Janez Perko

Modelling of Transport Phenomena by the Diffuse Approximate Method

Doctoral Dissertation

Mentor: Prof. dr. Božidar Šarler

School of Applied Sciences
Nova Gorica Polytechnic

2005

Acknowledgements

First and foremost, I would like to thank my supervisor Professor Božidar Šarler for his guidance and support as well as for a great deal of understanding over the past years. Special thanks also go to all who participated in my scientific evolution, in particular Professor Ching-Shyang Chen from the University of Nevada, Las Vegas, who introduced me to meshless methods, and Professor Dominique Gobin and his co-workers from Laboratory FAST at University Pierre et Marie Curie, Paris to whom I owe many clarifications of the physical aspects and also for proofreading of this work. The specific numerical method used in this work was chosen on the basis of contributions and later correspondence with Professor Hamou Sadat from Ecole Superieure d'Ingenieurs de Poitiers. I also thank him for his remarks and discussions. A great deal of education in the field of radionuclide transport was encouraged by the Slovenian Radwaste Agency (ARAO), who provided contacts and organized IAEA training awards. In this context I would like to thank Dr. Dirk Mallants from SCK-CEN, Belgium for his prompt help and for many valuable discussions and to Slovenian PA/SA team.
Next, I thank my co-workers for numerous debates. Special thanks go to Miha Založnik for his Smith and Hutton reference solution.
Last, but not least, I would like to thank my wife Tanja for her encouragements, assistance and help, my daughter Živa who was born during the writing of this work and to my parents for their continuous support.
This work is also dedicated to my late brother Martin.

Abstract

In this work, the numerical simulation of transport in porous media using the fully meshless Diffuse Approximate Method (DAM) is described. The physical background is related to mass, momentum, energy and species transport. Numerical examples provide an insight to the behavior and to the properties of DAM. First, numerical diffusion of DAM is analyzed on the Smith and Hutton problem. Next, the problem of unconfined flow is presented to prove the convergence of the method. On 1D transport of decaying species 4 explicit time stepping procedures (explicit Euler, explicit CBS, Adams-Bashforth, Runge-Kutta) are tested for their accuracy and efficiency. The solution of a realistic large-scale problem is shown on the case of coupled fluid flow and species transport trough the radwaste repository. The results for two radionuclides are presented. The problem of Darcy natural convection in a square enclosure is introduced to give the comparison between well developed Pressure Poisson and novel CBS velocity-pressure coupling algorithms. The ability of meshless methods to cope with problems defined on irregular domains is shown on the same problem. A complex problem of coupled mass, momentum, energy and species transport is presented as a last numerical example of double diffusive natural convection in a composite fluid-porous layer. Despite the fact that DAM has already been implemented on many physical problems, numerous open questions concerning accuracy and stability on non-uniform grids still exist. Stability and accuracy are improved in this work by a developed free parameter optimization algorithm. The algorithm is adopted to one parametric and to two parametric weight functions. In addition to the free parameter optimization, guidelines about the size of the domain of influence are given. The specific proper implementation of diffusive and advective terms into this type of meshless methods is described at the end.

Key words:

Diffuse approximate method, meshless method, transport in porous media, radioactive waste, CBS algorithm, optimization of free parameters.

Povzetek

Delo obravnava numerično simulacijo transporta v poroznem mediju s popolnoma brezmrežno Difuzijsko aproksimacijsko metodo (DAM). Fizikalni okvir obsega transport mase, gibalne količine, toplote in snovi. Obnašanje in lastnosti Difuzijske aproksimacijske metode so prikazane na numeričnih primerih. Na prvem, Smith and Huttonovem primeru, je analizirana numerična difuzija metode. S primerom prostega toka skozi izbrano področje je dokazana konvergenca in konsistentnost. Na enodimenzionalnem problemu transporta radionuklida sta analizirani natančnost in učinkovitost štirih časovnih shem (eksplicitna Eulerjeva, eksplicitna CBS, Adams-Bashforthova in Runge-Kutta). Simulacija toka podzemne vode in transporta radionuklidov skozi odlagališče radioaktivnih odpadkov prikazuje bolj stvaren in računsko večji je primer. Rezultati so podani za dva radionuklida. Primerjava med Poissonovim in CBS hitrostno-tlačnim algoritmom je narejena na primeru naravne konvekcije v Darcyjevem poroznem mediju v kotanji. Z istim primerom je prikazana zmožnost brezmrežnih metod za reševanje problemov na zahtevnejših geometrijah. Zadnji numerični primer dvojne difuzijske naravne konvekcije v domeni s porozno in kapljevinsko plastjo predstavlja kompleksen problem sklopljenih enačb transporta mase, gibalne količine, toplote in snovi. Kljub temu, da je bila Difuzijska aproksimacijska metoda uporabljena za reševanje mnogih fizikalnih problemov, pa so ostajala odprta številna vprašanja, ki se nanašajo na natančnost in stabilnost rešitve pri neuniformnih mrežah. Natančnost in stabilnost metode je v tem delu izboljšana z uporabo razvitega optimizacijskega algoritma za iskanje prostih parametrov. Algoritem je uporaben za optimizacijo eno in dvo parametričnih utežnih funkcij. Poleg optimizacije prostih parametrov so narejene tudi smernice glede velikosti vplivne domene. Na koncu je prikazana tudi pravilna implementacija difuzijskega in advekcijskega člena za ta tip brezmrežnih metod.

Ključne besede:

Difuzijska aproksimacijska metoda, brezmrežna methoda, transport v poroznem mediju, radioaktivni odpadki, optimizacija prostih parametrov.

Contents

List of Figures

List of Tables

List of symbols

For reasons of generalization, the units are denoted as M, L, T and Θ for mass, length, time and temperature, respectively.

	Lowercase Latin Characters	
c	free parameter	$[-]$
c_a	coefficient of porous medium geometry	$[-]$
c_F	form-drag constant	$[-]$
c_p	specific heat at constant pressure	$[L^2\,T^{-2}\,\Theta^{-1}]$
c_s	speed of sound magnitude	$[L\,T^{-1}]$
g	magnitude of gravitational acceleration	$[L\,T^{-2}]$
m	mass	$[M]$
$\boldsymbol{n}$	vector of normals	$[L]$
$\boldsymbol{p}$	position vector	$[L]$
r	radius of support	$[L]$
q	volumetric water flux	$[L^3\,T^{-1}]$
t	time	$[T]$
t_f	time in false transient	$[T]$
$t_{1/2}$	half-life	$[T]$
$\boldsymbol{v}$	pore water velocity	$[L\,T^{-1}]$
	Uppercase Latin Characters	
$\boldsymbol{A}$	left-hand side matrix	
A	aspect ratio	$[-]$
A	activity	$[Bq]$
A_m	molar activity	$[Bq/mole]$
C	concentration	$[M\,L^{-3}]$
C_{ads}	adsorbed concentration	$[M\,M^{-1}]$
D	diffusion coefficient	$[L^2\,T^{-1}]$
D^*	dispersion	$[L^2\,T^{-1}]$
D_p	molecular diffusion	$[L^2\,T^{-1}]$
Da	Darcy number	$[-]$

F	body force	$[M\,L^{-2}\,T^{-2}]$
$\mathcal{F}$	reference quality function	
$\mathcal{F}_{irr}$	reference quality function on irregular grid	
$\mathcal{F}_{uni}$	reference quality function on uniform grid	
K	bulk modulus	$[M\,L\,T^{-2}]$
K_d	distribution coefficient	$[M^{-3}\,L]$
Le	Lewis number	$[-]$
N	buoyancy ratio	$[-]$
N	number of nodes	$[-]$
P	pressure	$[M\,L^{-1}\,T^{-2}]$
$\hat{P}$	guessed pressure	$[M\,L^{-1}\,T^{-2}]$
$\breve{P}$	pressure correction	$[M\,L^{-1}\,T^{-2}]$
$\tilde{P}$	dimensionless pressure	$[-]$
R	retardation coefficient	$[-]$
Ra	Rayleigh number	$[-]$
Ra^*	porous media Rayleigh number	$[-]$
Re	Reynolds number	$[-]$
S	source per unit volume	$[M\,L^{-2}\,T^{-2}]$
$\bar{S}$	solubility limit	$[M\,L^{-3}]$
Sc	Schmidt number	$[-]$
T	temperature	$[\Theta]$
$\tilde{T}$	dimensionless temperature	[-]
$\boldsymbol{V}$	Darcy velocity vector	$[L\,T^{-1}]$
$\hat{\boldsymbol{V}}$	guessed velocity vector	$[L\,T^{-1}]$
$\breve{\boldsymbol{V}}$	velocity correction vector	$[L\,T^{-1}]$
$\tilde{\boldsymbol{V}}$	dimensionless velocity vector	[-]
$\mathcal{V}$	volume	$[L^{-3}]$
$\widehat{W}$	weight function	$[-]$

Lowercase Greek Characters

α	approximation coefficients	
α_{th}	thermal diffusivity	$[L^2\,T^{-1}]$
α_L	longitudinal dispersion	$[L]$
α_T	transversal dispersion	$[L]$
β	expansion coefficient	$[\Theta^{-1}]$
γ	artificial compressibility	$[L\,T^{-1}]$
δ	displacement	$[L]$
$\delta_{fill,uni}$	fill distance for uniform grid	$[L]$
δ_{fill}	fill distance	$[L]$
δ_{sep}	separation distance	$[L]$

ϵ	porosity	$[L^3\,L^{-3}]$
η	error	$[-]$
η_{con}	overall convergence criterion	$[-]$
η_{int}	internal convergence criterion	$[-]$
θ	saturation	$[-]$
ι	relaxation coefficient	
κ	hydraulic conductivity	$[L\,T^{-1}]$
λ	thermal conductivity	$[M\,L\,T^{-3}\,\Theta^{-1}]$
$\lambda_{1,2,3,}$	optimization step	$[-]$
$\bar{\lambda}$	decay constant	$[T^{-1}]$
μ	dynamic viscosity	$[M\,L^{-1}\,T^{-1}]$
ν	kinematic viscosity	$[L^2\,T^{-1}]$
ϱ	density	$[M\,L^{-3}]$
σ	radius of influence	$[L]$
τ	discrete time step level	$[-]$
φ	basis function	
ξ, ζ	general coordinate	$[L]$
χ	grid stretching factor	
ϕ	general scalar variable	
ω	object function forcing parameter	

Uppercase Greek Characters

Γ	boundary	
$\mathcal{K}$	intrinsic permeability	$[L^2]$
Λ	dimensionless viscosity	$[-]$
Ω	domain	
${}_n\Omega$	n'th domain of influence	
Υ	sharpness parameter	
Ψ	general function of scalar variable	

Subscripts

$_0$ or $_{ref}$	reference
$_b$	bulk
$_{conv}$	convective
$_{diff}$	diffusive
$_{eff}$	effective
$_f$	fluid
$_{i,j,k}$	counters
$_m$	mixture
$_{min}$	minimum
$_{max}$	maximum

$_P$	pressure
$_s$	solid
$_T$	thermal
$_{tot}$	total
$_C$	solutal
$_{x,y}$	Cartesian coordinates

Superscripts

^{D}	Dirichlet boundary condition
^{l}	internal iteration level
^{N}	Neumann boundary condition
^{R}	Robin boundary condition
^{T}	transposed

Abbreviations

BC	Boundary Condition
BEM	Boundary Element Method
CBS	Characteristic Based Split
DAM	Diffuse Approximate Method
DC	Direct-Chill
DOI	Domain of Influence
DRBEM	Dual Reciprocity Boundary Element Method
EFG	Element Free Galerkin
FDM	Finite Difference Method
FEM	Finite Element Method
FVM	Finite Volume Method
GKR	General Kernel Reproduction Method
LILW	Low and Intermediate Level Radioactive Waste
MLPG	Meshless Local Petrov Galerkin
MLS	Moving Least Squares
PAM	Point approximation method
PDE	Partial Differential Equation
PIM	Point Interpolation Method
PU	Partition of Unity
PUFE	Partition of Unity Finite Elements
RBFCM	Radial Basis Function Collocation Method
RKPM	Reproducing Kernel Particle Method
SPH	Smoothed particle hydrodynamics
SIMPLE	Semi Implicit Method
TTC	Tube Type Container

1 Introduction

Materials in general frequently fall under the class described as *porous media*. By a porous medium we mean a material consisting of a solid matrix with an interconnected void. The porous medium can be formed naturally or it can be fabricated. Transport phenomena in porous media thus govern many processes in engineering as well as in nature. For this reason it is not uncommon that much human effort is put into research of transport in porous media. Ever since the original work of Darcy [Darcy, 1856], the transport phenomena in porous media have been studied both experimentally and theoretically. Nowadays there are over 100 publications per year only on the problem of convection in porous media. The interest in transport in porous media seems to be increasing due to its wide application in industrial and natural systems. Despite the development of very sophisticated and relevant analytical techniques [Raghavan and Ozkan, 1994] a great majority of porous media models can be solved only by using discrete approximate solutions. In discrete approximate methods, the involved partial differential equations (PDE), which describe particular physical phenomena, are spatially and temporary discretized. The solution is obtained by the iterative solution of the involved equations. Numerous different numerical methods exist to solve the above mentioned problem.

Numerical simulations give us the possibility to have insight into processes which are difficult to understand or difficult to model experimentally such as for example change of flow structure in the case of double diffusive convection in a composite fluid-porous layer. For better understanding of such problems, an extensive analysis based on conditions and parameters variation is essential. The simulations are also needed in cases where transport processes are slow in nature. A good example of such process is a radionuclide release from the source and its transport, for example, through geological formations. Typically, radionuclides are transported by groundwater flow whose velocity can be very slow, so the transport can even be diffusion driven. Apart from the advection transport mechanism, most of the radionuclides are retarded due to sorption mechanisms. Thus the peak

concentrations at some location where this information is needed (e.g. the location of the critical group) could be reached even after several hundred thousand years. Therefore the simulation of such processes by discrete approximate method plays a key role in the safety assessment of nuclear waste disposal facilities.

The ever-increasing interest in porous media transport problems on physically complex situations and increasing development of computational capabilities results in many publications in the field of numerical implementation of transport phenomena in porous media. For example, the problem of Darcy natural convection in porous media was first numerically studied in [Chan *et al.*, 1970], using the finite difference method (FDM). A similar study was performed by [Hickox and Gartling, 1981] using the finite element method (FEM). [Prasad and Kulacki, 1984] pioneered the use of the finite volume method (FVM) for solving this problem. The problem of natural convection in porous media alone was extended to the examination of flow in a laterally heated rectangular enclosure partly filled with pure fluid and partly with a porous medium saturated by this fluid. The most comprehensive study is that in [Beckerman *et al.*, 1988]. [Beckerman and Viskanta, 1988] also made calculations for double diffusive convection during dendritic solidification of a binary mixture. The effects of double diffusion in a composite fluid-porous layer were first analyzed by [Gobin and Bennacer, 1996a].

After computers became available to the wider public, many commercial packages came onto the market. For example in the field of radionuclide transport one of the first commercial packages, Porflow for coupled groundwater flow, heat transfer and radionuclide transport was developed by [Runchal, 1982]. Many packages were developed later, based on different numerical methods. The Modflow 3D FDM based model for groundwater flow was released in 1988 [McDonald and Harbaugh, 1988], and the Tough 3D FVM based code in 1987 [Pruess, 1987] to name only a few. Traditional methods e.g. FDM, FVM described in [Ferziger and Perić, 1997] and [Patankar, 1980] or FEM ([Zienkiewicz and Taylor, 2000a], [Zienkiewicz and Taylor, 2000b], [Zienkiewicz and Taylor, 2000c]) are widely used in practice.

The emphasis of this work is in the field of meshless numerical methods. Our goal is to improve, analyze, and further develop one in this class, namely the Diffuse Approximate Method (DAM).

Our previous work

DAM was chosen on the basis of our previous work on different mesh-reduction and meshless methods. Our work began in 1998 with the comparison of different radial basis functions in DRBEM [Šarler and Perko, 1998]. For the

same method, a convergence study was made in [Perko *et al.*, 2000]. Actual application of DRBEM to natural convection in Darcy porous media was pioneered by [Šarler *et al.*, 2000] and for Darcy-Brinkman porous media in [Šarler *et al.*, 2004a]. The industrial applications were made by DRBEM first for direct-chill of aluminium slabs, where coupled momentum and energy equations are used [Šarler and Perko, 2000] and later, for radionuclide transport near a nuclear waste repository [Šarler and Perko, 2001]. The first attempt to solve natural convection in fluids by fully meshless radial basis collocation method (RBFCM) is by [Perko *et al.*, 2001b]. The same method was used for calculation of natural convection in solid-liquid systems [Perko *et al.*, 2001a]. The solution of natural convection in Darcy porous media by RBFCM is elaborated in [Šarler *et al.*, 2002], [Šarler *et al.*, 2004b] and for Darcy-Brinkman porous media in 2003 [Perko *et al.*, 2003].

From the knowledge acquired during previous work, the decision about the most appropriate meshless numerical method for a given physical problem was based on the following parameters which should be met to large extent as possible:

- Accuracy
- Stability
- Ease of implementation
- Ease of spatial and temporal discretization
- Possibility of grid refinement and adoption
- Consumption of memory
- Calculation time

A method that drew our attention was diffuse approximate method (DAM), because of the contribution of Sadat and his group who participated in numerical comparison exercise where numerical 2D simulation of melting from a vertical wall, dominated by natural convection in the liquid phase [Gobin and Le Quéré, 2000]. The results were compared between 13 authors. The DAM proved to be at least as accurate and stable as the other traditional methods even for high Prandtl and Rayleigh numbers.

The history of DAM

The Diffuse Approximate Method (DAM) was first presented in [Nayroles *et al.*, 1991], where the authors show the response of the method to the step

function. DAM was afterwards generalized in [Belytscko *et al.*, 1994b] and [Belytscko *et al.*, 1996a] who named it the Element Free Galerkin (EFG) method. They calculated the problem of fracture crack growth in [Belytscko *et al.*, 1994a], where different continuous and discontinuous weight functions were examined in order to determine the influence on the simulation of a crack, which is basically a discontinuity. The problem encountered with a discontinuity is that the effect of neighboring nodes differs to whether a certain node is on one or the other side of the crack.

In 1996 Sadat and his group [Sadat *et al.*, 1996] solved the problem of natural convection in porous media. In this work they made a simulation of Darcy flow in porous media inside differentially heated cavity for Porous media Rayleigh number Ra*=100 and Ra*=500. The results were compared to FEM solution and gave more accurate results at the same grid density. The solution of Navier-Stokes equations and comparison to established benchmark test with Reynolds number Re=5000 was found in 1998 [Couturier and Sadat, 1998b]. Afterwards they performed solid/liquid phase change phenomena simulations in [Couturier and Sadat, 1999]. Natural convection in fluids was first developed in [Couturier and Sadat, 1998a] and analyzed in [Sadat and Couturier, 2000] for 2D, and for 3D in 2002 in [Sophy and Sadat, 2002]. Sadat's group is presently devoted to research of magnetohydrodynamic problems [Sophy and Sadat, subm]. Probably the first industrial application of DAM is elaborated in [Šarler *et al.*, 2004c], where the method was implemented for calculation of solid-liquid phase change phenomena in direct-chill casting of aluminium slabs.

Overview of present work

Chapter 2: Meshless numerical methods

Although traditional methods are proved to be fast and robust, and are used to solve most of the numerical problems nowadays, they have a common spatial discretization problem, namely grid generation or polygonization. The calculation domain in traditional methods is spatially discretized by polygons, which is reflected in additional consumption of calculation time. To alleviate polygonization problems, there is an ongoing development of a new generation of numerical methods.

The polygonization problem is partly alleviated by so called mesh-reduction or semi-meshless methods, where domain integrals are transformed to boundary integrals only. A typical representative of mesh-reduction methods is the Boundary Element Method (BEM) [Brebbia *et al.*, 1984]. However,

the transformation of domain integrals in BEM is, due to fundamental solution, possible only for some PDEs. The improvement of BEM is made by the approximation of the resultant domain by a set of global approximation functions and subsequent representation of the domain integrals of these global approximation functions by the boundary integrals. This method is called the Dual Reciprocity Boundary Element Method (DRBEM) [Partridge *et al.*, 1992]. Nevertheless, the solution of boundary integrals involved in DRBEM is usually based on a cumbersome evaluation, since the polygonization of the boundary is always compulsory.

The idea to simplify mesh generation and to ease the solving of real and thus geometrically complex engineering problems leads to the development of fully meshless methods. Grid generation in the context of meshless methods is not needed. Consequently, the process of spatial discretization can, ideally, be fully automated. From the mathematical point of view, there are many ways to alleviate the polygonization by constructing meshless shape functions. Since there are many ways to construct meshless shape functions, there are many different meshless methods based on them, e.g. the meshless local Petrov-Galerkin (MLPG) method [Atluri and Zhu, 1998], the Point Interpolation Method (PIM) [Liu and Gu, 1999], the Point Assembly method (PAM) [Liu, 1999], the Finite Point Method *et al.* [Onate, 1996], Reproducing Kernel Particle Method [Liu *et al.*, 1993], which is an improvement of the Smooth Particle Method [Gingold and Monaghan, 1977], the Element Free Galerkin (EFG) method [Belytscko *et al.*, 1994b] which originates from the Diffuse Approximate Method, developed in [Nayroles *et al.*, 1991].

In Chapter 2 general definitions on meshless methods are given. In the beginning, the differences between the most typical numerical methods are presented. The main focus is on discretization issues which are typical for meshless methods, such as construction of support or domain of influence and node searching. At the end the classification of meshless methods is given.

Chapter 3: Physical background

The extent of the underlying physics in this chapter is developed only up to the level which is used in the cases shown below, starting with the general properties of porous media. The general transport equation is introduced and extended for specific transport processes. The mass and momentum conservation equations are presented for flow in porous media under various conditions. Therefore, the basic Darcy equation is extended to the Darcy-Brinkman and Darcy-Forchheimer equations. In addition, the formulation of the one domain Navier-Stokes - Darcy-Brinkman equation is given, which is

used for one-domain representation of porous-fluid domains.

The governing equation for energy transport in porous media is then presented. The energy equation is given for situations without phase transitions in the fluid phase.

Finally, the species transport, defined by the species conservation equation, is presented. The species conservation equation is given for non-decaying and non-sorbed species at the beginning. This equation is later expanded to decaying and sorbing species. Special attention is paid to the derivation and implementation of the diffusion tensor.

Chapter 4: Solution procedures

From the numerical point of view, partial differential equations can be solved in several ways. First, the description of several time-stepping procedures is introduced. The solution of non-coupled transport equations can be solved directly using these procedures. The momentum conservation equation, however, requires a more involved description, because its solution is constructed from the coupled mass and momentum conservation equations. Two algorithms are described in detail in this chapter. First the traditional Pressure Poison algorithm and then the more contemporary CBS algorithm [Zienkiewicz and Codina, 1995b] is described. The CBS algorithm is used for the first time in meshless methods in this work. The implementation of the CBS algorithm to solve the momentum equation is described separately, because it has never been applied before to any meshless method.

Chapter 5: The Diffuse Approximate Method

In this chapter the details about the DAM are given in more detail. First, the moving least squares principle is described and later applied to solve PDE. In addition, the implementation of boundary conditions of Dirichlet, Neumann and Robin type is provided.

Chapter 6: Numerical examples

The performance of the DAM is tested on several physical problems. In the beginning, the effect of numerical diffusion is analyzed on the Smith and Hutton problem. It is shown for the first time that the numerical diffusion of meshless methods is much lower than in traditional methods. The second case shows the simple but in nature very often encountered problem of unconfined flow. The problem involves flow through an open domain characterized by two materials with different hydraulic properties. The comparison

is made with a commercial package [Porflow, 2001]. The third case represents the 1D case of advective-diffusive transport of decaying and sorbing species. Several time-stepping procedures are tested and compared in terms of stability. Also the effect of time step on accuracy is shown. Later, both processes - water flow and radionuclide transport - are coupled. The release of radionuclides into groundwater are tested against the results provided by commercial package [Porflow, 2001]. This 2D case shows the capability of DAM to solve problems in realistic geometry and with realistic physical parameters. Next, the fluid flow in porous media is coupled to the energy conservation equation. This is a simple problem of Darcy natural convection in differentially heated cavity. Although the results have been already published by [Sadat *et al.*, 1996] for the same method and by [Šarler *et al.*, 2000] and [Perko *et al.*, 2003] for DRBEM and RBFCM respectively, this simulation is used for comparison between two velocity-pressure coupling algorithms; the traditional Pressure Poisson and contemporary CBS algorithm. To show that the method is capable to solve a given problem in an arbitrary geometry, one case of natural convection in the domain with complex shaped holes is presented. The latter case of double diffusive natural convection in a porous-fluid layer is environmentally and industrially important and numerically interesting. Here the problem is characterized by coupling of the momentum, energy and species conservation equations. The goal of this problem is to determine the effect of several parameters to heat and species transfer and their influence on flow pattern. The results are compared with the FVM solution. The described test cases show that DAM is well capable to deal with large-scale and geometrically complex problems.

Chapter 7: Advanced topics on DAM

Although in Chapter 6 several complex problems are solved, many details concerning DAM remain unexplained. This is because the great majority of published results as well as the results in Chapter 6 are produced on uniform grids. In order to be able to make the calculations on arbitrary grids, several issues need to be discussed in more details. Certainly the most important is the behavior of the method with respect to weight function and the involved free parameter. Second question that need to be answered is about the size of the domain of influence. The answers to both question are not definite, but they provide a good guideline for a future work. The application of advective and diffusive terms are discussed at the end. Their application needs some more attention than in other (locally conservative) numerical methods.

Main achievements

Although there is intense development in the field of meshless numerical methods, their involvement in realistic or even industrial applications is still rare. Therefore all problems presented in this work, except for natural convection in porous media, are new from the point of view of implementation into DAM and some of them even to other meshless methods. In addition to application of the method to sophisticated problems the following theoretical issues have been assessed:

- characterization of numerical diffusion in DAM
- effect of free parameter and weight function
- effect of size of the domain of influence
- effect of time stepping schemes
- effectiveness of CBS algorithm in meshless methods

2 Meshless methods

The interest in numerical modelling of industrial and environmental processes has grown increasingly in recent decades. This increasing interest is governed by different motivations; from the need to improve industrial processes in terms of efficiency, to the risk evaluations of various environmental issues. The numerical simulation of such problems requires the solving of the complex (partial) differential equations that govern these phenomena. The simulations can be made using various tools, usually by numerical methods. The most extensively used numerical methods are the finite difference method (FDM), the finite volume method (FVM), the finite element method (FEM) and for the past 20 years also the boundary element method (BEM). In these methods, the given geometry on which certain physical process takes place is discretized into *polygons*. Every such domain is represented (or discretized) by nodes. If there is a predefined connection between the given nodes this is called a *mesh*. In meshless methods, on the other hand, no predefined mesh is required and nodes can be ideally arbitrarily scattered within the problem domain and on the boundary. Because there is no need to create a mesh, and the nodes can be (ideally) created by a computer in a fully automated manner, the time an engineer would spend on conventional mesh generation can be reduced. Additionally this provides flexibility in adding, removing or adaptive re-meshing of nodes whenever and wherever needed.

Although traditional methods are implemented in many static, dynamic, linear or nonlinear stress analysis of solids, structures, as well as fluid dynamics and coupled problems, and are proved to be fast and robust, there are several limitations becoming increasingly evident, namely:

1. Creation of mesh for the problem domain is a prerequisite. Usually the engineer spends a lot of time in creating the mesh, which becomes a major component of the cost of a simulation project because the cost of CPU time is drastically decreasing. This is even more pronounced in 3D simulations and in complex geometries.

2. Numerical diffusion of advective flows is large in traditional methods

and strongly depends on the grid density.

3. Re-meshing and mesh adaptivity in traditional methods suffers from degradation of accuracy and additional numerical diffusion. Re-meshing requires "mappings" of field variables between meshes in successive stages in solving the problem.

4. For large 3D problems, the computational cost of re-meshing at each step becomes very high, even if an adaptive scheme is available.

5. Some traditional methods (FDM) require regularly distributed nodes.

6. Problems with free and moving boundaries are difficult to model.

2.1 Definitions related to meshless methods

The terminology in connection with meshless methods is still not unified. For a start, the whole branch of meshless methods is named differently in different literature. The term *meshless* was first used and is now also most commonly used [Belytscko *et al.*, 1996b]. In some literature the term *mesh free* appears, sometimes abbreviated as *MFree* [Liu, 2003]. The meaning of the two, however, is the same. Some authors oppose these terms since the definitions of *mesh*, *grid*, *cell*, *volume*, *element*, etc. are still not precisely defined. Therefore the common name *polygon free methods* is proposed, which is probably the most acceptable and non-ambiguous. Still in this work the term *meshless* will be used because nowadays it is most frequently applied in literature.

In addition to meshless methods there is also another similar branch of numerical methods, where only boundaries need to be polygonized. These numerical methods are called mesh-reduction methods or semi-meshless methods. Typical representatives of this branch are BEM [Wrobel and Aliabadi, 2002], [Brebbia *et al.*, 1984] and DRBEM [Partridge *et al.*, 1992].

2.1.1 Boundary and domain discretization

The shape of the problem domain is often very complex in reality. In addition, the intensity and complexity of the involved physical problem can also be high. To be able to cope with both the geometrical and physical complexity, this domain should be represented or discretized as precisely as possible into a finite number of nodes with known values of field variables. Usually the geometry is simplified to a reasonable representation due to the constraints

on time and computational resources. The details of the geometry should be modelled only if more accurate results are needed in these regions.

In classical numerical methods e.g. FVM and FEM, discretized nodes are connected in a predefined manner using nodal lines that form the polygons, shown in Figure 2.1(a) and 2.1(b). Meshing is performed to discretize the geometry created into elements or cells, or polygons in general. The solution on all individual elements forms the solution for the whole problem domain. Mesh generation is a very important part of the preprocess in FVM or FEM, and it can be a more or less time-consuming process. The domain has to be meshed properly into elements of specific shapes such as triangles or quadrilaterals in 2D. No overlapping and/or gaps are allowed. Connectivity must also be created between elements. This means that the information of all elements that are connected with a certain element has to be written. Many packages are designed mainly for meshing. Mesh generators for triangular grids are nowadays quite automated for 2D and also for 3D but they are time consuming. One of the main driving forces for the creation of a new

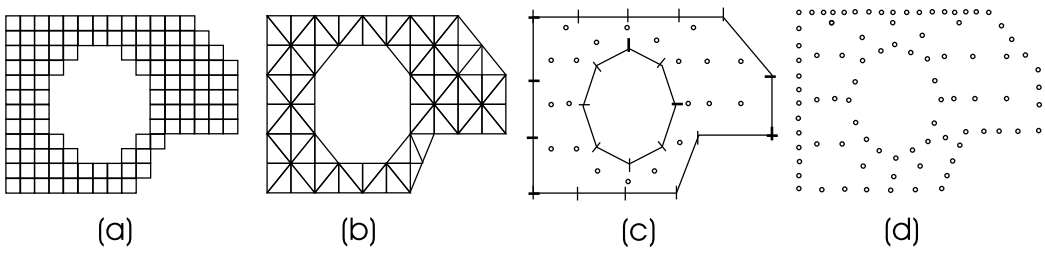


Figure 2.1: Discretization of geometry for different numerical methods; (a) FVM, (b) FEM, BEM, (c) DRBEM, (d) MSM.

numerical method is in fact the complexity of mesh generation. This led to the development of the BEM [Brebbia *et al.*, 1984] which is a weighted residual method for solving partial differential equations (PDEs), characterized by choosing an appropriate fundamental solution as a weighting function and by using the generalized Green's formula for complete transfer of one or more partial differential operators to the weighted function. The main comparative advantage of the BEM over the discrete approximative methods is demonstrated in cases where this procedure results in the boundary integral equations only. This turns out to be possible only for some PDEs. The Dual Reciprocity Boundary Element Method (DRBEM) [Partridge *et al.*, 1992] represents one of the possibilities for transforming the resultant domain integrals into a finite series of boundary integrals. The key point of the DRBEM is the approximation of the field in the domain by a set of global approximation functions and subsequent representation of the domain integrals of these

global approximation functions by the boundary integrals. The discretization of the domain is respectively represented only by grid nodes, shown in Figure 2.1(c). DRBEM belongs to the class of mesh-reduction or semi-meshless methods. The solution of boundary integrals involved in DRBEM is based on a cumbersome evaluation of regular, weakly-singular, strongly-singular, and hyper-singular integrals and polygonization of the boundary is still needed. Consequently, the more intense development of numerical methods is tending toward fully meshless numerical methods.

A given geometry or problem boundary and domain can be discretized only by setting a number of nodes, represented in Figure 2.1(d). Sometimes the term representation of the domain is used instead of discretization. The numerical solution is possible by the construction of shape function without predefined knowledge about the relationship between the nodes. Currently the most widely used methods for constructing meshless shape functions are the moving least squares (MLS) approximation and interpolation techniques. There are some others such as quasi-interpolation introduced by [Pollandt, 1997]. Each one of them has their good and bad qualities. The main advantage of MLS over interpolation is better stability and accuracy when the discretization of the domain is more unstructured [Buhmann, 2003]. As proposed in [Lazzaro and Montefusco, 2002] two quantities to measure the density of the data set are used in this work: separation distance δ_{sep} which is the half distance between the closest pair of nodes in the data set, and fill distance δ_{fill} which gives the radius of the largest inner empty space.

MLS also provides better approximation when strong non-linearities or jump conditions are involved. Their main drawback is that the constructed shape function does not possess the Kronecker delta function property. In other words, this means that the function value is not exact in the calculated node. The Kronecker delta property is important when dealing with boundary conditions of the second or third type, namely Neumann and Robin boundary conditions where the values (which should be as exact as possible) on the boundaries are calculated from the known derivatives. The condition of the Kronecker delta property is satisfied in the interpolation technique of the shape function construction. From the CPU time point of view, interpolation methods are usually a little bit faster, which is due to the construction of only a left-hand side matrix and right-hand side vector as opposed to approximation techniques where matrix multiplication is performed on both sides.

2.1.2 Support and domain of influence

The construction of shape function in all discrete numerical methods is made on the basis of known values in the neighboring nodes. The terms *support* and *domain of influence* are related to the quantity of the known values which is used for construction of the shape function. The difference between the domain of influence and the support is shown in Figure 2.2. The term

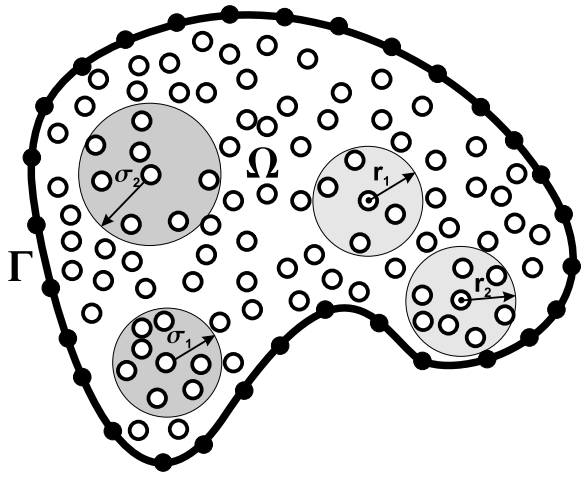


Figure 2.2: Support and influence domain.

support is used when the radius of support is given (in Figure 2.2 marked with r_1 and r_2, $r_1 = r_2$). The number of nodes inside the support varies (in Figure 2.2 from 5 nodes in r_1 to 8 nodes in r_2). The support is used for example in case of compact support radial basis functions. As a consequence the order of matrices involved varies as well. Therefore this approach can be used only when the nodal density does not vary too drastically. When the grid is more non-uniform the use of the domain of influence (DOI) is preferred. In the influence domain the number of nodes is defined, whereas the radius of influence σ varies in general ($\sigma_1 \neq \sigma_2$). Both the support and the influence domain can have different shapes, most often circular or rectangular. In the limit case where all nodes are inside the domain of influence or the support (now both terms are valid) we speak of *global interpolation methods* or *global approximation methods*.

2.1.3 Node searching

When the support or the domain of influence around the given reference node is constructed, we have to find the nodes that fall into this support or

domain of influence. Several problems arise here. First, it is very expensive if every node in the entire domain has to be checked against the node, especially if the number of nodes is large. With this we lose some of the attributes of using meshless methods. However, the problem can be alleviated by the construction of subdomains. One example of such an algorithm is the *Bucket algorithm for node searching* [Liu and Tu, 2002]. This is a very simple algorithm that divides the problem domain into buckets, each containing domain nodes up to a predefined number limit. The number limit is defined according to the problem size and maximum number of nodes allowed in an influence domain. The range of node searching can thus be reduced from the entire problem domain to a number of buckets that have overlaps with the domain of influence under construction. In other words, the nodes for each domain of influence are searched only from the nearest buckets, which reduces the overall time for node searching.

The second problem is in finding the most optimal nodes for construction of the shape function. Usually the nodes which fall into the support or the domain of influence are chosen on the basis of the smallest Euclidian distance. This kind of search is adequate in the case of uniform or less non-uniform grids. Sometimes the grids are strongly non-uniform and thus very unfavorable for use in meshless methods. Two cases of such unfavorable grids are shown in Figure 2.3. In Figure 2.3a uniform non-equidistant grid is shown. If we try to find for example the 9 nearest nodes, all will coincide in the same line. In such cases the constructed matrix is singular and no solution is obtained. Intuitively this is obvious, since the derivatives in y direction are impossible to calculate. The second case in Figure 2.3b shows a highly non-uniform grid. The solution for the reference node in this case carries the burden of the error which is due to the extrapolation of the values from the neighboring nodes. This problem can be circumvented by applying the

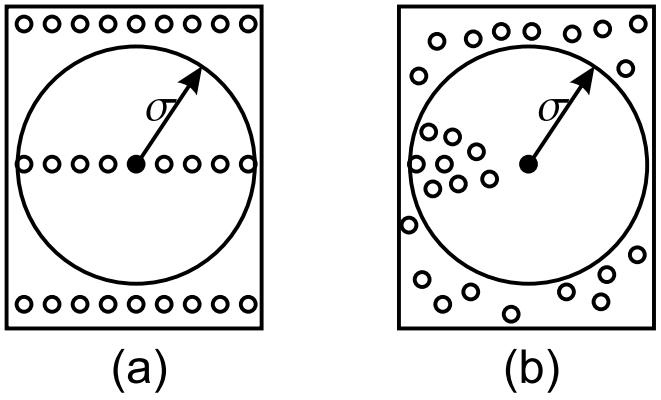


Figure 2.3: Two cases of unfavorable node distributions.

appropriate search algorithm. Similarly to the Bucket algorithm explained above, we can use the same idea here and name it the *Quartile bucket algorithm*. Consider 9 nodes in the domain of influence. It is most favorable for construction of the shape function that the nodes in the domain of influence are distributed around the reference node. Therefore we construct four buckets around the reference node. For 9 nodes each bucket will contain 2 nodes (in Figure 2.4 marked by grey circles). Now we search for the nearest

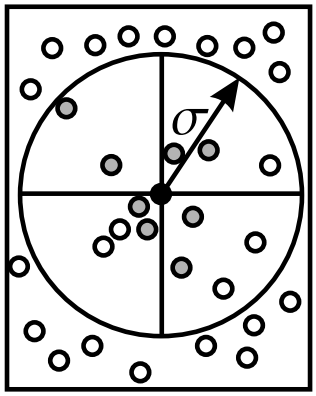


Figure 2.4: Schematic of Quartile bucket algorithm.

nodes in the first bucket, then in the second bucket and so on for the third and the fourth quartile. If there are no nodes in the problem domain to fill the bucket, the bucket is enlarged in the next quartile. This approach can be useful when dealing with highly non-uniform grids.

2.2 Classification of meshless methods

As described in the previous chapters, the main issue in meshless methods is the construction of the meshless shape function. The construction of the meshless shape function should be possible only from known the positions of the nodes without any predefined mesh. According to [Liu, 2003], a good method of shape function construction should satisfy the following basic requirements:

1. The nodal distribution can be arbitrary within reason. (*arbitrary nodal distribution*).

2. The algorithm must be stable (*stability*).

3. The constructed shape function should satisfy a certain order of consistency (*consistency*).

4. The domain for the field variable approximation/interpolation should be small compared with the entire problem domain (*locality*).

5. The algorithm should be computationally efficient (*efficiency*).

6. Ideally, the shape function should possess the Kronecker delta function property (*Kronecker delta property*).

7. Ideally, the field approximation using the shape function should be compatible throughout the problem domain (*compatibility*).

Satisfying the above listed requirements usually provides us with ease of implementation, numerical efficiency and accuracy. The first requirement is the general requirement of meshless methods.

The second requirement, *stability*, is necessary for solving realistic complex problems. The numerical stability of a method describes how a certain method responds to the differences between the calculation and the function being approximated. In a stable method, the errors due to the approximations get damped out as the computation proceeds. In an unstable method, any errors in processing get magnified as the calculation proceeds. Unstable methods quickly generate results that are physically incorrect and are useless for numerical processing [The Free Dictionary, 2004].

The third requirement, *consistency*, is essential for the convergence of the numerical results. For a method to be consistent, the truncation error must become zero when time step $\Delta t \rightarrow 0$ and/or the space between nodes tends towards zero.

The fourth requirement, *locality*, is not necessary, but it provides band shaped or small matrices which are much faster to solve. This requirement may also fall under requirement 5, where in addition unacceptably expensive shape function construction is taken into account. If the construction of the shape function is too expensive it will become unpractical no matter how good the solution is.

The 6th requirement, the *Kronecker delta property*, is not necessary as well, but it is very practical especially for imposing boundary conditions.

A number of ways to construct shape function have been proposed. The classification of the methods can be based on different properties, for example on the representation of shape function construction. There are three major categories. More detailed explanations about the specific methods listed below can be found in [Liu, 2003], [Atluri and Shen, 2002], or [Atluri, 2004].

1. Finite integral representation methods

   (a) Smoothed particle hydrodynamics method (SPH)

   (b) Reproducing kernel particle method (RKPM)
   (c) General kernel reproduction method (GKR)

2. Finite series representation methods
   (a) Moving least squares methods (MLS)
      i. MLS approximation
      ii. Generalized MLS approximation
   (b) Shepard function interpolation
   (c) Point interpolation method (PIM)
      i. Polynomial PIM
      ii. Radial PIM
   (d) Partition of unity method (PU)
      i. Partition of unity finite elements (PUFE)
      ii. *hp*-clouds
   (e) Radial basis function collocation method
      i. with global interpolation (RBFCM)
      ii. with compact support

3. Finite differential representation methods
   (a) Finite difference method for regular grids
   (b) Finite difference method for irregular grids

3 Physical background

Transport in porous media governs many processes in engineering and in nature. Processes can be described by differential equations if one independent variable is involved (e.g. time), or partial differential equations if the parameter is a function of more than one independent variable (e.g. time, space).

The descriptions of physical phenomena in this work extend only to the level of actual numerical implementation. More detailed description and derivation of relevant relations can be found in the appropriate literature e.g. [Bear, 1972], [Nield and Bejan, 1998], [Furbish, 1997].

Mass, momentum, heat and species transport in porous media are described in this work. On the macroscopic scale, the transport of momentum is governed by different equations for different flow regimes. Usually, the flow in porous media is described by Darcy's equation. For special cases of porous media flow, the Darcy equation can be adopted to Darcy-Brinkman or Darcy-Forchheimer formulations. In several instances, the corresponding domain can be occupied with both porous and fluid regions. One possibility to solve this problem is to use the one-domain approach by which both flow regimes can be described. In the one-domain approach, mixed Navier-Stokes and Darcy-Brinkman equations are often used. All above-mentioned formulations will be described below. The energy conservation equation for porous media in simple one-phase formulation without phase-change is presented. Transport of species is described by the species conservation equation, which will be developed for non-decaying and decaying species. Some important properties such as diffusion, dispersion and sorption are also given.

All three processes can be independent of each other, for example in transport of radionuclides, where water flow is not affected by the concentration of radionuclides. However, the processes can be coupled, as for example in the case of natural convection of binary fluids where flow is forced by thermal and concentration buoyancy forces.

But first we shall focus on the general properties of porous media.

3.1 Porosity

Porosity is the most obvious property of porous medium. Following the definition in [Nield and Bejan, 1998], the porosity ϵ of a porous medium is the fraction of the total volume of the medium that is occupied by void space. Thus $1 - \epsilon$ is the fraction that is occupied by solid. The pores can be either interconnected or isolated. The porosity which includes all - connected and isolated - pores is also termed total porosity. Effective porosity is introduced when not all voids are connected, so the effective porosity is the ratio of connected void to total volume. The effective porosity defines the permeability of a material. Effective porosity is typically less than total porosity. Porosity can vary in general from $0 \leq \epsilon \leq 1$. For media found in nature porosity does normally not exceed 0.6.

3.2 Density

Porous media in general is a multiphase system. The focus in this work is on saturated porous media, therefore the representative elementary volume is occupied only by a solid and a liquid phase. The densities can be averaged over the whole sample volume or over the solid phase. In the former case, when the mass of solid, denoted by m_s is averaged over r.e.v. $\mathcal{V}_{tot}$, we are speaking of *bulk density*

$$\varrho_b = \frac{m_s}{\mathcal{V}_{tot}}. \tag{3.1}$$

Bulk density of a porous medium is useful property since in practice it can be easily measured. If the mass of solid phase m_s is averaged over the volume of solid, this is called *solid density*

$$\varrho_s = \frac{m_s}{\mathcal{V}_s}. \tag{3.2}$$

These definitions are important because they describe the density properties of a porous medium, independent of any fluid present. Bulk density and density of solid can vary in space and time. The bulk density of saturated porous media can vary with fluid pressure as well [Furbish, 1997]. From the ratio between bulk and solid density we can obtain porosity

$$\epsilon = 1 - \frac{\varrho_b}{\varrho_s}. \tag{3.3}$$

3.3 Permeability and hydraulic conductivity

The specific permeability or intrinsic permeability $\mathcal{K}$ is the measure of the flow conductance of the porous matrix. If the permeability is larger, the resistance of porous matrix to flow is lower and vice versa. In the case of single-phase flow we abbreviate this to permeability. The coefficient $\mathcal{K}$ depends solely on the properties of the porous medium, but not on the fluid properties. For an anisotropic porous medium the permeability is written in tensor form, which in three dimensions can be written as

$$\boldsymbol{\mathcal{K}} = \begin{bmatrix} \mathcal{K}_{xx} & \mathcal{K}_{xy} & \mathcal{K}_{xz} \\ \mathcal{K}_{yx} & \mathcal{K}_{yy} & \mathcal{K}_{yz} \\ \mathcal{K}_{zx} & \mathcal{K}_{zy} & \mathcal{K}_{zz} \end{bmatrix}. \tag{3.4}$$

This tensor is symmetric, such as $\mathcal{K}_{xy} = \mathcal{K}_{yx}$, etc. No general relationship between effective porosity and permeability exists [Kaviany, 1995].

Especially in geology and when the fluid phase involved in transport is water, the property hydraulic conductivity κ is used. Hydraulic conductivity does not depend only on the properties of the porous matrix, but is the function of both the porous media and the fluid. The relation between hydraulic conductivity and permeability is given by

$$\kappa = \mathcal{K}\frac{\varrho_f\, g}{\mu}, \tag{3.5}$$

where ϱ_f, g and μ stand for density of fluid, magnitude of gravitational acceleration and dynamic viscosity, respectively. Similar to the permeability in anisotropic conditions, hydraulic conductivity κ also has to be written in tensor form $\boldsymbol{\kappa}$

$$\boldsymbol{\kappa} = \begin{bmatrix} \kappa_{xx} & \kappa_{xy} & \kappa_{xz} \\ \kappa_{yx} & \kappa_{yy} & \kappa_{yz} \\ \kappa_{zx} & \kappa_{zy} & \kappa_{zz} \end{bmatrix}. \tag{3.6}$$

The symmetry conditions are identical to those in permeability.

3.4 General transport equation

Before we proceed with the description of specific conservation equations, here the general transport equation [Šarler and Perko, 2000] is introduced. A general transport equation can be applied to a broad spectra of mass, energy, momentum, and species transfer problems.

$$\frac{\partial}{\partial t}[\varrho\,\mathcal{C}(\phi)] + \nabla\cdot[\varrho\,\boldsymbol{v}\,\mathcal{C}(\phi)] = -\nabla\cdot(-\boldsymbol{D}\,\nabla\phi) + S, \tag{3.7}$$

with ϱ, ϕ, t, v, $\boldsymbol{D}$, and S standing for density, transport variable, time, velocity, diffusion tensor and source, respectively. The scalar function $\mathcal{C}$ stands for possible more involved constitutive relations between the conserved and diffused properties. The application of all conservation equations used in present work into general transport equations is shown in Appendix A.

3.5 Mass and momentum conservation equation

First we recall the definition of porous media that this is a material consisting of a solid matrix with an interconnected void. Fluid in such material will, naturally, flow only through the pores. To obtain a continuum model, the velocities have to be averaged over the representative elementary volume. The velocity can be averaged over the volume element of the whole medium, or it can be averaged only over the volume of fluid. In the former case this quantity has been given different names such as seepage velocity, filtration velocity, and Darcy velocity, here denoted by $\boldsymbol{V}$. In the latter case we get pore water velocity $\boldsymbol{v}$ which is related to Darcy velocity $\boldsymbol{V}$ by the Dupuit-Forchheimer relationship $\boldsymbol{V} = \epsilon\,\boldsymbol{v}$. With this relation we can derive differential equations expressing conservation laws. The continuity equation or mass conservation equation for fluid phase can be written

$$\frac{\partial(\epsilon\,\varrho_f)}{\partial t} + \nabla\cdot(\varrho_f\,\mathbf{V}) = 0, \tag{3.8}$$

where ϱ_f is the density of fluid. If the porosity ϵ is constant, equation (3.8) is simplified to

$$\epsilon\frac{\partial\varrho_f}{\partial t} + \nabla\cdot(\varrho_f\,\mathbf{V}) = 0, \tag{3.9}$$

In addition, with assumed constant density the equation (3.9) further simplifies to

$$\nabla\cdot\mathbf{V} = 0. \tag{3.10}$$

When fluid is subjected to forces, these forces - according to Newton's second law - cause motion. The motion is described by the momentum conservation equation. Fluid flow in saturated porous media is most usually described by the Darcy equation

$$\nabla P = -\frac{\mu}{\mathcal{K}}\boldsymbol{V} + \boldsymbol{F}, \tag{3.11}$$

where ∇P is the pressure gradient and $\boldsymbol{F}$ represents volume forces acting on the fluid. These forces can result from thermal or solutal buoyancy. Approximation of buoyancy forces is described by the Boussinesq approximation

$$\boldsymbol{F} = \varrho_0 \, \mathbf{g} \, [1 - \beta_T \, (T - T_0) - \beta_C \, (C - C_0)], \tag{3.12}$$

where β_T and β_C are thermal and concentration expansion factors. Sometimes, especially in fabricated porous media, the porosities are large. In such porous media, when $\epsilon > 0.8$ [Rubinstein, 1986] another viscous term has to be added to the Darcy equation. This equation is then called *Darcy-Brinkman* equation

$$\nabla P = -\frac{\mu}{\mathcal{K}} \boldsymbol{V} + \nabla \cdot (\mu_{eff} \, \nabla \boldsymbol{V}) + \boldsymbol{F}, \tag{3.13}$$

where μ_{eff} is the effective viscosity. With the analogy to Navier-Stokes equations many authors (see [Nield and Bejan, 1998]) use acceleration and convective extensions to the Darcy equation (3.11).

$$\frac{\partial(\varrho_f \boldsymbol{v})}{\partial t} + \nabla \cdot (\varrho_f \boldsymbol{v}\boldsymbol{v}) = -\nabla P - \frac{\mu}{\mathcal{K}} \boldsymbol{V} + \nabla \cdot (\mu_{eff} \, \nabla \boldsymbol{V}) + \boldsymbol{F}. \tag{3.14}$$

Assuming constant porosity ϵ, equation (3.14) with the Dupuit-Forchheimer relation gives:

$$\frac{1}{\epsilon} \frac{\partial(\varrho_f \boldsymbol{V})}{\partial t} + \frac{1}{\epsilon^2} \nabla \cdot (\varrho_f \boldsymbol{V}\boldsymbol{V}) = -\nabla P - \frac{\mu}{\mathcal{K}} \boldsymbol{V} + \nabla \cdot (\mu_{eff} \, \nabla \boldsymbol{V}) + \boldsymbol{F}. \tag{3.15}$$

The transient term in equations (3.14) and (3.15) is derived on the assumption that the partial derivative with respect to time permutes with a volume average, but in general this is not valid. This term can be replaced by $\varrho \, \boldsymbol{c}_a \, \frac{\partial \boldsymbol{V}}{\partial t}$, where $\boldsymbol{c}_a$ is a constant tensor that depends on the geometry of the porous medium (see [Nield and Bejan, 1998] for details). In addition, some authors [Beck, 1972] and [Nield and Bejan, 1998] have stated that the use of the convective term $\nabla \cdot (\varrho_f \boldsymbol{v}\boldsymbol{v})$ can be questionable. Nevertheless, the convective term has little effect on flow when velocities are small. However, the equation becomes very useful in the *one-domain formulation* of composite fluid-porous layers, for example in the simulation of solidification process where mushy zone appears between the solid and liquid phase. This mushy zone can be macroscopically described as porous media varying from $0 \leq \epsilon \leq 1$. Porosity $\epsilon = 1$ defines a fluid and equation (3.15) becomes an ordinary Navier-Stokes equation, whereas porosity $\epsilon = 0$ defines a solid. Between these limiting values the equation exhibits Darcy-Brinkman behavior defined by equation (3.13). Instead of the use of convective term $\nabla \cdot (\varrho_f \boldsymbol{v}\boldsymbol{v})$, Forchheimer's equation can be used for large velocities

$$\nabla P = -\frac{\mathcal{K}}{\mu} \boldsymbol{V} - c_F \, \mathcal{K}^{1/2} \, \varrho \, |\boldsymbol{V}| \boldsymbol{V} + \boldsymbol{F}, \tag{3.16}$$

where c_F is a dimensionless form-drag constant.

Flow in an unsaturated porous medium exhibits quite different behavior due to the three phase system involved in an unsaturated porous medium, namely solid, fluid and air phase. This influences the resistance of the porous medium to flow in several ways. The first one is that fluid can flow only through the pores filled with fluid. The second effect is that due to capillarity large pores are dried first. Usually these effects are described in compact form as variable permeability of hydraulic conductivity, dependent on saturation θ, i.e. $\mathcal{K}(\theta)$ or $\kappa(\theta)$ for instance by the van Genuchten equation [van Genuchten, 1980]. Further description of models for unsaturated porous media will be dropped, because it exceeds the focus of this work.

3.6 Energy conservation equation

The temperature field is solved through the energy conservation equation. Assuming that there is thermal equilibrium so that $T_f = T_s = T$, where T_f and T_s are the temperatures of the solid and fluid phase, respectively, the energy conservation equation can be defined for the solid phase s

$$(1-\epsilon)(\varrho\, c_p)_s \frac{\partial T}{\partial t} = (1-\epsilon)\nabla\cdot(\boldsymbol{\lambda}_s\,\nabla T) + (1-\epsilon)\, S_{T,s}, \tag{3.17}$$

and for the liquid phase f

$$\epsilon\,(\varrho\, c_p)_f \frac{\partial T}{\partial t} + \nabla\cdot((\varrho\, c_p)_f\,\mathbf{V}\,T) = \epsilon\,\nabla\cdot(\boldsymbol{\lambda}_f\,\nabla T) + \epsilon\, S_{T,f}. \tag{3.18}$$

c_p is the specific heat at the constant pressure, $\boldsymbol{\lambda}$ is the thermal conductivity tensor, and S_T is the heat production per unit volume. In equation (3.17) it is assumed that the solid phase is not moving, which is not true in general. By assuming thermal equilibrium $T_f = T_s = T$ the summation of equations (3.18) and (3.17) gives

$$(\varrho\, c_p)_m \frac{\partial T}{\partial t} + \nabla\cdot((\varrho\, c_p)_f\,\mathbf{V}\,T) = \nabla\cdot(\boldsymbol{\lambda}_m\,\nabla T) + S_T, \tag{3.19}$$

c_p, λ, and S_T are mixture heat capacity, mixture thermal conductivity, and mixture heat production per unit volume of the medium, respectively, defined as

$$(\varrho\, c)_m = (1-\epsilon)(\varrho\, c_p)_s + \epsilon\,(\varrho\, c)_f, \tag{3.20}$$

$$\boldsymbol{\lambda}_m = (1-\epsilon)\boldsymbol{\lambda}_s + \epsilon\,\boldsymbol{\lambda}_f, \tag{3.21}$$

$$S_T = (1-\epsilon)\, S_{T,s} + \epsilon\, S_{T,f}. \tag{3.22}$$

3.7 Species conservation equation

The species equation defines the variation of the concentration field in the problem domain with given boundary conditions. For liquid phase, the species conservation equation with absence of any sources or sinks reads

$$\frac{\partial C_f}{\partial t} + \nabla \cdot (\boldsymbol{V}\, C_f) = \nabla \cdot (\bar{\boldsymbol{D}}\, \nabla C_f). \tag{3.23}$$

There are no diffusion and advection processes in the solid phase. Therefore we can write

$$\frac{\partial C_s}{\partial t} = 0. \tag{3.24}$$

C_f and C_s are the concentrations in liquid and solid phase, respectively. The combination of equations (3.23) and (3.24) gives the variation of total concentration C_{tot} (which is in fact the mixture concentration),

$$\frac{\partial C_{tot}}{\partial t} + \nabla \cdot (\boldsymbol{V}\, C_f) = \nabla \cdot (\bar{\boldsymbol{D}}\, \nabla C_f), \tag{3.25}$$

with $\boldsymbol{D}$ being the hydrodynamic dispersion coefficient. Two processes are incorporated in the hydrodynamic dispersion parameter: molecular diffusion and dispersion. Molecular diffusion is a thermochemical process, where mass is transported due to thermal or solutal gradients. On the other hand, dispersion is mechanical process, where spreading of the substance is caused due to the motion of the fluid. Although the physical basis is different for both processes, the macroscopic behavior is similar. Hence D can be partitioned into two parts

$$\bar{\boldsymbol{D}} = D_p\, \bar{\boldsymbol{I}} + \bar{\boldsymbol{D}}^*, \tag{3.26}$$

where D_p is the pore or effective molecular diffusion constant of the porous medium, $\bar{\boldsymbol{I}}$ the identity matrix and $\bar{\boldsymbol{D}}^*$ is a symmetric dispersion coefficient tensor. The elements of $\boldsymbol{D}^*$ are given by the rotation

$$\boldsymbol{D}^* = \tilde{\boldsymbol{D}}^{-1}\, \boldsymbol{D}'\, \tilde{\boldsymbol{D}}, \tag{3.27}$$

where $\tilde{\boldsymbol{D}}$ is the rotation matrix, defined in general 3D form as:

$$\begin{bmatrix} \cos\varphi_1\, \cos\varphi_2 & \sin\varphi_1\, \cos\varphi_2 & -\sin\varphi_2 \\ \cos\varphi_1\, \sin\varphi_2\, \sin\varphi_3 - \sin\varphi_1\, \cos\varphi_3 & \sin\varphi_1\, \sin\varphi_2\, \sin\varphi_3 - \cos\varphi_1, \cos\varphi_3 & \cos\varphi_2\, \sin\varphi_3 \\ \cos\varphi_1\, \sin\varphi_2\, \cos\varphi_3 + \sin\varphi_1\, \sin\varphi_3 & \sin\varphi_1\, \sin\varphi_2\, \cos\varphi_3 - \cos\varphi_1, \sin\varphi_3 & \cos\varphi_2\, \cos\varphi_3 \end{bmatrix}$$

with rotation φ_1, φ_2, and φ_3 around x, y, and z direction, respectively. $\boldsymbol{D}$' is the local dispersion tensor with longitudinal coefficient of dispersion D_L and

transverse coefficient of dispersion D_T

$$\boldsymbol{D'} = \begin{bmatrix} D_L & 0 & 0 \\ 0 & D_T & 0 \\ 0 & 0 & D_T \end{bmatrix} = \begin{bmatrix} \alpha_L\,\bar{v} & 0 & 0 \\ 0 & \alpha_T\,\bar{v} & 0 \\ 0 & 0 & \alpha_T\,\bar{v} \end{bmatrix}. \tag{3.28}$$

In the above equation α_L and α_T denote longitudinal and transverse dispersivities and $\bar{v}$ is the length of the velocity vector defined as

$$\bar{v} = \sqrt{v_x^2 + v_y^2 + v_z^2}. \tag{3.29}$$

In explicit form the dispersion coefficient tensor is written

$$D^*_{xx} = \frac{1}{\bar{v}}\left[\alpha_T\,(v_y^2 + v_z^2) + \alpha_L\,v_x^2\right], \tag{3.30}$$

$$D^*_{yy} = \frac{1}{\bar{v}}\left[\alpha_T\,(v_x^2 + v_z^2) + \alpha_L\,v_y^2\right], \tag{3.31}$$

$$D^*_{zz} = \frac{1}{\bar{v}}\left[\alpha_T\,(v_x^2 + v_y^2) + \alpha_L\,v_z^2\right], \tag{3.32}$$

$$D^*_{xy} = D^*_{yx} = \frac{v_x\,v_y}{\bar{v}}\left[\alpha_L - \alpha_T\right], \tag{3.33}$$

$$D^*_{xz} = D^*_{zx} = \frac{v_x\,v_z}{\bar{v}}\left[\alpha_L - \alpha_T\right], \tag{3.34}$$

$$D^*_{yz} = D^*_{zy} = \frac{v_y\,v_z}{\bar{v}}\left[\alpha_L - \alpha_T\right]. \tag{3.35}$$

In case the chemical reacts with the solid matrix (e.g. sorption onto the surfaces of mineral grains, organic carbon, iron oxides, hydroxides), the total concentration C_{tot} is partitioned into two terms [Jury and Flűhler, 1992]

$$C_{tot} = \epsilon\,C_f + \varrho_b\,C_{ads}, \tag{3.36}$$

where ϱ_b is bulk density and C_{ads} is adsorbed concentration. The quantity C_{ads} is expressed in units of mass of solute adsorbed per mass of dry solid. Assuming a linear sorption process, C_{ads} is expressed as

$$C_{ads} = K_d\,C_f, \tag{3.37}$$

where K_d is the distribution coefficient. Substituting C_{ads} in equation (3.36) by relation (3.37) gives

$$C_{tot} = \epsilon\,C_f + \varrho_b\,K_d\,C_f = \epsilon\,C_f\left(1 + \frac{\varrho_b\,K_d}{\epsilon}\right). \tag{3.38}$$

Finally, by inserting equations (3.38) into (3.25) yields

$$R\,\frac{\partial C_f}{\partial t} + \nabla \cdot (\boldsymbol{v}\, C_f) = \nabla \cdot (\mathbf{D}\, \nabla C_f), \tag{3.39}$$

where R is called the retardation factor, defined for linear sorption as defined in (3.38) by

$$R = 1 + \frac{\varrho_b\, K_d}{\epsilon}. \tag{3.40}$$

In fact, the retardation acts macroscopically by slowing down the transport of species as it directly influences the transient term. Linear sorption can be assumed only in the case when the sorption and desorption process may proceed sufficiently rapidly relative to local variation in concentration induced by advection and dispersion. More general definitions of equilibrium sorption are given by the Freundlich sorption isotherm or Langmuir sorption isotherm, but their description exceeds the focus of this work. When dealing with decaying substances the decay term is also included. This extension of equation (3.39) for a single radionuclide is

$$R\,\frac{\partial C_f}{\partial t} + \nabla \cdot (\boldsymbol{v}\, C_f) = \nabla \cdot (\mathbf{D}\, \nabla C_f) - \bar{\lambda}\, R\, C_f, \tag{3.41}$$

where $\bar{\lambda}$ is the first-order decay coefficient, defined as $\bar{\lambda} = \ln 2 / t_{1/2}$, with $t_{1/2}$ being the half life. In the absence of all transport mechanisms, the solution of (3.41) simplifies to

$$C_f = C_{f,0}\, e^{-\bar{\lambda} t}, \tag{3.42}$$

with initial concentration $C_{f,0}$. In addition, in decay chains the decay of a parent radionuclide leads to the generation of daughter radionuclide. In such case, the transport of several chemicals is involved in a sequential first-order decay chain. For two elements, liquid concentrations C_1 and C_2 (the subscript f is dropped) the coupled advection-dispersion equation becomes

$$R_1\,\frac{\partial C_1}{\partial t} + \nabla \cdot (\boldsymbol{v}\, C_1) = \nabla \cdot (\mathbf{D}\, \nabla C_1) - \bar{\lambda}_1\, R_1\, C_1, \tag{3.43}$$

$$R_2\,\frac{\partial C_2}{\partial t} + \nabla \cdot (\boldsymbol{v}\, C_2) = \nabla \cdot (\mathbf{D}\, \nabla C_2) - \bar{\lambda}_2\, R_2\, C_2 + \bar{\lambda}_1\, R_1\, C_1, \tag{3.44}$$

where the last term on the right-hand side of equation (3.44) accounts for the generation or production of C_2 owing to the decay of C_1.

4 Solution procedures

In many practical engineering applications extensive studies of fluid flow, heat and/or species transfer are essential. The given physical problems are mathematically described by appropriate differential or integral equations (see Chapter 3). In general, the equations are fully coupled, meaning that each equation depends on the solution of the other equations. For example, in fluid flow, the mass conservation equation and momentum conservation equations are coupled via density, velocity and pressure. In addition, the flow can be forced by thermal or solutal gradients, which in turn, are affected by fluid flow. For each case presented in Chapter 6 the exact solution procedure is provided.

Time discretization or integration over time plays, apart from space discretization, a key role in the solution of steady and unsteady problems. Some common time-stepping procedures are addressed in this chapter. The study is performed regardless of the particular type of space discretization. Next, the solution of coupled momentum and mass conservation equations for fluid flow are elaborated, since this is a case that requires special attention. The solution procedure is described for Navier-Stokes equations and the Darcy equation. Two procedures are presented here and compared in Chapter 6.5.

4.1 Time discretization

If the variation of transport variable $\phi(\boldsymbol{p})$ is time dependent we gain an additional dimension which is time. Similar to discretization of space, time also has to be discretized, but with one difference, which is the direction of influence. Whereas each source anywhere in the domain may have an effect anywhere else, the same source in time may affect only the events in the future. In other words, there is no reverse influence. The solution of a general time-dependent problem defined on a domain Ω with boundary Γ, described in equation (3.7) is constructed by the initial and boundary conditions. The value of transport variable $\phi(\boldsymbol{p}, t)$ at a node with position

vector $\boldsymbol{p}$ and the initial time t_0 is defined as

$$\phi(\boldsymbol{p}, t_0) = \phi_0(\boldsymbol{p}); \qquad \boldsymbol{p} = \Omega + \Gamma. \tag{4.1}$$

Basically we are interested in the solution of the transport variable at the new time $t^{\tau+1} = t^{\tau} + \Delta t$. Therefore, the solution of the transport variable at the new discrete time level $\phi(\boldsymbol{p}, t^{\tau+1})$ is calculated from the solution at the previous time level $\phi(\boldsymbol{p}, t^{\tau})$. Time step Δt denotes the difference between two discrete time levels. For easier presentation, the notation $\phi(\boldsymbol{p}, t^{\tau})$ is simplified to $\phi^{\tau}(\boldsymbol{p})$ and the general transport equation (3.7) is rewritten to

$$\frac{\partial[\varrho\, \mathcal{C}(\phi(\boldsymbol{p}))]}{\partial t} = \Psi(\phi(\boldsymbol{p})), \tag{4.2}$$

the term $\Psi(\phi(\boldsymbol{p}))$ represents all involved terms that might occur (e.g. advective, diffusive, sources or sinks). For the sake of simplicity, relation (4.2) is defined below for constant density and non-involved transport variable ϕ as

$$\frac{\partial \phi(\boldsymbol{p})}{\partial t} = \Psi(\phi(\boldsymbol{p})). \tag{4.3}$$

If small time steps are used, the solution is accurate regardless of the choice of time stepping procedure. The choice of time stepping procedure, however, has a great influence on stability, accuracy and computing time when larger time steps are preferred. A quite extensive description of the time stepping procedures can be found in [Ferziger and Perić, 1997]. The principal time stepping strategies are summarized below.

4.1.1 Explicit Euler method

This is the simplest method in which all fluxes and sources are evaluated using known values from previous time step τ. The general equation (4.3) can be discretized in the finite difference manner as

$$\frac{\phi^{\tau+1}(\boldsymbol{p}) - \phi^{\tau}(\boldsymbol{p})}{\Delta t} = \Psi(\phi^{\tau}(\boldsymbol{p})) + \mathcal{O}(\Delta t), \tag{4.4}$$

or

$$\phi^{\tau+1}(\boldsymbol{p}) = \phi^{\tau}(\boldsymbol{p}) + \Delta t\, \Psi(\phi^{\tau}(\boldsymbol{p})). \tag{4.5}$$

The truncation error $\mathcal{O}(\Delta t)$ denotes the first order accuracy in time. The only unknown is $\phi^{\tau+1}(\boldsymbol{p})$. All other values are taken from earlier time steps. Since the extrapolation to the value at the new discrete time level can have a significant error, this method has the lowest stability among all methods. Nevertheless, the explicit Euler method is most frequently used in our calculations for three reasons:

- calculation speed (no need for evaluation of the left-hand side of the equation)
- the memory consumption is low
- in many problems the processes experience transitions which are fairly fast. In such cases the time step should be small anyway to account for all transient effects.

4.1.2 Implicit method

If stability is a prime requirement, the implicit method is used. The implicit method is first order accurate in time with $\mathcal{O}(\Delta t)$.

$$\frac{\phi^{\tau+1}(\boldsymbol{p}) - \phi^{\tau}(\boldsymbol{p})}{\Delta t} = \Psi(\phi^{\tau+1}(\boldsymbol{p})) + \mathcal{O}(\Delta t), \tag{4.6}$$

or

$$\phi^{\tau+1}(\boldsymbol{p}) - \Delta t\, \Psi(\phi^{\tau+1}(\boldsymbol{p})) = \phi^{\tau}(\boldsymbol{p}). \tag{4.7}$$

The unknowns at the new discrete time level are $\phi^{\tau+1}(\boldsymbol{p}) - \Delta t\, \Psi(\phi^{\tau+1}(\boldsymbol{p}))$. Because all unknowns are on the left-hand side of equation, this makes them harder to program and they use more computer memory and computational time per time step. The implicit method is unconditionally stable. Even for large time steps the implicit method produces a stable but not necessarily physically correct solution. The implicit method is especially useful for solving steady problems since large time steps can be used.

4.1.3 Semi-implicit method

This method is a combination of the implicit and the explicit method.

$$\frac{\phi^{\tau+1}(\boldsymbol{p}) - \phi^{\tau}(\boldsymbol{p})}{\Delta t} = \iota\, \Psi(\phi^{\tau+1}(\boldsymbol{p})) + (1-\iota)\, \Psi(\phi^{\tau}(\boldsymbol{p})) + \mathcal{O}(\Delta t^2), \tag{4.8}$$

or

$$\phi^{\tau+1}(\boldsymbol{p}) - \iota\, \Delta t\, \Psi(\phi^{\tau+1}(\boldsymbol{p})) = \phi^{\tau}(\boldsymbol{p}) + (1-\iota)\Delta t\, \Psi(\phi^{\tau}(\boldsymbol{p})). \tag{4.9}$$

If parameter ι is set to $\iota = 0$ the method is explicit Euler and for $\iota = 1$ a fully implicit time scheme is employed. If $\iota = 0.5$ then the method is called Crank-Nicolson. This method is implicit in general, but it contains the contribution from the previous time level which makes it second order accurate in time.

4.1.4 Multilevel methods

All possible two-level methods have been presented previously. Additional time levels can be added to achieve higher-order approximation. The multipoint methods are derived by fitting a polynomial to the derivatives at a number of levels in time. If a Langrange polynomial is fit to the discrete time levels from m previous time steps $\phi^{\tau-m+1}(\boldsymbol{p})$, $\phi^{\tau-m+2}(\boldsymbol{p})$,...,$\phi^{\tau}(\boldsymbol{p})$, the explicit method of order $\mathcal{O}(m)$ is obtained. Methods of this type are called *Adams-Bashforth* methods. The first order method ($m = 1$) is explicit, while the second ($m = 2$) and the third ($m = 3$) are

$$\phi^{\tau+1}(\boldsymbol{p}) = \phi^{\tau}(\boldsymbol{p}) + \frac{\Delta t}{2}\left[3\,\Psi(\phi^{\tau}(\boldsymbol{p})) - \Psi(\phi^{\tau-1}(\boldsymbol{p}))\right] + \mathcal{O}(\Delta t^2) \tag{4.10}$$

and

$$\phi^{\tau+1}(\boldsymbol{p}) = \phi^{\tau}(\boldsymbol{p}) + \frac{\Delta t}{12}\left[23\,\Psi(\phi^{\tau}(\boldsymbol{p})) - 16\,\Psi(\phi^{\tau-1}(\boldsymbol{p})) + 5\,\Psi(\phi^{\tau-2}(\boldsymbol{p}))\right] + \mathcal{O}(\Delta t^3) \tag{4.11}$$

If data at $t^{\tau+1}$ is included in the interpolation polynomial, implicit methods known as *Adams-Moulton* methods are obtained. The first order is an implicit method, the second order is semi-implicit, and the third order is

$$\phi^{\tau+1}(\boldsymbol{p}) = \phi^{\tau}(\boldsymbol{p}) + \frac{\Delta t}{12}\left[5\,\Psi(\phi^{\tau+1}(\boldsymbol{p})) + 8\,\Psi(\phi^{\tau}(\boldsymbol{p})) - \Psi(\phi^{\tau-1}(\boldsymbol{p}))\right] + \mathcal{O}(\Delta t^3) \tag{4.12}$$

The multilevel approach has the advantage that it is relatively easy to construct and it requires only one evaluation of $\Psi(\phi)$ per time level. Its disadvantage is, because it requires data from many prior points, that it cannot be started with. For the evaluation of the first steps other lower level methods i.e. explicit method for Adams-Bashforth and implicit for Adams-Moulton methods have to be used.

4.1.5 Runge-Kutta Methods

The difficulties in starting multilevel methods can be alleviated by using time levels between two time steps. These methods are called Runge-Kutta methods. The second order Runge-Kutta method consists of two steps. The first one is a half-step predictor based on the explicit Euler methods.

$$\phi^{\tau+1/2}(\boldsymbol{p}) = \phi^{\tau}(\boldsymbol{p}) + \frac{\Delta t}{2}\,\Psi(\phi^{\tau}(\boldsymbol{p})). \tag{4.13}$$

It is followed by a midpoint corrector which makes the method second-order accurate

$$\phi^{\tau+1}(\boldsymbol{p}) = \phi^{\tau}(\boldsymbol{p}) + \Delta t\,\Psi(\phi^{\tau+1/2}(\boldsymbol{p})) + \mathcal{O}(\Delta t^2). \tag{4.14}$$

Higher order Runge-Kutta methods have been developed as well. The most commonly used is fourth order Runge-Kutta

$$\begin{aligned}
\phi_*^{\tau+1/2}(\boldsymbol{p}) &= \phi^\tau(\boldsymbol{p}) + \frac{\Delta t}{2}\,\Psi(\phi^\tau(\boldsymbol{p})), \\
\phi_{**}^{\tau+1/2}(\boldsymbol{p}) &= \phi^\tau(\boldsymbol{p}) + \frac{\Delta t}{2}\,\Psi(\phi_*^{\tau+1/2}(\boldsymbol{p})), \\
\phi_*^{\tau+1}(\boldsymbol{p}) &= \phi^\tau(\boldsymbol{p}) + \Delta t\,\Psi(\phi_{**}^{\tau+1/2}(\boldsymbol{p})), \\
\phi^{\tau+1}(\boldsymbol{p}) &= \phi^\tau(\boldsymbol{p}) + \frac{\Delta t}{6}\,\big[\Psi(\phi^\tau)(\boldsymbol{p}) + 2\,\Psi(\phi_*^{\tau+1/2}(\boldsymbol{p})) + \\
&\quad 2\,\Psi(\phi_{**}^{\tau+1/2}(\boldsymbol{p})) + \Psi(\phi_*^{\tau+1}(\boldsymbol{p}))\big] + \mathcal{O}(\Delta t^4).
\end{aligned} \tag{4.15}$$

The major problem with Runge-Kutta methods is that the function $\Psi(\phi)$ has to be evaluated several times in one time step. These methods are therefore more expensive for calculation than the multipoint methods of comparable order. However, they are more accurate and more stable than the multipoint methods of the same order.

4.1.6 Characteristic Based Split (CBS) algorithm

The CBS algorithm was introduced in [Zienkiewicz and Codina, 1995b] and [Zienkiewicz and Codina, 1995a] for the reason of stability improvement of the numerical procedure. The CBS is a time-iterative solution algorithm that was shown to be very general for the solution of transport equations. The basic idea of the CBS algorithm is to introduce an approach which lies somewhere between Lagrangian and Eulerian treatment [Zienkiewicz *et al.*, 1984]. So the values at the new time level $t^{\tau+1}$ are not calculated simply from the values in the spatial coordinate $\boldsymbol{p}$ in time level t^τ, but from the coordinate which is displaced for some value $\boldsymbol{\delta}$ from the original coordinate i.e. $\boldsymbol{p}-\boldsymbol{\delta}$ at time level t^τ. The solution is obtained by Taylor expansion around $\boldsymbol{\delta}$

$$\Psi(\phi^\tau(\boldsymbol{p}-\boldsymbol{\delta})) \approx \Psi(\phi^\tau(\boldsymbol{p})) - \boldsymbol{\delta}\cdot\nabla\,\Psi(\phi^\tau(\boldsymbol{p})) + \mathcal{O}(\boldsymbol{\delta}^2). \tag{4.16}$$

The displacement $\boldsymbol{\delta}$ is the distance travelled by the particle in the characteristic direction ξ; $\xi \in \{x, y\}$ in time step Δt, which is

$$\boldsymbol{\delta} = \overline{\boldsymbol{V}}\Delta t, \tag{4.17}$$

where $\overline{\boldsymbol{V}}$ is an average value of $\boldsymbol{V}$. The average velocity $\overline{\boldsymbol{V}}$ is defined here

$$\overline{\boldsymbol{V}}(\boldsymbol{p}) = \frac{\boldsymbol{V}^{\tau+1}(\boldsymbol{p}) + \boldsymbol{V}^\tau(\boldsymbol{p})}{2}. \tag{4.18}$$

For a fully explicit scheme the relation (4.18) simplifies to

$$\overline{\boldsymbol{V}}(\boldsymbol{p}) = \boldsymbol{V}^{\tau}(\boldsymbol{p}). \tag{4.19}$$

As explained above, the semi-implicit equation (4.9) is rewritten in a way that the right side of equation is introduced in Lagrangian coordinates. In other words, the stability of the procedure is improved by taking the information at positions along the characteristic $(\boldsymbol{p}-\boldsymbol{\delta})$ and not from the actual position $\boldsymbol{p}$.

$$\frac{\phi^{\tau+1}(\boldsymbol{p}) - \phi^{\tau}(\boldsymbol{p})}{\Delta t} = \iota\, \Psi(\phi^{\tau+1}(\boldsymbol{p})) + (1-\iota)\, \Psi(\phi^{\tau}(\boldsymbol{p}-\boldsymbol{\delta})), \tag{4.20}$$

where $0 < \iota < 1$ is set to 0 for fully explicit, 0.5 for semi-implicit, and 1 for fully implicit time scheme. By first order Taylor expansion (4.16) of equation (4.20) we obtain

$$\frac{\phi^{\tau+1}(\boldsymbol{p}) - \phi^{\tau}(\boldsymbol{p})}{\Delta t} = \iota\, \Psi(\phi^{\tau+1}(\boldsymbol{p})) + (1-\iota)\, [\Psi(\phi^{\tau}(\boldsymbol{p})) - \boldsymbol{\delta} \cdot \nabla \Psi(\phi^{\tau}(\boldsymbol{p}))]. \tag{4.21}$$

Inserting (4.17) into (4.21) and assuming (4.19) gives

$$\begin{aligned} &\frac{\phi^{\tau+1}(\boldsymbol{p}) - \phi^{\tau}(\boldsymbol{p})}{\Delta t} = \\ &\iota\, \Psi(\phi^{\tau+1}(\boldsymbol{p})) + (1-\iota)\, [\Psi(\phi^{\tau}(\boldsymbol{p})) - \Delta t\, \boldsymbol{V}^{\tau}(\boldsymbol{p}) \cdot \nabla \Psi(\phi^{\tau}(\boldsymbol{p}))]. \end{aligned} \tag{4.22}$$

For semi implicit scheme $\iota = 0.5$ the equation (4.22) reads

$$\begin{aligned} &\frac{\phi^{\tau+1}(\boldsymbol{p}) - \phi^{\tau}(\boldsymbol{p})}{\Delta t} = \\ &\frac{1}{2}\Psi(\phi^{\tau+1}(\boldsymbol{p})) + \frac{1}{2}[\Psi(\phi^{\tau}(\boldsymbol{p})) - \Delta t\, \boldsymbol{V}^{\tau}(\boldsymbol{p}) \cdot \nabla \Psi(\phi^{\tau}(\boldsymbol{p}))]. \end{aligned} \tag{4.23}$$

Further approximation can be made for explicit procedure by

$$\frac{1}{2}\Psi(\phi^{\tau+1}(\boldsymbol{p})) + \frac{1}{2}\Psi(\phi^{\tau}(\boldsymbol{p})) = \Psi(\phi^{\tau}(\boldsymbol{p})). \tag{4.24}$$

Now, the explicit CBS procedure gives

$$\frac{\phi^{\tau+1}(\boldsymbol{p}) - \phi^{\tau}(\boldsymbol{p})}{\Delta t} = \Psi(\phi^{\tau}(\boldsymbol{p})) - \frac{\Delta t}{2} \boldsymbol{V}^{\tau}(\boldsymbol{p}) \cdot \nabla \Psi(\phi^{\tau}(\boldsymbol{p})). \tag{4.25}$$

Finally, by rearrangement of (4.25) we obtain

$$\phi^{\tau+1}(\boldsymbol{p}) = \phi^{\tau}(\boldsymbol{p}) + \Delta t\, \Psi(\phi^{\tau}(\boldsymbol{p})) - \frac{\Delta t^2}{2}\, \boldsymbol{V}^{\tau}(\boldsymbol{p}) \cdot \nabla\, \Psi(\phi^{\tau}(\boldsymbol{p})). \tag{4.26}$$

In the CBS algorithm we gain the stabilizing term $\frac{\Delta t^2}{2}\, \boldsymbol{V}^{\tau}(\boldsymbol{p}) \cdot \nabla\, \Psi(\phi^{\tau}(\boldsymbol{p}))$ in addition to fully explicit Euler procedure.

4.2 Treatment of momentum conservation equation

Previously, several different time stepping procedures were explained. By knowing the initial and boundary conditions, a general transport equation can be solved by any of them. However, the solution can be obtained in a straightforward manner only for partial differential equations with one unknown transport variable, e.g. energy conservation equation with unknown temperature T or species conservation equation with unknown concentration C. However, the solution of the momentum conservation equation is more complex, since the velocity $\boldsymbol{V}$ and pressure P are involved in the same equation and are interdependent. Therefore, the solution has to be obtained by coupling of the momentum and mass conservation equations. Here the discussion is limited to primitive variable formulation, because it is the most general.

Two algorithms are explained here. The first one, the Explicit pressure correction algorithm, is used often in flow calculation. The second one, the CBS algorithm, is relatively new and it has not yet been implemented to any meshless method. The solution procedure is presented for a combined Navier-Stokes equation with Darcy term (3.15). Appropriate formulation for the solution of the Darcy equation is shown as well. The solution of the momentum conservation equation starts with known velocity and pressure boundary conditions and with initial velocity field at the initial time $t^{\tau=0}$ at each node with position $\boldsymbol{p}$

$$\boldsymbol{V}(\boldsymbol{p}, 0) = \boldsymbol{V}_0(\boldsymbol{p}); \qquad \boldsymbol{p} \in \Omega, \tag{4.27}$$

where the initial velocity field must obey the mass conservation principle

$$\nabla \cdot \boldsymbol{V}_0(\boldsymbol{p}) = 0, \tag{4.28}$$

with initial velocity $\boldsymbol{V}_0(\boldsymbol{p})$ at initial time $t^{\tau=0}$.

4.2.1 Explicit pressure Poisson solution of the momentum conservation equation

The solution of the explicit pressure Poisson algorithm is presented here for incompressible fluids. With some additional efforts in the pressure correction equation, the algorithm could be extended to compressible fluids as well. From the known velocity field at the time step t^{τ} (and initially from the

velocity field at $t^{\tau=0}$), first the pressure field is calculated by applying the divergence to the momentum conservation equation (3.15)

$$\nabla^2 P^\tau = \nabla \cdot \left(-\frac{\Delta(\varrho \boldsymbol{V})}{\epsilon\, \Delta t} - \frac{1}{\epsilon^2} \nabla \cdot (\varrho\, \boldsymbol{V}\, \boldsymbol{V}) - \frac{\mu}{\mathcal{K}} \boldsymbol{V} + \nabla \cdot (\mu_{eff}\, \nabla \boldsymbol{V}) + \boldsymbol{F} \right)^\tau \quad (4.29)$$

where $\Delta(\varrho \boldsymbol{V})^\tau = (\varrho \boldsymbol{V})^\tau - (\varrho \boldsymbol{V})^{\tau-1}$. Thus the Laplacian is produced on the left-hand side of equation (4.29), which is called the pressure Poisson equation. With explicit time stepping scheme, the Laplacian in the equation (4.29) cannot be calculated in one step. Therefore the pressure is calculated by the use of the false transient with false time t_f, which drives the pressure equation towards the steady-state

$$\begin{aligned} \frac{\partial P^\tau}{\partial t_f} &= -\nabla^2 P^\tau + \nabla \cdot \left(-\frac{\Delta(\varrho \boldsymbol{V})}{\epsilon\, \Delta t} - \frac{1}{\epsilon^2} \nabla \cdot (\varrho \boldsymbol{V} \boldsymbol{V}) - \right. \\ &\left. \frac{\mu}{\mathcal{K}} \boldsymbol{V} + \nabla \cdot (\mu_{eff}\, \nabla \boldsymbol{V}) + \boldsymbol{F} \right)^\tau . \end{aligned} \quad (4.30)$$

Equation (4.30) is solved iteratively by applying false time step Δt_f as

$$\begin{aligned} {}^{l+1}P^\tau &= {}^{l}P^\tau + \Delta t_f \left[-{}^{l}(\nabla^2 P)^\tau + \nabla \cdot {}^{l} \left(-\frac{\Delta(\varrho \boldsymbol{V})}{\epsilon\, \Delta t} - \right. \right. \\ &\left. \left. \frac{1}{\epsilon^2} \nabla \cdot (\varrho \boldsymbol{V} \boldsymbol{V}) - \frac{\mu}{\mathcal{K}} \boldsymbol{V} + \nabla \cdot (\mu_{eff}\, \nabla \boldsymbol{V}) + \boldsymbol{F} \right) \right]^\tau , \end{aligned} \quad (4.31)$$

where l denotes old internal iteration and $l+1$ new internal iteration. If the flow is not penetrating through the boundaries (confined flow), the pressure boundary conditions are Neumann boundary conditions, where equation (4.29) is multiplied by the normal on the boundary

$$\frac{\partial P^\tau}{\partial n_\Gamma} = {}^{l} \left[-\frac{\Delta(\varrho \boldsymbol{V})}{\epsilon\, \Delta t} - \frac{1}{\epsilon^2} \nabla \cdot (\varrho \boldsymbol{V} \boldsymbol{V}) - \frac{\mu}{\mathcal{K}} \boldsymbol{V} + \nabla \cdot (\mu_{eff}\, \nabla \boldsymbol{V}) + \boldsymbol{F} \right]^\tau \cdot \boldsymbol{n}_\Gamma \quad (4.32)$$

Now the pressure field and its derivatives in the internal time step are known and the intermediate velocity at the new time step can be obtained as

$$\begin{aligned} {}^{l+1}\hat{\boldsymbol{V}}^{\tau+1} &= \boldsymbol{V}^\tau + \frac{\epsilon\, \Delta t}{\varrho} \cdot \\ &\left[-\frac{1}{\epsilon^2} \nabla \cdot {}^{l} (\varrho \boldsymbol{V} \boldsymbol{V}) - {}^{l+1} (\nabla P) - \frac{\mu}{\mathcal{K}} {}^{l} \boldsymbol{V} + \nabla \cdot {}^{l} (\mu_{eff}\, \nabla \boldsymbol{V}) + \boldsymbol{F} \right]^\tau . \end{aligned} \quad (4.33)$$

The calculated velocity is denoted by $\hat{\boldsymbol{V}}$ because it does not satisfy the condition of mass conservation in general. To assure compliance with mass

conservation, pressure and velocity corrections have to be applied. Therefore the velocity $\hat{\boldsymbol{V}}$ is corrected by the velocity correction $\breve{\boldsymbol{V}}$. Written for incompressible conditions, the relation is

$$\nabla \cdot \boldsymbol{V}^{\tau+1} = \nabla \cdot (\hat{\boldsymbol{V}} + \breve{\boldsymbol{V}})^{\tau+1}. \tag{4.34}$$

Assuming that the velocity correction occurs exclusively due to the action of the pressure correction, we can write

$$\frac{\varrho}{\epsilon\,\Delta t}\breve{\boldsymbol{V}}^{\tau+1} = -\nabla \breve{P}^{\tau+1}. \tag{4.35}$$

The pressure correction $\breve{P}$ is again calculated by applying the divergence to both sides of equation (4.35), which in combination with (4.34) gives

$$\nabla^2 \breve{P}^{\tau+1} = -\frac{1}{\epsilon\,\Delta t}\nabla \cdot (\varrho\,\hat{\boldsymbol{V}})^{\tau+1}. \tag{4.36}$$

The natural boundary conditions for pressure correction are Neumann boundary conditions

$$\frac{\partial \breve{P}^{\tau+1}}{\partial n_\Gamma} = 0. \tag{4.37}$$

Now, in a similar manner to the pressure equation we can calculate the pressure correction by means of the false transient as

$$\frac{\partial \breve{P}^{\tau+1}}{\partial t_f} = -^{l}(\nabla^2 \breve{P})^{\tau+1} - \frac{1}{\epsilon\,\Delta t}\nabla \cdot^{l+1} (\varrho\,\hat{\boldsymbol{V}})^{\tau+1}. \tag{4.38}$$

Explicitly the equation (4.38) is calculated according to

$$^{l+1}\breve{P}^{\tau+1} =^{l} \breve{P}^{\tau+1} + \Delta t_f \left[-^{l}(\nabla^2 \breve{P}) - \frac{1}{\epsilon\,\Delta t}\nabla \cdot^{l+1} (\varrho\,\hat{\boldsymbol{V}}) \right]^{\tau+1}. \tag{4.39}$$

After the solution for pressure correction and its derivatives are obtained, the pressure and velocity fields are updated

$$^{l}P = {}^{l+1}P +^{l+1} \breve{P} \tag{4.40}$$

$$^{l}\boldsymbol{V}^{\tau+1} = {}^{l+1}\hat{\boldsymbol{V}}^{\tau+1} -^{l+1} \nabla \breve{P}\frac{\epsilon\,\Delta t}{\varrho}. \tag{4.41}$$

In order to proceed to the next time step, the conditions

$$\eta_P = |^{l+1}P^{\tau+1} -^{l} P^{\tau+1}| < \eta_{P,lim}, \tag{4.42}$$

$$\eta_V = |^{l+1}V^{\tau+1} -^{l} V^{\tau+1}| < \eta_{V,lim}, \tag{4.43}$$

have to be satisfied and $\eta_{P,lim}$ and $\eta_{V,lim}$ are internal measures of convergence for pressure and velocity, respectively. If the condition of internal convergence is not met, than we return to equation (4.31). The entire procedure is presented in Figure 4.1.

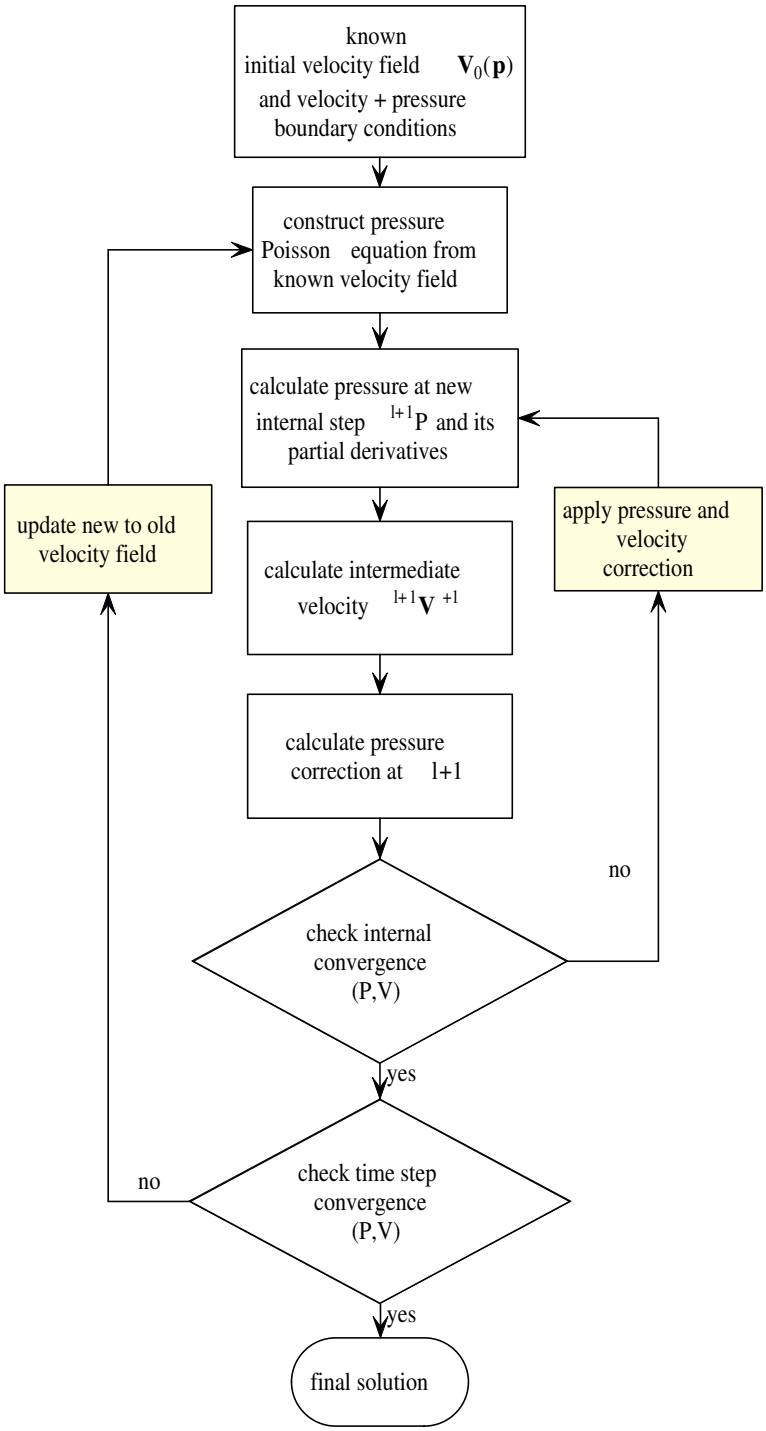


Figure 4.1: Flowchart for pressure Poisson solution of momentum equation.

Implementation of Darcy equation

In the previous section the solution procedure is presented for Navier-Stokes and similar non-linear equations. The Solution of the Darcy equation is slightly different. Recalling equation (3.11) we observe that the relation between the pressure gradient and the velocity is linear. Therefore the solution procedure is simplified because the velocity field is evaluated directly from the linear relation with the pressure gradient. Instead of an iterative solution of equation (4.33), velocity $\hat{\boldsymbol{V}}$ is obtained via the relation

$$^{l+1}\hat{\boldsymbol{V}} = \frac{\mathcal{K}}{\mu}(-\nabla P + \boldsymbol{F})^{l}. \tag{4.44}$$

The velocity and pressure correction procedures described below are the same as explained above. Since the Darcy equation is not transient, the time loop in Figure 4.1 is obsolete. Again in this case we have to solve the pressure equation by taking a divergence on both sides.

4.2.2 CBS solution of the momentum conservation equation

Many algorithms exist for solution of the momentum equation (SIMPLE, SIMPLER, SIMPLEC, PISO, etc) that are usually associated with a specific numerical method. Most of them are in semi-, quasi-, and fully implicit form, and are applicable in most cases to the problems of incompressible fluids. The CBS algorithm is well applicable for fully compressible flows in both explicit and semi-implicit forms. A further advantage of the CBS algorithm is the possibility of solving subsonic and supersonic flows using the same formulation. The solution of the momentum conservation equation by CBS algorithm is explained in this chapter. The solution procedure starts again with known velocity field and guessed pressure field at initial time with appropriate boundary conditions. If the variable ϕ is replaced by the mass flux $\varrho\boldsymbol{v} = \varrho\boldsymbol{V}/\epsilon$ and $\Psi(\phi) = -\nabla P - 1/\epsilon^2\,\nabla\cdot(\varrho\boldsymbol{V}\boldsymbol{V}) - \frac{\mu}{\mathcal{K}}\boldsymbol{V} + \nabla\cdot(\mu_{eff}\,\nabla\boldsymbol{V}) + \boldsymbol{F}$ into equation (4.26), we obtain the following solution

$$\begin{aligned} ^{l+1}\Delta(\widehat{\varrho\boldsymbol{V}})^{\tau+1} = {}^{l+1}(\widehat{\varrho\boldsymbol{V}})^{\tau+1} - (\varrho\boldsymbol{V})^{\tau} = \\ \epsilon\,\Delta t\,{}^{l}\left[-\nabla\bar{P} - \tfrac{\mu}{\mathcal{K}}\boldsymbol{V} - \tfrac{1}{\epsilon^2}\nabla\cdot(\varrho\boldsymbol{V}\boldsymbol{V}) + \nabla\cdot(\mu_{eff}\,\nabla\boldsymbol{V}) + \boldsymbol{F}\right]^{\tau+1} \\ -\tfrac{\Delta t^2}{2}{}^{l}\boldsymbol{V}^{\tau}\nabla\cdot{}^{l}\left[-\nabla\bar{P} - \tfrac{\mu}{\mathcal{K}}\boldsymbol{V} - \tfrac{1}{\epsilon^2}\nabla\cdot(\varrho\boldsymbol{V}\boldsymbol{V}) + \nabla\cdot(\mu_{eff}\,\nabla\boldsymbol{V}) + \boldsymbol{F}\right]^{\tau+1}. \end{aligned} \tag{4.45}$$

where $\Delta(\widehat{\varrho\boldsymbol{V}})$ is difference between mass flux per unit area in new and old time step and $\widehat{\varrho\boldsymbol{V}}$ is an intermediate value. Again $l+1$ and l represent new

and old intermediate iterative steps, respectively. The pressure gradient term $\nabla \bar{P}$ is defined as

$$\nabla \bar{P} = \iota \, \nabla P^{\tau+1} + (1 - \iota) \, \nabla P^{\tau}, \tag{4.46}$$

where parameter ι is 1 for implicit solution and 0 for explicit solution which is used in this work. At the initial time step $t^{\tau=0}$, the pressure gradient $\nabla \bar{P}$ is set to 0. Next, density change is obtained from mass conservation equation (3.9). Using temporal discretization of equation (3.9) and after rearranging, the relation for density change is

$${}^{l+1}\Delta \, \varrho^{\tau+1} = -\frac{\Delta t}{\epsilon} \nabla \cdot {}^{l+1}(\varrho \boldsymbol{V})^{\tau+\iota_1}. \tag{4.47}$$

The term $(\varrho \boldsymbol{V})^{\tau+\iota_1}$ is defined as

$${}^{l+1}(\varrho \boldsymbol{V})^{\tau+\iota_1} = \iota_1 \, {}^{l+1}(\varrho \boldsymbol{V})^{\tau+1} + (1 - \iota_1)^{l+1}(\varrho \boldsymbol{V})^{\tau}, \tag{4.48}$$

and ι_1 ranges between $0.5 < \iota_1 < 1$. The correction of velocities is again obtained from the pressure gradient

$${}^{l+1}(\varrho \boldsymbol{V})^{\tau+1} - {}^{l+1}\,(\widehat{\varrho \boldsymbol{V}})^{\tau+1} = \epsilon \, \Delta t \, {}^{l}\nabla \bar{P}. \tag{4.49}$$

Now inserting equations (4.48) and (4.49) into (4.47) gives for the explicit scheme

$$\begin{aligned} {}^{l+1}\Delta \, \varrho^{\tau+1} \; = -\tfrac{\Delta t}{\epsilon} \Big[\iota_1 \, \nabla \cdot {}^{l+1}(\widehat{\varrho \boldsymbol{V}})^{\tau+1} + (1 - \iota_1) \, \nabla \cdot {}^{l+1}\Delta(\varrho \boldsymbol{V})^{\tau+1} \Big] \\ + \Delta t^2 \, \iota_1 \, {}^{l}\nabla^2 P^{\tau}. \end{aligned} \tag{4.50}$$

Pressure change in one time step is obtained through density change and speed of sound c_s

$${}^{l+1}\Delta P^{\tau+1} = {c_s}^2 \, {}^{l+1}\Delta \varrho^{\tau+1}. \tag{4.51}$$

For real compressible fluids speed of sound c_s depends on pressure P and density ϱ.

$$c_s^2 = \frac{\partial P}{\partial \varrho} = \frac{\kappa \, P}{\varrho}, \tag{4.52}$$

where κ is the ratio of specific heats equal to c_p/c_v. For a fluid with small compressibility it holds that

$$c_s^2 = \frac{K}{\varrho}, \tag{4.53}$$

where K is the bulk modulus. The appropriate definition of c_s is case dependent. Naturally for incompressible fluids ($\Delta \varrho \to 0$), coefficient c_s must limit

toward infinity. Therefore, in the explicit scheme, it is impossible to implement relation (4.51). In order to solve this problem, Nithiarasu proposed an artificial compressibility (γ) instead of an infinite speed of sound [Nithiarasu, 2003]. The artificial compressibility γ is defined in [Massarotti *et al.*, 2004] to be of the following form

$$\gamma = max(\varepsilon, v_{conv}, v_{dif}), \tag{4.54}$$

where ε is a constant taken as 0.5, v_{conv} is the convective velocity and v_{diff} is the diffusive velocity.

$$v_{conv} = \sqrt{V\,V}, \tag{4.55}$$

$$v_{diff} = \frac{2}{h\,\mathrm{Re}}, \tag{4.56}$$

with h being minimum local element size.

On the basis of many tests, all results in this work were calculated using my own defined artificial compressibility with value

$$\gamma = \sqrt{\frac{\nu}{\Delta t}}. \tag{4.57}$$

Relation (4.57) has two advantages in comparison to the relation given in (4.56). The first, our relation is dimensionally consistent as opposed to (4.56). The second, minimum size of element, used in (4.56) has no meaning in meshless methods (or it can be the minimal distance between the nodes).

Implementation of Darcy equation

For time dependent Darcy equation we can write

$$\frac{1}{\epsilon}\frac{\partial(\varrho\boldsymbol{V})}{\partial t} = -\nabla P - \frac{\mu}{\mathcal{K}}\boldsymbol{V} + \boldsymbol{F}. \tag{4.58}$$

Similar to the procedure shown for the more general momentum equation the mass flow $\varrho\boldsymbol{V}/\epsilon$ is introduced for general variable ϕ and $\Psi(\phi) = -\nabla P - \frac{\mu}{\mathcal{K}}\boldsymbol{V} + \boldsymbol{F}$

$$\begin{aligned} {}^{l+1}\Delta(\widehat{\varrho\boldsymbol{V}}) &= {}^{l+1}(\widehat{\varrho\boldsymbol{V}})^{\tau+1} - (\varrho\boldsymbol{V})^{\tau} = \\ &\quad \epsilon\,\Delta t\,{}^{l}\left[-\nabla\bar{P} - \frac{\mu}{\mathcal{K}}\boldsymbol{V} + \boldsymbol{F}\right]^{\tau+1} \\ &- \quad \frac{\Delta t^2}{2}\boldsymbol{V}\nabla\cdot{}^{l}\left[\nabla P + \frac{\mu}{\mathcal{K}}\boldsymbol{V} - \boldsymbol{F}\right]^{\tau+1}. \end{aligned} \tag{4.59}$$

Solution procedure The CBS algorithm can be implemented in two ways: In Split 1 the pressure term is retained at its previous time step value, while in Split 2 the pressure term is completely removed from equation (4.45). Although the procedure in both cases is very similar, we will focus on Split 1. The solution procedure is constituted from three general steps

- **Step 1:** Calculate intermediate momentum equation (4.45)
- **Step 2:** Calculate density difference (4.50) and pressure difference (4.51)
- **Step 3:** Apply velocity correction from equation (4.49)

More precisely, for the explicit time scheme, the procedure goes

1. Set initial values in new time step ${}^{l}P^{\tau+1}$ and ${}^{l}\boldsymbol{V}^{\tau+1}$ to old values P^{τ} and $\boldsymbol{V}^{\tau}$.
2. Calculate pressure gradients ∇P from known pressure field in internal iteration l.
3. Calculate velocity gradients $\nabla \boldsymbol{V}$ and derivatives of convective term $\nabla \cdot (\varrho \boldsymbol{V}\boldsymbol{V})$ from ${}^{l}\mathbf{V}^{\tau+1}$.
4. Obtain partial derivatives of stabilization term $\nabla\Psi(\phi)$.
5. Calculate partial derivatives of $\Delta(\widehat{\varrho \boldsymbol{V}})$ from equation (4.45).
6. From equations (4.50) (for $\iota = 0$) and (4.51) obtain ΔP.
7. Update ${}^{l+1}P^{\tau+1} = {}^{l}P^{\tau+1} + \Delta P$.
8. Update ${}^{l+1}\boldsymbol{V}^{\tau+1} = \boldsymbol{V}^{\tau} + {}^{l+1}\Delta(\varrho \boldsymbol{V})^{\tau+1}/\varrho$.
9. Repeat from step 2 internally until the velocity difference $\eta_V = |{}^{l+1}\boldsymbol{V}^{\tau+1} - {}^{l}\boldsymbol{V}^{\tau+1}|$ is smaller than prescribed convergence criteria $\eta_{V,lim}$.
10. Proceed to the next time step.

If steady-state is needed the algorithm is repeated until $|\boldsymbol{V}^{\tau+1} - \boldsymbol{V}^{\tau}| < \eta_{V,con}$ and $\eta_{V,con}$ is the steady-state criterion.

5 Diffuse Approximate Method (DAM)

DAM belongs to the class of Moving Least Squares (MLS) methods. MLS, originated by mathematicians for data fitting and surface reconstruction, is often termed also local regression and loss. It can be categorized as a method of finite series representation of functions. MLS methods have been gaining popularity as an alternative for constructing meshless shape functions in recent years because it is generally considered to be one of the best schemes to interpolate data with reasonable accuracy [Atluri and Shen, 2002].

5.1 MLS Approximation

A subdomain ${}_n\Omega$, which is a neighborhood of a node $\boldsymbol{p}_n$ and is located within the problem domain Ω, is denoted as the domain of influence of node $\boldsymbol{p}_n$ (see Figure 5.1). To approximate the distribution of function ϕ in ${}_n\Omega$ over

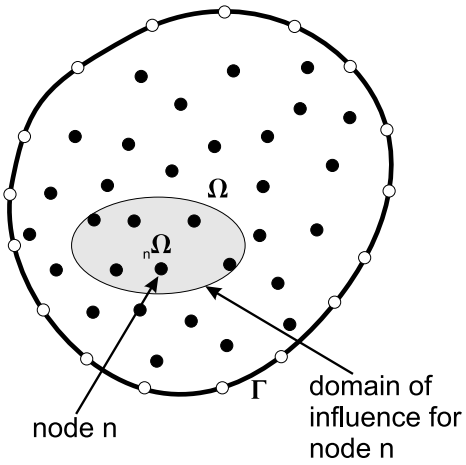


Figure 5.1: The problem domain with the domain of influence.

arbitrary located nodes $\boldsymbol{p}_i$; $i = 1, 2, ..., I$, where I is the number of nodes in the domain of influence ${}_n\Omega$, the MLS approximant $\hat{\phi}(\boldsymbol{p})$ at node $\boldsymbol{p}$ can be defined by

$$\hat{\phi}(\boldsymbol{p}) = \boldsymbol{\varphi}_k(\boldsymbol{p} - \boldsymbol{p}_i)^T \, \boldsymbol{\alpha}(\boldsymbol{p}), \tag{5.1}$$

In DAM, the basis is formed by polynomials. Therefore, $\boldsymbol{\varphi}_k(\boldsymbol{p}-\boldsymbol{p}_i) = [\varphi_1(\boldsymbol{p}-\boldsymbol{p}_i), \varphi_2(\boldsymbol{p}-\boldsymbol{p}_i), ..., \varphi_K(\boldsymbol{p}-\boldsymbol{p}_i)]$ is a complete monomial basis, K is the number of terms in the basis. For example for the 2D case φ_k are $\varphi_1 = 1$; $\varphi_2 = (p_x - p_{x\,i})$; $\varphi_3 = (p_y - p_{y\,i})$; $\varphi_4 = (p_x - p_{x\,i})^2$; $\varphi_5 = (p_x - p_{x\,i})\,(p_y - p_{y\,i})$; $\varphi_6 = (p_y - p_{y\,i})^2$, i.e. $K = 6$. The coefficients $\alpha_k(\boldsymbol{p})$ are obtained via minimization of a weighted functional

$$\mathcal{I}(\alpha_k(\boldsymbol{p})) = \sum_{i=1}^{I} {}_n\widehat{W}(\|\boldsymbol{p} - \boldsymbol{p}_i\|) \sum_{k=1}^{K} \Big[\alpha_k(\boldsymbol{p})\, \varphi_k(\|\boldsymbol{p} - \boldsymbol{p}_i\|) - \hat{\phi}(\boldsymbol{p}_i)\Big]^2, \tag{5.2}$$

${}_n\widehat{W}(\|\boldsymbol{p} - \boldsymbol{p}_I\|)$ is the weight function associated with the node n and is explained in more detail in Chapter 7.1. $\|\boldsymbol{p} - \boldsymbol{p}_I\|$ denotes L^2 norm. The minimization of $\mathcal{I}$ in equation (5.2) with respect to $\alpha_k(\boldsymbol{p})$ leads to the following linear relation between $\boldsymbol{\alpha}$ and $\hat{\boldsymbol{\phi}}$

$$\mathbf{A}\,\boldsymbol{\alpha} = \hat{\boldsymbol{\phi}}. \tag{5.3}$$

For each $n'th$ calculated node the left-hand side $K \times K$ matrix $\mathbf{A}$ and right-hand side vector $\hat{\boldsymbol{\phi}}$ are written in explicit form as

$${}_nA_{jk} = \sum_{i=1}^{I} \varphi_k(\boldsymbol{p}_i - \boldsymbol{p}_n)\, {}_n\widehat{W}(\boldsymbol{p}_i - \boldsymbol{p}_n)\, \varphi_j(\boldsymbol{p}_i - \boldsymbol{p}_n), \tag{5.4}$$

while vector ${}_nb_j$ on the right-hand side consists of

$${}_n\hat{\phi}_j = \sum_{i=1}^{I} \varphi_j(\boldsymbol{p}_i - \boldsymbol{p}_n)\, {}_n\widehat{W}(\boldsymbol{p}_i - \boldsymbol{p}_n)\, \hat{\phi}(\boldsymbol{p}_i). \tag{5.5}$$

The MLS approximation is well defined only when the matrix $\mathbf{A}$ is non-singular. Thus, a necessary condition for a well-defined MLS approximation is that the number of nodes in the domain of influence I is larger or equal to the number of the basis elements K, i.e. $I \geq K$.

5.2 From MLS approximation to DAM

The mathematical tool described in Section 5.1 can be effectively used in finding the solution of partial differential equations. For each node $\boldsymbol{p}_n \in {}_n\Omega$

we solve the system of equations (5.3). The values of function $\hat{\phi}$ are then calculated at each node $\boldsymbol{p}_n$ from the relation (5.1). Since the solution is sought exactly at node $\boldsymbol{p}_n$, the polynomial basis is

$$\boldsymbol{\varphi} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{5.6}$$

which gives us a simple relation for function value at node $\boldsymbol{p}_n$

$$\hat{\phi}(\boldsymbol{p}_n) = \alpha_1. \tag{5.7}$$

The first derivatives from known values α_k are calculated with the known derivatives of the basis functions. Hence we can write

$$\frac{\partial \hat{\phi}(\boldsymbol{p}_n)}{\partial \xi} = \frac{\partial \boldsymbol{\varphi}_k(\boldsymbol{p}_n - \boldsymbol{p}_n)^T}{\partial \xi}\, \boldsymbol{\alpha}(\boldsymbol{p}_n); \qquad \xi \in \{x, y\}, \tag{5.8}$$

with first-order polynomial basis partial derivatives at node $\boldsymbol{p}_n$

$$\frac{\partial \boldsymbol{\varphi}}{\partial x} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \frac{\partial \boldsymbol{\varphi}}{\partial y} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{5.9}$$

and the consequent derivatives

$$\frac{\partial \hat{\phi}(\boldsymbol{p}_n)}{\partial x} = \alpha_2, \qquad \frac{\partial \hat{\phi}(\boldsymbol{p}_n)}{\partial y} = \alpha_3. \tag{5.10}$$

For the second derivatives similarly

$$\frac{\partial^2 \hat{\phi}(\boldsymbol{p}_n)}{\partial \xi \partial \zeta} = \frac{\partial^2 \boldsymbol{\varphi}_k(\boldsymbol{p}_n - \boldsymbol{p}_n)^T}{\partial \xi \partial \zeta}\, \boldsymbol{\alpha}(\boldsymbol{p}_n); \qquad \xi, \zeta \in \{x, y\}, \tag{5.11}$$

with

$$\frac{\partial^2 \boldsymbol{\varphi}}{\partial x^2} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 2 \\ 0 \\ 0 \end{bmatrix}, \qquad \frac{\partial^2 \boldsymbol{\varphi}}{\partial x \partial y} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \qquad \frac{\partial^2 \boldsymbol{\varphi}}{\partial y^2} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 2 \end{bmatrix}. \tag{5.12}$$

with the derivatives

$$\frac{\partial^2 \hat{\phi}(\boldsymbol{p}_n)}{\partial x^2} = 2\,\alpha_4, \qquad \frac{\partial^2 \hat{\phi}(\boldsymbol{p}_n)}{\partial x \partial y} = \alpha_5, \qquad \frac{\partial^2 \hat{\phi}(\boldsymbol{p}_n)}{\partial y^2} = 2\,\alpha_6. \tag{5.13}$$

Considering a general partial differential equation with L as an arbitrary differential operator and B as an operator imposed for a chosen boundary condition we can write

$$\frac{\partial \phi(\boldsymbol{p}, t)}{\partial t} = L\,\phi(\boldsymbol{p}, t) + f(\boldsymbol{p}, t); \quad \boldsymbol{p} \in \Omega, \qquad B\,\phi = g(\boldsymbol{p}, t); \quad \boldsymbol{p} \in \Gamma. \tag{5.14}$$

The solution of equation (5.14) is obtained by iterative procedure, with appropriate replacement of differential operator L by α-s obtained from MLS approximation. An example of the explicit solution of the Laplace equation is given below.

Example: Solution of the Laplace equation The Laplace equation is defined as

$$\nabla^2 \phi(\boldsymbol{p}) = 0. \tag{5.15}$$

In the explicit time scheme, the solution should be obtained by a false transient such as

$$\frac{\partial \phi(\boldsymbol{p})}{\partial t_f} = \nabla^2 \phi(\boldsymbol{p}). \tag{5.16}$$

Operator L in the Laplace equation from equation (5.14) is in 2D then

$$L = \frac{\partial^2 \phi(\boldsymbol{p})}{\partial x^2} + \frac{\partial^2 \phi(\boldsymbol{p})}{\partial y^2} \tag{5.17}$$

After determination of initial and boundary conditions the solution procedure is the following:

1. From the values defined by the initial conditions obtain coefficients α_k in the first node, say in $\boldsymbol{p}_1$ from domain of influence ${}_1\Omega$.
2. Calculate new value, e.g. $\phi^{\tau+1}(\boldsymbol{p}_1) = \phi^{\tau}(\boldsymbol{p}_1) + \Delta t_f \left(\frac{\alpha_4}{2} + \frac{\alpha_6}{2}\right)$.
3. Go to the next node, say $\boldsymbol{p}_2$.
4. Repeat this procedure until the last node in calculation domain N.
5. Update old value $\phi^{\tau}(\boldsymbol{p}_n) = \phi^{\tau+1}(\boldsymbol{p}_n)$, $n = 1, 2, ..., N$.
6. Repeat until $\max\{|\phi^{\tau}(\boldsymbol{p}_n) - \phi^{\tau+1}(\boldsymbol{p}_n)|\} < \eta$ is met, where η is the convergence criterion.

5.2.1 Treatment of boundary conditions

The boundary can in general be divided into Γ_D, Γ_N, and Γ_R Dirichlet, Neumann and Robin type boundary conditions, respectively. These boundary conditions are at boundary node $\boldsymbol{p}$ with normal n_Γ defined through known functions ϕ^D_Γ, ϕ^N_Γ, ϕ^R_Γ, and reference value $\phi^R_{\Gamma,ref}$

$$\begin{aligned} \phi(\boldsymbol{p},t) &= \phi^D_\Gamma(\boldsymbol{p},t); \qquad \boldsymbol{p}\in\Gamma^D && (5.18)\\ \frac{\partial\phi(\boldsymbol{p},t)}{\partial n_\Gamma} &= \phi^N_\Gamma(\boldsymbol{p},t); \qquad \boldsymbol{p}\in\Gamma^N && (5.19)\\ \frac{\partial\phi(\boldsymbol{p},t)}{\partial n_\Gamma} &= \phi^R_\Gamma(\boldsymbol{p},t)[\phi(\boldsymbol{p},t)-\phi^R_{\Gamma,ref}(\boldsymbol{p},t)]; \qquad \boldsymbol{p}\in\Gamma^R. && (5.20) \end{aligned}$$

Dirichlet boundary conditions Dirichlet boundary conditions do not need any special treatment since the values $\phi(\boldsymbol{p})$; $\boldsymbol{p}\in\Gamma^D$ are prescribed. Usually they are omitted from the calculation in order to save CPU time.

Neumann boundary conditions With Neumann boundary conditions, however, the derivatives of the function $\phi(\boldsymbol{p})$ over the normal are known. From the given derivatives, we need to obtain the function values. Functional (5.2) is thus rewritten to

$$\mathcal{I}(\alpha_k(\boldsymbol{p})) = \sum_{i=1}^{I}\left[{}_n\widehat{W}(\|\boldsymbol{p}-\boldsymbol{p}_i\|)\sum_{k=1}^{K}\alpha_k(\boldsymbol{p})\,\frac{\partial\varphi_k(\|\boldsymbol{p}-\boldsymbol{p}_i\|)}{\partial n_\Gamma}-\hat{\phi}(\boldsymbol{p}_i)\right]^2, \quad (5.21)$$

with left hand side matrix $\mathbf{A}$ now being

$${}_nA_{jk} = \sum_{i=1}^{I}\frac{\partial\varphi_j(\boldsymbol{p}_i-\boldsymbol{p}_n)}{\partial n_\Gamma}\,{}_n\widehat{W}(\boldsymbol{p}_i-\boldsymbol{p}_n)\,\frac{\partial\varphi_k(\boldsymbol{p}_i-\boldsymbol{p}_n)}{\partial n_\Gamma}, \quad (5.22)$$

and right-hand side vector ${}_nb_j$

$${}_n\hat{\phi}_j = \sum_{i=1}^{I}\frac{\partial\varphi_j(\boldsymbol{p}_i-\boldsymbol{p}_n)}{\partial n_\Gamma}\,{}_n\widehat{W}(\boldsymbol{p}_i-\boldsymbol{p}_n)\,\phi^N_\Gamma(\boldsymbol{p}_i). \quad (5.23)$$

Function value $\phi(\boldsymbol{p})$ in each node is then represented as α_1.

Robin boundary conditions Robin boundary conditions can be implemented in two ways. The first way is to implement them as Neumann boundary conditions. Left-hand side matrix $\mathbf{A}$ is constructed the same as in equation (5.22). The right-hand side vector, however, is constructed from the

relation described in (5.20)

$$ {}_n\hat{\phi}_j = \sum_{i=1}^{I} \frac{\partial \varphi_j(\boldsymbol{p}_i - \boldsymbol{p}_n)}{\partial n_\Gamma} \, {}_n\widehat{W}(\boldsymbol{p}_i - \boldsymbol{p}_n) \, \phi_\Gamma^R(\boldsymbol{p}_i)[\phi(\boldsymbol{p}_i) - \phi_{\Gamma,ref}^R(\boldsymbol{p}_i)]. \tag{5.24} $$

Since the value $\phi(\boldsymbol{p}_i)$ in this case can only be the value from the previous iteration, it is necessary to obtain the solution iteratively. To circumvent this problem the Robin boundary conditions can be derived similar to the one explained in equation (5.2)

$$ \mathcal{I}(\alpha_k(\boldsymbol{p})) = \sum_{i=1}^{I} {}_n\widehat{W}(\|\boldsymbol{p} - \boldsymbol{p}_i\|) \sum_{k=1}^{K} \big[\phi_\Gamma^R \, (\alpha_k \, \varphi_k(\|\boldsymbol{p} - \boldsymbol{p}_i\|) - \phi_{\Gamma,ref}^R) - \alpha_k(\boldsymbol{p}) \frac{\partial \varphi_k(\|\boldsymbol{p} - \boldsymbol{p}_i\|)}{\partial n_\Gamma} \Big]^2 . \tag{5.25} $$

By the derivation of equation (5.26) the left-hand side expression is obtained

$$ \begin{aligned} {}_nA_{jk} \;=\; & \sum_{i=1}^{I} \left(\frac{\partial \varphi_j(\boldsymbol{p}_i - \boldsymbol{p}_n)}{\partial n_\Gamma} + \phi_\Gamma^R \, \varphi_j(\boldsymbol{p}_i - \boldsymbol{p}_n) \right) {}_n\widehat{W}(\boldsymbol{p}_i - \boldsymbol{p}_n) \cdot \\ & \cdot \left(\frac{\partial \varphi_k(\boldsymbol{p}_i - \boldsymbol{p}_n)}{\partial n_\Gamma} + \phi_\Gamma^R \, \varphi_k(\boldsymbol{p}_i - \boldsymbol{p}_n) \right), \end{aligned} \tag{5.26} $$

and the right-hand side vector

$$ {}_n\hat{\phi}_j = \sum_{i=1}^{I} \left(\frac{\partial \varphi_j(\boldsymbol{p}_i - \boldsymbol{p}_n)}{\partial n_\Gamma} + \phi_\Gamma^R \, \varphi_j(\boldsymbol{p}_i - \boldsymbol{p}_n) \right) {}_n\widehat{W}(\boldsymbol{p}_i - \boldsymbol{p}_n) \, \phi_\Gamma^R \, \phi_{\Gamma,ref}. \tag{5.27} $$

In the latter case no iterating is required. The drawback, however, is that the left-hand side matrix $\boldsymbol{A}$ has to be updated in each iteration.

6 Numerical examples

Several cases are presented in this section to provide an idea of the accuracy, stability and ability of DAM to solve a large spectrum of physically and numerically involved transport problems. In each selected example a certain property of DAM is exposed.

In the first example the level of numerical diffusion is analyzed. Next, the solution of unconfined flow through porous media is shown and compared to a commercial package in order to prove the consistency of the method. Transport of decaying species is the next problem that gained our attention. On this simple 1D problem the stability of different time stepping schemes is analyzed and compared against the analytical solution. In addition the behavior of different methods in terms of grid density is analyzed. The combination of two previously mentioned problems, unconfined flow and radionuclide transport, gives a basis for a more realistic case of performance assessment and safety analysis of the radioactive waste repository. Next, the porous media flow is coupled to the energy equation. The problem of natural convection in a rectangular enclosure is often used for the verification of numerical methods. The two already mentioned momentum-mass conservation coupling algorithms are compared. Finally, the momentum, energy, and species conservation equations for calculation of the double diffusive natural convection in a composite fluid-porous layer are considered in the last example.

All considered numerical examples in this chapter use uniform node arrangements. This is done for the reason of comparison of the results, because often the results are compared to FVM-based solutions. All nodes are arranged as shown in Figure 6.1. The grid in Figure 6.1 is denoted through the whole work by 3×3 although the overall number of nodes is $N_x \times N_y + 2\,N_x\,N_y$ where N_x and N_y represent the number of nodes in x and y direction of Cartesian coordinate, respectively. Thus for the shown 3×3 problem the number of nodes is $9 + 4 \cdot 3 = 21$.

When more appropriate, the distance between nodes is given. For a uniform grid, the spacing is denoted in length units $\Delta x = \Delta y$. For the

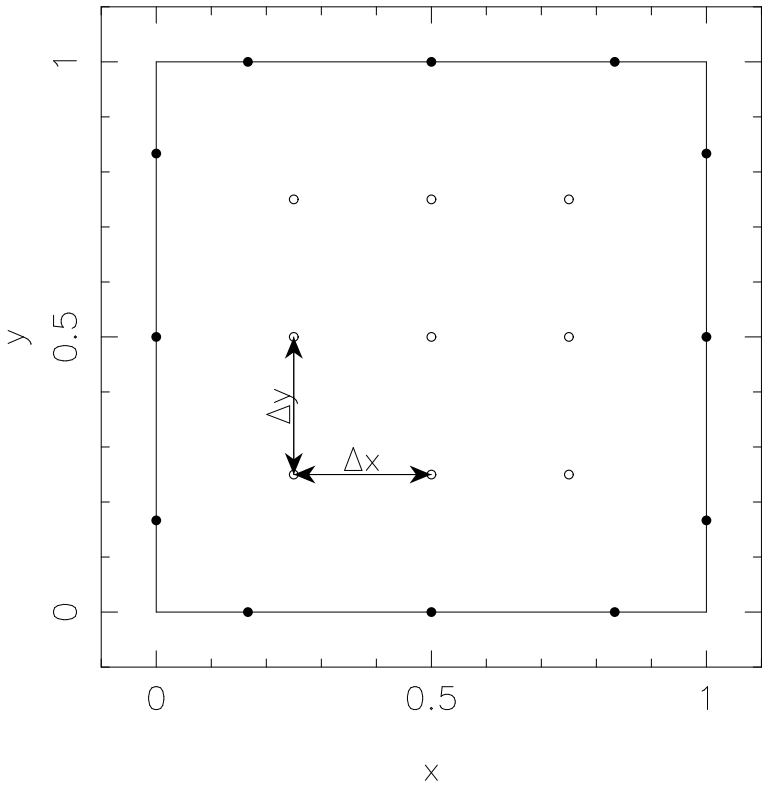


Figure 6.1: Definition of uniform 3×3 grid in DAM.

geometry in Figure 6.1, the spacing is $\Delta x = \Delta y = 0.25$ (the units of Δx and Δy are the same as the units of x and y). The number of nodes in the domain of influence is 9 in all cases and the corresponding free parameter c used in Gaussian weight function $\exp(-c\,r)$ is 15 where r is the radius of influence.

In most cases, especially in radionuclide transport calculations, a comparison is made with a commercial package [Porflow, 2001]. The Porflow code was used on the computer hardware of ARAO within the framework of the PA/SA assessment for the Slovenian LILW repository. Porflow is an FVM-based software tool for solution of multi-phase fluid flow, heat transfer, and mass transport problems in variably saturated porous or fractured media. The pre-processor is designed to allow control of data input and output through simple, format-free, English-like commands. Porflow can be used to simulate transient or steady state problems in Cartesian or cylindrical geometry. The geologic medium may be anisotropic and heterogenous, and may contain distinct embedded elements such as discrete fractures or boreholes within a porous matrix. The method of Nodal Point Integration is employed for integration of the governing differential equations by temporal and spatial

discretization over each control volume (polygon) of the physical domain. It leads to solutions that automatically conserve fluid, heat, and mass locally within every polygon, as well as for the entire flow domain. The source terms are approximated by a modified Newton-Raphson method. The polygons used to define the problem geometry can vary in size, but their shape is restricted to that of a quadrilateral, hexahedral or segment of a cylinder. For our reference solutions only quadrilateral elements are used.

For comparison of 1D radionuclide transport a Hydrus-1D package was used [Simunek *et al.*, 1999]. The Hydrus program is an FEM-based code for simulating the one-dimensional movement of water, heat, and multiple solutes in variably saturated media. The program numerically solves the Richard equation for saturated-unsaturated water flow and Fickian-based advection dispersion equations for heat and solute transport. The governing flow and transport equations are solved numerically using Galerkin-type linear finite element schemes. Integration in time is achieved using an implicit (backwards) finite difference scheme for both saturated and unsaturated conditions. Additional measures are taken to improve solution efficiency for transient problems, including automatic time step adjustment and adherence to preset ranges of the Courant and Peclet numbers. The water content term is evaluated using the mass conservation method proposed by [Celia *et al.*, 1990]. Possible options for minimizing numerical oscillations in the transport solutions include upstream weighing, artificial dispersion, and/or performance indexing.

6.1 Smith and Hutton problem for examination of numerical or artificial diffusion

Numerical or artificial diffusion is the unintentional smoothing of gradients associated with the discretization of the transport equations. Numerical diffusion leads to unphysical gradient smoothing, which is cumulative over time and is an irreversible process. The study of numerical diffusion is a very important issue. Its importance is based on the fact that all advective transport processes (excluding processes in a vacuum) include molecular diffusivity and dispersivity. In many cases, especially in slow processes such as radionuclide transport, physical diffusion plays a key role in transport. If the numerical diffusivity is in the same range or even larger than the physical diffusivity, the errors can be considerable. Therefore it is essential to determine the influence of numerical diffusion. In traditional methods numerical diffusion is larger if the direction of advection is not aligned with the orientation of polygon boundaries. The usual observation is that numerical diffusion is proportional to the grid size. The level of numerical diffusion in meshless methods is analyzed by the Smith and Hutton problem. FVM and DAM calculations are performed and compared for different grid densities.

6.1.1 Problem description

A good numerical test problem to evaluate various advection-diffusion schemes is posed in [Smith and Hutton, 1982]. The Smith and Hutton problem is to find the steady-state solution of the convection-diffusion equation

$$\nabla\phi = \frac{1}{\mathrm{Pe}}\nabla^2\phi \tag{6.1}$$

on the rectangular region shown in Figure 6.2. The amount of diffusion is controlled by the Peclet number (Pe), defined as

$$\mathrm{Pe} = \frac{L\,V}{\alpha_{th}}, \tag{6.2}$$

with characteristic length L, velocity V and thermal diffusivity α_{th}. The velocity field is derived from the stream function $\psi = (1-p_x^2)(1-p_y^2)$. From the known stream function the velocity component in the x direction

$$V_x(\boldsymbol{p}) = -\frac{\partial\psi}{\partial p_y} = 2\,p_y\,(1-p_x^2) \tag{6.3}$$

and in the y direction

$$V_y(\boldsymbol{p}) = \frac{\partial\psi}{\partial p_x} = -2\,p_x\,(1-p_y^2) \tag{6.4}$$

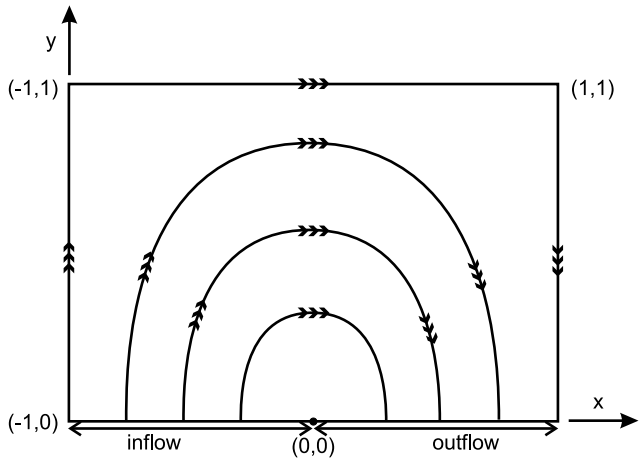


Figure 6.2: Schematics of the Smith and Hutton problem.

can be defined. All boundary conditions except the outflow boundary condition are of the Dirichlet type. The value of variable $\phi(\boldsymbol{p})$ is observed at the outflow boundary, thus having the Neumann boundary condition. The value of $\phi(\boldsymbol{p})$ is specified on the inflow boundary as

$$\phi(\boldsymbol{p}) = 1 + \tanh[\Upsilon\,(p_x + 1)];\ p_y = 0,\ -1 \leq p_x \leq 0. \tag{6.5}$$

On all other boundaries, except for the outflow portion, $\phi(\boldsymbol{p})$ is specified

$$\phi(\boldsymbol{p}) = 1 - \tanh(\Upsilon), \tag{6.6}$$

where Υ is an adjustable parameter that controls the "sharpness" of the rise of the profile at the inflow boundary. The value $\Upsilon = 10$ is used in the present example. Inflow function is rather sharp as shown in Figure 6.3. By the time

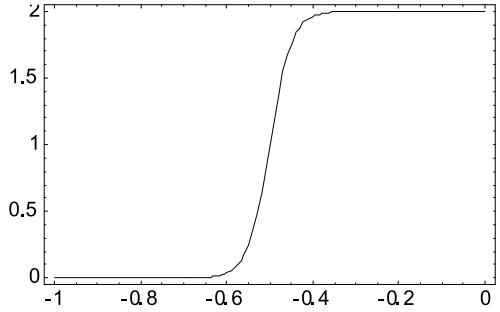


Figure 6.3: Inflow function for Smith and Hutton problem, $\Upsilon = 10$.

the variable $\phi(\boldsymbol{p})$ is transported from the inflow to outflow boundary it is diffused and flattened.

6.1.2 Numerical results

The present test shows the results for Pe=1000. Physical diffusion at Pe=1000 is small, so the resulting function at the output boundary is shaped very similarly to the input function in Figure 6.3. There is no analytical solution to this problem, so the results are compared to another numerical solution. In Figure 6.4 the results for three grid densities ($N_x = 22,\ N_y = 12$, $N_x = 42,\ N_y = 22$, $N_x = 82,\ N_y = 42$) are compared between two methods, FVM and DAM. The Limited Central Difference Scheme (CDS) is used [Leonard, 1997] in the FVM, because it provides the lowest numerical diffusion.

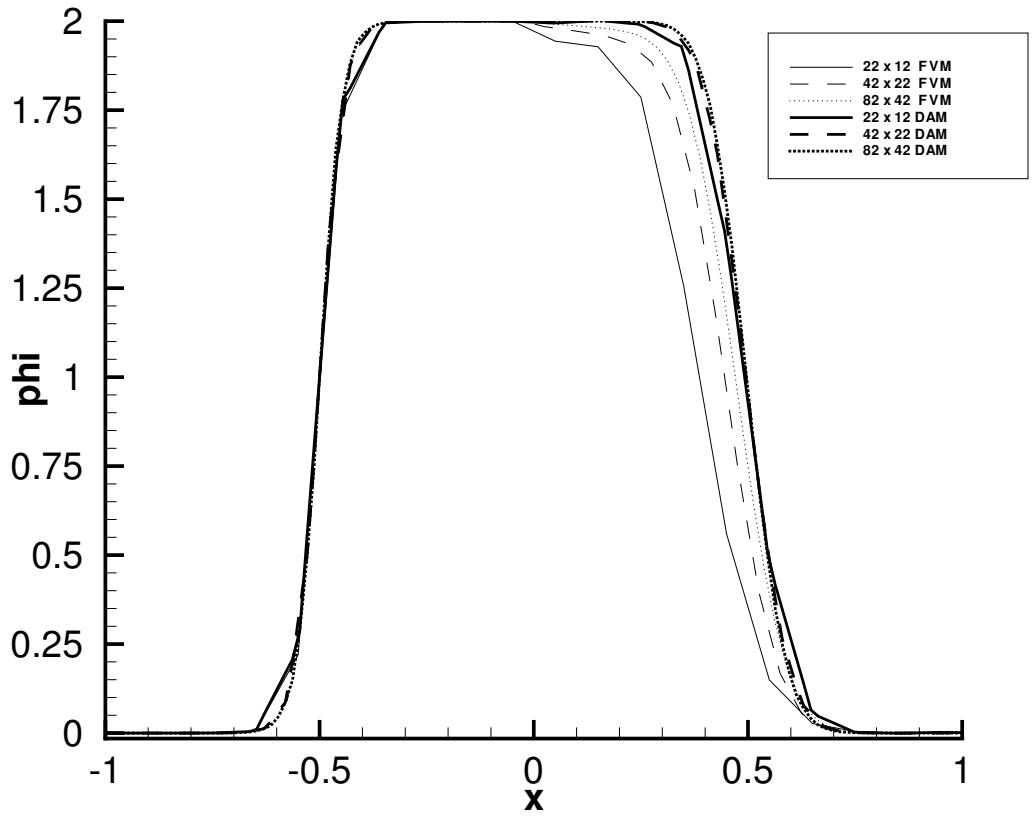


Figure 6.4: Smith and Hutton problem solution at inflow and outflow boundary for Pe=1000 and $\Upsilon = 10$.

6.1.3 Conclusions

FVM results in Figure 6.4 proved that the numerical diffusion is strongly dependent on grid density in traditional polygonization methods. With increasingly finer grids the influence of numerical diffusion decreases. On the other hand, the results obtained by DAM proved that the numerical diffusion and flattening is almost negligible even for very coarse grids. For realistic problems with small physical diffusion this represents an important feature of meshless methods. The DAM calculations were performed by 9 nodes in the domain of influence and c =15.

6.2 Unconfined flow in porous media

Water in nature enters a certain domain at one or more boundaries and leaves it at the other side. A typical example is groundwater flow, which usually results from rainfall. Some of that rainfall water is infiltrated into the ground and flows through different pathways. Usually we are interested only in the part of the flow path inside a certain domain. Therefore the problem is defined with a certain influx and outflux from this domain. Determining the exact pathway is of utmost importance if, for example, some substance is transported along with the fluid. This example tackles the performance of DAM for such situations.

6.2.1 Problem description

Consider a 5 m × 5 m domain of saturated porous medium (material 1) with hydraulic conductivity $\kappa = 10^{-6}$ m/s. Inside this domain there is a low permeable saturated porous layer (material 2) with hydraulic conductivity $\kappa = 10^{-9}$ m/s. The values taken for hydraulic conductivities correspond to the values for silt (higher permeability) and concrete or rock (lower permeability). At the top boundary the water flux enters with $q = 2.55$ m/y. At the bottom boundary the free flow velocity boundary condition is defined. The left and right boundaries are impermeable, therefore no water can enter or leave the domain from the side. The domain is defined as a unsymmetrical as shown in Figure 6.5, because the solution of symmetric problems is usually simpler. Velocity and pressure boundary conditions are shown in Figures 6.6(a) and 6.6(b), respectively. The bottom pressure boundary condition is defined through reference pressure $P_{ref} = 0$ Pa, because it is assumed that this boundary is at the level of the water table.

6.2.2 Governing equations and solution procedure

The problem of unconfined flow through Darcy porous media is defined by the mass conservation equation for incompressible fluids

$$\nabla \cdot \boldsymbol{V} = 0, \tag{6.7}$$

and the Darcy momentum conservation equation

$$\nabla P = -\frac{\mu}{\mathcal{K}} \boldsymbol{V}. \tag{6.8}$$

The solution is obtained by CBS algorithm with the simulation parameters listed in Table 6.1. The solution procedure is explained in Chapter 4.2.

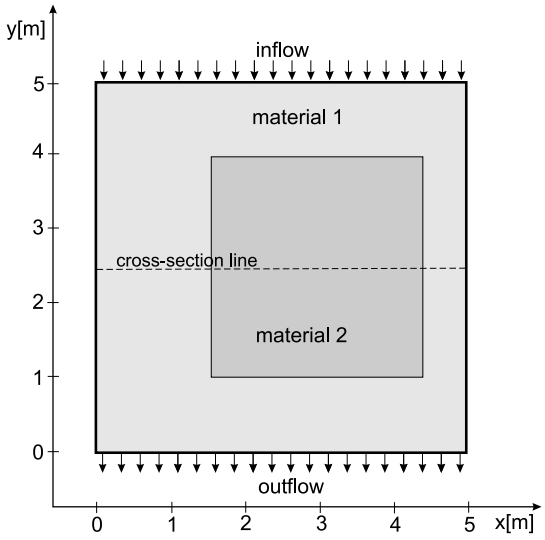


Figure 6.5: Problem definition for unconfined flow.

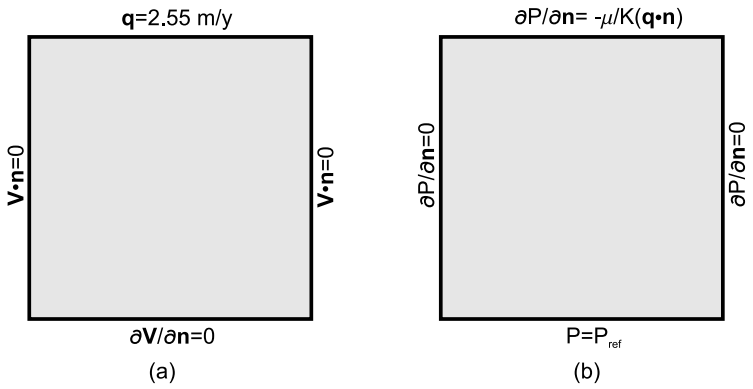


Figure 6.6: Boundary conditions: velocity (left) and pressure (right).

6.2.3 Numerical results

It is expected that most of the water flow entering the domain is diverted away from the domain with lower hydraulic conductivity. A small part of the water flows through the low permeability region as well. In Figure 6.7 the obtained flow structure is presented. The velocity vectors in the low permeability region are very short since the velocities in this region are almost three orders of magnitude smaller than the water flux around this low permeability region. For the sake of comparison, the results are compared at the middle of the domain height, namely at 2.5 m. There is no known analytical solution to this case, therefore the DAM results are compared to Porflow. The comparison is conducted on two grid densities with a uniform distribution of

Table 6.1: Simulation parameters for unconfined flow problem.

parameter	label	value
time step	Δt	0.001 s
internal convergence criteria	η_{int}	0.000001 m/s
total convergence criteria	η_{con}	0.00001 m/s
domain of influence	I	9
weight function free parameter	c	15

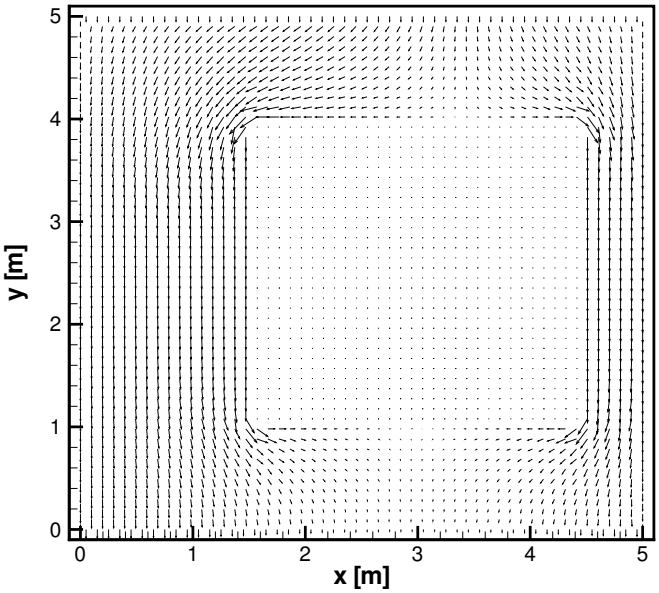


Figure 6.7: Calculated flow structure.

nodes. Grid densities used are $\Delta x = \Delta y = 10$ cm and $\Delta x = \Delta y = 5$ cm for coarser and finer grid, respectively, in both x and y directions. Results are presented in Figure 6.8.

6.2.4 Conclusions

From the results presented in Figure 6.8 it can be seen that the large velocity differences resulting from the material properties involved are frequent in flow calculations. The velocities in the low permeability region are several orders of magnitude removed from those in the high permeability region. Therefore the numerical method used must be well adapted for coping with strong non-

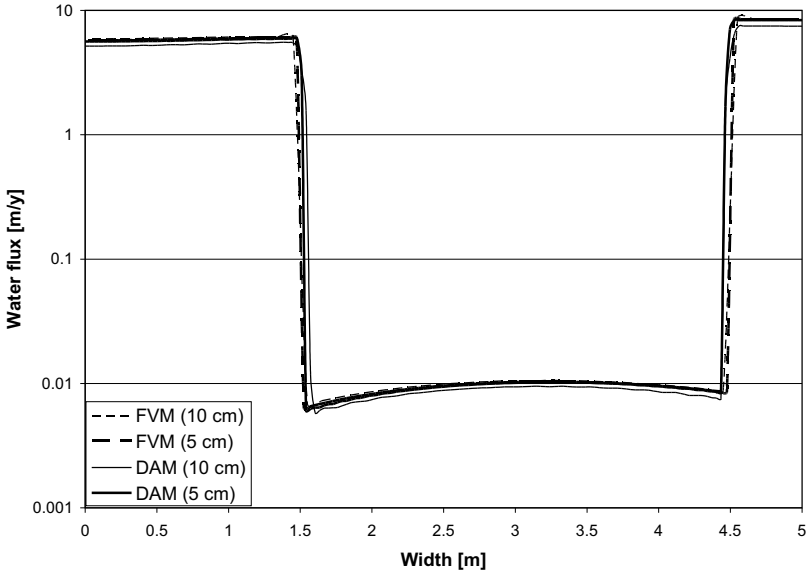


Figure 6.8: Comparison of the results over the cross-section at the mid height of the domain.

linearities. The qualitative comparison shows a good agreement between our results and the results obtained by the commercial package. In addition we can see that the flow structure and the qualitative results are not strongly dependent on the grid density. The results converge toward the same values for both methods used. The consistency of DAM is therefore demonstrated with the presented case.

6.3 Radionuclide transport

The third case under consideration is the transport of decaying species, taken from [Mallants, 2004]. Consider one-dimensional vertical transport of a sorbing species through a water-saturated concrete bunker. This test case is presented for several reasons. First, the consistency of DAM is compared against other numerical methods. Second, different time-stepping procedures are used and compared. And finally, the accuracy and convergence of results in terms of the chosen time step is examined.

6.3.1 Problem description

The length of the problem domain is 2.7 m. At 0.5 m from the top of the bunker, a 0.5 m high waste zone exists as shown in Figure 6.9. In the

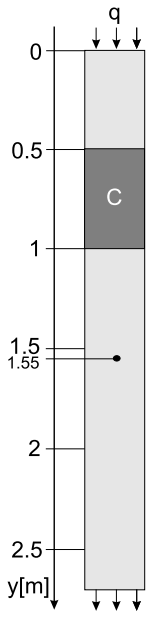


Figure 6.9: Radionuclide transport problem definition.

waste zone, the total amount of ^{226}Ra present is 0.0877 moles with initial activity of $7.25 \cdot 10^{11}$ Bq. The half-life $t_{1/2}$ of ^{226}Ra is 1600 y. We assume the waste is uniformly distributed in the waste zone, which has a unit length in horizontal directions x and z. The constant water flux q through the bunker is $4.53 \cdot 10^{-4}$ m/y. Considering a porosity $\epsilon = 0.37$, the pore water velocity $v = q/\epsilon = 1.224 \cdot 10^{-3}$ m/y. The sorption of radium onto the solid phase is described by the equilibrium distribution coefficient K_d, which is set to 0.653 m^3/kg. Bulk density ϱ_b=1660 kg/m^3, the retardation coefficient R,

calculated from equation (3.40) results in a value of 2930. In partitioning the radium between solid and liquid phase, the volumetric activity in liquid phase is determined from

$$C_f = \frac{m}{\mathcal{V}\,\epsilon\,R} = \frac{7.25 \cdot 10^{11}\,\mathrm{Bq}}{0.5\,\mathrm{m}^3 \cdot 0.37 \cdot 2930} = 1.37 \cdot 10^9\,\mathrm{Bq/m}^3, \tag{6.9}$$

where $\mathcal{V}$ is a volume of 1D waste region. Molecular diffusivity D_p is set to 0.0023 m^2/y, while longitudinal dispersion coefficient $\alpha_L = 0.1$ m.

6.3.2 Governing equations and solution procedure

The problem of decaying species is characterized by the species conservation equation

$$R\,\frac{\partial C_f}{\partial t} + \nabla \cdot (\boldsymbol{v}\,C_f) = \nabla \cdot (\mathbf{D}\,\nabla C_f) - \bar{\lambda}\,R\,C_f. \tag{6.10}$$

Since this simulation is made in 1D, the diffusion tensor $\boldsymbol{D}$ is simplified to scalar value D, defined as

$$D = D_p + \alpha_L\,v, \tag{6.11}$$

where D_p, α_L, and v are molecular diffusion, dispersion coefficient and pore water velocity, respectively. The results are tested on different time-stepping schemes. In this case the final time is defined instead of convergence criteria (due to decay, the steady state is at 0 $\mathrm{Bq/m}^3$). The calculation procedure is shown in Figure 6.10

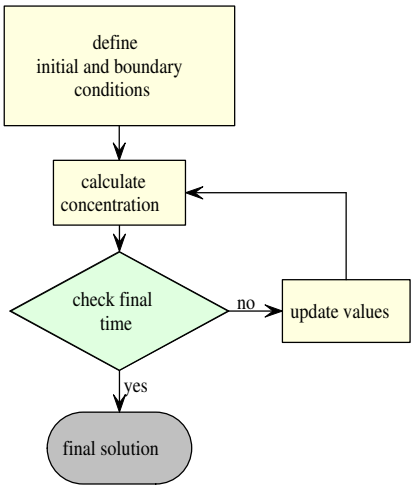


Figure 6.10: Calculation flowchart for the transport of decaying species.

6.3.3 Numerical results

The case of a decaying substance is chosen for several reasons. (1) Transport of radionuclides is characterized by a relatively slow movement. Thus all errors are accumulated in time. (2) The ratio between the initial source and observed volumetric activity in some more or less distant nodes can be and usually are of several orders of magnitude apart (easily up to 20). Numerically, this is not an easy task, because the numerical error can play a key role in the accuracy of the results. (3) The large source term originating from the decay can cause consequent instability of a numerical solution.

The volumetric activity is observed at the point positioned at 1.55 m from the top boundary. The comparative solution is obtained by the FVM-based commercial package Porflow and the FEM-based commercial package Hydrus. The results are calculated on four grid densities with spacing Δy=5 cm, 1.25 cm, 1 cm, and 0.5 cm and compared against the analytical solution given in [van Genuchten and Alves, 1982]. An analytical 1D solution for the volume-averaged species equation (3.41) is obtained for a semi-infinite system with the lower Neumann boundary condition

$$\frac{\partial C_f(\infty,t)}{\partial n_\Gamma} = 0; \qquad y \in \Gamma^N. \tag{6.12}$$

The top boundary conditions for a pulse-type injection using a well-mixed input solution is defined as a third-type or Robin boundary condition with

$$\begin{array}{l} \left(C_f - \frac{D}{V}\frac{\partial C_f}{\partial n}\right)_{y=0+} = C_0; \quad 0 < t \le t_0 \\ \left(C_f - \frac{D}{V}\frac{\partial C_f}{\partial n}\right)_{y=0+} = 0; \quad t > t_0, \end{array} \tag{6.13}$$

where t_0 is pulse duration and $y = 0+$ denotes the approaching of y toward $y = 0$ from the positive direction. In case of single step the initial condition becomes

$$C_f(y,0) = \left\{ \begin{array}{ll} C_1 & 0 \le y < y_1 \\ C_2 & y \ge y_1, \end{array} \right. , \tag{6.14}$$

with y_1 denoting the length of the concentration field with the concentration C_1. For the particular case, the concentration C_1 is the actual concentration in liquid phase C_f. At initial time the concentrations around this region are zero i.e. $C_2 = 0$. The analytical solution is

$$C_f(y,t) = \left\{ \begin{array}{ll} C_i\, A(y,t) + C_0\, B(y,t); & 0 < t \le t_0 \\ C_i\, A(y,t) + C_0\, B(y,t) - C_0\, B(y,t-t_0); & t > t_0 \end{array} \right. ,$$

where

$$\begin{aligned} A(y,t) &= \exp\left(\frac{\bar{\lambda}\,t}{R}\right)\cdot\left(1-\frac{1}{2}\,\mathrm{erfc}\left[\frac{R\,y-V\,t}{2\,(D\,R\,t)^{1/2}}\right]\right. \\ &- \left[\frac{V^2\,t}{\pi\,D\,R}\right]^{1/2}\exp\left[\frac{(R\,y-V\,t)^2}{4\,D\,R\,t}\right] \\ &+ \left.\frac{1}{2}\left[1+\frac{V\,y}{D}+\frac{V^2\,t}{D\,R}\right]\exp\left[\frac{V\,y}{D}\right]\,\mathrm{erfc}\left[\frac{R\,y+V\,t}{2\,(D\,R\,t)^{1/2}}\right]\right) \end{aligned} \tag{6.15}$$

and

$$\begin{aligned} B(y,t) &= \frac{V}{V+U}\,\exp\frac{(V-U)\,y}{2\,D}\,\mathrm{erfc}\left[\frac{R\,y-V\,t}{2\,(D\,R\,t)^{1/2}}\right] \\ &+ \frac{V}{V-U}\,\exp\frac{(V+U)\,y}{2\,D}\,\mathrm{erfc}\left[\frac{R\,y+V\,t}{2\,(D\,R\,t)^{1/2}}\right] \\ &+ \frac{V^2}{2\,\bar{\lambda}\,D}\,\exp\left(\frac{V\,y}{D}-\frac{\bar{\lambda}\,t}{R}\right)\,\mathrm{erfc}\left[\frac{R\,y+V\,t}{2\,(D\,R\,t)^{1/2}}\right], \end{aligned} \tag{6.16}$$

with

$$U = V\sqrt{1+\frac{4\,\bar{\lambda}\,D}{V^2}}. \tag{6.17}$$

The calculations are performed using three different numerical methods on all grid densities. The behavior of different methods in terms of accuracy is shown in Figure 6.11. The results shown for DAM are calculated by explicit Euler time scheme and time step $\Delta t = 50$ y, 9 nodes in the domain of influence and weight function free parameter c =15. Comparison shows that the DAM and FVM solutions are increasingly accurate with finer grids with overestimation of the results on coarser grids. Results obtained by FEM are less dependent on grid density. In the case of coarser grids, however, the solution is underestimated, which is problematic from the point of view of conservativity of the results. Conservativity is important in radionuclide transport calculations because the volumetric activity or flux obtained must be reflected in larger or equal results to those in reality and not lower. The exact peak volumetric activity at y=1.55 m resulting from the analytical solution is 575 Bq/m^3 at 15000 y. The relative errors at peak value are shown in Table 6.2. The next thing analyzed is the stability of different time-stepping schemes. Only the explicit schemes are analyzed. The results are presented on 1 cm node spacing for explicit Euler scheme, Adams-Bashforth scheme, Runge-Kutta scheme and explicit CBS scheme in terms of stability (maximum time step) and total calculation time. The calculation times in Table 6.3 are presented for maximal time step of each scheme and final time 40000

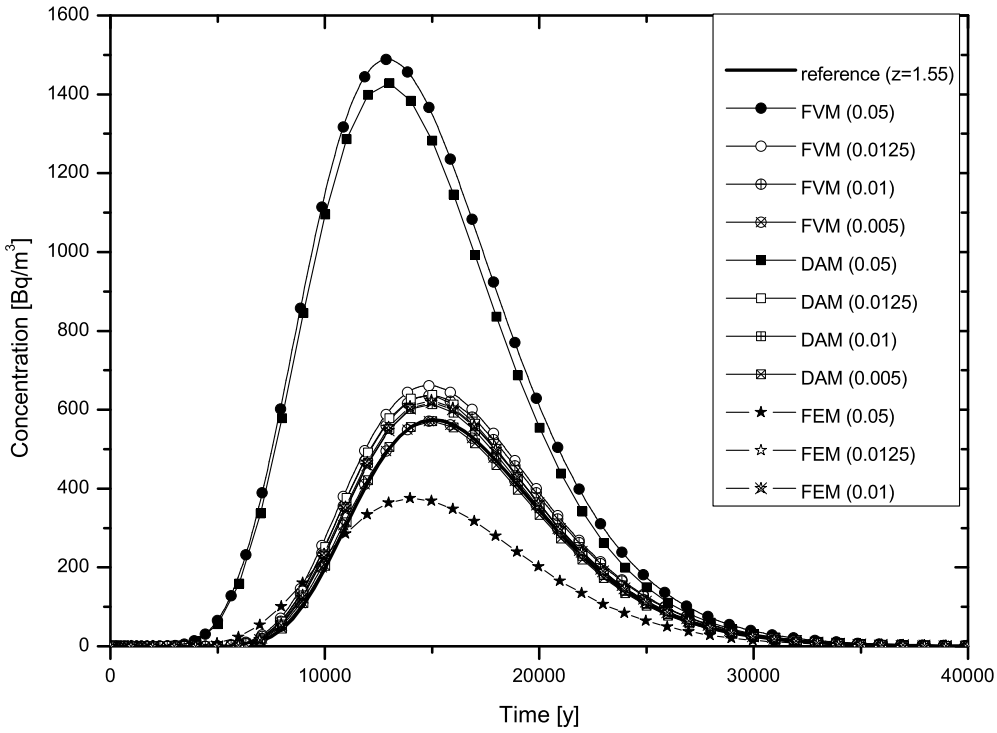


Figure 6.11: Volumetric activity versus time at y=1.55 m.

Table 6.2: Relative errors at the peak value.

Relative errors [%]				
	node spacing			
	10 cm	5 cm	1.25 cm	1 cm
FVM	158.85	15.02	10.57	0.57
FEM	-34.78	8.52	7.65	-
DAM	148.52	10.90	6.73	0.60

y. Another useful comparison between different time-stepping schemes concerns the accuracy in relation to the chosen time step. In Figure 6.12 the DAM results for 1 cm spacing between nodes and different time schemes are

[1]All calculations are performed on personal computer with 3 GHz CPU and 1 GB of RAM.

Table 6.3: Maximal time step for different time-stepping schemes and total calculation time.

	Scheme			
	Explicit	Adams-Bash.	Runge-Kutta	Explicit CBS
Max. Δt [y]	50	20	80	50
Time [s][1]	9.73	15.22	15.67	10.90

shown at various time steps. Maximal relative error is calculated as

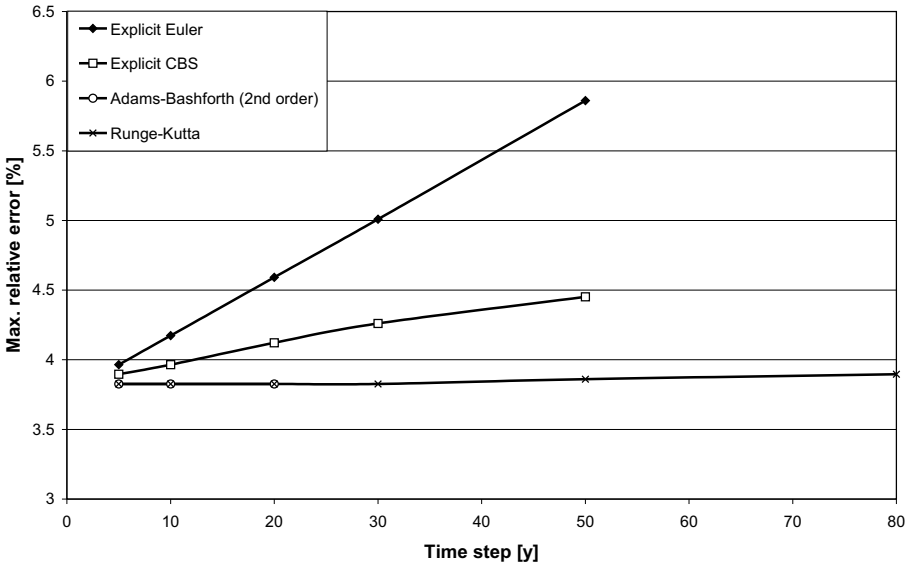


Figure 6.12: Maximal relative error versus time step.

$$\eta_C = sign\left(\frac{C(t) - C_{ref}(t)}{C_{ref}(t)}\right) max_t \left|\frac{C(t) - C_{ref}(t)}{C_{ref}(t)}\right|. \qquad (6.18)$$

6.3.4 Conclusions

From the results shown in Table 6.2 we can see that DAM gives more accurate results than FVM and sometimes also than FEM. Calculation time is hard to compare between the methods, since both commercial packages are well optimized for radionuclide transport while our code is in the testing stage. In addition, we didn't make a separate 1D method, therefore the domain is still 2D containing 3 nodes in the x direction. For pure 1D simulation the calculation times are expected to be considerably lower than they are

now. Computing times needed for the time period of 40000 y with time step $\Delta t = 10$ y are 23 s, 14 s and 3 s for DAM, Hydrus 1D and Porflow, respectively.

In Table 6.3 the comparison of different time-stepping schemes are presented. The most stable explicit scheme is the 4th order Runge-Kutta scheme. The Runge-Kutta procedure, however, needs four evaluations in one time step and is thus slower than the explicit Euler scheme. Adams-Bashforth schemes are supposed to be more stable than the pure explicit scheme, but it is shown in our case that this is quite different. We used a 2nd order Adams-Bashforth scheme for comparison. Also a 3rd order scheme was used, but it turns out to be even less stable than the 2nd order scheme. The reason for this is not known. Perhaps it is due to the type of problem or due to the specific numerical technique.

In addition, accuracy versus time is tested for different time schemes. The results presented in Figure 6.12 show strong dependence of errors on the time step chosen for the explicit Euler scheme. Lower dependence is obtained by explicit CBS. The lowest dependence is for Adams-Bashforth and Runge-Kutta schemes. The Adams-Bashforth scheme, however, allows only small time steps in comparison to the Runge-Kutta scheme.

6.4 Combined water flow and radionuclide transport

This is a more realistic example of radioactive waste release from the proposed repository, described in details in [Petkovšek *et al.*, 2002] and [Železnik *et al.*, 2003]. The procedure used in this example is typical for the direct calculations made for performance assessment and safety analysis of the radioactive waste repository.

6.4.1 Problem description

The repository consists of a multi-barrier system in a vault-type repository, where the following barriers prevent the waste from interacting with the environment: the waste packages, concrete fill (inside and outside the package), concrete structure (vault) and the geosphere (soil).

Realistic model

Solid radioactive wastes are packed in 200 l or 320 l (over-pack) steel drums, steel canisters $2 \times 2 \times 2.8$ m, and further in 864 l tube-type containers (TTC). Waste within the packages are sealed with concrete.

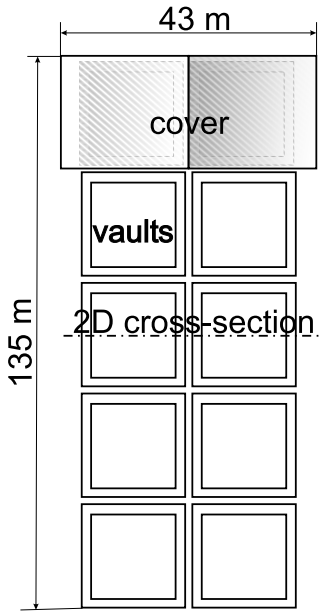


Figure 6.13: Aerial view of surface radwaste repository.

The TTCs are placed inside a concrete disposal container - concrete block, of dimensions $2.5 \times 2.5 \times 3.2$ m. After emplacement of the TTC, the concrete block is filled with poured concrete and covered with a prefabricated concrete lid. The drums are sealed in blocks of $8 \times 8 \times 3$ m.

All concrete blocks are disposed in 10 vaults, with dimensions $20 \times 20 \times 10$ m, with a 0.6 m thick bottom plate and 0.4 m thick walls. Gaps between blocks are backfilled with a mixture of native sand or bentonite clay and the whole vault is covered with a reinforced concrete plate. The whole repository structure consisting of 10 vaults is 135 m long, 43 m wide and 10 m high. An aerial view of the surface repository is depicted in Figure 6.13.

Numerical model

In reality the problem is, of course, defined in three dimensions. From the numerical point of view, 3D problems are usually too cumbersome, especially if the time frames considered are large. Therefore, the level of dimensionality is decreased by at least one dimension. Here the reduction to 2D is considered as sketched in Figure 6.14. The problem domain in Figure 6.14 is symmetrical

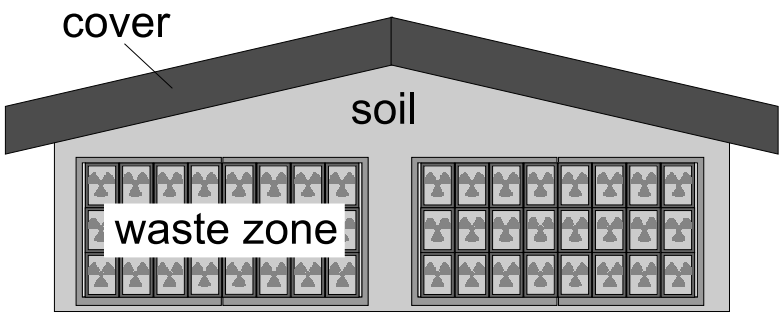


Figure 6.14: 2D scheme of radwaste repository.

with respect to the x coordinate. Further reduction of the calculation domain can be made then by taking only a half of one vault as depicted in Figure 6.15. Discretization of the domain is uniform with $\Delta x = \Delta y$ =10 cm spacing in both directions for easier comparison with the commercial package Porflow.

Materials considered are soil, concrete walls, backfill and concrete containers, marked in Figure 6.15 by numbers 1, 2, 3 and 4, respectively. The material properties of steel drums, TTC and concrete container are averaged. All together are considered as fine concrete with low permeability. The physical parameters are set as realistically as possible and were provided by SCK-CEN in Belgium [Mallants *et al.*, 2003].

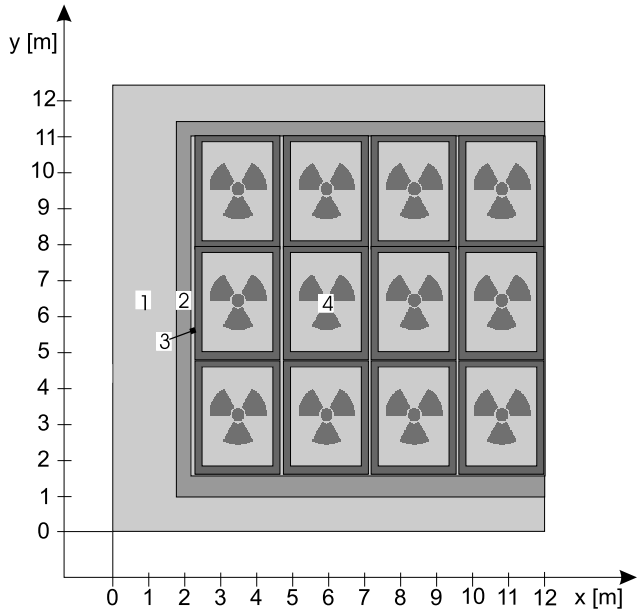


Figure 6.15: Schematic representation of the calculation domain.

Flow calculation In reality, the first engineered barrier for a surface repository is a cover which prevents the majority of the water from being infiltrated into the repository itself. In this work, however, the water flux that enters the calculation domain is assumed to be equal to the infiltration. Considering a rainfall of 1000 mm/y, 300 mm/y evapotranspiration and 100 mm/y runoff gives us 1000 mm/y-300 mm/y-100 mm/y=600 mm/y (0.6 m/y) infiltration. This water infiltrates at the top, as shown in Figure 6.5. So the top velocity boundary condition is of Dirichlet type with prescribed water flux. The velocity boundary conditions on the left and the right side are Neumann boundary conditions, because of the symmetry. The bottom boundary is characterized by free flow boundary condition with Neumann velocity and Dirichlet pressure boundary condition. The velocity and pressure boundary conditions are shown graphically in Figure 6.16(a) and in Figure 6.16(b), respectively. Assuming saturated conditions, the saturated hydraulic conductivities, porosities and bulk densities for each material are as listed in Table 6.4.

Radionuclide transport calculation Two radionuclides are considered for the comparison, short-lived ^3H with half-life $t_{1/2}$=12.3 y and long-lived ^{14}C with half-life $t_{1/2}$=5700 y with hypothetical initial total activity of 10^4 Bq for ^3H and 10^9 Bq for ^{14}C. Assuming the homogeneous distribution of each radionuclide over the whole repository, the corresponding volume per

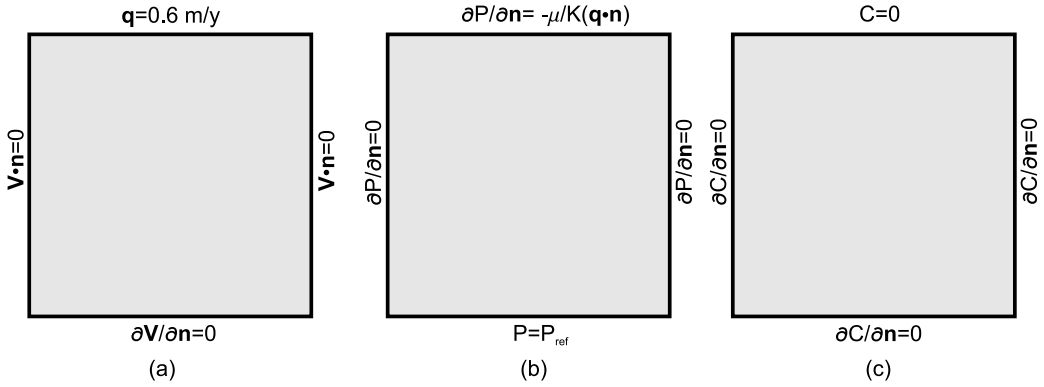


Figure 6.16: Velocity, pressure and concentration boundary conditions.

Table 6.4: Parameters for flow calculations.

Material	Hydraulic cond. [m/s]	Porosity [-]	Bulk density [kg/m^3]
1	10	0.42	1540
2	10^{-8}	0.16	2230
3	10^{-7}	0.36	1700
4	10^{-9}	0.16	2230

unit depth and half repository is $\mathcal{V} = 67.2$ m^3. Tritium transport is not retarded while the transport of carbon is much slower due to retardation. The retardation factors are calculated from bulk density in the waste area, the distribution coefficient and from porosity in this area.

$$R^{^3H} = 1 + \frac{K_d^{^3H}\,\varrho_b}{\epsilon} = 1, \tag{6.19}$$

$$R^{^{14}C} = 1 + \frac{K_d^{^{14}C}\,\varrho_b}{\epsilon} = 1 + \frac{2\,\mathrm{m^3/kg} \cdot 2226\,\mathrm{kg/m^3}}{0.16} = 27826. \tag{6.20}$$

As a consequence of different retardation factors, the peak value of carbon takes place later than the peak value of tritium. From known retardation factors we can calculate volumetric activities in water

$$C_f^{^3H} = \frac{m}{\mathcal{V}\,\epsilon\,R_{^3H}} = \frac{1.0 \cdot 10^3\,\mathrm{Bq}}{67.2\,\mathrm{m^3} \cdot 0.16 \cdot 1} = 93\,\mathrm{Bq/m^3}, \tag{6.21}$$

$$C_f^{^{14}C} = \frac{m}{\mathcal{V}\,\epsilon\,R_{^{14}C}} = \frac{1.0 \cdot 10^9\,\mathrm{Bq}}{67.2\,\mathrm{m^3} \cdot 0.16 \cdot 27826} = 3342\,\mathrm{Bq/m^3}. \tag{6.22}$$

It should be emphasized here, that volumetric activity can be used only in cases of single decaying species. However, if a multi-species decay chain is considered, the amount or mass (in moles) per unit volume of radionuclides

have to be used. The relation between activity and amount of material is made through molar activity as

$$N_r = \frac{A}{A_m} = \frac{A}{N_A\,\lambda} = \frac{A\,t_{1/2}}{N_A\,\ln 2}, \tag{6.23}$$

where N_r is amount of radionuclide in mole, A is activity in Bq, A_s is specific activity in Bq/mole and N_A is Avogadro's number. Physical properties for transport are listed in Table 6.5. Boundary conditions for radionuclide

Table 6.5: Physical parameters for radionuclide transport calculations.

Material	Diffusion	dispersion [m]		Distribution
	[m^2/y]	longitudinal	transversal	coefficient [m^3/kg]
^3H				
1	$2.14\cdot 10^{-4}$	0.01	0.001	0
2-4	$2.14\cdot 10^{-4}$	0.001	0.001	0
^{14}C				
1	$4.42\cdot 10^{-4}$	0.01	0.001	0.02
2-4	$4.42\cdot 10^{-4}$	0.001	0.001	2.00

transport are of Dirichlet type for top and left boundary with value $C = 0$ Bq/m^3, i.e. no radionuclides enter the domain. Bottom and right boundary conditions are of Neumann type due to the symmetry on the right side and due to free discharge of the radionuclides on the bottom. Appropriate concentration boundary conditions are shown in Figure 6.16(c).

6.4.2 Governing equations and solution procedure

Since the flow field is not affected by the concentration field, it can be calculated separately. In the case of radionuclide transport this detail is very important, because the time scale for calculation of the flow field is much different from the time scale of radionuclide transport. A steady-state flow field is achieved after a few years or after a few tens of years, whereas the time frames for peak concentrations can be measured even in several hundred thousand years.

For this reason the first flow field is determined and after that this flow field is applied or superimposed to the radionuclide transport calculations as shown in Figure 6.17. The problem of unconfined flow through the Darcy porous media is defined by Darcy momentum conservation equation

$$\nabla P = -\frac{\mu}{\mathcal{K}}\boldsymbol{V}. \tag{6.24}$$

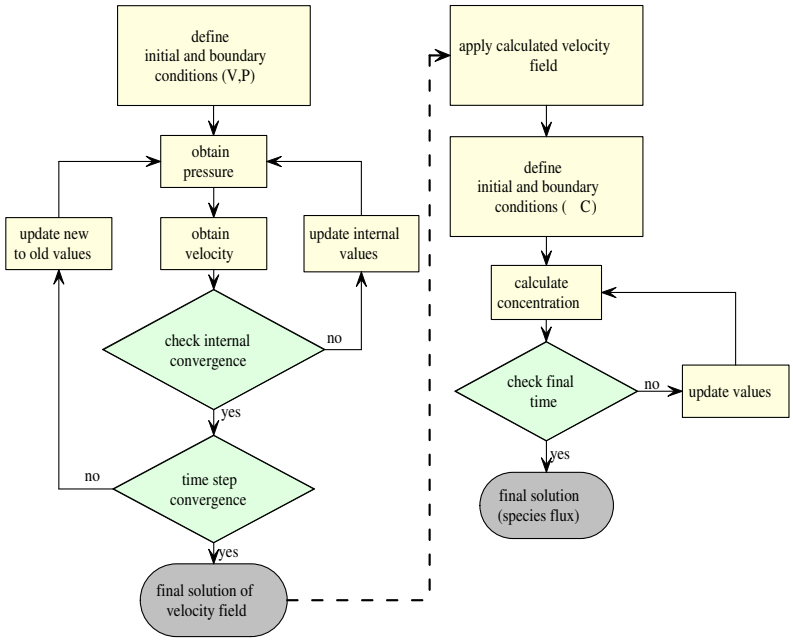


Figure 6.17: Calculation flowchart for combined water flow and radionuclide transport problem.

Table 6.6: Simulation parameters for water flow calculation.

parameter	label	value
number of nodes	$N_x \times N_y$	120×124
node spacing	$\Delta x = \Delta y$	10 cm
time step	Δt	0.001 y
internal convergence criteria	η_{int}	0.000001 m/y
total convergence criteria	η_{con}	0.00001 m/y
domain of influence	I	9
weight function free parameter	c	15

The solution is obtained by CBS algorithm with the following parameters.

The problem of decaying species is characterized by the species conservation equation

$$R\frac{\partial C_f}{\partial t} + \nabla \cdot (\boldsymbol{v}\, C_f) = \nabla \cdot (\mathbf{D}\, \nabla C_f) - \bar{\lambda}\, R\, C_f. \tag{6.25}$$

The solution is obtained by explicit algorithm using the parameters from Table 6.7.

Table 6.7: Simulation parameters for radionuclide transport through repository.

parameter	label	value
number of nodes	$N_x \times N_y$	120×124
node spacing	$\Delta x = \Delta y$	10 cm
time step (^{3}H)	Δt	0.001 y
time step (^{14}C)	Δt	10.00 y
domain of influence	I	9
weight function free parameter	c	15

6.4.3 Numerical results

Numerical results are given for both the flow field and radionuclide annual discharge. Comparison, however, is made only on the final results, that is for radionuclide discharge through the bottom boundary.

Flow calculation

Four materials are considered in the presented problem. Each is characterized by different saturated hydraulic conductivities. The first material (soil) has the largest hydraulic conductivity. When water "hits" the concrete walls of the repository, the concrete walls offer much more resistance to flow, characterized by lower hydraulic conductivity. Inside the repository there are containers with even lower hydraulic conductivities, but the backfill around them is more permeable. Therefore the majority of the flow flows in these thin gaps between the containers. The described flow structure is shown in the top part of Figure 6.18. Because the water flux inside the containers is considerably smaller than outside repository, the velocity profile is better seen in the bottom part of Figure 6.18. The velocity profile is made across the middle of the repository height. Here we can clearly observe the behavior of the water flow. As described earlier, the water flux is naturally higher around the repository and lower inside. In the gaps, the water flux is higher towards the middle of the repository length.

Radionuclide transport calculation

Radionuclide transport is characterized by several mechanisms, in general by diffusion, advection and retardation. Molecular diffusion is characteristic of each species, the host medium, which is water in this case and of the type of porous media through tortuosity. Often the data about tortuosity for different porous media is not known, therefore we assume that molecular diffusion is constant for all materials. Dispersion, on the other hand is characterized

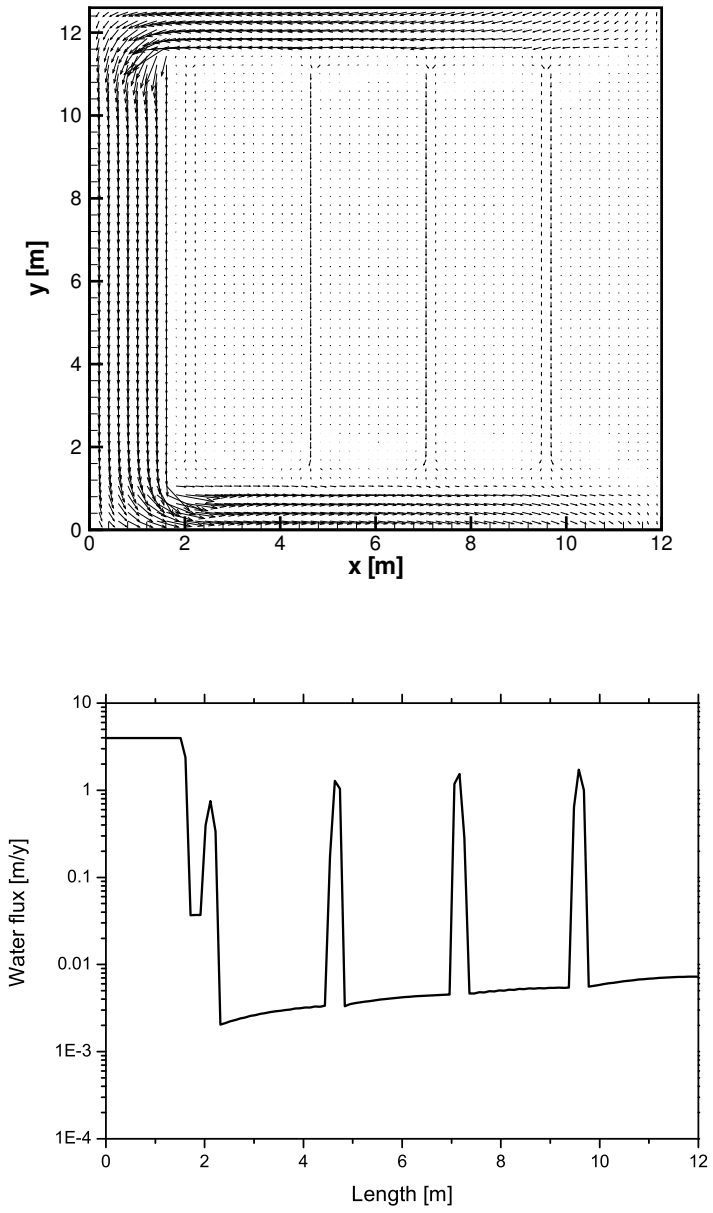


Figure 6.18: Flow structure in the repository (top) and flow at the middle of the repository (bottom).

by the movement of the species and from the type of porous media. The level of retardation is a function of material and the transported species. So

the number of involved parameters that effect the time of the peak and the peak value is relatively high. In addition the results are very sensitive to the initial position of the wastes. When the repository geometry is discretized, special care has to be taken on the waste placement. For this reason the grid is constructed in the same way in FVM and DAM calculations. Calculation results are given in terms of output annual discharge from the bottom (exit) boundary and are given in Figure 6.19.

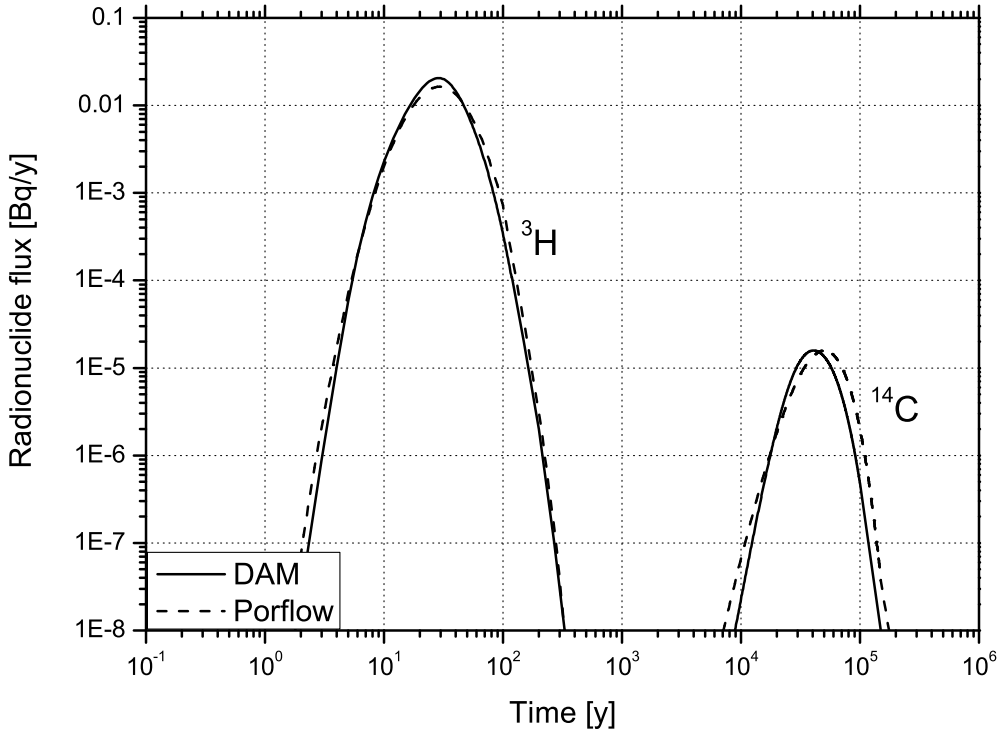


Figure 6.19: Annual discharge from the repository.

6.4.4 Conclusions

This test case is presented for several reasons. Perhaps the most important reason is the lack of realistic problem modelling using meshless methods. In the literature the problems are usually defined by dimensionless numbers on simple geometries using one homogeneous and isotropic material. In this

case a more realistic problem is considered, using primitive variables and real values. Four materials are considered with different properties. Simulation of radionuclide transport is numerically difficult because the initial concentration exhibits very intense jumps from zero outside to large concentrations in the waste region. Another problem is the large difference between input and output concentrations. So we are faced with a combination of large gradients which can cause numerical instabilities and great sensitivity of the system to the output results. The compared results are in excellent agreement with the solution obtained by the Porflow package.

6.5 Natural convection in porous media

Natural convection in porous media is a standard test case for code verification. Although the exact analytical solution is not known, some benchmark tests are available for comparison. From the first calculations of natural convection in the enclosure made by [Chan *et al.*, 1970] and [Shiralkar *et al.*, 1983], with the improvement of the hardware and computational capabilities, the results on a finer grid (50×50) are published by Ni and Beckermann in 1991 [Ni and Beckermann, 1991]. Natural convection in porous media in the square enclosure was calculated using mesh reduction DRBEM in 2000 [Šarler *et al.*, 2000] and by RBFCM in 2004 [Šarler *et al.*, 2004b]. In [Sadat *et al.*, 1996] a solution using DAM was presented in 1996. Much as the problem was calculated by several authors and using different numerical methods, we introduced it for the reason of comparison between the standard Pressure Poisson and CBS algorithm for the solution of a coupled pressure-velocity equation.

6.5.1 Problem description

The problem of natural convection in Darcy porous media under consideration is based on porous media Rayleigh (Ra^*) and the aspect ratio between height and length of the cavity A dimensionless numbers, defined as

$$\mathrm{Ra}^* = \frac{\mathcal{K}\, g\, \beta\, \Delta T\, H}{\nu\, \alpha_{th}}, \tag{6.26}$$

$$\mathrm{A} = \frac{H}{L}, \tag{6.27}$$

where $\mathcal{K}$ is permeability, g is the magnitude of gravitational acceleration, H is height of the enclosure, L is width of the enclosure, ΔT is temperature difference, ν is viscosity and α_{th} stands for thermal diffusivity

$$\alpha_{th} = \frac{\lambda}{\varrho\, c_p}, \tag{6.28}$$

with thermal conductivity λ. A graphical presentation of the problem is shown in Figure 6.20. Due to thermal gradients arising from the temperature difference between the hot and cold walls the fluid is subjected to buoyancy forces, here described by Bousinesq approximation

$$\varrho = \varrho_0\, [1 - \beta_T\, (T - T_0)], \tag{6.29}$$

where β_T is the coefficient of thermal expansion, and T_0 and ϱ_0 are reference temperature and density, respectively. Further assumptions and simplifications are:

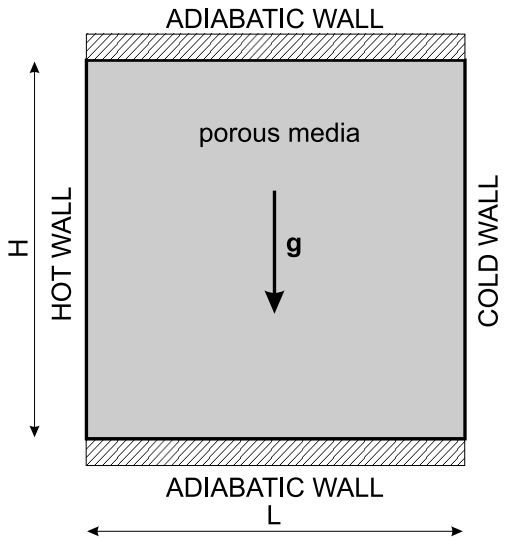


Figure 6.20: Geometry of the enclosure for natural convection in porous media.

- flow obeys Darcy behavior
- porous matrix is homogeneous and isotropic
- all physical properties are constant

The natural boundary conditions for Darcy flow are slip boundary conditions. In other words, the boundaries are impermeable for flow in the normal direction, but fluid can flow freely in the tangential direction. Velocity, pressure and temperature boundary conditions are shown in Figure 6.21.

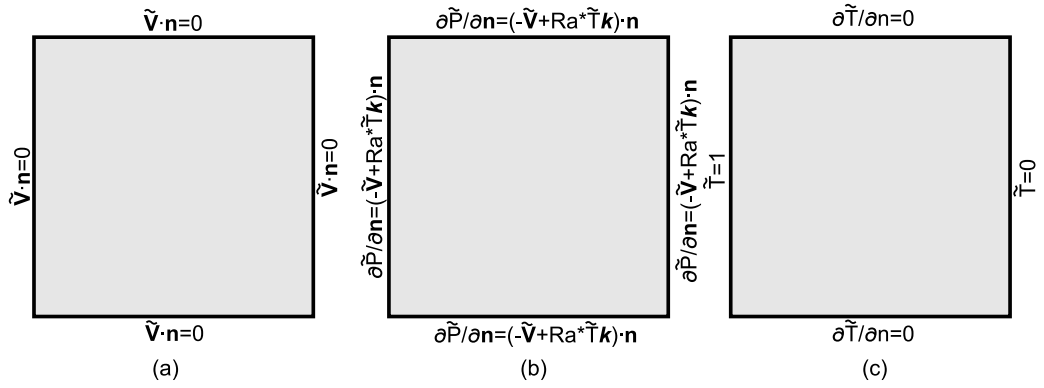


Figure 6.21: Velocity, pressure and temperature boundary conditions for the natural convection problem.

6.5.2 Governing equations and solution procedure

Since this case is defined by dimensionless numbers we can rewrite the mass conservation equation for incompressible fluid

$$\nabla \cdot \tilde{\boldsymbol{V}} = 0 \tag{6.30}$$

and Darcy equation (3.11) in dimensionless form

$$\tilde{\nabla} \tilde{P} = -\tilde{\boldsymbol{V}} + \mathrm{Ra}^* \, \tilde{\mathrm{T}} \, \boldsymbol{k}, \tag{6.31}$$

with $\boldsymbol{k}$ being the vector of the direction of gravitational acceleration and with other relations as follows. Dimensionless pressure is

$$\tilde{P} = \frac{H^2}{\alpha_{th} \, \mu} \, P, \tag{6.32}$$

dimensionless velocity

$$\tilde{\boldsymbol{V}} = \frac{H}{\alpha_{th}} \, \boldsymbol{V}, \tag{6.33}$$

and dimensionless temperature

$$\tilde{T} = \frac{T - T_{ref}}{(T_{max} - T_{min})}, \tag{6.34}$$

with reference temperature $T_{ref} = 1/2 \, (T_{max} - T_{min})$. The adjacent steady-state energy equation is

$$\tilde{\nabla} \cdot (\tilde{\boldsymbol{V}} \, \tilde{T}) = \tilde{\nabla}^2 \tilde{T}. \tag{6.35}$$

The solution procedure is as described in flow chart 6.22. Simulation parameters used are listed in Table 6.8. False time step is used in both CBS algorithm and pressure Poisson algorithm.

6.5.3 Numerical results

The calculation is performed for porous media Rayleigh number Ra^*=100 and aspect ratio A=1. A graphical presentation of the results is given in Figure 6.23, where the vectors show flow pattern and the lines represent isotherms. The results are compared for three grid densities, 30×30, 50×50 and 70×70 against the reference solution provided by Gobin *et al.* (can be found in [Šarler *et al.*, 2000]). The reference solution was assessed by fine-mesh FVM. The validation of the FVM code was performed over a large range of parameters for purely thermal natural convection in fluids or in

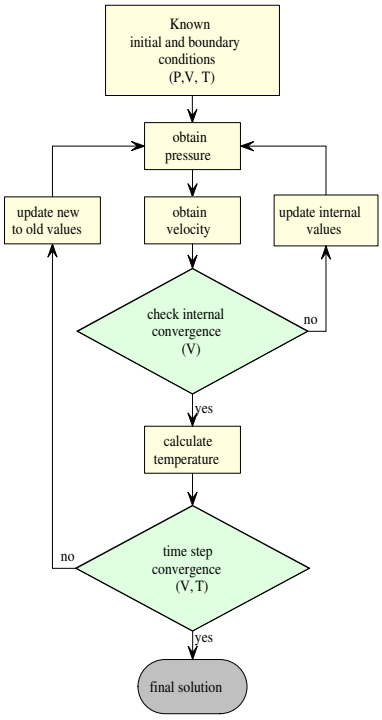


Figure 6.22: Calculation flowchart for natural convection.

porous media. The reference simulation presented here is based on the Darcy-Brinkman version of the code using a very low Darcy number, Da=10^{-8}. The FVM calculations were performed on a Cray-98 vector mainframe.

Temperature profiles for all three grid densities and reference solution along the insulated boundary are shown on the top of Figure 6.24. Velocity profiles are compared along the middle of the domain height for y component and along middle of the domain width for x velocity component. The comparison is shown only for one grid density (see bottom Figure 6.24) because optically we cannot distinguish between the results on different grids. As explained above, the solution is provided using two algorithms for coping with pressure-velocity equations. The first one is the classical pressure Poisson algorithm, which was used extensively in the past. The relatively new CBS algorithm is tested against the pressure Poisson algorithm. The results can be better compared in terms of maximal stream function ψ_{max}, maximal overall Nusselt number Nu_{max}, and maximal velocities on the middle of the domain. Stream function is calculated from

$$\psi_{max} = \int_{p_y^-}^{p_y} V_x \, \mathrm{d}p_y = - \int_{p_x^-}^{p_x} V_y \, \mathrm{d}p_x, \tag{6.36}$$

Table 6.8: Simulation parameters for natural convection.

number of nodes	parameter	label	value
30×30	false time step	Δt_f	0.001
	velocity convergence criteria		
	internal	$\eta_{int,\tilde{V}}$	10^{-6}
	steady state	$\eta_{con,\tilde{V}}$	10^{-5}
	temperature convergence criteria	$\eta_{\tilde{T}}$	10^{-6}
50×50	false time step	Δt_f	0.0001
	velocity convergence criteria		
	internal	$\eta_{int,\tilde{V}}$	10^{-6}
	steady state	$\eta_{con,\tilde{V}}$	10^{-5}
	temperature convergence criteria	$\eta_{\tilde{T}}$	10^{-6}
70×70	false time step	Δt_f	0.00005
	velocity convergence criteria		
	internal	$\eta_{int,\tilde{V}}$	10^{-7}
	steady state	$\eta_{con,\tilde{V}}$	10^{-6}
	temperature convergence criteria	$\eta_{\tilde{T}}$	10^{-7}
	domain of influence	I	9
	weight function free parameter	c	15

where p_y^- and p_x^- are positions on the bottom and the left side of the enclosure, respectively. Overall Nusselt number is defined as:

$$\mathrm{Nu} = \frac{\int_{p_y^-}^{p_y^+} \frac{\partial T(p_x^-, p_y)}{\partial x} \, \mathrm{d}y}{\mathrm{A}\,\Delta T}, \tag{6.37}$$

while the local Nusselt number is

$$\mathrm{Nu}(p_y) = \frac{H \, \frac{\partial T(p_x^-, p_y)}{\partial x}}{\mathrm{A}\,\Delta T}, \tag{6.38}$$

where p_y^- and p_y^+ are positions at the bottom and the top coordinates of the enclosure. The integration is made numerically using trapezoid rule. Maximal velocities are compared along the vertical for x component and along the horizontal middle cross-section for y component velocities. Reference values are given in Table 6.9. The reference values [Ni and Beckermann, 1991] and [Shiralkar *et al.*, 1983] were found in [Sadat *et al.*, 1996], where only data about Nusselt number is given. Previous calculations with DRBEM

[1] All calculations were performed on personal computer with 3 GHz CPU and 1 GB of RAM.

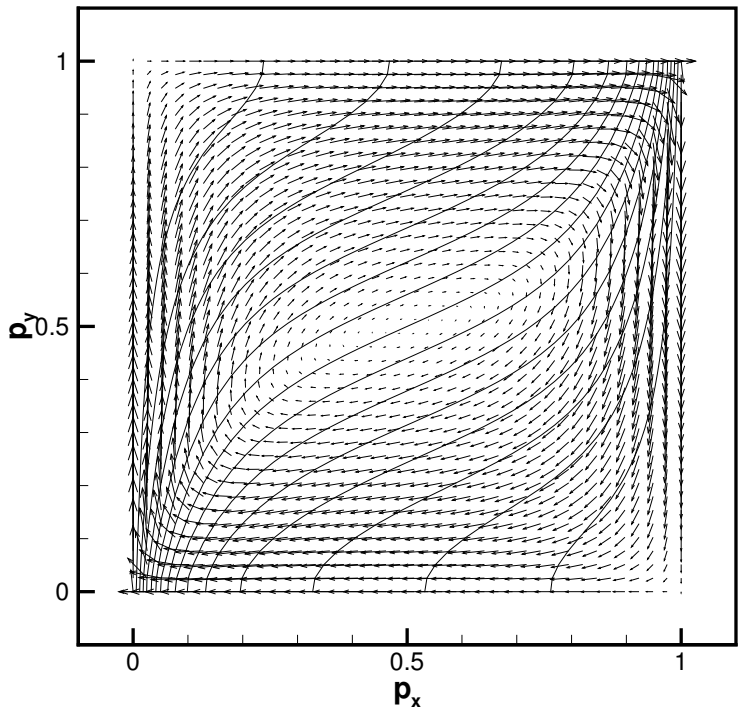


Figure 6.23: Results for Darcy natural convection for Ra*=100 and Pr=1.

Table 6.9: Reference values for natural convection in Darcy porous media for Ra*=100, A=1.

reference	grid	Nu^{ref}	ψ^{ref}	$V_{x,max}^{ref}$	$V_{y,max}^{ref}$
[Šarler *et al.*, 2000] (DRBEM)	30×30	3.149	4.7168	-	-
[Šarler *et al.*, 2004b] (RBFCM)	30×30	3.103	4.6201	-	-
[Gobin and Bennacer, 1996b] (FVM)	200×200	3.102	4.7357	35.8899	17.3805
[Sadat *et al.*, 1996] (DAM)	41×41	3.098	-	-	-
[Ni and Beckermann, 1991]	50×50	3.103	-	-	-
[Shiralkar *et al.*, 1983]	-	3.115	-	-	-

and RBFCM showed that the most sensitive results are those of local Nusselt number. In Figures 6.25 the convergence of local Nusselt numbers for two grid densities (coarser 30×30 and finer 70×70) is presented. Results calculated with DAM are given in Table 6.10 for three grid densities and for both algorithms.

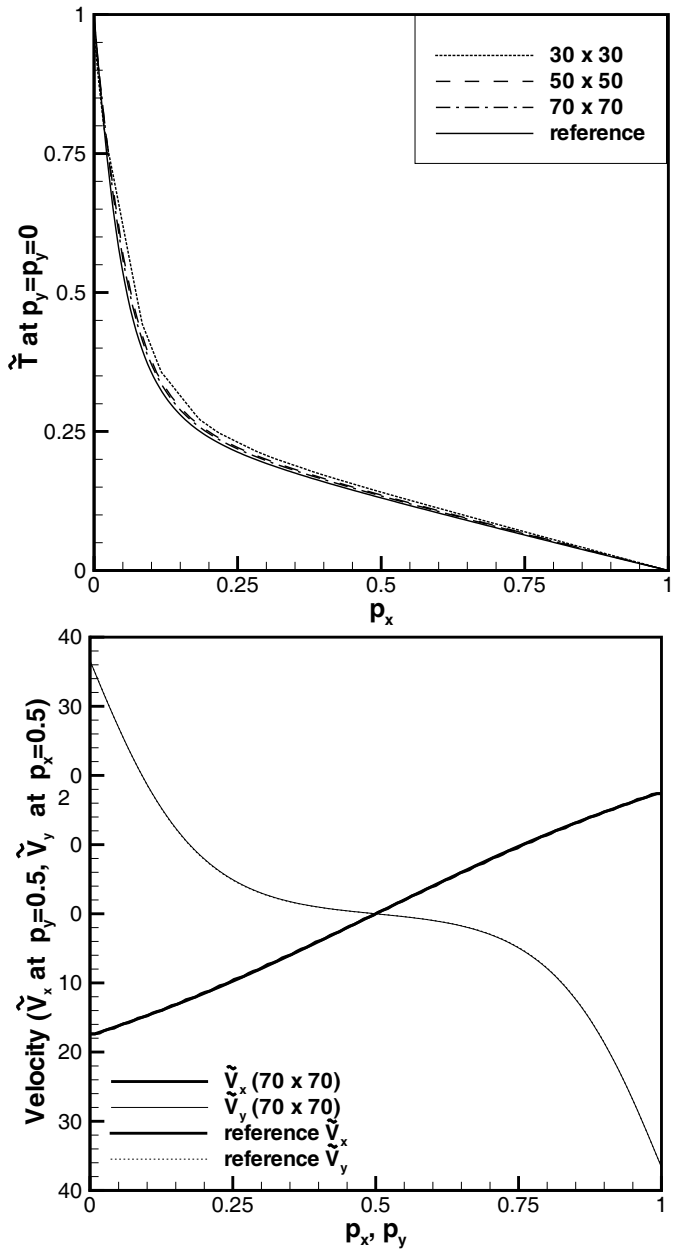


Figure 6.24: Temperature profile at the insulated boundary along x coordinate (top) and velocity components along the middle of the domain (bottom).

Table 6.10: Calculated results for natural convection in porous media (Ra^*=100, A=1).

Algorithm	Grid	Nu	ψ	$V_{x,max}$	$V_{y,max}$	time [min][1]
Poisson	30×30	3.033	4.7001	36.7584	16.9354	3.5
Poisson	50×50	3.056	4.7185	36.7083	17.2745	43
Poisson	70×70	3.077	4.719	36.72	17.3512	150
CBS	30×30	3.032	4.704	36.9457	16.8509	6
CBS	50×50	3.066	4.716	36.7625	17.2279	40
CBS	70×70	3.100	4.721	36.6724	17.3683	190

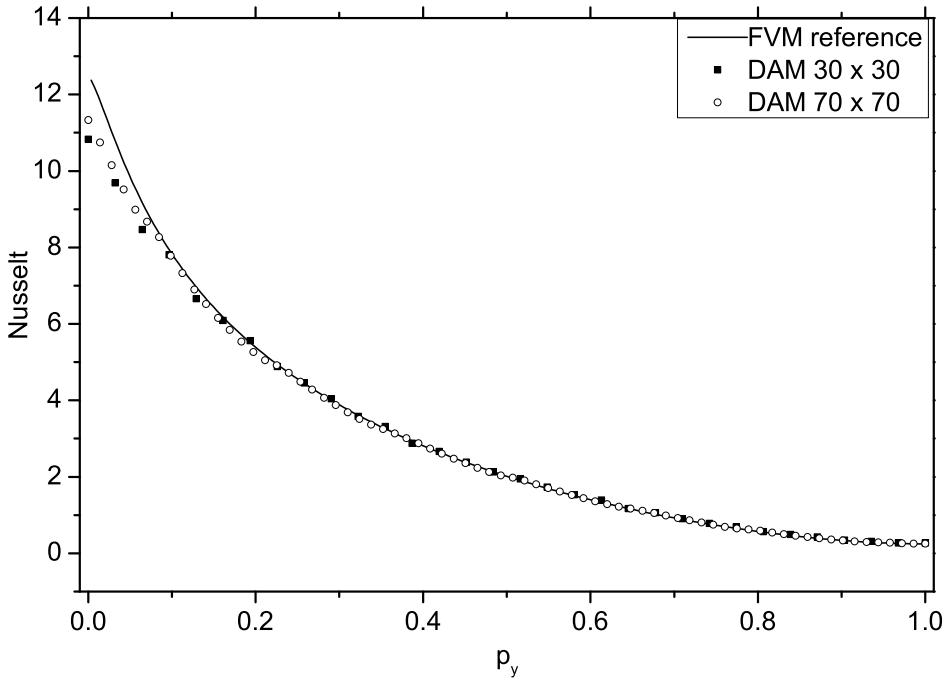


Figure 6.25: Local Nusselt number with 30×30 and 70×70 grid. Reference solution is denoted by solid line.

Natural convection on irregular geometry

A similar problem of natural convection in Darcy porous media is introduced on a domain containing holes in the shape of the letters *PhD* as shown in Figure 6.26. The boundary conditions are the same as defined in Figure 6.21.

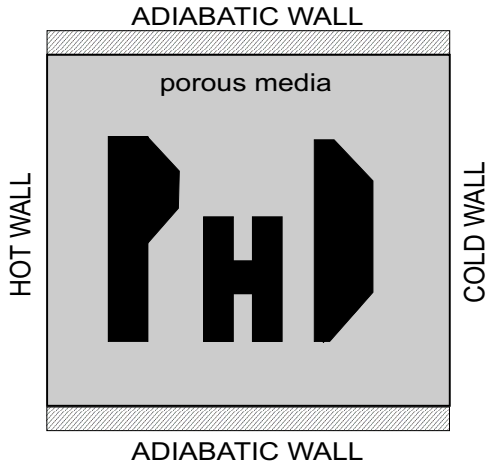


Figure 6.26: Problem geometry for natural convection in porous media on a domain with holes.

Other calculation parameters are given in Table 6.11. Because in this case the geometry is more difficult, the corners of the holes are given in tabular form. The coordinates are tabulated in anti-clockwise direction, because of the definition of normals in our code. Nodes that fall into the hole region are deleted. This case is meant just to show the capability of meshless methods for coping with the problems defined on irregular domains. The velocity boundary conditions for the holes are slip boundary conditions, and Neumann boundary conditions are used for the temperatures. The appropriate vector field and isotherms are shown in Figure 6.27. Temperature profiles are shown at top insulated boundary (p_y^+) and bottom insulated boundary (p_y^-) in Figure 6.28. Results in terms of Nusselt number and maximal stream function are given in Table 6.13.

Table 6.11: Simulation parameters for natural convection in porous media on irregular geometry.

parameter	label	value
Rayleigh number	Ra	100
Aspect ratio	A	1/1.1
Number of nodes	$N_x \times N_y$	50×55
false time step	Δt_f	0.00001
velocity convergence criteria (internal)	$\eta_{int,\tilde{V}}$	10^{-6}
velocity convergence criteria (steady-state)	$\eta_{con,\tilde{V}}$	10^{-5}
temperature convergence criteria	$\eta_{\tilde{T}}$	10^{-5}

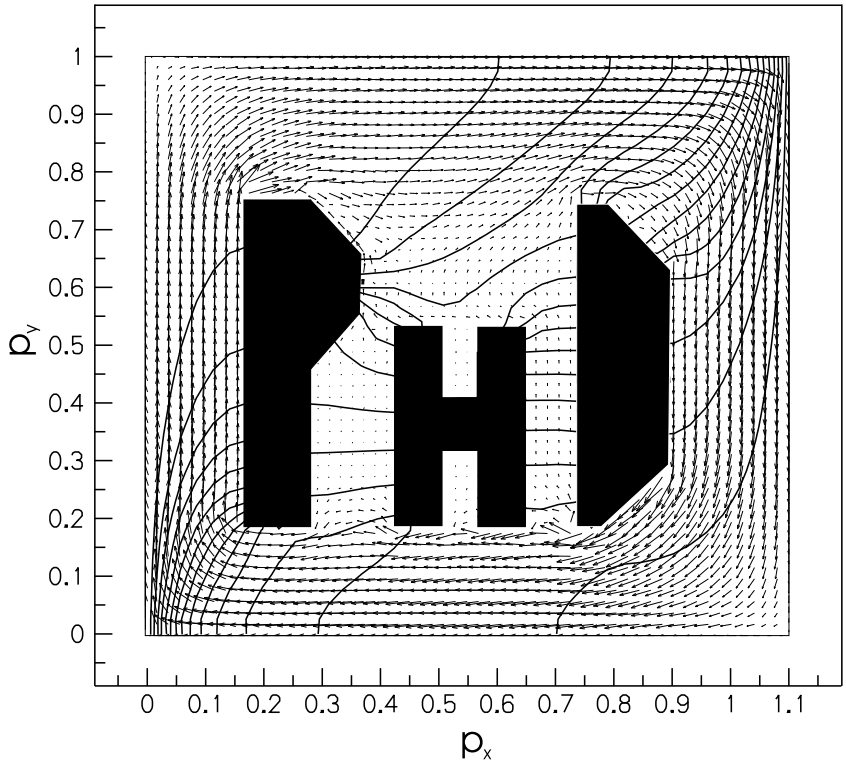


Figure 6.27: Solution of natural convection in porous media on irregular geometry for Ra*=100.

Table 6.12: Coordinates of internal holes.

P		H		D	
p_x	p_y	p_x	p_y	p_x	p_y
0.18	0.75	0.45	0.53	0.75	0.76
0.18	0.19	0.45	0.19	0.75	0.2
0.275	0.19	0.52	0.19	0.8	0.2
0.275	0.46	0.52	0.31	0.89	0.31
0.36	0.56	0.58	0.31	0.89	0.635
0.36	0.66	0.58	0.19	0.8	0.76
0.275	0.75	0.65	0.19	0.75	0.76
0.18	0.75	0.65	0.53		
		0.58	0.53		
		0.58	0.415		
		0.52	0.415		
		0.52	0.53		
		0.45	0.53		

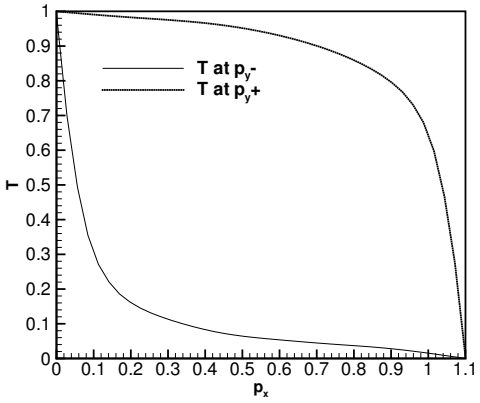


Figure 6.28: Nonsymmetric temperature profiles at the top and the bottom boundary.

Table 6.13: Calculated results for natural convection in porous media (Ra^*=100) with holes.

Nu	ψ
2.7615	4.0923

6.5.4 Conclusions

Although the problem of natural convection in a Darcy porous media has been solved many times and also by the same numerical method, it is presented here for the sake of comparison between two algorithms. It was shown that the accuracy is almost the same for the pressure Poisson as well as for the CBS solution procedure. The calculation time for steady-state solution with CBS algorithm, however, is larger than in the case of the pressure Poisson procedure. For the calculation of Darcy porous media flow this observation is logical, since one derivative more has to be evaluated. For the problems using extended Darcy equation e.g. Darcy-Brinkman equation or even where a porous-liquid layer is involved, the calculation time would be very similar or even smaller than the one for the pressure Poisson algorithm. In addition, the CBS algorithm is compact and the determination of artificial compressibility is straightforward, as opposed to finding the best relaxation factor in the pressure Poisson equation. It should be also noted that the procedures are not entirely comparable. While the code for the pressure Poisson equation solves the steady-state Darcy equation, the CBS procedure uses the Darcy equation in time-dependent form as defined in equation (4.59).

If DAM is compared to other mesh-reduction (DRBEM) and meshless (RBFCM) methods we can conclude that the DAM results are more accurate than those obtained by RBCFM at the same grid density. DRBEM results are a little bit more accurate than DAM results. However, the calculation time for the same problem calculated on a 30×30 grid takes a few minutes for DAM, while the calculation time for RBFCM is on the order of a few hours and on the order of several days for DRBEM.

At the end, a problem with irregular geometry is presented. The intention of the described geometry is to present the ability of meshless methods to cope with more complicated geometries. At this point we should emphasize that the distribution of domain and boundary nodes have to be *carefully* positioned. Otherwise the convergence of the simulation is very difficult to achieve.

6.6 Double diffusive natural convection in a composite fluid-porous layer

In nature fluids are not always pure, but composed of more than one substance. When such a fluid is subjected to thermal and compositional gradients, natural convection is induced by combined thermal and solutal buoyancy forces. This physical phenomena is called *double diffusive natural convection*. Both can drive the fluid in the same direction, or they can work in the opposite direction thus slowing down the movement of the fluid. In many industrial or environmental applications, a fluid can occupy a domain which is also partly characterized by porous media and thus termed a *composite fluid-porous layer*. As an example, in a solidification process there is a mushy zone between solid and liquid phase. The mushy zone is characterized by columnar or dendritic forms which can be macroscopically described as a saturated porous media. A good understanding of heat and mass transfer between the mushy and fluid regions may be of great importance in material processing because the internal structure and mechanical properties of the solid results from this coupling [Prescott and Incropera, 1996]. Similar problems are observed in nature for solute exchange in sediments in coastal fields, where transport phenomena take place at an interface between a fluid phase and a porous medium [Webster *et al.*, 1996]. The flow structure can vary greatly in such systems, depending on the governing parameters. The problem is characterized by a large number of parameters with interdependent influences. Therefore, the influences cannot be analyzed individually. The aim of this study, more thoroughly described in [Gobin *et al.*, 1998] and [Neculae, 2003], is to analyze the effect on heat and species transfer of a binary solute in a composite fluid-porous layer enclosed in a rectangular cavity. Until now, similar problems have been numerically analyzed for single porous Darcy-Brinkman [Goyeau *et al.*, 1996] or fluid layer [Gobin and Bennacer, 1996a] alone. The only results for double diffusive convection in a two layer cavity, which are also used for comparison, are provided by [Gobin *et al.*, 1998].

To avoid the uncertainties linked to programmers or computational errors it is important to have other results possibly provided by different authors and even better by different numerical methods. Therefore, the motivation for choosing the problem of double diffusive natural convection in a composite fluid-porous layer lies in its physical as well as numerical complexity.

6.6.1 Problem description

The binary fluid occupies a two-dimensional rectangular cavity, which is filled with porous media along the left vertical wall. The thickness of the porous layer is denoted by the dimensionless distance $\tilde{p}_x^{por}$. The porous layer is assumed to be isotropic and homogeneous and saturated with binary fluid. Different and uniform temperatures and concentrations are specified at the vertical walls and zero fluxes at the horizontal walls of the enclosure (see Figure 6.29). Further assumptions and simplifications are:

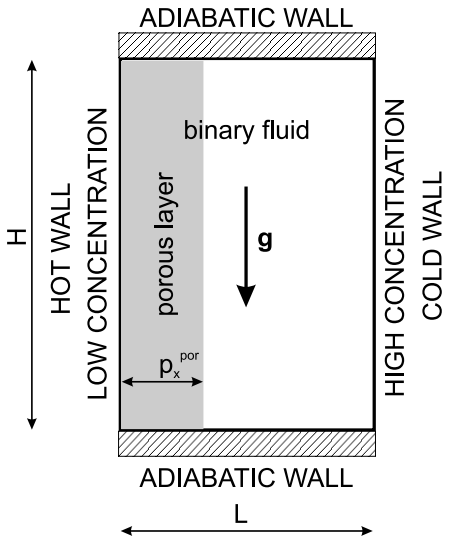


Figure 6.29: Geometry of enclosure for double diffusive natural convection in a composite fluid-porous layer.

- the flow is laminar
- the fluid is incompressible and Newtonian
- the porous matrix is in thermodynamic equilibrium with the fluid
- buoyancy forces are approximated by the Boussinesq approximation

$$\varrho = \varrho_0 \left[1 - \beta_T \left(T - T_0\right) - \beta_C \left(C - C_0\right)\right], \tag{6.39}$$

where $\beta_T > 0$ and $\beta_C < 0$ are the coefficients of thermal and solutal expansion, respectively, and T_0, C_0 and ϱ_0 are reference temperature, concentration, and density, respectively.

The problem is defined by coupled momentum, energy and species conservation equations. Furthermore, the physical problem is governed by the

different macroscopical properties of each layer. In this case two different domains and consequent possible strong non-linearity at the porous/fluid interface exist. Two different approaches are generally proposed: the two-domain approach and the one-domain approach. In the two-domain approach the fluid layer and the porous layer are considered separately. The Navier-Stokes equation is written in the fluid, while the flow in the porous medium is governed by the Darcy law (or one of its extensions; the Darcy-Brinkman formulation is used in the present work). In this case one must write the continuity of the velocity and of the shear stress at the interface. On the other hand we can use the one domain approach, where the fluid is considered as a pseudo-fluid and the composite region is treated as a continuum. This leads to solving only one modified Navier-Stokes equation, which also includes the Darcy term. The Darcy term in one-domain formulation dominates over other terms with lower permeability of porous medium. In this equation, the transition from the fluid to the porous medium is achieved through spatial variation of the permeability. This formulation has been widely used in previous numerical computations since it avoids explicit consideration of the conditions at the fluid/porous interface. In our case and in the reference case the one-domain approach was used.

6.6.2 Governing equations and solution procedure

Again, the case of double-diffusive natural convection in a composite porous-fluid layer is defined in dimensionless formulation. Since the one-domain approach was chosen, the appropriate momentum equation (3.15) with assumed incompressible fluid defined by

$$\tilde{\nabla} \cdot \tilde{\boldsymbol{V}} = 0 \tag{6.40}$$

can be re-cast into dimensionless form

$$\frac{1}{\mathrm{Pr}}\left[\frac{1}{\epsilon}\frac{\partial \tilde{V}}{\partial \tilde{t}} + \frac{1}{\epsilon^2}(\tilde{\boldsymbol{V}} \cdot \tilde{\nabla})\tilde{\boldsymbol{V}}\right] = -\frac{1}{\mathrm{Pr}}\tilde{\nabla}\tilde{P} - \frac{1}{\mathrm{Da}}\tilde{\boldsymbol{V}} + \Lambda\tilde{\nabla}^2\tilde{\boldsymbol{V}} + \mathrm{Ra}_T\,(\tilde{T} + \mathrm{N}\,\tilde{C})\,\boldsymbol{k}. \tag{6.41}$$

The definitions for $\tilde{P}$ and $\tilde{\boldsymbol{V}}$ are listed in Chapter 6.5. Dimensionless time is

$$\tilde{t} = \frac{\alpha_{th}\,t}{H^2} \tag{6.42}$$

and dimensionless viscosity Λ is defined as

$$\Lambda = \frac{\mu_{eff}}{\mu}. \tag{6.43}$$

The energy conservation equation in dimensionless form is

$$\frac{\partial \tilde{T}}{\partial \tilde{t}} + \tilde{\nabla} \cdot (\tilde{\boldsymbol{V}}\, \tilde{T}) = \tilde{\nabla}^2 \tilde{T}, \tag{6.44}$$

with dimensionless temperature

$$\tilde{T} = \frac{T - T_{ref}}{(T_{max} - T_{min})}, \tag{6.45}$$

and reference temperature $T_{ref} = 1/2\,(T_{max} - T_{min})$. In addition this problem is characterized by the dimensionless species conservation equation

$$\frac{\partial \tilde{C}}{\partial \tilde{t}} + \tilde{\nabla} \cdot (\tilde{\boldsymbol{V}}\, \tilde{C}) = \frac{1}{\mathrm{Le}} \tilde{\nabla}^2 \tilde{C}. \tag{6.46}$$

Dimensionless concentration is defined as

$$\tilde{C} = \frac{C - C_{ref}}{(C_{max} - C_{min})}, \tag{6.47}$$

where $C_{ref} = 1/2\,(C_{max} - C_{min})$ is reference concentration. Other dimensionless numbers are:

$$\mathrm{Ra_T} = \frac{g\,\beta\,\Delta T\,H^3}{\nu\,\alpha_{th}}, \tag{6.48}$$

$$\mathrm{Pr} = \frac{\nu}{\alpha_{th}}, \tag{6.49}$$

$$\mathrm{Sc} = \frac{\nu}{D}, \tag{6.50}$$

$$\mathrm{Da} = \frac{\mathcal{K}}{H^2}, \tag{6.51}$$

$$\mathrm{N} = \frac{\beta_C\,\Delta C}{\beta_T\,\Delta T}, \tag{6.52}$$

$$\mathrm{Le} = \frac{\alpha_{th}}{D}, \tag{6.53}$$

$$\tilde{p}_x^{por} = \frac{p_x^{por}}{L}, \tag{6.54}$$

$$\tag{6.55}$$

where Ra_T is thermal Rayleigh number, Pr is Prandtl number and Sc is Schmidt number, Da is Darcy number, N is buoyancy ratio, Le is Lewis number, $\tilde{p}_x^{por}$ is dimensionless width of porous medium, $\mathcal{K}$ is permeability of the porous medium, respectively, α_{th} is thermal diffusivity, D is diffusivity, H is height of the cavity, L is the width of the cavity, and p_x^{por} is dimensional

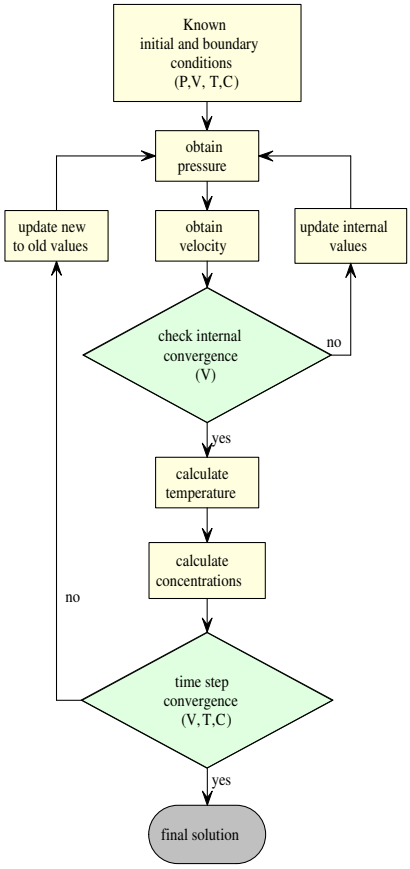


Figure 6.30: Calculation flowchart for double diffusive natural convection in fluid-porous layer.

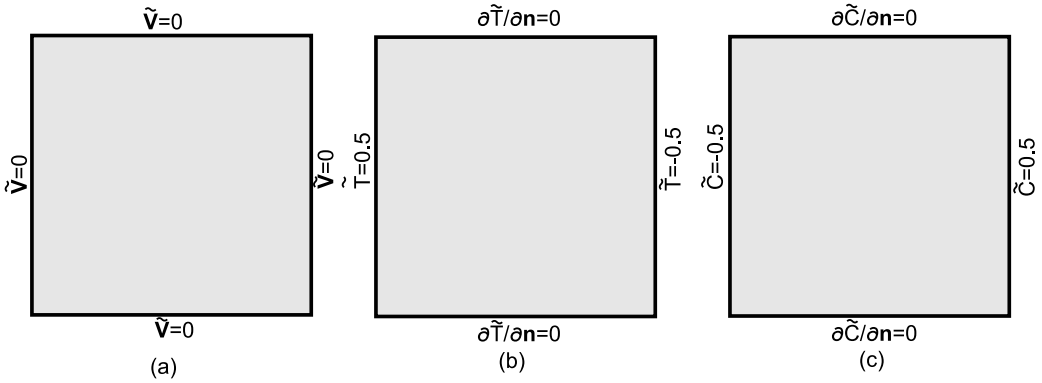


Figure 6.31: Velocity, temperature, and concentration boundary conditions for double diffusive natural convection in fluid-porous layer.

width of porous medium. The solution procedure is as described in flow chart

Table 6.14: Simulation parameters for double diffusive natural convection in fluid-porous layer.

parameter	label	value
number of nodes	$N_x \times N_y$	80×160
false time step	$\Delta\tilde{t}$	$2 \cdot 10^{-7}$
velocity convergence criteria (internal)	$\eta_{int,\tilde{V}}$	10^{-6}
velocity convergence criteria (steady-state)	$\eta_{con,\tilde{V}}$	10^{-6}
temperature convergence criteria	$\eta_{\tilde{T}}$	10^{-8}
concentration convergence criteria	$\eta_{\tilde{C}}$	10^{-8}
domain of influence	I	9
weight function free parameter	c	15

6.30. Simulation parameters used are listed in Table 6.14. All equations are solved using the CBS algorithm. Velocity, thermal, and concentration boundary conditions are presented in Figure 6.31. Pressure boundary conditions at all boundaries are derived from equation (6.41) multiplied by the boundary normal $\boldsymbol{n}$.

The problem of double diffusion natural convection in a composite fluid-porous layer is characterized by many parameters. Flow structure depends on the variation of all parameters, but it has been shown in [Gobin *et al.*, 1998] that variation of permeability in the porous layer has the most interesting and strong influence. For this reason, two sets of calculations are made based on the following parameters:

Table 6.15: Parameters for double diffusive convection in a composite porous-fluid layer.

Ra=10^6, Pr=10, Le=100, A=2, $\tilde{p}_x^{por}$=0.1					
N=1			N=10		
Da=10^{-2}	Da=10^{-5}	Da=10^{-8}	Da=10^{-2}	Da=10^{-5}	Da=10^{-8}
Case 1	Case 2	Case 3	Case 4	Case 5	Case 6

6.6.3 Numerical results

6 different cases are calculated in this chapter. All are characterized by the same Ra, Pr, Le, and $\tilde{p}_x^{por}$ numbers and aspect ratio A as can be seen in Table 6.15. The first set (Cases 1 to 3) is characterized by the equal influence of temperature and concentration field on buoyancy, i.e. N number is set to 1. Concentration buoyancy forces in the second set (Cases 4 to 6) are 10 times larger than for thermal forces. Both sets are characterized by three different values of Da numbers, Da=10^{-2}, Da=10^{-5}, and Da=10^{-8}. At Da=10^{-2} the porous layer gives almost no resistance to flow, while Da=10^{-8} represents a nearly impermeable layer. In this range both limiting cases are covered. Results are compared in terms of Nusselt and Sherwood numbers along the vertical walls which represent the dimensionless heat and mass transfer. The Nusselt number Nu is defined in (6.37). Similarly, the Sherwood number is

$$\mathrm{Sh} = \frac{\int_{p_y^-}^{p_y^+} D \frac{\partial C(p_x^-, p_y)}{\partial x} \, \mathrm{d}y}{A \, D \, \Delta C}, \tag{6.56}$$

with p_y^- and p_y^+ denoting minimum and maximum coordinate in the y direction, respectively, p_x^- the minimum coordinate in the x direction, and A is the aspect ratio defined in Chapter 6.5. In Figure 6.32 comparisons of stream lines, concentration and thermal field are shown for Case 2 and Case 3. The comparison shown in Figure 6.34 describes good agreement between the results obtained by DAM and FVM although the number of calculation nodes is approximately three times lower in the case of DAM. At N=1 no change of flow pattern is observed (see streamlines in Figure 6.32).

It has been shown in [Gobin *et al.*, 1998] and [Neculae, 2003] that more complex flow patterns are observed at N=10, where the flow structure is modified with increasing permeability of the porous layer. At Da=10^{-8} (Case 6) the flow is organized in one cell. While the Sherwood number is increasing monotonically with the most intense transition between Da=10^{-7} and Da=10^{-3} (Figure 6.35a), the Nusselt number undergoes one or more minima. The first local minimum is observed at approximately Da=$1.5{\cdot}10^{-7}$ and the second at Da=$1{\cdot}10^{-5}$ (see Figure 6.35b). Along with the local minima of Nusselt numbers flow also is reorganized into a larger number of cells.

The streamlines obtained by DAM are presented at Da=$1{\cdot}10^{-5}$ (Case 5), where flow pattern is reorganized from one cell into two cells (Figure 6.33b) and at Da=$1{\cdot}10^{-2}$ (Case 4) where a transition from two to three cells has occurred (Figure 6.33c). Tabulated results are listed in Table 6.16.

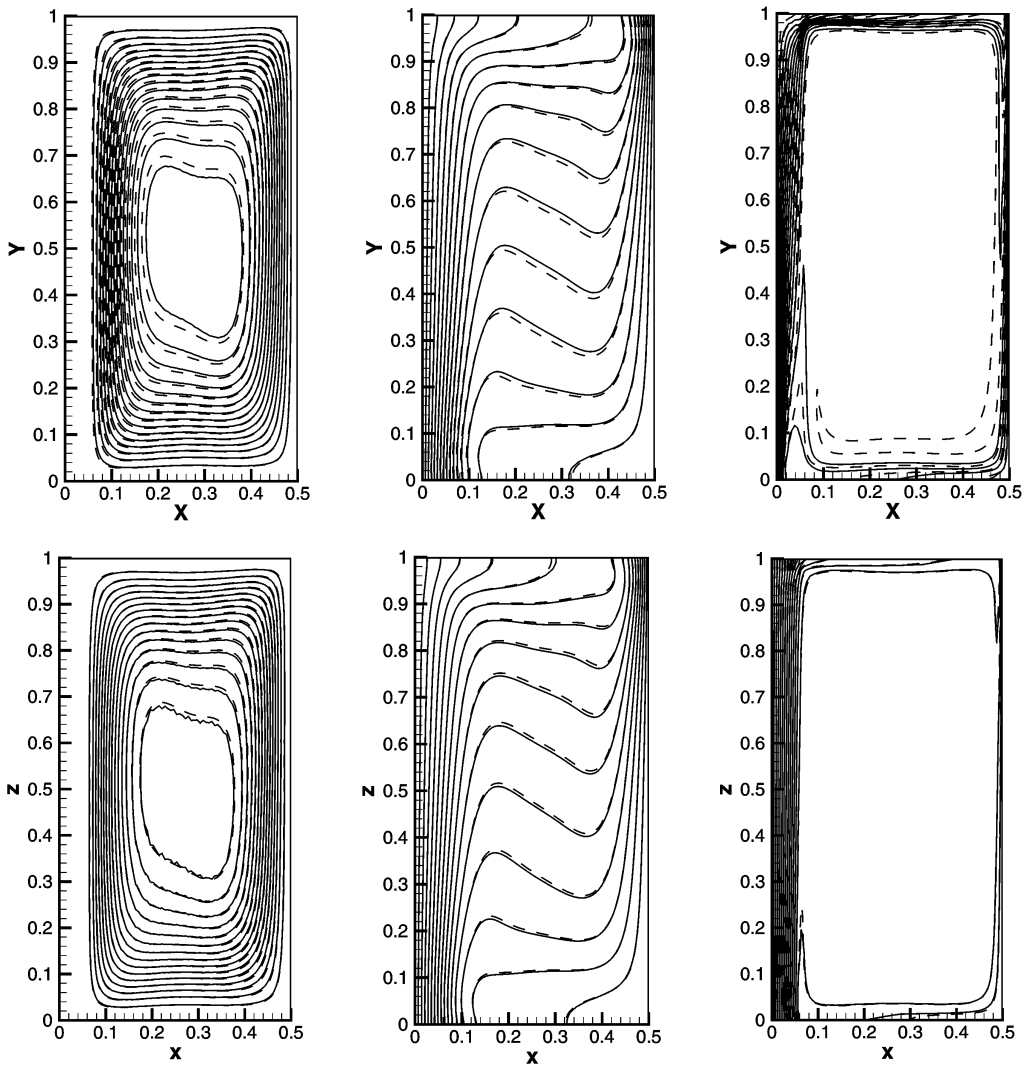
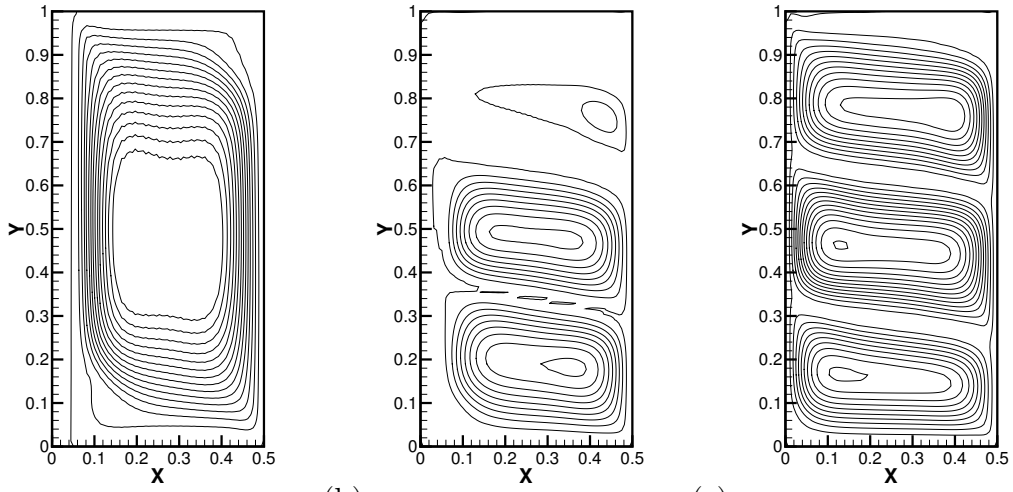


Figure 6.32: Comparisons of streamlines (left), temperature fields (mid), and concentration fields (right) for N=1 at $Da = 10^{-5}$ (top) and $Da = 10^{-8}$ (bottom). Solid lines represent DAM solutions and dashed lines FVM solutions.



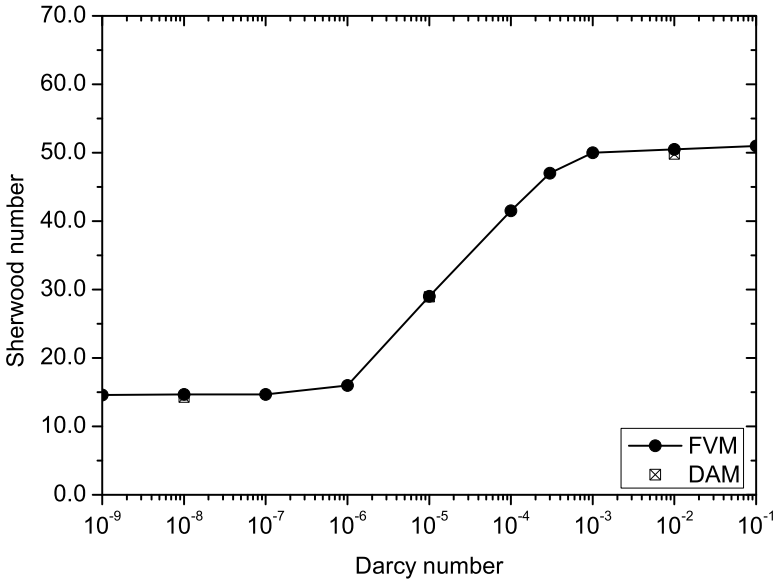
(a) (b) (c)

Figure 6.33: Streamlines for N=10 at $Da = 10^{-8}$ (a), $Da = 10^{-5}(b)$, and $Da = 10^{-2}$ (a). Contours range from values ψ =-10 to ψ =0 with equal spacing $\Delta\psi$.

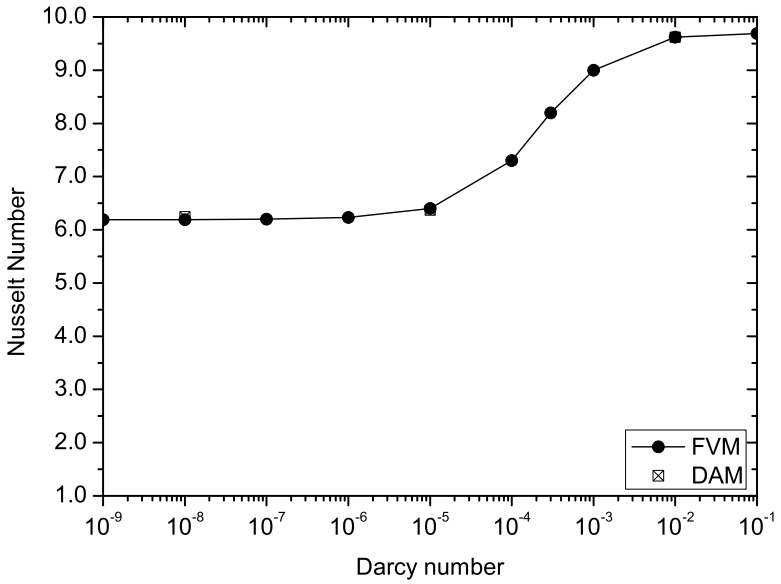
Table 6.16: Tabulated results for $\mathrm{Ra}_T = 10^6$, Pr=10, Le=100, A=2, $\tilde{p}_x^{por}$=0.1.

N=1				
	Nusselt number		Sherwood number	
	DAM	FVM	DAM	FVM
Da=10^{-2}	9.62	9.62	49.80	50.50
Da=10^{-5}	6.37	6.40	29.93	29.00
Da=10^{-8}	6.25	6.19	14.25	14.66

N=10				
	Nusselt number		Sherwood number	
	DAM	FVM	DAM	FVM
Da=10^{-2}	7.58	7.72	59.30	63.53
Da=10^{-5}	4.59	4.71	45.90	48.18
Da=10^{-8}	5.56	5.65	14.25	14.67

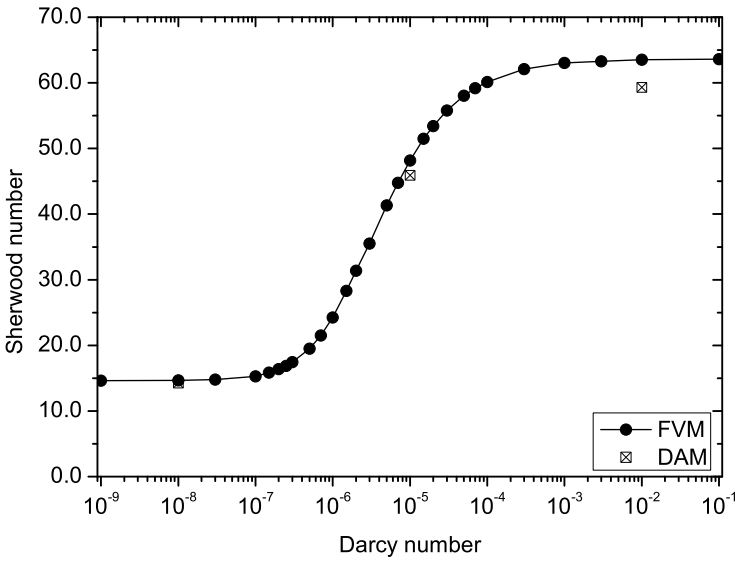


(a)

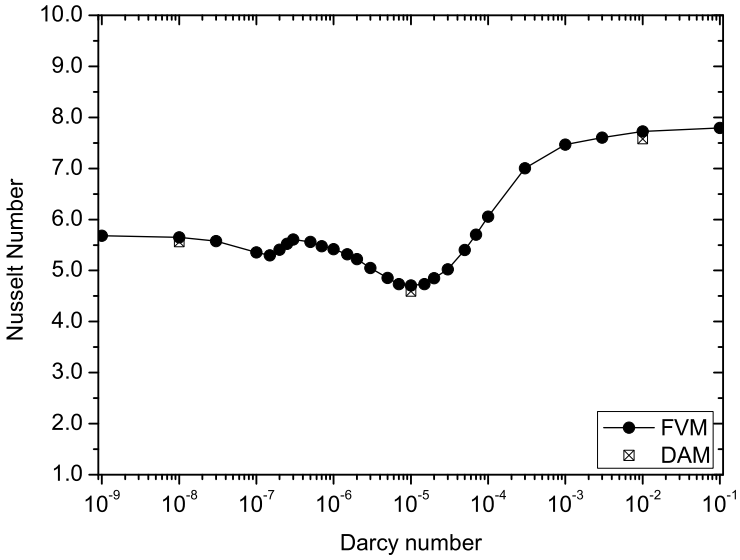


(b)

Figure 6.34: (a) The influence of the Darcy number on the average mass transfer, (b) The influence of the Darcy number on the average heat transfer at N=1.



(a)



(b)

Figure 6.35: (a) The influence of the Darcy number on the average mass transfer, (b) The influence of the Darcy number on the average heat transfer at N=10.

6.6.4 Conclusions for physical part

From the calculations shown in this chapter it can be observed that the flow structure can change considerably according to the conditions. However, the conditions for more complex flow structures are met only in a narrow spectrum of parameters. At buoyancy ratio N=1 the conditions for flow change are not met. With increasingly intense conditions (at higher N number and consequently larger Ra_C number) the flow pattern is modified to several cells. The full understanding of the problem is still not very clear. However, the explanation of [Gobin *et al.*, 2004] is that the evolution of the convective flow with increasing permeability results from the competition between two opposing effects. First, the higher permeability results in a better penetration of the flow into the porous layer and larger effect of the advection of the imposed temperature and concentration difference in the layer. The effective temperature and composition gradients governing the buoyancy is expected to grow and consequently force the flow to be accelerated. This results in higher heat and mass transfer which can be described by the Sherwood number (Figure 6.35(a)). On the other hand, due to larger thermal diffusivity over mass diffusivity, the central recirculation loop driven on the scale of the thermal boundary layer thickness is driven by a relatively smaller temperature difference. Locally, the thermal buoyancy force decreases and so does the intensity of the internal thermal loop and consequently average heat transfer. In the neighborhood of the first minimum in Figure 6.35(b) one may observe that the decrease in the Nusselt number is caused by a decrease of the main recirculation cell, due to the formation of a low velocity zone seen in Figure 6.33(b), where the heat transfer is mainly conductive. This *stagnant zone* is compositionally stratified as in typical double diffusive problems [Gobin and Bennacer, 1996a], while the heat transfer in this region is mainly conductive. With the increasing height of the stagnant zone, the local vertical concentration gradient decreases and becomes destabilized by the lateral temperature gradient, resulting in the formation of a secondary co-rotative cell and a sudden increase of the Nusselt number. With the decreasing permeability, again the formation of a stagnant cell and the consequent transition to three cells (Figure 6.33(c)) with corresponding increase of Nusselt number at $\mathrm{Da} \approx 2 \cdot 10^{-7}$ is observed. The evolution of the process is illustrated in Figure 6.35(b).

6.6.5 Conclusions for numerical part

The problem in question is important for this work because it also deals with the problems connected with environmental issues. In addition to heat

transfer, the species transfer is included as well. Although in many cases the species transport does not affect the flow structure (i.e. radionuclide or pesticide transport), it sometimes plays an important role as for example in estuaries. From the numerical point of view the problem is characterized by high Rayleigh numbers (Ra_T=10^6, Ra_C=10^9) and also very thin solutal boundary layers (high Schmidt number). These conditions are very likely to induce computational instabilities. The minimal number of grid points needed for simulation to converge for N=10 and Da=10^{-2} is 80×160 (A=2). This by itself shows the complexity of the problem. For N=10 and at higher Darcy numbers, the flow structures are quite complex and differences between methods are more visible. Still, the Nusselt number exhibits the same behavior and comparisons between the two methods show quite good agreement. Results for N=10 should be calculated on a finer grid to improve accuracy and to achieve better agreement with the reference solution.

7 Advanced topics on DAM

In Chapter 6 we presented the capability of DAM to solve complex and large scale physical problems. Nevertheless, the involved discretization of the physical domain was simple and can be calculated by any traditional method with the exception of the case of natural convection in the domain with holes. The main objective or advantage of meshless methods, however, is to solve problems on geometrically complex geometries and by arbitrary discretization of the physical domain. To achieve this goal, several issues typical for DAM (which can be extended to other meshless methods as well) are elaborated in this chapter. One of the most influential on stability and accuracy is certainly the choice of weight function. In addition, weight function is always characterized by a free parameter which is user-defined. For all previous cases, calculated on uniform grids or on slightly non-uniform grids, the value of the free parameter is the same in each subdomain ${}_n\Omega$. The idea about finding the most optimal weight function free parameter for non-uniform grids is presented and tested for different node arrangements.

The shape functions of local meshless methods are constructed from the nodes in the domain of influence or support. For this reason, the impact of the number of nodes in the domain of influence in terms of stability, accuracy and CPU time is studied.

In contrast to some traditional methods (FVM), meshless methods are not locally conservative. Therefore, some possible implementations of diffusive and advective terms are given at the end of this chapter.

7.1 Weight function

Weight functions $\widehat{W}$ play an important role in meshless methods. The influence of neighboring nodes $\boldsymbol{p}_i$ on the calculated node $\boldsymbol{p}_n$ is expressed in terms of weight function in each n'th subdomain ${}_n\Omega$ which has a strong influence on the accuracy, stability and conditioning of matrix $\mathbf{A}$ in (5.3). In general, the weight function has to satisfy the following four conditions

1. ${}_n\widehat{W}(\boldsymbol{p}-\boldsymbol{p}_i) > 0;\ \forall \boldsymbol{p}$ over ${}_{\mathrm{n}}\Omega$
2. ${}_n\widehat{W}(\boldsymbol{p}-\boldsymbol{p}_i) = 0;\ \forall \boldsymbol{p}$ outside ${}_{\mathrm{n}}\Omega$
3. $\widehat{W}$ is a monotonically decreasing function

The first condition, namely positivity, is not a mathematical function representation requirement, but is important to ensure a stable presentation of physical phenomena [Liu, 2003]. The second condition, compactness, is important because it enables the approximation to be generated from a local representation of nodes, i.e. from the neighboring nodes. The last condition is, again, not a mathematical requirement, but it is imposed on the physical consideration that more distant nodes have less influence than the nodes in the vicinity.

The choice of the weight function is more or less arbitrary as long as the weight function satisfies the conditions of positivity and compactness for local numerical methods. In the literature several different types of weight functions are given. Explicitly for DAM, the most extensively used (see for example [Sadat *et al.*, 1996], [Sophy and Sadat, 2002], ...) weight function is Gaussian

$$\widehat{W}(\boldsymbol{p}-\boldsymbol{p}_n) = \begin{cases} e^{-c\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|^2}{\sigma^2}} & ;\ \|\boldsymbol{p}-\boldsymbol{p}_n\| \le \sigma \\ 0 & ;\ \|\boldsymbol{p}-\boldsymbol{p}_n\| > \sigma \end{cases}, \tag{7.1}$$

where c is user-defined free parameter and $\|\boldsymbol{p}-\boldsymbol{p}_n\|$ is the Euclidian distance between two nodes $\boldsymbol{p}$ and $\boldsymbol{p}_n$. Parameter σ is defined as the largest Euclidian distance between node $\boldsymbol{p}$ and all other nodes considered in ${}_n\Omega$. [Belytscko *et al.*, 1996a] also used non free parameter splines: cubic spline and quadratic spline, defined as

$$\widehat{W}(\boldsymbol{p}-\boldsymbol{p}_n) = \begin{cases} \frac{2}{3} - 4\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|^2}{\sigma^2} + 4\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|^3}{\sigma^3} & ;\ \frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|}{\sigma} \le \frac{1}{2} \\ \frac{4}{3} - 4\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|}{\sigma} + 4\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|^2}{\sigma^2} - \frac{4}{3}\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|^3}{\sigma^3} & ;\ \frac{1}{2} < \frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|}{\sigma} \le 1 \\ 0 & ;\ \frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|}{\sigma} > 1 \end{cases} \tag{7.2}$$

for cubic spline and

$$\widehat{W}(\boldsymbol{p}-\boldsymbol{p}_n) = \begin{cases} 1 - 6\,\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|^2}{\sigma^2} + 8\,\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|^3}{\sigma^3} - 3\,\frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|^4}{\sigma^4} & ;\ \frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|}{\sigma} \leq 1 \\ 0 & ;\ \frac{\|\boldsymbol{p}-\boldsymbol{p}_n\|}{\sigma} > 1 \end{cases} \tag{7.3}$$

for quadratic spline. In fact the spline functions as well have a free parameter although it is hidden in the scaling factor. In this work the Gaussian weight function is used because of its simple adaptivity. Until now the scaling parameter c was usually heuristically defined by authors and it depended mostly on the grid density and on the distribution of nodes. In the case of randomly distributed nodes the parameter value can vary substantially. In the literature the range of proposed free parameter values vary from author to author. Explicitly for DAM, the heuristically defined c parameter ranges from $\ln 100 \approx 4.6$ [Prax *et al.*, 1998] to 6.25 found in [Belytscko *et al.*, 1996a] or almost 7 [Sadat and Couturier, 2000]. In general, for non-uniform grids, the free parameter may vary from one set of nodes to another. In addition, if the nodes are distributed non-uniformly, the free parameter can be different in different directions. For example consider a 2D problem with 9 support nodes. In figure 7.1a uniform distribution is presented. The most appropriate weight function for this case is intuitively radially symmetrical, as shown in Figure 7.1b. In Figure 7.2a uniform distribution of nodes is still shown,

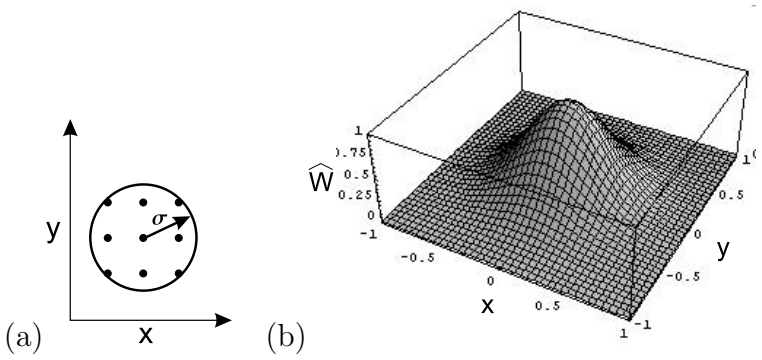


Figure 7.1: Uniform distribution. (a) Node distribution, (b) Weight function schematic.

but the grid spacing is different in the x and y directions. To obtain better conditioning of the matrix and consequently more accurate derivatives the weight function for this case should be different in each direction. In other words, the parameter c has to be larger in the x direction than in the y direction as shown in Figure 7.2b. Formulation of the weight function (7.1)

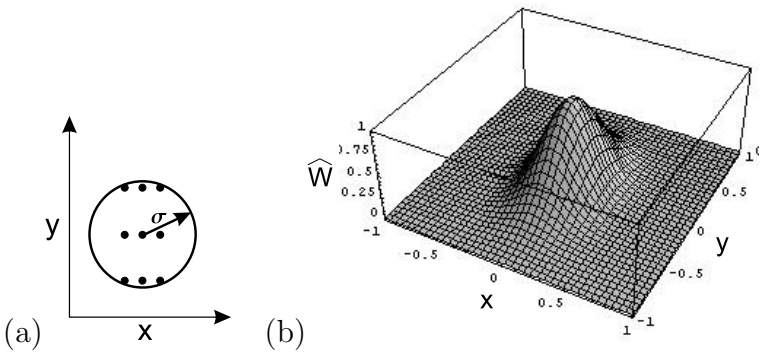


Figure 7.2: Uniform non-equidistant distribution. (a) Distribution of nodes, (b) Weight function schematic.

for the described case can be extended into

$$\widehat{W}(\boldsymbol{p}-\boldsymbol{p}_n) = \begin{cases} e^{-c_x \frac{(p_x - p_{x\,n})^2}{\sigma^2} - c_y \frac{(p_y - p_{y\,n})^2}{\sigma^2}} & ; \|\boldsymbol{p}-\boldsymbol{p}_n\| \leq \sigma \\ 0 & ; \|\boldsymbol{p}-\boldsymbol{p}_n\| > \sigma \end{cases} , \tag{7.4}$$

7.1.1 Optimization of weight function free parameter

The main problem of the construction of proper shape function is the local selection of free parameters c_x and c_y. In previous Chapter 6 single value of parameter $c = 15$ was used. Here, the focus is on finding the optimum c value for arbitrary grid. All cases consider 9 nodes in the domain of influence. The idea to adjust free parameters locally in each subdomain is developed here for the first time.

The procedure consists of three main steps. First, the measure against which the parameters are compared to has to be found. This measure is called the *reference quality function*. The reference quality function is used for evaluation of its numerical derivatives on uniform and non-uniform grids. Second, a function called the *object function*, which needs to be minimized, has to be constructed. The object function is constructed from the derivatives found by the evaluation of the reference quality function. And finally, the parameter values are evaluated using the appropriate *optimization procedure*. A schematic of the entire procedure is given in Figure 7.3. All steps are explained in more details below.

Reference quality function

The reference quality function $\mathcal{F}$ has to be a positive definite function on subdomain ${}_n\Omega$. The reference quality function should not be any of the used

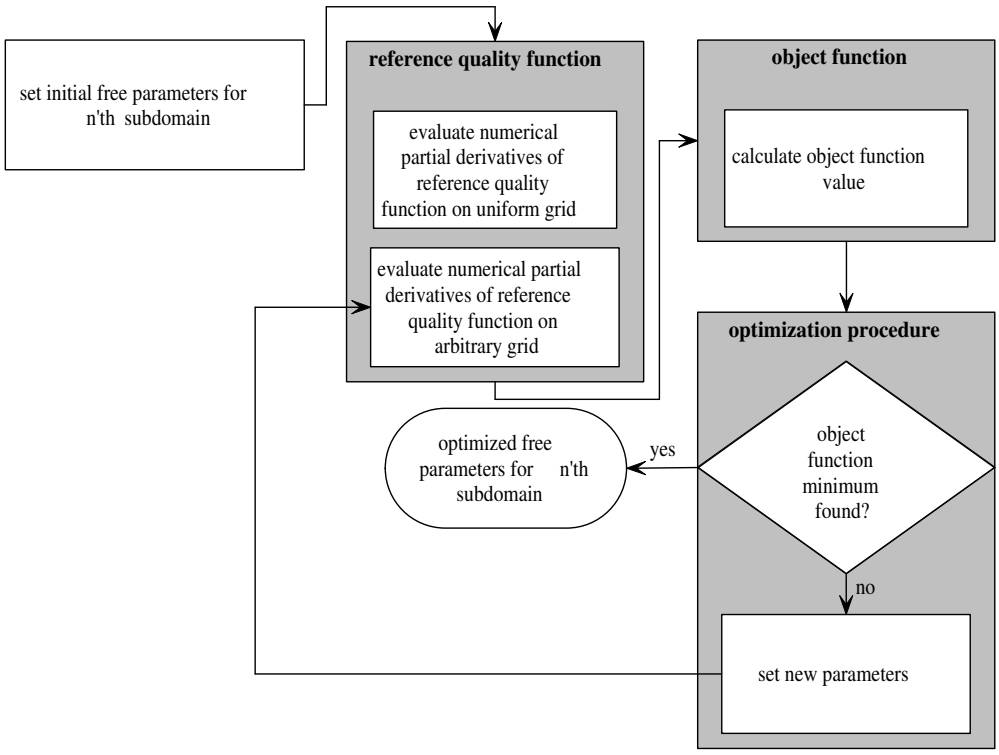


Figure 7.3: Free parameter optimization procedure flowchart.

basis functions. In this work we define $\mathcal{F}$ in such a way that $\mathcal{F}(\boldsymbol{p}-\boldsymbol{p}_n) = 1/\widehat{W}(\boldsymbol{p}-\boldsymbol{p}_n)$ inside ${}_n\Omega$ and 0 outside ${}_n\Omega$. Explicitly, the reference quality function is

$$\mathcal{F}(\boldsymbol{p}-\boldsymbol{p}_n) = \begin{cases} e^{c_x \frac{(p_x - p_{x\,n})^2}{\sigma^2} + c_y \frac{(p_y - p_{y\,n})^2}{\sigma^2}} & ; \|\boldsymbol{p}-\boldsymbol{p}_n\| \leq \sigma \\ 0 & ; \|\boldsymbol{p}-\boldsymbol{p}_n\| > \sigma \end{cases} . \tag{7.5}$$

The specific reference quality function in this work is chosen because the right-hand side values cancel and only basis functions remain. Because the domain of influence usually contains a relatively low number of nodes and because the chosen reference quality function is relatively steep, the second derivatives of $\mathcal{F}$ calculated on a given non-uniform grid cannot be compared to the analytically calculated partial derivatives. For this reason our second partial derivatives are obtained via least squares approximation of the reference quality function calculated on a uniform grid. The approximation of the reference quality function on the uniform grid is denoted by $\mathcal{F}_{uni}$, while the approximation of the reference quality function calculated on the posed irregular grid is marked by $\mathcal{F}_{irr}$. Both approximations of the functions and the

partial derivatives are obtained from equations (5.1-5.11). The dependence on grid density is alleviated by scaling of Euclidian distance between nodes in the basis functions by the radius of influence domain in each subdomain ${}_n\Omega$. Coefficients α are thus calculated from (5.3) by the least square approximation, where the right-hand side vector is due to our choice of reference quality function reduced from

$$ {}_n\hat{\phi}_j = \sum_{i=1}^{I} \varphi_j((\boldsymbol{p}_i - \boldsymbol{p}_n)/\sigma)\, {}_n\widehat{W}(\boldsymbol{p}_i - \boldsymbol{p}_n) \cdot \frac{1}{{}_n\widehat{W}(\boldsymbol{p}_i - \boldsymbol{p}_n)}, \tag{7.6} $$

to

$$ {}_n\hat{\phi}_j = \sum_{i=1}^{I} \varphi_j((\boldsymbol{p}_i - \boldsymbol{p}_n)/\sigma). \tag{7.7} $$

The approximation of the reference quality function on the uniform grid $\mathcal{F}_{uni}$ is calculated with $c_x = c_y = 12.5$ on the grid shown in Figure 7.4. This value is chosen because it gave the most optimal results on all grids tested. The

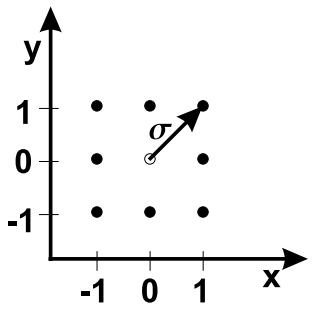


Figure 7.4: Scaled uniform node distribution ($\sigma = \sqrt{2}$).

evaluation of second derivatives on the uniform grid gives the single value $\frac{\partial^2 \mathcal{F}_{uni}}{\partial x^2} = \frac{\partial^2 \mathcal{F}_{uni}}{\partial y^2} = 6400$.

Object function

$\mathcal{J}$ is a function which needs to be minimized. The object function proposed in this work seeks for the minimum difference between the second partial derivatives of the reference quality function calculated on the uniform grid and the second partial derivatives of the same function found on the real grid. Thus we seek for a minimum of a functional $min\{\mathcal{J}(\Phi_{uni}, \Phi_{irr})\}$, where Φ_{uni} and Φ_{irr} are values or n'th derivatives of $\mathcal{F}_{uni}$ and $\mathcal{F}_{irr}$, respectively. The fact is that when solving PDE with meshless methods, the accuracy of the

solution will increase with better reproduction of the surface from known field variables, which leads to more accurate derivatives afterward applied to the given PDE. Therefore if the difference between derivatives of the known reference quality function and reproduced shape function are minimal then the solution of PDE will be most accurate.

The selection of object function has a strong influence on optimization results. However, finding the most appropriate one is not an easy task. Therefore, our object function is constructed from two parts. The first one is based on the assumption that the most accurate derivatives are obtained on a locally uniform grid. All results obtained on arbitrarily arranged grid nodes are then compared to the solution on the uniform grid. The second condition which should be satisfied is that second derivatives with respect to free parameter in the x and y directions are equal. Because errors of the second derivatives due to the truncation are larger, our object function is built only using second derivatives as follows

$$\begin{aligned} \mathcal{J} &= \left(\frac{\partial^2 \mathcal{F}_{irr}}{\partial\, x^2} - \frac{\partial^2 \mathcal{F}_{uni}}{\partial\, x^2}\, \omega_x^4 \right)^2 \\ &+ \left(\frac{\partial^2 \mathcal{F}_{irr}}{\partial\, y^2} - \frac{\partial^2 \mathcal{F}_{uni}}{\partial\, y^2}\, \omega_y^4 \right)^2 \\ &+ 16 \left(\omega_x^2 + \omega_y^2 \right)^2 \cdot \left(\frac{\partial^2 \mathcal{F}_{irr}}{c_x\, \partial\, x^2} - \frac{\partial^2 \mathcal{F}_{irr}}{c_y\, \partial\, y^2} \right)^2 . \end{aligned} \tag{7.8}$$

Object function forcing parameters ω_x and ω_y are introduced to prevent the free parameters from being too different and too extreme and are defined as $\omega_x = c_y/c_x$ and $\omega_y = c_x/c_y$. The first two terms in equation (7.8) are used to minimize the error between the second derivatives obtained on the uniform grid and the second derivatives obtained on the arbitrary grid. The third term in equation (7.8), however, is used to force the second derivatives to be equal with respect to c_x and c_y. Equivalence of the second derivatives is forced more intensively when c_x and c_y are farther apart. The constant 16 is set as a weight between the first two terms and the last term. For one parametric formulation, equation (7.8) simplifies to

$$\begin{aligned} \mathcal{J} &= \left(\frac{\partial^2 \mathcal{F}_{irr}}{\partial\, x^2} - \frac{\partial^2 \mathcal{F}_{uni}}{\partial\, x^2} \right)^2 \\ &+ \left(\frac{\partial^2 \mathcal{F}_{irr}}{\partial\, y^2} - \frac{\partial^2 \mathcal{F}_{uni}}{\partial\, y^2} \right)^2 \\ &+ 64 \left(\frac{\partial^2 \mathcal{F}_{irr}}{c\, \partial\, x^2} - \frac{\partial^2 \mathcal{F}_{irr}}{c\, \partial\, y^2} \right)^2 . \end{aligned} \tag{7.9}$$

The goal is to minimize the object function (7.9) for one parametric weight function or (7.8) for two parametric weight function.

Optimization procedure

The optimization procedure can be chosen by the user. There are many algorithms available [Fortran, 2003]. In cases where $\mathcal{J}$ consists of more than one local minima, special precaution and/or special optimization procedures should be used. The choice of optimization procedure also has a great influence on the optimization time. For testing purposes a simple but robust tree procedure is used:

1. Set the initial value of free parameter c_0.

2. Search for minimum around initial value with large step λ_1: $c^1_{min} = c^0 \pm \lambda_1$.

3. Search for minimum around c^1_{min} with step λ_2: $c^2_{min} = c^1_{min} \pm \lambda_2$; $\lambda_2 = \lambda_1\, \varpi/2$.

4. Refine search around c^2_{min}: $c^3_{min} = c^2_{min} \pm \lambda_3$; $\lambda_3 = \lambda_2\, \varpi/2$.

5. Update new minimum ($c^2_{min} = c^3_{min}$).

6. Return to Step 4 if $c^3_{min} < c^2_{min}$.

7. Exit when condition in Step 6 is not satisfied.

λ_1 is a user-defined step and $\varpi \approx 0.618$ is the golden section value. One parametric and two parametric weight functions are used in optimization procedure in this work. In the latter case the weight function (7.4) has two free parameters, therefore free parameters c and λ are second-order vectors. The starting position is set to $\mathbf{c}_0 = \{15, 15\}$ and initial step $\boldsymbol{\lambda}_1 = \{10, 10\}$. This procedure can be even expanded to four degrees of freedom for optimization of each weight function quadrant in possible future work.

7.1.2 Numerical examples

The procedure proposed above is tested on a single convective-diffusive PDE using Dirichlet boundary conditions.

Problem description

The optimization procedure is tested on the 2D Burger PDE which describes the wave propagation in 2D domain. The Burger PDE is chosen because it contains first and second order derivatives and is time dependent which is important for the calculation of real convection-diffusion physical problems. The second important feature of the Burger equation is that the analytical solution is known. The dimensionless time period considered for calculation is $\tilde{t}_{max}$=4. In this time the "wave" propagates through the whole domain, stretching from 0 to 1 in x and y direction, which has two effects. The first is that each node is subjected to the strong nonlinearity of the propagating wave in a certain time. The second is that errors from one time step to another in an iterative process are accumulated. The Burger PDE is defined as

$$\frac{\partial \phi}{\partial \tilde{t}} + \phi \left(\frac{\partial \phi}{\partial x} + \frac{\partial \phi}{\partial y} \right) = \nu \left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right); \qquad (x, y) \in \Omega \tag{7.10}$$

where $\nu = 0.05$ is the dimensionless diffusion constant. The analytical solution is

$$\phi(\tilde{p}_x, \tilde{p}_y, \tilde{t}) = \frac{1}{1 + \exp\left(\frac{\tilde{p}_x + \tilde{p}_y - \tilde{t}}{2\,\nu} \right)}. \tag{7.11}$$

Boundary conditions are Dirichlet boundary conditions with values from (7.11) at the boundary.

Solution procedure

At initial time $t_0 = 0$ values for ϕ are taken from the analytical solution, thus

$$\phi(\tilde{p}_x, \tilde{p}_y, 0) = \frac{1}{1 + \exp\left(\frac{\tilde{p}_x + \tilde{p}_y}{2\,\mu} \right)}. \tag{7.12}$$

From this initial condition equation (7.10) is temporarily discretized in an explicit manner. Variable $\phi(\tilde{p}_x, \tilde{p}_y)$ at time $t^{\tau+1}$ is calculated explicitly from variable ϕ at time t^{τ} as

$$\phi^{\tau+1} = \phi^{\tau} + \frac{1}{\Delta t} \left(\nu \left(\frac{\partial^2 \phi^{\tau}}{\partial x^2} + \frac{\partial^2 \phi^{\tau}}{\partial y^2} \right) - \phi^{\tau} \left(\frac{\partial \phi^{\tau}}{\partial x} + \frac{\partial \phi^{\tau}}{\partial y} \right) \right), \tag{7.13}$$

where the derivatives are calculated as described in equations (5.1-5.11). Time step Δt is 0.001.

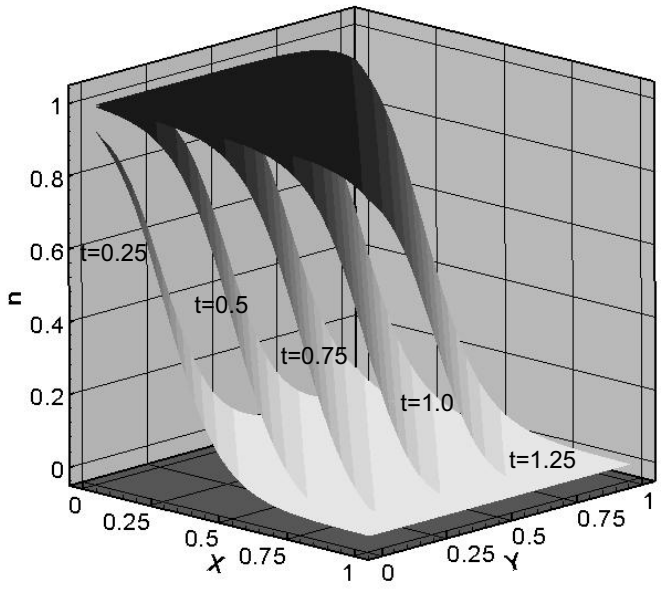


Figure 7.5: Analytical solution of the Burger equation at times 0.25, 0.5, 0.75, 1, and 1.25.

Grids used

The solution of the Burger equation is sought on different node distributions. First, the tests are carried out on random node distribution, and later on grids with finer discretization at the boundaries. Random grids are constructed here in a way that nodes are initially distributed uniformly with fill distance $\delta_{fill,uni}$. From this initial position each node is displaced randomly by δ_x in the x direction and δ_y in the y direction. Displacement is described by $\delta = 0\%$ for no displacement ($\delta_{fill} = \delta_{sep}$), i.e. uniform grid and $\delta = 50\%$ for maximal displacement to half of the initial distance $\delta_{fill,uni}/2$ between the original neighboring nodes ($\delta_{sep} = 0$, $\delta_{fill} = \delta_{fill,uni}/2$). In latter case the nodes can coincide at the same position. Grids with 30×30 domain nodes for $\delta = 0(\%)$ and $\delta = 45\%$ are shown in Figure 7.6 and Figure 7.7, respectively. For the sake of comparison the same seed number is used in the random generator (Fortran RANDOM_NUMBER routine) in all cases [Fortran, 2003].

In practice random grids are not used very frequently. Usually grids are regular, but still non-uniform as presented in Figure 7.8. An example of such grid with finer node distribution near the boundary for improving calcula-

tion accuracy and for better representation of boundary layers is described for example in [Sadat and Couturier, 2000] and here expanded for variable stretching with

$$\boldsymbol{p} = \boldsymbol{p}_{min} + \frac{\boldsymbol{p}_{max} - \boldsymbol{p}_{min}}{2} \left(1 + \frac{\tanh\left(\frac{2\chi \boldsymbol{p}_r - \frac{\boldsymbol{p}_{max} + \boldsymbol{p}_{min}}{2}}{\boldsymbol{p}_{max} - \boldsymbol{p}_{min}} \right)}{\tanh \chi} \right), \quad (7.14)$$

where $\boldsymbol{p}_r$ corresponds to the position of the nodes on an $N \times N$ uniform grid, χ is stretching factor and $\boldsymbol{p}_{max}$ and $\boldsymbol{p}_{min}$ stand for maximum and minimum positions in the domain Ω, respectively.

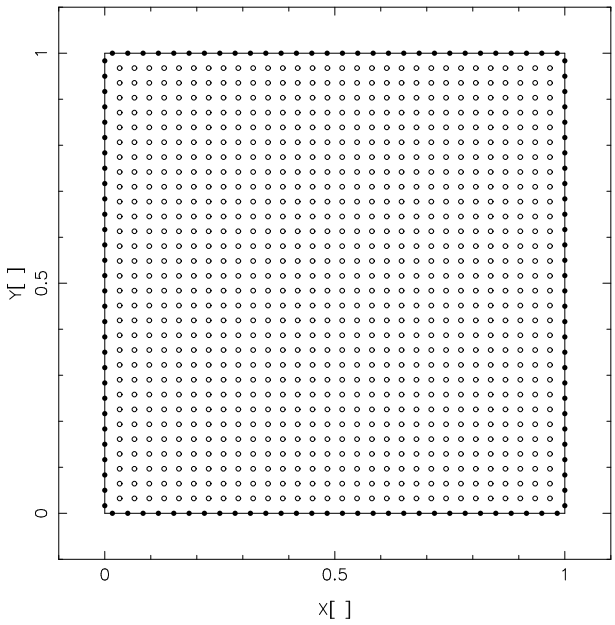


Figure 7.6: 30×30 with $\delta = 0\%$ (uniform) grid.

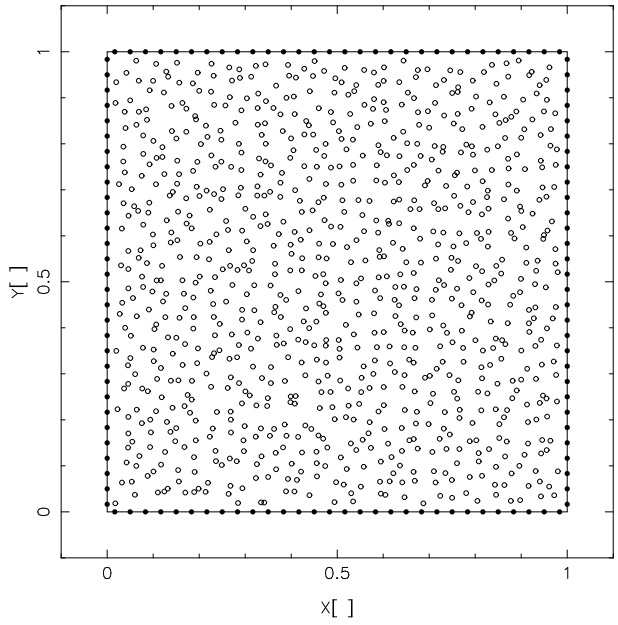


Figure 7.7: 30×30 with $\delta = 45\%$ random grid.

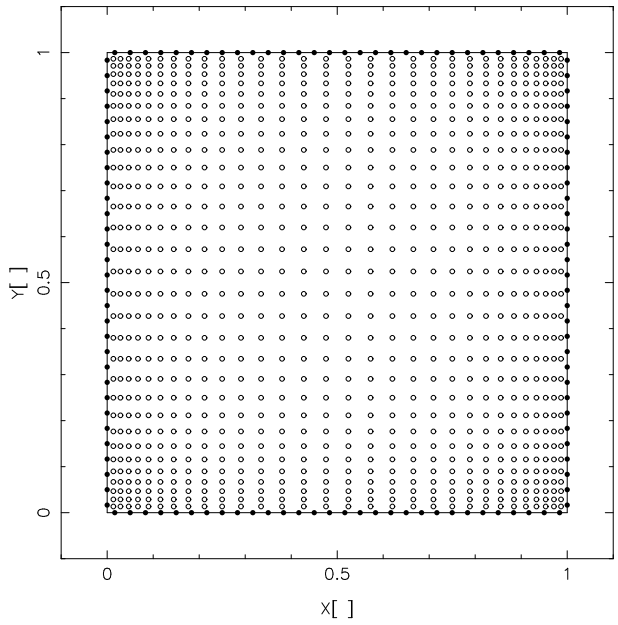


Figure 7.8: 30×30 with stretching factor 1.3 grid.

The following tests have been employed on:

- Random grids, defined by maximal possible displacement δ parameter

  - 20×20 grid with $\delta = 0\%$
  - 30×30 grid with $\delta = 0\%$
  - 40×40 grid with $\delta = 0\%$
  - 20×20 grid with $\delta = 25\%$
  - 30×30 grid with $\delta = 25\%$
  - 40×40 grid with $\delta = 25\%$
  - 20×20 grid with $\delta = 35\%$
  - 30×30 grid with $\delta = 35\%$
  - 40×40 grid with $\delta = 35\%$
  - 20×20 grid with $\delta = 45\%$
  - 30×30 grid with $\delta = 45\%$
  - 40×40 grid with $\delta = 45\%$

- Stretched grid
  - 30×30 grid with $\chi = 1.3$

Errors are calculated in terms of maximal relative error between analytical and numerical solution over the whole domain in each time step

$$\eta(t) = \max \left\{ \frac{|\phi^n_{analytic} - \phi^n_{numeric}|}{\phi^n_{analytic}} \right\}, \tag{7.15}$$

where $t \in \{0, t_{max}\}$ is time, and $n = 1, ..., N$, where N is number of nodes.

Example 1: Uniform grid

In the literature, most of the calculations are performed on uniform grids. It turns out that the results are quite insensitive on a wide range of parameter c (roughly from 10-30) although the error decreases with increasing c as shown in Figure 7.9. The results in terms of $\eta(t)$ are presented for four cases of uniform c parameters and for optimized parameters c_x and c_y. Optimized results are inside this range with value 12.5 as set for the reference solution. Naturally, in this case c_x and c_y are equal after the optimization. Therefore both one parametric and two parametric optimizations gave the same results.

Example 2: Slightly non-uniform grid ($\delta = 25\%$)

Nodes are displaced from the initial uniform distribution by up to 25% in this case. $\eta(t)$ is roughly twice the error on the uniform grid. Still, one can observe the same tendency of lower errors for larger c parameter values (Figure 7.10). Again optimized results are very close to the best uniform value of c. The variation of parameter c_x and c_y values is small, since the grid is not strongly non-uniform. However, the difference in the results between one parametric (denoted by Optimized 1P) and two parametric (denoted by Optimized 2P) weight functions is visible.

Example 3: Non-uniform grid ($\delta = 35\%$)

Behavior $\eta(t)$ in Figure 7.11 indicates that the spectrum of uniform free parameter values with converged numerical solution is drastically narrowed and the best free parameter value is around 10. At this point the optimized solution is better than any other solution obtained by the uniform free parameter value.

Example 4: Extremely non-uniform grid ($\delta = 45\%$)

The range of valid free parameters at $\delta = 45\%$ is further narrowed to values between 9 and 10 (Figure 7.12) (Note that curves are now marked with the same signs for different c values). Therefore, in real cases, when dealing with strongly unstructured grids, it is quite difficult to determine the most appropriate free parameter value. Again, the optimized solution gave better results than any other solution obtained by unique free parameter value. The range of free parameters c_x and c_y in this case is between 5 and 20 and the variation of values in different subdomains is large.

For more scattered grid nodes the solution can not be obtained by any free parameter value. With the optimization proposed in this work the stability

of the solution is provided also for larger displacements for the same seed number.

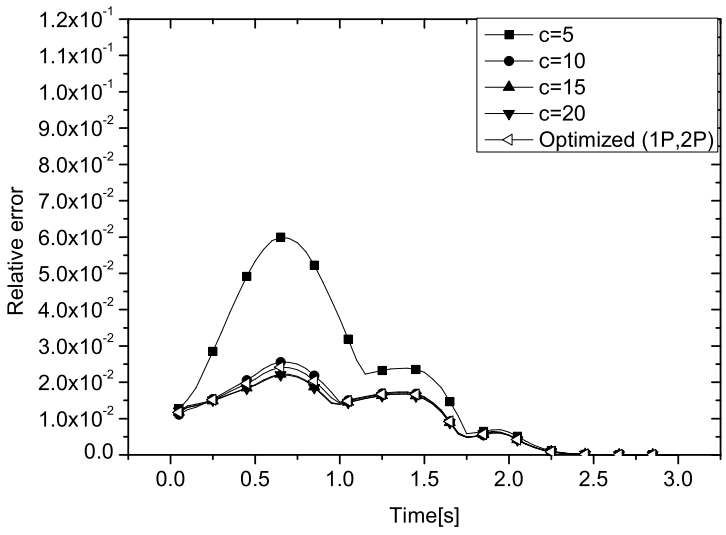


Figure 7.9: Uniform grid, i.e. 30×30 with $\delta = 0\%$ grid.

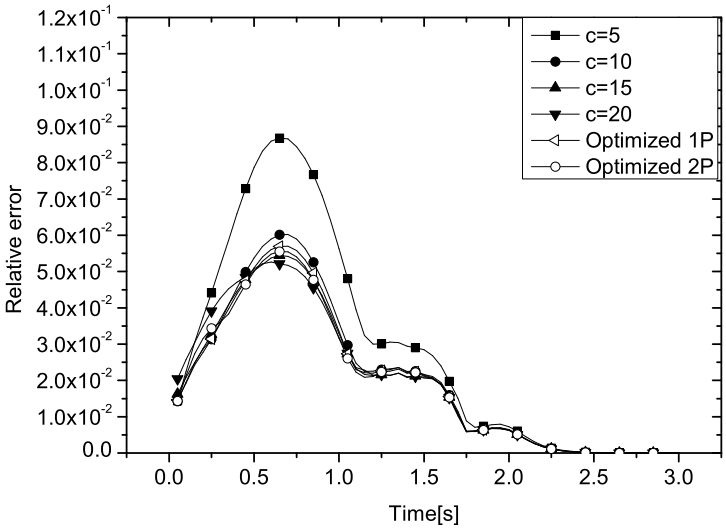


Figure 7.10: 30×30 with $\delta = 25\%$ grid.

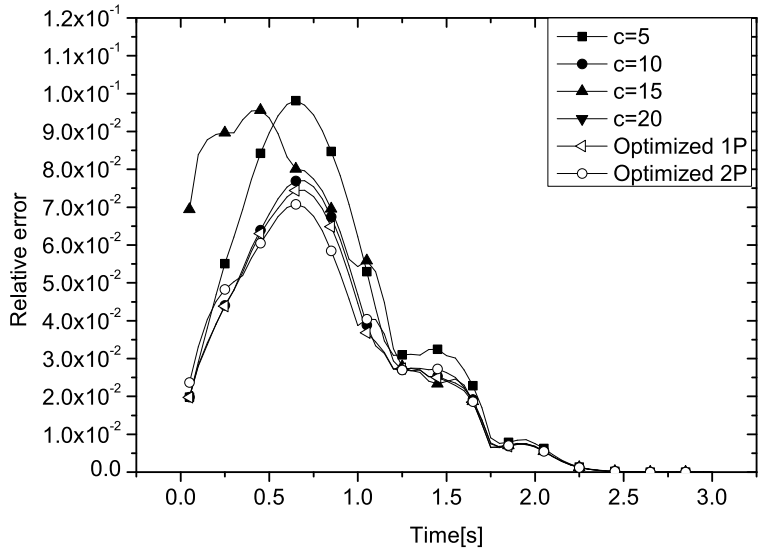


Figure 7.11: 30×30 with $\delta = 35\%$ grid.

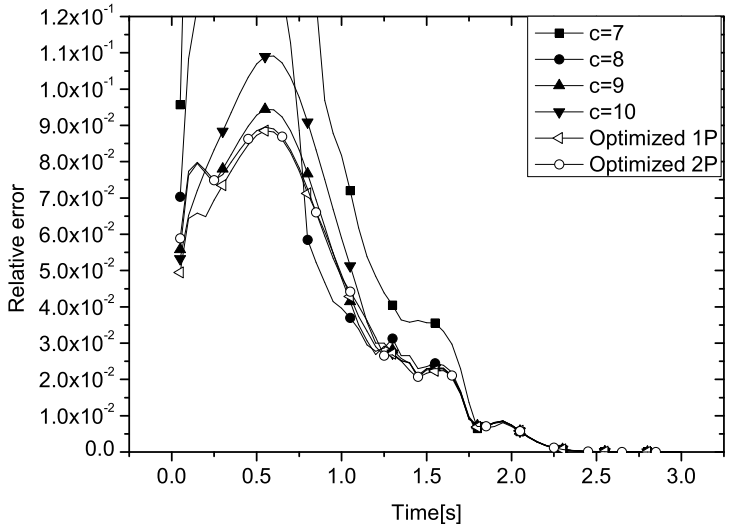


Figure 7.12: 30×30 with $\delta = 45\%$ grid.

Maximal relative errors for three different grid densities are presented in Tables 7.1-7.3. Tabulated results are given for larger spectra of free parameters and for different grid densities, 20×20, 30×30, and 40×40, respectively. Errors marked bold are the smallest. The tendency of free parameter c to

Table 7.1: Maximal relative errors (in %) on 20×20 grid

c	0%	25%	35%	45%
5	16.210	24.015	27.831	26.643
7	11.015	18.540	23.032	**23.885**
8	9.665	16.936	21.570	-
9	8.806	15.797	20.486	-
10	8.266	14.957	19.632	-
11	7.930	14.301	**19.454**	-
15	7.474	13.320	24.771	-
20	**7.419**	**12.553**	-	-
Opti 1P	7.631	14.444	21.023	22.583
Opti 2P	7.631	14.032	20.398	22.358

be smaller for a larger degree of non-uniformity is clearly seen. The best stability in this case is observed to be between 5 and 7 for the uniform free

parameter value, which is the range given in the literature. Similar behavior is observed for the finer 30×30 grid in Table 7.2. The best stability is assured

Table 7.2: Maximal relative errors (in %) on 30×30 grid

c	0%	25%	35%	45%
5	5.997	8.680	9.817	-
7	3.739	6.642	8.373	34.310
8	3.157	6.602	8.056	13.580
9	2.790	6.274	7.852	**9.442**
10	2.557	6.021	**7.530**	10.910
11	2.421	5.856	7.639	-
15	2.227	5.444	9.560	-
20	**2.202**	**5.215**	-	-
Opti 1P	2.299	5.708	7.462	8.862
Opti 2P	2.299	5.555	7.256	8.952

for c around 10. For the even finer 40×40 grid the range of valid parameters is narrowed down. Here results are presented only for up to $\delta = 0.35$, because for $\delta = 0.45$ the simulation diverged for all cases due to the larger probability of achieving the given theoretical limit δ_{sep}.

Table 7.3: Maximal relative errors on 40×40 grid

c	0%	25%	35%
5	2.459	3.895	3.916
7	1.237	2.779	**3.135**
8	1.189	2.474	**3.135**
9	1.161	2.275	3.194
10	1.145	2.153	3.366
11	1.136	**2.084**	-
15	**1.127**	2.094	-
20	1.144	2.172	-
Opti 1P	1.138	1.976	3.034
Opti 2P	1.138	1.891	3.418

Example 5: Structured non-uniform grid ($\chi = 1.3$)

This is a more realistic node arrangement used in practice. Similar structured but non-uniform grids are obtained for example with Delaney triangulation, remeshing of grids, etc. Errors are again smaller for larger values of c as observed in Example 1 and Example 2, but for larger values than 10 the simulation diverges (Figure 7.13), which is similar to critical values in Example 3 and Example 4. The variation of parameters can be presented graphically.

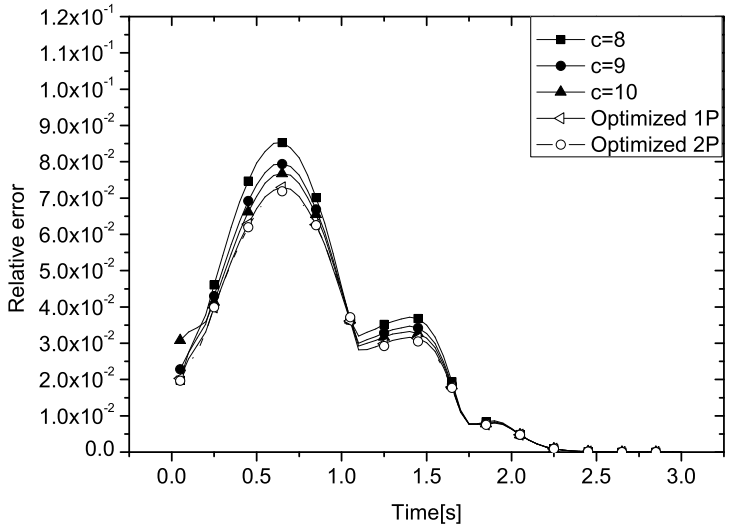


Figure 7.13: Regular non-uniform 30×30 grid with stretching factor $\chi = 1.3$.

Figure 7.14 represents the spatial distribution of free parameters c_x and c_y. The parameters are symmetrical with respect to the x and y axis and range from 7.5 to 18.5. Similar results are obtained by using one parametric optimization.

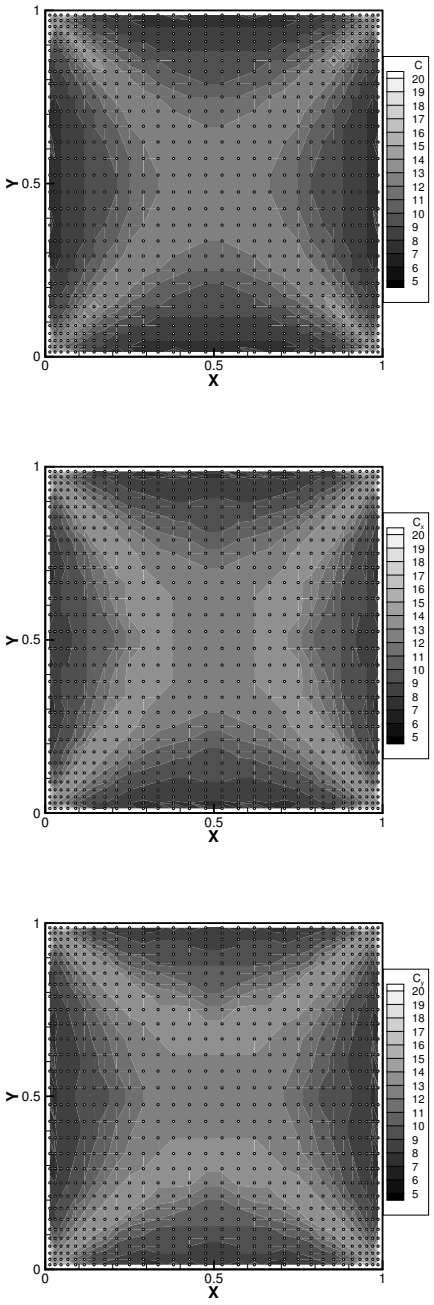


Figure 7.14: Free parameter values c for one parametric optimization (top), and two parametric c_x (center) and c_y (bottom).

Statistical comparison

For the sake of comparison, the results in Tables 7.1-7.3 are calculated on the node distribution with the same seed number, which determines the sequence of quasi-random numbers. As a consequence the node distribution is the same, only the displacement is bigger for larger δ values. For real comparison several runs have to be performed with random seed number and same δ parameter. Naturally maximal overall error will vary from one node distribution to another. In Table 7.4 results are presented for 100 runs and are given in terms of average error, standard error deviation and percentage of diverged runs. Maximal errors over 10% are excluded from the calculation of average error and standard deviation and are treated as diverged and listed separately as *diverged*. Parameter c for $\delta = 25\%$ is set to 15 which is the average value between $c = 20$ in Table 7.2 and $c = 11$ in Table 7.3). For $\delta = 35\%$ the free parameter c is set to 10.

Table 7.4: Average overall maximal error, standard deviation and number of diverged simulations.

			average error	standard deviation	% diverged
$\delta=$ 25%	grid 30×30	opti 1P	4.89	0.412	0
		opti 2P	4.75	0.363	0
		c=15	4.63	0.539	0
	grid 40×40	opti 1P	1.95	0.150	0
		opti 2P	1.90	0.141	0
		c=15	1.97	0.261	2
$\delta =$ 35%	grid 30×30	opti 1P	6.60	0.83	4
		opti 2P	6.57	0.77	3
		c=10	6.65	0.85	14
	grid 40×40	opti 1P	3.21	0.845	6
		opti 2P	3.13	0.932	6
		c=10	3.30	0.831	15

Table 7.4 clearly shows that stability of the optimized weight function is greatly improved. Although the average errors are sometimes just a little bit bigger in case of optimized parameters than for uniform parameter, optimized free parameters still give smaller standard deviation. While for $\delta = 25\%$ and grid density 30×30 the optimized and non-optimized solutions have 100% convergence of calculations, on the finer 40×40 grid the non-optimized

solution diverged in 2% of cases. For $c = 20$ (which showed the best accuracy for the same conditions in Table 7.2) on the 30×30 grid, the simulation diverged in 14% of cases.

With strongly non-uniform grids ($\delta = 35\%$) the differences in the number of converged simulations is much larger for both grid densities. In the case of the 30×30 grid there are almost 4 times more converged results in the case of optimized free parameters and in the case of the 40×40 grid more than 2 times.

The difference in the number of converged results between one parametric and two parametric optimization is negligible. However, the average accuracy and standard deviation is smaller in case of two parametric optimization.

7.1.3 Conclusions

From the results shown above one can conclude that the free parameter values have indeed a strong influence on the accuracy and the stability of a simulation. The idea developed in this work is original from two standpoints. First, the optimization procedure is set locally on each subdomain ${}_n\Omega$. Therefore each node is optimized by the same local reference quality function according to the specific node distribution in ${}_n\Omega$. Second, this idea is extended to two parametric optimization. Non radially symmetric weight functions have already been presented in [Prax and Sadat, 1996] for flows in channels, but the idea about different values c_x and c_y in each ${}_n\Omega$ is introduced here for the first time.

As described previously in this chapter, the optimization time depends mostly on the choice of optimization procedure. The results show that the accuracy and stability of one parametric optimization is usually not much worse than two parametric optimization. Therefore the decision between the two choices (and possibly even between multi parametric optimization) is up to the user. Optimization time, however, increases with the number of the optimization degree of freedom as shown in Table 7.5. The time for 100 calculation nodes and one parametric optimization ($t_{ref} = 1.81$ s)[1] is taken as reference and all others are calculated as

$$\eta_t = \frac{t}{t_{ref}}, \tag{7.16}$$

[1]Pentium III, 500 MHz.

Table 7.5: Relative optimization time for one parametric (1P) and two parametric optimization (2P).

Number of nodes	1P	2P
100	1	2.65
400	3.66	11.54
900	8.13	26.38
1600	15.48	49.33
2500	27.61	84.39
10000	173.62	404.47

7.2 Size of the domain of influence

Another property of local meshless methods is that the representation of the problem domain is made via construction of a shape function, which is formed from I neighboring nodes in the subdomain ${}_n\Omega$, which usually contains a much smaller number of nodes, than the whole discretized domain, i.e. $I \ll N$, and N is the total number of nodes. The neighboring nodes can be chosen based on two principles. In the first principle, the total number of neighboring nodes I is set. In the second principle, the shape and the size of the subdomain ${}_n\Omega$ is set. The shape is usually chosen to be a sphere (3D) or a circle (2D) with the radius (size) σ centered around the node $\boldsymbol{p}_n$. In former case ${}_n\Omega$ is denoted as domain of influence (DOI), while in latter case ${}_n\Omega$ is called support [Liu, 2003]. In DAM the number of interpolating nodes is the same for each interpolation patch, therefore the term domain of influence is used. The size of the domain of influence is a user choice. In general it can embrace the whole calculation domain including boundaries $\Omega+\Gamma$. Such methods are called *global representation methods*. Their common disadvantage is in fully populated matrices, which lead to large CPU time consumption. Another disadvantage is lower ability to interpolate strong non-linearities or even discontinuities in the calculation domain. On the other hand, the lowest number of nodes in the domain of influence is the number of the closest neighbors (i.e. 2 in 1D, 4 in 2D, 6 in 3D). Accuracy and stability is usually lower for the same grid density in the case of smaller domain of influence, but calculation speed drastically increases. Methods based on MLS have additional limitation, namely that the number of nodes in the domain of influence cannot be lower than the number of support functions. As an example in DAM the lowest number of nodes in 2D is 6 due to Taylor truncation of the polynomial at the second order.

As we can see from the above-mentioned behavior, we seek for a kind of optimum between accuracy, stability, and calculation speed. The investigation of the behavior of calculation results is again carried out on the Burger equation (7.10) on a 30×30 grid. The properties of unstructured grids are the same as defined in Section 7.1.

In order to find the optimum number of nodes in the DOI, the simulation results must be represented and compared in terms of

- accuracy
- stability
- CPU time

This is not an easy task since the accuracy and stability are determined through at least two interconnected properties, namely number of nodes in the DOI and free parameter c value. CPU time, however, is related only to the number of nodes in the DOI. For this reason two sets of calculations have been introduced. In the first set, uniform free parameter value $c = 15$ is used for all runs in combination with different number of nodes in DOI. The purpose of this first set is to investigate the stability and calculation time of the simulations.

The second set is characterized by the fixed number of nodes in the DOI with different values of free parameter c.

7.2.1 Numerical results

Results of testing for the first set of calculations are given in Table 7.6. In the first column the number of nodes inside the domain of influence are

Table 7.6: The effect of the number of support nodes in the domain of influence on accuracy and stability.

No. of nodes in DOI	relative time	Maximal relative errors in [%]			
		$\delta = 0$	$\delta = 25\%$	$\delta = 35\%$	$\delta = 45\%$
6	1	2.213	-	-	-
7	3.89	**2.191**	5.826	-	-
8	6.80	2.248	**5.312**	-	-
9	8.03	2.319	5.445	9.560	-
10	14.40	3.396	5.773	**7.533**	-
15	33.13	4.640	6.586	8.156	-
20	50.43	4.690	7.725	9.167	**10.138**

listed. We started from the minimum possible number (6) and took 20 as the largest number of nodes. The time for 6 nodes is taken as reference and all others are calculated as in equation (7.16), where t is the CPU time of each simulation and t_{ref} is the time needed for calculation with a minimal number of nodes. Errors are calculated in terms of maximal relative error between analytical and numerical solutions over the whole domain in each time step as in equation 7.15 and the analytical solution is defined in equation (7.11). As explained in Section 7.1 the level of non-uniformity is indicated by δ. The most accurate results are written in boldface, whereas diverged simulations are indicated by $-$. Table 7.6 clearly shows that the stability improves with an increasing number of nodes in the domain of influence. In Table 7.6 it can also be observed that the results obtained with a larger number of nodes are not the most accurate and are dependent on the number of nodes in the DOI. There are at least two reasons for this. The first one is that the optimal free parameter c is the same as shown for 9 nodes in the DOI. The second reason is that the calculated derivatives are "smeared" when a larger number of nodes in the DOI is used.

The second part of the test is dedicated to the investigation of the influence of the free parameters for 15 nodes in the domain of influence. Results are presented in Table 7.7. From Table 7.7 we can conclude that most opti-

Table 7.7: Maximal relative error with respect to free parameter c for 15 nodes and $\delta = 35\%$.

c	Maximal Relative error in [%]
10	10.100
15	8.156
20	**7.812**
25	8.079

mal parameter at 15 nodes in the domain of influence is 20, which is twice as much as for 9 nodes. The answer for this lies in the fact that the influence of more distant nodes should be diminished for good representation of large derivatives involved in the Burger equation because with a larger number of nodes in the domain the results are more smeared.

7.2.2 Conclusions

What is the final answer about the optimal number of nodes in the DOI? For moderately unstructured grids it is better to use a lower number of nodes from the accuracy point of view as well as from the time consumption point of view. Only in the case of a highly unstructured grid is the use of a higher

number of nodes justified. The most optimal number of neighboring nodes included in the domain of influence is - on our knowledge - 9. The reason for this is that time consumption is only 10% larger than in case of 6 nodes and accuracy and stability are assured. Intuitively this number is most usually taken because for uniform grids this represents the neighboring nodes on the left, right, up, down, and all diagonal nodes in 2D.

7.3 Notes on implementation of diffusive and advective term

With calculations of physically involved problems (e.g. double diffusive natural convection, radionuclide transport) some issues about the implementation of advective and diffusive terms into DAM (and possibly to other meshless methods) became more pronounced. Similar observations can be found in [Sophy, 2002]. Although the physical description of advective and diffusive terms is very clear, their numerical implementation is far more ambiguous.

7.3.1 Diffusive term

Consider first the diffusive term. In conservative form for a general scalar variable ϕ the diffusive term is defined as

$$\nabla \cdot (\boldsymbol{D}\, \nabla \phi), \tag{7.17}$$

where $\boldsymbol{D}$ is the diffusion tensor. It turns out that the evaluation of the diffusive term in the described form is not very effective, because it is constructed by two evaluations of the first derivatives. From practical testing we have shown that the numerical error from two evaluations of the first derivative is larger than from one evaluation of the second derivative. Therefore better results are provided by mathematical expansion of equation (7.17) to

$$\nabla \cdot (\boldsymbol{D}\, \nabla \phi) = (\nabla \cdot \boldsymbol{D}) \cdot \nabla \phi + \boldsymbol{D} : [(\nabla^2 \phi)\, \boldsymbol{I}], \tag{7.18}$$

where $\boldsymbol{I}$ is identity tensor

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7.19}$$

and double dot product is

$$\begin{aligned} \boldsymbol{D} : (\nabla^2 \phi\, \boldsymbol{I}) &= \begin{bmatrix} D_{xx} & D_{yx} & D_{zx} \\ D_{xy} & D_{yy} & D_{zy} \\ D_{xz} & D_{yz} & D_{zz} \end{bmatrix} : \begin{bmatrix} \frac{\partial^2 \phi}{\partial x^2} & 0 & 0 \\ 0 & \frac{\partial^2 \phi}{\partial y^2} & 0 \\ 0 & 0 & \frac{\partial^2 \phi}{\partial z^2} \end{bmatrix} = \\ &= D_{xx} \frac{\partial^2 \phi}{\partial x^2} + D_{yy} \frac{\partial^2 \phi}{\partial y^2} + D_{zz} \frac{\partial^2 \phi}{\partial z^2} \end{aligned} \tag{7.20}$$

In the case of constant diffusion D equation (7.18) simplifies to

$$D \nabla^2 \phi. \tag{7.21}$$

7.3.2 Advective term

As in the case of the diffusion term, the application of the advective term needs more attention as it was observed on more intensive problems with thin boundary layers, e.g. double diffusive natural convection. The partial derivatives of the advective term in conservative form

$$\nabla \cdot (\mathbf{V}\,\phi) \tag{7.22}$$

have to be evaluated in the transport equation with advection term. One can calculate the derivatives from equation (5.8) by coefficients α calculated from the system of equations (5.3) where in the right-hand side vector (5.5) $\hat{\phi}$ is replaced by $\phi\, V_\xi$

$$_n\hat{\phi}_j = \sum_{i=1}^{I} \varphi_j(\mathbf{p}_i - \mathbf{p}_n)\; _n\widehat{W}(\mathbf{p}_i - \mathbf{p}_n)\, \phi(\mathbf{p}_i)\, V_\xi(\mathbf{p}_i), \tag{7.23}$$

where $\xi \in \{x, y\}$. The described formulation has two weaknesses. Consider the case of incompressible fluids (which is the most usual simplification). When the steady-state simulation is performed sometimes the condition of incompressibility ($\nabla \cdot \boldsymbol{V} = 0$) is not obtained during the simulation. If a transport equation is coupled using non-converged velocity field, this would act as a source or a sink. In the case of meshless methods this happens quite often, since they are not locally conservative. The solution to the mentioned problem with incompressible fluids with $\nabla \cdot \boldsymbol{V} = 0$ is to recast equation (7.22) to

$$\nabla \cdot (\mathbf{V}\,\phi) = \mathbf{V} \cdot \nabla\, \phi, \tag{7.24}$$

and use only the derivatives of the scalar variable ϕ. This formulation works fine only when the fluid is assumed to be incompressible and when the problems are not very intense, because the stability of the simulation is lower than with the conservative formulation.

However, the same problem emerges near boundaries. If the calculated subdomain contains the boundary nodes, the result may be not correct, especially when the boundary layers are thin. The reason for this is shown in Figure 7.15. First consider the domain of influence denoted by *case 1*. Visually it can be seen that velocity partial derivatives $\frac{\partial V_x}{\partial x}$ are of the same order as $\frac{\partial V_y}{\partial y}$ which insures incompressibility. In the second case, denoted by *case 2* in Figure 7.15, the domain of influence contains domain and boundary nodes. Due to the influence of boundary nodes with $V_x = V_y = 0$ the approximated absolute value of partial derivatives $\left|\frac{\partial V_x}{\partial x}\right|$ is larger than $\left|\frac{\partial V_y}{\partial y}\right|$. This problem is more pronounced when a coarser grid is used. The mentioned problem can

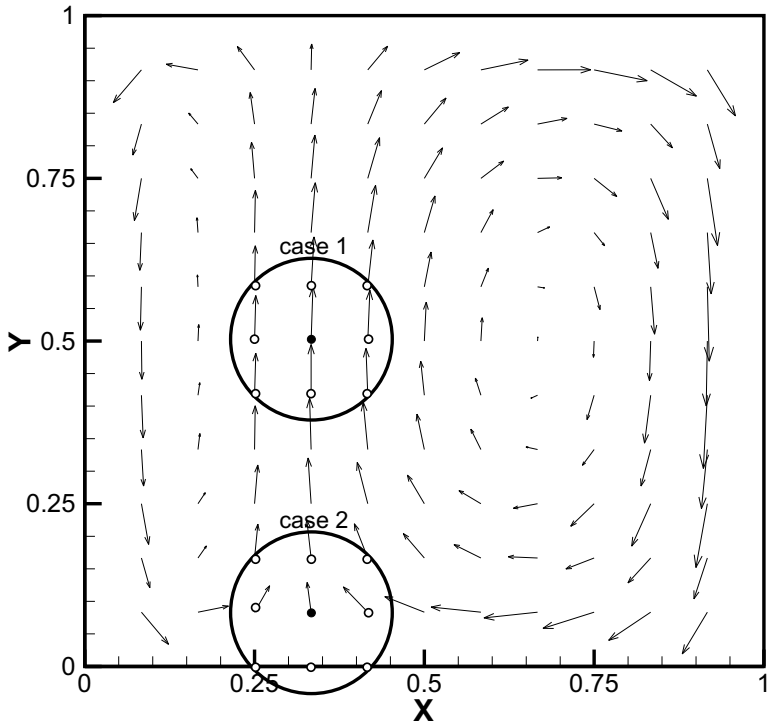


Figure 7.15: Description of incompressibility issue for advective term.

be solved by expanding equation (7.22) to

$$\nabla \cdot (\mathbf{V}\,\phi) = \phi\,(\nabla \cdot \mathbf{V}) + \mathbf{V} \cdot \nabla\phi \tag{7.25}$$

and using this relation instead of variable $\hat{\phi}$ in the right-hand side vector (5.5). In this way partial derivatives are calculated locally in each node only for variable ϕ. The solution of equation (7.25) is much more stable than the one described by equation (7.24) due to averaging (smearing) of derivatives. However, the previous statement holds only for approximation methods with absence of the Kronecker delta property.

8 Conclusions

This chapter is devoted to recapitulation of the work carried out and to recommendations for future research. In the beginning we should emphasize that much work has already been accomplished in the field of meshless numerical methods. However, there is a very limited number of contributions where meshless methods have been used for the simulation of realistic or even industrial problems. This lack of involvement in more realistic problems is related to the ongoing development of meshless methods. Consequently the developers of meshless methods try to develop the most effective way of solving (partial) differential equations and not to implement certain method to more complicated and hard-to-verify problems. In other words, we are faced with fast development of the tool, but very poor transfer of this tool to users without extensive knowledge of a certain meshless method. The reason lies in the relatively broad spectra of methods, but usually quite shallow understanding of the details of each meshless method. Nevertheless, the "devil" is in the details.

Our primary intention in this work is to pick out one promising meshless method and to make an analysis of as many as possible of its properties. DAM has been selected on the basis of previous work with several mesh-reduction and meshless methods. DAM in many cases proved to be effective for solving large-scale and more complicated problems. However, many questions remained unanswered. To fill this gap, several properties of DAM are systematically analyzed. The acquired conclusions can be (with some care in interpretation) extrapolated to other meshless methods as well. Our contributions are summarized below.

Free parameters: The most important property which is responsible for the accuracy and stability of several meshless methods is the evaluation and determination of free parameters. Free parameters are associated with either basis functions or the weight function. The latter is used in the Diffuse Approximate Method. In our opinion, the inability to determine free parameter values in a simple and straightforward manner is the largest reason for the

absence of meshless methods in the engineering world. A way of determining the free parameter has been worked out and tested in Chapter 7.1. Tests show an improvement of the accuracy and stability of the method against non-optimized free parameters. Perhaps the whole optimization algorithm is not the absolutely best chosen, but it provides a possible way of tackling this problem. In addition to radially symmetric weight function (one parametric optimization), radially nonsymmetric weight function is also implemented and optimized (two parametric optimization) for the first time in meshless methods.

Size of the domain of influence: The next previously unexplored field is the size of the domain of influence. This, to our knowledge, has never been analyzed before. The tests in Chapter 7.2 are performed on the Burger equation for domains of influence containing from 6 up to 20 nodes. It is shown that the accuracy and the stability are situated on opposite banks. While the accuracy is better with a lower number of nodes considered, the stability improves with a higher number of nodes. So the answer cannot be given just in terms of a single number. It depends on the intensity of the involved physical problem as well as on the uniformity of the grid. The more intense the problem is (i.e. the derivatives are high) and the more non-uniform the grid, the more we are forced to focus on the stability (larger number of nodes in the DOI). On the other hand if the problem is not intense and the grid is uniform, a lower number of nodes can be used in the domain of influence.

Numerical or artificial diffusion: In traditional methods, where the polygonization of the problem geometry is necessary, the numerical diffusion is strongly related to grid density, i.e. a finer grid provides lower numerical diffusion. It is shown here probably for the first time that the meshless methods do not exhibit the same behavior to such extent. We can almost state that the effect of numerical diffusion is negligible in meshless methods. This feature is very important in the simulation of transport phenomena. Especially in cases where molecular diffusion and dispersion in nature is low, this proves the great advantage of meshless methods against traditional methods.

Time-stepping procedures: On the problem of 1D radionuclide transport in Chapter 6.3 different time-stepping procedures are applied. Several implicit schemes, namely Euler explicit, explicit CBS, Adams-Bashforth and Runge-Kutta are tested. It is shown that the best stability is obtained by the fourth-order Runge-Kutta procedure. Due to evaluation of half-time-step

derivatives, however, CPU time is larger than for the Euler explicit method. The explicit CBS procedure was shown to be as stable as the Euler explicit procedure, but the order of accuracy is larger. The stability of Adams-Bashforth methods proved, against our expectations, to be lower than that of the Euler explicit, but the order of accuracy is higher.

Pressure-velocity coupling: Until now many algorithms such as SIMPLE, SIMPLER, PISO, etc. have been presented for the solution of velocity fields. However, it is hard to state which one is the most effective. Most of the algorithms are used solely for incompressible fluids. One of the newest algorithms developed for the solution of compressible and incompressible fluids is the CBS algorithm. Up to now it has been used in connection with FEM only. In this work the CBS algorithm is implemented and tested for the first time on any meshless method. On the case of natural convection in porous media in Chapter 6.5 it is shown that it provides good accuracy, although the calculation time for the Darcy equation is a little bit longer. Nevertheless, the algorithm is quite robust for application and its use is preferred over the use of the pressure Poison algorithm because the stability of the simulation is improved. The improvement of the stability in the context of meshless methods is not as large as the authors claim it to be in the context of FEM. However, since the algorithm is of second-order accuracy, the influence of time step truncation error is smaller than in the explicit pressure Poisson algorithm.

Application to physically more involved problems: While testing of numerical methods is performed on simple cases with constant material properties, regular grids, etc. the goal of the whole development of numerical methods is its application to real problems. Real problems, in general, involve complex geometries, nonhomogeneous and anisotropic physical properties. An example of a more realistic problem is shown in Chapter 6.4 dealing with the groundwater flow and radionuclide transport through a radwaste repository. The physical properties are still simplified due to the lack of measured data, but they are as realistic as possible. The repository geometry is also realistic. The comparison of radionuclide fluxes transported to groundwater show excellent agreement with the commercial package. Although the DAM calculations took considerably more CPU time than the calculations made by the commercial package, the results are very promising. This case shows the ability of DAM to cope with step-function initial conditions (initial concentration field) and good calculation accuracy which is necessary for such calculations where the difference between input values

and output values can be 20 orders of magnitude or even more.

Application to numerically more involved problems: Usually the problems used for testing of a certain method are not numerically very profound. Therefore it was a challenge to make a complex simulation with thin boundary layers and intense heat and mass transfer, which is a hard nut to crack even for highly developed traditional methods. Such a case of double diffusive natural convection in a composite fluid-porous layer is presented in Chapter 6.6. The purpose of this problem is to examine the effect of the permeability of the porous layer on flow structure and on heat and species transfer. The comparison with results obtained by FVM shows good agreement to DAM results despite the high complexity of the flow structure.

Other features: In addition to the above-mentioned work, important conclusions about the specific implementation of advective and diffusive terms into DAM are presented in Chapter 7.3. Another new feature is the implementation of Robin boundary conditions. Unfortunately, the physical scope of this work isn't broad enough to present the implementation of Robin boundary conditions. However, the results can be seen in [Šarler *et al.*, 2004c].

Future work: Although we tried to reasonably describe and analyze all important properties of DAM, several questions remained unanswered and left for future work. In this work a lot of effort is devoted to the development of an optimization algorithm for the optimization of the free parameter of the weight function. However, relatively little emphasis is put on the optimal search algorithm for the construction of subdomains. Perhaps the optimal distribution of nodes in a subdomain has as much impact on accuracy and stability as the optimal weight function. Therefore, more sophisticated analysis should probably be carried out in this field.

All calculations given in this work are made by explicit time-stepping schemes. In some cases it would also be valuable to include semi- or fully-implicit schemes, especially for steady-state calculations.

Despite the fact that DAM proved to be an accurate and stable method which can easily compare to traditional methods, it also has a few disadvantages. The most obvious disadvantage is the lack of the Kronecker delta property. In other words, the function value in a calculated node after approximation is not the same as the function value used. The effect of the Kronecker delta property mostly affects the boundary. In the absence of the Kronecker delta property the nodes with Dirichlet boundary conditions

should be left out of the calculation or must be treated separately by imposing given values after each iteration. In addition, the accuracy of Neumann and Robin boundary conditions can also be lower. This issue should be tackled in the future.

Appendix A

A.1 Conservation equations in the form of the general transport equation

All equations used in this work can be written in form of the general transport equation 3.7

$$\frac{\partial}{\partial t}[\varrho\,\mathcal{C}(\phi)] + \nabla\cdot[\varrho\,\boldsymbol{v}\,\mathcal{C}(\phi)] = -\nabla\cdot(-\boldsymbol{D}\,\nabla\phi) + S \tag{A.1}$$

Appropriate values for ϕ, $\mathcal{C}(\phi)$, S and $\boldsymbol{D}$ are give in Table A.1

Table A.1: Values of ϕ, $\mathcal{C}(\phi)$, S and $\boldsymbol{D}$ for the general transport equation.

conservation equation	reference	ϕ	$\mathcal{C}(\phi)$	S	$\boldsymbol{D}$
mass	3.9	1	ϵ	0	0
Darcy	3.11	0	0	$-\nabla P - \frac{\mu}{\mathcal{K}}\boldsymbol{V} + \boldsymbol{F}$	0
Darcy-Brinkman	3.13	$\boldsymbol{V}$	1	$-\nabla P - \frac{\mu}{\mathcal{K}}\boldsymbol{V} + \boldsymbol{F}$	μ_{eff}
Navier-Stokes+Darcy	3.15	$\boldsymbol{V}$	$\boldsymbol{V}/\epsilon$	$-\nabla P - \frac{\mu}{\mathcal{K}}\boldsymbol{V} + \boldsymbol{F}$	μ_{eff}
Darcy-Forchheimer	3.16	0	0	$-\nabla P - c_f\,\mathcal{K}^{1/2}\,\varrho\,\lvert\boldsymbol{V}\rvert\boldsymbol{V} + \boldsymbol{F}$	0
Energy	3.19	T	$c_p\,T$	S_T	$\boldsymbol{\lambda}$
Species	3.25	C	C	$-\bar{\lambda}_2\,R_2\,C_2 + \bar{\lambda}_1\,R_1\,C_1$	$\boldsymbol{D}$

A.2 Dimensionless numbers

The dimensionless numbers and variables used in this work are summarized here.

Aspect ratio	A	$\frac{H}{L}$
Buoyancy ratio	N	$\frac{\beta_C \Delta C}{\beta_T \Delta T}$
Darcy number	Da	$\frac{\mathcal{K}}{H^2}$
Dimensionless distance	$\tilde{p}$	$\frac{p}{\Delta p}$
Dimensionless pressure	$\tilde{P}$	$\frac{H^2}{\alpha_{th}\,\mu}$P
Dimensionless velocity	$\tilde{V}$	$\frac{H}{\alpha_{th}}$V
Dimensionless temperature	$\tilde{T}$	$\frac{T-T_{ref}}{\Delta T}$
Dimensionless viscosity	Λ	$\frac{\mu_{eff}}{\mu}$
Lewis number	Le	$\frac{\alpha_{th}}{D}$
Local Nusselt number	Nu(y)	H$\frac{\partial T(y)/\partial x}{A\,\Delta T}$
Local Sherwood number	Sh(y)	H$\frac{\partial C(y)/\partial x}{A\,\Delta C}$
Prandtl number	Pr	$\frac{\nu}{\alpha_{th}}$
Rayleigh number	Ra	$\frac{\varrho\ g\,\beta\,\Delta T\,H^3}{\alpha_{th}\,\mu}$
Porous media Rayleigh number	Ra*	$\frac{\varrho\,\mathcal{K}\,g\,\beta\,\Delta T\,H}{\alpha_{th}\,\mu}$
Schmidt number	Sc	$\frac{\nu}{D}$

Bibliography

Atluri, S. N. (2004). *The Meshless Method (MLPG) for domain & BIE Discretizations*. CREST. Tech Science Press, Los Angeles.

Atluri, S. N. and Shen, S. (2002). *The Meshless Local Petrov-Galerkin (MLPG) Method*. CREST. Tech Science Press, Los Angeles.

Atluri, S. N. and Zhu, T. (1998). A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. *Computational Mechanics*, 22:117–127.

Bear, J. (1972). *Dynamics of Fluids in Porous Media*. Elsevier, London.

Beck, J. L. (1972). Convection in a box of porous material saturated with fluid. *Physics of Fluids*, 15(8):1377–1383.

Beckerman, C. and Viskanta, R. (1988). Double-diffusive convection due to melting. *Journal of Heat and Mass Transfer*, 31:2077–2089.

Beckerman, C., Viskanta, R., and Ramadhyani, S. (1988). Natural convection in vertical enclosures containing simultaneously fluid and porous layers. *Journal of Fluid Dynamics*, 186:257–284.

Belytscko, T., Gu, L., and Lu, Y. Y. (1994a). Fracture and cracks growth by element free Galerkin methods. *Modelling and Simulations in Material Science and Engineering*, 2:519–534.

Belytscko, T., Krongauz, Y., Fleming, M., Organ, D., and Liu, W. K. (1996a). Smoothing and acceleration computations in the element free Galerkin method. *Computational Methods in Applied Mathematics*, 74:116–126.

Belytscko, T., Krongauz, Y., Organ, D., Fleming, M., and Krysl, P. (1996b). Meshless methods: an overview and recent developments. *Computational Methods in Applied Mechanical Engineering*, 139:3–47.

Belytscko, T., Lu, Y. Y., and Gu, L. (1994b). Element-free Galerkin methods. *International Journal of Numerical Methods in Engineering*, 37:229–256.

Brebbia, C. A., Telles, J. C. F., and Wrobel, L. C. (1984). *Boundary Element Techniques*. Theory and Applications in Engineering. Springer Verlag, Berlin.

Buhmann, M. D. (2003). *Radial basis functions: theory and implementations*. Cambridge University Press, Cambridge.

Celia, M. A., Bouloutas, E. T., and Zarba, R. L. (1990). A general mass-conservative numerical solution for the unsaturated flow equation. *Water Resources Research*, 26:14831496.

Chan, B. K. C., Ivey, C. M., and Barry, J. M. (1970). Natural convection in enclosed porous media with rectangular boundaries. *Wărme Stoffűbertragung*, 7:22–30.

Cooper, J. R., Randle, K., and Sokhi, R. S. (2002). *Radioactive Releases in the Environment - Impact and Assessment*. John Wiley & Sons, New York.

Couturier, S. (1999). *Contribution à l'étude des Transferts Conjugués en Présence de Changement de Phase par une Méthode de Collocation Utilisant l'Approximation Diffuse*. PhD thesis, Université de Poitiers.

Couturier, S. and Sadat, H. (1998a). A meshless method for solving incompressible fluid flow equations. *European Journal of Finite Element*, 7:825–841.

Couturier, S. and Sadat, H. (1998b). Résolution des équations de navier stokes dans la formulation en variables primitives par approximation diffuse. *Mécanique des fluides numérique*, 326:117–119.

Couturier, S. and Sadat, H. (1999). Melting driven by natural convection: A comparison exercise. *Interantional Journal of Thermal Sciences*, 38:5–26.

Darcy, H. (1856). *Les Fontaines Publiques de la Ville de Dijon*. Dalmont, Paris.

Dieses, A. E. (2000). *Numerical Methods for Optimization Problems in Water Flow and Reactive Solute Transport Processes of Xenobiotics in Soils*. PhD thesis, Ruprecht-Karls-Universitat, Hedilberg.

Ferziger, J. H. and Perić, M. (1997). *Computational Methods for Fluid Dynamics*. Springer Verlag, Berlin.

Fortran (2003). Compaq Visual Fortran 6.6.

Furbish, D. J. (1997). *Fluid Physics in Geology*. Oxford University Press, New York.

Gingold, R. A. and Monaghan, J. J. (1977). Smooth particle hydrodynamics: Theory and application to nonspherical stars. *Monthly Notices of the Royal Astronomical Society*, 181:375–389.

Gobin, D. and Bennacer, R. (1996a). Cooperating thermosolutal convection in enclosures: 1. scale analysis and mass transfer. *International Journal of Heat and Mass Transfer*, 39:2671–2681.

Gobin, D. and Bennacer, R. (1996b). Cooperating thermosolutal convection in enclosures, 2. Heat transfer and flow structure. *International Journal of Heat and Mass Transfer*, 39:2683–2697.

Gobin, D., Goyeau, B., and Neculae, A. (2004). Convective heat and solute transfer in partially porous cavities. *Journal of Heat Transfer*.

Gobin, D., Goyeau, B., and Songbe, J. P. (1998). Double diffusive natural convection in a composite fluid-porous layer. *Journal of Heat Transfer*, 120:234–242.

Gobin, D. and Le Quéré, P. (2000). Melting from an isothermal vertical wall: Synthesis of a numerical comparison exercise. *Computer Assisted Mechanics and Engineering Sciences*, 7:289–306.

Goyeau, B., Songbe, J. P., and Gobin, D. (1996). Numerical study of double diffusive natural convection in a porous cavity using the Darcy-Brinkman formulation. *International Journal of Heat and Mass Transfer*, 39:1363–1378.

Hickox, G. E. and Gartling, D. K. (1981). A numerical study of natural convection in a horizontal porous layer subjected to an end-to-end temperature difference. *Journal of Heat Transfer*, 103:797–802.

Jury, W. A. and Flűhler, H. (1992). Transport of chemicals through soil: Mechanisms, models and field applications. *Advances in Agronomy*, 47:141–201.

Kaviany, M. (1995). *Principles of Heat Transfer in Porous Media*. Springer Verlag, Berlin.

Lazzaro, D. and Montefusco, L. B. (2002). Radial basis functions for the multivariate interpolation of large scattered data sets. *Journal of Computational and Applied Mathematics*, 140:521–636.

Leonard, B. P. (1997). *Advances in Numerical Heat Transfer*, volume 1, chapter Bounded Higher-Order Upwind Multidimensional Finite-Volume Convection-Diffusion Algorithms, pages 1–54. Taylor and Francis, New York.

Liu, G. R. (1999). A point assembly method for stress analysis for solid. In et al., V. S., editor, *Impact Response of Materials & Structures*, pages 475–480. Oxford, Oxford University Press, New York.

Liu, G. R. (2003). *Mesh Free Methods: Moving Beyond the Finite Element Method*. CRC Press, London, New York.

Liu, G. R. and Gu, Y. T. (1999). A point interpolation method. In *Fourth Asia-Pacific Conference on Computational Mechanics*, pages 1009–1014. Singapore.

Liu, G. R. and Tu, Z. H. (2002). An adaptive procedure based on background cells for meshless methods. *Computational Methods in Applied Mechanical Engineering*, 191:1923–1942.

Liu, W. K., Adee, J., and Jun, S. (1993). Reproducing kernel and wavelet particle methods for elastic and plastic problems. In Benson, D. J., editor, *Advanced Computational Methods for Material Modeling*, 180/PVP 268 ASME, pages 175–190. ASME Publications, New York.

Mallants, D. (2004). Basic concepts of water flow, solute transport and heat flow in soil and sediments. BLG-Report 991, SCK/CEN, Mol, Belgium.

Mallants, D., Volckaert, G., and Labat, S. (2003). Parameter values used in the performance assessment of the disposal of low level radioactive waste at the nuclear zone mol-dessel. volume 2: Annexes to the data collection forms for engineerd barriers. Report R-3352, SCK/CEN, Mol, Belgium.

Massarotti, N., Arpino, F., and Nithiarasu, P. (2004). Fully explicit and semi-implicit CBS procedures for incompressible flows. In Neittaanmäki, P., Rossi, T., Korotov, S., Oñate, E., Périaux, J., and Knőrzer, D., editors, *Book of Abstracts: ECCOMAS 2004*, volume 2, page 240, Jyväskylä, Finland. University of Jyväskylä.

McDonald, M. G. and Harbaugh, A. W. (1988). A modular three-dimensional finite-difference groundwater flow model. U.S. Geological Survey Techniques of Water-Resources Investigations, Reston, Virginia.

Nayroles, B., Touzot, G., and Villon, P. (1991). L'approximation diffuse. *Mécanique des Milieux Continus*, 2:293–296.

Neculae, A. P. (2003). *Contribution à l'étude des phénomènes de transport dans une zone de croissance dendritique*. PhD thesis, Université Pierre et Marie Curie (Paris VI) et Universite de l'Ouest de Timişoara (Roumanie).

Ni, J. and Beckermann, C. (1991). Natural convection in vertical enclosure filled with anisotropic porous media. *Journal of Heat Transfer*, 113:1033–1037.

Nield, D. A. and Bejan, A. (1998). *Convection in Porous Media*. Springer-Verlag, Berlin.

Nithiarasu, P. (2003). An efficient artificial compressibility (AC) scheme based on split (CBS) method for incompressible flows. *International Journal for Numerical Methods in Engineering*, 56:1815–1845.

Onate, E. (1996). A finite point method in computational mechanics applications to convective transport and fluid flow. *International Journal of Numerical Methods in Engineering*, 39:3839–3866.

Partridge, P. W., Brebbia, C. A., and Wrobel, L. C. (1992). *The Dual Reciprocity Boundary Element Method*. Elsevier, London.

Patankar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow*. Hemisphere, New York.

Perko, J., Chen, C. S., and Šarler, B. (2001a). A polygon-free numerical solution of steady natural convection in solid-liquid systems. Computational Modelling of Free Moving Boundary Problems: Moving Boundaries VI, pages 111–122. WIT Press, Southampton.

Perko, J., Šarler, B., Chen, C. S., and Kuhn, G. (2001b). A meshless approach to natural convection. In Atluri, S., Nishioka, T., and Kikuchi, M., editors, *Advances in Computational Engineering & Sciences, ICCES'01*, pages 150–157, Mexico. Tech Science Press, Los Angeles.

Perko, J., Šarler, B., and Kuhn, G. (2003). Natural convection in porous media - a meshless solution of the brinkman extended darcy model. *Proceedings in Applied Mathematics and Mechanics*, 2:511–512.

Perko, J., Šarler, B., and Rek, Z. (2000). Convergence study of dual reciprocity BEM for navier-stokes equations. *Zeitschrift fűr Angewandte Mathematik und Mechanik*, 80:S839–S840.

Petkovšek, B., Perko, J., Brenčič, M., and Kalin, J. (2002). PA/SA calculations on influence of Slovenian LILW repository on environment. Final report, Zavod za gradbeništvo, Ljubljana, Slovenija.

Pollandt, R. (1997). Solving nonlinear differential equations of mechanics with the boundary element and radial basis functions. *International Journal for Numerical Methods in Engineering*, 40:61–73.

Porflow (2001). Porflow 128k, version 4.00. ACRi software.

Prasad, V. and Kulacki, F. A. (1984). Convective heat transfer in a rectangular porous cavity - effects of aspect ratio of flow structure and heat transfer. *Journal of Heat Transfer*, 106:158–165.

Prax, C. and Sadat, H. (1996). Co-located diffuse approximation for channel flows. *Mechanics Research Communications*, 23:61–66.

Prax, C., Salagnac, P., and Sadat, H. (1998). Diffuse approximation and control-volume-based finite-element methods: A comparative study. *Numerical Heat Transfer, Part B*, 34:303–321.

Prescott, P. and Incropera, F. P. (1996). Convection heat and mass transfer in alloy solidification. *Advances in Heat Transfer*, 28:231–338.

Pruess, K. (1987). TOUGH user's guide. Report LBL-20700, Lawrence Berkeley Laboratory, Berkeley, CA, and Nuclear Regulatory Commission Report NUREG/CR-4645, Washington, D.C.

Raghavan, R. and Ozkan, E. (1994). *A Method for Computing Unsteady Flows in Porous Media*. Pitman research notes in mathematical series. Longman Scientific and Technical, Harlow.

Rubinstein, J. (1986). Effective equations for flow in random porous media with a large number of scales. *Journal of Fluid Mechanics*, 170:379–383.

Runchal, A. K. (1982). PORFLOW: A mathematical model for coupled ground water flow, heat transfer and radionuclide transport in porous media. ACRi/TN-006.

Sadat, H. and Couturier, S. (2000). Performance and accuracy of a meshless method for laminar natural convection. *Numerical Heat Transfer, Part B*, 37:455–467.

Sadat, H., Prax, C., and Salagnac, P. (1996). Diffuse approximate method for solving natural convection in porous media. *Transport in Porous Media*, 22:215–223.

Sahimi, M. (1995). *Flow Transport in Porous Media and Fractured Rock*. VCH, New York.

Shiralkar, G. S., Haajizadeh, M., and Tien, C. L. (1983). Numerical study of high rayleigh number convection in a porous enclosure. *Numerical Heat Transfer*, 6:223–234.

Simunek, M., Sejna, M., and van Genuchten, M. T. (1999). The HYDRUS software package for simulating two-dimensional movement of water, heat, and multiple solutes in variably saturated media. IGWMC - TPS - 53, International Ground Water Modeling Center, Colorado School of Mines, Golden, Colorado, page 251.

Smith, R. M. and Hutton, A. G. (1982). The numerical treatment of advection: A performance comparison of current methods. *Numerical Heat Transfer*, 5:439–461.

Sophy, T. (2002). *Contribution à l'analyse de la Convection Thermomagnétique par une Méthode de Collocation basée sur l'approximation Diffuse*. PhD thesis, Université de Poitiers.

Sophy, T. and Sadat, H. (2002). A three dimensional formulation of a meshless method for solving fluid flow and heat transfer problems. *Numerical Heat Transfer, Part B: Fundamentals*, 41(5):193–198.

Sophy, T. and Sadat, H. (subm). On the numerical solution of magnetohydrodynamic equations. *International Journal of Numerical Fluid Flow*.

Šarler, B., Gobin, D., Goyeau, B., Perko, J., and Power, H. (2000). Natural convection in porous media - dual reciprocity boundary element method solution of the darcy model. *International Journal of Numerical Methods in Fluids*, 33:279–312.

Šarler, B., Gobin, D., Goyeau, B., Perko, J., and Power, H. (2004a). Dual reciprocity boundary element method solution of natural convection in darcy-brinkman porous media. *Engineering Analysis with Boundary Elements*, 28:23–41.

Šarler, B. and Perko, J. (1998). A comparison of different radial basis functions in DRBEM solution of the navier-stokes equations. In Kassab, A., Chopra, M., and Brebbia, C. A., editors, *Boundary Elements XX*, volume 4 of *International Series on Advances in Boundary Elements*, pages 461–470, Orlando. Computational Mechanics Publications; Southampton.

Šarler, B. and Perko, J. (2000). DRBEM solution of temperature and velocities in dc cast aluminium slabs. In Brebbbia, C., Kassab, A., Chopra, M., and Divo, E., editors, *Fourteenth International Conference on Boundary Element Technology, BETECH XIV*, volume 10 of *Advances in Boundary Elements*, pages 357–369. WIT Press, Southampton.

Šarler, B. and Perko, J. (2001). DRBEM simulation of radionuclide transport near nuclear waste repository. volume 10 of *International Series on Advances in Boundary Elements*, pages 309–318. WIT Press, Southampton.

Šarler, B., Perko, J., and Chen, C. S. (2002). Natural convection in porous media - radial basis function collocation method solution of the Darcy model. In Mang, H., Rammerstorfer, F., and Eberhardsteiner, J., editors, *Proceedings of the Fifth World Congress on Computational Mechanics (WCCM V)*, pages 1–16. University of Technology, Institute for Strenght of Materials, Institute for Lightweight Structures and Aerospace Engineering.

Šarler, B., Perko, J., and Chen, C. S. (2004b). Radial basis collocation method solution of natural convection in porous media. *International Journal of Numerical Methods for Heat and Fluid Flow*, 14:187–212.

Šarler, B., Vertnik, R., and Perko, J. (2004c). Solution of temperature field in DC cast aluminum alloy slab by the diffuse approximate method. In Tadeu, A. and Atluri, S. N., editors, *Advances in Computational and Experimental Engineering and Sciences : Proceedings of the 2004 International Conference on Computational & Experimental Engineering & Sciences*, pages 1364–1371. Tech Science Press, Los Angels.

The Free Dictionary (Accessed: 18. Nov. 2004). http://encyclopedia.thefreedictionary.com. Internet source.

van Genuchten, M. T. (1980). A closed-form equation for predicting the hydraulic conductivity of unsaturated soils. *Soil Science Society of America Journal*, 44:892–898.

van Genuchten, M. T. and Alves, W. J. (1982). Analytical solutions of the one-dimensional convective-dispersive solute transport equation. USDA

ARS Technical Bulletin Number 1661, U.S. Salinity Laboratory, 4500 Glenwood Drive, Riverside, CA 92501.

Webster, I. T., Norquay, S. J., Ross, F. C., and Wooding, R. A. (1996). Solute exchange by convection within estuarine sediments. *Estuarine, Coastal and Shelf Science*, 42:171–183.

Wrobel, L. C. and Aliabadi, M. (2002). *The Boundary Element Method*. John Wiley & Sons, New York.

Zienkiewicz, O. C. and Codina, R. (1995a). A general algorithm for compressible and incompressible flow, part 1. the split characteristic based scheme. *International Journal of Numerical Methods in Fluids*, 20:869–85.

Zienkiewicz, O. C. and Codina, R. (1995b). Search for a general fluid mechanic algorithm. In Caughley, D. and Hafez, M., editors, *Frontiers of Computational Fluid Dynamics*, pages 101–13. John Wiley & Sons, New York.

Zienkiewicz, O. C., Loehner, R., Morgan, K., and Nikazawa, S. (1984). *Finite Elements in Fluids*, volume 5. John Wiley & Sons, New York.

Zienkiewicz, O. C. and Taylor, R. L. (2000a). *The Finite Element Method*, volume 1: The basis. Butterworth-Heinemann, Oxford.

Zienkiewicz, O. C. and Taylor, R. L. (2000b). *The Finite Element Method*, volume 2: Solid mechanics. Butterworth-Heinemann, Oxford.

Zienkiewicz, O. C. and Taylor, R. L. (2000c). *The Finite Element Method*, volume 3: Fluid dynamics. Butterworth-Heinemann, Oxford.

Železnik, N., Petkovšek, B., Kalin, J., Perko, J., and Brenčič, M. (2003). Executive summary of PA study. In *Performance Assessment for LILW Repository in Slovenia*, Ljubljana, Slovenija. ARAO.