Development of an edge-based finite volume solver for porous media flow applications.

Rees, Ian

Rees, Ian (2004) *Development of an edge-based finite volume solver for porous media flow applications..* thesis, Swansea University.
http://cronfa.swan.ac.uk/Record/cronfa43012

CIVIL & COMPUTATIONAL ENGINEERING CENTRE
UNIVERSITY OF WALES SWANSEA

DEVELOPMENT OF AN EDGE-BASED FINITE VOLUME SOLVER FOR POROUS MEDIA FLOW APPLICATIONS

IAN REES
BSc.(Hons.), MBA

THESIS SUBMITTED TO THE UNIVERSITY OF WALES IN CANDIDATURE
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

APRIL 2004

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed ...(candidate)

Date ......27/05/04......................

STATEMENT 1

This dissertation is being submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

Signed (candidate)

Date ......27/05/04......................

STATEMENT 2

This dissertation is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed ....(candidate)

Date ......27/05/04......................

STATEMENT 3

I hereby give consent for my dissertation, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ....(candidate)

Date ......27/05/04......................

This work was funded jointly by the *Norwegian Geotechnical Institute* under the Force Contract and a Doctoral Training Grant from the *Engineering and Physical Sciences Research Council.*

SUMMARY

The work presented in this thesis constitutes the first phase in the development of a solver to handle the highly non-linear systems that describe the flow of fluids through porous media. Aspects researched are the mathematical description, discretization as well as the computer code implementations. With regards to the spatial discretization, a computational efficient edge-based vertex-centred finite volume scheme is proposed, with the application of a 'compact stencil' to calculate second derivative terms. With regards to the software implementation, a novel label based approach was employed using FORTRAN that greatly improved the flexibility of the code. The modelling capabilities of the proposed numerical scheme were validated successfully through the simulation of problems for which benchmark numerical solutions or analytic solutions exist.

ACKNOWLEDGEMENTS

I am eternally grateful for the guidance and support of my supervisors, Professor R.W. Lewis and Dr. I. Masters. This work would not have been possible had it not been for their vision, advice and patience. I would also like to extend my thanks to Dr. A.G. Malan for his assistance and willingness to share his knowledge on issues related to my research. His contribution is greatly valued and appreciated.

I have been very fortunate during my research to meet an excellent group of people, many of whom I now regard as friends. There are too many to mention here; but they know who they are. I thank them all for making the last three years so much more enjoyable.

Finally, and most importantly, my sincere gratitude goes out to my parents for their unwavering support and encouragement throughout my time in Swansea.

CONTENTS

Chapter 1: Introduction

Chapter 2: Spatial Discretization Procedures

Chapter 3: Advanced Programming Methods And Data Structures For Computational Modelling

Chapter 4: Phase Transformation

Chapter 5: Saturated-Unsaturated Groundwater Flow

Chapter 6: Modelling Aquifer Contaminant Transport

Chapter 7: Coupled Contaminant Transport Through a Saturated Porous Medium

Chapter 8: Conclusions

NOMENCLATURE

Γ	Surface [L^2].
Γ_{mn}	Control volume bounding surface associated with the edge connecting nodes *m* and *n* [L^2].
Γ^b_{mn}	Control volume bounding surface of the edge connecting nodes *m* and *n* which coincides with the computational domain boundary [L^2].
$\mathbf{B}_{mn}$	Boundary edge coefficient of edge connecting nodes *m* and *n* [L^2].
$\mathbf{C}_{mn}$	Internal edge coefficient of edge connecting nodes *m* and *n* [L^2].
$\mathbf{n}$	Unit normal vector.
t	Time [T].
Ω	Volume [L^3].
Ω_m	Discrete control volume associated with node *m* [L^3].
Υ_{mn}	Edge connecting nodes *m* and *n*.
x_i	Cartesian spatial coordinate component *j* [L].
x	Cartesian spatial coordinate in the *x* direction [L].
y	Cartesian spatial coordinate in the *y* direction [L].
S_w	Saturation of water (dimensionless).
S_g	Saturation of gas (dimensionless).
S_s	Specific storage of the soil [L^{-1}].
h	Pressure head of water [L].
α	Thickness of the flow domain [L].
k_{ij}	Intrinsic conductivity tensor [LT^{-1}].
k_r	Relative permeability to water (dimensionless).
C	Specific moisture capacity [L^{-1}].
θ	Moisture content of the soil (dimensionless).
q	Source/sink term [L^3T^{-1}].
p_a	Air pressure [$ML^{-1}T^{-2}$].
p_w	Water pressure [$ML^{-1}T^{-2}$].
γ_w	Specific weight of water.
p_c	Capillary pressure between air and water [$ML^{-1}T^{-2}$].

ϕ	Porosity (dimensionless).
c	Concentration of pollutant [$\mathrm{ML^{-3}}$].
$\mathbf{D}$	Dispersion coefficient tensor [$\mathrm{L^{2}T^{-1}}$].
$\mathbf{v}$	Velocity [$\mathrm{LT^{-1}}$].
g	Function that incorporates chemical reactions [$\mathrm{T^{-1}}$].
λ	Decay rate coefficient [$\mathrm{M^{-1}T^{-1}L^{3}}$].

Mathematical operators

$\cdot$	Vector inner product.
$\lvert\bullet\rvert$	Norm of $\bullet$ (absolute value in case of a scalar).
$\partial\,\bullet$	Partial derivative of $\bullet$.
$\partial_j^{fv}\,\bullet$	Finite volume approximation of the spatial partial derivative of $\bullet$ with respect to x_j.
$\nabla\,\bullet$	Gradient operator of $\bullet$.
$\Delta\,\bullet$	Increment in $\bullet$.

Dimensionless Numbers

Pe	Peclet number: $v_x \Delta x / D$
Cr	Courant number: $v_x \Delta t / \Delta x$

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

1.1 BACKGROUND

The origins of this project lie in the work done in Swansea over the last two decades by research students under the supervision of Prof. R.W. Lewis on the numerical modelling of fluid flow through porous media. More recently, research activities within the group have been targeted towards the simulation of the behaviour of producing petroleum reservoirs. The development of a capacity to predict the exploitation of structurally complicated and fractured oil reservoirs is essential for the rational use of investment capital. A poor understanding of how the reservoir behaves during production may lead to inept, costly and inefficient development schemes. The area of reservoir simulation applies the concepts and techniques of mathematical modelling to the analysis of the behaviour of petroleum reservoir systems. The basic flow model consists of the partial differential equations which govern the unsteady-state flow of all fluid phases in the reservoir medium.

Traditionally, these reservoir simulators consisted only of a flow model to simulate the three-phase flow (water, oil, gas) through the porous reservoir system. These models contained no geomechanical effects, and were predominantly solved using the finite difference method as described by Aziz and Settari [1] and Crichlow [7]. Although acceptable results are obtainable using the finite difference scheme, it is severely restricted when the geometric flexibility of the method is considered. Geometrically complex geological features, such as faults, cross stratified beds and large scale inclined strata, can have very significant effects on the flow of reservoir fluids. In two dimensions, only regular quadrilateral elements, and regular brick elements in three dimensions are available to the finite difference practitioner. Standard finite difference methodology lacks the geometric flexibility required for modeling flow through or around such structures. Triangle based discretization methods, by contrast, offer an attractive means to discretize and simulate flow through geometrically complex features accurately. Multi phase fluid systems have been successfully simulated using the triangle based finite element method by numerous researchers e.g. ([18]-[26]).

As deeper hydrocarbon bearing formations were being detected and explored, Gutierrez and Lewis [14] recognised that the role of geomechanics in petroleum reservoir engineering was becoming increasingly important. The increasing power, and extensive use of computers, in addition to the concurrent development of numerical techniques made possible much more precise analyses, with the eventual aim of developing the capability for making predictive field scale simulations. The stress, fluid pressure and temperature conditions encountered at large depth give rise to strange situations where conventional reservoir modelling fails to provide an accurate analysis [6]. Since comprehensive coupling between the stresses and pore fluid pressures was first rationalised by Biot [4], many advances have been made by various researchers, notably Lewis and Schreffler [21], on the development of a fully coupled fluid flow and geomechanics model. This model has been successfully extended and utilised by Sukirman [37] and Lewis and Sukirman ([24],[22],[23]) to model the subsidence bowl above a compacting saturated oil reservoir.

A major departure from the traditional reservoir modelling approach was made by Barenblatt *et al.* [3], who utilised an overlapping continuum technique (double porosity model) to simulate fluid flow in fractured porous media. This work was followed closely by Warren and Root [38], who presented an analytical solution for single-phase, unsteady-state flow in naturally fractured reservoirs. Ghafouri [13], Ghafouri and Lewis [12], and Lewis and Ghafouri [19] included the double-porosity model into the coupled system of Sukirman [37], and it was concluded that the reservoir behaviour was highly affected by both fissure permeability and the fissure intervals. The double porosity model provided more realistic results than those with a conventional single average porosity.

The effects of temperature on the reservoir system were introduced to the coupled reservoir simulator (CORES), and a temperature-dependent double porosity model in a deforming fissured reservoir was presented by Pao [31], Masters *et al.* [29] and Pao *et al.* [32]. By neglecting the temperature effects, Pao and Lewis ([30],[20]) presented a mathematical formulation of a three-phase, three-dimensional fluid flow and rock deformation in fractured reservoirs, and a field scale numerical example was employed to show the applicability of the scheme. In addition, Lewis *et al.* [25] worked on the inclusion of an elasto-plastic model into CORES to attempt to simulate the effect of deformation of a structurally weak reservoir rock such as chalk.

During the 1990's, several researchers, including Fung *et al.* [11] and Sonier and Eymard [34], began working with the Control Volume Finite Element (CVFE) method for the solution of multi-phase flow problems in porous media. The CVFE method was proposed in the area of computational fluid dynamics for solving the Navier-Stokes Equations ([2],[33]) where flexible gridding as well as local mass, momentum, and energy conservation are achieved. The control-volume approach enforces local mass conservation and permits a direct physical interpretation of the resulting discrete equations without losing the flexible grid geometry. As reported by Fung *et al.* [11], these are significant advantages over the classical Galerkin or variational finite-element methods. At this time, problems were being encountered with CORES due to the program architecture preventing the easy modification of the governing equations. It was decided that a substantial re-write was necessary, and knowing the deficiencies of the Galerkin finite element scheme in calculating fluid flow, an alternate scheme was sought. Several researchers modelling fluid flow in the computational fluid dynamics community were praising the novel edge-based vertex-centred finite volume scheme (as per Malan *et al.* [28], Sorensen [35]). The scheme was conservative, and the edge-based representation of the domain facilitated computational and storage efficiency. Work commenced in the Spring of 2001 to write from new an edge-based finite volume source code, with good program architecture to facilitate easy modification, to solve porous media flow problems. This scheme had not been used previously in simulating the highly non-linear flow of fluid through porous media. The modeling of the fluid flow through porous media is complex on a number of levels. The development of such capabilities is subdivided into three main tasks viz. formulation of the governing partial differential equations to describe the physics of the problem, the discretization and solution of these equations and, last but not least, the translation of the resulting numerical scheme into efficient computer code.

1.2 SCOPE OF WORK AND RESEARCH CONTRIBUTIONS

The work documented in this thesis constituted the first phase in the development of a simulator for modelling fluid flow through porous media. The main research activities were:

- Suitable discretization and solution procedures were selected and refined. Discretization should support structured and unstructured computational meshes to ensure applicability to complex geometries.
- On a computational level, memory efficiency and a clean program architecture were paramount.
- The transcription of the algorithm to computer code was achieved by building on the latest in programming science. Here, techniques were developed for furnishing computationally efficient code.

A summary of the research contributions made during the above activities follows:

- A purely edge-based spatial discretization technique was employed for the discretization of the governing equations describing the physical system. This is, to the author's knowledge, the first instance in which an edge-based finite volume scheme has been applied to strongly non-linear diffusion dominated equations describing the multi-phase flow of fluid through a porous continua. This technique offers significant computational cost savings as compared to element based methods (Luo *et al.* [27]) while being naturally applicable to unstructured and hybrid grids.

- The edge-based technique was enhanced by an alternative calculation procedure, utilizing a more compact stencil (Crumpton *et al.* [8]) for the case of the non-linear diffusion type partial differentials.
- With regards to the software implementation, a novel label based approach was employed. In the interest of computational performance, a data structure was proposed such that the modification of code at all levels was possible with the minimum of labour.

A number of publications were, as a result, forthcoming from the research, and are listed next.

1.3 PUBLICATION LIST

The following publications were produced during the course of the research project:

- W.K.S. Pao, R.W. Lewis, X. Yang, I. Masters, I. Rees, The Effects of Cross Coupling Terms in Double Porosity Model and its Significant in Fluid Transport, In proceedings: IMA Conference on Modelling Flow in Oil Reservoirs, BP Institute, University of Cambridge (2002).
- I. Rees, R.W. Lewis, I. Masters and A.G. Malan, Material Variation Using A Compact Stencil Vertex-Centred Edge-Based Finite Volume Scheme, In proceedings: The 11th Annual Conference of the Association for Computational Mechanics in Engineering (ed. M.A. Wheel), pp. 5-9, Glasgow, 2003.
- I. Rees, I. Masters, A.G. Malan and R.W. Lewis, An Edge-Based Finite Volume Scheme For Saturated-Unsaturated Groundwater Flow, *Computer Methods in Applied Mechanics and Engineering,* to appear (2004).
- I. Masters, I. Rees and R.W. Lewis, Advanced programming methods and data structures for computational modelling using edge based finite volume methods, Submitted for review to *International Journal for Numerical and Analytical Methods in Geomechanics.*

- I. Masters, I. Rees, R.W. Lewis and A.G. Malan, Modelling Aquifer Contaminant Transport Via An Edge Based Algorithm, In proceedings: The 12th Annual Conference of the Association for Computational Mechanics in Engineering (ed. H.R. Thomas), pp. 7-10, Cardiff, 2004.

1.4 THESIS OUTLINE

The thesis is split into eight chapters, including an introduction and conclusion. The following is a synopsis of each chapter.

- *Chapter One: Introduction*. In addition to giving an outline of the thesis, this chapter contains an overview of the work background, scope and research contributions made.
- *Chapter Two: Spatial Discretization Procedures*. The chosen spatial discretization algorithm is described in detail. Enhancements made to the scheme by the application of a more compact stencil in the interest of improved accuracy are discussed and presented. Finally, derivative calculations on homogeneous and non-homogeneous domains are presented and compared against analytic solutions for accuracy.
- *Chapter Three: Advanced Programming Methods And Data Structures For Computational Modelling*. This chapter focuses on the chosen programming methodology. Novel program architecture is proposed which goes beyond conventional numerical programming practices.
- *Chapter Four: Phase Transformation*. The highly non-linear one-dimensional problem of solidification and melting of metal ingots is presented. This problem is chosen as its mathematical description is very similar to the groundwater flow problems that need to be modelled, and analytic solutions exist for the test case. This work is the initial building blocks of the groundwater flow solver, and proves that highly non-linear diffusion dominated problems are solved successfully using the chosen scheme.

- *Chapter Five: Saturated-Unsaturated Groundwater Flow*. Tests were conducted to evaluate the ability of the proposed algorithm as well as the efficacy of the technology for highly non-linear flow problems on two-dimensional domains. The infiltration of water into a vertical soil column, and across a homogeneous and zoned earth dam section is considered. The spatial and temporal accuracy was assessed by comparing with published numerical solutions where available, and the enhancements made to the scheme by application of the compact stencil is demonstrated.
- *Chapter Six: Modelling Aquifer Contaminant Transport*. This chapter discusses the numerical tests conducted to evaluate the accuracy of the numerical formulation as well as the model's ability to describe the flow of a contaminant through a domain saturated with water using only the contaminant flow equation. Varying rates of advection/diffusion on the equation system is considered, and the effect of contaminant decay on the system is investigated. The spatial and temporal accuracy of the solution was assessed by comparing with published and analytic solutions where available. The enhancements made to the scheme in the interest of improved accuracy by application of the compact stencil are demonstrated.
- *Chapter Seven: Fully Coupled Contaminant Transport Through a Saturated Porous Medium*. In this chapter, tests were conducted to evaluate the ability of the proposed algorithm to model the fully coupled system of flow equations describing the flow of a contaminant through a domain saturated with water. The flow equation, and the transport equation are fully coupled by the Darcy velocity term. The test case considered examines the injection and subsequent extraction of a contaminant from the same well, and the effect of the dispersion and decay terms on the system are investigated.
- *Chapter Eight: Conclusion*. The original research contributions are summarized and recommendations made for the continuation of the work through future research.

REFERENCES

[1] Aziz K. and Settari A., Petroleum Reservoir Simulation, Applied Science, London, 1979.

[2] Baliga B.R. and Patankar S.V., A New Finite-Element Formulation for Convection-Diffusion Problems, *Num. Heat Trans.*, **3**: 393-409, 1980.

[3] Barenblatt G.I., Zheltov I.P. and Kochina I.N., Basic concepts in the theory of seepage of homogeneous liquids in fissured rocks, *J. Appl. Math. Mech. USSR*, **24**: 1286-1303, 1960.

[4] Biot M.A., General theory of three-dimensional consolidation, *J. Appl. Phys.*, **12**: 155-164, 1941.

[5] Cavendish J.C., Price H.S. and Varga R.S., Galerkin Methods for the Numerical Solution of Boundary Value Problems, *Soc. Pet. Eng. J.*, 204-220, 1970.

[6] Charlez PhA., Rock Mechanics, Volume 1: Theoretical Fundementals, Editions Technip, France, 1991.

[7] Chrichlow H.B., Modern Reservoir Engineering – A simulation Approach, Prentice Hall, New Jersey, 1977.

[8] Crumpton P.I., Moinier P. and Giles M.B., An unstructured algorithm for high Reynolds number on highly stretched meshes. Numerical Methods in Laminar and Turbulent Flow (ed. C. Taylor and J. T. Cross), pages 561-572, 1997.

[9] Dalen V., Simplified finite element models for reservoir flow problems, *SPE 7196*, 1979.

[10] Finder G.F. and Friend E.O., Application of Galerkin's Procedure to Aquifer Analysis, *Water Resour. Res.*, 8(1): 108-120, 1972.

[11] Fung L.S.-K., Hiebert A.D. and Nghiem L., Reservoir simulation with a control-volume finite-element method, *SPE 21224*, 1991.

[12] Ghafouri H.R. and Lewis R.W., A finite element double porosity model for heterogeneous deformable porous media, *International Journal for Numerical and Analytical Methods in Geomechanics*, **20**: 831-844, 1996.

[13] Ghafouri H.R., Finite element modelling of multi-phase flow through deformable fractured porous media, PhD Thesis, University of Wales Swansea, 1996.

[14] Gutierrez M. and Lewis R.W., The role of geomechanics in reservoir simulation, Proceedings of the SPE/ISRM EUROCK'98, (SPE paper 47392), Trondheim, Norway, 1998.

[15] Javandel A. and Witherspoon P.A., Application of the Finite Element Method to Transient Flow in Porous Media, *Soc. Pet. Eng. J.*, 241-252, 1968.

[16] Kebaiti A.L. and Thomas G.W., A Two-Phase Coning Model Using Alternating Direction Galerkin Procedure, paper *SPE 5724* presented at the SPE-AIME Symposium on Numerical Simulation of Reservoir Performance, Los Angeles, Feb. 1976.

[17] Langsrud O., Simulation of Two-Phase Flow by Finite Element Methods, paper *SPE 5725* presented at the SPE-AIME Symposium on Numerical Simulation of Reservoir Performance, Los Angeles, Feb. 19-20, 1976.

[18] Lemonnier P.A., Improved of reservoir simulation by a triangular discontinuous finite element method, *SPE 8249*, 1979.

[19] Lewis R.W. and Ghafouri H.R., A novel finite element double porosity model for multiphase flow through deformable fractured porous media, *International Journal for Numerical and Analytical Methods in Geomechanics*, **21**(11): 789-816, 1997.

[20] Lewis R.W. and Pao W.K.S., Numerical simulation of three-phase flow in deforming fractured reservoirs, *Oil & Gas Science and Technology-Revue de L Institut Francais Du Petrole*, **57**(5): 499-514, 2002.

[21] Lewis R.W. and Schrefler B.A., The finite element method in the static and dynamic deformation and consolidation of porous media (2^{nd} edn.). Wiley: England, 1998.

[22] Lewis R.W. and Sukirman Y., A Coupled Implicit Flow Model For Deforming Porous-Media, Computational Mechanics: From Concepts to Computations, Vols. 1 and 2, 573-578, 1993.

[23] Lewis R.W. and Sukirman Y., Finite-element modeling for simulating the surface subsidence above a compacting hydrocarbon reservoir, *International Journal for Numerical and Analytical Methods in Geomechanics*, **18**(9): 619-639, 1994.

[24] Lewis R.W. and Sukirman Y., Finite-Element Modeling Of 3-Phase Flow In Deforming Saturated Oil-Reservoirs, *International Journal for Numerical and Analytical Methods in Geomechanics*, **17** (8): 577-598, 1993.

[25] Lewis R.W., Masters I., Guiterrez M. and Wilkes S., 3D elastoplastic analysis in soft rock petroleum reservoirs, in M. Cross (ed.), Proc. 8th Annual Conference of the Association for Computational Mechanics in Engineering, University of Greenwich, 202-205, 2000.

[26] Lewis R.W., Vemer E.A. and Zienkiewicz O.C., A Finite Element Approach to Two-Phase Flow in Porous Media, Finite Elements in Fluids, John Wiley & Sons Ltd., London, 1975.

[27] Luo H., Baum J.D. and Lohner R., Edge-based finite-element scheme for the Euler equations. *AIAA*, **32**(6): 1183-1190, 1994.

[28] Malan A.G., Lewis R.W. and Nithiarasu P., An improved unsteady, unstructured, artificial compressibility, finite volume scheme for viscous incompressible flows: Part I. Theory and implementation. *International Journal for Numerical Methods in Engineering*, **54**(5): 695-714, 2002.

[29] Masters I., Pao W.K.S. and Lewis R.W., Coupling temperature to a double-porosity model of deformable porous media, *International Journal for Numerical Methods in Engineering,* **49**: 421-438, 2000.

[30] Pao W.K.S. and Lewis R.W., Three-dimensional finite element simulation of three-phase flow in a deforming fissured reservoir, *Comput. Methods Appl. Mech. Engrg.*, **191**: 2631-2659, 2002.

[31] Pao W.K.S., Aspects of continuum modelling and numerical simulations of coupled multiphase deformable non-isothermal porous continua, PhD Thesis, University of Wales Swansea, 2000.

[32] Pao W.K.S., Lewis R.W. and Masters I., A fully coupled hydro-thermo-poro-mechanical model for black oil reservoir simulation, *International Journal for Numerical and Analytical Methods in Geomechanics*, **25**: 1229-1256, 2001.

[33] Schneider G. and Raw M.J., A Skewed Positive Influence Coefficient Upwinding procedure for Control Volume Based Finite Element Convection Diffusion Computation, *Num. Heat Trans.*, **9**: 1-26, 1986.

[34] Sonier F. and Eymard R., Mathematical and Numerical Properties of Control-Volume Finite-Element Scheme for Reservoir Simulation, *SPE 25267*, 1993.

[35] Sorensen K.A., A Multigrid Acceleration Procedure for the Solution of Compressible Fluid Flows on Unstructured Hybrid Meshes. PhD Thesis, School of Engineering, University of Wales Swansea, 2001.

[36] Spivak, Price H.S. and Settari A., Solution of the Equations for Multi-Dimensional Two-Phase Immiscible Flow by Variational Methods, *Soc. Pet. Eng. J.*, 27-41, 1977.

[37] Sukirman Y., Petroleum reservoir simulation coupling flow and subsidence, PhD Thesis, University of Wales Swansea, 1993.

[38] Warren J.E. and Root P.J., The behaviour of naturally fractured reservoirs, *SPE J. Trans. AIME*, **228**: 244-255, 1963.

[39] Zienkiewicz O.C. and Parekh C.J., Transient Field Problems: Two-Dimensional and Three- Dimensional Analysis by Isoparametric Finite Elements, *International Journal for Numerical Methods in Engineering*, **2**: 61-71, 1970.

Chapter 2

SPATIAL DISCRETIZATION PROCEDURES

2.1 INTRODUCTION

This chapter focuses on the spatial discretization aspect of the chosen numerical procedure. The governing equations describing groundwater flow problems which are the main focus of this work are strongly non-linear, and in the case of contaminant transport problems, are highly coupled. Hence, an analytical solution of the problem is impossible. A numerical technique is therefore sought, which will enable subdivision of the system of equations into a number of discrete expressions that are solvable. Here, this construction is performed by an edge-based vertex-centred finite volume approach. A short discussion comparing this method with alternative discretisation approaches is presented. This method is formulated for unstructured and structured meshes, and the dual mesh definitions of these meshes are described in detail. Moreover, the purely edge-based technique is enhanced by utilizing an alternative calculation procedure, and it is demonstrated that this results in a significant improvement in accuracy when calculating derivative terms. Finally, the calculation of derivative terms on a non-homogeneous domain using the standard and alternative calculation procedure is considered.

2.2 WEIGHTED RESIDUAL METHODS

Problems concerning the flow of fluid through porous media are complex from a numerical point of view as the governing equations contain widely varying mathematical characteristics and measured non-linearity. Most numerical schemes in use today, which are applicable to such problems where the governing equations are of a differential type, can be considered a member of the *Weighted*

Residual Method (WRM) family of discretisation procedures. Well known examples of this technique are the finite element and finite volume procedures.

In WRM, a system of n differential equations of the form

$$f_m\left(\mathbf{Q}_m(\mathbf{x},t),\mathbf{x},t\right)=0\ ,\quad \mathbf{x}\in\Omega,\quad m\in[1,n] \tag{2.1}$$

is considered, with initial and boundary conditions $\mathbf{Q}_{\mathrm{m}}(\mathbf{x},0)$ and $f_m=g_m\left(\mathbf{Q}_m(\Gamma)\right)$ is solved. Here $\mathbf{Q}_m(\mathbf{x})$ denote the dependent and $\mathbf{x}$ and t the independent variables. Ω designates the spatial extent of the domain, and Γ is the bounding surface. By subdividing the domain Ω into discrete volumes Ω_m and integrating over each, the system of *weak forms* is introduced in the manner

$$\int_{\Omega^m} f_m\left(\mathbf{Q}_m(\mathbf{x},t),\mathbf{x},t\right)\mathbf{w}(\mathbf{x})\,d\Omega=0 \tag{2.2}$$

where $\mathbf{w}(\mathbf{x})$ is a vector of length N of independent weighting functions. The result is a set of $N\times n$ discrete equations. The objective now is to obtain an approximation for the dependent variables or unknowns, $\tilde{\mathbf{Q}}$, which satisfy the set of equations (2.1) and the initial and boundary conditions prescribed to the system.

2.3 FINITE ELEMENT METHODS

In finite element methods, the approximate solution is constructed by setting

$$\tilde{\mathbf{Q}}=\sum_{P=1}^{N}\mathbf{Q}_P S_P(\mathbf{x}) \tag{2.3}$$

where $\left\{S_P(\mathbf{x}),P\in[1,N]\right\}$ is a finite set of independent basis-functions and $\left\{\mathbf{Q}_P\in\mathbb{R}^n,P\in[1,N]\right\}$ is a set of unknowns and constants. The result is a system of equations with $N\times n$ unknowns. Different choices of weighting functions results in the construction of different families of discretisation schemes. Spectral schemes result from setting weighting and basis functions to global trigonometric functions, while applying the Dirac delta for the weighting functions yields finite difference schemes. The simplest and most commonly used finite element method is the Galerkin method which is constructed by setting the weighting and basis functions to be identical

The spatial domain is subdivided into discrete non-overlapping elements (Figure 2.1) from which the basis-functions are constructed over the elements. As shown the method is applicable to unstructured grids, making it suitable to complex geometries. Element borders are referred to as edges and nodes designate where the edges intersect. A dependent variable set $\mathbf{Q}_p$ (of dimension n) is associated with each node (of which there are N).

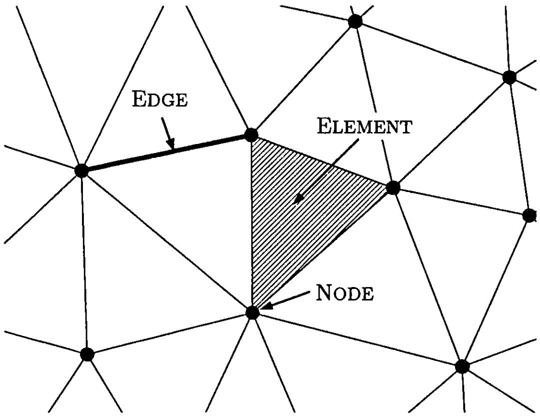


Figure 2.1: Illustration of the nomenclature used for unstructured meshes; here demonstrated on a 2-D triangular mesh.

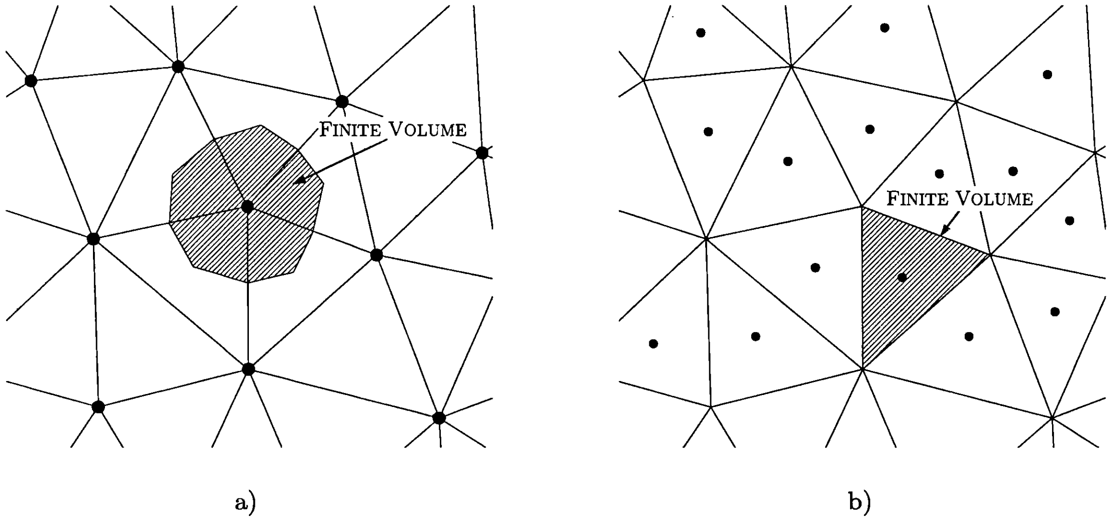


Figure 2.2: Schematics of the two finite volume domain composition variants applied to unstructured grids, a) vertex-centred, and b) cell-centred. The filled circles denote the locations of nodes, and hence the discrete unknowns.

2.4 FINITE VOLUME METHODS

The finite volume scheme results by choosing the weighting function in (2.2) as unity to split the computational domain into a set of subdomains

$$\mathbf{w}_m = \begin{cases} 1, & \mathbf{x} \in \Omega_m \\ 0, & \mathbf{x} \notin \Omega_m \end{cases} \tag{2.4}$$

where Ω_m denotes the non-overlapping subdomain associated with $\mathbf{w}_m$ such that $\{\Omega_m \subset \Omega\}$. Each non-overlapping domain, Ω_m, is constructed from meshes similar to the ones used in finite element schemes, and such finite volume schemes are divided into two families: *element-based/cell-centred* and *node-based/vertex-centred*. The difference between these two families of schemes is found in the way the subdomain sets, or *control volumes*, $\{\Omega_m\}$ are constructed.

In vertex-centred methods, the dependent variables are stored at nodes, and the control volumes are constructed around each node in such a way that a node resides in one subdomain only. The set of connecting surfaces between these subdomains is referred to as the *dual-mesh*. An illustration of a vertex-centred control-volume definition of an unstructured triangular mesh is shown in Figure 2.2a. Cell-centred methods do not require the creation of a dual mesh, but instead use the elements of the mesh themselves as control-volume definitions with unknowns being stored at the element centres. An illustration of a cell-centred control-volume definition of an unstructured triangular mesh is shown in Figure 2.2b.

In comparison, the cell-centred variant is significantly more memory intensive (a factor of two in 2-D) on unstructured grids (Foy and Dawes [5]) in part due to the ratio between the elements and nodes (also evident from Figure 2.2b). In addition, the vertex-centred method allows for edge-based computation which offers a significant advantage in computational cost as compared to element based methods as pointed out by Luo *et al.* [8]. This is because looping over the elements (element based methods) requires the computation of the same edge contribution twice (in 2-D), and more than four times in 3-D. Another advantage of the vertex-centred scheme is that it is similar, and in some cases equivalent (e.g. Onate *et al.* [13]), to extensively used finite difference and finite element methods. Node-base schemes further avoid the complexities associated with cell-centred schemes when applying boundary conditions. Considering this in conjunction with the storage and computational advantages, the edge-based vertex-centred scheme is deemed to be the superior finite volume approach.

When considering vertex-based algorithms, variations exist in the method employed to calculate second order derivatives. These range from purely edge-based (as per Malan *et al.* [10], Sorensen [15]) to element-based techniques such as the method developed by Turner and Ferguson [18]. The former will be utilised as it lends naturally to the memory efficient edge-based storage structure. An edge-based representation of the domain is employed as it facilitates computational and storage efficiency. This discretisation scheme is naturally applicable to unstructured meshes and is thus suitable where complex geometries are prevalent. This further facilitates the use of solution-adapted grids which have been shown to be both accurate and computationally efficient (Lewis *et al.*

[7] and Nithiarasu and Zienkiewicz [12]). Moreover, hybrid grids are also supported, with the benefits of increased accuracy and performance as demonstrated by Chen and Kallinderis [3] and Sorensen *et al.* [15].

2.5 DUAL-MESH CONSTRUCTION

The dual-mesh is constructed by connecting *element geometric centres* and *edge midpoints,* and in 3-D face-centroids, in such a way that only one node is present in each control volume as per Vahdati *et al.* [19], and the dependent variables are stored at these nodes. This is shown schematically for a boundary node m on a 2-D domain in Figure 2.3, where the black filled circles denote nodes, the white circles denote element geometric centres, the squares denote edge midpoints and the dual is denoted by dashed lines. We shall limit our interest to the 2-D case. The control volume (subdomain) associated with node m is designated Ω_m and the bounding surfaces by Γ_m and Γ_m^b respectively. The latter denote internal (dual) and computational boundary surfaces respectively. This method is suitable for use on unstructured meshes.

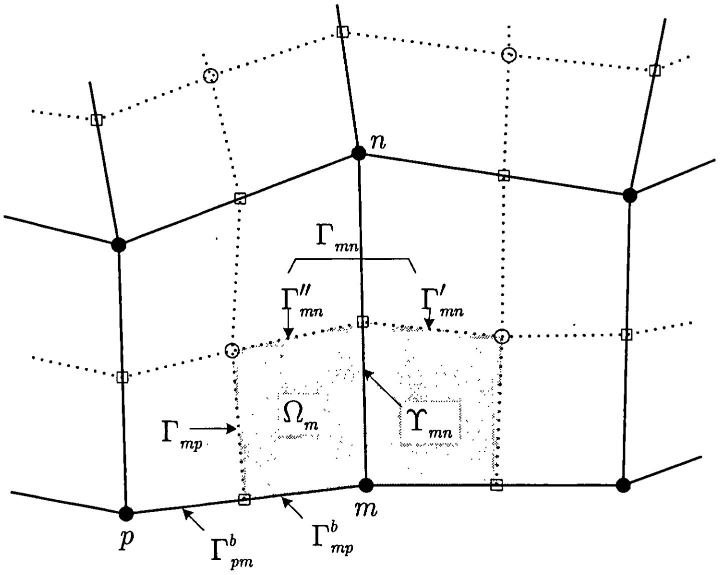


Figure 2.3: Schematic of the construction of the median-dual-mesh (dashed lines) on an unstructured quadrilateral grid. Points shown within elements depict the element geometric centres.

Internal and boundary surfaces are constructed such that $\Gamma_{mp} \cap \Gamma_{mp}^{b} \neq \varnothing$ and $\left|\Gamma_{mp} \cap \Gamma_{mp}^{b}\right| = 0$, where $|\bullet|$ returns the length of its argument. The bounding surface Γ_{m} is composed of a number of surface segments which are defined in an edge-wise manner. For example, Γ_{mn} is the segment associated with the edge Υ_{mn} which connects nodes m and n. Γ_{mn} is constructed by summation of the surface segments attached to this edge i.e. $\Gamma_{mn} = \Gamma'_{mn} \cup \Gamma''_{mn}$ (as shown in Figure 2.3). Domain boundary surfaces are similarly composed of segments where Γ_{mp}^{b} denotes the boundary surface sector associated with the boundary edge Υ_{mp}.

In order to exploit the computational advantages associated with edge-based matrix assembly, bounding surface information is stored in an edge-wise manner. These are termed as *edge coefficients*, and in the case of internal edges on a 2-D domain as shown in Figure 2.3 are defined as

$$\mathbf{C}_{mn} = \mathbf{n}'_{mn}\left|\Gamma'_{mn}\right| + \mathbf{n}''_{mn}\left|\Gamma''_{mn}\right| \qquad \text{if} \quad \Gamma_{mn} \not\subset \partial\Omega \tag{2.5}$$

where, m and n denote the nodes attached by edge Υ_{mn}, and $\mathbf{n}'_{mn}$, $\mathbf{n}''_{mn}$ and $\mathbf{n}_{mn}$ denote unit vectors orthogonal to Γ'_{mn}, Γ''_{mn} and Γ_{mn} respectively. For consistency sake these vectors are oriented from node m to node n, and are outward pointing with respect to the control volume Ω. The domain boundary contributions are computed in a similar fashion where the associated boundary edge coefficient is given by

$$\mathbf{B}_{mp} = \mathbf{n}_{mp}^{b}\left|\Gamma_{mp}^{b}\right| \qquad \text{if} \quad \Gamma_{mn} \subset \partial\Omega \tag{2.6}$$

where, $\mathbf{n}_{mp}^{b}$ denotes the outward pointing normal vector to boundary bounding surface segment Γ_{mp}^{b}. From equation (2.5) it can be seen that the internal edge coefficients are antisymmetric, i.e.

$$\mathbf{C}_{mn} = -\mathbf{C}_{nm} \tag{2.7}$$

which means that the internal edge coefficients only need to be stored once for every edge. From equations (2.5) and (2.6) it follows that

$$\sum_i C^j_{mi} + \sum_i B^j_{mi} = 0\,, \tag{2.8}$$

where i is the set of nodes directly connected to node m by a single edge. Hence by (2.8), the scheme is consistent in the case that the surface integral of a constant flux is numerically zero. The identity,

$$\mathbf{B}_{mp} = \mathbf{B}_{pm} \tag{2.9}$$

for the boundary coefficients is also true as long as the edge midpoints are used in the dual mesh construction. This means that the boundary coefficients also need to be stored once only for each edge. Since the definitions of the internal and boundary edge coefficients are purely geometrical, they need only be calculated once, unless the mesh is deformed.

The governing partial differential equations to be solved exhibit a combination of two types of mathematical characteristics viz. hyperbolic and elliptic. The hyperbolic aspect is convective in nature while the elliptic is diffusive. Numerically each is to be treated appropriately in order to facilitate notionally second order accuracy and stability. This is considered next.

2.6 DEVELOPMENT OF THE EDGE-BASED ALGORITHM

Initially, the governing equations must be written in a weak form, and then each term in the weak form of the equation system is dealt with separately as given below.

2.6.1 First Derivative Terms

The spatial derivative terms in the weak form equation are discretised by application of Green's theorem, and by the assumption that the spatial derivative is linear over the control volume Ω_m surrounding the node m, and equal to its approximated value at the midpoint of the edge. We implement the standard index notation but with no summation over indices.

For the purpose of describing the spatial discretization algorithm, a generic first derivative term is defined as

$$\mathfrak{M} = \frac{\partial \zeta}{\partial x_i} \tag{2.10}$$

where $\zeta =$ an arbitrary parameter. Equation (2.10) cast into a weak form follows viz.:

$$\int_{\Omega_m} \frac{\partial \zeta}{\partial x_i} d\Omega_m = \int_{\Gamma_{\Omega_m}} \zeta \; n^i_{mn} d\Gamma_m \tag{2.11}$$

An averaged gradient $\overline{\partial \zeta / \partial x_i}$ on the control volume associated with the vertex m is defined as the quantity satisfying:

$$\left.\overline{\frac{\partial \zeta}{\partial x_i}}\right|_{\Omega_m} = \frac{1}{|\Omega_m|} \int_{\Omega_m} \frac{\partial \zeta}{\partial x_i} d\Omega_m \tag{2.12}$$

where $|\Omega_m| = \int_{\Omega_m} d\Omega_m$. This is a first-order accurate discretization of the first-order derivative of ζ taken at the node m, namely $\left.\partial \zeta / \partial x_i\right|_m$. That is,

$$\left.\frac{\partial \zeta}{\partial x_i}\right|_m = \left.\overline{\frac{\partial \zeta}{\partial x_i}}\right|_{\Omega_m} + O(\Delta x) \tag{2.13}$$

assuming that ζ is a twice continuously differentiable function, we obtain the correct estimate of the approximation error,

$$\begin{aligned}\left|\left.\overline{\frac{\partial\zeta}{\partial x_i}}\right|_{\Omega_m} - \left.\frac{\partial\zeta}{\partial x_i}\right|_m\right| \int_{\Omega_m} d\Omega_m &\le \int_{\Omega_m}\left|\frac{\partial\zeta}{\partial x_i} - \left.\frac{\partial\zeta}{\partial x_i}\right|_m\right| d\Omega_m \\ &\le \max_{x\in\Omega_m}\left|\frac{\partial^2\zeta(x)}{\partial x_i^2}\right|\Delta x\int_{\Omega_m} d\Omega_m = O\left(\Delta x^3\right)\end{aligned} \tag{2.14}$$

where $i = x,y$ and $\bar{\zeta}_m$ is the average ζ across the control volume Ω_m. Applying the divergence theorem to $\partial\zeta/\partial x_i$ we have:

$$\begin{aligned}\int_{\Omega_m}\frac{\partial\zeta}{\partial x_i}d\Omega_m &= \int_{\Gamma_{\Omega_m}}\zeta\, n^i_{mn}d\Gamma_m \\ &\approx \sum_{\Upsilon_{mn}\cap\Omega_m}\bar{\zeta}_{mn}C^i_{mn} + \sum_{\Upsilon^b_{mn}\cap\Omega_m}\zeta_m B^i_{mn} \\ &= \sum_{edges}\left.\frac{\widehat{\partial\zeta}}{\partial x_i}\right|_{mn}\end{aligned} \tag{2.15}$$

where Υ_{mn} denotes the edge connecting nodes m and n, and Υ^b_{mn} denotes the edge connecting boundary nodes m and n. The edge coefficients $\mathbf{C}_{mn}$ and $\mathbf{B}_{mn}$ are defined as per equations (2.5) and (2.6). Γ_m is the surface of the control volume Ω_m and $\bar{\zeta}_{mn}$ is the average flux along the edge. The term $\left.\widehat{\partial\zeta}/\partial x_i\right|_{mn}$ is the partial contribution to the derivative associated with edge$_{mn}$. The edge-based scheme is numerically equivalent to a vertex-centred scheme, however, it is this aspect of the formulation that enables a computationally efficient implementation. The standard method of calculating the average value of a quantity that contributes towards the flux calculation along an edge (Vahdati *et al.* [19], Crumpton *et al.* [4], Sorensen *et al.* [16] and Malan *et al.* [9]) is:

$$\bar{\zeta}_{mn} = \frac{1}{2}(\zeta_m + \zeta_n) \tag{2.16}$$

Finally, the calculation of a surface integral for the spatial derivative over the control volume surface for node m is then given by the formula

$$\boxed{\begin{aligned}\int_{\Omega_m}\frac{\partial\zeta}{\partial x_i}d\Omega_m &= \left.\frac{\partial\zeta}{\partial x_i}\right|_m \Omega_m \\ &= \sum_{\Upsilon_{mn}\cap\Omega_m}\bar{\zeta}_{mn}C^i_{mn} + \sum_{\Upsilon^b_{mn}\cap\Omega_m}\zeta_m B^i_{mn}\end{aligned}} \tag{2.17}$$

Thus, the last term only contributes if m is a boundary node. The boundary terms are here treated in the classic finite volume way, using a local midpoint rule. On a structured mesh this relation is thought of as second order accurate.

Note that expressions (2.7) and (2.8) guarantee the numerical scheme to be conservative, i.e. the sum of the numerical fluxes in the computational domain equals the flux over the computational domain boundary

$$\sum_m\left[\sum_{\Upsilon_{mn}\cap\Omega_m}\bar{\zeta}_{mn}C^i_{mn} + \sum_{\Upsilon^b_{mn}\cap\Omega_m}\zeta_m B^i_{mn}\right] = \sum_m\sum_{\Upsilon^b_{mn}\cap\Omega_m}\zeta_m B^i_{mn} \tag{2.18}$$

2.6.2 Second Derivative Terms

The elliptic terms are those containing second order spatial derivatives and are numerically far less problematic than hyperbolic terms. For the purpose of describing the spatial discretization algorithm, a generic second derivative term is defined as

$$\mathfrak{A} = \frac{\partial}{\partial x_i}\left[\alpha\frac{\partial\zeta}{\partial x_j}\right] \tag{2.19}$$

where α is a scalar phenomenological coefficient and ζ is an arbitrary parameter. In the case of some of the problems considered in this work, notably the equations that describe fluid flow in porous media, α is a highly non-linear function of the material relative permeability. The scalar phenomenological coefficient α is an edge-based term defined as follows

$$\alpha = \alpha_{mn} = \frac{1}{2}\left(\alpha_m + \alpha_n\right), \tag{2.20}$$

where the term α_{mn} is the value of α applied to edge_{mn}. Malan [9] showed that handling these phenomenological coefficient in this way results in formal second order accuracy being reinstated while maintaining the edge-based nature of the scheme. This enhancement is expected to be of significance specifically where the terms are non-linear. A similar averaging technique has in principle been considered for finite element schemes but abandoned due to the computational costs involved (Fletcher [6]). In our case the edge-based nature of the scheme again serves to minimize the additional computational cost as it involves one additional addition and division per edge computation.

Equation (2.19) cast into a weak form follows viz.:

$$\int_{\Omega_m} \frac{\partial}{\partial x_i}\left[\alpha_{mn}\frac{\partial \zeta}{\partial x_j}\right] d\Omega_m = \int_{\Gamma_{\Omega_m}} \left[\alpha_{mn}\frac{\partial \zeta}{\partial x_j}\right] n^i_{mn} d\Gamma_m \tag{2.21}$$

Edge-based finite volume methods do not allow the calculation of second order derivatives with one loop over edges, which is a deficiency as compared to finite element methods. Elliptic terms are calculated by looping over all edges twice, whereby the first derivative is approximated in the first sweep. Along these lines the discretization of (2.21) starts by calculating the first derivative of the scalar field ζ by application of equation (2.17), and dividing through by the control volume as follows;

$$\left.\frac{\partial \zeta}{\partial x_i}\right|_m \approx \partial_i^{fv}\zeta_m \equiv \frac{1}{|\Omega_m|}\left[\sum_{\Upsilon_{mn}\cap\Omega_m} \overline{\zeta}_{mn} C^i_{mn} + \sum_{\Upsilon^b_{mn}\cap\Omega_m} \zeta_m B^i_{mn}\right] \tag{2.22}$$

where ∂_i^{fv} denotes the finite volume approximation of a first derivative with respect to x_i. Here, the midpoint rule has been applied for the volume integral.

Setting $J^{ij}_m = \alpha_{mn}\partial^{fv}_i\zeta_m$, the boundary integral of the gradient is now evaluated directly as

$$\boxed{\int\limits_{\Omega_m} \frac{\partial}{\partial x_i}\left[\alpha_{mn}\frac{\partial \zeta}{\partial x_j}\right] d\Omega_m \approx \sum_{\Upsilon_{mn}\cap\Omega_m} \overline{J^{ij}}_{mn} C^j_{mn} + \sum_{\Upsilon^b_{mn}\cap\Omega_m} J^{ij}_m B^j_{mn}} \tag{2.23}$$

where the standard method employed to calculate the average term $\overline{J^{ij}}_{mn}$ is described as per equation (2.16).

Note that in the first derivative loop, if the first derivative term (2.10) is proceeded by a phenomenological coefficient α, then we proceed in a similar manner as described here for second derivative terms, and equation (2.17) is amended to be

$$\boxed{\begin{aligned} \int\limits_{\Omega_m} \alpha_{mn}\frac{\partial \zeta}{\partial x_i} d\Omega_m &= \alpha_{mn}\left.\frac{\partial \zeta}{\partial x_i}\right|_m \Omega_m \\ &= \sum_{\Upsilon_{mn}\cap\Omega_m} \alpha_{mn}\bar{\zeta}_{mn} C^i_{mn} + \sum_{\Upsilon^b_{mn}\cap\Omega_m} \alpha_{mn}\zeta_m B^i_{mn} \end{aligned}} \tag{2.24}$$

Equation (2.23) is second order accurate discretisation on smooth structured meshes, and simplex meshes, but in the case of non-mixed derivatives the computational stencil spans across five nodes in each direction. For a uniform structured mesh, this means that the nodes are decoupled from the closest neighbouring nodes in this term, hence the contribution from the nodes closest to the node at which the derivative is being calculated is effectively nullified. In such cases the discretization error is twice that obtained with a three-node stencil and an alternative is to be sought. Moreover, it is usually desirable from a discretisation point of view to employ schemes that are as compact as possible.

2.6.3 Compact Stencil

A disadvantage of the edge-based method is that when calculating non-mixed second-derivative terms via the standard finite volume approach, equation (2.23) results in a two-layer (five-node) stencil. Hence, the contribution from the nodes adjacent to the node at which the derivative is calculated is not included in the calculation. The author has shown that this produces large spatial oscillations emanating from the discontinuous material boundaries in a non-homogeneous domain [14].

Turner and Ferguson [18] proposed a more compact scheme whereby nodes not attached to an edge but belonging to the attached elements are brought into the calculation, resulting in a three-node stencil. The drawback is however that it is not purely edge-based with potential storage penalties. Swanson and Turkel [17] developed an alternative compact scheme whereby the first derivatives are calculated using equation (2.22) at edge midpoints using an additional edge-based control volume. However, an additional set of edge coefficients are required for this formulation. In addition, the computation of the boundary integrals become more complicated in the sense that the contribution of the integral associated with the edge becomes dependent on all of the nodes in the elements sharing the edge. This results in the loss of the purely edge-based nature of the scheme.

In addressing this issue, a more compact stencil as proposed by Crumpton *et al.* [4] is implemented. This results in a one-layer (three-node) stencil, and the increase in accuracy observed when calculating elliptic terms using the compact stencil as compared to the standard edge-based finite volume scheme is significant as shown by Malan [9]. The compact stencil scheme is put into effect by splitting derivatives calculated at edge midpoints ($\overline{J^{ij}}_{mn}$) into components tangential and normal to the edge, as:

$$\overline{J^{ij}}_{mn} C^{j}_{mn} = \left(\overline{J^{ij}}_{mn}\Big|_{\mathbf{tang}} + \overline{J^{ij}}_{mn}\Big|_{\mathbf{norm}} \right) C^{j}_{mn} \tag{2.25}$$

A new tangential component is calculated as

$$\overline{J^{ij}_{mn}}\Big|^{*}_{\mathbf{tang}} = \alpha_{mn} \frac{\zeta_n - \zeta_m}{\left|\mathbf{x}_n - \mathbf{x}_m\right|} t^{mn}_{j} \tag{2.26}$$

which constitutes the directional derivative along the edge with $\mathbf{t}^{mn}$ designating the unit vector tangential to the edge. The term $|\mathbf{x}_n - \mathbf{x}_m|$ describes the length of the edge between nodes m and n. This tangential component is second order accurate at the edge midpoint but only requires two nodes in its evaluation.

The normal component is calculated by averaging the flux in the normal fashion and subtracting the tangential component.

$$\left.\overline{J^{ij}}_{mn}\right|_{\textbf{norm}} = \overline{J^{ij}}_{mn} - \left(\overline{J^{il}}_{mn} t_l^{mn}\right) t_j^{mn} \tag{2.27}$$

Equation (2.23) now becomes:

$$\boxed{\begin{aligned}&\int_{\Gamma_{\Omega_m}} \left[\alpha_{mn}\frac{\partial \zeta}{\partial x_j}\right] n^i_{mn}\, d\Gamma_m \approx \\ &\qquad \sum_{\Upsilon_{mn}\cap\Omega_m} \left(\left.\overline{J^{ij}_{mn}}\right|^*_{\textbf{tang}} + \left.\overline{J^{ij}_{mn}}\right|_{\textbf{norm}}\right) C^j_{mn} + \sum_{\Upsilon^b_{mn}\cap\Omega_m} J^{ij}_m B^j_{mn}\end{aligned}} \tag{2.28}$$

In the case where m is an internal node and the basis vector of the implicated edge coefficients are reasonably aligned with the associated edge, the normal component $\left.\overline{J^{ij}_{mn}}\right|_{\textbf{norm}}$ is small and a three-node stencil results. The additional computational cost compared to the standard method is marginal and this technique is viewed as an elegant method of improving accuracy while conserving the edge-based nature of the scheme.

Sorensen [15] reports that the five-node stencil of the standard second derivative calculation did not show itself to be a problem when solving Computational Fluid Dynamics (CFD) problems on unstructured hybrid meshes. However, it shall be seen that when considering non-homogeneous domains, and point sources such as injection/producing wells, the difference obtained in the solution using the standard scheme and the compact stencil scheme is great (see sections 2.9.1.2, 6.4 and 7.4.2).

2.6.4 Point Source Term

The integration of the point source term, q, within a governing equation yields the following:

$$\boxed{\int_{\Omega_m} q d\Omega_m \approx q_m \Omega_m} \tag{2.29}$$

The above relation is notionally second order accurate.

2.6.5 Time Derivative Term

Finally we consider a time derivative term within a governing equation. Integration, and the application of the Euler backward difference scheme yields:

$$\boxed{\int_{\Omega_m} \frac{\partial \zeta}{\partial t} d\Omega_m \approx \left.\frac{\partial \zeta}{\partial t}\right|_m \Omega_m \approx \frac{\zeta_m^{n+1} - \zeta_m^n}{\Delta t} \cdot \Omega_m} \tag{2.30}$$

Depending on which second derivative scheme is used, by combining equations (2.17), (2.23), (2.28), (2.29) and (2.30) we form the discretised scheme for a governing equation using the standard vertex-centred edge-based finite volume scheme.

2.7 COMPARISON WITH AN ANALYTICAL SOLUTION – DIFFUSION EQUATION

In order to ensure that the derivative routines had been coded correctly and were free from programming errors, it was decided that the code should be compared against the analytical solution for the linear diffusion equation, as presented by Mills [11]. The two-dimensional heat conduction equation considered in this test case is shown here,

$$\frac{\partial}{\partial x}\left(\alpha_x \frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(\alpha_y \frac{\partial T}{\partial y}\right)=\frac{\partial T}{\partial t} \tag{2.31}$$

where the following dimensionless constants were applied; $k = 2.0$, $\rho = 2050.0$, $c_p = 945.36586$ and $\alpha_x = \alpha_y = {}^{k}\!/_{\rho c_p}$. The subroutine used to calculate the analytical solution for this problem is included in Appendix A. The equation was solved for T using the standard vertex-centred edge-based finite volume scheme described.

This governing equation was chosen as it represents a linear, diffusion-dominated flow problem, which is very similar to the porous media fluid flow equations the solver will ultimately be handling. The domain over which the equation was applied was a rectangle of dimension (0.1×0.01) metres and is illustrated in Figure 2.4a. This domain was discretised using 594 regular triangular elements, and the computational mesh is shown in Figure 2.4b.

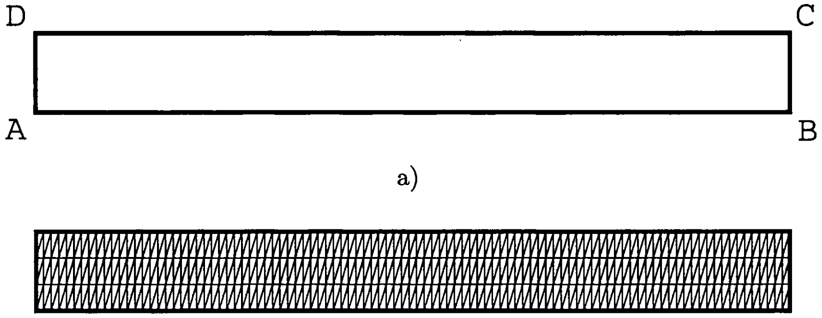
The initial and boundary conditions used for this test case were;

$$T(x,y,0) = 625 \tag{2.32}$$

$$T(x,y,t) = 125 \qquad \text{on side A-D} \tag{2.33}$$

$$\partial T/\partial y = 0 \qquad \text{on sides A-B \& C-D} \tag{2.34}$$

$$\partial T/\partial x = 0 \qquad \text{on side B-C} \tag{2.35}$$



a)

b)

Figure 2.4: Illustration of a) the domain boundary with boundary labels, and b) the computational domain with triangular elements used for the analytical comparison problem.

Figure 2.5 shows contour plots of the front propagating through the domain at $t = 100$, 2000, 5000 and 10000 seconds respectively. The illustration clearly displays the gradual cooling of the bar originating from side D-A, which is as expected. Also of note is that the cooling front is uniform along the width of the domain.

A graph of the calculated values of T along the line $y = 0.005$ m, $x \in [0, 0.1]$m through the domain at $t = 100$, 2000, 5000 and 10000 seconds respectively are shown in Figure 2.6, along with the analytical solution for each time. This line runs through the centre of the domain and along its whole length.

There is a very close correlation between the numerical solution and the analytical solution for the two-dimensional heat conduction problem considered here. In conclusion, it is assumed that the standard derivative routines within the solver are working correctly for linear problems.

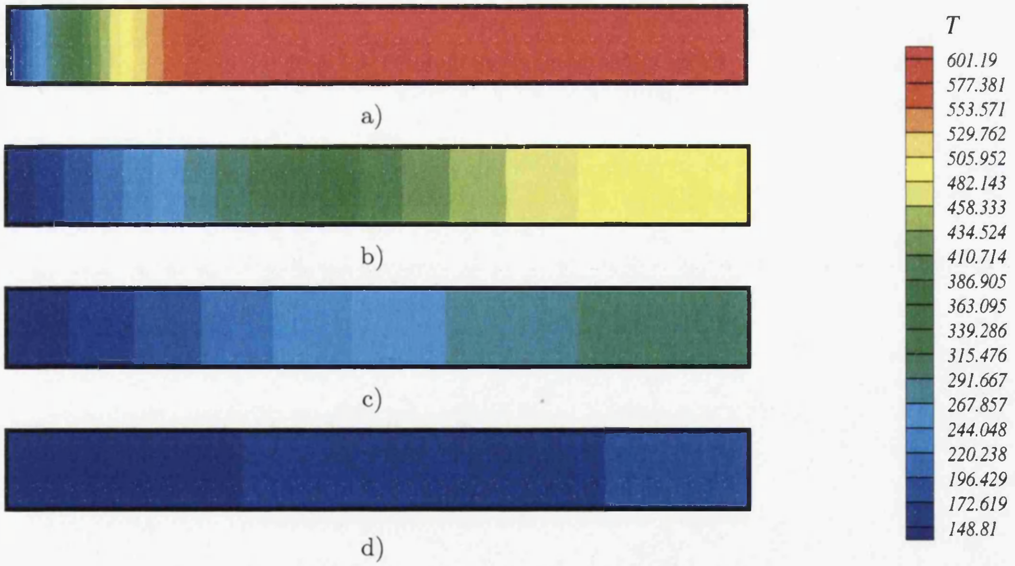


Figure 2.5: Contour plots of the front propagating through the domain at a) 100, b) 2000, c) 5000 and d) 10000 seconds respectively.

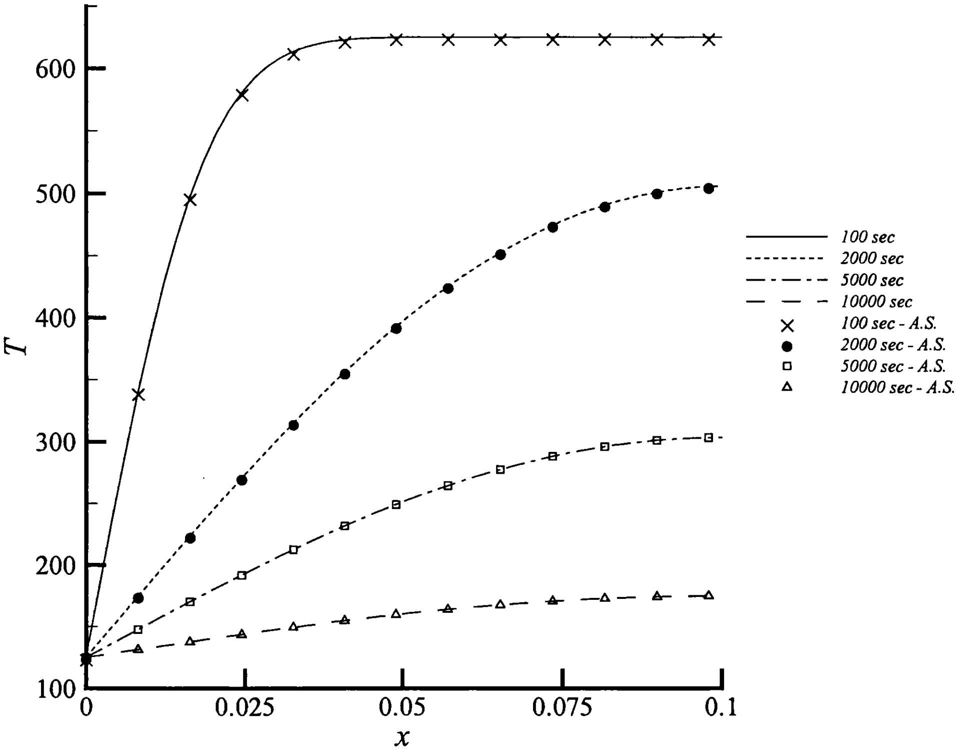


Figure 2.6: Comparison between numerical results and the corresponding analytical solution along the line $y = 0.005$ through the domain at $t = 100$, 2000, 5000 and 10000 seconds respectively.

2.8 NUMERICAL SOLUTION ACCURACIES USING STANDARD AND COMPACT STENCIL FINITE VOLUME SCHEMES

In order to determine the improvement in accuracy obtained by utilising the compact stencil scheme, and to verify the precision of the derivative calculation routines the following tests were conducted. A primitive variable field defined by the function;

$$f(x,y) = \cos(4\pi x) \cdot \cos(4\pi y) \tag{2.36}$$

was applied to a domain of side 1 unit. Harmonic functions, like the one described in equation (2.36) above are known to be difficult to capture due to the infinite series of Taylor terms required to describe them [9]. For this comparison both standard and compact stencil flux-averaging methods are employed. Figure 2.7 illustrates an unstructured mesh that has been distorted to depict the harmonic field.

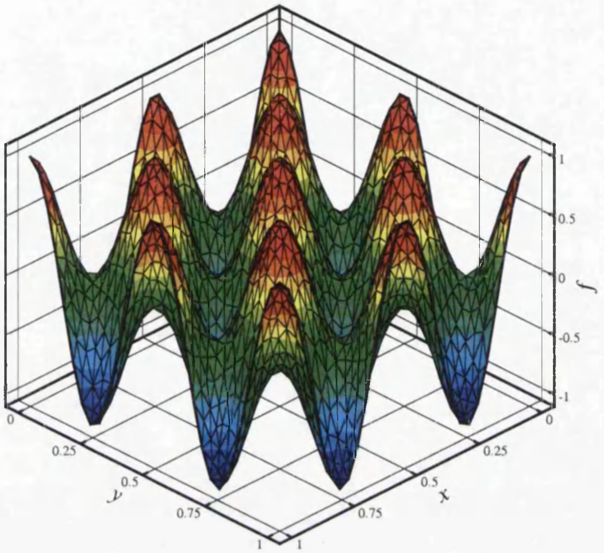


Figure 2.7: The unstructured mesh distorted to depict the harmonic field.

In order to verify the accuracy of the standard vertex-centred edge-based finite volume scheme and the compact stencil finite volume scheme in second calculating derivative terms, the following test was conducted. The primitive variable defined by equation (2.36) was applied to the computational domain, and the second derivative with respect to x (i.e. $\partial^2 f/\partial x^2$) calculated across the domain using the standard and compact stencil schemes on unstructured and structured meshes of varying densities. The results obtained for the derivative calculation using the two schemes each mesh along the poly-line with equation $y = 0.1$ across the domain were then compared to the analytical solution.

Note that for the function described by equation (2.36), the following hold true,

$$\frac{\partial f}{\partial x} = -4\pi\cos(4\pi y)\sin(4\pi x), \tag{2.37}$$

$$\frac{\partial f}{\partial y} = -4\pi \cos(4\pi x)\sin(4\pi y)\,, \tag{2.38}$$

and,

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial^2 f}{\partial y^2} = -16\pi^2 \cos(4\pi y)\cos(4\pi x)\,. \tag{2.39}$$

Also, the value for the constant π was taken to be $\pi = 3.1415926535897932$.

2.8.1 Unstructured Grids

The square domain of side 1 unit was discretised using unstructured triangular elements and is shown in Figure 2.8. The mesh has 35 nodes per boundary, 2356 computational nodes and 6929 elements.

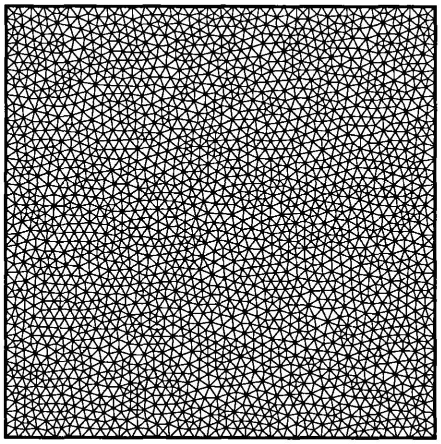


Figure 2.8: Computational domain of side 1 discretised into 6929 unstructured triangular elements with 2356 computational nodes and 35 nodes per boundary.

The calculated results for the second derivative with respect to x obtained for each mesh along the poly line $y = 0.1$ m, $x \in [0,1.0]$m across the domain using the standard and compact stencil schemes are shown in Figure 2.9, along with the analytical solution for the problem. The analytical solution for the second derivative of the primitive variable field with respect to x is described by equation (2.39).

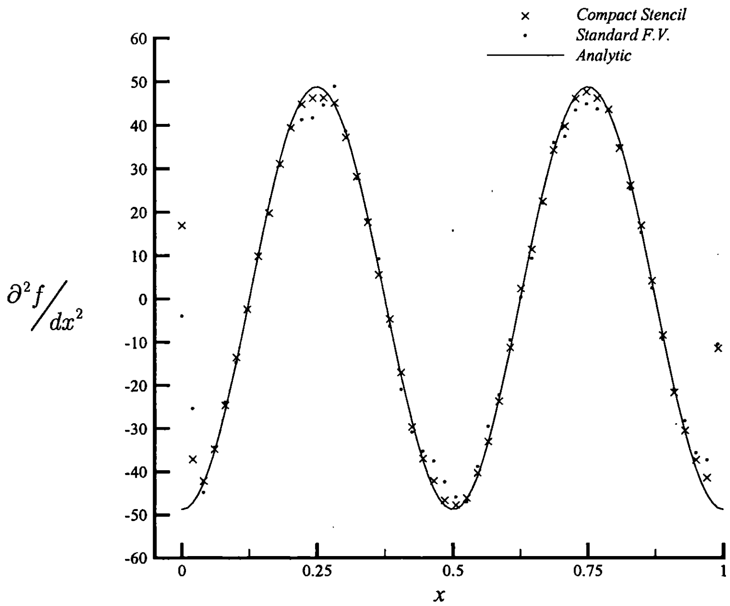


Figure 2.9: Comparison of the predicted values along $y = 0.1$, using the standard and compact stencil edge-based flux averaging algorithms for the second derivative w.r.t. x, to the exact solution on an unstructured mesh with 35 nodes on each boundary.

As can be seen from Figure 2.9, the compact stencil solution captures the peak of the oscillatory solution much more accurately than the standard edge-based scheme. Moreover, in the regions between these peaks the standard scheme produces a solution that lags slightly behind the analytical. This lag is not present in the compact stencil plot. The solution using both schemes diverges from the analytical on the boundaries of the domain. This is a defect inherent to the scheme and cannot be avoided, and is due to the accuracy of the scheme reducing from second order at internal nodes, to first order at, or on adjacent nodes, to the boundary. This anomaly is explained in detail by Malan [9].

In conclusion, the compact stencil finite volume scheme provides a marked improvement over the accuracy obtained by the standard vertex-centred edge-based finite volume scheme when calculating second derivative terms.

2.8.2 Structured Grids

The improvement in solution accuracy using the compact stencil scheme as opposed to the standard finite volume scheme has already been established. The following test was considered in order to compare the solutions obtained for derivative calculations using the standard vertex-centred edge-based finite volume scheme, and the compact stencil finite volume scheme on structured and unstructured grids.

The same domain as considered for the unstructured test above was used, however the domain was now discretised using structured quadrilateral elements. The mesh considered for the test has the same boundary node density as the unstructured mesh shown in Figure 2.8, that is 35 nodes per boundary; and is shown in Figure 2.10.

The same primitive variable field as defined by equation (2.36) was applied to the structured domain. The first derivative with respect to x (i.e. $\partial f / \partial x$) of the primitive variable field was calculated using the standard vertex-centred edge-based finite volume scheme, and the second derivative with respect to x (i.e. $\partial^2 f / \partial x^2$) calculated across the domain using the standard and compact stencil schemes on the mesh illustrated in Figure 2.10. The results obtained for these derivative calculations on the structured meshes along the line $y = 0.1$ m, $x \in [0,1.0]$m through the domain were compared to the solution obtained on an unstructured mesh of the same boundary node density.

The results obtained for the first derivative comparison using the structured mesh is shown in Figure 2.11. The results obtained for the second derivative comparison using the compact stencil scheme and the standard finite volume scheme on the structured mesh are shown in Figure 2.12 and Figure 2.13 respectively.

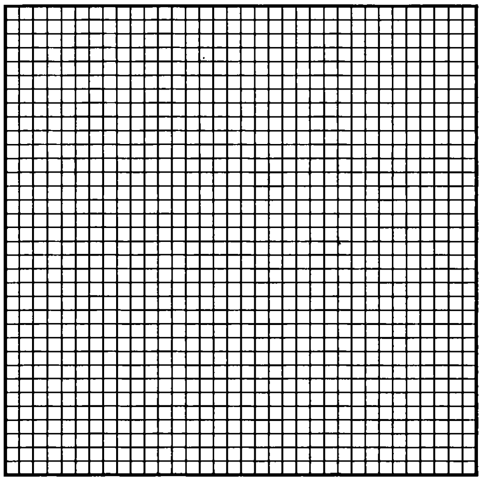


Figure 2.10: Computational domain of side 1 discretised into unstructured quadrilateral elements with 35 nodes per boundary, 1225 computational nodes and 1156 elements.

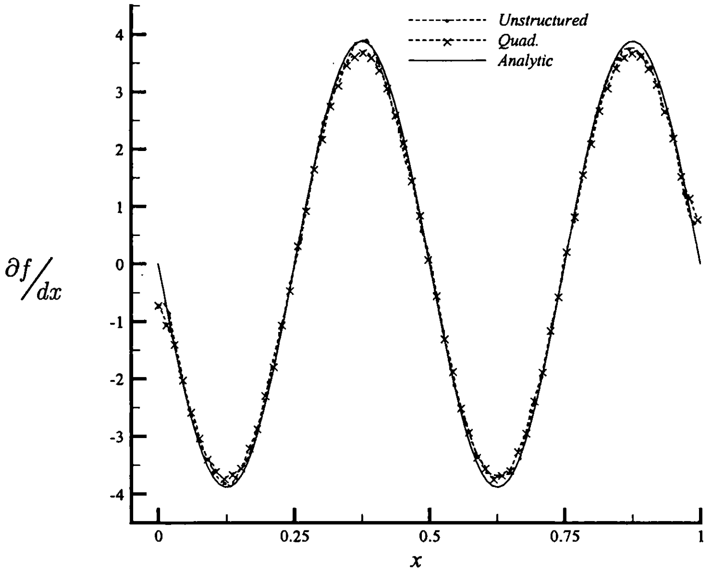


Figure 2.11: Comparison of the solutions obtained for the first derivative w.r.t. x calculation along $y = 0.1$ using the standard F.V. scheme on a structured and an unstructured mesh with 35 nodes on each to the exact solution.

As is evident from Figure 2.11, the solution obtained for the first derivative of the primitive variable with respect to x is very similar for both structured and unstructured meshes. However, the unstructured mesh provides a better solution at the peak of the solution cycle. This is probably due to the large difference in the number of computational nodes between structured and unstructured meshes with the same boundary node density. The unstructured mesh contained 2356 nodes (see Figure 2.8) while the structured mesh contained 1225 nodes.

The solution obtained for the second derivative of the primitive variable with respect to x using the compact stencil scheme is very similar for both structured and unstructured meshes as shown in Figure 2.12. The solutions obtained compares well with the analytical solution for the problem. However, there is a slight improvement in the solution when using the unstructured grids for the same reason as described for the first derivative case.

The solution obtained using the standard finite volume scheme is shown in Figure 2.13. For this case, the structured grid provides a much better solution to the derivative calculation than the unstructured. This is due to inability of the standard scheme to handle the irregular orientation of the edges within an unstructured grid as compared to the structured grid, whose edges are always in line with the coordinate axis in this case. This happens as the standard scheme calculates derivative contributions along the length of the edge. As the compact scheme calculates a contribution to the derivative at a node normal and tangent to an edge, it is independent of edge orientation.

In conclusion, there is little difference in the accuracy of first derivative calculations carried out on structured and unstructured grids of comparable boundary node density. As regards the calculation of second derivatives, as long as the compact stencil scheme is employed then again there is little difference between the two mesh types. However, if the standard second derivative algorithm is applied, then structured grids prove to be superior.

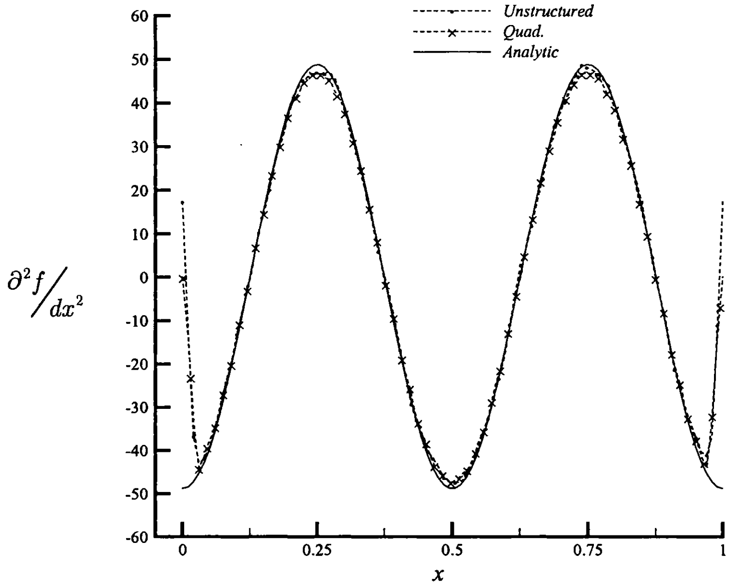


Figure 2.12: Comparison of the solutions obtained for the second derivative w.r.t. x calculation along $y = 0.1$ using the compact stencil scheme on a structured and an unstructured mesh with 35 nodes on each to the exact solution.

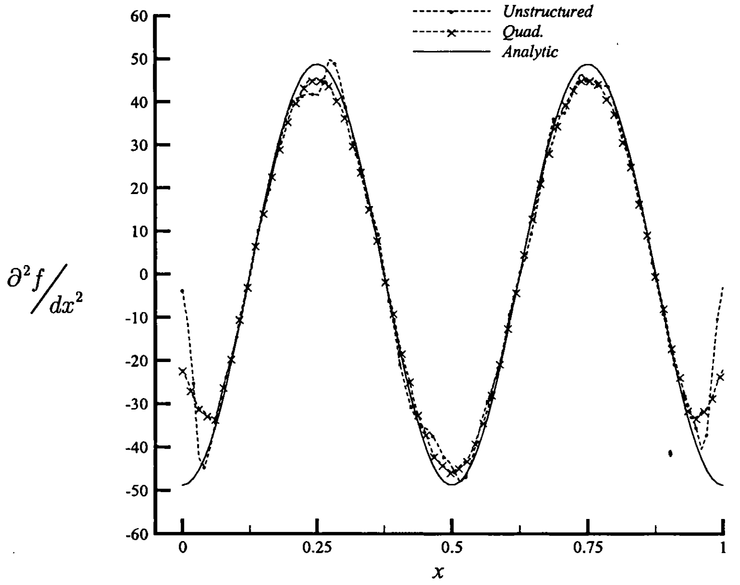


Figure 2.13: Comparison of the solutions obtained for the second derivative w.r.t. x calculation along $y = 0.1$ using the standard F.V. scheme on a structured and an unstructured mesh with 35 nodes on each to the exact solution.

2.9 DERIVATIVE CALCULATIONS ON NON-HOMOGENEOUS DOMAINS

The overall objective of the project is to model fluid flow through porous media. In most instances, this is a purely diffusive, although highly non-linear problem, and is described by diffusion equations such as equation (2.31). From a mathematical perspective, similar equations are seen in petroleum reservoir problems [1], and groundwater flow problems [2]. Here, a linear version of equation (2.31) will be solved on non-homogeneous, isotropic domains using the edge-based finite volume scheme as already described. Although this problem is trivial when considering other more widely used computational schemes, edge-based schemes originate in the Computational Fluid Dynamics community, and thus far, to the authors' knowledge, they have only been applied to homogeneous-isotropic domains. Hence, this section describes how to apply a vertex-centred edge-based finite volume scheme to a non-homogeneous domain.

The domain, of side 1 unit, is shown in Figure 2.14a along with markers to distinguish between each boundary. Markers A to D lie on the corners of the domain, while markers E to H lie on the geometric mid points of the boundaries. The domain has been discretised using 100 regular quadrilateral elements of equal edge length, and the computational grid is shown in Figure 2.14b.

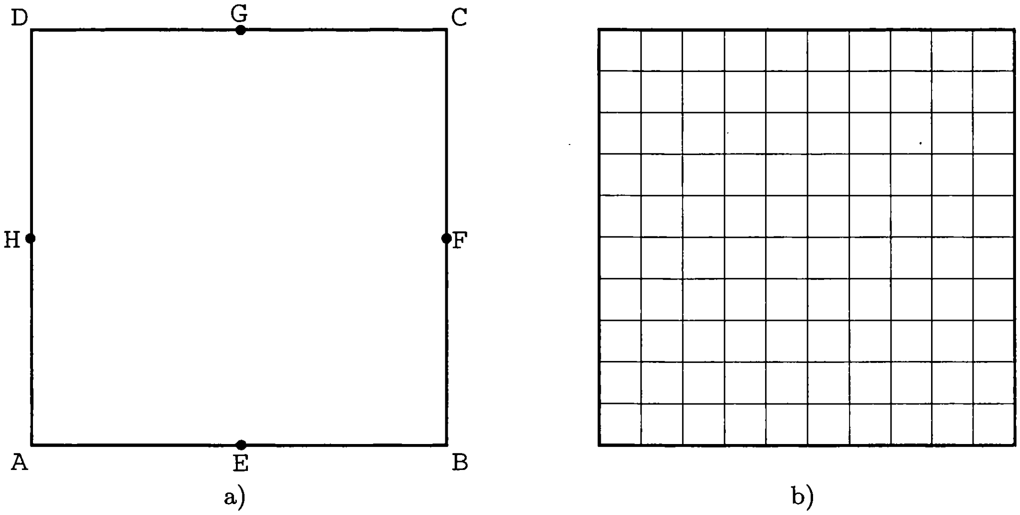


Figure 2.14: Illustration of a) the domain boundary of side 1 unit with boundary labels, and b) the computational domain with quadrilateral elements used for the non-homogeneous domain problem.

In order to establish the correct procedure and formulation for handling derivatives over non-homogeneous domains, the following simple test case was considered.

2.9.1 Test 1 – Vertical Zones

The domain was partitioned into two isotropic zones: 1 and 2, with the boundary between the two zones lying on the line with equation $x = 0.5$ (that is the line connecting points E and G in Figure 2.14a). This partitioned domain is shown in Figure 2.15 below.

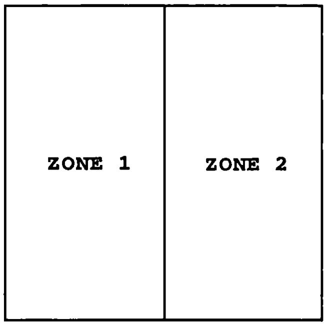


Figure 2.15: The vertically partitioned domain used for the non-homogeneous Test 1.

Each zone was prescribed a different set of dimensionless parameter values as shown in Table 2.1 below.

	Zone 1	*Zone 2*
α_x	10.0	50.0
α_y	10.0	50.0

Table 2.1: Values for the dimensionless parameter α applied to the non-homogeneous domain in Test 1.

Note, for the purpose of this illustration, the parameters in α_x, α_y and T in equation (2.31) are regarded as having no unit or physical meaning. The objective being to formulate and successfully solve a simple diffusion problem on a non-homogeneous, isotropic domain.

The initial and boundary conditions used for this test case were;

$$T(x,y,0) = 0.0 \tag{2.40}$$

$$T(x,y,t) = 1.0 \qquad \text{on side A-D} \tag{2.41}$$

$$T(x,y,t) = 0.0 \qquad \text{on side B-C} \tag{2.42}$$

$$\partial T/\partial y = 0.0 \qquad \text{on sides A-B \& D-C} \tag{2.43}$$

2.9.1.1 Analytical Solution For Non-Homogeneous Case

The analytical solution for the variable T along the zone boundary E-G is derived in the following manner. Assuming that the flux across the domain is constant, then,

$$Q = \alpha \frac{\partial T}{\partial x} \tag{2.44}$$

where Q is the flux. Let the value of T on the left hand vertical boundary be defined as T_1, the value of T on the right hand vertical boundary be defined as T_2, and the value of T on the boundary between the two zones (that is along the line $x = 0.5$) be defined as T_c. In addition, let the value of α in zone 1 be α_1, the value of α in zone 2 be α_2, and the length of boundary A-B be dx. Hence, by the assumption that the flux is constant in both zones, we have,

$$2\alpha_1 \frac{(T_c - T_1)}{dx} = 2\alpha_2 \frac{(T_2 - T_c)}{dx} \tag{2.45}$$

which simplifies to,

$$T_c = \frac{\alpha_2 T_2 + \alpha_1 T_1}{\alpha_1 + \alpha_2} \tag{2.46}$$

By substituting known values for the terms in (2.46); that is $\alpha_1 = 10.0$, $\alpha_2 = 50.0$, $T_1 = 1.0$ and $T_2 = 0.0$, we get,

$$T_c = \frac{1}{6} \tag{2.47}$$

This is the analytical solution for the value of T along the line $x = 0.5$ through the domain for this particular case. This will be used to compare the accuracies of the solutions obtained using the various derivative calculation algorithms.

The problem, equation (2.31), was solved numerically using a timestep length of $dt = 0.0001$ seconds by calculating the required derivatives using the standard and compact stencil edge-based finite volume scheme. Profiles of the calculated parameter T along the line $y = 0.5$ m, $x \in [0, 1.0]$m through the domain at $t = 0.025$ sec using the standard and compact stencil schemes are illustrated in Figure 2.16 and Figure 2.17 respectively, and the calculated value of T at the zone boundary compared against the analytical solution for accuracy.

2.9.1.2 Results - Test 1

The inherent disadvantage of standard edge-based finite volume schemes due to the 'leapfrog' manner in which derivatives are calculated using the five-node stencil has already been discussed. This impediment is obvious when the results shown in Figure 2.16 are considered. The large spatial oscillations emanating from the discontinuous material boundaries in the non-homogeneous domain are a direct result of the de-coupling of the solution when implementing the standard edge-based scheme. However, the solution obtained using the compact stencil formulation (Figure 2.17) contains no spatial oscillations, and agrees well with the analytical solution. Hence, the compact stencil formulation is consistent for inhomogeneous isotropic domains, and should be used wherever possible for increased accuracy across the material boundary.

The problem discussed in this section (Test 1 –Vertical Zones) was re-run with different boundary conditions applied to boundary B-C. The condition now applied to this boundary was

$$\alpha_x \frac{\partial T(x,y,t)}{\partial x} = 10.0 \qquad \text{on side B-C}. \tag{2.48}$$

All other parameters remained the same as above. The results obtained are shown in Figure 2.18, where plots of the calculated parameter T along the line $y = 0.5$ m, $x \in [0,1.0]$m through the domain at t = 0.01, 0.02, 0.03, 0.05, 0.07, 0.1, 0.2 and 0.4 seconds are illustrated.

It can be seen that the solution curve obtained by plotting along the line $y = 0.5$ on the domain at various time steps from 0.01 seconds to 0.4 seconds, when a steady state solution has been reached. At t = 0.4 seconds a steady state solution is reached, and it is noted that the plot for this time has a gradient of $\alpha_x \, \partial T/\partial x$, which is exactly the analytical solution for this case.

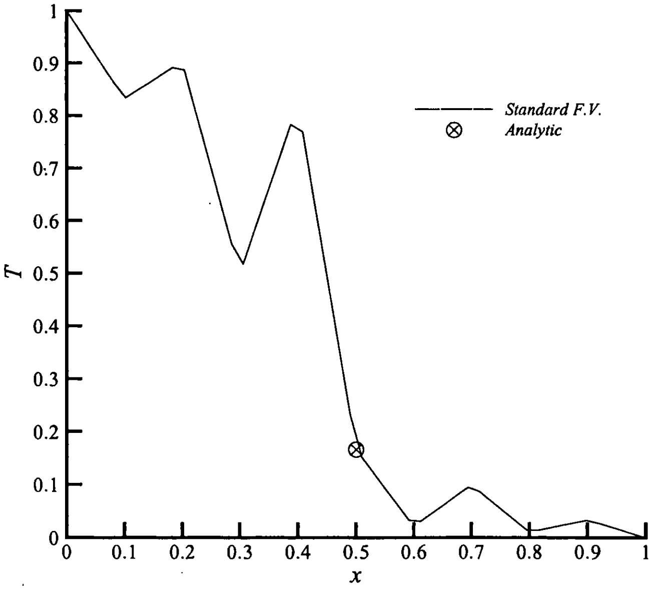


Figure 2.16: A profile of the calculated parameter T along the line $y = 0.5$ m, $x \in [0,1.0]$m through the domain at t = 0.025 sec. using the standard edge-based FV scheme.

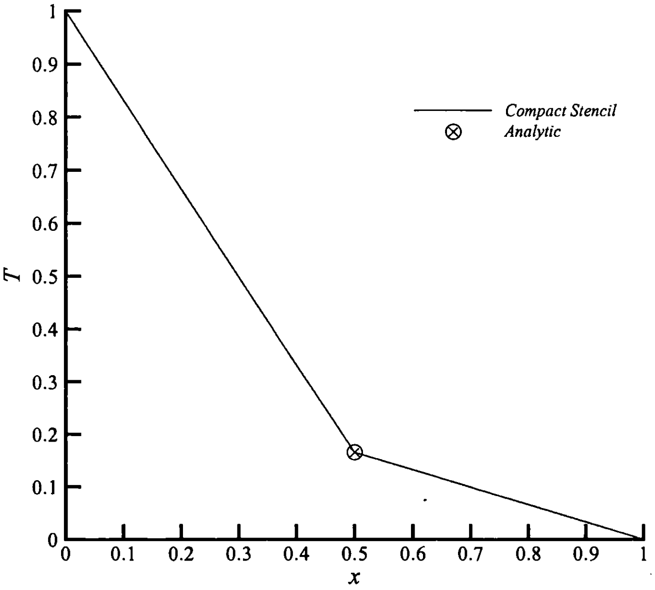


Figure 2.17: A profile of the calculated parameter T along the line $y = 0.5$ m, $x \in [0,1.0]$m through the domain at $t = 0.025$ sec. using the compact stencil edge-based FV scheme.

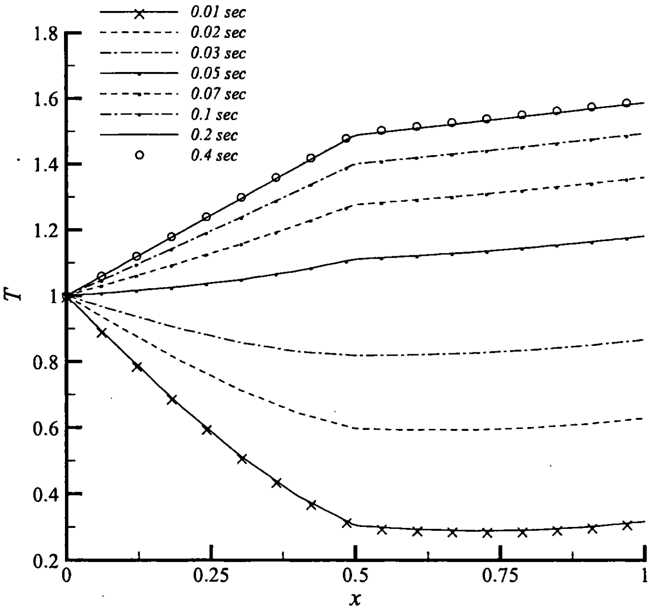


Figure 2.18: Plots of the calculated parameter T along the line $y = 0.5$ m, $x \in [0,1.0]$m through the domain at $t =$ 0.01, 0.02, 0.03, 0.05, 0.07, 0.1, 0.2 and 0.4 sec. using the compact stencil edge-based FV scheme.

2.9.2 Test 2 – Narrow Zone of Preferential Flow

This test was conducted to simulate flow through a narrow zone of preferential flow, and the same domain as shown in Figure 2.14a was used. The domain was partitioned into two zones, with a thin diagonal band (Zone 2) used to represent the narrow zone of preferential flow as shown in Figure 2.19a. The domain was discretised using an unstructured grid of 1606 triangular elements (see Figure 2.19b).

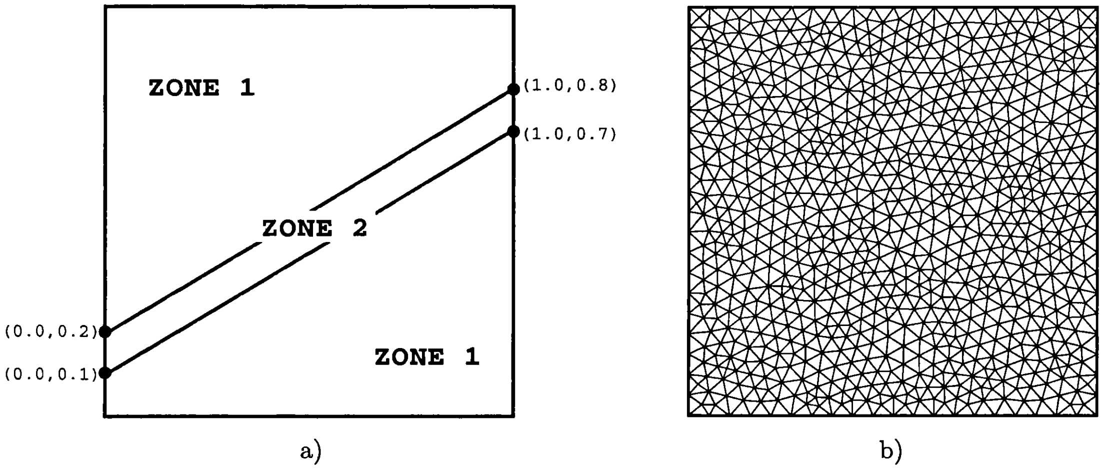


Figure 2.19: Illustration of a) the domain with simulated fracture, and b) the unstructured grid used for the non-homogeneous problem Test 2.

Although the domain is inhomogeneous, each zone is isotropic and was prescribed a different set of dimensionless parameter values as shown in Table 2.2 below.

	Zone 1	*Zone 2*
α_x	1.0	10.0
α_y	1.0	10.0

Table 2.2: Values for the dimensionless parameter α applied to the non-homogeneous domain in Test 2.

Note that the narrow zone of preferential flow has α values ten times greater than the remainder of the domain. The initial and boundary conditions used for this test case were;

$$T(x,y,0) = 0.0 \tag{2.49}$$

$$T(x,y,t) = 1.0 \qquad \text{on side A-D} \tag{2.50}$$

$$T(x,y,t) = 0.0 \qquad \text{on side B-C} \tag{2.51}$$

$$\partial T\big/\partial y = 0.0 \qquad \text{on sides A-B \& D-C} \tag{2.52}$$

Contours of the calculated parameter T at various times are shown in Figure 2.20. As expected, the front movement is accelerated in the path of preferential flow; that is through Zone 2.

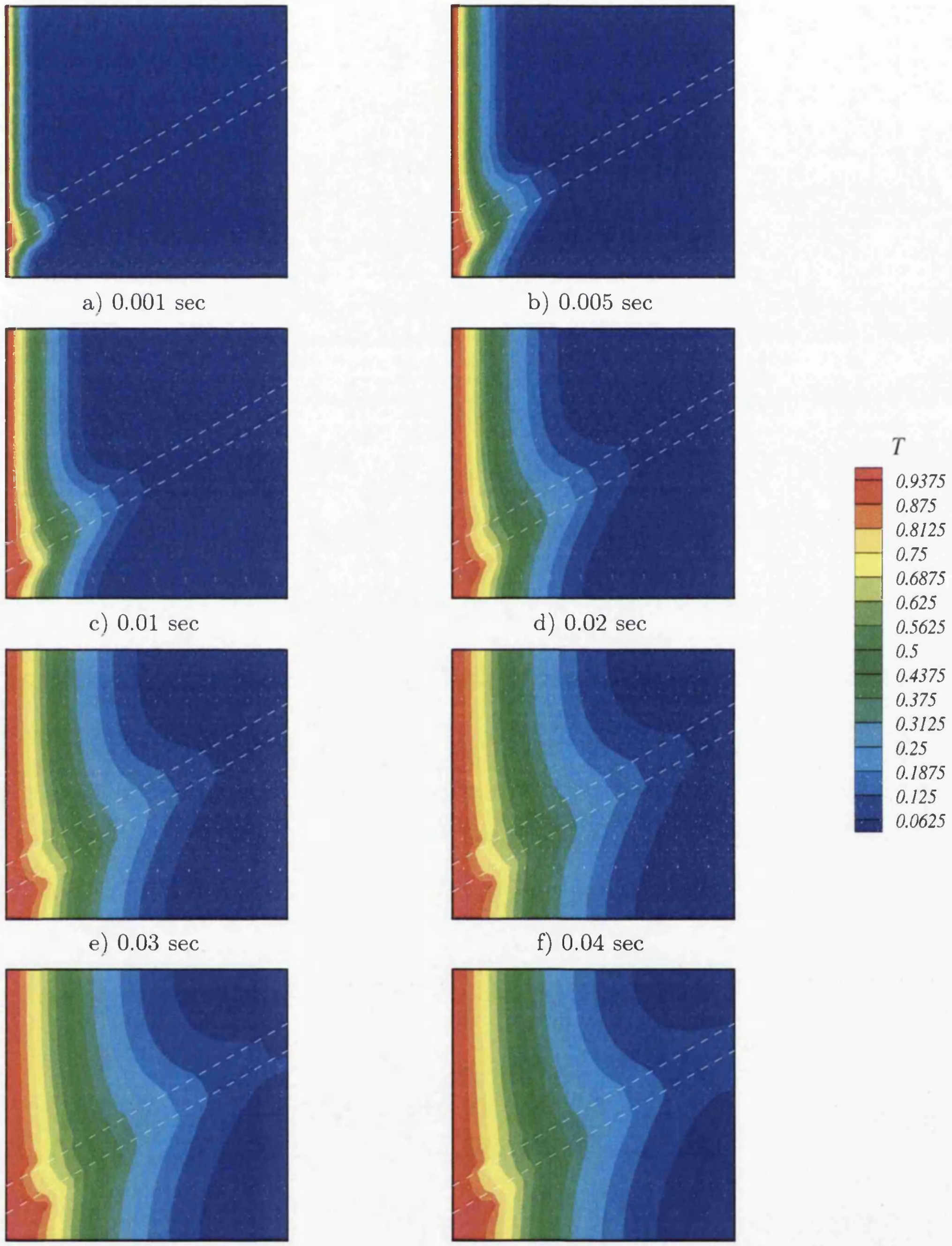
Hence, it can be concluded that the compact stencil scheme, unlike the standard finite volume scheme, successfully calculates second derivative terms across material boundaries.

2.10 CLOSURE

In this chapter the spatial discretization aspect of the work was discussed with the chosen scheme being the finite volume vertex-centred dual-cell method. Further, an edge-based solution procedure was employed which offers significant performance benefits as compared to element-based methods. As a benchmarking exercise, the standard derivative routines were utilised to solve a linear case of the two-dimensional heat conduction equation, and a close correlation between the numerical and analytical solution for this problem was achieved.

Enhancements to the chosen scheme involved the use of a compact stencil algorithm for the discretization of diffusive second derivative terms. The increased accuracy of the compact scheme over the standard scheme was demonstrated by the calculation of second derivatives of a harmonic function on structured and unstructured grids. When calculating second derivative terms of a harmonic function on unstructured grids, it was shown that the solution obtained using the compact stencil scheme provided a marked improvement over standard finite volume scheme. Moreover, it was shown that there was little difference between the solutions obtained using the compact scheme on structured and unstructured grids with an equal boundary node density.

Finally, the numerical solution to simple diffusion problems on two different inhomogeneous domains using the chosen scheme was considered. It is believed that this is the first documented case of the application of the edge-based finite volume scheme on problems over inhomogeneous domains. It is shown that the application of the standard scheme results in large spatial oscillations emanating from the discontinuous material boundaries within the domain. This problem was overcome by the utilisation of the compact stencil, and accurate solutions to the test cases were presented. Hence, the compact stencil formulation is consistent for inhomogeneous domains, and should be used wherever possible for increased accuracy across the material boundary.



a) 0.001 sec
b) 0.005 sec
c) 0.01 sec
d) 0.02 sec
e) 0.03 sec
f) 0.04 sec
g) 0.045 sec
h) 0.05 sec

Figure 2.20: Contours of the calculated parameter T at $t = 0.001$, 0.005, 0.01, 0.02, 0.03, 0.04, 0.045 and 0.05 sec. for Test 2.

REFERENCES

[1] Aziz K. and Settari A., Petroleum Reservoir Simulation. Elsevier, 1979.

[2] Bear J., Hydraulics of Groundwater (McGraw-Hill Series in Water Resources and Environmental Engineering), McGraw Hill College Div; 1979.

[3] Chen A.J. and Kallinderis Y., Adaptive hybrid (prismatic-tetrahedral) grids for incompressible, *International Journal for Numerical Methods in Fluids*, **26**:1085-1105, 1998.

[4] Crumpton P.I., Moinier P. and Giles M.B., An Unstructured Algorithm for High Reynolds Number Flows on Highly Stretched Meshes, in: C. Taylor, J. T. Cross, ed., Numerical Methods in Laminar and Turbulent Flow (Pineridge Press, 1997) 561-572.

[5] de Foy B. and Dawes W., Unstructured pressure-correction solver based on a consistent discretization of the Poisson equation, *International Journal for Numerical Methods in Fluids*, **34**:463-478, 2000.

[6] Fletcher C.A.J., Computational Techniques for Fluid Dynamics, volume I. Springer-Verlag, second edition, 1991.

[7] Lewis R.W., Huang H.C., Usmani A.S. and Cross J.T., Finite element analysis of heat-transfer and problems using adaptive remeshing including application to solidification problems, *International Journal for Numerical Methods in Engineering*, **32**(4):767-781, 1991.

[8] Luo H., Baum J.D. and Lohner R., Edge-based finite-element scheme for the Euler equations, *AIAA*, **32**(6):1183-1190, 1994.

[9] Malan A.G., Investigation Into The Continuum Thermodynamic Modelling Of Investment Casting Shell-Mould Drying, PhD Thesis, School of Engineering, University of Wales Swansea, 2002.

[10] Malan A.G., Lewis R.W. and Nithiarasu P., An improved unsteady, unstructured, artificial compressibility, finite volume scheme for viscous incompressible flows: Part I. Theory and implementation, *International Journal for Numerical Methods in Engineering*, **54**(5):695-714, 2002.

[11] Mills A.F., Heat Transfer, Prentice Hall, 1998. TJ260.M52

[12] Nithiarasu P. and Zienkiewicz O.C., Adaptive mesh generation for mechanics problems. *International Journal for Numerical Methods in Engineering*, **47**(1-3): 629-662, 2000.

[13] Onate E., Cervera M. and Zienkiewicz O.C., A finite volume format for structural problems, *International Journal for Numerical Methods in Engineering*, **37**(2):181-201, 1994.

[14] Rees I., Lewis R.W., Masters I. and Malan A.G., Material Variation Using A Compact Stencil Vertex-Centred Edge-Based Finite Volume Scheme, In proceedings: M.A. Wheel,

ed., The 11th Annual Conference of the Association for Computational Mechanics in Engineering, Glasgow, 5-9, 2003.

[15] Sorensen K.A., A Multigrid Acceleration Procedure for the Solution of Compressible Fluid Flows on Unstructured Hybrid Meshes. PhD Thesis, School of Engineering, University of Wales Swansea, 2001.

[16] Sorensen K.A., Hassan O., Morgan K. and Weatherill N.P., Agglomerated multigrid on hybrid unstructured meshes for compressible flow. *International Journal For Numerical Methods In Fluids*, **40**(3-4): 593-603, 2002.

[17] Swanson R.C. and Turkel E., Multistage schemes with multigrid for Euler and Navier-Stokes equations. NASA Technical Paper 3631, 1997.

[18] Turner I.W. and Ferguson W.J., An unstructured mesh cell-centered control volume method for simulating heat and mass transfer in porous media: Application to softwood drying, Part I: The isotropic model. *Applied Mathematical Modeling*, **19**:654-667, 1995.

[19] Vahdati M., Morgan K., Peraire J. and Hassan O., A cell-vertex upwind unstructured grid solution procedure for high-speed compressible viscous flow. In proceedings International Conference on Hypersonic Aerodynamics, pages 12.1-12.22, London, 1989. Royal Aeronautical Society.

Chapter 3

ADVANCED PROGRAMMING METHODS AND DATA STRUCTURES FOR COMPUTATIONAL MODELLING

3.1 INTRODUCTION

This chapter describes the author's efforts in producing a new type of computer program for the solution of numerical models. This work is based upon an earlier study of the use of an object-oriented language, C++, for the implementation of a finite element model [13]. An alternative title for this chapter would be 'Beyond Object-Oriented Finite Elements' as one has carefully considered all aspects of the code, including the programming style, the data structure and the underlying mathematics of the numerical scheme. In all of these areas, improvements have been made over and above the object-oriented approach reported in earlier work by Masters *et al.* [13], Lewis *et al.* [10], and Cross *et al.* [2]. The generation of a numerical model must consider the separate issues of the governing equations, the numerical representation of those equations, the data structure that describes the model, the choice of programming language and finally the implementation and code management issues. A discussion regarding the computational speed and optimisation issues relating to the main programming languages is presented. Then the optimal data structures for the most common numerical schemes are proposed, with the data structure for the implemented scheme described in detail. These issues are considered as a whole in this chapter, and as a consequence ten golden rules for numerical modelling are proposed. The implementation of an explicit edge-based scheme using a novel data structure is introduced, and this is shown to have over 91% code re-use. Hence, code written with the architecture described here is highly flexible and applicable to many different problems.

When the finite element method was developed in the 1960's computing facilities were very different to those available today. Programming languages were still in their infancy and the compilers available did not necessarily produce fast code. The scientific community was interested in using the computer as a tool, using simple programs to obtain the results they required. The programming language had to have instructions that were simple to use, suited to the programming of relatively short codes and that had an efficient compiler to make the most of the limited computer hardware. Given these criteria, there was only one programming option available, FORTRAN, and implementation of numerical schemes in this language was described in many books and papers. The textbooks that came to be regarded as key works, such as Hinton and Owen [7] or Smith and Griffiths [18] are still used by many people and the program structures used in those example programs forms the underlying style of many finite element codes.

Most code developers have taken advantage of new techniques for reducing CPU time, including vector and parallel computing and the use of advanced matrix vector routines. One example is the very good performance in both speed and memory use available through preconditioned iterative matrix solvers [3]. However, one area that has seen little change is the underlying structure and style of programs. The reluctance to change follows the very good argument that "if it ain't broke, don't fix it", and hence a very stable programming environment results. However, this argument breaks down with the sheer size of finite element codes; current research codes are large, complex pieces of software designed to deal with a physically accurate, highly non-linear engineering problems. Large software has code maintenance issues and therefore can be difficult to change, especially where a team of programmers are involved. For this reason, several researchers have turned to more structured programming languages, for example, FORTRAN 95 or C. C++ has also been used where code reuse and protection of data is required.

C++ is a good programming language that forces the user into good practice. Code produced this way is very compact, easy to read and the data is hidden. There is a high level of flexibility in the code and sections of code can easily be re-used in future development. Previous work by Cross *et al.* [2] and

Masters *et al.* [12] reports progress made in this area. It is sufficient to say that object-oriented methods have been used to generate numerical code for a wide range of application areas including stress analysis ([5],[4]), hypersonic shock waves [1], structural dynamics [16], shell structures [15], plasticity [14], electromagnetics [17] and contact problems [6]. However, there are some disadvantages of the object-oriented approach, which are covered in more detail in the data structures section below.

This chapter sets out to deal with the conflict between development time, structured data and CPU time. Can we still get the code re-use of C++ with the computational speed of raw FORTRAN or C operations based on pointers[1]? Can our data structures be as clear and understandable as C, without the hidden data of C++ and the unnecessary complexity of a legacy FORTRAN code?

In the next section a more detailed look is taken at each of the key areas of numerical programming: the data structures employed, computational speed and the numerical schemes. The author's own strategy designed to overcome these problems is then presented. Finally, conclusions are drawn from the work.

3.2 DATA STRUCTURES

A data structure is the way in which information is stored within the model. A good data structure is quick to access when running and has meaningful variable names that relate easily to each other. In this section we discuss several data structures and the way in which they are used within a numerical model.

One of the problems with numerical modelling is where a code is still in use but has been around for a long time. This 'legacy code' contains some very powerful modelling capability, but modifying it for a new application is a daunting project. As an illustrative example, one code that previous researchers within the group have worked on has been derived from the PLASCON code

---

[1] For those unfamiliar with C, pointer based operations are where the loop variable is the actual location in memory of the data and compilers can turn this into fast arithmetic. FORTRAN compiliers with the `-Ofast` flag can achieve the same arithmetic speed in the majority of cases.

supplied with the first edition of Lewis and Schrefler [9] published in 1987. Fragments of this code are even earlier and can be traced back to the late seventies. Over the last 15 years, this code has had at least five programmers working on it and has grown to over 8000 lines of FORTRAN and a further 3000 lines of library routines from three external sources. Over that period there have been additions to the data structure but no fundamental re-write. For example, the code required to form the Jacobian matrix for an iso-parametric finite element is one of the oldest sections of this program. The data structure that holds the input data for that calculation (the nodal co-ordinates and the element connectivity data) is therefore over 25 years old. It would be impossible to modify the way in which this data was stored and used without unseen implications in other areas of the program. Part of this organically grown data structure is shown in Figure 3.1 below.

A programming language that allows structured data goes some way to removing some of the issues of code flexibility and clarity. A simple example of this is storing the dimensions of a matrix alongside the data within the same data structure. In this instance C is used, but an identical structure is possible in FORTRAN 95.

```
struct matrix { int rows; int columns; double *values; };
struct matrix a;
```

This sets up a matrix called `a` of size `rows` by `columns` containing an array of `values`. The size of the data will be set up later in the code. This reduces dramatically the number of individual items of data and is now common practice for numerical programmers using C or FORTRAN 95. Another way of looking at this concept is that the type of data called `matrix` contains three other types of data and those types of data could also contain other data in a nested fashion.

```
  COMMON   /INPOUT/  IIN,    IOUT, IOUT1, IOUT2, IOUT3, IOUT4
  COMMON   /SPACE1/  NB,  NSK,  BUF,  KSIZE, KSIZE1,KSIZE2,NRHS,
 1                   ND, NR,   NW,    NSFR,  NGNG, NARRAY,NSMITP,
 2                   NSMITT,NINSCH,NTPLOT,NNTV,IPRINT
  COMMON   /SPACE2/  NRAD,  NPSR,  NTET, NPST,R1W, IELEM, INODE,
 1                   IELMAT,NROW, NLAYER,  NCOL,  ISTOP, ISTORE
  COMMON   /CONTR1/  ISTR,  ISTRES,IPSTR, ICS1, NSTEP, INDP,
 1                   MAT,   IVELOC,JIG,   IPP,   LPL,   ISTR1, IN,
 2                   IDISP, IBOUND,INIT,  ILL, NS1, ICALC, ISYMM,
 3                   NCOUNT,ISTEP, IBO, ITER,  NBAND, NEL,   IGJG,
  COMMON   /PRINT/   IWRIT,IWRITT, INTAP, IOUTAP
  COMMON   /OLDPRSS/ OLDOPR(NUEL,8),OLDWPR(NUEL,8),OLDGPR(NUEL,8)
  COMMON   /DIM/     N1, N2, N3, N4, N5, N6, N7, N8, N9, N10, N11,
 2   N15,   N16, N17, N18, N19, N20, N21, N22, N23,   N24,   N25,
 3   N26, N27, N28, N29, N30, N31, N32, N33, N34, N35,
 5   N36,   N37,   N38,   N39,   N40,   N41,   N42
  COMMON   /VAR/     TIMA,TIN, TFRAC, PLATK,DENS, ELHFLW,ELFLOW
  COMMON /PVTQX/       TKRW,TKRO,FVFW,FVFO,SPCG,OFVF(NUEL,8)
 1 ,SFO,FVFOO(NUEL,8),TKW,TKROG,VFWO(NUEL,8),SFVFW,OFVFW(NUEL,8)
  COMMON /COMLTY/    VCOG,COMPO,VCOO,VCOW,DVCOW
  COMMON /DINIT/ PREOI(NUEL),PREWI(NUEL),PREGI(NUEL),SOM(NUEL),
 1  FVFOI,RSOI,PCWOI,PCGOI,PCONV,TKRGI,TKROI,TKROGI,TKROWI,
 2  TKRWI,FVFWI,TKROCW,PCWIN(NUNO),PCGIN(NUNO),FACTW,FACTO,FACTG
  COMMON /CMBE/     CPOM(8,8),CODO(8,8),COSO(8,24),TOLO(8,8)
 1              ,GMBE(8,8),OLDOP(NUNO)  ,TOM(8,8)
  COMMON /PHASE/     COSC(8,24),SATRSW,SWOLD(NUEL,8),SWI,SGI,SWCO
  COMMON /ZON /      JMODEL,IPVT,IZONE(NUEL),INDPT(NUNO)
  COMMON   /GAS/     SATRGO(NUEL,8),PCGOO(NUEL,8),
 1                   SPCGAS,PCGO,SATRG,FVFGO(NUEL,8),TKRG,
 2  OFVFG(NUEL,8),SFVFS,PGAS,  FVFG,RSOO(NUEL,8),SRSO,RSN(NUEL,8),
 3  TOWCUT(120,30),PGOR(120,30)
  COMMON   /COGAS/ GDE(24,8),GWE(8,8),GOE(8,8),GG1(8,8),GG2(8,8),
 1                 GG3(8,8),GG4(8,8),GG5(8,24),GG6(8,8),GG0(8,8)
  DIMENSION      GASHTD(MDF1,MNTV),  NNT(MNTV)
  DIMENSION      NFIX(MP),    CORD(MP,NDIM),NOP(ME,8),  IMAT(ME),
 1 NBC(MP),R1(MRHS),  BIT4Q(MP),STMOV(JN),NSTEPT(NSTPP)
  DIMENSION   GASHT(IGS,MDF1,MP),IONARY(MP),NODEL(NSIZE2)
  DIMENSION   A(MTOT),  BB(MTOTB)
  DIMENSION   NLT(NLO),TFACEL(NSTEP2),  TFAC(NSTEP1,NLO1)
  DIMENSION   PLK(NSIZE1,NSIZE1),COC(NSIZE1,NSIZE2),
 3 LH(NSIZE2,NSIZE2),   FLG(NSIZE2,NSIZE2),  QLAD(NSIZE2),
 5 SLOAD(NSIZE1),  STLOAD(NSIZE1),  QHLOD(NSIZE2),
 8 COM(NSIZE2,NSIZE1),  COTU(NSIZE1,NSIZE2),  TLH(NSIZE2,NSIZE2),
 B TLL(NSIZE2,NSIZE2),  TLG(NSIZE2,NSIZE2),  TSL(NSIZE2,NSIZE2)
```

Figure 3.1: FORTRAN 77 legacy data structure example.

A global data structure for a finite element model using this approach could be structured in the following way:

Mesh
Contains
Elements
Contains
Gauss points
Nodes
Contains
Co-ordintates
Data

In order to understand the issues of an object-oriented approach, the first stage is to clarify how the data structure described above could be extended to a set of objects. The fundamental difference between a structured programming language and an object-oriented one is the ability to attach functions to a structure in the same way as data. This significantly improves the flexibility of the code. For example, if our C matrix data structure given above was extended to C++ and a function called `determinant()` was attached to that data structure then the simple line

```
D= a.determinant();
```

would return the determinant of `a`. However, this approach becomes very powerful if more than one data structure has the same functions attached. If we now define a banded matrix structure

```
struct bandedMatrix { … ; double determinant(); };
```

and define `a` as this type,

```
struct bandedMatrix a;
```

then `d=a.determinant();` still works and the code requires no changes.

The overall concept behind object-oriented programming is that objects ask each other for data and do not know (and do not need to know) how that data is stored or manipulated. An example would be making a telephone call. By dialling the number you ask the exchange to connect you to another telephone. You do not want to know about cables and data transmission, but are interested in the results of the application of that technology. The process is hidden behind a wall and communication through the wall is carefully controlled.

We now return to numerical computation. The two fundamental stages of a finite element operation are: 1.) generation of a global matrix, and 2.) solution of a global matrix. This is where there is a conceptual problem with the mesh data structure above. If we wish to have distinct, separate data structures for the mesh and the matrix then the mesh object asks each element object for a local stiffness matrix $\mathbf{K}^e$, which is then passed over the wall to the matrix object to put into the correct locations in the global matrix. In theory, a fully object based system has a range of mesh objects for different physical problems, for example, stress analysis, groundwater flow, heat transfer, etc. These mesh objects communicate with a range of matrix objects, which have different compact storage methods such as banded, skyline, or sparse. To keep the strict data structure then some very disciplined programming is required. The same discipline is also required for the solution stage, the implementation of a matrix inverse routine that works with various matrix formats is fairly time consuming. Finally, the result for that step is handed back over the wall to the mesh object and distributed back to the nodes.

A rational evaluation of this whole process shows that the disciplined finite element programmer, who is writing a highly structured and flexible finite element system in an object-oriented style, spends most of the time rearranging data from one form to another (element to mesh to matrix). After lengthy discussions with other software developers, it is the authors' conclusion that the programmer is never rigorous enough in testing to trust the system completely. Therefore, when a change is made to the model (perhaps an extra term in the system equation) the program still works, but because the point where the change is made (a function associated with an element) is so far removed from the point at which it makes a difference (during the inversion of the global matrix) then debugging is very difficult. Consequently, the issues of the data structure are such that organising the structure takes up the developers thinking and time, rather than dealing with issues related directly to the underlying numerical model.

One will discuss later how the choice of numerical scheme influences the complexity of the data structure. However, for all numerical models in all languages, a strong recommendation is to split pre-processing from the main solver routines into a separate program. The pre-processor takes mesh data together with initial and boundary conditions and forms all of the help arrays needed for solution. In the finite element context, for example, these would include the gauss point co-ordinates, shape function derivatives, and other data required for each element. This data should be output to files, thus allowing external checking before being read into the main solver program.

3.3 COMPUTATIONAL SPEED

The issue of computational speed for numerical models dominates discussion at all levels. It is generally accepted that there will never be enough computing power and that models continue to grow in size. However, the reality is that a modest desktop PC has far higher performance than yesterday's mainframe machines and a realistic solution of many problems can be achieved in a matter of hours if not minutes. Much has been written about loop unrolling, use of temporary variables, reduction of floating-point calculations and removing `if` statements from loops, and one does not intend to repeat that here. Rather, we consider two important questions: 1.) Which programming language gives the fastest compiled code? 2.) On balance, how much effort should go into optimisation of code for speedup?

3.3.1 Choice of Programming Language and the Resulting Computational Speed

If we limit our discussion to FORTRAN, C and C++ then each of these languages can be used to produce fast code. FORTRAN compilers are excellent, the data structures are simple and fast code often results from using appropriate compiler options. Manipulation of memory is left to the compiler and in most cases the fastest code results. In C it is possible to carry out pointer-based arithmetic and it is guaranteed that this is the fastest code (the same as FORTRAN). C programmers tend to use a larger number of short functions than FORTRAN programmers so this adds an overhead, but if the number of floating point operations is 'large' compared to the number of function calls then this is negligible.

One observation of code speed comparison is that a problem needs to be 'large' compared to both the speed of the processor and the size of cache for the comparison to be valid. Under these conditions, perceived differences between languages rapidly disappear.

In C++, one of the most powerful functions is dynamic data using templates, where the type of data is not specified. For example, it is perfectly possible to write a template for a matrix object that would carry out operations on float, double and complex data without requiring code changes and the code is then very compact and flexible. However, the type of data is determined at run time (not compile time) and the computational overhead is huge. If this dynamic type allocation is removed then it can be proved [11] that operations are much faster and equivalent to FORTRAN.

Therefore, recognising that optimisation for small problems is not really necessary, the choice of programming language does not significantly affect runtime. If the problem to be solved is sufficiently large and if dynamic type allocation is avoided, then there is no real performance difference between these three languages.

3.3.2 Optimisation of Code for Speedup

Before optimisation for speed is attempted, code profiling is normally carried out. For most numerical schemes, it is found that the majority of computational effort is included in the matrix operations. Matrix/vector libraries such as BLAS are available for most programming languages and optimised compiled versions are often included with Unix systems. Provided that matrix operations are optimised, the developer's efforts in speeding up the rest of the code will only produce small improvements in speed and should not be given a high priority.

The computational resources available grow daily. In a paper published in 1997, Masters *et al.* [12] reported on efforts to build the first generation of sustained Teraflop computer. The current Top 500 [8] computer has a speed of 36 Teraflops and 58 machines on this list have over a Teraflop capability. This shows that the processing capability at the top end has doubled five times in the last six years. On this basis there is a valid argument not to carry out any speed optimisation at all and to simply replace hardware every year to halve the computation time.

In conclusion, provided that matrix operations are optimised, the choice of data structure and numerical scheme has more influence on runtime than the choice of programming language and the developer's efforts in optimisation.

3.4 NUMERICAL SCHEMES

Most numerical programmers have a particular physical problem to solve before they start writing a code. In the planning stage, it is best to consider all schemes available and to use the simplest system that is capable of solving the target problem. Table 3.1 below gives the most common numerical schemes and suggests the most efficient way of dealing with the data in each case. Knowledge of the data structure required should inform the choice of programming language used to implement the scheme.

Scheme	*Approximation*	*Best Data Structure*
Finite Element	Mesh of elements and nodes	Multi-level structured data
Finite Difference	Grid of difference points	Matrix
Meshless (fixed nodes)	Unconnected nodes	Tree structure and node list
Meshless (moving nodes)	Unconnected points interacting locally	Dynamically changing Tree structure
Finite Volume (edge based)	Edges	List and Matrix

Table 3.1: Data structures required for various numerical methods.

Obviously, not all schemes are applicable to all problems and there are many variations on the above. The appropriateness of a scheme should always be considered before a final decision is made.

Ten golden rules of developing a numerical code follow from this discussion, and are listed in Figure 3.2.

1. The data structure used in the program should match the data structure of the numerical scheme employed.
2. The management of data and data structures should not dominate the development of a program.
3. Code should be re-used wherever possible.
4. Pre-processing of data should be separated from the main solver routines, preferably into separate programs that communicate via data files.
5. Mesh generation and visualisation should be taken care of by readily available software.
6. Good comments and in-code documentation is essential.
7. Make good use of library routines, especially for matrix/vector operations and matrix solution.
8. A good programmer can write, on average, 20 lines of validated, documented code in a day.
9. The time take to validate code is proportional to the square of the number of lines to be checked.
10. Reducing development time is more important than reducing CPU time.

Figure 3.2: Ten golden rules for the successful development of a numerical code.

3.5 A STRATEGY FOR CODE DEVELOPMENT

It is beneficial to establish exactly what numerical programming is attempting to do. The underlying motivation for modelling is to interpret/understand some form of physical problem. The first approximation is that the physics is interpreted as a set of equations, normally in differential form. This is then approximated again into a pointwise (discretised) approximation of the integral over the domain via the application of some local trial solutions (e.g. shape functions). The primary variables follow from the minimisation of that integral.

The first stage in our proposal for a numerical scheme is to remove part of this complication. Consider a set of parameters $\underset{\sim}{P}$ in a domain Ω that are approximated by a set of discrete point values $\hat{\underset{\sim}{P}}$, such that,

$$\hat{\underset{\sim}{P}}(x,y) \approx \underset{\sim}{P}(x,y) \quad , \quad \forall x,y \in \Omega \tag{3.1}$$

The pointwise values are stored in a matrix $\mathbf{P}$ such that,

$$\hat{\underset{\sim}{P}}(x,y) \subset \mathbf{P} \quad , \quad \forall x,y \in \Omega \tag{3.2}$$

Observing that for a set of discrete points i, the point co-ordinates $(x_i,\ y_i)$ are of an equivalent structure (length) to the primary variables. Hence, they can also be stored within the matrix $\mathbf{P}$.

$$x_i, y_i \subset \hat{\underset{\sim}{P}} \tag{3.3}$$

If ψ is one of our variables, then two further variables can be formed, namely,

$$\frac{\partial\hat{\psi}}{\partial x} \approx \frac{\widehat{\partial\psi_i}}{\partial x_i} \subset \hat{\underset{\sim}{P}} \quad , \quad \theta_i = \theta\left(\psi_i\right) \subset \hat{\underset{\sim}{P}} \tag{3.4}$$

Therefore, all node-based variables, derivatives of variables and non-linear relationships can be stored within $\mathbf{P}$.

In order to implement this strategy, all that is required is a simple data structure (a matrix) and a set of numerical processes to find the dependent variables $(\theta(\psi_i))$ and the derivatives. With the ability to find all of these relationships, it should be possible to solve any differential equation by performing operations on variables within $\mathbf{P}$ to form new variables to add to $\mathbf{P}$. A similar arrangement is constructed for edge-based data storage. Hence, all data within the solver is stored inside two large matrices of length `NNODES` and `NEDGES` respectively; where `NNODES` is the number of nodes in the domain, and `NEDGES` is the number of edges in the domain.

The second stage of our strategy is the numerical scheme chosen to form the derivatives. The elegant and robust vertex-centred edge-based finite volume approximation, discussed at length in chapter 2, is implemented. This is a relatively short procedure to implement and works with two simple data structures; a list of edges and a discretised field variable. The data associated with an edge is the node numbers for the two ends and the edge coefficient in each co-ordinate direction.

The third stage of the strategy is the use of an explicit formulation. All operations can then be undertaken quickly on a local level. Although the number of time steps in an analysis may be larger than an implicit method, memory use is orders of magnitude smaller and the code will make good use of cache and vector/parallel hardware if available.

From this conceptually simple approach, implementation is easy and the developer can concentrate on the differential equation that describes the underlying continuous function, rather than implementation issues.

3.6 IMPLEMENTATION OF PROGRAMMING STRATEGY

In this section, we describe the implementation of a computer program that uses the advanced programming methods and data structures proposed so far in this chapter to solve problems concerning the flow of fluids through a porous medium. If the resulting program is clear, reliable and flexible with lots of code reuse, then it can be concluded that the method is a success. This section describes the implementation choices and describes the final structure of the code.

The first decision required is the choice of a programming language. At the start of the project a number of library routines were available. C++ was considered as there were matrix operation routines, iterative solvers and various checking and I/O utilities in existence. However, the simple matrix structure of the data does not require the complexity of C++ and with the overall philosophy of simplicity this is not the best option. In FORTRAN, there were available routines for matrix operations, iterative solvers, curve lookup routines and interaction with TecPlot [19]. TecPlot is a visualisation package, which was used in this project to generate meshes and to interpret results. The decision was taken to use a FORTRAN 95 compiler, however, most coding used a more traditional FORTRAN 77 style for clarity of the code. Data management is avoided by the use of a header file. Each subroutine has an `include "header.fi"` statement where all data is managed. This is actually quite a simple structure due to the fact that most data is contained within the single matrix **P**. By implication, this means that data is stored in a common format, but by careful use of parameter statements to dimension the common block, this is not a real issue. The structure of the code is as follows:

1. The TecPlot program is used interactively to generate a mesh file.
2. A small data file is written to describe initial and boundary conditions attached to the mesh
3. A pre-processor program reads the files, checks them, breaks the mesh into edges and generates a simple file format containing only the data necessary to complete a solution.
4. A second small data file is written that contains the solver parameters such as time step length, convergence criteria, etc.
5. The solver reads in the pre-processor file and generates output.
6. TecPlot is used for visualisation.

The pre-processor has three main components, which are stored in separate FORTRAN files for clarity. A header file (127 lines) as discussed earlier is used to set out all of the data structures required. The utilities (888 lines) are used for file I/O and the checking of input data. The main routine (1678 lines) generates the data required for solution, in particular the edge coefficients, the boundary coefficients, the control volume sizes and the material property data attached to each edge. The advantage of this pre-processor is that it can operate on any number of nodes per element and is completely independent of the governing equations. Therefore, the 2693 lines of this code should not require modification at all in the future.

The main program has 5309 lines in total (excluding the BLAS library, but including comment lines) and is split up as follows:

- A header file (300 lines) used in the same way as described above.
- A main program (745 lines). This contains a number of routines that implement the numerical algorithm, including the main time step loop. All other routines are called from here.
- Utilities (2278 lines) for file I/O, initialisation of data and the implementation of a restart function. File interaction is a laborious process in any language and requires lots of lines. To ensure clarity a large number of comments are also required, hence this section is the longest in the program.
- Boundary condition routines (119 lines). A self-contained section, so hence a self-contained file.
- Numerical error checking (360 lines). Mass and energy balance calculations can be carried out on the solution to check the numerical accuracy.
- Derivative routines (404 lines). Due to the large amount of effort in the pre-processing and data structures, the most important routines, where most of the computational effort takes place, are actually quite short.
- Curve handling routines (934 lines) together with a header file (28 lines). These self-contained curve routines contain their own I/O and error

checking. They are mainly used to generate non-linear variables $\theta\left(\psi_i\right)$ within the code. To enable these routines to be integrated with the main code, a set of wrapper routines (141 lines) are used to convert data from the program structure to the utilities structure. No data manipulation is carried out in the wrapper; the objective is to achieve clear code at the expense of one additional function call.

This is quite a large implementation and at first glance does not appear to be well written. However, to implement a completely different governing equation would only require a rewrite to the main routine. There may be some minor changes in the output routines but the only significant change is to the algorithm, which is concentrated in one place. Therefore out of 8002 lines of code, only 745 need to be considered, or alternatively, 91% of the code is reusable without modification.

3.7 CLOSURE

This chapter has shown that the implementation of a numerical method requires thought and should not be seen as an inconsequential part of the process of research into numerical modelling techniques. If chosen with care, and some ground rules are carefully considered, the implementation will aid rather than hinder the development.

A discussion of data structures showed that data should be as simple as possible and only as complex as is necessary. Getting the data right is much more important than an over-emphasis on the computational speed. However, for the chosen implementation and because the data structure is simple, the computationally expensive routines are a small part of the code and have been easy to optimise. Also, the author concludes that an edge-based scheme brings together good computational simplicity with good numerical results. A strategy of making no distinction between variables and treating each in the same way leads to a conceptually elegant matrix structure and a straightforward implementation.

As a result of this strategy, 91% of the code can be re-used. Moreover, the code can be easily amended to handle additional terms in the governing equations, or to solve a new numerical problem with little effort. Hence, the proposed methodology has resulted in a highly flexible numerical tool.

It is estimated that code development based on the innovative techniques described here will require 30% less effort than the traditional methods in general use. When the method is applied more generally to other engineering problems, this novel program architecture will significantly reduce the time consuming issues that arise from code development and code alteration which emerge when using large finite element packages. In conclusion, therefore, if the ten golden rules are accepted and the methodology implemented, this approach will significantly accelerate the development of numerical methods in all areas of engineering.

REFERENCES

[1] Budge K.G. and Peery J.S., RHALE – A MMALE shock physics code written in C++, *International Journal of Impact Engineering*, **14**: 107-120, 1993.

[2] Cross J.T., Masters I. and Lewis R.W., Why you should consider object-oriented programming techniques for finite element methods, *International Journal of Numerical Methods for Heat and Fluid Flow*, **9**: 333-347, 1999.

[3] Davey K. and Ward M. J., A successive preconditioned conjugate gradient method for the minimization of quadratic and nonlinear functions, *Applied Numerical Mathematics*, **35**: 129-156, 2000.

[4] Dubois-Pelerin Y. and Zimmermann T., Object-oriented finite element programming 3. An efficient implementation in C++, *Computer Methods in Applied Mechanics and Engineering*, **108**: 165-183, 1993.

[5] Dubois-Pelerin Y., Zimmermann T. and Bomme P., Object-oriented finite element programming 2. A prototype program in Smalltalk, *Computer Methods in Applied Mechanics and Engineering*, **98**: 361-397, 1992.

[6] Feng Z.Q., 2D or 3D frictional contact algorithms and applications in a large deformation context, *Communications in Numerical Methods in Engineering*, **11**: 409-416, 1995.

[7] Hinton E. and Owen D.R.J., Finite Element Programming, Academic Press, 1977.

[8] http://www.top500.org 26/01/2004

[9] Lewis R.W. and Schrefler B.A., The Finite Element Method in the Deformation and Consolidation of Porous Media, J. Wiley & Sons, Chichester, 1987.

[10] Lewis R.W., Masters I. and Cross J.T., Automatic Timestep Selection for the Super-Time-Stepping Acceleration on Unstructured Grids Using Object-Oriented Programming, *Communications in Numerical Methods in Engineering,* **13**: 249-260, 1997.

[11] Malan A.G., Lewis R.W., Ransing R.S. and Malan J.R., An object-oriented data structure for high-performance explicit numerical codes, ACME 2001: 9th Annual Conference of the Association of Computational Mechanics in Engineering (ed. Chan H C), Birmingham, 143-146, 2001.

[12] Masters I., Cross J.T. and Lewis R.W., A brief review of object-oriented finite element methods, in R. W. Lewis and J. T. Cross (ed.), Proc 10th International Conference on Numerical Methods in Thermal Problems, (Swansea, Pineridge Press, 1997), 766-776.

[13] Masters I., Usmani A.S., Cross J.T. and Lewis R.W., Finite Element Analysis of Solidification using Object-Oriented and Parallel Techniques, *International Journal For Numerical Methods In Engineering*, **40**: 2891-2909, 1997.

[14] Mentray P. and Zimmermann T., Object-oriented non-linear finite element analysis – Application to J2 plasticity, *Computers and Structures*, **49**: 767-777, 1993.

[15] Ohtsubo H., Kawamura Y. and Kubota A., Development of the object-oriented finite element system – Modify, *Engineering with Computers*, **9**: 187-197, 1993.

[16] Pidaparti R.V.M. and Hudli A.V., Dynamic analysis of structures using object-oriented techniques, *Computers and Structures*, **49**: 149-156, 1993.

[17] Silva E.J., Mesquitta R.C., Saldanha R.R. and Palmeira P.F.M., 'An Object-Oriented Finite -Element Program for Electromagnetic Field Computation', *IEEE Transactions on Magnetics* , **30**(5): 3618-3621, 1994.

[18] Smith I.M. and Griffiths D.V., Programming the Finite Element Method, 3rd Edition, John Wiley & Sons, Chichester, 1997.

[19] TecPlot homepage, www.amtec.com

Chapter 4

PHASE TRANSFORMATION

4.1 INTRODUCTION

The goal of this work is to develop a vertex-centred edge-based finite volume solver for problems involving fluid flow in porous media. The governing equations for such systems are diffusive in nature [1]. However, these diffusion-based equations contain highly non-linear terms. As this was the first attempt at solving a realistic engineering problem governed by a highly non-linear equation system using the proposed scheme, it was deemed necessary to compare the accuracy of the numerical solution against a verified and accepted result. Hence, the search for a one-dimensional problem involving fluid flow in porous media with an analytic solution was conducted. Finding such a test case proved difficult, and an alternate, yet mathematically similar problem was considered. The problem considered involves the phase transformation of a one-component material on a fixed grid using the enthalpy method. This example has been used for such comparisons in several papers ([14],[17],[18]), and contains all the relevant components required from the one-dimensional initial test case.

Phase transformations take place in familiar ways such as solidification, melting, vaporisation and condensation. Furthermore, the processes considered here have been restricted to melting and solidification, involving only the solid and liquid phases of materials. The most familiar phase change process is undoubtedly the melting of ice and the solidification of water. Other applications for the modelling of phase change includes glass forming, the freezing and freeze-drying of food products, semiconductor crystal growth, and the analysis of fire resisting containers [11]. In addition, the modelling of phase change has been used for geotechnical applications including soil freezing for excavations and construction over permafrost ([5],[6]). However, the single most important application of phase change modelling involves metal processing. These applications include the solidification of castings and ingots, welding and electro slag melting.

Melting and solidification are phase transformation processes which are accompanied by either absorption or release of thermal energy. A moving boundary exists that separates the two phases of different thermo-physical properties and at which thermal energy is adsorbed or liberated. When considering the solidification of binary and multi-component alloys, the physical phenomena become more complicated due to the phase transformations taking place over a range of temperatures corresponding to each component of the alloy. The lowest temperature corresponding to the fully liquid phase is called the 'liquidus' and the highest temperature corresponding to the solid phase is called the 'solidus'. The material between the solidus and the liquidus temperatures is partly solid and partly liquid and resembles a porous medium; therefore it is referred to as the mushy zone.

A complete understanding of the phase change phenomenon involves the analysis of the various processes that accompany it. From a macroscopic point of view, the most important of these processes is heat transfer. This process is complicated by the release or absorption of the latent heat of fusion at the liquid/solid interface. Several methods have been used by researchers to take account of the latent heat, which are divided into fixed and moving mesh methods. These methods are referred to as the 1-domain and the 2-domain methods respectively. As the names suggest, 1-domain methods involve the solutions of a continuous system with an implicit representation of phase change, while in the 2-domain or 'front tracking' methods, the solid and the liquid regions are treated separately and the phase change interface is explicitly determined as a moving boundary. A comprehensive treatment of the subject of moving boundary problems in general appears in a book by John Crank [2], while a good account of the early effort in the numerical modelling of phase change is given by Samonds [19]. The finite difference scheme has been utilised extensively in solving solidification problems by many researchers, for example Henzel and Kevarian

([8],[9],[10]), Weatherwax and Riegger [23] and Hansen [7]. While the first use of the finite element method was by Soliman and Fakhroo [20] in the analysis of ingot casting. A useful review of the methods used in a finite element context can be found in a paper by Dalhuijsen and Segal [4].

The classical solidification problem involves considering the conservation of energy in the domain Ω by dividing it into two distinct domains Ω_l and Ω_s, where $\Omega_l + \Omega_s = \Omega$. The energy conservation is written as,

$$\rho_l c_l \frac{\partial T}{\partial t} = \nabla \cdot k_l \nabla T \qquad \text{in } \Omega_l \tag{4.1}$$

and

$$\rho_s c_s \frac{\partial T}{\partial t} = \nabla \cdot k_s \nabla T \qquad \text{in } \Omega_s \tag{4.2}$$

where, T is the temperature, c is the specific heat, ρ is the density and k the thermal conductivity of the material; and the subscripts l and s denote liquid and solid phase respectively. In addition to the initial and boundary conditions, the complete description of the problem involves the interface conditions on the phase change boundary Γ_{sl}, which are,

$$\begin{gathered} T_{\Gamma_{sl}} = T_f \\ k_l \left(\frac{\partial T}{\partial x} \right)_l - k_s \left(\frac{\partial T}{\partial x} \right)_s = \rho L \frac{d\vartheta}{dt} \qquad \text{on } \Gamma_{sl} \end{gathered} \tag{4.3}$$

where, ϑ represents the position of the interface, $d\vartheta/dt$ the interface velocity, L the latent heat, and T_f the phase change temperature. The main problem in solving the classical problem lies in tracking the interface boundary position ϑ.

4.2 MODELLING PHASE TRANSFORMATION

As mentioned earlier, front tracking methods have been used to solve the classical problem by the use of moving meshes, or the transformation of coordinates

([3],[16],[24]), to explicitly satisfy the interface conditions. These methods can be used to solve isothermal phase change problems with good accuracy but become too complicated, even impossible, when faced with complex interface shapes which vary non-monotonically with time. Furthermore, front tracking methods cannot be readily used in the case of freezing over a range of temperatures [12]. Due to the limitations of the 2-domain methods, the subsequent discussions will be limited to the 1-domain methods. The 1-domain or fixed grid methods offer a more general solution as they account for the phase change conditions implicitly without attempting to *a priori* establish the position of the front. These methods are based on a weak formulation of the classical problem, which is commonly referred to as the *enthalpy formulation.* A single energy conservation equation is written for the whole domain as,

$$\frac{\partial H}{\partial t} = \nabla \cdot k \nabla T \qquad \text{in } \Omega \tag{4.4}$$

where H is the enthalpy function, and is fully defined in Appendix B.

4.3 EFFECTIVE HEAT CAPACITY

Among the fixed mesh methods, one of the most commonly used methods has been the *effective heat capacity* method. This method was derived from writing equation (4.4) as;

$$\frac{dH}{dT}\frac{\partial T}{\partial t} = \nabla \cdot k \nabla T \qquad \text{in } \Omega \tag{4.5}$$

Comparing with the standard heat conduction equations such as equations (4.1) and (4.2), one can write,

$$c_{eff} = \frac{dH}{dT} \tag{4.6}$$

where c_{eff} is the effective heat capacity, which can be evaluated directly from equations (B.2) in Appendix B, as shown in reference [4] as,

$$\begin{aligned} c_{eff} &= \rho c_s && (T < T_s) \\ c_{eff} &= \rho c_f + \frac{L}{T_l - T_s} && (T_s \le T \le T_l) \\ c_{eff} &= \rho c_l && (T > T_l) \end{aligned} \tag{4.7}$$

The typical variation of the effective heat capacity and enthalpy with temperature is illustrated in Figure 4.1. Huang and Usmani [12] state that if this directly evaluated specific heat is used, it will be necessary to maintain an interval of temperatures for the evolution of latent heat; otherwise the effective heat capacity will become infinite. Therefore, this method cannot accurately model an iso-thermal change of phase due to the requirement of a temperature range. Due to the step like behaviour of c_{eff} around the phase change interval, numerical oscillations may occur, making the achievement of a convergent solution difficult [15].

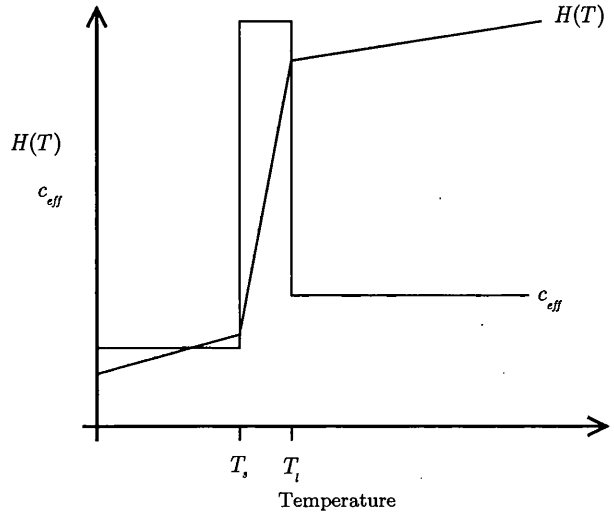


Figure 4.1: Variation of the effective heat capacity and enthalpy with temperature.

4.4 ENTHALPY METHOD

In order to overcome the difficulties encountered in using a directly evaluated effective capacity, the use of an averaging technique is advocated. These techniques are generally referred to as the *enthalpy method.* The enthalpy method enables the heat capacity to be defined as a smooth function of temperature [15]. Morgan *et al.* [14] and Lemmon [13] suggest an approximation which is reported to work satisfactorily,

$$\frac{dH}{dT}=\left(\frac{\left(\frac{dH}{dx}\right)^2+\left(\frac{dH}{dy}\right)^2}{\left(\frac{dT}{dx}\right)^2+\left(\frac{dT}{dy}\right)^2}\right)^{\frac{1}{2}} \tag{4.8}$$

This will be averaging technique implemented into the scheme. However, several researchers have used different averaging techniques with varying rates of success. Some of the more commonly encountered techniques are listed in Appendix C. In using the above techniques in a finite element analysis it is common practice to interpolate H from the nodal values using the same basis function as for T thus obtaining a smoothing effect. The appearance of space derivatives in the above equations ensures the inclusion of a phase change.

The enthalpy method, although reasonably accurate and simple to implement in any heat transfer code, suffers from a number of deficiencies some of which have been discussed in ([21],[22]). The main drawbacks can be listed as follows,

- Isothermal phase changes cannot be modelled correctly due to the necessity of a temperature range for the evolution of latent heat.
- The method requires small spatial and temporal step sizes; otherwise convergence becomes difficult. Hence this does not allow for quick computer runs.

- Due to the requirements of small mesh size and timesteps an analysis can become prohibitively expensive if a difficult problem is encountered, such as that with a narrow heating range and large latent heat.

Another method used in the calculation of phase change problems is the *heat source method* first suggested in the finite element context by Rolph and Bathe [17] and then by Roose and Storrer [18]. However, this method lacks the mathematical theory that supports the enthalpy method so well, and unlike the enthalpy method is difficult to implement. In spite of these drawbacks it offers an attractive alternative to the enthalpy method, as it does not suffer from any of the deficiencies of the enthalpy method listed above. The heat source method has not been implemented in this work. For details regarding the implementation of the scheme the reader should refer to Rolph and Bathe [17].

4.5 BENCHMARK EXAMPLES

A one-dimensional benchmark example of solidification as shown in Figure 4.2 is solved by the enthalpy method discussed above and compared with the analytic solution. The material properties, dimensions, boundary and initial conditions are as shown in the same figure. T_o and T_f refer to the initial and freezing temperatures respectively. No particular units are necessary. This example has been used for such comparisons in several papers ([14],[17],[18]). The temperature vs. time curve at a chosen point ($x = 1.0$ here) in the domain is used for the comparisons. Dalhuijsen and Segal [4] claim that this method is more reliable than the normally used interface position vs. time curves. The same example with reversed temperatures as shown in Figure 4.3 will be used to solve a melting example.

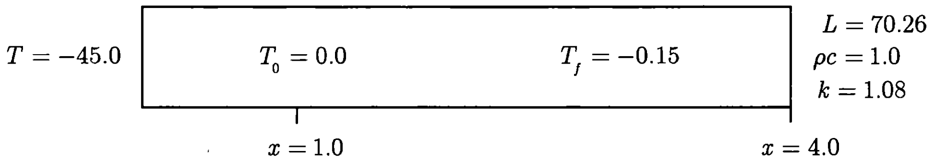


Figure 4.2: Solidification example.

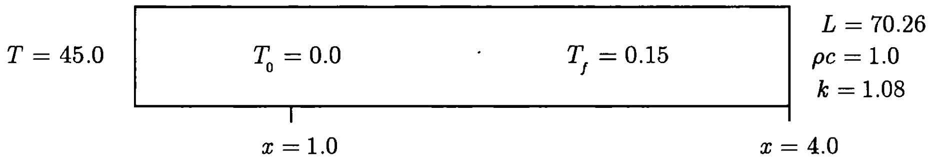


Figure 4.3: Melting example.

The analytical solutions for 1-D solidification and melting were taken from [12] and were implemented into the solver to generate temperature time curves at specified points in the domain, and are included in Appendix D and E respectively.

4.6 SOLUTION PROCEDURE

The expansion of the heat conduction equations (4.5) on a one-dimensional domain Ω gives;

$$\frac{dH}{dT}\frac{\partial T}{\partial t}=\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) \tag{4.9}$$

The spatial derivative terms in equation (4.9) are discretised by application of Green's theorem, and by the assumption that the spatial derivative is linear over the control volume Ω_m. Integration over the control volume gives;

$$\int_{\Omega}\frac{dH}{dT}\frac{\partial T}{\partial t}\,d\Omega=\int_{\Omega}\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)\,d\Omega \tag{4.10}$$

The application of the Euler backward difference scheme, and the edge-based discretisation procedures discussed in chapter 2 gives for all nodes m in the domain;

$$\left.\frac{dH}{dT}\right|_m \frac{T_m^{n+1}-T_m^n}{\Delta t}\cdot\Omega_m = \left.\frac{\partial}{\partial x}\left(k_{mn}\frac{\partial T^n}{\partial x}\right)\right|_m \Omega_m \tag{4.11}$$

where T_m^n = temperature at node m at time n, and T_m^{n+1}= temperature at node m at time *n+1*. The term k_{mn} is the value of k applied to edge$_{mn}$, as this problem an isotropic, homogeneous domain, $k = 1.08$ everywhere. The enthalpy H is obtained directly from the enthalpy curves (see Figure 4.6 and Figure 4.12) and is evaluated using a linear interpolation of the value at $T=T^n$. The enthalpy term dH/dT is calculated using the approximation reported by Morgan *et al.* [14] and Lemmon [13] (see equation (4.8)), but reduced to a one-dimensional case, that is;

$$\frac{dH}{dT} = \left(\frac{\left(\frac{dH}{dx}\right)^2}{\left(\frac{dT}{dx}\right)^2}\right)^{\frac{1}{2}} = \frac{\left(\frac{dH}{dx}\right)}{\left(\frac{dT}{dx}\right)} \tag{4.12}$$

where in this case $T=T_m^n$, and $H_m = f\left(T_m^n\right)$. By introducing the terms Φ and Ξ defined as;

$$\Phi_m = \left.\frac{dH}{dT}\right|_m = \frac{\left.\frac{dH}{dx}\right|_m}{\left.\frac{dT^n}{dx}\right|_m} \quad , \quad \Xi_m = \left.\frac{\partial}{\partial x}\left(k_{mn}\frac{\partial T_m^n}{\partial x}\right)\right|_m \Omega_m \tag{4.13}$$

Equation (4.11) simplifies to;

$$\Phi_m \frac{T_m^{n+1} - T_m^n}{\Delta t} \cdot \Omega_m = \Xi_m \tag{4.14}$$

Re-arranging equation (4.14), gives the linear update for the primary variable T as:

$$T_m^{n+1} = \frac{\Xi_m \Delta t}{\Phi_m \Omega_m} + T_m^n \tag{4.15}$$

Due to the highly non-linear nature of the problem a more efficient non-linear explicit primary update algorithm was successfully introduced. The primary variable update is performed in the following manner. Defining $T_m^{n,k}$ as a value of T for timestep n, iteration k and node m, we initiate each timestep with $T_m^{n+1,1} = T_m^n$ for all nodes. Then, a variable $\overline{T}$ is proposed;

$$\overline{T}_m = \beta T_m^{n+1,k} + (1-\beta) T_m^n \tag{4.16}$$

where $\{\beta \in \mathbb{R} : \beta \in [0,1]\}$. Derivatives and non-linear variables that are functions of T in equation (4.15) are discretized using values of $\overline{T}$, to form the left hand side of the equation. Hence, equation (4.15) becomes,

$$T_m^{n+1} = \frac{\overline{\Xi}_m \Delta t}{\overline{\Phi}_m \Omega_m} + T_m^n \tag{4.17}$$

where,

$$\overline{\Phi}_m = \left.\frac{d\overline{H}}{dT}\right|_m = \frac{\left.\frac{d\overline{H}}{dx}\right|_m}{\left.\frac{d\overline{T}_m}{dx}\right|_m} \quad , \quad \overline{\Xi}_m = \frac{\partial}{\partial x}\left(k_{mn} \frac{\partial \overline{T}_m}{\partial x}\right)\Omega_m \tag{4.18}$$

and $\overline{H}_m = f\left(\overline{T}_m^n\right)$. If the system has converged, then the following relationship holds true.

$$\left|\ \left\|T_m^{n+1,k+1}\right\| - \left\|T_m^{n+1,k}\right\|\ \right| \leq \varepsilon \tag{4.19}$$

Where $\{\varepsilon \in \mathbb{R} : \varepsilon \ll 1\}$ is a user defined convergence criterion. Once convergence is achieved the cycle begins for the next time step with $T_m^n = T_m^{n+1,k+1}$. However, if equation (4.19) is not satisfied, then the non-linear loop is repeated until a satisfactory outcome is achieved, or the non-linear counter exceeds a specified value.

A flowchart of the solution procedure is given in Figure 4.5, where the symbols correspond to those used in equations (4.9) to (4.19). If we define the primary variable of the problem as $T \in \left\{T_m^n, \overline{T}_m\right\}$, then let the set of secondary variables calculated or derived from the primary set be $\mathbb{S}(T)$, where;

$$\mathbb{S}(T) = \left\{H_m, \overline{H}_m\right\} \ . \tag{4.20}$$

Hence in Figure 4.5, the term **CALCULATE** $\mathbb{S}(T)$ implies the calculation of the set of secondary variables using the primary variable *T*. The terms H_m and $\overline{H}_m$ are values from the enthalpy curve specific to each problem (see Figure 4.6 and Figure 4.12), where $H_m = f\left(T_m^n\right)$ and $\overline{H}_m = f\left(\overline{T}_m\right)$; and, H_m and $\overline{H}_m$ lie on the same curve. These terms are calculated through linear interpolation of the tabulated values of the functions.

4.7 PHASE TRANSFORMATION RESULTS – 1-D

The mesh used in both the 1-D solidification and melting examples considered in this chapter is a one-dimensional grid with equally spaced nodes. The necessary input data required to generate the mesh and run the problem are shown in Table 4.1.

No. of nodes	1001
No. of edges	1000
No. time steps	400000
Δt	0.00001
Δx	0.004
k	1.08
β	0.667
ε	0.000001

Table 4.1: Input data for the 1-D solidification and melting problems.

4.7.1 Solidification - 1-D

The solidification problem was solved using the vertex-centred edge-based finite volume scheme developed here. The initial and boundary conditions applied to the domain are shown in Figure 4.4, and the enthalpy curve used is shown in Figure 4.6 and tabulated in Table 4.2.

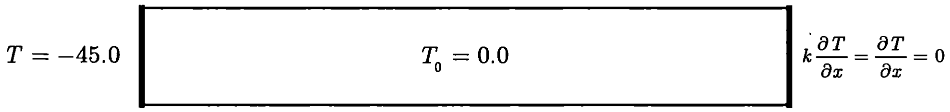


Figure 4.4: Initial and boundary condition applied for the solidification problem.

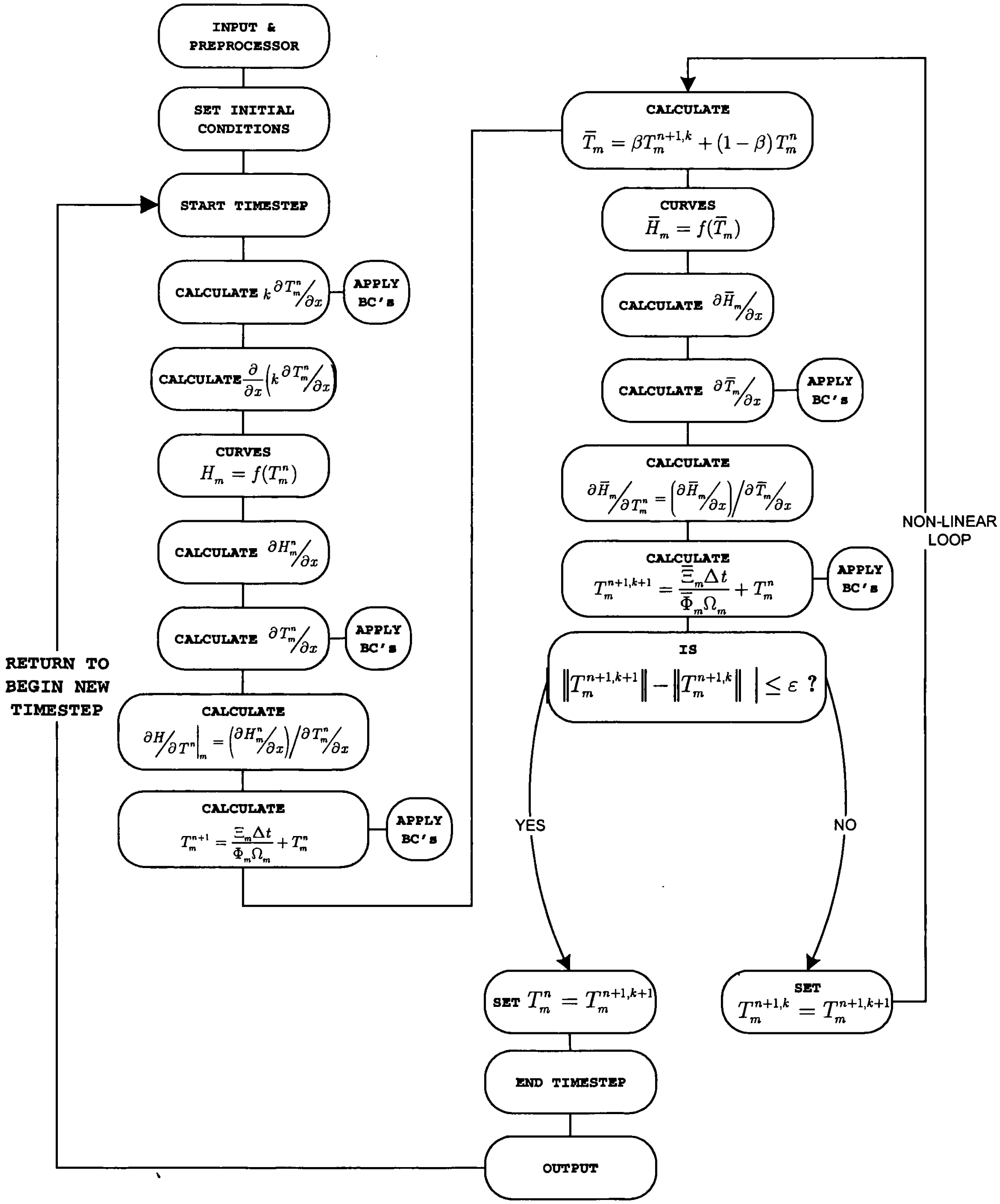


Figure 4.5: Program flow chart for the 1-D phase transformation problem.

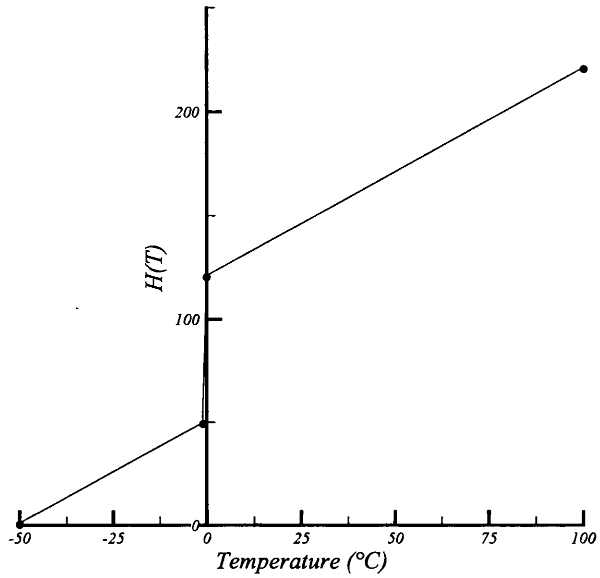


Figure 4.6: The enthalpy curve used for the solidification test case.

Temp. (°C)	*H(T)*
-50	1.0
-1.15	49.85
-0.15	121.11
5.0	126.26
100	221.26

Table 4.2: The tabulated enthalpy curve used for the solidification test case.

As can be seen from Figure 4.7, the solution for the calculated temperature using the enthalpy method, obtained using the finite volume scheme closely matches the analytic solution for the solidification problem at the point $x = 1$ in the domain over time. The plot in Figure 4.8 shows a comparison between the calculated finite volume solution and the solution obtained for the same problem by Huang and Usmani [12] using the finite element method on a mesh with 20-9 noded elements. Again, there is a very good comparison between the results, with the finite volume solution being a smoother plot than the finite element solution.

This is probably due to the greater mesh density used in the simulation, thus the zone of transition between the liquid and solid material is captured without loss of accuracy. Also, Figure 4.9 shows that the finite volume solution is closer to the analytic solution than Huang and Usmani finite element solution for the same problem.

Let the calculated solution at the point $x = 1.0$ in the domain at time t seconds using the finite volume scheme and the Huang and Usmani finite element solution [12] be defined as FV and FE respectively. We define the analytic solution for the problem at the same point and time as AS. An absolute error term, dT, is introduced, which is defined as the modulus of the difference between either the calculated finite volume solution or the finite element solution, and the analytic solution at that particular point in time, that is;

$$dT = \left|(\cdot) - \mathrm{AS}\right| \tag{4.21}$$

where $(\cdot) =$ FV or FE. This error term will determine the extent in which the solutions differ from the analytic, and hence the more accurate solution established. A plot of the absolute error, dT, against time for both the calculated finite volume solution and the Huang and Usmani finite element solution for the solidification test case is shown in Figure 4.10. This error plot indicates that there is little difference in the accuracy of both schemes between $t = 0$ and 2 seconds, however, the finite volume scheme calculates the temperature at the zone of transition between the liquid and solid material with much less error than the finite element method. This is evident from the peaks on the graph at $t = 1$ second. Between $t = 2$ and 4 seconds, the error observed in the finite volume solution as compared to the finite element solution is less. The major difference in the two error plots is the smoothness of the finite volume plot compared to the finite element plot. This confirms the comments already made regarding the smoothness of the two solutions (see Figure 4.8).

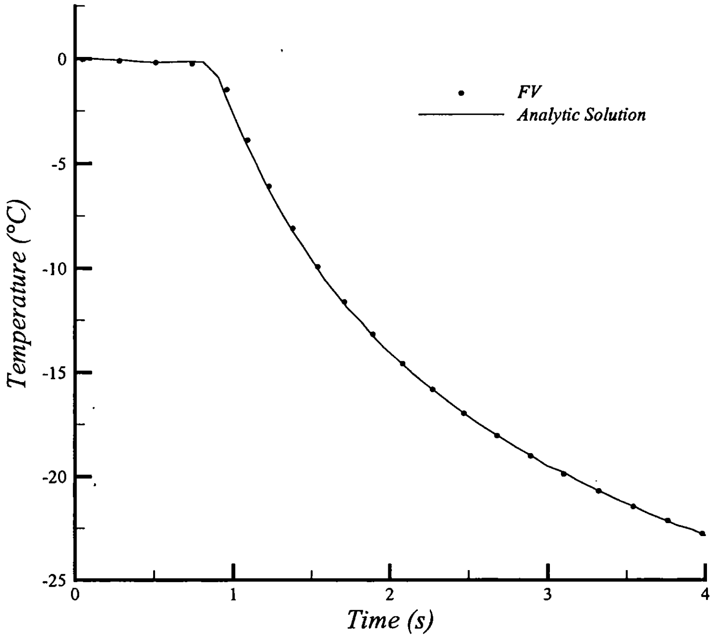


Figure 4.7: The calculated FV solution compared against the analytic solution for solidification over time at the point $x = 1$ on the 1-D domain.

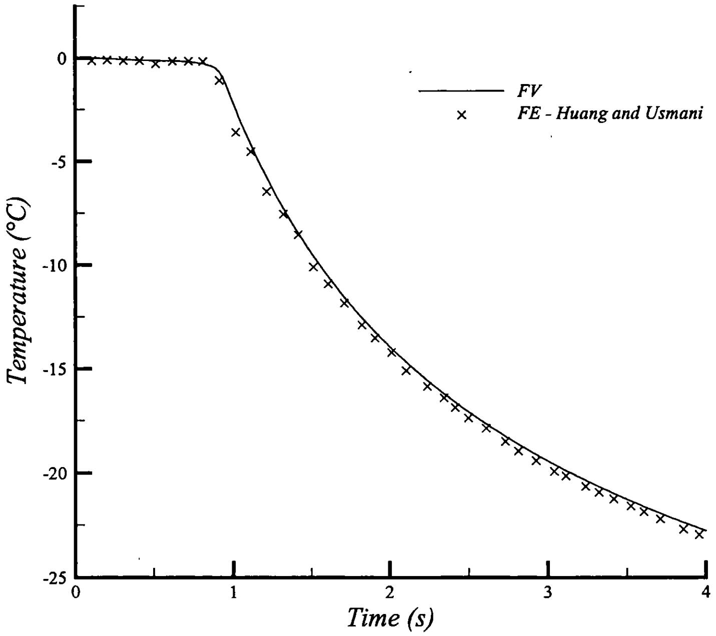


Figure 4.8: The calculated FV solution compared against the Huang and Usmani FE solution [12] for solidification over time at the point $x = 1$ on the 1-D domain.

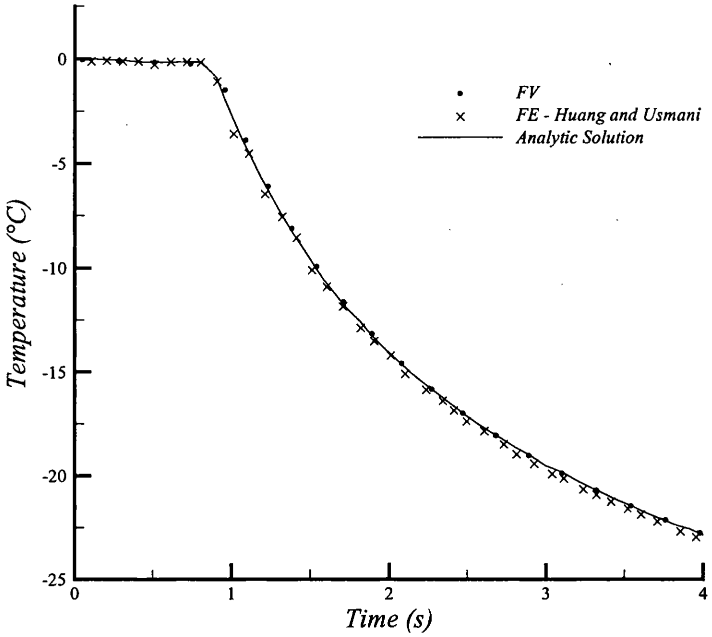


Figure 4.9: The calculated FV solution, the Huang and Usmani FE solution [12], and the analytic solution for solidification over time at the point $x = 1$ on the 1-D domain.

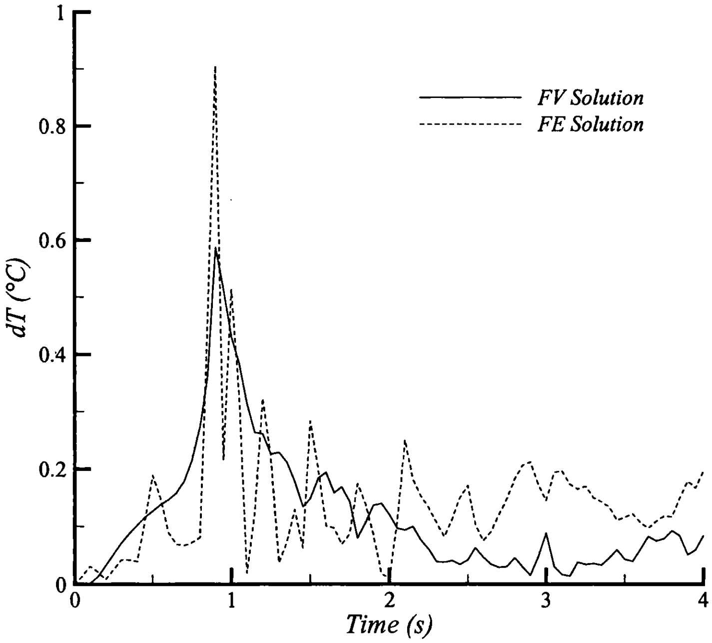


Figure 4.10: Calculated absolute error in the FV solution and the Huang and Usmani FE solution [12] for solidification over time at the point $x = 1$ on the 1-D domain.

4.7.2 Melting - 1-D

The melting problem was solved using the vertex-centred edge-based finite volume scheme developed here. The initial and boundary conditions applied to the domain are shown in Figure 4.11, and the enthalpy curve used is shown in Figure 4.12 and tabulated in Table 4.3.

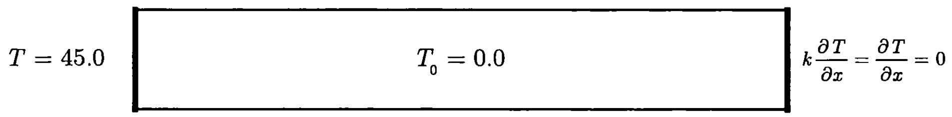


Figure 4.11: Initial and boundary conditions applied for the melting problem.

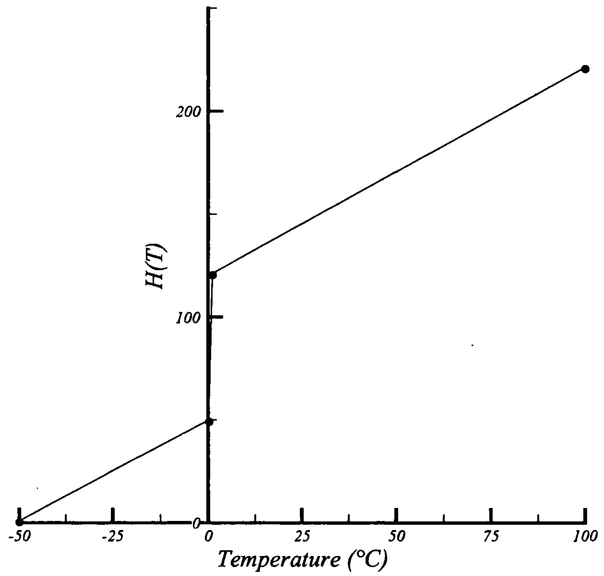


Figure 4.12: The enthalpy curve used for the melting test case.

Temp. (°C)	H(T)
-50	1.0
0.15	49.85
1.15	121.11
5.0	126.26
100	221.26

Table 4.3: The tabulated enthalpy curve used for the melting test case.

Again, similar trends are observed with the solution to the melting case as already seen with the solidification problem above. The calculated temperature at the point $x = 1$ using the finite volume scheme almost exactly matches the analytic solution as shown in Figure 4.13. The only discrepancy is in the last 1 second, when the calculated solution differs slightly from the analytic. However, the calculated solution is everywhere smooth, and is free from temporal oscillations. It is noted that the calculated solution captures the zone of transition almost exactly. The solution obtained by Huang and Usmani [12] for the same problem using the finite element method encounters difficulty in this area, as seen from Figure 4.14 and Figure 4.15. Moreover, the finite element solution does not display the smoothness of plot as obtained using the finite volume scheme, however as discussed above, this may be due to the different mesh densities used.

Figure 4.16 presents the calculated absolute error in the finite volume solution and the Huang and Usmani [12] finite element solution for this case at the point $x = 1$ on the 1-D domain. The absolute error is calculated as per equation (4.21). In terms of accuracy between $t = 0$ and 3 seconds, the finite volume solution is superior to the finite element solution in this case. Moreover, this plot only reinforces what has already been said, that is that the improvement in accuracy at the zone of transition between solid and liquid material (approximately $t = 1$ second) gained by employing the finite volume scheme over the Huang and Usmani finite element scheme is great. However, at $t > 3$ seconds, the finite element solution displays improved accuracy over the finite volume scheme. This is evident from inspection of Figure 4.15, where the finite volume is seen to diverge from the analytic solution in this region.

As perceived in the error plot for the solidification test case in Figure 4.10, the melting error plots show similar trends as regards smoothness of solution; where the finite element error plot displays increased noise as compared to the finite volume case. Again, this only underlines the comments made on the smoothness of the solutions obtained using the two schemes (see Figure 4.14).

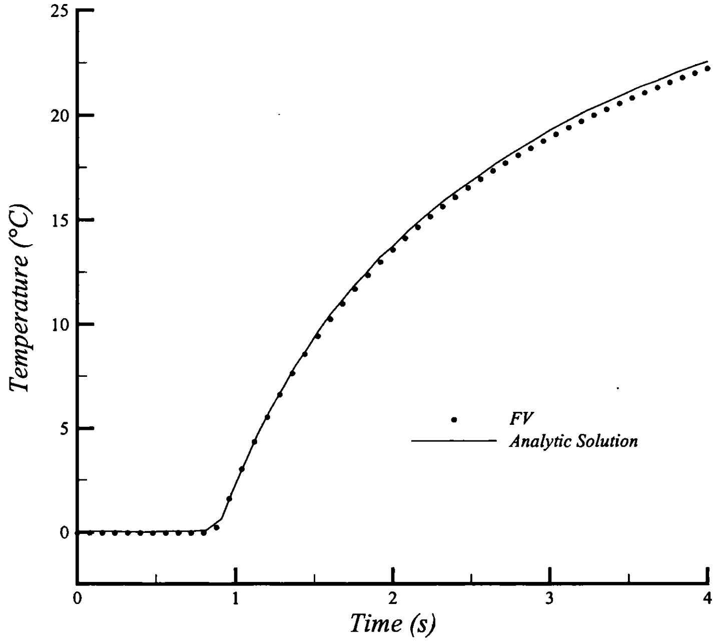


Figure 4.13: The calculated FV solution compared against the analytic solution for melting over time at the point $x = 1$ on the 1-D domain.

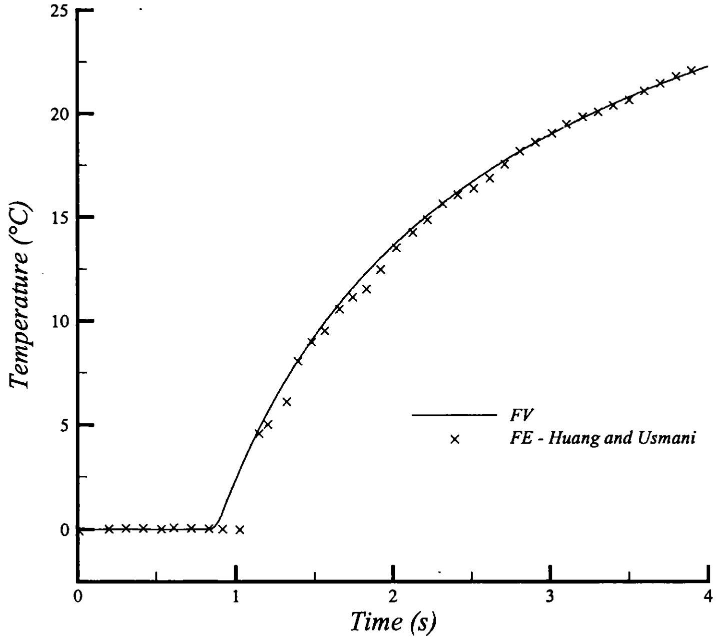


Figure 4.14: The calculated FV solution compared against the Huang and Usmani FE solution [12] for melting over time at the point $x = 1$ on the 1-D domain.

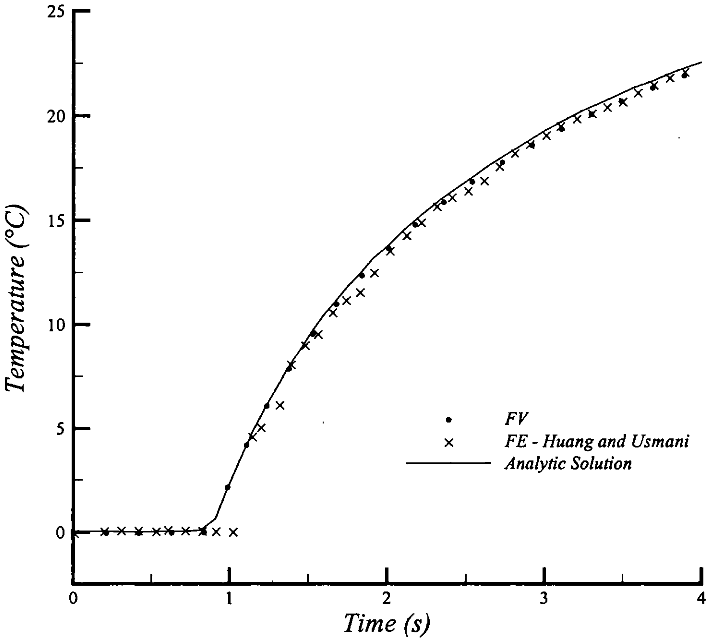


Figure 4.15: The calculated FV solution, the Huang and Usmani FE solution [12], and the analytic solution for melting over time at the point $x = 1$ on the 1-D domain.

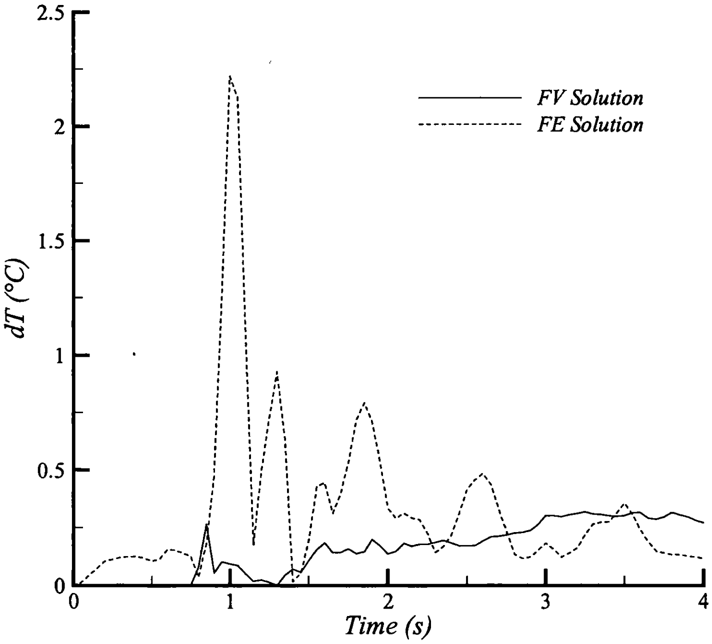


Figure 4.16: Calculated absolute error in the FV solution and the Huang and Usmani FE solution [12] for melting over time at the point $x = 1$ on the 1-D domain.

4.8 SOLIDIFICATION – 2-D

The solidification problem described and solved above in one dimension is here considered on a two-dimensional domain. The computational domain has dimensions 4×1 units as shown in Figure 4.17a. The domain was discretised using structured quadrilateral elements, with 200 nodes in the x direction and 20 nodes in the y direction, as illustrated in Figure 4.17b. In the x direction, the nodes have an exponential distribution with initial spacing of 0.01. In all the mesh contains 4000 nodes and 3781 elements.

The expansion of the heat conduction equations (4.5) on a two-dimensional domain Ω gives;

$$\frac{dH}{dT}\frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) \tag{4.22}$$

The solution procedure is identical to that of the one-dimensional case, with the exception of a second dimension. The input parameters tabulated in Table 4.1, and the enthalpy curve used for the 1-D solidification problem shown in Figure 4.6 were used in this case also. The initial and boundary conditions applied to the domain are as shown in Figure 4.18.

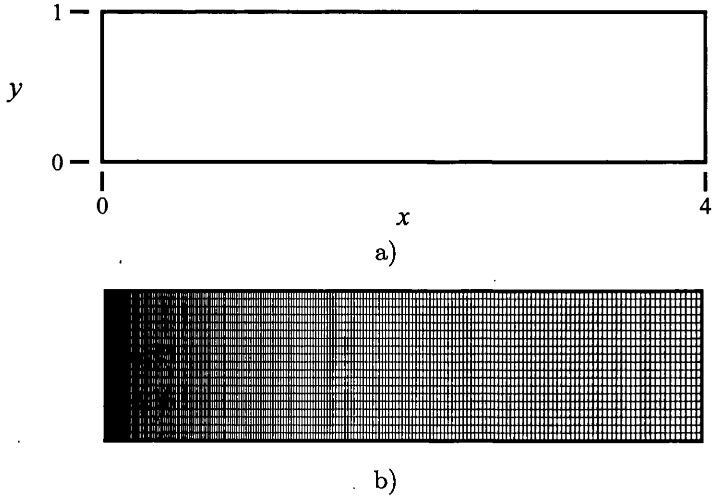


Figure 4.17: a) boundary of the computational domain, and b) the structured quadrilateral mesh used in the 2-D solidification problem.

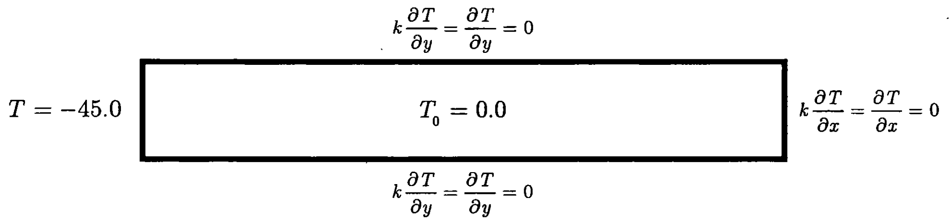


Figure 4.18: Initial and boundary condition applied for the 2-D solidification problem.

The only difference in the formulation is that the enthalpy term dH/dT is calculated using the approximation reported by Morgan *et al.* [14] and Lemmon [13] (see equation (4.8)), that is;

$$\frac{dH}{dT} = \left(\frac{\left(\frac{dH}{dx}\right)^2 + \left(\frac{dH}{dy}\right)^2}{\left(\frac{dT}{dx}\right)^2 + \left(\frac{dT}{dy}\right)^2} \right)^{\frac{1}{2}} \tag{4.23}$$

The plot of the calculated temperature at the node with coordinates (1, 0.5) obtained using the finite volume scheme over time is shown in Figure 4.19, along with the same analytic solution as was used in 1-D. The calculated values closely match the analytic solution for the solidification problem.

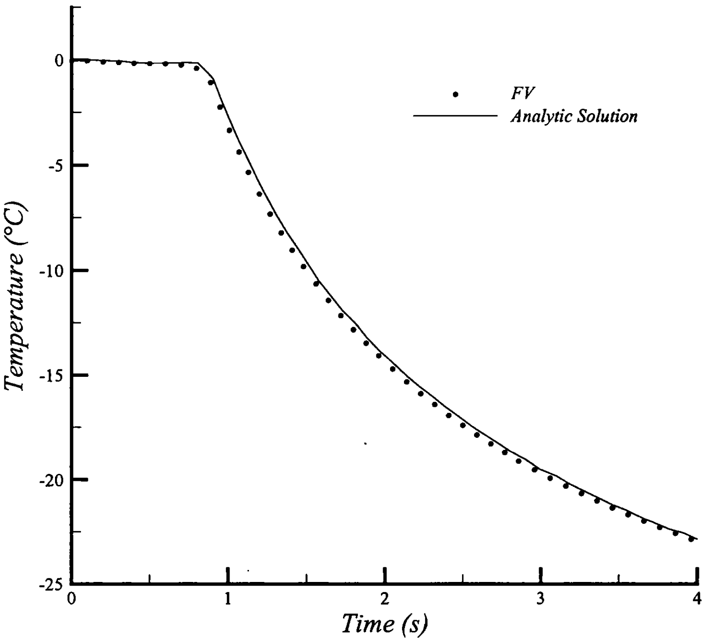


Figure 4.19: The calculated FV solution compared against the analytic solution for solidification over time at the point $x = 1$ on the 2-D domain.

4.9 CLOSURE

The phase transformation problem considered here is not related to the field of study, that being fluid flow in porous media. However, the governing equations are diffusive in nature, and are mathematically very similar to the equations that govern the flow of fluids through porous materials. The problem considered here also contain highly non-linear properties as will be encountered when considering the material property curves of fluids, such as permeability and capillary pressure curves. Hence, this problem was considered as it contained many aspects that are necessary in the development of a solver for fluid flow in porous media.

The proposed edge-based finite volume scheme is shown to have accurately solved the phase change problem, both for solidification and melting on a one-dimensional domain. The calculated solution was compared to the analytic and published numerical solutions for the same problem with excellent correspondence. Further, the solidification problem was solved on a two-dimensional domain using a structured quadrilateral element mesh. This also converged to the analytical solution.

This chapter has proved that the chosen edge-based finite volume scheme handles problems whose discretised form contains a highly non-linear capacity (mass) matrix $\mathbf{M}(\underset{\sim}{\mathbf{u}})$ in equations of the following type;

$$\mathbf{M}(\underset{\sim}{\mathbf{u}})\frac{\partial \underset{\sim}{\mathbf{u}}}{\partial t} = \mathbf{K}\underset{\sim}{\mathbf{u}} \qquad (4.24)$$

where, $\underset{\sim}{\mathbf{u}}$ is a vector of some arbitrary parameter, and $\mathbf{K}$ is a stiffness matrix. In order to solve the governing equation for fluid flow in porous media, the scheme must be able to handle the above equation system, but with the added complication of the addition of a highly non-linear term $\mathbf{K} = \mathbf{K}(\underset{\sim}{\mathbf{u}})$. In this chapter, $\mathbf{K}$ was simply a constant. This added complication will be considered in Chapter 5 - Saturated-Unsaturated Groundwater Flow.

In conclusion, we can be assured that the building blocks of the solver are in place, and significant progress towards a porous media fluid flow code has been achieved.

REFERENCES

[1] Bear J., Hydraulics of Groundwater (McGraw-Hill Series in Water Resources and Environmental Engineering), McGraw Hill College Div., 1979.

[2] Crank J., Free and Moving Boundary Problems, Clarendon Press, Oxford, 1984.

[3] Crank J., How to deal with moving boundaries in thermal problems, In R.W. Lewis, K. Morgan and O.C. Zienkiewicz, editors, *Numerical Methods in Heat Transfer,* Wiley, Chichester, 1981.

[4] Dalhuijsen A.J. and Segal A., Comparison of finite element techniques for solidification problems, *International Journal for Numerical Methods in Engineering,* **23**: 1807-1829, 1986.

[5] DelGuidice S., Comini G. and Lewis R.W., Finite element simulation of freezing processes in soils, *International Journal of Numerical and Analytical Methods in Geomechanics*, **2**: 223-235, 1978.

[6] Frivik P.E., Thorbergsen E., DelGuidice S. and Comini G., Thermal design of pavement structures in seasonal frost areas, *Journal of Heat Transfer,* **99**: 533-540, 1977.

[7] Hansen P.H., Numerical simulation of the solidification process, In proceedings: *Solidification and Casting of Metals,* Sheffield, U.K., 1979. The Metals Society.

[8] Henzel J.G. and Keverian J., 'Comparison of calculated and measured solidification patterns in a variety of steel castings', *AFS Cast Metals Research Journal,* 19-36, 1965.

[9] Henzel J.G. and Keverian J., Predicting casting solidification patterns with a computer, *Foundry,* 50-53, 1964.

[10] Henzel J.G. and Keverian J., Predicting casting solidification patterns in a steel valve casting by means of a digital computer, *Metal Eng. Quarterly,* 39-44, 1965.

[11] Huang H.C. and Lewis R.W., Adaptive analysis for heat flow problems using error estimation techniques, In *Sixth International Conference for Numerical Methods in Thermal Problems,* Swansea, U.K., July 1989, Pineridge Press, Swansea.

[12] Huang H.-C. and Usmani A.S, Finite Element Analysis for Heat Transfer, Springer-Verlag, 1994.

[13] Lemmon E.C., Multidimensional integral phase change approximations for finite element conduction codes, In R.W. Lewis, K. Morgan and O.C. Zienkiewicz, editors, *Numerical Methods in Heat Transfer,* Wiley, Chichester, 1981.

[14] Lewis R.W., Morgan K. and Zienkiewicz O.C., An improved algorithm for heat conduction problems with phase change, *International Journal for Numerical Methods in Engineering,* **12**: 1191-1195, 1978.

[15] Lewis R.W., Morgan K., Thomas H.R. and Seetharamu K.N., The Finite Element Method in Heat Transfer Analysis, John Wiley & Sons, 1996.

[16] O'Neill K. and Lynch D.R., A finite element solution of freezing problems using a continuously deforming coordinate system, In R.W. Lewis, K. Morgan and O.C. Zienkiewicz, editors, *Numerical Methods in Heat Transfer,* Wiley, Chichester, 1981.

[17] Rolph W.D. and Bathe K.J., An efficient algorithm for analysis of non-linear heat transfer with phase changes, *International Journal for Numerical Methods in Engineering,* **18**: 119-134, 1982.

[18] Roose J. and Storrer O., Modelization of phase changes by ficticious heat flow, *International Journal for Numerical Methods in Engineering,* **20**: 217-225, 1984.

[19] Samonds M.T., Finite element simulation of solidification in sand mould and gravity die castings, PhD Thesis, University of Wales Swansea, 1985.

[20] Soliman J.I. and Fakhroo E.A., Finite element solution of heat transmission in steel ingots, *Journal of Mechanical Eng. Science,* **14**: 19-24, 1972

[21] Viskanta R., Phase change heat transfer, In G.A. Lane, editor, *Solar Heat Storage Latent Heat Materials,* CRC Press, 1983.

[22] Voller V.R., Cross M. and Markatos N.C., An enthalpy method for convection/diffusion phase change, *International Journal for Numerical Methods in Engineering,* **24**: 271-284, 1987.

[23] Weatherwax R. and Riegger O.K., Computer-aided solidification study of a die cast aluminium piston, *AFS Transactions,* **85**: 317-322, 1977.

[24] Yoo J. and B. Rubinsky B., Numerical computation using finite elements for the moving interface in heat transfer problems with phase transformations, *Numerical Heat Transfer,* **6**: 209-222, 1983.

Chapter 5

SATURATED-UNSATURATED GROUNDWATER FLOW

5.1 INTRODUCTION

This chapter is concerned with the application of the developed vertex-centred edge-based finite volume scheme in solving problems involving saturated-unsaturated fluid flow in porous media. The governing equation for the flow of water in a slightly compressible and partly saturated soil is described by the Richards equation [28] which contains highly non-linear material property terms. Numerical modelling of water flow in variably saturated soils is needed in a wide range of applications, and the Richards equation model is typically used to describe variably saturated flows [14]. Richards equation is defined by coupling a statement of flow continuity with the Darcy equation and is commonly cast into one of three forms: 1) moisture based form, 2) pressure head based form, or 3) a mixed or coupled form. A full description of each equation form is presented in Kavetski *et al.* [15]. Dozens of papers have been published on numerical solutions to the Richards equation, and Tocci *et al.* [32] reports that Richards equation is considered a state-of-the-art treatment of groundwater flow problems in an air-water system. The most common approaches currently use low-order finite difference or finite element methods in space to solve the governing equations. For stability consideration, most of the existing approaches solve the equation with a fully implicit approach and use pressure head, which is continuous in both saturated and unsaturated zones, as the primary variable. This requires estimates of soil hydraulic properties at the new time level, giving a non-linear system that has to be solved by iteration. The application of Richards equation presents one of the most significant challenges in numerical solution of variably saturated flow, and considerable effort has been expended over the last few decades in solving it (e.g.[6],[29],[36],[23],[10],[32],[12]). It has been shown that such formulations are capable of solving fluid flow problems through vertical soil columns and earth dam sections ([12],[35],[37] etc.). However, due to certain simplifications made to the equation system (outlined below), the full analysis of multi-phase problems cannot be modelled using the Richards equation. In these situations, such as the simulation of the three-phase system that occurs in petroleum reservoirs (oil-water-gas), a coupled system of flow equations for each phase must be solved (see Chrichlow [5] and Aziz and Settari [2]).

The flow of fluid through an unsaturated soil is predominantly an immiscible two-phase flow problem. Using this two-phase formulation it is necessary to consider the mass balance of the air and fluid phases. Hence, the flow model is composed of two material balance equations for water and air (gas) respectively, plus an algebraic equation of fluid saturation continuity,

$$S_w + S_g = 1.0 \tag{5.1}$$

where S_w is the saturation of the fluid phase and S_g the saturation of the gas phase in the system. In addition, to fully describe the problem Forsyth *et. al* ([10],[9]) proposed an algebraic equation expressing the link between water saturation and the capillary pressure between air and water.

A simplification to this approach may be effected by assuming that the air mass flow through the system is negligible as compared to the liquid mass flow, and may therefore be ignored. In this case, the problem reduces to a single-phase formulation as shown by Gottardi and Venutelli [11]. Gottardi and Venutelli showed that the partially saturated flow model may then be described by the water mass balance equation, and the equations expressing the initial and boundary conditions [12]. Valid numerical solutions of this equation are also difficult to obtain due to the strong non-linearity of the soil relative permeability against water moisture content and of the capillary pressure versus the water moisture content relationship ([11],[3]).

The application of the edge-based finite volume method for modelling groundwater flow in saturated-unsaturated soils is presented, and the flow equation is solved by applying a vertex-centred edge-based finite volume scheme. The flexibility of the scheme is demonstrated by its ability to handle both structured and unstructured grids (comprising both triangular and quadrilateral elements), and the ability to deal with highly non-linear material properties on homogeneous, and inhomogeneous domains. An acceleration algorithm is employed which utilises an explicit non-linear loop for the primary variable update. In order to evaluate the performance of the edge-based scheme in solving groundwater flow problems, three test cases are presented. The first two cases model the vertical infiltration of water into an initially dry soil column by applying a prefixed pressure head and a prefixed flow rate at the top of the column respectively analysed by Gottardi and Venutelli [12]. The third case models the infiltration flow of water across an initially dry earth dam. An optimisation exercise was conducted to determine the optimum input values for the parameters within the non-linear iterative loop for the pressure head update, and these values then applied to the above-mentioned test cases. Finally, conclusions are drawn on the efficiency and accuracy of the scheme.

5.2 GOVERNING EQUATIONS

Under the hypothesis that Darcy's law holds true, Bear [3] showed that the pressure head form of the Richards equation, written for the flow domain Ω, can be written using indicial notation as:

$$\frac{\partial}{\partial x_i}\left[\alpha k_{ij} k_r \left(\frac{\partial h}{\partial x_j} + \frac{\partial y}{\partial x_j}\right)\right] - \alpha q_w = \alpha\left(C + S_w S_s\right)\frac{\partial h}{\partial t} \qquad \text{on } \Omega \qquad (5.2)$$

where h is the pressure head [L], x_i $(i = 1,2)$ are the Cartesian coordinates (x_3 is in the vertical direction and positive upward) [L], y is the distance from the datum plane measured in the vertical direction positive upward [L], α is the thickness of the flow domain [L], k_{ij} is the intrinsic conductivity tensor (conductivity of the completely saturated porous medium) [LT^{-1}], and k_r is the relative permeability to water ($0 \leq k_r \leq 1$), which is a function of the water saturation. The specific moisture capacity C, [L^{-1}], is defined by;

$$C = \frac{d\theta}{dh} \tag{5.3}$$

where θ (dimensionless) is the moisture content of the soil. S_s is the specific storage [L^{-1}] and S_w is the water saturation ($0 \leq S_w \leq 1$) (dimensionless). S_s reflects the combined elastic behaviour of the porous medium and the water. This represents the volume of water instantaneously released from storage per unit bulk volume of saturated soil when h is lowered by a unit. q_w is a source/sink term (positive for sources and negative for sinks), that is, the volume of water per unit bulk volume of soil that enters or leaves the system in a unit time.

In this context the air pressure in the unsaturated soil is considered to be constant, and equal to the atmospheric pressure. By definition, we have

$$h = h(\theta) = \frac{(p_w - p_a)}{\gamma_w} = \frac{-p_c(\theta)}{\gamma_w} \tag{5.4}$$

where p_a and p_w are, respectively, the air and water pressures [ML^{-1}T^{-2}], and γ_w is the specific weight of water. p_c is the capillary pressure between air and water [ML^{-1}T^{-2}] and is a function, for each soil, of its moisture content θ. From equation (5.4) we have;

$$p_a = p_w + p_c(\theta) = \overline{p}_a = const \tag{5.5}$$

where $\overline{p}_a$ is the discretised from of p_a, and hence is a volume averaged quantity. From this equation, assuming the hypothesis that the air pressure is constant, the water pressure is directly proportional to the capillary pressure

($p_w = const - p_c$). Therefore, in the partially saturated zones, p_w and h are functions of the water moisture.

In the absence of hysteresis, the water relative permeability is a single value function of θ, i.e.,

$$k_r = k_r(\theta) \tag{5.6}$$

where the specific function is determined by the type of soil.

The water moisture content θ and the water saturation S_w are linked by the relationship

$$\theta = \phi S_w = \theta(h) \tag{5.7}$$

where ϕ is the soil porosity.

As seen from Eq. (5.4), the pressure head h is positive in the saturated zones (where $p_w \geq p_a$) and negative in the unsaturated ones (where $p_w < p_a$). Equations (5.3),(5.6) and (5.7) characterize the soil.

The initial conditions for the pressure head and moisture content are:

$$h(x_i,0) = h_0(x_i) \quad \text{on } \Omega, \tag{5.8}$$

$$\theta(x_i,0) = \theta_0(x_i) \quad \text{on } \Omega, \tag{5.9}$$

The Dirichlet boundary condition for the fixed pressure head is denoted,

$$h(x_i,t) = \tilde{h}(x_i,t) \qquad \text{on } \Gamma \tag{5.10}$$

where $\tilde{h}$ is a specified function and Γ is the boundary of the computational domain.

The Neuman boundary condition describing the fixed water rate flux is

$$-\alpha k_{ij} k_r \left(\frac{\partial h}{\partial x_j} + \frac{\partial y}{\partial x_j} \right) n_i = q_b(x_i,t) \qquad \text{on } \Gamma \tag{5.11}$$

where n_i is the i^{th} component of the outward pointing unit normal to the boundary on which the fixed water rate condition is to be applied, and q_b is a fixed water flow rate [L^2T^{-1}].

5.3 SOLUTION PROCEDURE

By implying that the second mixed derivatives are zero, the expansion of the Richards equation (5.2) on a two dimensional domain Ω gives;

$$\begin{aligned}&\frac{\partial}{\partial x}\left[\alpha k_{xx}k_r\left(\frac{\partial h}{\partial x}+\frac{\partial y}{\partial x}\right)\right]+\frac{\partial}{\partial y}\left[\alpha k_{yy}k_r\left(\frac{\partial h}{\partial y}+\frac{\partial y}{\partial y}\right)\right]-\alpha q_w\\&=\alpha\left(C+S_wS_s\right)\frac{\partial h}{\partial t}\end{aligned}\tag{5.12}$$

Using the fact that;

$$\frac{\partial h}{\partial x_i}+\frac{\partial y}{\partial x_i}=\frac{\partial}{\partial x_i}(h+y)=\frac{\partial \zeta}{\partial x_i}\tag{5.13}$$

Equation (5.12) simplifies to:

$$\begin{aligned}&\frac{\partial}{\partial x}\left[\alpha k_{xx}k_r\left(\frac{\partial \zeta}{\partial x}\right)\right]+\frac{\partial}{\partial y}\left[\alpha k_{yy}k_r\left(\frac{\partial \zeta}{\partial y}\right)\right]-\alpha q_w\\&=\alpha\left(C+S_wS_s\right)\frac{\partial h}{\partial t}\end{aligned}\tag{5.14}$$

The spatial derivative terms in equation (5.14) are discretised by application of Green's theorem, and by the assumption that the spatial derivative is linear over the control volume Ω_m. Integration over the control volume, and the application of the Euler backward difference scheme as discussed in chapter 2 gives for all nodes m in the domain;

$$\begin{aligned}&\frac{\partial}{\partial x}\left[\left(\alpha k_{xx}k_r\right)_{mn}\left(\frac{\partial\zeta}{\partial x}\right)\right]\Bigg|_m \Omega_m + \frac{\partial}{\partial y}\left[\left(\alpha k_{yy}k_r\right)_{mn}\left(\frac{\partial\zeta}{\partial y}\right)\right]\Bigg|_m \Omega_m - \left[\alpha q_w\right]_m \Omega_m\\ &= \alpha_m\left(C+S_wS_s\right)_m \frac{h_m^{n+1}-h_m^n}{\Delta t}\cdot\Omega_m\end{aligned} \qquad (5.15)$$

where h_m^n = pressure head at node m at time n, and h_m^{n+1} = pressure head at node m at time $n+1$. The term $\left(\alpha k_{ij}k_r\right)_{mn}$ is the value of $\alpha k_{ij}k_r$ applied to edge$_{mn}$. Inside this bracket lies the non-linear relative water permeability term $\left(k_r\right)_{mn}$, which is calculated as follows;

$$\left(k_r\right)_{mn} = f\left(\theta_{mn}\right) = f\left(h_{mn}\right) \qquad (5.16)$$

where;

$$h_{mn} = \frac{1}{2}\left(h_m + h_n\right) \qquad (5.17)$$

The specific moisture capacity C at node m, is defined as;

$$C_m = f\left(h_m^n\right) \qquad (5.18)$$

and the calculated water saturation S_w at node m, is given by;

$$\left(S_w\right)_m = {}^{\theta_m}\!/_{\phi} \qquad (5.19)$$

where the moisture content θ_m is defined as:

$$\theta_m = f\left(h_m^n\right) \qquad (5.20)$$

Within the solver, the water moisture, moisture content and moisture capacity for each soil are described by a table of values. These sets of discrete values are then manipulated through linear interpolation.

By equation (5.13), the first derivative terms are described as:

$$\left.\frac{\partial \zeta}{\partial x_i}\right|_m = \frac{\partial h_m^n}{\partial x_i} + \frac{\partial y_m}{\partial x_i} = \frac{\partial}{\partial x_i}\left(h_m^n + y_m\right) \tag{5.21}$$

Re-arranging equation (5.15), gives the linear update for the primary variable h as:

$$\begin{aligned} h_m^{n+1} = & \left\{ \left.\frac{\partial}{\partial x}\left[\left(\alpha k_{xx} k_r\right)_{mn}\left(\frac{\partial \zeta}{\partial x}\right)\right]\right|_m \Omega_m \right. \\ & \left. + \left.\frac{\partial}{\partial y}\left[\left(\alpha k_{yy} k_r\right)_{mn}\left(\frac{\partial \zeta}{\partial y}\right)\right]\right|_m \Omega_m - \left[\alpha q_w\right]_m \Omega_m \right\} \frac{\Delta t}{\alpha_m \left(C + S_w S_s\right)_m} \cdot \frac{1}{\Omega_m} + h_m^n \end{aligned} \tag{5.22}$$

In previous work [27], it has been shown that a linear update for the primary variable has a limited timestep length. Hence a more efficient non-linear explicit primary update algorithm was successfully developed.

The primary variable update, which is the pressure head h in this work, is performed in the following manner. Defining $h_m^{n,k}$ as a value of h for timestep n, iteration k and node m, we initiate each timestep with $h_m^{n+1,1} = h_m^n$ for all nodes. Then, a variable $\bar{h}$ is proposed;

$$\bar{h}_m = \beta h_m^{n+1,k} + (1-\beta) h_m^n \tag{5.23}$$

where $\{\beta \in \mathbb{R} : \beta \in [0,1]\}$. Derivatives and non-linear variables that are functions of h in equation (5.22) are discretized using values of $\bar{h}$, to form the left hand side of the equation.

$$\Re_m = \left\{ \frac{\partial}{\partial x}\left[\left(\alpha k_{xx} k_r\left(\bar{h}_m\right)\right)_{mn} \left(\frac{\partial \bar{\zeta}}{\partial x}\right)\right]\Bigg|_m \Omega_m + \frac{\partial}{\partial y}\left[\left(\alpha k_{yy} k_r\left(\bar{h}_m\right)\right)_{mn} \left(\frac{\partial \bar{\zeta}}{\partial y}\right)\right]\Bigg|_m \Omega_m - \left[\alpha q_w\right]_m \Omega_m \right\} \frac{1}{\Omega_m} \tag{5.24}$$

where,

$$\left.\frac{\partial \bar{\zeta}}{\partial x_i}\right|_m = \left.\frac{\partial \bar{h}}{\partial x_i}\right|_m + \left.\frac{\partial y}{\partial x_i}\right|_m = \left.\frac{\partial}{\partial x_i}\left(\bar{h} + y\right)\right|_m \tag{5.25}$$

Similarly, the variables preceding the discretized time derivative term in equation (5.22) are calculated.

$$\aleph_m = \frac{\Delta t}{\alpha_m \left(C\left(\bar{h}\right) + S_w\left(\bar{h}\right) S_s\right)\Big|_m} \cdot \frac{1}{\Omega_m} \tag{5.26}$$

The primary variable update is then performed by employing equations (5.24) and (5.26) in the following manner.

$$h_m^{n+1,k+1} = \aleph_m \Re_m + h_m^n \tag{5.27}$$

If the system has converged, then one of the following relationships holds true.

$$\left\| h_m^{n+1,k+1} - h_m^{n+1,k} \right\| \leq \varepsilon \tag{5.28}$$

or,

$$\left| \, \left\| h_m^{n+1,k+1} \right\| - \left\| h_m^{n+1,k} \right\| \, \right| \leq \varepsilon \tag{5.29}$$

Where $\{\varepsilon \in \mathbb{R} : \varepsilon \ll 1\}$ is a user defined convergence criterion. The reader should note that (5.28) is more usual, however, in practice, results obtained using (5.29) are sufficiently similar for the problem under consideration that use of it is acceptable. Implementation of this norm also requires less memory and is computationally more efficient. Once convergence is achieved the cycle begins for the next time step with $h_m^n = h_m^{n+1,k+1}$. However, if equation (5.29) is not satisfied, then we apply a successive over relaxation (SOR) parameter $\{\gamma \in \mathbb{R} : \gamma \in [0,1]\}$ such that [1]:

$$\begin{aligned} h_{SOR} = h_m^{n+1,k+1} &= \gamma\left(\aleph_m \Re_m + h_m^n\right) + (1-\gamma)\, h_m^{n+1,k} \\ &= \gamma h_m^{n+1,k+1} + (1-\gamma)\, h_m^{n+1,k} \end{aligned} \tag{5.30}$$

The value of $h_m^{n+1,k+1}$ calculated from equation (5.30) is then used to calculate $\bar{h}_m$ in equation (5.23) on the next iteration, and the process is repeated until convergence is achieved. The values for the non-linear parameters used were $\beta = 0.5$, $\gamma = 1.0$ and $\varepsilon = 1.0 \times 10^{-05}$. These values were decided upon after numerical experiments to determine the optimal values (see section 5.4.2.2). These were used for the analyses in section 5.4.

A flowchart of the solution procedure is given in Figure 5.1, where the symbols correspond to those used in equations (5.12) to (5.30). If we define the primary variable of the problem as $h \in \{h_m^n, \bar{h}_m\}$, then let the set of secondary variables calculated or derived from the primary set be $\mathbb{S}(h)$, where;

$$\mathbb{S}(h) = \{C, \theta, k_r, S_w\} \tag{5.31}$$

Hence in Figure 5.1, the term **CALCULATE** $\mathbb{S}(h)$ implies the calculation of the set of secondary variables using the primary variable h. The terms C, θ and k_r are dependant on the soils used in each test case, where $C = f(h)$, $\theta = f(h)$ and $k_r = f(\theta)$. These terms are calculated through linear interpolation of the tabulated values of the functions (see Figure 5.4, Figure 5.10 and Figure 5.21). The water saturation S_w is calculated directly from equation (5.7).

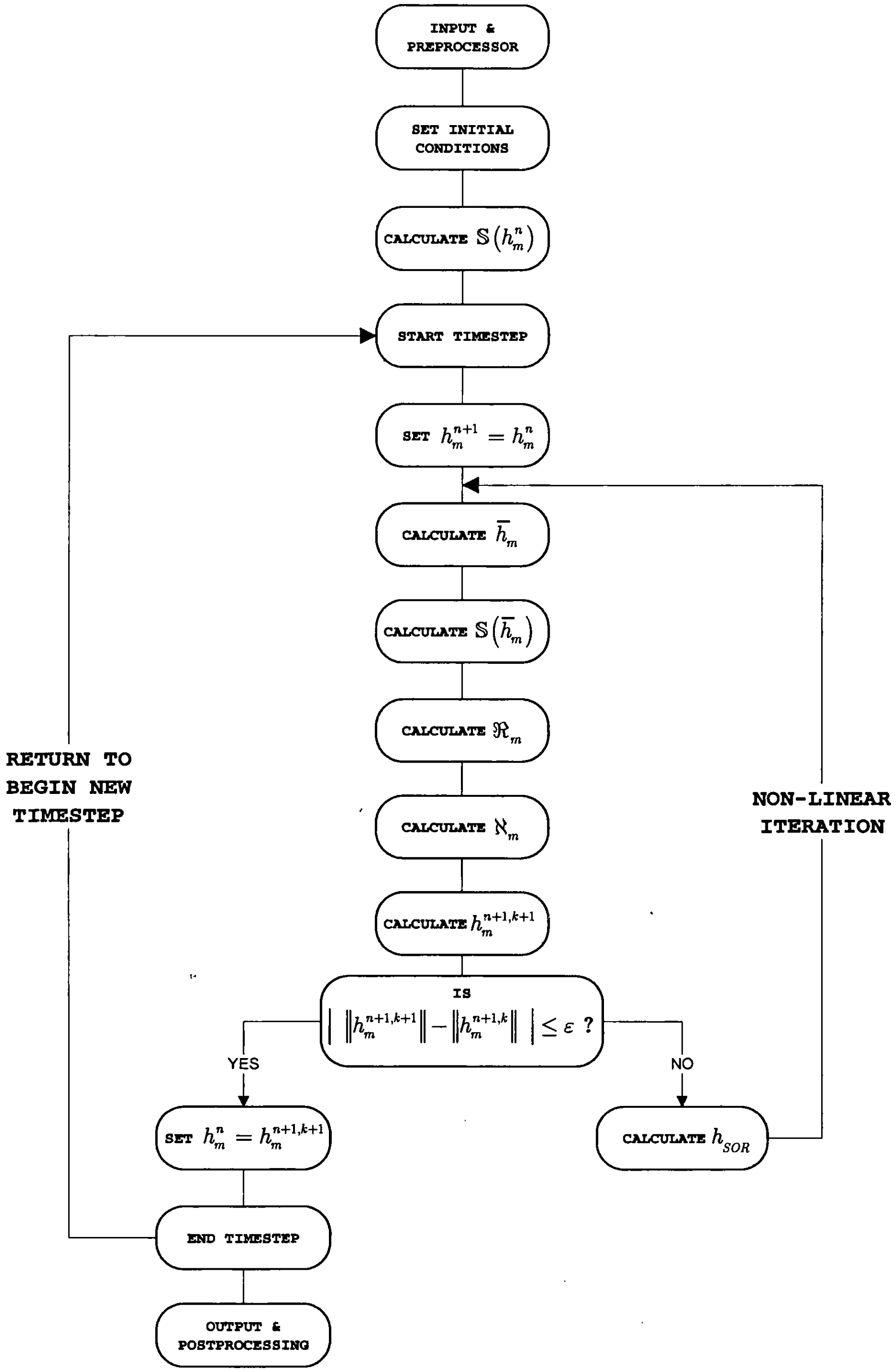


Figure 5.1: Program flow chart for saturated-unsaturated fluid flow in porous media.

5.4 COMPARISON WITH PREVIOUS NUMERICAL SOLUTIONS

In order to evaluate the performance of the edge-based scheme in solving groundwater flow problems, three test cases are presented. In the first test case, the infiltration of water into an initially dry soil column is achieved using a prefixed pressure head at the top of the column. The second test case simulates groundwater flow through a different soil using a prefixed flow rate (see equation (5.11)) at the top of the column. Finally, the third test case models the infiltration flow of water across an initially dry earth dam. The specific material functions $k_r = k_r(\theta)$, $\theta = \theta(h)$ and $C = C(h)$ for each soil are given for each test case.

5.4.1 Test Case 1 - Prefixed Pressure Head

This test case models the infiltration flow of water across an isotropic-homogeneous silt column 75 m in depth, and 10 m in width shown in Figure 5.2 along with the labels A, B, C, and D to distinguish between boundaries. Initially, a mesh independent solution study was done. The domain was discretised using three meshes of varying densities as shown in Figure 5.3. Mesh 1 was discretised using structured triangular elements, with 3 nodes along boundary A-B, and 19 nodes along boundary C-D. Mesh 2 was discretised using unstructured triangular elements, with 5 nodes along boundary A-B, and 40 nodes along boundary C-D. Mesh 3 was discretised using unstructured triangular elements, with 10 nodes along boundary A-B, and 60 nodes along boundary C-D. A comparison of the number of nodes and elements present in each mesh is tabulated in Table 5.1.

	N°. Nodes	*N°. Elements*
Mesh 1	57	72
Mesh 2	450	812
Mesh 3	1010	1882

Table 5.1: Data concerning the meshes used in the mesh independent solution study for Test Case 1.

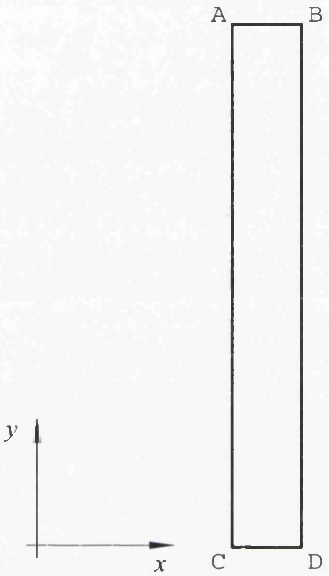


Figure 5.2: The soil column 75 m in depth, and 10 m in width along with the labels A, B, C, and D to distinguish between boundaries used in Test Case 1.

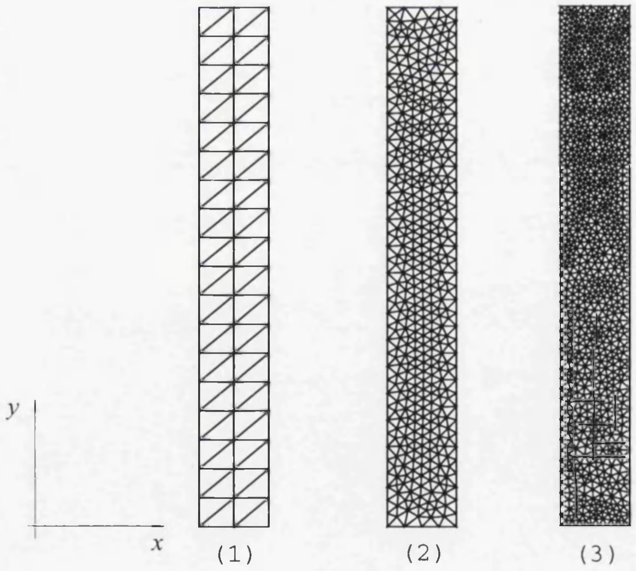


Figure 5.3: Three meshes of various densities used in the mesh independent solution study for Test Case 1.

The initial and boundary conditions for the simulation were the initial pressure head corresponding to the residual water saturation ($h_r = -1000$ m) for all nodes at time zero, a prefixed pressure head ($h_1 = -1\ m$) on side A-B and a residual water saturation ($h_r = -1000$ m) on side C-D, that is;

$$h(x,y,0) = -1000 \text{ m} \tag{5.32}$$

$$h(x,0,t) = -1 \text{ m} \tag{5.33}$$

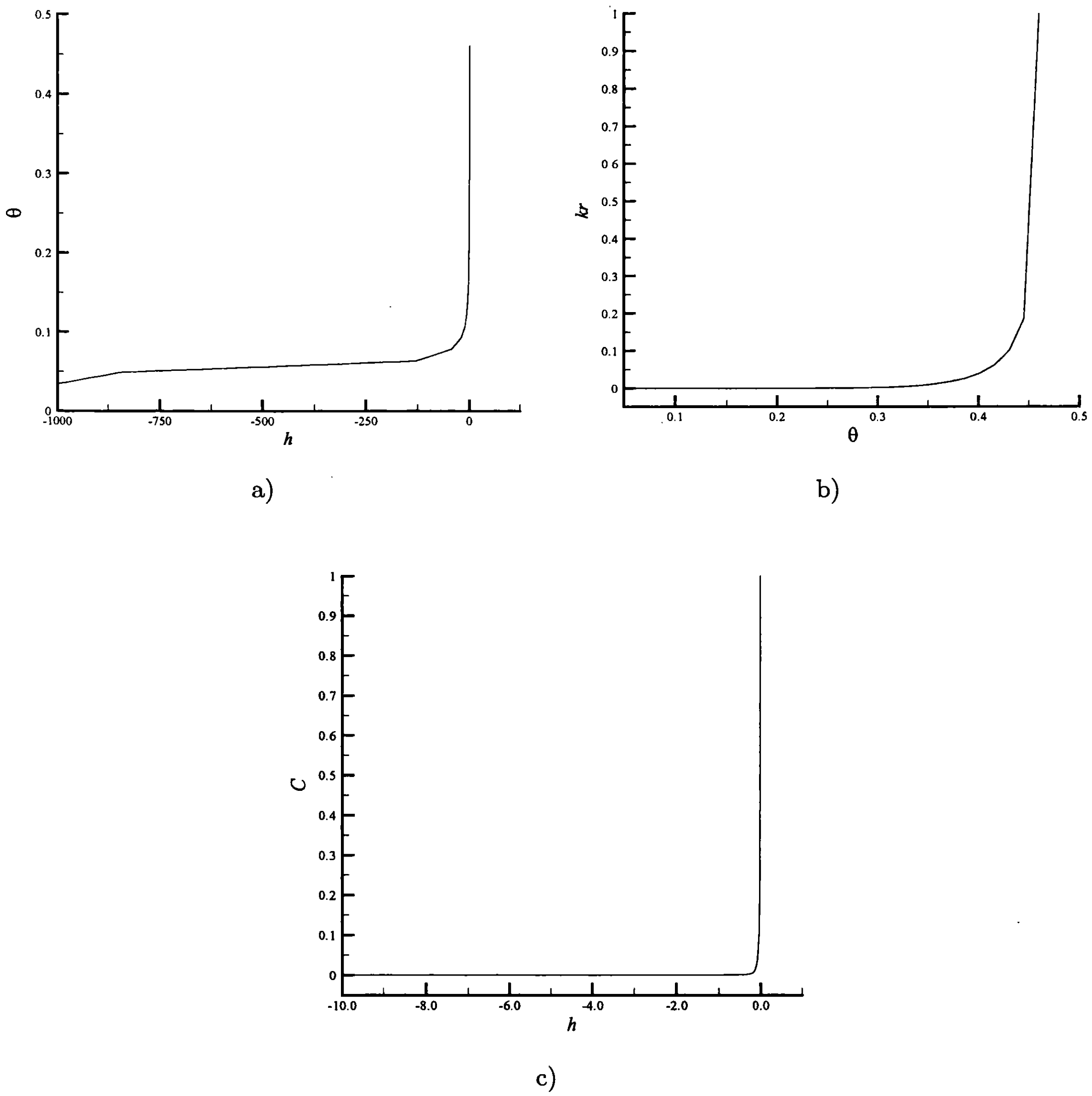
$$h(x,75,t) = -1000 \text{ m} \tag{5.34}$$

$k_x\left(ms^{-1}\right)$	$k_y\left(ms^{-1}\right)$	ϕ	$S_s\left(m^{-1}\right)$
1.515×10^{-05}	1.515×10^{-05}	0.46	1.0×10^{-05}

Table 5.2: Material properties for silt used in Test Case 1.

The highly non-linear specific material functions $k_r = k_r(\theta)$, $\theta = \theta(h)$ and $C = C(h)$ for the material used in this test are shown in Figure 5.4, while the other material properties are given in Table 5.2. Within the solver, each soil property shown in Figure 5.4 is described by a table of published values [12]. These sets of discrete values are then manipulated through linear interpolation.

The compact stencil scheme was employed to solve the governing equations of the problem and the solution obtained for the front position at 5 days was calculated for the three meshes. The results of the mesh independent solution study is shown in Figure 5.5. This shows profile of the calculated pressure head h along the line $x = 5.0$ m, $y \in [0, 75.0]$m through the domain at 5 days using mesh 1, 2 and 3. As expected, it is evident from the solution that the finer the mesh density, the water front position is calculated more efficiently and accurately. The front calculated using the coarse mesh 1 is spread across a vertical distance of 9.4 metres. Mesh 2 displays a front position at 5 days which is much sharper and more defined than that seen with mesh 1. Here, the front is captured within a vertical distance of 2.3 metres. The front position calculated on mesh 3 is sharper still, and is captured within a vertical distance of 0.9 metres. The simulation time for this 5-day run on all thee meshes was relatively short, with the longest time (9 minutes) being taken to calculate the solution on the denser mesh 3. This simulation time is acceptable, and hence it was decided to use mesh 3 for the validation of the scheme.



a)

b)

c)

Figure 5.4: Graphs of the material property non-linear variation of a) pressure head h against moisture content θ , b) moisture content θ against the relative permeability to water k_r , and c) pressure head h against moisture capacity C for the material silt used in Test Case 1.

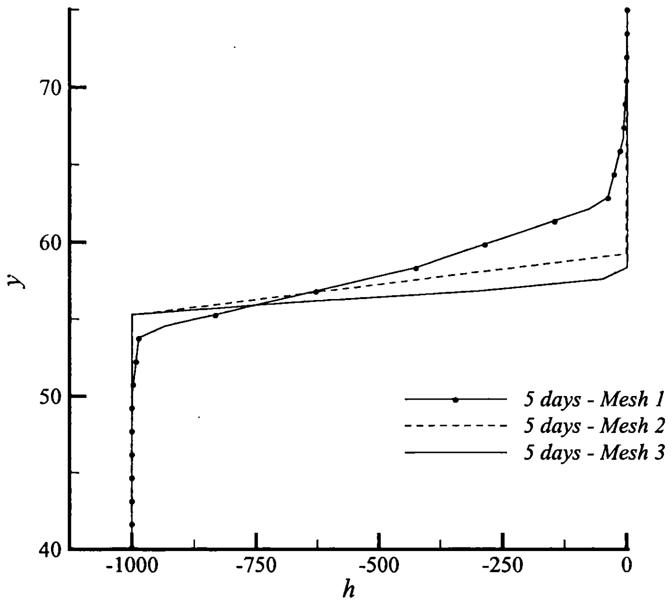


Figure 5.5: A profile of the calculated pressure head h along the line $x = 5.0$ m, $y \in [0, 75.0]$m through the domain at 5 days for Test Case 1, using mesh 1, 2 and 3.

For Test Case 1, the compact stencil scheme was employed to solve the governing equations of the problem and the solution obtained for the front position at 5, 10 and 15 days against the solution obtained using the FE method obtained by Gottardi [12]. The total simulation time for this test case was 15 days. The results obtained for the moisture content θ, water saturation S_w and the pressure head h along the line $x = 5.0$ m, $y \in [0, 75.0]$m through the domain at 5, 10 and 15 days are shown in Figure 5.6, Figure 5.7 and Figure 5.8.

Figure 5.6 displays the profile of the water saturation S_w together with the solution obtained by Gottardi [12] using the FE method. It is clearly shown that the computed solution captures the front much more sharply than the FE solution. The FE mesh used by Gottardi was much coarser than the mesh used (mesh 3 in Figure 5.3), containing 120 triangular elements and 78 nodes, and so this result is expected.

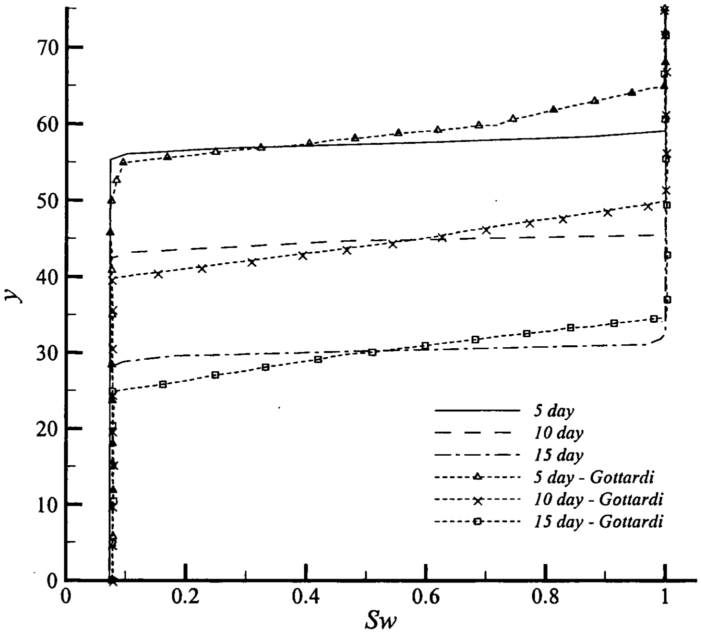


Figure 5.6: A profile of the calculated water saturation S_w along the line $x = 5.0$ m , $y \in [0, 75.0]$m through the domain at 5, 10 and 15 days using mesh 3, compared to Gottardi's solution for Test Case 1.

Figure 5.7 shows the profile of the pressure head h, and Figure 5.8 shows the profile of the moisture content θ along the line $x = 5.0$ m through the domain at 5, 10 and 15 days. Again, the front is sharp, and the transition between the totally saturated and partially saturated regions is well defined as is expected from the saturation plot in Figure 5.6. However, no published solutions were available to compare against for these two plots.

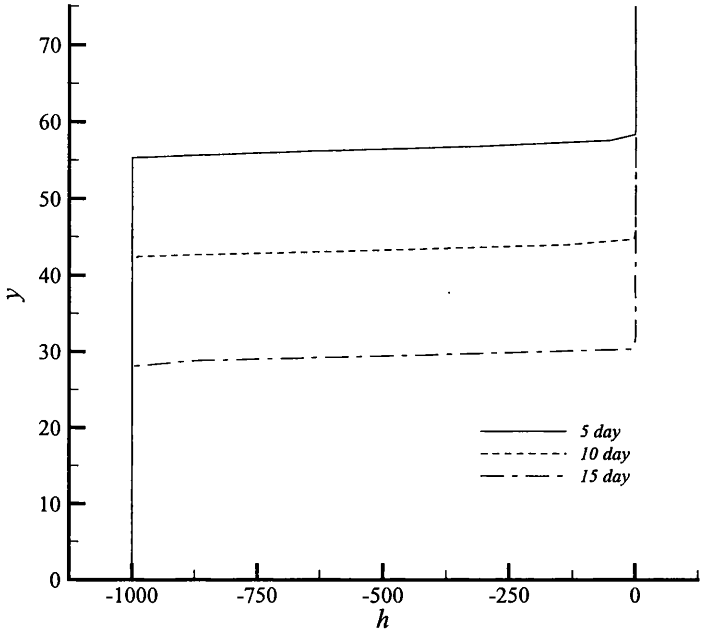


Figure 5.7: A profile of the calculated pressure head h along the line $x = 5.0$ m, $y \in [0, 75.0]$m through the domain at 5, 10 and 15 days using mesh 3 for Test Case 1.

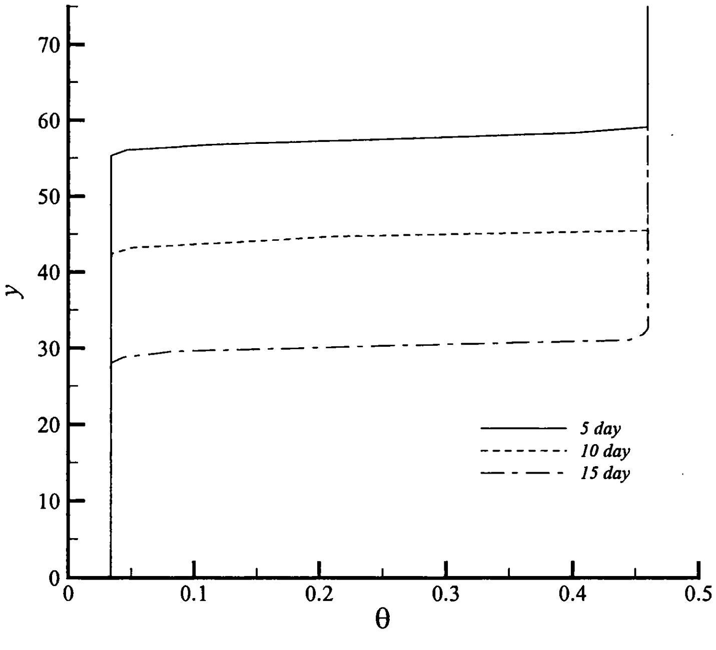


Figure 5.8: A profile of the moisture content θ along the line $x = 5.0$ m, $y \in [0,75.0]$m through the domain at 5, 10 and 15 days using mesh 3 for Test Case 1.

5.4.2 Test Case 2 - Prefixed Flow Rate

This test case considers a prefixed flow rate applied to the top of the soil column to model the infiltration of surface water into the ground. An investigation is undertaken in section 5.4.2.2 to determine the optimal values of the non-linear parameters $(\beta,\gamma,\varepsilon)$ to be used in equations (5.23), (5.29) and (5.30). Once these values are determined, a 10-day simulation was undertaken using both the standard and compact stencil finite volume schemes.

The infiltration flow of water across an isotropic-homogeneous sand column 75 m in depth, and 10 m in width is modelled by applying a boundary condition of constant prefixed specific water rate (q_1) along boundary A-B. The domain was discretised using a structured quadrilateral element mesh (891 elements and 1000 nodes), and is shown in Figure 5.9 along with the labels A, B, C, and D to distinguish between boundaries. For this test case, the standard finite volume and the compact stencil scheme were employed to solve the governing equations, and the solutions obtained from each scheme compared against a FE solution obtained by Gottardi [13]. The initial and boundary conditions for the simulation were,

$$h(x,y,0) = -1.0 \text{ m} \tag{5.35}$$

$$q(x,0,t) = 10 \text{ m}^3/\text{day} \tag{5.36}$$

$$h(x,0,t) = -1.0 \text{ m} \tag{5.37}$$

that is, the initial pressure head corresponding to the residual water saturation for all nodes at time $t = 0$, a Neuman boundary condition of constant prefixed specific water rate applied to side A-B to simulate the infiltration of water into the soil and finally a Dirichlet boundary condition of prefixed pressure head is applied to side C-D. For consistency of units, the domain is assumed to have unit thickness. The material properties for sand were applied to the soil column in question. These properties are shown in Table 5.3 and the non-linear variation of k_r, θ and C are shown in Figure 5.10.

$k_x\left(ms^{-1}\right)$	$k_y\left(ms^{-1}\right)$	ϕ	$S_s\left(m^{-1}\right)$
9.4398×10^{-05}	9.4398×10^{-05}	0.43	1.0×10^{-05}

Table 5.3: Material properties for sand used in Test Case 2.

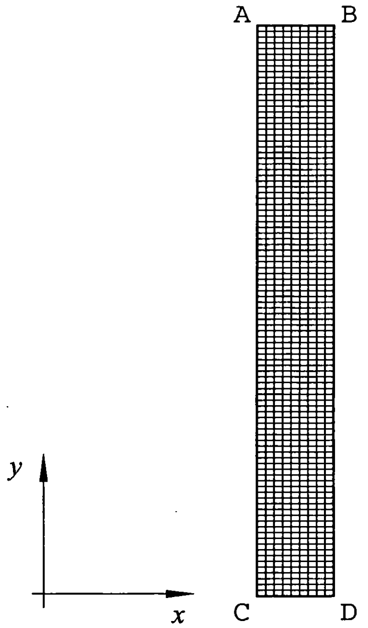
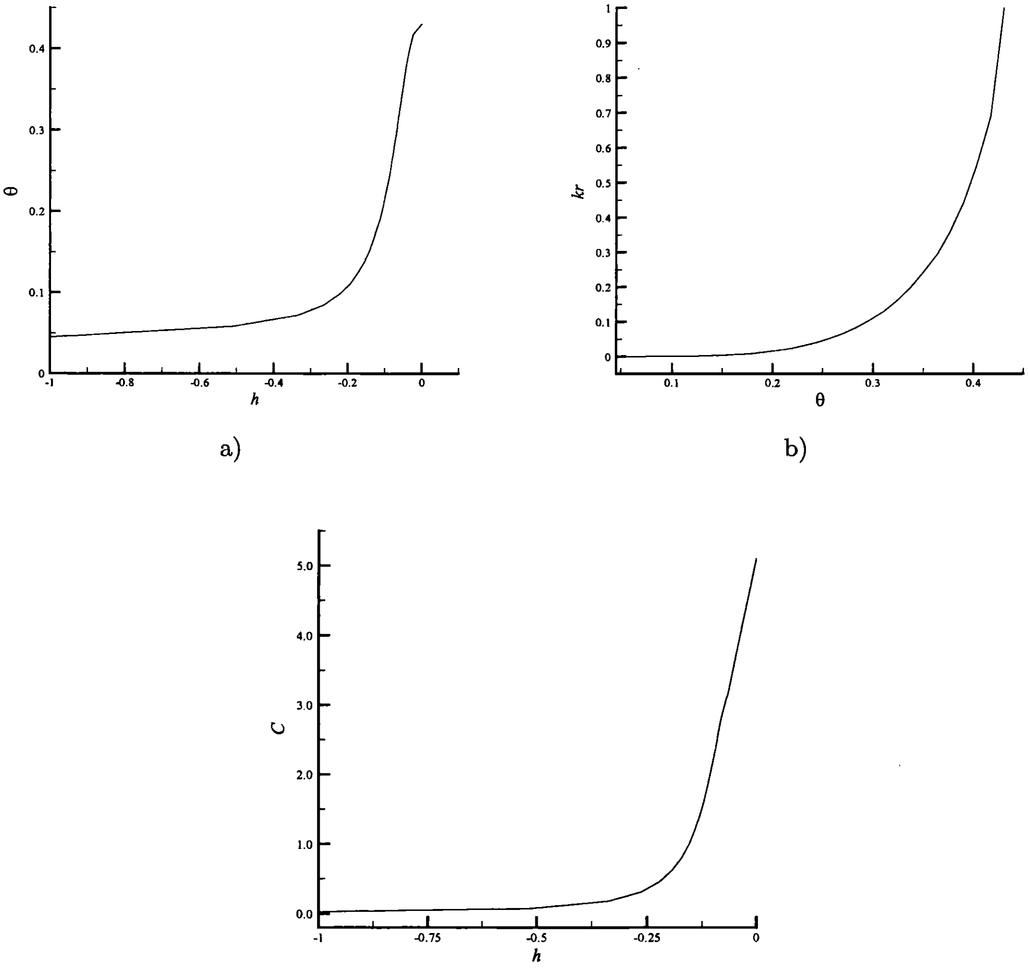


Figure 5.9: The structured grid used for the groundwater flow problem described in Test Case 2 with boundary labels A, B, C and D included.

Firstly, numerical experiments are conducted in sections 5.4.2.1 to 5.4.2.4 to determine the optimal values of the non-linear parameters $(\beta,\gamma,\varepsilon)$ to be used in equations (5.23), (5.29) and (5.30).



a)

b)

c)

Figure 5.10: Graphs of the material property non-linear variation of a) pressure head h against moisture content θ , b) moisture content θ against the relative permeability to water k_r , and c) pressure head h against moisture capacity C for the material sand used in Test Case 2.

5.4.2.1 Reference Solution

Initially, the primary variable update for the pressure head h was performed in a simple linear fashion using equation (5.22) only. The maximum timestep length which produced a stable solution using this linear update model was $dt = 1.0$ second. In Figure 5.11, the profile of the water saturation along the soil column (Sand) for $t = 50{,}000$ seconds using a timestep of 1.0 second is shown. This graph was obtained by plotting the water saturations along the line $x = 5.0$ m, $y \in [0, 75.0]$m running through the domain at the above-mentioned time. As this solution was obtained using a small timestep with a linear variable update, this was used as the reference solution for the numerical experiments that followed.

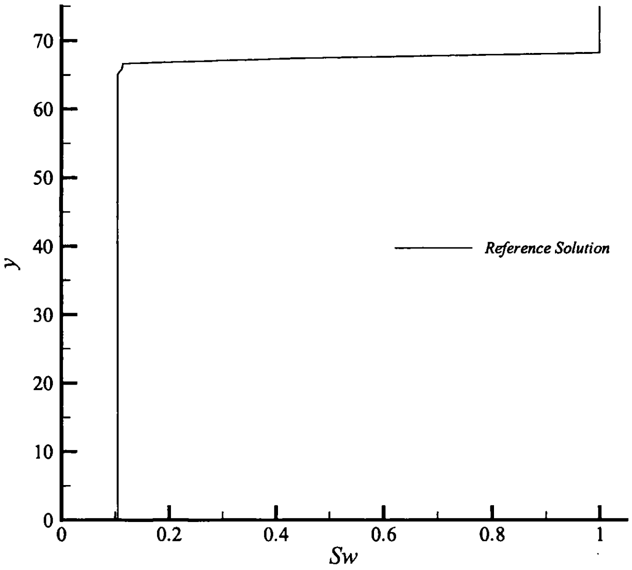


Figure 5.11: Water saturation profile at $t = 50{,}000$ seconds along the line $x = 0.5$ with $dt = 1.0$ second.

5.4.2.2 Investigation Of Non-Linear Parameters At Small Timestep Lengths

Having established a reference solution in the above work, variation in the non-linear parameters can be investigated. The objective of this part of the study is to establish optimal values for the non-linear parameters without causing degradation in the solution. Three values of β, γ and ε were chosen and all possible combinations (27 runs) were investigated. The test case was identical to that given above, with an analysis time of 50,000 seconds and a timestep of 1.0 second. The parameters and the corresponding total iteration count are given in Table 5.4. Note that in the table, the set of non-linear parameters that results in both the maximum and minimum iteration counts are highlighted.

The level of error in each of these solutions has to be established, and in all cases the final front position at time $t = 50{,}000$ seconds is identical. From the results shown, it appears that the optimal values, that is the non-linear parameters which produce a good solution with the lowest iteration count, are given by Run 7. These non-linear parameters are $\beta = 0.5$, $\gamma = 1.0$ and $\varepsilon = 1.0 \times 10^{-05}$.

5.4.2.3 Effect Of Time Step Length On Iteration Count

The objective of this stage is to retain the optimal non-linear parameters given above, and extend the time step length to find a maximum value for dt which gives an acceptable solution to the problem. The runs attempted, and the final iteration count for each, are given in Table 5.5. Speedup is defined as the ratio of the total number of iterations for an analysis to the iteration count for the smallest timestep size.

It can be clearly seen that as the timestep length increases the total iterations required decreases. However, beyond a particular step length, the solutions are not stable and a solution cannot be reached. Figure 5.12 shows the number of iterations per step for the $dt = 50.0$ second case, and it can be seen that although a few steps require a significant number of iterations, most converge in less than ten iterations.

Figure 5.13 shows the change in convergence criteria between iterations (see equation (5.29)). In most cases convergence is fast; however some timesteps require a few more steps before a suitable solution is reached. Trends are visible in this type of plot but do not appear to be conclusive.

Run #	β	γ	ε	Total No. Iterations
1	0.5	0.8	1.0E-05	127802
2	0.5	0.8	1.0E-08	342912
3	0.5	0.8	1.0E-12	629759
4	0.5	0.9	1.0E-05	114516
5	0.5	0.9	1.0E-08	264142
6	0.5	0.9	1.0E-12	464635
7	***0.5***	***1.0***	***1.0E-05***	***97307***
8	0.5	1.0	1.0E-08	136637
9	0.5	1.0	1.0E-12	208029
10	0.666	0.8	1.0E-05	127826
11	0.666	0.8	1.0E-08	342968
12	0.666	0.8	1.0E-12	629835
13	0.666	0.9	1.0E-05	114526
14	0.666	0.9	1.0E-08	264102
15	0.666	0.9	1.0E-12	464857
16	0.666	1.0	1.0E-05	97327
17	0.666	1.0	1.0E-08	138565
18	0.666	1.0	1.0E-12	215008
19	1.0	0.8	1.0E-05	127865
20	1.0	0.8	1.0E-08	342904
21	***1.0***	***0.8***	***1.0E-12***	***630282***
22	1.0	0.9	1.0E-05	114530
23	1.0	0.9	1.0E-08	263873
24	1.0	0.9	1.0E-12	465448
25	1.0	1.0	1.0E-05	97365
26	1.0	1.0	1.0E-08	142459
27	1.0	1.0	1.0E-12	227899

Table 5.4: Investigation of non-linear parameters at $dt = 1.0$ second.

dt (s)	No. Time Steps	Did Code Run?	Total No. Iterations	Speedup
1	50,000	YES	97,307	1.0
10	5,000	YES	11,103	8.8
50	1,000	YES	3,623	26.9
80	625	STOP STEP 370	N/A	N/A
100	500	STOP STEP 130	N/A	N/A

Table 5.5: Effect of time step length on iteration count and speedup.

In order to check the accuracy of the solution obtained, values along the centre vertical line of the domain ($x = 5$ m) at $t = 50{,}000$ seconds for the three time step lengths which produced a suitable solution is shown in Figure 5.14. It can be clearly seen that for the stable solutions the results are also accurate, and the increase in timestep length has no effect on the quality of the solution obtained.

5.4.2.4 Investigation Into The Influence Of Gamma

Careful consideration of Table 5.4 shows that the value of γ has the largest influence on the stability of the scheme and the number of non-linear iterations required to obtain a solution. Therefore, for large timesteps it is relevant to investigate changes in γ. With a timestep length of 50 seconds, seven runs were undertaken with γ values from 0.4 to 0.9. For these runs $\beta = 0.5$ and $\varepsilon = 1.0 \times 10^{-05}$ were used. A graph showing the total iteration count for this case is given in Figure 5.15. As expected, iteration count decreases with increasing gamma until an unstable point is reached for γ values greater than 0.8. This shows that a trial and error approach can give a stable solution for large timesteps.

For a timestep length of 50 seconds, a maximum speedup of 4.83 was achieved when $\gamma = 0.8$ (as compared to the reference solution). However, careful reference to Table 5.5 shows that more improvement in runtime can be achieved by a slight reduction in the timestep length and the use of standard values of the non-linear parameters rather than searching for optimal γ at the largest possible value of dt.

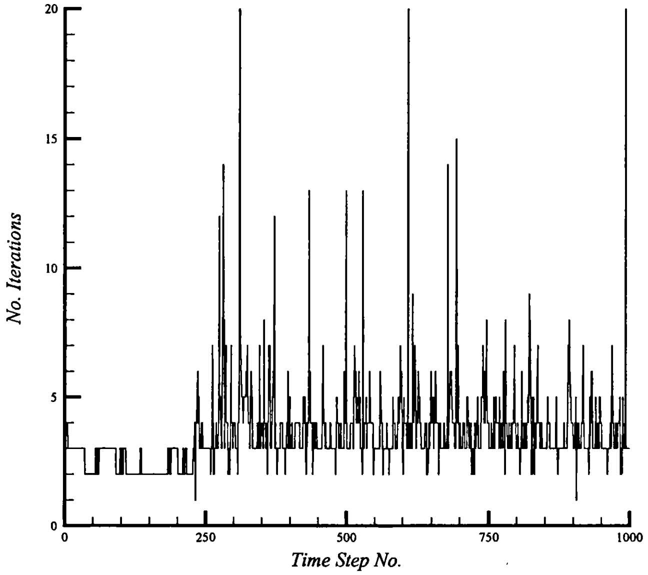


Figure 5.12: A plot of the number of iterations per timestep required for $dt = 50.0$ second, using the non-linear variables of Run 7.

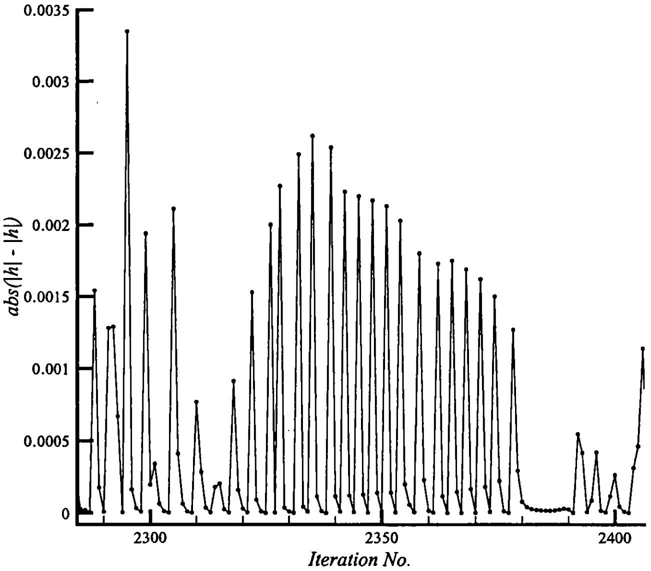


Figure 5.13: A plot showing the relationship between $\mathrm{abs}\left(\left\|h^{n+1,k+1}\right\| - \left\|h^{n+1,k}\right\|\right)$ and the total iteration number between 2300 and 2400, using the non-linear variables of Run 7 and $dt = 50.0$ second.

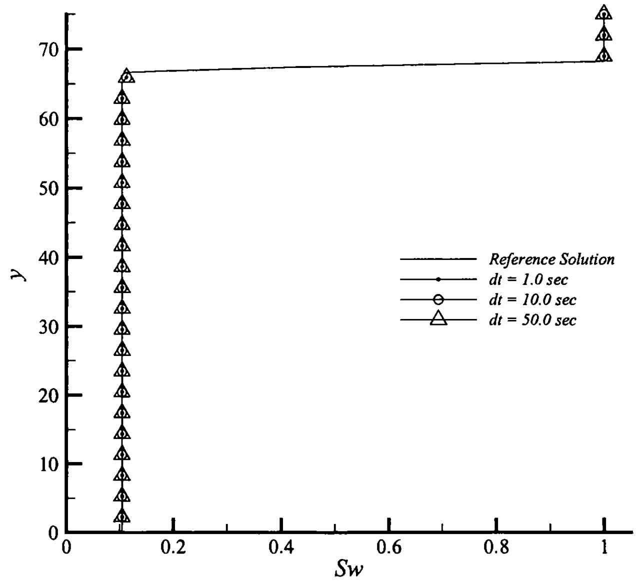


Figure 5.14: Water saturation profile at t = 50,000 seconds along the line $x = 0.5$ with $dt = 1$, 10, 50 second compared to the reference solution, using the non-linear variables of Run 7.

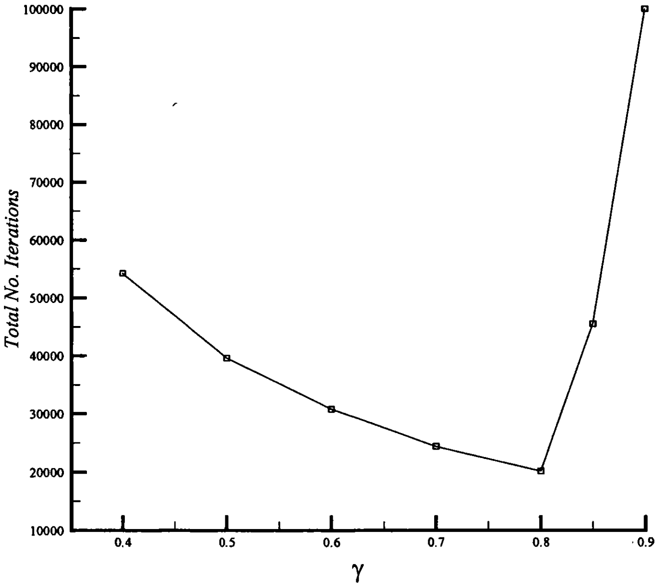


Figure 5.15: A graph showing the effect of γ on the total iteration count.

5.4.2.5 Investigation of the accuracy of the optimal non-linear parameters on a full solution.

The results shown in Figure 5.16, Figure 5.17 and Figure 5.18 are plots along the line $x = 5.0$ m through the domain at 2, 5 and 10 days respectively. Figure 5.16 displays the profile of the pressure head h along this line at the above mentioned times calculated using the standard finite volume scheme and the compact stencil scheme along with the solution obtained by Gottardi [13] using the FE method. Again, as was evident from Test Case 1, there is considerably less smearing of the front with the finite volume methods as compared to the FE solution. However, Figure 5.16 shows that the compact stencil method captures the pressure head front more efficiently than the standard finite volume method, and thus demonstrates the benefit of applying the three-node stencil to capture non-linear variations in the primary variable as compared to the standard five-node stencil.

Similar results are shown in Figure 5.17 and Figure 5.18. Figure 5.17 shows a comparison of the solution obtained for the water saturation S_w along the line $x = 5.0$ m through the domain obtained using the compact stencil scheme and the standard finite volume scheme at 5, 10 and 15 days compared to Gottardi's FE solution. The compact stencil solution again produces a sharper front than the standard finite volume solution, with the compact stencil solution in advance of the standard finite volume solution by 1.5 - 2 metres. Similarly for Figure 5.18, which shows a comparison of the solution obtained for the moisture content θ along the line $x = 5.0$ m through the domain obtained using the compact stencil scheme and the standard finite volume scheme at 2, 5 and 10 days compared to Gottardi's solution.

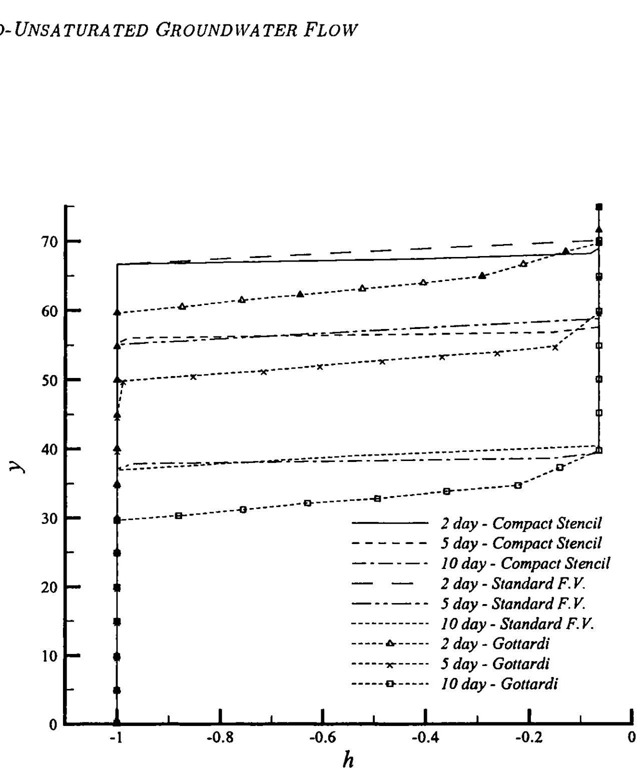


Figure 5.16: A comparison of the solution obtained for the pressure head h along the line $x = 5.0$ m, $y \in [0, 75.0]$m through the domain obtained using the compact stencil scheme and the standard finite volume scheme at 2, 5 and 10 days compared to Gottardi's solution for Test Case 2. Solutions obtained using the non-linear variables of Run 7 and $dt = 50.0$ second.

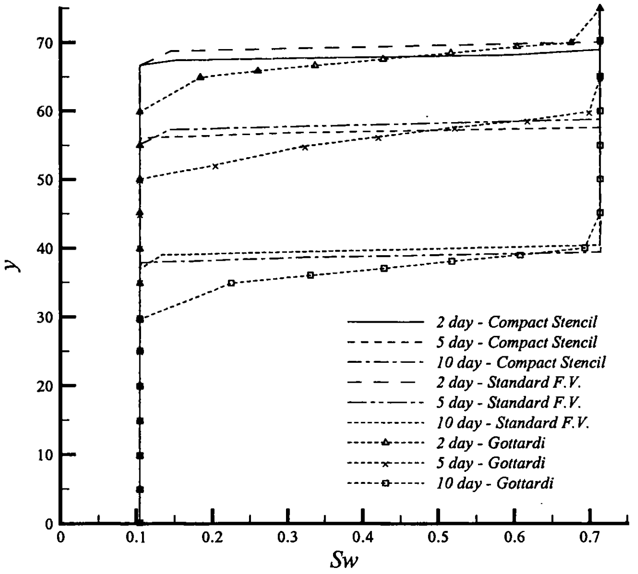


Figure 5.17: A comparison of the solution obtained for the water saturation S_w along the line $x = 5.0$ m, $y \in [0, 75.0]$m through the domain obtained using the compact stencil scheme and the standard finite volume scheme at 2, 5 and 10 days compared to Gottardi's solution for Test Case 2. Solutions obtained using the non-linear variables of Run 7 and $dt = 50.0$ second.

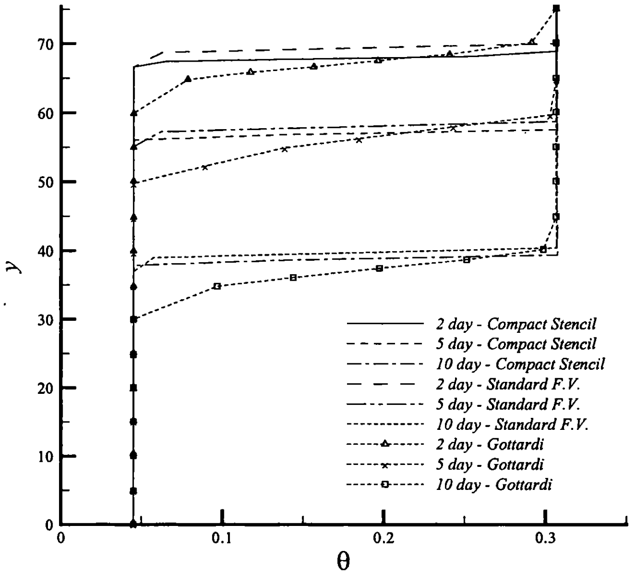


Figure 5.18: A comparison of the solution obtained for the moisture content θ along the line $x = 5.0$ m, $y \in [0, 75.0]$m through the domain obtained using the compact stencil scheme and the standard finite volume scheme at 2, 5 and 10 days compared to Gottardi's solution for Test Case 2. Solutions obtained using the non-linear variables of Run 7 and $dt = 50.0$ second.

5.4.3 Test Case 3 – Infiltration Flow Across An Earth Dam

This test case models the infiltration flow of water across an initially dry earth dam. Earthfill dams for the storage of water have been used since the early days of civilisation. Today, as in the past, the earthfill dam continues to be the most common type of small dam, principally because its construction involves utilisation of materials in their natural state with a minimum of processing. Soils placed in earth dams should be sufficiently impervious in order to reduce seepage to a manageable rate, and be sufficiently stable to provide sturdy side batters to the structure. The type of dam embankment will depend on the quantities of suitable soils available. Generally there are two types of small dam embankments;

1) homogeneous, and,
2) zoned.

A purely homogeneous type of dam is composed of a single kind of material. However, the most common type of earth dam section is that in which an impervious core is flanked by zones of materials considerably more pervious. The pervious zones enclose, support and protect the impervious core; the upstream pervious zone affords stability against rapid drawdown; and the downstream pervious zone acts as a drain to control the line of seepage. The zoned embankment consists of a central clay core with more pervious material forming a shell on each side. Some of the benefits of a zoned dam can also be obtained structurally when the soil is placed selectively or when different construction methods are employed in different portions of the embankment. In the first case, a central clay core with more pervious material forming a shell on each side is constructed. In the second case, zones of lower permeability are created in otherwise homogeneous embankments by using either more compaction or a higher construction water content such as the Tres Marias Dam in Brazil ([33],[24]). Using more compaction will make a considerable difference in the permeability of some materials, especially of residual soils and other materials that break down as they are being rolled. Using a higher content of construction water to create zones of lower permeability is a method chosen most frequently when the embankment materials are clayey soils. Additional water during compaction makes these soils considerably more impervious than they are when compacted on the dry side of optimum water content (at the same dry density), and the difference persists with time. The difference in permeability, which may be 10 to 100 times, is due to an actual difference in the structure of the compacted soil ([16],[17]). The saturated-unsaturated non-linear model developed in this work is able to model easily these changes in construction methods and permeability and this section sets out to investigate these phenomena. The following test case simulates the transient infiltration flow of water across the earth dam whose boundary is shown in Figure 5.19, for a sudden rising of the water level to point B.

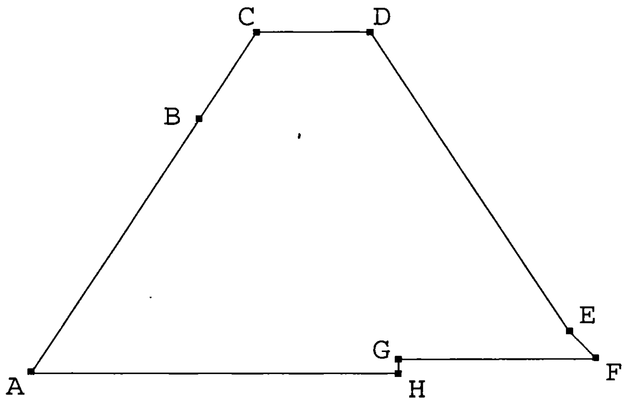


Figure 5.19 : The boundary of the earth dam used in Test Case 3 along with the positions of markers.

The coordinates of the eight points indicated in Figure 5.19 lying on the dam boundary are shown in Table 5.6.

	Coordinates (m)	
	x	*y*
A	0.0	0.0
B	6.0	9.0
C	8.0	12.0
D	12.0	12.0
E	19.0	1.5
F	20.0	0.5
G	13.0	0.5
H	13.0	0.0

Table 5.6: Coordinates of boundary markers for the earth dam boundary.

An internal drain is an essential feature of all but the smallest dams unless the downstream is so pervious that it acts as a drain. The purpose is firstly to carry away any seepage that goes through the core or the cut-off and secondly to prevent the saturation of the upper part of the downstream shell by rain or spray falling on the dam [31]. A horizontal drainage blanket is applied to the dam cross-section along the boundary connecting points G and F in Figure 5.19. This type of drain is widely used in dams of moderate height.

The initial and boundary conditions used for this test case were;

$$h(x,y,0) = -3000.0\,\mathrm{m} \tag{5.38}$$

$$h(x,y,t) = -y\gamma_w \qquad \text{on side A-B} \tag{5.39}$$

$$h(x,y,t) = 0 \qquad \text{if } q_w > 0 \text{ on side F-G} \tag{5.40}$$

$$q_w = 0 \qquad \text{on sides B-C, C-D, D-E, E-F, G-H and A-H} \tag{5.41}$$

In these equations γ_w is the specific weight of water, and q_w is the specific flow rate of water on the seepage surface F-G of the dam drain. The water level (point B) was taken as being the datum for y. The flow domain was discretised in an unstructured manner using triangular elements by using 398 nodes and 730 elements as shown in Figure 5.20.

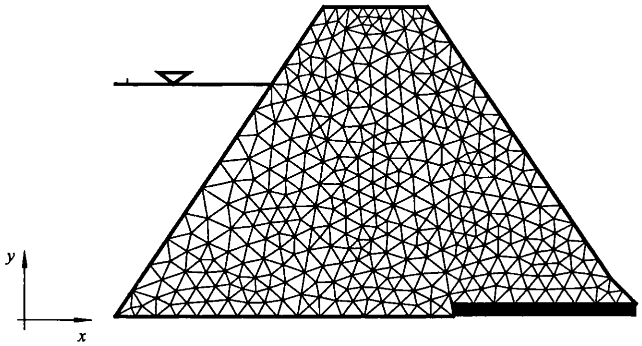
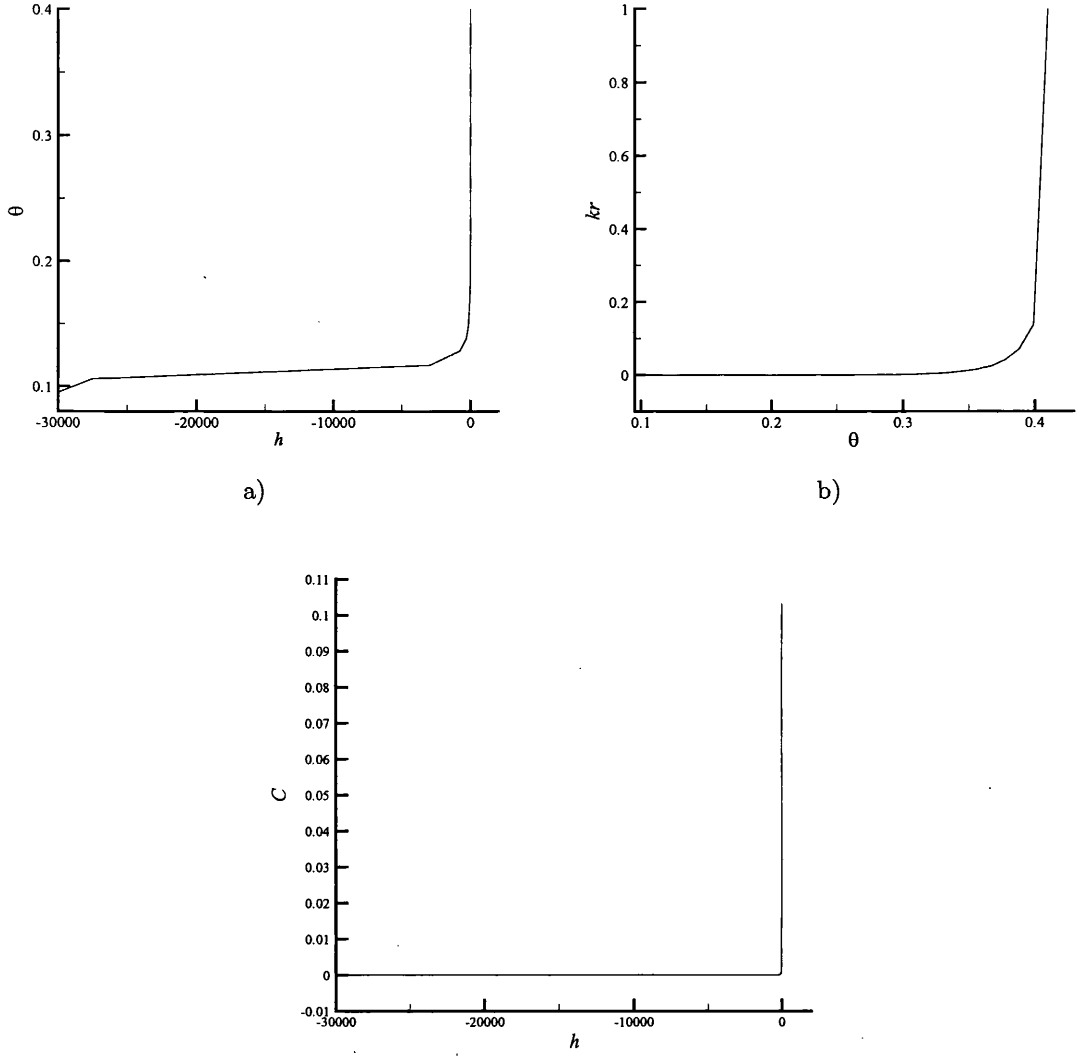


Figure 5.20 : The unstructured grid of 398 nodes and 730 elements.

The material properties for clay loam [12] were applied to the discretised earth dam in Figure 5.20. These properties are shown in Table 5.7 and the specific material functions $k_r = k_r(\theta)$, $\theta = \theta(h)$ and $C = C(h)$ for the material are given in Figure 5.21.

$k_x\left(ms^{-1}\right)$	$k_y\left(ms^{-1}\right)$	ϕ	$S_s\left(m^{-1}\right)$
1.2303×10^{-07}	1.2303×10^{-07}	0.41	1.0×10^{-05}

Table 5.7: Material properties for clay loam.



a)

b)

c)

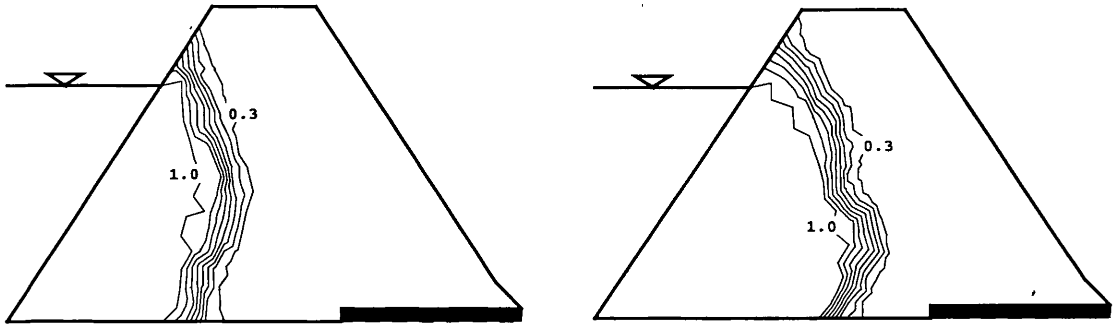
Figure 5.21: Graphs of the material property non-linear variation of a) pressure head h against moisture content θ , b) moisture content θ against the relative permeability to water k_r , and c) pressure head h against moisture capacity C for the material clay loam used in Test Case 3.

5.4.3.1 Homogeneous Section

The first model presented here is of the dam with a homogeneous section of clay loam. Contour plots of the water saturation, S_w, and the pressure head, h, through the section after 100, 200, 300, 400, 500 and 600 days are shown in Figure 5.22 and Figure 5.23 respectively. While Figure 5.24 shows the pressure head and water saturation profiles along the line $y = 6.0$ m through a homogeneous dam section at 100, 200, 300, 400, 500 and 600 days.

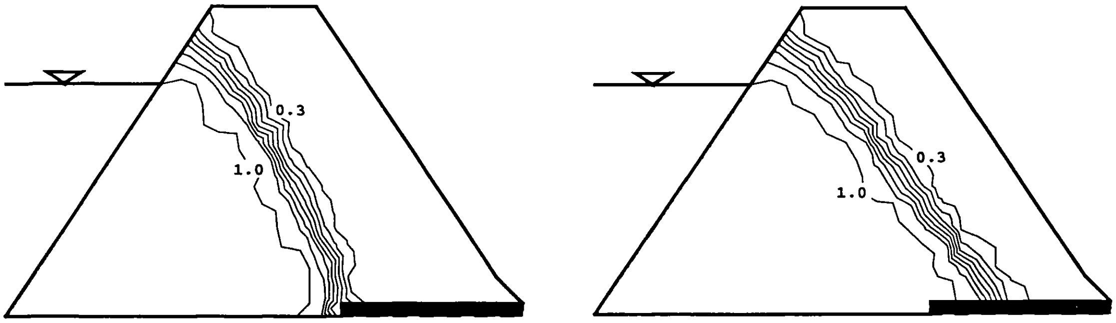
As can be seen from the water saturation profiles in Figure 5.22, the homogeneous clay loam dam provides little resistance to the seepage water. Hence, a purely homogeneous clay-loam earthfill dam is of little practical use. However, a steady state solution is reached after approximately 500 days. This shows that the seepage water is being drained from the system proving that the horizontal drainage blanket has been effectively modelled. Thus the saturation front is unable to reach the downstream dam boundary. This observation is supported by the pressure head contours shown in Figure 5.23.

The water saturation plot shown in Figure 5.24b illustrate the front position at each specified time. As can be seen, there is little smearing of the front, and is most cases the front is captured within a distance of less than 1 metre. The front position is equally spaced for the first 400 days, with the front position at 500 and 600 days lying close together. Again, this is directly due to the influence of the downstream drain at later times. Similar deductions can be made concerning the pressure head front shown in Figure 5.24a. As the unstructured mesh used for this problem was relatively coarse, the ability of the scheme to capture such sharp fronts on coarse meshes is undoubtedly a positive outcome. As already shown in the mesh density investigation of section 5.4.1, the sharpness of the front increases with the density of the mesh.



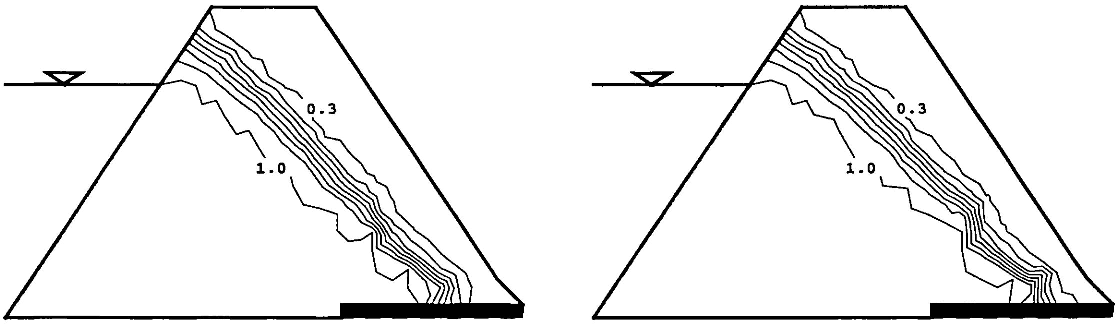
a) 100 days

b) 200 days



c) 300 days

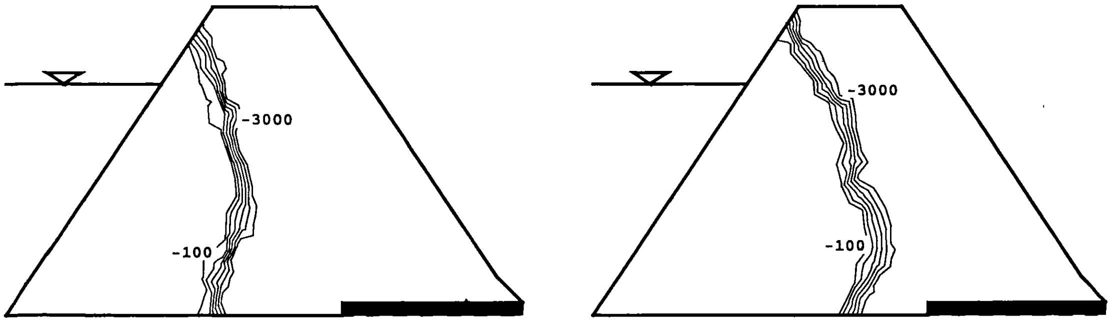
d) 400 days



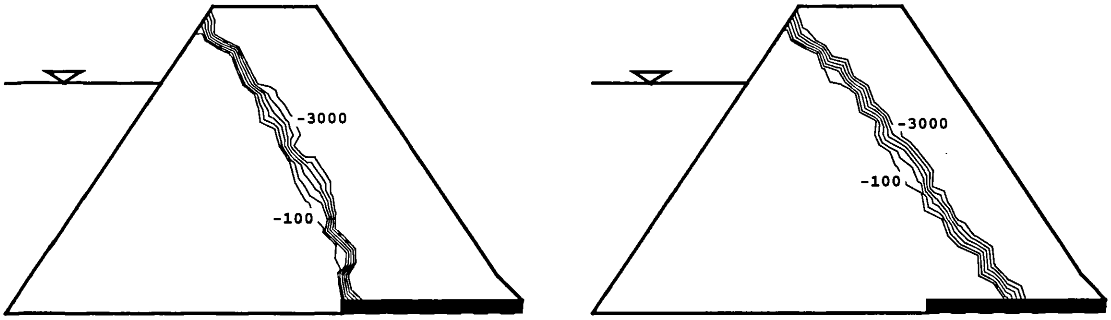
e) 500 days

f) 600 days

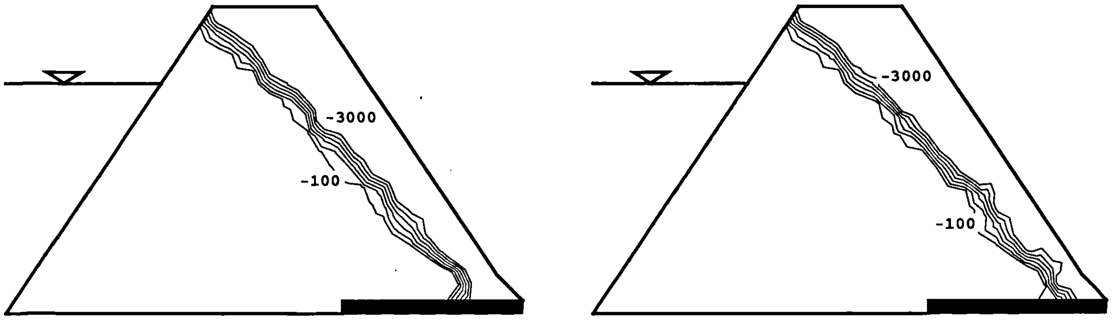
Figure 5.22: Water saturation, S_w, contours (0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0) across a homogeneous clay loam dam after a)100, b)200, c)300, d)400, e)500 and f)600 days respectively.



a) 100 days b) 200 days

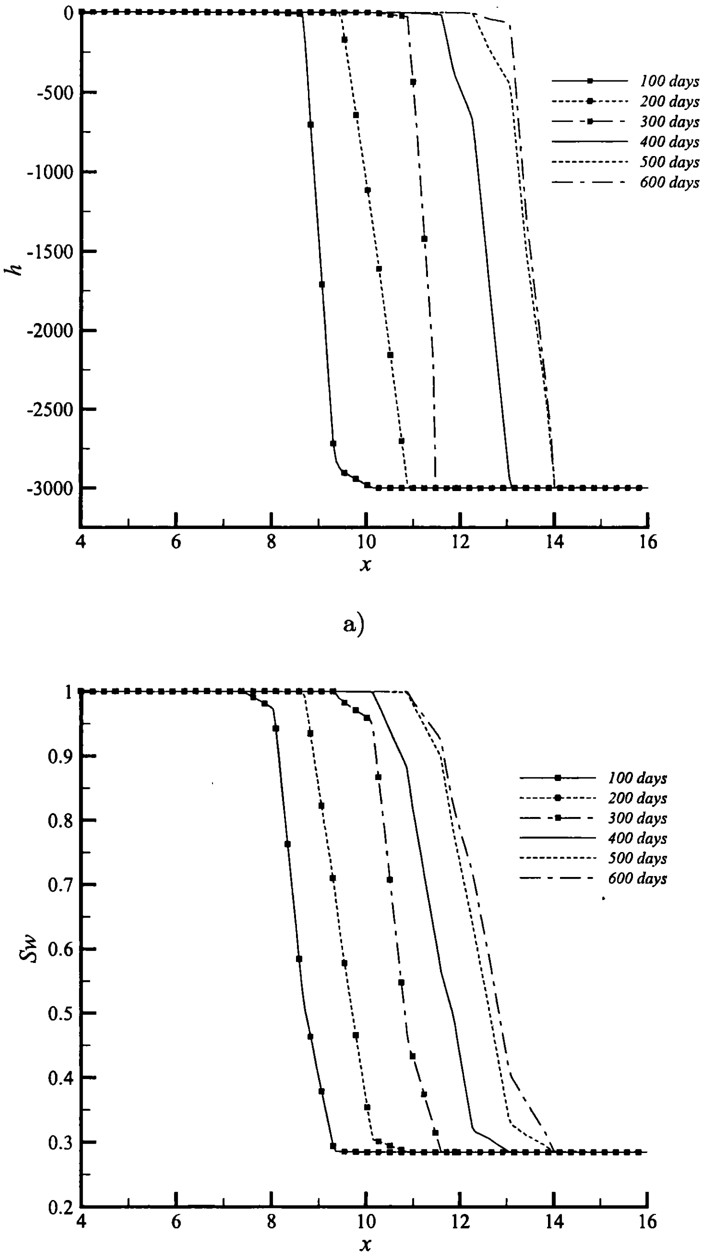


c) 300 days d) 400 days



e) 500 days f) 600 days

Figure 5.23: Pressure head, *h*, contours (-3000, -2500, -2000, -1500, -1000, -500 and –100m) across a homogeneous clay loam dam after a)100, b)200, c)300, d)400, e)500 and f)600 days respectively.



a)

b)

Figure 5.24: a) pressure head, *h*, and b) water saturation, S_w, profiles along the line $y = 6.0$ m, $x \in [4,16]$m through a homogeneous clay loam dam using the compact stencil scheme at 100, 200, 300, 400, 500 and 600 days.

5.4.3.2 Non-Homogeneous Section - Sloped Clay Loam Core

The simulation discussed in the previous section where a homogeneous dam was considered was repeated using a zoned embankment type dam constructed of clay loam with a core permeability of a) 1/10th, and b) 1/100th, that of clay loam. The core is flanked by clay loam with material properties as tabulated in Table 5.7 and Figure 5.22. Other than the permeability, the core had material properties identical to that of clay loam. Contour plots of the water saturation, S_w, through the zoned dam section after 100, 200, 300, 400, 500 and 600 days are shown in Figure 5.25 and Figure 5.28, while pressure head, h, contours at the same times through the dam section are shown in Figure 5.26 and Figure 5.29. Figure 5.25 and Figure 5.26 shows the results obtained with a core permeability 1/10th that of clay loam, while Figure 5.28 and Figure 5.29 show the results obtained with a core permeability 1/100th that of clay loam. A dashed line has been superimposed onto the contour plots to indicate the boundary of the sloping core of the earthfill dam. Figure 5.27 and Figure 5.30 shows the pressure head and water saturation profiles along the line $y = 6.0$ m through the zoned dam section with core permeability 1/10th and 1/100th that of the remainder of the dam material respectively, at 100, 200, 300, 400, 500 and 600 days.

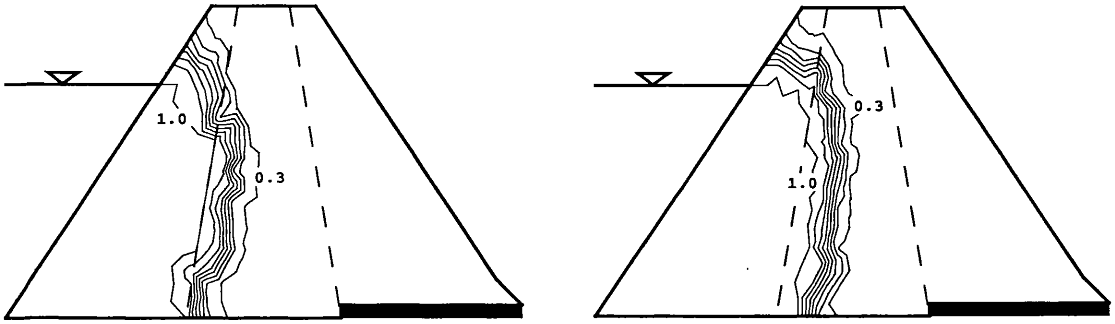
When comparing the results obtained for the water saturation and pressure head contours on both zoned dam sections (in Figure 5.25, Figure 5.28, Figure 5.26 and Figure 5.29) to the homogeneous dam results (Figure 5.22 and Figure 5.23), the benefit of the former is evident. By compacting the core material, thus reducing its permeability, the resistance to the flow of seepage water through the dam embankment is beyond question. For the case shown in Figure 5.25, where the core has a permeability 1/10th that of clay loam, the result after 600 days shows clearly that the water saturation front has yet to emerge through the downstream boundary of the core. Compare this with the result obtained for the same time but on a homogeneous dam (Figure 5.22), where the saturation front had reached the far side of the horizontal drainage blanket by the same time. While the results shown in Figure 5.28, where the core had a permeability 1/100th that of clay loam, shows that the front position after 100, 200, 300, 400, 500 and 600 days is similar. The front position at 100 days is at the dam core, and after 600 days the front has shown little movement from this position. The core is virtually impenetrable to the seepage, thus holding the front back to a greater degree than the other two cases presented in Test Case 3.

The effect of the compacted core on the ease of travel of the seepage water is evident from inspection of Figure 5.24, Figure 5.27 and Figure 5.30 which show the water saturation and pressure head profiles along the line $y = 6.0$ m through the dam section. For all cases, the water saturation and pressure head front position at 100 days is approximately 9 metres. However; this is where the similarity in the results end. For the homogeneous section (Figure 5.24), the water saturation and pressure head front at 600 days is at 14 metres. The zoned dam section with core permeability 1/10th that of the remainder (Figure 5.27) has a water saturation and pressure head front position at 600 days of 12 and 11 m respectively. Finally, the zoned dam section with core permeability 1/100th that of the remainder (Figure 5.30) has a water saturation and pressure head front position at 600 days of 10 and 9.5 metres respectively. These plots clearly illustrate the benefits of the zoned section dams in preventing the flow of water as compared to the homogeneous dams.

Plots of the pressure head, h, and water saturation, S_w, at the centre point of the dam section (coordinate (10,6) metres) against time for the three dam sections considered are shown in Figure 5.31. The chosen point lies along the centre line of the dam section, and hence along the centre line of the dam core. The calculated solution for the pressure head at this point is considered first (see Figure 5.31a). The plot for the homogeneous dam section shows a rapid increase in the pressure head at this point, with the calculated value increasing by 3000m within $t = 100$ and 200 days. While the plot at the same point for the zoned dam section with core permeability 1/100th that of the remainder shows no increase in the pressure head until $t = 450$ days, then a gradual increase is observed over the remainder of the simulation time. These two results are as expected, however, the results obtained for the pressure head for the zoned dam section with core permeability 1/10th that of the remainder is curious. It would not be unreasonable to assume that this plot would display a trend lying between the two mentioned plots, however, this is not the case. For approximately the first 200 days, the plot closely follows the homogeneous section solution, however, the onset of the pressure increase is delayed somewhat. Between 200 and 400 days, the solution contains a temporal oscillation, before once more agreeing with the homogenous section solution for $t > 400$ days. This under-prediction of the solution is most probably due to slight spatial oscillations in the solution at this point. This is evident on inspection of Figure 5.26a and b, where there exists oscillations in the front position at 200 days (Figure 5.26a) which are not present at 30 days (Figure 5.26b). Inspection of the unstructured mesh used for this test case (Figure 5.20) shows a definite coarseness of the mesh in the core region as compared to the flanks, and in all probability this oscillation is mesh dependent.

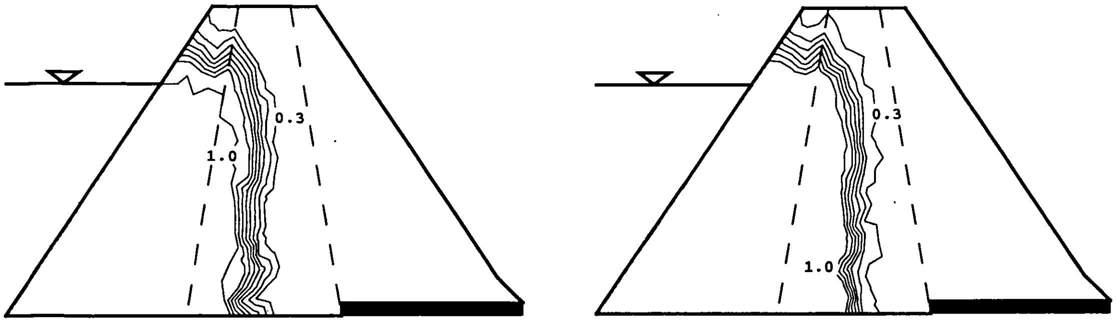
The water saturation plot for the three dam sections at the coordinate (10,6) metres is illustrated in Figure 5.31b. Here, the plots for the homogeneous section and the less permeable zoned dam show similar trends to the pressure head curves in Figure 5.31a, with the zoned dam showing little change from the initial conditions hence the front has not passed the point plotted here. In this case, the plot for the zoned dam section with core permeability 1/10th that of the remainder shows a gradual increases in the water saturation between $t = 150$ and 600 days as compared to the homogeneous case. This is an indication of the ease with which the front passes through the point in the two cases in question; the zoned dam offering considerably more resistance to the flow. In this plot, the large temporal oscillation in the pressure head solution shows itself to a lesser degree.

It must be noted that this section does not consider the dam material stability issues relating to the different construction methods, and the build up of water in front of the dam core.



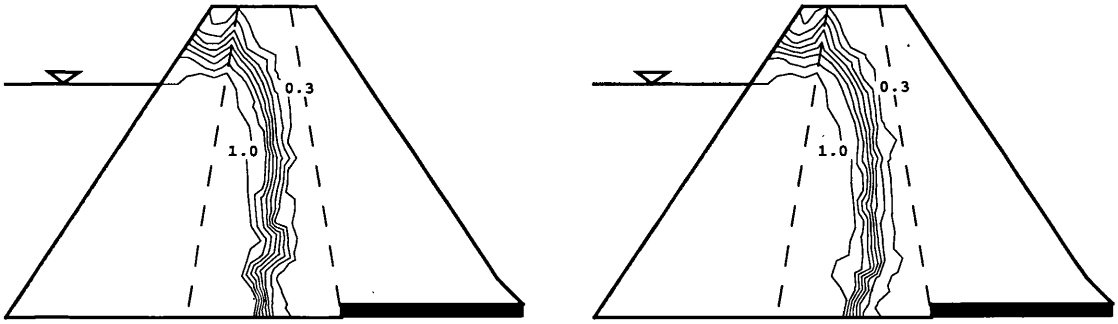
a) 100 days

b) 200 days



c) 300 days

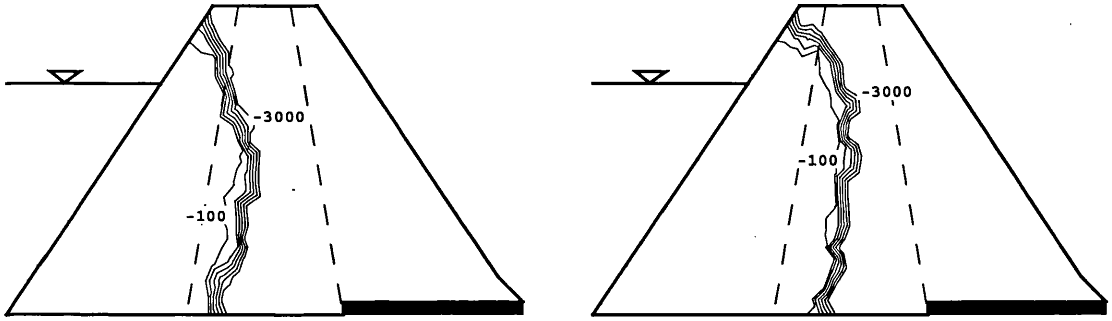
d) 400 days



e) 500 days

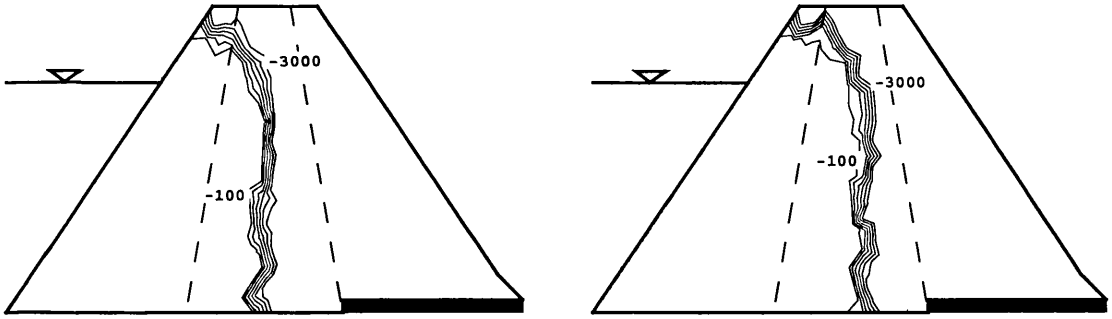
f) 600 days

Figure 5.25: Water saturation, S_w, profiles (0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0) across a zoned clay loam dam with a compacted core after a)100, b)200, c)300, d)400, e)500 and f)600 days respectively. Core permeability is 1/10th that of the remainder of the dam material.



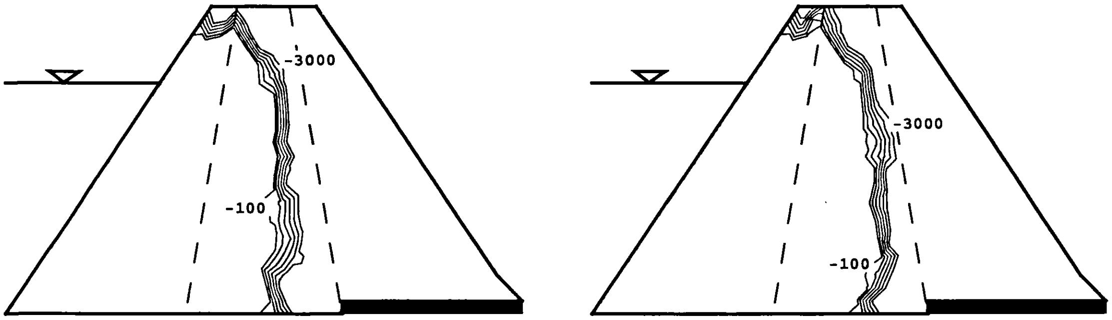
a) 100 days

b) 200 days



c) 300 days

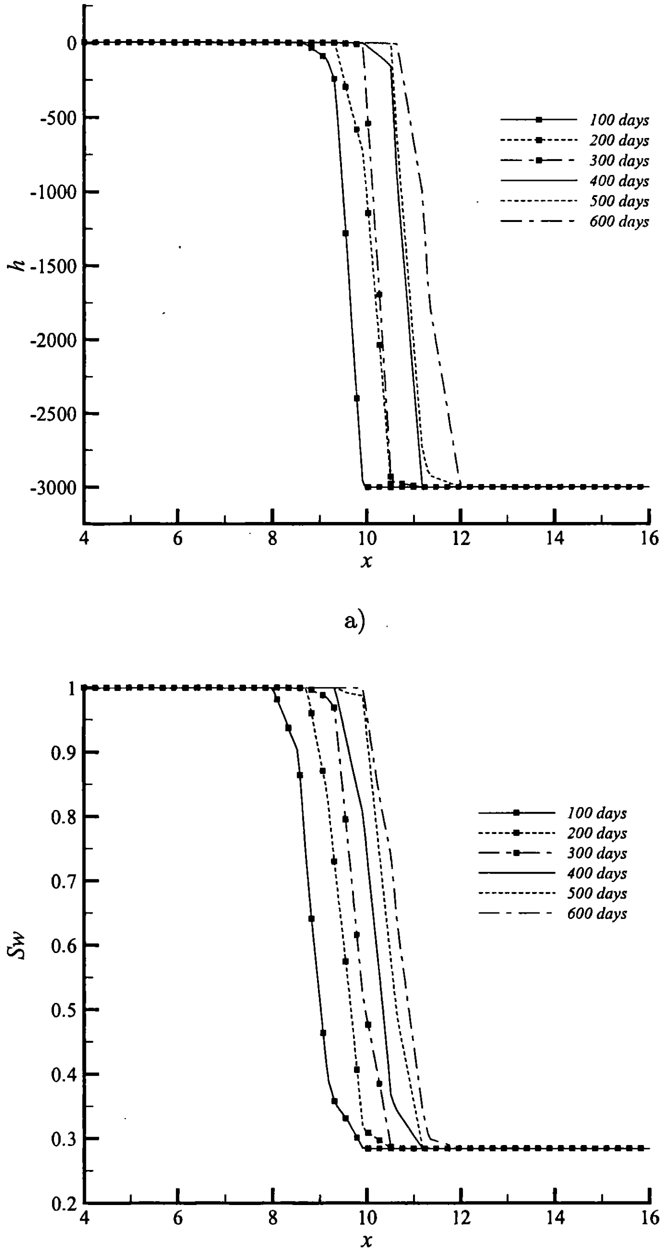
d) 400 days



e) 500 days

f) 600 days

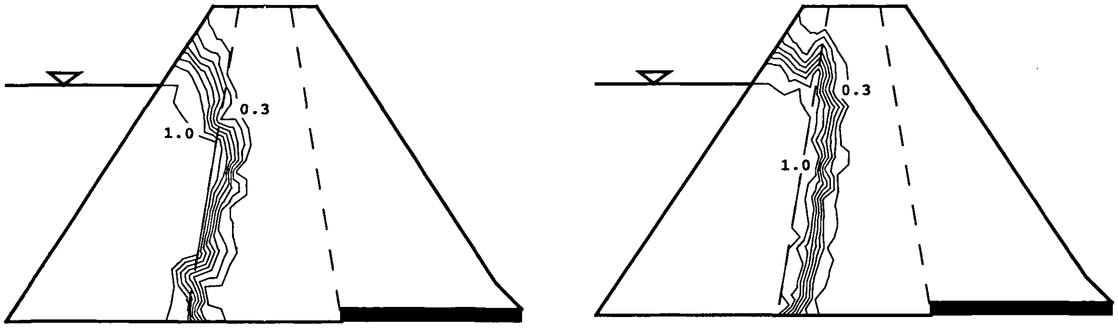
Figure 5.26: Pressure head, h, contours (-3000, -2500, -2000, -1500, -1000, -500 and –100m) across a zoned clay loam dam with a compacted core after a)100, b)200, c)300, d)400, e)500 and f)600 days respectively. Core permeability is 1/10$^{\text{th}}$ that of the remainder of the dam material.



a)

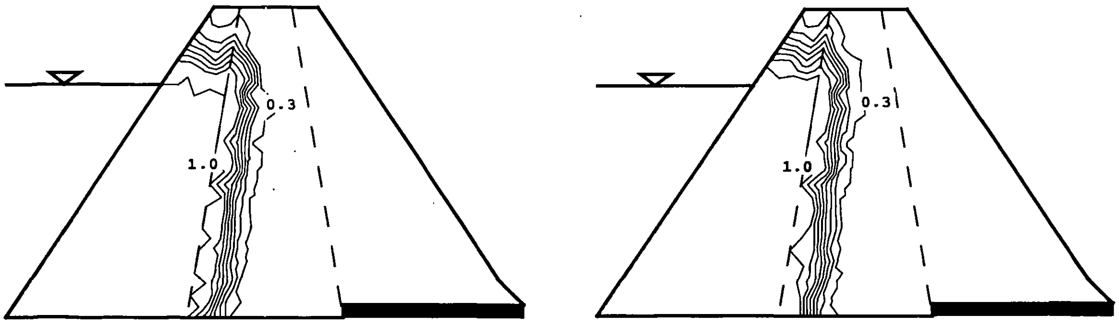
b)

Figure 5.27: a) pressure head, *h*, and b) water saturation, S_w , profiles along the line $y = 6.0$ m, $x \in [4,16]$m through a zoned clay loam dam using the compact stencil scheme at 100, 200, 300, 400, 500 and 600 days. Core permeability is $1/10^{\text{th}}$ that of the remainder of the dam material.



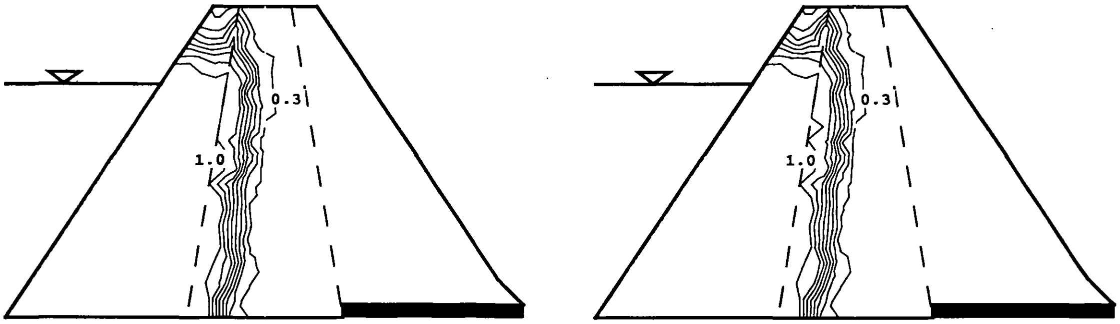
a) 100 days

b) 200 days



c) 300 days

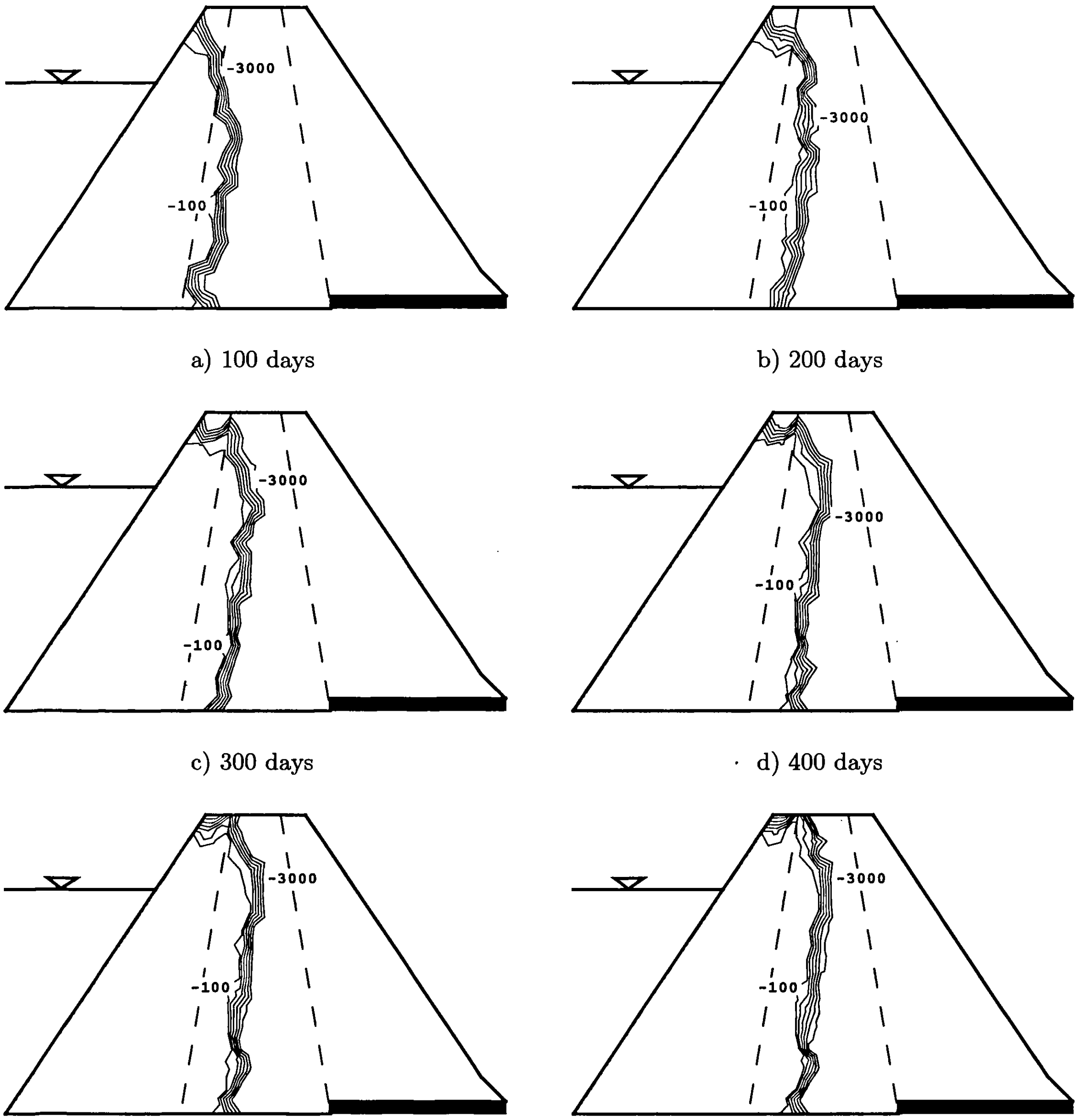
d) 400 days



e) 500 days

f) 600 days

Figure 5.28: Water saturation, S_w, profiles (0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0) across a zoned clay loam dam with a compacted core after a)100, b)200, c)300, d)400, e)500 and f)600 days respectively. Core permeability is 1/100$^{\text{th}}$ that of the remainder of the dam material.



a) 100 days

b) 200 days

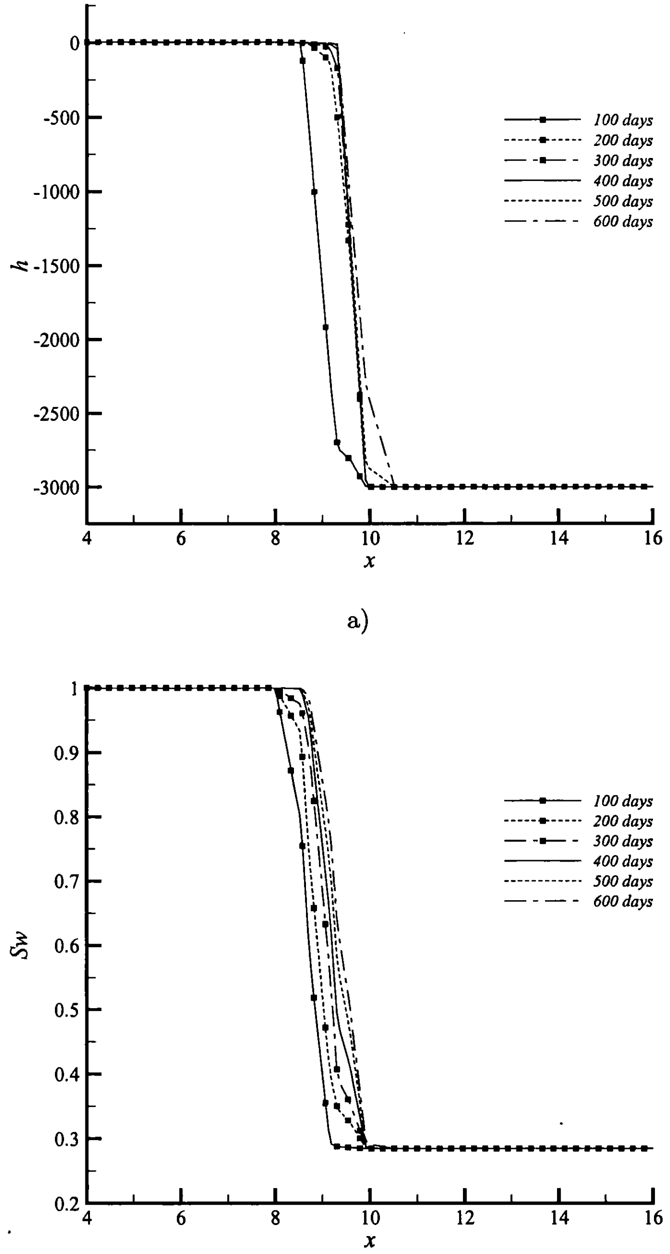
c) 300 days

d) 400 days

e) 500 days

f) 600 days

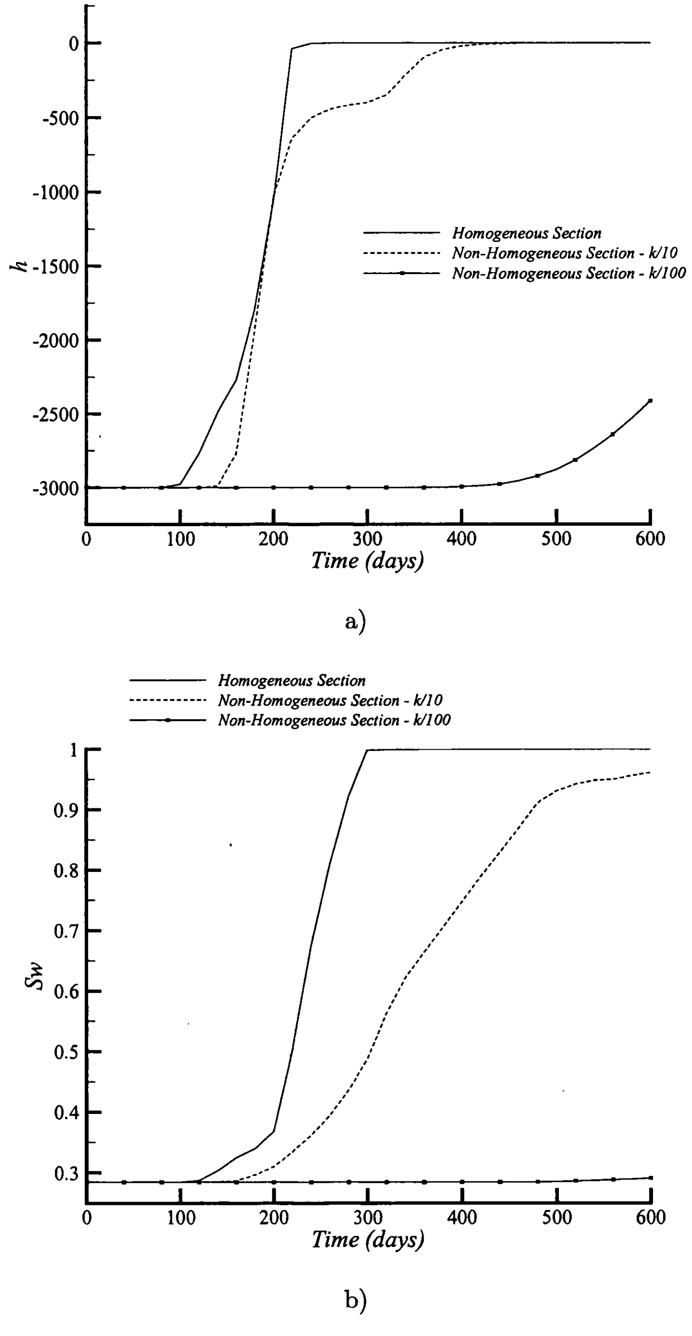
Figure 5.29: Pressure head, *h*, contours (-3000, -2500, -2000, -1500, -1000, -500 and –100m) across a zoned clay loam dam with a compacted core after a)100, b)200, c)300, d)400, e)500 and f)600 days respectively. Core permeability is 1/100th that of the remainder of the dam material.



a)

b)

Figure 5.30: a) pressure head, h, and b) water saturation, S_w , profiles along the line $y = 6.0$ m, $x \in [4,16]$m through a zoned clay loam dam using the compact stencil scheme at 100, 200, 300, 400, 500 and 600 days. Core permeability is 1/100th that of the remainder of the dam material.



a)

b)

Figure 5.31: Plots of, a) pressure head, h, and b) water saturation, S_w at the point (10,6) m against time for the three dam sections considered.

5.5 CLOSURE

This chapter has presented a vertex-centred edge-based finite volume formulation for groundwater flow problems. The scheme is capable of handling highly non-linear material parameters and is made more accurate by employing the compact stencil discretisation approach. The scheme is edge-based and is highly scalable between small and large meshes, both structured and unstructured, of an arbitrary number of vertices per element.

Using a successive over relaxation non-linear iteration loop, the analysis time for the solution of the given problem was accelerated by an order of magnitude. By applying specific values of β, γ, and ε to the non-linear update loop, much larger timesteps were achievable as compared to the linear update solution. In general, the scheme has been shown to be robust with respect to the non-linear parameters. For a very large timestep length, the stability of the solution is highly dependent on the non-linear loop parameters. If this problem is experienced, then a slight reduction in the step length will, in general, produce a stable solution and good speedup.

The test cases show that the standard finite volume method and compact stencil formulations have been successfully implemented for solving problems concerning two-dimensional flow in unsaturated soils. The first and second test case presents the infiltration of water into an initially dry soil column using a prefixed pressure head at the top of the column, and a prefixed flow rate at the top of the column respectively. For both test cases, the finite volume scheme captured the saturation front much more accurately than the finite element solution published by Gottardi ([12],[13]). Moreover, the results show that the compact stencil scheme produces less smearing of the non-linear front as compared to the standard scheme. Finally, the third test case models the infiltration flow of water across an initially dry compacted clay dam. Both homogeneous and inhomogeneous dam sections were considered, and the scheme produced excellent solutions for each case.

To the authors' knowledge, this is the first instance in which the application of the elegant edge-based finite volume scheme has been used to solve the case of highly non-linear saturated-unsaturated groundwater flow through a porous medium.

REFERENCES

[1] Avrashi J., Michael O. and Rosenhouse G., An Adaptive Successive Over Relaxation Domain Decomposition By The Boundary Spectral Strip Method, *International Journal For Numerical Methods In Engineering*, **40**(8): 1383-1397, 1997.

[2] Aziz K. and Settari A., Petroleum Reservoir Simulation, Applied Science, London, 1979.

[3] Bear J., Hydraulics of Groundwater (McGraw-Hill Series in Water Resources and Environmental Engineering), McGraw Hill College Div., 1979.

[4] Celia M.A., Bouloutas E.T. and Zarba R.L., A general mass-conservative numerical solution for the unsaturated flow equation, *Water Resources Res.*, **26**(7): 1483–1496, 1990.

[5] Chrichlow H.B., Modern Reservoir Engineering – A simulation Approach, Prentice Hall, New Jersey, 1977.

[6] Cooley R.L., Some new procedures for numerical solution of variably saturated flow problems. *Water Resources Res.,* **19**: 1271–1285, 1983.

[7] Crumpton P.I., Moinier P. and Giles M.B., An Unstructured Algorithm for High Reynolds Number Flows on Highly Stretched Meshes, in: C. Taylor, J.T. Cross, ed., Numerical Methods in Laminar and Turbulent Flow (Pineridge Press, 1997) 561-572.

[8] de Foy B. and Dawes W.. Unstructured pressure-correction solver based on a consistent discretization of the Poisson equation. *International Journal for Numerical Methods in Fluids*, **34**: 463-478, 2000.

[9] Forsyth P.A., Comparison of the single-phase and two-phase numerical model formulation for saturated-unsaturated groundwater-flow, *Computers Methods in Applied Mechanics and Engineering*, **69**: 243-259, 1988.

[10] Forsyth P.A., Wu Y.S. and Pruess K., Robust numerical methods for saturated-unsaturated flows with dry initial conditions in heterogeneous media, *Advances in Water Resources*, **18**: 25-38, 1995.

[11] Gottardi G and Venutelli M., Richards: Computer program for the numerical simulation of one-dimensional filtration into unsaturated soil, *Computers & Geosciences*, **19**(9): 1239-1266, 1993.

[12] Gottardi G and Venutelli M., UPF: two-dimensional finite-element groundwater flow model for saturated–unsaturated soils, *Computers & Geosciences*, **27**(2): 179-189, 2001.

[13] Gottardi G., Private communication, 2003.

[14] Huyakorn P.S. and Pinder G.F., Computational methods in subsurface flow, Academic Press, San Diego, 1985.

[15] Kavetski D., Binning P. and Sloan S.W., Adaptive time stepping and error control in a mass conservative numerical solution of the mixed form of Richards equation, *Advances in Water Resources*, **24**(6): 595-605, 2001.

[16] Lambe T.W., 'The Engineering Behaviour of Compacted Clays', *Transactions, American Society of Civil Engineers*, vol. 125, p. 718, 1960

[17] Lambe T.W., 'The Structure of Compact Clays', *Transactions, American Society of Civil Engineers*, **125**, p. 682, 1960.

[18] Luo H., Baum J.D. and Lohner R., Edge-based finite-element scheme for the Euler equations. *AIAA*, **32**(6): 1183-1190, 1994.

[19] Malan A.G., Investigation Into The Continuum Thermodynamic Modelling Of Investment Casting Shell-Mould Drying, PhD Thesis, School of Engineering, University of Wales Swansea, 2002.

[20] Malan A.G., Lewis R.W. and Nithiarasu P., An Improved Unsteady, Unstructured, Artificial Compressibility, Finite Volume Scheme for Viscous Incompressible Flows: Part I. Theory and Implementation, *International Journal For Numerical Methods In Engineering*, **54**(5): 695-714, 2002.

[21] Masters I., Usmani A.S., Cross J.T. and Lewis R.W., Finite Element Analysis of Solidification using Object-Oriented and Parallel Techniques, *International Journal For Numerical Methods In Engineering*, **40**: 2891-2909, 1997.

[22] Onate E., Cervera M. and Zienkiewicz O.C., A finite volume format for structural problems. *International Journal For Numerical Methods In Engineering*, **37**(2): 181-201, 1994

[23] Paniconi C. and Putti M., A comparison of Picard and Newton iteration in the numerical solution of multidimensional variably saturated flow problems. *Water Resources Res.*, **30**: 3357–3374, 1994.

[24] Queiroz L.A., 'Compressible Foundation at Tres Marias Earth Dam', Proceedings First Panamerican Conference on Soil Mechanics and Foundation Engineering, Mexico (1959), vol. II, p. 763.

[25] Rathfelder K. and Abriola L.M., Mass conservative numerical solutions of the head-based Richards equation, *Water Resources Res.*, **30**(9): 2579–2586, 1994.

[26] Rees I., Lewis R.W., Masters I. and Malan A.G., Material Variation Using A Compact Stencil Vertex-Centred Edge-Based Finite Volume Scheme, In proceedings: M.A. Wheel, ed., The 11th Annual Conference of the Association for Computational Mechanics in Engineering, Glasgow, 5-9, 2003.

[27] Rees I., Lewis R.W., Masters I. and Malan A.G., Two Dimensional Vertex-Centred Edge-Based Finite Volume Groundwater Flow Model For Saturated-Unsaturated Soils, Internal Report, Civil & Computational Engineering Centre, University of Wales Swansea, 2003.

[28] Richards L.A., Capillary conduction of liquids through porous mediums. *Physics,* **1**: 318–333, 1931.

[29] Ross P.J., Efficient numerical methods for infiltration using Richards' equation. *Water Resources Res.*, **26**: 279–290, 1990.

[30] Sorensen K.A., Hassan O., Morgan K. and Weatherill N.P., Agglomerated multigrid on hybrid unstructured meshes for compressible flow. *International Journal For Numerical Methods In Fluids*, **40**(3-4): 593-603, 2002.

[31] Sowers G.F. and Sally H.L., 'Earth and Rockfill Dam Engineering', Asia Publishing House, London, 1962.

[32] Tocci M.D., Kelley C.T., Miller C.T. and Kees C.E., Inexact Newton methods and the method of lines for solving Richards' equation in two space dimensions. *Comput. Geosci.,* **2**(4): 291–309, 1999.

[33] Tres Marias Dam, *Water Power,* part I (December 1959); part II (January 1960); part III (February 1960).

[34] Vahdati M., Morgan K., Peraire J. and Hassan O., A cell-vertex upwind unstructured grid solution procedure for high-speed compressible viscous flow. In proceedings International Conference on Hypersonic Aerodynamics, London, 1989. Royal Aeronautical Society, pages 12.1-12.22.

[35] van Dam J.C. and Feddes R.A., Numerical simulation of infiltration, evaporation and shallow groundwater levels with the Richards equation, *Journal of Hydrology*, **233**(1-4): 72-85, 2000.

[36] Zaidel J. and Russo D., Estimation of finite difference interblock conductivities for simulation of infiltration into initially dry soils. *Water Resources Res.*, **28**: 2285–2295, 1992.

[37] Zhang X., Bengough A.G., Crawford J.W. and Young I.M., Efficient methods for solving water flow in variably saturated soils under prescribed flux infiltration, *Journal of Hydrology*, **260**(1-4): 75-87, 2002.

Chapter 6

MODELLING AQUIFER CONTAMINANT TRANSPORT

6.1 INTRODUCTION

In this chapter, the modelling of contaminant transport through aquifers over a horizontal plane by the application of the vertex-centred edge-based finite volume scheme is presented. De Wiest [4] defines an *aquifer* as a geologic formation, or a group of formations, which (i) contains water and (ii) permits significant amounts of water to move through it under ordinary field conditions. These aquifers may be classed as *confined* or *unconfined.* A *confined aquifer* is one bounded from above and below by impervious formations, where an *unconfined aquifer* has one or more pervious boundaries allowing the recharge and leakage of the groundwater contained within it. As aquifers are comprised of layers of rock or unconsolidated deposits, generally their dimensions along the horizontal plane (that is along the x and y coordinate axes) are an order of magnitude greater than their dimensions in the vertical direction. Hence, representing an aquifer as a two-dimensional domain is justified in simplified cases as considered in this chapter.

Today, our environment suffers more and more from the by-products of man's industrial activities, and hence water quality is a major concern in any development and management of a water resources system. Bear [2] states that as the quality of both surface and groundwater resources deteriorate as a result of pollution, special attention should be devoted to the pollution of groundwater in aquifers due to their very slow velocity. Therefore, the study and simulation of contaminant transport in groundwater is of major importance, and in the 1970s the main aim of numerical models changed from the study of groundwater flow to problems of spreading of various substances in porous media [10]. Bear [2] defines term 'pollutant' to denote dissolved matter carried with the water and accumulating in the aquifer, without inferring that concentrations have necessarily reached dangerous levels. Groundwater pollution is usually traced back to four sources:

- *Environmental.* This type of pollution is due to the environment through which the flow of groundwater takes place. For example, in flow through carbonate rocks, water dissolves small, yet sometimes significant amounts of the rock. Seawater intrusion, or the pollution of good quality aquifers by invading brackish groundwater from adjacent aquifers as a result of disturbing an equilibrium that existed between the two bodied of water, are also examples of environmental pollution.
- *Domestic.* Domestic pollution may be caused by accidental breaking of sewers, by percolation from septic tanks, by rain infiltrating through sanitary landfills, or by artificial recharge of aquifers by sewage water after being treated to different levels. Biological contaminants (e.g. bacteria and viruses) are also related to this source.
- *Industrial.* In many cases, a single sewage disposal system serves both industrial and residential areas. In this case, one cannot separate between industrial and domestic pollution, although their compositions – and hence the type of treatment they require and the pollution they cause – are completely different. Heavy metals, for example, constitute a major problem in industrial waste. Industrial waste may also contain specially non-deteriorating toxic compounds and radioactive materials.
- *Agriculture.* This is due to irrigation water and rainwater dissolving and carrying fertilisers, salts, herbicides, etc., as they infiltrate through the ground surface and replenish the aquifer.

Hence, the simulation of mass transport in porous media is dealt with in this chapter, where the considered mass is that of some solute (pollutant) moving with the solvent (water) in the interstices of a porous medium. The transport of pollutants in porous media can be divided into two main groups; miscible displacement – pollutants are soluble in water, and immiscible displacement – pollutants are insoluble in water.

- *Transport of Miscible Pollutants.*

Consider saturated flow through a porous medium, and let a portion of the flow domain contain a certain mass of solute (pollutant). The quantity of a solute in groundwater is represented by its concentration, and is defined as the substance's quantity per unit of water volume. The solute will be referred to as a *tracer.* Bear [2] states that as flow take place, the tracer gradually spreads and occupies an ever-increasing portion of the flow domain, beyond the region it is expected to occupy according to the average flow alone. The spreading phenomenon is called *hydrodynamic dispersion* in a porous medium, and it is a macroscopic reflection of a real movement of particles in pores. It is a non-steady, irreversible process (in the sense that the initial tracer distribution cannot be obtained by reversing the flow) that creates a transition zone where the concentration continuously changes from the minimal to the maximal value.

De Wiest [4] reports that one of the earliest observations of these phenomena is presented by Slichter [18], who used an electrolyte as a tracer in studying the movement of groundwater. Slichter observed that at an observation well downstream of a continuous injection point, the tracer's concentration increases gradually, and that even in a uniform (average) flow field the tracer advances in the direction of the flow in a pear-like shape that becomes longer and wider as it advances.

The Galerkin finite element method has been extensively used to solve miscible contaminant transport problems as reported by Pinder and Gray [16]. In Gray and Pinder [6], the effects of various finite difference and finite element methods for solving the one-dimensional convective-dispersive equation are investigated. Numerical solutions to such problems characteristically exhibit either oscillations in concentration in the neighbourhood of a sharp front or a smearing of the front, and the relative merits of certain finite difference and finite element schemes for solving such problems are considered in terms of frontal smearing and concentration overshoot. Results show that both finite difference and finite element schemes generate numerical solutions which exhibit an oscillatory behaviour behind, and in some cases in front of the concentration front. Procedures to correct such oscillations are presented, and they conclude that the finite element method was found to be superior to the finite difference method for solution of the convective-dispersive equation.

Selim and Mansell [17] present mathematical solutions of the differential equations governing reactive solute transport in a finite soil column for a continuous solute input, and a pulse-type, or slug input at the soil surface. These solutions incorporated reversible linear adsorption as well as irreversible solute adsorbtion. Comparisons were made with the mathematical solutions developed by Cleary and Adrian [3] and Lindstrom *et al.* [12], and it was concluded that for high velocity flow all three solutions are in agreement, but the Selim and Mansell solution is superior for flows with a Peclet number of $Pe < 20$.

- *Transport of Immiscible Pollutants*

This part of the transport of pollutants was developed mainly by solving problems connected with oil exploration. Oils are a typical group that are almost insoluble, and hence they occupy an entire area in a porous medium, called a phase. As a result this is often called a multi phase flow. The most frequently consideed cases of multi phase flows is two phase flow (e.g. oil-water, water-air) and three-phase flow (e.g. oil-water-gas) systems. Multi phase fluid systems have been successfully simulated using various numerical techniques by numerous researchers e.g. ([9],[11],[20],[24]).

The developed edge-based finite volume scheme will be applied to solve a contaminant transport problem where the contaminant is released into the aquifer from a strip source at the upstream end of a homogeneous and isotropic domain. The problem will be run on structured quadrilateral element meshes of different densities using the standard and compact stencil finite volume schemes, and the solution compared against an analytic solution and a published finite element solution. Numerical experiments will be conducted to find the optimal timestep length for these problems. The modelling of pollutant slugs entering the domain at the upstream boundary will then be considered, and the exponential decay of the pollutant in the slug and continuous injector boundary investigated. A pollutant slug can be defined as the introduction of a pollutant into an aquifer for a finite time, after which no further release occurs. This could be due to an accidental breaking of sewers or such pollutant carrying vessels, which would release pollutant into the groundwater until the problem was resolved. For all cases, varying levels of diffusion and advection in the equation system will be taken into account, and highly diffusion dominant, diffusion and advection equally dominant, and advection dominant problems will be considered. Finally, conclusions will be drawn from the work undertaken.

6.2 GOVERNING EQUATIONS

The governing equation for contaminant transport through aquifers over a horizontal plane can be obtained by vertically integrating the three-dimensional equation [2]. A simplified form of this equation is shown in (6.1), and it will be used to describe the aquifer contaminant transport problem considered here.

$$\nabla.(\mathbf{D}\nabla c)-\nabla.(\mathbf{v}c)+g=\frac{\partial c}{\partial t} \tag{6.1}$$

where ∇ is the dell operator; c is the concentration; t is time; $\mathbf{v}$ is velocity; $\mathbf{D}$ is the dispersion coefficient tensor and g incorporates chemical reactions. Note that $\mathbf{v}=\left(v_{xx},v_{yy}\right)$.

For problems where the exponential decay (radioactive decay) of the contaminant species over time is considered, then the chemical reaction term in equation (6.1) is defined as follows,

$$g = -\lambda c \tag{6.2}$$

where $\lambda =$ decay rate coefficient. By substituting equation (6.2) into (6.1), and making the assumption that c is constant with respect to x, then the decay term can be described as follows,

$$\frac{dc}{dt} = -\lambda c \tag{6.3}$$

It can be seen that,

$$c = e^{-\lambda t} \tag{6.4}$$

is a solution to (6.3) and hence proves that the decay term (6.2) used is indeed exponential decay.

6.3 SOLUTION PROCEDURE

The solution procedure employed to solve the contaminant transport problem using the edge-based finite volume scheme defined in chapter 2 is presented in this section. By implying that the second mixed derivatives are zero, the expansion of the transport equation (6.1) on a two dimensional domain Ω gives;

$$\frac{\partial}{\partial x}\left(D_{xx}\frac{\partial c}{\partial x}\right) + \frac{\partial}{\partial y}\left(D_{yy}\frac{\partial c}{\partial y}\right) - v_{xx}\frac{\partial c}{\partial x} - v_{yy}\frac{\partial c}{\partial y} - \lambda c = \frac{\partial c}{\partial t} \tag{6.5}$$

Integration over the control volume gives:

$$\begin{aligned}&\int_{\Omega}\frac{\partial}{\partial x}\left(D_{xx}\frac{\partial c}{\partial x}\right)\,d\Omega+\int_{\Omega}\frac{\partial}{\partial y}\left(D_{yy}\frac{\partial c}{\partial y}\right)\,d\Omega\\&-\int_{\Omega}v_{xx}\frac{\partial c}{\partial x}\,d\Omega-\int_{\Omega}v_{yy}\frac{\partial c}{\partial y}\,d\Omega-\int_{\Omega}\lambda c\,d\Omega=\int_{\Omega}\frac{\partial c}{\partial t}\,d\Omega\end{aligned}\tag{6.6}$$

The spatial derivative terms in equation (6.6) are discretised by application of Green's theorem, and by the assumption that the spatial derivative is linear over the control volume Ω_m. The application of the edge-based discretisation procedures and the Euler backward difference scheme as discussed in chapter 2 gives for all nodes m in the domain;

$$\begin{aligned}&\frac{\partial}{\partial x}\left[\left(D_{xx}\right)_{mn}\left(\frac{\partial c^n}{\partial x}\right)\right]\Bigg|_m\Omega_m+\frac{\partial}{\partial y}\left[\left(D_{yy}\right)_{mn}\left(\frac{\partial c^n}{\partial y}\right)\right]\Bigg|_m\Omega_m\\&-\left(v_{xx}\right)_{mn}\left(\frac{\partial c^n}{\partial x}\right)\Bigg|_m\Omega_m-\left(v_{yy}\right)_{mn}\left(\frac{\partial c^n}{\partial y}\right)\Bigg|_m\Omega_m-\lambda c_m^n\Omega_m=\frac{c_m^{n+1}-c_m^n}{\Delta t}\cdot\Omega_m\end{aligned}\tag{6.7}$$

The term $\left(v_i\right)_{mn}$ and $\left(D_i\right)_{mn}$ are the velocity and dispersion coefficient components respectively applied to edge_{mn}.

The diffusion terms in equation (6.7) are grouped into one term Φ_m as shown in (6.8), and the advection terms grouped into the term Θ_m as shown in (6.9);

$$\Phi_m=\frac{\partial}{\partial x}\left[\left(D_{xx}\right)_{mn}\left(\frac{\partial c^n}{\partial x}\right)\right]\Bigg|_m\Omega_m+\frac{\partial}{\partial y}\left[\left(D_{yy}\right)_{mn}\left(\frac{\partial c^n}{\partial y}\right)\right]\Bigg|_m\Omega_m\tag{6.8}$$

$$\Theta_m=-\left(v_{xx}\right)_{mn}\left(\frac{\partial c^n}{\partial x}\right)\Bigg|_m\Omega_m-\left(v_{yy}\right)_{mn}\left(\frac{\partial c^n}{\partial y}\right)\Bigg|_m\Omega_m\tag{6.9}$$

As we assume that the flow is to be unidirectional along the x-axis, the advection term in equation (6.9) simplifies to,

$$\begin{aligned}\Theta_m &= -\left(v_{xx}\right)_{mn}\left(\frac{\partial c^n}{\partial x}\right)\Bigg|_m \Omega_m - \cancelto{0}{\left(v_{yy}\right)_{mn}\left(\frac{\partial c^n}{\partial y}\right)\Bigg|_m} \Omega_m \\ &= -\left(v_{xx}\right)_{mn}\left(\frac{\partial c^n}{\partial x}\right)\Bigg|_m \Omega_m\end{aligned} \tag{6.10}$$

Substituting equations (6.8) and (6.10) in (6.7) results in the following expression.

$$\Phi_m + \Theta_m - \lambda c_m^n \Omega_m = \frac{c_m^{n+1} - c_m^n}{\Delta t} \cdot \Omega_m \tag{6.11}$$

Rearranging (6.11) gives the explicit linear update for the pollutant concentration as:

$$\begin{aligned}c_m^{n+1} &= \left[\left(\Phi_m + \Theta_m\right) - \lambda c_m^n \Omega_m\right]\frac{\Delta t}{\Omega_m} + c_m^n \\ &= \left(\Phi_m + \Theta_m\right)\frac{\Delta t}{\Omega_m} - \lambda c_m^n \Delta t + c_m^n\end{aligned} \tag{6.12}$$

A flowchart of the solution procedure is given in Figure 6.1, where the symbols correspond to those used in equations (6.5) to (6.12).

6.4 TEST CASE - CONTAMINANT TRANSPORT FROM AN UPSTREAM STRIP SOURCE

This test case will consider the common problem of two-dimensional transport from a strip source at the upstream end of the domain. Similar problems have been considered by a number of researchers ([5],[21],[6]), and the test case considered here is taken from a publication by Yeh [23]. The flow through the domain is assumed to be unidirectional along the x direction having a velocity equal to 1.0 m/s. The medium is homogeneous and isotropic, and the longitudinal dispersivity is assumed equal to the lateral dispersivity hence $D_x = D_y$.

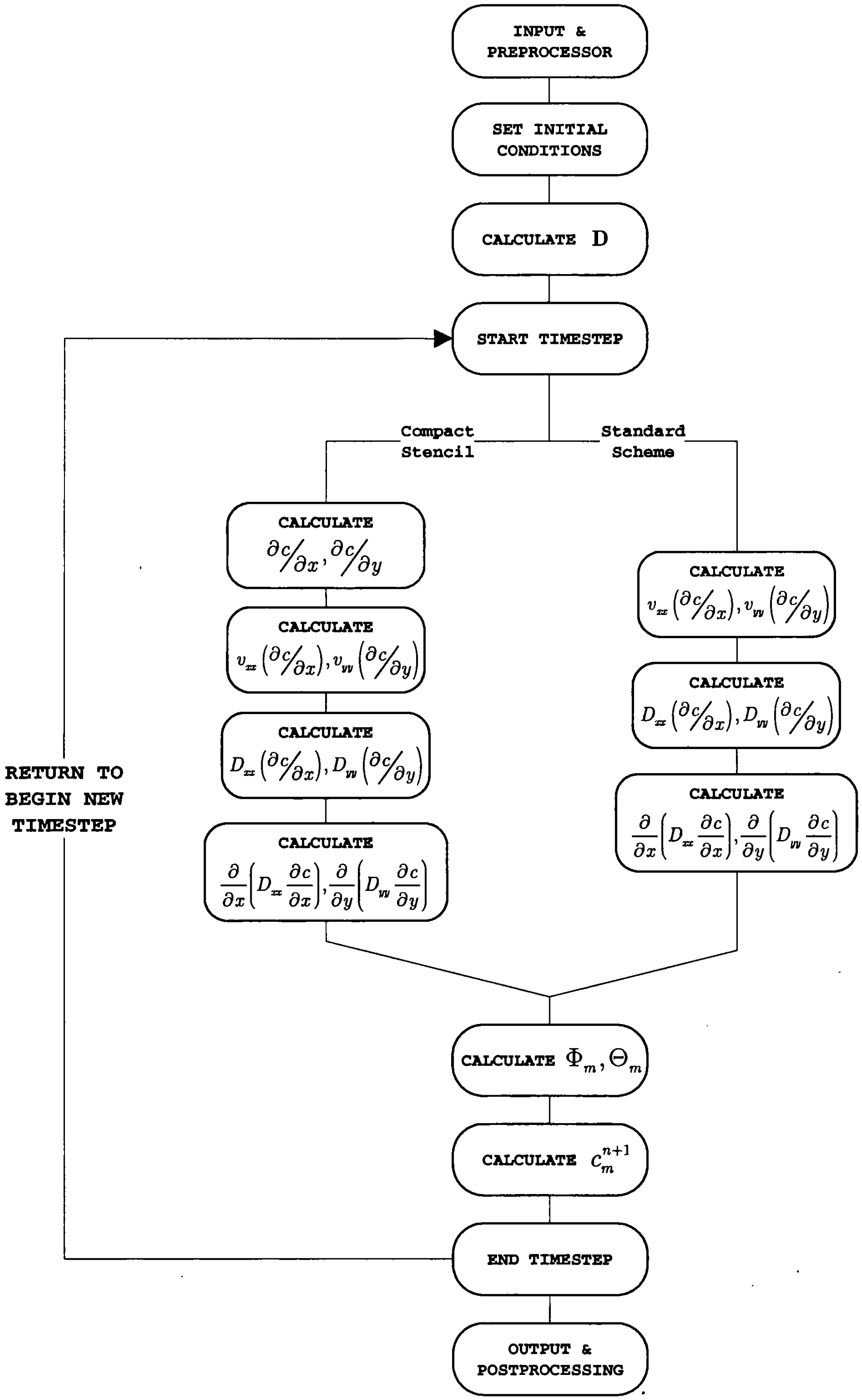


Figure 6.1: Program flow chart for the pollutant transport problem.

The dimensions of the domain to be considered is 5×10 metres (5 m in the y direction, 10 m in the x direction) and is shown in Figure 6.2, along with markers to enable the identification of each boundary.

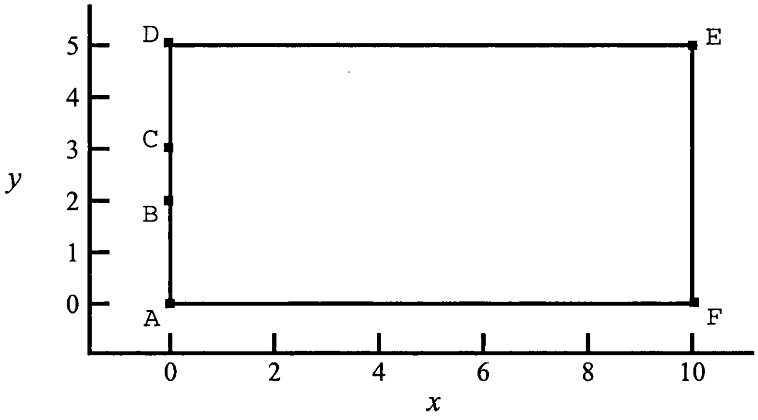


Figure 6.2: The boundary of the domain considered in the contaminant transport test case, along with boundary labels.

The initial and boundary conditions for this problem are;

$$\begin{aligned} c_0 &= 0.0 \text{ at } t = 0, \\ c &= 1.0 \text{ on side B-C} \\ c &= 0.0 \text{ on side A-B and C-D}, \\ \partial c / \partial y &= 0.0 \text{ on side D-E and A-F}, \\ \partial C / \partial x &= 0.0 \text{ on side E-F.} \end{aligned}$$

In contrast to the groundwater flow models, the models of miscible pollutant transport have to deal with the effects of an advective transport. This makes the solution difficult, and, as we shall see later, a new error emerges, which causes an instability of the front of a transition zone by high velocities (that is when the advective term has a great influence). To assess the influence, the Peclet number (Pe) and Courant number (C_r) is introduced as;

$$Pe = \frac{v_x \Delta x}{D} \tag{6.13}$$

and,

$$C_r = \frac{v_x \Delta t}{\Delta x} \tag{6.14}$$

respectively, where Δx is the minimum edge length for the mesh, and Δt the timestep length. It is apparent that the higher the velocity, the larger the Peclet number. When the velocity equals zero, only a molecular diffusion comes into play and $Pe = 0$. The other limit case is a pure advective transport when $D = 0$ and $Pe = \infty$. Three cases, with Peclet numbers of 0.05, 1.0 and 50.0 are considered. They represent three distinct cases of contaminant transport problems: 1) highly dispersion dominant, 2) dispersion and advection equally dominant, and 3) highly advection dominant respectively. These Peclet numbers are made up by varying the dispersion coefficient from 10.0 to 0.5 and 0.01 respectively.

For the numerical simulation, the domain is discretised into two distinct meshes, Mesh 1 and 2 which are illustrated in Figure 6.3. Mesh 1 contains 10 x 20 quadrilateral (square) elements, resulting in 11 x 21 = 231 nodes, and has an edge length (Δx) of 0.5 m. Mesh 2 contains 21 x 41 quadrilateral (square) elements, resulting in 22 x 42 = 924 nodes, and has an edge length of 0.2439024 m.

The three cases, with Peclet numbers of 0.05, 1.0 and 50.0 are solved and the results obtained are discussed in subsequent sections. The numerical results are displayed graphically and are compared to the analytical solution to the problem, and the solution obtained by Yeh [23] using a finite element scheme.

6.4.1 Dispersion Dominant System - Pe = 0.05

The results for the contaminant transport test case with a Peclet number of 0.05, using the standard and compact stencil finite volume scheme on Mesh 1 and 2 are shown in Figure 6.4 to Figure 6.6. The chosen Peclet number represents a highly dispersion dominant system, and the solutions were obtained using a time step length of $dt = 0.001$ seconds.

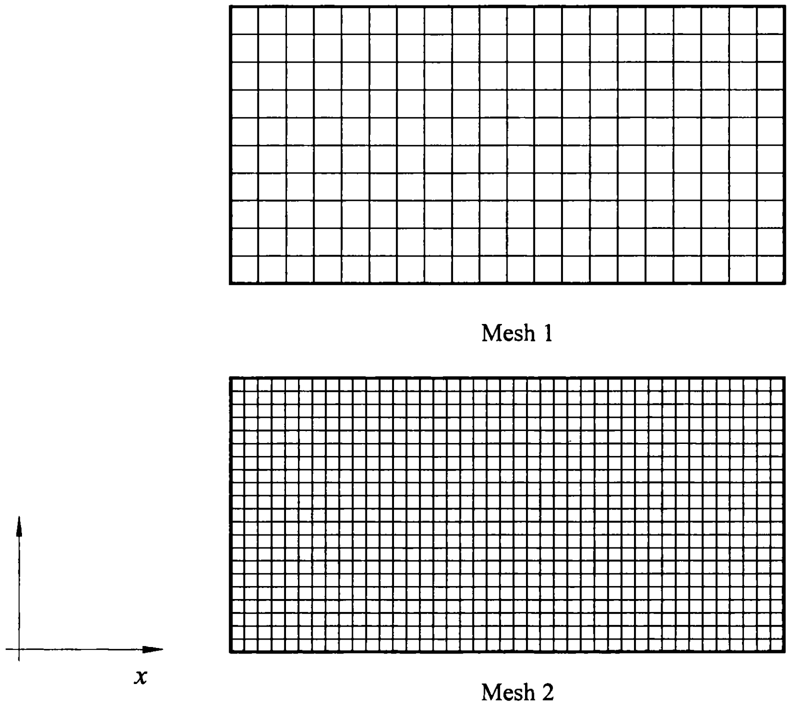


Figure 6.3: The coarse mesh (Mesh 1) and the finer mesh (Mesh 2) used for the contaminant transport problem.

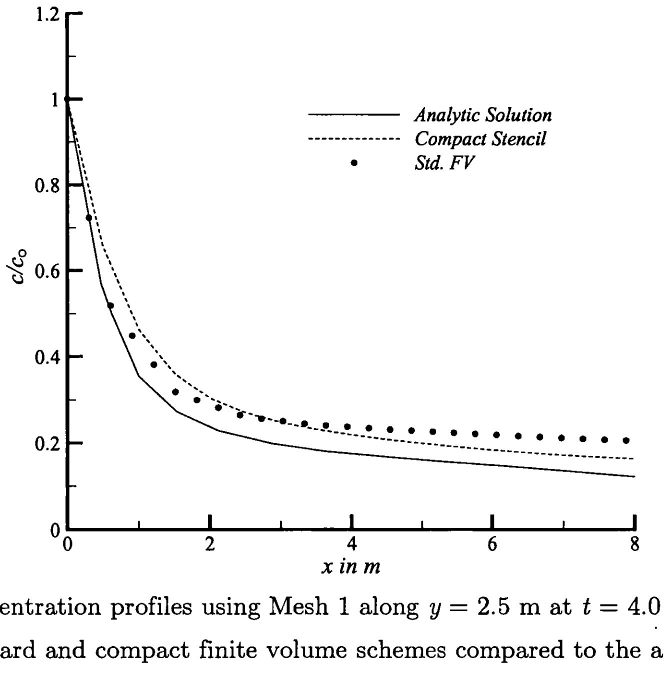


Figure 6.4: Concentration profiles using Mesh 1 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 0.05$, using the standard and compact finite volume schemes compared to the analytical solution.

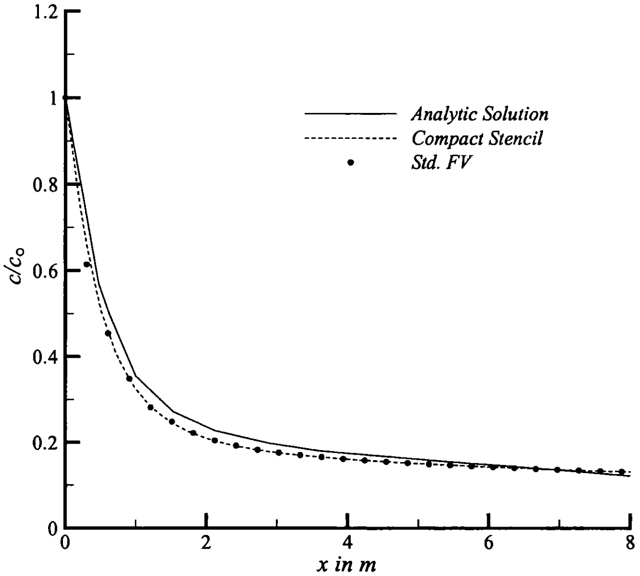


Figure 6.5: Concentration profiles using Mesh 2 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 0.05$, using the standard and compact finite volume schemes compared to the analytical solution.

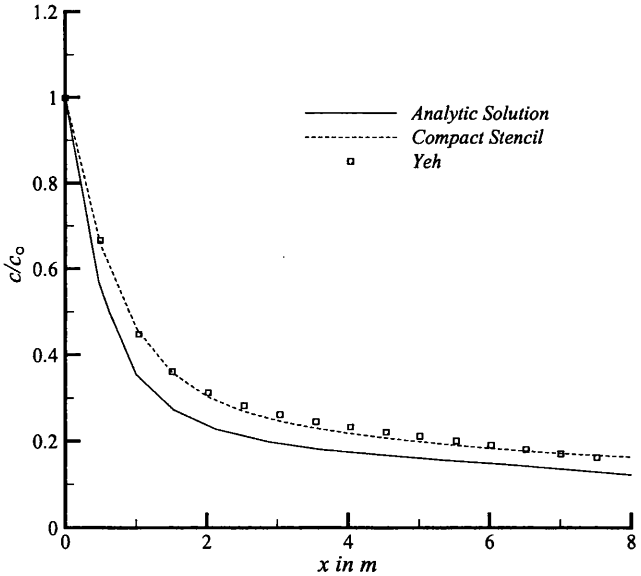


Figure 6.6: A comparison of the concentration profiles using Mesh 1 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 0.05$, using the compact stencil scheme compared to the analytical solution and the solution obtained by Yeh [23].

Figure 6.4 to Figure 6.6 show the concentration profiles along the horizontal centre line of domain at $t = 4.0$ seconds. The profiles were calculated using the standard finite volume scheme and the compact stencil scheme, and are compared to the analytic solution for the set problem. Using the coarser Mesh 1, Figure 6.4 shows that the standard scheme captures the exact solution for approximately the first 0.5 metres of the domain. This solution then diverges away from the analytic, and the plot is not smooth. Although the solution obtained using the compact scheme is smooth, it nowhere matches the analytic solution. However, it is much closer to the profile of the analytic solution than the standard scheme solution. When the finer Mesh 2 is used, the difference in the solution obtained using the standard and compact schemes along the horizontal centre line is negligible, as shown in Figure 6.5. However, it is encouraging that the result obtained using the compact scheme is almost identical to the finite element result of Yeh [23] (see Figure 6.6). The mesh is the same in both cases but the analysis very different. Yeh [23] used an upstream weighted implicit method with successive over relaxation for this problem; whereas finite volume solution is a linear explicit method.

The main benefit of the compact scheme is displayed in the contour plots shown in Figure 6.7 and Figure 6.8, which show the concentration contours at $t = 4.0$ seconds for Mesh 1 and 2 respectively using; a) the compact stencil scheme, and b) the standard second derivative scheme. The disadvantage of the standard five-node stencil second derivative calculation as compared to the compact stencil scheme is clearly evident. Figure 6.7b shows the concentration obtained on the coarse Mesh 1 at 4.0 seconds. At the edges of the concentration front, where there are regions with large differentials, the leapfrog nature of the standard scheme has clearly introduced a spatial oscillation into the solution. For an identical case with the compact stencil (Figure 6.7a) these oscillations are not present. Similar observations can be made regarding Figure 6.8, where the concentration contour plots calculated using both schemes on the finer Mesh 2 are shown. Again, the standard scheme results in spatial oscillations, and the compact scheme solution is everywhere smooth.

By considering the plots along the designated centre line and the contour plots of the domain, it can be concluded that the compact scheme produces a more accurate solution than the standard scheme for diffusion dominant flow problems.

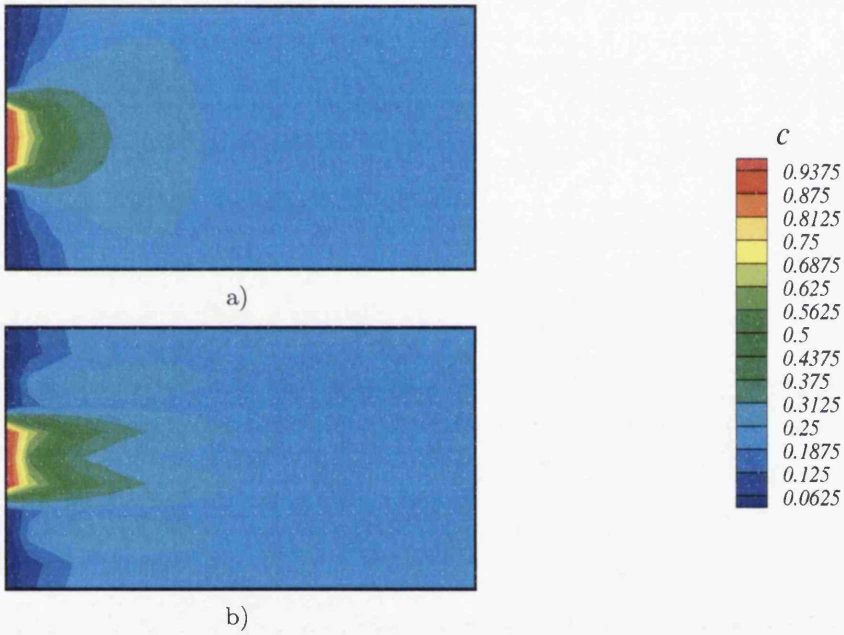


Figure 6.7: Concentration contours at $t = 4.0$ sec. for $Pe = 0.05$ for Mesh 1 using a) the compact stencil scheme, and b) the standard second derivative scheme.

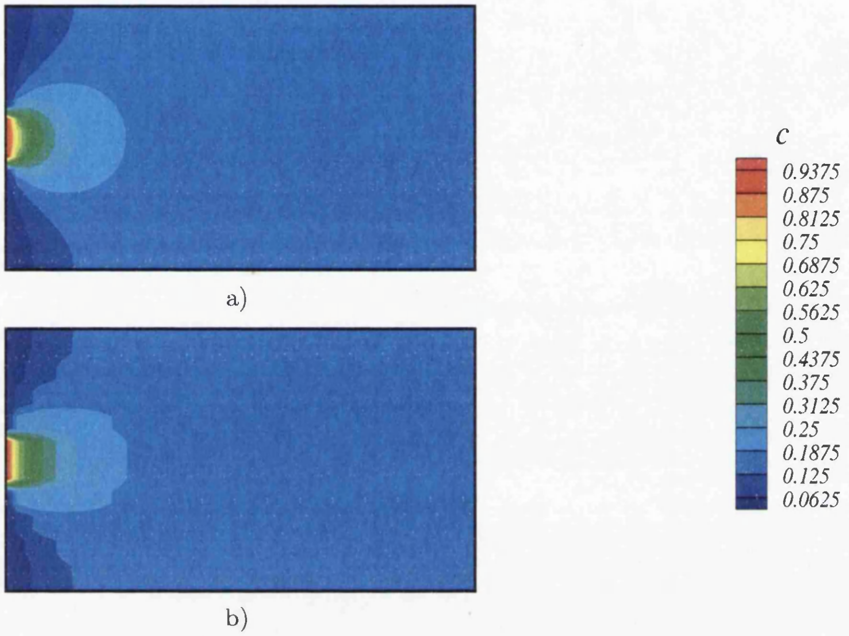


Figure 6.8: Concentration contours at $t = 4.0$ sec. for $Pe = 0.05$ for Mesh 2 using a) the compact stencil scheme, and b) the standard second derivative scheme.

6.4.2 Dispersion And Advection Equally Dominant System - Pe = 1.0

A Peclet number of 1.0 represents a dispersion and advection equally dominant system, and in this case the results obtained for the contaminant transport problem using the standard and compact stencil finite volume scheme are shown in Figure 6.9 to Figure 6.11. Results obtained using both Mesh 1 and 2 are illustrated, and the solutions were obtained using a time step length of $dt = 0.001$ seconds.

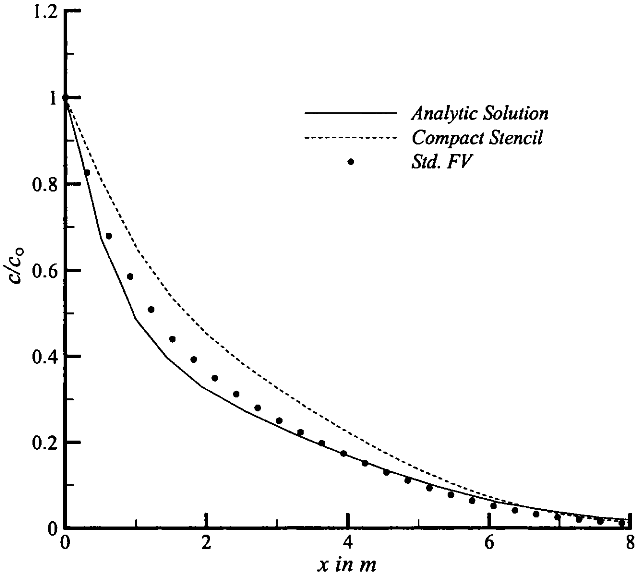


Figure 6.9: Concentration profiles using Mesh 1 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 1.0$, using the standard and compact finite volume schemes compared to the analytical solution.

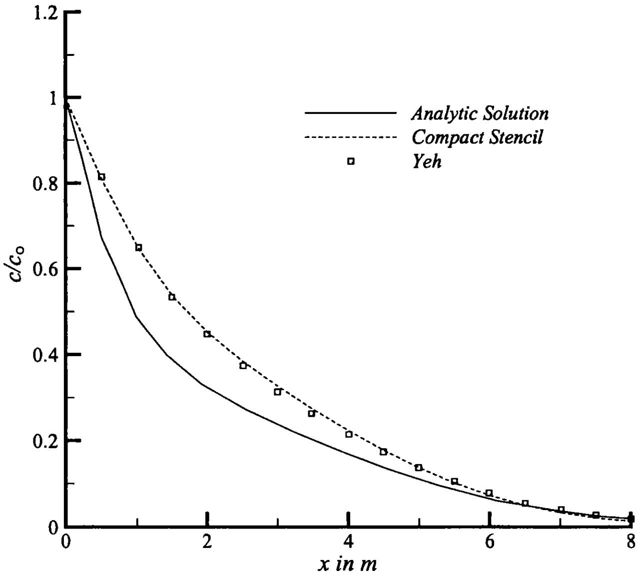


Figure 6.10: A comparison of the concentration profiles using Mesh 1 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 1.0$, using the compact stencil scheme compared to the analytical solution and the solution obtained by Yeh [23].

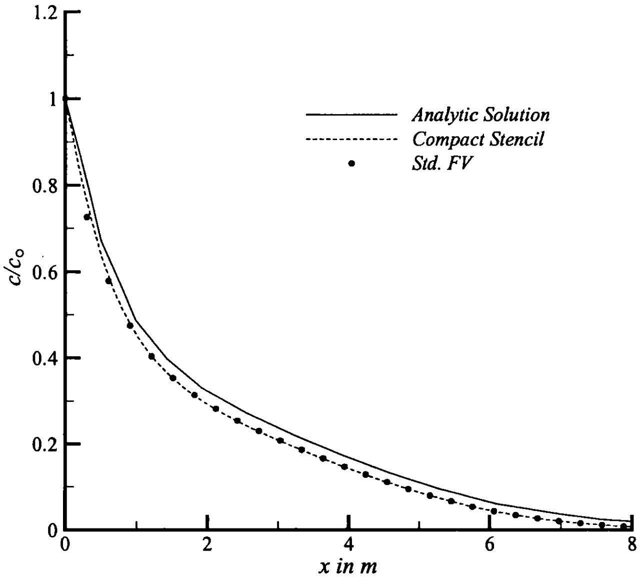


Figure 6.11: Concentration profiles using Mesh 2 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 1.0$, using the standard and compact finite volume schemes compared to the analytical solution.

Using the coarser Mesh 1, it is evident from Figure 6.9 that the standard scheme produces a solution which is much closer to the analytic than the compact stencil solution. However, on inspection of the concentration contours for Mesh 1, illustrated in Figure 6.12, it can be seen that the compact scheme produces markedly smoother contours along the whole domain when compared to the standard scheme contours. As the plot in Figure 6.9 only takes the values of the concentration along the horizontal centre line through the domain, it can be hypothesised that the values obtained for the standard scheme were fortunate to be close to the analytic solution as the concentration front oscillates in space as shown in Figure 6.12. However, as we are now considering a dispersion and advection equally dominant system, the benefit of the compact scheme, which is only used to calculate second derivative terms, is reduced. The advection terms are now beginning to affect the solution, and may be the cause of the difference between the accuracy of the compact and standard solutions in Figure 6.9. However, when the finer Mesh 2 is used, the difference in the calculated solution between the standard and compact schemes is negligible, as shown in Figure 6.11. Once more, the compact scheme result is almost identical to the computationally expensive finite element method result of Yeh [23], as shown by Figure 6.10.

Again, the advantage of the compact scheme as compared to the standard scheme is incontestable from the concentration contour plots shown in Figure 6.12 and Figure 6.13. The solution across the entire domain obtained using the compact scheme is smooth for both meshes, while the standard scheme solution has large spatial oscillations.

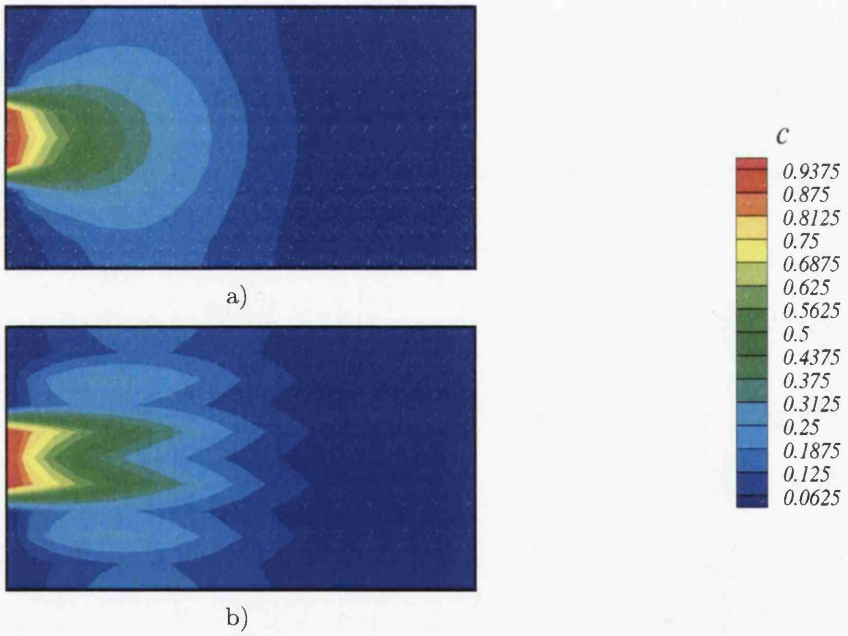


Figure 6.12: Concentration contours at $t = 4.0$ sec. for $Pe = 1.0$ for Mesh 1 using a) the compact stencil scheme, and b) the standard second derivative scheme.

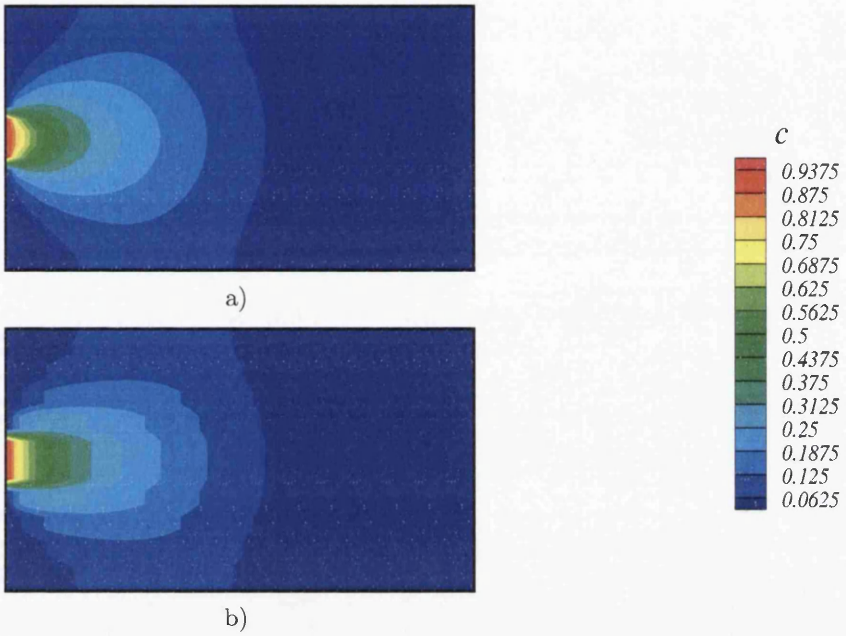


Figure 6.13: Concentration contours at $t = 4.0$ sec. for $Pe = 1.0$ for Mesh 2 using a) the compact stencil scheme, and b) the standard second derivative scheme.

6.4.3 Highly Advection Dominant System - Pe = 50.0

A Peclet number of 50.0 represents a highly advection dominant system, and the results obtained using the standard and compact stencil finite volume scheme for the contaminant transport problem are shown in Figure 6.14 to Figure 6.16, where they are also compared to the analytical solution. The solutions were obtained using a time step length of $dt = 0.001$ second, and Figure 6.14 to Figure 6.16 show the concentration profiles along the line $y = 2.5$ m through the domain at $t = 4.0$ seconds. Results for both Mesh 1 and Mesh 2 are illustrated.

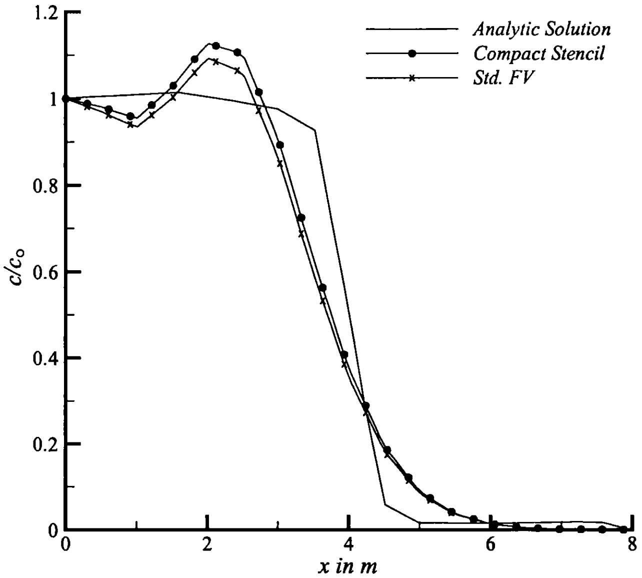


Figure 6.14: Concentration profiles using Mesh 1 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 50.0$, using the standard and compact finite volume schemes compared to the analytical solution.

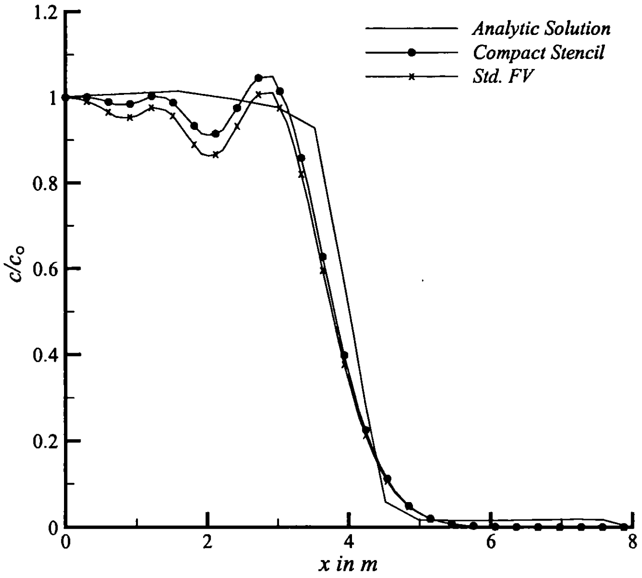


Figure 6.15: Concentration profiles using Mesh 2 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 50.0$, using the standard and compact finite volume schemes compared to the analytical solution.

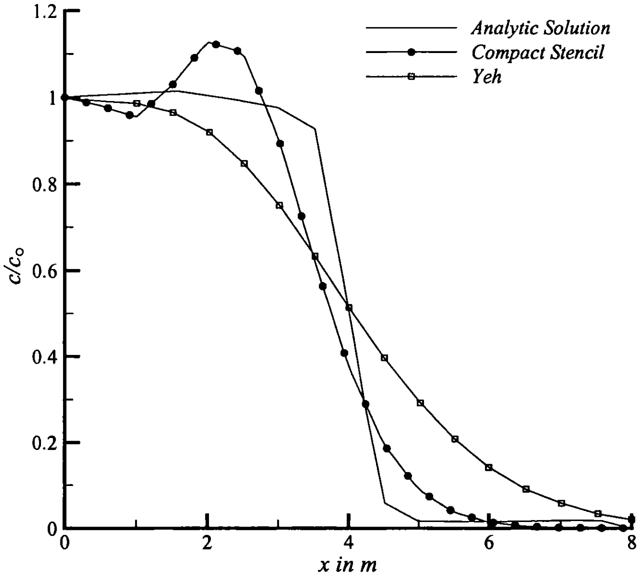


Figure 6.16: A comparison of the concentration profiles using Mesh 1 along $y = 2.5$ m at $t =$ 4.0 sec. for $Pe = 50.0$, using the compact stencil scheme compared to the analytical solution and the solution obtained by Yeh [23].

The front tracking capability of the scheme is shown by this set of results. This problem is dominated by the first order advection term and diffusion is minimal. Consequently the elliptic terms calculated by the more accurate compact stencil play a diminished role in the algorithm. Using the coarser Mesh 1, Figure 6.14 shows that both the standard and compact schemes produce comparable solutions. The same can be deduced from Figure 6.15, which shows the solutions obtained on the finer Mesh 2. Due to the highly advection dominant system, the elliptic terms calculated by the more accurate compact stencil play a diminished role in the algorithm. Hence, both solution schemes produce similar results. This is also shown by the concentration contour plots obtained using both schemes on both computational domains, shown in Figure 6.17 and Figure 6.18. Furthermore, it can be deduced from both Figure 6.14 and Figure 6.15 that the finite volume scheme closely captures the concentration front, with the sharper front obtained using the denser Mesh 2. In comparison to Yeh's solution [23] obtained on Mesh 1

(Figure 6.16), the front capturing capability of the edge-based finite volume scheme is superior to the front smearing of the finite element scheme employed in his work. Moreover, the solution obtained in the vicinity of the front is much closer to the analytic solution for the problem. The only drawback to the scheme is the oscillations observed in the solution behind the front. This is due to the way in which derivatives are calculated within the scheme. If the nodes contributing terms to the derivative calculated at node m span across the front, then oscillations will be generated, and will propagate. The numerical discretisation described using the edge-based finite volume scheme described in section 2.6.1 (First derivative terms) is a typical central-difference approach, which is known to be unstable for hyperbolic-type equations. Over the history of the use of such schemes in CFD related problems, stabilising such schemes while retaining second order accuracy has been one of the main concerns [7]. It was soon found that the addition of dissipation would make the scheme stable by damping the oscillations originating from the advection terms. The stabilising dissipation can be thought of as adding a simplified discrete harmonic term to the governing equations. This led to first order upwind schemes which are, although stable, too dissipative to produce accurate solutions for reasonable mesh sizes. The need for improved accuracy resulted in the discovery of several second order accurate approaches ([1],[8],[13],[22]). One class of such schemes are the *artificial dissipation* schemes where a third order biharmonic operator is added to the equations and has been successfully implemented by Malan [14], Malan *et al.* [15] and Sorensen [19]. However, as the problems we are interested in generally have low Peclet numbers, these schemes have not been implemented into the solution algorithm.

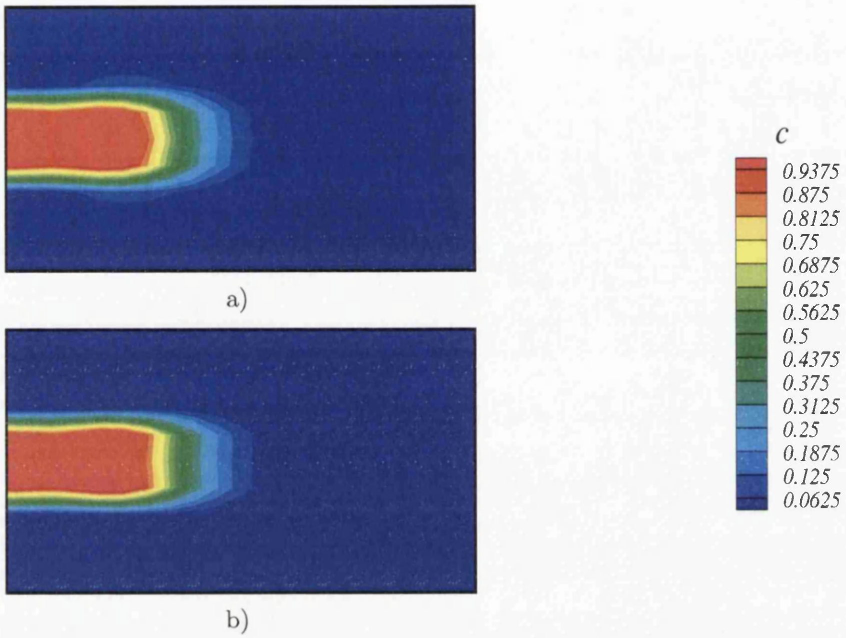


Figure 6.17: Concentration contours at $t = 4.0$ sec. for $Pe = 50.0$ for Mesh 1 using a) the compact stencil scheme, and b) the standard second derivative scheme.

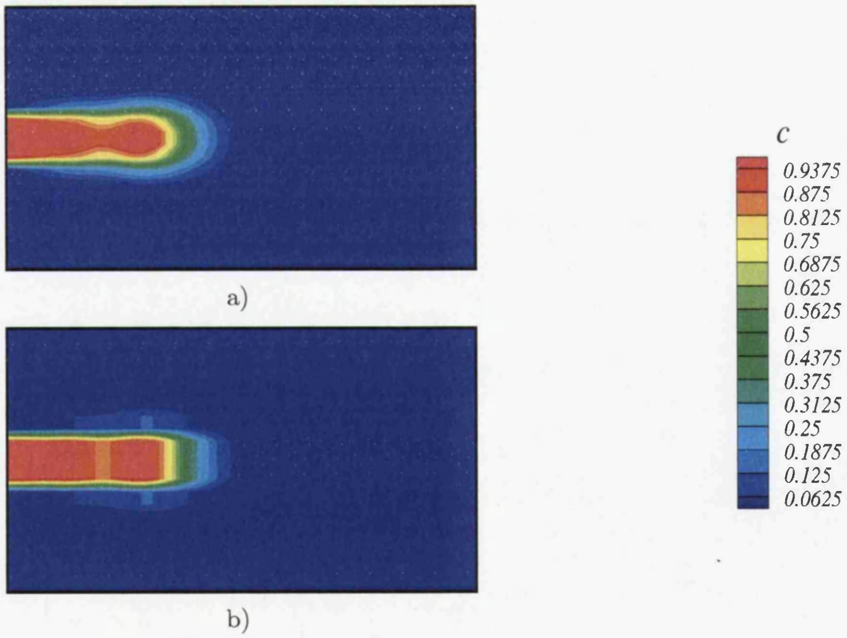


Figure 6.18: Concentration contours at $t = 4.0$ sec. for $Pe = 50.0$ for Mesh 2 using a) the compact stencil scheme, and b) the standard second derivative scheme.

6.5 TIME STEP LENGTH EXERCISE

Numerical experiments have shown that for Peclet numbers of 0.05 and 1.0, using both standard and compact stencil finite volume schemes on Mesh 1 and 2, the maximum allowable timestep size for the problem was $dt = 0.001$ seconds. Timesteps smaller than this produced exactly the same result, while larger timesteps cause the solution to become unstable, and convergence did not occur. Table 6.1 summarises the results of a similar numerical experiment run for a Peclet number of 50.0 carried out using the compact stencil finite volume scheme on Mesh 2 only, using timesteps of 1.0, 0.5, 0.1, 0.01, 0.001 and 0.00001 seconds. If the simulation produced a suitable solution then 'OK' is seen in the table, otherwise, if the solution did not converge the corresponding cell contains '--'. In addition the Courant number (equation (6.14)) for the mesh using the designated timestep length is shown.

Δt (sec.)	*Compact Stencil*	C_r
1.0	--	4.1
0.5	--	2.05
0.1	OK	0.41
0.01	OK	0.041
0.001	OK	0.41×10^{-02}
0.00001	OK	0.41×10^{-04}

Table 6.1: Timestep length exercise using the compact stencil finite volume scheme on Mesh 2 with $Pe = 50.0$

Figure 6.19 shows the calculated concentration profile along the horizontal centre line through the domain using Mesh 2 at $t = 4$ seconds with timesteps of 0.5, 0.1, 0.01, 0.001 and 0.00001 seconds. The solution was obtained using the compact stencil finite volume scheme.

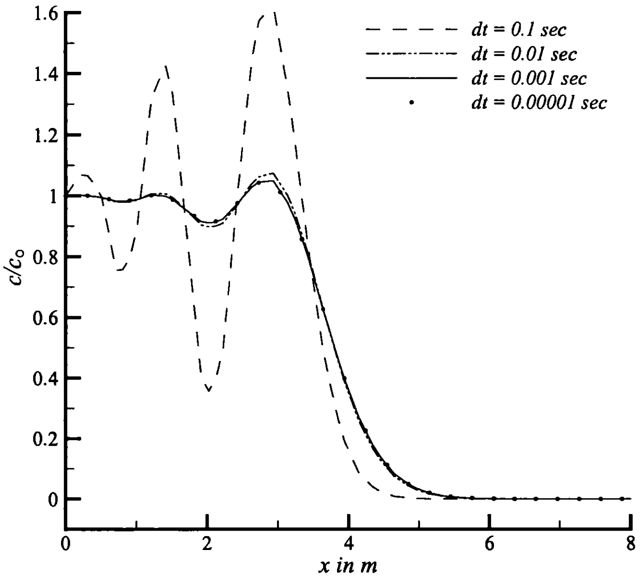


Figure 6.19: Concentration profiles on Mesh 2 along $y = 2.5$ m at $t = 4$ sec. for $Pe = 50.0$, using the compact stencil scheme, and timesteps of 0.1, 0.01, 0.001 and 0.00001 seconds.

The general trend seen from the result is that the larger timesteps greatly amplify the oscillations seen in the region behind the front. For the results obtained using a timestep of $dt = 1.0$ and 0.5 seconds, such large oscillations were produced behind the front that these results could not be included on the plot in Figure 6.19. However, the front position for both omitted plots was similar to that seen with a timestep of 0.1 seconds.

On reducing the timestep down to 0.1, 0.01, 0.001 and 0.00001 seconds, it is seen that the solution obtained converge towards the solution for the problem using this scheme. As the timestep is reduced, the amplitude of the oscillations in the plot is also reduced, and the front becomes sharper. This trend is observed for all cases, until exactly the same solutions are achieved with a timestep of 0.001 and 0.00001 seconds. It is interesting to note that both the compact stencil and the standard finite volume schemes display the same general trend when different timestep lengths are considered.

6.6 TEST CASE - CONTAMINANT TRANSPORT FROM AN UPSTREAM STRIP SOURCE WITH DECAY

In the previous sections we considered the transport of pollutants when they do not react with the porous medium's skeleton, or degrade over time. This is only the simplest case and in real situations these phenomena can be neglected only on some occasions. The derivation and application of terms to describe the sorption and adsorption of the dissolved pollutant by the porous media is described in detail by Kovraik [10]. By making the assumption that the pollutant does not react with the porous medium, thus the processes of sorption and adsorption do not occur, the radioactive decay of this ideal tracer is described by equation (6.2) as per Bear [2], De Wiest [4] and Kovraik [10]. The problem discussed in the previous section is repeated here using and a range of levels of contaminant decay to investigate the effect of decay on the concentration of pollutant. The rate of decay of the contaminant species is determined by the coefficient λ in equation (6.2). This decay coefficient represents the rate of decay of the contaminant species per second (i.e. the reciprocal of the solute's mean half life). Other degrading processes (of chemical or microbiological manner) can be represented by different values of the constant λ. The decay coefficient values chosen for this investigation and the corresponding levels of decay associated with each value at the end of the simulation time is shown in Table 6.2.

λ	***Decay %***
0.0	0.0
0.05	18.13
0.1	32.97
0.2	55.07
0.5	86.47

Table 6.2: Decay coefficient values and the corresponding levels of decay at $t = 4$ sec.

The calculated concentration profiles using Mesh 2 along the horizontal centre line through the domain at time $t = 4.0$ seconds using decay coefficients

(λ) of 0.0, 0.05, 0.1, 0.2 and 0.5, for Pe = 0.05, 1.0 and 50.0 are shown in Figure 6.20, Figure 6.21 and Figure 6.22 respectively.

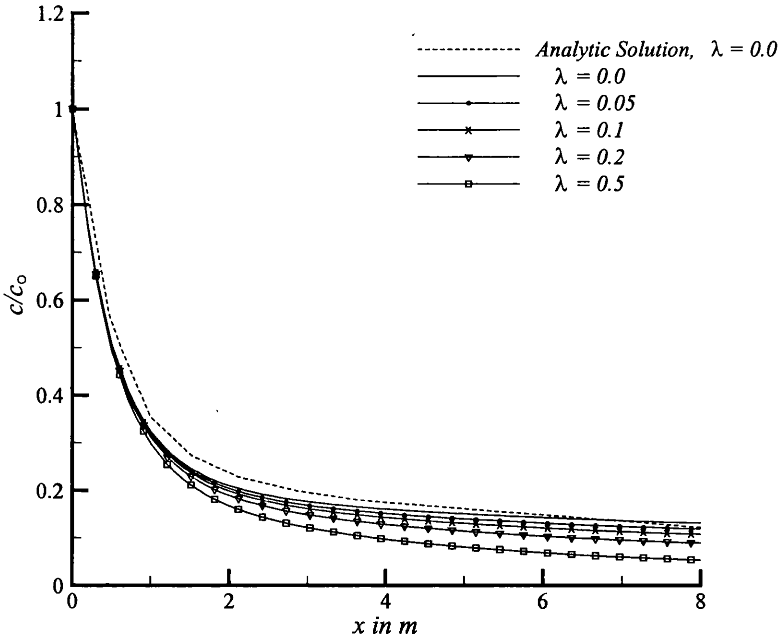


Figure 6.20: Concentration profiles using Mesh 2 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe =$ 0.05, using decay coefficients (λ) of 0.0, 0.05, 0.1, 0.2 and 0.5.

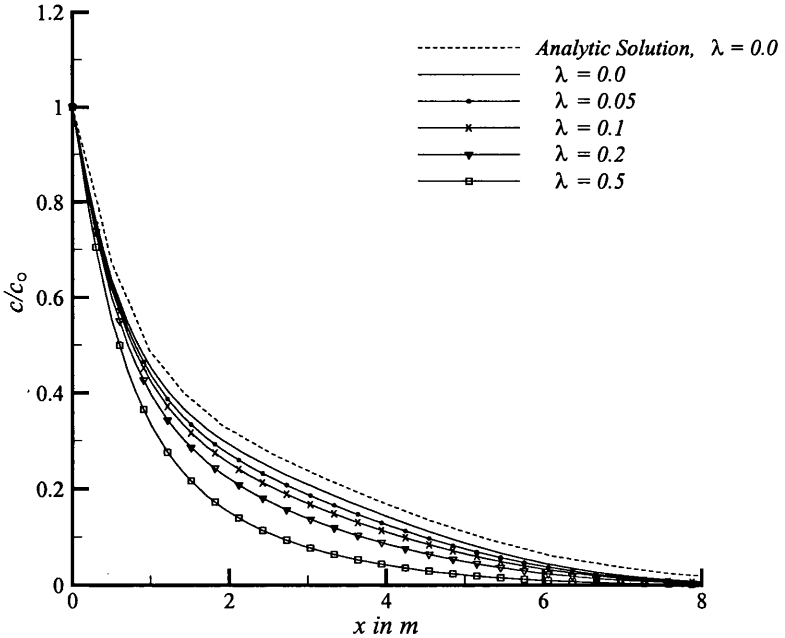


Figure 6.21: Concentration profiles using Mesh 2 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 1.0$, using decay coefficients (λ) of 0.0, 0.05, 0.1, 0.2 and 0.5.

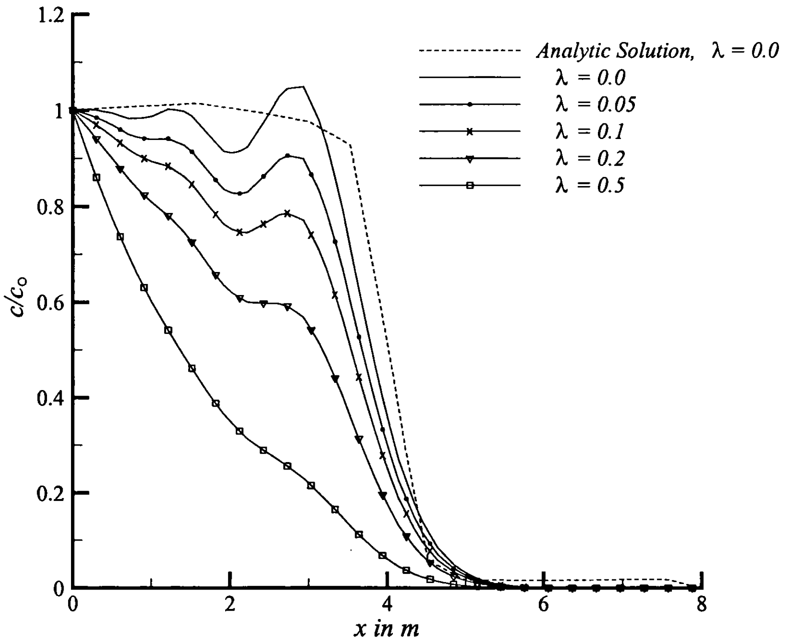


Figure 6.22: Concentration profiles using Mesh 2 along $y = 2.5$ m at $t = 4.0$ sec. for $Pe = 50.0$, using decay coefficients (λ) of 0.0, 0.05, 0.1, 0.2 and 0.5.

For the three cases shown, the required level of decay after 4 seconds is achieved. The first two cases, Figure 6.20 and Figure 6.21, show decay plots that are smooth, and in general are of the same shape, but with decreased concentration levels, as the numerical solution achieved with no decay. In the advection dominant case shown in Figure 6.22, as the rates of decay increase, the sharpness of the front reduces and the spatial oscillations behind the front position are damped as compared to the numerical solution with no decay. It is of note that at high rates of decay ($\lambda = 0.2, 0.5$), the pollutant is decaying faster than the travel of the front across the domain, and for the case of $\lambda = 0.5$, the pollutant species is virtually extinct except at the upstream inflow boundary where $x = 0$.

6.7 TEST CASE - TRANSPORT OF A CONTAMINANT SLUG FROM AN UPSTREAM STRIP SOURCE

In this case, we consider a problem of two-dimensional transport from a strip source at the upstream end. Previously, the pollutant is injected at a constant rate throughout the simulation by application of a fixed concentration boundary condition along boundary B-C (see Figure 6.2). Here, we wish to model the situation where a pollutant slug injected into the domain for 1 second, and then the injection of pollutant is ceased. This is achieved by applying the following boundary conditions to the domain:

$$\begin{aligned} c_0 &= 0.0 \text{ at } t = 0, \\ c &= \begin{cases} 1.0 \text{ , when t} \leq 1.0 \text{ sec,} \\ 0.0 \text{ , when t} > 1.0 \text{ sec,} \end{cases} \quad \text{on side B-C} \\ c &= 0.0 \text{ on side A-B and C-D,} \\ \partial c / \partial y &= 0.0 \text{ on side D-E and A-F,} \\ \partial c / \partial x &= 0.0 \text{ on side E-F.} \end{aligned}$$

Once more, the flow is assumed to be unidirectional along the x - axis having a velocity equal to 1.0 m/s. The medium is homogeneous and isotropic, and the longitudinal dispersivity is assumed equal to the lateral dispersivity. Three cases with Peclet numbers of 0.05, 1.0 and 50.0 were considered, and the results obtained for each case are shown below. All numerical simulations in this section were run on the finer Mesh 2 (see Figure 6.3).

6.7.1 Contaminant Slug in a Dispersion Dominant System - Pe = 0.05

The results obtained for the numerical simulation of a contaminant slug in a highly dispersion dominant system with a Peclet number of 0.05 using the compact stencil scheme are shown in Figure 6.23, Figure 6.24 and Figure 6.25. The solutions were obtained using a time step length of $dt = 0.001$ seconds. Figure 6.23 and Figure 6.24 show the concentration profiles along the horizontal centre line of the domain (i.e. $y = 2.5$) at various times in the simulation, while

Figure 6.25 shows the concentration contours at t = 0.1, 1.1, 1.2, 1.3 and 1.5 seconds.

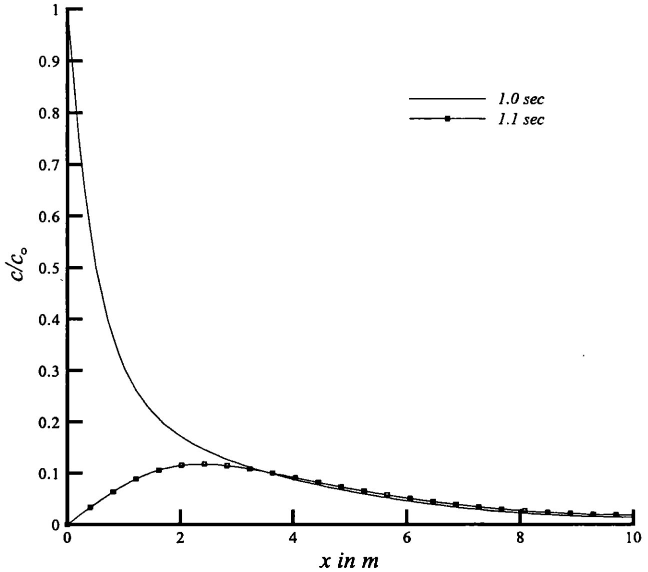


Figure 6.23: Concentration profiles using Mesh 2 along y = 2.5 at t = 1.0 and t = 1.1 sec. for Pe = 0.05, using the compact stencil scheme.

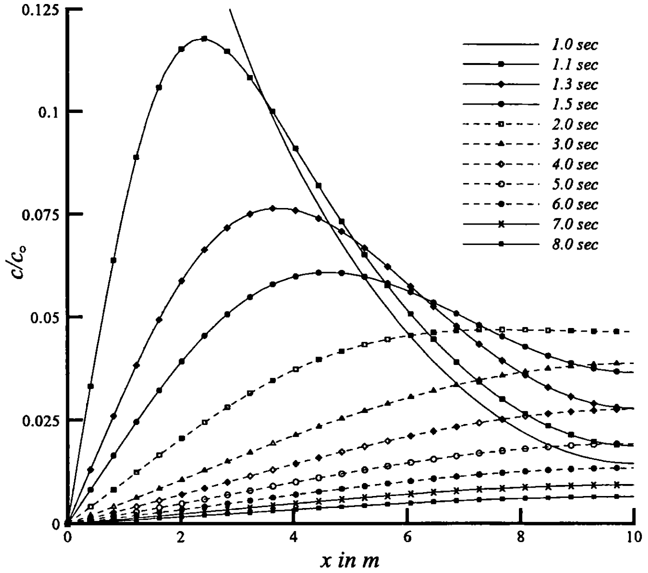


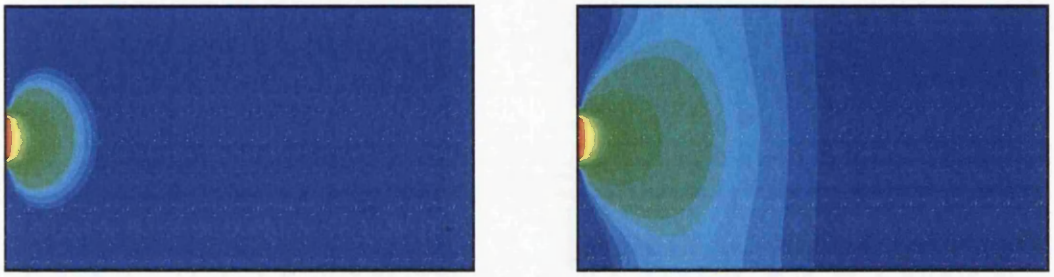
Figure 6.24: Concentration profiles using Mesh 2 along y = 2.5 at various times between 1.0 and 8.0 sec. for Pe = 0.05 using the compact stencil scheme.

The plot of the concentration profiles using along the horizontal centre line of the domain at t = 1.0 and t = 1.1 seconds for a Peclet number of $Pe = 0.05$ is illustrated in Figure 6.23. This clearly shows the situation along this line at the end of the period of injection of the pollutant slug (t = 1.0 seconds), and immediately after it (t = 1.1 seconds) when the injection of pollutant has ceased. The pollutant concentration at the point $x = 0.0$ m drops from $c = 1.0$ at $t = 1.0$ seconds, to $c = 0.0$ at $t = 1.1$ seconds. This is as described by the boundary conditions above, and hence one is assured that the slug injection condition is correct.

Figure 6.24 shows concentration profiles along the same line as shown in Figure 6.23, but for times between 1.0 and 8.0 seconds. It is seen that for the solution obtained at $t = 1.1$ seconds, the concentration profile has a sharp narrow peak. Solutions at increasing times show increasing rounding of the peak, the maximum concentration is reduced, and the slug is seen to diffuse throughout the whole domain. This outcome is supported by considering the pollutant concentration contours show in Figure 6.25. The high Peclet number enforced on this problem ensures that the equation system is diffusion dominant. Once the slug is released from boundary B-C at $t \geq 1.0$ second, it diffuses quickly into the domain. After 1.5 seconds, using the contour scale shown, no variability in pollutant concentration is visible in the domain.

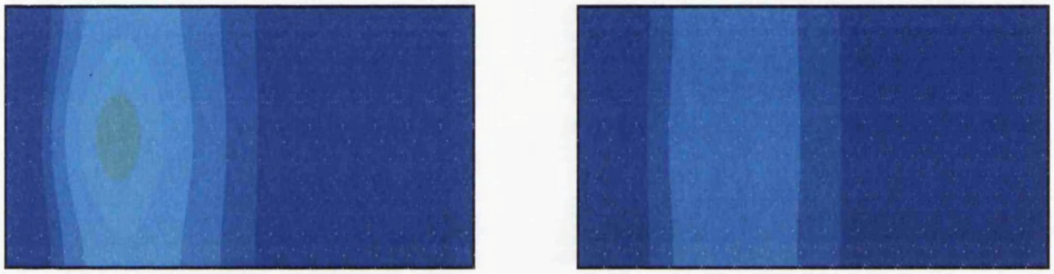
6.7.2 Contaminant Slug in a Dispersion And Advection Equally Dominant System - Pe = 1.0

A Peclet number of 1.0 represents a dispersion and advection equally dominant system. The results modelling a contaminant slug through the domain under these conditions using the compact stencil scheme are shown in Figure 6.26 through to Figure 6.28. The solutions were obtained using a time step length of $dt = 0.001$ seconds. Figure 6.26 show the concentration profiles along $y = 2.5$ at various times in the simulation, while Figure 6.27 shows the concentration contours at t = 1.0, 1.1, 2.0, 4.0, 6.0 and 8.0 seconds. Figure 6.28 illustrates a three dimensional representation of the concentration contours, with pollutant concentration on the vertical axis, at $t = 1.1$, 1.5, 2.0 and 3.0 seconds.



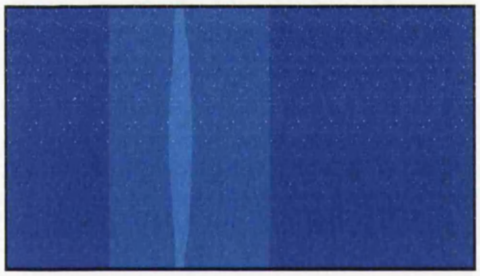
a) 0.1 sec.

b) 1.0 sec.



c) 1.1 sec.

d) 1.2 sec.



e) 1.3 sec.

f) 1.5 sec.

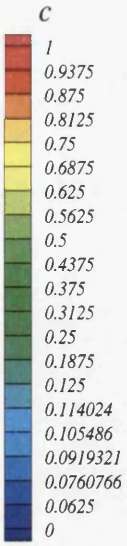


Figure 6.25: Concentration contours at various times for $Pe = 0.05$ on Mesh 2 using the compact stencil scheme.

By comparing Figure 6.24 and Figure 6.26, it is evident that the there is far less diffusion in the latter case. This is a direct result of the choice of Peclet number. The concentration plots of Figure 6.26 shows that the pollutant slug has not diffused through the domain as quickly as the previous case. The slug remains as an entity until it approaches the far side boundary at $t = 5.0$ seconds. Then there occurs a build up of concentration in this region due to the no-flow condition applied to boundary E-F. Similar conclusions can be made concerning the results obtained using $Pe = 0.05$. However, in this case the slug approaches the far side boundary almost immediately after the slug injection has come to an end (see Figure 6.26).

Figure 6.27 reinforces what has already been said, and it is clear that the reduced rate of diffusion causes the slug to remain as a region of high concentration for an extended period as compared to the previous case (see Figure 6.25). The 3-D concentration plots of Figure 6.28 clearly illustrate the path of the pollutant slug. It is seen that the centre point of the slug loses concentration as the slug diffuses out towards the horizontal boundaries of the domain (boundaries A-F and D-E).

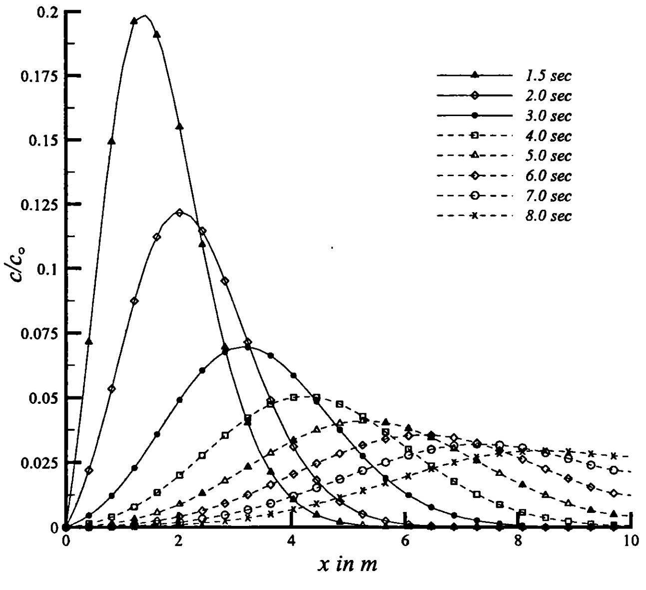
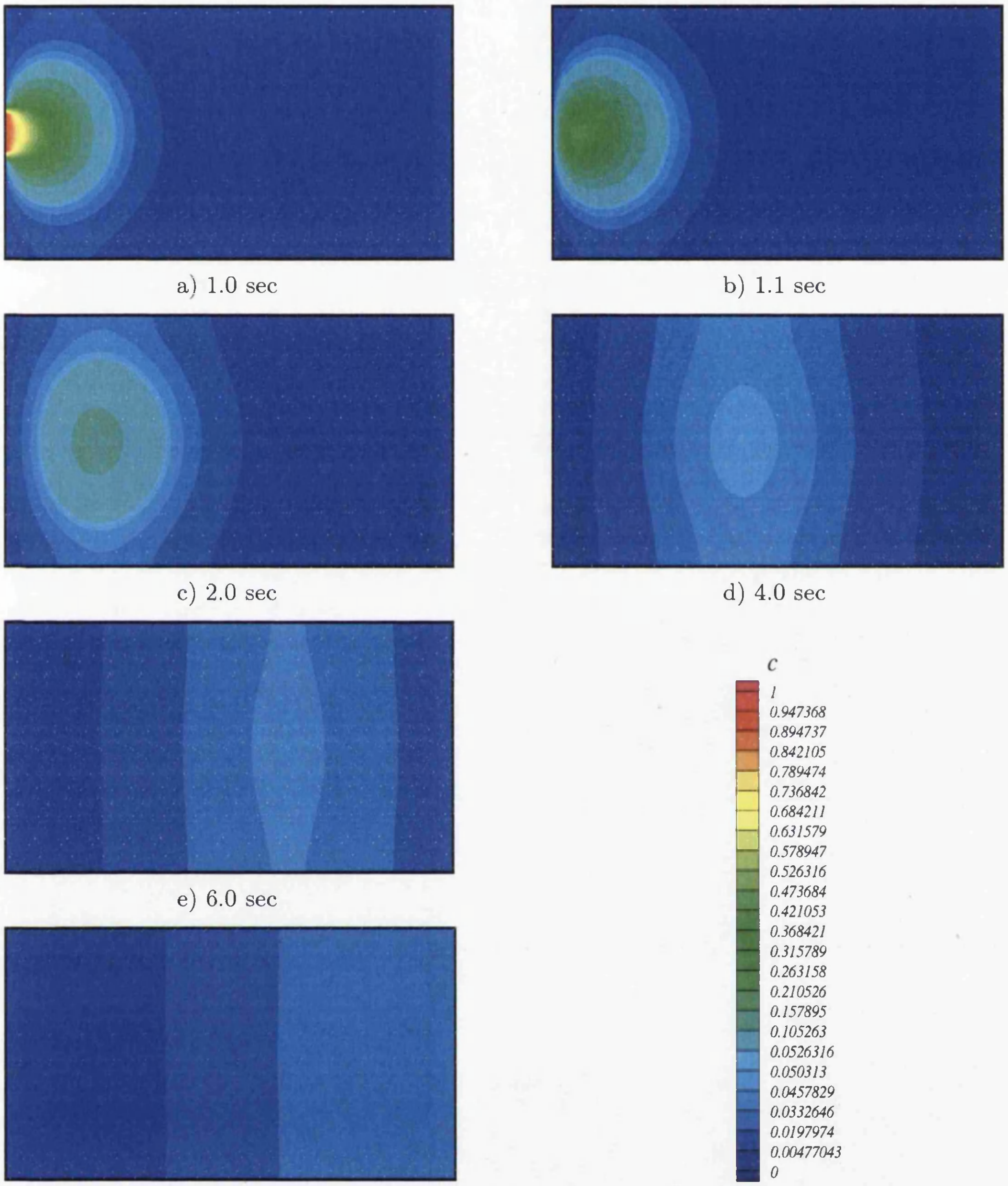


Figure 6.26: Concentration profiles using Mesh 2 along $y = 2.5$ at various times between 1.5 and 8.0 sec. for $Pe = 1.0$ using the compact stencil scheme.



a) 1.0 sec

b) 1.1 sec

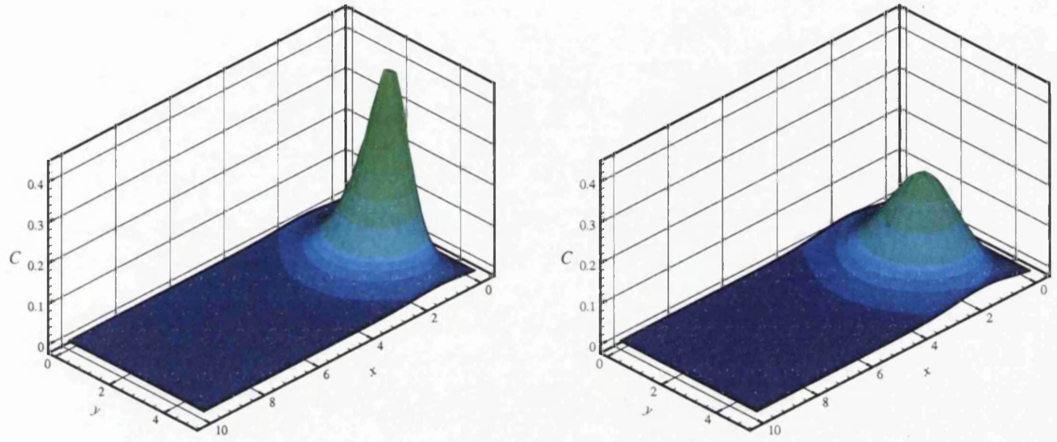
c) 2.0 sec

d) 4.0 sec

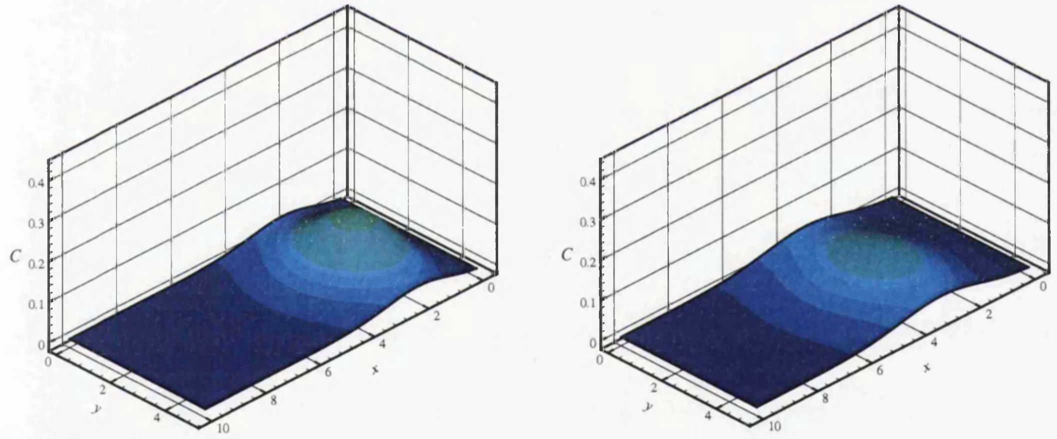
e) 6.0 sec

f) 8.0 sec

Figure 6.27: Concentration contours at various times for $Pe = 1.0$ on Mesh 2 using the compact stencil scheme.



a) 1.1 sec. b) 1.5 sec.



c) 2.0 sec. d) 3.0 sec.

Figure 6.28: 3-D Concentration contours with pollutant concentration on the vertical axis at t = 1.1, 1.5, 2.0 and 3.0 sec., and Pe = 1.0 for Mesh 2 using the compact stencil scheme. (The contour legend is the same as that in Figure 6.27).

6.7.3 Contaminant Slug in a Highly Advection Dominant System - $Pe = 50.0$

The results for a highly advection dominant system, a result of the chosen Peclet number of 50.0, using the compact stencil scheme are shown in Figure 6.29 and Figure 6.30. The solutions were obtained using a time step length of $dt = 0.001$ second. Figure 6.29 show the concentration profiles along $y = 2.5$ at various times in the simulation, while Figure 6.30 shows the concentration contours at $t = 0.5$, 1.1, 2.0, 4.0, 6.0 and 8.0 second.

From Figure 6.29, it is evident that as the maximum concentration at the centre of the slug reduces slightly as time increases, and the width of the slug increases accordingly due to the small rate of diffusion. However, on comparison with the other two cases considered, the rate of diffusion of the pollutant slug is miniscule, and the slug remains as a body of high concentration throughout the simulation time as clearly illustrated by Figure 6.30. Moreover, the slug does not make contact with the far side boundary at all, again as a result of the small diffusion rates. The oscillations caused by the advection terms, as discussed in section 6.4.3, are visible in both the concentration profile and contours (Figure 6.29 and Figure 6.30 respectively). Suitable approaches to correct this have been mentioned, but have not been implemented into the scheme at this time.

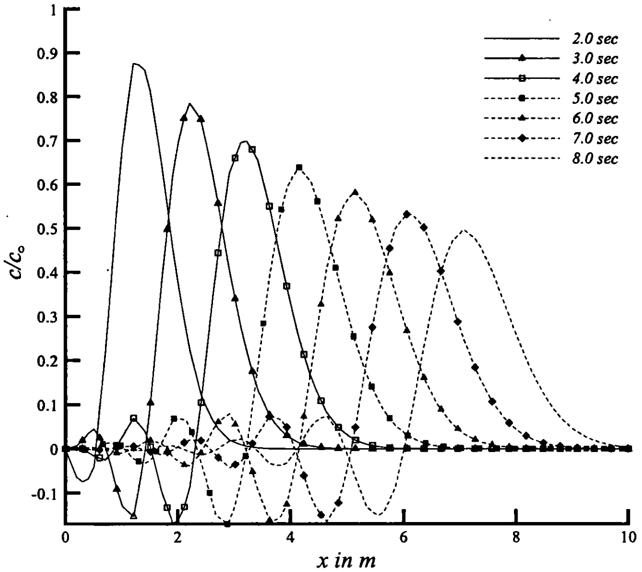


Figure 6.29: Concentration profiles using Mesh 2 along $y = 2.5$ at $t = 1.0$, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0 and 8.0 sec. for $Pe = 50.0$ using the compact stencil scheme.

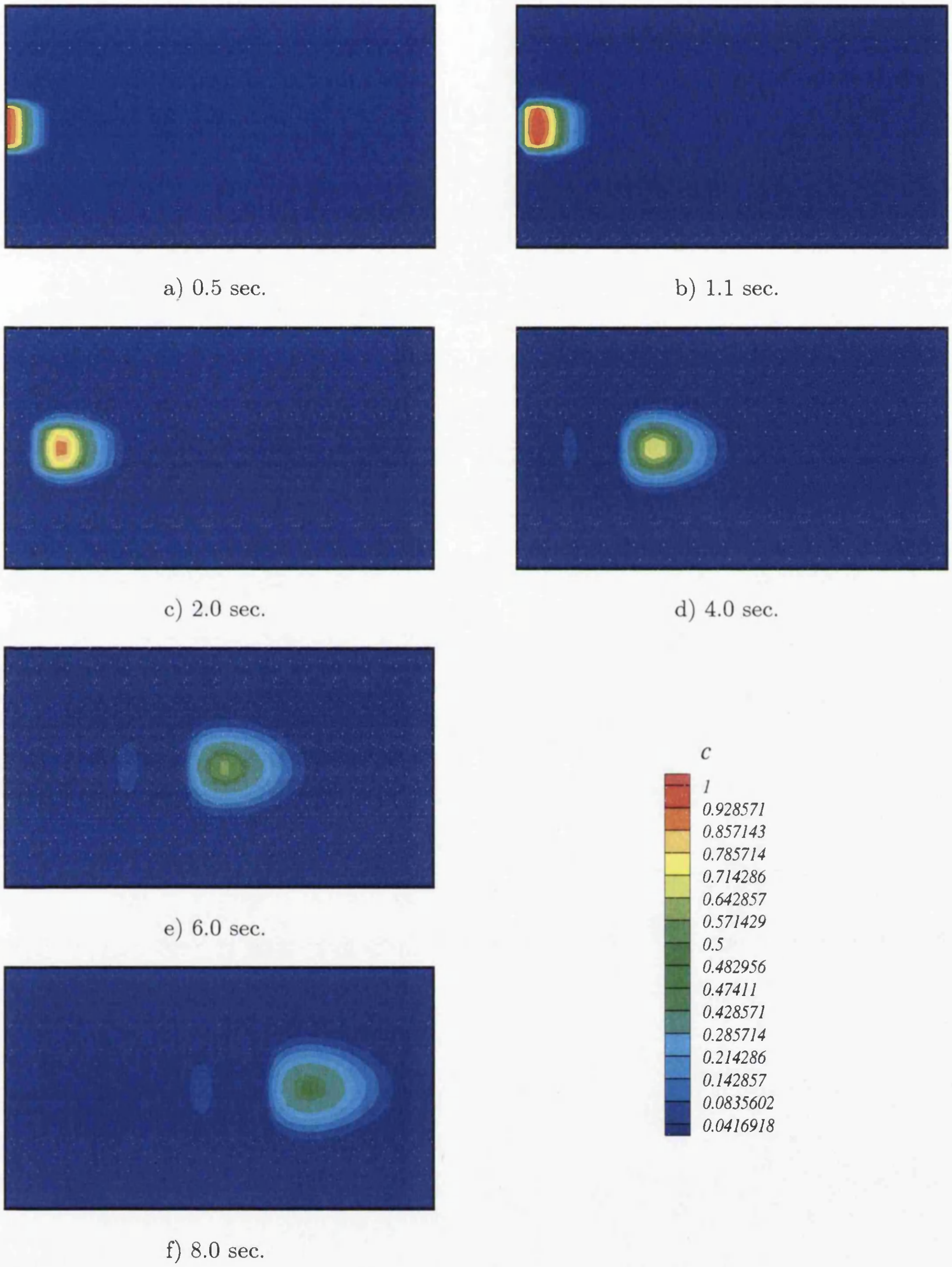


Figure 6.30: Concentration contours at various times for $Pe = 50.0$ on Mesh 2 using the compact stencil scheme.

6.8 TEST CASE - TRANSPORT OF A CONTAMINANT SLUG FROM AN UPSTREAM STRIP SOURCE WITH DECAY

In this case, we consider a problem of two-dimensional transport of a pollutant slug from a strip source at the upstream end of the domain, with the effect of exponential decay on the pollutant over time. The initial and boundary conditions, and the domain size and mesh used are identical to those used in section 6.4. The problem is solved for the three Peclet numbers considered throughout this chapter (0.05, 1.0 and 50.0), and a range of levels of contaminant decay to investigate the effect of decay on the concentration of the pollutant slug. The values for the decay coefficient λ chosen for this investigation, and the corresponding levels of decay associated with each value at the end of the simulation time as used in section 6.6 are shown in Table 6.2. The results obtained are shown in Figure 6.31, Figure 6.32 and Figure 6.33 respectively. The plots show the concentration profiles along the horizontal centre line of the domain at various times in the simulation. The solutions were obtained using a time step length of $dt = 0.001$ second on Mesh 2.

In all cases the plots are smooth, and show the required rates of decay for each decay coefficient as listed in Table 6.2.

6.9 CLOSURE

In this chapter the modelling of contaminant transport through aquifers over a horizontal plane by the application of the vertex-centred edge-based finite volume scheme has been presented. To the authors' knowledge, this is the first application of this scheme in solving contaminant transport problems.

Firstly, the contaminant transport from a strip source at the upstream end of a homogeneous and isotropic domain was considered. The flow was assumed to be unidirectional. The standard and compact stencil finite volume schemes were successfully utilised to solve the problem for varying Peclet number and mesh density. The numerical solutions were compared to the analytical solution for the problem and a close correlation was achieved. The solutions also compared well to a published numerical solution obtained using the more computationally expensive finite element method.

The disadvantage of the standard five-node stencil was clearly evident from a spatial oscillation in the contour plots of the solutions. However, the excellent compact scheme was comparable to the analytical solution. The front tracking capability of the scheme is shown by the solution to the highly advection dominant system. In addition, by adding non-zero values of the decay coefficient into the governing equation, the decay of the pollutant species over time was successfully modelled. Finally, the simulation of the injection of a pollutant slug into the system for a set time was successfully dealt with, and the same problem was considered with decay of the contaminant species.

Therefore, this efficient and elegant numerical method can provide accurate solutions for a wide range of contaminant transport problems.

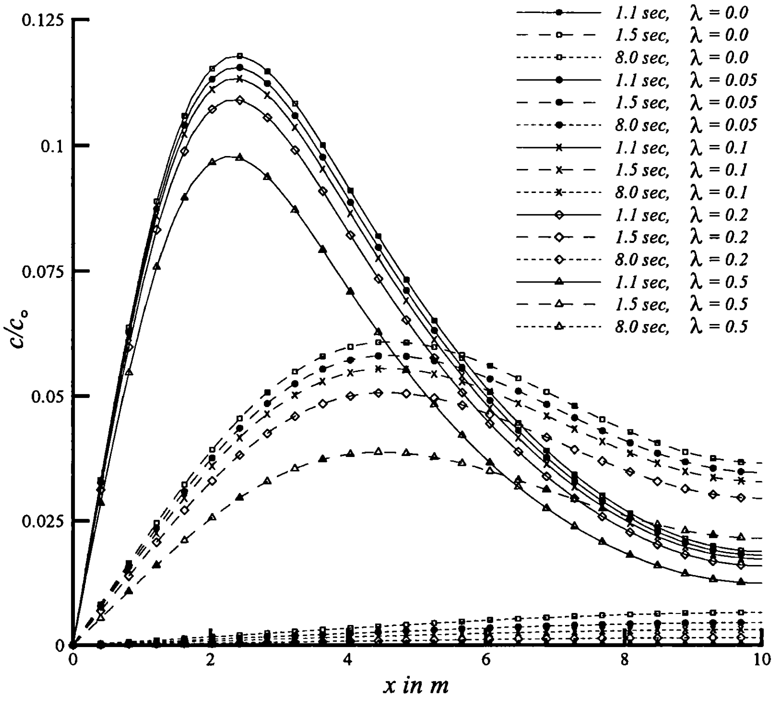


Figure 6.31: Concentration profiles using Mesh 2 along $y = 2.5$, at $t = 1.1$, 1.5 and 8.0 sec., for $Pe = 0.05$, using the compact stencil scheme, and $\lambda = 0.0$, 0.05, 0.1, 0.2 and 0.5.

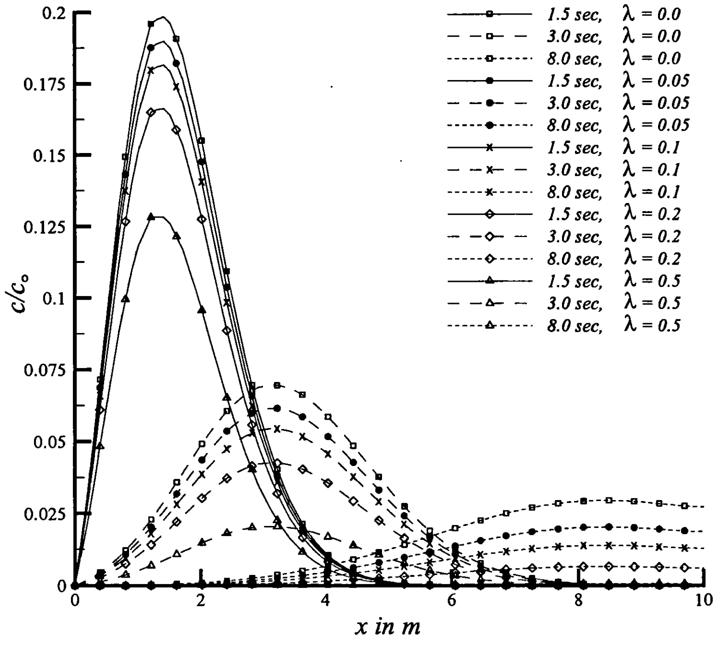


Figure 6.32: Concentration profiles using Mesh 2 along $y = 2.5$, at $t = 1.5$, 3.0 and 8.0 sec., for $Pe = 1.0$, using the compact stencil scheme, and $\lambda = 0.0$, 0.05, 0.1, 0.2 and 0.5.

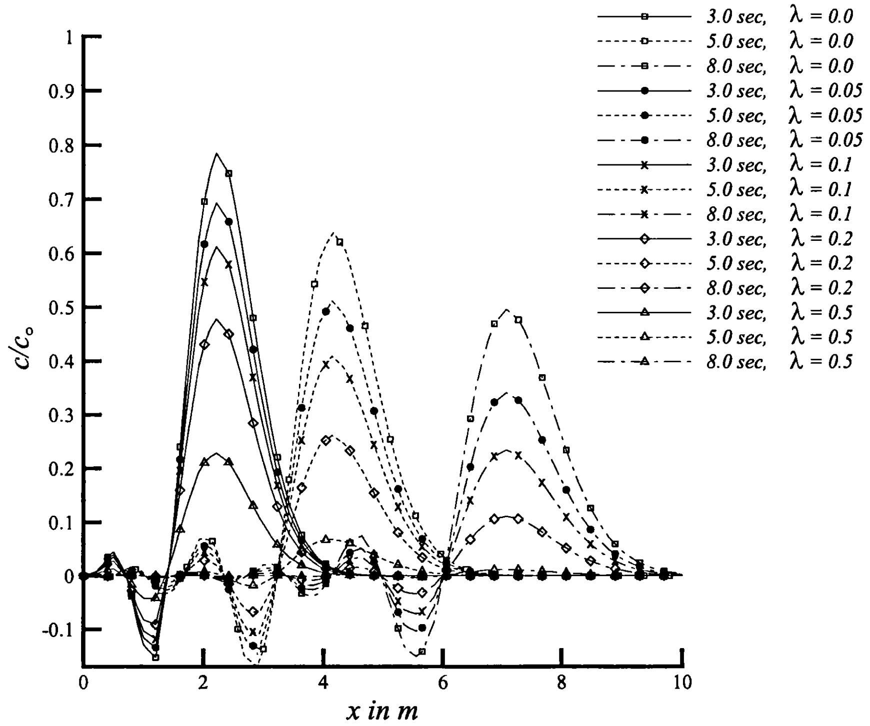


Figure 6.33: Concentration profiles using Mesh 2 along $y = 2.5$, at $t = 3.0$, 5.0 and 8.0 sec., for $Pe = 50.0$ using the compact stencil scheme, and $\lambda = 0.0$, 0.05, 0.1, 0.2 and 0.5.

REFERENCES

[1] Beam R.M. and Warming R.F., An implicit finite-difference algorithm for hyperbolic systems in conservation-law form - application to Eulerian gasdynamic equations, *Journal of Computational Physics*, **22**:87-110, 1976.

[2] Bear J., Hydraulics of Groundwater (McGraw-Hill Series in Water Resources and Environmental Engineering), McGraw Hill College Div., 1979.

[3] Cleary R.W. and Adrian D.D., Analytic solution of the convective-dispersive equation for cation adsorption. Soil Sci. Soc. Amer. Proc., **37**: 197-199, 1973.

[4] De Wiest R.J.M., ed. Flow Through Porous Media, Academic Press, New York, 1969.

[5] Gottardi G. and Venutelli M., Moving finite element model for one-dimensional infiltration in unsaturated soil. *Water Resources Research*, **28**(12): 3259-3267, 1992.

[6] Gray W.G. and Pinder G.F., An analysis of the numerical solution of the transport equation. *Water Resources Research*, **12**(3): 547-556, 1976.

[7] Hirsch C., Numerical Computation of Internal and External Flows, *Computational Methods for Inviscid and Viscous Flows*, volume 2, Wiley 1990.

[8] Jameson A., Schmidt W. and Turkel E., Numerical simulation of the Euler equations by finite volume methods using Runge-Kutta timestepping schemes, *AIAA 5th Computational Fluid Dynamics Conference,* 1981.

[9] Kebaiti A.L. and Thomas G.W., A Two-Phase Coning Model Using Alternating Direction Galerkin Procedure, paper *SPE 5724* presented at the SPE-AIME Symposium on Numerical Simulation of Reservoir Performance, Los Angeles, Feb. 1976.

[10] Kovraik K., Numerical Models in Groundwater Pollution, Springer-Verlag, Berlin, 2000.

[11] Lewis R.W., Vemer E.A. and Zienkiewicz O.C., A Finite Element Approach to Two-Phase Flow in Porous Media, Finite Elements in Fluids, John Wiley & Sons Ltd., London, 1975.

[12] Lindstrom F.T., Hague R., Freed V.H and Boersma L., Theory on movement of some herbicides in soils, Linear diffusion and convection of chemicals in soils. *Environ. Sci. Technol.*, **1**: 561-565, 1967.

[13] MacCormack R.W. and Baldwin B.S., A numerical method for solving the Navier-Stokes equations with application to shock-boundary layer interactions, *AIAA paper*, 1-75, 1975.

[14] Malan A.G., Investigation Into The Continuum Thermodynamic Modelling Of Investment Casting Shell-Mould Drying, PhD Thesis, School of Engineering, University of Wales Swansea, 2002.

[15] Malan A.G., Lewis R.W. and Nithiarasu P., An improved unsteady, unstructured, artificial compressibility, finite volume scheme for viscous incompressible flows: Part I. Theory and implementation. *International Journal for Numerical Methods in Engineering*, **54**(5): 695-714, 2002.

[16] Pinder G.F. and Gray W.G., Finite element simulations in surface and subsurface hydrology. Academic, Orlando, Fla., 1977.

[17] Selim H.M. and Mansell R.S., Analytical solution of the equation for transport of reactive solutes through soils. *Water Resources Research*, **12**(3): 528-532, 1976.

[18] Slichter C.S., Field measurements of the rate of movement of underground waters, Water Sup. Paper No. 140, U.S. Geol. Surv., 1905.

[19] Sorensen K.A., A Multigrid Acceleration Procedure for the Solution of Compressible Fluid Flows on Unstructured Hybrid Meshes. Doctor of Philosophy Thesis submitted to the University of Wales Swansea, Swansea, 2002.

[20] Spivak S., Price H.S. and Settari A., Solution of the Equations for Multi-Dimensional Two-Phase Immiscible Flow by Variational Methods, *Soc. Pet. Eng. J.,* 27-41, 1977.

[21] Sun N.-Z. and Yeh W.W.-G., A proposed upstream weight numerical method for simulating pollutant transport in groundwater. *Water Resources Research*, **19**(6): 1489-1500, 1983.

[22] von Neumann J. and Richtmyer R.D., A method for the numerical calculations of hydrodynamical shocks, *Journal of Mathematical Physics,* **21**:232-237, 1950.

[23] Yeh G.T., An orthogonal-upstream finite element approach to modeling aquifer contaminant transport, *Water Resources Research*, **22**(6): 952-964, 1986.

[24] Zienkiewicz O.C. and Parekh C.J., Transient Field Problems: Two-Dimensional and Three- Dimensional Analysis by Isoparametric Finite Elements, *Int. J. Num. Meth. Eng.*, **2**: 61-71, 1970.

Chapter 7

COUPLED CONTAMINANT TRANSPORT THROUGH A SATURATED POROUS MEDIUM

7.1 INTRODUCTION

In this chapter the coupled model of miscible contaminant transport through a saturated porous medium, such as an aquifer, is presented. The aquifer is represented by a horizontal plane and the governing equations are discretised by the application of the vertex-centred edge-based finite volume scheme. The presented model solves Richards equation for flow and a transport equation for each contaminant species, the two being coupled by the Darcy velocity term. This is an extension to the problem stated in chapter 6, which considered the solution of the transport equation for the contaminant species only, and a constant flow velocity was assumed.

The numerical solution of transport problems with sharp transition or high advection are surprisingly difficult to solve using traditional finite element and finite difference schemes ([3]-[11]). When the advective term becomes dominant, the transport equation tends to become hyperbolic and the solutions based on traditional finite element and finite difference methods are always plagued with either spurious oscillations (overshoot) or excessive slope of the concentration front (smearing), due to numerical dispersion. Yeh and Ward [9] and McDonald and Harbaugh [6] have presented coupled flow and contaminant transport models, called FEMWATER and MODFLOW respectively. The flow models used are based on a standard Galerkin finite element approximation for the calculation of the pressure head. Holder *et al.* [4] states that this approach is not element wise conservative, and velocities are at best first-order accurate. Holder *et al.* [4] present a three-dimensional groundwater model called FLOTRAN, which solves the governing equations for flow and contaminant transport using the finite element method. They claim that the model provides excellent mass balance characteristics, fast solution times, and flexibility in choosing modelling conditions, such as grid definition boundary conditions, and wells. In addition, FLOTRAN incorporates linear sorption, bio-degradation and first-order decay into its solution of the ground water contaminant transport equations, and can model multiple components, allowing the user to simulate several contaminants in one model run.

Gottardi [2] presents a finite element program for modeling two-dimensional pollutant transport in confined and unconfined aquifers, where the flow and pollutant transport equations are solved by using either a Galerkin finite element or a control-volume finite element formulation. The absorption and biological decay of the pollutant are integrated into the governing equations used. The solution obtained using the two schemes for two problems of which the analytical solution is known are compared. The two schemes produced similar results. For definitions and a brief literature review dealing with similar problems, please refer back to section 6.1.

The coupled miscible contaminant transport resulting from the injection and subsequent extraction of a contaminant into an aquifer from the same well on a homogeneous and isotropic domain will be considered. The problem will be run on a structured quadrilateral element using the standard and compact stencil finite volume schemes for a range of Peclet numbers. Numerical experiments are then conducted to investigate the effect of the rate of dispersion on the numerical solution. The effect of radioactive decay on the solution is then investigated, and finally, conclusions will be drawn from the work undertaken.

7.2 GOVERNING EQUATIONS

The equations that describe the coupled flow and contaminant transport are set out in this section. The flow equation is described by Holder *et al.* [4] and is of the form,

$$\frac{\partial \theta(\psi)}{\partial t} + S_s \frac{\partial h}{\partial t} + \nabla.\mathbf{v} = q \tag{7.1}$$

where $\mathbf{v}$ is the Darcy velocity, q represents sources and sinks, S_s the specific storage, θ the moisture content, h the hydraulic head, and $\psi = h - z$. The Darcy velocity is defined by;

$$\mathbf{v} = \begin{bmatrix} v_{xx} & v_{xy} \\ v_{xy} & v_{yy} \end{bmatrix} \tag{7.2}$$

By making the same assumption as in chapter 5 concerning the relationship between the water saturation and moisture content, we have,

$$S_w = \frac{\theta}{\phi} \tag{7.3}$$

where ϕ is the material porosity. Assuming that the domain is fully saturated with water at time $t = t_0$, the flow equation (7.1) simplifies to;

$$S_s \frac{\partial h}{\partial t} + \nabla.\mathbf{v} = q \tag{7.4}$$

and this is the form of the equation that is considered. The simplification is justified due to the water saturation $S_w = 1.0$, which implies that the moisture content θ is constant from equation (7.3). By Darcy's law,

$$\mathbf{v} = -\mathbf{K}(\psi)\nabla h \tag{7.5}$$

where the hydraulic conductivity tensor **K** is defined as,

$$\mathbf{K} = \begin{bmatrix} k_{xx} & k_{xy} \\ k_{xy} & k_{yy} \end{bmatrix} \tag{7.6}$$

The Darcy velocities and moisture content computed from the flow are used in the system of transport equations for each contaminant species. These equations model the advection, diffusion, and chemical reactions of species in the system. Each equation is of the form

$$\frac{\partial \theta c}{\partial t} + \nabla.(\mathbf{v}c - \mathbf{D}\nabla c) = q\hat{c} + g \tag{7.7}$$

where c is the contaminant concentration, **D** the diffusion/dispersion tensor, and g incorporates chemical reactions, and/or decay of the contaminant species. g may be a function of several species in the system. At sources $\hat{c} = 1.0$, while at sinks $\hat{c} = c$. Consequently, c is defined as a dimensionless ratio of contaminant present compared to initial or source concentration. The dispersion tensor is defined by;

$$\mathbf{D} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{7.8}$$

Due to the fact that the mixed derivative terms will be omitted from the formulation, the off diagonal terms in tensors (7.2), (7.6) and (7.8) are zero.

7.3 SOLUTION PROCEDURE

The solution procedure utilised to solve the described problem is discussed here. By implying that the second mixed derivatives are zero, the expansion of the flow equation (7.4) gives;

$$S_s\frac{\partial h}{\partial t}-\frac{\partial}{\partial x}\left(k_{xx}\frac{\partial h}{\partial x}\right)-\frac{\partial}{\partial y}\left(k_{yy}\frac{\partial h}{\partial y}\right)=q \tag{7.9}$$

The spatial derivative terms in the equation are discretised by application of Green's theorem, and by the assumption that the spatial derivative is linear over the control volume Ω_m. Integration over the control volume gives,

$$\int_\Omega S_s\frac{\partial h}{\partial t}\;d\Omega-\int_\Omega\frac{\partial}{\partial x}\left(k_{xx}\frac{\partial h}{\partial x}\right)\;d\Omega-\int_\Omega\frac{\partial}{\partial y}\left(k_{yy}\frac{\partial h}{\partial y}\right)\;d\Omega=\int_\Omega q\;d\Omega \tag{7.10}$$

and the application of the edge based discretisation and the Euler backward difference scheme as discussed in chapter 2 gives for all nodes m in the domain;

$$\begin{aligned}&S_s\big|_m\left(\frac{h_m^{n+1}-h_m^n}{\Delta t}\right)\Omega_m\\&=\frac{\partial}{\partial x}\left(\left(k_{xx}\right)_{mn}\frac{\partial h^n}{\partial x}\right)\bigg|_m\Omega_m+\frac{\partial}{\partial y}\left(\left(k_{yy}\right)_{mn}\frac{\partial h^n}{\partial y}\right)\bigg|_m\Omega_m+q_m\Omega_m\end{aligned} \tag{7.11}$$

where h_m^n = pressure head at node m at time n, and h_m^{n+1}= pressure head at node m at time $n+1$. Hence, rearranging results in the pressure head update which is given by equation (7.12).

$$\begin{aligned}h_m^{n+1}&=\left(\frac{\partial}{\partial x}\left(\left(k_{xx}\right)_{mn}\frac{\partial h^n}{\partial x}\right)\bigg|_m\Omega_m+\frac{\partial}{\partial y}\left(\left(k_{yy}\right)_{mn}\frac{\partial h^n}{\partial y}\right)\bigg|_m\Omega_m+q_m\Omega_m\right)\frac{\Delta t}{\Omega_m\,S_s\big|_m}+h_m^n\\&=\Lambda_m\frac{\Delta t}{\Omega_m\,S_s\big|_m}+h_m^n\end{aligned} \tag{7.12}$$

where,

$$\Lambda_m=\frac{\partial}{\partial x}\left(\left(k_{xx}\right)_{mn}\frac{\partial h^n}{\partial x}\right)\bigg|_m\Omega_m+\frac{\partial}{\partial y}\left(\left(k_{yy}\right)_{mn}\frac{\partial h^n}{\partial y}\right)\bigg|_m\Omega_m+q_m\Omega_m \tag{7.13}$$

Once the pressure head is updated for timestep $n+1$, the contaminant transport equation can then be considered. The transport equation, that is equation (7.7), is coupled to the flow equation (equation (7.1)) through the Darcy velocity terms v_{xx} and v_{yy}, defined by equation (7.14) below. This coupling is clearly shown on the program flow chart illustrated in Figure 7.1.

$$v_{xx}\Big|_m = \left(k_{xx}\right)_{mn} \frac{\partial h^n}{\partial x}\bigg|_m \quad , \quad v_{yy}\Big|_m = \left(k_{yy}\right)_{mn} \frac{\partial h^n}{\partial y}\bigg|_m \tag{7.14}$$

Again, by implying that the second mixed derivatives are zero, expansion of the contaminant transport equation (7.7) gives;

$$\begin{gathered} \frac{\partial \theta c}{\partial t} = -\frac{\partial}{\partial x}\left(v_{xx}c\right) - \frac{\partial}{\partial y}\left(v_{yy}c\right) + \\ \frac{\partial}{\partial x}\left(D_{xx}\frac{\partial c}{\partial x}\right) + \frac{\partial}{\partial y}\left(D_{yy}\frac{\partial c}{\partial y}\right) + q\hat{c} + g \end{gathered} \tag{7.15}$$

Again, the spatial derivative terms in the equation are discretised by application of Green's theorem. Integrating over the control volume produces,

$$\begin{gathered} \int_\Omega \frac{\partial \theta c}{\partial t}\ d\Omega = -\int_\Omega \frac{\partial}{\partial x}\left(v_{xx}c\right)\ d\Omega - \int_\Omega \frac{\partial}{\partial y}\left(v_{yy}c\right)\ d\Omega + \\ \int_\Omega \frac{\partial}{\partial x}\left(D_{xx}\frac{\partial c}{\partial x}\right)\ d\Omega + \int_\Omega \frac{\partial}{\partial y}\left(D_{yy}\frac{\partial c}{\partial y}\right)\ d\Omega + \\ \int_\Omega q\hat{c}\ d\Omega + \int_\Omega g\ d\Omega \end{gathered} \tag{7.16}$$

and the application of the edge based discretisation procedure and the Euler backward difference scheme gives for all nodes m in the domain;

$$\begin{aligned}\left(\frac{c_m^{n+1}-c_m^{n}}{\Delta t}\right)\Omega_m\theta &= -\frac{\partial}{\partial x}\left(v_{xx}c_m^{n}\right)\bigg|_m \Omega_m - \frac{\partial}{\partial y}\left(v_{yy}c_m^{n}\right)\bigg|_m \Omega_m + \frac{\partial}{\partial x}\left(\left(D_{xx}\right)_{mn}\frac{\partial c^{n}}{\partial x}\right)\bigg|_m \Omega_m \\ &\quad + \frac{\partial}{\partial y}\left(\left(D_{yy}\right)_{mn}\frac{\partial c^{n}}{\partial y}\right)\bigg|_m \Omega_m + q_m\hat{c}_m\Omega_m - \lambda c_m^{n}\Omega_m \\ &= \Xi_m - \lambda c_m^{n}\Omega_m\end{aligned} \tag{7.17}$$

where,

$$\begin{aligned}\Xi_m &= -\frac{\partial}{\partial x}\left(v_{xx}c_m^{n}\right)\bigg|_m \Omega_m - \frac{\partial}{\partial y}\left(v_{yy}c_m^{n}\right)\bigg|_m \Omega_m + \frac{\partial}{\partial x}\left(\left(D_{xx}\right)_{mn}\frac{\partial c^{n}}{\partial x}\right)\bigg|_m \Omega_m \\ &\quad + \frac{\partial}{\partial y}\left(\left(D_{yy}\right)_{mn}\frac{\partial c^{n}}{\partial y}\right)\bigg|_m \Omega_m + q_m\hat{c}_m\Omega_m - \lambda c_m^{n}\Omega_m\end{aligned} \tag{7.18}$$

Note that the term g which governs the chemical reactions within the system has been substituted with equation (6.2). Rearranging equation (7.17) results in the contaminant concentration update given here.

$$c_m^{n+1} = \frac{\Xi_m \Delta t}{\Omega_m \theta} - \frac{\lambda c_m^{n} \Delta t}{\theta} + c_m^{n} \tag{7.19}$$

The problem is now fully discretised, and all variables calculated at the given timestep. A flowchart of the processes undertaken within the source code in generating a solution to this problem is described in Figure 7.1.

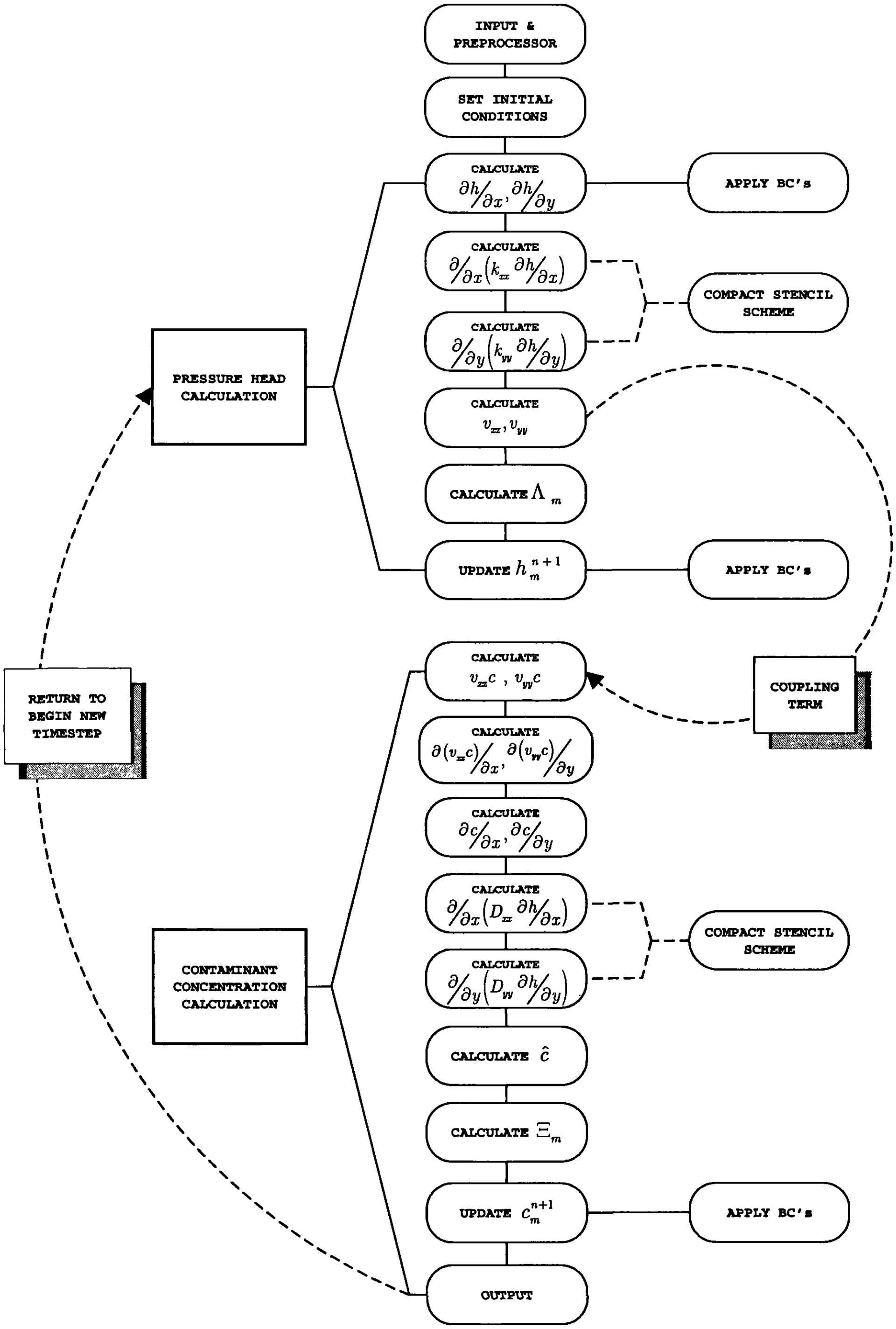


Figure 7.1: Program flow chart for the coupled flow problem.

7.4 RADIAL FLOW: INJECTION AND SUBSEQUENT EXTRACTION FROM THE SAME WELL

The test case considers the injection and subsequent extraction of a contaminant into an aquifer from the same well described by the governing equations (7.1) and (7.7). The problem is based around the test case presented by Holder *et. al* [4]. However, due to insufficient information in the publication, their results could not be replicated exactly.

In this test case, contaminated water is injected into an aquifer for a time $t = t_1$, then the flow is reversed, and water is removed from the aquifer. The aquifer is modelled as a two-dimensional square domain of side 9000 metres, and is illustrated in Figure 7.2 along with labels to distinguish between boundaries. The domain was discretised into 2304 structured quadrilateral elements as shown in Figure 7.3. The total run time for the simulation is set at t = 250 days. The well flow reversal time t_1 is defined as t_1 = 100 days, where t is the time since the beginning of the injection cycle, and the well is applied to node number 1201 at coordinate (4500, 4500) m. The well flow rate, q [M^3T^{-1}] is assumed to be the same in injection and recovery and is defined as follows;

$$q(4500, 4500, t) = \begin{cases} 1.0 \text{ m}^3\text{s}^{-1}, & t \le t_1 \\ -1.0 \text{ m}^3\text{s}^{-1}, & t > t_1 \end{cases} \tag{7.20}$$

The initial and boundary conditions for this problem are;

$$\begin{aligned} h(x,y,0) &= 0.0 \\ c(x,y,t) &= 0.0 \text{ for all nodes except the well,} \\ c(4500,4500,t) &= 1.0 \text{ for } t \le t_1, \\ \partial c/\partial x &= 0.0 \text{ on sides A-B and C-D,} \\ \partial c/\partial y &= 0.0 \text{ on sides A-D and B-C.} \end{aligned}$$

Other parameters necessary to run the problem are listed in Table 7.1 below.

Parameter	*Symbol*	*Value*
Hydraulic conductivity	K	$5.0\times10^{-6}\mathrm{ms}^{-1}$
Specific storage	S_s	$1.0\times10^{-4}\mathrm{m}^{-1}$
Porosity	ϕ	0.30
Ratio of longitudinal to transverse dispersivity	α_L/α_T	1.0
Domain size in x direction	--	9000 m
Domain size in y direction	--	9000 m
Edge length in x direction	Δx	187.5 m
Edge length in y direction	Δy	187.5 m
Number of nodes in x direction	N_x	49
Number of nodes in y direction	N_y	49
Number of nodes in mesh	--	2401
Number of elements in mesh	--	2304
Number of edges in mesh	--	4704
Concentration of contaminant in injected water	c_0	100%

Table 7.1: Parameters for coupled pollutant transport problem.

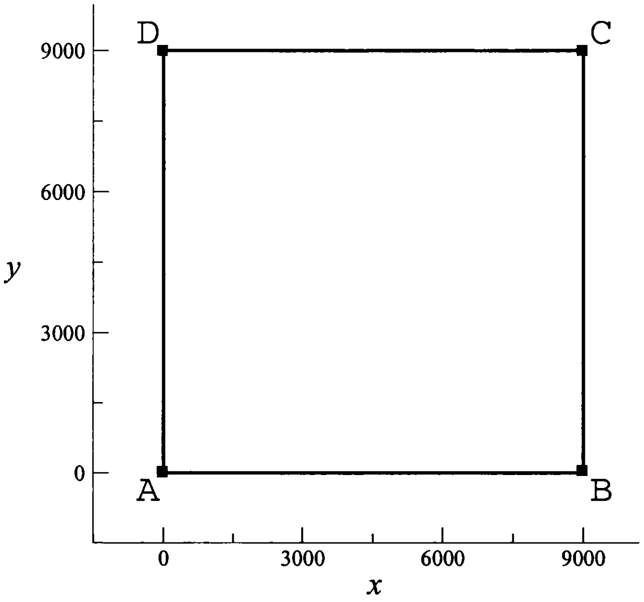


Figure 7.2: Square domain of side 9000 metres used for the test case, along with boundary markers.

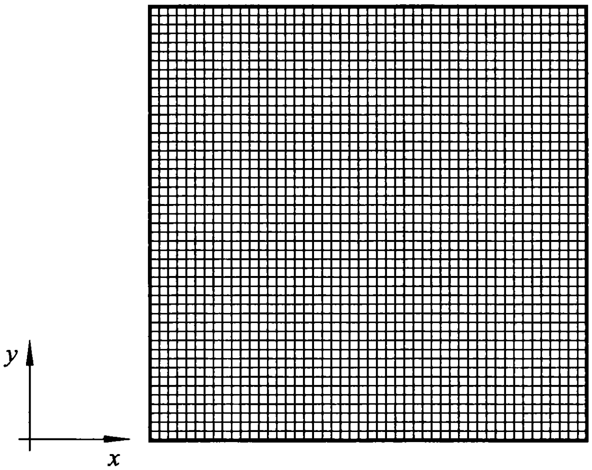


Figure 7.3: Domain discretised into 2304 structured quadrilateral elements.

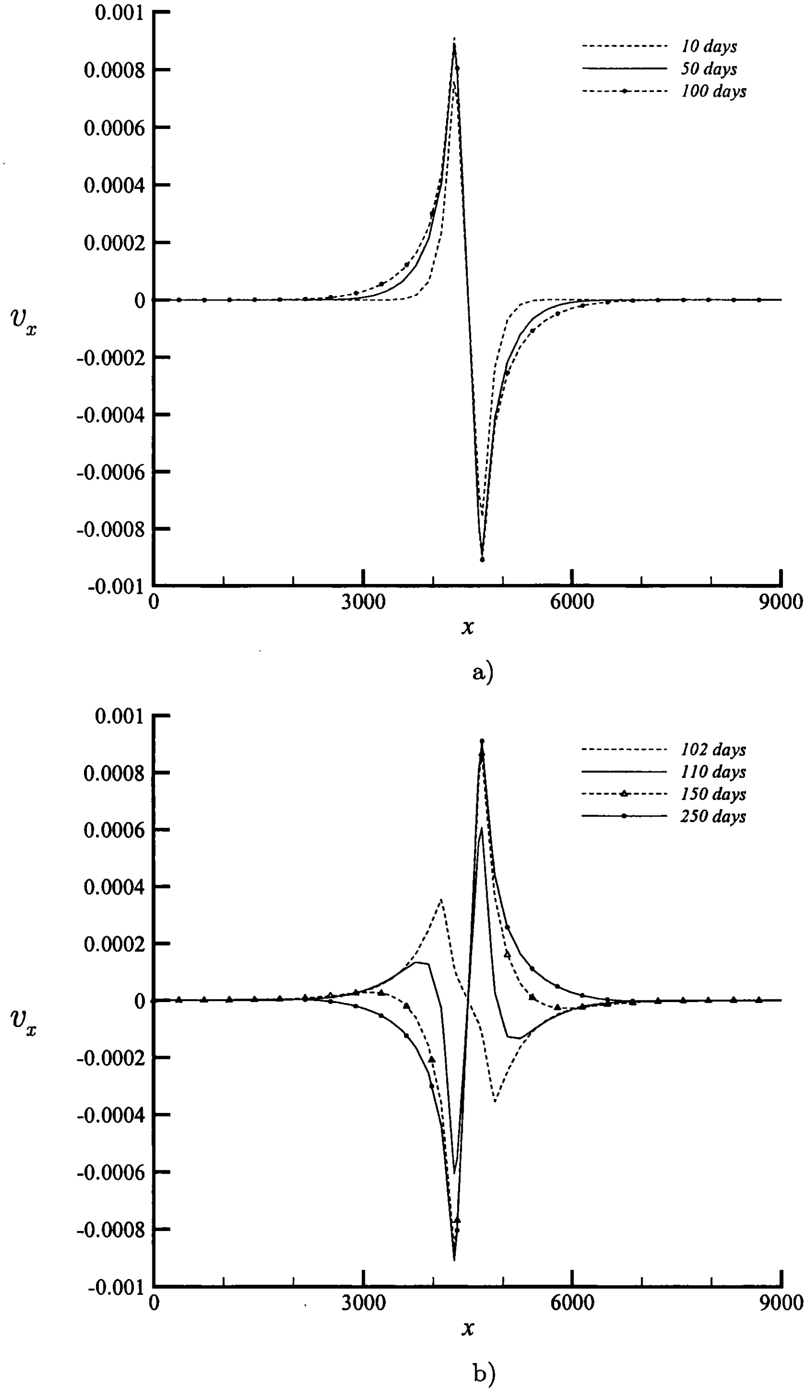
7.4.1 Varying Dispersion Coefficient Test

In order to investigate the effect of increasing diffusion rates on the contaminant transport equation (7.7), several cases were considered with Peclet numbers of 1.0, 2.0, 4.0, 5.0, 10.0, 20.0 and 40.0. The extremities of this scale represent highly diffusion dominated, and highly advection dominated systems. Yeh [10] and Sun and Yeh [8] describe the mesh Peclet number as;

$$Pe = \frac{v_x \Delta x}{D} \tag{7.21}$$

where Pe = Peclet number, Δx = edge length, D = dispersion coefficient, and v_x = Darcy velocity in the x direction.

Equations (7.4) and (7.7) are coupled by the Darcy velocity term (7.5). This coupling acts in one way only, that is from the flow equation (7.4) to the transport equation (7.7). By uncoupling the system and solving the flow equation only for the given problem, the following solution was obtained for the Darcy velocity in the x direction, v_x, along the horizontal centre line of the domain (see Figure 7.4). From this graph it can be seen that the calculated Darcy velocities lie approximately within the range [-0.0008, 0.0008] ms^{-1}. Of note is the behaviour of the plot near the well during the injection and extraction cycles. During the injection cycle ($t \leq 100$ days) the pressure head at the well is large compared to the surrounding nodes. Hence, theoretically, the gradient of the velocity plot (Figure 7.4) will be positive in the region $(0 \leq x < 4500)\,\mathrm{m}$, and negative in the region $(4500 > x \geq 9000)\,\mathrm{m}$. Relatively speaking, this is what is seen in Figure 7.4a, with the only discrepancy being the transition from a positive to a negative gradient is not instantaneous. However, this transition seems to have been handled well by the scheme, and is captured in a relatively short spatial distance. During the extraction cycle ($t > 100$ days) the pressure head at the well is now small compared to the surrounding nodes, and hence the gradient of the velocity graph in the regions either side of the well is reversed when compared to the injection phase as shown in Figure 7.4b.



a)

b)

Figure 7.4: Calculated velocity (v_x) profiles along the line $x \in [0, 4500]$m, $y = 4500$m through the domain at a) 10, 50 and 100 days, and b) 102, 110, 150 and 250 days.

By applying the maximum calculated Darcy velocity of 0.0008 ms^{-1}, and the chosen Peclet numbers, the theoretical maximum dispersion for each set can be calculated from equation (7.21). The chosen Peclet numbers along with the corresponding calculated dispersion rates and an assigned label for recognition of results are shown in Table 7.2, where $D_{xx} = D_{yy} = D_i$.

Pe	D_i	***Label***
1.0	0.15	D_1
2.0	0.075	D_2
4.0	0.0375	D_3
5.0	0.03	D_4
10.0	0.015	D_5
20.0	0.0075	D_6
40.0	0.00375	D_7

Table 7.2: Chosen Peclet numbers and calculated maximum dispersion values.

The calculated Peclet numbers should provide a wide range of solutions. Case D_1, with $Pe = 1.0$, will have a rate of diffusion 40 times greater than case D_7, which has $Pe = 40.0$. Results were obtained using an explicit scheme with a timestep of $dt = 10.0$ seconds, with the calculated variables h and c plotted for the following nodes over the 250 day simulation:

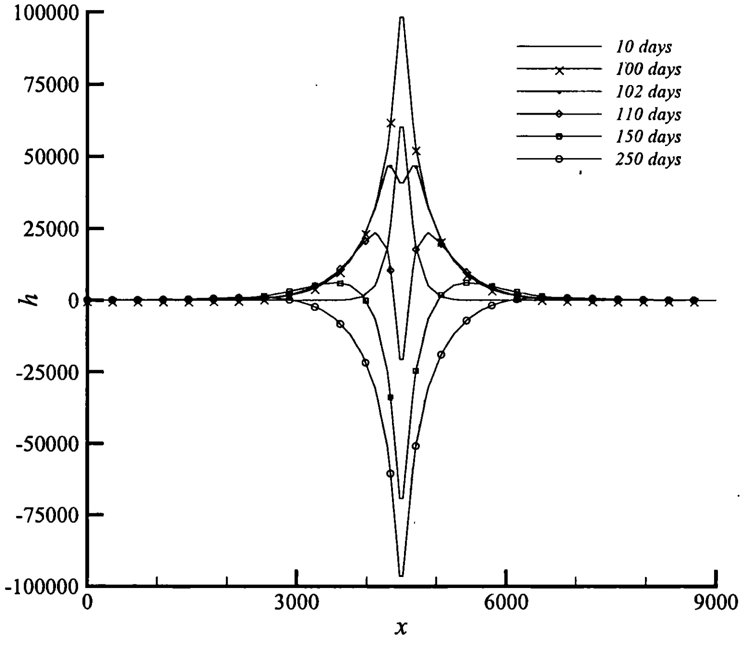
Node Number	***Spatial Coordinate (m)***	
	x	y
1201	4500.00	4500.00
1198	3937.50	4500.00
1195	3375.00	4500.00
1189	2250.00	4500.00
1185	1500.00	4500.00
1179	375.00	4500.00

Table 7.3: Coordinates of nodes at which result plots are given for the varying dispersion coefficient test.

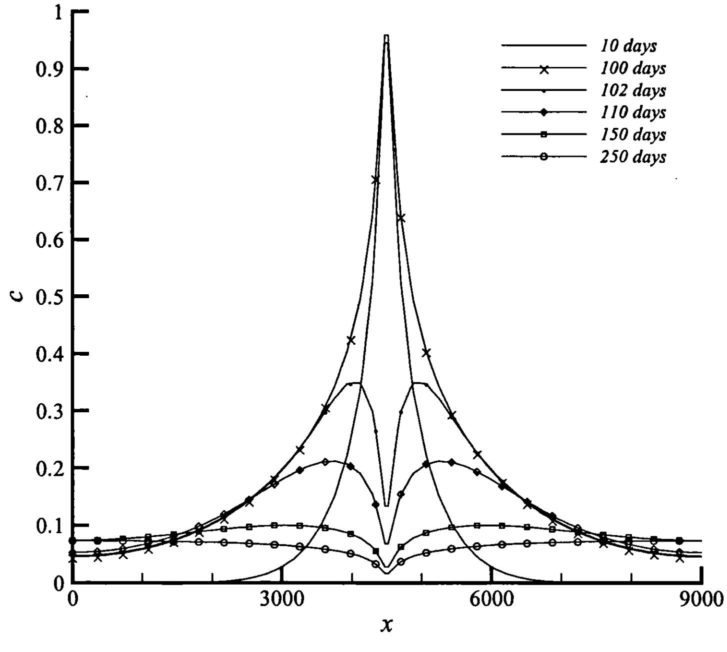
The results obtained by running the simulation as described above but using dispersion value D_1 are considered first. This dispersion value equals 0.15 and corresponds to a maximum Peclet number of 1.0, as shown in Table 7.2. Hence, this run has the maximum diffusion rate of all dispersion values that are to be considered. The results obtained using this dispersion rate are described fully here, and are then used as the benchmark with which results from using different dispersion rates are compared against. Figure 7.5 shows the pressure head and concentration profiles along a line through the centre of the domain, which intersects the well, at various times throughout the simulation. It is of note that differing dispersion rates have no effect on the pressure head plot shown in Figure 7.5a. This plot is the same no matter which dispersion value is used; hence it is only shown here. Both plots are smooth, and they seem to handle the reversal in the well condition from injection to subsequent extraction at time t_1 = 100 days very well. As the diffusion rates from this case are high, it is clearly seen from Figure 7.5b that the pollutant fills the whole domain after 250 days.

Figure 7.6 and Figure 7.7 shows the pressure head and concentration values at the nodes whose coordinates are tabulated in Table 7.3 against time, respectively. Again, differing dispersion rates (D) have no effect on the nodal pressure head plot shown in Figure 7.6; hence it is shown here once only.

Figure 7.6a and Figure 7.7a show the pressure head and concentration values at the well against time. The well reversal at time t_1 = 100 days is clearly captured on both plots. The reduction in pressure head for times $t > t_1$ is gradual, whereas the pollutant concentration reduces by approximately 90% between $t = t_1$ and $t = t_1 + dt$. This is as expected, as the pollutant injected at the source at $t \leq t_1$ is instantaneously removed at the same rate at $t > t_1$, hence causing large differences in the concentration levels at injection and extraction periods.

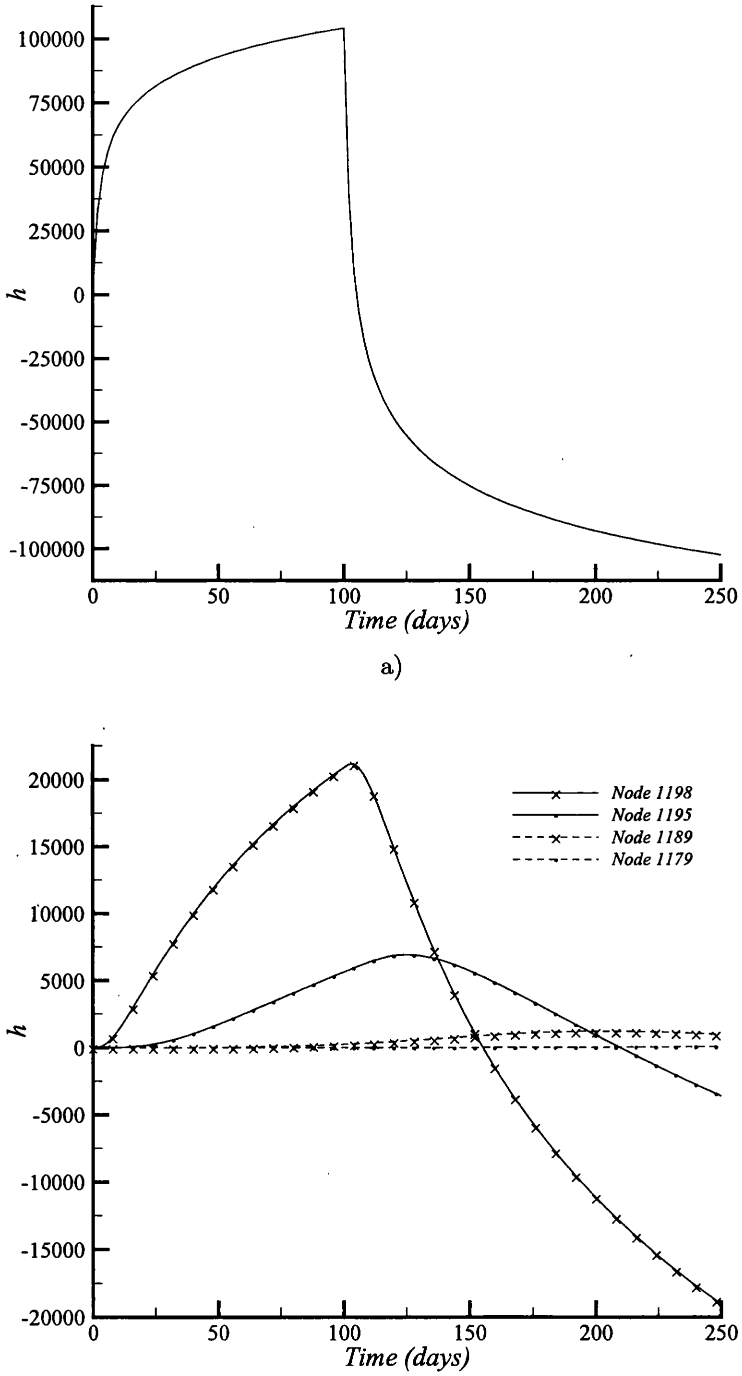


a)



b)

Figure 7.5: a) pressure head (h), and b) pollutant concentration (c) profiles along the line $x \in [0, 4500]$m, $y = 4500$m through the domain at 10, 100, 102, 110, 150 and 250 days using dispersion D_1.



a)

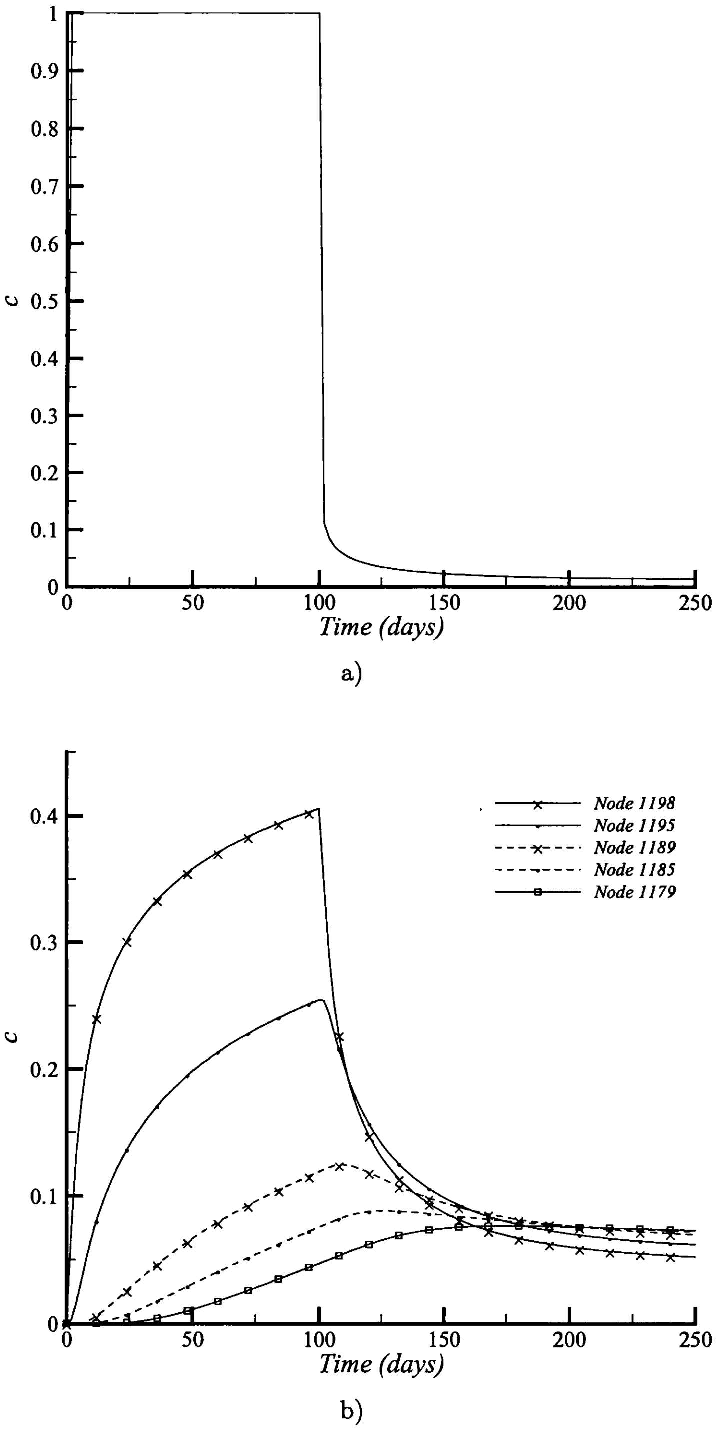
b)

Figure 7.6: Nodal pressure head (h) profiles over time at a) the well, node 1201, and b) nodes 1198, 1195, 1189 and 1179, using dispersion D_l.

Figure 7.6b and Figure 7.7b show the pressure head and concentration values at nodes 1198, 1195, 1189 and 1179 against time. These nodes occupy positions on the line y = 4500m, but at increasing distances away from the well node as shown in Table 7.3. The nearest, node 1198, is 562.5 m from the well, while the furthest, node 1179, is 4125 m from the well.

As can be seen from Figure 7.6b and Figure 7.7b, the transition point between a positive and a negative gradient on a nodal plot gets further away from time t_1 = 100 days as the position of the node becomes further away from the well. This is expected, due to the time taken for information to propagate through the domain. As the well condition is reversed at time t_1 = 100 days on node 1201, by the time this information has propagated to nodes lying further away from this position time has passed. The pressure head plot for node 1179 on Figure 7.6b shows no change over time. This corresponds to what is shown in Figure 7.5 as this node lies in the area showing no change in h. The pollutant concentration plot, Figure 7.7b, shows that the further the node is from the well, for times $t < t_1$, the greater the concentration of pollutant compared to closer lying nodes. This is directly associated with the propagation of information through the domain and the diffusion rate inherent in the equations.

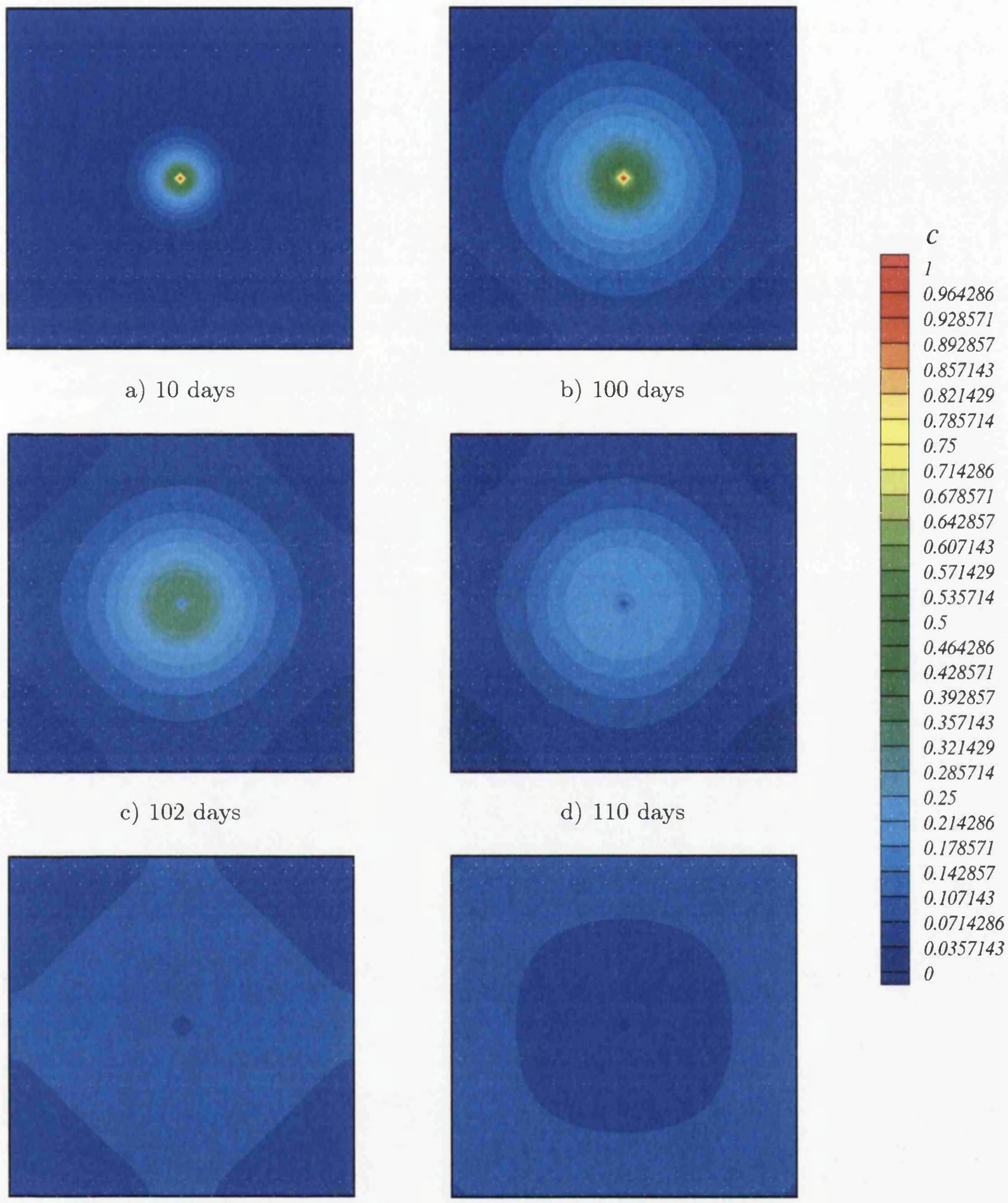
Figure 7.8 shows pollutant concentration (c) contours at 10, 100, 102, 110, 150 and 250 days using dispersion D_1. The contours clearly display radial flow surrounding an injecting/producing well, and the plots are symmetrical about the x and y axes. Reversal in the well at time t_1 = 100 days is evident from the contour plots, with the peak at the well at time t = 100 days becoming inverted at the same point at t = 102 days. This is shown more cleary in Figure 7.9, where three-dimensional pollutant concentration (c) contours at 102, 150, 200 and 250 days ($t > t_1$) are shown. The inverted peak due to the injection and subsequent extraction from the same well is evident in all the plots. Of interest is the continued diffusion of the pollutant concentration out towards the domain boundaries during this well reversal period. Note that the contour legend for Figure 7.9 (not shown) is the same as that for Figure 7.8.



a)

b)

Figure 7.7: Nodal pollutant concentration (c) profiles over time at a) the well, node 1201, and b) nodes 1198, 1195, 1189 and 1179, using dispersion D_l.



a) 10 days

b) 100 days

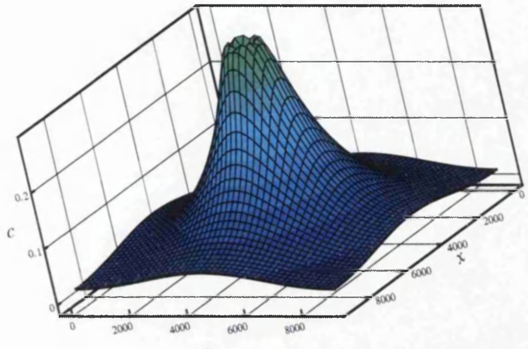
c) 102 days

d) 110 days

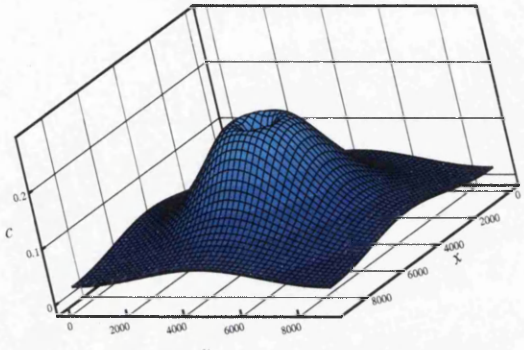
e) 150 days

f) 250 days

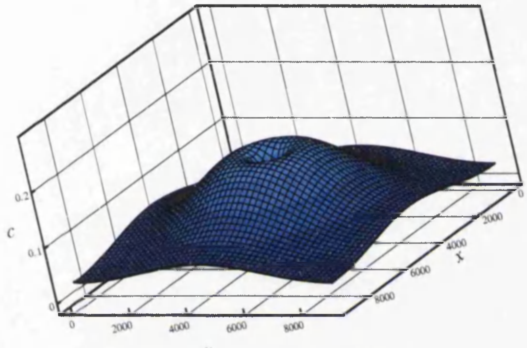
Figure 7.8: Pollutant concentration (c) contours at 10, 100, 102, 110, 150 and 250 days using dispersion D_I.



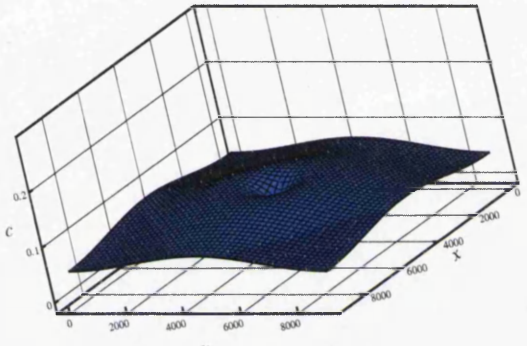
a) 102 days



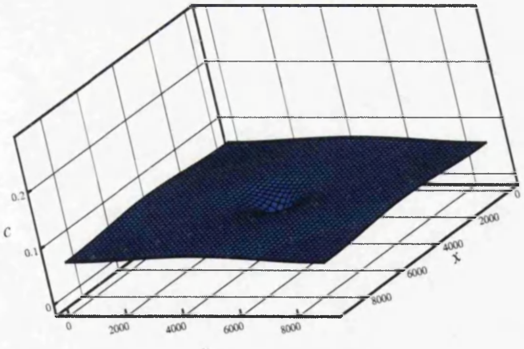
b) 110 days



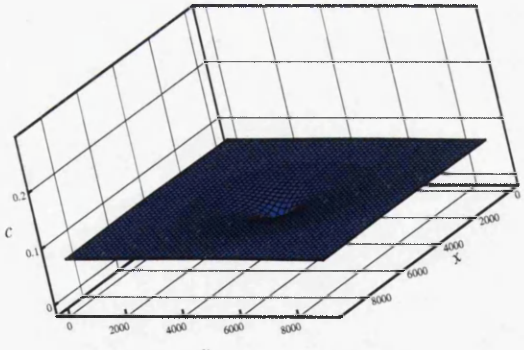
c) 120 days



d) 150 days



e) 200 days



f) 250 days

Figure 7.9: 3-D pollutant concentration (c) contours at 102, 150, 200 and 250 days using dispersion D_l.

The results obtained for the problem using the benchmark dispersion value D_1 have been described in the previous section. Here, the results obtained by using the other six dispersion values described in Table 7.2 are discussed and compared against the benchmark solution.

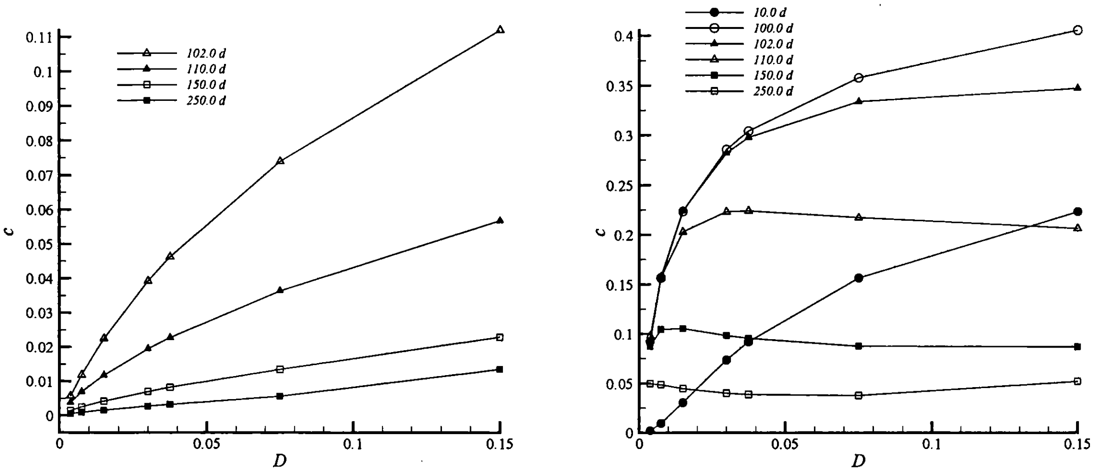
Once all simulations using the different dispersion coefficients were completed, it was necessary to compare the effect that this coefficient had on the solution. The calculated value of the pollutant concentration at particular nodes using the seven dispersion rates at specific times were compared. This is displayed in Figure 7.10, which shows plots of the chosen dispersion coefficient (D) against the calculated pollutant concentration (c) at t = 10, 100, 102, 110, 150 and 250 days for nodes 1201, 1198, 1195, 1189, 1185 and 1179. The six graphs in Figure 7.10 can be separated into two distinct types of behaviour. The first three plots will be dealt with first.

Figure 7.10a, b and c are of data related to nodes 1201, 1198, 1195 which have an x coordinate of 4500.00, 3937.50 and 3375.00 metres respectfully (see Table 7.3). The dispersion against concentration plots for each node are shown for 10, 100, 102, 110, 150 and 250 days. Note that the plot at t = 10 days was omitted from the node 1201 plot (Figure 7.10a), as the scale of the graph with the plot included prevents the easy recognition of the separate plots for each time. These three plots show that the dispersion coefficient produce very little difference with respect to the calculated concentration in this region of the domain. Hence, the primary change in the pollutant concentration is due to other factors. Therefore, the advective terms in equation (7.7) must drive the process. A significant contributor to these terms is the Darcy velocity (equation (7.5)), which is the coupling term between the flow and the transport equations.

Figure 7.10d, e and f These plots are of data related to nodes 1189, 1185 and 1179 which have an x coordinate of 2250.00, 1500.00 and 375.00 metres respectfully (see Table 7.3). The dispersion against concentration plots for each node are shown for 10, 100, 102, 110, 150 and 250 days. Here, the dispersion coefficient produces a significant difference with respect to the calculated concentration in this region of the domain. Hence, the primary change in the pollutant concentration is due to the diffusive terms in equation (7.7). By studying the Darcy velocity plot through the domain (Figure 7.4), it is seen that the calculated velocity in the region occupied by these nodes is zero, hence nullifying the effect of the advective terms in equation (7.5). In conclusion, in the outer regions of the domain, the change in the pollutant concentration is dominated by dispersion/diffusion terms, and the small change in the x coordinate between nodes 1195 and 1189 produces a large influence in results due to the effect of the coupling term.

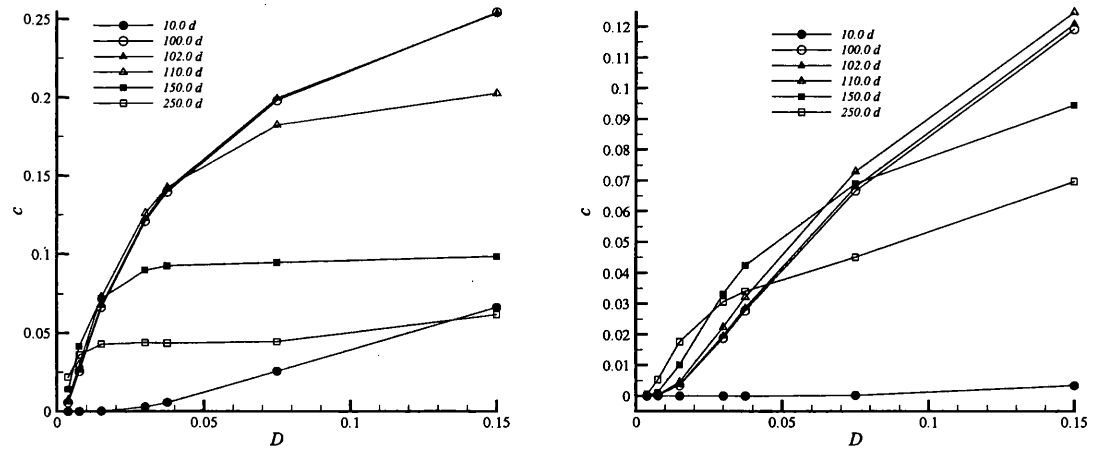
Figure 7.11 shows plots of the pollutant concentration profiles along the line $x \in [0, 4500]$m, $y = 4500$m through the domain at 10, 100, 102, 110, 150 and 250 days using the seven chosen dispersion values. It has been concluded from Figure 7.10 that in the vicinity of the well, the dispersion coefficient produces very little difference with respect to the calculated pollutant concentration, and the primary change in the concentration is due to the advective terms in equation (7.7). Additionally, beyond a radial distance of approximately 2000 metres from the well, the dispersion coefficient produces a significant difference with respect to the calculated concentration, and the primary change in the pollutant concentration is due to the diffusive terms in equation (7.7). These facts are supported by Figure 7.11, where for the same time, differences in the dispersion coefficient produces little difference to the result near to the well. This can be seen from all the plots in Figure 7.11, where for the first 1000 metres or so around the well the solutions obtained at a particular time using all dispersion coefficients are similar. Increasing the radial distance further reduces the magnitude of the Darcy velocity (see Figure 7.4) until it becomes insignificant at approximately 2000 metres. This transition from an advective dominant system, to a diffusion dominant system occurs in the region between 1000 and 2000 metres away from the well. Exceeding a radial distance of 2000 metres, the diffusion terms in the transport equation become dominant, resulting in large differences in results using different dispersion coefficients. As the dispersion coefficient decreases, hence the rate of diffusion encountered decreases, resulting in the pollutant having a restricted flow to the domain boundaries. This is evident from all plots, but especially clear in Figure 7.11d, e and f. Considering Figure 7.11f for illustrative purposes which shows the results obtained at 250 days, we have that for dispersion D_1, the pollutant fills the entire domain to a concentration of 0.0725 at the $x = 0.0$ metre boundary. However, a dispersion of D_7 (the smallest rate considered), the pollutant only occupies a region of approximately 2500 metres radially from the well at the same time.

Plots of the nodal pollutant concentration against time at nodes 1201, 1198, 1195, 1189, 1185 and 1179 using the seven chosen dispersion values are displayed in Figure 7.12. Again, it is evident from these plots the effect of the dispersion coefficient and the Darcy velocity coupling term has on the pollutant concentration at different locations within the domain. It is observed that for the plot for node 1198 (Figure 7.12b), the concentration using each dispersion coefficient tend towards the same solution at $t > t_1$. This is a direct consequence of the dominance of the advective terms in this region of the domain, as already discussed. However, the diffusive terms, which include the effect of the dispersion coefficient, causes the range of results seen for $t < t_1$. This trend is repeated, but the increase in the effect of the diffusion terms, and the decrease in the advective terms is apparent as the distance from the well increases. At node 1189 (Figure 7.12d), where the coupling velocity term is now small (see Figure 7.4), the different dispersion coefficients utilised has a much greater effect on the solution. The range in results, resulting from the different dispersion coefficients, is now apparent at $t > t_1$, with the result using the minimum dispersion value, D_7, showing no change over the simulation.



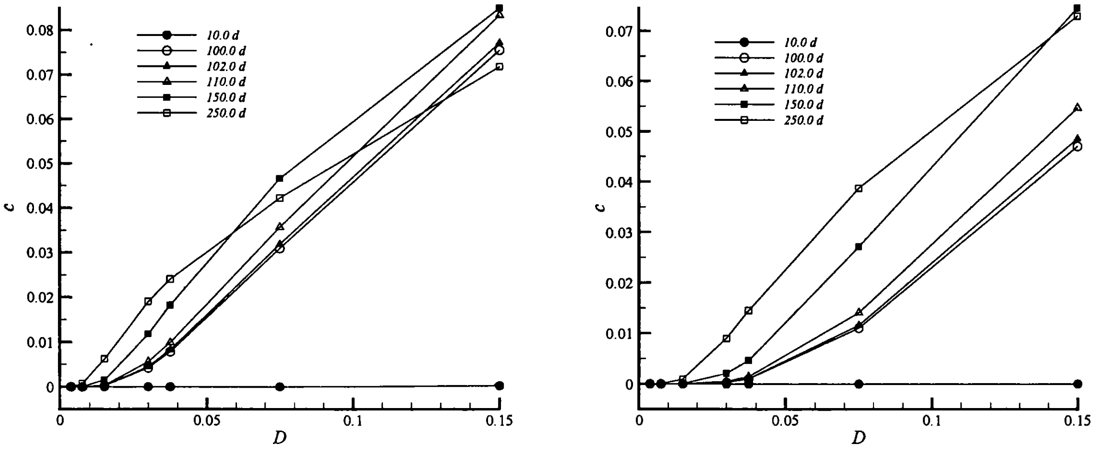
a) Node 1201

b) Node 1198



c) Node 1195

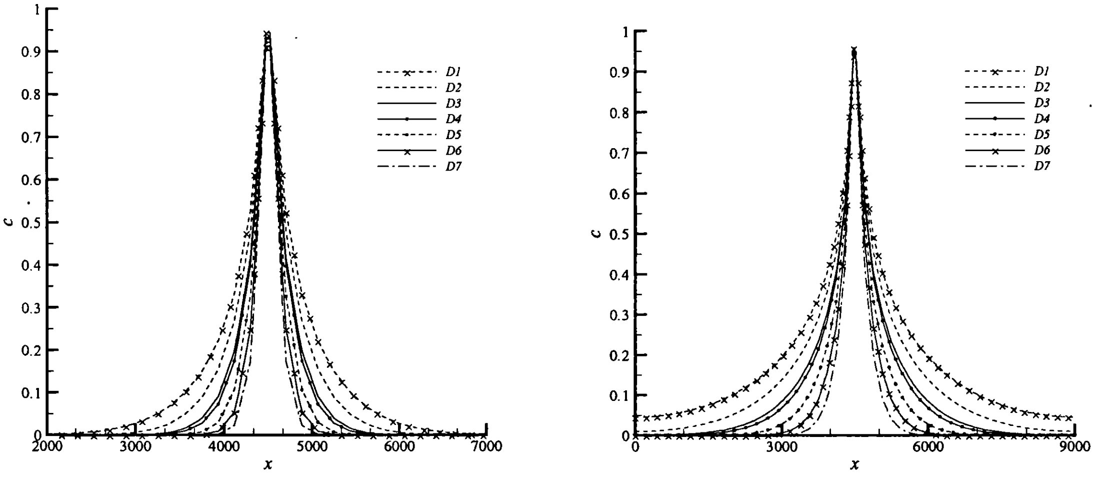
d) Node 1189



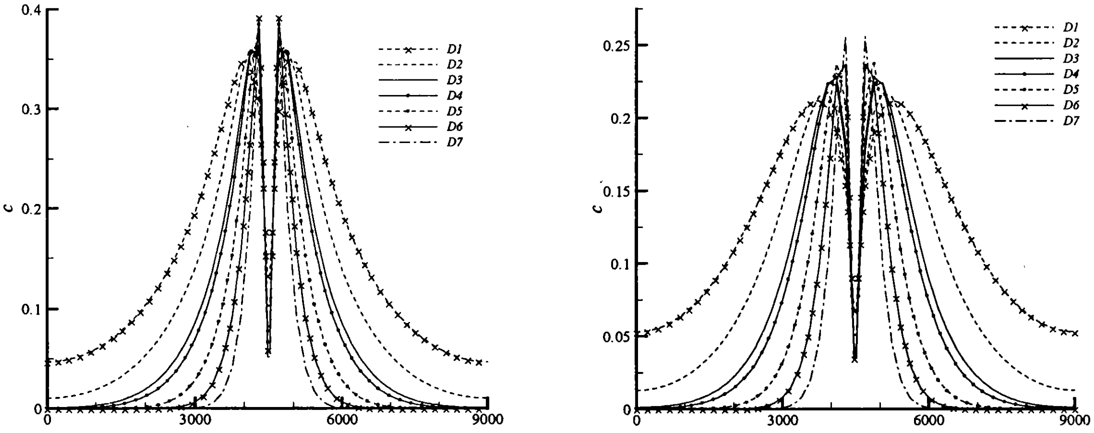
e) Node 1185

f) Node 1179

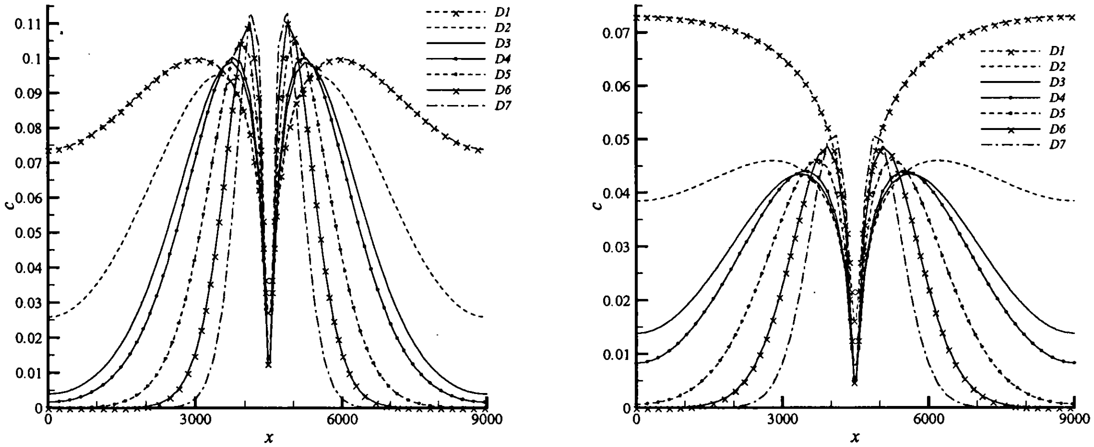
Figure 7.10: Plots of the chosen dispersion value (D) against pollutant concentration (c) at 10, 100, 102, 110, 150 and 250 days for nodes 1201, 1198, 1195, 1189, 1185 and 1179.



a) 10 days
b) 100 days

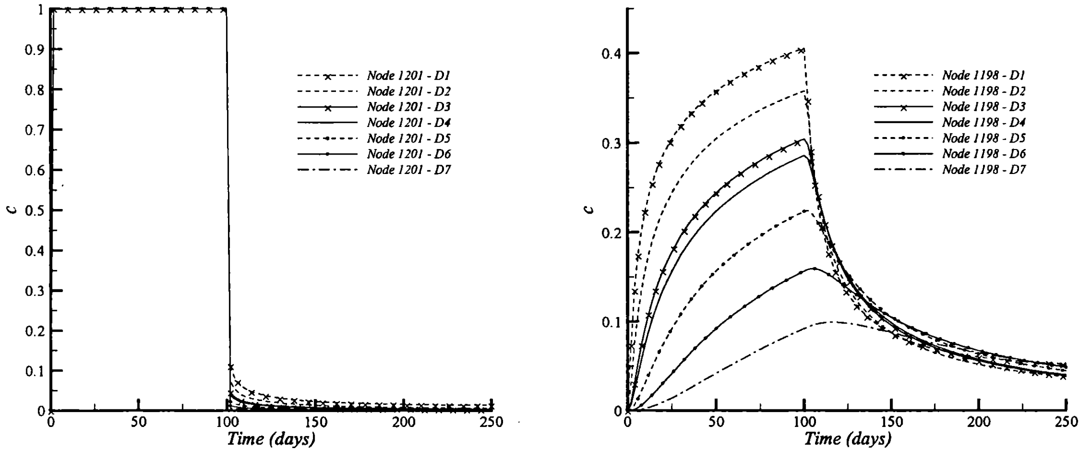


c) 102 days
d) 110 days

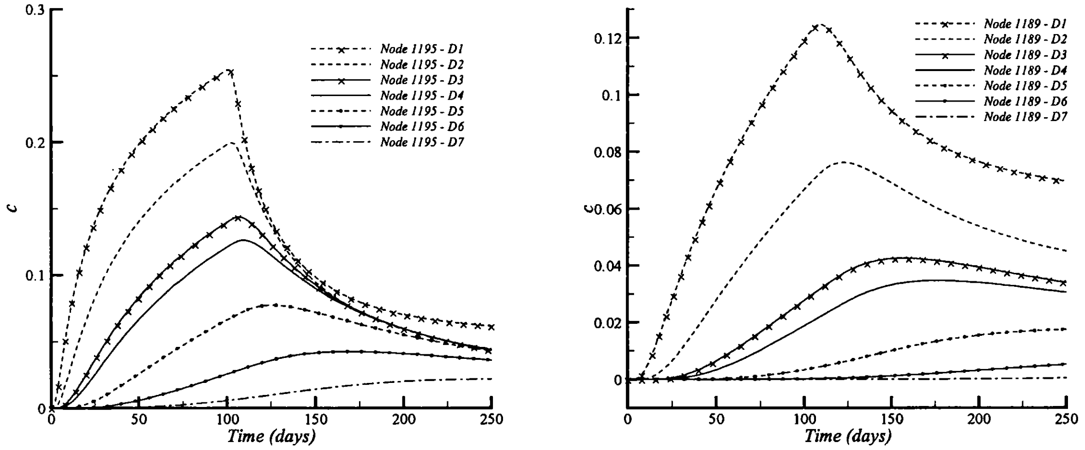


e) 150 days
f) 250 days

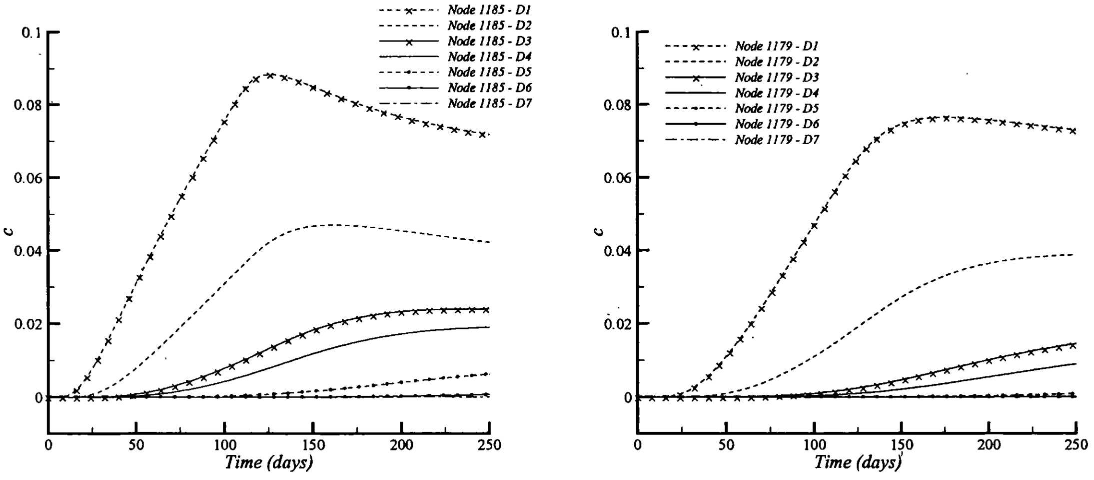
Figure 7.11: Pollutant concentration (c) profiles along the line $x \in [0, 4500]$m, $y = 4500$m through the domain at 10, 100, 102, 110, 150 and 250 days using the seven chosen dispersion values.



a) Node 1201 b) Node 1198



c) Node 1195 d) Node 1189



e) Node 1185 f) Node 1179

Figure 7.12: Nodal pollutant concentration (c) profiles over time at nodes 1201, 1198, 1195, 1189, 1185 and 1179 using the seven chosen dispersion values.

7.4.2 Results – Standard Finite Volume Scheme

All second derivative terms in the discretised equations (7.12) and (7.17) were calculated using the compact stencil scheme. As already discussed in earlier chapters, this scheme produces an improvement in the result, but also handles the calculation of derivatives on non-homogeneous domains much better than the standard finite volume scheme. In the authors' opinion, nothing demonstrates the superiority of the compact scheme over the standard scheme than using the scheme to solve a problem that has a point source such as the one considered here.

The results obtained by running the problem as described above using the standard finite volume scheme to calculate second derivatives are shown here. The dispersion value used equals D_1 and corresponds to a maximum Peclet number of 1.0, as shown in Table 7.2. The problem considered here is exactly the same as the problem in the previous section, where the compact scheme was utilised. As already shown in Figure 7.5b, the compact scheme produces smooth solutions for this problem. Figure 7.13 shows the comparison between the contour plots obtained using the standard and compact schemes at 10, 100 and 102 days. The plots are displayed on a three-dimensional axis with pollutant concentration on the vertical axis, and the contour legend for this diagram is the same as that used for Figure 7.8. It is evident from Figure 7.13 that the standard edge-based finite volume scheme produces large spatial oscillations emanating from the well at coordinate (4500, 4500) m. If it were not for the oscillations, it is undeniable that both schemes produce similar results. These spatial oscillations are a direct result of the five-node stencil which is the basis of the second derivative calculation using the standard scheme. A consequence of this is the decoupling of the odd and even nodes within the domain. Moreover, if a node behaves as a point source - as in this case - the nodes adjacent to it do not 'see' it when calculating second derivative terms, hence the spatial oscillations. This problem is eradicated when the three-node stencil of the compact scheme is used. In previous simulations, the improvement in the solution obtained when using the compact scheme over the standard scheme has been notable, but not to the extent shown in this case where the increased versatility of the compact stencil is apparent.

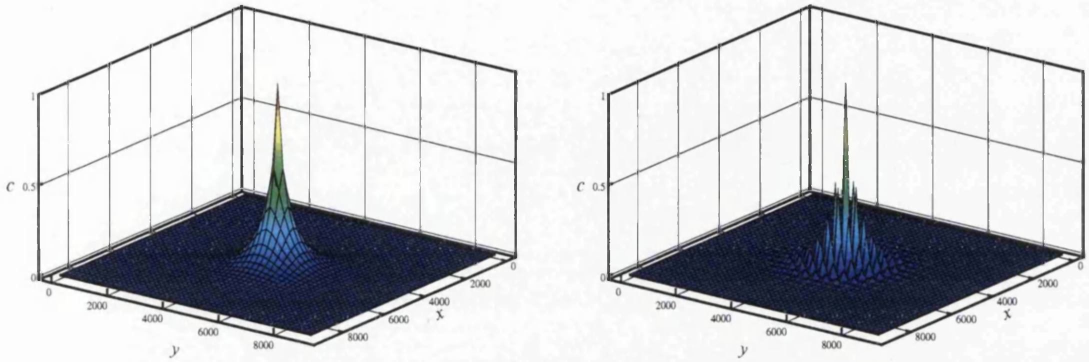
7.4.3 Pollutant Decay

When the exponential decay of the contaminant species is taken into account, the variable g in the contaminant species flow equation (7.7) is substituted by $(-\lambda c)$, as has already been seen in equation (6.2). The rate of decay of the contaminant species is determined by the decay coefficient λ, which represents the rate of decay of the contaminant species with time.

The problem discussed above is repeated here using dispersion coefficient D_5, and a range of decay levels. The decay coefficient values chosen for this investigation and the corresponding levels of decay at the end of the simulation associated with each value at the end of the simulation time are shown in Table 7.4.

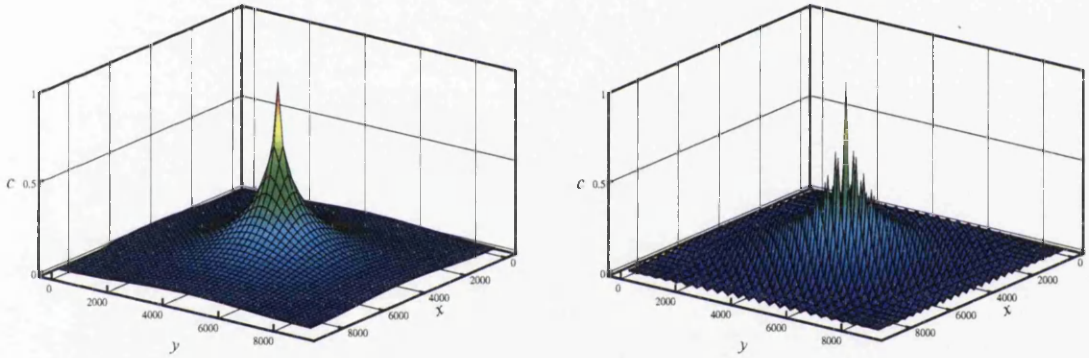
λ	***Decay %***
0.00000001	19.5
0.00000002	35.1
0.00000003	47.7
0.00000004	57.9
0.00000005	66.1
0.00000006	72.7

Table 7.4: Decay coefficient values and the corresponding levels of decay at $t = 250$ days.



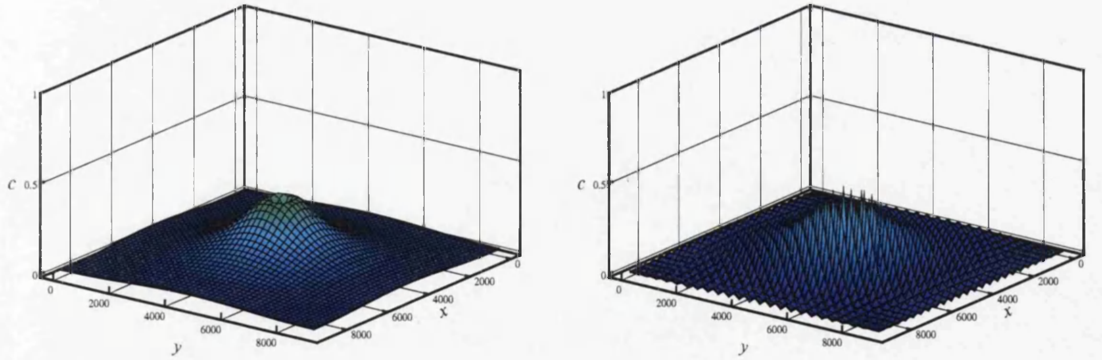
a) 10 days – Compact Stencil

b) 10 days – Standard F.V.



c) 100 days – Compact Stencil

d) 100 days – Standard F.V.



e) 102 days – Compact Stencil

f) 102 days – Standard F.V.

Figure 7.13: A comparison between the solutions obtained using the standard and compact stencil for the pollutant concentration (c) contours at 10, 100 and 102 days using dispersion D_I.

Figure 7.14 shows the pollutant concentration profiles against time at nodes 1201, 1198, 1195, 1189, 1185 and 1179 using the decay coefficients listed in Table 7.4. As can be seen, the pollutant decay function is acting as expected, with increasing levels of decay experienced as the simulation time increases until the percentage decay specified in Table 7.4 is achieved at $t = 250$ days. Similar observations can be made about Figure 7.15, which shows the pollutant concentration profiles along the line $x \in [0, 4500]$m, $y = 4500$m through the domain at 120, 150, 200 and 250 days and varying levels of decay coefficient λ. It is noted that as time progresses, the plot lines for the different decay coefficients used become more fanned out due to the increasing rate of decay.

Plots of the decay coefficient against pollutant concentration at 102, 110, 150 and 250 days for nodes 1201, 1198, 1195, 1189, 1185 and 1179, are shown in Figure 7.16. The plots for nodes 1201, 1198 and 1195 lie within the region of dominant advection as already discussed. In this region, the pollution decay is masked by the predominant advection terms in equation (7.7), and the varying levels of decay have little effect on the pollutant concentration. However, nodes 1189, 1185 and 1179 lie within the region of dominant diffusion, and here the effect of the pollutant decay term is apparent. The further the point is away from the well, and the later the time, the greater the effect of the decay term is on the pollution concentration.

7.5 CLOSURE

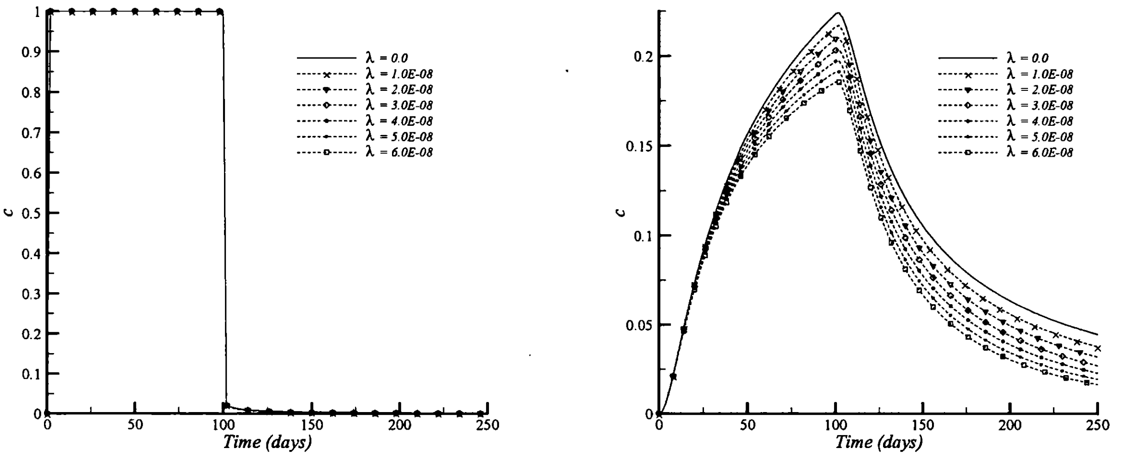
A coupled model for pollutant transport through a porous medium has been presented in this chapter, where the flow and transport equations are coupled via the Darcy velocity term. The governing equations were discretised using the edge-based finite volume scheme, and the problem solved on structured quadrilateral element meshes.

The initial test case considered the injection of a contaminant species into a homogeneous, isotropic aquifer for 100 days; then the flow was reversed denoting the extraction of the contaminant from the same well for the subsequent

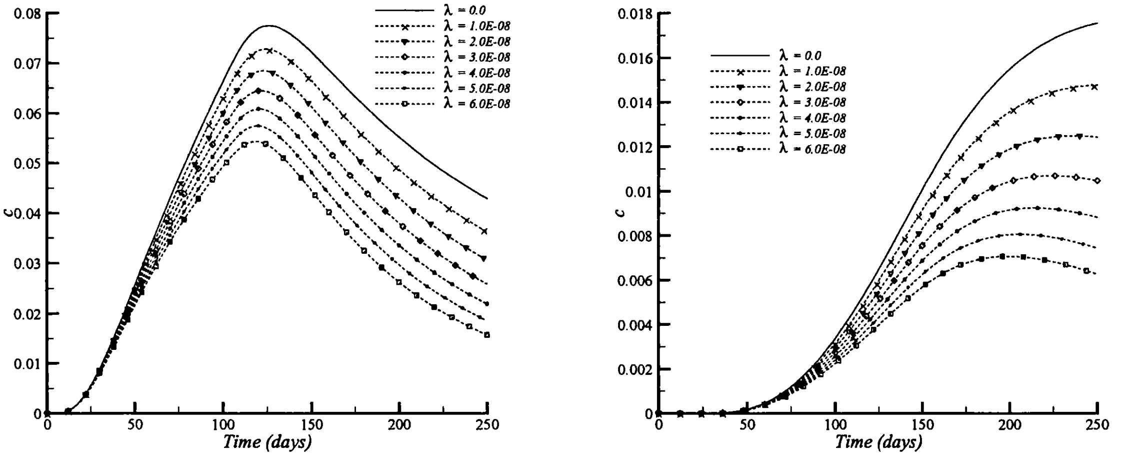
150 days. The injection and extraction rates were set at equal, but opposite rates. The calculated pressure head for this problem was a smooth curve, and the scheme successfully handled the reversal in the well condition from injection to extraction. An investigation into the effect of the dispersion coefficient on the contaminant concentration was undertaken, and seven different dispersion rates were considered. These ranged from highly diffusion dominated to highly advection dominated systems. It is evident from the results that due to the strong effect of the coupling Darcy velocity term, the domain was partitioned into zones of advection and diffusion dominant flow. This transition from an advective dominant system, to a diffusion dominant system occurred at approximately a radial distance of 1500 metres from the well. Exceeding this point, the diffusion terms in the transport equation became dominant, resulting in large differences in results from different dispersion coefficients. This is directly due to the high Darcy velocity encountered in the vicinity of the well dominating the diffusion terms in favour of the advective processes. Finally, a decay term was added to the governing equations, and the effect of pollutant decay was simulated for several decay rate coefficient values.

The superiority of the compact stencil formulation over the standard finite volume scheme was clearly demonstrated by comparing their solutions. As the test case contained a point source (i.e. a well), the standard scheme's five-node stencil produced large spatial oscillations emanating from the well. This is an inherent disadvantage of the scheme due to the decoupling of the odd and even nodes within the domain. The application of the three-node compact stencil was shown to overcome this difficulty; proving the versatility and power of the scheme.

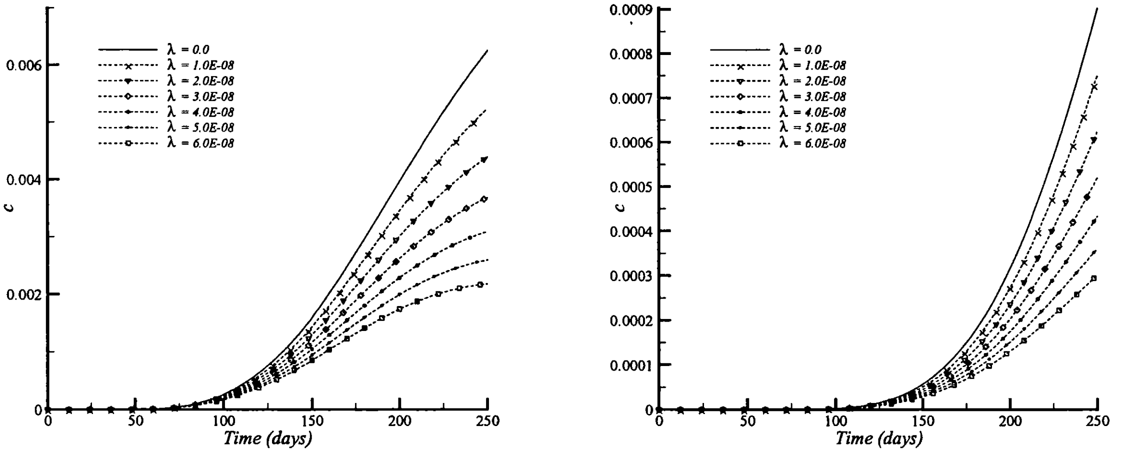
Although the edge-based finite volume scheme has been employed in solving coupled systems of equations in the past (see Lewis and Malan [5]), the author classifies the utilisation of the scheme in a coupled pollutant transport problems as being an original contribution.



a) Node 1201 b) Node 1198

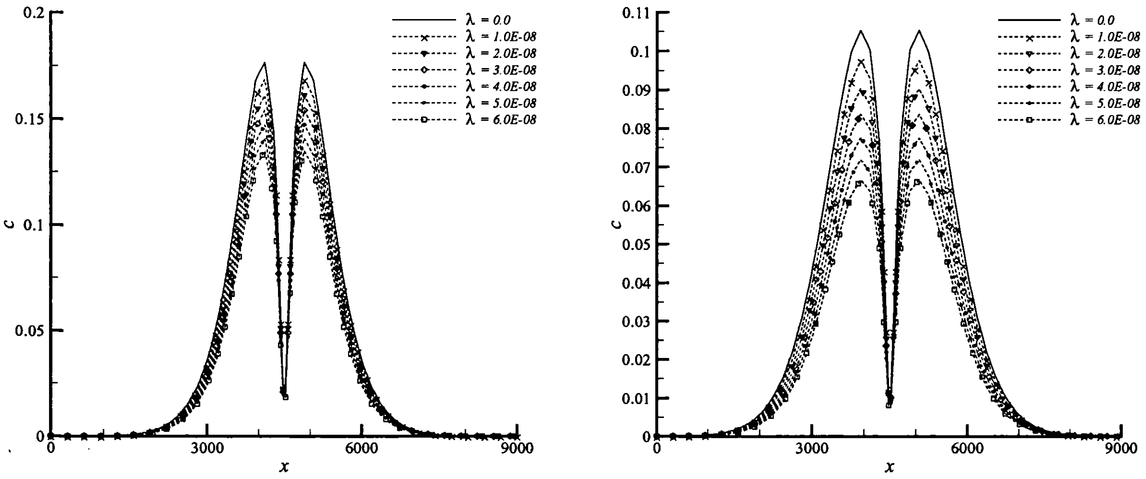


c) Node 1195 d) Node 1189



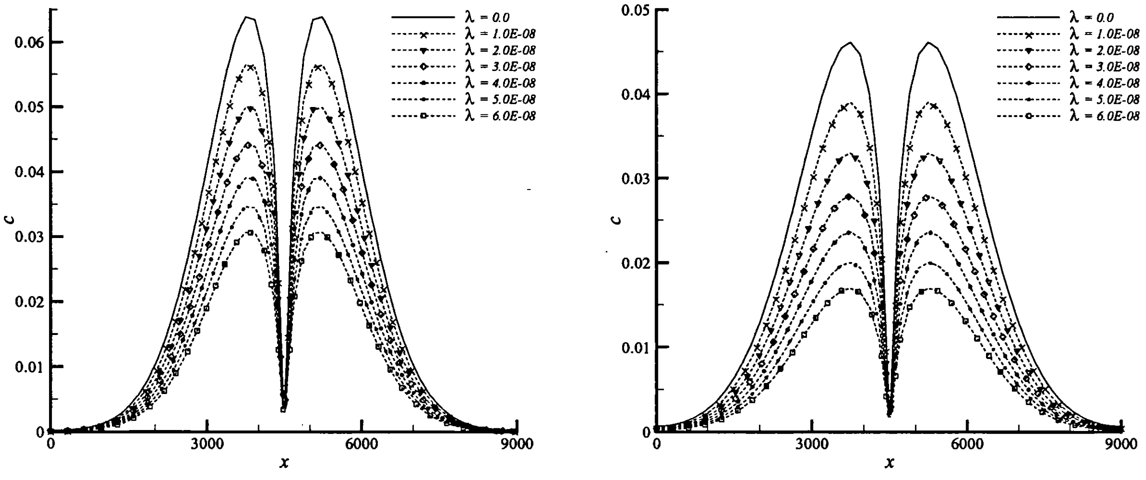
e) Node 1185 f) Node 1179

Figure 7.14: Nodal pollutant concentration (c) profiles over time at nodes 1201, 1198, 1195, 1189, 1185 and 1179 using dispersion D_5, and varying levels of decay λ.



a) 120 days

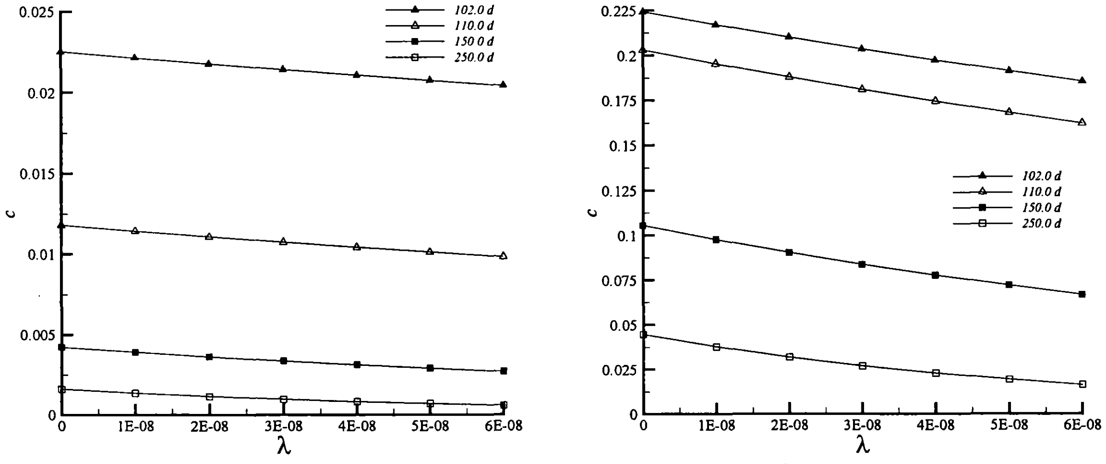
b) 150 days



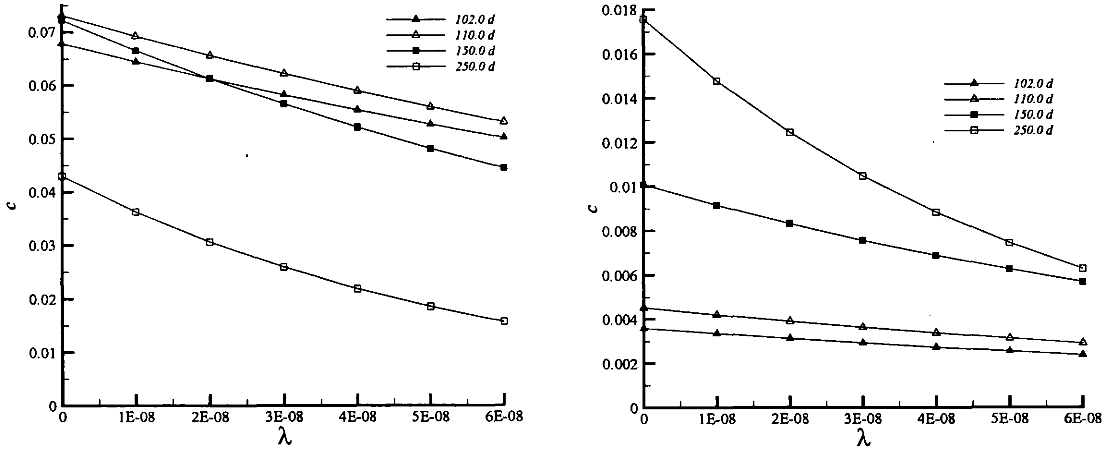
c) 200 days

d) 250 days

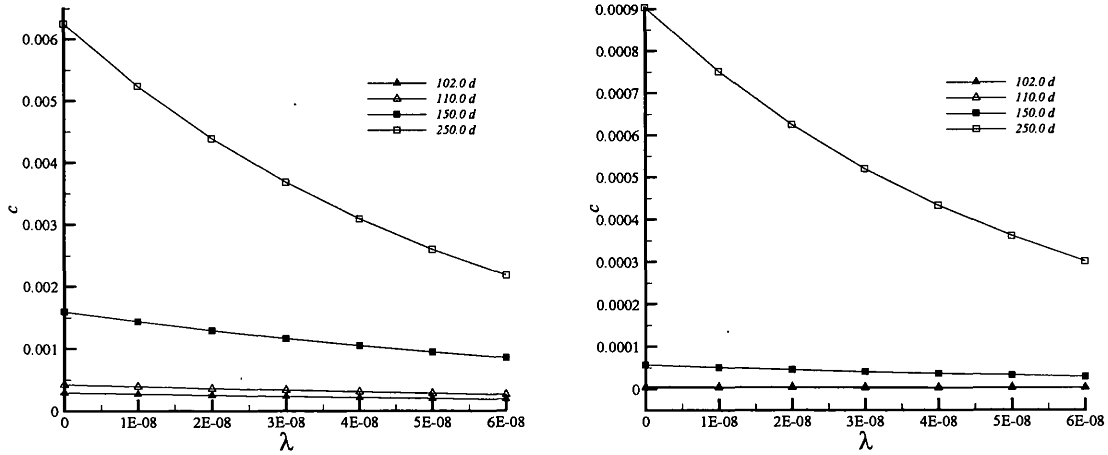
Figure 7.15: Pollutant concentration (c) profiles along the line $x \in [0, 4500]$m, $y = 4500$m through the domain at 120, 150, 200 and 250 days using dispersion D_5, and varying levels of decay λ.



a) Node 1201 b) Node 1198



c) Node 1195 d) Node 1189



e) Node 1185 f) Node 1179

Figure 7.16: Plots of the chosen level of decay (λ) against pollutant concentration (c) at 102, 110, 150 and 250 days for nodes 1201, 1198, 1195, 1189, 1185 and 1179, using dispersion D_5.

REFERENCES

[1] Dick E., Accurate Petrov-Galerkin methods for transient convective diffusion problems, *International Journal for Numerical Methods in Engineering*, **19**: 1425-1433, 1983.

[2] Gottardi G., POLF: Two-dimensional finite element model for predictiong areal flow of pollutant in confined and unconfined aquifers, *Computers & Geoscienes*, **24**(6): 509-522, 1998.

[3] Gray W.G. and Pinder G.F., An analysis of the numerical solution of the transport equation, *Water Resources Research*, **12**(3): 547-555, 1976.

[4] Holder A.W., Bedient P.B. and Dawson C.N., FLOTRAN, a three dimensional ground water model, with comparisons to analytical solutions and other models, *Advances in Water Resources*, **23**: 517-530, 2000.

[5] Lewis R.W. and Malan A.G., Continuum thermodynamic modelling of drying capillary particulate materials via an edge-based algorithm, *Computer Methods in Applied Mechanics and Engineering*, to appear, 2004.

[6] McDonald M.G. and Harbaugh A.W., A modular three-dimensional finite-difference ground-water flow model. Open File Report 83-875, US Department of the Interior, USGS National Center, Reston, Virginia, 1984.

[7] Narasimham T.N. and Witherspoon P.A., An integrated finite difference method for analyzing fluid flow in porous media, *Water Resources Research*, **17**(6): 1665-1675, 1976.

[8] Sun N.-Z. and Yeh W.W.-G., A proposed upstream weight numerical method for simulating pollutant transport in groundwater, *Water Resources Research*, **19**(6): 1489-1500, 1983.

[9] Yeh G.T. and Ward D.S., FEMWATER, a finite element model of water flow through saturated-unsaturated porous media. ORNL-5567, Oak Ridge National Laboratory, Oak Ridge, Tennessee, 1980.

[10] Yeh G.T., An orthogonal-upstream finite element approach to modeling aquifer contaminant transport, *Water Resources Research*, **22**(6): 952-964, 1986.

[11] Yu C.C. and Heinrich J.C., Petrov-Galerkin method for multidimensional, time dependent, convective-diffusion equations, *International Journal for Numerical Methods in Engineering*, **24**: 2201-2215, 1987.

Chapter 8

CONCLUSION

8.1 ACHIEVEMENTS

The work presented in this thesis constitutes the first phase in the development of a simulator for modelling fluid flow through porous media. The main research activities were:

- Suitable discretization and solution procedures were investigated, and the edge-based finite volume technique chosen.
- The edge-based technique was enhanced by an alternative calculation procedure, utilizing a more compact stencil for the case of the non-linear diffusion type partial differentials.
- With regards to the software implementation, a novel label based approach was employed.

The chosen discretization procedures were successfully applied to model:

- The highly non-linear process of phase transformation of a one-component material using the enthalpy method.
- Saturated-unsaturated fluid flow in porous media described by the highly non-linear Richards equation.
- Miscible contaminant transport through aquifers over a horizontal plane.
- A coupled miscible contaminant transport and fluid flow through a saturated porous medium.

8.2 CONCLUSIONS

The vertex-centred edge-based finite volume scheme is a recently developed technique that has been proven to be accurate, while also being superior to traditional element based methods in terms of computational efficiency. The chosen scheme was enhanced by the application of a compact stencil algorithm for the discretization of diffusive second derivative terms, and the increased accuracy of this chosen scheme was demonstrated many times against the standard finite volume algorithm. Moreover, it was shown that the compact stencil scheme was the only available option when calculating second derivative terms on non-homogeneous domains.

For implementation into a computer code, a novel label based approach was employed in the interest of computational performance. A data structure was presented such that the modification of code at all levels was possible with the minimum of labour. As a result of this chosen data structure, the computationally expensive routines were a small part of the code and were easily optimised. The strategy of making no distinction between variables and treating each in the same way lead to a conceptually elegant matrix structure and a straightforward implementation. Consequently, a highly flexible software package is developed which had 91% code re-use, that can be modified to solve engineering problems in a variety of fields with little alteration to the source code, and no alteration to the data structure.

The edge-based finite volume scheme accurately solved the highly non-linear phase change problem, for both solidification and melting. This problem was considered due to it being very similar mathematically to the equations that govern the flow of fluids through porous media, and analytic solutions were available for comparison. The solution was shown to be superior to published numerical solutions obtained using the finite element method.

The next validation considered was a groundwater flow problem described by the Richards equation. The test cases showed that the standard finite volume method and compact stencil formulations presented were successfully implemented for solving problems concerning two-dimensional flow in unsaturated soils using both structured and unstructured meshes. The test cases considered consisted of the infiltration of water into an initially dry soil column, and the infiltration of groundwater across an earth dam of homogeneous and non-homogeneous cross sections. Results showed that the compact stencil scheme produced less smearing of the saturation front as compared to the standard scheme. Also, a successive over relaxation non-linear iteration loop was implemented with the effect that the analysis time for the solution of the given problem was accelerated by an order of magnitude.

A large number of test cases for contaminant transport were analysed and where analytical solutions were available, good correlations were achieved. The superiority of the compact stencil scheme over the standard finite volume formulation was clearly evident from these test cases, with the standard scheme introducing a spatial oscillation. Decay of the pollutant species over time was investigated, and the injection of a pollutant slug into the system for a set time was successfully modelled.

Finally, a coupled model for pollutant transport through a porous medium was presented. The flow and transport equations were coupled via the Darcy velocity term, and discretised using the edge-based finite volume scheme. The problem solved on a structured quadrilateral element mesh. The test case considered the injection and subsequent extraction of a contaminant species into a homogeneous, isotropic aquifer. The scheme successfully handled the reversal in the well condition from injection to extraction, and the numerical solution for both the pressure head and pollutant concentration was everywhere smooth. It was shown that the Darcy velocity term had an overwhelming effect on the contaminant transport system in the vicinity of the well. In this region the Darcy velocities were large, resulting in an advection dominant system.

To the authors' knowledge, this was the first instance for the vertex-centred edge-based finite volume scheme has been utilised in solving highly non-linear groundwater flow and contaminant transport problems.

8.3 RECOMMENDATIONS

The foundation has been laid for a robust model capable of handling the governing equations of fluid flow through porous media. Further work is required to furnish a comprehensive simulator for petroleum reservoir applications, which is the long-term aim of the project. The issues involved are,

- The extension of the model to consider two-phase, and multi-phase immiscible flow problems, such as is encountered in petroleum reservoir systems (oil, water, gas). All the required elements for this phase in the development are in place.
- Extension of the code to include a third spatial dimension. This is deemed of importance, as geological formations are often highly three-dimensional. This would mainly involve the extension of the pre-processor to handle three-dimensional domains. Extending the solver into three-dimensions would not pose a great difficulty due to the edge-based nature of the scheme.
- The coupling of a stress analysis module needs to be realised if the solver is to be applied to solve complex problems of fluid flow in deformable porous media. One course of action would be to couple the already developed finite volume fluid flow model to the existing finite element stress model in CORES (see chapter 1). With the computational advantages of the edge-based finite volume scheme over finite element schemes, a more attractive alternative would be to write an edge-based finite volume stress model and couple this to the existing edge-based fluid flow model. Finite volume schemes have recently emerged as a viable numerical method for stress analysis in solid structures, and fluid-structure interaction has been successfully modelled using the finite volume scheme by Greenshields *et al.* [1], and Maneeratana and Ivankovic [2]. Hence, the re-writing of the existing stress model in an edge wise manner could prove to be the way forward. This development will be based on the innovative programming techniques described in chapter 3. It is estimated that code

development will require 30% less effort due to this approach, significantly accelerating the development process.

REFERENCES

[1] Greenshields C.J., Weller H. G. and Ivankovic A., The Finite Volume Method for Coupled Fluid Flow and Stress Analysis, *Computer Modelling and Simulation in Engineering*, 4: 213-218, 1999.

[2] Maneeratana K. and Ivankovic A., Finite Volume Method for Geometrically Non-linear Stress Analysis Problems, (Proc. 7th ACME conference on Computational Mechanics in Engineering, Durham, 1999).

Appendix A

```
      subroutine analytical(xcoord, tempra)
      include 'explicit.fi'

      K = 2.0
      rho = 2050.0
      Cp  = 9.4536586E+02
      alpha = K/(rho*Cp)
      L = 0.1
      A1 = 1.273
      lambda1 = sqrt(2.467)
      Fo = alpha*(dt*i_time_steps)/(L*L)
      T0 = 6.2500000E+02
      Te = 125.0

      l1a = 1.5707963
      l2a = 4.7123890
      l3a = 7.8539816
      l4a = 10.995574
      l5a = 14.137167
      l6a = 17.278760
      l7a = 20.420352
      l8a = 23.561945
      l9a = 26.703538
      l10a = 29.84513021
      l11a = 32.98672286
      l12a = 36.12831552
      l13a = 39.26990817

      A1a = 2.0*sin(l1a)/(l1a + sin(l1a)*cos(l1a))
      A2a = 2.0*sin(l2a)/(l2a + sin(l2a)*cos(l2a))
      A3a = 2.0*sin(l3a)/(l3a + sin(l3a)*cos(l3a))
      A4a = 2.0*sin(l4a)/(l4a + sin(l4a)*cos(l4a))
      A5a = 2.0*sin(l5a)/(l5a + sin(l5a)*cos(l5a))
      A6a = 2.0*sin(l6a)/(l6a + sin(l6a)*cos(l6a))
      A7a = 2.0*sin(l7a)/(l7a + sin(l7a)*cos(l7a))
      A8a = 2.0*sin(l8a)/(l8a + sin(l8a)*cos(l8a))
      A9a = 2.0*sin(l9a)/(l9a + sin(l9a)*cos(l9a))
      A10a = 2.0*sin(l10a)/(l10a + sin(l10a)*cos(l10a))
      A11a = 2.0*sin(l11a)/(l11a + sin(l11a)*cos(l11a))
      A12a = 2.0*sin(l12a)/(l12a + sin(l12a)*cos(l12a))
      A13a = 2.0*sin(l13a)/(l13a + sin(l13a)*cos(l13a))

      f1a = cos(l1a*(xcoord-L)/L)
      f2a = cos(l2a*(xcoord-L)/L)
      f3a = cos(l3a*(xcoord-L)/L)
      f4a = cos(l4a*(xcoord-L)/L)
      f5a = cos(l5a*(xcoord-L)/L)
      f6a = cos(l6a*(xcoord-L)/L)
      f7a = cos(l7a*(xcoord-L)/L)
      f8a = cos(l8a*(xcoord-L)/L)
      f9a = cos(l9a*(xcoord-L)/L)
      f10a = cos(l10a*(xcoord-L)/L)
      f11a = cos(l11a*(xcoord-L)/L)
      f12a = cos(l12a*(xcoord-L)/L)
      f13a = cos(l13a*(xcoord-L)/L)

      tempra = Te + (T0-Te)*( A1a*exp(-l1a*l1a*Fo)*f1a +
     .        A2a*exp(-l2a*l2a*Fo)*f2a + A3a*exp(-l3a*l3a*Fo)*f3a +
     .        A4a*exp(-l4a*l4a*Fo)*f4a + A5a*exp(-l5a*l5a*Fo)*f5a +
     .        A6a*exp(-l6a*l6a*Fo)*f6a + A7a*exp(-l7a*l7a*Fo)*f7a +
     .        A8a*exp(-l8a*l8a*Fo)*f8a + A9a*exp(-l9a*l9a*Fo)*f9a +
     .        A10a*exp(-l10a*l10a*Fo)*f10a +
     .        A11a*exp(-l11a*l11a*Fo)*f11a  +
     .        A12a*exp(-l12a*l12a*Fo)*f12a +
     .        A13a*exp(-l13a*l13a*Fo)*f13a )

      return
      end
```

FORTRAN subroutine to calculate the analytical solution for the diffusion equation (2.31).

Appendix B

DEFINITION OF THE ENTHALPY FUNCTION

The enthalpy function, H, is defined by Lewis *et al.* [2] for isothermal phase change as,

$$
\begin{aligned}
H(T) &= \int_{T_r}^{T} \rho c_s(T)\ dT && \left(T < T_f\right) \\
H(T) &= \int_{T_r}^{T_f} \rho c_s(T)\ dT + \rho L + \int_{T_f}^{T} \rho c_l(T)\ dT && \left(T \geq T_f\right)
\end{aligned}
\qquad (B.1)
$$

and for phase change over an interval of temperatures T_s to T_l, which are the solidus and the liquidus respectively, we have,

$$
\begin{aligned}
H(T) &= \int_{T_r}^{T} \rho c_s(T)\ dT &, \quad & \left(T < T_f\right) \\
H(T) &= \int_{T_r}^{T_s} \rho c_s(T)\ dT + \int_{T_s}^{T} \left(\rho\left(\frac{dL}{dT}\right) + \rho c_f(T)\right) dT &, \quad & \left(T_s \leq T \leq T_l\right) \\
H(T) &= \int_{T_r}^{T_s} \rho c_s(T)\ dT + \rho L + \int_{T_s}^{T_l} \rho c_f(T) dT & & \\
&\quad + \int_{T_l}^{T} \rho c_l(T) dT &, \quad & \left(T > T_l\right)
\end{aligned}
\qquad (B.2)
$$

where, c_f is the specific heat in the freezing interval, L is the latent heat and T_r is a reference temperature lower than T_s. The mathematical validity of this formulation is discussed in detail by Crank [1], who has proved the equivalence of the classical and weak formulations and the existence and uniqueness of the solution for the weak formulation.

REFERENCES

[1] Crank J., Free and Moving Boundary Problems, Clarendon Press, Oxford, 1984.

[2] Lewis R.W., Morgan K., Thomas H.R. and Seetharamu K.N., The Finite Element Method in Heat Transfer Analysis, John Wiley & Sons, 1996.

Appendix C

ENTHALPY METHOD AVERAGING TECHNIQUES

Some of the commonly used averaging techniques mentioned in the relevant references ([1],[5]) are listed below. The simplest approximation used (for 2-D) is;

$$\frac{dH}{dT} = \frac{1}{2}\left(\frac{\frac{dH}{dx}}{\frac{dT}{dx}} + \frac{\frac{dH}{dy}}{\frac{dT}{dy}} \right) \tag{C.1}$$

This method has been reported to cause oscillations in certain circumstances [5]. Another method, reported by DelGuidice *et al.* [2] is,

$$\frac{dH}{dT} = \left(\frac{\left(\frac{dH}{dx}\right)\left(\frac{dT}{dx}\right) + \left(\frac{dH}{dy}\right)\left(\frac{dT}{dy}\right)}{\left(\frac{dT}{dx}\right)^2 + \left(\frac{dT}{dy}\right)^2} \right) \tag{C.2}$$

Morgan *et al.* [5] presented the use of a simple backward difference approximation,

$$\left(\frac{dH}{dT}\right)_n = \frac{\left(H_n - H_{n-1}\right)}{\left(T_n - T_{n-1}\right)} \tag{C.3}$$

where n represents the timestep number. where n represents the timestep number. This scheme restricts the timestep severely according to reference [1] if a correct heat balance is to be maintained. Lewis and Roberts [4] claim that the last scheme is computationally quicker than the abovementioned averaging techniques.

REFERENCES

[1] Dalhuijsen A.J. and Segal A., Comparison of finite element techniques for solidification problems, *International Journal for Numerical Methods in Engineering,* **23**: 1807-1829, 1986.

[2] DelGuidice S., Comini G. and Lewis R.W., Finite element simulation of freezing processes in soils, *International Journal of Numerical and Analytical Methods in Geomechanics*, **2**: 223-235, 1978.

[3] Lemmon E.C., Multidimensional integral phase change approximations for finite element conduction codes, In R.W. Lewis, K. Morgan and O.C. Zienkiewicz, editors, *Numerical Methods in Heat Transfer,* Wiley, Chichester, 1981.

[4] Lewis R.W. and Roberts P.M., Finite element simulation of solidification problems, Applied Scientific Research, 44: 61-92, 1987.

[5] Lewis R.W., Morgan K. and Zienkiewicz O.C., An improved algorithm for heat conduction problems with phase change, *International Journal for Numerical Methods in Engineering,* **12**: 1191-1195, 1978.

[6] Lewis R.W., Morgan K., Thomas H.R. and Seetharamu K.N., The Finite Element Method in Heat Transfer Analysis, John Wiley & Sons, 1996.

[7] Viskanta R., Phase change heat transfer, In G.A. Lane, editor, *Solar Heat Storage Latent Heat Materials,* CRC Press, 1983.

[8] Voller V.R., Cross M. and Markatos N.C., An enthalpy method for convection/diffusion phase change, *International Journal for Numerical Methods in Engineering,* **24**: 271-284, 1987.

Appendix D

ANALYTICAL SOLUTION - SOLIDIFICATION

The analytical solution for a semi-infinite 1-D domain at a temperature T_∞, subject to a surface temperature of zero degrees, is calculated according to the following formulae:

The position of the solidification front X is obtained from,

$$X = 2\lambda \left(k_s t\right)^{1/2} \tag{D.1}$$

The temperature in the solid zone $\left(x \leq X \quad \text{or} \quad T \leq T_f\right)$, is given by,

$$T = \frac{T_f}{erf\lambda} erf \frac{x}{2\left(k_s t\right)^{1/2}} \tag{D.2}$$

The temperature in the liquid zone $\left(x \geq X \quad \text{or} \quad T \geq T_f\right)$, is given by,

$$T = T_\infty - \frac{T_\infty - T_f}{erfc\lambda \left(\frac{k_s}{k_l}\right)^{1/2}} erf \frac{x}{2\left(k_s t\right)^{1/2}} \tag{D.3}$$

where, λ in the above equations is obtained from,

$$\frac{e^{-\lambda^2}}{erf\lambda} - \left(\frac{k_s}{k_l}\right)^{1/2} \frac{T_\infty - T_f}{T_f} \frac{e^{-\lambda^2 \frac{k_s}{k_l}}}{erfc\lambda \left({k_s}/{k_l}\right)^{1/2}} = \frac{\lambda L \pi^{1/2}}{c_s T_f} \tag{D.4}$$

Here, *erf* is the error function and $erfc(x) = 1 - erf(x)$. The values of the error function were approximated from the formula below:

$$erf(x) = 1 - \left(a_1 b + a_2 b^2 + a_3 b^3 + a_4 b^4 + a_5 b^5\right) e^{-x^2} + \varepsilon(x) \qquad \text{(D.5)}$$

where,

$$b = \frac{1}{1 + px}$$

$$p = 0.3275911$$

$$a_1 = 0.254829529$$

$$a_2 = -0.284496736$$

$$a_3 = 1.421413741$$

$$a_4 = -1.453152027$$

$$a_5 = 1.061405429$$

$$\varepsilon(x) \leq 1.5 \times 10^{-7}$$

Appendix E

ANALYTICAL SOLUTION - MELTING

The analytical solution for a semi-infinite 1-D domain at a temperature of zero degrees, subject to a surface temperature of T_o, is calculated according to the following formulae:

The position of the melting front X is obtained from,

$$X = 2\lambda\left(k_l t\right)^{1/2} \tag{E.1}$$

The temperature in the liquid zone $\left(x \leq X \quad \text{or} \quad T \geq T_f\right)$, is given by,

$$T = T_o - \frac{T_o - T_f}{erfc\lambda} erf \frac{x}{2\left(k_l t\right)^{1/2}} \tag{E.2}$$

The temperature in the solid zone $\left(x \geq X \quad \text{or} \quad T \leq T_f\right)$, is given by,

$$T = \frac{T_f}{erfc\lambda\left({k_l}/{k_s}\right)^{1/2}} erfc \frac{x}{2\left(k_s t\right)^{1/2}} \tag{E.3}$$

where, λ in the above equations is obtained from,

$$\frac{e^{-\lambda^2}}{erf\lambda} - \left(\frac{k_s}{k_l}\right)^{1/2} \frac{T_f}{T_o - T_f} \frac{e^{-\lambda^2 \frac{k_l}{k_s}}}{erfc\lambda\left({k_l}/{k_s}\right)^{1/2}} = \frac{\lambda L \pi^{1/2}}{c_s\left(T_o - T_f\right)} \tag{E.4}$$

Here, erf is the error function and $erfc(x) = 1 - erf(x)$. The values of the error function were approximated from the formula below:

$$erf(x) = 1 - \left(a_1 b + a_2 b^2 + a_3 b^3 + a_4 b^4 + a_5 b^5\right) e^{-x^2} + \varepsilon(x) \tag{E.5}$$

where,

$$b = \frac{1}{1 + px}$$

$$p = 0.3275911$$

$$a_1 = 0.254829529$$

$$a_2 = -0.284496736$$

$$a_3 = 1.421413741$$

$$a_4 = -1.453152027$$

$$a_5 = 1.061405429$$

$$\varepsilon(x) \leq 1.5 \times 10^{-7}$$