the international HYDROCOIN project for the verification of groundwater models (Swedish Nuclear Power Inspectorate 1986). This test case is addressed to model variable-density groundwater flow over a hypothetical salt dome, where the geometry is largely simplified. Attention is paid to the physics

of flow of a binary fluid, whose density strongly depends on its salt concentration. The geometry and boundary conditions of the test problem are shown in Fig. 14.7. The cross-section of the model extends horizontally 900m and vertically 300m. The aquifer is considered to be homogeneous and isotropic. The pressure/hydraulic head varies linearly on the top of the aquifer. The other sides are impervious to flow. The concentration on the top is set equal to zero at the inflow domain. The middle section of the base represents the top of the salt dome with normalized mass concentration of solute equal unity. All remaining parts of the boundary are closed for diffusive fluxes (i.e. $\partial C/\partial n = 0$). The simulation parameters are listed in Tab. 14.8. In general, we obtain the same stratified system, as Herbert et al. (1988) found. This becomes obvious when studying the streamline contours which provide a good visualization of the hydrodynamic system (Fig. 14.7b). A fresh water region with higher velocities is observed in the upper part, where flow is driven by the superimposed pressure gradient on the top of the aquifer (Fig. 14.7a) and there is a brine pool along the bottom, where flow with small velocities recirculates. The outflow of the saltwater is focused on the upper right-hand corner (Fig. 14.7c). With decreasing diffusivities the fresh water zone is moving deeper into the reservoir, in other words, there are higher limits for the rise of solute.

Table 14.8: Simulation parameters for the salt dome problem (HYDROCOIN level 1 case 5)

Symbol	Quantity	Value	Unit
D_m	coefficient of molecular diffusion	$0, 1.39 \times 10^{-8}$	$m^2 s^{-1}$
\mathbf{g}	vector of gravity acceleration	9.80665	ms^{-2}
k	permeability of the porous medium	10^{-12}	m^2
Ra_S	Rayleigh number of solute	400	—
Δp	pressure difference at the top	10^5	Pa
α_L	longitudinal dispersivity	20	m
α_T	transversal dispersivity	2	m
β_C	volumetric concentration expansion	0.2036108	—
n	porosity	0.2	—
μ	dynamic viscosity	8.9×10^{-4}	$kg m^{-1} s^{-1}$
ρ^w, ρ^s	density of water and solute	$(0.997, 1.2) \times 10^3$	$kg m^{-3}$

Simulation technique

Herbert et al. (1988) who applied the NAMMU code for the simulation of the salt dome problem developed a modified standard Galerkin finite element method by using mixed finite element interpolations for the field variables - pressure and solute concentration. Velocities are computed continuously by applying a separate finite element scheme for the velocity function. Furthermore,

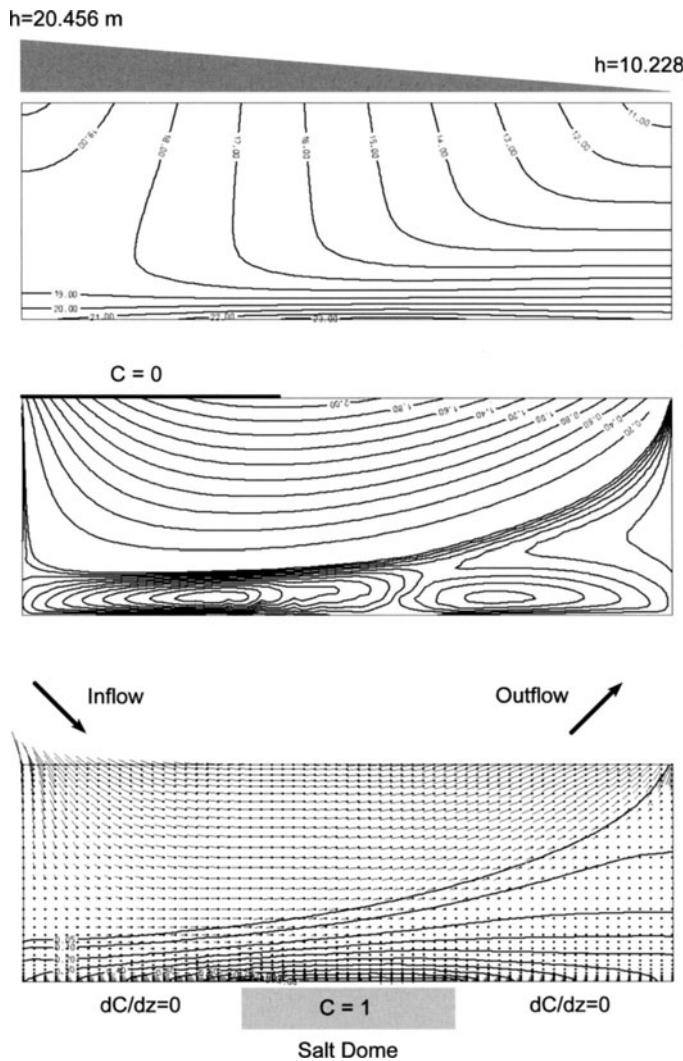


Figure 14.7: Definition of the salt dome problem (HYDROCOIN Level 1 Case 5) - domain and boundary conditions: (a) hydraulic head, (b) streamline, (c) velocity and salinity distributions

Herbert et al. (1988) employed a steady-state strategy to obtain the long-term solution. They used parameter stepping for the diffusivities and dispersivities. However, such parameter increments must be small enough to guarantee satisfactory convergence by the Newton-Raphson iteration scheme. Kolditz and Diersch (1993) applied successfully a similar parameter stepping procedure for the numerical simulation of hydrothermal convections in thermoelastic fractures. In contrast to Herbert et al. (1988), we calculated the fully transient evolution of the convection system. Steady-state solutions were reached after about 200 years total simulation time. We used two different techniques to find the steady-state solution. First, the ROCKFLOW simulations were performed with comparatively small time steps (governed by the Courant criterion in the explicit approach). The comparison of the 200 with the 400 years results, where no more changes of the salinity distributions were detected, indicated that steady-state was reached. Secondly, the FEFLOW simulations were conducted with an automatic time step control based on a predictor-corrector method (second order Adams-Bashforth/trapezoid scheme). After about 100 years, the time steps grew very rapidly, which means that only very small changes in pressure and salinity distributions occur. We stopped the simulation after a simulation time of about 200 years. Oldenburg and Pruess (1995) achieved steady-state solutions after about 100 years (starting from a fresh water pool as initial condition).

Results and discussion

In the past, the salt dome problem was analyzed for different values of molecular diffusivity and mechanical dispersivity (Tab. 14.9).

Table 14.9: Values of molecular diffusivity and dispersivities for the salt dome problem

Parameters Model	Molecular diffusivity D_m	Longitudinal dispersivity α_L	Transverse dispersivity α_T
Diffusive	$5 \times 10^{-6} m^2 s^{-1}$	0	0
	$5 \times 10^{-7} m^2 s^{-1}$	0	0
	$5 \times 10^{-8} m^2 s^{-1}$	0	0
Diffusive-dispersive	$5 \times 10^{-8} m^2 s^{-1}$	$20m$	$2m$
	$1.39 \times 10^{-8} m^2 s^{-1}$	$20m$	$2m$
Dispersive	0	$20m$	$2m$

Kolditz (1994) studied different diffusive/dispersive models of the salt dome problem and compared the results with prior findings presented by Herbert et al. (1988) and Oldenburg and Pruess (1995). In general, our results agree with those of Herbert et al. (1988) rather than with those by Oldenburg and Pruess (1995). Recently, Rivera and Jones (1995) confirmed the prior NAMMU

results by Herbert et al. (1988) using the new 6.2 version of this simulator. The most interesting cases for that inconsistent results exist will be discussed below (Tab. 14.9, emphasized values). Figure 14.8 shows the computed steady-state salinity distributions for the case of $D_m = 1.39 \times 10^{-8} m^2 s^{-1}$ and $L = 20m$, $\alpha_T = 2m$. Note, the results by Herbert et al. 20 correspond to values of $D_m = 1.39 \times 10^{-8} m^2 s^{-1}$ (Fig. 14.8a), whereas Oldenburg and Pruess (1995) used a zero molecular diffusivity (Fig. 14.8b). We computed the models for both values of molecular diffusivity (Figs 14.7c and 8c) and found nearly identical salinity distributions. Oldenburg and Pruess (1995) computed a quite different salinity distribution. They called their results a 'fully swept-forward' pattern (Fig. 14.8b). There is a substantial difference in the number of convection cells. This swept-forward pattern covers only one instead of two recirculating cells near the aquifer bottom (Fig. 14.7b). Oldenburg and Pruess (1995) argued that for large dispersive fluxes ($D > 2 \times 10^{-7} m^2 s^{-1}$) the amount of brine rising from the bottom source is sufficient to cause a clockwise recirculation against the overall counterclockwise flow imposed by the pressure difference along the aquifer top. In contrast, the buoyancy forces are insufficient to overwhelm the pressure-driven flow for small dispersive fluxes (i.e. $D < 10^{-7} m^2 s^{-1}$). As a consequence, the brine is swept forward from left to right with the overall flow. Kolditz (1994) obtained similar swept-forward regimes for the case of lower Rayleigh numbers, i.e. if buoyancy effects are less significant. Different levels of the approximation of the density variations are employed for the analysis of the salt dome problem, i.e. level 2 (Fig. 14.8c) and level 3 (Fig. 14.8d). The good agreement of the simulation results indicates that the different levels of density approximations have no influence on the steady-state salinity distribution in this case.

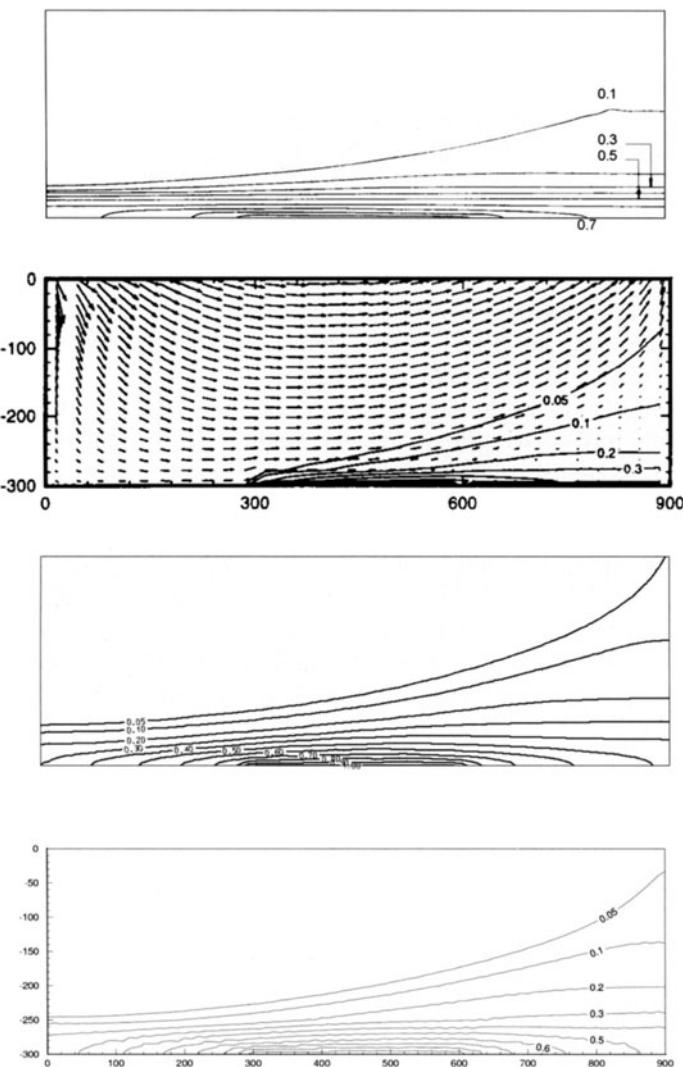


Figure 14.8: Results for the salt dome problem: steady state salinity contours computed by (a) Herbert et al. (1988), (b) Oldenburg and Pruess (1995), (c) FEFLOW simulation, (d) ROCKFLOW simulation

Bibliography

- [1] Bastian, P., UG - Ein Programmabaukasten zur schnellen adaptiven Lösung partieller Differentialgleichungen. Preprint, Institut für Wissenschaftliches Rechnen, Universität Stuttgart, 1992.
- [2] Bear, J. & Bachmat, Y., Introduction to modeling of transport phenomena in porous media. Kluwer Academic Publisher, Dordrecht, 1990.
- [3] Clauser, C. & Kiesner, S., A conservative, unconditionally stable, second-order three-point differencing scheme for the diffusion-convection equation. Geophys. J. R. Astr. Soc., 91 (1987) 557-568.
- [4] Croucher, A. E. & O'Sullivan, M. J., The Henry problem for saltwater intrusion. Water Resources Research, 31(7) (1995) 1809-1814.
- [5] Desai, C. S. & Contractor, D. N., Finite element analysis of flow, diffusion, and salt water intrusion in porous media. In: Formulation and Comp. Algorithms in Finite Element Analysis, Bathe, K.-J. et al. (ed), Cambridge, Massachusetts, MIT, 1977.
- [6] Diersch, H.-J., Primitive variables finite element solutions of free convection flow in porous media. Zeitschr. Angewandte Math. Mech. (ZAMM), 61(7) (1981) 325-337.
- [7] Diersch, H.-J., Finite element modelling of recirculating density-driven saltwater intrusion processes in groundwater. Advances in Water Resources, 11(1) (1988) 25-43.
- [8] Diersch, H.-J., Interactive, graphics-based finite-element simulation system - FEFLOW - for modeling groundwater flow and contaminant transport processes. WASY - Gesellschaft für wasserwirtschaftliche Planung und Systemforschung mbH, Berlin, 1994.
- [9] Donea, J., A Taylor-Galerkin method for convective transport problems. Int. J. Numerical Methods in Engineering, 20 (1984) 101-119.
- [10] Elder, J. W., Numerical experiments with a free convection in a vertical slot. J. Fluid Mech., 24(4) (1966) 823-843.
- [11] Elder, J. W., Transient convection in a porous medium. J. Fluid Mech., 27(3) (1967) 609-623.
- [12] Evans, D. G. & Raffensperger, J. P., On the stream function for variable density groundwater flow. Water Resources Research, 28(8) (1992) 2141-2145.
- [13] Frind, E. O., Simulation of long-term transient density dependent transport in groundwater. Adv. Water Resour., 5 (1982) 73-97.
- [14] Häfner, F., Sames, D. & Voigt, H.-D., Wärme- und Stofftransport - Mathematische Methoden. Springer Verlag, Berlin - Heidelberg - New York, ISBN 3-540-54665-0, 626S, 1992.
- [15] Hassanizadeh, M. S. & Gray, W. G., General conservation equation for multi-phase systems: 2. Mass, momentum, energy, and entropy equations. Adv. Water Resour., 2 (1979) 191-203.
- [16] Hassanizadeh, M. S., Derivation of basic equations of mass transport in porous media: 2. Generalized Darcy's and Fick's laws. Adv. Water Resour., 9 (1986) 207-222.

- [17] Hassanzadeh, M. S. & Leijnse, A., On the modeling of brine transport in porous media. *Water Resources Research*, 24(3) (1988) 321-330.
- [18] Hassanzadeh, M. S. & Leijnse, A., A non-linear theory of high-concentration-gradient dispersion in porous media. *Adv. Water Resour.*, 18(4) (1995) 203-215.
- [19] Henry, H. R., Interfaces between salt water and fresh water in coastal aquifers. US Geological Survey Water-Supply Paper 1613-C, Sea Water in Coastal Aquifers: C35-C70, 1964.
- [20] Herbert, A. W., Jackson, C. P. & Lever, D. A., Coupled groundwater flow and solute transport with fluid density strongly dependent on concentration. *Water Resources Research*, 24(19) (1988) 1781-1795.
- [21] Holzbecher, E., Numerische Modellierung von Dichteströmungen im porösen Medium. In: Mitteilungen Nr.117, Institut für Wasserbau und Wasserwirtschaft, Technische Universität, Berlin, 1991.
- [22] Horton, C. W. & Rogers, F. T., Convection currents in a porous medium. *J. Appl. Phys.*, 16 (1945) 367- 369.
- [23] Huyakorn, P. S. & Pinder, G. F., Computational methods in subsurface flow. Academic Press, 1983.
- [24] Kolditz, O. & Diersch, H.-J., Quasi-steady-state strategy for numerical simulation of geothermal circulation in hot dry rock fractures. *Int. J. Non-Linear Mechanics*, 28(4) (1993) 467-481.
- [25] Kolditz, O., Benchmarks and examples for numerical groundwater simulations. In: Diersch H.-J., FEFLOW Manual, WASY - Gesellschaft für wasserwirtschaftliche Planung und Systemforschung, Berlin, 1994.
- [26] Kolditz, O., Ratke, R., Zielke, W. & Diersch, H. -J., Coupled physical modelling for the analysis of groundwater systems. In: Notes on Numerical Fluid Mechanics, vol. 51, Vieweg-Verlag, Braunschweig-Wiesbaden, 1995.
- [27] Kolditz, O., Ratke, R., Diersch, H.-J. & Zielke, W. (1998): Coupled groundwater flow and transport: 1. Verification of variable density flow and transport models. *Advances in Water Resources*, 21(1): 27-46.
- [28] Kröhn, K.-P., Simulation von Transportvorgängen im klüftigen Gestein mit der Methode der Finiten Elemente. Bericht Nr. 29/1991, Institut für Strömungsmechanik, Universität Hannover, Dissertationsschrift, 1991.
- [29] Kröhn, K.-P. & Zielke, W., FE-Simulation von Transportvorgängen im klüftigen Gestein. Deutsche Gewässerkundliche Mitteilungen, 35 (3/4) (1991) 82-88.
- [30] Lapwood, E. R., Convection of a fluid in a porous medium. *Proc. Cambridge Phil. Soc.*, 44 (1948) 508-521.
- [31] Leijnse, T. & Hassanzadeh, S. M., Verification of the METROPOL code for density dependent flow in porous media. Report No. 728528002, Rijksinstituut voor Volksgezondheid en Milieuhygiëne RIVM, Bilthoven, 1989.
- [32] Leijnse, T., Three-dimensional modeling of coupled flow and transport in porous media. Ph.D. thesis, University of Notre Dame, Indiana, 1992.
- [33] Lever, D. A. & Jackson, C. P., On the equation for the flow of concentrated salt solution through a porous medium. US Department of Energy, Report No. DOE/RW / 85.100, 1985.

- [34] Mendoza, C. A., VapourT users guide (version 2.11). Waterloo Center for Groundwater Research, University of Waterloo, Canada, 1990.
- [35] Nield, D. A. & Bejan, A., Convection in porous media. Springer-Verlag, 1992.
- [36] Oldenburg, C. M. & Pruess, K., Dispersive transport dynamics in a strongly coupled groundwater-brine flow system. *Water Resources Research*, 31(2) (1995) 289-302.
- [37] Oostrom, M., Hayworth, J. S., Dane, J. H. & Guven, O., Behavior of dense aqueous phase leachate plumes in homogeneous porous media. *Water Resources Research*, 28(10) (1992) 2123-2134.
- [38] Pinder, G. F. & Cooper, H. H., A numerical technique for calculating the transient position of the saltwater front. *Water Resources Research*, 6(3) (1970) 875-882.
- [39] Ratke, R., Zur Lösung der Strömungs- und Transportgleichung bei veränderlicher Dichte. Technischer Bericht, Institut für Strömungsmechanik und Elektronisches Rechnen im Bauwesen, Universität Hannover, 1995.
- [40] Reeves, M., Ward, D. S., Johns, N. D. & Cranwell, R. M., Theory and implementation of SWIFT II, the Sandia waste-isolation flow and transport model for fractured media. Report No. SAND83-1159, Sandia National Lab., Albuquerque, 1986.
- [41] Rivera, A. & Johns, R., Reverification of the HYDROCOIN test case 5 with NAMMU6.2 and comparison with TOUGH2. Presented on 1st GAMM Seminar on Modelling and Computation in Environmental Sciences, Okt. 12-13, Stuttgart, Germany, 1995.
- [42] Schelkes, K. & Vogel, P., Paleohydrological information as an important tool for groundwater modeling of Gorleben site. In: Paleohydrological Methods and their Applications to Radioactive Waste Disposal, Proc. OECD/NEA Workshop, Paris, France, 1992.
- [43] Schincariol, R. A. & Schwartz, F. W., An experimental investigation of variable density flow and mixing in homogeneous and heterogeneous media. *Water Resources Research*, 26(10) (1990) 2317-2329.
- [44] Schincariol, R. A., Schwartz, F. W. & Mendoza, C. A., On the generation of instabilities in variable density flow. *Water Resources Research*, 30(4) (1994) 913-927.
- [45] Segol, G., Pinder, G. F. & Gray, W. G., A Galerkin finite element technique for calculating the transient position of the saltwater front. *Water Resources Research*, 11(2) (1975) 343-347.
- [46] Segol, G., Classic groundwater simulations - Proving and improving numerical models. PTR Prentice Hall, 531pp, 1994.
- [47] Senger, R. K. & Fogg, G. E., Stream functions and equivalent freshwater heads for modeling regional flow of variable-density groundwater: 1. Review of theory and verification. *Water Resources Research*, 26(9) (1990) 2089-2096.
- [48] Swedish Nuclear Power Inspectorate, HYDROCOIN - an international project for studying groundwater hydrology modeling strategies. Level 1 Final Report, Stockholm, 1986.
- [49] van der Molen, W. H. & van Ommen, H. C., Transport of solutes in soils and aquifers. *J. Hydrology*, 100 (1988) 433-451.

- [50] Vogel, P., Zur Theorie binärer Fluidgemische in porösen Medien. Technischer Bericht, Archiv-Nr. 113882, Bundesanstalt für Geowissenschaften und Rohstoffe, Hannover, 1995.
- [51] Voss, C. I. & Souza, W. R., Variable density flow and solute transport simulation of regional aquifers containing a narrow freshwater-saltwater transition zone. *Water Resources Research*, 23(10) (1987) 1851-1866.
- [52] Wooding, R. A., Free convection of fluid in a vertical tube filled with porous material. *J. Fluid Mech.*, 13 (1962) 126-144.

Chapter 15

Multiphase Flow in Deformable Porous Media

15.1 Macroscopic Balance Equations

We consider both cases: multiphase flow in deformable as well as non-deformable (static) porous media. In addition to flow of two fluid phases (compressible and incompressible fluids) we also apply the Richards approximation, which is valid for most cases of infiltration in soils. We assume isothermal conditions. For additional information the reader should refer e.g. to Bear & Bachmat (1990) and Lewis & Schrefler (1998).

15.1.1 Mass Conservation in Static Porous Media

Fluid Phases

We start from the macroscopic mass balance equation for a fluid phase γ .

$$\frac{\partial(nS^\gamma\rho^\gamma)}{\partial t} + \nabla \cdot (nS^\gamma\rho^\gamma\mathbf{v}^\gamma) = Q_{\rho^\gamma} \quad (15.1)$$

Partial differentiation gives

$$\begin{aligned} S^\gamma\rho^\gamma\frac{\partial n}{\partial t} + S^\gamma n\frac{\partial\rho^\gamma}{\partial t} + n\rho^\gamma\frac{\partial S^\gamma}{\partial t} \\ + S^\gamma\rho^\gamma\nabla \cdot (n\mathbf{v}^\gamma) + nS^\gamma\nabla \cdot (\rho^\gamma\mathbf{v}^\gamma) + n\rho^\gamma\nabla \cdot (S^\gamma\mathbf{v}^\gamma) = Q_{\rho^\gamma} \end{aligned} \quad (15.2)$$

which can be rewritten in terms of total derivatives with respect to the gaseous phase.

$$S^\gamma\rho^\gamma\frac{d^\gamma n}{dt} + S^\gamma n\frac{d^\gamma\rho^\gamma}{dt} + n\rho^\gamma\frac{d^\gamma S^\gamma}{dt} = Q_{\rho^\gamma}$$

We offer balance equations in both partial and material derivative notations. The partial derivative notation is used for the Eulerian description of motion, whereas the material derivative notation is necessary for the Lagrangian description.

For non-deformable porous media we assume:

- Porosity does not change: $d^\gamma n/dt = 0$

Then the mass balance equation simplifies to

$$S^\gamma \frac{d^\gamma \rho^\gamma}{dt} + \rho^\gamma \frac{d^\gamma S^\gamma}{dt} = \frac{Q_{\rho^\gamma}}{n}$$

or in terms of partial derivatives

$$S^\gamma \frac{\partial \rho^\gamma}{\partial t} + \rho^\gamma \frac{\partial S^\gamma}{\partial t} + S^\gamma \nabla \cdot (\rho^\gamma \mathbf{v}^\gamma) + \rho^\gamma \nabla \cdot (S^\gamma \mathbf{v}^\gamma) = \frac{Q_{\rho^\gamma}}{n} \quad (15.3)$$

The last two terms on the left hand side can be combined.

$$S^\gamma \frac{\partial \rho^\gamma}{\partial t} + \rho^\gamma \frac{\partial S^\gamma}{\partial t} + \nabla \cdot (S^\gamma \rho^\gamma \mathbf{v}^\gamma) = \frac{Q_{\rho^\gamma}}{n} \quad (15.4)$$

Porous Medium

Based on the general assumption of local thermodynamic equilibrium, the fluid mass balance equation of the porous medium can be obtained by superposing the phase related equations. For the case of two-phase flow we consider a non-wetting phase (gaseous) $\gamma = g$ and a wetting phase (water) $\gamma = w$.

Dividing the equation (15.4) by ρ^g we yield

$$\frac{S^g}{\rho^g} \frac{\partial \rho^g}{\partial t} + \frac{\partial S^g}{\partial t} + \frac{1}{\rho^g} \nabla \cdot (S^g \rho^g \mathbf{v}^g) = \frac{Q_{\rho^g}}{n \rho^g} \quad (15.5)$$

Dividing the equation (15.4) for water by ρ^w and substituting $S^w = 1 - S^g$ in the temporal derivation term we yield

$$\frac{S^w}{\rho^w} \frac{\partial \rho^w}{\partial t} - \frac{\partial S^g}{\partial t} + \frac{1}{\rho^w} \nabla \cdot (S^w \rho^w \mathbf{v}^w) = \frac{Q_{\rho^w}}{n \rho^w} \quad (15.6)$$

Superposing the above equations we obtain the fluid mass balance equation of the porous medium

$$\begin{aligned} \frac{S^g}{\rho^g} \frac{\partial \rho^g}{\partial t} + \frac{S^w}{\rho^w} \frac{\partial \rho^w}{\partial t} + \frac{1}{\rho^g} \nabla \cdot (S^g \rho^g \mathbf{v}^g) + \frac{1}{\rho^w} \nabla \cdot (S^w \rho^w \mathbf{v}^w) \\ = \frac{Q_{\rho^g}}{n \rho^g} + \frac{Q_{\rho^w}}{n \rho^w} \end{aligned} \quad (15.7)$$

Therefore, we have three unknown field functions: gas density ρ^g , water density ρ^w , and gas (or water) saturation S^g .

15.1.2 Mass Conservation in Deformable Porous Media

Now we consider the more general case of deformable porous media. In this case we have to take into account the solid phase as well.

Solid Phase

We start from the macroscopic mass balance equation for the solid phase.

$$\frac{\partial(1-n)\rho^s}{\partial t} + \nabla \cdot ((1-n)\rho^s \mathbf{v}^s) = 0 \quad (15.8)$$

Partial differentiation and extraction of material derivatives with respect to solid movement gives

$$-\rho^s \frac{d^s n}{dt} + (1-n) \left(\frac{d^s \rho^s}{dt} + \rho^s \nabla \cdot \mathbf{v}^s \right) = 0 \quad (15.9)$$

We assume:

- Incompressibility of solid grains: $d\rho^s/dt = 0$

Then we obtain the simplified mass balance equation

$$-\frac{d^s n}{dt} + (1-n) \nabla \cdot \mathbf{v}^s = 0 \quad (15.10)$$

For density variations we yield

$$\frac{d^s n}{dt} = (1-n) \nabla \cdot \mathbf{v}^s \quad (15.11)$$

Fluid Phases

Again we start from the macroscopic mass balance equation for a fluid phase γ .

$$\frac{\partial(nS^\gamma\rho^\gamma)}{\partial t} + \nabla \cdot (nS^\gamma\rho^\gamma \mathbf{v}^\gamma) = Q_{\rho^\gamma} \quad (15.12)$$

Now we consider fluid velocities relative to the deforming solid phase.

$$\mathbf{v}^\gamma = (\mathbf{v}^\gamma - \mathbf{v}^s) + \mathbf{v}^s \quad (15.13)$$

Then the mass balance equation for a fluid phase γ can be written as

$$\frac{\partial(nS^\gamma\rho^\gamma)}{\partial t} + \nabla \cdot (nS^\gamma\rho^\gamma(\mathbf{v}^\gamma - \mathbf{v}^s)) + \nabla \cdot (nS^\gamma\rho^\gamma \mathbf{v}^s) = Q_{\rho^\gamma} \quad (15.14)$$

Partial differentiation and extraction of material derivatives with respect to solid movement gives

$$\begin{aligned} S^\gamma \rho^\gamma \frac{d^s n}{dt} + n S^\gamma \frac{d^s \rho^\gamma}{dt} + n \rho^\gamma \frac{d^s S^\gamma}{dt} + \nabla \cdot (n S^\gamma \rho^\gamma (\mathbf{v}^\gamma - \mathbf{v}^s)) + n S^\gamma \rho^\gamma \nabla \cdot \mathbf{v}^s \\ = Q_{\rho^\gamma} \end{aligned} \quad (15.15)$$

Using eqn (15.11) for substitution of porosity changes and splitting up the total derivatives again we obtain

$$\begin{aligned} S^\gamma \rho^\gamma \nabla \cdot \mathbf{v}^s + n S^\gamma \left(\frac{\partial \rho^\gamma}{\partial t} + \mathbf{v}^s \cdot \nabla \rho^\gamma \right) + n \rho^\gamma \left(\frac{\partial S^\gamma}{\partial t} + \mathbf{v}^s \cdot \nabla S^\gamma \right) + \\ + \nabla \cdot (n S^\gamma \rho^\gamma (\mathbf{v}^\gamma - \mathbf{v}^s)) = Q_{\rho^\gamma} \end{aligned} \quad (15.16)$$

Rearranging the terms we yield the following mass balance equation for a fluid phase γ in a deformable porous medium.

$$\begin{aligned} n S^\gamma \frac{\partial \rho^\gamma}{\partial t} + n \rho^\gamma \frac{\partial S^\gamma}{\partial t} + \nabla \cdot (n S^\gamma \rho^\gamma (\mathbf{v}^\gamma - \mathbf{v}^s)) + S^\gamma \rho^\gamma \nabla \cdot \mathbf{v}^s + \\ + n \mathbf{v}^s \cdot (S^\gamma \nabla \rho^\gamma + \rho^\gamma \nabla S^\gamma) = Q_{\rho^\gamma} \end{aligned} \quad (15.17)$$

Porous Medium

Based on the general assumption of local thermodynamic equilibrium, the mass balance equation of the porous medium can be obtained by superposing the phase related equations, equations (15.11),(15.17).

For two-phase flow of a non-wetting (gaseous) and a wetting (water) phase in a deformable porous medium we have

$$\begin{aligned} (S^g \rho^g + S^w \rho^w) \nabla \cdot \mathbf{v}^s + n \left(S^g \frac{\partial \rho^g}{\partial t} + S^w \frac{\partial \rho^w}{\partial t} \right) + n \left(\rho^g \frac{\partial S^g}{\partial t} + \rho^w \frac{\partial S^w}{\partial t} \right) + \\ + \nabla \cdot (n S^g \rho^g (\mathbf{v}^g - \mathbf{v}^s)) + \nabla \cdot (n S^w \rho^w (\mathbf{v}^w - \mathbf{v}^s)) + \\ + n \mathbf{v}^s \cdot (S^g \nabla \rho^g + \rho^g \nabla S^g + S^w \nabla \rho^w + \rho^w \nabla S^w) \\ = Q_{\rho^g} + Q_{\rho^w} \end{aligned} \quad (15.18)$$

15.1.3 Momentum Conservation

General phase momentum conservation is described in section 3.3.2. For linear momentum conservation in porous media we assume, in general, that inertial forces can be neglected (i.e. $d\mathbf{v}/dt \approx 0$) and body forces are gravity at all.

Solid Phase

Taking the above general assumptions into account we derive the following balance equation of solid phase linear momentum.

$$(1-n)\rho^s \mathbf{g} + \nabla \cdot ((1-n)\boldsymbol{\sigma}^s) = 0 \quad (15.19)$$

Fluid Phases

Taking the above general assumptions into account and assuming furthermore that internal fluid friction is small in comparison to friction on the fluid-solid interface and that turbulence effects can be neglected we obtain the Darcy law for each fluid phase γ in multiphase flow.

$$n(\mathbf{v}^\gamma - \mathbf{v}^s) = -\frac{k_{\text{rel}}^\gamma \mathbf{k}}{\mu^\gamma} (\nabla p^\gamma - \rho^\gamma \mathbf{g}) \quad (15.20)$$

Porous Medium

The equilibrium condition for the porous medium is obtained by summing the phase-related equations.

$$\rho \mathbf{g} + \nabla \cdot \boldsymbol{\sigma} = 0 \quad (15.21)$$

with density of porous medium:

$$\rho = (1-n)\rho^s + nS^w\rho^w + nS^g\rho^g \quad (15.22)$$

and with total stress in porous medium:

$$\boldsymbol{\sigma} = (1-n)\boldsymbol{\sigma}^s - n(S^w p^w + S^g p^g) \mathbf{I} \quad (15.23)$$

15.2 Constitutive Relationships

15.2.1 Saturation

Saturation of a fluid phase γ is defined as the volumetric fraction ϵ^γ related to the sum of all fluid phases volumetric fractions.

$$S^\gamma = \frac{\epsilon^\gamma}{\sum_{(\gamma)} \epsilon^\gamma} \quad (15.24)$$

The sum of saturations of all fluid phases must be equal to unity.

$$\sum_{(\gamma)} S^\gamma = 1 \quad (15.25)$$

Effective saturation is defined as (e.g. Brooks & Corey (1964))

$$S_{\text{eff}}^{\gamma} = \frac{S^{\gamma} - S_r^{\gamma}}{1 - S_r^{\gamma}} \quad (15.26)$$

Moisture content is defined as the product of porosity and saturation.

$$\theta^{\gamma} = n S^{\gamma} \quad (15.27)$$

Applying the chain rule, we can express saturation changes in following way.

$$dS^{\gamma} = \frac{dS^{\gamma}}{dp^{\gamma}} dp^{\gamma} \quad (15.28)$$

15.2.2 Density

Solids

From the mass conservation equation $d^s(\rho^s V^s) = 0$ the following constitutive equation for solid density can be derived.

$$\frac{1}{\rho^s} \frac{d^s \rho^s}{dt} = \frac{1}{1-n} \left(\frac{\alpha - n}{K_s} \frac{d^s p^s}{dt} - (1-\alpha) \nabla \cdot \mathbf{v}^s \right) \quad (15.29)$$

A substantial assumption we use later is incompressibility of solid grain, which does not mean that the porous medium is not compressible, i.e. porosity can change due to deformation.

$$\frac{d^s \rho^s}{dt} = 0 \quad (15.30)$$

Compressible Fluids

Equation of state for an ideal gas is

$$\rho^g = \frac{M}{RT} p^g \quad (15.31)$$

Incompressible Fluids

Equation of state for water is

$$\rho^w = \rho_0^w (1 - \beta_T^w T + \beta_p (p^w - p_0^w)) \quad (15.32)$$

$$\beta_p = \frac{1}{\rho^w} \frac{d\rho^w}{dp^w} \quad (15.33)$$

Porous Medium

Density of the multiphase medium is defined as sum of all partial phase densities.

$$\rho = (1 - n)\rho^s + nS^w\rho^w + nS^g\rho^g \quad (15.34)$$

15.2.3 Capillary Pressure

As a consequence of interfacial tension a discontinuity in fluid pressure exists across the interface that separates two immiscible fluids. The partial pressure difference between two phases is denoted as capillary pressure, which is a function of saturation.

$$p_c^{\alpha\beta} = p^\beta - p^\alpha = f(S^\alpha) \quad (15.35)$$

In general, capillary pressure is the difference between partial pressures of non-wetting and wetting phases.

$$p_c = p^{nw} - p^w = f(S^w) \quad (15.36)$$

Capillary pressure is always positive: $p_c > 0, \forall S$. It is often assumed that air is at a constant atmospheric pressure taken as zero $p^g = 0$. This means, the macroscopic pressure of water in the unsaturated zone is always negative due to suction.

Capillary pressure must be measured for given soils and pairs of fluids. In general, these experiments are conducted for equilibrium conditions with no fluid in motion. Various authors have proposed analytical functions for capillary pressure - saturation - relationships. In the following we list the implemented models (see Rockflow manual for details).

1 - User-defined function

$$p_c = f(S_i^g) = f(1 - S_i^w) \quad (15.37)$$

2 - Linear Function

$$p_c = \begin{cases} p_{c\max} & S^w < S_r^w \\ (1 - S_{\text{eff}}^w)p_{c\max} & S_r^w < S^w < S_{\max}^w \\ 0 & S^w > S_{\max}^w \end{cases} \quad (15.38)$$

3 - Potential Function

$$p_c = a(S_{\text{eff}}^w)^b \quad (15.39)$$

4 - van Genuchten Model (1980)

$$S_{\text{eff}} = \frac{S^w - S_r^w}{1 - S_r^w} = (1 + (\alpha p_c)^n)^m \quad , \quad p_c > 0 \quad (15.40)$$

$$p_c = \begin{cases} 0 & S^w > S_{\max}^w \\ \frac{\rho^w g}{\alpha} (S_{\text{eff}}^{-1/m} - 1)^{1/n} & S_r^w < S^w < S_{\max}^w \\ p_{c\max} & S^w < S_r^w \end{cases} \quad (15.41)$$

5 - Haverkamp et al. (1977)

The formulas are given in terms of pressure head $h = p^w/g\rho^w$ and moisture content $\theta = nS^w$.

$$\theta = \frac{\alpha(\theta_s - \theta_r)}{\alpha + |h|^\beta} + \theta_r \quad (15.42)$$

$$h = \left(-\frac{\alpha}{\theta} (\theta - \theta_s + \theta_r) \right)^{1/\beta} \quad (15.43)$$

Table 15.1: Model parameter

θ	volumetric water (moisture) content	$[cm^3/cm^3]$
θ_r	residual volumetric water content	$[cm^3/cm^3]$
θ_s	saturated volumetric water content	$[cm^3/cm^3]$
$h(\theta)$	soil water pressure head relative to the atmosphere	$[cm]$
α		1.611×10^6
β		$[Pa^{-1}]$

6 - Brooks & Corey Model (1964)

$$S_{\text{eff}} = \frac{S^w - S_r^w}{1 - S_r^w} = \left(\frac{p_b}{p_c} \right)^\lambda \quad , \quad p_c \geq p_b \quad (15.44)$$

$$p_c = \begin{cases} 0 & S^w > S_{\max}^w \\ p_b \left(\frac{1}{S_{\text{eff}}} \right)^{1/\lambda} & S_r^w < S^w < S_{\max}^w \\ p_{c\max} & S^w < S_r^w \end{cases} \quad (15.45)$$

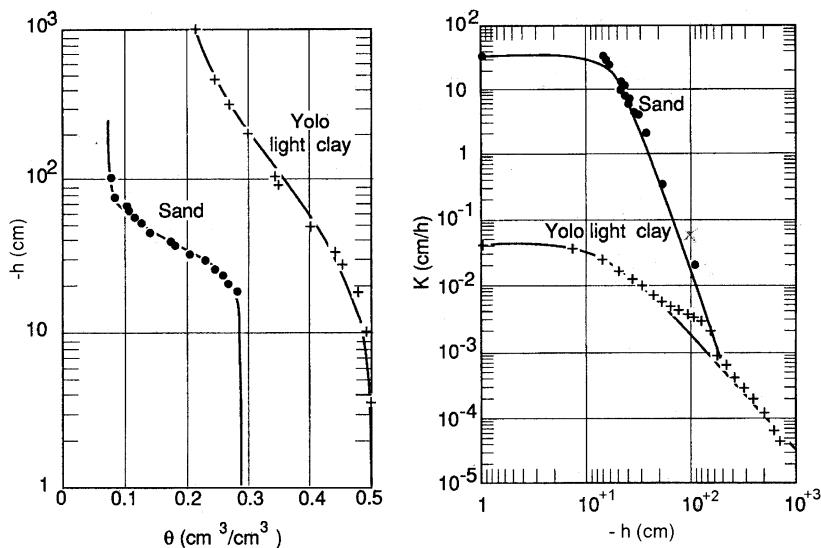


Figure 15.1: Hydraulic properties of unsaturated soil (Segol 1995)

p_b is the so-called bubbling pressure, the minimum pressure at which the gaseous phase exists, λ is the pore size distribution index.

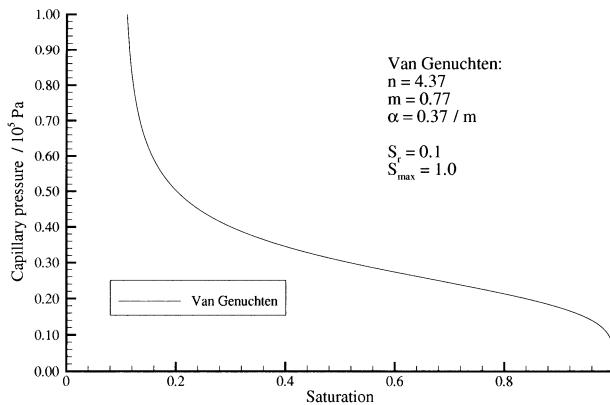


Figure 15.2: Capillary pressure / saturation relationship

The capillary pressure/saturation relationships $p_c(S)$ differ for drainage and rewetting (imbibition) (Mualem 1974, 1978). This phenomenon is called hys-

teresis. Reasons for capillary pressure hysteresis are: (i) varying pore shape (ink-bottle effect), (ii) contact angle hysteresis (raindrop effect), (iii) entrainment of non-wetting fluids, (iv) swelling and shrinking of solid grains.

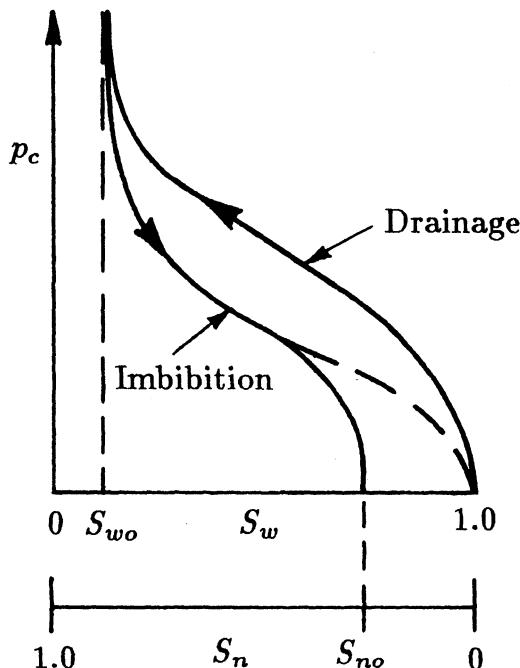


Figure 15.3: Capillary hysteresis, Source: Bear & Bachmat 1990, with $S_w = S^w$, $S_{wo} = S_r^w$, $S_n = S^g$, $S_{no} = S_r^g$

Table 15.2: Rockflow models

$p_c(S)$ model	cp[0]	cp[1]	cp[2]	cp[3]	cp[4]	cp[5]
0 No capillary pressure $p_c = 0$						
1 User-defined curve		Curve				
2 Linear function		S_r^w	S_{\max}^w	$p_{c\max}$		
3 Parabolic function		S_r^w	S_{\max}^w	a	b	
4 van Genuchten (1980)		S_r^w	S_{\max}^w	α	m	n
5 Haverkamp et al. (1977)		S_r^w	S_{\max}^w	A	b	c
6 Brooks & Corey (1966)		S_r^w	p_b	λ		

Rockflow functions:

```
void CECalcCap_AMM(long element,double satu,double *kap12)
double CECalcSatuFromCap_AMM(long element,double kap12)
```

15.2.4 Relative Permeability

To introduce the concept of relative permeability we recall the Darcy law for flow of multiple fluid phases through porous media.

$$nS^\gamma \mathbf{v}^\gamma = -\frac{k_{\text{rel}}^\gamma(S^\gamma)\mathbf{k}}{\mu^\gamma}(\nabla p^\gamma - \rho^\gamma \mathbf{g}) \quad (15.46)$$

Effective permeability of a porous medium $k_{\text{rel}}^\gamma(S^\gamma)\mathbf{k}$ containing more than one fluid depends on a large extent on saturations.

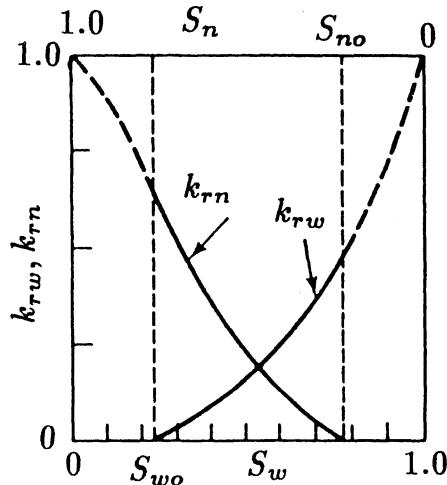


Figure 15.4: Relative permeability functions, Source: Bear & Bachmat 1990, with $S_w = S^w$, $S_{wo} = S_r^w$, $S_n = S^g$, $S_{n0} = S_r^g$, $k_{rn} = k_{\text{rel}}^g$, $k_{rw} = k_{\text{rel}}^w$

Numerous relationships between relative permeability and saturation are given in the literature. We list some of the most common which are implemented in the Rockflow code.

0 - Perfect mobile phases

$$k_{\text{rel}}^g = 1 \quad , \quad k_{\text{rel}}^w = 1 \quad (15.47)$$

1 - User-defined function

$$k_{\text{rel}}^g = f(S_i^g) \quad , \quad k_{\text{rel}}^w = f(S_i^w) \quad (15.48)$$

2 - Linear function

$$k_{\text{rel}}^g = \begin{cases} 0 & S^g < S_1^g \\ \frac{S^g - S_1^g}{S_2^g - S_1^g} & S_1^g < S^g < S_2^g \\ 1 & S^g > S_2^g \end{cases} \quad (15.49)$$

$$k_{\text{rel}}^w = \begin{cases} 0 & S^w < S_1^w \\ \frac{S^w - S_1^w}{S_2^w - S_1^w} & S_1^w < S^w < S_2^w \\ 1 & S^w > S_2^w \end{cases} \quad (15.50)$$

3 - Potential function

$$k_{\text{rel}}^g = \begin{cases} 0 & S^g < S_1^g \\ \left(\frac{S^g - S_1^g}{S_2^g - S_1^g} \right)^a & S_1^g < S^g < S_2^g \\ 1 & S^g > S_2^g \end{cases} \quad (15.51)$$

$$k_{\text{rel}}^w = \begin{cases} 0 & S^w < S_1^w \\ \left(\frac{S^w - S_1^w}{S_2^w - S_1^w} \right)^a & S_1^w < S^w < S_2^w \\ 1 & S^w > S_2^w \end{cases} \quad (15.52)$$

Such a function was introduced e.g. by Huyakorn & Pinder (1983).

$$k_{\text{rel}}^w = \frac{(S^w - 0.2)^2}{0.36} \quad (15.53)$$

4 - van Genuchten Model (1980)

$$k_{\text{rel}}(h) = \frac{1 - (\alpha h)^{n-2} [1 + (\alpha h)^n]^{-m}}{[1 + (\alpha h)^n]^{2m}} \quad (15.54)$$

5 - Haverkamp et al. (1977)

$$k_{\text{rel}}(h) = K_s \frac{A}{A + |h|^\beta} \quad (15.55)$$

$$h = \left(-\frac{\alpha}{\theta} (\theta - \theta_s + \theta_r) \right)^{1/\beta} \quad (15.56)$$

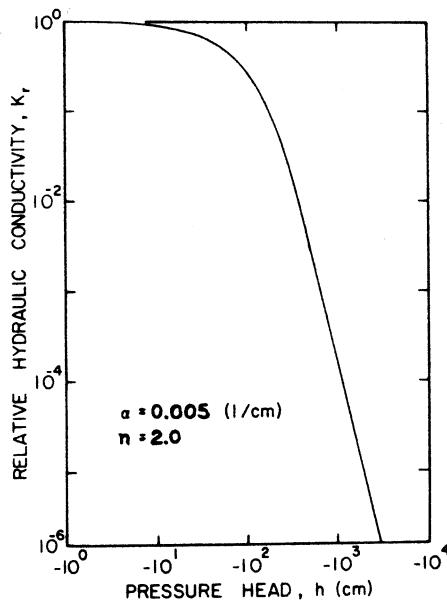


Figure 15.5: Relative permeability functions (Source: van Genuchten Model (1980))

6 - Brooks & Corey (1966)

$$S_{\text{eff}} = \frac{S^w - S_r^w}{S_{\max}^w - S_r^w} \quad (15.57)$$

$$k_{\text{rel}}^w = S_{\text{eff}}^4 \quad (15.58)$$

Rockflow functions:

```
void CECalcRelPerm_AMM(long element, int phasen, double *satu, double
*relperm)
```

15.2.5 Total and Effective Stress

Due to the fundamental symmetry assumption there are 6 instead of 9 components of stress and strain tensors. Using vector notation stress and strain tensors can be written in following form.

$$\boldsymbol{\sigma} = [\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{31}]^T \quad (15.59)$$

$$\boldsymbol{\varepsilon} = [\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{33}, 2\varepsilon_{12}, 2\varepsilon_{23}, 2\varepsilon_{31}]^T \quad (15.60)$$

Table 15.3: Rockflow models

$k_{\text{rel}}(S)$ model	rp[1]	rp[2]	rp[3]	rp[4]	rp[5]	rp[6]
rp[0] Perfectly mobile phases						
1 User-defined curve	1st	2nd	...			
2 Linear function	S_1^g	S_2^g		S_1^w	S_2^w	
3 Parabolic function	S_1^g	S_2^g	a	S_1^w	S_2^w	b
4 van Genuchten (1980)	S_r^w	S_{\max}^w	α	m	n	
5 Haverkamp et al. (1977)	S_r^w	S_{\max}^w	α	A	β	
6 Brooks & Corey (1966)	S_r^w	S_{\max}^w				

Introducing the symmetric gradient operator

$$\mathbf{L} = \begin{bmatrix} \partial/\partial x & 0 & 0 \\ 0 & \partial/\partial y & 0 \\ 0 & 0 & \partial/\partial z \\ \partial/\partial y & \partial/\partial x & 0 \\ 0 & \partial/\partial z & \partial/\partial y \\ \partial/\partial z & 0 & \partial/\partial x \end{bmatrix} \quad (15.61)$$

and the specific unit tensor

$$\mathbf{m}^T = [1, 1, 1, 0, 0, 0] \quad (15.62)$$

we can rewrite the balance equation of linear momentum for the porous medium (equilibrium equation) as follows

$$\mathbf{L}^T \boldsymbol{\sigma} + \rho \mathbf{g} = \mathbf{0} \quad (15.63)$$

Divergence of strain velocities can be expressed by displacements as follows.

$$\nabla \cdot \mathbf{v}^s = \mathbf{m}^T \frac{\partial \boldsymbol{\varepsilon}}{\partial t} = \mathbf{m}^T \mathbf{L} \frac{\partial \mathbf{u}}{\partial t} \quad (15.64)$$

Total stress in the porous medium is defined as the sum of all phase-related partial stresses.

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^s + \boldsymbol{\sigma}^w + \boldsymbol{\sigma}^g \quad (15.65)$$

Solid stress is by definition

$$\boldsymbol{\sigma}^s = (1-n)(\boldsymbol{\sigma}_e^s - p^s \mathbf{I}) = \underbrace{(1-n)\boldsymbol{\sigma}_e^s}_{\boldsymbol{\sigma}'} - (1-n)(S^w p^w + S^g p^g) \mathbf{I} \quad (15.66)$$

Effective stress is, therefore, defined as

$$\boldsymbol{\sigma}' = (1 - n)\boldsymbol{\sigma}_e^s + p^s \mathbf{I} = (1 - n)\boldsymbol{\sigma}_e^s + (S^g p^g + S^w p^w) \mathbf{I} \quad (15.67)$$

Total stress now can be written as

$$\begin{aligned} \boldsymbol{\sigma} &= \boldsymbol{\sigma}' - (1 - n)(S^w p^w + S^g p^g) \mathbf{I} - n S^w p^w \mathbf{I} - n S^g p^g \mathbf{I} \\ &= \boldsymbol{\sigma}' - (S^w p^w + S^g p^g) \mathbf{I} \end{aligned} \quad (15.68)$$

Because effective stress is responsible for deformation process in the solid matrix (accept of grain compressibility), stress-strain relationships for porous media are established in term of effective stresses.

$$d\boldsymbol{\sigma}' = \mathbf{D}^s d\boldsymbol{\varepsilon} \quad (15.69)$$

with the strain rate tensor \mathbf{D}^s .

$$\boldsymbol{\sigma}' = \lambda \text{tr} \boldsymbol{\varepsilon}^{el} \mathbf{I} + 2\mu \boldsymbol{\varepsilon}^{el}, \quad \boldsymbol{\varepsilon}^{el} = \boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^{in} \quad (15.70)$$

Strain and displacement are related by the following relationship with denoted matrix operations.

$$\underbrace{\boldsymbol{\varepsilon}}_{[6 \times 1]} = \underbrace{\mathbf{L}}_{[6 \times 3]} \underbrace{\mathbf{u}}_{[3 \times 1]} \quad (15.71)$$

For the stress-strain relation the generalized isotropic Hooke's law is used.

$$\boldsymbol{\sigma}' = \mathbf{C} \boldsymbol{\varepsilon} = \mathbf{C} \mathbf{L} \mathbf{u} \quad (15.72)$$

with

$$\mathbf{C} = \begin{bmatrix} \lambda + 2G & \lambda & \lambda \\ \lambda & \lambda + 2G & \lambda \\ \lambda & \lambda & \lambda + 2G \end{bmatrix} \begin{matrix} G \\ G \\ G \end{matrix} \quad (15.73)$$

15.3 Governing Equations

The followings steps are conducted to derive the governing equations for multi-phase flow in deformable porous media:

- Applying the constitutive relationships (sec 15.2) to the balance equations (sec 15.1),

- Combining mass and momentum balance equations for fluid flow,
- Introducing physically based simplifications

The finally derived governing equations are the basis for the discretization process introduced the next section. These formulas are specially framed.

We consider three cases:

1. Static Porous Medium - Two Phase Flow (sec 15.3.1),
2. Deformable Porous Medium - Richards Approximation (sec 15.3.2),
3. Deformable Porous Medium - Two Phase Flow (sec 15.3.3).

15.3.1 Static Porous Medium - Two Phase Flow

At first we consider a simplified case: two phase flow in a static porous medium.

Mass Balance of Porous Medium

The set of governing equations for two phase flow in a static porous medium consists of the fluid mass balance equation of the porous medium (15.7) for determination of gas pressure p^g and the fluid phase mass balance (15.4) for determination of water saturation S^w .

We apply the following constitutive relationships to the fluid mass balance equation of the porous medium .

- Equations of state for gas and water density, eqn (15.31) and eqn (15.33)
- Definition of capillary pressure: $p^w = p^g - p_c$, eqn (15.36)
- Extended Darcy law for phase-related advective fluid fluxes, eqn (15.46)

Pressure of the compressible fluid phase p^g

$$\begin{aligned} & \left(\frac{S^g}{\rho^g} \frac{M}{RT} - S^w \beta_p^w \right) \frac{\partial p^g}{\partial t} + S^w \beta_p^w \frac{\partial p_c}{\partial t} \\ & - \frac{1}{\rho^g} \nabla \cdot \left(\rho^g \frac{k_{\text{rel}}^g \mathbf{k}}{\mu^g} (\nabla p^g - \rho^g \mathbf{g}) \right) - \frac{1}{\rho^w} \nabla \cdot \left(\rho^w \frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} (\nabla (p_c - p^g) - \rho^w \mathbf{g}) \right) \\ & = \frac{Q_{\rho^g}}{n \rho^g} + \frac{Q_{\rho^w}}{n \rho^w} \end{aligned} \quad (15.74)$$

We apply the following constitutive relationships and assumptions to the fluid phase mass balance equation.

- Temporal changes of water density can be neglected.
- Extended Darcy law for phase-related advective fluid fluxes, eqn (15.46)

Saturation of the incompressible fluid phase S^w

$$\rho^w \frac{\partial S^w}{\partial t} - \nabla \cdot \left(\rho^w \frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} (\nabla p^w - \rho^w \mathbf{g}) \right) = \frac{Q_{\rho^w}}{n} \quad (15.75)$$

Constitutive relationships

$$\begin{aligned} p_c &= f(S^w) \\ k_{\text{rel}}^w &= f(S^w) \\ k_{\text{rel}}^g &= f(1 - S^w) \end{aligned} \quad (15.76)$$

15.3.2 Deformable Porous Medium - Richards Approximation

Now we consider deformation of the porous medium, but we simplify the multiphase flow problem employing the Richards approximation. According to the Richards approximation we assume that air pressure is constant everywhere, $p^g = 0$. Then we have to consider only the continuity equation for the incompressible fluid phase (e.g. water) due to equation (15.17).

Mass Balance of Porous Medium

We employ the following constitutive relationships and assumptions to the mass balance equation of the incompressible fluid phase (15.17).

- Eliminating $d^s n / dt$ using the solid phase mass conservation, eqn (15.9),
- Equation of state for water density, eqn (15.33)
- Extended Darcy law for phase-related advective fluid fluxes, eqn (15.46)
- Divergence of strain velocities in terms of displacements, eqn (15.64)

Governing equation for determination of water pressure p^w .

$$\begin{aligned} n \rho^w \left(\frac{\partial S^w}{\partial p^w} + S^w \beta_p \right) \frac{\partial p^w}{\partial t} + S^w \rho^w \mathbf{m}^T \mathbf{L} \frac{\partial \mathbf{u}}{\partial t} \\ - \nabla \cdot \left(\rho^w \frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} (\nabla p^w - \rho^w \mathbf{g}) \right) = Q_{\rho^w} \end{aligned} \quad (15.77)$$

Frequently, the Boussinesq approximation is employed. In this case, spatial density variations are neglected in the fluid mass balance equation. Then the equation of fluid mass conservation can be divided by ρ^w and reduces to

$$n \left(\frac{\partial S^w}{\partial p^w} + S^w \beta_p \right) \frac{\partial p^w}{\partial t} + S^w \mathbf{m}^T \mathbf{L} \frac{\partial \mathbf{u}}{\partial t} - \nabla \cdot \left(\frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} (\nabla p^w - \rho^w \mathbf{g}) \right) = \frac{Q_{\rho^w}}{\rho^w} \quad (15.78)$$

Constitutive relationships

$$\begin{aligned} S^w &= S^w(p_c) = S^w(p^w) \\ k_{\text{rel}}^w &= k_{\text{rel}}^w(S^w) \end{aligned} \quad (15.79)$$

Momentum Balance of Porous Medium

We employ the following constitutive relationships and assumptions to the momentum balance equation of the porous medium (15.21)/(15.63).

- Density of the porous medium, equation (15.34)
- Total stress tensor equation (15.68)
- Definition of strain tensor in terms of displacement equation (15.71)
- Hooke's law for elastic deformation equation (15.72)

Governing equation for determination of solid displacements \mathbf{u}

$$((1-n)\rho^s + S^w n \rho^w) \mathbf{g} + \nabla \cdot (\mathbf{C} \mathbf{L} \mathbf{u} - S^w p^w \mathbf{m}) = 0 \quad (15.80)$$

15.3.3 Deformable Porous Medium - Two Phase Flow

Now we consider the complex case of multiphase flow in a deformable porous medium. There are 3 unknown field functions to determine: gas pressure p^g , water saturation S^w , solid displacements \mathbf{u} .

Mass Balance of Porous Medium

We employ the following constitutive relationships and assumptions to the mass balance equation of the porous medium (15.18).

- Equations of state for gas and water density, eqn (15.33), eqn (15.31)
- Extended Darcy law for phase-related advective fluid fluxes, eqn (15.46)
- Definition of capillary pressure: $p^w = p^g - p_c$, equation (15.36)
- Auxiliary relationship for saturations: $S^g = 1 - S^w$
- Expressing strain velocities by displacements as primal variables, eqn (15.64)

Governing equation for determination of gas pressure p^g .

$$\begin{aligned} & n \left(S^g \frac{M}{RT} + S^w \rho^w \beta_p \right) \frac{\partial p^g}{\partial t} - n S^w \rho^w \beta_p \frac{\partial p_c}{\partial t} + n(\rho^g - \rho^w) \frac{\partial S^g}{\partial t} \\ & + (S^g \rho^g + S^w \rho^w) \mathbf{m}^T \mathbf{L} \frac{\partial \mathbf{u}}{\partial t} \\ & - \nabla \cdot \mathbf{k} \left((\rho^g \frac{k_{\text{rel}}^g}{\mu^g} + \rho^w \frac{k_{\text{rel}}^w}{\mu^w}) \nabla p^g - ((\rho^g)^2 \frac{k_{\text{rel}}^g}{\mu^g} + (\rho^w)^2 \frac{k_{\text{rel}}^w}{\mu^w}) \mathbf{g} + \rho^w \frac{k_{\text{rel}}^w}{\mu^w} \nabla p_c \right) \\ & = Q_{\rho^g} + Q_{\rho^w} \end{aligned} \quad (15.81)$$

We apply the following constitutive relationships and assumptions to the fluid phase mass balance equation (15.17).

- Temporal changes of water density can be neglected.
- Extended Darcy law for phase-related advective fluid fluxes, eqn (15.46)

Saturation of the incompressible fluid phase S^w

$$n \rho^w \frac{\partial S^w}{\partial t} - \nabla \cdot \left(\rho^w \frac{k_{\text{rel}}^w}{\mu^w} (\nabla p^w - \rho^w \mathbf{g}) \right) + S^w \rho^w \mathbf{m}^T \mathbf{L} \frac{\partial \mathbf{u}}{\partial t} = Q_{\rho^w} \quad (15.82)$$

Constitutive relationships

$$\begin{aligned} p_c &= f(S^w) \\ k_{\text{rel}}^w &= f(S^w) \\ k_{\text{rel}}^g &= f(1 - S^w) \end{aligned} \quad (15.83)$$

Momentum Balance of Porous Medium

We conduct the same procedure as in the previous section to the momentum balance equation of the porous medium (15.21)/(15.63) taking into account gas pressure.

Governing equation for determination of solid displacements \mathbf{u}

$$((1-n)\rho^s + nS^g\rho^g + nS^w\rho^w)\mathbf{g} + \nabla \cdot (\mathbf{C}\mathbf{L}\mathbf{u} - S^g p^g \mathbf{m} - S^w p^w \mathbf{m}) = 0 \quad (15.84)$$

15.4 Finite Element Formulations

In this section we focus on the finite element formulations for flow of two immiscible fluids in deformable porous media. At first we consider a simplified case for the flow problem using the Richards approximation.

15.4.1 Mass Balance of Porous Medium

The discretization process for determination of fluid pressure of incompressible phase p^w is based on governing equation (15.78).

$$\begin{aligned} n \left(\frac{\partial S^w}{\partial p^w} + S^w \beta_p \right) \frac{\partial p^w}{\partial t} + S^w \mathbf{m}^T \mathbf{L} \frac{\partial \mathbf{u}}{\partial t} \\ - \nabla \cdot \left(\frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} (\nabla p^w - \rho^w \mathbf{g}) \right) = Q_{\rho^w} \end{aligned} \quad (15.85)$$

Method of weighted residuals provides us the weak formulation of the multiphase flow PDE.

$$\begin{aligned} \int_{\Omega} \omega \left(n \left(\frac{\partial S^w}{\partial p^w} + S^w \beta_p \right) \frac{\partial p^w}{\partial t} + S^w \mathbf{m}^T \mathbf{L} \frac{\partial \mathbf{u}}{\partial t} \right) d\Omega \\ - \int_{\Omega} \omega \left(\nabla \cdot \left(\frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} (\nabla p^w - \rho^w \mathbf{g}) \right) \right) d\Omega = \int_{\Omega} \omega \frac{Q_{\rho^w}}{\rho^w} d\Omega \end{aligned} \quad (15.86)$$

In order to eliminate second-order derivatives we use partial differentiation and Gauss-Green integral theorem.

$$\begin{aligned} \int_{\Omega} \omega \left(n \left(\frac{\partial S^w}{\partial p^w} + S^w \beta_p \right) \frac{\partial p^w}{\partial t} + S^w \mathbf{m}^T \mathbf{L} \frac{\partial \mathbf{u}}{\partial t} \right) d\Omega \\ + \int_{\Omega} \nabla \omega \cdot \left(\frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} (\nabla p^w - \rho^w \mathbf{g}) \right) d\Omega = \int_{\Omega} \omega \frac{Q_{\rho^w}}{\rho^w} d\Omega - \int_{\Gamma} \omega q_n^w d\Gamma \end{aligned} \quad (15.87)$$

with the fluid mass normal flux term

$$q_n^w = - \frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} (\nabla p^w - \rho^w \mathbf{g}) \cdot \mathbf{n} \quad (15.88)$$

All unknown functions are approximated by following trial solutions based on nodal values and interpolation functions.

$$\mathbf{p}^w = \sum_{i=1}^{np} \phi_i \hat{\mathbf{p}}_i^w \quad , \quad S^w = \sum_{i=1}^{np} \phi_i \hat{\mathbf{S}}_i^w \quad , \quad \mathbf{u} = \sum_{i=1}^{np} \phi_i \hat{\mathbf{u}}_i \quad (15.89)$$

Now the spatial discretization process is completed and we derive the following system of algebraic equations.

$$\mathbf{C} \frac{d\hat{\mathbf{p}}^w}{dt} + \mathbf{D} \frac{d\hat{\mathbf{u}}}{dt} + \mathbf{K} \hat{\mathbf{p}}^w = \mathbf{f} + \mathbf{g} + \mathbf{s} \quad (15.90)$$

with:

$$\begin{aligned} \mathbf{C} &= \int_{\Omega} \omega_p \left(n \frac{\partial S^w}{\partial p^w} + n S^w \beta_p \right) \phi_p d\Omega && \text{Fluid capacitance matrix} \\ \mathbf{D} &= \int_{\Omega} \omega_p (S^w \mathbf{m}^T \mathbf{L}) \phi_u d\Omega && \text{Solid capacitance matrix} \\ &&& (\text{Dilatation coupling matrix}) \\ \mathbf{K} &= \int_{\Omega} \nabla \omega_p \left(\frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} \right) \nabla \phi_p d\Omega && \text{Conductance matrix} \end{aligned}$$

$$\begin{aligned} \mathbf{g} &= \int_{\Omega} \nabla \omega_p \left(\frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} \rho^w \mathbf{g} \right) d\Omega && \text{Gravitation matrix} \\ \mathbf{s} &= \int_{\Omega} \omega_p \frac{Q_{\rho^w}}{\rho^w} d\Omega && \text{Fluid source term} \\ \mathbf{r} &= - \int_{\Gamma} \omega_p q_n^w d\Gamma && \text{Boundary fluid flux term} \end{aligned}$$

Constitutive relationships coupling the equations

$$S^w(p^w) \quad , \quad k_{\text{rel}}^w(S^w) \quad (15.91)$$

Time discretization using first-order finite difference schemes yields the following scheme.

$$\begin{aligned} &\left[\frac{\mathbf{C}^{n+1}}{\Delta t} + \theta \mathbf{K}^{n+1} \right] \hat{\mathbf{p}}^{n+1} + \frac{\mathbf{D}^{n+1}}{\Delta t} \hat{\mathbf{u}}^{n+1} \\ &= \left[\frac{\mathbf{C}^n}{\Delta t} - (1-\theta) \mathbf{K}^n \right] \hat{\mathbf{p}}^n + \frac{\mathbf{D}^n}{\Delta t} \hat{\mathbf{u}}^n + \mathbf{f} + \mathbf{g} + \mathbf{s} \end{aligned} \quad (15.92)$$

As a result of domain decomposition into finite elements and interpolation restricted on elements using shape functions, N_p and N_u , we obtain finite element matrices as follows.

$$\mathbf{C} = \bigcup_e \mathbf{C}^e \quad (15.93)$$

$\mathbf{C}^e = \int_{\Omega} \mathbf{N}_p \left(n \frac{\partial S^w}{\partial p^w} + n S^w \beta_p \right) \mathbf{N}_p d\Omega$	Fluid capacitance matrix
$\mathbf{D}^e = \int_{\Omega} \mathbf{N}_p (S^w \mathbf{m}^T \mathbf{L}) \mathbf{N}_u d\Omega$	Solid capacitance matrix (Dilatation coupling matrix)
$\mathbf{K}^e = \int_{\Omega} \nabla \mathbf{N}_p \left(\frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} \right) \nabla \mathbf{N}_p d\Omega$	Conductance matrix

$\mathbf{g}^e = \int_{\Omega} \nabla \mathbf{N}_p \left(\frac{k_{\text{rel}}^w \mathbf{k}}{\mu^w} \rho^w \mathbf{g} \right) d\Omega$	Gravitation matrix
$\mathbf{s}^e = \int_{\Omega} \mathbf{N}_p \frac{Q_{\rho^w}}{\rho^w} d\Omega$	Fluid source term
$\mathbf{r}^e = - \int_{\Gamma} \mathbf{N}_p q_n^w d\Gamma$	Boundary fluid flux term

All material properties refer now to corresponding elements.

15.4.2 Equilibrium Problem

The discretization process for determination of fluid pressure of incompressible phase p^w is based on governing equation (15.80).

The Galerkin method yields the weak formulation of equilibrium condition

$$\int_{\Omega} \mathbf{N}_u [\mathbf{L}^T (\boldsymbol{\sigma}' - S^w p^w \mathbf{m}) + \rho \mathbf{g}] d\Omega = 0 \quad , \quad \forall \mathbf{N}_u \in H_0^1 \quad (15.94)$$

Integration by parts using the Gaussian integral theorem and rearranging unknown and known functions to left and right hand side, respectively, yields

$$\begin{aligned} \int_{\Omega} (\mathbf{L} \mathbf{N}_u)^T \boldsymbol{\sigma}' d\Omega &= \\ \left[\int_{\Omega} (\mathbf{L} \mathbf{N}_u)^T S^w \mathbf{m} \mathbf{N}_p d\Omega \right] \mathbf{p}^w + \int_{\Omega} \mathbf{N}_u^T \rho \mathbf{g} d\Omega + \int_{\Gamma} \mathbf{N}_u \mathbf{t} d\Gamma \end{aligned} \quad (15.95)$$

with the outward momentum flux vector

$$\mathbf{t} = \boldsymbol{\sigma}' \cdot \mathbf{n} \quad (15.96)$$

We introduce following operator

$$\mathbf{LN}_u = \mathbf{B} \quad (15.97)$$

The system of resulting algebraic equations can be written now as

$$\int_{\Omega} \mathbf{B}^T \boldsymbol{\sigma}' d\Omega = \mathbf{Q} \mathbf{p}^w + \mathbf{r} \quad (15.98)$$

with

$$\begin{aligned} \mathbf{Q} &= \int_{\Omega} \mathbf{B}^T S^w \mathbf{m} \mathbf{N}_p d\Omega \\ \mathbf{r} &= \int_{\Omega} \mathbf{N}_u^T \rho \mathbf{g} d\Omega + \int_{\Gamma} \mathbf{N}_u \mathbf{t} d\Gamma \end{aligned} \quad (15.99)$$

For stress-strain relation the generalized isotropic Hooke's law (15.72) is used. Then we have.

$$\int_{\Omega} \mathbf{B}^T \boldsymbol{\sigma}' d\Omega = \left[\underbrace{\int_{\Omega} \mathbf{B}^T \mathbf{C} \mathbf{B} d\Omega}_{\mathbf{K}} \right] \mathbf{u} \quad (15.100)$$

The resulting system of algebraic equation for determination of displacements is

$$\mathbf{K} \mathbf{u} = \mathbf{Q} \mathbf{p}^w + \mathbf{r} \quad (15.101)$$

with following element matrices:

$$\mathbf{K}^e = \int_{\Omega^e} \mathbf{B}^T \mathbf{C} \mathbf{B} d\Omega^e \quad \text{Stiffness matrix}$$

$$\mathbf{Q}^e = \int_{\Omega} \mathbf{B}^T S^w \mathbf{m} \mathbf{N}_p d\Omega \quad \text{Pressure coupling matrix}$$

$$\mathbf{r}^e = \int_{\Omega} \mathbf{N}_u^T \rho \mathbf{g} d\Omega + \int_{\Gamma} \mathbf{N}_u \mathbf{t} d\Gamma \quad \text{Load vector}$$

15.5 Implementation

In this chapter we describe the implementation of the RF consolidation model. In general, the model is prepared for 2 phases (1 fluid and 1 solid phase) and 2 components (water and salt) in a deformable porous medium under non-isothermal conditions.

Table 15.4: Processes and RF Kernels.

solid phase deformation	RF-DM
multi-phase fluid flow (pressures)	RF-MMP
multi-phase fluid flow (saturations)	RF-MMS

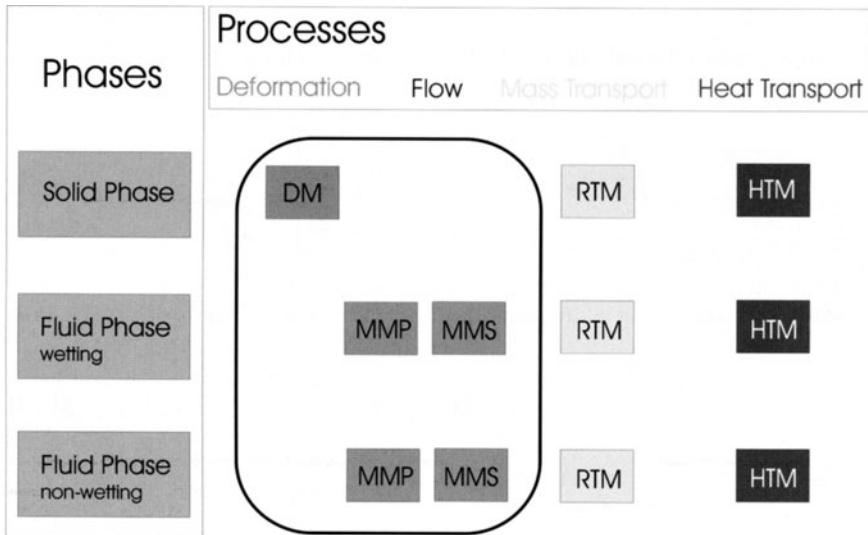


Figure 15.6: RF-FEM Kernels and couplings for the consolidation/salt-transport model.

According to the object-oriented (OO) concept of RF we can use common objects (Tab. 15.5). To generate new problem-oriented models the user has to specify model-specific objects (Tab. 15.6), which we describe in the following sections. The RF C-code is completely compatible to C++, which makes it possible to use class libraries (e.g. MFC for Windows) for the GUI.

Table 15.5: Common objects and source files.

Purpose	Object	Source files
Nodes	NOD	nodes.h/c
Elements	ELE	elements.h/c
Time stepping	TIM	rftime.h/c
Data output	OUT	rfoutput.h/c
Numerical parameter	NUM	numerics.h/c
Initial conditions	IC	rfinitc.h/c
Boundary conditions	BC	rfbc.h/c
Source-sink terms	SS	rfsousin.h/c
Material properties	MAT	material.h/c
Solver (linear/non-linear)	SOL	rfsolver.h/c
Grid adaptation	ADP	gridadap.h/c
Functions	FCT	tools.h/c
Graphical User Interface (GUI)	SHELL	rfshell.h/cpp

Table 15.6: Model-specific objects and source files.

Purpose	Object	Source files
Model configuration	MOD	mod_0010.h/c
Time discretization	LOP	loop0010.h/c
FE matrices	Deformation	cel_dm.h/c
	Fluid flow	cel_sm.h/c
	Multiphase flow	cel_mmp.h/c
	Multiphase flow	cel_mms.h/c
	Component transport	cel_rtm.h/c
	Heat transport	cel_htm.h/c
Equation systems	Deformation	cgs_dm.h/c
	Fluid flow	cgs_sm.h/c
	Multiphase flow	cgs_mmp.h/c
	Multiphase flow	cgs_mms.h/c
	Component transport	cgs_rtm.h/c
	Heat transport	cgs_htm.h/c

15.5.1 Model

Simulation Control Data

models.h/c

- Data types

```
int deformation;
int deformation_plastic;
int number_of_load_steps;
int load_step_number;
int simulation;
int max_pu_iter;
```

- Data construction

```
CreateModelData()
DestroyModelData()
```

- Data configuration

```
ConfigModel()
```

Model Node Data

mod_0010.h/c

- Data construction for unknown functions (i.e. fluid pressure, solid displacements, salt concentration, temperature)

```
CreateModelNodeData()
DestroyInternNodeData()
```

- Data access to unknown functions

MMPGetNodePress	= GetNodePress0010;
MMPGetSaturation	= GetGPSaturation0010;
MMPGetNodeSatu	= GetNodeSaturation0010;
DMGetNodeDisplacementX0	= getNodeDisplacementX0_0010;
DMGetNodeDisplacementY0	= getNodeDisplacementY0_0010;
DMGetNodeDisplacementZ0	= getNodeDisplacementZ0_0010;

- Data access to nodal data

MMPSetNodeFlux	= SetNodeFlux0010;
MMPGetNodeFlux	= GetNodeFlux0010;
MMPGetNodeRelativePermeability	= CalcNodeRelativePermeabilityM1;

- Data transfer from linear equation solver to model node data

```

LINEAR_SOLVER *TransferNodeValuesVec(LINEAR_SOLVER *ls, long index, int shift)
    void TransferNodeValsVec(double *ergebnis,int idx, int shift)
        void SetNodeVal ( long number, int idx, double val )
    {
        ((Knoten *) nodelist->elements[number])->nval[idx] = val;
    }
}

```

Model Element Data

mod_0010.h/c

- Data construction for element resultants (i.e. fluid velocity, stresses, strains)

```

CreateModelElementData()
DestroyInternElementData()

```

- Data access to element resultants

```

MMPGetRelativePermeability      = CalcRelativePermeabilityM0;
MMPGetSaturationPressureDependency = GetSaturationPressureDependencyM0;

DMSetElementStrainXX           = SetElementStrainXX0010;
DMSetElementStrainYY           = SetElementStrainYY0010;
DMSetElementStrainXY           = SetElementStrainXY0010;
DMSetElementStressXX           = SetElementStressXX0010;
DMSetElementStressYY           = SetElementStressYY0010;
DMSetElementStressXY           = SetElementStressXY0010;

```

- Data construction for element matrices

```

void *InitElementMatrices0010 ( void )
void *DestroyInternElementData0010 ( void *data )

```

- Data access to element matrices

```

MMPSetElementCapacitanceMatrix      = SetElementCapacitanceMatrixPhase0010;
MMPSetElementConductivityMatrix     = SetElementConductivityMatrixPhase0010;
MMPSetElementGravityVector          = SetElementGravityVectorPhase0010;
MMPSetElementStrainCouplingMatrixX  = SetElementStrainCouplingMatrixX0010;
MMPSetElementStrainCouplingMatrixY  = SetElementStrainCouplingMatrixY0010;
MMPGetElementCapacitanceMatrix      = GetElementCapacitanceMatrixPhase0010;
MMPGetElementConductivityMatrix     = GetElementConductivityMatrixPhase0010;
MMPGetElementGravityVector          = GetElementGravityVectorPhase0010;
MMPGetElementStrainCouplingMatrixX  = GetElementStrainCouplingMatrixX0010;
MMPGetElementStrainCouplingMatrixY  = GetElementStrainCouplingMatrixY0010;
MMPGetElementCapillarityVector      = GetElementCapillarityVectorPhase0010;
MMPGetElementSaturationMassMatrix   = GetElementSaturationMassMatrix0010;

DMSetElementStiffnessMatrix         = SetElementStiffnessMatrix0010;
DMSetElementPressureCouplingMatrixX = SetElementPressureCouplingMatrixX0010;
DMSetElementPressureCouplingMatrixY = SetElementPressureCouplingMatrixY0010;
DMSetElementGravityForceVector      = SetElementGravityForceVector0010;
DMGetElementStiffnessMatrix         = GetElementStiffnessMatrix0010;
DMGetElementPressureCouplingMatrixX = GetElementPressureCouplingMatrixX0010;
DMGetElementPressureCouplingMatrixY = GetElementPressureCouplingMatrixY0010;
DMGetElementGravityForceVector      = GetElementGravityForceVector0010;

```

15.5.2 Loop

loop0010.h/c

PreTimeLoop0010()

- Construct and initialize all solver objects (SOL)
- Define object names for all initial conditions objects (IC)
- Define object names for all boundary conditions objects (BC)
- Define object names for all sink-source term objects (SS)
- Set initial or restart conditions

TimeLoop0010()

The scheme of the time loop is given on the following page.

PostTimeLoop0010()

- Data destruction

Iteration Scheme

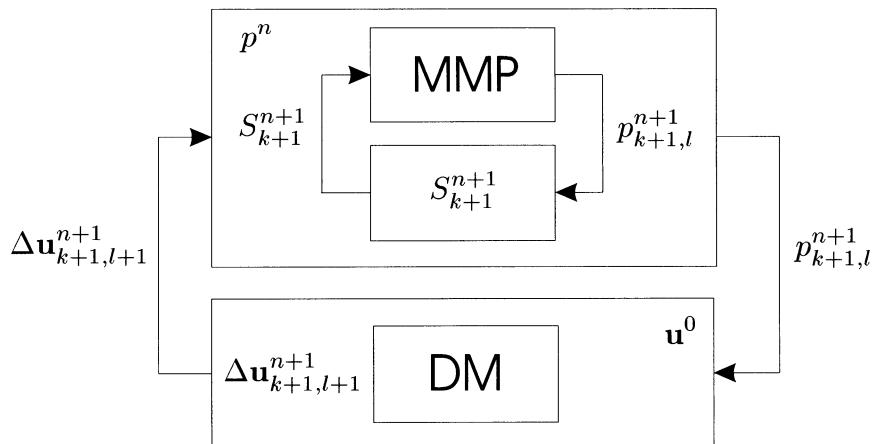


Figure 15.7: Iteration scheme, n - time level, k - multiphase iteration, l - consolidation iteration

15.5.3 Element Matrices and Resultants

cel_mmp.h/c, cel_dm.h/c

- Calculation and storage of element matrices

```
void CalcEle2D_DM(long index)

void CalcElementMatrix_MMP(long index, int phase, int matrixtyp)
    CalcEle2D_MMP(index, phase, matrixtyp);
```

- Calculation of element resultants in Gaussian points: strain and stress tensors

```
double *Calc2DStrainTensor0010(long index, double r, double s);
double *Calc2DStressTensor0010(long index, double r, double s);
```

- Functions for plastic material behavior (deformation_plastic==1)

```
double *Calc2DFlowVector(double *stress);
double *Calc2DRowVector(double *flow_vector, ...);
double *Calc2DElasticityMatrix(double lambda, double shear_module);
double Calc2DStressStrainSlope(double *d_stress, double *d_strain);
```

15.5.4 Equation System

cgs_mmp.h/c, cgs_dm.h/c

- Assembling of system matrices

```
void MakeEQS_MMP_Richards(double *rechts, double *ergebnis_MMP)
    MMTM0699CalcSaturationRichards(ergebnis_MMP, results_MMS);
    MMTM0699RebuildMatricesPressureRichards();
    MMTM0699RebuildMatricesSaturation(-1);
    MMPAssembleMatricesRichards(phase, ergebnis_MMP, rechts);

void MakeGS2D_DM(long index, double *rechts, double *ergebnis);
```

15.5.5 Solver

rfsolver.h/c

- Construction of solver objects

```
lgs_dm = CreateLinearSolver(0, 2*NodeListLength);
nlgs_mmp = CreateNonlinearSolver(speichertechnik_flow, NodeListLength);
```

- Configuration of solver objects **LINEAR_SOLVER**

```
InitializeNonLinearSolver(nlgs_mmp,
    DEFAULT_NAME_SOURCE_VOLUME_FLUID_PHASE(1),
    DEFAULT_NAME_SOURCE_MASS_FLUID_PHASE(1),
    "FREE_OUTFLOW1",
    DEFAULT_NAME_LINEAR_SOLVER_PROPERTIES_FLUID_PHASE(1),
    DEFAULT_NAME_NONLINEAR_SOLVER_PROPERTIES_FLUID_PHASE(1));

InitializeLinearSolverVec2(lgs_dm,
    name_load_solid_phase,      /*DEFAULT_NAME_LOAD_SOLID_PHASE(1), */
    name_load_solid_phase,      /*DEFAULT_NAME_LOAD_SOLID_PHASE(2), */
    name_displacement_x_phase,
    name_displacement_y_phase,
    DEFAULT_NAME_LINEAR_SOLVER_PROPERTIES_FLUID_PHASE(1));
```

- Set names of source-sink terms for solver object, specify multiplicator functions

```
for (i=0;i<number_of_phases;i++) {
    SetSourceSink(DEFAULT_NAME_SOURCE_MASS_FLUID_PHASE(i+1));
    SetSourceSink(DEFAULT_NAME_SOURCE_VOLUME_FLUID_PHASE(i+1));
}

SetSourceSinkLinearSolverVec(ls,sousin_name1,1);
SetSourceSinkLinearSolverVec(ls,sousin_name2,2);
SetSourceSinkFunctionLinearSolver(lgs_dm,2,NULL);
```

- Set names of boundary conditions for solver object

```
for(i=0;i<number_of_phases;i++)
    SetBoundaryConditions(DEFAULT_NAME(i+1));
SetBoundaryConditions("FREE_OUTFLOW1");

SetBoundaryConditionsLinearSolverVec(ls,bc_name,1);
SetBoundaryConditionsLinearSolverVec(ls,bc_name2,2);
```

- Set assembling function for the solver object

```
SetAssembleFunctionNonlinearSolver(nlgs_mmp,MakeEQS_MMP_Richards,1);
SetAssembleFunctionLinearSolver(lgs_dm,MakeGS_DM);
```

- Execution of solver including boundary conditions and source-sink terms incorporation

```
ExecuteLinearSolverVec(lgs_dm,NULL);
IncorporateBoundaryConditionsVec(ls->bc_name,ls->b,2,0);
IncorporateBoundaryConditionsVec(ls->bc_name2,ls->b,2,1);
```

- Data transfer between model and solver objects

```

TransferNodeValuesFromModelToMemoryVectorLinearSolver(lgs_dm,index);
TransferNodeValuesVec(lgs_dm,index_disp_x_0010,0);
TransferNodeValuesVec(lgs_dm,index_disp_y_0010,1);

```

15.5.6 Materials

The object for material properties MAT is used to assign the required material properties to fluids, solids, porous medium, and chemical species. Data structures for material properties are designed in particular for multiphase-multicomponent systems. Material properties can be specified for several fluid and solid phases, the fractured-porous medium itself, as well as for chemical components, which may be dissolved in a fluid phase or adsorbed to a solid phase.

```

Sγ(θ,r,s,t)
// Time collocated saturation in Gaussian points (r,s,t)
double GetGPSaturation0010
    (int phase,long index,double r,double s,double t,double theta)

kγrel(Sγ(θ,r,s,t))
// Time collocated relative permeability in Gaussian points
double CalcRelativePermeabilityM0
    (int phase,long index,double r,double s,double t,double theta)
    GetGPSaturation0010(phase,index,r,s,t,theta);
    CECalcRelPerm_AMM(index, number_of_phases,satu,kr);

kγrel(Sγ(θ,i))
// Time collocated relative permeability based on nodal saturation values
double CalcNodeRelativePermeabilityM1
    (int phase,long element,long node,double theta)
    CECalcRelPerm_AMM(element,number_of_phases,satu, kr);

dSγ/dpC(Sγ(θ,r,s,t) + 0.5dSγ,Sγ(θ,r,s,t) - 0.5dSγ)
// Calculation of dS/dp
// based on time collocated saturations in Gaussian points
double GetSaturationPressureDependencyM0
    (int phase,long index,double r,double s,double t,double theta)
    GetGPSaturation0010(phase,index,r,s,t,theta);
    CECalcCap_AMM(index, satu2 - 5.e-2, &kap1);

```

15.5.7 Data Input

- Boundary conditions for displacements

```
#BOUNDARY_CONDITIONS_DISPLACEMENT_X
int FctBoundaryConditionsDisplacementX()
#BOUNDARY_CONDITIONS_DISPLACEMENT_Y
int FctBoundaryConditionsDisplacementY()
```

- Nodal load terms

```
#LOAD_SOLID_PHASE_X
int FctLoadSolidPhaseX()
#LOAD_SOLID_PHASE_Y
int FctLoadSolidPhaseY()
```

- Material properties for solid phase

```
#SOLID_PROPERTIES
int FctSolidProperties()
```

15.6 Examples

15.6.1 Benchmark Problem - Liakopoulos Experiment

The first example is a drainage test based on an experiment by Liakopoulos (1965). Desaturation takes place due to gravitational effects. This example was studied previously by several authors, e.g. Liakopoulos (1965), Narasimhan & Witherspoon (1978), Zienkiewicz et al. (1990), Schrefler & Zhan (1993). Therefore, this example is well suited as benchmark, moreover, because of the lack of any analytical solutions for this type of coupled, non-linear problems. In this test example, multiphase flow in a deforming porous medium is studied. At first we employ the Richards approximation, i.e. air remains at atmospheric pressure.

The physical experiment of Liakopoulos was conducted in a column packed with so-called Del Monte sand. Moisture content and tension at several points along the column were measured with tensiometers (Fig. (15.8), Fig. (15.9)).

The capillary pressure $p_c(S^w)$

$$p_c = \left(\frac{1 - S^w}{1.9722} \times 10^{11} \right)^{\frac{1}{2.4279}} \quad (15.102)$$

as well as the relative permeability relationships $k_{\text{rel}}(S^w)$

$$k_{\text{rel}}^w = 1 - 2.207(1 - S^w)^{1.0121} \quad (15.103)$$

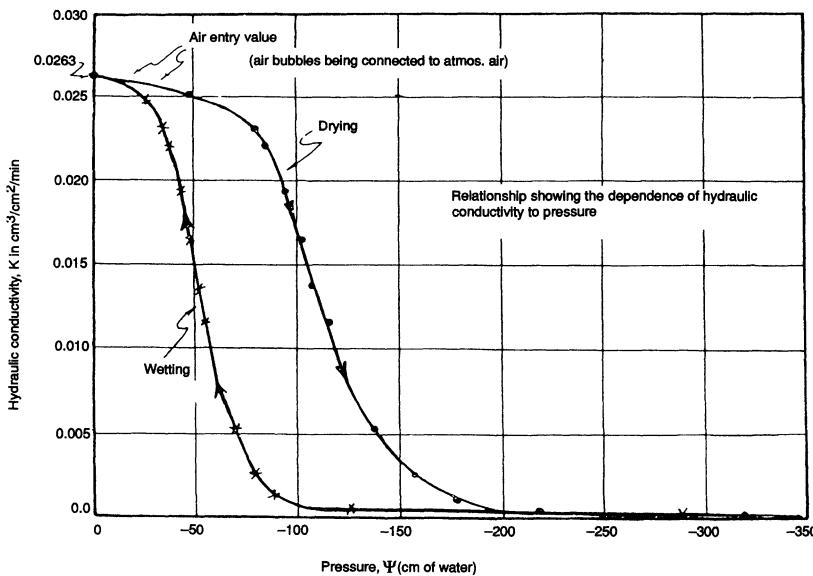


Figure 15.8: Hydraulic conductivity vs pressure, with $K = k\rho^w g / \mu^w$ and $\Psi = p^w / \rho^w g$ (Liakopoulos 1965)

Table 15.7: Model parameter

Young's modulus	E	1.3 MPa
Poisson's ratio	ν	0.4
Solid grain density	ρ^s	2000kgm^{-3}
Liquid density	ρ^w	1000kgm^{-3}
Porosity	n	0.2975
Permeability	k	$4.5 \times 10^{-13} \text{m}^2$
Water viscosity	μ^w	10^{-3}Pas
Gravity	g	9.806ms^{-2}

fit the measured data for saturations larger than 0.84.

The physical parameter are given in the table below.

The following figures show history of simulated water pressure p^w , water saturation S^w and vertical solid displacement u_y^s .

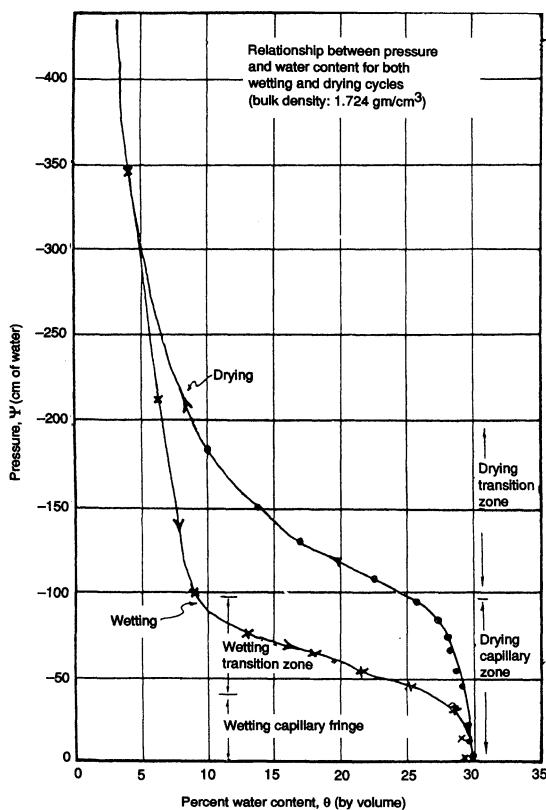


Figure 15.9: Hydraulic head vs water content, with $\Psi = p^w / \rho^w g$ and $\theta = nS^w$ (Liakopoulos 1965)

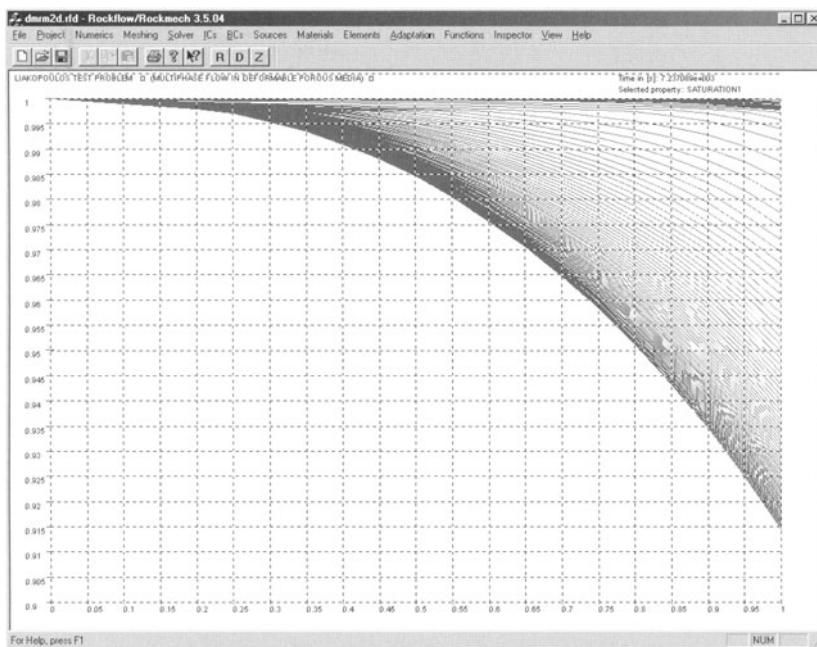
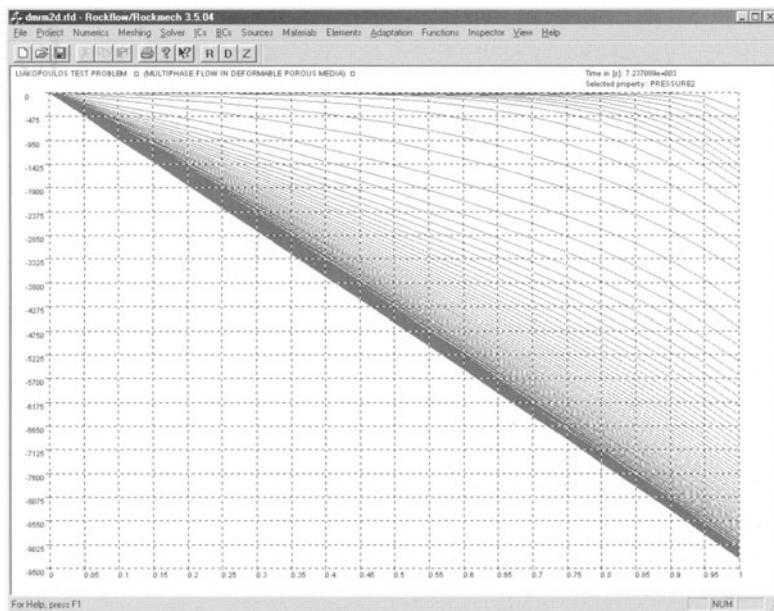
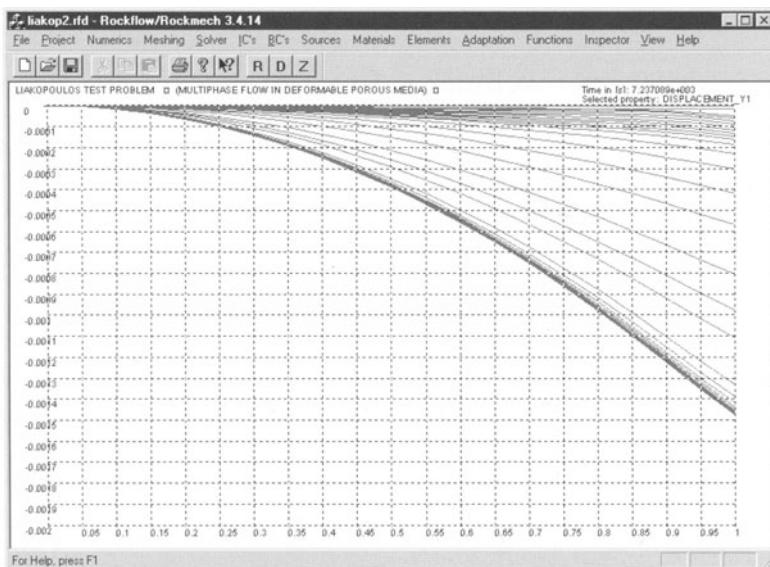


Figure 15.10: Simulated water pressure p^w

15.6.2 Moisture Swelling of Bentonites

Clays and bentonites receive increasing attention as technical buffer materials in geological barriers in order to isolate waste. Therefore, detailed knowledge of the behavior of these materials under a large variety of thermodynamical conditions is of special interest. In particular, we are interested in the varying water saturations leading to swelling and shrinking processes. Analysis of the complex material behavior of clays and bentonites requires mathematical modeling of the non-linear processes.

Engineered Barrier Systems (EBS) are basic elements of many waste repository concepts in geologic formations. Clay-like materials such as bentonites have been proved as efficient buffer materials for reducing the release of contaminants to the environment. Bentonites are often used as sealing and filling materials of waste disposals. The evaluation of the efficiency of the deposition concept requires a basic understanding of the occurring processes not only in the buffer material but in the near-field zone, too. These processes are highly complex and include heat transport, fluid flow as well as mechanical reaction of the porous medium itself. Complexity results in thermo-hydro-mechanical (THM) coupling phenomena (Gens et al. 1998). Numerical modeling can be an effective way to improve the understanding of those processes and to evaluate the long-term performance of those engineered barrier systems.

Figure 15.11: Simulated water saturation S^w Figure 15.12: Simulated solid displacements u_y^s

Clay-like materials swell and shrink with wetting and drying processes. Swelling and shrinking - also denoted as hydration - are hydromechanical coupling phenomena. They result in a dramatic alteration of material properties. The swelling effect results from additional embedding of water molecules into the solid matrix. This water becomes immobile. As a consequence, effective porosity (for fluid flow) and intrinsic permeability will decrease. With drying of the material, the porosity will be recovered to a certain degree. Those moisture transport as well as phase changes (evaporation and condensation) are controlled by heat exchange processes. Swelling behavior additionally depends on the mineral composition of the material. If the volume expansion of the material is restrained, swelling pressure is observed. This pressure can be measured by use of oedometers or triaxial cells. The relationship between swelling pressure and void ratio can be used as a measure for the swelling or compression properties of the material.

Experimental analysis and modeling of engineered barrier systems are part of several projects such as DECOVALEX and FEBEX. DECOVALEX is an international co-operative research project supported by a number of national waste management organizations. Jing et al. (1999) report on numerical analysis of in-situ experiments at Kamaishi mine in Japan. They evaluate the behavior of the complete heater-buffer-rock system. Additionally, they compare several conceptual models for HM and THM processes. Another experimental site is located in Switzerland. In the Grimsel hard rock laboratory, the NAGRA (National cooperative for the disposal of Radioactive Waste) and the ENRESA (Spanish agency for management and storage of radioactive waste) are carrying out a full-scale heating test of an engineered barrier system made up of compacted bentonite. Recently, the Federal Institute of Geosciences and Natural Resources (BGR) conducted experiments to investigate swelling phenomena of compacted bentonites.

Details of the numerical model for simulation of two-phase flow in swelling porous media can be found in Kolditz (2001) and Kolditz et al. (2001). We use the material model by Börgesson et al. (1995) and Studer et al. (1984) for swelling effects resulting in porosity and permeability changes. The numerical model is implemented in the context of multi-purpose simulator Rockflow, which is based on object-oriented programming techniques. As an example we consider the swelling of an X-80 bentonite during a saturation process. The data used for the simulations were from the VEGAS-GMT test performed by the Federal Institute of Geosciences and Natural Resources (Dr. Shao, Dr. Wallner).

Fig. 15.13 shows the simulated water pressure build-up in several observation points. The measured data can be reproduced very well, except in the first experimental period. This is due to the measurements which does not account for capillary effects. Therefore, only with beginning significance of water pressures, the data can be evaluated with the two-phase model.

These results have to be considered as preliminary ones. It has to be pointed out, that the employed permeability dependency on swelling pressure not re-

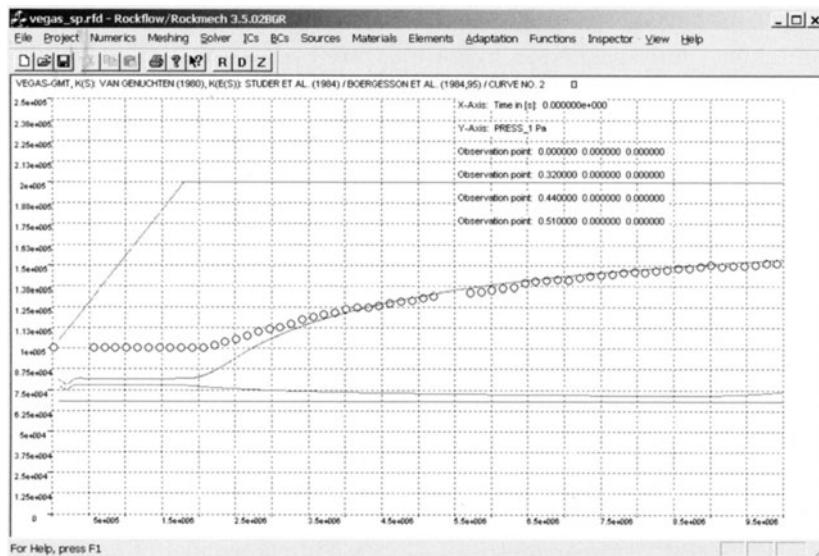


Figure 15.13: Simulated water pressure p^w

ally accounts for deformation induced permeability changes - it is an empirical relationship. Those deformation processes in variable saturated porous media, including non-isothermal effects, will be on the schedule for future work.

Bibliography

- [1] Alonso E E, Vaunat J, Gens A (1999): Modelling the mechanical behaviour of expansive clays. *Engineering Geology*, 54, 173-183.
- [2] Bear J & Bachmat Y (1990): *Introduction to modeling of transport phenomena in porous media*, Kluwer Academic Publishers.
- [3] Börgesson L, Johannesson L-E, Sanden, T, Herneling J (1995): *Modelling of the physical behavior of water saturated clay barriers - Laboratory tests, material models and finite element application*. SKB Technical Report, 95-20.
- [4] Brooks R N & Corey A T (1964): *Properties of porous media affecting fluid flow* J Irrig Drain Div ASCE, 92 (IR2): 61-8.
- [5] Fujita T, Chijimatsu H, Ishikawi H (1997): *Fundamental properties of bentonite OT-9607*. Technical Note 11-96-04, Power Reactor and Nuclear Fuel Development Corporation (PNC).
- [6] Gens A, Garcia-Molina A J, Olivella S, Alonso E E & Huertas F (1998): Analysis of a full scale in-situ test simulating repository conditions. *Int. J. Anal. Num. Meth. Geomech.*, 22:515-548.
- [7] Haverkamp R, Vauclin M, Touma J, Wierenga P J & Vachaud G (1977): *A comparison of numerical simulation models for one-dimensional infiltration* , J Soil Science Society of America, 41: 284-94.
- [8] Huyakorn P S & Pinder G F (1983): *Computational methods in subsurface flow*, Academic Press, New York - London.
- [9] Jing L, Stephansson O, Börgesson L, Chijimatzu M, Kautsky F, Tsang C-F (1999) : *DECOWALEX II Project*. Technical Report - Task 2C, SKB Report 99:23, Stockholm.
- [10] Kolditz O, Habbar A, Kaiser R, Rother T & Thorenz C (1999), *ROCKFLOW - Theory and Users Manual*, Release 3.4, Groundwater Modeling Group, Institute of Fluid Mechanics, University of Hannover, (www.rockflow.de).
- [11] Kolditz (2001): A fully coupled thermo-hydro-mechanical (THM) model. Rockflow-Preprint [2001-2], Institute of Fluid Mechanics, University of Hannover.
- [12] Kolditz O, Kohlmeier M & Shao H (2001): THM Modeling: Numerical Modelling of Moisture Swelling in Unsaturated Porous Media. Rockflow-Preprint [2001-3], Technical Report to the Bundesanstalt für Geowissenschaften und Rohstoffe (BGR), Centre of Applied Geosciences, University of Tübingen, Institute of Fluid Mechanics, University of Hannover.
- [13] Lewis R W & Schrefler B A (1998): *The finite element method in the static and dynamic deformation and consolidation of porous media*, Wiley & Sons.
- [14] Liakopoulos A (1965): *Retention and distribution of moisture in soils after infiltration has ceased* , Bulletin of the International Association for scientific hydrology, 10: 58-69.
- [15] Narasimhan T N & Witherspoon P A (1978): *Numerical model for saturated-unsaturated flow in deformable porous media* , Water Resources Research, 14: 1017-34.
- [16] Schrefler B A & Zhan X (1993): *A fully coupled model for water flow and air flow in deformable porous media* , Water Resources Research, 29: 155-67.

- [17] Segol G (1995): *Classic groundwater simulations*, PTR Prentice Hall.
- [18] Studer J, Ammann W, Meier P, Müller Ch & Glauser E (1984): *Verfüllen und Versiegeln von Stollen, Schächten und Bohrlöchern*. Nagra Technischer Bericht, 84-33, Baden, Schweiz.
- [19] Thorenz C (2001): *Model adaptive simulation of multiphase and density driven flow in fractured and porous media*, PhD thesis, in preparation, Groundwater Modeling Group, Institute of Fluid Mechanics, University of Hannover.
- [20] van Genuchten M (1980): *A closed-form equation for predicting the hydraulic conductivity of unsaturated soils*, J Soil Science Society of America, 44: 892-98.
- [21] Zienkiewicz O C, Xie Y M, Schrefler B A, Ledesma A & Bicanic N (1990): *Static and dynamic behaviour of soils: a rational approach to quantitative solutions, II. Semi-saturated problems*, Proc R Soc London A, 429: 311-21.

Index

- approximation
 - Boussinesq 35 50 309 321
 - Boussinesq extended 310 321
 - discrete 82
 - Richards 349
 - solution 82 93
- aquifer
 - porous 195
 - fractured 195
- axiom
 - Lax 83
- benchmarks
 - Henry problem 315
 - Elder problem 318
 - salt dome problem 323
 - Liakopoulos 364
- characteristics 63 68 81
- class 208
 - heredity 208
 - hierarchy 208
- condition
 - boundary 60 70 108 166 174 212 218 279
 - boundary - Cauchy 71
 - boundary - Dirichlet 71
 - boundary - Neumann 71
 - boundary - no-slip 71
 - initial 108 212 285
- constitutive relationships 52 59 198
- coordinates
 - local 145 148 150 152 155
 - material 4
 - spatial 4
 - system 158 159
 - transformation 160 168
 - transformation matrix 168
 - transformation tensor 161
- criterion
 - Courant 113
 - Courant-Friedrichs-Lowy 113
- Peclet 118 120
- tolerance 92
- von Neumann 104 256
- data
 - constructs 208
 - container 218
 - types 208
- element 215 225
 - finite 130
 - Hermitian 139
 - implementation 233
 - Lagrangian 139
 - matrix 361
 - 1-D linear bar 140
 - 1-D quadratic bar 144
 - 1-D in 3-D space 157
 - 2-D bilinear quadrilateral 148
 - 2-D triangular 145 233
 - 2-D in 3-D space 159
 - 3-D hexahedral 154
 - 3-D tetrahedral 150
 - 3-D triangular prismatic 152
- equation
 - advection 110 185
 - algebraic 82 167 212 217 222 361
 - balance 305 307 333
 - Burgers 63 122
 - conservation mass 12 24 51 333
 - conservation momentum 13 336
 - Darcy 22
 - Darcy-Weisbach 244
 - diffusion 100 161 183
 - discretization 132 174 209
 - elliptic 63 66
 - equilibrium 55 337
 - Euler 21 65
 - Euler-Lagrange 133
 - Fokker-Planck 70
 - Forchheimer 242

- general balance 11 50 59 173 197
208 305 333
 - Hagen-Poiseuille 242
 - hyperbolic 63 68
 - Kolmogorov-Prandtl 40
 - Laplace 63 67
 - linear 87 223
 - macroscopic balance 307
 - Navier-Stokes 21 35 36 65 122
241
 - non-linear 92 93 123 223
 - of state 198
 - ordinary differential 117
 - parabolic 63 67
 - partial differential (PDE) 63 82
 - Reynolds 34 36
 - solver 82
 - Stokes 21 65
 - transport 35 116 166
 - wave 63 69
- flow**
- channeling 246
 - density-dependent 301
 - divergence 8 12
 - eddies 30 39
 - fracture 241
 - incompressible 12 42 64
 - inviscid 64
 - irrotational 64
 - laminar 242
 - multiphase 333 369
 - non-linear 241 255 259 261
 - non-linear laminar 242
 - pathline 4
 - potential 67
 - streamline 4
 - subsonic 64
 - supersonic 64
 - two-phase 348 350
 - tube 242
 - turbulent 29 242
 - unsteady 64
- fluid**
- homogeneous 12
 - immiscible 45
 - Newtonian 20
 - non-Newtonian 244
- flux** 9 71 173 194 178 197
- advective 10
 - diffusive 10
- energy 23 75
 - heat 61
 - macroscopic 306
 - mass 72
 - microscopic 305
 - momentum 13 72
 - total 7
- force**
- body 53
 - external 13
 - inertial 13 22 53 336
 - pressure 53
 - surface 14
 - viscous 14 20 54
- formulation**
- convective 309
 - divergent 309
 - Eulerian 4 7 173 334
 - Lagrangian 3 5 334
- function**
- basis 133
 - derivative 97 98 99 141 174
 - derivative material 4
 - derivative - time 176
 - interpolation 134 137
 - mapping - isoparametric 139 143
 - mapping - subparametric 139
 - mapping - superparametric 139
 - polymorphism 208
 - polynomial - Lagrangian 144
 - prototypes 208
 - shape 137 138 139 140 146 148
152 153 155 158 160
 - solution - trial 134
 - test 135
 - virtual 208
 - weighting 134 135
- grid**
- adaptation 131 217
 - Delaunay triangulation 195
 - discretization - domain 130 210
 - generation 131
 - node 215 225
 - unstructured 174
- integration**
- analytical 147 152 154
 - numerical 182
- kinematics** 17
- law**
- conservation 3 136

- cubic 245
- Darcy 54 242 303 307 337 343
- Fick 10 303 305
- Fourier 23 60
- Hooke 347
- Newton 13 20
- thermodynamics 22 26
- material 212 225 363
 - bentonite 367
 - clay 367
 - compressibility 308
 - density - fluid 53 338
 - density - mass, momentum, energy 5
 - density - porous medium 338
 - density - solid 53 338
 - density - solute 308
 - friction factor 244 246
 - heat conduction 23
 - hydraulic conductivity 250
 - hydraulic resistance coefficient 244
 - hydraulic radius 245
 - permeability 221 243 246 250
 - permeability relative 343
 - porosity 221
 - roughness fracture 246 260
 - saturation 337
 - storativity 308
 - viscosity - dynamic 21
 - viscosity - kinematic 21
- matrix
 - advection 168
 - capacitance 168
 - capacitance fluid 353
 - capacitance solid 353
 - chemical solution 168
 - Cholesky decomposition 88
 - conductance 353
 - diffusion 168
 - dilatation coupling 353
 - eigenvalues 63 103
 - element 164 168 361
 - global 164
 - gravity 353
 - Jacobian 93 124 148 149 154 156 158 165 169
 - LU decomposition 88
 - mass 136
 - non-symmetrical 89
 - pressure coupling 355
- medium
 - stiffness 136 355
 - symmetrical 89
- method
 - deformable 333
 - fractured 46 245
 - fractured-porous 46 271 293
 - fracture network 260
 - fracture single 255
 - multiphase 45
 - porous 245 301
 - porous fractured 46
- numerical methods
 - analytical 81
 - boundary 81
 - boundary representation 194
 - block elimination 88
 - block iteration 88
 - Bubnov-Galerkin 135
 - Cholesky decomposition 88
 - collocation 108 135
 - conjugate gradient (CG) 88
 - cord slope 92
 - dynamic relaxation 92
 - finite difference 81 87 97 129 135 209
 - finite element 81 87 129 135 164 209
 - finite volume 81 87 135 173 209
 - Galerkin 81 135 163 166 251
 - Gaussian elimination 88
 - Gauss-Seidel 88
 - Jacobian 88
 - Krylov subspace 89
 - least square 135
 - Newton-Raphson 92 93
 - numerical 81
 - object-oriented 193
 - particle 4
 - Picard 92 311
 - preconditioning 90
 - preconditioned conjugate gradient (PCG) 311
 - Rayleigh-Ritz 133
 - Richardson 88
 - spectral 137
 - solver - direct 88
 - solver - iterative 88 89
 - stability matrix 103 106
 - stability Neumann 103
 - subdomain 135

- variational 81 133
- von Neumann 111
- weighted residuals 81 134
- model**
 - anisotropic 282
 - continuum 45
 - data 208
 - deterministic 274 287
 - fractal 246
 - fracture-matrix 274
 - fracture network 260
 - geometric 194 277
 - $k - \varepsilon$ 36 40
 - mixing length 36 39
 - network 195
 - statistic 47
 - stochastic 274 287
- number**
 - conditioning 90
 - Courant 111 124
 - Neumann 102 124
 - Peclet 118 320
 - Prandtl-Schmidt 38 43
 - Rayleigh 303 307 312
 - Reynolds 21 29 65 244 245
- numerical**
 - accuracy 84
 - consistency 83 84 86 100 111 112 118
 - convergence 83 83 86 92 112
 - diffusion 113
 - discretization - time 256
 - dispersion 115 312
 - dissipation 115
 - error 84
 - residuum 93 135
 - stability 83 84 86 100 108 111 116 119 121
 - stability - von Neumann analysis 116
 - truncation 84
- object** 214
 - class 208
 - design 208
 - element related 210 219
 - encapsulation 208
 - equation system - linear 223
 - equation system - non-linear 223
 - GUI 228 230
 - implementation 214
- interface 215
- linear equation system (LES) 223
- material 220
- mathematical 208
- methods 193 208 215
- node related 210 217
- non-linear equation system (NLES) 223
- numerical 209
- list 202
- orientation 356
- oriented programming 207
- properties 208 214
- process**
 - adiabatic 26
 - clogging 196
 - consolidation 50 196
 - convection 196 301
 - convection mixed 304
 - deformation 13 17 18 40
 - discretization 82
 - dissipation 24 40 42
 - dissipation - heat 23
 - hysteresis capillary 342
 - interphase 45
 - moisture swelling 367
 - movement - rotation 18
 - movement - translation 18
 - orientation 204
 - osmosis 196 198 302
 - production - heat 61
 - radiation 23
 - reactions - chemical 23
 - saltwater intrusion 315
 - solution 86
 - stability - physical 32 303
 - stability - hydrodynamic 32
 - thermo-hydro-mechanical (THM) 369
 - transport - mass 58
 - transport - heat 61 64 66 271
 - transport - turbulent 36
 - turbulence 29
 - work - reversible, irreversible 23
- quantity**
 - diffusivity - turbulent 38
 - dilatation 4 52
 - dispersion - hydrodynamic 59
 - dispersion - thermal 61
 - displacement 4 18 346

- heat - capacity 60
 - heat - diffusion 61
 - heat - dispersion 61
 - heat - source 61
 - energy - conservation 23
 - energy - internal 23 26
 - energy - kinetic 23 25
 - energy - sources 24
 - energy - total 23
 - energy - turbulent kinetic 38 40
 - enthalpy 26
 - entropy 26
 - extensive 5
 - intensive 5
 - hydraulic head 307
 - moisture content 340
 - mass - concentration 58 307
 - mass - conservation 51
 - mass - fraction 58 307
 - momentum - conservation 53
 - pressure 14
 - pressure capillary 339
 - pressure head 340
 - roughness 73
 - saturation 59
 - strain 345
 - strain rate 347
 - strain velocity 346
 - stress 13
 - stress - effective 55 345
 - stress - normal 14
 - stress - Reynolds 29 36 39 54
 - stress - tensor 14
 - stress - total 55 345
 - stress - viscous 14 19
 - velocity - mass-weighted 10 305
 - velocity - mean 34
 - velocity - particle 9
 - velocity - volume-weighted 305
 - viscosity - eddy 38
 - viscosity - turbulent 38
 - work 23
- RockFlow (RF)
- consolidation model 356
 - FEM 203
 - GUI 203 228
 - HGM 203
 - kernel 199 213 225 356
 - loop 226 360
 - model 199 213 224 358
- objects 200
 - solver 361
- scale
- averaging - rules 34 48
 - averaging - time 34
 - averaging - volume 47
 - macroscopic 45
 - microscopic 45
- scheme
- Adams-Bashforth 312
 - algebraic 100
 - centered 179
 - Crank-Nicolson 100 107 108 114 121 125 311
 - difference - backward 98 118
 - difference - centred 99 118
 - difference - forward 98 118
 - DuFort-Frankel 100
 - Euler-Taylor 115
 - Euler-Taylor-Galerkin (ETG) 311
 - explicit 100 108 121 124
 - forward-time-centered-space (FTCS) 101 110 121 124
 - implicit 100 105 108 121 125
 - iteration 360
 - Lax-Wendroff 113 124
 - leapfrog 113
 - preconditioning - diagonal 91
 - preconditioning - incomplete Cholesky 91
 - preconditioning - over-relaxation 91
 - preconditioning - polynomial 91
 - Richardson 100
 - trapezoid 312
 - upwind 111 116 117 118 181
 - volume - cell-centered 175 179
 - volume - cell-vertex 175
- series
- Fourier 69 111 115
 - Taylor 93 97
- simulation
- direct 36
 - network 157 159
 - large eddy 36
- site
- Rosemanowes 271
 - Soultz-sous-Forêts 261 293
- software
- application 227 229

- graphical user interface (GUI)
 - 228
 - implementation 356
- solver
- linear 87
 - non-linear 91 257
- theorem
- Gauss-Ostrogradskian integral 8
 - 13 162 163 176 182 185
 - Reynolds transport theorem 9
- volume
- area - calculation 176
 - calculation 176
 - control 4
 - representative elementary 46 242