

Unsteady Pipe-Flow Using the Petrov-Galerkin Finite Element Method

George Gerber

Thesis presented in partial fulfillment of the requirements for the degree of
Master of Engineering at the University of Stellenbosch.



Supervisor: Dr Philippe Maincon

April 2004

Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety or in part submitted it at any university for a degree.

Synopsis

Presented here is an Eulerian scheme for solving the unsteady pipe-flow equations. It is called the Characteristic Dissipative Petrov-Galerkin finite element algorithm. It is based on Hicks and Steffler's open-channel finite element algorithm [5]. The algorithm features a highly selective dissipative interface, which damps out spurious oscillations in the pressure field while leaving the rest of the field almost unaffected. The dissipative interface is obtained through upwinding of the test shape functions, which is controlled by the characteristic directions of the flow field at a node. The algorithm can be applied to variable grids, since the dissipative interface is locally controlled. The algorithm was applied to waterhammer problems, which included reservoir, deadend, valve and pump boundary conditions. Satisfactory results were obtained using a simple one-dimensional element with linear shape functions.

Samevatting

'n Euleriese skema word hier beskryf om die onbestendige pypvloei differensiaal vergelykings op te los. Dit word die Karakteristieke Dissiperende Petrov-Galerkin eindige element algoritme genoem. Die algoritme is gebaseer op Hicks en Steffler se oop-kanaal eindige element algoritme [5]. In hierdie algoritme word onrealistiese ossilasies in die drukveld selektief gedissipeer, sonder om die res van die veld te beïnvloed. Die dissiperende koppelvlak word verkry deur stroomop weegfunksies, wat beheer word deur die karakteristieke rigtings in die vloeiveld, by 'n node. Die algoritme kan dus gebruik word op veranderbare roosters, omdat die dissiperende koppelvlak lokaal beheer word. Die algoritme was toegepas op waterslag probleme waarvan die grenskondisies reservoirs, entpunte, kleppe en pompe ingesluit het. Bevredigende resultate was verkry vir hierdie probleme, al was die geïmplementeerde element een-dimensioneel met lineere vormfunksies.

Acknowledgement

I would like to express my gratitude to Philippe Maincon, who assisted me in the finite element method and the solution of non-linear problems.

I am also grateful to my wonderful family, for supporting me through my studies.

Contents

| | |
|---|------------|
| Nomenclature | vii |
| 1 Introduction | 1 |
| 1.1 Motivation for this Thesis | 1 |
| 1.2 A Brief Example of Waterhammer | 1 |
| 1.3 Literature study | 3 |
| 1.3.1 The turbulent history of CFD and finite element methods | 3 |
| 1.4 Outline | 5 |
| 2 Unsteady Pipe Flow from First Principles | 6 |
| 2.1 Introduction | 6 |
| 2.2 Conservation laws governing fluid dynamics | 7 |
| 2.3 Non-dimensional parameters that indicate the state of a fluid | 7 |
| 2.4 Fluid motion description: | |
| Lagrangian and Eulerian perspectives | 9 |
| 2.4.1 Lagrangian perspective | 9 |
| 2.4.2 Eulerian perspective | 10 |
| 2.5 Momentum conservation equation for a control body | 10 |
| 2.5.1 Forces on the control body | 11 |
| 2.5.2 Newton's second law | 12 |
| 2.6 Mass conservation equation for a control body | 14 |
| 2.6.1 Accounting for fluid elasticity | 16 |
| 2.6.2 Accounting for thin-walled pipe deformation | 17 |
| 2.6.3 Pipe support constraints | 18 |
| 3 Basic Computational Fluid Dynamics Theory | 20 |
| 3.1 Introduction | 20 |
| 3.2 Classification of Partial Differential Equations | 21 |
| 3.2.1 Important properties of PDE's | 21 |
| 3.2.2 Method of Characteristics | 22 |
| 3.2.3 Eigenvalue Method for a system of first-order PDEs | 22 |

CONTENTS

ii

| | | |
|----------|---|-----------|
| 3.2.4 | Classification of a single first-order PDE | 25 |
| 3.3 | Hyperbolic PDE's | 26 |
| 3.4 | Parabolic PDE's | 27 |
| 3.5 | Elliptic PDE's | 28 |
| 3.6 | Conservation and Non-Conservation forms of the governing equations . . . | 29 |
| 3.6.1 | Conservation form | 29 |
| 3.6.2 | Non-Conservation form | 31 |
| 3.6.3 | Sub-classes of the Conservation form | 32 |
| 3.7 | Jacobians of the Flux Vectors | 32 |
| 3.8 | Shock Capturing and Shock Fitting | 33 |
| 3.9 | Convection, Diffusion and the numerical difficulties that they introduce . . | 34 |
| 3.9.1 | Convection (Macroscopic transport) | 35 |
| 3.9.2 | Diffusion (Microscopic transport) | 36 |
| 3.9.3 | The form of convective and diffusive fluxes in differential equations | 37 |
| 3.9.4 | Eulerian, Lagrangian and Eulerian-Lagrangian schemes | 38 |
| 4 | Discretizing the Pipe-Flow Equations | 40 |
| 4.1 | Introduction | 40 |
| 4.2 | Why the Bubnov-Galerkin finite element is unsuitable | 41 |
| 4.2.1 | How Bubnov-Galerkin approximates derivatives in equations in the non-conservation form | 41 |
| 4.2.2 | Why upwinding is used to approximate odd-ordered derivatives in equations in non-conservation form | 43 |
| 4.2.3 | Finding the upwind direction through flux-vector splitting | 44 |
| 4.2.4 | Why discontinuous test functions introduces hyperbolic behaviour | 45 |
| 4.3 | Evolution from a Bubnov-Galerkin to a Petrov-Galerkin FEM | 46 |
| 4.3.1 | An open-channel flow problem, containing a hydraulic jump | 46 |
| 4.4 | Discretization of the unsteady pipe-flow equations | 50 |
| 4.4.1 | Governing equations | 51 |
| 4.4.2 | Trial functions | 52 |
| 4.4.3 | Approximate solution | 52 |
| 4.4.4 | Test functions | 52 |
| 4.4.5 | Jacobian of the coordinate transformation | 54 |
| 4.4.6 | Petrov-Galerkin finite element | 55 |
| 4.4.7 | θ implicit finite difference scheme | 58 |
| 4.4.8 | Residual of an element, with boundary conditions imposed | 60 |
| 4.4.9 | Jacobian matrix of an element | 60 |
| 4.5 | Solution algorithm | 63 |
| 4.5.1 | Newton-Raphson iteration | 63 |

| | |
|-----------------|---|
| <i>CONTENTS</i> | iii |
| 4.5.2 | Computationally efficient form of Newton-Raphson iteration 64 |
| 4.5.3 | Overall algorithm 64 |
| 4.6 | Summary of algorithm 65 |
| 4.6.1 | Overall solving process 65 |
| 4.6.2 | Newton-Raphson iterative solver 65 |
| 4.7 | General remarks concerning the algorithm 66 |
| 5 | Boundary Conditions 69 |
| 5.1 | Introduction 69 |
| 5.2 | Procedure for implementing boundary conditions 70 |
| 5.3 | Deadends 70 |
| 5.4 | Reservoirs 71 |
| 5.5 | Valves 71 |
| 5.5.1 | In-line Valve 71 |
| 5.5.2 | Reservoir valve 73 |
| 5.6 | Pumps 73 |
| 5.6.1 | Single pump-trip equations 74 |
| 5.6.2 | Series pump equations 83 |
| 5.6.3 | Parallel pump equations 83 |
| 6 | Stability Analysis 84 |
| 6.1 | Introduction 84 |
| 6.2 | Behaviour of numerical solution and error 84 |
| 6.3 | Fourier series approximation of the solution 85 |
| 6.4 | Linearization of the governing equations 86 |
| 6.4.1 | Linearizing the continuity equation 86 |
| 6.4.2 | Linearizing the momentum equation 87 |
| 6.5 | Petrov-Galerkin finite element formulation 87 |
| 6.6 | Fourier Stability analysis 88 |
| 6.7 | Amplitude figure 91 |
| 6.8 | Phase shift figure 91 |
| 7 | Test Cases 95 |
| 7.1 | Introduction 95 |
| 7.2 | Steady-state model 96 |
| 7.2.1 | Model objectives 96 |
| 7.2.2 | Flow problem statement 96 |
| 7.2.3 | Steady-state hand calculations 97 |
| 7.2.4 | Determining if the scheme has reached a steady-state 97 |

CONTENTS

iv

| | | |
|----------|---|------------|
| 7.2.5 | Verifying that the correct steady state was reached | 98 |
| 7.2.6 | Steady-state mass conservation | 99 |
| 7.3 | Rapid valve closure model | 100 |
| 7.3.1 | Model objectives | 100 |
| 7.3.2 | Flow problem statement | 102 |
| 7.3.3 | Joukowsky maximum head calculation | 102 |
| 7.3.4 | Comparison of the computed results with those of Streeter and Lai | 103 |
| 7.3.5 | Courant number dependency | 105 |
| 7.3.6 | Grid independence | 106 |
| 7.3.7 | Upwinding parameter dependency | 106 |
| 7.4 | Pump model | 110 |
| 7.4.1 | Model objectives | 110 |
| 7.4.2 | Flow problem statement | 110 |
| 7.4.3 | Comparison of computed results | 111 |
| 8 | Conclusion | 116 |
| 8.1 | Practical use of Characteristic Dissipative Galerkin scheme | 116 |
| 8.2 | Potential applications for industry | 116 |
| 8.3 | Further work | 117 |
| 8.3.1 | Finite elements from boundary conditions | 117 |

List of Figures

| | | |
|-----|--|----|
| 1.1 | Motion of a waterhammer wave | 2 |
| 2.1 | Control body for momentum conservation | 11 |
| 2.2 | Control body for mass conservation | 14 |
| 2.3 | Strain directions for a thin pipe-wall | 17 |
| 3.1 | Characteristic directions and Characteristic lines for a hyperbolic problem | 24 |
| 3.2 | Domain of dependence and influence for a hyperbolic PDE | 26 |
| 3.3 | Domain of dependence and influence for a parabolic PDE | 28 |
| 3.4 | Domain of dependence and influence for an elliptic PDE | 29 |
| 3.5 | Shock Capturing Approach | 34 |
| 3.6 | Shock Fitting Approach | 34 |
| 3.7 | Spatial Variation of Flux and Primitive Variables across a Shock Front . . | 35 |
| 4.1 | Examples of trial and test functions for Bubnov-Galerkin and Petrov-Galerkin schemes | 44 |
| 4.2 | Characteristic Directions for different Flow Types. | 45 |
| 4.3 | Analytic solution, showing hydraulic jump | 47 |
| 4.4 | Bubnov-Galerkin solution, showing spurious waves | 47 |
| 4.5 | Bubnov-Galerkin solution, with artificial viscosity added | 48 |
| 4.6 | Overall solving process | 66 |
| 4.7 | Newton-Raphson solver for time-step $n+1$ | 68 |
| 5.1 | Valve Discharge coefficient vs Opening size | 72 |
| 5.2 | χ -formulas for different zones of pump operation | 75 |
| 5.3 | Pump characteristic curves | 76 |
| 6.1 | Artificial damping for various combinations of upwinding and Courant numbers | 93 |
| 6.2 | Phase shifts for various combinations of upwinding and Courant numbers . | 94 |
| 7.1 | Two reservoirs connected by a steel pipeline | 96 |
| 7.2 | Steady-state convergence of time derivatives | 99 |

LIST OF FIGURES

vi

| | | |
|------|---|-----|
| 7.3 | Steady-state convergence of flow variables | 100 |
| 7.4 | Initial- and steady-state hydraulic grade lines | 101 |
| 7.5 | Non-conservation of mass - Steady state solution after 3300 time steps . . . | 102 |
| 7.6 | Two reservoirs connected by a steel pipeline, with a ball valve situated at the downstream reservoir | 102 |
| 7.7 | Head-time variation at valve | 104 |
| 7.8 | Computed valve head-time series by Streeter and Lai | 104 |
| 7.9 | Valve closure relationships implemented by Streeter and Lai | 105 |
| 7.10 | Courant number sensitivity analysis | 107 |
| 7.11 | Mesh coarseness sensitivity analysis | 108 |
| 7.12 | Sensitivity analysis of the upwinding parameter | 109 |
| 7.13 | Two reservoirs connected by a steel pipe, with a pump at midpoint | 111 |
| 7.14 | Computed and measured response of Tracy pumping plant | 112 |
| 7.15 | Pump behaviour after power failure | 114 |
| 7.16 | Hydraulic Gradelines and velocity distribution at zero and fifty seconds . . | 115 |

Nomenclature

Latin symbols

| | |
|-----------------------------|---|
| a | Speed of sound of water in a pipe (wavespeed) |
| a | Acceleration of control body |
| \tilde{a} | A property of the fluid (e.g. temperature, concentration) |
| A | Cross-sectional area of control body |
| A_j | Pump head- or torque-Fourier coefficient |
| \bar{A} | Convection matrix |
| $\frac{\bar{A}}{ \bar{A} }$ | Upwinding matrix, derived from convection matrix \bar{A} , whose elements have been normalised by its eigenvalues |
| B_j | Pump head- or torque-Fourier coefficient |
| \bar{B} | Source/sink vector |
| c | Speed of sound of water in a channel (wavespeed) |
| c | Specific heat constant |
| c | Circumference of the pipe |
| c_1 | Pipe support condition |
| \bar{c} | Vector containing the global cartesian coordinate values of a finite element |
| <i>Courant</i> | Courant number |
| C_d | Discharge coefficient of the valve |
| C^m | Function whose derivatives are interelement continuous up to degree m |
| \bar{C} | Convection matrix of discretised governing equations |
| dx | Longitudinal length of a control body |
| D | Pipe diameter |
| D | Exact solution |
| \bar{D} | Unknown head-velocity coefficient matrix |
| E | Pipe-wall elasticity modulus |
| E | Error due to round-off errors |
| \bar{f} | Residual of uncondensed element |
| \bar{f}_r | Residual of condensed element (only contains retained dofs) |
| F | Force |
| Fr | Froude number |

NOMENCLATURE

viii

| | |
|-------------------|--|
| F_i | Head or torque data value for χ_i |
| \bar{F} | Flux vector in cartesian x-space, containing functions of flux variables |
| $\bar{\bar{F}}_r$ | Residual of global equation system, based on condensed elements |
| g | Gravity acceleration constant |
| $ g $ | Norm of the complex eigenvalues of the amplification matrix |
| \bar{G} | Flux vector in cartesian y-space, containing functions of flux variables |
| $\bar{\bar{G}}$ | Amplification matrix of the numerical scheme |
| h | Pressure in metres head |
| h_{rsvr} | Head at reservoir |
| h_1, h_2 | Element's pressure dofs |
| h_{1b} | Head dof at pipe B's first node |
| H | Pressure in metres head |
| H_j | Joukowski head |
| \mathbf{H} | Head of pump (subscript "rated" indicates flow at optimum efficiency) |
| ΔH | Loss of head across the valve |
| i | Global equation system's Newton-Raphson iteration number |
| j | Node number |
| \bar{j} | Jacobian of uncondensed element |
| $\bar{\bar{j}}_r$ | Jacobian of condensed element (only contains retained dofs) |
| \bar{J} | Jacobian matrix of the coordinate transformation |
| $\bar{\bar{J}}_r$ | Jacobian of global equation system, based on condensed elements |
| k | Thermal conductivity |
| k | Wave number |
| k | Pump's Newton-Raphson iteration number |
| K | Elasticity of fluid |
| \bar{K} | Element coefficient matrix |
| L | Length of pipeline |
| L | Domain length |
| L | Length of finite element |
| L | Symbol for the dimension of length |
| \bar{L} | Source/sink vector |
| m | Mass of control body |
| M | Mach number |
| M | Symbol for the dimension of mass |
| \bar{M} | Mass matrix of discretised governing equations |
| n_1, n_2 | Scalar trial functions |
| \bar{n} | Surface normal vector |
| N | Numerical solution |

NOMENCLATURE

ix

| | |
|----------------------------|---|
| N | Number of elements that a wavelength spans |
| N | Trial function. Can be scalar, vector or matrix of functions |
| N | Pump speed (subscript "rated" indicates speed at optimum efficiency) |
| N^* | Test function. Can be scalar, vector or matrix of functions |
| p | Water pressure |
| \bar{P} | Known head-velocity coefficient matrix |
| $\bar{P} \cdot \bar{\phi}$ | Mass-momentum flux vector |
| q | Heat flux |
| q | Flow per unit width |
| \bar{q} | Convective flux of property in fluid |
| Q | Volumetric flux |
| Q | Flow through pump (subscript "rated" indicates flow at optimum efficiency) |
| \bar{Q} | Flux vector containing known values, based on the previous time-step's solution |
| R | Residual of approximated differential equation |
| Re | Reynold's number |
| s | Speed at which control body expands radially |
| S | Surface of control body |
| t | time |
| t | pipe-wall thickness |
| Δt | Time-step size |
| T | Temperature |
| T | Symbol for the dimension of time |
| T | Torque of pump (subscript "rated" indicates flow at optimum efficiency) |
| T_j | Temperature dof at node j |
| T_p | Natural period of pipeline |
| \bar{T} | Vector of temperature dofs |
| u | Flow velocity in the x-direction |
| u | Displacement |
| u | Convection velocity |
| \bar{U} | Solution vector containing flux variables |
| Δt | Time-step size |
| v | Flow velocity in the y-direction |
| v_0 | Initial steady-state velocity |
| $v_1 v_2$ | Element's velocity dofs |
| v_{2a} | Velocity dof at pipe A's second node |
| \bar{v} | Vector of velocities for x,y,z-directions |
| V | Volume of control body |
| \bar{V} | Matrix whose sum with the trial functions matrix gives the test function |

NOMENCLATURE

x

| | |
|------------|---|
| w | Flow velocity in the z-direction |
| $WH(\chi)$ | Dimensionless pump head function |
| $WT(\chi)$ | Dimensionless pump torque function |
| x | x-space |
| x_0 | Integration constant |
| Δx | Length of an element |
| \bar{x} | Global coordinate vector |
| y | y-space |
| y | Depth of flow in the channel |
| z | z-space |
| z | Elevation of the pipe above a specified datum |

Greek symbols

| | |
|----------------------|--|
| α | Slope of the pipe |
| α | Dimensionless pump speed variable |
| β | Dimensionless pump torque variable |
| γ | Dimensionless pump head variable |
| ε | Upwinding coefficient |
| ε | Pipe-wall strain |
| ε_1 | Circumferential pipe-wall strain |
| ε_2 | Radial pipe-wall strain |
| ε_3 | Longitudinal pipe-wall strain |
| ζ_k | Fourier coefficients for wavenumber k of the series |
| θ | Scalar parameter controlling the level of implicitness of discretised equation |
| λ | Friction factor for moving fluid in the pipe |
| $\lambda_{PQ_{ii}}$ | Characteristic direction of $\bar{P} \cdot \bar{Q}$ |
| $\lambda_{PR_{ii}}$ | Characteristic direction of $\bar{P} \cdot \bar{R}$ |
| $\bar{\lambda}_{PQ}$ | Eigenvalue matrix of $\bar{P} \cdot \bar{Q}$ |
| $\bar{\lambda}_{PR}$ | Eigenvalue matrix of $\bar{P} \cdot \bar{R}$ |
| μ | Elastic modulus |
| ν | Kinematic viscosity of fluid |
| ν | Poisson's ratio for three-dimensional strain |
| ν | Dimensional pump flow variable |
| ξ | Natural spatial coordinate |
| ρ | Density of fluid |
| ρ_k | Fourier coefficient for wavenumber k of the series |
| $\bar{\rho}$ | Density of property in fluid |

NOMENCLATURE

xi

| | |
|-------------------------------------|--|
| $\bar{\rho}_{\pm\frac{\lambda}{2}}$ | Density of property of fluid at $x = \pm\frac{\lambda}{2}$ |
| $\tilde{\rho}_0$ | Density of property of fluid at $x = 0$ |
| σ | Pipe-wall stress |
| σ_1 | Circumferential pipe-wall stress (hoop stress) |
| σ_2 | Radial pipe-wall stress |
| σ_3 | Longitudinal pipe-wall stress |
| τ | Shear stress |
| ϕ_0 | Flux per unit surface of fluid at $x = 0$ |
| $\phi_{\pm\frac{\lambda}{2}}$ | Flux per unit surface at $x = \pm\frac{\lambda}{2}$ |
| $\bar{\phi}$ | Vector containing element's pressure and velocity fields |
| $\bar{\phi}_c$ | Vector of condensed dofs |
| $\bar{\phi}_r$ | Vector of retained dofs |
| $\bar{\phi}^0$ | Initial solution vector of dofs |
| $\bar{\phi}^n, \bar{\phi}^{n+1}$ | Vector containing element's pressure and velocity fields for time-levels n and $n + 1$ |
| $\hat{\phi}$ | Vector containing approximate solution of pressure and velocity fields |
| $\Delta\bar{\phi}_r$ | Vector containing the increments of the retained dofs |
| $\bar{\Delta}_{\bar{\phi}}$ | Differential operator with respect to the element's dofs |
| Φ | Phase angle of the numerical scheme |
| Φ_e | Phase angle of the exact solution |
| χ | Angle describing the zone of operation of the pump |
| $\bar{\psi}$ | Vector containing the velocity and pressure fields |
| ω | Angular velocity of pump impeller |

Mathematical symbols

| | |
|-------------------------|---------------------------------|
| \square | First order tensor (vector) |
| $\bar{\square}$ | Second order tensor (matrix) |
| $\bar{\nabla}$ | Gradient vector operator |
| $\dot{\square}$ | Time derivative |
| $\square \cdot \square$ | Dot product of two tensors |
| \square^T | Transpose of a matrix or vector |
| \square^{-1} | Inverse of a tensor/matrix |
| $ \square $ | Absolute value |

Chapter 1

Introduction

1.1 Motivation for this Thesis

It has been said that the computer is incredibly fast, accurate and stupid; users are unbelievably slow, inaccurate and brilliant - the combination is an opportunity beyond imagination... (Author unknown)

To err is human... but for a real state-of-the-art catastrophe, you need a computer! (Philippe Maincon)

Many pipe- and channel-flow problems can be solved using simple hydraulic equations. However, these equations are nearly always applied to problems with simple geometry, due to the limited time available to solve them. Computational fluid dynamics (CFD), an established science, has opened the doors to solving problems of simple-physics-complex-geometry or even complex-physics-complex-geometry. The range of problems, which can be solved by these methods are numerous. The problems can range from the purely theoretical (with very little practical application) to very important, practical problems. Waterhammer is a practical problem in pipeline design, whose loadings require serious consideration during design and operation. Traditionally, the differential equations describing waterhammer were solved by the finite difference method. Today, the finite element method has also established itself as a viable method in solving these equations, but with the ability to model problems with greater flexibility. This thesis explores this unsteady flow problem and how the finite element method is applied to solve it.

1.2 A Brief Example of Waterhammer

Figure 1.1 shows the motion of a transient (Waterhammer pressure wave) along a pipe, with a reservoir and valve respectively at each end. Fluid transients are initial- boundary value problems. It is an initial value problem, because at the start of the motion

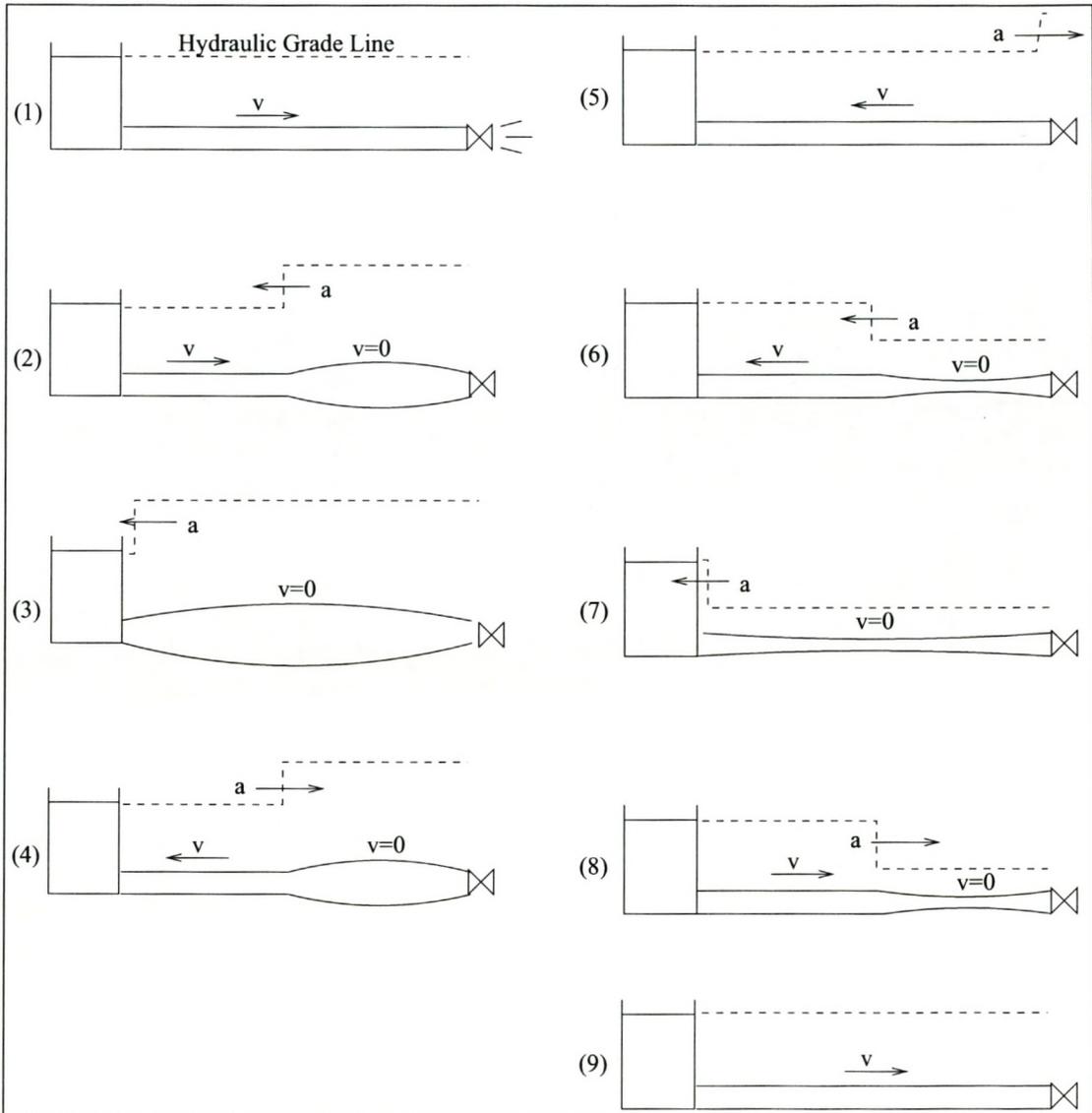


Figure 1.1: Motion of a waterhammer wave

the velocity and pressure has to be specified along the length of the pipe. It is also a boundary value problem, because the motion is influenced by the properties of the valve and reservoir. The pressure at the reservoir and its associated pipe-end will always be $H(m)$. When a waterhammer wave tries to introduce a different pressure than $H(m)$ at the reservoir end the reservoir will readjust the pressure back to $H(m)$ by changing the flow velocity at its end. When the valve at the other end of the pipe is closed no flow will exist. When a flow velocity disturbance reaches this end, which does not satisfy zero flow velocity, then the valve will readjust the flow velocity back to zero by changing the pressure at its end. The motion is described in eight steps (refer to fig. 1.1):

1. Initially, steady-, uniform-flow occurred between the reservoir and valve.
2. Waterhammer motion starts when the valve is instantly closed. (The valve generates a positive pressure wave, which satisfies zero flow)
3. The positive pressure wave reaches the reservoir end and disturbs the reservoir's boundary condition (pressure equal to $H(m)$).
4. The reservoir changes the flow velocity in such a manner that the pressure at its end is restored back to $H(m)$. This disturbance in the flow velocity moves back to the valve.
5. The disturbance reaches the valve and violates its boundary condition.
6. The valve restores its zero-flow condition by changing the pressure. This pressure wave moves back to the reservoir.
7. The wave reaches the reservoir and violates its boundary condition.
8. The reservoir restores its pressure $H(m)$ by changing the flow velocity, which moves back to the valve.
9. The flow velocity disturbance reaches the valve and the the whole motion from steps 1 to 8 is repeated, until all the excess energy in the system has been dissipated through friction.

1.3 Literature study

1.3.1 The turbulent history of CFD and finite element methods

The finite element method came about during the 1950's, when engineers seeked a "force-balance" method for analyzing large structural systems for aircraft. One of the first papers, which was presented on this topic was by Turner et al. [15]. The application of

the method to non-structural problems, such as elementary flow and electromagnetism, soon followed through the work of Zienkiewicz [16]. Continued research on the method, soon replaced the "force balance" concept with a more robust theoretical development based on variational calculus and Rayleigh-Ritz methods. However, it is this variational approach which slowed the development of the method for flow problems and gave the finite difference and finite volume methods their headstart. One of the main obstacles was that the momentum equation, based on the Eulerian reference-frame, is explicitly non-linear. This means that it cannot be known for certain if a functional, which is necessary for a variational approach exists, let alone be found. This is why the bulk of CFD research is based on finite difference and finite volume methods. (Finite difference methods solve partial differential equations by substituting derivatives with difference equations, derived from Taylor expansions. Finite volume methods integrate the conservation equations over a discretization, using the divergence theorem and replacing cell fluxes with difference quotients.) Notwithstanding the problems given above, research on the finite element method continued, because of its attractive ability to use unstructured meshes, while the two alternatives cannot. Weighted residual finite element methods, such as the Galerkin method, later appeared on the scene. These methods are in many cases equivalent to the Ritz-methods, but without the problems introduced by functionals. Today, there exists a large number of finite element algorithms, which are able to solve various fluid flow problems satisfactorily. It has further been shown that in some cases the Galerkin finite element method is a generalization of the finite difference and finite volume methods. The contributions of various researchers to the finite element method, within the context of waterhammer problems, are given below in chronological order.

- 1965** Zienkiewicz uses the finite element method, for the first time, to solve elementary flow problems [16].
- 1972** Oden presents one of the first weighted residual finite element formulations for the Navier-Stokes equations [12].
- 1973** Dupont proves that the traditional Bubnov-Galerkin finite element method does not achieve optimal accuracy for hyperbolic partial differential equations, as it does for elliptic and parabolic ones (The differential equations describing waterhammer is of the hyperbolic type). [3].
- 1974** Dendy [2] suggests a Petrov-Galerkin finite element formulation for the scalar advection equation, which is also a hyperbolic partial differential equation, with the following test function: $N^* = N + \varepsilon \frac{\partial N}{\partial x}$.
- 1976** Raymond and Garder shows that an upwinding coefficient of $\varepsilon = \frac{2}{\sqrt{15}}$, minimises phase errors for Dendy's test function [13].

- 1981** Johnson shows that for a hyperbolic partial differential equation, the discontinuities of the exact solution across the characteristics are numerically captured in an almost exact fashion by the Dissipative Galerkin scheme, even when the characteristics are skew to the computational grid [7].
- 1982** Hughes and Brooks provides error analysis for various Petrov-Galerkin formulations on the scalar advection-diffusion equations [6].
- 1984** Katopodes applies the Dissipative Galerkin scheme on the non-conservation form of the open-channel equations [9]. He further shows how the scheme selectively damps the short, spurious wavelengths, in the solution [10]. He then extends the scheme for two dimensional open-channel flow [11].
- 1992** Hicks and Steffler formulates the Characteristic Dissipative Galerkin scheme for open-channel flows in conservation form. They also show that this scheme compares favourably to the commonly used four-point implicit finite difference scheme, known as the Preissman scheme [5].

1.4 Outline

- Chapter 2** discusses the fundamentals of fluid dynamics and shows how the mass and momentum conservation equations are derived from first principles.
- Chapter 3** shows how to identify the different types of partial differential equations and what numerical obstacles each type presents.
- Chapter 4** discusses why the traditional Galerkin finite element is unable to solve the unsteady pipe-flow equations. The Petrov-Galerkin finite element is presented as an alternative and its implementation shown. The chapter ends by explaining how the algorithm is implemented to solve these non-linear equations.
- Chapter 5** explains how the different boundary conditions for waterhammer were implemented.
- Chapter 6** introduces stability analysis as a method to determine the numerical performance of the finite element method, for various ranges of parameter values.
- Chapter 7** presents the results of various waterhammer test cases, to which the Petrov-Galerkin finite element was applied.
- Chapter 8** the conclusion, summarizes findings and suggestions for further work.

Chapter 2

Unsteady Pipe Flow from First Principles

2.1 Introduction

This chapter discusses the fundamentals of fluid dynamics and shows how the momentum and mass conservation differential equations for unsteady pipe flow is derived from first principles. The topics presented are:

Conservation laws governing fluid dynamics. This section discusses the flow field variables encountered in fluid dynamics, as well as the equations which are used to solve these fields.

Non-dimensional parameters that indicate the state of a fluid. The Reynold's, Mach and Froude numbers are presented and it is shown how these parameters help to determine the dominant forces which are acting on the fluid. The flow problem can then be simplified by neglecting the insignificant forces.

Fluid motion description. The Lagrangian and Eulerian perspectives are presented, by which a flow is described mathematically. Each perspective produces a unique form of the governing equation, which has far-reaching effects on numerical schemes.

Momentum conservation equation for a control body. The momentum equation is derived from first principles and its underlying assumptions are given.

Mass conservation equation for a control body. The continuity equation is derived from first principles and its underlying assumptions are given.

2.2 Conservation laws governing fluid dynamics

The flow of a fluid is described mathematically by five dependent variables. These variables are pressure, density and flow velocity in the x , y , z -directions (p, ρ, v, u, w). These five variables depend on four independent variables, namely position in space (x, y, z) and time (t). To solve for the five dependent variables we need five independent constraints/equations. These equations are called the conservation laws. They are

- Conservation of mass.
- Conservation of momentum in the x , y , z -directions.
- Conservation of energy.

In many civil engineering applications, the density of the fluid can be regarded as constant, leaving only four dependent variables. Further, for one dimensional problems the number of dependent variables reduces to two (p, v). We therefore require only two conservation laws. This thesis regards the flow as one dimensional and the fluid density as constant. Consequently, only the conservation laws of mass and momentum in the x -direction are required.

2.3 Non-dimensional parameters that indicate the state of a fluid

This section presents the Reynold, Mach and Froude numbers, which helps us to identify the dominating influences on the flow. This enables us to reduce our generalised problem into a more specific one, by neglecting the smaller influences.

Reynolds number The Reynolds number is defined as

$$Re = \frac{vD}{\nu} \tag{2.1}$$

where

- v is the flow velocity
- D is the characteristic length (in the case of a pipe, this is the diameter)
- ν is the kinematic viscosity of the fluid

The Reynolds number indicates the relative magnitude of the inertia forces and viscous forces. Reynolds numbers close to zero, indicate that inertia forces are

negligible. This type of flow is *laminar*. High Reynolds numbers indicate that viscous forces are negligible. This type of flow is *turbulent*.

The kinematic viscosity (ν) relates the shear stress with the rate of shear deformation within the fluid. Governing equations which take viscosity into account are called **Navier-Stokes equations** and those that neglect it are called **Euler equations**. In this thesis we will not use the full Navier-Stokes equations. The equations used will be simpler, by introducing only a single friction term, containing a friction factor λ that accounts for viscosity and surface roughness.

Mach number The Mach number is defined as

$$M = \frac{v}{a} \quad (2.2)$$

where

- v is the flow velocity
- a is the speed of sound for the fluid

The Mach number indicates the relative magnitude of the flow speed and the speed of sound of the fluid. This number provides a measure of the compressibility, or change in density, due to motion. A mach number less than 0.14 implies less than 1% change in density due to motion. Mach numbers up to three are common for fighter aircraft. Since pipe-flow involves low flow speeds ($0 - 2m/s$) and high sound speeds ($300 - 1200m/s$) the pipe-flow problems for this thesis will involve low Mach numbers and hence low compressibility.

Froude number The Froude number applies to open-channel flows, but is presented here because this thesis uses a number of open-channel flow examples. The Froude number is defined as

$$Fr = \frac{v}{\sqrt{gy}} \quad (2.3)$$

where

- v is the flow velocity
- y is the the depth of flow in the channel
- g is the gravity acceleration constant

The Froude number provides a measure of the relative importance of inertia and gravity forces. For very small Froude numbers, gravity is able to keep the water surface flat and the resistance to motion associated with the generation of surface waves is negligible.

Table 2.1: *Flow Classification*

| Viscosity (ν) | Density (ρ) | |
|--|--|---|
| | Incompressible (ρ constant) | Compressible (ρ varies) |
| Inviscid flow ($\nu = 0$) | Potential/Irrotational flow (if vorticity is zero) | Gas dynamics (with "k=0") |
| Boundary layer flow ($\nu \neq 0$) (Viscosity important near boundary) | Laminar flow (<i>low Re</i>) Turbulent flow (<i>high Re</i>) | Heat transfer |
| Separated flow ($\nu \neq 0$) (Viscosity important everywhere) | Laminar flow (<i>very low Re</i>) Turbulent flow (<i>mod. to high Re</i>) | Heat transfer |

Table 2.1 gives an indication of the flow problem type, by rating the importance of viscosity and compressibility. In this thesis, we are looking at a problem with predominantly high Reynolds numbers, low Mach numbers (i.e. an incompressible fluid), where the viscosity is important near the boundary layer.

2.4 Fluid motion description: Lagrangian and Eulerian perspectives

2.4.1 Lagrangian perspective

The Lagrangian perspective is based on a **control body**. The body can have any shape, but the collection of particles contained within this shape are fixed. Our focus therefore is on a volume, which is moving with the fluid such that the same fluid particles are always inside it. The perspective provides us with the ability to monitor how the flow fields change with time for those particles. The position of the control body is completely described by its initial position and time.

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} f(x_0, y_0, z_0, t) \\ g(x_0, y_0, z_0, t) \\ h(x_0, y_0, z_0, t) \end{Bmatrix} \quad (2.4)$$

The velocity vector of the control body is obtained by taking the time derivative of the position vector.

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} \dot{f}(x_0, y_0, z_0, t) \\ \dot{g}(x_0, y_0, z_0, t) \\ \dot{h}(x_0, y_0, z_0, t) \end{pmatrix} \quad (2.5)$$

The velocity vector for a fixed set of particles, can therefore be completely described by their initial position and time, using the Lagrangian method. This also applies to the pressure field. In section 3.6 we will see that the equations obtained from the Lagrangian perspective are in non-conservation form. This form has advantages, as well as disadvantages for a numerical scheme. This will become clearer in section 3.8, which discusses shock-capturing and shock-fitting.

2.4.2 Eulerian perspective

The Eulerian perspective is based on a **control volume** fixed in space, with fluid moving through it (e.g. a length of buried pipe). Particles move through this volume. The volume's shape can also be arbitrary. This perspective therefore provides us with the ability to monitor how the flow fields change with time within that specific volume. The velocity-field depends on the location of the fixed control volume and the time (t). This also applies to the pressure field.

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} p(x, y, z, t) \\ q(x, y, z, t) \\ r(x, y, z, t) \end{pmatrix} \quad (2.6)$$

In section 3.6 we will see that the equations obtained from the Eulerian perspective are in conservation form. This form also has advantages and disadvantages for numerical schemes, which will be discussed in section 3.8.

2.5 Momentum conservation equation for a control body

In this section the momentum equation is derived for a control body, using the Lagrangian perspective. The control body moves along the inside of the pipe and its cross-sectional shape is the same as the pipe's. The rate of change of the cross-section is small. The equation is based on Newton's second law and assumes that there is little variation in fluid density

The variables used in the derivation of the equation are

V is the volume of the control body

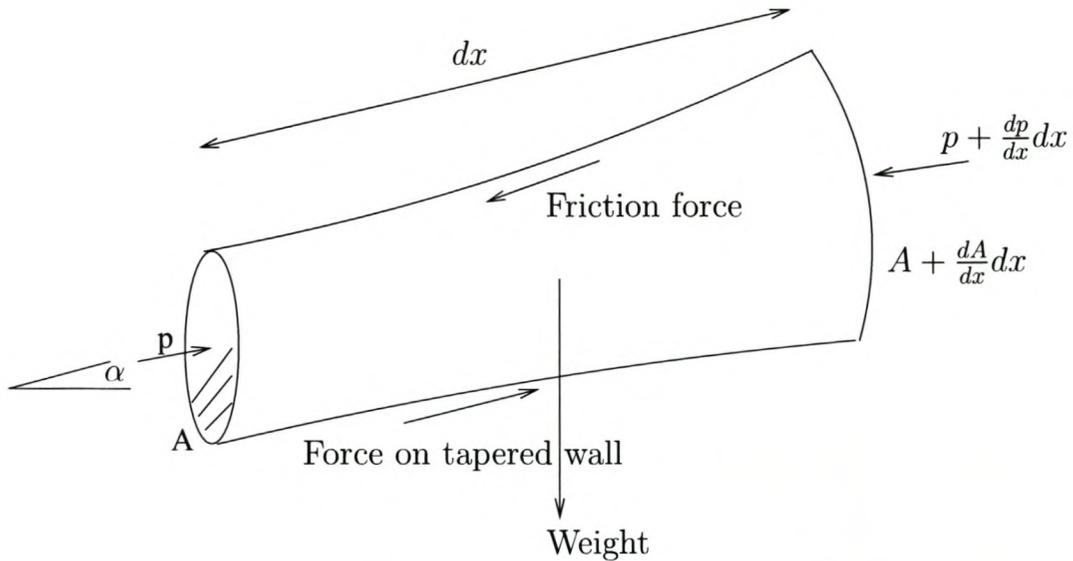


Figure 2.1: Control body for momentum conservation

A is the cross-sectional area of the control body

dx is the longitudinal length of the control body

p is the pressure within the fluid

v is the flow velocity of the fluid

H is the hydraulic head of the fluid

ρ is the density of the fluid

λ is the friction factor for the moving fluid in the pipe

g is the gravitational acceleration constant

D is the diameter of the pipe

z is the elevation of the pipe, above a specified datum

α is the slope of the pipe

2.5.1 Forces on the control body

The forces exerted on the control body can be divided into surface forces and body forces. Second order terms are neglected throughout.

Surface forces

Fluid pressure on transverse ends This force is equal to the nett value of the pipe end's pressure, times its cross-sectional area.

$$F_1 = pA - \left(p + \frac{\partial p}{\partial x} dx \right) \left(A + \frac{\partial A}{\partial x} dx \right) \quad (2.7)$$

$$= -A \frac{\partial p}{\partial x} dx - p \frac{\partial A}{\partial x} dx \quad (2.8)$$

The $\frac{\partial p}{\partial x} \frac{\partial A}{\partial x} dx^2$ term is negligible.

Tapered pipe-wall force This axial force is equal to the product of the average pressure within the body (mid-length pressure) and the change of cross section area.

$$F_2 = \left(p + \frac{\partial p}{\partial x} \frac{dx}{2} \right) \frac{\partial A}{\partial x} dx \quad (2.9)$$

$$= p \frac{\partial A}{\partial x} dx \quad (2.10)$$

The $\frac{\partial p}{\partial x} \frac{\partial A}{\partial x} dx^2$ term is negligible.

Friction force due to pipe-wall This force is equal to the product of the fluid shear stress at the pipe-wall and the pipe's internal surface.

$$F_3 = -\tau \pi D dx \quad (2.11)$$

$$= - \left(\frac{\lambda \rho v |v|}{8} \right) \pi D dx \quad (2.12)$$

where λ is found from the Moody diagram in any hydraulics text book.

Body forces

Gravitational force This force is equal to the product of the fluid density, the body length, the average cross-sectional area of the control body and the gravity acceleration component along the pipe axis.

$$F_4 = -\rho V g \sin(\alpha) dx = -\rho g \left(A + \frac{\partial A}{\partial x} \frac{dx}{2} \right) dx \sin(\alpha) \quad (2.13)$$

$$= -\rho g A dx \sin(\alpha) \quad (2.14)$$

2.5.2 Newton's second law

Newton's second law states that the product of the control body's mass and its acceleration is equal to the sum of the external forces acting on the body.

$$\sum F = ma \quad (2.15)$$

$$F_1 + F_2 + F_3 + F_4 = \rho V \dot{v} \quad (2.16)$$

Substitution of the surface and body forces gives

$$-A \frac{\partial p}{\partial x} dx - \frac{\lambda \rho v |v|}{8} \pi D dx - \rho g A dx \sin(\alpha) = (\rho A dx) \dot{v} \quad (2.17)$$

Simplifying gives

$$(\rho A dx) \dot{v} + A \frac{\partial p}{\partial x} dx + \frac{\lambda \rho v |v|}{8} \pi D dx + \rho g A dx \sin(\alpha) = 0 \quad (2.18)$$

Dividing by $\rho A dx$ gives

$$\dot{v} + \frac{\left(\frac{\partial p}{\partial x}\right)}{\rho} + \frac{\lambda v |v| \pi D}{8A} + g \sin(\alpha) = 0 \quad (2.19)$$

$$\dot{v} + \frac{\left(\frac{\partial p}{\partial x}\right)}{\rho} + \frac{\lambda v |v|}{2D} + g \sin(\alpha) = 0 \quad (2.20)$$

Expanding the total derivative gives

$$\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} + \frac{\lambda v |v|}{2D} + \frac{\left(\frac{\partial p}{\partial x}\right)}{\rho} + g \sin(\alpha) = 0 \quad (2.21)$$

Further, the hydraulic gradeline head is defined as

$$h = z + \frac{p}{\rho g} \quad (2.22)$$

Taking the partial derivative with respect to x and assuming ρ to be constant gives

$$\frac{\partial h}{\partial x} = \frac{\partial z}{\partial x} + \frac{1}{\rho g} \frac{\partial p}{\partial x} \quad (2.23)$$

Substituting $\frac{\partial z}{\partial x} = \sin(\alpha)$ in the above equation gives

$$\frac{\partial h}{\partial x} = \sin(\alpha) + \frac{1}{\rho g} \frac{\partial p}{\partial x} \quad (2.24)$$

Multiplying by g gives

$$g \frac{\partial h}{\partial x} = \frac{\left(\frac{\partial p}{\partial x}\right)}{\rho} + g \sin(\alpha) \quad (2.25)$$

Substitution of equation 2.25 into 2.21 gives the momentum equation

$$\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} + \frac{\lambda v |v|}{2D} + g \frac{\partial h}{\partial x} = 0 \quad (2.26)$$

This is the momentum equation with h and v as unknowns. It holds for:

- Liquids, since ρ was assumed to be constant
- Converging and diverging flow
- Flows for which the friction factor λ is the same as in steady flow

2.6 Mass conservation equation for a control body

In this section the equation for the continuity of mass is derived for a control body, using the Lagrangian perspective. The control body moves along the pipe and its cross-sectional shape is determined by the pipe's cross-section. The equation accounts for the elasticity of the fluid, as well as the elasticity of the pipe wall.

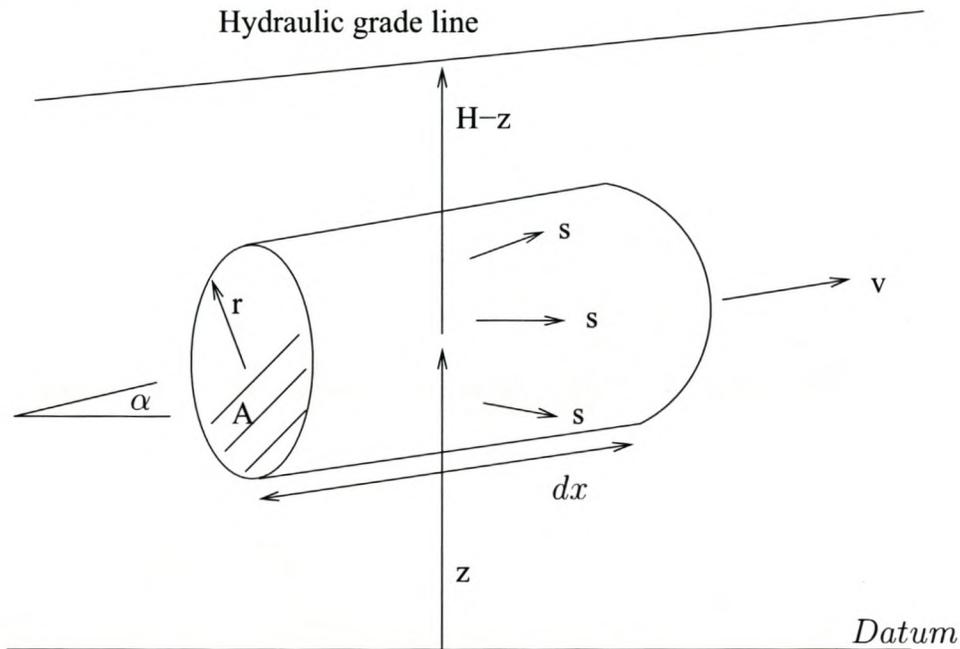


Figure 2.2: Control body for mass conservation

The variables used in the derivation of the equation are

V is the volume of the control body

A is the cross-sectional area of the control body

dx is the length of the control body along the pipe axis

p is the pressure within the fluid

v is the flow velocity of the fluid

h is the hydraulic head of the fluid

ρ is the density of the fluid

K is the elasticity of the fluid

σ and ε are the stress and strain in the pipe-wall

ν is Poisson's ratio for three-dimensional strain

a is the wavespeed of water in a pipe

g is the gravitational acceleration constant

D is the diameter of the pipe

r is the radius of the pipe

c is the circumference of the pipe

z is the elevation of the pipe, above a specified datum

α is the slope of the pipe

s is the speed at which the control body expands radially

v is the speed of the control body along the pipe axis

The mass of a control body can be written as

$$\rho V = m \quad (2.27)$$

Taking the time derivatives of this equation gives

$$\dot{\rho}V + \rho\dot{V} = \dot{m} \quad (2.28)$$

For a control body the mass is constant, therefore

$$\dot{\rho}V + \rho\dot{V} = 0 \quad (2.29)$$

Dividing by ρV gives

$$\frac{\dot{\rho}}{\rho} + \frac{\dot{V}}{V} = 0 \quad (2.30)$$

Substituting $V = A dx$ in the above equation gives

$$\frac{\dot{\rho}}{\rho} + \frac{\dot{A}dx + A\dot{d}x}{A dx} = 0 \quad (2.31)$$

$$\frac{\dot{\rho}}{\rho} + \frac{\dot{A}}{A} + \frac{\dot{d}x}{dx} = 0 \quad (2.32)$$

Now let us examine \dot{V} .

For a control volume, \dot{V} is the rate at which volume/mass is moving through its surface S .

For a control body, \dot{V} is the rate at which the volume of the body expands or contracts.

Further, Gauss's divergence theorem can be applied to control volumes and control bodies.

If we use the divergence theorem, we can say

$$\dot{V} = \int_S \bar{n} \cdot \bar{v} dS = \int_V \bar{\nabla} \cdot \bar{v} dV \quad (2.33)$$

Assume that V is the volume of an infinitesimal control body and that $\bar{\nabla} \cdot \bar{v}$ is constant for such a small body. We can therefore easily evaluate the integral and get

$$\dot{V} = \bar{\nabla} \cdot \bar{v} V \quad (2.34)$$

$$\frac{\dot{V}}{V} = \bar{\nabla} \cdot \bar{v} \quad (2.35)$$

Substituting $V = dxA$ into equation 2.35 and evaluating $\bar{\nabla} \cdot \bar{v}$ gives

$$\frac{\dot{dx}A + dx\dot{A}}{dxA} = \frac{\partial v}{\partial x} + \frac{\partial s}{\partial r} \quad (2.36)$$

$$\frac{\dot{dx}}{dx} + \frac{\dot{A}}{A} = \frac{\partial v}{\partial x} + \frac{\partial s}{\partial r} \quad (2.37)$$

We are however, only interested in the rate of deformation of the control body along the pipe axis (in the x-direction)

$$\frac{\dot{dx}}{dx} = \frac{\partial v}{\partial x} \quad (2.38)$$

Substituting the above equation into equation 2.32 gives

$$\frac{\dot{\rho}}{\rho} + \frac{\dot{A}}{A} + \frac{\partial v}{\partial x} = 0 \quad (2.39)$$

2.6.1 Accounting for fluid elasticity

The relationship between the volumetric strain of a body of fluid and its corresponding pressure is given by

$$dp = -K \frac{dV}{V} \quad (2.40)$$

Dividing by dt gives

$$\dot{p} = -K \frac{\dot{V}}{V} \quad (2.41)$$

Substituting equation 2.41 into equation 2.30 gives the usefull relationship

$$\dot{p} = K \frac{\dot{\rho}}{\rho} \quad (2.42)$$

$$\frac{\dot{p}}{K} = \frac{\dot{\rho}}{\rho} \quad (2.43)$$

2.6.2 Accounting for thin-walled pipe deformation

The equations for the circumference of a circle and its first derivative are

$$c = 2\pi r \quad (2.44)$$

$$dc = 2\pi dr \quad (2.45)$$

The equation for the area of a circle and its first derivative are

$$A = \pi r^2 \quad (2.46)$$

$$dA = 2\pi r dr \quad (2.47)$$

The hoop strain in a thin-walled pipe is given by

$$\varepsilon_1 = \frac{dc}{c} \quad (2.48)$$

Substituting equations 2.44 and 2.45 into equation 2.48 gives

$$\varepsilon_1 = \frac{2\pi dr}{2\pi r} = \frac{\frac{1}{r}(2\pi r dr)}{2\pi r} \quad (2.49)$$

Substituting equations 2.46 and 2.47 into equation 2.49 gives

$$\varepsilon_1 = \frac{dA}{2A} \quad (2.50)$$

Dividing by dt gives

$$2\varepsilon_1 = \frac{\dot{A}}{A} \quad (2.51)$$

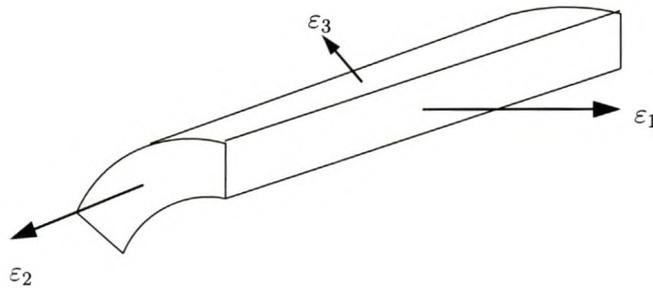


Figure 2.3: Strain directions for a thin pipe-wall

The total hoop strain ε_1 is found by using Hooke's law

$$\varepsilon_1 = \frac{1}{E} (\sigma_1 - \nu\sigma_2 - \nu\sigma_3) \quad (2.52)$$

where σ_2 and σ_3 are the axial and radial stresses respectively. We can neglect σ_3 , therefore

$$\varepsilon_1 = \frac{1}{E} (\sigma_1 - \nu\sigma_2) \quad (2.53)$$

Taking the time derivatives gives

$$\dot{\epsilon}_1 = \frac{1}{E} (\dot{\sigma}_1 - \nu \dot{\sigma}_2) \quad (2.54)$$

Substituting equation 2.54 into equation 2.51 gives the usefull relationship

$$\frac{\dot{A}}{A} = \frac{2}{E} (\dot{\sigma}_1 - \nu \dot{\sigma}_2) \quad (2.55)$$

Finally, substituting equations 2.55 and 2.43 into equation 2.39 gives

$$\frac{\dot{p}}{K} + \frac{2}{E} (\dot{\sigma}_1 - \nu \dot{\sigma}_2) + \frac{\partial v}{\partial x} = 0 \quad (2.56)$$

2.6.3 Pipe support constraints

Equation 2.56 provides us with the flexibility to model pipes with different support conditions, due to the pipe-wall stress terms. Three support conditions are considered. They are

1. Pipe anchored at its upstream end only
2. Pipe anchored against axial movement
3. Pipe anchored with expansion joints throughout its length

For all three support conditions the time derivative of the pipe-wall hoop stress is

$$\dot{\sigma}_1 = \frac{\dot{p}D}{2t} \quad (2.57)$$

where

- p is the internal pressure
- D is the pipe diameter
- t is the wall thickness

Assuming quasi-static axial pipe behaviour, the time derivative of the longitudinal stress for the above three support conditions is

1. $\dot{\sigma}_2 = \frac{\dot{p}A}{\pi Dt} = \frac{\dot{p}D}{4t}$
2. $\dot{\sigma}_2 = \nu \dot{\sigma}_1$
3. $\dot{\sigma}_2 = 0$

Substitution of the above stress rates into equation 2.56 yields the following generalised equation

$$\frac{\dot{p}}{\rho} + a^2 \frac{\partial v}{\partial x} = 0 \quad (2.58)$$

where the square of the pipe wavespeed (a) is defined as

$$a^2 = \frac{\frac{K}{\rho}}{1 + c_1 \left(\frac{KD}{Et}\right)} \quad (2.59)$$

and the different c_1 values correspond to the different pipe support conditions

1. $c_1 = 1 - \frac{\nu}{2}$
2. $c_1 = 1 - \nu^2$
3. $c_1 = 1$

Further, the piezometric head is given by:

$$p = \rho g (h - z) \quad (2.60)$$

Dividing by dt gives

$$\dot{p} = \rho g (\dot{h} - \dot{z}) \quad (2.61)$$

Expanding the total derivatives gives

$$\dot{p} = \rho g \left(v \frac{\partial h}{\partial x} + \frac{\partial h}{\partial t} - v \frac{\partial z}{\partial x} - \frac{\partial z}{\partial t} \right) \quad (2.62)$$

Substituting (2.62) in (2.58) yields the final continuity equation

$$v \frac{\partial h}{\partial x} + \frac{\partial h}{\partial t} - v \sin \alpha + \frac{a^2}{g} \frac{\partial v}{\partial x} = 0 \quad (2.63)$$

This continuity equation applies to

- compressible liquids
- elastic pipes
- quasi-static axial pipe behaviour

Chapter 3

Basic Computational Fluid Dynamics Theory

3.1 Introduction

This chapter discusses how one can distinguish between different partial differential equations, what properties the different types possess, as well as the numerical challenges that they introduce. The topics presented are

Classification of partial differential equations. This section shows how to classify partial differential equations using the eigenvalue method. The method is also used for deriving the characteristic and compatibility equations. We will later see the importance of the characteristics when they are embedded into the test shape functions of a Petrov-Galerkin finite element.

Hyperbolic partial differential equations. (HPDE's) This section discusses the properties of HPDE's. HPDE's describe problems such as unsteady pipe and open-channel flow or steady two-dimensional supercritical flow in open-channels. This thesis is concerned with hyperbolic partial differential equations.

Parabolic partial differential equations. (PPDE's) This section discusses the properties of PPDE's. PPDE's describe problems such as: pollutant diffusion in water, unsteady groundwater seepage, heat conduction or two-dimensional critical flow in open-channels.

Elliptic partial differential equations. (EPDE's) This section discusses the properties of EPDE's. EPDE's are encountered in steady pipe and open-channel flow problems or two-dimensional subcritical flow in open-channels.

Conservation and non-conservation forms of the governing equations. This section shows how to distinguish between the stated forms. It also discusses why the

non-conservation form, which we will use, produces spurious oscillations for most numerical schemes.

Jacobians of the flux vectors. This section shows how to transform a differential equation which is in conservation form into the form which the eigenvalue method uses. This transformation is not used in this thesis. It is mentioned, because of its use in the open-channel equations, which are related to the pipe flow equations.

Shock capturing and shock fitting. This section shows two different ways of determining the influence that shock waves have on the flow fields in our solution. It also shows why we need to proceed with caution, when we use the shock capturing method with our equations in non-conservation form.

Convection, diffusion and the numerical difficulties that they introduce. This section shows how convection and diffusion terms can be identified in a partial differential equation. The flows studied in this thesis are convection dominated. The convective terms introduce dispersive errors in our solution.

3.2 Classification of Partial Differential Equations

Any PDE (partial differential equation) can be classified into one of three distinct groups (a set of equations might be a mixture of groups). Each group has a unique mathematical behaviour. The groups are: HPDE's, PPDE's and EPDE's. HPDE's usually describe unsteady convective flows (which are the focus of this thesis). PPDE's usually describe unsteady diffusive flows. EPDE's usually describe steady flows. A numerical scheme needs to conform to the behaviour of its PDE in order to converge to the correct solution. There are a number of ways to classify PDE's. Two are: the traditional "method of characteristics" and the eigenvalue method. Both methods will be explained, but further developments will be based on the eigenvalue method, since the test shape functions of the Petrov-Galerkin finite element uses the eigenvalues obtained from the eigenvalue method.

3.2.1 Important properties of PDE's

Before we look at how to classify PDE's and how each type behaves, it is important to note some general properties of PDE's:

- The derivatives with the highest-order appearing in a PDE control the behaviour of the PDE. This means that the classification procedures, shown in the following sections, are primarily applied to the highest-order derivatives.
- Any PDE of order higher than one, can be represented as a system of first-order derivatives.

- Coordinate transformations do not alter the PDE's type. This is a crucial point, because at some point in the derivation of our finite element, we will need to transform our coordinate system from global coordinates (x) to natural coordinates (ξ).

3.2.2 Method of Characteristics

The method of characteristics is a method for transforming a PDE (conditionally) into an ODE (ordinary differential equation). Finite difference schemes were then built on these ODE's. By understanding the conditions, which makes this transformation possible, we know the behaviour of the PDE. These conditions represent lines in xt -space (called characteristic lines) and it is only on those lines that the ODE (called the compatibility equations) is defined. In contrast, the PDE is defined everywhere within the domain. The concepts, such as characteristic lines and compatibility equations explained here, are also used in the eigenvalue method, which will be explained in detail next.

3.2.3 Eigenvalue Method for a system of first-order PDEs

Consider a set of general quasi-linear PDE's:

$$a_1 \frac{\partial u}{\partial t} + b_1 \frac{\partial u}{\partial x} + c_1 \frac{\partial u}{\partial y} + d_1 \frac{\partial v}{\partial t} + e_1 \frac{\partial v}{\partial x} + f_1 \frac{\partial v}{\partial y} = L_1 \quad (3.1)$$

$$a_2 \frac{\partial u}{\partial t} + b_2 \frac{\partial u}{\partial x} + c_2 \frac{\partial u}{\partial y} + d_2 \frac{\partial v}{\partial t} + e_2 \frac{\partial v}{\partial x} + f_2 \frac{\partial v}{\partial y} = L_2 \quad (3.2)$$

These two equations can represent two unsteady velocity-fields (dependent variables u and v). They have three independent variables (time t and x, y -space). These equations are general, because every combination of first order derivatives for the dependent variables are present.

Now let

$$\bar{w} = \begin{Bmatrix} u \\ v \end{Bmatrix}$$

Equations 3.1 and 3.2 can be written in matrix form.

$$\begin{bmatrix} a_1 & d_1 \\ a_2 & d_2 \end{bmatrix} \cdot \frac{\partial \bar{w}}{\partial t} + \begin{bmatrix} b_1 & e_1 \\ b_2 & e_2 \end{bmatrix} \cdot \frac{\partial \bar{w}}{\partial x} + \begin{bmatrix} c_1 & f_1 \\ c_2 & f_2 \end{bmatrix} \cdot \frac{\partial \bar{w}}{\partial y} = \begin{Bmatrix} L_1 \\ L_2 \end{Bmatrix} \quad (3.3)$$

$$\bar{P} \cdot \frac{\partial \bar{w}}{\partial t} + \bar{Q} \cdot \frac{\partial \bar{w}}{\partial x} + \bar{R} \cdot \frac{\partial \bar{w}}{\partial y} = \bar{L} \quad (3.4)$$

$$\frac{\partial \bar{w}}{\partial t} + \bar{P}^{-1} \cdot \bar{Q} \cdot \frac{\partial \bar{w}}{\partial x} + \bar{P}^{-1} \cdot \bar{R} \cdot \frac{\partial \bar{w}}{\partial y} = \bar{P}^{-1} \cdot \bar{L} \quad (3.5)$$

If $\bar{P}^{-1} \cdot \bar{Q}$ and $\bar{P}^{-1} \cdot \bar{R}$ has got multiple non-zero values in a column, then the two equations are coupled (the equations share derivatives of certain dependent variables).

We can uncouple these equations, because $\bar{P}^{-1} \cdot \bar{Q}$ and $\bar{P}^{-1} \cdot \bar{R}$ are square matrices. The process is called "diagonalizing of matrices" or writing matrices in quadratic form.

$$\frac{\partial \bar{w}}{\partial t} + \bar{T} \cdot \bar{\lambda}_{PQ} \cdot \bar{T}^{-1} \cdot \frac{\partial \bar{w}}{\partial x} + \bar{S} \cdot \bar{\lambda}_{PR} \cdot \bar{S}^{-1} \cdot \frac{\partial \bar{w}}{\partial y} = \bar{P}^{-1} \cdot \bar{L} \quad (3.6)$$

$\bar{\lambda}_{PQ}$ and $\bar{\lambda}_{PR}$ are matrices containing non-zero values only on the main diagonal. These entries correspond to the eigenvalues of $\bar{P}^{-1} \cdot \bar{Q}$ and $\bar{P}^{-1} \cdot \bar{R}$ respectively. Each eigenvalue is a characteristic direction (See figure 3.1). This means $\lambda_{PQ_{ii}} = \frac{dx}{dt}_i$ and $\lambda_{PR_{ii}} = \frac{dy}{dt}_i$, where $i \in [1, 2]$. Matrices T and S consists of columns of eigenvectors. Each eigenvector corresponds to an eigenvalue in $\bar{\lambda}_{PQ}$ or $\bar{\lambda}_{PR}$. We have now uncoupled the continuity and momentum equations, in the characteristic directions. They are still PDE's. By examining the eigenvalues of $\bar{\lambda}_{PQ}$ for the xt-plane or $\bar{\lambda}_{PR}$ for the yt-plane we can classify the system. The system is:

- Hyperbolic if both eigenvalues are real and distinct.
- Parabolic if both eigenvalues are real and equal.
- Elliptic if both eigenvalues are complex.

An eigenvalue analysis on the differential equations for waterhammer (eq. 2.26 and eq. 2.63), reveals that the system is hyperbolic.

$$\bar{\lambda}_{PQ} = \begin{bmatrix} v + a & 0 \\ 0 & v - a \end{bmatrix} \quad (3.7)$$

Furthermore, if we know what the eigenvalues are, then we are able to derive the characteristic and compatibility equations. This is presented next.

Obtaining the characteristic equations from the eigenvalue method

In the previous section we mentioned that it is possible to transform a PDE into an ODE, subject to the condition that this ODE (also known as a compatibility equation) is only defined when used with its corresponding characteristic equation. These characteristic equations represent lines in xt or yt-space. They are obtained by integrating the characteristic directions:

$$\frac{dx}{dt} = \lambda_{PQ_{ii}} \quad (3.8)$$

$$dx = \lambda_{PQ_{ii}} dt \quad (3.9)$$

$$x = \int \lambda_{PQ_{ii}} dt + x_0 \quad (3.10)$$

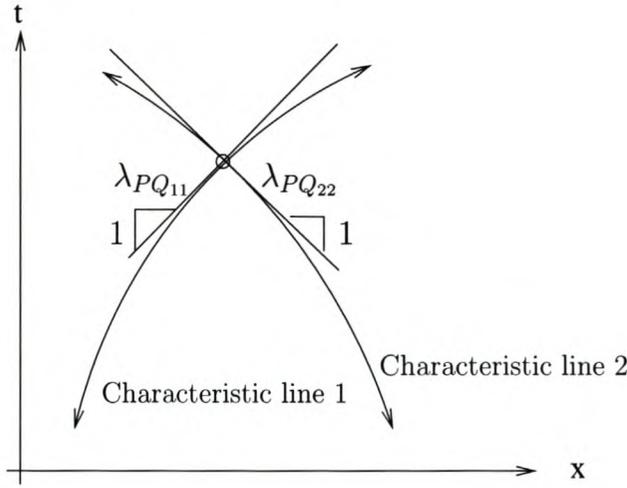


Figure 3.1: *Characteristic directions and Characteristic lines for a hyperbolic problem*

Equation 3.8 is a general characteristic equation in xt -space. The characteristic equations for our pipe-flow PDE's are

$$x = \int (v + a) dt + x_0 \quad (3.11)$$

$$x = \int (v - a) dt + x_0 \quad (3.12)$$

Also note that the number of characteristic equations, which a system has is equal to the number of equations appearing in the system.

Obtaining the compatibility equations from the eigenvalue method

The PDE set (eq. 3.6) can be transformed into an ODE set (for the xt -plane) as follows: Premultiplication by \bar{T}^{-1} gives

$$\bar{T}^{-1} \cdot \left\{ \frac{\partial \bar{w}}{\partial t} + \bar{T} \cdot \bar{\lambda}_{PQ} \cdot \bar{T}^{-1} \cdot \frac{\partial \bar{w}}{\partial x} + \bar{S} \cdot \bar{\lambda}_{PR} \cdot \bar{S}^{-1} \cdot \frac{\partial \bar{w}}{\partial y} \right\} = \bar{T}^{-1} \cdot \bar{P}^{-1} \cdot \bar{L} \quad (3.13)$$

Substituting $\bar{O} = \bar{T}^{-1} \cdot \bar{w}$ in the previous equation set gives

$$\frac{\partial \bar{O}}{\partial t} + \bar{\lambda}_{PQ} \cdot \frac{\partial \bar{O}}{\partial x} + \bar{T}^{-1} \cdot \bar{S} \cdot \bar{\lambda}_{PR} \cdot \bar{S}^{-1} \cdot \bar{T} \cdot \frac{\partial \bar{O}}{\partial y} = \bar{T}^{-1} \cdot \bar{P}^{-1} \cdot \bar{L} \quad (3.14)$$

Vector \bar{O} contains the characteristic variables; Riemann variables or Riemann invariants. The equation set (eq. 3.14) can be written as an ODE set by using the definition of "total derivatives".

Substituting $\lambda_{PQ_{ii}} = \frac{dx}{dt}_i$ in eq. 3.14 and temporarily freezing all y derivatives, by setting

y equal to a constant, gives for xt -space

$$\frac{\partial \bar{O}}{\partial t} + \frac{dx}{dt} \frac{\partial \bar{O}}{\partial x} = \bar{T}^{-1} \cdot \bar{P}^{-1} \cdot \bar{L} \quad (3.15)$$

$$\frac{d\bar{O}}{dt} = \bar{T}^{-1} \cdot \bar{P}^{-1} \cdot \bar{L} \quad (3.16)$$

Equation set 3.16 is known as the compatibility equations. It is equivalent to the PDE set except that they are only valid on the characteristic lines in xt -space. The solution for the governing equations can therefore be found by solving the compatibility equations on the domain defined by the characteristic equations. The compatibility equations for our pipe-flow PDE's are given by

$$\frac{g}{a} \frac{dh}{dt} + \frac{dv}{dt} - \frac{g}{a} v \sin(\alpha) + \frac{\lambda|v|v}{2D} = 0 \quad (3.17)$$

$$-\frac{g}{a} \frac{dh}{dt} + \frac{dv}{dt} + \frac{g}{a} v \sin(\alpha) + \frac{\lambda|v|v}{2D} = 0 \quad (3.18)$$

3.2.4 Classification of a single first-order PDE

The eigenvalue method, presented in the previous section, can also be used to determine the characteristic and compatibility equation for a single first-order PDE (single first-order PDE's with two independent variables are of the hyperbolic type). Consider the first order PDE

$$a \frac{\partial u}{\partial t} + b \frac{\partial u}{\partial x} + c \frac{\partial u}{\partial y} = L \quad (3.19)$$

Divide the equation by (a)

$$\frac{\partial u}{\partial t} + \frac{b}{a} \frac{\partial u}{\partial x} + \frac{c}{a} \frac{\partial u}{\partial y} = \frac{L}{a} \quad (3.20)$$

The convection matrices ($\bar{P}^{-1} \cdot \bar{Q}$ and $\bar{P}^{-1} \cdot \bar{R}$) in equation 3.5 are represented as convection scalars ($\frac{b}{a}$ and $\frac{c}{a}$) in equation 3.20. These scalars are also equivalent to the eigenvalue matrices ($\bar{\lambda}_{PQ}$ and $\bar{\lambda}_{PR}$) in equation 3.6. It follows therefore, that for xt -space (temporarily freezing all y derivatives by setting y equal to a constant) the characteristic equation is given by

$$x = \int \frac{b}{a} dt + x_0 \quad (3.21)$$

while the compatibility equation is given by

$$\frac{\partial u}{\partial t} + \frac{b}{a} \frac{\partial u}{\partial x} = \frac{L}{a} \quad (3.22)$$

$$\frac{du}{dt} = \frac{L}{a} \quad (3.23)$$

Now that we know how to classify a PDE and derive its characteristic and compatibility equations, it is appropriate to examine what makes each PDE group unique. This is discussed in the following three sections.

3.3 Hyperbolic PDE's

In fluid dynamics many flows are governed by equation systems that are hyperbolic in character (e.g. low-friction unsteady convective flows). These systems have very little or no dissipation. This means solutions are characterised by wave-trains that propagate with little or no loss in amplitude. Consider the hyperbolic equation:

$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = 0 \quad (3.24)$$

where u is the flow velocity. The boundary conditions are

$$u(x, 0) = \sin(\pi x) \quad (3.25)$$

$$u(0, t) = u(1, t) = 0 \quad (3.26)$$

$$\frac{\partial u}{\partial t}(x, 0) = 0 \quad (3.27)$$

The analytic solution is

$$u(x, t) = \sin(\pi x) \cos(\pi t) \quad (3.28)$$

By examining the last factor of the solution, it is clear that this hyperbolic equation

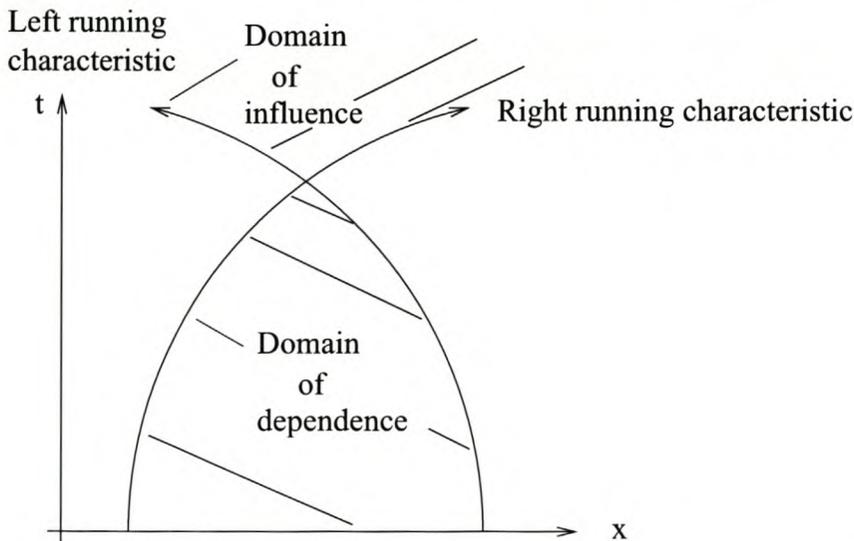


Figure 3.2: Domain of dependence and influence for a hyperbolic PDE

has no dissipation with respect to time (the solution exhibits a simple harmonic motion). This implies that if the initial data (or boundary conditions) contain discontinuities, they will be transmitted into the interior along the characteristic lines, without attenuation of the discontinuity for linear equations.

Numerical schemes, in general, have trouble accomodating such discontinuities, without introducing spurious oscillations. Numerical methods are usually designed to introduce small amounts of artificial dissipation to avoid these spurious oscillations. However, if a scheme introduces too much artificial dissipation, then the wave-amplitude of the true solution gets damped unnecessarily and the speed at which the wave propagates is adversely affected.

Another feature of hyperbolic equations is that derivatives normal to the characteristic directions are discontinuous, while those along the characteristic directions are continuous. The reason for this is that information can only be transmitted along the characteristic lines, not across them. If a disturbance is created at a point in the domain, then at that moment in time, no other point will experience its effect. It is only through the characteristic lines that the disturbance is communicated to the other points in the domain.

3.4 Parabolic PDE's

Parabolic PDE's are propagation problems, which include dissipative mechanisms, such as heat conduction or pollutant concentrations diffusing in water. They are typified by solutions which march forward in time, but diffuses from its origin in space. In contrast to hyperbolic equations, their derivatives normal to the characteristic direction are continuous. For steady two dimensional boundary layer flow, the characteristics are normal to the flow direction. Therefore a disturbance introduced at any point within or on the boundary of the domain will influence any part of the domain for $t \geq t_i$. However, the magnitude of the disturbance quickly attenuates in moving away from its origin. Examine the parabolic equation:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad (3.29)$$

with boundary conditions:

$$u(x, 0) = \sin(\pi x) \quad (3.30)$$

$$u(0, t) = u(1, t) = 0 \quad (3.31)$$

and analytic solution:

$$u(x, t) = \sin(\pi x) e^{-\pi^2 t} \quad (3.32)$$

We can see that a disturbance will get damped out if sufficient time is given. If we want our domain to contain disturbances all the time, then we would need to introduce these disturbances continuously through the boundary conditions.

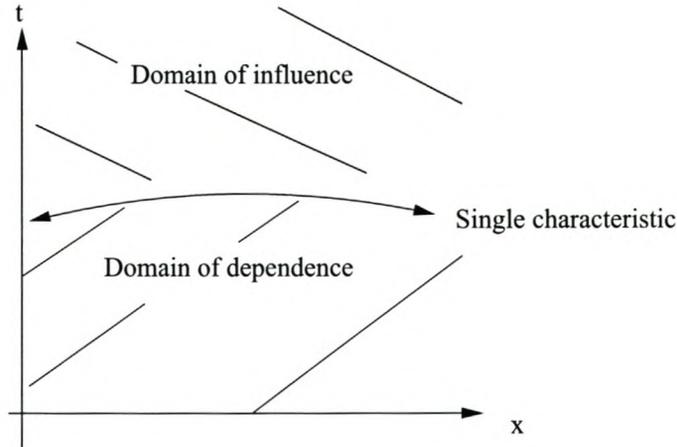


Figure 3.3: *Domain of dependence and influence for a parabolic PDE*

3.5 Elliptic PDE's

Elliptic equations describe steady-state problems. This means that they are independent of time. The most important feature concerning elliptic PDE's is that a disturbance introduced at an interior point, influences all other points in the computational domain, although away from its origin, its influence will be small. This implies that in seeking computational solutions to elliptic problems, it is necessary to consider the whole domain. Examine an elliptic equation:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (3.33)$$

with boundary conditions:

$$u(x, 0) = \sin(\pi x) \quad (3.34)$$

$$u(x, 1) = \sin(\pi x) e^{-\pi} \quad (3.35)$$

$$u(0, y) = u(1, y) = 0 \quad (3.36)$$

and analytic solution:

$$u(x, y) = \sin(\pi x) e^{-\pi y} \quad (3.37)$$

By examining the last factor in the solution, it can be seen that discontinuities are smoothed out in the interior.

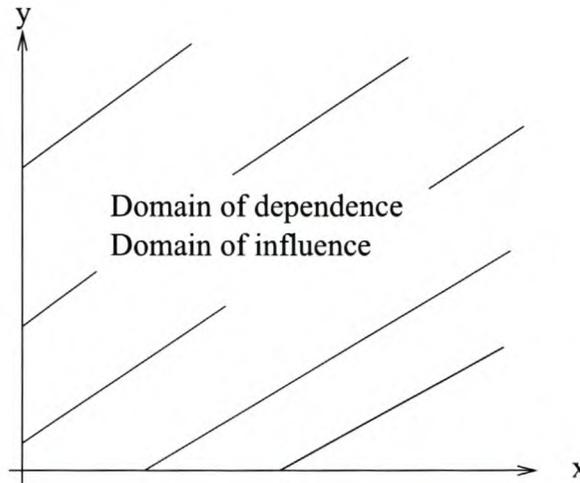


Figure 3.4: *Domain of dependence and influence for an elliptic PDE*

3.6 Conservation and Non-Conservation forms of the governing equations

We have mentioned in sections 2.4.1, 2.4.2 that the Lagrangian perspective produces equations in non-conservation form, while the Eulerian perspective produces equations in conservation form. This section explains how to identify each of these forms. It discusses the different properties of the two forms and shows that the **primitive variables** (h, v, ρ) are used exclusively by the non-conservation form, while the **flux variables** (mass, momentum, energy) are used by the conservation form. The distinction between primitive and flux variables are important, because primitive variables can have discontinuities within their solutions (due to shock fronts), but not the flux variables (the conservation laws imposes continuity of mass, momentum and energy). Numerical schemes often produce spurious oscillations due to sharp gradients or discontinuities in the solution. Unfortunately, the governing equations for unsteady pipe-flow is only available in non-conservation form. The numerical scheme employed to solve these equations, will therefore need to be robust enough to handle sharp gradients and discontinuities.

3.6.1 Conservation form

The following equation is an example of a differential equation written in Conservation form

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} = 0 \quad (3.38)$$

or in compact tensor notation

$$\frac{\partial \rho}{\partial t} + \bar{\nabla} \cdot \rho \bar{v} = 0 \quad (3.39)$$

Notice that the **flux variables** ($\rho = \text{mass}$ and $\rho u, \rho v, \rho w = \text{momentum}$) appear only inside the derivative terms and not as coefficients for other derivative terms. The rule that identifies the above equation as being a conservation form is: **Coefficients of the derivative terms are either constant or, if variable, their derivatives appear nowhere in the equation.** Equations written in conservation form are convenient to manipulate numerically and to program, because they have a generic form:

$$\frac{\partial \bar{U}}{\partial t} + \frac{\partial \bar{F}}{\partial x} + \frac{\partial \bar{G}}{\partial y} + \frac{\partial \bar{H}}{\partial z} = \bar{L} \quad (3.40)$$

When more than one equation (each having the above form) are combined in a single system, then $\bar{U}, \bar{F}, \bar{G}, \bar{H}$ and \bar{L} are vectors. For example inviscid flow has the following vectors:

$$\bar{U} = \left\{ \begin{array}{c} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho \left(e + \frac{v^2}{2} \right) \end{array} \right\} \quad (3.41)$$

$$\bar{F} = \left\{ \begin{array}{c} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ \rho u \left(e + \frac{v^2}{2} \right) + pu \end{array} \right\} \quad (3.42)$$

$$\bar{L} = \left\{ \begin{array}{c} 0 \\ \rho f_x \\ \rho f_y \\ \rho f_z \\ \rho (u f_x + v f_y + w f_z) + \rho \dot{q} \end{array} \right\} \quad (3.43)$$

\bar{U} is the solution vector, whose elements contain the **flux variables**, and \bar{L} is the source/sink vector. \bar{F}, \bar{G} and \bar{H} are flux-vectors along each of the cartesian axes, and whose elements are **functions of the flux variables**. The first element in any of the above vectors, corresponds to the conservation of mass equation. The second, third and

fourth elements, correspond to the momentum equations in the x, y, z -directions. The last element corresponds to the conservation of energy equation. Equation 3.40 can be written in compact tensor notation as

$$\frac{\partial \bar{U}}{\partial t} + \bar{\nabla} \cdot \bar{f}(\bar{U}) = \bar{L} \quad (3.44)$$

\bar{G} and \bar{H} are the equivalents of \bar{F} for the y and z -directions. When the control volume is fixed in space (Eulerian perspective), we are concerned with the flux of mass, momentum and energy into and out of the volume. The aim therefore is to solve for vector \bar{U} , which contains the flux variables. The primitive variables (p, ρ, v) are solved after \bar{U} has been found, by using their relationships with the flux variables.

For unsteady flows the conservation form can easily be used in a time marching scheme (provided there are boundary conditions to guide the solution), by writing:

$$\frac{\partial \bar{U}}{\partial t} = \bar{L} - \frac{\partial \bar{F}}{\partial x} - \frac{\partial \bar{G}}{\partial y} - \frac{\partial \bar{H}}{\partial z} \quad (3.45)$$

For steady flows the conservation form can also be used in a spatial marching scheme (provided there are boundary conditions to guide the solution), by writing:

$$\frac{\partial \bar{F}}{\partial x} = \bar{L} - \frac{\partial \bar{G}}{\partial y} - \frac{\partial \bar{H}}{\partial z} \quad (3.46)$$

3.6.2 Non-Conservation form

As mentioned previously, these equations are obtained from the Lagrangian perspective, which is based on a control body of moving fluid. It is because of this movement that total derivatives appear within these equations:

$$\frac{du}{dt} = \frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} \frac{\partial x}{\partial t} \quad (3.47)$$

where $\frac{\partial x}{\partial t}$ is the velocity of the control body.

The one dimensional heat conduction equation is an example of a differential equation in non-conservation form

$$\rho c \frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2} + \frac{\partial k}{\partial x} \frac{\partial T}{\partial x} \quad (3.48)$$

where

- T is temperature
- k is the thermal conductivity
- ρ is the substance density
- c is the specific heat constant

The rule given in the previous section indicates that this is a non-conservation form, since the coefficient k of a derivative term and the gradient of k appears in the same equation.

Another important property is that the spatial and time derivatives operate on the primitive variables. As will be seen later, this introduces problems when implementing a shock-capturing model.

3.6.3 Sub-classes of the Conservation form

Equations written in conservation form can be classified further into a strong formulation and a weak formulation. A strong formulation has no spatial derivatives outside the divergence and unsteady flow terms. e.g.

$$\frac{\partial \bar{U}}{\partial t} + \frac{\partial \bar{F}}{\partial x} + \frac{\partial \bar{G}}{\partial y} + \frac{\partial \bar{H}}{\partial z} = \bar{L} \quad (3.49)$$

A weak formulation has a number of spatial derivatives outside the divergence and unsteady flow terms. e.g.

$$\frac{\partial(\rho u)}{\partial t} + \bar{\nabla} \cdot (\rho u \bar{V}) = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial x} + \frac{\partial \tau_{zx}}{\partial x} + \rho f_x \quad (3.50)$$

The difference between weak and strong form, becomes important when supersonic steady state flows are examined. In this case the temporal derivative would fall away, leaving only spatial derivatives and source/sink terms. Because the strong form does not have multiple spatial derivatives for a single direction, it is possible to implement a spatial marching scheme on it. It is however not possible to implement a spatial marching scheme on the weak form, because of its multiple spatial derivative in one direction (e.g. the x -direction)

3.7 Jacobians of the Flux Vectors

The eigenvalue method is used to classify quasi-linear PDE's. By quasi-linear we mean that the coefficients of the derivative terms (which contain dependent variables) are either constant or dependent variables, but not derivatives of the dependent variables themselves. For example: $(v \frac{\partial v}{\partial x}$ or $2 \frac{\partial v}{\partial x}$), but not $(\frac{\partial v}{\partial x} \frac{\partial v}{\partial x})$. Equations in non-conservation form are already in quasi-linear form, but not so for equations in conservation form. This section shows how to transpose an equation in conservation form into a quasi-linear form, which is required for the eigenvalue method. Examine

$$\frac{\partial \bar{U}}{\partial t} + \frac{\partial \bar{F}}{\partial x} + \frac{\partial \bar{G}}{\partial y} + \frac{\partial \bar{H}}{\partial z} = \bar{L} \quad (3.51)$$

where the flux vectors are non-linear functions of \bar{U} ; $\bar{F}(\bar{U})$, $\bar{G}(\bar{U})$ and $\bar{H}(\bar{U})$. We can expand the equation to

$$\frac{\partial \bar{U}}{\partial t} + \bar{A} \cdot \frac{\partial \bar{U}}{\partial x} + \bar{B} \cdot \frac{\partial \bar{U}}{\partial y} + \bar{C} \cdot \frac{\partial \bar{U}}{\partial z} = \bar{L} \quad (3.52)$$

where

$$\bar{\bar{A}} \equiv \frac{\partial \bar{F}}{\partial \bar{U}}, \bar{\bar{B}} \equiv \frac{\partial \bar{G}}{\partial \bar{U}}, \bar{\bar{C}} \equiv \frac{\partial \bar{H}}{\partial \bar{U}} \quad (3.53)$$

$\bar{\bar{A}}$, $\bar{\bar{B}}$ and $\bar{\bar{C}}$ are called the jacobians of the flux vectors. The derivatives of the dependent variables (the derivatives containing \bar{U}) now appear linearly, hence the equation is in quasi-linear form. We can now classify the PDE using the eigenvalue method. The PDE's behaviour is dictated by the eigenvalues of the jacobian matrices $\bar{\bar{A}}$, $\bar{\bar{B}}$ and $\bar{\bar{C}}$. The inverse of these eigenvalues gives the slopes of the characteristic lines in xt-space, yt-space, zt-space.

3.8 Shock Capturing and Shock Fitting

The effect of shock waves on the flow fields can be determined through two methods: Shock capturing and Shock fitting.

Shock capturing (fig. 3.5) is used when we do not know the position and form of the shock wave. A large computational domain, with appropriate boundary conditions, is defined. The flow fields on this large domain is solved and the location and form of the shock wave is extracted from the solution. This approach works well for equations in conservation form (whose derivatives operate on flux variables), since the solution of the flux variables are continuous throughout the domain.

However, numerical schemes often produce spurious oscillations in a shock capturing approach for equations in the non-conservation form. The reason is that the derivatives in the non-conservation form equations operate on primitive variables, whose solution is discontinuous across a shock front. Figure 3.7 shows schematically the the flow field solutions of primitive and flux variables.

Shock fitting (fig. 3.6) is used, when we do know the location and form of the shock wave. A smaller domain can be defined between the shock wave and the necessary boundary conditions. The flow fields within this domain can then be solved. This approach works well for equations in conservation form and non-conservation form, since the shock front does not appear within the domain. It exists only on the boundary of the domain and its effect is introduced on the flow fields by way of a boundary condition.

The pipe-flow equations (eq. 2.26 and 2.63) for waterhammer are only available in non-conservation form, due to the substitution of pipe- and water-elasticity equations into the continuity equation. Ideally, we would like to implement a shock fitting approach to avoid

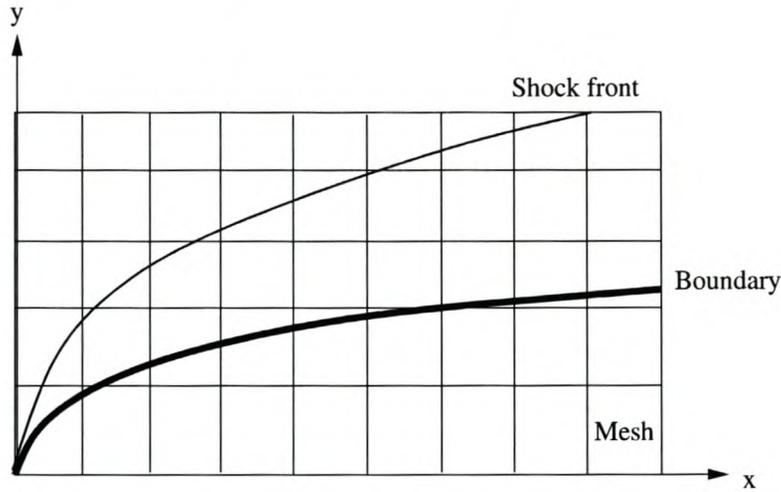


Figure 3.5: *Shock Capturing Approach*

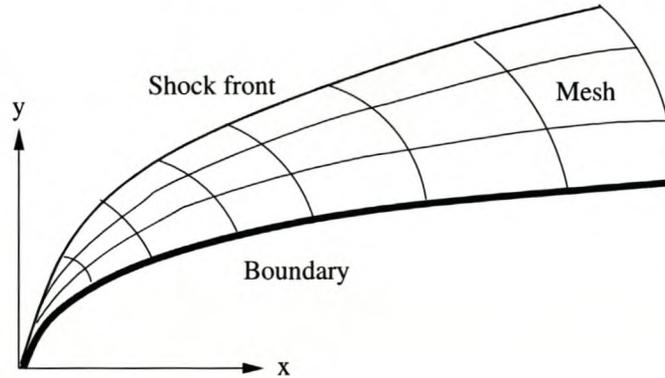


Figure 3.6: *Shock Fitting Approach*

spurious oscillations due to the discontinuous flow fields of the primitive variables. Unfortunately, waterhammer's nature is incompatible with a shock fitting approach, since multiple shock waves may exist in a pipe network, each one *moving in different directions*. We are therefore forced to use a shock capturing approach, with equations in non-conservation form. The challenge is therefore, to find a numerical scheme which will be able to selectively remove or damp spurious oscillations.

3.9 Convection, Diffusion and the numerical difficulties that they introduce

This section discusses convective and diffusive transport, as well as how these two types of phenomena are described mathematically. It also mentions that numerical schemes

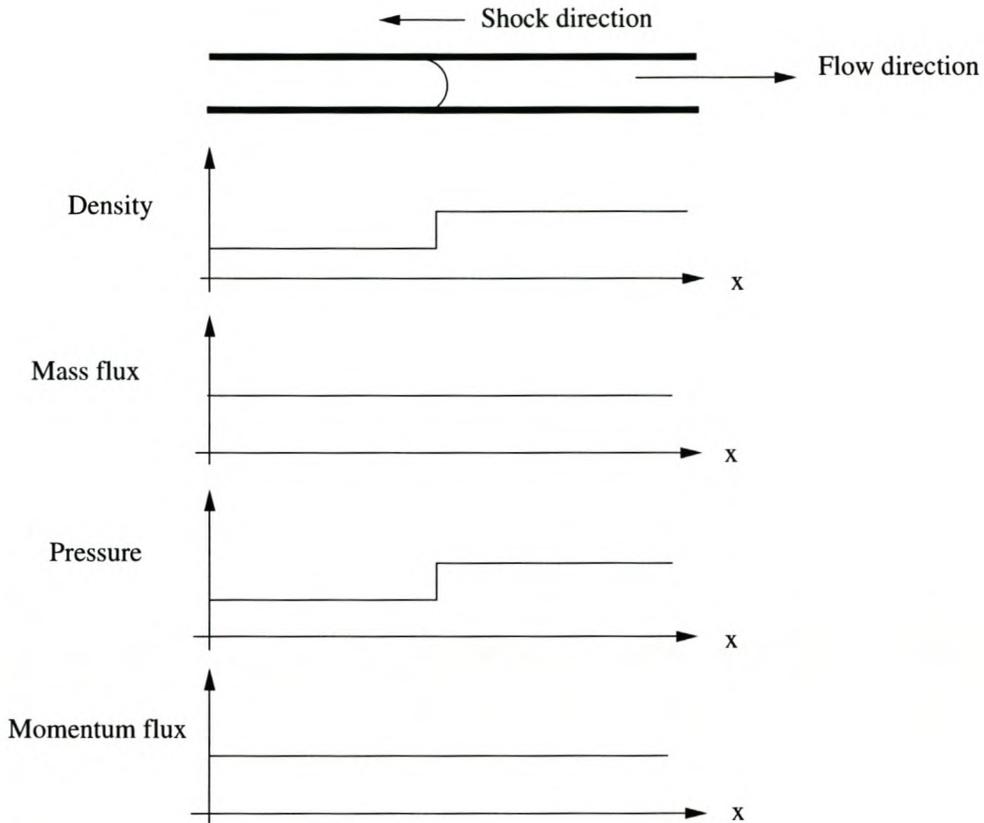


Figure 3.7: *Spatial Variation of Flux and Primitive Variables across a Shock Front*

suffer from dispersive errors due to the odd-ordered derivatives introduced by convective terms, while dissipative errors are introduced by even-ordered derivatives introduced by diffusive terms.

3.9.1 Convection (Macroscopic transport)

Convection is the transport of a property, because of its entrainment within a moving fluid. The properties could be mass, momentum or temperature. The convective flux of a property is defined as the product of the property density and the volumetric flux of the moving fluid. Where

- **Property density** is defined as the amount of property per unit volume
- **Property flux** is defined as the amount of property per unit time.

Table 3.1 shows the different properties that can be convected, their densities and their fluxes. Where:

- ρ is the mass density of the property

- v is the flow velocity
- Q is the volumetric flux
- A is the cross-sectional area of the stream tube

Table 3.1: *Different properties that can be convected, their densities and fluxes*

| \tilde{a} (property) | $\tilde{\rho}$ (property density) | \tilde{q} (convective property flux) |
|------------------------|-----------------------------------|--|
| Mass | ρ | ρQ or $\rho v A$ |
| Momentum | ρv | $\rho v Q$ or $\rho v v A$ |

An important mathematical property of convective fluxes is that they are described by a zero'th order derivative (Notice that there are no derivatives in the property flux column of table 3.1). At the end of this section we will see that these zero'th order derivatives introduces dispersion errors in our numerical solution.

3.9.2 Diffusion (Microscopic transport)

Diffusion occurs when macroscopic nonuniformity in a property \tilde{a} (described by gradients of \tilde{a}), triggers transport at the molecular level e.g. viscosity, diffusion or heat conduction. This diffusive movement is described mathematically as follows (using the same symbols defined in the previous section):

Imagine that we have a property being convected at two points ($x = +\frac{\lambda}{2}$ and $x = -\frac{\lambda}{2}$) at the same speed (v), but in opposite directions. The flux per unit surface of the two points are:

$$\phi_{+\frac{\lambda}{2}} = \tilde{\rho}_{+\frac{\lambda}{2}} Q \quad (3.54)$$

$$\phi_{-\frac{\lambda}{2}} = \tilde{\rho}_{-\frac{\lambda}{2}} Q \quad (3.55)$$

Hence the total diffusive property flux at $x = +\frac{\lambda}{2}$ and $x = -\frac{\lambda}{2}$ per unit surface is:

$$\phi_{+\frac{\lambda}{2}, -\frac{\lambda}{2}} = \phi_{+\frac{\lambda}{2}} - \phi_{-\frac{\lambda}{2}} = \left(\tilde{\rho}_{+\frac{\lambda}{2}} - \tilde{\rho}_{-\frac{\lambda}{2}} \right) Q \quad (3.56)$$

The total diffusive property flux at the origin ($x = 0$) is obtained by substituting a linearised Taylor series, whose base is at the origin, for $\tilde{\rho}_{+\frac{\lambda}{2}}$ and $\tilde{\rho}_{-\frac{\lambda}{2}}$.

$$\phi_{+\frac{\lambda}{2}, -\frac{\lambda}{2}} = \left(\tilde{\rho}_{+\frac{\lambda}{2}} - \tilde{\rho}_{-\frac{\lambda}{2}} \right) Q \quad (3.57)$$

$$\phi_0 = \left(\tilde{\rho}_0 + \frac{\partial \tilde{\rho}_0}{\partial x} \left(\frac{\lambda}{2} \right) - \tilde{\rho}_0 - \frac{\partial \tilde{\rho}_0}{\partial x} \left(-\frac{\lambda}{2} \right) \right) Q \quad (3.58)$$

$$\phi_0 = \frac{\partial \tilde{\rho}_0}{\partial x} \lambda Q \quad (3.59)$$

or simply

$$\phi = K \frac{\partial \bar{\rho}}{\partial x} \quad (3.60)$$

where K is a constant.

The point of this derivation was to show that diffusion as a flux is described by a first-order derivative. At the end of this section we will see that these first order derivatives introduces dissipation errors in our numerical solution. Examples of microscopic transport are:

Stress

$$\bar{\tau} = \mu \bar{\nabla} \bar{u} \quad (3.61)$$

where

- τ is stress
- μ is elastic modulus
- u is displacement

Heat conduction

$$\bar{q}'' = -k \bar{\nabla} T \quad (3.62)$$

where

- q is heat flux
- k is thermal conductivity
- T is temperature

3.9.3 The form of convective and diffusive fluxes in differential equations

To describe the **change** in flux from one side of an infinitesimal control volume to the other, we can use a linearised Taylor series expansion. This net change usually reduces to the gradient of the flux multiplied with the distance between the two sides of the control volume e.g.

$$\text{Net}tp = \left(p + \frac{dp}{dx} dx\right) - p \quad (3.63)$$

$$\text{Net}tp = \frac{dp}{dx} dx \quad (3.64)$$

where

- $p = \bar{\rho}Q$ for convective fluxes
- $p = K \frac{\partial \bar{p}}{\partial x}$ for diffusive fluxes

Summarised: *The convective and diffusive flux terms have the order of their derivatives increased by one, when used in a differential equation.* By deriving our equations from small control bodies/volumes and using Taylor expansions, we are now able to interpret the physical meaning/origin of the flux terms in our differential equations. Examples are:

- $\bar{\nabla} \cdot (\rho \bar{v})$ is the nett change in mass flux per unit length due to convection.
- $\bar{\nabla} \cdot (\rho \bar{v} \bar{v})$ is the nett change in momentum flux per unit length due to convection.
- $-\nu \bar{\nabla} \cdot \bar{\nabla} \bar{v} = -\nu \frac{\partial^2 \bar{v}}{\partial x^2}$ is the nett velocity flux per unit length due to viscous diffusion.

It was mentioned in this section that the two types of transport phenomena are described by derivatives, having respectively different orders. These different orders introduces numerical diffusion errors in the solution. **Numerical diffusion** is the combined effect of dissipation and dispersion. **Dissipation** reduces all gradients in the solution, whether physically correct or numerically induced. It is the direct result of even-ordered derivative terms. **Dispersion** distorts the phase relation between various waves (Our solution can be decomposed into the components of a Fourier series). Dispersion is the direct result of odd-ordered derivatives. This thesis studies convection dominated flows, which means that the odd-ordered derivatives, in our differential equations, will introduce dispersive errors into our solution.

3.9.4 Eulerian, Lagrangian and Eulerian-Lagrangian schemes

It is appropriate at this point to mention the different numerical schemes, which can be used to solve the unsteady pipe-flow differential equations, within the finite element context, before we show in the next chapter how the equations are solved. There are three broad groups of schemes for solving hyperbolic PDE's [8]: Eulerian, Lagrangian and Eulerian-Lagrangian schemes (These schemes should not be confused with the Eulerian and Lagrangian perspectives).

Eulerian schemes

Eulerian schemes are localised schemes, which are based on a mesh/grid that does not change with time. The spatial derivatives are approximated using the information at the neighbouring nodes. They account for the hyperbolic character of the PDE (usually by up-winding) and also introduce dissipation terms to eliminate spurious oscillations. The disadvantages of these schemes are that, for either stability (for explicit time-stepping cases)

or accuracy (for implicit time-stepping cases) reasons, the time-step size is limited by some form of the Courant-Friedrichs-Levy (CFL) condition. Common Eulerian schemes are: streamline-upwind Petrov-Galerkin (SUPG), Galerkin/least-squares and Taylor-Galerkin schemes. This thesis implements the streamline-upwind Petrov-Galerkin scheme.

Lagrangian schemes

Lagrangian schemes are based on computational meshes, which move along the characteristic lines of the hyperbolic PDE (i.e. the mesh changes with time). These schemes are theoretically well suited for advection problems, but practically difficult to implement, because the mesh can become easily distorted, due to the dilatation of the fluid particles.

Eulerian-Lagrangian schemes

Eulerian-Lagrangian schemes uses a Lagrangian tracking algorithm along the characteristic lines, while computations are based on a fixed "Eulerian" mesh/grid. Nodes from the grid are tracked backwards (or forwards) along the characteristics over the time-step. Numerical information is then projected from the previous time-step on the Eulerian mesh/grid, onto the "Lagrangian" mesh/grid. The advantage of this scheme is that the CFL-condition in terms of stability/accuracy is significantly relaxed.

Chapter 4

Discretizing the Pipe-Flow Equations

4.1 Introduction

This chapter focusses on

- Why the Bubnov-Galerkin finite element is unable to solve the unsteady pipe-flow equations.
- Which properties of the Petrov-Galerkin finite element makes it succesful in solving the pipe-flow equations and why.
- Discretizing the pipe-flow equations using the Petrov-Galerkin finite element method.
- The algorithm used to solve the resulting non-linear set of equations

The chapter starts by showing that the Bubnov-Galerkin finite element poorly approximates the odd-ordered derivatives of our pipe-flow equations in a shock capturing approach, because the equations are in a non-conservation form. The reason why Bubnov-Galerkin schemes perform poorly for equations in non-conservation form is because their trial and test functions are identical. This results in sampling data from the wrong domain of dependence. The chapter then presents upwinding as a method to sample the correct domain of dependence.

It also mentions that the continuous test functions implemented by the Bubnov-Galerkin scheme leads to a globally coupled system of equations. This means that a disturbance at a point within the domain, influences the whole domain immediately, which is natural for elliptic and parabolic differential equations, but not for hyperbolic PDE's. Discontinuous test functions are more appropriate for hyperbolic PDE's.

An open-channel flow example, which contains a hydraulic jump, is used to illustrate the results obtained for the Bubnov-Galerkin finite element. Various remedial measures

are applied to the poor Bubnov-Galerkin finite element and the "improved" results are discussed.

The last part of the chapter shows the derivation of a Characteristic Dissipative Galerkin finite element (a special Petrov-Galerkin element) for the unsteady pipe-flow equations. This finite element produces a non-linear set of equations. Finally, the algorithm used to solve these non-linear equations is discussed.

4.2 Why the Bubnov-Galerkin finite element is unsuitable

Bubnov-Galerkin schemes (schemes where the trial functions are identical to the test functions) applied to equations in non-conservation form, encounter difficulties when approximating derivatives in areas where the solution is discontinuous. Parasitic oscillations often appear near shock wave fronts, which reduces accuracy at those locations and may even introduce instability.

In section 3.2.3 it was shown how the characteristic lines were obtained from an eigenvalue analysis. We saw that the unsteady pipe-flow equations were hyperbolic and that the domain of dependence for a point in xt -space is defined by the characteristic lines. Furthermore, the characteristic lines shows us the speed and directions in which information is propagated throughout the flow field. It would seem natural that a numerical scheme, that wants to solve these equations, should be consistent with the speed and direction in which information propagates throughout the flow field. It should not draw numerical information from outside the domain of dependence of a given grid point; this would compromise the accuracy of the solution.

Part of the reason why the Bubnov-Galerkin finite element performs poorly for hyperbolic problems is that it uses the wrong domain of dependence for calculating the flow field values at a node. More specifically, it is the domain of dependence of the odd-ordered derivatives, appearing in our pipe-flow equations, which the Bubnov-Galerkin scheme misrepresents.

4.2.1 How Bubnov-Galerkin approximates derivatives in equations in the non-conservation form

This section shows how the Bubnov-Galerkin scheme represents odd- and even-ordered derivatives, as well as what the influence of the misrepresented odd-ordered derivatives is

on the global system of equations.

Let us consider a differential equation containing an odd-ordered derivative, as well as an even-ordered derivative; such as the viscous Burger's equation:

$$\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial x} - \nu \frac{\partial^2 T}{\partial x^2} = 0 \quad (4.1)$$

where

- T is temperature
- u is the convection velocity
- ν is the coefficient of viscosity

When odd-ordered derivatives (e.g. $u \frac{\partial T}{\partial x}$) are evaluated on a uniform grid, the Bubnov-Galerkin method produces algebraic formulas, whose coefficients are zero for the dof (degree of freedom) at which the function is centred. In other words, there is zero contribution from the odd-ordered derivatives to the global coefficient matrix's diagonal. This is not the case for even-ordered derivatives. Therefore entries on the diagonal only have contributions coming from the diffusive terms (e.g. $\nu \frac{\partial^2 T}{\partial x^2}$). This is easily proven.

Bubnov-Galerkin applied to odd-ordered derivatives

Let us apply the Bubnov-Galerkin scheme on a simple differential equation, consisting only of an odd-ordered derivative

$$u \frac{\partial T}{\partial x} = 0 \quad (4.2)$$

Define a two-node element. Linear shape functions are used to keep things as simple as possible. The shape functions are:

$$\bar{N} = \begin{bmatrix} \frac{1-\xi}{2} \\ \frac{1+\xi}{2} \end{bmatrix} \quad \xi \in [-1, 1] \quad (4.3)$$

$$(4.4)$$

Applying Bubnov-Galerkin

$$\int_{-1}^1 \bar{N} \cdot R d\xi = 0 \quad (4.5)$$

with R the residual, the result is

$$u \int_{-1}^1 \bar{N} \frac{\partial \bar{N}^T}{\partial \xi} d\xi \cdot \bar{T} = 0 \quad (4.6)$$

Evaluation of the integral produces the element coefficient matrix

$$\bar{K} = \frac{u}{2} \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \quad (4.7)$$

Assembly of element coefficient matrices, produces the global coefficient matrix (only a part shown below)

$$\frac{u}{2} \begin{bmatrix} -1 & 1 & 0 & 0 & \dots \\ -1 & 0 & 1 & 0 & \dots \\ 0 & -1 & 0 & 1 & \dots \\ 0 & 0 & -1 & 0 & 1 \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix} \quad (4.8)$$

If we ignore the first and last rows of the global coefficient matrix, we can see that a single equation has the general form

$$-\frac{u}{2}T_{j-1} + 0T_j + \frac{u}{2}T_{j+1} \quad (4.9)$$

where T_j is the dof at which the function is centred.

Bubnov-Galerkin applied to even-ordered derivatives

For even-ordered derivatives there are contributions to the coefficient matrix's main diagonal. This is also easily proven. Examine the differential equation having only an even-ordered derivative.

$$\nu \frac{\partial^2 T}{\partial x^2} = 0 \quad (4.10)$$

Applying Bubnov-Galerkin and integrating by parts once (with the same set of shape functions which were used for the previous proof), produces

$$\frac{2\nu}{\Delta x} \left\{ \int_{-1}^1 \frac{\partial \bar{N}}{\partial \xi} \frac{\partial \bar{N}^T}{\partial \xi} d\xi + \left[\bar{N} \frac{\partial \bar{N}}{\partial \xi} \right]_{-1}^1 \right\} \cdot \bar{T} \quad (4.11)$$

Integration and assembly of element stiffness matrices produces the general equation

$$\frac{\nu}{\Delta x} (-T_{j-1} + 2T_j - T_{j+1}) \quad (4.12)$$

4.2.2 Why upwinding is used to approximate odd-ordered derivatives in equations in non-conservation form

We have seen in subsection 3.9.3 that convection is described in a differential equation by odd-ordered derivatives. If equation 4.1 is dominated by convection (ν small i.e. Reynolds high), then the matrix is not diagonally dominant and the Bubnov-Galerkin solution is often oscillatory. Even with artificial viscosity (therefore increasing the weight of the diagonal), the results still show some oscillations, although much smaller.

Upwind schemes are designed to numerically simulate more properly the direction of propagation of information in a flow field along the characteristic lines. This technique avoids zero coefficients associated with the odd-ordered derivative for the dof at which the function is centred.

This requires test shape functions which are directionally biased at the dof where the function is centred (It accounts for the direction of propagation). Therefore if the shock front is at the dof being evaluated, the shape function weighs the information on the one side of the dof heavier than on the other side of it. Figure 4.1 shows the different trial and test functions for the Bubnov-Galerkin and Petrov-Galerkin schemes on a one-dimensional mesh.

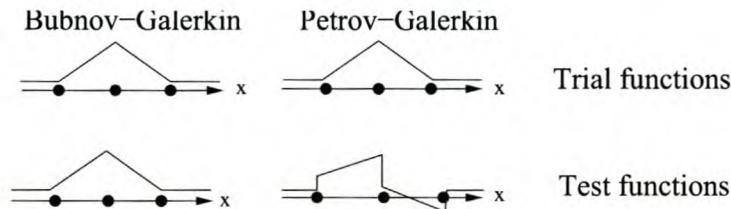


Figure 4.1: *Examples of trial and test functions for Bubnov-Galerkin and Petrov-Galerkin schemes*

4.2.3 Finding the upwind direction through flux-vector splitting

In previous sections we have mentioned that, in order to properly simulate the direction of information propagation we need an upwind scheme. The flux-vector splitting method, is a method with which one can determine the upwind direction for the flow at a node. The Petrov-Galerkin finite element scheme incorporates this method in its discontinuous test functions.

Consider an unsteady convection equation:

$$\frac{\partial \bar{w}}{\partial t} + \bar{A} \frac{\partial \bar{w}}{\partial x} = 0 \quad (4.13)$$

The first term is a local time derivative and will typically be treated by a finite difference discretization. The second term is a convection term. This term contains an odd-ordered derivative and therefore requires some type of upwind finite element scheme. Matrix \bar{A} , in the above equation, usually has multiple non-zero entries in a column. This means that the equations are coupled. By uncoupling the equations, we are also splitting the total flux contribution. The eigenvalue structure of matrix \bar{A} is used to uncouple the system. This is shown below.

$$\frac{\partial w}{\partial t} + T \begin{bmatrix} \lambda_{PQ_1} & 0 \\ 0 & \lambda_{PQ_2} \end{bmatrix} T^{-1} \frac{\partial w}{\partial x} = 0 \quad (4.14)$$

$$\frac{\partial w}{\partial t} + T \begin{bmatrix} \lambda_{PQ_1} & 0 \\ 0 & 0 \end{bmatrix} T^{-1} \frac{\partial w}{\partial x} + T \begin{bmatrix} 0 & 0 \\ 0 & \lambda_{PQ_2} \end{bmatrix} T^{-1} \frac{\partial w}{\partial x} = 0 \quad (4.15)$$

In the above example, we diagonalized matrix \bar{A} . We then expand this eigenvalue matrix into a sum of matrices containing zeros and only one eigenvalue. We now have two convective terms, since our problem is one-dimensional. Figure 4.2 portrays the characteristic directions ($\frac{dt}{dx} = \frac{1}{\lambda}$) for a point on the domain boundary. These characteristic directions tells us which direction is upwind.

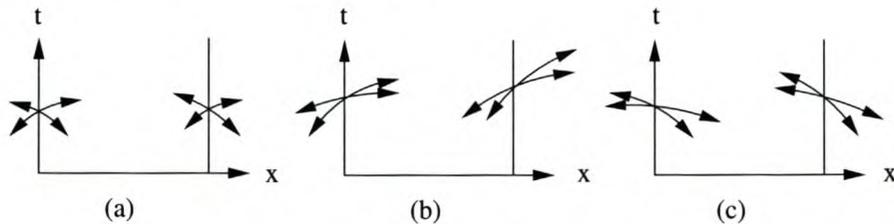


Figure 4.2: *Characteristic Directions for different Flow Types.*

- Figure 4.2 (a) Sub-critical flow: When λ_{PQ_1} and λ_{PQ_2} have different signs, we know that the flow at the point being evaluated is subcritical. Upwind is in *both* the directions of the left and right boundaries.
- Figure 4.2 (b) Super-critical flow downstream: When λ_{PQ_1} and λ_{PQ_2} are both positive, the flow is supercritical in the positive x -direction at the point being evaluated. Upwind is in the direction of the left boundary.
- Figure 4.2 (c) Super-critical flow upstream: When λ_{PQ_1} and λ_{PQ_2} are both negative, the flow is supercritical in the negative x -direction at the point being evaluated. Upwind is in the direction of the right boundary.

4.2.4 Why discontinuous test functions introduces hyperbolic behaviour

The hyperbolic character of our differential equation requires that changes in data at a point only affects the solution locally around that point. The system strives for some degree of uncoupling.

According to Johnson [7], Galerkin methods using continuous trial functions will lead to globally coupled systems of linear equations, i.e. systems where a change in data at one

node will affect the solution at all nodes. This is natural for the elliptic and parabolic problem, but not so for the hyperbolic problem. Discontinuous test functions are therefore required to provide some degree of uncoupling for the global system of linear equations.

4.3 Evolution from a Bubnov-Galerkin to a Petrov-Galerkin FEM

This section describes how a hyperbolic flow problem, containing a discontinuity, is initially solved using the Bubnov-Galerkin scheme. The solution obtained is not realistic. Various improvements are added (some have been discussed in the previous section), to the Bubnov-Galerkin scheme to obtain better results. By adding the improvements one at a time, we can see how the Bubnov-Galerkin scheme, with its poor solution, evolves into a Petrov-Galerkin scheme with a realistic solution.

4.3.1 An open-channel flow problem, containing a hydraulic jump

Imagine that we would like to simulate a hydraulic jump, based on the shallow-water differential equations.

$$\frac{\partial y}{\partial t} + \frac{\partial q}{\partial x} = 0 \quad (4.16)$$

$$\frac{\partial q}{\partial t} + (c^2 - u^2) \frac{\partial y}{\partial x} + 2u \frac{\partial q}{\partial x} = 0 \quad (4.17)$$

where

- q is the flow per unit width
- y is the depth of flow
- u is the flow velocity
- c is the wavespeed in the channel

(Note that these equations have strong ties with the pipe-flow differential equations 2.26 and 2.63). The Bubnov-Galerkin approach would be to minimise the residual with respect to each test function, where the test functions are the same as the trial functions.

$$\langle N, R \rangle = 0$$

where N is the test function and R is the residual of the governing equations in non-conservation form.

The solution of our differential equation is generally not globally smooth. If our equations contained a diffusive term (e.g. $\alpha \frac{\partial^2 q}{\partial x^2}$) then for $\alpha > 0$ there would be narrow regions within the domain, where the solution varies rapidly (steep gradients). If $\alpha = 0$, then the solution might even be discontinuous (A hydraulic jump is a typical example of a natural discontinuity in our solution).

When we compare the analytical solution, with that of the Bubnov-Galerkin approach (figures 4.3 and 4.4), the latter does not provide satisfactory results. The Bubnov-Galerkin scheme has proved itself to be effective in solving elliptic and parabolic differential equations. However, we see here that the optimisation of its test functions, with respect to the residual, does not provide accurate results; when applied to hyperbolic partial differential equations. The non-linear coefficient (u), from the odd-order derivative, in the

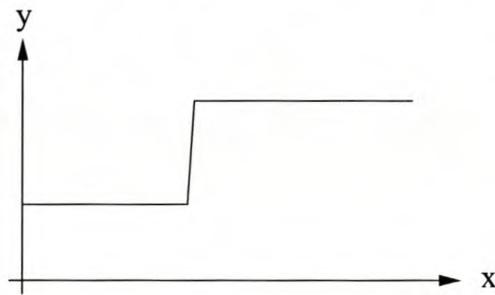


Figure 4.3: *Analytic solution, showing hydraulic jump*

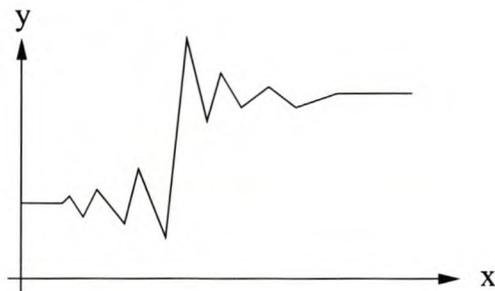


Figure 4.4: *Bubnov-Galerkin solution, showing spurious waves*

momentum equation (eq. 4.17) produces parasitic waves i.e. spurious oscillations. These oscillations, due to numerical error in the solution, do not decay as time progresses, but instead they grow beyond control and eventually the entire flow domain is occupied by these parasitic waves.

First improvement: Adding artificial viscosity explicitly by introducing a diffusive term

A possible solution to reduce these spurious oscillations is to add artificial viscosity to the numerical scheme. By adding such a diffusive term ($\alpha \frac{\partial^2 q}{\partial x^2}$) we are damping out the spurious oscillations.

Furthermore, the physics of our modeled system is starting to change from hyperbolic ($\alpha = 0$; where a change in data at a point only affects the solution locally) to parabolic ($\alpha > 0$; where a change in data at a point influences the solution at all points). The so called "streamline diffusion method" is based on this approach, where an artificial diffusion term (acting only in the direction of the streamlines) is added. The results of this approach are stable, but not very accurate, because the hydraulic jump (a natural discontinuity in the solution) has been smeared (figure 4.5). A stability analysis would reveal that all wavelengths, in a Fourier decomposition, gets damped; resulting in the observed smearing.



Figure 4.5: *Bubnov-Galerkin solution, with artificial viscosity added*

Second improvement: Adding artificial viscosity implicitly by using continuous upwind test functions

Another way of introducing an artificial diffusion term is by replacing the test function, which is identical to the trial function, by an upwind test function. A diffusive term materializes in our weak solution when upwind test functions are used (The next section gives an example of such a diffusive term).

Third improvement: Adding artificial viscosity implicitly by using discontinuous upwind test functions

A further improvement is to replace the upwind-continuous ($C^0, C^1, C^2 \dots$) test functions with discontinuous (C^{-1}) ones. We can see that the hyperbolic character of our differential equation requires that changes in data at a point only affects the solution locally around

that point. The system strives for some degree of uncoupling. This degree of uncoupling is controlled by the size in the discontinuity of the upwind discontinuous test functions.

A mechanism to control the size in the discontinuity is provided by the characteristics of the system and its implementation is called the "Characteristic Dissipative Galerkin scheme". This means that the discontinuity in the test functions is dynamically controlled by the characteristic directions at the node as the flow changes for each time step. Dendy [2] has proposed the following discontinuous test function for a single hyperbolic differential equation.

$$\bar{N}^* = \bar{N} + \omega \frac{\partial \bar{N}}{\partial \xi} \quad (4.18)$$

where

- \bar{N} is a vector continuous trial functions
- ω is a parameter determining the level of dissipation
- ξ is the spatial variable for a normalised domain

A value of $\omega = \frac{2}{\sqrt{15}}$ minimises phase error and provides optimum dissipation levels for pure convection [13]. This test function (fig.4.1) has proved to be largely successful.

For this Petrov-Galerkin method, the basis function and its spatial gradient is made orthogonal to the solution. The Petrov-Galerkin weak solution of the open channel equations (eq. 4.16 and 4.17), can be interpreted as the Galerkin approximation of the following system:

$$\frac{\partial y}{\partial t} + \frac{\partial q}{\partial x} - \omega \left[\frac{\partial^2 q}{\partial x \partial t} + 2u \frac{\partial^2 q}{\partial x^2} + (c^2 - u^2) \frac{\partial^2 y}{\partial x^2} \right] = 0 \quad (4.19)$$

$$\begin{aligned} \frac{\partial q}{\partial t} + (c^2 - u^2) \frac{\partial y}{\partial x} + 2u \frac{\partial q}{\partial x} \\ - \omega \left[(c^2 - u^2) \left(\frac{\partial^2 y}{\partial x \partial t} + \frac{\partial^2 q}{\partial x^2} \right) + 2u \left(\frac{\partial^2 q}{\partial x \partial t} + 2u \frac{\partial^2 q}{\partial x^2} + (c^2 - u^2) \frac{\partial^2 y}{\partial x^2} \right) \right] = 0 \end{aligned} \quad (4.20)$$

When ω approaches zero, the solution approaches the standard Galerkin solution of the original PDE set. When $\omega \neq 0$, higher order derivatives are introduced into the numerical solution.

The important feature of this test function is that the presence of the gradient of the shape function N , in the formulation is equivalent to the weak solution of flow equations containing higher order derivatives (which provides numerical dissipation), but with the advantage of being highly selective in damping out short wavelengths. It can be seen from a stability analysis, that the introduced dissipation error selectively damps out fourth-order frequencies (σ_n^4 ; typical spurious frequencies).

Fourth improvement: A discontinuous test function accounting for coupled hyperbolic equations

The test function presented in equation 4.18, does not apply to coupled hyperbolic equations. To uncouple these equations and implement the correct direction of upwinding we require the non-conservation form of the equation set, because it contains the convection matrix (See also subsection 4.2.3); which we can diagonalize. We then change this test function further, by dividing each eigenvalue with the absolute value of itself. Our test function for coupled and uncoupled equation sets is:

$$\bar{N}^* = \bar{N} + \omega \frac{\bar{A}}{|\bar{A}|} \cdot \frac{\partial \bar{N}}{\partial \xi} \quad (4.21)$$

$\frac{\bar{A}}{|\bar{A}|}$ merely accounts for the similarity transformation necessary to convert the coupled equations into uncoupled form and guarantee the correct sign on the dissipation terms. It therefore uses the characteristic directions of the differential equations to determine the relative upwinding.

The dissipation level's most important feature lies in the direct dependence of the mesh size and local wave speed. The dissipation level is allowed to vary according to the local characteristics of both the flow and computational conditions.

4.4 Discretization of the unsteady pipe-flow equations

This section discretizes the unsteady pipe-flow partial differential equations, spatially, by using the Characteristic Dissipative Galerkin finite element scheme. The discontinuous test function (eq. 4.21) described in the previous section is used.

4.4.1 Governing equations

The governing equations for pipe-flow are conservation of mass and momentum respectively (equations 2.26 and 2.63):

$$\frac{\partial h}{\partial t} + v \frac{\partial h}{\partial x} - v \sin \alpha + \frac{a^2}{g} \frac{\partial v}{\partial x} = 0 \quad (4.22)$$

$$\frac{\partial v}{\partial t} + g \frac{\partial h}{\partial x} + v \frac{\partial v}{\partial x} + \frac{\lambda v |v|}{2D} = 0 \quad (4.23)$$

which can be written as a velocity-pressure formulation in matrix form

$$\frac{\partial \bar{\psi}}{\partial t} + \bar{A} \cdot \frac{\partial \bar{\psi}}{\partial x} + \bar{B} \cdot \bar{\psi} = 0 \quad (4.24)$$

with the vector of unknowns given by

$$\bar{\psi} = \begin{Bmatrix} h \\ v \end{Bmatrix} \quad (4.25)$$

the convection-matrix given by

$$\bar{A} = \begin{bmatrix} v & \frac{a^2}{g} \\ g & v \end{bmatrix} \quad (4.26)$$

and the source/sink-matrix given by

$$\bar{B} = \begin{bmatrix} 0 & -\sin \alpha \\ 0 & \frac{\lambda |v|}{2D} \end{bmatrix} \quad (4.27)$$

Premultiplying \bar{A} with the scalar density ρ , followed by a dimensional analysis gives

$$\frac{1}{L^2} \begin{bmatrix} \frac{M}{T} & M & \frac{M}{T} & M \\ \frac{M}{T^2} & \frac{M}{T} & \frac{M}{T^2} & \frac{M}{T} \\ \frac{M}{T} & M & \frac{M}{T} & M \\ \frac{M}{T^2} & \frac{M}{T} & \frac{M}{T^2} & \frac{M}{T} \end{bmatrix} \quad (4.28)$$

The first and third rows have dimensions $\frac{M}{T}$ and M . These dimensions are similar to that of mass convection (Table 3.1). The second and fourth rows have dimensions $\frac{M}{T^2}$ and $\frac{M}{T}$. These dimensions are similar to that of momentum convection (Table 3.1). \bar{A} is therefore a convection-matrix, since it describes the convection of mass and momentum in the system.

\bar{B} is a source/sink-matrix, which describes the rate at which mass and momentum is introduced or removed from the system.

4.4.2 Trial functions

The solution consists of two fields (h and v). For a simple two node element each field will have two dofs per element. Furthermore, because the highest order derivative in our governing equations is one, we only need 1D linear shape functions. The trial function set is chosen as:

$$\bar{N} = \begin{bmatrix} n_1 & 0 & n_2 & 0 \\ 0 & n_1 & 0 & n_2 \end{bmatrix} \quad (4.29)$$

$$n_1 = \frac{1 - \xi}{2} \quad \xi \in [-1, 1] \quad (4.30)$$

$$n_2 = \frac{1 + \xi}{2} \quad (4.31)$$

4.4.3 Approximate solution

The approximate solution ($\hat{\phi}$) is obtained through the dot product of the trial functions with the element's dofs ($\bar{\phi}$). Note that the hat ($\hat{\quad}$) indicates an approximate value and not the exact one.

$$\hat{\phi} = \begin{Bmatrix} \hat{h} \\ \hat{v} \end{Bmatrix} = \bar{N} \cdot \bar{\phi} = \sum_j N_{ij} \phi_j \quad (4.32)$$

where the element's dofs are

$$\bar{\phi} = \begin{Bmatrix} h_1 \\ v_1 \\ h_2 \\ v_2 \end{Bmatrix} \quad (4.33)$$

4.4.4 Test functions

The test function presented in equation 4.21 is used, because of its ability to handle coupled hyperbolic differential equations. This ability stems from two of its properties; that is, discontinuity and upwind weighting (See fig. 4.1). The test function \bar{N}^* is

$$\bar{N}^* = \bar{N} + \omega \begin{pmatrix} \hat{A} \\ |\hat{A}| \end{pmatrix}^T \cdot \frac{\partial \bar{N}}{\partial \xi} \quad (4.34)$$

or compactly

$$\bar{N}^* = \bar{N} + \hat{V} \quad (4.35)$$

where

- ω is the parameter controlling the level of upwinding
- $\hat{\hat{A}}$ is the approximate convection matrix from the non-conservation form of our governing equations (eq. 4.26)
- $\frac{\hat{\hat{A}}}{|\hat{\hat{A}}|}$ is not matrix $\hat{\hat{A}}$ divided by the determinant of itself, but a notation indicating that each eigenvalue of $\hat{\hat{A}}$ should be divided by the absolute value of itself.

By diagonalizing $\hat{\hat{A}}$ and dividing each eigenvalue by the absolute value of itself, $\frac{\hat{\hat{A}}}{|\hat{\hat{A}}|}$ can be written as

$$\frac{\hat{\hat{A}}}{|\hat{\hat{A}}|} = \frac{\begin{bmatrix} \hat{v} & a^2 \\ g & \hat{v} \end{bmatrix}}{\begin{vmatrix} \hat{v} & a^2 \\ g & \hat{v} \end{vmatrix}} \quad (4.36)$$

$$= \frac{\begin{bmatrix} 1 & 1 \\ -\frac{g}{a} & \frac{g}{a} \end{bmatrix} \begin{bmatrix} \hat{v} - a & 0 \\ 0 & \hat{v} + a \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -\frac{g}{a} & \frac{g}{a} \end{bmatrix}^{-1}}{\begin{vmatrix} \hat{v} - a & 0 \\ 0 & \hat{v} + a \end{vmatrix}} \quad (4.37)$$

$$= \begin{bmatrix} 1 & 1 \\ -\frac{g}{a} & \frac{g}{a} \end{bmatrix} \begin{bmatrix} \frac{\hat{v} - a}{|\hat{v} - a|} & 0 \\ 0 & \frac{\hat{v} + a}{|\hat{v} + a|} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -\frac{g}{a} & \frac{g}{a} \end{bmatrix}^{-1} \quad (4.38)$$

$$= \begin{bmatrix} \frac{\hat{v} + a}{2|\hat{v} + a|} + \frac{\hat{v} - a}{2|\hat{v} - a|} & \frac{a(\hat{v} + a)}{2g|\hat{v} + a|} - \frac{a(\hat{v} - a)}{2g|\hat{v} - a|} \\ \frac{g(\hat{v} + a)}{2a|\hat{v} + a|} - \frac{g(\hat{v} - a)}{2a|\hat{v} - a|} & \frac{\hat{v} + a}{2|\hat{v} + a|} + \frac{\hat{v} - a}{2|\hat{v} - a|} \end{bmatrix} \quad (4.39)$$

Example test shape functions for sub-critical flows

The test shape function matrix $\bar{\bar{N}}^*$ is identical for upstream and downstream subcritical flows.

Sub-critical flow downstream/upstream ($a = 1200$ m/s, $v = \pm 1.5$ m/s, $\omega = \frac{2}{\sqrt{15}}$)

The upwinding matrix $\frac{\hat{\hat{A}}}{|\hat{\hat{A}}|}$ is

| | |
|--------|---------|
| 0 | 122.324 |
| 0.0082 | 0 |

The corresponding test shape function matrix $\bar{\bar{N}}^*$ is

| | | | |
|--------|---------|-------|--------|
| n_1 | -0.0021 | n_2 | 0.0021 |
| -31.58 | n_1 | 31.58 | n_2 |

Example test shape functions for super-critical flows

The test shape function matrix $\bar{\bar{N}}^*$ for upstream super-critical flow is the mirror image of that of downstream super-critical flow.

Super-critical flow downstream ($a = 0.3$ m/s, $v = 1.5$ m/s, $\omega = \frac{2}{\sqrt{15}}$)

The upwinding matrix $\frac{\hat{A}}{|\hat{A}|}$ is

| | |
|---|---|
| 1 | 0 |
| 0 | 1 |

The corresponding test shape function matrix $\bar{\bar{N}}^*$ is

| | | | |
|---------------|---------------|---------------|---------------|
| $n_1 - 0.258$ | 0 | $n_2 + 0.258$ | 0 |
| 0 | $n_1 - 0.258$ | 0 | $n_2 + 0.258$ |

Super-critical flow upstream ($a = 0.7$ m/s, $v = -1.5$ m/s, $\omega = \frac{2}{\sqrt{15}}$)

The upwinding matrix $\frac{\hat{A}}{|\hat{A}|}$ is

| | |
|----|----|
| -1 | 0 |
| 0 | -1 |

The corresponding test shape function matrix $\bar{\bar{N}}^*$ is

| | | | |
|---------------|---------------|---------------|---------------|
| $n_1 + 0.258$ | 0 | $n_2 - 0.258$ | 0 |
| 0 | $n_1 + 0.258$ | 0 | $n_2 - 0.258$ |

4.4.5 Jacobian of the coordinate transformation

Our finite element formulation will require some sort of mapping between local coordinates (ξ) and global coordinates (x), since our shape functions are defined in local coordinates and our model is described in global coordinates. This mapping is done through isoparametric construction:

$$\bar{x} = \bar{\bar{N}} \cdot \bar{c} \quad (4.40)$$

where the 3D implementation has

$$\bar{x} = \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} \quad (4.41)$$

$$\bar{c} = \begin{Bmatrix} x_1 \\ \dots \\ x_n \\ y_1 \\ \dots \\ y_n \\ z_1 \\ \dots \\ z_n \end{Bmatrix} \quad (4.42)$$

\bar{c} is a vector containing the global cartesian coordinate values of a finite element. The Jacobian \bar{J} of the coordinate transformation is given by

$$\bar{J} = (\bar{\nabla} \bar{x})^T \quad (4.43)$$

$$= (\bar{N} \cdot \bar{c}) \bar{\nabla} \quad (4.44)$$

$$= \bar{N} \bar{\nabla} \cdot \bar{c} \quad (4.45)$$

Since we are working with a one dimensional two-node element, equation 4.45 reduces to

$$J = \begin{bmatrix} \frac{-1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} \quad (4.46)$$

$$= \frac{x_2 - x_1}{2} \quad (4.47)$$

$$= \frac{\Delta x}{2} \quad (4.48)$$

Transformation of an infinitesimal length in global coordinates to normal coordinates can now be done by

$$dx = |det \bar{J}| d\xi \quad (4.49)$$

$$= \frac{\Delta x}{2} d\xi \quad (4.50)$$

where $|det \bar{J}|$ is the absolute value of the determinant of matrix \bar{J}

4.4.6 Petrov-Galerkin finite element

Using the following notation

$$\langle M, N \rangle = \int_{\Omega} M \cdot N d\Omega \quad (4.51)$$

The Petrov-Galerkin finite element is obtained from

$$\langle \bar{N}^{*T}, \bar{R} \rangle = 0 \quad (4.52)$$

where \bar{R} is the residual of the velocity-pressure formulation of equation 4.24, given by

$$\bar{R} = \frac{\partial \hat{\phi}}{\partial t} + \hat{A} \cdot \frac{\partial \hat{\phi}}{\partial x} + \hat{B} \cdot \hat{\phi} \quad (4.53)$$

with

$$\hat{A} = \begin{bmatrix} \hat{v} & \frac{a^2}{g} \\ g & \hat{v} \end{bmatrix} \quad (4.54)$$

$$\hat{B} = \begin{bmatrix} 0 & -\text{Sin}\alpha \\ 0 & \frac{\lambda|\hat{v}|}{2D} \end{bmatrix} \quad (4.55)$$

Substituting the residual \bar{R} in eq. 4.52 gives

$$\langle \bar{N}^{*T}, \frac{\partial \hat{\phi}}{\partial t} + \hat{A} \cdot \frac{\partial \hat{\phi}}{\partial x} + \hat{B} \cdot \hat{\phi} \rangle = 0 \quad (4.56)$$

or written as an integral

$$\int_a^b \left\{ \bar{N}^{*T} \cdot \left(\frac{\partial \hat{\phi}}{\partial t} + \hat{A} \cdot \frac{\partial \hat{\phi}}{\partial x} + \hat{B} \cdot \hat{\phi} \right) \right\} dx = 0 \quad (4.57)$$

Substituting equation 4.32 for $\hat{\phi}$ gives

$$\int_a^b \left\{ \bar{N}^{*T} \cdot \left(\frac{\partial}{\partial t} (\bar{N} \cdot \bar{\phi}) + \hat{A} \cdot \frac{\partial}{\partial x} (\bar{N} \cdot \bar{\phi}) + \hat{B} \cdot (\bar{N} \cdot \bar{\phi}) \right) \right\} dx = 0 \quad (4.58)$$

Applying the time derivative on the dof vector and the spatial derivative on the shape function matrix gives

$$\int_a^b \left\{ \bar{N}^{*T} \cdot \left(\bar{N} \cdot \frac{\partial \bar{\phi}}{\partial t} + \hat{A} \cdot \frac{\partial \bar{N}}{\partial x} \cdot \bar{\phi} + \hat{B} \cdot \bar{N} \cdot \bar{\phi} \right) \right\} dx = 0 \quad (4.59)$$

Grouping the time derivative- and spatial derivative-terms together gives

$$\int_a^b \left\{ \bar{N}^{*T} \cdot \bar{N} \cdot \frac{\partial \bar{\phi}}{\partial t} \right\} dx + \int_a^b \left\{ \bar{N}^{*T} \cdot \left(\hat{A} \cdot \frac{\partial \bar{N}}{\partial x} \cdot \bar{\phi} + \hat{B} \cdot \bar{N} \cdot \bar{\phi} \right) \right\} dx = 0 \quad (4.60)$$

Introducing natural coordinates by substituting equation 4.50 into equation 4.60 gives:

$$\left\{ \int_{-1}^{+1} \bar{N}^{*T} \cdot \bar{N} \frac{\Delta x}{2} d\xi \right\} \cdot \frac{\partial \bar{\phi}}{\partial t} + \left\{ \int_{-1}^{+1} \bar{N}^{*T} \cdot \left(\hat{A} \cdot \frac{\partial \bar{N}}{\partial \xi} \frac{2}{\Delta x} + \hat{B} \cdot \bar{N} \right) \frac{\Delta x}{2} d\xi \right\} \cdot \bar{\phi} = 0 \quad (4.61)$$

or simply

$$\bar{M} \cdot \frac{\partial \bar{\phi}}{\partial t} + \bar{C} \cdot \bar{\phi} = 0 \quad (4.62)$$

with the mass-matrix given by

$$\bar{M} = \int_{-1}^{+1} \bar{N}^{*T} \cdot \bar{N} \frac{\Delta x}{2} d\xi \quad (4.63)$$

and the convection-matrix given by

$$\bar{C} = \int_{-1}^{+1} \bar{N}^{*T} \cdot \left(\hat{A} \cdot \frac{\partial \bar{N}}{\partial \xi} + \hat{B} \cdot \bar{N} \frac{\Delta x}{2} \right) d\xi \quad (4.64)$$

A dimensional analysis of \bar{M} reveals that all entries are of dimension L^2 . Premultiplying \bar{M} with the scalar density ρ (dimensions $\frac{M}{L^3}$), changes the dimensions of its elements to $\frac{M}{L}$. \bar{M} can therefore be called the mass-matrix. It describes the mass-fluxes and forces introduced into the system, by the mass of the particles.

Premultiplying \bar{C} also with the scalar density ρ , followed by a dimensional analysis gives the same dimensional matrix as the one defined in equation 4.28. \bar{C} can therefore also be called a convection-matrix, since it describes the mass-fluxes and forces introduced into the system by the convection of mass and momentum in the system. A last remark about \bar{C} is that, although \bar{C} looks similar to a stiffness-matrix, in the structural sense, it is not. For linear structural problems the stiffness matrix is conveniently also the Jacobian of the equation system's residual. \bar{C} is not the Jacobian of this equation system's residual. The Jacobian will be defined in a later section.

Two-point Gauss quadrature was used to numerically integrate \bar{M} and \bar{C} . The approximate value \hat{v} was evaluated at the Gauss points, with the formula

$$\hat{v} = v_1 n_1 + v_2 n_2 \quad (4.65)$$

where n_1 and n_2 are the shape functions defined in equation 4.29.

It is interesting to note that for the open-channel equations (eq. 4.16 and 4.17), Hicks and Steffler uses a variant of the Petrov-Galerkin scheme presented above [5]. Instead of eq. 4.52 i.e.

$$\left\langle \bar{N}^T + \hat{V}^T, \bar{R}_{non-conservation\ form} \right\rangle \quad (4.66)$$

the Petrov-Galerkin formulation Hicks and Steffler uses reads

$$\left\langle \bar{N}^T, \bar{R}_{\text{conservation form}} \right\rangle + \left\langle \hat{\bar{V}}^T, \bar{R}_{\text{non-conservation form}} \right\rangle \quad (4.67)$$

This formulation conserves the flux variables and maintains the high level of selective damping. This formulation is possible, because the conservation form and non-conservation form of the open-channel equations are both available, in contrast to the pipe-flow equations.

4.4.7 θ implicit finite difference scheme

In the previous sections we have seen how the spatial derivatives were discretized using 1D Petrov-Galerkin finite elements. The time derivative however, is discretized using a θ implicit finite difference scheme. The finite difference scheme is implicit, meaning that a system of equations will need to be solved for each time-step.

The θ implicit finite difference scheme is derived from a Taylor series expansion as follows.

$$\bar{\phi}^{n+1} = \bar{\phi}^{n+\frac{1}{2}} + \frac{\partial \bar{\phi}^{n+\frac{1}{2}}}{\partial t} \left(\frac{\Delta t}{2} \right) + O(\Delta t^2) \quad (4.68)$$

$$\bar{\phi}^n = \bar{\phi}^{n+\frac{1}{2}} - \frac{\partial \bar{\phi}^{n+\frac{1}{2}}}{\partial t} \left(\frac{\Delta t}{2} \right) + O(\Delta t^2) \quad (4.69)$$

Eliminating $\bar{\phi}^{n+\frac{1}{2}}$ by substituting equation 4.69 into 4.68 gives a second-order accurate approximation of a central time derivative

$$\frac{\bar{\phi}^{n+1} - \bar{\phi}^n}{\Delta t} = \frac{\partial \bar{\phi}^{n+\frac{1}{2}}}{\partial t} = \frac{1}{2} \left(\frac{\partial \bar{\phi}^{n+1}}{\partial t} + \frac{\partial \bar{\phi}^n}{\partial t} \right) \quad (4.70)$$

More flexibility can be added to this equation by introducing the scalar θ , which controls the level of implicitness

$$\frac{\bar{\phi}^{n+1} - \bar{\phi}^n}{\Delta t} = \theta \frac{\partial \bar{\phi}^{n+1}}{\partial t} + (1 - \theta) \frac{\partial \bar{\phi}^n}{\partial t} \quad (4.71)$$

When ($\theta = 0$) the difference scheme is in explicit form. When ($\theta = 1$) the difference scheme is fully implicit. Other values of θ creates a scheme which is a combination of both. Implicit schemes usually encounter accuracy problems, for high Courant numbers, while explicit schemes encounter stability problems. This motivates the importance of a stability analysis of the algorithm for different values of θ , ω and Courant numbers (Chapter 6).

The advantage of using an implicit scheme is that we will not be restricted to Courant numbers of less than one. The Courant number is an indication of how accurately the

numerical scheme samples data from the domain of dependence. The Courant number is defined as

$$Courant = \frac{a\Delta t}{L} \quad (4.72)$$

where

- a is the wavespeed
- Δt is the time step size
- L is the length of the element

By premultiplying equation 4.62 with \bar{M}^{-1} we obtain equations for the time derivatives at the current ($n + 1$) and previous (n) time levels.

$$\frac{\partial \bar{\phi}^{n+1}}{\partial t} = \bar{M}^{-1} \cdot \{ \bar{C}^{n+1} \cdot \bar{\phi}^{n+1} \} \quad (4.73)$$

$$\frac{\partial \bar{\phi}^n}{\partial t} = \bar{M}^{-1} \cdot \{ \bar{C}^n \cdot \bar{\phi}^n \} \quad (4.74)$$

Equations 4.73 and 4.74 are then substituted into equation 4.71. After some reshuffling (and assuming that \bar{M} is not a function of $\bar{\phi}$) the following equation is obtained

$$\{ \bar{M} + \theta \Delta t \bar{C}^{n+1} \} \cdot \bar{\phi}^{n+1} = \{ \bar{M} + \Delta t (\theta - 1) \bar{C}^n \} \cdot \bar{\phi}^n = 0 \quad (4.75)$$

or simply

$$\bar{D}(\bar{\phi}^{n+1}) \cdot \bar{\phi}^{n+1} = \bar{P}(\bar{\phi}^n) \cdot \bar{\phi}^n \quad (4.76)$$

where

$$\bar{D}(\bar{\phi}^{n+1}) = \bar{M} + \theta \Delta t \bar{C}^{n+1} \quad (4.77)$$

$$\bar{P}(\bar{\phi}^n) = \bar{M} + (\theta - 1) \Delta t \bar{C}^n \quad (4.78)$$

\bar{D} is called the unknown head-velocity coefficient matrix, or simply the coefficient matrix

\bar{P} is called the known head-velocity coefficient matrix

$\bar{P} \cdot \bar{\phi}$ is called the mass-momentum flux vector

Note that \bar{C} depends on the vector $\bar{\phi}$, because it contains the dependent variable \hat{v} . Therefore, $D(\bar{\phi})^{n+1}$, $P(\bar{\phi})^n$ also depends on the vector $\bar{\phi}$.

4.4.8 Residual of an element, with boundary conditions imposed

The boundary conditions in chapter 5 are functions of head and velocity dofs. These functions may range from simple prescribed values (e.g. constant head due to a reservoir) to non-linear relations (e.g. a valve's velocity dofs). A boundary condition's dofs can be divided into condensed dofs ($\bar{\phi}_c^{n+1}$) and retained dofs ($\bar{\phi}_r^{n+1}$). A general relationship between the condensed and retained dofs of a boundary condition can be written as

$$\bar{\phi}_c^{n+1} = \bar{g}(\bar{\phi}_r^{n+1}) \quad (4.79)$$

From equation 4.79 it can be seen that if a reasonable estimate of the boundary condition's $\bar{\phi}_r^{n+1}$ is available, then its corresponding $\bar{\phi}_c^{n+1}$ can be determined (e.g. the equivalent of equation 4.79 for an inline valve is equation 5.5). The boundary conditions in this thesis are implemented by way of a condensation contragradient transformation on the element coefficient matrices.

The condensation transformation begins by reordering the dofs of element equation 4.76 into two groups, namely condensed and retained dofs

$$\begin{bmatrix} \bar{D}_{rr} & \bar{D}_{rc} \\ \bar{D}_{cr} & \bar{D}_{cc} \end{bmatrix} \begin{Bmatrix} \bar{\phi}_r^{n+1} \\ \bar{\phi}_c^{n+1} \end{Bmatrix} = \begin{Bmatrix} \bar{Q}_r \\ \bar{Q}_c \end{Bmatrix} \quad (4.80)$$

where

- r and c stands for retained and condensed, respectively
- \bar{Q} is a flux vector containing known values, based on the previous time-step's solution
- $\bar{D}_{??}$ is a sub-matrix of the element coefficient matrix, which is dependent on $\bar{\phi}^{n+1}$

Substitution of equation 4.79 into the first equation of the system defined by equation 4.80 gives

$$\bar{D}_{rr} \cdot \bar{\phi}_r^{n+1} + \bar{D}_{rc} \cdot \bar{g}(\bar{\phi}_r^{n+1}) = \bar{Q}_r \quad (4.81)$$

The residual of the condensed element ($dofs = \bar{\phi}_r$) is therefore

$$\bar{f}_r^{n+1} = \bar{D}_{rr} \cdot \bar{\phi}_r^{n+1} - \left(\bar{Q}_r - \bar{D}_{rc} \cdot \bar{g}(\bar{\phi}_r^{n+1}) \right) \quad (4.82)$$

4.4.9 Jacobian matrix of an element

Residual of an element (A convenient form for formulating the Jacobian)

By rearranging equation 4.71 we get the useful equation

$$\frac{\partial \bar{\phi}^{n+1}}{\partial t} = \left(\frac{1}{\theta \Delta t} \right) \bar{\phi}^{n+1} - \left[\left(\frac{1}{\theta \Delta t} \right) \bar{\phi}^n + \left(\frac{1-\theta}{\theta} \right) \frac{\partial \bar{\phi}^n}{\partial t} \right] \quad (4.83)$$

or simply

$$\frac{\partial \bar{\phi}^{n+1}}{\partial t} = \beta_1 \bar{\phi}^{n+1} - \bar{\beta}_2 \quad (4.84)$$

The head and velocity time derivatives can be written in this form

$$\frac{\partial h}{\partial t} = \beta_1 h - \beta_2 \quad (4.85)$$

$$\frac{\partial v}{\partial t} = \beta_1 v - \beta_3 \quad (4.86)$$

Substituting equations 4.85 and 4.86 into equation 4.53 gives the following form of the differential equation's residual

$$\bar{R} = \left\{ \begin{array}{l} \beta_1 h - \beta_2 + v \frac{\partial h}{\partial x} + \frac{a^2}{g} \frac{\partial v}{\partial x} - v \sin(\alpha) \\ \beta_1 v - \beta_3 + g \frac{\partial h}{\partial x} + v \frac{\partial v}{\partial x} + \frac{\lambda |v| v}{2D} \end{array} \right\} \quad (4.87)$$

For an element, equation 4.52 will typically not be exactly zero, due to the initial guess of $\bar{\phi}^{n+1}$ not being the correct solution. The element's residual \bar{f}^{n+1} can therefore be defined as

$$\bar{f}^{n+1} = \int_a^b (\bar{N}^{*T} \cdot \bar{R}) dx \quad (4.88)$$

Substituting equation 4.87 into equation 4.88 gives

$$\bar{f}^{n+1} = \int_a^b \left(\bar{N}^{*T} \cdot \left\{ \begin{array}{l} \beta_1 h - \beta_2 + v \frac{\partial h}{\partial x} + \frac{a^2}{g} \frac{\partial v}{\partial x} - v \sin(\alpha) \\ \beta_1 v - \beta_3 + g \frac{\partial h}{\partial x} + v \frac{\partial v}{\partial x} + \frac{\lambda |v| v}{2D} \end{array} \right\} \right) dx \quad (4.89)$$

Rewriting this equation in terms of local coordinates gives

$$\bar{f}^{n+1} = \int_{-1}^1 \left(\bar{N}^{*T} \cdot \left\{ \begin{array}{l} \beta_1 h - \beta_2 + v \frac{\partial h}{\partial \xi} \frac{2}{\Delta x} + \frac{a^2}{g} \frac{\partial v}{\partial \xi} \frac{2}{\Delta x} - v \sin(\alpha) \\ \beta_1 v - \beta_3 + g \frac{\partial h}{\partial \xi} \frac{2}{\Delta x} + v \frac{\partial v}{\partial \xi} \frac{2}{\Delta x} + \frac{\lambda |v| v}{2D} \end{array} \right\} \right) \frac{\Delta x}{2} d\xi \quad (4.90)$$

or simply

$$\bar{f}^{n+1} = \int_{-1}^1 (\bar{N}^{*T} \cdot \bar{\gamma}) \frac{\Delta x}{2} d\xi \quad (4.91)$$

Jacobian of an element, without imposed boundary conditions

The Jacobian of an element is defined as

$$\bar{j}^{n+1} = (\bar{\nabla}_{\bar{\phi}} \bar{f}^{n+1})^T \quad (4.92)$$

where

$$\bar{\nabla}_{\bar{\phi}} = \left\{ \begin{array}{l} \frac{\partial}{\partial h_1} \\ \frac{\partial}{\partial v_1} \\ \frac{\partial}{\partial h_2} \\ \frac{\partial}{\partial v_2} \end{array} \right\} \quad (4.93)$$

Substituting equation 4.91 into equation 4.92 gives the Jacobian of an unconstrained element (i.e. no boundary conditions are imposed)

$$\bar{j}^{n+1} = \int_{-1}^1 \left(\bar{N}^{*T} \cdot \bar{\gamma} \bar{\nabla}_{\bar{\phi}} \right) \frac{\Delta x}{2} d\xi \quad (4.94)$$

where

$$\bar{\gamma} \bar{\nabla}_{\bar{\phi}} = \begin{bmatrix} \frac{\partial \gamma_1}{\partial h_1} & \frac{\partial \gamma_1}{\partial v_1} & \frac{\partial \gamma_1}{\partial h_2} & \frac{\partial \gamma_1}{\partial v_2} \\ \frac{\partial \gamma_2}{\partial h_1} & \frac{\partial \gamma_2}{\partial v_1} & \frac{\partial \gamma_2}{\partial h_2} & \frac{\partial \gamma_2}{\partial v_2} \end{bmatrix} \quad (4.95)$$

with

$$\frac{\partial \gamma_1}{\partial h_1} = \beta_1 n_1 + v \frac{\partial n_1}{\partial \xi} \frac{2}{\Delta x} \quad (4.96)$$

$$\frac{\partial \gamma_1}{\partial h_2} = \beta_1 n_2 + v \frac{\partial n_2}{\partial \xi} \frac{2}{\Delta x} \quad (4.97)$$

$$\frac{\partial \gamma_1}{\partial v_1} = n_1 \frac{\partial h}{\partial \xi} \frac{2}{\Delta x} + \frac{a^2}{g} \frac{\partial n_1}{\partial \xi} \frac{2}{\Delta x} - n_1 \text{Sin}(\alpha) \quad (4.98)$$

$$\frac{\partial \gamma_1}{\partial v_2} = n_2 \frac{\partial h}{\partial \xi} \frac{2}{\Delta x} + \frac{a^2}{g} \frac{\partial n_2}{\partial \xi} \frac{2}{\Delta x} - n_2 \text{Sin}(\alpha) \quad (4.99)$$

$$\frac{\partial \gamma_2}{\partial h_1} = g \frac{\partial n_1}{\partial \xi} \frac{2}{\Delta x} \quad (4.100)$$

$$\frac{\partial \gamma_2}{\partial h_2} = g \frac{\partial n_2}{\partial \xi} \frac{2}{\Delta x} \quad (4.101)$$

$$\frac{\partial \gamma_2}{\partial v_1} = \beta_1 n_1 + \frac{2}{\Delta x} \left(n_1 \frac{\partial v}{\partial \xi} + v \frac{\partial n_1}{\partial \xi} \right) + \frac{\lambda n_1 |v|}{2D} + \frac{\lambda n_1 v^2}{2D|v|} \quad (4.102)$$

$$\frac{\partial \gamma_2}{\partial v_2} = \beta_1 n_2 + \frac{2}{\Delta x} \left(n_2 \frac{\partial v}{\partial \xi} + v \frac{\partial n_2}{\partial \xi} \right) + \frac{\lambda n_2 |v|}{2D} + \frac{\lambda n_2 v^2}{2D|v|} \quad (4.103)$$

Jacobian of an element, with imposed boundary conditions

In section 4.4.8 we saw that different boundary conditions introduced different condensed and retained dofs. The relationship between the condensed dofs and the retained dofs was given by equation 4.79. Furthermore, by condensation it was possible to express our element residual in terms of retained dofs. If our element residual is defined in terms of retained dofs, then the element's Jacobian matrix also has to be defined in terms of retained dofs. The condensed Jacobian matrix of an element ($dofs = \bar{\phi}_r$), is therefore

$$\bar{j}_r^{n+1} = \left(\bar{\nabla}_{\bar{\phi}_r} \bar{f}_r^{n+1} \right)^T \quad (4.104)$$

whose derivation is similar to that of an unconstrained element, except that equation 4.79 is substituted into equation 4.87. The consequences of equations 4.82 and 4.104 is that each boundary condition will introduce a unique element residual and Jacobian, which will need to be derived and programmed.

4.5 Solution algorithm

4.5.1 Newton-Raphson iteration

Element equation 4.76 has been partitioned into two parts: The left hand side of the equation depends on $\bar{\phi}^{n+1}$ (which contains the unknown dofs for the current time-step). While the right hand side depends only on $\bar{\phi}^n$ (which contains the known dofs for the previous time-step). The right hand side is therefore completely known (it can be called the element flux-vector \bar{Q}) and it remains to solve only the dofs on the left hand side, essentially vector $\bar{\phi}^{n+1}$. Assembly of the element equations (eq. 4.76) produces a non-linear equation system. To solve the flow variables (v and h), we require the element coefficient matrices (\bar{D} and \bar{P}) of equation 4.76 and the appropriate boundary conditions. However, these element coefficient matrices and boundary conditions are dependent on the flow variables for which we want to solve. An iterative solver is required, to solve such a problem. The solver used in this thesis is the Newton-Raphson method. For a given time-step (e.g n+1), the method is given by

$$\bar{\phi}_r^{i+1} = \bar{\phi}_r^i - \left(\bar{J}_r^i \right)^{-1} \cdot \bar{F}_r^i \quad (4.105)$$

or simply

$$\bar{\phi}_r^{i+1} = \bar{\phi}_r^i + \Delta \bar{\phi}_r \quad (4.106)$$

where

- i is the iteration number
- $\bar{\phi}_r^{i+1}$ is the approximation for the global retained dof vector at iteration i+1
- $\bar{\phi}_r^i$ is the approximation for the global retained dof vector at iteration i
- \bar{J}_r^i is the global equation system's condensed Jacobian at iteration i
- \bar{F}_r^i is the global equation system's condensed residual at iteration i

The condensed **global residual** at time n+1 (\bar{F}_r^{n+1}) is obtained by assembling all the element residuals (\bar{f}_r^{n+1}), which was derived in section 4.4.8. A considerable amount of computational time and computer storage is saved by assembling the residuals of each element, instead of calculating the global residual from the globally assembled \bar{D} and \bar{P} matrices (eq. 4.76).

Further, the condensed **global Jacobian** matrix at time n+1 (\bar{J}_r^{n+1}) is found by assembling all the element Jacobian matrices (\bar{j}_r^{n+1}), which was derived in section 4.4.9.

An improved estimate of the **retained dofs** at time $n+1$ ($\bar{\phi}_r^{n+1}$) can be obtained by substituting the global residual and global Jacobian matrix into the iterative Newton-Raphson scheme (eq. 4.105). Consequently, an improved estimate of the **condensed dofs** $\bar{\phi}_c^{n+1}$ can be obtained by substituting the improved retained dofs $\bar{\phi}_r^{n+1}$ into equation 4.79. (An alternative method for implementing boundary conditions is presented in section 8.3.1 of the conclusion of this thesis)

4.5.2 Computationally efficient form of Newton-Raphson iteration

The Newton-Raphson equation (eq. 4.105) can be written in a more computationally efficient manner, as follows:

$$\bar{\phi}_r^{i+1} = \bar{\phi}_r^i - \left(\bar{J}_r^i\right)^{-1} \cdot \bar{F}_r^i \quad (4.107)$$

$$\bar{\phi}_r^{i+1} - \bar{\phi}_r^i = - \left(\bar{J}_r^i\right)^{-1} \cdot \bar{F}_r^i \quad (4.108)$$

$$\Delta \bar{\phi}_r = - \left(\bar{J}_r^i\right)^{-1} \cdot \bar{F}_r^i \quad (4.109)$$

$$\bar{J}_r^i \cdot \Delta \bar{\phi}_r = - \bar{F}_r^i \quad (4.110)$$

\bar{J}_r^i can then be decomposed into an upper and lower matrix, giving

$$\bar{L} \cdot \bar{U} \cdot \Delta \bar{\phi}_r = - \bar{F}_r^i \quad (4.111)$$

$$\Delta \bar{\phi}_r = - \bar{U}^{-1} \cdot \bar{L}^{-1} \cdot \bar{F}_r^i \quad (4.112)$$

Computation of the inverses of \bar{L} and \bar{U} is much less expensive, computationally, than the inverse of \bar{J}_r^i . This thesis solved the non-linear equation system by using MINPACK's "hybrd" solver routine (public domain software). The routine only requires a user-defined function for the equation system's residual (eq. 4.53). It approximates the Jacobian by finite differences by calling the residual function, with an approximate solution as argument. MINPACK also has a routine, called "hybrj", which uses user-defined functions of the residual and Jacobian. The reason for using these routines, instead of Newton-Raphson, was simplicity and better convergence, when poor initial estimates are provided for the solver.

4.5.3 Overall algorithm

By considering equation 4.105, we set the initial guess of the current time-step's retained dof vector $\bar{\phi}_r^i$ equal to the retained dof vector of the previous time-step $\bar{\phi}_r^n$. We then compute the condensed and uncondensed/unconstrained element residuals and Jacobian matrices ($dofs = \bar{\phi}_r$) at the current time-step ($n+1$). The condensed global residual

and Jacobian (\bar{F}_r^i and \bar{J}_r^i) are found through assembly of the condensed and uncondensed element residuals and Jacobians. The condensed global residual and Jacobian matrix are then substituted into the Newton-Raphson equation (eq. 4.105), whose evaluation provides an improved approximation of the retained dofs $\bar{\phi}_r^{i+1}$ at time $n+1$.

If the norm of our global residual is found to be too large we use the computed $\bar{\phi}_r^{i+1}$ as the next guess for iteration $i+1$ and recompute the condensed dofs, as well as the condensed and uncondensed element residuals and Jacobian matrices (still time-step $n+1$).

Again we assemble these element residuals and Jacobian matrices. The condensed global residual and Jacobian matrix are then substituted into the Newton-Raphson equation, which is then evaluated. This whole process is repeated ($i = 2, 3, 4, \dots$) until an acceptably small residual norm for time step $n+1$ is found.

We then march one time-step forward ($n+2$) and use the converged solution of our previous time-step ($\bar{\phi}_r^{n+1}$) as the initial guess of the current time step ($\bar{\phi}_r^i$). We repeatedly compute the condensed global residual (as in the previous paragraphs) until an acceptably small residual norm is reached. With the new solution we march to the next time step ($n+3$). This whole procedure is repeated until we have reached the final time-step.

4.6 Summary of algorithm

4.6.1 Overall solving process

The overall algorithm for the Petrov-Galerkin finite element scheme starts at time ($t = 0$) with the initial solution $\bar{\phi}^0$ as input. It calls the Newton-Raphson solver, which returns the current time-step's converged solution for the retained dofs $\bar{\phi}_r^{n+1}$. The retained dofs are then used to compute the condensed dofs $\bar{\phi}_c^{n+1}$. The time-step's solution $\bar{\phi}^{n+1}$ is then assembled from the retained and condensed dofs. The process then saves the solution as output and moves to the next time-step. The whole process is repeated, until the final time-step is reached in which case the algorithm stops. See figure 4.6.

4.6.2 Newton-Raphson iterative solver

For time-step $n+1$, the Newton-Raphson solver starts with initial guess that $\bar{\phi}_r^i$ is equal to the previous time-step's retained dof vector $\bar{\phi}_r^n$. It then calculates the condensed dofs $\bar{\phi}_c^i$ by substituting the guessed retained dof vector into equation 4.79. It then computes each unconstrained element's residual and Jacobian matrix. It also computes the condensed

```

 $\bar{\phi}^0, t = 0$ 
Save  $(t, \bar{\phi}^0)$ 
Set the current time-step's solution  $\bar{\phi}^n = \bar{\phi}^0$ 
while  $t \leq tMax$  do
     $t = t + \Delta t$  (i.e. proceed to next time-step  $n+1$ )
    Get converged  $\bar{\phi}_r^{n+1}$  by calling Newton-Raphson solver with  $\bar{\phi}^n$  as argument (fig. 4.7)
    Solve condensed dofs:  $\bar{\phi}_c^{n+1} = \bar{g}(\bar{\phi}_r^{n+1})$  (Chapter 5, boundary conditions)
     $\bar{\phi}^{n+1} = [\bar{\phi}_r^{n+1}, \bar{\phi}_c^{n+1}]^T$ 
    Save  $(t, \bar{\phi}^{n+1})$ 
    Set  $\bar{\phi}^n = \bar{\phi}^{n+1}$ 
end

```

Figure 4.6: Overall solving process

residual and Jacobian, for elements constrained by boundary conditions. The condensed global residual and Jacobian are then assembled (i.e. \bar{F}_r^i and \bar{J}_r^i). If the norm of the global residual is too big, then a better estimate of the retained dofs $\bar{\phi}_r^{i+1}$ is calculated using the Newton-Raphson equation. The process is repeated, until the residual norm is acceptably small. The process then sets $\bar{\phi}_r^{n+1}$ equal to the converged $\bar{\phi}_r^i$. It then returns $\bar{\phi}_r^{n+1}$ as output. See figure 4.7.

4.7 General remarks concerning the algorithm

It is appropriate at this point to discuss qualitatively what the algorithm does, when it solves the unsteady pipe-flow equations.

The driving force of the algorithm is the disturbances introduced at each node in the model. These disturbances are propagated between nodes by shockwaves, or more appropriately characteristic lines. A hyperbolic PDE has two characteristic lines at any point within its domain. Therefore, the number of shockwaves in our discretized model is equal to twice the number of nodes.

Furthermore, the model assumes a flat shockfront. There does not exist a quick buildup of pressure as the shockwave reaches a node. Rather, the node is instantaneously influenced by the full magnitude of the shock wave.

Another point of note, is that at first the mesh for a model might seem rather coarse (the physical length of a pipe element generally ranges from tens to thousands of metres). The reason for using such large elements is, because of the high wavespeeds encountered in waterhammer problems (a ranges from 300m/s to 1400m/s). This means that for a time-step of length Δt , the distance travelled by the shockwave is $L = a\Delta t$. A Courant number of one, implies that we place two adjacent nodes a distance $a\Delta t$ apart, such that each node intercepts the information/disturbance sent by its neighbour, precisely at each time-step.

```

 $\bar{\phi}^n$ 
 $i = 0$ 
Set  $\bar{\phi}_r^i = \bar{\phi}^n$ 
while ( $i \leq iMax$  or  $\bar{\phi}_r^{n+1}$  unconverged ) do
     $i = i + 1$ 
     $\bar{\phi}_c^i = \bar{g}(\bar{\phi}_r^i)$  (eq. 4.79)
    for unconstrained  $Element_k$  to  $Element_n$  do
        Compute  $Element_k$ 's residual  $\bar{f}_r^i$  (eq. 4.82)
        Compute  $Element_k$ 's jacobian matrix  $\bar{j}_r^i$  (eq. 4.94)
        Add  $Element_k$ 's residual to the condensed global residual  $\bar{F}_r^i$ 
        Add  $Element_k$ 's Jacobian to the condensed global Jacobian matrix  $\bar{J}_r^i$ 
    end
    for constrained  $Element_l$  to  $Element_n$  do
        Compute  $Element_l$ 's condensed residual  $\bar{f}_r^i$  (eq. 4.82)
        Compute  $Element_l$ 's condensed jacobian matrix  $\bar{j}_r^i$  (eq. 4.104)
        Add  $Element_l$ 's residual to the condensed global residual  $\bar{F}_r^i$ 
        Add  $Element_l$ 's Jacobian to the condensed global Jacobian matrix  $\bar{J}_r^i$ 
    end
    if  $norm(\bar{F}_r^i) \leq$  Convergence Criteria
        then Set  $\bar{\phi}_r^{n+1} = \bar{\phi}_r^i$ 
            return  $\bar{\phi}_r^{n+1}$ 
        else Compute a better estimate  $\bar{\phi}_r^{i+1} = \bar{\phi}_r^i - (\bar{J}_r^i)^{-1} \cdot \bar{F}_r^i$ 
    end
end
end

```

Figure 4.7: *Newton-Raphson solver for time-step $n+1$*

Chapter 5

Boundary Conditions

5.1 Introduction

This chapter discusses the different boundary conditions encountered in unsteady pipe flow and how they were implemented, within the Petrov-Galerkin finite element scheme. There are three main types of boundary conditions:

- **Dirichlet/Essential boundary condition** The variable to be solved for, is expressed as a function. e.g. $u = f$ on ∂R
- **Neumann/Non-Essential boundary condition** The variable to be solved for, is expressed as a derivative. e.g. $\frac{\partial u}{\partial n} = f$ on ∂R
- **Robin (mixed) boundary condition** The variable to be solved for, is expressed as a combination of derivatives and non-derivatives e.g. $\frac{\partial u}{\partial n} + ku = f$ on ∂R

Boundary conditions are needed to guide our solution as time progresses. Applying integration by parts on the weak solution produces the PDE's natural boundary conditions. Usually if we had Neumann boundary conditions, we could substitute them for the natural boundary conditions obtained by integration by parts. However, the boundary conditions within this thesis are not of the Neumann type. All of them are of the Dirichlet type. For Dirichlet boundary conditions no integration by parts was necessary. Implementation of the boundary conditions was by a condensation contragradient transformation. There are a large number of boundary conditions for pipe-flow, unfortunately, only a small set could be presented, due to the limited time available. The boundary conditions discussed are:

- Pipe deadends
- Reservoirs
- Valves

- Pump power failures

5.2 Procedure for implementing boundary conditions

The boundary conditions described in this chapter are functions of head and velocity dofs. As stated in section 4.4.8, the dofs of a boundary condition $\bar{\phi}$ can be divided into condensed dofs $\bar{\phi}_c$ and retained dofs $\bar{\phi}_r$. This chapter shows what dofs should be condensed and what dofs should be retained for each boundary condition. It also shows the relationship between the condensed dofs and the retained dofs (eq. 4.79). This relationship is used to derive the condensed residual and Jacobian of a constrained element (See sections 4.4.9 and 4.4.9). These condensed residuals and Jacobians, together with the unconstrained element's residuals and Jacobians are then assembled into the condensed global residual and Jacobian of retained dofs. The condensed global residual and Jacobian in turn is substituted in the Newton-Raphson equation and the retained dofs consequently solved. The retained dofs are then used to compute the condensed dofs using the same boundary condition relationships. (A simpler method for implementing boundary conditions is presented in section 8.3.1 of the conclusion of this thesis)

5.3 Deadends

The pipe deadend is the simplest boundary condition. No flow can go through a deadend and therefore the velocity at the deadend is always zero. This means that depending at what node of the pipe the deadend is located, $v_1 = 0$ or $v_2 = 0$. This is a so called **Dirichlet boundary condition**. Let us assume that the deadend is located at node 1 of the pipe. Dof v_1 is therefore a condensed dof and dofs (h_1, h_2, v_2) are retained dofs. The approximate flow fields at any point within the pipe element is therefore given by

$$h = h_1 n_1 + h_2 n_2 \quad (5.1)$$

$$v = v_2 n_2 \quad (5.2)$$

Where n_1 and n_2 are the shape functions defined in equation 4.29. Substitution of equations 5.1 and 5.2 into equation 4.90 gives the residual of the constrained pipe element \bar{f}_r^{n+1} . Further, the condensed Jacobian \bar{j}_r^{n+1} of the constrained pipe element is found by substituting the above residual into equation 4.104 with $\bar{\nabla}_{\bar{\phi}_r}$ defined as

$$\bar{\nabla}_{\bar{\phi}_r} = \left\{ \begin{array}{c} \frac{\partial}{\partial h_1} \\ \frac{\partial}{\partial h_2} \\ \frac{\partial}{\partial v_2} \end{array} \right\} \quad (5.3)$$

5.4 Reservoirs

The reservoir boundary condition is almost the opposite of the pipe deadend. For the reservoir, the pressure head (water level in this case) can be assumed to be a constant. This means that depending at what node of the pipe the reservoir is located $h_1 = \text{constant}$ or $h_2 = \text{constant}$. This is also a **Dirichlet boundary condition**. Let us assume that the reservoir is located at node 1 of the pipe. Then exactly the same procedure as for the deadend can followed, except that dof h_1 is condensed away and dofs (v_1, h_2, v_2) are retained.

5.5 Valves

The rate at which valves are opened and closed determines the magnitude of transients within a pipe network. The following assumptions are made in the derivation of the valve equations:

- Fluid that flows through the valve is incompressible.
- The inertial effects of the accelerating and decelerating flow through the valve is negligible.

The valve equation is based on the steady-state orifice equation

$$v = C_d \sqrt{2g\Delta H} \quad (5.4)$$

where

- C_d being the discharge coefficient, which varies with the valve position (degree of openness). See figure 5.1.
- ΔH is the loss of head across the valve (The head needed to push the required flow through the valve).
- g is gravity acceleration

Two simple valve configurations are : In-line valves and reservoir valves.

5.5.1 In-line Valve

Both ends of an in-line valve are coupled to an adjacent pipe. This means that an in-line valve boundary condition, is applied to two pipe elements. Assume that the in-line valve is connected to two pipes: A and B. The valve is connected to pipe A's second node and to pipe B's first node. If v_{2a} and h_{2a} denotes the velocity and head dofs of pipe A's second

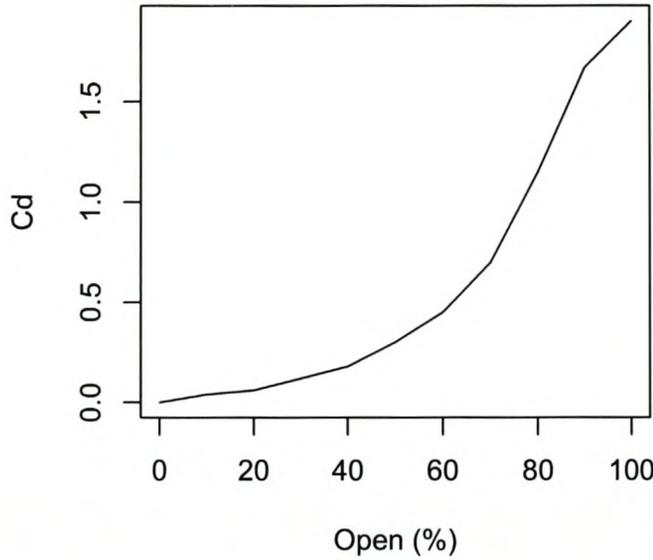


Figure 5.1: Valve Discharge coefficient vs Opening size

node and v_{1b} and h_{1b} denotes the velocity and head dofs of pipe B's first node, then the in-line valve equations applicable are:

$$v_{2a} = +C_d \sqrt{2g(h_{2a} - h_{1b})} \quad h_{2a} > h_{1b} \quad (5.5)$$

$$v_{2a} = -C_d \sqrt{2g(h_{1b} - h_{2a})} \quad h_{1b} > h_{2a} \quad (5.6)$$

$$v_{1b} = v_{2a} \quad (5.7)$$

Pipe A's v_{2a} dof is condensed, while dofs (h_{1a}, v_{1a}, h_{2a}) are retained. Pipe B's v_{1b} dof is condensed, while dofs (h_{1b}, h_{2b}, v_{2b}) are retained. For simplicity let us assume that $h_{2a} > h_{1b}$. Then the approximate flow fields at any point within pipe element A is given by

$$h = h_{1a}n_1 + h_{2a}n_2 \quad (5.8)$$

$$v = v_{1a}n_1 + C_d \sqrt{2g(h_{2a} - h_{1b})}n_2 \quad (5.9)$$

Where n_1 and n_2 are the shape functions defined in equation 4.29. Substitution of equations 5.8 and 5.9 into equation 4.90 gives the condensed residual of the constrained pipe element \bar{f}_r^{n+1} . Further, the condensed Jacobian \bar{j}_r^{n+1} of the constrained pipe element is

found by substituting the above residual into equation 4.104 with $\bar{\nabla}_{\bar{\phi}_r}$ defined as

$$\bar{\nabla}_{\bar{\phi}_r} = \left\{ \begin{array}{c} \frac{\partial}{\partial h_{1a}} \\ \frac{\partial}{\partial v_{1a}} \\ \frac{\partial}{\partial h_{2a}} \end{array} \right\} \quad (5.10)$$

The condensed residual and Jacobian of pipe element B can be derived in the same manner as pipe element A above.

5.5.2 Reservoir valve

Only one end of a reservoir valve is connected to a pipe element (say pipe A), while the other end is connected to a reservoir supplying a constant head of h_{rsvr} . If the first node of pipe A is connected to the reservoir valve then the differential in head across the valve is the difference between h_{rsvr} and h_{1a} . The reservoir valve equations applicable are:

$$v_{1a} = +C_d \sqrt{2g(h_{rsvr} - h_{1a})} \quad h_{rsvr} > h_{1a} \quad (5.11)$$

$$v_{1a} = -C_d \sqrt{2g(h_{1a} - h_{rsvr})} \quad h_{1a} > h_{rsvr} \quad (5.12)$$

$$(5.13)$$

Pipe A's v_{1a} dof is condensed, while dofs (h_{1a}, h_{2a}, v_{2a}) are retained. For simplicity let us assume that $h_{rsvr} > h_{1a}$. Then the approximate flow fields at any point within pipe element A is given by

$$h = h_{1a}n_1 + h_{2a}n_2 \quad (5.14)$$

$$v = C_d \sqrt{2g(h_{rsvr} - h_{1a})}n_1 + v_{2a}n_2 \quad (5.15)$$

Substitution of equations 5.14 and 5.15 into equation 4.90 gives the condensed residual of the constrained pipe element \bar{f}_r^{n+1} . Furthermore, the condensed Jacobian \bar{j}_r^{n+1} of the constrained pipe element is found by substituting the above residual into equation 4.104 with $\bar{\nabla}_{\bar{\phi}_r}$ defined as

$$\bar{\nabla}_{\bar{\phi}_r} = \left\{ \begin{array}{c} \frac{\partial}{\partial h_{1a}} \\ \frac{\partial}{\partial h_{2a}} \\ \frac{\partial}{\partial v_{2a}} \end{array} \right\} \quad (5.16)$$

5.6 Pumps

Pumps can cause transients during startup or power failures. Two simple pump configurations are: booster pumps (in-line pumps) and reservoir pumps (pumps with a constant upstream pressure). The equations governing a pump's behaviour during power failure will be shown in subsection 5.6.1. Later in subsections 5.6.2 and 5.6.3 it is shown how these equations can be extended to pumps in series and parallel.

5.6.1 Single pump-trip equations

Both ends of a booster pump are coupled to an adjacent pipe. This means that a booster pump boundary condition, is applied to two pipe elements. Assume that the booster pump is connected to two pipes: A and B. The pump is connected to pipe A's second node and to pipe B's first node. Therefore, v_{2a} and h_{2a} denotes the velocity and head dofs of pipe A's second node and v_{1b} and h_{1b} denotes the velocity and head dofs of pipe B's first node.

The following assumptions are made in the derivation of the pump-trip equations:

- Rotary inertia of moving parts are significant.
- The steady state characteristics also holds for the unsteady state situations.
- Pump similitude laws are valid (i.e. dynamic similarity holds for identical pumps).

The behaviour of the pump is governed by three equations:

- Head equilibrium (A continuity equation)
- Torque equilibrium (An equation of motion)
- Flow continuity (A continuity equation)

The main variables used to describe the behaviour of a pump are:

- **Q** Flow amount. ($\frac{m^3}{s}$). (Independent variable)
- **N** Pump speed. (rpm) (Independent variable)
- **H** Pressure head. (m) (Dependent variable)
- **T** Pump torque. (kNm) (Dependent variable)

The dependent variables (**H** and **T**) can be solved at any moment, if we know the independent variables (**Q** and **N**). To simplify our work we will rewrite the main pump variables, mentioned above, in dimensionless form

$$\nu = \frac{\mathbf{Q}}{\mathbf{Q}_{rated}} = \frac{v}{v_{rated}} \quad (5.17)$$

$$\alpha = \frac{\mathbf{N}}{\mathbf{N}_{rated}} \quad (5.18)$$

$$\gamma = \frac{\mathbf{H}}{\mathbf{H}_{rated}} \quad (5.19)$$

$$\beta = \frac{\mathbf{T}}{\mathbf{T}_{rated}} \quad (5.20)$$

where "rated" indicates values at optimum pumping efficiency.

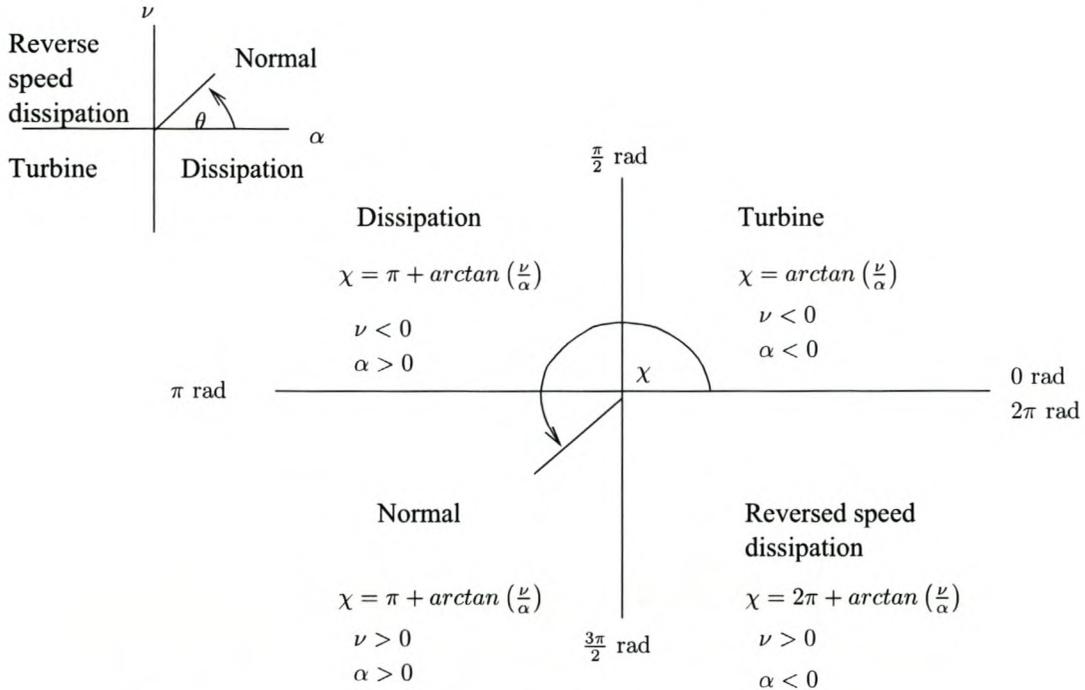


Figure 5.2: χ -formulas for different zones of pump operation

Test data for a pump can be obtained from manufacturers and then be transposed into the useful non-dimensional forms given below:

- $\frac{\gamma}{\alpha^2 + \nu^2}$ is tabulated against $\chi(\nu, \alpha)$ where $\chi \in \{0; 2\pi\}$ is found in figure 5.2
- $\frac{\beta}{\alpha^2 + \nu^2}$ is tabulated against $\chi(\nu, \alpha)$ where $\chi \in \{0; 2\pi\}$ is found in figure 5.2

A computer implementation will read in the manufacturer's test data points, and then use some sort of interpolation scheme for in-between values. For simplicity we will define $\frac{\gamma}{\alpha^2 + \nu^2}$ as $WH(\chi)$ and $\frac{\beta}{\alpha^2 + \nu^2}$ as $WT(\chi)$.

Head equilibrium

The head equilibrium equation assumes that the sum of the heads/pressures across the pump is zero.

$$h_{1b} = h_{2a} + \mathbf{H} \quad (5.21)$$

with

$$\mathbf{H} = \mathbf{H}_{rated} \gamma = \mathbf{H}_{rated} (\alpha^2 + \nu^2) WH(\chi) \quad (5.22)$$

where

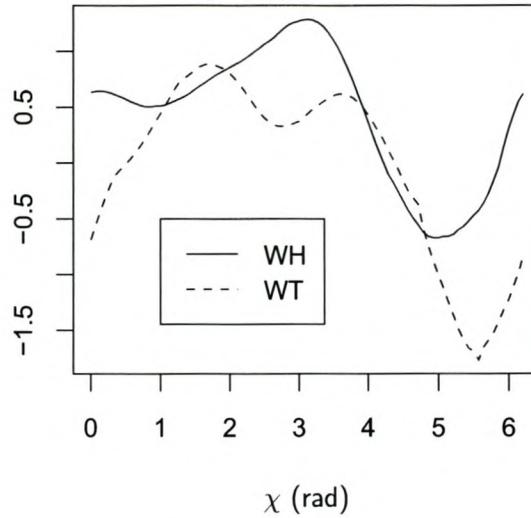


Figure 5.3: *Pump characteristic curves*

- h_{1b} is the head downstream of the pump
- h_{2a} is the head upstream of the pump
- \mathbf{H} is the head added by the pump (Total dynamic head)

The final form of the head equilibrium equation therefore reads

$$h_{1b} = h_{2a} + \mathbf{H}_{rated} (\alpha^2 + \nu^2) WH(\chi) \quad (5.23)$$

Torque equilibrium

The torque equilibrium equation assumes that the resultant torque is equal to the product of the inertia and the acceleration of the rotating parts.

$$\mathbf{T} = -I \frac{d\omega}{dt} \quad (5.24)$$

where

- \mathbf{T} is the pump torque
- ω is the angular velocity $[\frac{rad}{s}]$
- I is the inertia of the motor $[kg\ m^2]$

The unbalanced torque \mathbf{T} is approximated by an implicit θ -method, similar to the one used in section 4.4.7. θ was chosen as 0.5 with \mathbf{T}_0 and \mathbf{T}_p the respective torque values at

the start and end of the time-step, whose length is Δt . Equation 5.24 therefore becomes

$$\frac{1}{2}(\mathbf{T}_0 + \mathbf{T}_p) = -I \frac{d\omega}{dt} \quad (5.25)$$

$$(\mathbf{T}_0 + \mathbf{T}_p) \int_0^{\Delta t} dt = -2I \int_{\omega_0}^{\omega} d\omega \quad (5.26)$$

$$(\mathbf{T}_0 + \mathbf{T}_p) \Delta t = -2I (\omega - \omega_0) \quad (5.27)$$

Substituting the following into equation 5.27 (with the definitions of α and β given by equations 5.18 and 5.20 and the subscript "0" denoting the value at the start of the time-step)

$$\omega_0 = \mathbf{N}_{rated} \frac{2\pi}{60} \alpha_0 \quad (5.28)$$

$$\omega = \mathbf{N}_{rated} \frac{2\pi}{60} \alpha \quad (5.29)$$

$$\mathbf{T}_0 = \beta_0 \mathbf{T}_{rated} \quad (5.30)$$

$$\mathbf{T}_p = \beta \mathbf{T}_{rated} \quad (5.31)$$

gives

$$\beta = I \frac{\mathbf{N}_{rated}}{\mathbf{T}_{rated}} \frac{\pi}{15} \frac{(\alpha_0 - \alpha)}{\Delta t} - \beta_0 \quad (5.32)$$

Substituting $\beta = WT(\chi)(\alpha^2 + \nu^2)$ in equation 5.32 gives the torque equilibrium equation

$$WT(\chi) (\alpha^2 + \nu^2) - I \frac{\mathbf{N}_{rated}\pi}{15\mathbf{T}_{rated}\Delta t} (\alpha_0 - \alpha) + \beta_0 = 0 \quad (5.33)$$

Flow continuity

No flow is stored within the pump, therefore

$$v_{1b} = v_{2a} \quad (5.34)$$

Pipe coefficient equations

A single booster pump is connected to two nodes. Each node has two dofs i.e. (h_{2a}, v_{2a}) and (h_{1b}, v_{1b}) . As mentioned earlier there are two independent variables, which control the behaviour of a pump (α and ν). To determine the state of the pump therefore requires solving six unknowns $(h_{2a}, v_{2a}, h_{1b}, v_{1b}, \alpha, \nu)$. Up to this point four equations have been presented to solve the six unknowns (eq. 5.17, 5.23, 5.33, 5.34). Two additional equations are still required. The last two are obtained from equation 4.76 for the upstream and downstream pipes adjacent to the pump.

Expanding equation 4.76 gives

$$\begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & d_{34} \\ d_{41} & d_{42} & d_{43} & d_{44} \end{bmatrix} \begin{Bmatrix} h_1^{n+1} \\ v_1^{n+1} \\ h_2^{n+1} \\ v_2^{n+1} \end{Bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix} \begin{Bmatrix} h_1^n \\ v_1^n \\ h_2^n \\ v_2^n \end{Bmatrix} \quad (5.35)$$

since \bar{P} and $\bar{\phi}$ are known, the right hand side reduces to a known flux vector.

$$\begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & d_{34} \\ d_{41} & d_{42} & d_{43} & d_{44} \end{bmatrix} \begin{Bmatrix} h_1^{n+1} \\ v_1^{n+1} \\ h_2^{n+1} \\ v_2^{n+1} \end{Bmatrix} = \begin{Bmatrix} r_{h1} \\ r_{v1} \\ r_{h2} \\ r_{v2} \end{Bmatrix} \quad (5.36)$$

for clarity remove the superscript $n+1$

$$\begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & d_{34} \\ d_{41} & d_{42} & d_{43} & d_{44} \end{bmatrix} \begin{Bmatrix} h_1 \\ v_1 \\ h_2 \\ v_2 \end{Bmatrix} = \begin{Bmatrix} r_{h1} \\ r_{v1} \\ r_{h2} \\ r_{v2} \end{Bmatrix} \quad (5.37)$$

Upstream pipe equation (Pipe A): For the upstream pipe, the last equation in equation 5.37 is used.

$$d_{41}h_{1a} + d_{42}v_{1a} + d_{43}h_{2a} + d_{44}v_{2a} = r_{v2} \quad (5.38)$$

Making h_{2a} the subject of the equation gives the first of the two additional equations

$$h_{2a} = \frac{1}{d_{43}} (-d_{41}h_{1a} - d_{42}v_{1a} - d_{44}v_{2a} + r_{v2}) \quad (5.39)$$

Downstream pipe equation (Pipe B): For the downstream pipe, the first equation in the system defined by equation 5.37 is used.

$$d_{11}h_{1b} + d_{12}v_{1b} + d_{13}h_{2b} + d_{14}v_{2b} = r_{h1} \quad (5.40)$$

Making h_{1b} the subject of the equation gives the second of the two additional equations

$$h_{1b} = \frac{1}{d_{11}} (-d_{12}v_{1b} - d_{13}h_{2b} - d_{14}v_{2b} + r_{h1}) \quad (5.41)$$

Solving the pumptrip equations by a local Newton-Raphson iterative scheme

The six unknowns ($h_{2a}, v_{2a}, h_{1b}, v_{1b}, \alpha, \nu$) are solved by six equations (eq. 5.17, 5.23, 5.33, 5.34, 5.41 and 5.39, which are summarised below

$$v_{1b} = v_{2a} \quad (5.42)$$

$$v = v_{rated}\nu \quad (5.43)$$

$$h_{1b} = h_{2a} + \mathbf{H}_{rated}(\alpha^2 + \nu^2) WH(\chi) \quad (5.44)$$

$$WT(\chi)(\alpha^2 + \nu^2) - I \frac{\mathbf{N}_{rated}\pi}{15\mathbf{T}_{rated}\Delta t}(\alpha_0 - \alpha) + \beta_0 = 0 \quad (5.45)$$

$$h_{2a} = \frac{1}{d_{43}}(-d_{41}h_{1a} - d_{42}v_{1a} - d_{44}v_{2a} + r_{v2}) \quad (5.46)$$

$$h_{1b} = \frac{1}{d_{11}}(-d_{12}v_{1b} - d_{13}h_{2b} - d_{14}v_{2b} + r_{h1}) \quad (5.47)$$

Substituting equations 5.42, 5.43 and the two pipe coefficient equations (eq. 5.46 and 5.47) into equations 5.44 and 5.45 produces two non-linear equations, which can only be solved iteratively using a scheme such as Newton-Raphson.

$$\frac{1}{d_{43}}(r_{v2} - d_{44}v_{rated}\nu - d_{41}h_{1a} - d_{42}v_{1a}) - \frac{1}{d_{11}}(r_{h1} - d_{12}v_{rated}\nu - d_{13}h_{2b} - d_{14}v_{2b}) + WH(\chi)(\alpha^2 + \nu^2)\mathbf{H}_{rated} = 0 \quad (5.48)$$

$$WT(\chi)(\alpha^2 + \nu^2) + \beta_0 - C(\alpha_0 - \alpha) = 0 \quad (5.49)$$

with

$$C = I \frac{\mathbf{N}_{rated}\pi}{15\mathbf{T}_{rated}\Delta t} \quad (5.50)$$

The pump-trip boundary condition uses its own Newton-Raphson solver for solving these two non-linear equations (It is possible to implement the two equations into the global Newton-Raphson solver, but since these two equations are strongly non-linear, it is computationally more efficient to solve the 'local' two by two coefficient equation system). The rest of this section describes how to obtain accurate residuals (for the two equations) and their corresponding derivatives for the local Newton-Raphson scheme.

Usually pump manufacturers provide a finite set of data points for WH and WT. It is important to use an accurate interpolation scheme when only a limited amount of data points is available, since convergence of the Newton-Raphson scheme depends on accurate derivatives. A least-squares Fourier series was used to approximate $WH(\chi)$ and $WT(\chi)$

[1].

$$WH(\chi) = \frac{A_0}{2} + \sum_{j=1}^m (A_j \cos j\chi + B_j \sin j\chi) \quad (5.51)$$

$$WT(\chi) = \frac{A_0}{2} + \sum_{j=1}^m (A_j \cos j\chi + B_j \sin j\chi) \quad (5.52)$$

where A_j, B_j are Fourier coefficients given in table 5.6.1. These coefficients describe a pump with a specific speed of 35 (SI units).

The advantage of using a Fourier series is that its derivatives are easily obtained.

$$\frac{\partial WH}{\partial \chi} = \sum_{j=1}^m (-A_j j \sin j\chi + B_j j \cos j\chi) \quad (5.53)$$

$$\frac{\partial WT}{\partial \chi} = \sum_{j=1}^m (-A_j j \sin j\chi + B_j j \cos j\chi) \quad (5.54)$$

The following equations were used to generate the the coefficients A_j, B_j for a least-squares Fourier series:

$$A_j = \frac{1}{L} \sum_{i=1}^{2L} F_i \cos j\chi_i \quad (5.55)$$

$$B_j = \frac{1}{L} \sum_{i=1}^{2L} F_i \sin j\chi_i \quad (5.56)$$

where (χ_i, F_i) are data points, obtained from pump manufacturers

The two non-linear equations 5.48 and 5.49, with α and ν as the only unknowns, are solved through Newton-Raphson iteration

$$\begin{Bmatrix} \alpha \\ \nu \end{Bmatrix}^{k+1} = \begin{Bmatrix} \alpha \\ \nu \end{Bmatrix}^k - \left(\begin{bmatrix} \frac{\partial R_{WH}}{\partial \alpha} & \frac{\partial R_{WH}}{\partial \nu} \\ \frac{\partial R_{WT}}{\partial \alpha} & \frac{\partial R_{WT}}{\partial \nu} \end{bmatrix}^k \right)^{-1} \cdot \begin{Bmatrix} R_{WH} \\ R_{WT} \end{Bmatrix}^k \quad (5.57)$$

or simply

$$\bar{u}^{k+1} = \bar{u}^k - \left((\bar{R} \bar{\nabla}_{\bar{u}})^k \right)^{-1} \cdot \bar{R}^k \quad (5.58)$$

where the residuals of equations 5.48 and 5.49 are

$$R_{WH} = \frac{1}{d_{43}} (r_{v2} - d_{44} v_{rated} \nu - d_{41} h_{1a} - d_{42} v_{1a}) - \frac{1}{d_{11}} (r_{h1} - d_{12} v_{rated} \nu - d_{13} h_{2b} - d_{14} v_{2b}) + WH(\chi) (\alpha^2 + \nu^2) \mathbf{H}_{rated} \quad (5.59)$$

$$R_{WT} = WT(\chi) (\alpha^2 + \nu^2) + \beta_0 - C (\alpha_0 - \alpha) \quad (5.60)$$

Table 5.1: *WH and WT Fourier coefficients for a pump with a specific speed of 35 (SI units)*

| j | A_{WH} | B_{WH} | A_{WT} | B_{WT} |
|-----|-----------------|-----------------|-----------------|-----------------|
| 0 | 8.71159090e-01 | 0.00000000e+00 | -4.38636400e-02 | 0.00000000e+00 |
| 1 | -4.47933490e-01 | 5.71955345e-01 | -6.93199750e-01 | 6.80745913e-01 |
| 2 | 4.15500240e-01 | 1.20402879e-01 | -1.42992420e-01 | 4.72949111e-01 |
| 3 | 5.33410400e-02 | -5.37834401e-02 | 1.48393900e-01 | 1.78954755e-03 |
| 4 | 5.62422000e-02 | 2.37557743e-02 | 9.48542300e-02 | 6.25092864e-02 |
| 5 | 1.49694300e-02 | -7.29799554e-03 | 3.97621000e-02 | -1.19597601e-02 |
| 6 | 5.97240000e-03 | -6.17257368e-03 | 1.27662700e-02 | 1.90232788e-03 |
| 7 | 2.82290000e-03 | -1.16669814e-02 | 6.53027000e-03 | 1.27629764e-02 |
| 8 | -9.88351000e-03 | -1.44912950e-02 | 8.83786000e-03 | 2.09535030e-03 |
| 9 | -1.68786100e-02 | -8.76135502e-03 | 7.67048000e-03 | 1.93717669e-03 |
| 10 | -1.41355000e-02 | -5.19973455e-03 | 9.25291000e-03 | 6.94022326e-03 |
| 11 | -1.39485600e-02 | -6.58374573e-03 | 1.78455600e-02 | 7.16630181e-03 |
| 12 | -1.86265600e-02 | -3.30456304e-03 | 2.12019700e-02 | 1.53978370e-03 |
| 13 | -1.54946300e-02 | -9.59258902e-04 | 1.54385900e-02 | -6.45409081e-03 |
| 14 | -1.51934700e-02 | 3.75783076e-04 | 1.30334800e-02 | -2.48073645e-03 |
| 15 | -1.45470200e-02 | -1.50512736e-03 | 1.80898800e-02 | 1.04007862e-03 |
| 16 | -1.43418800e-02 | 1.29547083e-04 | 2.07219500e-02 | -1.42490419e-03 |
| 17 | -1.55764400e-02 | -1.65149218e-03 | 1.32148700e-02 | -4.24413805e-03 |
| 18 | -1.57036700e-02 | -2.67165969e-04 | 1.09998600e-02 | 2.57616542e-04 |
| 19 | -1.53273100e-02 | 3.08152063e-04 | 1.62238800e-02 | 2.53961155e-03 |
| 20 | -1.33947500e-02 | -1.45771119e-04 | 1.97847800e-02 | 1.72931321e-03 |

The Jacobian is defined as

$$\left[\begin{array}{cc} \frac{\partial R_{WH}}{\partial \alpha} \text{ total} & \frac{\partial R_{WH}}{\partial \nu} \text{ total} \\ \frac{\partial R_{WT}}{\partial \alpha} \text{ total} & \frac{\partial R_{WT}}{\partial \nu} \text{ total} \end{array} \right]^k = \left[\begin{array}{cc} \frac{\partial R_{WH}}{\partial \chi} \frac{\partial \chi}{\partial \alpha} + \frac{\partial R_{WH}}{\partial \alpha} & \frac{\partial R_{WH}}{\partial \chi} \frac{\partial \chi}{\partial \nu} + \frac{\partial R_{WH}}{\partial \nu} \\ \frac{\partial R_{WT}}{\partial \chi} \frac{\partial \chi}{\partial \alpha} + \frac{\partial R_{WT}}{\partial \alpha} & \frac{\partial R_{WT}}{\partial \chi} \frac{\partial \chi}{\partial \nu} + \frac{\partial R_{WT}}{\partial \nu} \end{array} \right]^k \quad (5.61)$$

with

$$\frac{\partial \chi}{\partial \alpha} = \frac{-\nu}{\nu^2 + \alpha^2} \quad (5.62)$$

$$\frac{\partial \chi}{\partial \nu} = \frac{\alpha}{\nu^2 + \alpha^2} \quad (5.63)$$

$$\frac{\partial R_{WH}}{\partial \alpha} = WH(\chi) \mathbf{H}_{rated} 2\alpha \quad (5.64)$$

$$\frac{\partial R_{WT}}{\partial \alpha} = WT(\chi) 2\alpha + C \quad (5.65)$$

$$\frac{\partial R_{WH}}{\partial \nu} = WH(\chi) 2\nu \mathbf{H}_{rated} + v_{rated} \left(\frac{d_{12}}{d_{11}} - \frac{d_{44}}{d_{43}} \right) \quad (5.66)$$

$$\frac{\partial R_{WT}}{\partial \nu} = WT(\chi) 2\nu \quad (5.67)$$

$$\frac{\partial R_{WH}}{\partial \chi} = \frac{\partial WH}{\partial \chi} (\alpha^2 + \nu^2) \mathbf{H}_{rated} \quad (5.68)$$

$$\frac{\partial R_{WT}}{\partial \chi} = \frac{\partial WT}{\partial \chi} (\alpha^2 + \nu^2) \quad (5.69)$$

After α and ν has been solved for the current time-step (n+1) and iteration (i), using the pump's Newton-Raphson iterative scheme (k=1,2,3...), the condensed residuals and Jacobians of the two constrained pipe element's can be computed as follows:

Pipe A's h_{2a} and v_{2a} dofs are condensed, while dofs h_{1a} and v_{1a} are retained. Pipe B's h_{1b} and v_{1b} dofs are condensed, while dofs h_{2b} and v_{2b} are retained. Then the approximate flow fields at any point within pipe element A is given by

$$h = h_{1a} n_1 + \frac{1}{d_{43}} (-d_{41} h_{1a} - d_{42} v_{1a} - d_{44} v_{2a} + r_{v2}) n_2 \quad (5.70)$$

$$v = v_{1a} n_1 + v_{rated} \nu n_2 \quad (5.71)$$

Substitution of equations 5.70 and 5.71 into equation 4.90 gives the condensed residual of the constrained pipe element \bar{f}_r^{n+1} . Furthermore, the condensed Jacobian \bar{j}_r^{n+1} of the constrained pipe element is found by substituting the above residual into equation 4.104 with $\bar{\nabla}_{\bar{\phi}_r}$ defined as

$$\bar{\nabla}_{\bar{\phi}_r} = \left\{ \begin{array}{c} \frac{\partial}{\partial h_{1a}} \\ \frac{\partial}{\partial v_{1a}} \end{array} \right\} \quad (5.72)$$

The condensed residual and Jacobian of pipe element B can be derived in the same manner as pipe element A above.

5.6.2 Series pump equations

Pumps can be considered to be in series, when the pipe length separating pumps are smaller than $a \Delta t$. With (a) being the wavespeed of the pipe.

Head equilibrium equation

The head equilibrium equation is extended to accomodate the \mathbf{H} and *valve head loss* of the second pump.

$$\mathbf{H}_s + \mathbf{H}_1 + \mathbf{H}_2 = \mathbf{H}_p \quad (5.73)$$

Torque equilibrium equations

An additional torque equilibrium equation appears for each additional pump, similar to equation 5.33.

5.6.3 Parallel pump equations

Head equilibrium equation

For each parallel line a head equilibrium equation is introduced, similar to equation 5.23.

Torque equilibrium equations

A torque equilibrium equation appears for each parallel pump, similar to equation 5.33. For multiple pumps, the computer program will need to automatically increase or decrease the number of pump equations when individual pumps fail or start.

Chapter 6

Stability Analysis

6.1 Introduction

This chapter focusses on the numerical stability and accuracy of the Petrov-Galerkin finite element, which was derived for unsteady pipeflow in chapter 4.

The chapter begins by showing that the computed solution and its error have the same numerical behaviour, because both satisfy the weak solution. Therefore, to predict the behaviour of numerical errors we only need an understanding of the numerical solution's behaviour. The chapter shows how the solution, at any given time level, can be approximated by a Fourier series. Fourier series of the velocity and head fields are constructed. These series are then substituted into the linearised forms of the governing equations, which in turn is recast into finite element form. If the amplitude of any Fourier component increases with time, the finite element scheme is unstable.

The value of a stability analysis lies in the ability to predict the range of values that various parameters can assume, without causing the scheme to become unstable or excessively inaccurate with respect to wave timing and magnitude. The chapter ends by discussing the amplitude and phase-shift plots obtained from such an analysis. An important point from this discussion is that the scheme does not become unstable for Courant numbers greater than one. It does however, become less accurate.

6.2 Behaviour of numerical solution and error

Consider the numerical solution obtained by our scheme

$$N = D + E \tag{6.1}$$

where

- N is the numerical solution

- D is the exact solution
- E is the error due to round-off errors

If N and D satisfy the weak solution then E must do so. This means that the numerical error and the numerical solution both possess the same growth property in time and either could be used to examine stability. Any perturbation of the input values at the n 'th time level will either be prevented from growing without bound for a stable system or will grow without bound for an unstable system. Stability analysis utilizes the superposition principle of a Fourier series. Unfortunately our pipeflow differential equations are quasi-linear and not linear. This means that if we substitute the Fourier series solution of v and h in our quasi-linear equations, then the superposition principle will not apply anymore. To retain the superposition property, we need our differential equations in a linearized form. The stability analysis done here, can therefore only partially indicate to us if the Petrov-Galerkin scheme is stable or not. It cannot predict instabilities associated with the nonlinearities of the problem.

6.3 Fourier series approximation of the solution

The velocity and head fields, at the current time level, can be written as a Fourier series (in complex exponential form) as follows:

$$h_j^n = \sum_{k=1}^M \zeta_k^n e^{ik\left(\frac{x}{L}\right)(j\Delta x)} \quad (6.2)$$

$$v_j^n = \sum_{k=1}^M \varrho_k^n e^{ik\left(\frac{x}{L}\right)(j\Delta x)} \quad (6.3)$$

$$N = \frac{2L}{\Delta x} \quad (6.4)$$

$$(6.5)$$

where

j is the node number

n is the current time level

k is the wave number (for a particular wavelength; k is the number of half-wavelengths which span the domain)

L is the length of the domain

Δx is the length of an element (it is assumed that the domain is discretised into elements of equal length)

N is the resolution with which Fourier components of our solution is described (i.e. the number of elements that a particular wavelength spans). In the present chapter short wavelengths refer to wavelengths spanning only a few elements, while long wavelengths span many elements.

ζ_k, ϱ_k are the Fourier coefficients, for wavenumber k of the series.

$$i = \sqrt{-1}$$

e is Euler's number (2.7183)

6.4 Linearization of the governing equations

This section shows how to linearise the unsteady pipe flow equations.

Let

$$h = h_0 + \Delta h \quad (6.6)$$

$$v = v_0 + \Delta v \quad (6.7)$$

6.4.1 Linearizing the continuity equation

Substitution of eq. 6.6 in eq. 2.63 gives

$$\begin{aligned} \frac{\partial h_0}{\partial t} + \frac{\partial \Delta h}{\partial t} + v_0 \frac{\partial h_0}{\partial x} + v_0 \frac{\partial \Delta h}{\partial x} + \Delta v \frac{\partial h_0}{\partial x} \\ + \Delta v \frac{\partial \Delta h}{\partial x} - v_0 \text{Sin}\alpha - \Delta v \text{Sin}\alpha + \frac{a^2}{g} \frac{\partial v_0}{\partial x} + \frac{a^2}{g} \frac{\partial \Delta v}{\partial x} = 0 \end{aligned} \quad (6.8)$$

Assuming h_0 satisfies eq. 2.63, then

$$\frac{\partial h_0}{\partial t} + v_0 \frac{\partial h_0}{\partial x} - v_0 \text{Sin}\alpha + \frac{a^2}{g} \frac{\partial v_0}{\partial x} = 0 \quad (6.9)$$

further, $\Delta v \frac{\partial \Delta h}{\partial x}$ is negligible, therefore eq. 6.8 simplifies to

$$\frac{\partial \Delta h}{\partial t} + \Delta v \frac{\partial h_0}{\partial x} + v_0 \frac{\partial \Delta h}{\partial x} - \Delta v \text{Sin}\alpha + \frac{a^2}{g} \frac{\partial \Delta v}{\partial x} = 0 \quad (6.10)$$

6.4.2 Linearizing the momentum equation

Substitution of eq. 6.7 in eq. 2.26 gives

$$\begin{aligned} \frac{\partial v_0}{\partial t} + \frac{\partial \Delta v}{\partial t} + g \frac{\partial h_0}{\partial x} + g \frac{\partial \Delta h}{\partial x} + v_0 \frac{\partial v_0}{\partial x} + \Delta v \frac{\partial v_0}{\partial x} \\ + v_0 \frac{\partial \Delta v}{\partial x} + \Delta v \frac{\partial \Delta v}{\partial x} + \frac{\lambda}{2D} v_0^2 + \frac{\lambda}{2D} v_0 \Delta v + \frac{\lambda}{2D} \Delta v^2 = 0 \end{aligned} \quad (6.11)$$

Assuming v_0 satisfies eq. 2.26, then

$$\frac{\partial v_0}{\partial t} + g \frac{\partial h_0}{\partial x} + v_0 \frac{\partial v_0}{\partial x} + \frac{\lambda}{D} v_0^2 = 0 \quad (6.12)$$

further, $\Delta v \frac{\partial \Delta v}{\partial x}$ and $\frac{\lambda}{D} \Delta v^2$ are negligible, therefore eq. 6.11 simplifies to

$$\frac{\partial \Delta v}{\partial t} + g \frac{\partial \Delta h}{\partial x} + \Delta v \frac{\partial v_0}{\partial x} + v_0 \frac{\partial \Delta v}{\partial x} + \frac{\lambda}{2D} v_0 \Delta v = 0 \quad (6.13)$$

Equations 6.10 and 6.13 can be written in tensor notation as

$$\frac{\partial \bar{\phi}}{\partial t} + \bar{A} \cdot \frac{\partial \bar{\phi}}{\partial x} + \bar{B} \cdot \bar{\phi} = 0 \quad (6.14)$$

with

$$\bar{\phi} = \begin{Bmatrix} \Delta h \\ \Delta v \end{Bmatrix} \quad (6.15)$$

$$\bar{A} = \begin{bmatrix} v_0 & \frac{a^2}{g} \\ g & v_0 \end{bmatrix} \quad (6.16)$$

$$\bar{B} = \begin{bmatrix} 0 & \frac{\partial h_0}{\partial x} - \text{Sin}\alpha \\ 0 & \frac{\partial v_0}{\partial x} + \frac{\lambda}{D} v_0 \end{bmatrix} \quad (6.17)$$

6.5 Petrov-Galerkin finite element formulation

The linearised equations obtained in the previous section are recast into finite element form in this section. Applying the Characteristic Dissipative Galerkin formulation

$$\langle \bar{N}^{*T}, \bar{R} \rangle = 0 \quad (6.18)$$

to eq. 6.14 gives a finite element very similar to 4.62

$$\bar{M} \cdot \frac{\partial \bar{\phi}}{\partial t} + \bar{C} \cdot \bar{\phi} = 0 \quad (6.19)$$

with

$$\bar{M} = \int_{-1}^{+1} \bar{N}^{*T} \cdot \bar{N} \frac{\Delta x}{2} d\xi \quad (6.20)$$

and

$$\bar{C} = \int_{-1}^{+1} \bar{N}^{*T} \cdot \left(\hat{A} \cdot \frac{\partial \bar{N}}{\partial \xi} + \hat{B} \cdot \bar{N} \frac{\Delta x}{2} \right) d\xi \quad (6.21)$$

Applying θ implicit time differencing to eq. 6.19 gives

$$\left\{ \bar{M} + \theta \Delta t \bar{C}^{n+1} \right\} \cdot \bar{\phi}^{n+1} = \left\{ \bar{M} + (\theta - 1) \Delta t \bar{C}^n \right\} \cdot \bar{\phi}^n \quad (6.22)$$

or simply

$$\bar{D} \cdot \bar{\phi}^{n+1} = \bar{P} \cdot \bar{\phi}^n \quad (6.23)$$

where n denotes the current time-step. This is a linear system, since matrices \bar{D} and \bar{P} contain only known values. These element matrices are then assembled to get a system matrix. An important point to note, is that equations 6.23 and 4.110 have the same form, and in fact is equivalent. This can be verified by expanding both equations. Therefore, the linearized equation system derived in this section is the same as the one obtained from the Newton-Raphson approximation. This shows that a stability analysis provides an indication of the convergence ability of the Newton-Raphson algorithm.

6.6 Fourier Stability analysis

In this section the Fourier series approximations of the velocity and head fields are substituted into the finite element formulated in the previous section, which was based on the linearized differential equations. This system of equations is then reshuffled in such a way that we can easily view the growth/decay of the Fourier coefficients.

A single finite element, having nodes j and $j+1$, of equation 6.23 can be written as

$$\bar{D} \cdot \begin{Bmatrix} \Delta h_j^{n+1} \\ \Delta v_j^{n+1} \\ \Delta h_{j+1}^{n+1} \\ \Delta v_{j+1}^{n+1} \end{Bmatrix} = \bar{P} \cdot \begin{Bmatrix} \Delta h_j^n \\ \Delta v_j^n \\ \Delta h_{j+1}^n \\ \Delta v_{j+1}^n \end{Bmatrix} \quad (6.24)$$

We then substitute Δh and Δv with equations equivalent to equation 6.2. We now consider how *any* component of the series grows/decays. For an arbitrary wavenumber (say $k = 1$ for simplicity) and defining $m = \frac{\pi}{L}$, we obtain

$$\bar{D} \cdot \begin{Bmatrix} \zeta^{n+1} e^{imj\Delta x} \\ \varrho^{n+1} e^{imj\Delta x} \\ \zeta^{n+1} e^{im(j+1)\Delta x} \\ \varrho^{n+1} e^{im(j+1)\Delta x} \end{Bmatrix} = \bar{P} \cdot \begin{Bmatrix} \zeta^n e^{imj\Delta x} \\ \varrho^n e^{imj\Delta x} \\ \zeta^n e^{im(j+1)\Delta x} \\ \varrho^n e^{im(j+1)\Delta x} \end{Bmatrix} \quad (6.25)$$

Expanding the terms in each vector gives

$$\bar{D} \cdot \begin{Bmatrix} \zeta^{n+1} e^{imj\Delta x} \\ \varrho^{n+1} e^{imj\Delta x} \\ \zeta^{n+1} e^{imj\Delta x} e^{im\Delta x} \\ \varrho^{n+1} e^{imj\Delta x} e^{im\Delta x} \end{Bmatrix} = \bar{P} \cdot \begin{Bmatrix} \zeta^n e^{imj\Delta x} \\ \varrho^n e^{imj\Delta x} \\ \zeta^n e^{imj\Delta x} e^{im\Delta x} \\ \varrho^n e^{imj\Delta x} e^{im\Delta x} \end{Bmatrix} \quad (6.26)$$

or, in expanded form

$$\begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & d_{34} \\ d_{41} & d_{42} & d_{43} & d_{44} \end{bmatrix} \cdot \begin{Bmatrix} \zeta^{n+1} e^{imj\Delta x} \\ \varrho^{n+1} e^{imj\Delta x} \\ \zeta^{n+1} e^{imj\Delta x} e^{im\Delta x} \\ \varrho^{n+1} e^{imj\Delta x} e^{im\Delta x} \end{Bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix} \cdot \begin{Bmatrix} \zeta^n e^{imj\Delta x} \\ \varrho^n e^{imj\Delta x} \\ \zeta^n e^{imj\Delta x} e^{im\Delta x} \\ \varrho^n e^{imj\Delta x} e^{im\Delta x} \end{Bmatrix} \quad (6.27)$$

Notice that the matrix equation above contains four equations, but only two unknowns (ζ_h, ϱ_v) for each time level. We can therefore reduce the the four equations to two, by back-substituting the third equation in eq. 6.27 into the first and the fourth into the second, to get

$$\begin{bmatrix} d_{11} + e^{im\Delta x} d_{13} & d_{12} + e^{im\Delta x} d_{14} \\ d_{21} + e^{im\Delta x} d_{23} & d_{22} + e^{im\Delta x} d_{24} \end{bmatrix} \cdot \begin{Bmatrix} \zeta^{n+1} e^{imj\Delta x} \\ \varrho^{n+1} e^{imj\Delta x} \end{Bmatrix} = \begin{bmatrix} p_{11} + e^{im\Delta x} p_{13} & p_{12} + e^{im\Delta x} p_{14} \\ p_{21} + e^{im\Delta x} p_{23} & p_{22} + e^{im\Delta x} p_{24} \end{bmatrix} \cdot \begin{Bmatrix} \zeta^n e^{imj\Delta x} \\ \varrho^n e^{imj\Delta x} \end{Bmatrix} \quad (6.28)$$

or simply

$$\bar{K} \cdot \begin{Bmatrix} \zeta^{n+1} e^{imj\Delta x} \\ \varrho^{n+1} e^{imj\Delta x} \end{Bmatrix} = \bar{R} \cdot \begin{Bmatrix} \zeta^n e^{imj\Delta x} \\ \varrho^n e^{imj\Delta x} \end{Bmatrix} \quad (6.29)$$

Eliminating $e^{imj\Delta x}$ on both sides gives

$$\bar{K} \cdot \begin{Bmatrix} \zeta^{n+1} \\ \varrho^{n+1} \end{Bmatrix} = \bar{R} \cdot \begin{Bmatrix} \zeta^n \\ \varrho^n \end{Bmatrix} \quad (6.30)$$

Premultiplying both sides with the inverse of \bar{K} , provides us with a suitable view of studying how the Fourier coefficients grows/decays from one timestep to the other.

$$\begin{Bmatrix} \zeta^{n+1} \\ \varrho^{n+1} \end{Bmatrix} = \bar{K}^{-1} \cdot \bar{R} \cdot \begin{Bmatrix} \zeta^n \\ \varrho^n \end{Bmatrix} \quad (6.31)$$

$\bar{\bar{G}} = \bar{\bar{K}}^{-1} \cdot \bar{\bar{R}}$ is known as the amplification matrix. The norm of the spectral radius $\rho(\bar{\bar{G}})$ of the amplification matrix, determines if the system will be stable or not. (The spectral radius of a matrix is the eigenvalue with the greatest norm.) A norm less than one is stable and a norm greater than one is unstable. The eigenvalues for $\bar{\bar{G}}$ are complex numbers, whose norm is defined as

$$|g| = \sqrt{Re(\rho(\bar{\bar{G}}))^2 + Im(\rho(\bar{\bar{G}}))^2} \quad (6.32)$$

The spectral radius of the amplification matrix can also be used to see how the speed of waves described by the scheme differs from their true counterparts. This is known as the phase shift between the computed and real waves. The phase angle of the numerical scheme is

$$\Phi = \left| \arctan \left(\frac{Im(\rho(\bar{\bar{G}}))}{Re(\rho(\bar{\bar{G}}))} \right) \right| \quad (6.33)$$

The phase angle of the exact solution is

$$\Phi_e = m\Delta x Courant = \frac{\pi}{L} a\Delta t \quad (6.34)$$

The phase shift is therefore

$$\frac{\Phi}{\Phi_e} = \frac{\left| \arctan \left(\frac{Im(\rho(\bar{\bar{G}}))}{Re(\rho(\bar{\bar{G}}))} \right) \right|}{m\Delta x Courant} \quad (6.35)$$

A phase shift value greater than one indicates that the Fourier component of the numerical solution has a wave speed faster than the correct solution. While a phase shift of less than one indicates that the numerical solution has a wavespeed slower than the exact solution.

The value of $N = 2$, is referred to as the "cutoff" or Nyquist number. This number corresponds to the shortest wavelength that can be resolved on a grid (the length of the domain is equal to the length of an element). This means that information is automatically lost at sub-grid scales of this number. This is a reason for doing a sensitivity analysis of a model on different mesh sizes, to determine at what resolution do we obtain grid-independence. The quality of a numerical scheme for a hyperbolic system can be determined by examining the scheme's amplitude and phase-shift properties. These properties are examined next.

6.7 Amplitude figure

Figure 6.1 plots the damping coefficient/amplification factor against N (eq. 6.4), for different Courant numbers and upwinding values (ω). Damping coefficients larger than one indicates numerical instability, while coefficients smaller than one indicates a stable system. Figure 6.1 shows that the scheme is stable for all selected Courant numbers and upwinding values.

Figures 6.1a,b,c shows that the damping coefficient of the Fourier components decreases with decreasing N . This means that the amplitudes of the short wavelengths (known for creating spurious oscillations, due to dispersive errors) are damped more by the scheme than the longer wavelengths. This is highly desired.

Since Courant numbers in excess of one do not produce an unstable system, we have the option to use large timesteps and save computational time. It comes at a cost, however, because some of the medium wavelengths will be unnecessarily damped, reducing the accuracy of the solution.

Comparison of figures 6.1a and b reveals that increasing the upwinding parameter, increases the selective numerical damping. However, comparing figures 6.1b and c reveals that large upwinding values, close to one, reduces the quality of the dissipative interface (weak damping is introduced for both the short and medium wavelengths).

6.8 Phase shift figure

Figure 6.2 plots the shift in phase between the exact solution and the numerical solution against N , for different Courant numbers and upwinding values (ω). The plot shows that the phase shift is at its greatest for small wavelengths. This means that short wavelengths have larger phase-shift errors than the longer wavelengths.

Figures 6.1a and 6.2a shows why the Bubnov-Galerkin scheme ($\omega = 0$) is dispersive. Phase shift errors are introduced for the short wavelengths, which are only weakly damped by the dissipative interface. Figure 6.2a also shows that the relative celerity is larger than one, for Courant numbers smaller than one. Relative celerities smaller than one are associated with Courant numbers greater than one.

Figure 6.2b shows the phase shift plot for an optimal upwinding value of $\omega = \frac{2}{\sqrt{(15)}}$. We can see that phase shift errors are restricted to short wavelengths, which the selective dissipative interface strongly damps.

Figure 6.2c shows that large upwinding values, close to one, introduces phase shift errors for both short and medium wavelengths. This undesirable result is hidden in part by the dissipative interface, which also becomes less selective for large upwinding values. It must be remembered that the focus of the scheme is to modify only the short wavelengths and not the larger ones, since it is only the short wavelengths that introduces spurious oscillations for solutions containing sharp gradients. Therefore, large upwinding values introduces additional problems, which are not accounted for by the scheme.

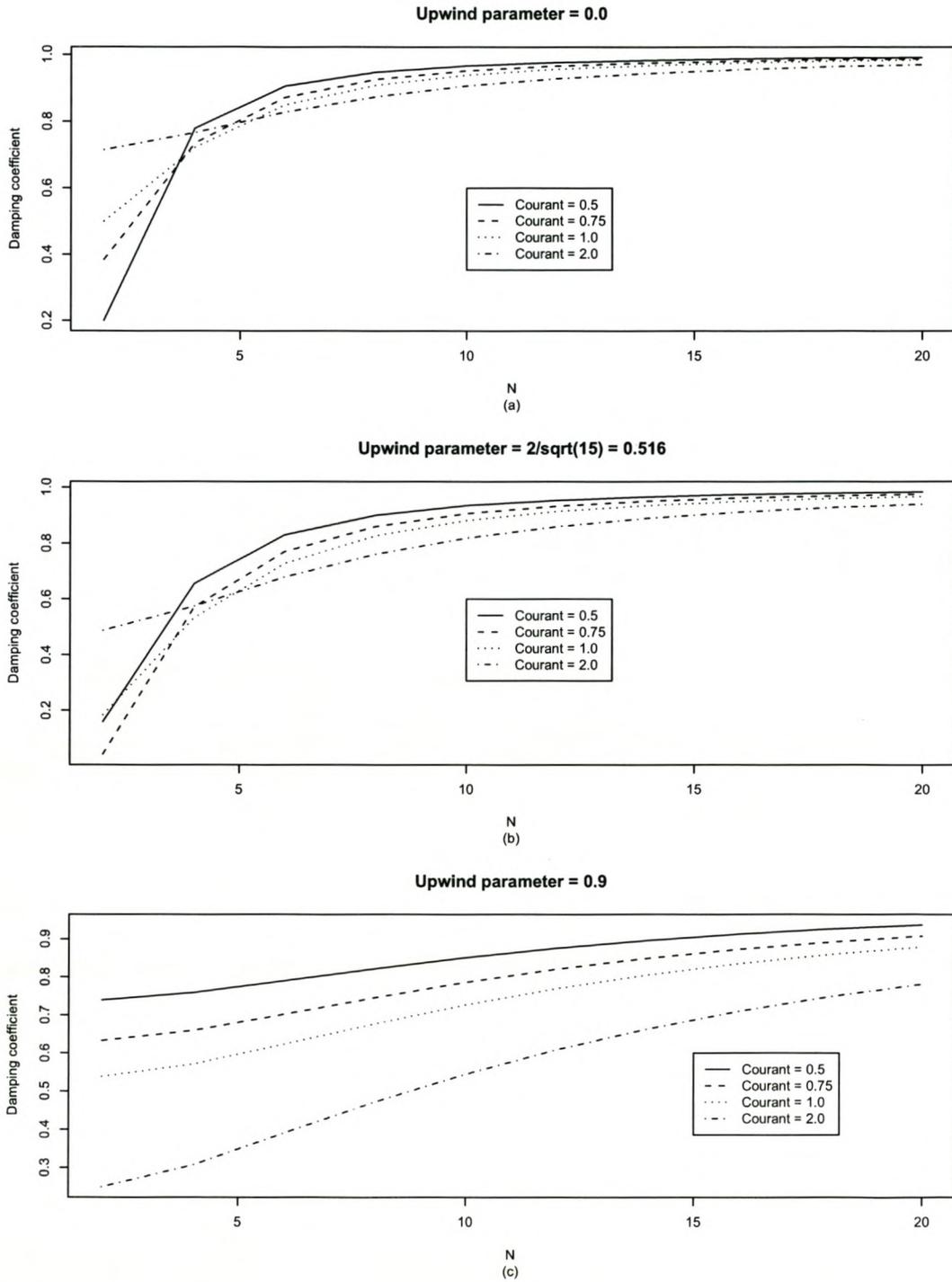


Figure 6.1: Artificial damping for various combinations of upwinding and Courant numbers

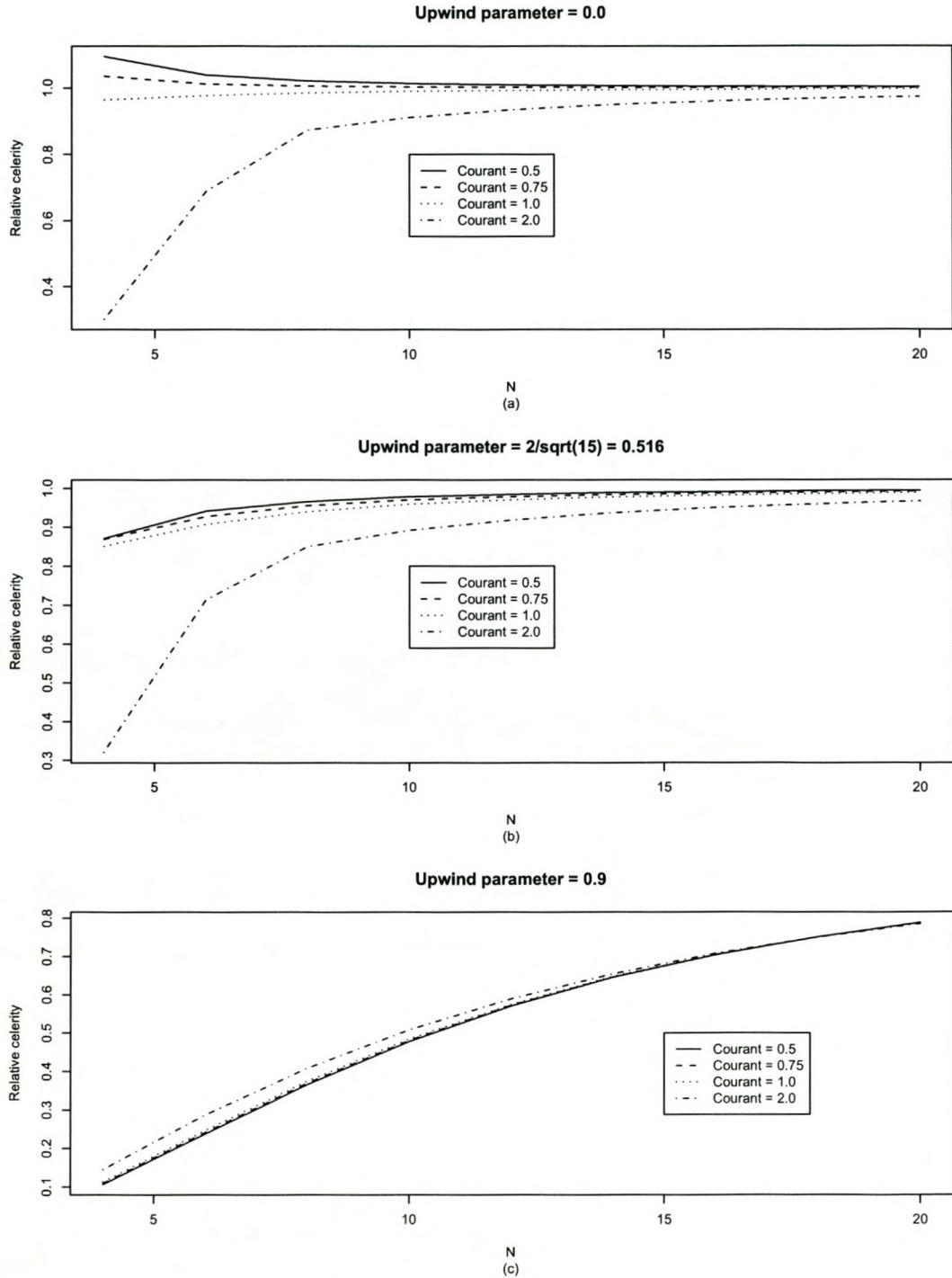


Figure 6.2: Phase shifts for various combinations of upwinding and Courant numbers

Chapter 7

Test Cases

7.1 Introduction

This chapter examines the results obtained from a pipeline model, operated under various conditions. The conditions were:

- Steady-state flow
- Rapid valve closure
- Pump power failure

The sensitivity of the results to:

- the Courant number
- the upwinding parameter
- the number of elements used to solve the problem

was studied. The computed results were compared to hydraulic analytic solutions, as well as documented experimental data. The finite element method gave good results in all the implemented models. For example, it predicted correctly the magnitude of the surge pressure, due to rapid valve closure. The period of the returning transient was also correct. It also proved successful in attaining steady-state conditions, given sufficient time to dampen out the excess energy through friction. It further showed to be robust enough to handle sensitive boundary conditions, such as a pump-trip. The method however, also has some weaknesses. Firstly, it is unable to conserve mass, momentum and energy precisely, although for all practical purposes the magnitude of the error is negligible. This was expected, since the governing equations used were in non-conservation form. Its second weakness is that it is unable to eliminate all the spurious oscillations, following a steep gradient or discontinuity. The results are however, significantly better than those of the Bubnov-Galerkin scheme and compares well with finite difference schemes, such as the Preismann scheme [10].

7.2 Steady-state model

7.2.1 Model objectives

The focus of this model is to determine whether the finite element scheme is able to approach the correct steady state flow, when the simulation is run long enough. The section is divided in the following parts:

- Flow problem statement
- Steady-state hand calculations, with which we can compare our computed results
- How to determine if the scheme is converging to *some* steady-state
- Conclusion that computed results agree with hand calculations
- Inability of scheme to conserve mass in the steady-state

7.2.2 Flow problem statement

We would like to model the steady state flow for a burst pipe. The problem can be simplified to two reservoirs connected by a 1000m long steel pipeline (Figure 7.1). The diameter, roughness and wavespeed of the pipe is 0.5m, 0.06mm and 1200m/s respectively. The upstream reservoir's operating level is 100m and the downstream reservoir's is 52.93m. Initially there is no flow, with a static hydraulic grade line of 100m. At time zero an isolating valve is instantly opened at the lower reservoir. What is the steady-state flow? In this problem the effect of cavitation has been ignored.

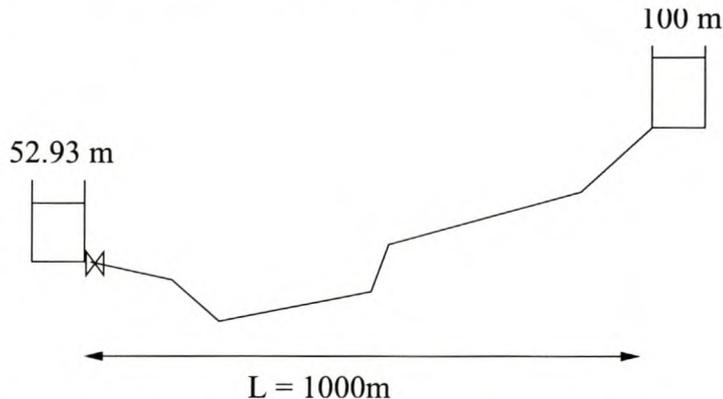


Figure 7.1: Two reservoirs connected by a steel pipeline

Table 7.1: *Analytical hydraulic grade line*

| Chainage (m) | Head (m) | Velocity (m/s) |
|--------------|----------|----------------|
| 0.0 | 52.93 | -5.97 |
| 100.0 | 57.65 | -5.97 |
| 200.0 | 62.38 | -5.97 |
| 300.0 | 67.11 | -5.97 |
| 400.0 | 71.83 | -5.97 |
| 500.0 | 76.56 | -5.97 |
| 600.0 | 81.29 | -5.97 |
| 700.0 | 86.01 | -5.97 |
| 800.0 | 90.74 | -5.97 |
| 900.0 | 95.47 | -5.97 |
| 1000.0 | 100.19 | -5.97 |

7.2.3 Steady-state hand calculations

The pressure differential between the two reservoirs is 47.07m. Using the Colebrook-White Darcy-Weisbach equation for pipes, the steady-state flow velocity is 5.97m/s. The Colebrook-White Barr equation provides us with a friction factor of $\lambda = 0.013$. This means that the friction headloss over a 100m pipe section is 4.73m. We can now determine the steady-state hydraulic grade line, with the calculated headloss. Table 7.2.3 gives the hydraulic grade line for every 100 meters.

The natural period of the pipeline at the downstream reservoir is given by

$$T_p = \frac{2L}{a} \quad (7.1)$$

$$= 1.67s \quad (7.2)$$

where

- a is the wavespeed
- L is the length of the pipe

This means that a transient caused at the downstream reservoir (e.g. a pipe burst) needs 1.67s to return to this reservoir.

7.2.4 Determining if the scheme has reached a steady-state

No variation in head and flow velocity should occur for steady flow, as time passes. This means that $\frac{\partial h}{\partial t} = \frac{\partial v}{\partial t} = 0$. However large values for the time derivatives may be expected

during the early parts of the simulation, because of the non-equilibrium initial conditions. It is therefore prudent to select initial conditions which are realistic and not too far from the steady-state conditions. Unrealistic initial conditions will create excessively large time derivatives which will cause the scheme to diverge. If our time derivatives are shrinking with time, we can be sure that the scheme is approaching a steady state. Rearranging equations 2.26 and 2.63 provides us with the equations for $\frac{\partial h}{\partial t}$, $\frac{\partial v}{\partial t}$.

$$\frac{\partial h}{\partial t} = -v \frac{\partial h}{\partial x} + v \sin \alpha - \frac{a^2}{g} \frac{\partial v}{\partial x} \quad (7.3)$$

$$\frac{\partial v}{\partial t} = -g \frac{\partial h}{\partial x} - v \frac{\partial v}{\partial x} - \frac{\lambda v |v|}{2D} \quad (7.4)$$

In order to solve these equations we require the values of v , $\frac{\partial v}{\partial x}$, $\frac{\partial h}{\partial x}$. Evaluation of $\frac{\partial v}{\partial x}$, $\frac{\partial h}{\partial x}$ at a node requires at least C^1 trial functions, however our trial functions are C^0 . This means that there is no interelement compatibility of first order (and higher) derivatives. We can therefore only evaluate $\frac{\partial v}{\partial x}$, $\frac{\partial h}{\partial x}$ within an element. The velocity v is given by equation 4.32

$$v = \frac{1 - \xi}{2} v_1 + \frac{1 + \xi}{2} v_2 \quad (7.5)$$

taking the spatial derivative (in global coordinates) gives

$$\frac{\partial v}{\partial x} = \frac{v_2 - v_1}{\Delta x} \quad (7.6)$$

similarly

$$\frac{\partial h}{\partial x} = \frac{h_2 - h_1}{\Delta x} \quad (7.7)$$

The most accurate values for v are obtained at $\xi = \frac{+1}{\sqrt{3}}$, which are the Gauss points for a second order rule, since two point Gauss quadrature were performed on the elements (the Gauss point of a first order rule would also have produced accurate results).

In figures 7.2a and 7.2b it can be seen that $\frac{\partial h}{\partial t}$, $\frac{\partial v}{\partial t}$ oscillate around zero. This oscillation is due to the expansion and compression of the water as the transient passes the node. The period of these oscillations corresponds to the pipe's natural period.

We also see that $\frac{\partial v}{\partial t}$, $\frac{\partial h}{\partial t}$ approach zero quickly during the early part of the simulation, but this rapid convergence slows down as the steady-state conditions are approached. This is due to the big potential difference, between steady-state and initial conditions, during the early parts of the simulation. This potential difference pushes the flow fields rapidly towards the steady-state solution. As the flow fields approach equilibrium, the potential difference diminishes, resulting in a slower convergence towards steady-state.

7.2.5 Verifying that the correct steady state was reached

Table 7.2.3 shows that at chainage 100m, $h = 57.65m$ and $v = -5.97m/s$.

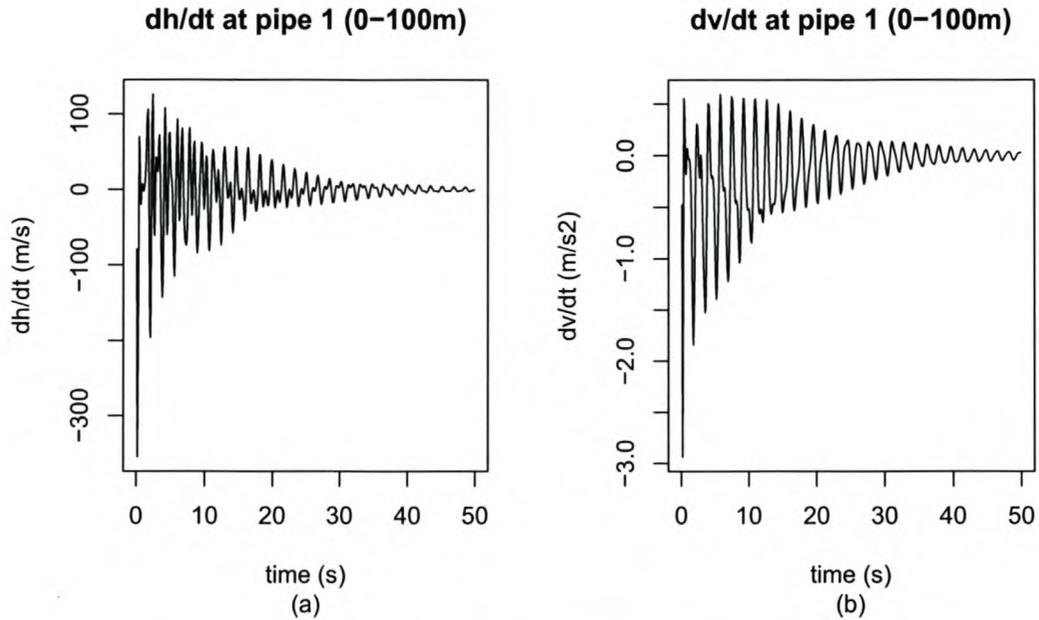


Figure 7.2: *Steady-state convergence of time derivatives*

Figures 7.3a and 7.3b indicates that h, v at this node approaches these tabulated values, with time.

Figure 7.4 shows that the computed and hand calculated steady-state hydraulic grade lines are almost identical. From this model we are able to conclude that if reasonable initial conditions are given, the finite element scheme reaches the correct steady state, given sufficient time.

7.2.6 Steady-state mass conservation

This section examines the ability of the scheme to conserve mass under steady flow conditions. In section 3.8 we mentioned that our scheme will be unable to conserve the flux variables, because our governing equations are written in non-conservation form. This means that our scheme will not conserve mass, momentum and energy exactly. This statement is verified in figure 7.5, where the steady-state velocities are plotted against chainage. It can be seen that for each pipe section, the inflow and outflow velocities are different. The pipeline has a constant diameter, therefore mass is being created/destroyed within the pipe. These velocity errors are however for all practical purposes negligible and the computed results are therefore satisfactory.

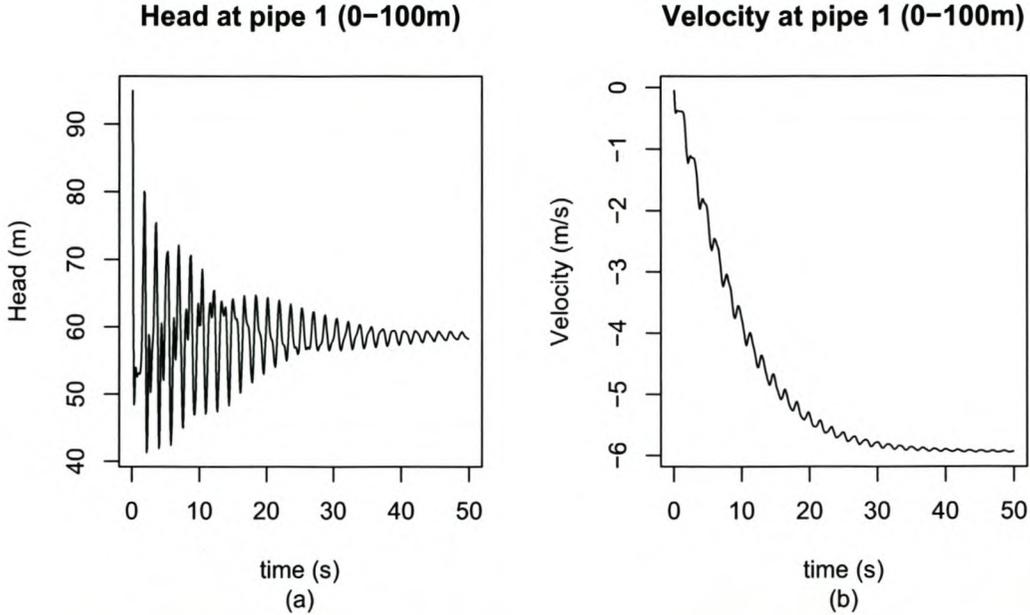


Figure 7.3: *Steady-state convergence of flow variables*

7.3 Rapid valve closure model

7.3.1 Model objectives

This model focusses on how accurately the finite element scheme can predict the magnitude and timing of a transient, caused by the rapid closure of a ball valve. By rapid closure we mean the time that it takes to close the valve, is equal or less than the time it takes for the transient to move through the pipeline and return to the valve. The computed results were compared to the Joukowsky maximum head equation as well as the experiments done by Streeter and Lai [14]. This section is divided in the following parts:

- Flow problem statement
- Joukowsky maximum head calculations, with which we can compare our computed results
- Comparison of the computed results with those of Streeter and Lai
- Courant number dependency
- Grid independence
- Upwinding parameter dependency

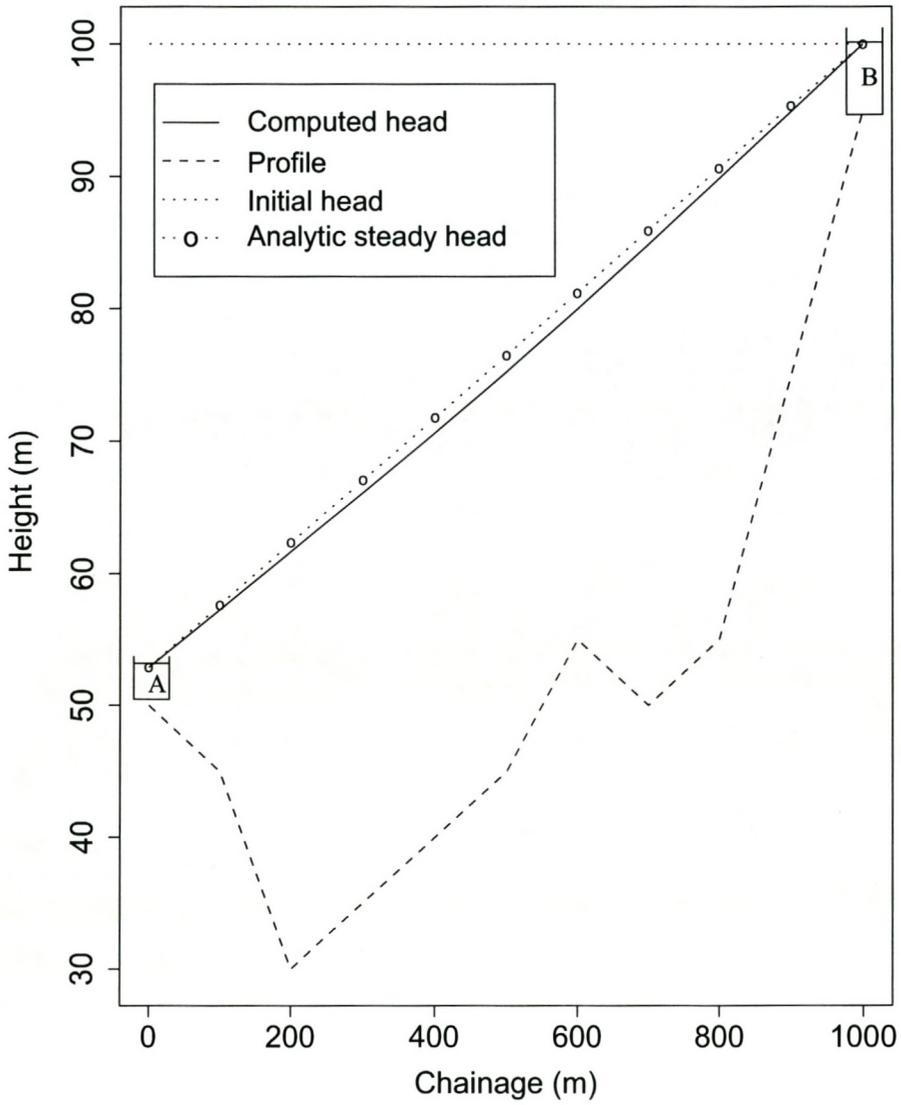


Figure 7.4: Initial- and steady-state hydraulic grade lines

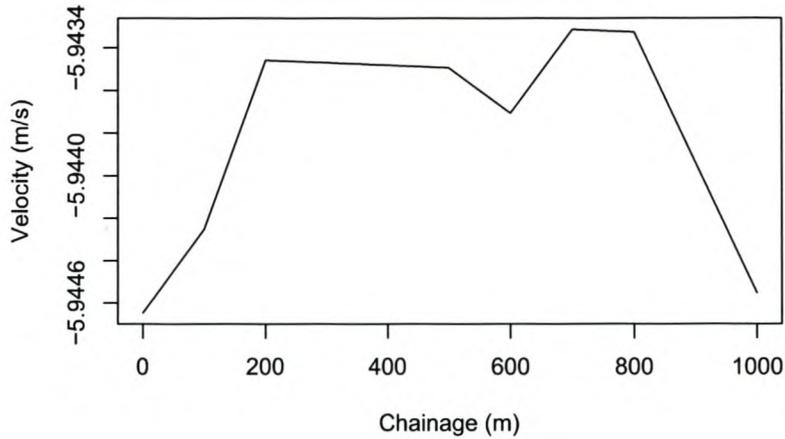


Figure 7.5: *Non-conservation of mass - Steady state solution after 3300 time steps*

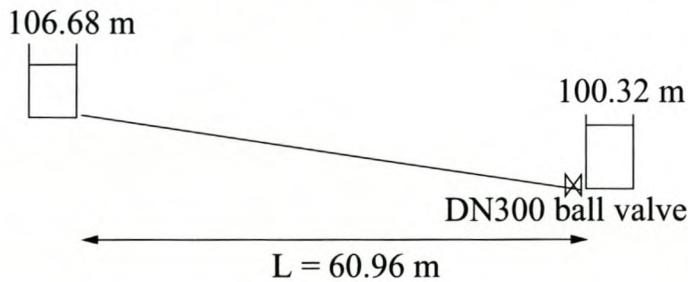


Figure 7.6: *Two reservoirs connected by a steel pipeline, with a ball valve situated at the downstream reservoir*

7.3.2 Flow problem statement

Two reservoirs are connected by a 60.96m long horizontal copper pipeline. The diameter, roughness and wavespeed of the pipeline is 0.011m, 0.003mm and 1355m/s respectively. The upstream reservoir's operating level is 106.68m, while the downstream reservoir operates at 100.32m. A DN300 ball valve is connected to the downstream reservoir. The ball valve is closed within 0.09s and the maximum pressure at valve closure is required. In this problem the effect of cavitation has been ignored.

7.3.3 Joukowsky maximum head calculation

The pressure differential between the two reservoirs is 6.36m. Using the Colebrook-White Darcy-Weisbach equation for pipes, the steady-state flow velocity is 0.84m/s. The natural

period of the pipeline at the ball valve is given by:

$$T_p = \frac{2L}{a} \quad (7.8)$$

$$= 0.09s \quad (7.9)$$

Joukowsky's maximum head equation may be used, since the time of valve closure is equal to the natural period (In practice to simulate fast valve-closure it is better to use a valve closing time less than 80 percent of the pipeline period). The maximum head at the valve will therefore be the sum of the steady-state head and the Joukowsky head. The Joukowsky head is

$$H_J = \frac{av_0}{g} \quad (7.10)$$

$$= 117m \quad (7.11)$$

where

- a is the wavespeed of the pipe (1355m/s)
- v_0 is the initial steady-state velocity (0.84m/s)
- g is the gravity acceleration constant

The maximum head is therefore 217m. This value is in good agreement with Streeter and Lai's work presented in figure 7.8, where the maximum head at 0.09s is 213m (700ft). These plots are based on simplified valve closure relationships and do not represent the real shape of a transient caused by the closure of a ball valve.

7.3.4 Comparison of the computed results with those of Streeter and Lai

In the present research, the flow domain was discretized into ten finite elements, each having a length of 6.096m. The Courant number, ω , Δt and θ were 1, $\frac{2}{\sqrt{15}}$, 0.0045s and 0.5 respectively.

The initial steady-state conditions were assumed to be zero flow, with a static hydraulic grade line of 106m. The simulation was run with the valve fully open all the time. The proper computed steady-state results were obtained from this simulation after 5400 timesteps (i.e. 24.4s). This steady-state solution was then used as the initial condition for the transient analysis, where the valve is closed within 0.09s. Figure 7.7 shows the computed variation in head with respect to time at the valve. This figure shows a small spurious oscillation at the moment the valve is fully closed (i.e the moment where maximum pressure is reached). Due to this spurious oscillation the maximum is overestimated by 11m head. A conservative five percent error. However, the finite element scheme

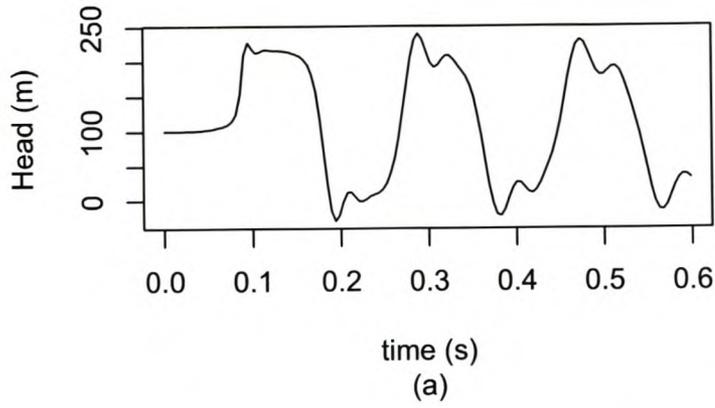


Figure 7.7: Head-time variation at valve

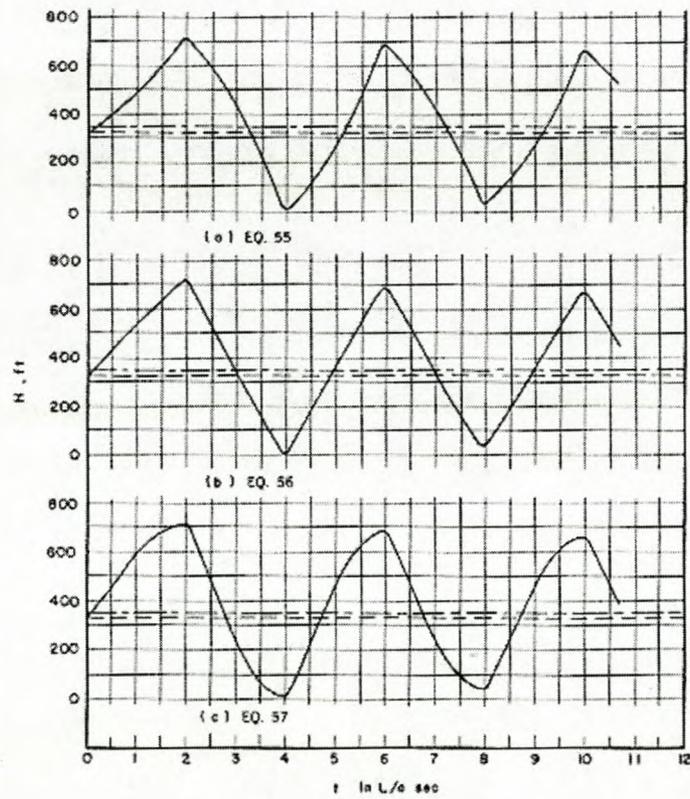


Figure 7.8: Computed valve head-time series by Streeter and Lai

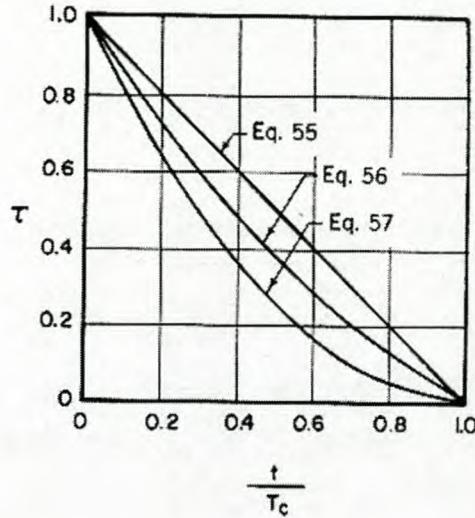


Figure 7.9: *Valve closure relationships implemented by Streeter and Lai*

quickly rectifies itself to the realistic value of 217m after the oscillation. This value agrees well with our hand calculation, as well as Streeter and Lai's example (figure 7.8). The time that it takes for the low pressure transient (a reservoir inverts a transient from high to low or vice-versa) to return to the valve corresponds to our hand-calculated value of 0.09s. The existence of the spurious oscillation can be ascribed to the sensitivity of the finite element scheme towards sharp gradients and discontinuities in the flow fields. From figure 7.7 we can see that each time the wavefront is reflected from a boundary another spurious oscillation is added to the previous one. The result is that the shape of the wave is degraded after repeated reflections off the boundaries. The consequence of this, is that if there existed multiple transients within a pipeline (due to pipe branches or cavitation) inaccurate superposition of two or more passing waves will be shown.

7.3.5 Courant number dependency

This section examines how dependent the solution is on the Courant number. A Courant number of one, indicates that the numerical scheme samples the domain of dependence precisely. Values of less than one indicates that the scheme samples all the data within the domain of dependence, but also from the surrounding regions. This means that the solution loses accuracy, because irrelevant data is being used. Courant numbers in excess of one indicates that the scheme samples only part of the domain of dependence. Information is lost. This data loss usually results in the numerical scheme becoming unstable. There are two reasons why we would want to use Courant numbers close to one. The first reason is accuracy. The second is that large Courant numbers use large

timesteps. Large timesteps reduces the computational time.

The effect of the Courant number was determined by comparing the maximum head at the valve for different timestep lengths. It was decided not to compare the spurious head at ($t = 0.09s$) with those of Streeter and Lai. Advantage was taken of the fact that the maximum pressure stays constant for a short period after the ball valve has closed. The maximum head was therefore sampled at 1.2s. Ten elements, each with length 6.096m were used. The scheme's parameters were $\omega = \frac{2}{\sqrt{15}}$ and $\theta = 0.5$. Figures 7.10a,b shows that the maximum head at 1.2s (behind the spurious oscillation) is close to the correct value. It can also be seen that for this particular problem, the scheme is largely independent of the Courant number and does not become unstable for Courant numbers in excess of one.

7.3.6 Grid independence

This section examines how sensitive the solution is to the number of elements used. The number of elements was varied, while the other parameters were held constant (Courant = 1, $\omega = \frac{2}{\sqrt{15}}$ and $\theta = 0.5$). The maximum head was also sampled at 1.2s.

From figure 7.11 we can see that the solutions for four and less elements differ significantly. The difference in the solution for seven or more elements are comparatively small. We can therefore say that our solution becomes independent of the number of elements implemented, once we start using more than approximately seven elements. It can also be seen that the computed maximum head at 1.2s, for a grid independent model, is close to the correct results.

7.3.7 Upwinding parameter dependency

This section examines how dependent the solution is on the upwinding parameter (ω). The upwinding parameter was varied, while the other parameters were held constant (Ten elements, Courant = 1 and $\theta = 0.5$). The maximum head was also sampled at 1.2s. It is interesting to note that an upwinding value of zero, simplifies the Petrov- Galerkin finite element scheme to that of a Bubnov-Galerkin one.

In section 4.3 it was mentioned that the Bubnov-Galerkin scheme is unable to handle sharp gradients in the flow fields and as a result excessively large spurious oscillations appear. This can clearly be seen in figure 7.12a.

Figure 7.12b shows that an upwinding coefficient of one introduces very strong, selective damping of the spurious oscillations. It would appear that a value of one is ideal, however such strong damping adversely affects the timing of the wave, due to the introduction of artificial viscosity. Figure 7.12c shows the damping for a theoretically optimal upwinding value of ($\omega = \frac{2}{\sqrt{15}}$). At this value mild, selective damping is introduced, but

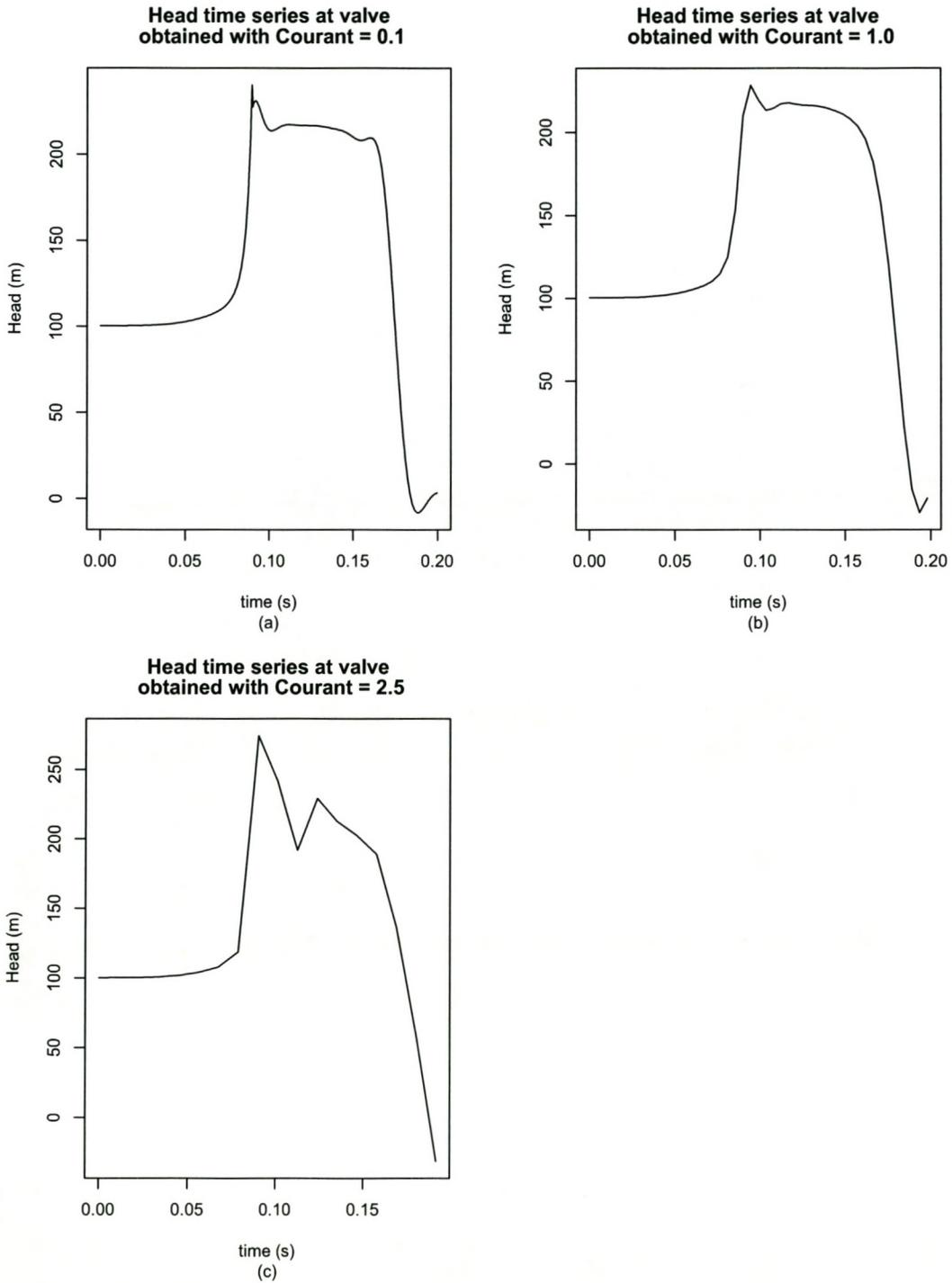


Figure 7.10: Courant number sensitivity analysis

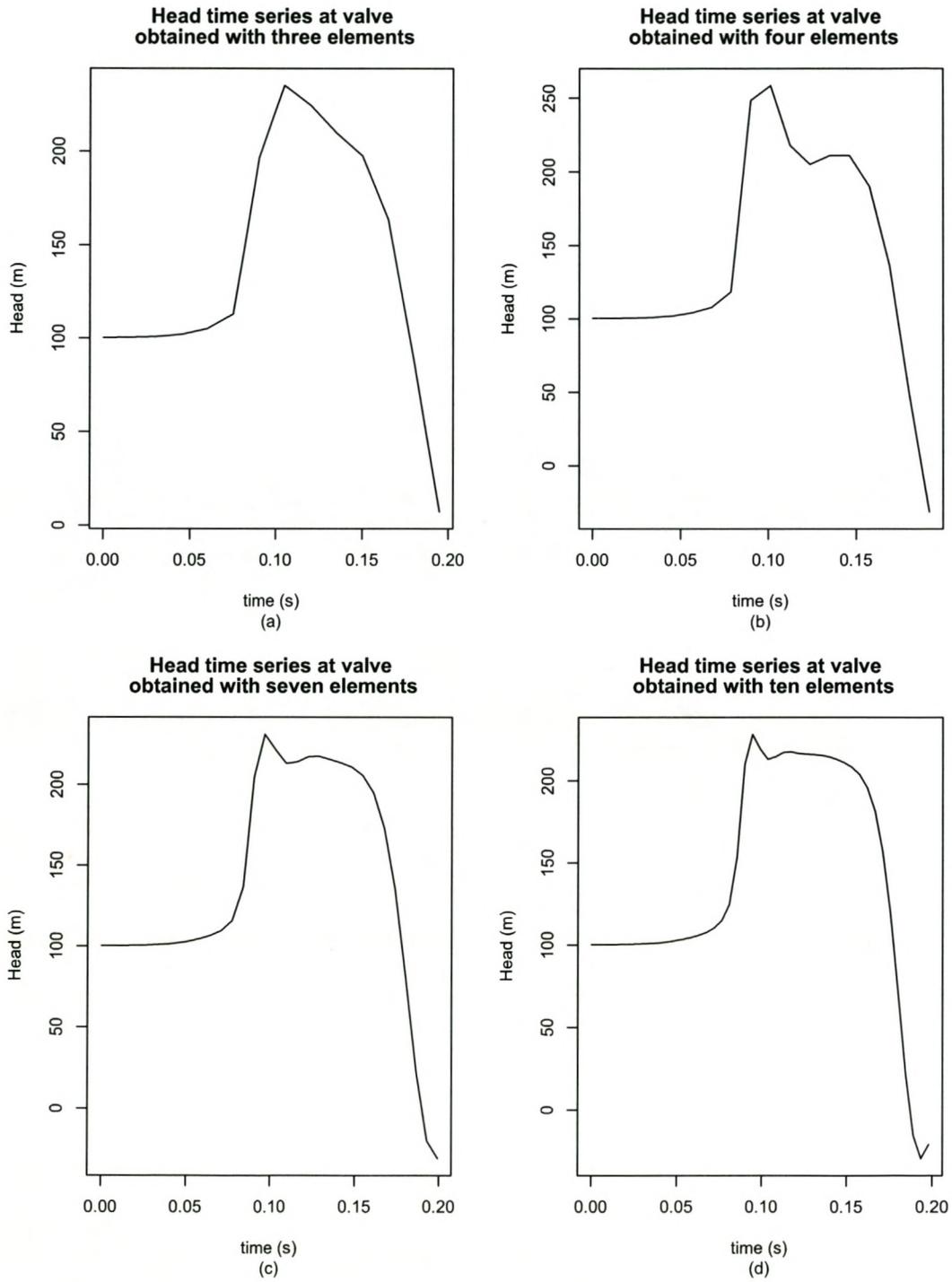


Figure 7.11: Mesh coarseness sensitivity analysis

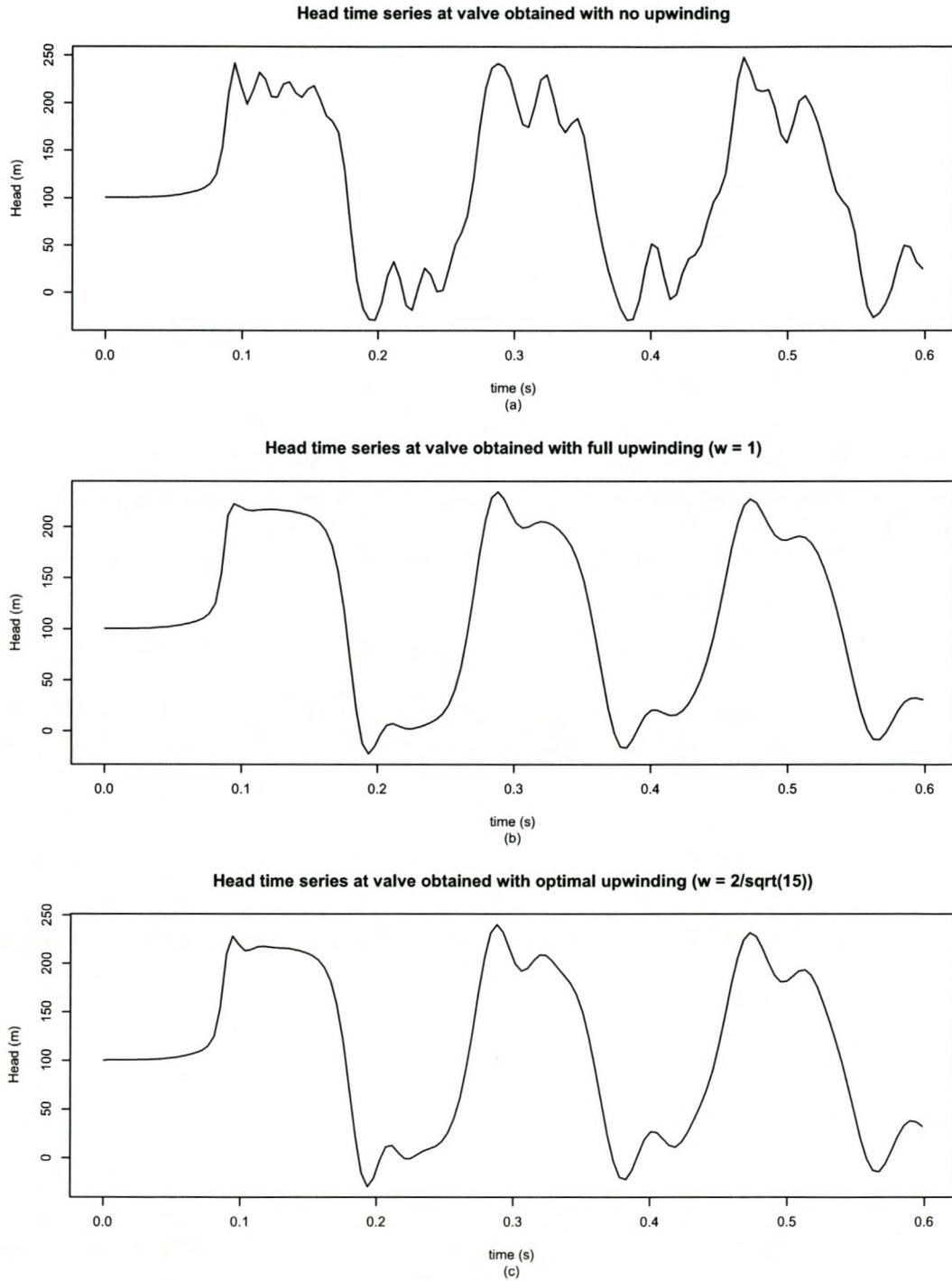


Figure 7.12: *Sensitivity analysis of the upwinding parameter*

without disturbing the timing of the wave too much. Figures 7.12a,b,c shows that the maximum head value is close to the correct value for a large range of upwinding values (except the low, spurious ones). We can therefore conclude that for this particular problem, the solution is largely independent of the upwinding parameter.

7.4 Pump model

7.4.1 Model objectives

The focus of this model was to see if sensitive boundary conditions, such as a pump-trip, can be implemented in the finite element scheme. By sensitive we mean that the boundary condition is described by a set of non-linear equations. This equation set requires its own Newton-Raphson iterative scheme. The results of this Newton-Raphson scheme is used in turn by the global system's Newton-Raphson scheme. An example of an insensitive boundary condition is that of the pipe-deadend. This boundary condition only imposes that its velocity dof is zero at all times. It was difficult to evaluate the model quantitatively, because data from other numerical experiments was not found. It was however, possible to compare the model qualitatively with the work of Wylie and Streeter [4], who analysed the transients for a pump-trip at the U.S. Bureau of Reclamation's Tracy pumping plant. The qualitative results compared well, to those of Wylie and Streeter. The section is divided into the following parts:

- Flow problem statement
- Comparison of computed results

7.4.2 Flow problem statement

Two reservoirs are connected by a 1200m long horizontal steel pipeline. The diameter, roughness and wavespeed of the pipeline is 0.5m, 0.0001mm and 1200m/s respectively. The upstream reservoir's operating level is 10.0m, while the downstream reservoir operates at 25.02m. The initial steady-state flow velocity is 1.5m/s. A booster pump is located at the pipe midpoint.

The pump has the following characteristics:

- Specific speed = 35 (SI units; a centrifugal pump)
- Head at optimum efficiency = 19.2m
- Torque at optimum efficiency = 1374N.m
- Flow at optimum efficiency = $0.294 \frac{m^3}{s}$

$$D = 0.5m$$

$$\text{roughness} = 0.0001mm$$

$$v_0 = 1.5m/s$$

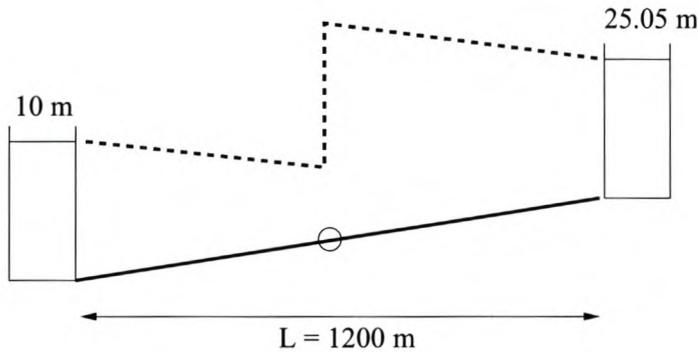


Figure 7.13: Two reservoirs connected by a steel pipe, with a pump at midpoint

- Motor speed at optimum efficiency = 592 rpm (value was determined for a pump of specific speed equal to 35, pump head and flow as given above)
- Motor inertia = 18.57 kg.m^2
- Suction and discharge diameters = 0.5m (large, but needed to fit pipe's diameter)

How does the pump behave after the power fails at time zero? In this problem the effect of cavitation has been ignored.

7.4.3 Comparison of computed results

Figure 7.14 presents the measured and computed results for a power failure at the U.S. Bureau of Reclamation's Tracy pumping plant [4]. The numerical scheme was a forward time, centred space explicit finite difference scheme, based on the method of characteristics. The system comprised of two identical parallel pumps, pumping water from a reservoir, through a pipeline whose length and diameter was 1564m and 4.57m respectively. The pump characteristics were

- Head at optimum efficiency = 60m
- Flow at optimum efficiency = $21.72 \frac{m^3}{s}$
- Motor speed at optimum efficiency = 180rpm

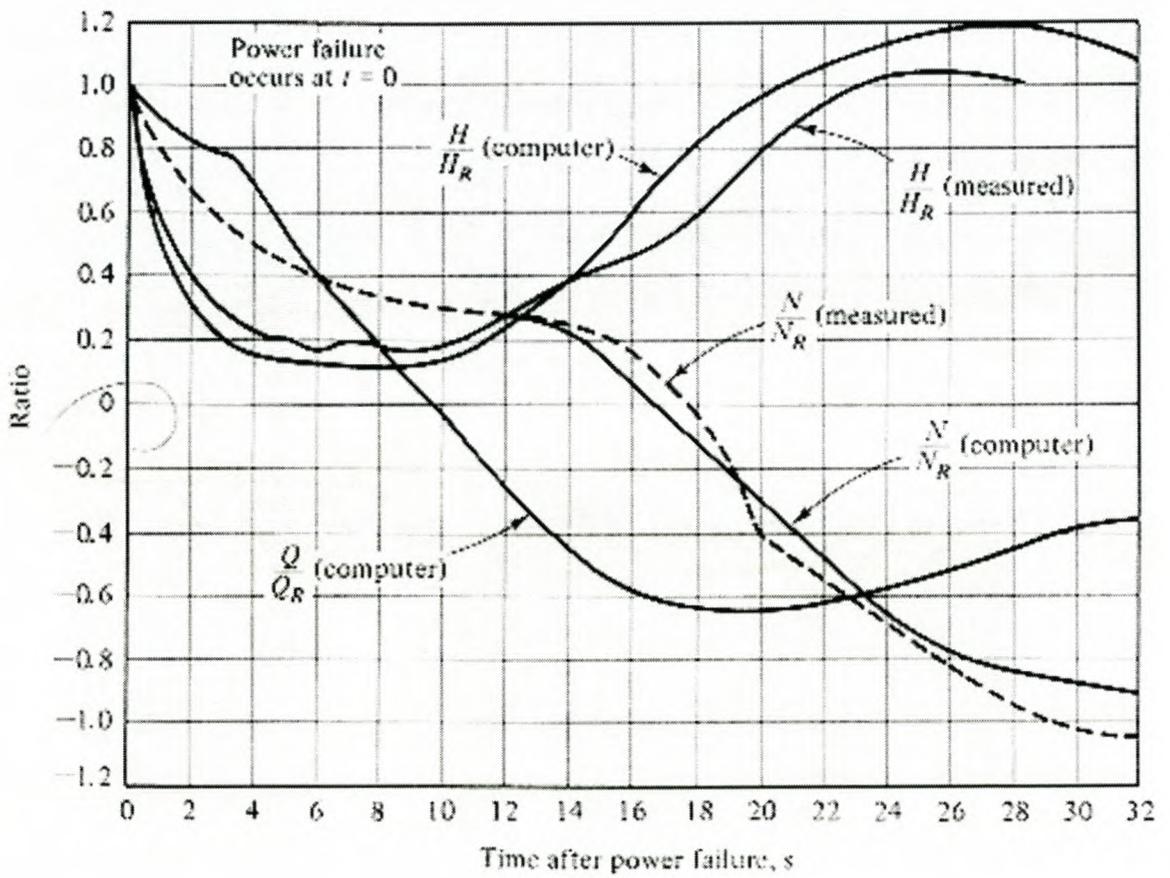


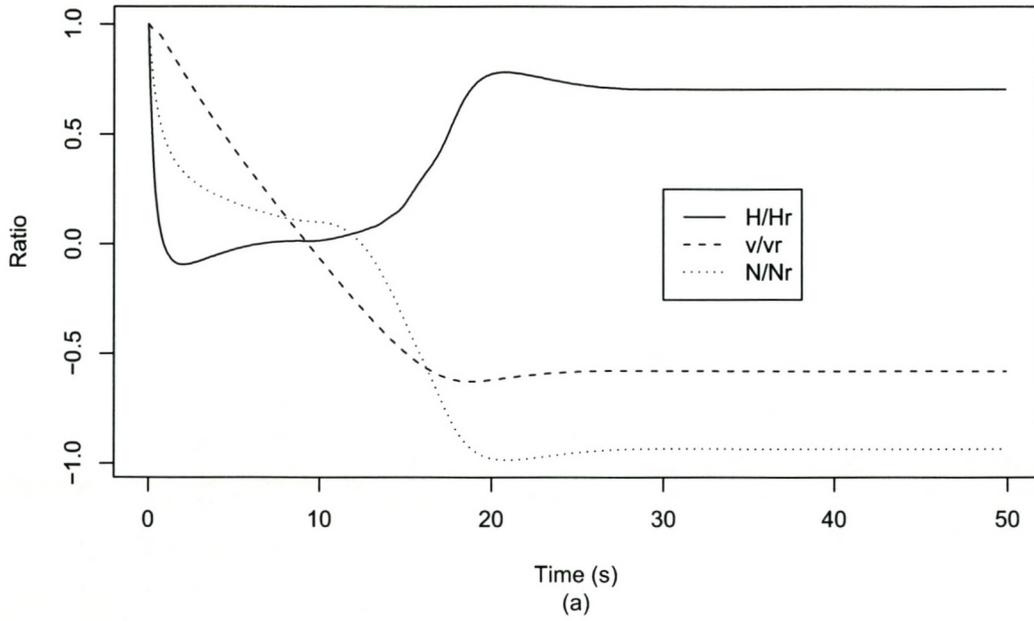
Figure 7.14: Computed and measured response of Tracy pumping plant

Figure 7.15a shows the Petrov-Galerkin solution for the flow problem defined in the previous section. It can be seen that the relative position of the speed, head and flow curves at any moment for the Petrov-Galerkin solution compares reasonably well with those of figure 7.14. Both plots show that the pump speed approaches, but never exceeds, the rated speed after pump-trip when it is acting as a turbine.

From figure 7.15b we can see how the pumps transition from normal to turbine operation varies smoothly after power-failure (See figure 5.2 for the operating zones defined by χ).

Figure 7.16 shows the steady state hydraulic gradelines, before and after power failure. Before power failure, the pump adds a dynamic head, which is larger than the head differential of the reservoirs. The steady state hydraulic gradeline after power failure shows that the head absorbed by the pump is less than the head differential of the reservoirs, which is realistic. This also explains why the steady state speed during turbine operation is not exactly equal to one.

Qualitative description of pump behaviour after power failure



Pump operating zone – time series

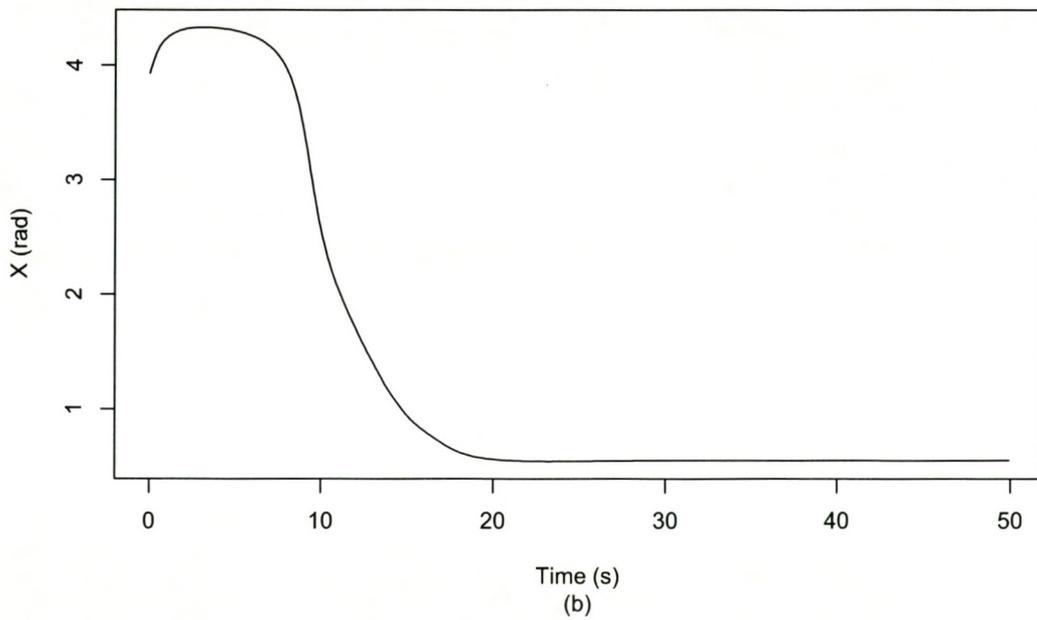


Figure 7.15: *Pump behaviour after power failure*

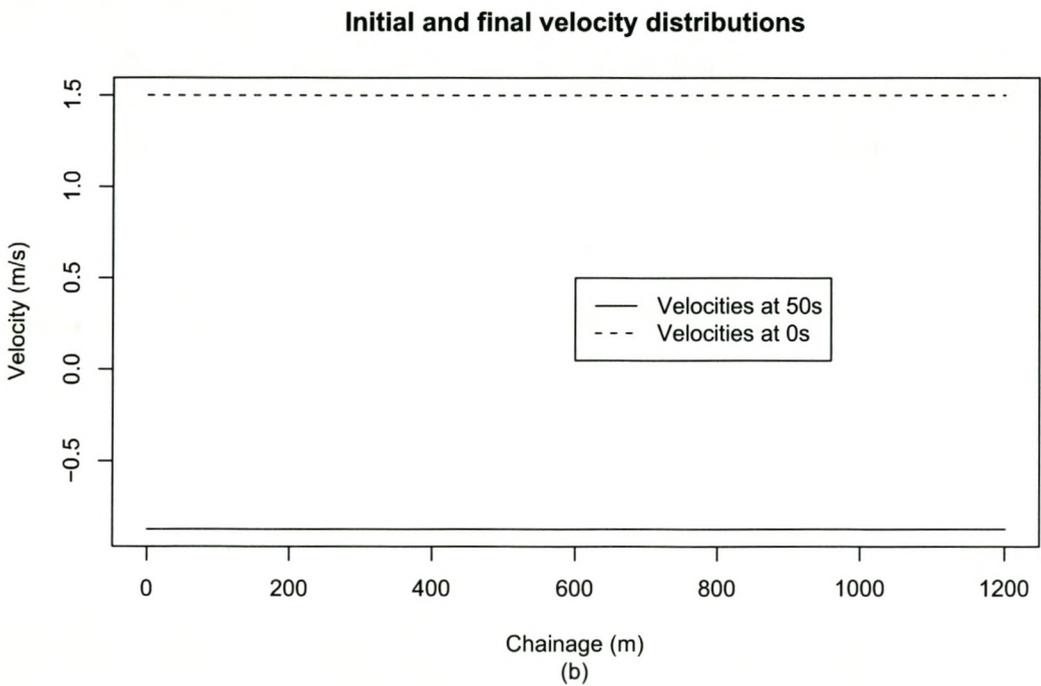
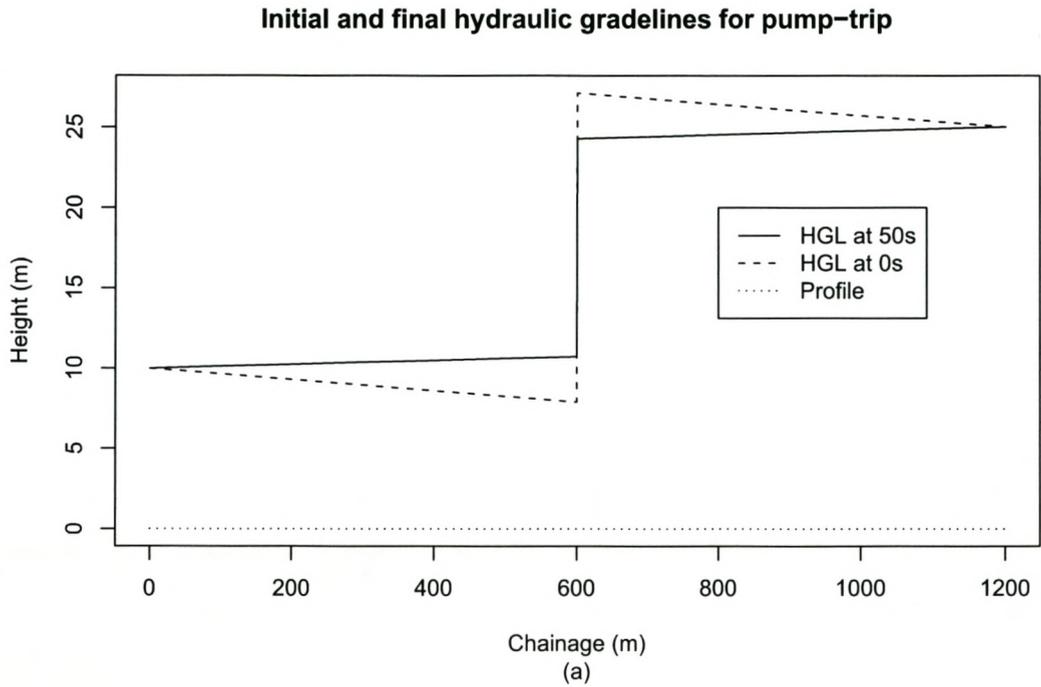


Figure 7.16: *Hydraulic Gradelines and velocity distribution at zero and fifty seconds*

Chapter 8

Conclusion

8.1 Practical use of Characteristic Dissipative Galerkin scheme

The Characteristic Dissipative Galerkin finite element has shown to be successful in solving the unsteady pipe-flow partial differential equations. It is an elegant and attractive method for a number of reasons:

1. The upwinding coefficient (ω) provides the flexibility to switch from a Petrov-Galerkin scheme ($\omega > 0$) to a Bubnov-Galerkin scheme ($\omega = 0$). The advantage of this, is that steady pipe-flows (which are elliptic) can also be solved by this scheme. This is helpful, since it reduces the computing time considerably, in setting up initial (steady-state) conditions for a dynamic analysis. This requires very little change in the computer code.
2. The method compares favourably, in terms of overall accuracy and stability with the Lax and Preismann finite difference schemes (which had been the work horses, for many years, in industrial applications) [10].
3. Complex and simple boundary conditions can be implemented in a simple, consistent manner. This facilitates simple, clear computer code.

8.2 Potential applications for industry

The Characteristic Dissipative Galerkin finite element has the ability to solve a number of practical engineering problems. A small list is given below.

1. The scheme can be extended to more than one dimension, which provides the opportunity to study velocity profiles in pipes or open-channels. e.g. A 2D open-channel

model could help identify areas in channels susceptible to scour or sediment deposition.

2. The scheme can also be extended to other hyperbolic systems, such as multilayer fluids, 2D depth-averaged flow or sediment transport problems [5].
3. The possibility exists for "mixing" different finite elements into a single model (e.g. An elastic pipe finite element could be used in conjunction with our unsteady flow finite element, to produce a pipe-fluid-interaction model).

8.3 Further work

In retrospect, the implementation of the finite element for this thesis could have been done better by:

1. Linearizing the equations for the boundary conditions and then creating elements of the boundary conditions themselves (see the following section for a short example).
2. Implementing a more efficient solver for the non-linear equations.
3. Implementing cavitation and airvalve boundary conditions, to extend the applicability of the unsteady pipe-flow model.
4. Implementing an unsteady friction model, instead of assuming steady-state friction factors for the flow at a node for each time-step.

8.3.1 Finite elements from boundary conditions

One of the improvements mentioned is, linearizing the boundary condition equations and then creating elements of the boundary conditions themselves. This subsection shortly shows how to create a valve element.

Examine the valve boundary condition equation:

$$v_1 = C_d \sqrt{2g(h_1 - h_0)} \quad (8.1)$$

A valve element can be created by linearizing this equation as follows:

Substituting

$$h_0 = h_{00} + \Delta h_0 \quad (8.2)$$

$$h_1 = h_{10} + \Delta h_1 \quad (8.3)$$

$$v_1 = v_{10} + \Delta v_1 \quad (8.4)$$

into equation 8.1 gives

$$\left(\frac{v_{10}}{gCd^2} \right) \Delta v_1 + \Delta h_0 - \Delta h_1 = 0 \quad (8.5)$$

Which can be written in matrix form as:

$$\left[\begin{array}{cccc} 1 & 0 & -1 & \frac{v_{10}}{gCd^2} \end{array} \right] \cdot \left\{ \begin{array}{c} \Delta h_0 \\ \Delta v_0 \\ \Delta h_1 \\ \Delta v_1 \end{array} \right\} = 0 \quad (8.6)$$

This valve element can now easily be assembled with the pipe elements into equation 4.110.

Bibliography

- [1] DE ALMEIDA, A. and E.KOELLE, *Fluid Transients in Pipe Networks*.
First edition. Southampton Boston: Computational Mechanics Publications, 1992.
- [2] DENDY, J., "Two Methods of Galerkin-Type Achieving Optimum L^2 Rates of Convergence for First Order Hyperbolics." *SIAM Journal of Numerical Analysis*, 1974.
- [3] DUPONT, T., "Galerkin methods for first-order hyperbolics: an example.." *SIAM Journal of Numerical Analysis*, 1973.
- [4] E.B.WYLIE and V.L.STREETER, *Fluid Transients*. First edition. Oxford: McGraw-Hill Inc., 1978.
- [5] HICKS, F. and STEFFLER, P., "Characteristic Dissipative Galerkin Scheme for Open-Chanel Flow." *ASCE, Journal of Hydraulic Engineering*, 1992.
- [6] HUGHES, T. and BROOKS, A., "A theoretical framework for Petrov-Galerkin methods with discontinuous weighting functions: Application to the streamline-upwind procedure." in *Finite Elements in Fluids*.
- [7] JOHNSON, C., "Finite Element Methods for Convection-Diffusion Problems." in *Fifth International Symposium on Computing Methods in Engineering and Applied Sciences*, 1981.
- [8] KAAZEMPUR-MOFRAD, M. and ETHIER, C., "An efficient characteristic Galerkin scheme for the advection equation in 3-D." *Comput.Methods Appl. Mech. Engrg*, 2002.
- [9] KATOPODES, N., "A Dissipative Galerkin Scheme for Open-Channel Flow." *ASCE, Journal of Hydraulic Engineering*, 1984.
- [10] KATOPODES, N., "Fourier Analysis of Dissipative FEM Channel Flow Model." *ASCE, Journal of Hydraulic Engineering*, 1984.
- [11] KATOPODES, N., "Two-Dimensional Surges and Shocks in Open Channels." *ASCE, Journal of Hydraulic Engineering*, 1984.

BIBLIOGRAPHY

120

- [12] ODEN, J., *Finite Elements of Nonlinear Continua*. McGraw-Hill, 1972.
- [13] RAYMOND, W. and GARDER, A., "Selective damping in a Galerkin method for solving wave problems with variable grids." *Monthly Weather Review*, 1976.
- [14] STREETER, V. L. and LAI, C., "Water-hammer analysis including fluid friction." *ASCE, Journal of Hydraulic Engineering*, 1962.
- [15] TURNER, M., "Stiffness and deflection analysis of complex structures." *J. Aero. Sci.*, 1956.
- [16] ZIENKIEWICZ, O. and CHEUNG, Y., "Finite elements in the solution of field problems." *The Engineer*, 1965.