# Pressure formulation and adaptive control of numerical algorithms for transient flow in pipe networks

A. J. Kriel

Thesis submitted for the degree Doctor of Philosophy at the Potchefstroom Campus of the North-West University

Supervisor: Prof. P. G. Rousseau

Co-supervisor: Prof. C. G. du Toit

November 2012

## Abstract

Fluid flow network simulation codes are commonly used as a design and analysis tool for many engineering problems such as gas distribution networks, power plants and heat pumps. Two formulations of conservation of momentum have been widely applied in fluid flow network simulation models namely those based on static pressure and those based on total pressure. The total pressure formulations are convenient in that they eliminate the difficulties associated with the calculation of the convective terms and components such as pipe junctions are treated in a straightforward manner based on total pressure losses. However, the different formulations of total pressure for compressible and incompressible flow require different formulations of the momentum conservation equation, which is inconvenient for implementation in a generic network simulation code. In this thesis a unified total pressure formulation is first derived which is valid for all fluids and therefore eliminates the inconvenience of switching between the compressible and incompressible formulations. A non-iterative method for the solution of the non-isothermal discretised equations based on the total pressure formulation is then introduced and consistency is illustrated. The method appears to be very stable for subsonic flows, while rapid steady state convergence is observed. A systematic comparison is also done with traditional static pressure based methods and the similarities and differences between the two formulations are illuminated.

The different time scales involved in the simulation of transient flow in fluid networks are problematic when conventional fixed time step methods are used for time-wise integration. The time scales associated with acoustic and kinematic wave phenomena as well as storage effects can differ by orders in magnitude. This thesis also presents a simple adaptive time step algorithm which can be readily used in conjunction with all the commonly used first order methods for fluid flow networks. Two test problems are selected to demonstrate the efficiency and savings obtained with this procedure. The adaptive time step algorithm correctly selects appropriate time steps for all phenomena and significant computational savings are observed for accurate integration. In addition, a procedure is implemented which automatically selects the appropriate integration method. The resulting algorithm is a fully adaptive algorithm which switches between a fully implicit method and a semi-implicit method. Two test problems are once again used to demonstrate the efficiency and savings. The fully adaptive algorithm correctly selects appropriate methods for all phenomena and significant additional computational savings are observed.



*Keywords:* Transient flow; Implicit methods; Pipe networks; Adaptive time step

## Opsomming

Vloeinetwerk simulasiekodes word algemeen gebruik as 'n ontwerp en analise-gereedskapstuk vir talle ingenieursprobleme soos gasverspreidingsnetwerke, kragstasies en hittepompe. Twee formulerings van momentumbehoud word wyd toegepas in vloeinetwerk simulasie modelle naamlik dié wat gebaseer is op statiese druk en dié wat gebaseer is op totale druk. Die totale druk formulerings is gerieflik in die sin dat dit die probleme geassosieer met die berekening van die konvektiewe terme elimineer en pyp aansluitings word op 'n eenvoudige manier hanteer wat gebaseer is op totale druk verliese. Die verskillende formulerings van die totale druk vir samedrukbare en onsamedrukbare vloei vereis egter verskillende formulerings van momentumbehoud, wat ongerieflik is vir die implementering in 'n generiese netwerk simulasie kode. In hierdie tesis word 'n veralgemeende totale druk formulering afgelei wat geldig is vir alle vloeiers en die ongerief van die skakel tussen die samedrukbare en onsamedrukbare formulerings elimineer. 'n Nie-iteratiewe metode vir die oplos van die nie-isotermies gediskretiseerde vergelykings wat gebaseer is op die totale druk formulering word dan bekendgestel en konsekwentheid word geïllustreer. Die metode is baie stabiel vir subsoniese vloei, terwyl vinnige bestendige toestand konvergensie waargeneem word. 'n Sistematiese vergelyking word ook gedoen met tradisionele statiese druk gebaseerde metodes en die ooreenkomste en verskille tussen die twee formulerings word uitgebeeld.

Die verskillende tydskale wat betrokke is in die simulasie van transiënte vloei in vloeinetwerke is problematies wanneer konvensionele konstante tyd stap metodes gebruik word vir tydintegrasie. Tydskale wat verband hou met akoestiese en kinematiese golfverskynsels sowel as stooreffekte verskil noemenswaardig in grootte. Hierdie tesis bied 'n eenvoudige aanpasbare tyd stap algoritme wat maklik gebruik kan word in samewerking met al die algemene eerste-orde-metodes wat gebruik word in vloeinetwerke. Twee toetsprobleme is gekies om die doeltreffendheid en besparing wat verkry word met hierdie proses te demonstreer. Die aanpasbare tyd stap algoritme selekteer korrek gepaste tydstappe vir alle verskynsels en aansienlike besparings vir akkurate integrasie word waargeneem. 'n Prosedure word ook geïmplementeer wat outomaties die gepaste integrasie metode selekteer. Die resulterende algoritme is 'n volle aanpasbare algoritme wat skakel tussen 'n volle implisiete metode en 'n semi-implisiete metode. Twee toetsprobleme word weereens gebruik om die effektiwiteit en besparings te illustreer. Die volle aanpasbare algoritme selekteer toepaslike metodes korrek vir alle verskynsels en merkwaardige verdere berekeningskoste besparings word waargeneem.



*Sleutelwoorde:* Transiënte vloei; Implisiete metodes; Pypnetwerke; Aanpasbare tydstap

*To my parents*

# Contents

# List of Figures

# List of Tables

# Nomenclature

| | |
|---|---|
| $\Delta t$ | change in $t$, s |
| $\Delta x$ | change in $x$, m |
| $\dot{m}$ | mass flow rate, $\mathrm{kg} \cdot \mathrm{s}^{-1}$ |
| $\dot{Q}$ | heat transfer rate, kW |
| $\dot{q}$ | heat transfer rate per unit volume, $\mathrm{kW} \cdot \mathrm{m}^{-3}$ |
| $\mathcal{V}$ | volume, $\mathrm{m}^3$ |
| $A$ | area, $\mathrm{m}^2$ |
| $a$ | acoustic velocity, $\mathrm{m} \cdot \mathrm{s}^{-1}$ |
| $c_p$ | constant pressure-specific heat, $\mathrm{kJ} \cdot \mathrm{kg}^{-1}\mathrm{K}^{-1}$ |
| $c_v$ | constant volume-specific heat, $\mathrm{kJ} \cdot \mathrm{kg}^{-1}\mathrm{K}^{-1}$ |
| $D$ | pipe diameter, m |
| $E$ | specific internal energy (including kinetic terms), $\mathrm{kJ} \cdot \mathrm{kg}^{-1}$ |
| $e$ | specific internal energy, $\mathrm{kJ} \cdot \mathrm{kg}^{-1}$; element connectivity entry |
| $f$ | Darcy-Weisbach friction factor |
| $g$ | gravitational acceleration, $\mathrm{m} \cdot \mathrm{s}^{-2}$ |
| $h$ | specific enthalpy, $\mathrm{kJ} \cdot \mathrm{kg}^{-1} \cdot \mathrm{K}^{-1}$ |
| $M$ | Mach number |
| $N$ | number of cells or faces |
| $n$ | node connectivity entry |
| $p$ | pressure, kPa |
| $R$ | Gas constant, $\mathrm{kJ} \cdot \mathrm{kg}^{-1} \cdot \mathrm{K}^{-1}$ |
| $s$ | specific entropy, $\mathrm{kJ} \cdot \mathrm{kg}^{-1} \cdot \mathrm{K}^{-1}$; sign of connectivity list element |
| $T$ | temperature, °C |
| $t$ | temporal coordinate, s |
| $u$ | velocity, $\mathrm{m} \cdot \mathrm{s}^{-1}$ |
| $V$ | volume, $\mathrm{m}^3$ |
| $x$ | spatial coordinate, m |

*Greek symbols*

| | |
|---|---|
| $\alpha$ | implicit weight factor |
| $\epsilon$ | error norm |
| $\gamma$ | ratio of specific heat capacities |
| $\Omega$ | control volume |
| $\partial\Omega$ | control surface |
| $\phi$ | scalar quantity |
| $\pi$ | 3.141593 |
| $\rho$ | density, $\mathrm{kg} \cdot \mathrm{m}^{-3}$ |

| | |
|---|---|
| $\tau$ | truncation error |

*Scripts*

| | |
|---|---|
| 0 | stagnation properties |
| $e$ | exit |
| $i$ | cell; inlet; spatial index |
| $j$ | face |
| $n$ | temporal index; neighbouring cell |

*Abbreviations*

| | |
|---|---|
| *ATol* | Absolute error Tolerance |
| *RTol* | Relative error Tolerance |
| CFD | Computational Fluid Dynamics |
| CFL | Courant-Friedrich-Lewy |
| FIBS | Fully Implicit Back Substitution method |
| MOC | Method Of Characteristics |
| MOL | Method Of Lines |
| ODE | Ordinary Differential Equation |
| PCIM | Pressure Correction Implicit Method |
| PISO | Pressure-Implicit with Splitting of Operators |
| SIMPLE | Semi-Implicit Method for Pressure-Linked Equations |
| TVD | Total Variation Diminishing |

# Chapter 1

# Introduction

## 1.1 Background

Many thermal-fluid system engineering applications which consist of different interacting components can be adequately described by a so-called pipe or fluid flow network approach. Such an approach usually consists of one-dimensional and lumped models which are connected in an arbitrary manner. Typically, pipes are treated as one-dimensional elements, while other components such as pumps and turbines are described by lumped models.

Pipe network simulation codes are widely used as a design and analysis tool for many engineering problems. Problems commonly described by a pipe network approach include water and gas distribution networks, power plants and cycles, air conditioners, heat pumps and internal combustion engines. Examples of popular simulation codes are RELAP5, SINDA/FLUINT, TRAC-P and Flownex. This study focuses on some techniques for the accurate and efficient simulation of transient flow in pipe networks.

### 1.1.1 Modelling approaches and solution algorithms

With such a wide range of applications, the accurate and efficient simulation of compressible and incompressible transient flow in pipe networks is essential. Transient fluid flow in pipes can be described by the one-dimensional Euler equations. Source terms account for frictional forces, gravitational forces, heat transfer, area variation etc. Various simplifying assumptions are often made, such as neglecting convective inertia or assuming isothermal flow.

Historically, a total pressure formulation of conservation of momentum has been preferred for incompressible pipe flow. This has the advantage of eliminating problems associated with the convective momentum terms and area variation as well as pipe junctions and boundary treatment. A similar total pressure formulation has also been used for ideal gas flows, which again benefit from these advantages.

Numerous methods have been used to solve the Euler equations for the simulation of transient flow in pipes. Extensive research has been done in the development and design of these methods, which range from the explicit compressible methods from the literature of hyperbolic conservation laws and the method of characteristics, to pressure correction based methods which are prevalent in the computational fluid dynamics literature. In the case of explicit methods for pipes, junctions and other components have been treated in a separate manner, usually with a non-linear solver in conjunction with pipe boundaries. In contrast, pressure based implicit methods can usually be integrated directly with junctions and other components.

These methods all have their advantages and disadvantages. Explicit methods are subject to stability criteria, which usually correspond to the Courant-Friedrich-Lewy (CFL) number associated with the fastest wave speed. These methods are preferred for the resolution of fast transient phenomena where wave phenomena are present and a small time step in the neighbourhood of the stability criteria is needed for accuracy. While implicit methods require the solution of a (usually) non-linear system of equations, these methods are much more stable than explicit methods, and stability criteria are usually much less stringent or even absent. This makes implicit methods computationally efficient for the simulation of slow transients, in particular transients which can be accurately modelled by a much larger time step as prescribed by the stability conditions of explicit methods. In addition, several semi-implicit methods also exist. In particular, such methods usually have a stability criterion based on the material CFL number, which is much less stringent than that of explicit methods for low speed flows. Usually the resulting system of equations for these methods are less complex than that of an implicit method.

### 1.1.2 Transient phenomena and adaptive time stepping

As noted earlier, in the simulation of transient flow in pipe networks, different time scales are typically involved. These commonly include, but are not limited to, acoustic and kinematic wave phenomena as well as storage, heat transfer and other effects introduced by various component models. Wave phenomena are usually the results of a rapid change in boundary conditions or component parameters. Examples include pipe rupture and accident scenarios, instantaneous valve closure and opening, compressor start up etc. Slow transients can be associated with storage tanks, heat storage and release and slowly varying boundary conditions and variations in component operating points. Since the presence of fast transients is typically of a short duration in comparison to that of slower transients, implicit methods have been commonly used as the method of choice for a general purpose simulation code.

For problems involving different time scales, adaptive time step algorithms provide an efficient means of accurate integration in time. Such algorithms usually obtain some measure of the discretisation error, and adjust the time step accordingly. This means that the largest possible time step which provides sufficient accuracy is always used. These methods have been well established in the literature of ordinary differential equations (ODEs) (Hairer et al. 1987, Hairer & Wanner 1991). For partial differential equations, such as flow problems, a method of lines (MOL) discretisation allows conventional methods for ODEs to be applied.

In addition to adaptive time stepping, switching between explicit and implicit methods provides further gains in efficiency. A typical approach is to use an explicit method and once stiffness is encountered, i.e. the explicit method is driven by stability as opposed to accuracy, the implicit method is used. Methods for detecting stiffness are once again available in ODE literature. If stability criteria are known for the explicit method, switching to and from an implicit method becomes a relatively straightforward task.

## 1.2 Problem statement

The literature study that was undertaken as part of this study revealed that despite this field being well established, certain issues still need further study and refinement.

This thesis focuses on the following issues:

- From the literature it is clear that incompressible flow in pipe networks are almost unanimously done with the total pressure formulation. Compressible flow, on the other hand, are done either with the static pressure formulation or a separate total pressure formulation found in Greyvenstein (2002) for ideal gases. The problem at hand is thus the development of a generalised total pressure formulation, which is valid for all fluids and reduces to the appropriate formulations under the correct assumptions. Next, non-iterative methods such as the PISO algorithm which are based on static pressure formulations need to be extended to the total pressure formulation. If this is done, fully iterative methods such as the family of SIMPLE algorithms can be developed in a straightforward manner.

- While adaptive time step algorithms have been used for the simulation of transient flow in pipe networks, this work is limited. In particular, the simultaneous treatment of very fast phenomena such as acoustic effects and slow effects associated with storage seems scarce. Furthermore, existing adaptive time step algorithms utilise classical methods from the ODE literature such as embedded Runge-Kutta, Rosenbrock and multi-step methods. Methods commonly used in the CFD literature, such as that of the PISO algorithm and semi-implicit methods, cannot always be categorised as one of these methods. Therefore, if these methods are used for pipe network simulation, a computationally efficient adaptive time step algorithm valid for these methods needs to be developed. Efficient treatment of transients of vastly different time scales also needs to be demonstrated.

- As previously mentioned, adaptive time stepping for pipe networks have been done with a single, usually implicit method. With different methods of varying degrees of implicitness available for transient flow simulation, an obvious extension of the adaptive time step algorithm is the addition of the automatic selection of the appropriate time integration method. The result will be an efficient fully adaptive algorithm.

## 1.3 Objectives

The objectives of this study can be organised into three parts. The objectives are:

- – Develop a generalized 1D total pressure formulation for conservation of momentum which is valid for all homogeneous fluids.
  - – Extend the family of non-iterative solution algorithms based on operator-splitting to the total pressure formulation.
  - – Investigate and compare methods based on static and total pressure with regard to additional aspects such as stability and steady state convergence.
- – Develop an adaptive time step algorithm which can be used with the methods developed in this study as well as any other commonly used method.
  - – Demonstrate the computational efficiency of an adaptive time step algorithm in comparison to a conventional fixed time step method as applied to pipe networks which consist of different transient phenomena.
- – In addition to adaptive time stepping, implement an automatic algorithm switching procedure which automatically selects the appropriate time integration method (implicit or explicit/semi-implicit).
  - – Demonstrate the computational efficiency of such a fully adaptive algorithm in comparison to conventional fixed and adaptive time step methods as applied to pipe networks which consist of different transient phenomena.

## 1.4 Methodology

The methodology used in order to achieve these objectives are summarised next.

**Theoretical work** Discretisation of the governing equations and construction of the solution algorithms were primarily done based on the finite volume method. In all the derived theoretical results such as the generalised total pressure formulation and error estimates for adaptive time stepping, standard techniques and theorems from calculus and numerical analysis were employed.

**Numerical modelling** A C++ program was written for the simulation of simple pipe networks. Modelled components include pipe elements and junctions. Four solution algorithms together with adaptive time stepping were included, i.e. segregated methods based on a static and total pressure formulation, a computationally efficient semi-implicit method as well as FIBS (Garland 1998). The program uses a text file input for all input data, which include the network layout described in Chapter 4 as well as boundary conditions and discretisation information. An output text file is generated which contains the results and other relevant information. A direct solver was used throughout.

**Verification and validation** Numerical results were verified and validated with solutions and results obtained from the literature. Analytical solutions of Riemann problems, in particular the shock tube problem of Sod (Wesseling 2001), were used for verification and validation of the unsteady 1D flow. Additionally, for 1D flow, the analytical solution of the Fanno flow problem was used which includes the frictional effects. Finally, the methods used were compared with numerical results from the literature in order to validate the simulation of a simple pipe network which also includes a pipe junction. Results from Botha (2003) were primarily used for this purpose.

**Demonstrations** Test problems were constructed to demonstrate and compare various aspects of different methods. These problems were selected or specifically constructed to highlight aspects of interest, such as a network which consists of different transient phenomena where the virtues of an adaptive algorithm would be highlighted.

## 1.5 Limitations

The limitations of this study include the following:

- A generic direct solver based on Gaussian elimination was used throughout this study. While sufficient for all purposes of this study, iterative and sparse linear solvers will provide greater efficiency, especially for large systems of equations.

- Pipe network components are limited to pipes and junctions. Treatment of components such as pumps, turbines and heat exchangers was beyond the scope of this study.
- The fully adaptive algorithm switches between methods based on a staggered grid. Additional methods such as the classical methods from hyperbolic conservation laws which are based on a collocated grid were not considered, as this would introduce additional complexity such as interpolation between staggered and collocated variables.

## 1.6 Outline of the thesis

**Chapter 2** contains the literature study.

**Chapter 3** focusses on the simulation of transient flow in pipes. Relevant theory, the generalised total pressure formulation and corresponding non-iterative solution algorithm are presented. Comparisons with regard to various aspects are done with conventional static pressure methods.

**Chapter 4** primarily discusses the treatment of pipe junctions and validates the methods under consideration for a simple pipe network with results found in literature. Additional information is given with regard to computational cost.

**Chapter 5** introduces an adaptive time step algorithm which is valid for all commonly used first order methods. Efficiency and accuracy are demonstrated and compared with traditional constant time step methods.

**Chapter 6** discusses a fully adaptive algorithm, which consists of both adaptive time stepping and automatic switching between an explicit/semi-implicit and implicit methods. Efficiency and accuracy are demonstrated and compared with traditional constant time step methods as well as the adaptive time step methods of Chapter 5.

**Chapter 7** summarises the conclusion of the study. Recommendations for further work are also given.

# Chapter 2

# Literature survey

## 2.1 Introduction

In this chapter an overview is given of the most common techniques used in the simulation of flow in pipelines and networks as found in the literature. The chapter consists of primarily three sections. In the first section, a brief review of the literature is given on the solution of the classical steady state incompressible networks. The second section comprises of the primary focus of the literature, i.e. the solution of transient flow problems. Finally, the third section focuses on the treatment of pipe junctions.

## 2.2 Incompressible steady flow in pipe networks

A considerable number of papers have been published on the solution of steady state incompressible flow in pipe networks. Since this is not the main focus of this study, only a few pointers to relevant literature are given. Methods are generally based on node or loop formulations and graph-theoretical models such as described in Kesavan & Chandrashekar (1972). The oldest method for solving the equations can be traced back to the Hardy Cross successive substitution method (Cross 1936). The Hardy Cross method often encountered convergence problems, which lead to the development of coupled methods. A summary and evaluation of modern methods for steady networks based on Newton's method and methods originating from CFD can be found in Altman & Boulos (1995), Wood & Rayes (1981), Greyvenstein & Laurie (1994) and Kelkar & Patankar (2003).

## 2.3 Transient flow in pipe networks

### 2.3.1 Method of characteristics (MOC)

A semi-analytical method, the well-known method of characteristics, was originally developed for fast transients in liquid pipelines (Wylie & Streeter 1993). The method transforms the partial differential equations of continuity and momentum into ordinary differential equations. These equations are valid along lines in the time-distance plane called characteristics. Van Ravenswaay (1998), Wylie & Streeter (1993) and Chaudhry (1979) gave a thorough derivation and implementation of this method. Numerous articles and other texts are available on the implementation of this method. For a further overview, refer to Thorley & Tiley (1987) or Kim (2008).

Wylie & Streeter (1993) stated that the characteristics method has several advantages over other methods, which include an explicit solution, well established stability criterion, high accuracy and is based on a physical interpretation. The characteristics method can also handle discontinuities efficiently (Thorley & Tiley 1987). Thorley & Tiley also recommended the characteristic method in the case of rapid transients in single pipelines. The primary disadvantage is the strict adherence to the time step-distance interval relationship. Furthermore, the Courant-Friedrich-Lewy (CFL) condition is necessary for stability and convergence.

Various interpolation schemes for a fixed-grid MOC also exist, which can be done over time, space or a combination of both. Kim (2008) gave a thorough overview of interpolation based MOC schemes. Issa & Spalding (1972) used linear interpolation between a rectangular grid superimposed on the characteristic mesh and favourable results were obtained for the shock tube problem as compared to experimental results.

For a complex pipe network, it is necessary that the same time step is used for all elements while ensuring that the CFL condition is fulfilled. This is to ensure that boundary values at junctions can be used (Chaudhry 1979). This can lead to problems, especially if some pipes are short relative to others. In this case Chaudhry recommended the use of an implicit method combined with the characteristic method. Afshar & Rohani (2008) proposed an implicit MOC with the advantage of handling any configuration of devices in a pipeline system. Issa & Spalding (1972) described a characteristic method for unsteady one-dimensional compressible frictional flow with

heat transfer.

The method has been extended to deal with isothermal gas flows, but severe limitations arise when applying it to more general situations. For non-isothermal gas flow the sonic velocity varies and the temporal and/or spatial increments have to be adjusted accordingly. Similar problems arise for sub-networks with different types of fluids and heat exchangers (Greyvenstein 2002). In addition, conservation of energy has to be dealt with in non-conservative form, which can have undesired consequences. For two-phase flows, the method of characteristics also becomes problematic (Chaudhry 1979, Kim 2008).

These factors make the characteristic method a non-viable choice for general non-linear pipe network simulation.

### 2.3.2 Explicit finite difference and finite volume methods

Compressible methods are used extensively for high speed flow where there is a strong coupling between density and pressure. For hyperbolic problems, such as the compressible Euler equations, explicit methods are generally efficient (Leveque 2002) since the domain of dependence is always bounded. For high speed flow, there is no great disparity between the eigenvalues. This means that the Courant-Friedrich-Lewy (CFL) condition will not impose large restrictions.

Explicit finite volume methods for homogeneous hyperbolic systems are well established and can be found in Leveque (2002). For the more specific case of gas dynamics and the Euler equations, see Laney (1998). Techniques include the Lax-Friedrichs method, the Lax-Wendroff method, high-resolution versions of Godunov's method and various total variation diminishing (TVD) and flux difference splitting schemes capable of capturing shocks and the investigation of wave propagation.

Implementation of these standard methods does not necessarily carry over to problems with source terms. Methods can be developed for these problems, but higher order derivatives might be complicated to compute, making it difficult to develop higher-order numerical methods (Leveque 2002).

The treatment of the source terms can be done efficiently via operator splitting techniques. This comprises of solving the homogeneous hyperbolic system followed by the solution of an ordinary differential equation. The advantage of this is that the

standard methods for homogeneous hyperbolic problems can be used, together with an ordinary differential equation solver (Leveque 2002). The splitting, however, introduces a first-order error, regardless of the schemes used. Leveque (2002) added that the coefficient of the first order term may be much smaller than the coefficients of the second-order term, and the splitting method will produce generally favourable results. Another disadvantage of this approach is that steady state solutions will depend on the temporal discretisation, which may be undesired.

On the specific application of shock capturing methods for the simulation of transient flow in pipelines, Thorley & Tiley (1987) gave an overview of some commonly used flux difference splitting schemes. Thorley & Tiley also stated that considerable computer time is required to split the flux difference. Dukhovnaya & Adewumi (2000) used a TVD scheme for the simulation of non-isothermal transients in a two-phase flow single pipeline, which preserved shocks with minimal dispersion. Operator splitting techniques combined with TVD schemes have also been applied successfully to gas flow in pipeline networks (See for example Banda et al. (2005)).

Thorley & Tiley (1987) also described other commonly used explicit methods for transient flow of compressible fluids in pipelines. These include the first order method of Lax and higher order schemes such as the Lax-Wendroff-, two-step Lax-Wendroff- and MacCormack methods. Thorley & Tiley stated that due to conditional stability these explicit methods are generally not suited for the analysis of large networks and simulation over long time periods. Furthermore, difficulties can arise with the natural implementation of boundary conditions. However, explicit methods, such as the two-step Lax-Wendroff method, have been implemented successfully in transient networks (See for example Botha (2003) and Chang (2002)).

Difficulties are encountered when compressible methods are applied to low speed flow problems. Standard compressible methods applied to low speed flow problems (when the Mach number is below 0.3) give rise to accuracy problems due to the weak coupling between pressure and density and loss of efficiency due to stiffness. These problems have been remedied through preconditioning techniques (van der Heul et al. 2003, Wesseling 2001). For unsteady flows, however, time accuracy has to be restored. This can be done with dual time stepping techniques, which are expensive, and cannot be called efficient (Wesseling 2001). Alternatively, time accurate preconditioners can

be used, eliminating the need for dual time stepping. The resulting equations are, however, expensive to solve (Wesseling 2001).

Thermal hydraulic networks consist of multiple hyperbolic problems, and the CFL condition may impose undesired restrictions since a uniform time increment is to be used throughout the network. If short pipe elements are present, or non-uniform spatial discretisation is used, discretisation induced stiffness can arise and the CFL condition imposes severe limitations on the overall step size (Chua & Dew 1984). This is one of the major disadvantages of explicit methods.

### 2.3.3 Implicit finite difference and finite volume methods

Implicit methods are effective in dealing with stiff equations and slow transients, since these methods are unconditionally stable and not subjected to the CFL condition (Thorley & Tiley 1987). Disparity between the wave speeds and component models can result in stiffness. Discretisation induced stiffness can also arise, e.g. short pipes in the network (Chua & Dew 1984). The first part of this section reviews implicit pressure based methods, while the second part of the review comprises of alternative methods.

#### Implicit pressure based methods

In computational fluid dynamics, pressure based methods constructed on a staggered grid were initially developed for incompressible flow. In these original methods, pressure is usually obtained by manipulating the continuity and momentum equations. These methods include the family of pressure correction methods, such as the fully iterative SIMPLE algorithm and its variants, the non-iterative PISO algorithm, fractional step methods, etc. For transient simulations, a method of lines (MOL) approach is usually adopted.

Traditional incompressible methods have been extended towards a unified algorithm with great success. For transient flow, non-iterative pressure based methods have also been developed, which are much more efficient than traditional SIMPLE algorithms (van der Heul et al. 2000). These include the compressible pressure correction methods of van der Heul et al. (2000) and Wesseling (2001). Various non-iterative semi-implicit methods have also been developed, which treat only acoustic terms im-

plicitly. Examples are the works of Casulli & Greenspan (1984), Kwatra et al. (2009), the non-conservative approach by Yoon & Yabe (1999) and the methods for two phase flow as described in Prosperetti & Tryggvason (2007). In addition to conventional computational fluid dynamics, traditional pressure based methods have also been successfully applied and extended to pipelines and networks (See for example Walter et al. (2000) and Majumdar & Ravindran (2010)).

A tube-collector-model was presented by Walter et al. (2000) describing the fluid flow in a network of tubes. The treatment of junctions and boundary conditions were also discussed in detail. The difference between explicit and implicit calculations were shown, as well as some results of the transient simulation of a Heat Recovery Steam Generator (HRSG). According to Walter et al., the necessary CFL condition implies a very small time step for explicit methods. This is due to the fact that the speed of sound in water is very high. Consequently the simulation by using explicit methods is very time consuming. The implicit method was solved by using the SIMPLE-algorithm.

Greyvenstein (2002) developed a fully implicit finite difference method for the analysis of transient flows in pipe networks, termed the pressure correction implicit method (PCIM), which is based on the fully iterative SIMPLE-algorithm. The method is valid for isothermal as well as non-isothermal flows and for both fast and slow transients. With the introduction of stagnation properties, conservation of momentum has been formulated in terms of total pressures. This has the advantages of eliminating difficulties associated with the convective terms, as well as a more natural treatment of pipe junctions and boundary conditions. A downside is that separate formulations are needed for incompressible and compressible gas flows, in contrast to a conventional static pressure formulation. The method compared favourably with the MOC and two-step Lax-Wendroff methods. A similar method was extensively verified by Van Ravenswaay (1998) and compared to the work of Kiuchi (1994).

#### Other implicit methods

Thorley & Tiley (1987) provided an overview of the most commonly used implicit methods for transient flow of compressible fluids in pipelines. These include the fully implicit method, the semi-implicit Crank-Nicholson method, the central difference method and the characteristic finite difference method. The latter method can also be used effi-

ciently to simulate transient homogeneous two-phase flow. The main disadvantages of these methods are the required solution of a system of non-linear equations at each time step and excessive smoothing of fast transients and shocks. Thorley & Tiley (1987) and Chaudhry (1979) stated that implicit methods are more suited for the analysis of large networks than the method of characteristics and explicit methods.

Herran-Gonzalez et al. (2009) derived various simplified models for the gas distribution network problem. A MATLAB-Simulink library was built in order to solve the problem with the MOC and Crank-Nicolson methods. Behbahani-Nejad & Bagheri (2008) developed a transient flow simulation based on transfer functions and MATLAB-Simulink. This was done by using local linearisation and Laplace transforms. Behbahani-Nejad & Bagheri compared their results with those of the conventional finite difference schemes. Sufficient accuracy was obtained, and Behbahani-Nejad & Bagheri claimed that the proposed method is computationally more efficient than other methods.

Islam & Chaudhry (1998) developed an implicit finite difference scheme for the solution of transient networks. Compressibility effects were neglected due to slowly varying pressures and flows. The solution algorithm was based on the loop equations which is popular in steady state networks. The assumption of incompressibility makes this method suitable for transient simulation. For compressible flow the loop method will not carry over directly to the transient case. Furthermore, Islam & Chaudhry also calculated the transport, mixing and decay of a constituent. The one-dimensional transport equation was solved with an operator splitting technique together with explicit schemes.

A gas flow network formulation was introduced by Engl (1996) for the modelling of complex gas transmission systems. A second-order TVD scheme was used for the pipe equations, while an implicit predictor-corrector method similar to the Adams-Moulton method was used for the chambers and other connections. The predictor-corrector method was solved with Newton's method.

Various techniques were compared by Emara-Shabaik (2004) for the simulation of transient flow in a single pipeline. These include an alternating-space method, centred-spaced methods such as the Crank-Nicolson scheme and a predictor-corrector method based on MacCormack's method. Emara-Shabaik found that the Crank-Nicolson scheme produced unrealistic oscillations, while MacCormack's method gave

the most satisfactory results of the methods that were compared.

Arfaie & Anderson (1991) compared different implicit finite difference methods commonly used for isothermal hyperbolic fluid transient problems. These included the Crank-Nicholson, fully implicit and four-point centred-difference schemes. Arfaie & Anderson stated that the latter scheme is the most popular method found in the literature for unsteady pipe flow. Comparisons were made with a MOC benchmark solution and these three methods. The Crank-Nicolson method produced results with strong oscillations which were not improved by increasing the number of nodes, while the fully-implicit method showed excessive smoothing of sharp transients. Arfaie & Anderson (1991) confirmed that the four-point centred-difference scheme is the best of these in this regard and can be further improved by implementing a spatial weighing factor.

Guy (1967) described the implicit finite difference method for the case of isothermal gas networks. Guy used an iterative method which solves a set of tridiagonal matrices, one for each pipe. This is done by estimating the pressures at each junction. This approach eliminates the need for a global matrix inversion. It is expected, however, to decrease the convergence rate, but with the hope of decreasing the computational cost.

A fully implicit finite difference method for the calculation of unsteady isothermal gas flow in pipeline networks was described by Kiuchi (1994). The finite difference representation is similar to that of the four-point centred-difference scheme described by Arfaie & Anderson (1991) and Thorley & Tiley (1987). Kiuchi used the Newton-Raphson method as solution algorithm to solve the finite difference equations. He also performed a Von Neumann stability analysis on the finite difference method which showed that the method is unconditionally stable. Kiuchi compared the results with those of the method of characteristics, the two-step Lax-Wendroff method and the method presented by Guy (1967) and found excellent agreement. He stated that the computational time can be greatly reduced by using the implicit method.

Abbaspour & Chapman (2008) described another implicit finite difference method for solution of the continuity, momentum and energy equations for flow within a gas pipeline. The gas is treated as non-isothermal, and Abbaspour & Chapman stated that this is essential for the accurate simulation of pipeline flow, in particular for fast transients. The resulting set of difference equations consists of mass flow rate,

temperature and pressure, and are solved with the Newton-Raphson method. It was also concluded that the convective inertia term plays an important role in gas flow analysis and must be retained. Results were compared with Kiuchi's method, and when the same assumptions were made, results were found to be identical.

Majumdar & Ravindran (2010) described an implicit finite volume formulation also based on a staggered grid technique. The authors used a combination of Newton's method and the successive substitution method to solve the equations. Majumdar & Ravindran stated that usually the coupling among mass conservation, momentum conservation and equation of state is stronger than other equations such as the energy equation, and a decoupling of the equations leads to significant memory and computer time savings. Majumdar & Ravindran proposed that the strongly coupled equations are solved by Newton's method, while the weakly coupled equations are solved by the successive substitution method. They also implemented Broyden's method, a quasi-Newton method based on a multidimensional generalization of the secant method, which enables efficient evaluation and inversion of the Jacobian matrix at the cost of convergence speed. They proposed and compared four different methods for solving the non-linear algebraic systems of equations arising from network flow models. The four categories are: a Newton solver, a Broyden solver, a hybrid Newton-successive substitution (Newton-SS) solver and a hybrid Broyden-successive substitution (Broyden-SS) solver, where the hybrid methods utilize the afore-mentioned decoupling. Majumdar & Ravindran compared these methods for various problems and confirmed that the hybrid methods performs considerably better than the fully coupled solvers. They also concluded that the hybrid Broyden-successive substitution solver is superior to the hybrid Newton-successive substitution solver. Further details of this work can be found in Majumdar & Ravindran (2010) and Majumdar (1999).

An algorithm for thermal hydraulic simulation based on the work of Porsching was successfully implemented into the program SOPHT (Simulation of PHWR Heat Transport) (Garland 1998). Porching's method involves the linearisation of the conservation laws and successive back substitution of the equations. The result is a single linear system of equations for the mass flow rate which can be solved by traditional means, while all other unknowns can be calculated by simple back-substitutions. The method however, is only semi-implicit, with fully implicit treatment of flow and explicit treat-

ment of enthalpy. This will likely result in stability criteria based on the material CFL condition. Details of this method can be found in Garland (1998) and Leung (1997).

Garland (1998) extended Porching's method in order to maintain the fully implicit formulation, which he termed the Fully Implicit Back Substitution method (FIBS). Similar to Porsching's method, the result is a single system of equations for mass flow rate resulting from the linearisation of the conservation laws, while all other unknowns can be calculated by simple back-substitutions. The method, however, includes the entire calculation and storage of a matrix inverse in order to maintain the fully implicit formulation, which can be an expensive calculation. Garland's FIBS method is based on the linearisation of the equations and can be regarded as a single step of Newton's method. Therefore the procedure can be repeated iteratively for a single time-step if a higher solution accuracy is desired.

### 2.3.4 Finite element methods

Finite element methods have not been widely used for transient flow simulation of compressible fluids in pipelines (Thorley & Tiley 1987, Kim 2008), mainly due to the high computing time and storage requirements. MOL techniques are usually used for the computation of transients, where finite elements are used in the spatial domain and finite differences in the time domain.

Thorley & Tiley and Kim (2008) did however give several advantages of finite element methods over traditional methods. These include non-uniform distribution of nodal points and efficient handling of boundary conditions and variable size system components.

### 2.3.5 Adaptive time stepping and the method of lines (MOL)

An alternative method for the treatment of partial differential equations is to discretise the spatial derivative first, reducing the partial differential equation to an ordinary differential equation. The advantage of this approach is that all the available techniques for ordinary differential equations can be used to solve the problem, which include implicit schemes appropriate for stiff equations. Efficient methods for solving initial value problems can generally be categorised as linear multistep methods or Runge-Kutta methods. The MOL can further be used effectively together with automatic

step-size control. Hairer et al. (1987) discussed the implementation of variable step size methods for ordinary differential equations. A method of lines approach has also been used successfully in conjunction with spatial adaptivity, which is an efficient approach for hyperbolic problems with moving shocks. For a comprehensive review on both temporal and spatial adaptivity, refer to Vande Wouwer et al. (2001).

Since the resulting ordinary differential equations contain errors arising from the spatial discretisation, the use of very high-order methods are generally not recommended (Fletcher 1991). Furthermore, Suwan & Anderson (1992) stated that although high-order methods may be appropriate for parabolic problems, it may not give better accuracy for hyperbolic problems.

Chua & Dew (1984) designed an efficient variable-step integrator for the solution of transient gas networks, which is based on the control of the local error. This method has reduced computational time compared to constant-step algorithms. Chua & Dew experimented with different numerical schemes, and found the weighted implicit methods as well as Rosenbrock methods among the most efficient. The simplified Newton method was used to solve the nonlinear system of equations of the implicit methods. This study was done for isothermal flow and wave phenomena were not considered.

Tentis et al. (2003) successfully used the method of lines together with Gear's backward difference formula with automatic step size control to solve the unsteady gas flow transmission system. The authors also obtained improvements by implementing an adaptive coarse grid. This approach was found to be significantly more efficient compared to that of the fine uniform grid. Other authors, such as Chaczykowski (2009), have also successfully applied Gear's method with automatic step size control.

An extensive comparison of commonly used adaptive time stepping methods was done by John & Rang (2010) for the incompressible Navier-Stokes equations. These included the $\alpha$-scheme, diagonal-implicit Runge-Kutta (DIRK) methods and Rosenbrock-Wanner (ROW) methods. Here, the $\alpha$-scheme is a generic time integration scheme where $\alpha$ is a parameter that controls the degree of implicitness. Embedded methods were used to control the time step. It was found that several ROW methods outperformed the other methods, in particular the Crank-Nicolson scheme with constant time step. The efficiency of the ROW methods can be attributed to the fact that only a single LU-decomposition is necessary to solve the multiple stages. This means that these

methods will lose efficiency and the advantage over other methods for large systems where iterative methods are preferred.

A new time integration method was introduced by Berzins (1995) for convection dominated problems. The method of lines approach was adopted and time step selection was done based on a balancing of temporal and spatial contributions to the discretisation error. The idea is to select the time step such that the total discretisation error is dominated by the spatial discretisation error. Furthermore, the time integration should not be carried out with a much higher accuracy than that obtained in space. See Berzins (1995) for further details on various strategies based on this idea.

The method of lines approach was used by Hornëll & Lötstedt (2001) to solve a conservation law. The authors used an explicit embedded Runge-Kutta method for automatic time-step selection based on the local discretisation error. Hornëll & Lötstedt stated that the error estimates at discontinuities are too pessimistic. A filter was then introduced which based the error estimate on smooth parts of the solution only. The technique was justified numerically with a one dimensional Riemann problem as well as a two dimensional Burgers' equation.

For multi step methods, error estimates are more straightforward to obtain than for Runge-Kutta methods (Shampine et al. 2003), and can be obtained by interpolation using values at previous time steps. The result is that error estimates can be obtained explicitly. A similar approach was described in Gresho et al. (1980) as applied to the incompressible Navier-Stokes equations. In this approach the trapezoidal method was used as the primary integrator, while an explicit second order Adams-Bashforth method was used in order to obtain an error estimate. This method was successfully demonstrated in Kay et al. (2010) to work efficiently for the incompressible Navier-Stokes equations with different time scales. A similar approach is used in the software SINDA/FLUINT.

Richardson extrapolation is a third technique for error approximation for adaptive time stepping. This approach is usually more expensive than an embedded technique. A comparison between these techniques was given in Kaps et al. (1985) who pointed out several advantages of Richardson extrapolation. For further details on Richardson extrapolation and initial value problems, see Hairer et al. (1987).

Lewandowski (2000) used the MOL together with Runge-Kutta-Chebyshev methods

with extended stability regions for transient modelling of gas pipeline networks. This is an explicit method, which trades accuracy for stability. Lewandowski stated that the Runge-Kutta-Chebyshev methods can only be applied successfully to parabolic equations. By omitting terms in the original complete governing equations, the gas model becomes parabolic and the Runge-Kutta-Chebyshev methods can be implemented. Computational cost was greatly reduced with this method. However, the altered governing equations limits this method's applicability.

Various other authors have successfully implemented the method of lines in thermal hydraulic networks. These include Osiadacz & Cahczykowski (2001) and Van Ravenswaay (1998). Transient networks based on electrical analogies together with the method of lines have also been successfully implemented by Nicolet (2007) and Tao & Ti (1998).

## 2.4 Pipe junctions

Since junctions are encountered in this thesis, a brief review of the literature regarding this aspect is given. Numerous works on the treatment of pipe junctions can be found in the internal combustion engine literature (Benson et al. 1963-4, Bingharn & Blair 1985, Corberan 1992, William-Louis et al. 1998). Various solution methods, in conjunction with the method of characteristics for the pipe branches, have been proposed.

Pipe junctions are commonly treated as quasi-steady, while pressure loss models are commonly used. Static pressure loss models can be found in Benson et al. (1963-4) and Bingharn & Blair (1985) while losses based on total pressures models can be found in Bassett et al. (2001) and William-Louis et al. (1998). Constant pressure theory (or more correctly, uniform pressure theory (Corberan 1992)) assumes an equal pressure at each junction end (Benson et al. 1963-4). This model is realistic when the velocities are low and the junction geometry is such that pressure losses are minimal (Corberan 1992).

If pipe junctions are treated as plenums, i.e. the volume of the junction and associated transient effects are included, explicit methods can be used in a straightforward manner. However, in the case of a quasi-steady model of junctions, the balance equations are given by a set of coupled non-linear algebraic equations. If explicit methods

are used for pipes, non-linear solvers such as Newton's method are usually adopted to solve the resulting coupled equations at junctions (Chae et al. 2006, Bingharn & Blair 1985). From the potential difficulties in solving these equations, an alternative iterative method was also given in William-Louis et al. (1998). This method consists of decoupling the equations by considering a single incoming and outgoing branch at an iteration. The method was found to be reliable and stable.

In the case of explicit treatment of pipe elements, non-iterative methods have also been developed for the treatment of junctions. A non-iterative method is given by Chae et al. (2006). For this approach, a quasi-linear set of equations is given which does not sacrifice accuracy (Chae et al. 2006). The model was based on time-dependant boundary conditions for hyperbolic conservation laws found in Thompson (1987). The model was implemented and tested for the calculation of behaviour of nonlinear wave interaction in a branched pipe. Good agreements were obtained with previous non-linear solutions and experimental data. A non-iterative finite volume method on junction coupling and boundary treatment for flow networks were also developed by Hong & Kim (2009). A ghost junction cell was used while considering linear momentum interchange and wall effects, which eliminates the need for any empirical correlations. The method was found to be computationally more efficient than conventional methods. Satisfactory results were obtained, considering no junction loss correlations were used.

Pressure based methods, however, such as those from the computational fluid dynamics literature, can be directly extended to treat pipe junctions based a uniform pressure or simple empirical formulation. Pipe networks consisting of pipes and pipe junctions based on simple loss correlations or uniform pressure theory have been successfully integrated and solved with conventional pressure correction methods (See for example Walter et al. (2000) and Garland (1998)).

## 2.5 Conclusion

Methods for transient networks are well established. Implicit methods are suitable for slow transients and problems such as complex network transient simulation where stiffness is likely to be encountered, and is therefore commonly used as a default method for simulation codes. Explicit methods and the method of characteristics are more

suitable for single pipelines and simple networks where only fast transients are present. Semi-implicit methods, on the other hand, provide a balance between stability and computational effort.

While a total pressure formulation of conservation of momentum is commonly used for the incompressible flow in pipe networks, compressible flows are commonly either treated with the regular static pressure formulation or a total pressure formulation for ideal gases found in Greyvenstein (2002). The total pressure formulation has a potential advantage of eliminating problems associated with the convective momentum terms as well as pipe junctions and boundaries. The total pressure formulations do, in contrast with static pressure methods, require separate formulations for gas flow and liquid flow.

Although many authors adopted a generic approach of the solution of implicit methods, such as Newton's method, notable work were done by Majumdar & Ravindran (2010), Van Ravenswaay (1998), Garland (1998) and Greyvenstein (2002) on this aspect. Majumdar & Ravindran adopted a decoupling of the equation in a hybrid Newton approach, while Greyvenstein and Walter et al. adopted conventional segregated methods from CFD literature. Garland's FIBS took a different approach by maintaining the coupling of all conservation laws while performing back substitutions. The linear system of equations resulting from the FIBS is expected to be more complex than the equations resulting from the PCIM and the work of Majumdar & Ravindran, and the algorithm requires significant matrix computations and storage.

Adaptive time step methods are well established and provide an efficient means of treating transient phenomena consisting of different time scales. Adaptive time step methods have been applied to pipe networks to a limited extent. In particular, generic methods from the ODE literature have been used (Chua & Dew 1984, Gresho et al. 1980). Application to both wave phenomena and slower effects needs to be addressed. Furthermore, conventional CFD methods, which are commonly used in pipe networks as well, need to be implemented efficiently in adaptive time step algorithms.

# Chapter 3

# Transient flow in 1D pipes

## 3.1 Introduction

In the light of the aspects presented in Chapters 1 & 2, this chapter presents a non-iterative unified total pressure correction algorithm for the simulation of transient flow in one-dimensional pipes, suitable for all fluids. Consistency of the algorithm is presented through standard test problems. This is followed by a stability investigation and comparison of the total pressure and static pressure methods. Finally, steady state convergence of the various methods is investigated and compared. Non-pipe elements are not considered at this time.

## 3.2 Governing equations

For pipe elements, flow is assumed to be quasi one dimensional, assuming that the flow properties are uniform across any given cross section. The governing equations for quasi one dimensional pipe flow are given next in fully conservative differential form for the case where body/gravitational forces are neglected (Städtke 2006, Greyvenstein 2002).

**Continuity**

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u A)}{A \partial x} = 0. \tag{3.1}$$

Here $x$ and $t$ are the spatial and temporal coordinates. $\rho$ is the density, $u$ the velocity and $A$ the cross-sectional area.

**Conservation of momentum**

$$\frac{\partial m}{\partial t} + \frac{\partial (muA)}{A\partial x} + \frac{\partial p}{\partial x} = -\frac{f\rho u|u|}{2D}. \tag{3.2}$$

Here $m = \rho u$ is the momentum per unit volume, $p$ is the static pressure, $D$ the hydraulic diameter and $f$ the Darcy-Weisbach friction factor.

**Conservation of energy**

$$\frac{\partial (\rho E)}{\partial t} + \frac{\partial (\rho E u A)}{A\partial x} + \frac{\partial (upA)}{A\partial x} = \dot{q}. \tag{3.3}$$

Here $E$ is the energy defined by $E = e + \frac{1}{2}u^2$, $e$ is the internal energy and $\dot{q}$ the heat transfer rate per unit volume.

To eliminate the convective momentum term in (3.2), conservation of momentum is written in non-conservative form with the use of (3.1):

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{1}{\rho}\frac{\partial p}{\partial x} = -\frac{fu|u|}{2D}. \tag{3.4}$$

For incompressible flow this can be written as

$$\frac{\partial u}{\partial t} + \frac{1}{\rho}\frac{\partial p_0}{\partial x} = -\frac{fu|u|}{2D}, \tag{3.5}$$

where the total pressure $p_0$ is defined by $p_0 = p + \frac{1}{2}\rho u^2$.

For ideal gas flows, a similar formulation based on total pressures can be obtained, as found in Greyvenstein (2002):

$$\frac{\partial u}{\partial t} + \frac{1}{\rho}\frac{p}{p_0}\frac{\partial p_0}{\partial x} + \frac{u^2}{2T_0}\frac{\partial T_0}{\partial x} = -\frac{f|u|u}{2D}. \tag{3.6}$$

Here the total pressure $p_0$ and total temperature $T_0$ are obtained from the local isentropic relations

$$p_0 = p\left(1 + \frac{\gamma - 1}{2}M^2\right)^{\frac{\gamma}{\gamma-1}} \tag{3.7}$$

and

$$T_0 = T\left(1 + \frac{\gamma - 1}{2}M^2\right) \tag{3.8}$$

respectively, with $T$ the static temperature and $M$ the local Mach number defined as $u/\sqrt{\gamma RT}$. $\gamma$ is the ratio of specific heats and $R$ the specific gas constant. The set of equations is closed with an equation of state, which is given for an ideal gas by

$$p = \rho RT. \tag{3.9}$$

It can be argued that (3.6) reduces to (3.5) as $u \to 0$ since $p/p_0 \to 1$. This, however, is a very limited case. More general situations such as incompressible flow where $u \neq 0$, two-phase flow as well as real gas effects are not accommodated by (3.6), which pose severe restrictions on its usefulness as a unified formulation.

## 3.3 Generalized total pressure formulation

In this section a generalized total pressure formulation is introduced, which is valid for an arbitrary fluid. From the definition of total pressure and enthalpy, it follows that

$$s(p, h) = s_0(p_0, h_0) \tag{3.10}$$

$$h = h_0 - \frac{1}{2}u^2, \tag{3.11}$$

where $h_0$ is the total enthalpy and $s_0$ the entropy associated with that state. This can be written in differential form as

$$ds = ds_0 \tag{3.12}$$

$$dh = dh_0 - udu. \tag{3.13}$$

Let us also define $\rho_0 = f(p_0, h_0)$ where $f$ is the normal thermodynamic fluid property relationship. From the thermodynamic relations we also have

$$ds = \frac{1}{T}dh - \frac{1}{\rho T}dp \tag{3.14}$$

$$ds_0 = \frac{1}{T_0}dh_0 - \frac{1}{\rho_0 T_0}dp_0. \tag{3.15}$$

Substituting (3.14) and (3.15) in (3.12) gives

$$\frac{1}{T}dh - \frac{1}{\rho T}dp = \frac{1}{T_0}dh_0 - \frac{1}{\rho_0 T_0}dp_0. \tag{3.16}$$

After eliminating $dh$ with (3.13) and rearranging, this leads to

$$\frac{1}{\rho}dp + udu = \frac{T}{\rho_0 T_0}dp_0 + \left(1 - \frac{T}{T_0}\right)dh_0. \tag{3.17}$$

Combining (3.4) with (3.17) gives the final form of the generalized total pressure formulation:

$$\frac{\partial u}{\partial t} + \frac{T}{\rho_0 T_0}\frac{\partial p_0}{\partial x} + \left(1 - \frac{T}{T_0}\right)\frac{\partial h_0}{\partial x} = -\frac{f|u|u}{2D}. \tag{3.18}$$

For incompressible flows we may write that $\rho_0 = \rho$. Furthermore $T_0 = T$ regardless of velocity $u$, which follows from the fact that $s = s(T) = s(T_0)$. Therefore we again obtain the result of (3.5). For ideal gases we have that $p = \rho RT$, $p_0 = \rho_0 RT_0$ and $T_0 = T + u^2/(2c_p)$. Therefore we again obtain the result of (3.6).

## 3.4 Discretisation

A staggered grid is adopted for pipe elements and is illustrated in Figure 3.1. Scalar quantities are located at the centroid of a scalar control volume, while velocities are located at the faces of a scalar control volume. In the following sections the discretisation and solution algorithm will be illustrated for the unified formulation of (3.18).

For a pipe, consider the scalar control volume centred at $i$ as illustrated in Figure 3.1.

**Continuity** A finite volume discretisation of (3.1) leads to

$$\mathcal{V}_i \frac{d\rho_i}{dt} + (\rho u A)_{i+1/2} - (\rho u A)_{i-1/2} = 0. \tag{3.19}$$

**Conservation of energy** Similarly, a finite volume discretisation of (3.3) leads to

$$\mathcal{V}_i \frac{d(\rho E)_i}{dt} + \left[(\rho E + p)\, uA\right]_{i+1/2} - \left[(\rho E + p)\, uA\right]_{i-1/2} = \dot{Q}_i. \tag{3.20}$$

Here $\mathcal{V}$ is the volume of the control volume and $\dot{Q}$ the heat transfer rate.

In the above equations, the scalar quantities at the faces $i + \frac{1}{2}$ and $i - \frac{1}{2}$ need to be approximated. A scalar quantity $\phi$ at face $i + \frac{1}{2}$ can be approximated in various ways. For upwind interpolation, this can be written as

$$\phi_{i+1/2} = \begin{cases} \phi_i & \text{if } u_{i+1/2} \geq 0 \\ \phi_{i+1} & \text{if } u_{i+1/2} < 0 \end{cases}, \tag{3.21}$$

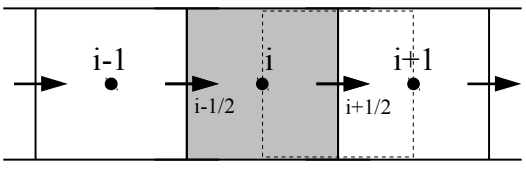



**Figure 3.1:** Finite volume discretisation for a staggered grid.

while central interpolation gives

$$\phi_{i+1/2} = \frac{\phi_i + \phi_{i+1}}{2}. \tag{3.22}$$

Higher order interpolation methods can be used as well.

### Discretisation of conservation of momentum

For a pipe, consider a control volume centred at $i + \frac{1}{2}$ as illustrated in Figure 3.1. A finite volume spatial discretisation of the non-conservative form of conservation of momentum (3.4) leads to

$$\frac{du_{i+1/2}}{dt} + u_{i+1/2}\frac{u_{i+1} - u_i}{\Delta x} + \frac{1}{\rho_{i+1/2}}\frac{p_{i+1} - p_i}{\Delta x} = -\frac{f_{i+1/2}u_{i+1/2}|u_{i+1/2}|}{2D_{i+1/2}}. \tag{3.23}$$

For the total pressure formulation (3.18), discretisation gives

$$\frac{du_{i+1/2}}{dt} + \frac{1}{\rho_{0_{i+1/2}}}\frac{T_{i+1/2}}{T_{0_{i+1/2}}}\frac{p_{0_{i+1}} - p_{0_i}}{\Delta x} + \left(1 - \frac{T_{i+1/2}}{T_{0_{i+1/2}}}\right)\frac{h_{0_{i+1}} - h_{0_i}}{\Delta x} = -\frac{f_{i+1/2}u_{i+1/2}|u_{i+1/2}|}{2D_{i+1/2}}. \tag{3.24}$$

Suitable interpolation schemes can be used as mentioned in the previous section.

### Temporal discretisation

The semi-discrete equations represented in the previous section are of the form

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(\mathbf{y}), \tag{3.25}$$

which is a stiff ordinary differential equation (ODE).

In this paper the $\alpha$-scheme will be used to integrate the system of equations, which takes the form

$$\frac{\mathbf{y}^{n+1} - \mathbf{y}^n}{\Delta t} = (1 - \alpha)\mathbf{f}(\mathbf{y}^n) + \alpha\mathbf{f}(\mathbf{y}^{n+1}), \tag{3.26}$$

with $\alpha \in [0, 1]$ and $n$ the discrete temporal index. For $\alpha = 0$ the method reduces to the explicit Euler method, for $\alpha = 1$ the implicit Euler method and for $\alpha = 0.5$ the trapezoidal method. The method is second order accurate for the latter. It is A-stable for $0.5 \leq \alpha \leq 1$ and L-stable for $\alpha = 1$. For a balance between L-stability and accuracy, it is common practice to choose $\alpha$ in the neighbourhood of 0.7. For each time step, an implicit $\alpha$-scheme ($\alpha > 0$) will require the solution of an implicit system of non-linear equations for $\mathbf{y}^{n+1}$, which will be discussed next.

## 3.5 Solution algorithm

The $\alpha$-scheme results in an implicit system of non-linear equations. These equations can be solved exactly with fully coupled solvers or segregated solvers such as the family of SIMPLE algorithms. This is the approach of Greyvenstein's method (Greyvenstein 2002). Alternatively, approximate solutions can be utilized while ensuring that the errors introduced do not dominate the temporal discretisation. This is the idea of the PISO algorithm (Issa 1986). This leads to a non-iterative method. In this paper a similar non-iterative algorithm will be presented, based on the generalized total pressure formulation.

In order to construct the pressure correction procedure, the first step in the algorithm is an explicit velocity predictor $\tilde{u}_{i+1/2}$ from conservation of momentum:

$$\begin{aligned}
&\frac{\tilde{u}_{i+1/2} - u^n_{i+1/2}}{\Delta t} \\
&+ \alpha \left[ \frac{1}{\rho^n_{0_{i+1/2}}} \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}} \frac{p^n_{0_{i+1}} - p^n_{0_i}}{\Delta x} + \left(1 - \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}}\right) \frac{h^n_{0_{i+1}} - h^n_{0_i}}{\Delta x} + \frac{f_{i+1/2} |u^n_{i+1/2}| \tilde{u}_{i+1/2}}{2D_{i+1/2}} \right] \\
&+ (1-\alpha) \left[ \frac{1}{\rho^n_{0_{i+1/2}}} \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}} \frac{p^n_{0_{i+1}} - p^n_{0_i}}{\Delta x} + \left(1 - \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}}\right) \frac{h^n_{0_{i+1}} - h^n_{0_i}}{\Delta x} + \frac{f_{i+1/2} |u^n_{i+1/2}| u^n_{i+1/2}}{2D_{i+1/2}} \right] \\
&= 0.
\end{aligned} \tag{3.27}$$

Next, the velocity $u^*_{i+1/2}$ is obtained, based on the new total pressures $p^*_{0_{i+1}}$ and $p^*_{0_i}$ such that continuity is satisfied. From conservation of momentum:

$$\begin{aligned}
&\frac{u^*_{i+1/2} - u^n_{i+1/2}}{\Delta t} \\
&+ \alpha \left[ \frac{1}{\rho^n_{0_{i+1/2}}} \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}} \frac{p^*_{0_{i+1}} - p^*_{0_i}}{\Delta x} + \left(1 - \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}}\right) \frac{h^n_{0_{i+1}} - h^n_{0_i}}{\Delta x} + \frac{f_{i+1/2} |u^n_{i+1/2}| u^*_{i+1/2}}{2D_{i+1/2}} \right] \\
&+ (1-\alpha) \left[ \frac{1}{\rho^n_{0_{i+1/2}}} \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}} \frac{p^n_{0_{i+1}} - p^n_{0_i}}{\Delta x} + \left(1 - \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}}\right) \frac{h^n_{0_{i+1}} - h^n_{0_i}}{\Delta x} + \frac{f_{i+1/2} |u^n_{i+1/2}| u^n_{i+1/2}}{2D_{i+1/2}} \right] \\
&= 0.
\end{aligned} \tag{3.28}$$

(3.27) is subtracted from (3.28) to give the velocity correction defined by $\delta u^*_{i+1/2} = u^*_{i+1/2} - \tilde{u}_{i+1/2}$:

$$\frac{\delta u^*_{i+1/2}}{\Delta t} + \alpha \left( \frac{1}{\rho^n_{0_{i+1/2}}} \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}} \frac{\delta p^*_{0_{i+1}} - \delta p^*_{0_i}}{\Delta x} + \frac{f_{i+1/2} |u^n_{i+1/2}| \delta u^*_{i+1/2}}{2D_{i+1/2}} \right) = 0, \tag{3.29}$$

or

$$\delta u^*_{i+1/2} = \alpha \frac{1}{\rho^n_{0_{i+1/2}}} \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}} \Bigg/ \left(1 + \alpha \Delta t \frac{f_{i+1/2}|u^n_{i+1/2}|}{2D_{i+1/2}}\right) \left(\delta p^*_{0_i} - \delta p^*_{0_{i+1}}\right), \tag{3.30}$$

which can be written as

$$\delta u^*_{i+1/2} = c_{i+1/2} \left(\delta p^*_{0_i} - \delta p^*_{0_{i+1}}\right) \tag{3.31}$$

where

$$c_{i+1/2} = \alpha \frac{1}{\rho^n_{0_{i+1/2}}} \frac{T^n_{i+1/2}}{T^n_{0_{i+1/2}}} \Bigg/ \left(1 + \alpha \Delta t \frac{f_{i+1/2}|u^n_{i+1/2}|}{2D_{i+1/2}}\right). \tag{3.32}$$

Similarly, for $u^*_{i-1/2}$:

$$\delta u^*_{i-1/2} = c_{i-1/2} \left(\delta p^*_{0_{i-1}} - \delta p^*_{0_i}\right) \tag{3.33}$$

and

$$c_{i-1/2} = \alpha \frac{1}{\rho^n_{0_{i-1/2}}} \frac{T^n_{i-1/2}}{T^n_{0_{i-1/2}}} \Bigg/ \left(1 + \alpha \Delta t \frac{f_{i-1/2}|u^n_{i-1/2}|}{2D_{i-1/2}}\right). \tag{3.34}$$

Here $\delta p^*_{0_{i+1}}$ and $\delta p^*_{0_i}$ are the first pressure corrections defined by $\delta p^*_{0_{i+1}} = p^*_{0_{i+1}} - p^n_{0_{i+1}}$ and $\delta p^*_{0_i} = p^*_{0_i} - p^n_{0_i}$.

The linearised continuity equation about the previous time step $n$ for a control volume centred at $i$ is

$$\begin{aligned}
&\mathcal{V}_i \frac{\rho\left(p^n_{0_i}, e^n_i, u^n_i\right) + \left(\frac{\partial \rho}{\partial p_0}\right)^n_i \delta p^*_{0_i} - \rho^n_i}{\Delta t} \\
&+ \alpha \left\{ \left[ \rho^n_{i+1/2} u^*_{i+1/2} + u^n_{i+1/2} \left(\frac{\partial \rho}{\partial p_0}\right)^n_{i+1/2} \delta p^*_{0_{i+1/2}} \right] A_{i+1/2} \right. \\
&\qquad \left. - \left[ \rho^n_{i-1/2} u^*_{i-1/2} - u^n_{i-1/2} \left(\frac{\partial \rho}{\partial p_0}\right)^n_{i-1/2} \delta p^*_{0_{i-1/2}} \right] A_{i-1/2} \right\} \\
&+ (1-\alpha)\left(\rho^n_{i+1/2} u^n_{i+1/2} A_{i+1/2} - \rho^n_{i-1/2} u^n_{i-1/2} A_{i-1/2}\right) = 0.
\end{aligned} \tag{3.35}$$

Here it has been assumed that $\rho = \rho\left(p_0, e, u\right)$. For an ideal gas it can be shown that

$$\frac{\partial \rho}{\partial p_0} = \frac{\rho}{p_0}. \tag{3.36}$$

By applying upwind differencing of the density at faces $i + \frac{1}{2}$ and $i - \frac{1}{2}$ and substi-

tuting the velocity correction as defined in (3.31) leads to

$$\begin{aligned}
&\mathcal{V}_i \frac{\rho\left(p_{0_i}^n, e_i^n, u_i^n\right) + \left(\frac{\partial\rho}{\partial p_0}\right)_i^n \delta p_{0_i}^* - \rho_i^n}{\Delta t} \\
&+ \alpha \left[\rho_{i+1/2}^n \left(\delta u_{i+1/2}^* + \tilde{u}_{i+1/2}\right) + \max\left(u_{i+1/2}^n, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_i^n \delta p_{0_i}^* \right. \\
&\qquad \left. + \min\left(u_{i+1/2}^n, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_{i+1}^n \delta p_{0_{i+1}}^*\right] A_{i+1/2} \\
&- \alpha \left[\rho_{i-1/2}^n \left(\delta u_{i-1/2}^* + \tilde{u}_{i-1/2}\right) + \min\left(u_{i-1/2}^n, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_i^n \delta p_{0_i}^* \right. \\
&\qquad \left. + \max\left(u_{i-1/2}^n, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_{i-1}^n \delta p_{0_{i-1}}^*\right] A_{i-1/2} \\
&+ r_i^n = 0,
\end{aligned} \tag{3.37}$$

with

$$r_i^n = (1-\alpha)\left(\rho_{i+1/2}^n u_{i+1/2}^n A_{i+1/2} - \rho_{i-1/2}^n u_{i-1/2}^n A_{i-1/2}\right). \tag{3.38}$$

Substitution of (3.31) and (3.33) into (3.37) leads to the final set of equations for total pressure correction, which takes the form

$$a_{i,i-1}\delta p_{0_{i-1}}^* + a_{i,i}\delta p_{0_i}^* + a_{i,i+1}\delta p_{0_{i+1}}^* = b_i, \tag{3.39}$$

where

$$\begin{aligned}
a_{i,i} = &\frac{\mathcal{V}_i}{\Delta t}\left(\frac{\partial\rho}{\partial p_0}\right)_i^n \\
&+ \alpha \left\{ \begin{array}{l} \rho_{i+1/2}^n c_{i+1/2} A_{i+1/2} + \rho_{i-1/2}^n c_{i-1/2} A_{i-1/2} \\ + \left[\max\left(u_{i+1/2}^n, 0\right) A_{i+1/2} + \max\left(-u_{i-1/2}^n, 0\right) A_{i-1/2}\right] \left(\frac{\partial\rho}{\partial p_0}\right)_i^n \end{array} \right\}
\end{aligned} \tag{3.40}$$

$$a_{i,i-1} = -\alpha \left[\rho_{i-1/2}^n c_{i-1/2} + \max\left(u_{i-1/2}^n, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_{i-1}^n\right] A_{i-1/2} \tag{3.41}$$

$$a_{i,i+1} = -\alpha \left[\rho_{i+1/2}^n c_{i+1/2} + \max\left(-u_{i+1/2}^n, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_{i+1}^n\right] A_{i+1/2} \tag{3.42}$$

$$b_i = -r_i^n - r_i^*, \tag{3.43}$$

and

$$r_i^* = \mathcal{V}_i \frac{\rho\left(p_{0_i}^n, e_i^n, u_i^n\right) - \rho_i^n}{\Delta t} + \alpha\rho_{i+1/2}^n \tilde{u}_{i+1/2} A_{i+1/2} - \alpha\rho_{i-1/2}^n \tilde{u}_{i-1/2} A_{i-1/2}. \tag{3.44}$$

Equation (3.39) is solved for the total pressure correction $\delta p_0^*$, after which the total pressure $p_0^*$ and velocity $u^*$ can be updated from (3.30). The preliminary static pressure $p^*$ can be updated from the newly obtained total pressure and velocity with the relations (3.10) and (3.11) together with the equation of state to give an expression of the form $p^* = p\left(p_{0_i}^*, e_i^n, u_i^*\right)$.

The density $\rho^*$ is updated from continuity (3.35):

$$\rho^* = \rho\left(p_{0_i}^n, e_i^n, u_i^n\right) + \left(\frac{\partial \rho}{\partial p_0}\right)_i^n \delta p_{0_i}^*. \tag{3.45}$$

Next, $\rho E^*$ is solved from conservation of energy, where the available velocities $u^*$ and pressures $p^*$ are used. This gives:

$$\begin{aligned}
&\mathcal{V}_i \frac{(\rho E)_i^* - (\rho E)_i^n}{\Delta t} \\
&+ \alpha \left\{ \left[ (\rho E)_{i+1/2}^* + p_{i+1/2}^* \right] u_{i+1/2}^* A_{i+1/2} - \left[ (\rho E)_{i-1/2}^* + p_{i-1/2}^* \right] u_{i-1/2}^* A_{i-1/2} \right\} \\
&+ (1-\alpha) \left\{ \left[ (\rho E)_{i+1/2}^n + p_{i+1/2}^n \right] u_{i+1/2}^n A_{i+1/2} - \left[ (\rho E)_{i-1/2}^n + p_{i-1/2}^n \right] u_{i-1/2}^n A_{i-1/2} \right\} \\
&+ - \left[ \alpha \dot{Q}_i^{n+1} + (1-\alpha) \dot{Q}_i^n \right] = 0.
\end{aligned} \tag{3.46}$$

With $(\rho E)^*$ known, internal energy can be calculated from

$$e_i^* = (\rho E)_i^* / \rho_i^* - \frac{1}{2}\left(u_i^*\right)^2, \tag{3.47}$$

after which the static enthalpy $h^*$ and total enthalpy $h_0^*$ can also be updated.

The final step in the solution procedure is the calculation of the second total pressure correction $\delta p_0^{**} = p_0^{**} - p_0^*$. This step is not necessary for constant density flows. This is carried out in a similar manner as presented earlier, and will be described next.

Once again, a velocity predictor $\tilde{u}_{i+1/2}$ is obtained explicitly from conservation of momentum, with the latest pressure $p_0 = p_0^*$ and enthalpy $h_0 = h_0^*$ used in the derivatives:

$$\begin{aligned}
&\frac{\tilde{u}_{i+1/2} - u_{i+1/2}^n}{\Delta t} \\
&+ \alpha \left[ \frac{1}{\rho_{0_{i+1/2}}^n} \frac{T_{i+1/2}^n}{T_{0_{i+1/2}}^n} \frac{p_{0_{i+1}}^* - p_{0_i}^*}{\Delta x} + \left(1 - \frac{T_{i+1/2}^n}{T_{0_{i+1/2}}^n}\right) \frac{h_{0_{i+1}}^* - h_{0_i}^*}{\Delta x} + \frac{f_{i+1/2} |u_{i+1/2}^n| \tilde{u}_{i+1/2}}{2 D_{i+1/2}} \right] \\
&+ (1-\alpha) \left[ \frac{1}{\rho_{0_{i+1/2}}^n} \frac{T_{i+1/2}^n}{T_{0_{i+1/2}}^n} \frac{p_{0_{i+1}}^n - p_{0_i}^n}{\Delta x} + \left(1 - \frac{T_{i+1/2}^n}{T_{0_{i+1/2}}^n}\right) \frac{h_{0_{i+1}}^n - h_{0_i}^n}{\Delta x} + \frac{f_{i+1/2} |u_{i+1/2}^n| u_{i+1/2}^n}{2 D_{i+1/2}} \right] \\
&= 0.
\end{aligned} \tag{3.48}$$

Next, the velocity $u^{**}_{i+1/2}$ is obtained, based on the new total pressures $p^{**}_{0_{i+1}}$ such that continuity is satisfied. From conservation of momentum:

$$\begin{aligned}
&\frac{u^{**}_{i+1/2} - u^{n}_{i+1/2}}{\Delta t} \\
&+ \alpha \left[ \frac{1}{\rho^{n}_{0_{i+1/2}}} \frac{T^{n}_{i+1/2}}{T^{n}_{0_{i+1/2}}} \frac{p^{**}_{0_{i+1}} - p^{**}_{0_i}}{\Delta x} + \left( 1 - \frac{T^{n}_{i+1/2}}{T^{n}_{0_{i+1/2}}} \right) \frac{h^{*}_{0_{i+1}} - h^{*}_{0_i}}{\Delta x} + \frac{f_{i+1/2} |u^{n}_{i+1/2}| u^{**}_{i+1/2}}{2D_{i+1/2}} \right] \\
&+ (1-\alpha) \left[ \frac{1}{\rho^{n}_{0_{i+1/2}}} \frac{T^{n}_{i+1/2}}{T^{n}_{0_{i+1/2}}} \frac{p^{n}_{0_{i+1}} - p^{n}_{0_i}}{\Delta x} + \left( 1 - \frac{T^{n}_{i+1/2}}{T^{n}_{0_{i+1/2}}} \right) \frac{h^{n}_{0_{i+1}} - h^{n}_{0_i}}{\Delta x} + \frac{f_{i+1/2} |u^{n}_{i+1/2}| u^{n}_{i+1/2}}{2D_{i+1/2}} \right] \\
&= 0.
\end{aligned} \tag{3.49}$$

(3.48) is subtracted from (3.49) to give the velocity correction defined by $\delta u^{**}_{i+1/2} = u^{**}_{i+1/2} - \tilde{u}_{i+1/2}$:

$$\frac{\delta u^{**}_{i+1/2}}{\Delta t} + \alpha \left( \frac{1}{\rho^{n}_{0_{i+1/2}}} \frac{T^{n}_{i+1/2}}{T^{n}_{0_{i+1/2}}} \frac{\delta p^{**}_{0_{i+1}} - \delta p^{**}_{0_i}}{\Delta x} + \frac{f_{i+1/2} |u^{n}_{i+1/2}| \delta u^{**}_{i+1/2}}{2D_{i+1/2}} \right) = 0, \tag{3.50}$$

or

$$\delta u^{**}_{i+1/2} = \alpha \frac{\Delta t}{\Delta x} \frac{1}{\rho^{n}_{0_{i+1/2}}} \frac{T^{n}_{i+1/2}}{T^{n}_{0_{i+1/2}}} \Bigg/ \left( 1 + \alpha \Delta t \frac{f_{i+1/2} |u^{n}_{i+1/2}|}{2D_{i+1/2}} \right) \left( \delta p^{**}_{0_i} - \delta p^{**}_{0_{i+1}} \right), \tag{3.51}$$

which can be written as

$$\delta u^{**}_{i+1/2} = c_{i+1/2} \left( \delta p^{**}_{0_i} - \delta p^{**}_{0_{i+1}} \right). \tag{3.52}$$

Similarly, for $u^{**}_{i-1/2}$:

$$\delta u^{**}_{i-1/2} = c_{i-1/2} \left( \delta p^{**}_{0_{i-1}} - \delta p^{**}_{0_i} \right). \tag{3.53}$$

Here $\delta p^{**}_{0_{i+1}}$ and $\delta p^{**}_{0_i}$ are the second pressure corrections defined by $\delta p^{**}_{0_{i+1}} = p^{**}_{0_{i+1}} - p^{*}_{0_{i+1}}$ and $\delta p^{**}_{0_i} = p^{**}_{0_i} - p^{*}_{0_i}$.

The linearised continuity equation about the preliminary solution $*$ for a control volume centred at $i$ is

$$\begin{aligned}
&\mathcal{V}_i \frac{\rho \left( p^{*}_{0_i}, e^{*}_{i}, u^{*}_{i} \right) + \left( \frac{\partial \rho}{\partial p_0} \right)^{*}_{i} \delta p^{**}_{0_i} - \rho^{n}_{i}}{\Delta t} \\
&+ \alpha \left\{ \left[ \rho^{*}_{i+1/2} u^{**}_{i+1/2} + u^{*}_{i+1/2} \left( \frac{\partial \rho}{\partial p_0} \right)^{*}_{i+1/2} \delta p^{**}_{0_{i+1/2}} \right] A_{i+1/2} \right. \\
&\qquad \left. - \left[ \rho^{*}_{i-1/2} u^{**}_{i-1/2} - u^{*}_{i-1/2} \left( \frac{\partial \rho}{\partial p_0} \right)^{*}_{i-1/2} \delta p^{**}_{0_{i-1/2}} \right] A_{i-1/2} \right\} \\
&+ (1-\alpha) \left( \rho^{n}_{i+1/2} u^{n}_{i+1/2} A_{i+1/2} - \rho^{n}_{i-1/2} u^{n}_{i-1/2} A_{i-1/2} \right) = 0.
\end{aligned} \tag{3.54}$$

By applying upwind differencing of the density at faces $i+\frac{1}{2}$ and $i-\frac{1}{2}$ and applying the definition of the velocity correction in (3.54) leads to

$$\begin{aligned}
&\mathcal{V}_i \frac{\rho\left(p_{0_i}^*, e_i^*, u_i^*\right) + \left(\frac{\partial\rho}{\partial p_0}\right)_i^* \delta p_{0_i}^{**} - \rho_i^n}{\Delta t} \\
&+ \alpha \left[ \rho_{i+1/2}^* \left(\delta u_{i+1/2}^{**} + \tilde{u}_{i+1/2}\right) + \max\left(u_{i+1/2}^*, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_i^* \delta p_{0_i}^* \right. \\
&\qquad \left. + \min\left(u_{i+1/2}^*, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_{i+1}^* \delta p_{0_{i+1}}^* \right] A_{i+1/2} \\
&- \alpha \left[ \rho_{i-1/2}^n \left(\delta u_{i-1/2}^{**} + \tilde{u}_{i-1/2}\right) + \min\left(u_{i-1/2}^*, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_i^* \delta p_{0_i}^{**} \right. \\
&\qquad \left. + \max\left(u_{i-1/2}^*, 0\right) \left(\frac{\partial\rho}{\partial p_0}\right)_{i-1}^* \delta p_{0_{i-1}}^{**} \right] A_{i-1/2} \\
&+ r_i^n = 0.
\end{aligned} \tag{3.55}$$

Substitution of (3.52) and (3.53) into (3.55) leads to the final set of equations for the second total pressure correction, which again takes the form

$$a_{i,i-1}\delta p_{0_{i-1}}^{**} + a_{i,i}\delta p_{0_i}^{**} + a_{i,i+1}\delta p_{0_{i+1}}^{**} = b_i, . \tag{3.56}$$

with newly obtained coefficients. In the second total pressure correction, linearisation of the continuity equation has been done about the preliminary solution indicated by $*$. Alternatively, linearisation could have been performed about the solution at the previous time step $n$. The resulting equations would have the same coefficients as the first pressure correction.

Equation (3.56) is solved for the second total pressure correction $\delta p_0^{**}$, after which the total pressure $p_0^{**}$ and velocity $u^{**}$ can be updated from (3.51). The pressure $p^{**}$ can be updated in the same manner as in the first total pressure correction.

The density $\rho^{**}$ is once again updated from continuity (3.54):

$$\rho^{**} = \rho\left(p_{0_i}^*, e_i^*, u_i^*\right) + \left(\frac{\partial\rho}{\partial p_0}\right)_i^* \delta p_{0_i}^*. \tag{3.57}$$

At this stage the solution will be consistent in time. This means that the difference between this approximate and the exact solutions of the equations resulting from the $\alpha$-scheme will be at least $O(\Delta t^2)$. Additional iterations can be performed in order to better satisfy the temporal discretised equations (such as PISO and its variants), and will not be discussed here. This completes the derivation of the non-iterative generalized total pressure based method. A summary will be given next.

**Summary** A minimalistic non-iterative procedure based on a generalized total pressure formulation can be summarised as follows:

1. Assemble and solve the linearised momentum equations for velocity $\tilde{u}$. This is done explicitly for the total pressure method.

2. Assemble and solve the total pressure-correction equation and obtain $p_0^*$. Update the velocity $u^*$ from conservation of momentum and density $\rho^*$ from continuity as well as $p^*$ from the total pressure relation.

3. Assemble and solve the energy equation for internal energy $e^{n+1} = e^*$.

4. Solve the second linearised momentum equations for velocity $\tilde{u}$.

5. Assemble and solve the second total pressure-correction equation and obtain $p_0^{n+1} = p_0^{**}$. Update the velocity $u^{n+1} = u^{**}$ from conservation of momentum and density $\rho^{n+1}$ from continuity as well as $p^{n+1} = p^{**}$ from the total pressure relation.

6. Advance to the next time step.

The derivation of a static pressure based method is similar and will not be presented here. The main differences are the calculation of a possible implicit velocity predictor as well as constructing the pressure correction equation from the static pressure $p$ rather than total pressure $p_0$.

## 3.6 Validation

The validation of the non-iterative method based on the generalized total pressure method is discussed in this section. Two commonly used test problems are considered. The first is a Riemann problem which is commonly used in the computational fluid dynamics literature. The second is the sudden closure of a valve in a pipeline, which is an established test problem in the field of pipe networks and pipelines. Results are compared with conventional static pressure methods, based on both conservative and non-conservative formulations.

**Riemann problems** For Riemann problems, the initial data consists of two uniform states that are separated by a discontinuity at the origin. As a first Riemann test problem, consider the shock tube problem of Sod (Wesseling 2001). For $x \leq 0$ the initial conditions are $p_L$, $\rho_L$ and $u_L$. For $x > 0$ the initial conditions are $p_R$, $\rho_R$ and $u_R$. Initial conditions are summarised as

$$p_L = 1 \qquad \rho_L = 1 \qquad u_L = 0 \qquad p_R = 0.1 \qquad \rho_R = 0.125 \qquad u_R = 0.$$

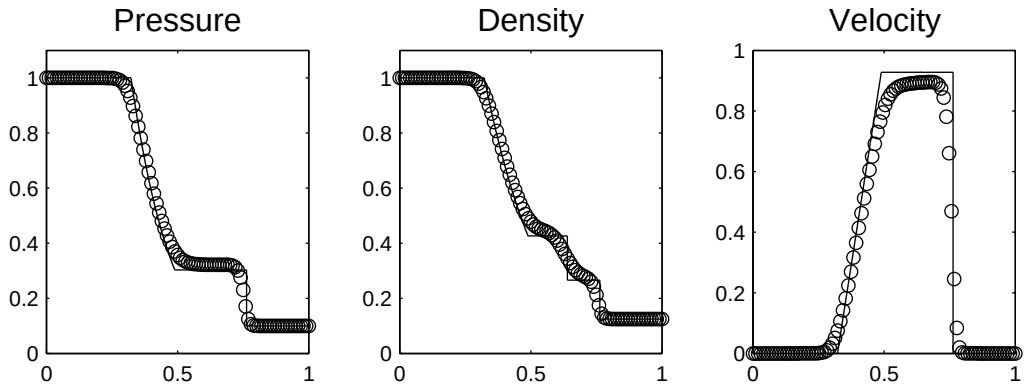
The values of $\gamma = 1.4$ and $R = 1$ were chosen for this numerical experiment.

The numerical results are illustrated in Figure 3.2 for three methods at $t = 0.15$. The results are provided in the form of pressure, density and velocity distributions along the length of the tube for each of three selected methods. The three methods are the total pressure method derived above, a non-conservative static pressure method based on (3.4) and a conservative static pressure method based on (3.2). For all three methods, $\Delta t/\Delta x = 1/4$ was chosen. The analytical solution is given by the solid line in all graphs.

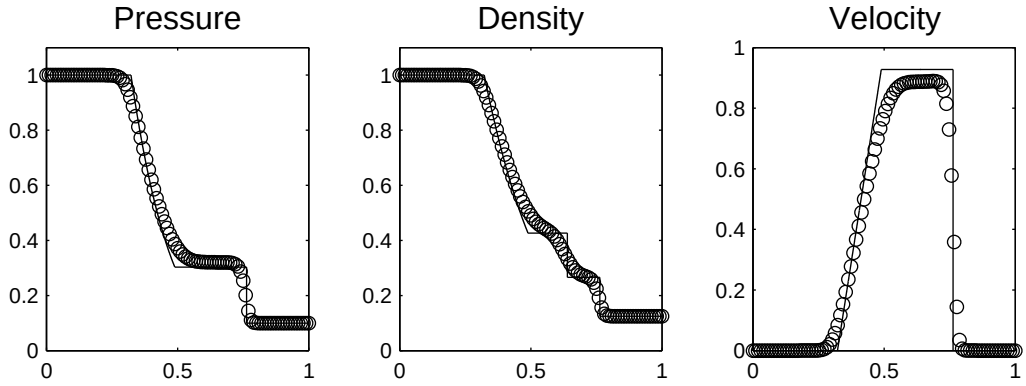
The total pressure method compares favourably with the analytical solution, and yields results similar to that of the non-conservative static pressure method. Both these non-conservative methods will converge to the incorrect shock speed (Laney 1998), although discrepancies are usually negligible for most fluid flow network applications. The conservative method will converge to the correct shock speed.

To illustrate some inconsistent variations, some results of inconsistent total pressure methods are given for Sod's tests problem in Figure 3.3. Figure 3.3a illustrates a non-iterative method which updates enthalpy $h$ rather than $e$. The inconsistency is due to the presence of the term $\partial p/\partial t$, which uses the yet inconsistent pressure. Figure 3.3b illustrates a non-iterative method that does not include a second pressure update, leaving the pressure inconsistent at the new time step.

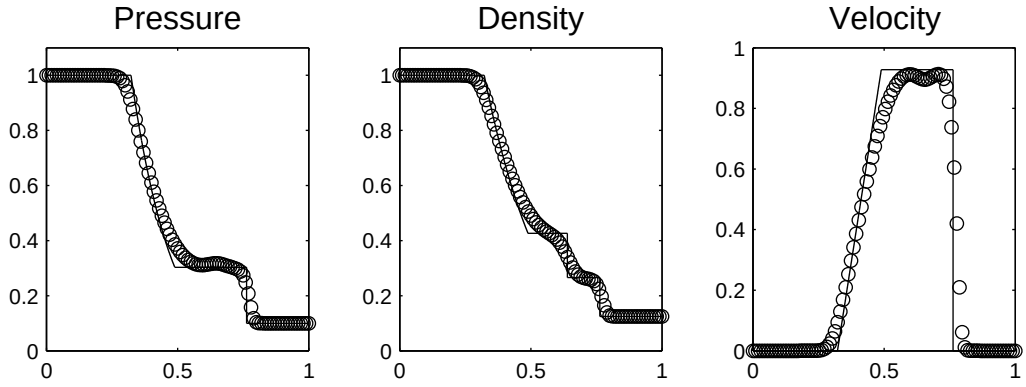
**Sudden valve closure** The test problem in this section is the sudden closure of a downstream valve in a single pipe with fluid initially at steady state. It has been established as a common test problem for pipelines and can be found in numerous publications (See for example Greyvenstein (2002), Majumdar & Ravindran (2010)), where the method of characteristics has been primarily used as a benchmark solution.




**(a)** Total pressure formulation



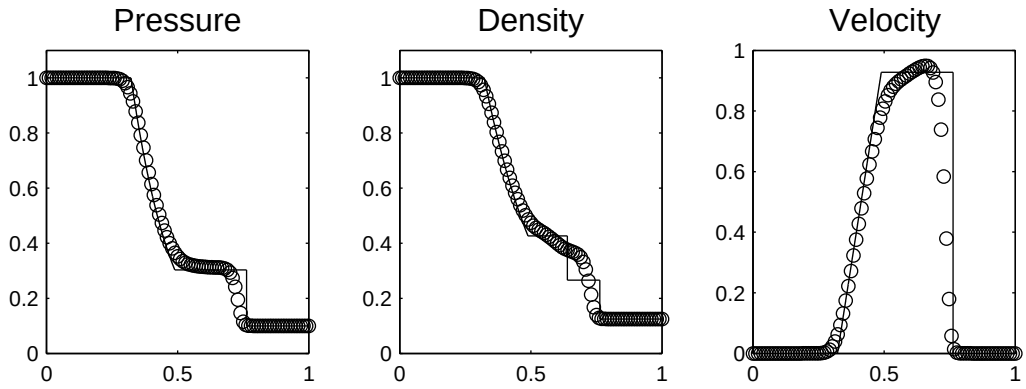

**(b)** Non-conservative static pressure formulation.



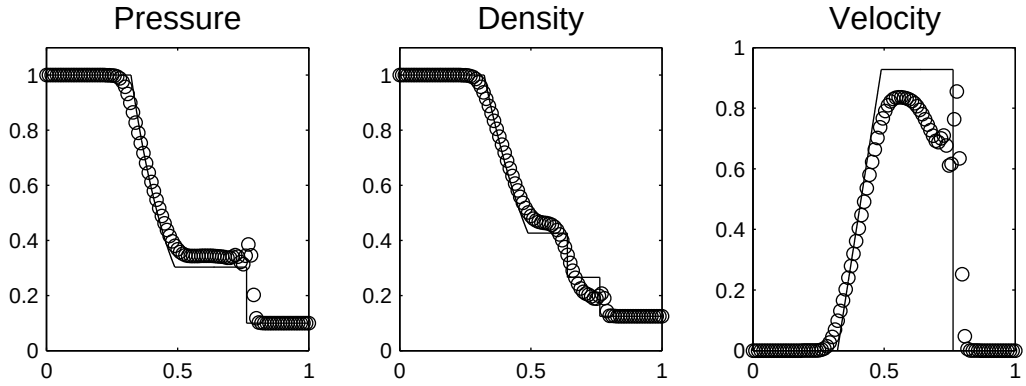

**(c)** Conservative static pressure formulation.

**Figure 3.2:** Sod's Shock Tube problem for static and total pressure formulations. $t = 0.15$.

The specific problem considered is the non-isothermal problem found in Greyvenstein (2002). The fluid is helium. The pipe is 20 m long and has a 0.5 m diameter. A constant friction factor of 0.02 was used. The initial conditions were the steady state solution in which the inlet pressure was 700 kPa, inlet temperature was 300 K and the mass flow rate was 44.86 kg.s$^{-1}$. The pipe was divided into 20 equal increments. At the




**(a)** Inconsistent formulation, enthalpy based.



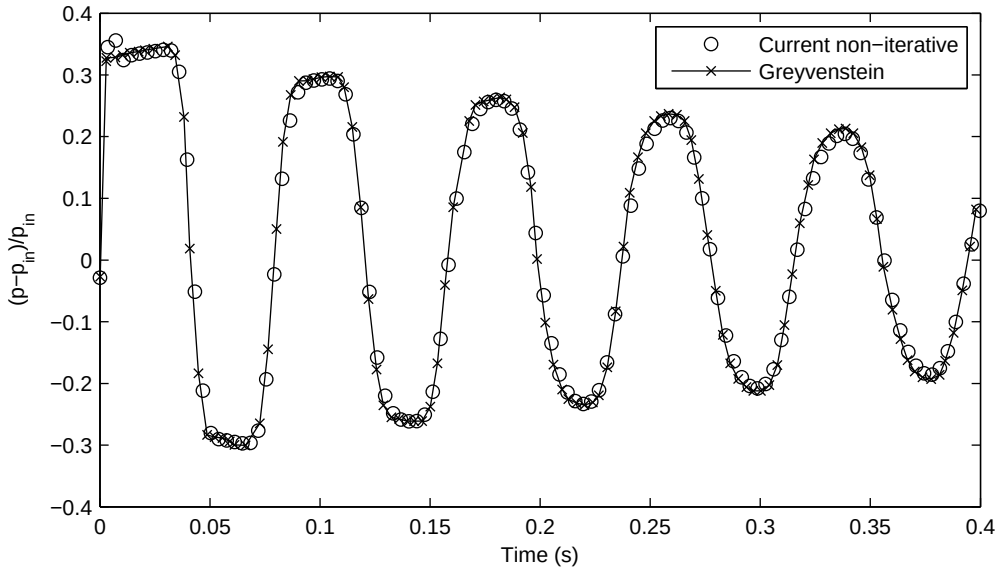

**(b)** Inconsistent formulation, no pressure update.

**Figure 3.3:** Sod's Shock Tube, inconsistent formulations. $t = 0.15$.

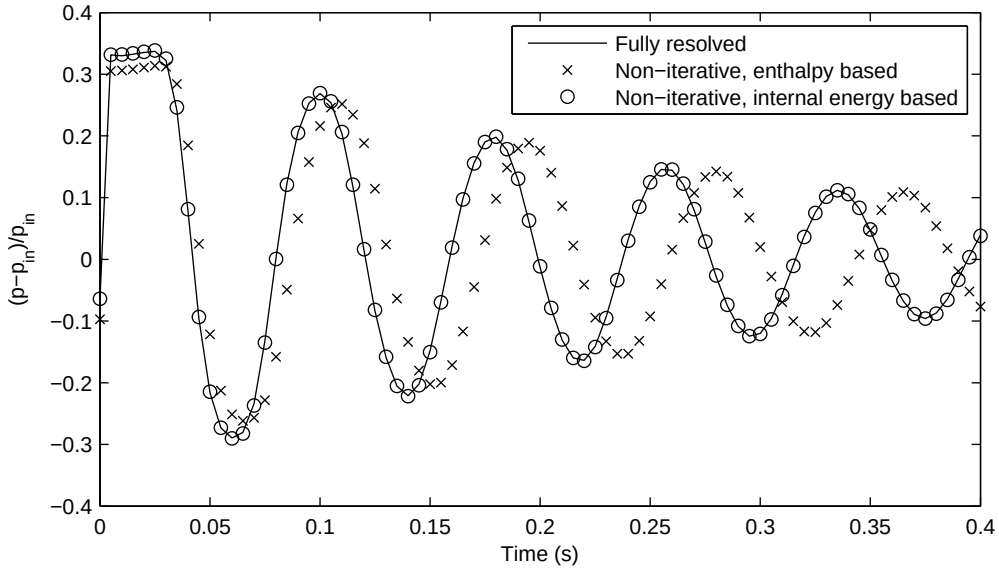
onset of the transient the valve at the outlet was closed instantaneously, corresponding to the outlet boundary condition for mass flow $\dot{m} = 0$ kg.s$^{-1}$. A time step of 0.0009 s was used.

Figure 3.4a shows the results of the non-iterative total pressure formulation introduced earlier compared with the total pressure method from Greyvenstein (2002). The variation of the pressure at the valve position versus the elapsed time is illustrated after the sudden closure. In both methods, a value of $\alpha = 0.6$ was used. The results compare favourably. The only notable difference is right at the onset of the transient where some overshoots are observed with the current method. This is only partially a result of the increased explicit treatment of certain terms in the current non-iterative method. This phenomena is mostly attributed to the spatial discretisation used, and can be largely eliminated by spatial mesh refinement.

Similar to the previous section, an inconsistent formulation based on updating the enthalpy rather than the internal energy is illustrated in Figure 3.4b. In addition, the




**(a)** Comparison with results from Greyvenstein (2002). $\alpha = 0.6$.




**(b)** Illustration of inconsistent formulations. $\alpha = 1$.

**Figure 3.4:** Pressure variation at the valve, total pressure formulation.

fully resolved solution of the $\alpha$-scheme is presented which corresponds to the method of Greyvenstein (2002). For these two methods as well as the current non-iterative method, a value of $\alpha = 1$ was chosen, and the fully resolved solution will correspond to the implicit Euler method. Once again the consistency of the current method can be observed. In addition, as compared to Figure 3.4a, a value of $\alpha = 1$ gives considerably more numerical dissipation. The initial overshoots, however, are reduced.

## 3.7 Stability investigation

Stability of the current non-iterative generalized total pressure formulation is compared with commonly used non-iterative static pressure methods. Comparisons between the fully iterative methods which exactly solve the equations resulting from the $\alpha$-scheme were not considered, as these methods would likely be unconditionally stable. The current method is compared with the following two static pressure methods:

- Static pressure method with explicit treatment of convective terms in conservation of momentum.
- Static pressure method with an implicit velocity predictor step from conservation of momentum.

The non-conservative formulations of conservation of momentum have been used throughout, as it has been found to be more efficient and stable than its conservative counterpart. Furthermore, the non-conservative form highlights differences solely attributed to conservation of momentum, since continuity and conservation of energy are treated the same for all methods. Stability was investigated with test problems consisting of various Riemann problems with different initial states that are quite close together, which means that the Riemann problem can be regarded as a small perturbation of the initial conditions. Similar investigations can be found in van der Heul et al. (2001). The initial conditions of the test problems for various velocities are given in Table 3.1, where the subscripts $L$ and $R$ correspond to the left and right hand side initial states respectively

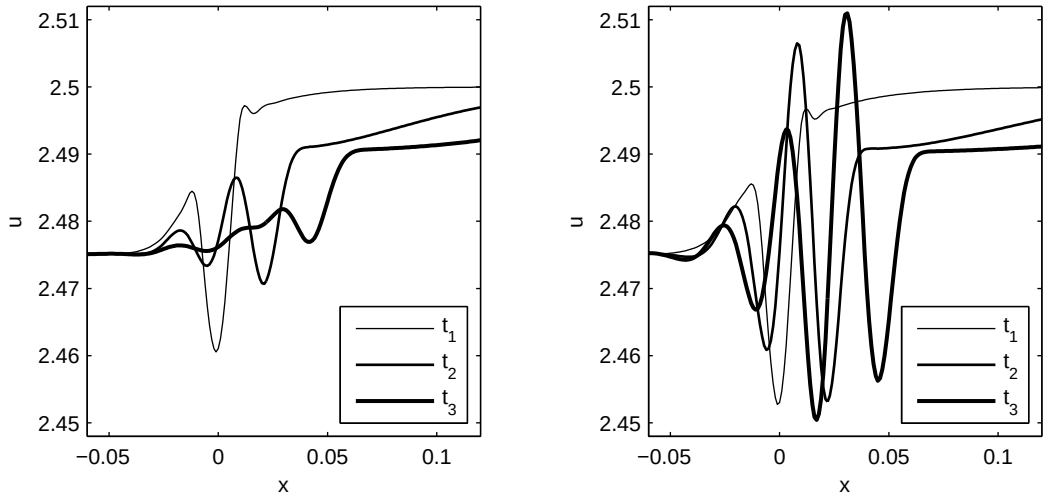
Stability was investigated for the various Mach numbers $M$ resulting from the different test problems. The maximum value of the CFL number (CFL $= |u|\Delta t/\Delta x$) which provided stable integration was then obtained by varying $\Delta t$. Examples of stable and unstable integration with the same non-iterative total pressure method is presented in Figure 3.5. This is the third test problem in Table 3.1. For this case values of $\gamma = 1.4$ and $R = 1$ were used, which corresponds to a Mach number of $M = 1.5$. The figure shows the velocity distribution along the first ten percent of the length of the tube at three different time steps namely $t_1 = 0.0025$, $t_2 = 0.01$ and $t_3 = 0.0175$.

In Figure 3.5a, integration is stable for CFL $= 3.25$ and oscillations are damped. In Figure 3.5b, integration is unstable for the slightly larger CFL number of 4 and

**Table 3.1:** Initial conditions for Riemann test problems.

| | $p_L$ | $\rho_L$ | $u_L$ | $p_R$ | $\rho_R$ | $u_R$ |
|---|---|---|---|---|---|---|
| 1 | 2.01 | 1.01 | 1.485 | 2 | 1 | 1.5 |
| 2 | 2.01 | 1.01 | 1.98 | 2 | 1 | 2 |
| 3 | 2.01 | 1.01 | 2.475 | 2 | 1 | 2.5 |
| 4 | 2.01 | 1.01 | 2.97 | 2 | 1 | 3 |
| 5 | 2.01 | 1.01 | 3.465 | 2 | 1 | 3.5 |
| 6 | 2.01 | 1.01 | 3.96 | 2 | 1 | 4 |
| 7 | 2.01 | 1.01 | 4.455 | 2 | 1 | 4.5 |
| 8 | 2.01 | 1.01 | 4.95 | 2 | 1 | 5 |

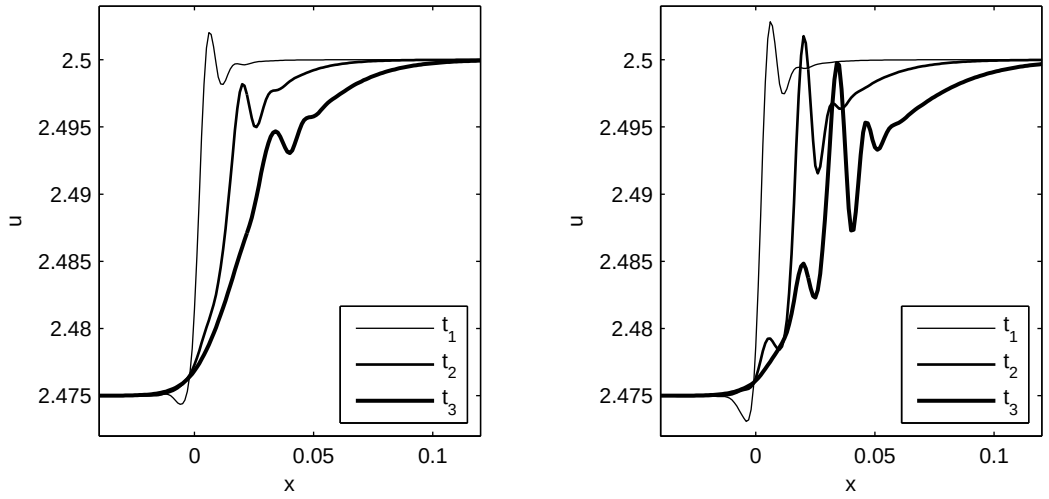
oscillations are amplified. Similarly, examples of stable and unstable integration with the static pressure formulation are presented in Figure 3.6 for the third test problem in Table 3.1. In Figure 3.6a, integration is stable for CFL = 1 and oscillations are damped, while in Figure 3.6b, integration is unstable for the slightly larger CFL number of 1.25 and oscillations are amplified once again. Results differ from that of Figure 3.5 due to the different CFL numbers, while accurate integration is not of importance. These




**(a)** Stable integration obtained with CFL = 3.25.

**(b)** Unstable integration obtained with CFL = 4.

**Figure 3.5:** Velocity distribution in the neighbourhood of $x = 0$ obtained with stable and unstable integration of the third test problem with the total pressure method, $t_1 < t_2 < t_3$.




**(a)** Stable integration obtained with CFL = 1.

**(b)** Unstable integration obtained CFL = 1.25.

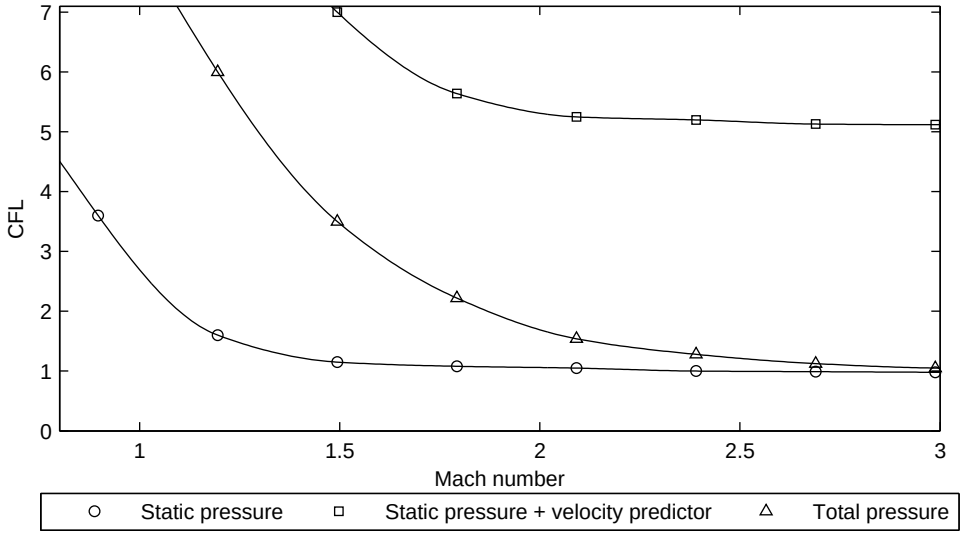
**Figure 3.6:** Velocity distribution in the neighbourhood of $x = 0$ obtained with stable and unstable integration of the third test problem with the static pressure method with explicit convective terms, $t_1 < t_2 < t_3$.

examples illustrate that the proposed new total pressure formulation provides better stability than the static pressure formulation with explicit convection terms.

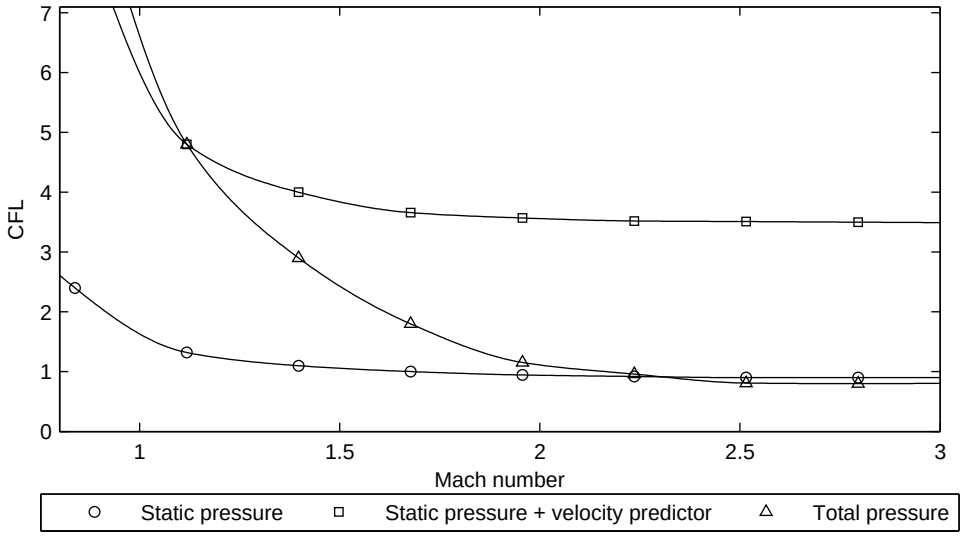
The experiments were conducted for all eight test problems of Table 3.1, and results are summarized in Figure 3.7. The entire procedure was done twice, i.e. for $\gamma = 1.4$ and $\gamma = 1.6$. Figure 3.7a is the case for $\gamma = 1.4$, while 3.7b the case for $\gamma = 1.6$.

All three methods have been found to be highly stable for subsonic flow. The static pressure formulation with explicit convective terms is generally the most restrictive, even for subsonic flows. For increasing Mach numbers, stability windows for all three methods decrease rapidly. For high Mach numbers the static pressure method with explicit convective momentum treatment as well as the total pressure based method become highly restrictive, whereas the method with velocity prediction is superior. The rapid decrease of stability of the total pressure based method can be attributed to the treatment of the $\partial h_0/\partial x$ term in conservation of momentum. For high speed flows this term will become significant and explicit treatment will give rise to instabilities.

From Figure 3.7 it is clear that stability for all three methods decrease as $\gamma$ increases. This is likely attributed to the increased velocity (for fixed Mach number), which can introduce instabilities with the segregated procedure. A non-conservative formulation




**(a)** Stability plot for $\gamma = 1.4$.




**(b)** Stability plot for $\gamma = 1.6$.

**Figure 3.7:** Maximum CFL vs Mach number.

of conservation of energy which eliminates the kinetic energy term may remedy this problem.

For subsonic flows, common in fluid flow networks, the total pressure method appears to be sufficiently unconditionally stable, whereas the static pressure method without velocity predictor may have undesired restrictions in the range $0.7 < \mathrm{M} < 1$. However, the effect of the frictional term has not been included in this investigation. The algebraic frictional term can be naturally discretised implicitly and will aid stability in all methods as velocity increases. It can therefore be concluded that both the

total pressure method and static pressure methods have sufficient stability properties for general purpose subsonic fluid flow network simulation.

## 3.8 Steady state solution

Convergence towards steady state is discussed in this section for the methods based on different momentum formulations. This was done by marching the solution in pseudo time with the different methods. Temporal accuracy is not important, and the only criteria for step size selection was convergence speed towards steady state. Test problems are limited to subsonic flows, as the presence of steady supersonic flows in pipes is highly unlikely.

**Incompressible flow through a variable area duct** The first test problem is the incompressible flow through a 10 m long variable area duct and is illustrated in Figure 3.8. The fluid had a constant density of 1000 kg.m$^{-3}$. The area decreases linearly from 0.03 m$^2$ to 0.01 m$^2$. Friction was neglected, and the duct was divided into 100 increments. The problem was advanced in time, with the initial conditions summarised in Table 3.2.

The energy update and second pressure correction were omitted for each time step calculation, since for incompressible flow conservation of energy decouples from continuity and conservation of momentum. The static pressure method with implicit

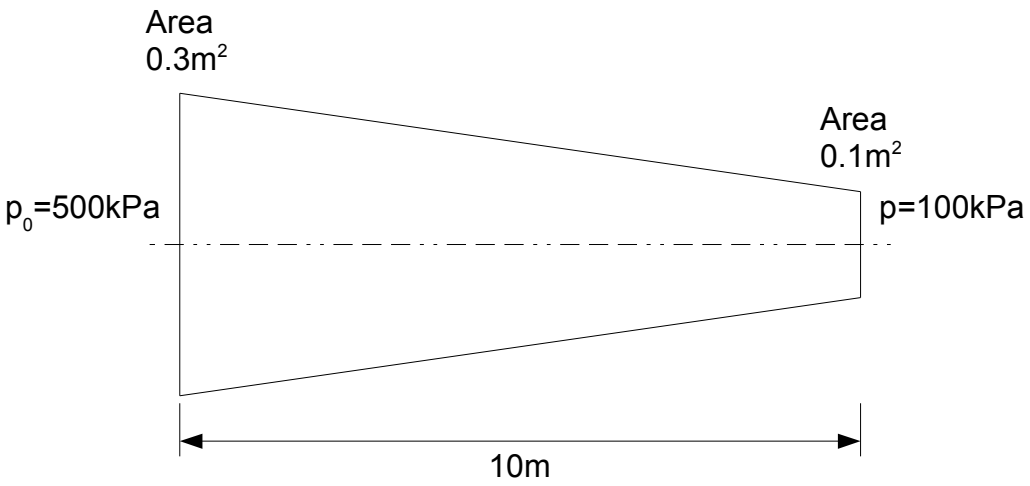



**Figure 3.8:** Incompressible flow through variable area duct.

**Table 3.2:** Initial conditions for Incompressible flow problem.

| | |
|---|---|
| Pressure (kPa) | 100 |
| Velocity ($m.s^{-1}$) | 0 |

momentum prediction was not included, since it was found to have slower convergence, albeit the more stable static pressure method.

Figure 3.9 illustrates the number of pseudo time steps, or iterations, required to reach steady state. The static pressure formulation with explicit convective momentum terms required a minimum of 11 iterations, while the total pressure formulation required 7 iterations. The convergence of the static pressure formulation is clearly slower and more sensitive to the pseudo time step $\Delta t$ than the total pressure formulation. The stability analysis of the previous section did not include the effect of a variable flow area, and the static pressure formulation is expected to suffer from accompanying stability issues. On the other hand, the total pressure formulation eliminates all difficulties resulting from the convective terms of incompressible flow.

The steady state solutions for both formulations are compared with the analytical

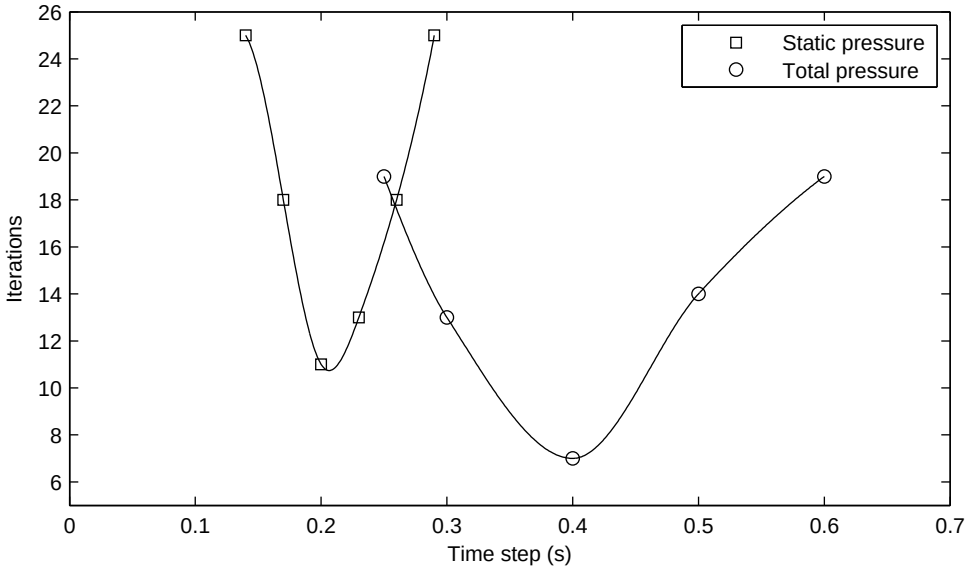
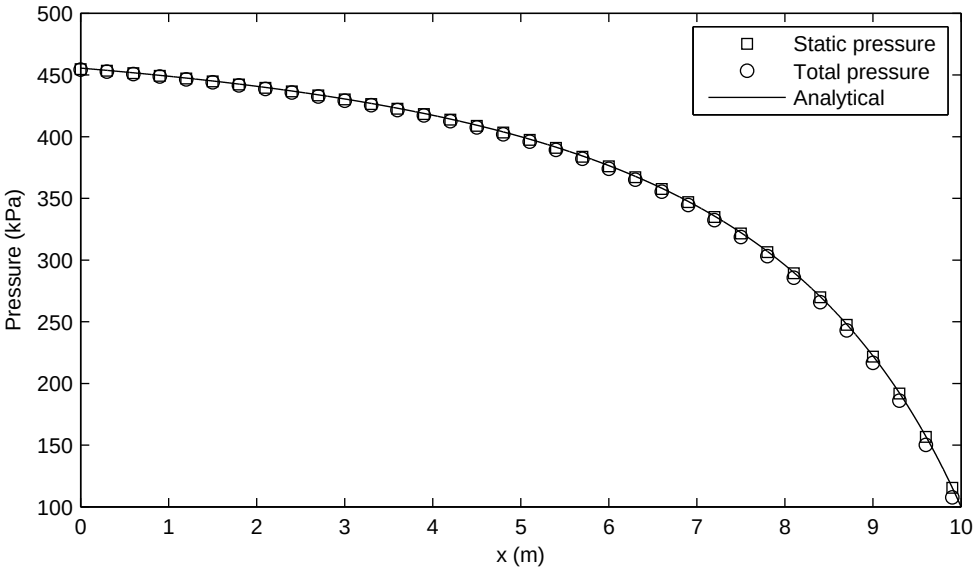
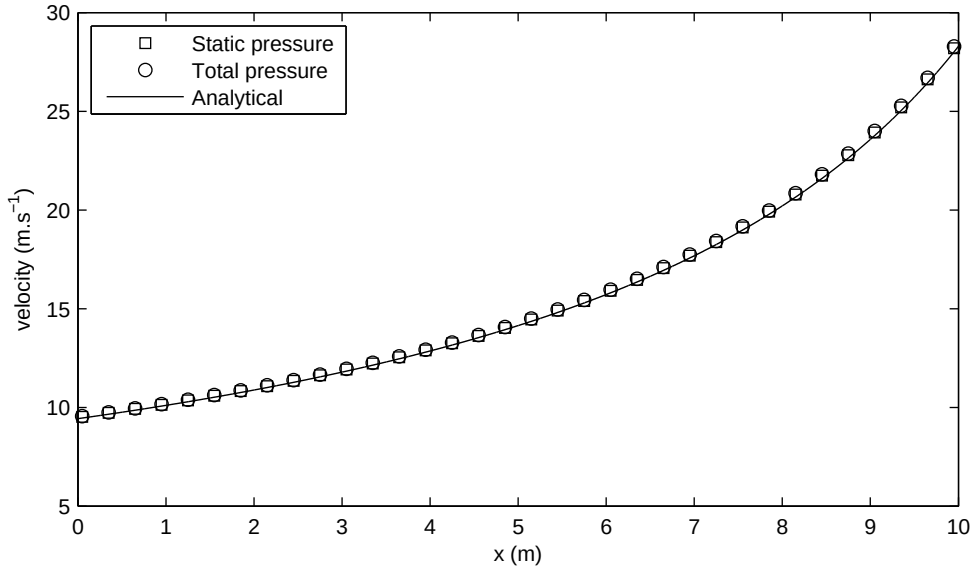



**Figure 3.9:** Number of iterations to reach steady state vs. time step.




**(a)** Pressure vs. length




**(b)** Velocity vs. length

**Figure 3.10:** Pressure and velocity distributions of steady incompressible flow for various methods.

solution in Figure 3.10. The analytical solution of this problem is given by

$$u = \frac{A_e}{A}\sqrt{\frac{2}{\rho}\left(p_{0i} - p_e\right)} \tag{3.58}$$

$$p = p_{0i} - \frac{1}{2}\rho u^2. \tag{3.59}$$

Deviations are attributed to discretisation errors, in particular the treatment at boundaries. In the present case, in order to treat the inlet total pressure with the

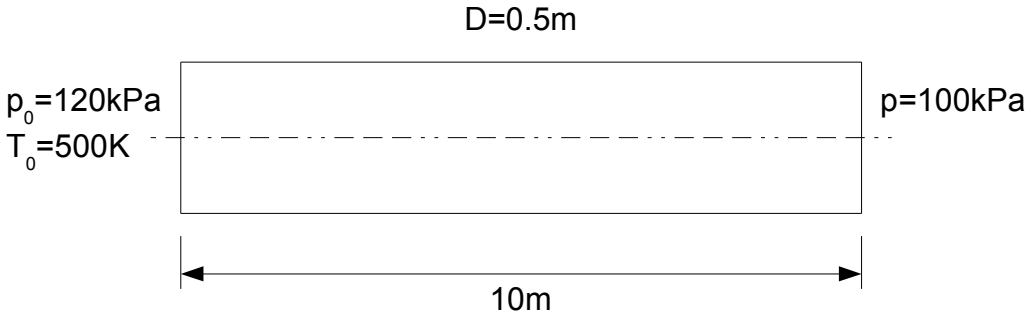



**Figure 3.11:** Fanno flow test problem.

static pressure formulation, zeroth order extrapolation of velocity has been used as a result of the staggered grid. The outlet static pressure is treated naturally with the formulation. The treatment is similar for the total pressure formulation. However, in this case the method treats the inlet total pressure naturally, while extrapolation is used for the outlet static pressure. With outlet velocities varying greatly, the extrapolation thereof at the exit will result in larger inaccuracies as opposed to the inlet. Hence the static pressure formulation will give slightly better results in this particular case, as is evident in Figure 3.10.

**Fanno flow problem** A similar test case was done for the Fanno flow problem, i.e. compressible flow with friction in a constant area duct, as illustrated in Figure 3.11. The fluid was air. A constant friction factor of $f = 0.02$ was chosen and the pipe was divided into 100 increments. Once again the problem was advanced in time, with the initial conditions summarised in Table 3.3. As temporal accuracy and consistency is not important, the second pressure correction was omitted. For the total pressure formulation, the term in conservation of momentum containing $\partial h_0/\partial x$ is exactly zero for steady state adiabatic flow and was omitted in order to potentially increase steady state convergence.

**Table 3.3:** Initial conditions for Fanno flow problem.

| | |
|---|---|
| Pressure (kPa) | 100 |
| Temperature (K) | 400 |
| Velocity (m.s$^{-1}$) | 0 |

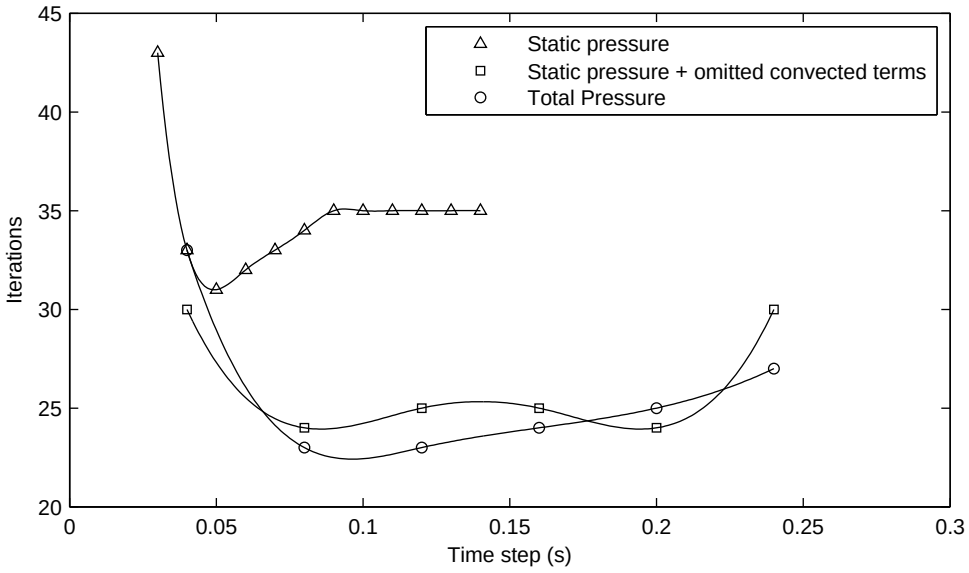



**Figure 3.12:** Number of iterations to reach steady state vs. time step.

For this problem, the non-conservative static pressure formulation with explicit convective momentum terms was again compared with the total pressure formulation. Once again the static pressure method with implicit momentum prediction was not included, since it was found to have slower convergence for this problem. In addition, the performance of a static pressure formulation with omitted convective terms was included, in order to demonstrate the effect of these terms.

Figure 3.12 illustrates the number of pseudo time steps required to reach steady state. The static pressure formulation with explicit convective momentum terms required a minimum of 31 iterations, while the total pressure formulation required 23 iterations. In the present case, even with a moderate Mach number of below 0.5, the convergence of the static pressure formulation is clearly slower and more sensitive to the pseudo time step $\Delta t$ than the other methods. Results of the static pressure formulation with omitted convective terms are similar to that of the total pressure formulation. It can hence be concluded that the total pressure formulation effectively eliminates the difficulties associated with the convective terms from a steady state solution perspective.

The analytical solution can be obtained from the following equations:

$$\frac{p}{p^*} = \frac{1}{M}\left(\frac{(\gamma+1)/2}{1+[(\gamma-1)/2]\,M^2}\right)^{1/2} \tag{3.60}$$

**Table 3.4:** Comparison of mass flux.

| Method | Mass flux |
|---|---|
| Analytical | 139.486 |
| Static pressure formulation | 139.565 |
| Total pressure formulation | 139.555 |

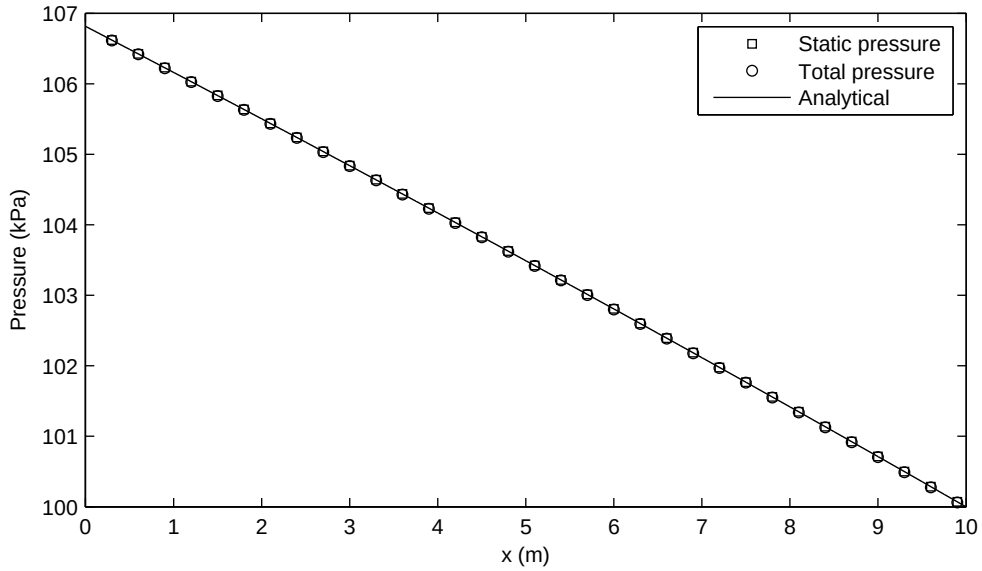
$$\frac{p_0}{p_0^*} = \frac{1}{M}\left(\frac{1+[(\gamma-1)/2]\,M^2}{(\gamma+1)/2}\right)^{(\gamma+1)/2(\gamma-1)} \tag{3.61}$$

$$\frac{f\,(x-x^*)}{D} = \left(\frac{\gamma+1}{2\gamma}\right)\ln\left(\frac{1+[(\gamma-1)/2]\,M^2}{(\gamma+1)/2}\right) - \frac{1}{\gamma}\left(\frac{1}{M^2}-1\right) - \frac{\gamma+1}{2\gamma}\ln M^2 \tag{3.62}$$

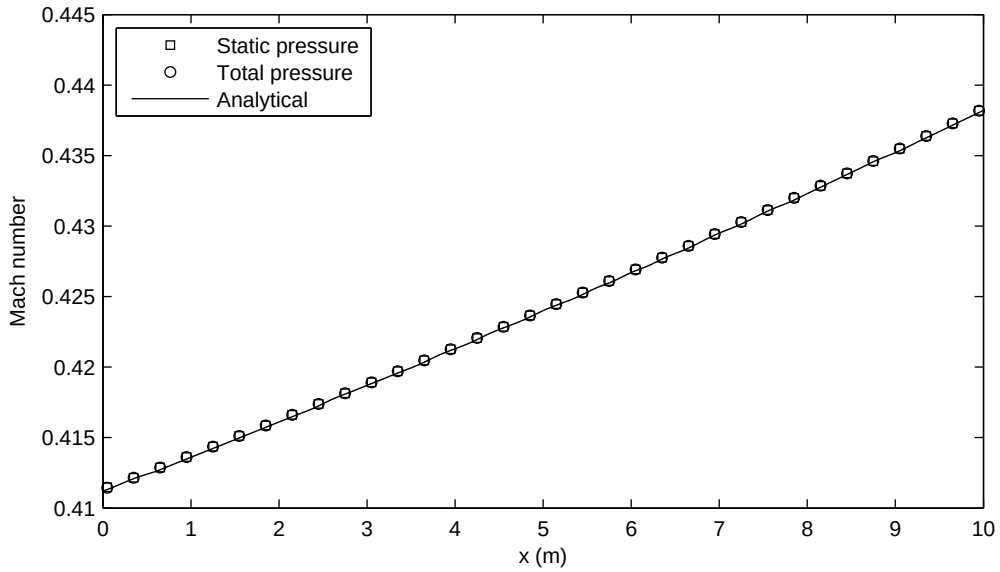
as found in Zucker & Biblarz (2002). Here the asterisk $*$ indicates the hypothetical position where $M = 1$. The steady state solutions for both formulations are compared with the analytical solution in Figure 3.13. The mass flux comparison is summarised in Table 3.4. Deviations are attributed to discretisation errors, particularly the upwind treatment of convected scalars. Once again boundaries have been treated with zeroth order extrapolation of velocity. As opposed to the incompressible problem, both formulations have approximately similar accuracy, since similar errors are introduced with boundary treatment, as evident from the velocity variation in Figure 3.13.

## 3.9 Conclusion

In this chapter a unified 1D total pressure formulation for conservation of momentum was presented which is valid for all fluids. For incompressible fluids and ideal gases, the unified formulation reduces to the respective separate formulations found in Greyvenstein (2002). In addition, a non-iterative solution algorithm was presented for the total pressure method, similar to the ideas of the PISO algorithm. Consistency of the non-iterative algorithm was demonstrated with test problems such as selected Riemann problems and a sudden valve closure problem from the literature. The current non-iterative method was compared with different static pressure methods from a stability point of view using test problems, and was found to be sufficiently stable for subsonic flow conditions, while the static pressure formulation with implicit velocity prediction was found to be the most stable for high speed flows. The instabilities are associated with the sequential, or segregated, solution of the discretised conservation laws, which




**(a)** Pressure vs. length




**(b)** Mach number vs. length

**Figure 3.13:** Pressure and Mach number distributions of steady compressible flow for various methods.

can be overcome by using a fully coupled solution method. Finally, acceleration towards steady state was demonstrated with two test problems which were selected to demonstrate the effects of the treatment of convective terms in conservation of momentum. The first steady state problem is the incompressible flow through a variable area duct. The second problem is a Fanno flow problem at a moderate Mach number. The total pressure method was shown to converge rapidly towards steady state, eliminating

convergence issues associated with the convective terms. It can hence be concluded that methods based on total pressure formulations should be more convenient for application in network simulation codes and that a generic formulation for all fluids can be implemented.

# Chapter 4

# Pipe networks

## 4.1 Introduction

In this chapter, the relevant theory and algorithms are presented as applicable to pipe networks. The first section discusses the compact representation of pipe networks with the aid of node and element connectivity lists. The second section briefly discusses the treatment of pipe junctions in the solution algorithm. In the third section, validation of the methods are done with the aid of solutions obtained in the literature. In the final section of this chapter, a brief discussion is given related to the computational costs associated with the methods under consideration.

## 4.2 Pipe network representation

Pipe networks can be represented using *digraphs* (directed graphs). A graph consists of points, called *vertices*, and lines connecting them, called *edges*. In the engineering literature, these are commonly referred to as *nodes* and *elements* respectively. A digraph is a graph where each edge (element) has a direction from its initial vertex (node) to its terminal vertex (node). Figure 4.1 shows an example digraph.

In pipe networks, pipes are represented by elements, with the origin of its local axis defined at its initial node, with positive direction towards the second node of the element. Pipe junctions and other components can be represented by nodes.

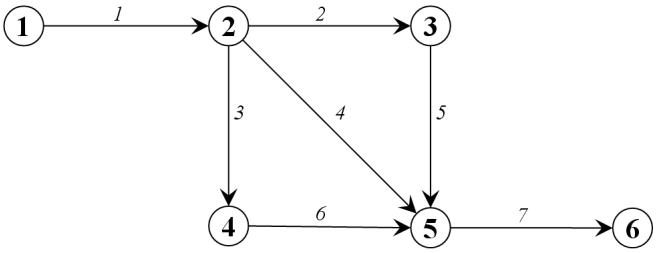



**Figure 4.1:** Digraph consisting of 6 vertices and 7 edges.

### Node and element connectivity lists

A digraph can be uniquely defined by either a node or element connectivity list. For the node connectivity list, the node is listed, followed by all incident elements. If the direction of the element is outwards from the node, a positive value is assigned, while a negative value is assigned otherwise. In the element connectivity list, the element is listed, followed by its initial and terminal nodes, in that order. Alternatively, a connectivity matrix can be used. The list is preferred for its compactness, especially in the case of sparse networks.

The node and element connectivity lists for the graph in Figure 4.1 are given in Table 4.1a and 4.1b respectively. In the node connectivity list, each entry is denoted by $n[i][j]$, for example $n[2][4] = 4$ and $n[5][2] = -5$. A similar convention is used for the element connectivity list, and the entries are denoted by $e[i][j]$.

**Table 4.1:** Network connectivity.

**(a)** Node connectivity list.

| Node | Incident Elements |
|---|---|
| 1 | 1 |
| 2 | $-1, 2, 3, 4$ |
| 3 | $-2, 5$ |
| 4 | $-3, 6$ |
| 5 | $-4, -5, -6, 7$ |
| 6 | $-7$ |

**(b)** Element connectivity list.

| Element | End Nodes |
|---|---|
| 1 | 1, 2 |
| 2 | 2, 3 |
| 3 | 2, 4 |
| 4 | 2, 5 |
| 5 | 3, 5 |
| 6 | 4, 5 |
| 7 | 5, 6 |

## 4.3 Pipe junctions

A pipe junction can be primarily treated as a *branch junction* or *plenum*. A junction of pipes is called a branch junction when it is considered to be quasi-steady, i.e has negligible volume compared to the system as a whole. Alternatively, pipe junctions can be treated as plenums, for which the volume of the junction and associated transient effects are included. For more details on the treatment of junctions, see Corberan (1992) and the references therein.

Consider the pipe junction in Figure 4.2. The conservation law for pipe junction $i$ can be written in integral form for a control volume $\Omega_i$, which can be arbitrary and include or exclude pipe segments. The control volume has neighbouring faces, associated with each connecting pipe, denoted by $j$.

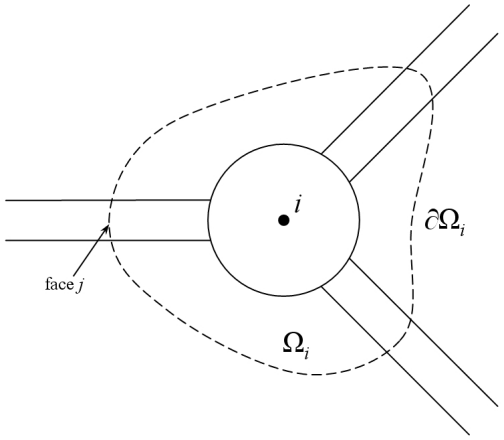



**Figure 4.2:** Control volume for a pipe junction $i$ with face neighbours $j$.

**Continuity** Continuity for control volume $\Omega_i$ can be written as

$$\frac{\partial}{\partial t}\iiint_{\Omega_i} \rho d\mathcal{V} + \iint_{\partial\Omega_i} \rho \mathbf{V} \cdot d\mathbf{A} = 0.$$

The second integral can be rewritten as a sum over all neighbouring faces $j$:

$$\iint_{\partial\Omega_i} \rho \mathbf{V} \cdot d\mathbf{A} = \sum_j (\rho \mathbf{V} \cdot \mathbf{A})_j = \sum_j (\rho u A s)_j,$$

where $s_j$ is the sign of the element $n[i][j]$ in the node connectivity list as described in section 4.2. The final form of continuity is

$$\frac{\partial}{\partial t}\iiint_{\Omega_i} \rho d\mathcal{V} + \sum_j (\rho u A s)_j = 0. \tag{4.1}$$

**Conservation of energy** Conservation of energy for control volume $\Omega_i$ can be written as

$$\frac{\partial}{\partial t}\iiint_{\Omega_i}(\rho h_0 - p)d\mathcal{V} + \iint_{\partial\Omega_i}\rho h_0 \mathbf{V}\cdot d\mathbf{A} = \dot{Q}_i.$$

Once again, the second integral can be rewritten as a sum over all neighbouring faces $j$:

$$\iint_{\partial\Omega_i}\rho h_0 \mathbf{V}\cdot d\mathbf{A} = \sum_j(\rho h_0 \mathbf{V}\cdot\mathbf{A})_j = \sum_j(\rho h_0 uAs)_j.$$

The final form of conservation of energy is

$$\frac{\partial}{\partial t}\iiint_{\Omega_i}(\rho h_0 - p)d\mathcal{V} + \sum_j(\rho h_0 uAs)_j = \dot{Q}_i. \tag{4.2}$$

**Conservation of momentum**

For pipe junctions, empirical correlations can be used or simplified flow assumptions can be made in an attempt to account for momentum conservation. Possibly the simplest approach is based on uniform pressure, which will be used in this study. This approach is mathematically sound, and assumes a uniform pressure for all incoming and outgoing waves.

**Uniform pressure theory** The uniform pressure theory states that at any instant the pressure at the end of each branch of a pipe junction is equal. Furthermore, if the junction is modelled by a plenum, the pressure of the junction volume is assumed uniformly distributed over space. For the pipe junction in Figure 4.3 this can be written as

$$p_j = p_i \qquad j = 1, 2, ... \tag{4.3}$$

A uniform total pressure model can be formulated in exactly the same manner.

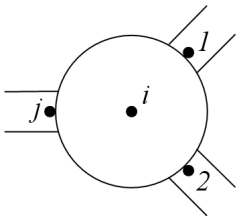



**Figure 4.3:** Pipe junction $i$ with branch ends $j$.

### 4.3.1 Pipe junction discretisation

A staggered grid is also adopted for pipe junctions. Consider the pipe junction in Figure 4.4. Each junction has neighbouring faces $j$ associated with each connected pipe. Associated with each face $j$ is a neighbouring node $n_j$. The control volume $\Omega_i$ for each junction is chosen to include only the volume of the junction (which may be zero). Although this is perfectly valid, an alternative approach will be used in this study, which will be described next.

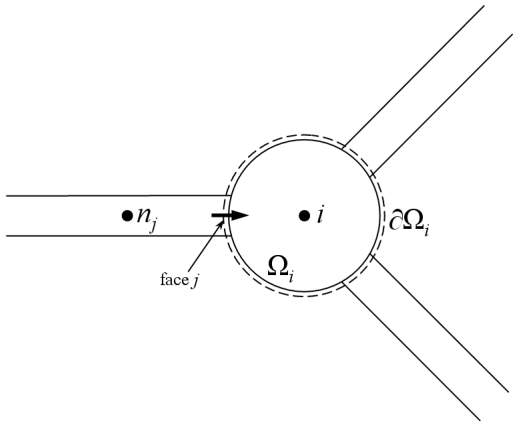



**Figure 4.4:** Scalar control volume for a pipe junction.

As opposed to the discretisation illustrated in Figure 4.4 consider the discretisation in Figure 4.5. The control volume $\Omega_i$ associated with each junction is chosen to include the volume of the junction (if the junction is modelled by a branch, this will be zero), as well as the volume of half an increment of each connecting pipe, as illustrated in Figure 4.5. The main advantage of this approach is the efficient implementation of uniform static or total pressure junctions and junctions with pressure loss coefficients. Although this is not the ideal approach from a component-wise model, such as the discretisation in Figure 4.4, this approach is adequate for the purpose of this study. Special care should be taken in the case of a total pressure model, as variables such as density and static pressure are discontinuous at junctions. The semi-discretised continuity and conservation of energy equations for a control volume can now be presented for (4.1) and (4.2).

**Continuity**

$$\mathcal{V}_i \frac{d\rho_i}{dt} + \sum_j (\rho u A s)_j = 0. \tag{4.4}$$

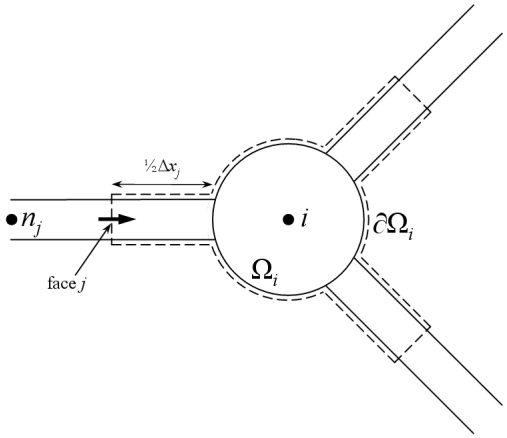



**Figure 4.5:** Alternative scalar control volume for a pipe branch node.

**Conservation of energy**

$$\mathcal{V}_i \frac{d(\rho_i h_{0i} - p_i)}{dt} + \sum_j (\rho h_0 u A s)_j = Q. \tag{4.5}$$

Here $j$ is the neighbouring face corresponding to the connected pipe. The scalar quantities at the faces need to be approximated, and are done in a similar way as with pipes. A scalar quantity $\phi$ at face $j$ can be approximated in various ways. For upwind interpolation, this can be written as

$$\phi_j = \left\{ \begin{array}{ll} \phi_i & \text{if } u_j s_j \geq 0 \\ \phi_{n_j} & \text{if } u_j s_j < 0 \end{array} \right\}, \tag{4.6}$$

while central interpolation gives

$$\phi_j = \frac{\phi_i + \phi_{n_j}}{2}. \tag{4.7}$$

### 4.3.2 Solution algorithm

With the discretisation of junctions done in the manner illustrated in Figure 4.5, the methods for pipes are readily extended to include junctions. More specifically, static pressure methods can be extended for uniform static pressure junctions and total pressure methods for uniform total pressure junctions. The solution procedure is briefly discussed next. The argument is valid for any suitable temporal discretisation such as the $\alpha$-scheme.

For control volume $i$ in Figure 4.5 of a uniform pressure junction, the face velocity $u_j$ can be written in terms of the cell $i$ and neighbouring cell $n_j$ from conservation of

momentum (3.4). This takes the form:

$$u_j = s_j c_j^1 \left(p_i - p_{n_j}\right) + c_j^0, \tag{4.8}$$

where the constants $c_j^1$ and $c_j^0$ are introduced for convenience. From this step, the manipulation of continuity and conservation of momentum proceeds in a similar fashion as that of single pipes to construct the pressure correction equation. A pressure (or pressure correction) equation of the form

$$a_{ii} p_i + \sum_j a_{ij} p_{n_j} = b_i \tag{4.9}$$

is obtained for the junction in Figure 4.5. The approach for a total pressure method proceeds in exactly the same manner if a uniform total pressure junction model is used. This equation is likely diagonal dominant, depending on the discretisation of the convected scalars. A tridiagonal structure is lost at junctions, however.

It should be stated that the discretised equations for continuity and conservation of energy for pipes can be expressed in a similar manner as the discretised equations for pipe branches. The corresponding values for the scalar control volume $i$ of Figure 3.1 are summarized in Table 4.2. For more complex junction treatment, however, a separate formulation of pipes and junctions is likely more appropriate.

**Table 4.2:** Unified numbering applied to a pipe segment of Figure 3.1 for node $i$.

| | $j = 1$ | $j = 2$ |
|---|---|---|
| $s_j$ | $-1$ | $1$ |
| face$_j$ | $i - 1/2$ | $i + 1/2$ |
| $n_j$ | $i - 1$ | $i + 1$ |

## 4.4 Validation

### 4.4.1 Total pressure method

For the validation of the treatment of a simple network consisting of a pipe junction, the non-iterative total pressure method presented in the previous chapter is compared

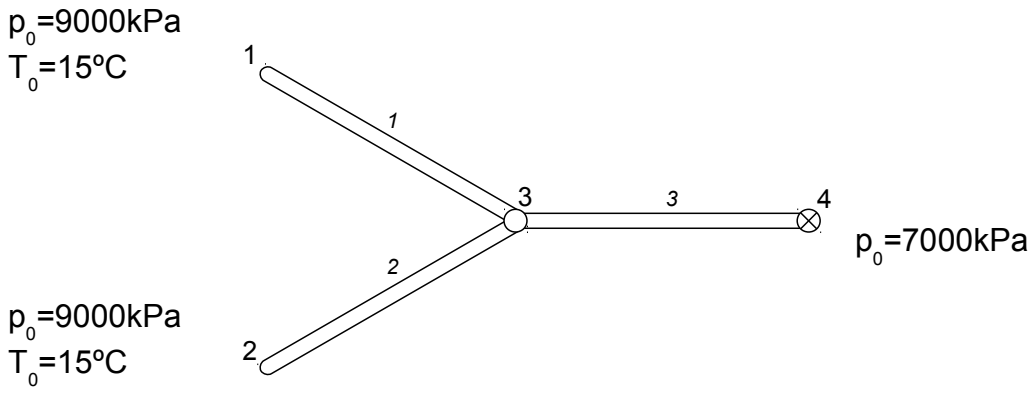
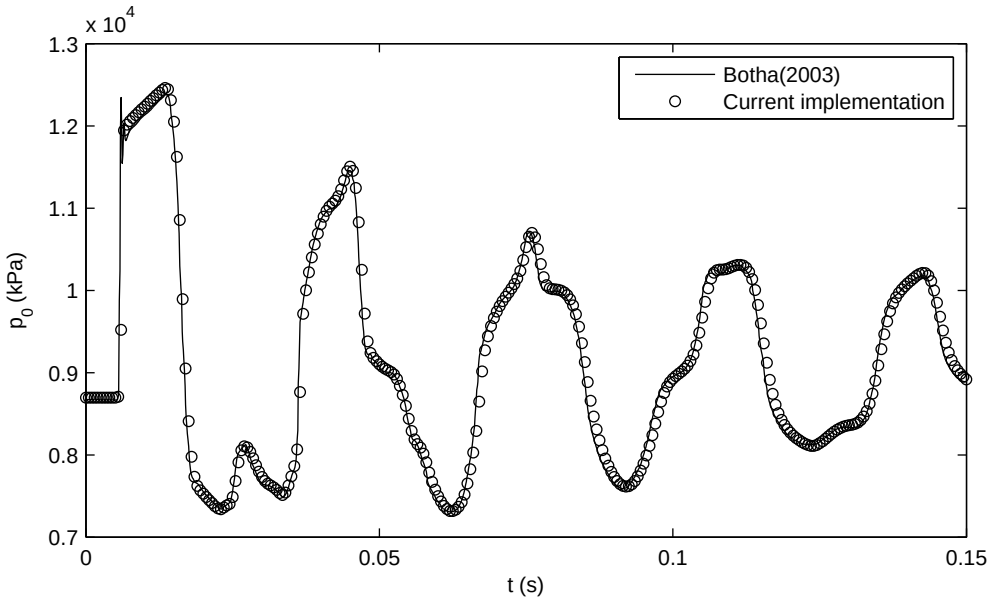



**Figure 4.6:** A simple branching network test problem for the total pressure method (Botha 2003).

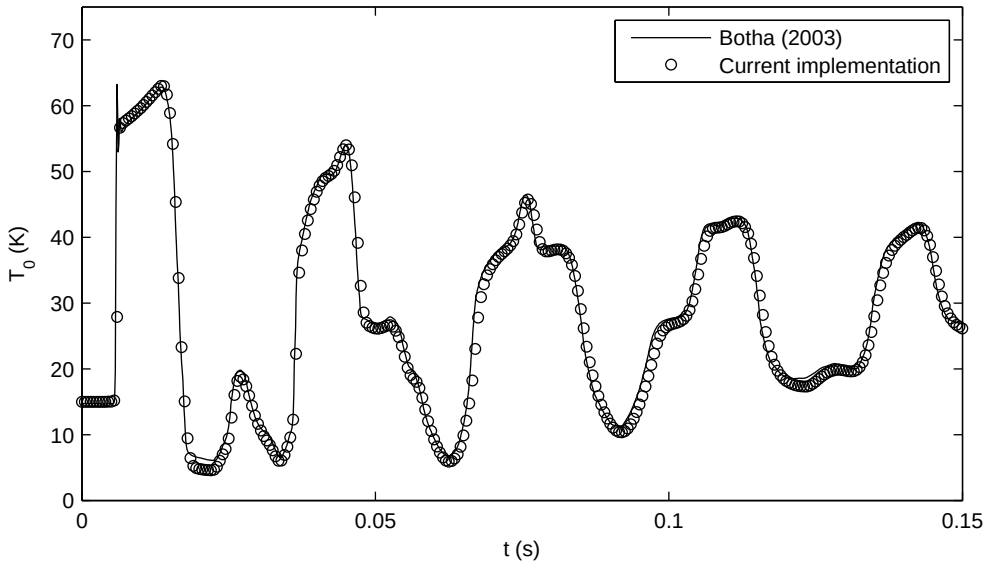
with results from the literature. The test problem and results that were used for this purpose are that of the simple branching network found in Botha (2003). In Botha (2003), the solution as obtained with a validated implicit method was given and is used in this section as a benchmark solution. Another potential benchmark problem and solution, based on static pressure losses, were given in Chae et al. (2006).

The test problem as found in Botha (2003) is given in Figure 4.6 together with the boundary conditions. It is a simple branching network which consists of three pipes and a junction node. After steady state was achieved with the given boundary conditions, the valve at node 4 was closed instantaneously. The fluid is helium. All three pipe characteristics are identical, with a length of 5 m and a diameter of 0.065 m. All three pipes were divided into 40 equal increments. In Botha (2003), friction factors were calculated as a function of the Reynolds number and relative pipe roughness. In this paper, a constant friction factor of 0.0165 was used throughout, which corresponds approximately to the friction factor at steady state conditions given in Botha (2003). A time step size of $t = 0.00005$ s and a value of $\alpha = 0.7$ were used for the remainder of this chapter, unless stated otherwise.

The pressure and temperature at the junction (node 3) as a function of time after the instant of valve closure are given in Figure 4.7. The solution as found in Botha (2003) as well as that of the current non-iterative total pressure method are shown. Clearly all complex wave phenomena at the junction node as well as damping due to frictional forces are similar and no significant deviations are observed.




**(a)** Pressure distribution at the junction node.




**(b)** Temperature distribution at the junction node.

**Figure 4.7:** Validation of the non-iterative total pressure method for a simple branching network.

### 4.4.2 Static pressure methods

The static pressure methods for a uniform static pressure junction give a notable different solution as opposed to a total pressure method at moderate velocities. For this reason, in order to validate the static pressure methods used in this thesis, the test problem in Figure 4.6 was slightly modified and is illustrated in Figure 4.8. In

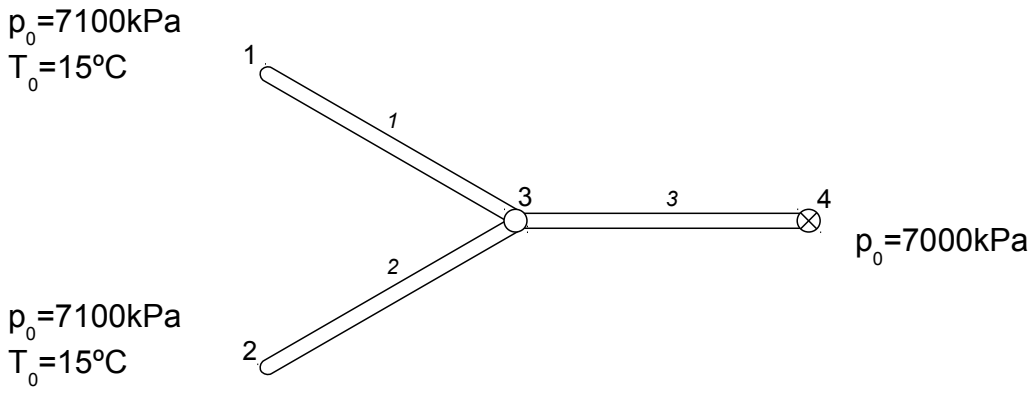



**Figure 4.8:** The modified branching network test problem for static pressure methods.

particular, the boundary pressures are significantly lowered to result in a lower speed problem such that $T_0 \approx T$ and $p_0 \approx p$. Results of three static pressure methods are then compared with the validated non-iterative total pressure method.

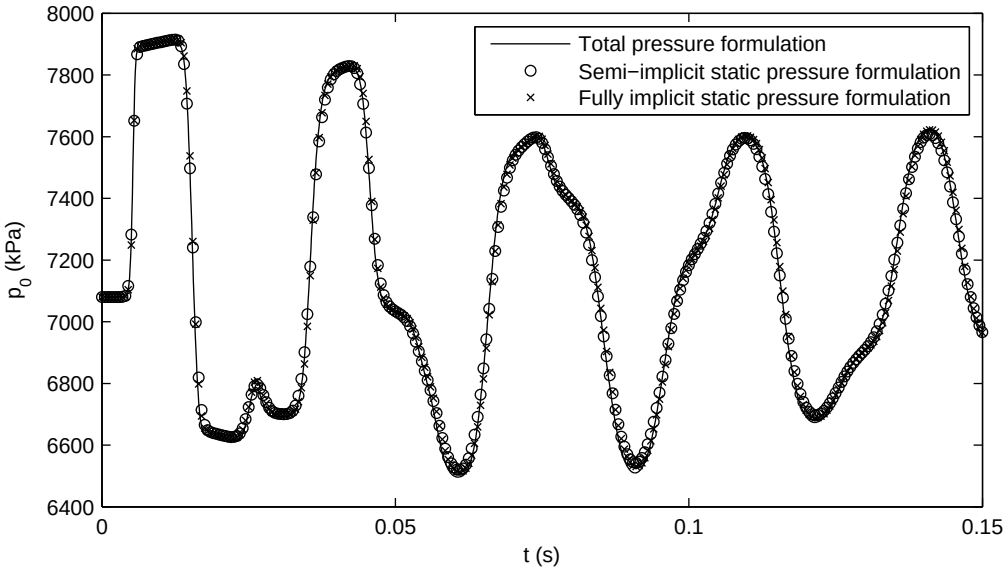
The first static pressure based method is the non-iterative method based on the $\alpha$-scheme as found in Chapter 3. The second method of interest in this paper is a semi-implicit method, of which the derivation can be found in Appendix A. The idea of the method is to treat only acoustic terms implicitly, while kinematic terms are treated explicitly. This leads to a more efficient solution algorithm which comprises of only a pressure equation. However, a stability criterion exists based on the material Courant number:

$$\max |u| \frac{\Delta t}{\Delta x} \leq 1. \tag{4.10}$$

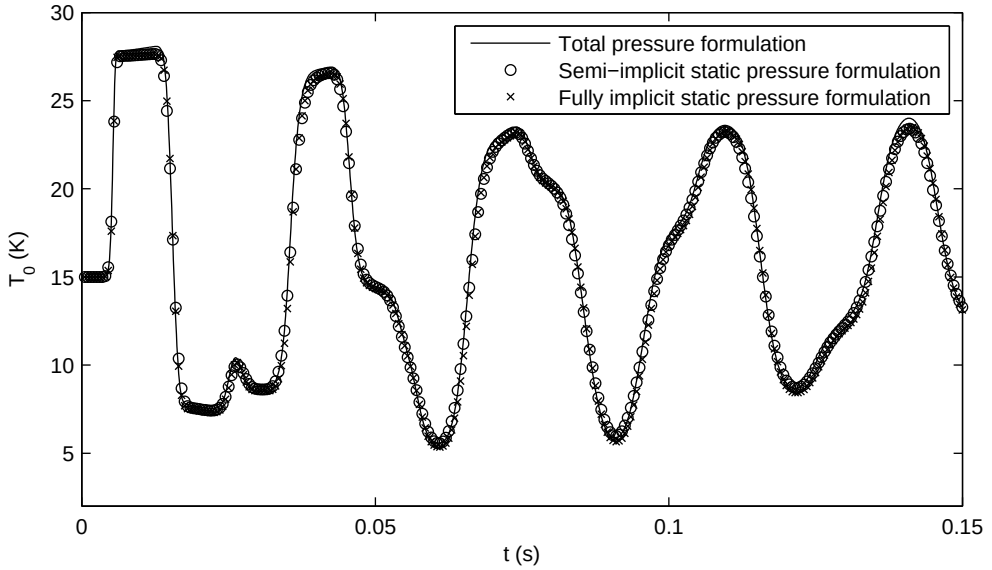
A third method is the fully implicit back substitution method (FIBS) of Garland (1998). Although not utilised in this thesis, the results are nonetheless given. The method is based on the $\alpha$-scheme, while the calculation of a matrix inverse and a number of matrix multiplications are done in an attempt to reduce the system of equations to a system consisting of only mass fluxes. The entire method can also be found in Appendix A.

The pressure and temperature at the junction (node 3) as obtained with the semi-implicit and implicit static pressure methods after the instant of valve closure are given in Figure 4.9. The solutions compare well with the already validated total pressure method.

The pressure and temperature at the junction (node 3) as obtained with FIBS



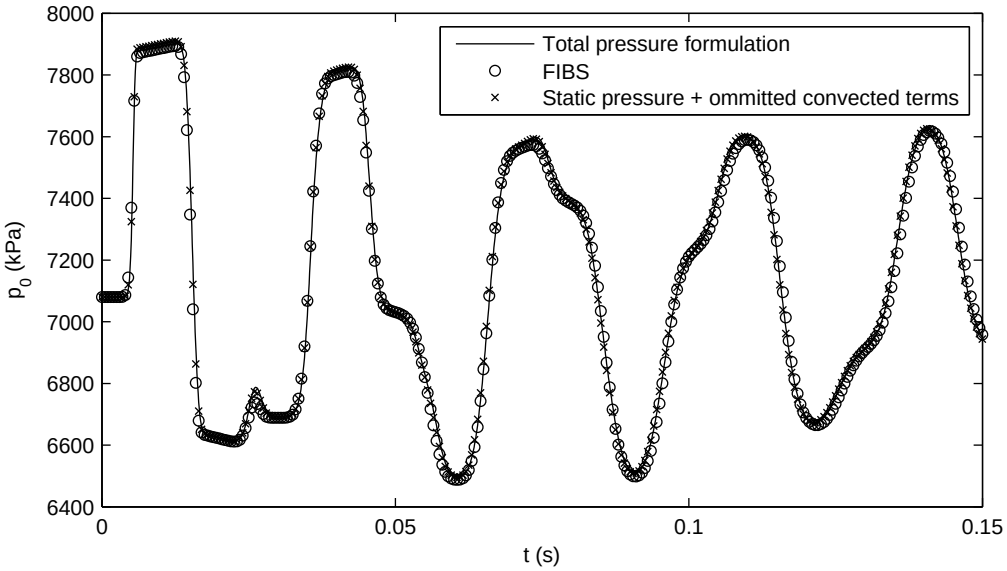

**(a)** Pressure distribution at the junction node.



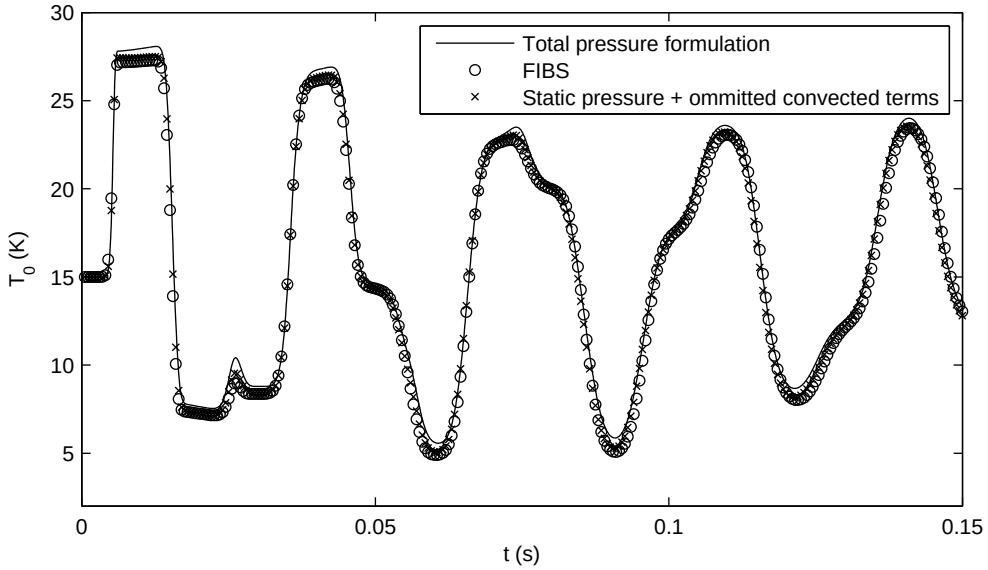

**(b)** Temperature distribution at the junction node.

**Figure 4.9:** Validation of the semi-implicit and non-iterative implicit static pressure method for a simple branching network.

after the instant of valve closure is given in Figure 4.10. Slight deviations of the temperature distribution is observed. This can likely be attributed to the omitted convective momentum terms of the method. A static pressure method with omitted convective momentum terms is also illustrated in Figure 4.10 to demonstrate this.




**(a)** Pressure distribution at the junction node.




**(b)** Temperature distribution at the junction node.

**Figure 4.10:** Validation of FIBS for a simple branching network.

## 4.5 Computational cost

Although a detail investigation of computational cost and performance is beyond the scope of this thesis, costs associated with these four methods are briefly discussed in this section. The system of linear equations was solved with a generic direct method. In the case of Garland's FIBS, generic matrix routines were used for matrix multiplication and inversion. All calculations were performed on a 2.67 GHz Intel® Core™2 Duo

**Table 4.3:** Computational time (in seconds) for various methods. TP = total pressure method; $SP_1$ = implicit static pressure method; $SP_2$ = semi-implicit static pressure method.

| $1/\Delta x$ | TP | $SP_1$ | $SP_2$ | FIBS |
|---|---|---|---|---|
| 5 | 1.134 | 1.047 | 0.409 | 3.735 |
| 10 | 6.453 | 6.334 | 2.188 | 28.156 |
| 15 | 19.797 | 19.532 | 6.656 | 116.640 |
| 20 | 45.859 | 45.531 | 15.360 | 307.000 |
| 25 | 88.437 | 87.860 | 29.453 | 550.000 |
| 30 | 149.609 | 149.047 | 49.953 | 928.200 |

processor. Of course, sparse or iterative solvers are more efficient for large systems and should be included in a thorough study and comparison of the methods under consideration.

For the test problem of the previous section, the computational time (in seconds) for various methods is summarised in Table 4.3. For all methods, the transient was solved up to $t = 0.15$ s and $\Delta t = 0.0001$ s. This corresponds to 1500 time steps. The results are given for increasingly finer meshes which scales with $1/\Delta x$. The implicit total and static pressure methods (TP and $SP_1$ in Table 4.3) has similar computational times, with the total pressure method slightly more expensive. This is due to the additional costs associated with the calculation of both static and total pressures and temperatures. However, for large systems where the cost is expected to be dominated by the solution of the system of linear equations, these differences will be less apparent. The semi-implicit method ($SP_2$) is the most efficient, while FIBS is by far the most expensive. The reason for the expensive nature of FIBS is attributed to the large number of matrix multiplications as well as the calculation and storage of a matrix inverse (see Appendix A for the algorithm). Sparse matrix techniques will result in significant savings, and is essential in a comparison study to do any of these methods justice.

The computational cost can be expressed relative to that of the semi-implicit method, and is summarised in Table 4.4. This is presented graphically in Figure 4.11.

**Table 4.4:** Relative computational time (in seconds) for various methods. TP = total pressure method; SP = implicit static pressure method. Computational time are expressed relative to the semi-implicit method.

| $1/\Delta x$ | TP | SP | FIBS |
|---|---|---|---|
| 5 | 2.773 | 2.560 | 9.132 |
| 10 | 2.949 | 2.895 | 12.868 |
| 15 | 2.974 | 2.934 | 17.524 |
| 20 | 2.986 | 2.964 | 19.987 |
| 25 | 3.003 | 2.983 | 18.674 |
| 30 | 2.995 | 2.984 | 18.581 |

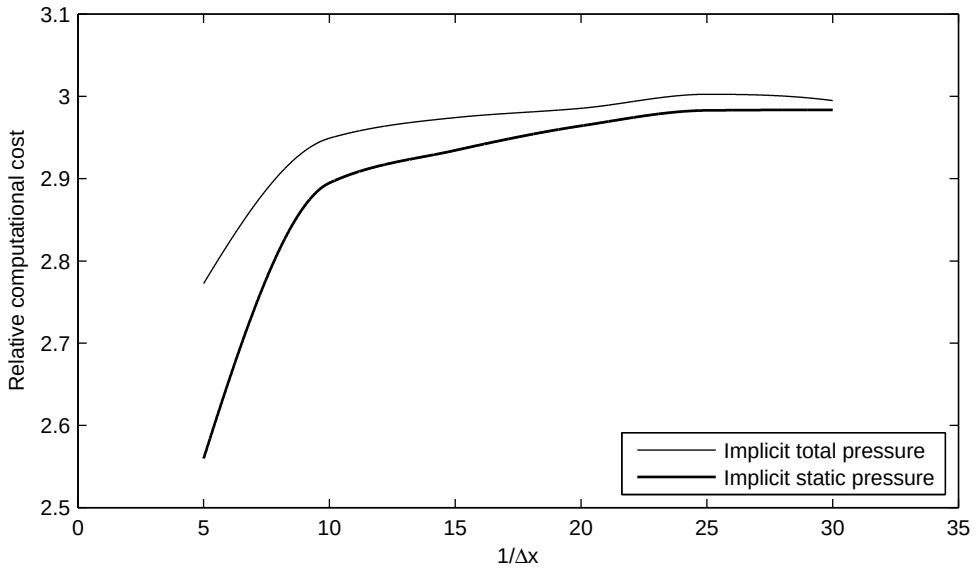



**Figure 4.11:** Relative computational time for various methods as the grid is refined.

For a fine grid, the relative computational cost approaches three for both the implicit methods. This is expected since the minimalistic implicit methods of Chapter 3 solve for pressure twice and energy once, while the semi-implicit method solves for pressure only once and all other values are obtained from post-computations (refer to Appendix A). Although this result was obtained using a generic direct linear solver, similar relative costs are expected whenever the computational cost is dominated by the solution of the linear equations.

## 4.6 Conclusion

This chapter discussed some network specific aspects. In particular, network representation was represented uniquely by node or element connectivity lists. The conservation equations for simple pipe junctions were given, and solution procedures were briefly discussed. The treatment of other components were not discussed as it is beyond the scope of this thesis.

Four methods were validated for the treatment of a simple network consisting of a pipe junction. These were the total pressure method, fully implicit and semi-implicit static pressure methods as well as the fully implicit back substitution method (FIBS) of Garland (1998). For simplicity, a uniform total pressure junction model was used for the total pressure method, while a uniform static pressure junction model was used for all static pressure based methods. All phenomena associated with wave interaction at the junction and frictional effects were correctly handled by all methods. In addition to validation, a rigorous summary was given with respect to the computational costs associated with all methods where direct matrix methods were used. As expected, the semi-implicit method was the most efficient, while FIBS had the highest computational time.

# Chapter 5

# Adaptive time step algorithm

## 5.1 Introduction

From the literature it is clear that adaptive time stepping techniques have been successfully applied to selected flow problems. This chapter addresses the simultaneous treatment of very fast wave phenomena as well as slow storage effects, on which the literature seems to be scarce. Furthermore, as opposed to the conventional Runge-Kutta and multi step methods, other widely used methods for flow problems such as semi-implicit methods and predictor corrector methods have not been widely used for adaptive time stepping. This chapter addresses this issue by providing an error estimation valid for any first order method as applied to pipe networks. Efficiency of the adaptive time step algorithm is demonstrated for pipe networks associated with different time scales.

## 5.2 Adaptive time step methods

**Semi-discrete equations**

After a suitable spatial discretisation of the governing equations such as those given in Chapters 3 & 4, the resulting semi-discrete equations can be written as the ordinary differential equation

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(\mathbf{y}). \tag{5.1}$$

Here $\mathbf{y}$ consists of the solution vectors $\mathbf{y} = (\boldsymbol{\rho},\ \mathbf{u},\ \boldsymbol{\rho E}\ )^T$ which are defined by

$$\boldsymbol{\rho} := \boldsymbol{\rho}(t) = (\rho_1(t),\ \rho_2(t), \ldots,\ \rho_{N_s})^T$$
$$\boldsymbol{u} := \boldsymbol{u}(t) = (u_1(t),\ u_2(t), \ldots,\ u_{N_u})^T$$
$$\boldsymbol{\rho E} := \boldsymbol{\rho E}(t) = (\rho E_1(t),\ \rho E_2(t), \ldots,\ \rho E_{N_s})^T$$

and associated with continuity, conservation of momentum and conservation of energy respectively. $N_u$ and $N_s$ are the number of discrete velocities and scalars associated with the spatial discretisation.

### Error estimation

In this chapter, adaptive time stepping will be accomplished in a similar fashion as with embedded techniques, i.e by comparing methods of different order. If a single adaptive method is used for general pipe networks, an implicit method with good stability is preferred. Fast transients for which explicit methods may be preferred are usually of a short duration, as opposed to slow transients which span over long time intervals. Hence implicit methods are generally more efficient if both phenomena are present.

The $\alpha$-scheme is once again used to integrate the system of equations. The solution $\mathbf{y}^{n+1}$ is obtained by solving these equations approximately by means of a the PISO-based non-iterative method of Chapter 3. $\alpha = 0.7$ is used in this study, and the method is hence first order. From a Taylor series expansion for the true solution $\mathbf{y}\,(t_n + \Delta t)$, the local truncation error of the $\alpha$-scheme can be shown to be

$$\boldsymbol{\tau} = \Delta t^2 \left(\frac{1}{2} - \alpha\right) \ddot{\mathbf{y}}^n + O\left(\Delta t^3\right), \tag{5.2}$$

which can be approximated by interpolation using values at previous time steps. The error of the solution obtained with the PISO-based algorithm will generally not be the same as the exact $\alpha$-scheme, which can be classified as a multi-step method, unless additional iterations are performed. This means that conventional techniques for error estimates of multi-step methods cannot be used, and an alternative approach is therefore employed. Similar arguments hold for conventional predictor-corrector and other semi-implicit methods which are neither Runge-Kutta nor multi-step methods.

Once the first order solution is obtained, a second order explicit solution $\tilde{\mathbf{y}}^{n+1}$ can be obtained with trapezoidal integration:

$$\tilde{\mathbf{y}}^{n+1} = \mathbf{y}^n + \frac{\Delta t}{2}\left[\mathbf{f}\left(\mathbf{y}^n\right) + \mathbf{f}\left(\mathbf{y}^{n+1}\right)\right], \tag{5.3}$$

where the first order solution $\mathbf{y}^{n+1}$ is used. The first order implicit step serves as the primary method, whilst the second order method is used only for error estimation of the first order scheme. Advancing the solution with the second order method, i.e. local extrapolation, is not used as it is not sufficiently stable.

The difference between the two methods gives an approximation of the truncation error of the first order method:

$$\boldsymbol{\tau}^{n+1} = \mathbf{y}^{n+1} - \tilde{\mathbf{y}}^{n+1}, \tag{5.4}$$

or alternatively for this case

$$\boldsymbol{\tau}^{n+1} = \left(\boldsymbol{\tau}_{\rho}^{n+1}, \boldsymbol{\tau}_{u}^{n+1}, \boldsymbol{\tau}_{\rho E}^{n+1}\right)^{T}, \tag{5.5}$$

where

$$\boldsymbol{\tau}_{\rho}^{n+1} = \boldsymbol{\rho}^{n+1} - \tilde{\boldsymbol{\rho}}^{n+1} \tag{5.6}$$

$$\boldsymbol{\tau}_{u}^{n+1} = \mathbf{u}^{n+1} - \tilde{\mathbf{u}}^{n+1} \tag{5.7}$$

$$\boldsymbol{\tau}_{\rho E}^{n+1} = (\boldsymbol{\rho E})^{n+1} - \left(\tilde{\boldsymbol{\rho E}}\right)^{n+1}. \tag{5.8}$$

This procedure remains a single step method and no additional storage is required for the solution vector at previous time steps in order to obtain the estimate.

If the $\alpha$-scheme is solved exactly, i.e. the resulting non-linear equations are solved up to tolerance with a suitable method such as the SIMPLE algorithm, (5.3) and (5.4) reduce to an appropriate approximation of the truncation error if $\alpha \neq 0.5$, i.e.

$$\begin{aligned}\mathbf{y}^{n+1} - \tilde{\mathbf{y}}^{n+1} &= \Delta t \left(\frac{1}{2} - \alpha\right) \left[\mathbf{f}(\mathbf{y}^{n+1}) - \mathbf{f}(\mathbf{y}^{n})\right] \\ &\approx \Delta t^2 \left(\frac{1}{2} - \alpha\right) \ddot{\mathbf{y}}^{n}.\end{aligned}$$

For $\alpha$ close to 0.5, this procedure fails since the truncation error will be dominated by higher order terms. No such problems were encountered for $\alpha = 0.7$. If $\alpha$ close to 0.5 is desired, a third order method can be used for error estimation.

The error norm is defined as

$$\epsilon^{n+1} = \max\left(\epsilon_{\rho}^{n+1}, \epsilon_{u}^{n+1}, \epsilon_{\rho E}^{n+1}\right) \tag{5.9}$$

where

$$\epsilon_{\rho}^{n+1} = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{\left|\rho_i^{n+1} - \tilde{\rho}_i^{n+1}\right|}{Atol_{\rho} + Rtol_{\rho}\left|\rho_i^{n+1}\right|} \tag{5.10}$$

$$\epsilon_{u}^{n+1} = \frac{1}{N_u}\sum_{i=1}^{N_u}\frac{\left|u_i^{n+1} - \tilde{u}_i^{n+1}\right|}{Atol_{u} + Rtol_{u}\left|u_i^{n+1}\right|} \tag{5.11}$$

$$\epsilon_{\rho E}^{n+1} = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{\left|(\rho E)_i^{n+1} - \left(\tilde{\rho E}\right)_i^{n+1}\right|}{Atol_{\rho E} + Rtol_{\rho E}\left|(\rho E)_i^{n+1}\right|}. \tag{5.12}$$

Here $Atol_{\rho}$, $Atol_{\rho}$ and $Atol_{\rho E}$ are user specified absolute tolerances corresponding to $\rho$, $u$ and $\rho E$ respectively. Similarly, $Rtol_{\rho}$, $Rtol_{\rho}$ and $Rtol_{\rho E}$ are user specified relative tolerances corresponding to $\rho$, $u$ and $\rho E$ respectively. Other norms such as RMS can be used as well. Weighted norms with respect to volume can also be of use, for example.

**Time step size adjustment** The time step size is to be obtained such that $\epsilon \leq 1$. The difference between the solution of two methods of order $p$ and $p+1$ gives an approximation of the error of the $p$-th order method, i.e. $\epsilon = C\Delta t^{p+1} + O\left(\Delta t^{p+2}\right)$ or

$$\epsilon \approx C\Delta t^{p+1}, \tag{5.13}$$

where $C$ is some constant. From this, a new time step $\Delta t_{new}$ can be calculated such that $\epsilon \leq 1$ is satisfied:

$$\epsilon_{new} \approx \epsilon\left(\Delta t_{new}/\Delta t\right)^{p+1} \leq 1 \tag{5.14}$$

or

$$\Delta t_{new} \leq \Delta t \cdot \left(1/\epsilon\right)^{1/(p+1)}. \tag{5.15}$$

(5.15) is multiplied with a safety factor $fac$, in the range of 0.8, in order to increase the probability of an acceptable error. In addition, the step size is not allowed to increase too fast, so that we can write

$$\Delta t_{new} = \Delta t \cdot \min\left[facmax, fac \cdot \left(1/\epsilon\right)^{1/(p+1)}\right], \tag{5.16}$$

where $facmax$ is usually chosen in the range of 1.2. This type of control can be viewed as analogous to an integral (I) type controller. Other controls can be used as well, such as PI and PID (Hairer & Wanner 1991, Valli et al. 2005).

The adaptive time step algorithm for the first order method, i.e. $p = 1$, used in this paper is summarised in Figure 5.1. The first step is to obtain the input data

1. Input data: $\mathbf{y}^n$, $t^n$, $\Delta t$ and $n_{rej} = 0$.
2. Compute the first order solution $\mathbf{y}^{n+1}$ with any appropriate method such as PISO or SIMPLE algorithms.
3. Compute the second order solution $\tilde{\mathbf{y}}^{n+1}$ from (5.3).
4. Compute the error estimate $\epsilon^{n+1}$ from (5.9).
5. Compute the new time step:
   (a) $\Delta t = \Delta t \cdot \mathtt{min}\left[facmax, fac \cdot (1/\epsilon)^{1/2}\right]$
   (b) $\Delta t = \mathtt{max}\left(\Delta t, \Delta t_{min}\right)$
   (c) $\Delta t = \mathtt{min}\left(\Delta t, \Delta t_{max}\right)$
6. If $((\epsilon > 1.0)$ and $(\Delta t > \Delta t_{min}))$ then
   (a) $n_{rej} = n_{rej} + 1$
   (b) Return to step 2

   Else
   (a) $\mathbf{y}^n = \mathbf{y}^{n+1}$, $t^n = t^{n+1}$
   (b) Return to step 1

   End if

**Figure 5.1:** Adaptive time step algorithm.

from the previous time step, or the initial values at the beginning of the transient. A counter $n_{rej}$ is initialised to obtain the number of rejected steps at each time step. The next step is to obtain a first order solution with a desired method, such as a PISO or SIMPLE based algorithm. This is followed by the explicit calculation of the second order solution using (5.3) and then eventually obtaining the error estimate from (5.9). A potential next time step size is then computed, while ensuring it remains between user specified minimum and maximum step sizes $\Delta t_{min}$ and $\Delta t_{max}$. If $\epsilon > 1.0$, the current time step is repeated with the newly calculated $\Delta t$. If $\epsilon \leq 1.0$ or $\Delta t = \Delta t_{min}$ the solution is accepted and advanced to the next time step.

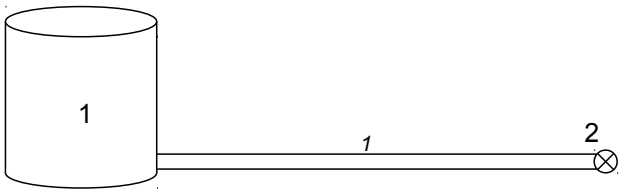



**Figure 5.2:** A single tank and pipe test problem.

## 5.3 Test problems

In this section, test problems are selected in order to demonstrate the benefits of an adaptive time step algorithm as applied to transient flow in pipe networks. The problems are mainly chosen to contain both fast and slow transients, i.e wave phenomena and storage effects. Results will be compared with a constant step method from an efficiency perspective.

### 5.3.1 A single tank and pipe

The first test problem under consideration is a tank connected to a pipe, as illustrated in Figure 5.2. At the end of the pipe (at node 2) a valve is situated which can be opened to ambient conditions. Initially, the flow is at rest, with initial conditions other than ambient conditions, which are summarised in Table 5.1. The pipe and tank characteristics are summarised in Table 5.2. The fluid is air, with $\gamma = 1.4$ and $R = 287$ J·kg$^{-1}$·K$^{-1}$. A spatial increment of $\Delta x = 0.1$ was used, which corresponds to 50 increments of the pipe. $\alpha = 0.7$ was used throughout. For the adaptive time step algorithm, tolerances of $Atol = Rtol = 1 \times 10^{-4}$ were chosen. Ideally, these tolerances can be based on the spatial discretisation error (Berzins 1995). Nevertheless these values are adequate for the purposes of this thesis.

**Table 5.1:** Initial conditions for the problem in Figure 5.2.

| | Pressure (kPa) | Temperature (°C) |
|---|---|---|
| Pipe and tank | 120 | 30 |
| Ambient | 100 | 20 |

**Table 5.2:** Pipe and tank characteristics for the problem in Figure 5.2.

**(a)** Tank

| Tank | Volume ($m^2$) |
|---|---|
| 1 | 10 |

**(b)** Pipe

| Pipe | Diameter (m) | Length (m) | Friction factor |
|---|---|---|---|
| 1 | 0.02 | 5 | 0.02 |

At $t = 0$, the valve was opened instantaneously. This induced rapid transients associated with wave phenomena for a short duration. Eventually, wave phenomena were damped out by the frictional source term in the conservation of momentum, and the transient consisted of the slow storage effects associated with the volume of the tank and pipe. At $t = 60$, the valve was then closed again instantaneously. This once again induced rapid wave phenomena until the fluid came to rest. The time steps required for accurate integration of these different phenomena differs greatly, and it is shown here that an adaptive time step algorithm is an efficient means of treating such problems.

Figure 5.3 illustrates the velocity at the exit of the pipe due to the instantaneous valve opening at $t = 0$. Results are given that were obtained with constant time steps as well as the adaptive algorithm. For purposes of a clear illustration, not all data points are shown. For explicit methods, the time step is usually taken as $\Delta t \leq \Delta x / \left( |u|_{max} + a \right)$, where $a$ is the speed of sound, to accurately capture all wave phenomena. For the current implicit method, a constant time step $\Delta t = 0.0001$ was used which was found to capture wave phenomena up to acceptable accuracy. A second constant step size of $\Delta t = 0.001$ was also used which did not capture wave phenomena accurately, as seen in Figure 5.3.

At approximately $t = 0.2$, the fast transients resulting from the instantaneous valve opening was completed, and was followed by the slow transient associated with storage effects for the remainder of the transient. The velocity at the exit of the pipe during the entire transient is illustrated in Figure 5.4, which overshadows the duration of the fast transient in Figure 5.3. The pressure and temperature in the tank during the entire transient are illustrated in Figures 5.5 and 5.6 respectively. Results are once again given for a constant time step as well as the adaptive algorithm. The relatively large constant time step of $\Delta t = 0.001$ was used which gives sufficient accuracy for the slow

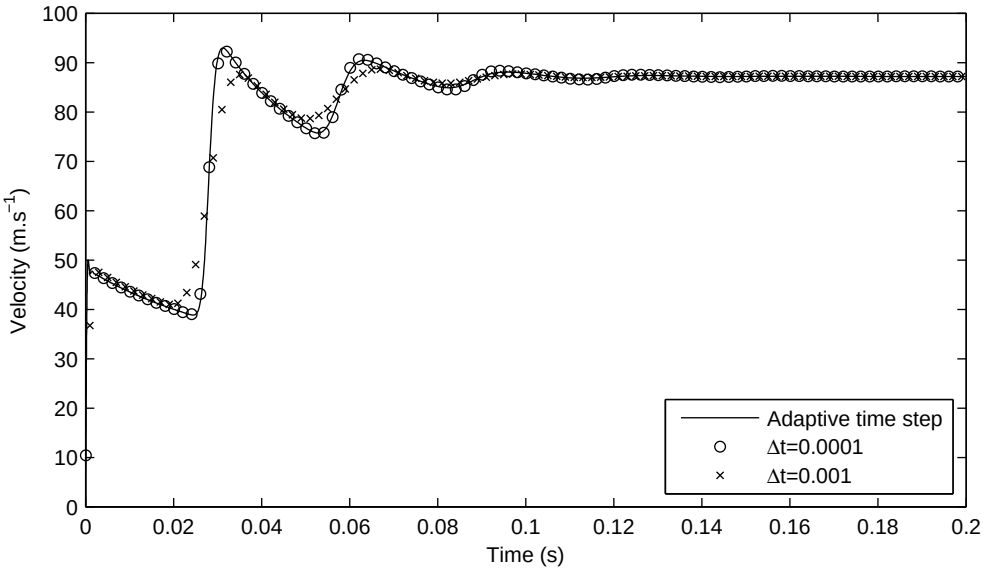



**Figure 5.3:** The solution of the velocity at the exit of the pipe at the onset of the fast transient due to the instantaneous valve opening as calculated with the adaptive and a constant time step methods.

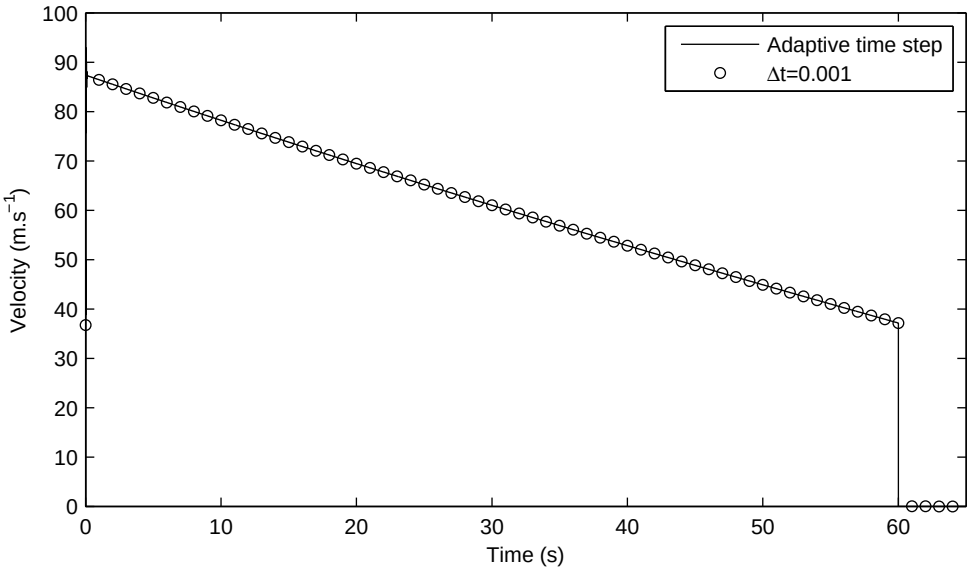



**Figure 5.4:** The solution of the velocity at the exit of the pipe during the entire transient as calculated with the adaptive and a constant time step methods.

transient. The adaptive time step method selected a larger step size of approximately 0.03 for the slow transient, which provides sufficient accuracy as evident in Figures 5.5 and 5.6.

At $t = 60$ when the valve was closed instantaneously the second fast transient took

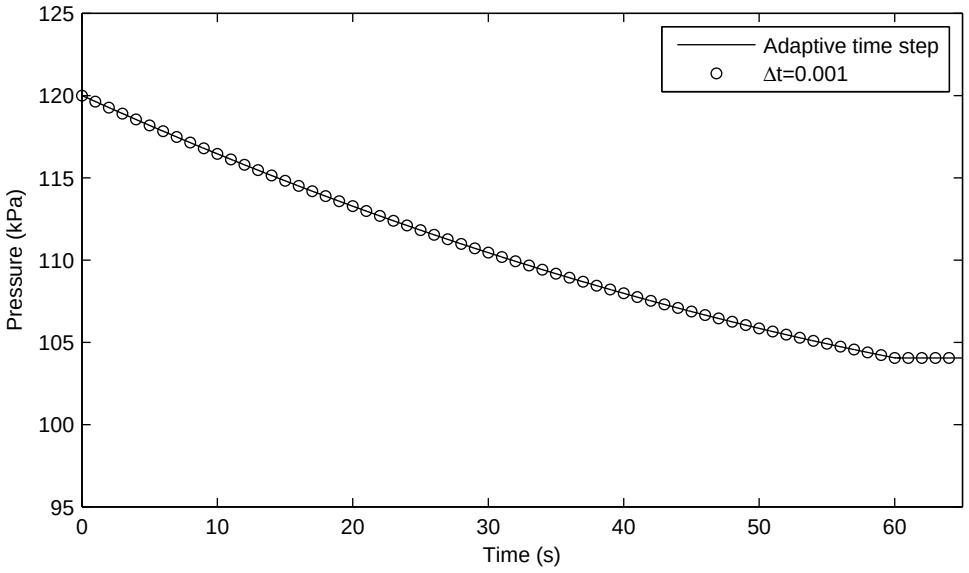



**Figure 5.5:** The solution of the pressure in the tank as calculated with the adaptive and a constant time step methods.

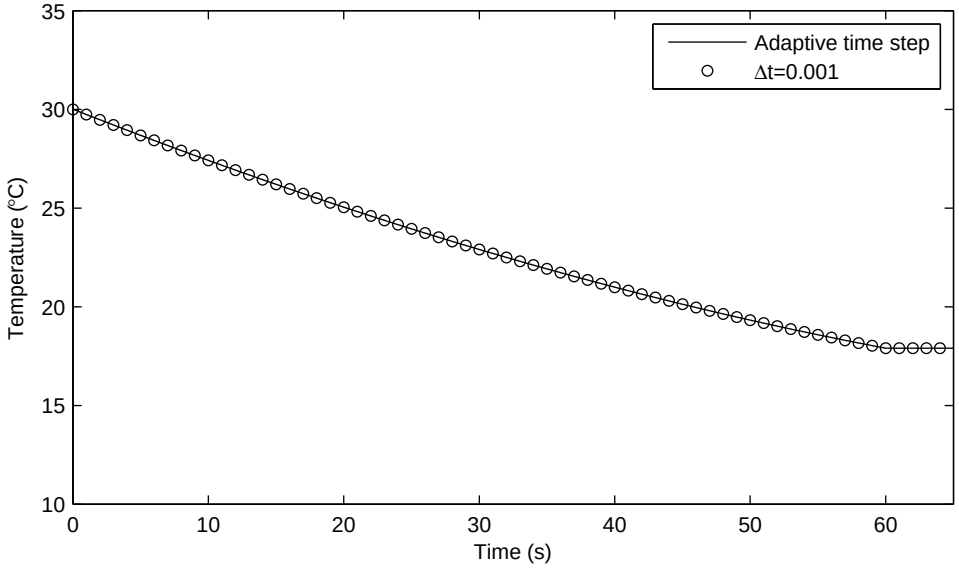



**Figure 5.6:** The solution of the temperature in the tank as calculated with the adaptive and a constant time step methods.

place. The pressure and temperature at the middle of the pipe during the fast transient are illustrated in Figures 5.7 and 5.8 respectively. Once again, results are illustrated for constant time step sizes $\Delta t = 0.0001$ and $\Delta t = 0.001$. Again it can be seen that the step size of $\Delta t = 0.001$ provides insufficient accuracy. The results of $\Delta t = 0.0001$ and the adaptive time step algorithm are once again similarly accurate.

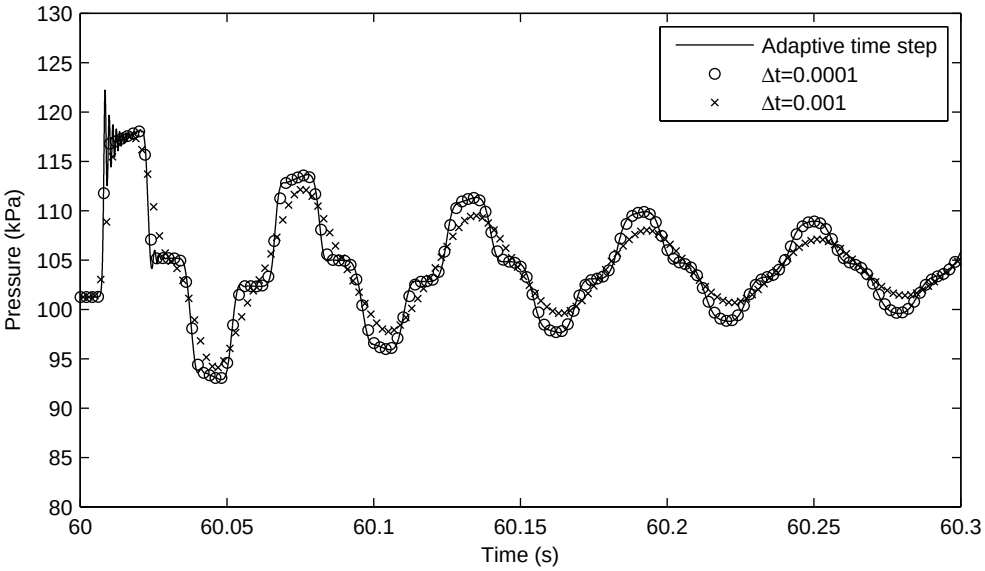



**Figure 5.7:** The solution of the pressure at the middle of the pipe at the onset of the fast transient due to the instantaneous valve closure as calculated with the adaptive and constant time step methods.

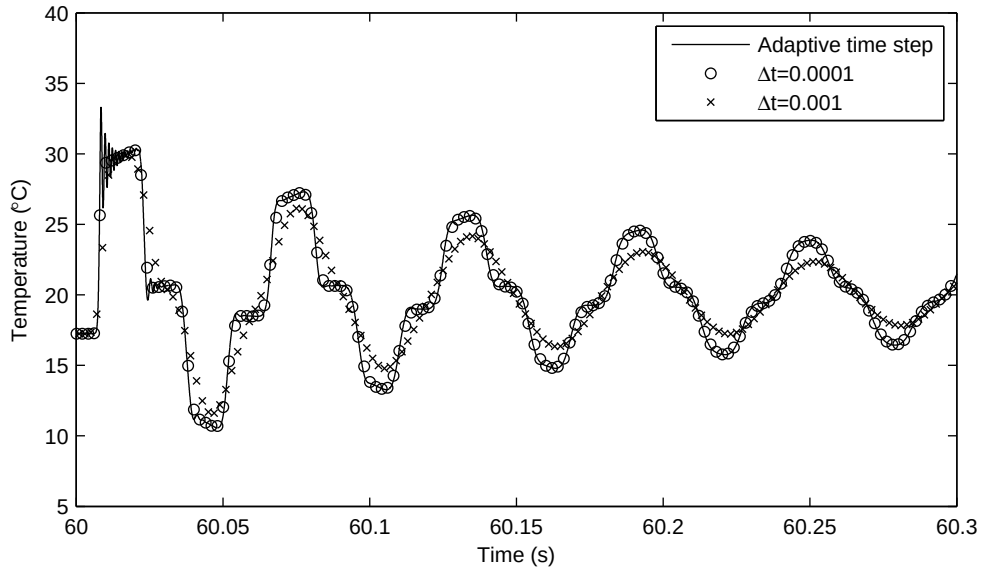



**Figure 5.8:** The solution of the temperature at the middle of the pipe at the onset of the fast transient due to the instantaneous valve closure as calculated with the adaptive and constant time step methods.

The oscillations at the onset of the transient, as seen in Figures 5.7 & 5.8 are largely attributed to the spatial discretisation used, rather than the time integration scheme. To see this, a detailed illustration of the initial transient is given in Figure 5.9 with the

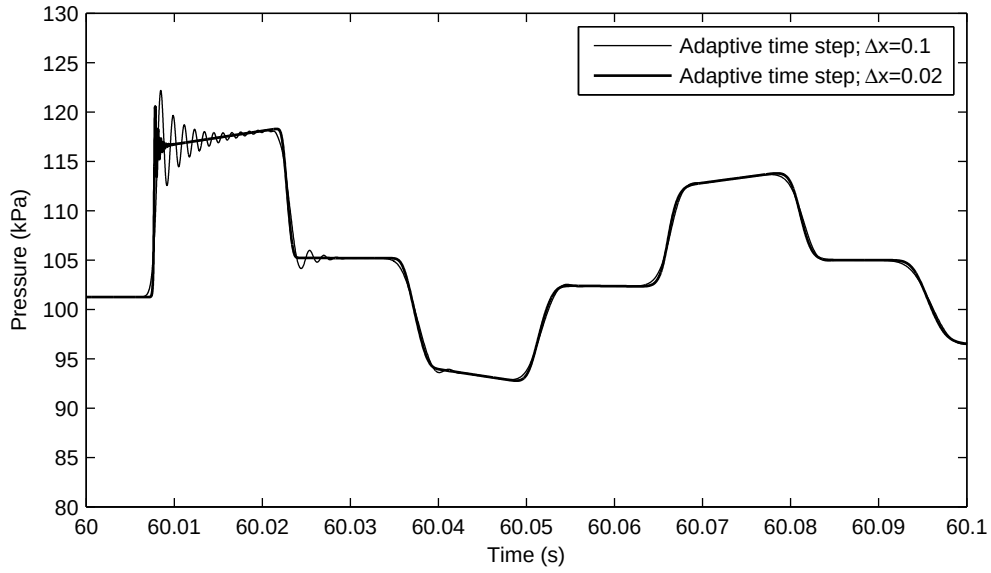



**Figure 5.9:** A detail view of the initial pressure solution as obtained with different spatial meshes.

current mesh as well as a finer mesh of $\Delta x = 0.02$. The improved resolution obtained with the finer spatial mesh is apparent, and the oscillations are significantly reduced. This illustrates the importance of spatial discretisation error control in conjunction with the temporal error control as described in Berzins (1995). These methods, however, fall beyond the scope of this study.

Figure 5.10 illustrates the time step sizes used throughout the entire transient as determined by the adaptive time step algorithm. Initially, the time step was small, corresponding to the initial fast wave phenomena. Then in the neighbourhood of $1 \times 10^3$ steps, the step size was increased. This corresponds to the slow transient as described previously where a large time step provides sufficient accuracy. As can be seen, the time step was also steadily increased for this part as the slow transient progressed. Finally, at approximately $3 \times 10^3$ steps, the time step was reduced to provide accurate integration of the second fast transient. At the end of the transient, the step size increased rapidly up to the maximum step size specified $\Delta t_{max}$. From this it is clear that the adaptive time step algorithm effectively detects and treats all the different time scales that are present, even those introduced by an instantaneous change in boundary conditions.

The area under the curve shown in Figure 5.10 gives the actual time covered by

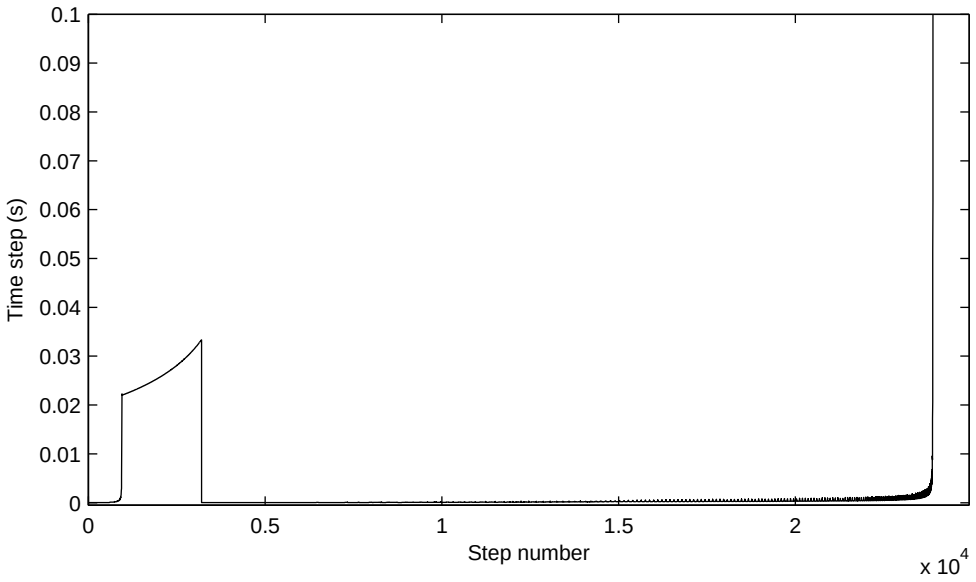



**Figure 5.10:** Time step size vs. step number for the entire transient.

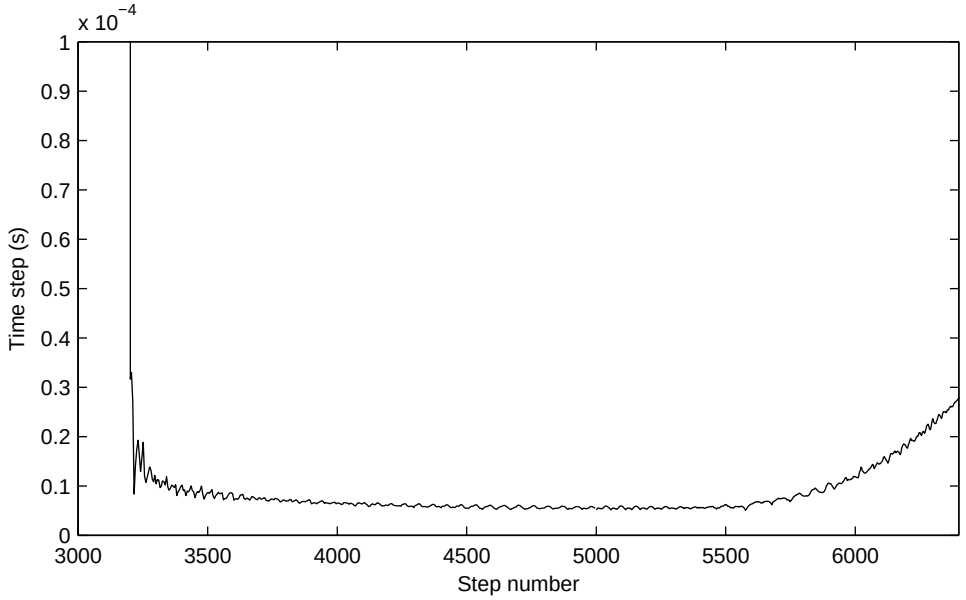



**Figure 5.11:** Time step size vs. step number for the fast transient during the valve closure.

the adaptive algorithm. It is clear that the duration of the slow transient was longer than that of the fast transients, while computationally the fast transient dominated the simulation cost. Figure 5.11 illustrates the time step size during the second fast transient in more detail. The time step remained approximately constant for most of the duration, as expected for hyperbolic problems. The step size was in the vicinity of $\Delta t = 1 \times 10^{-5}$. This is a result of the tolerances used in this study, i.e $Atol = Rtol = 1 \times 10^{-4}$. Alternatively, a minimum time step can be specified if these values

**Table 5.3:** Computational cost for adaptive and fixed time step methods .

| Method | Time steps | Rejected steps | Computational time (s) |
|---|---|---|---|
| Adaptive time step | 23886 | 210 | 8.329 |
| $\Delta t = 0.0001$ | 650000 | – | 213.844 |
| $\Delta t = 0.001$ | 65000 | – | 21.384 |

are considered to small, or the tolerances can be based on the spatial error. Typically, a minimum time step of $\Delta t = \Delta x / \left( |u|_{max} + a \right)$ can be prescribed. Another option would be to filter the contribution of the discontinuous part of the solution to the error, as described in Hornëll & Lötstedt (2001).

To summarize the results obtained, the computational costs associated with the constant and adaptive time step methods are given in Table 5.3. The transient was solved up to $t = 65$ s, which corresponds to approximately the end of the transient. All simulations were done on a 2.67 GHz Intel® Core™2 Duo processor while a generic direct matrix inversion has been adopted to solve the linear system of equations. The computational time of the adaptive time algorithm was 8.329 s. The two constant time step methods $\Delta t = 0.0001$ s and $\Delta t = 0.001$ s required 213.844 s and 21.384 s respectively. From this, the adaptive time step method is more than 25 times more efficient than the $\Delta t = 0.0001$ s step method and more than 2.5 times more efficient than the $\Delta t = 0.001$ s step method.

The total time steps required by the adaptive algorithm is 23866. In addition to the 210 rejected steps, this required a total of $23866 + 210 = 24096$ effective implicit time steps. The two constant time step methods $\Delta t = 0.0001$ s and $\Delta t = 0.001$ s required 650000 and 65000 steps respectively. From this perspective, the adaptive algorithm is approximately 27 times more efficient than the $\Delta t = 0.0001$ s step method and roughly 2.7 times more efficient than the $\Delta t = 0.001$ s step method, only slightly more than the savings based on computational time. This indicates that the additional function evaluations required for error control do not contribute significantly to the overall expense of the implicit method, since this is done with the inexpensive explicit error estimate. The constant time step of $\Delta t = 0.001$ s provided only accurate integration of the slow transient, and is not suitable for the fast transient. Significant savings are observed by using the adaptive time step algorithm in order to obtain an accurate

solution.

### Algorithm performance at higher wave speeds

Typical pipe flow problems can consist of fluids with high wave speeds, such as Helium. Very high wave speeds such as those associated with water hammer are also commonly encountered. The successful treatment of problems associated with very high wave speeds is briefly demonstrated in this section. This was done again with the previous test problem with two adjustments. Firstly, the value of $R$ was increased with a factor of 1000, i.e. $R = 287 \times 10^3$. This gave rise to a much higher wave speed. Secondly, the valve is closed at $t = 2$ s.

The pressure in the tank for this problem during the entire transient is illustrated in Figure 5.12. Results are given for a constant time step as well as the adaptive algorithm. The constant time step of $\Delta t = 0.0001$ was used which gave sufficient accuracy for the slow transient. The adaptive time step method selected a larger step size of approximately 0.0008 for the slow transient. Again it is clear that the slow transient is resolved accurately.

At $t = 2$ when the valve was closed instantaneously the second fast transient took place. The pressure at the middle of the pipe during the fast transient is illustrated

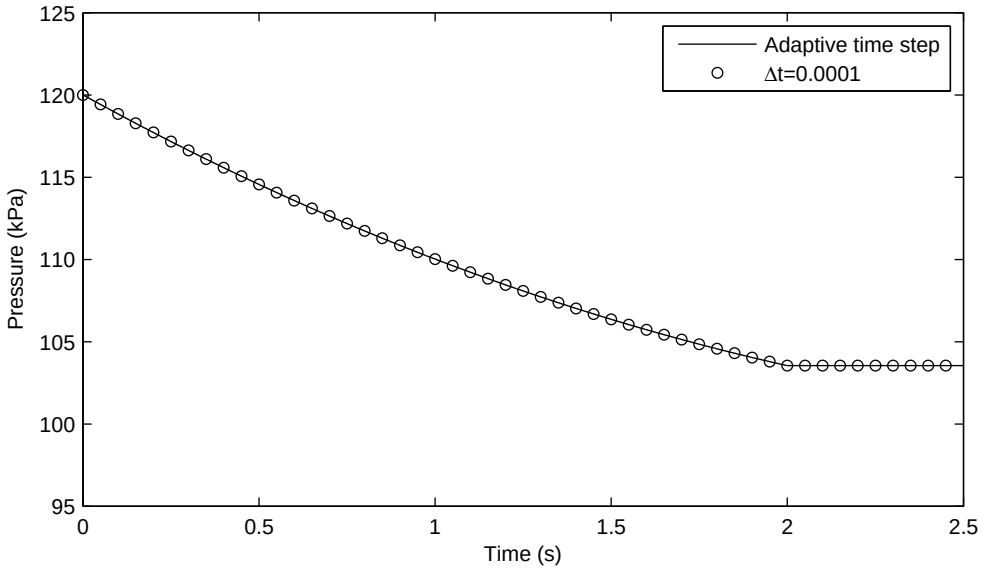



**Figure 5.12:** The solution of the pressure in the tank as calculated with the adaptive and a constant time step methods for a fluid with a very high wave speed.

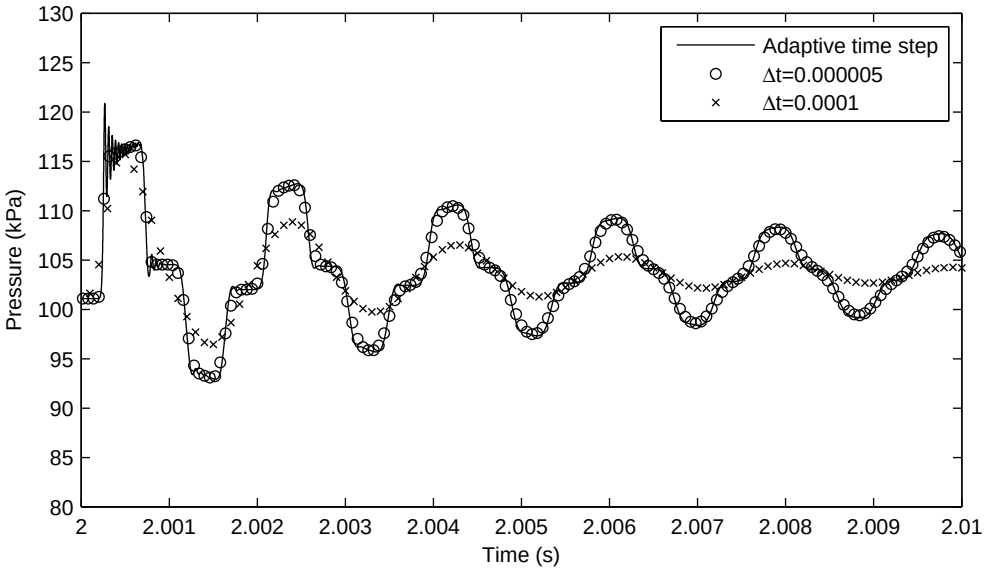



**Figure 5.13:** The solution of the pressure at the middle of the pipe at the onset of the fast transient due to the instantaneous valve closure for a fluid with a very high wave speed.

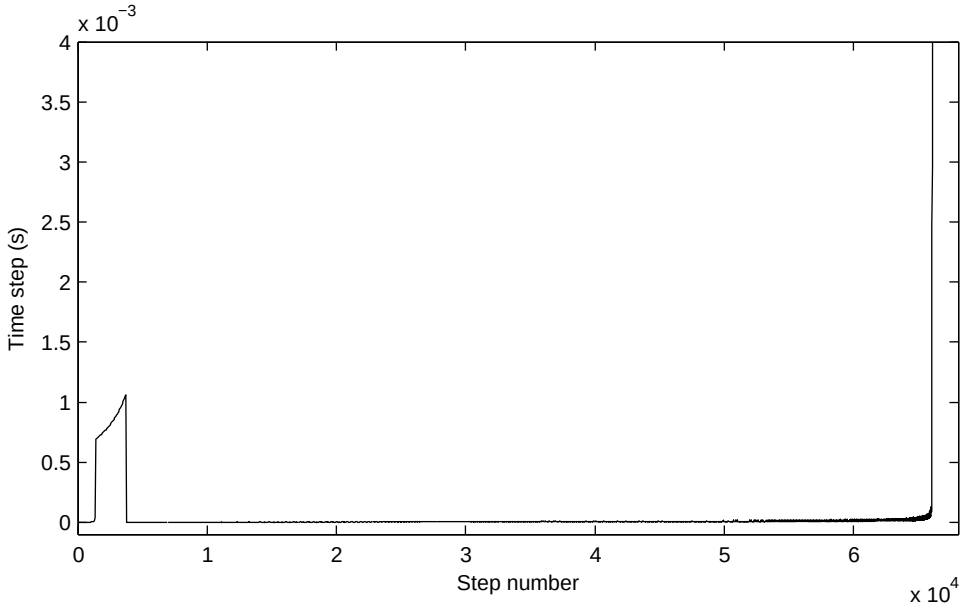



**Figure 5.14:** Time step size vs. step number for the entire transient for a fluid with a very high wave speed.

in Figure 5.13. Results are illustrated for constant time step sizes $\Delta t = 0.000005$ and $\Delta t = 0.0001$. The step size of $\Delta t = 0.0001$ provides insufficient accuracy for the fast transient. The step size of $\Delta t = 0.000005$ was necessary to obtain an accurate solution, which is again similar to the result of the adaptive time step algorithm.

Figure 5.14 illustrates the time step sizes used throughout the entire transient as

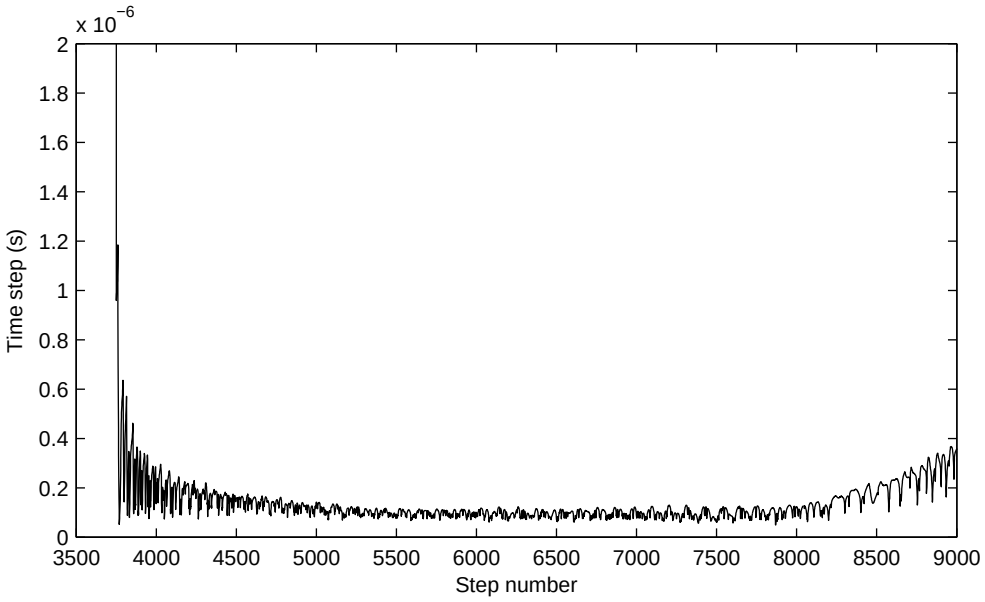



**Figure 5.15:** Time step size vs. step number for the fast transient during the valve closure for a fluid with a very high wave speed.

determined by the adaptive time step algorithm. The result is as exactly as expected, and the discussion is similar to that in the previous section. Figure 5.15 illustrates the time step size during the fast transient after valve closure in more detail. The extremely small time step used by the algorithm to accurately capture the very high speed wave phenomena is observed. From this it can be concluded that the adaptive time step algorithm performs correctly in the presence of very high wave speeds, and problems such as water hammer will be treated successfully and similar computational savings are expected.

### 5.3.2 A simple branching network with tanks

The second test problem under consideration is two tanks connected by three pipes, as illustrated in Figure 5.16. At the end of the pipe (at node 4) is a valve which can be opened to ambient conditions. Initially, the flow is at rest, with initial conditions other than ambient conditions, which are summarised in Table 5.4. The pipe and tank characteristics are summarised in Table 5.5. The fluid is air, with $\gamma = 1.4$ and $R = 287$ J·kg$^{-1}$·K$^{-1}$. A spatial increment of $\Delta x = 0.1$ was used for all pipes. $\alpha = 0.7$ was used throughout. For the adaptive time step algorithm, tolerances of $Atol = Rtol = 1 \times 10^{-4}$ were chosen as in the previous test problem.

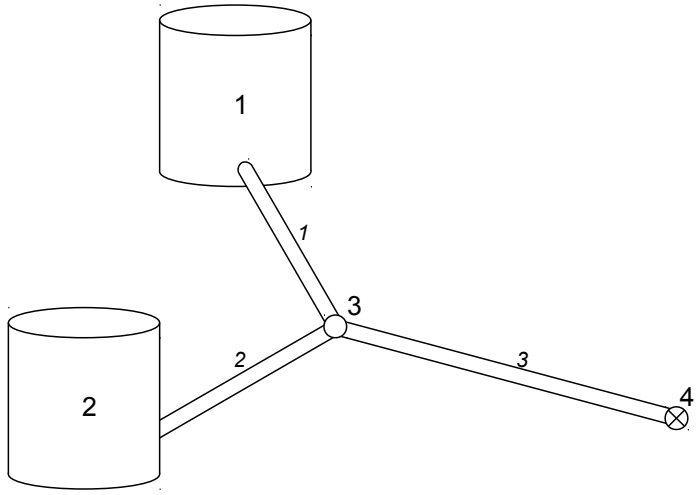



**Figure 5.16:** A simple branching network consisting of two tanks and three pipes.

The characteristics of the pipes and tanks were chosen to illustrate different time scales associated with the slow storage effects. From Table 5.5, the volume of tank 1 is larger than that of tank 2, pipe 1 is longer than pipe 2 and the diameter of pipe 1 is smaller than that of pipe 2. These values all contributed to a rapid blow down of the second tank (node 2) and a much slower blow down of the first tank (node 1). Thus, different time scales existed in the slow storage effects as well, and it is demonstrated that the adaptive time step algorithm can treat these effects efficiently.

As in the first test problem, at $t = 0$, the valve was opened instantaneously which induced rapid transients associated with wave phenomena for a short duration. Wave phenomena were damped out eventually and the transient consisted of the slow storage effects associated with the volumes of the tanks and pipes. At $t = 60$, the valve was closed instantaneously which again induced rapid wave phenomena. At the end of the wave phenomena, the network stabilised until a uniform state was reached.

**Table 5.4:** Initial conditions for the problem in Figure 5.16.

| | Pressure (kPa) | Temperature (°C) |
|---|---|---|
| Pipes and tanks | 120 | 30 |
| Ambient | 100 | 20 |

**Table 5.5:** Pipe and tank characteristics for the problem in Figure 5.16.

**(a)** Tanks

| Tank | Volume ($m^3$) |
|---|---|
| 1 | 10 |
| 2 | 5 |

**(b)** Pipes

| Pipe | Diameter (m) | Length (m) | Friction factor |
|---|---|---|---|
| 1 | 0.02 | 5 | 0.02 |
| 2 | 0.04 | 2 | 0.02 |
| 3 | 0.05 | 3 | 0.02 |

Figure 5.17 illustrates the velocity at the exit of pipe 3 due to the instantaneous valve opening at $t = 0$. Again, results given were obtained with constant time steps as well as the adaptive algorithm. The step size of $\Delta t = 0.001$ again did not capture wave phenomena accurately, as seen in Figure 5.17.

The pressure and temperature in the tanks during the entire transient are illustrated in Figures 5.18 and 5.19 respectively. The faster transient associated with tank 2 and slower transient associated with tank 1 can be clearly seen. At $t = 60$ when the valve was closed, the two tanks stabilised until a uniform pressure was reached. Results are again given for the adaptive time step and constant time step of $\Delta t = 0.001$.

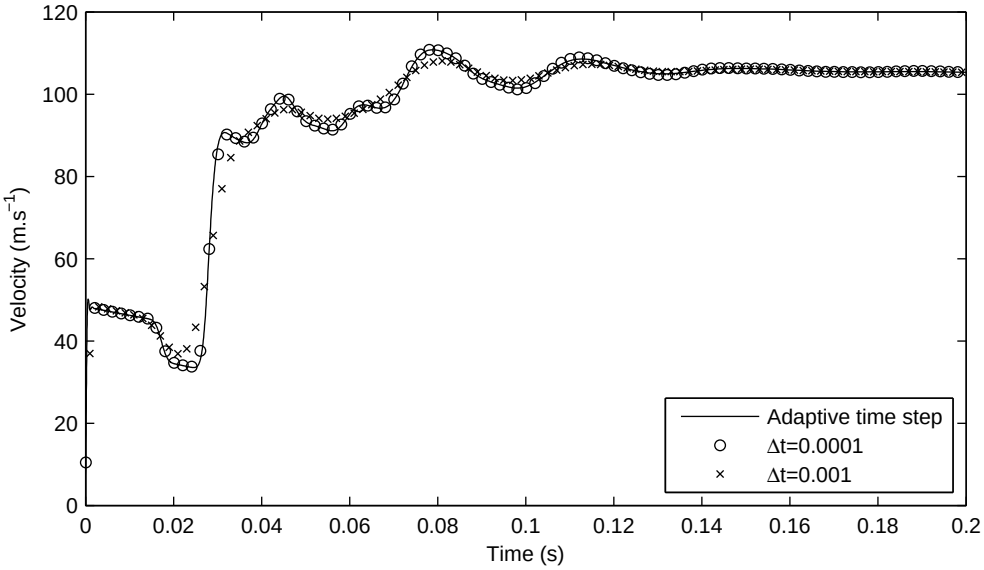



**Figure 5.17:** The solution of the velocity at the exit of the pipe at the onset of the fast transient due to the instantaneous valve opening as calculated with the adaptive and a constant time step methods.

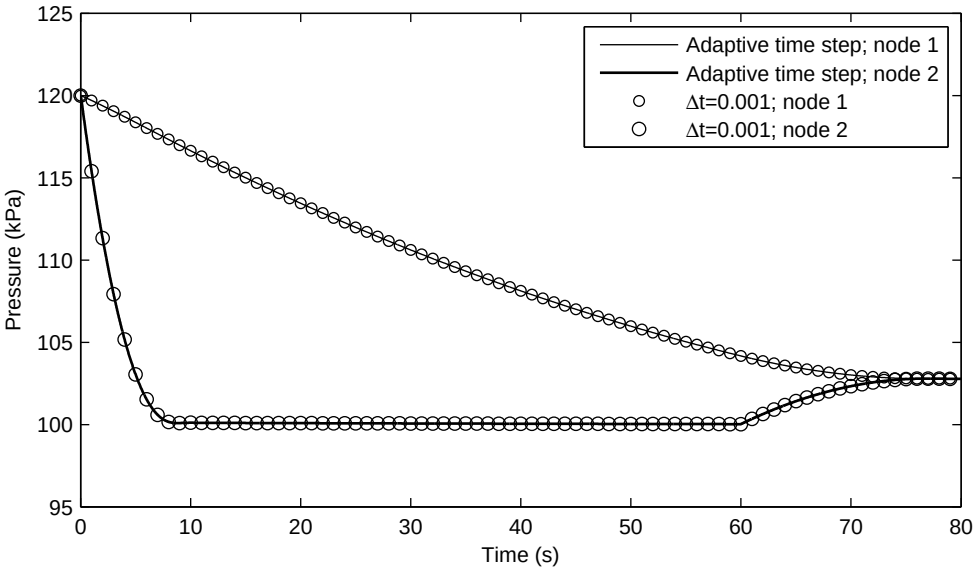



**Figure 5.18:** The solution of the pressure in the tanks as calculated with the adaptive and a constant time step methods.

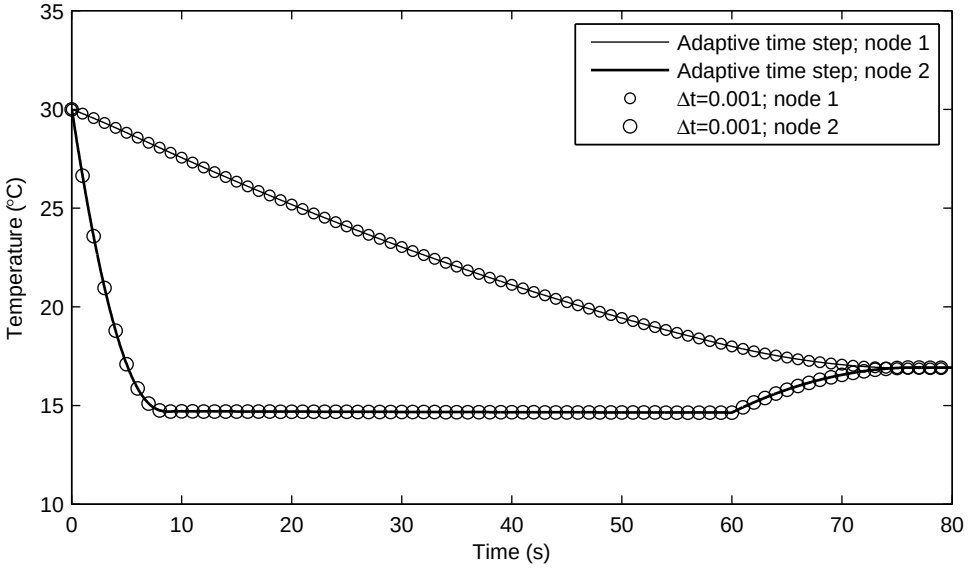



**Figure 5.19:** The solution of the temperature in the tanks as calculated with the adaptive and a constant time step methods.

At $t = 60$ the valve was closed instantaneously. The pressure and temperature at the junction (node 3) during the onset of the resulting fast transient are illustrated in Figures 5.20 and 5.21 respectively. Once again, results are illustrated for constant time step sizes $\Delta t = 0.0001$ and $\Delta t = 0.001$. Again it can be seen that the step size of $\Delta t = 0.001$ provided insufficient accuracy while the results of $\Delta t = 0.0001$ and the

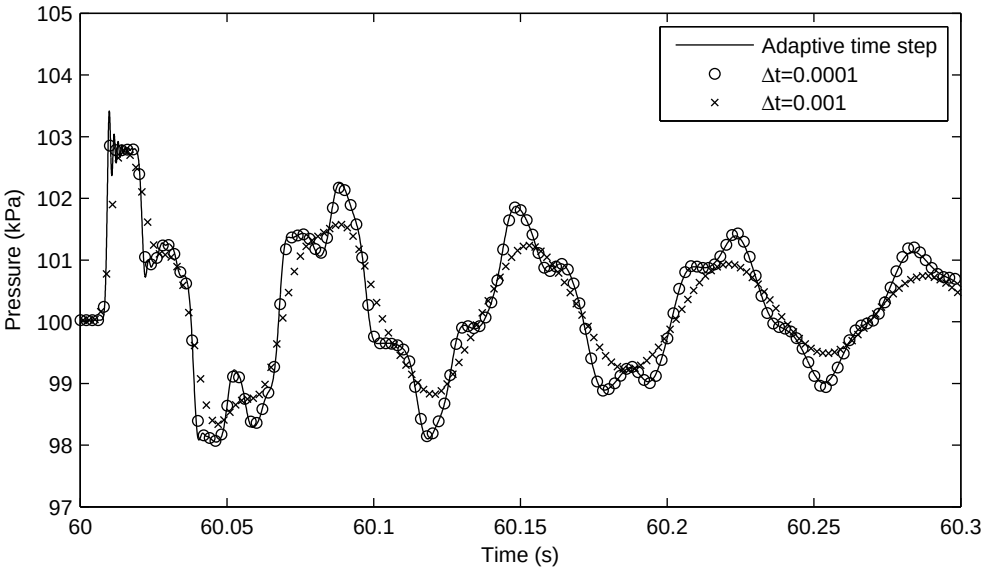



**Figure 5.20:** The solution of the pressure at the junction (node 3) of the pipe at the onset of the fast transient due to the instantaneous valve closure as calculated with the adaptive and constant time step methods.

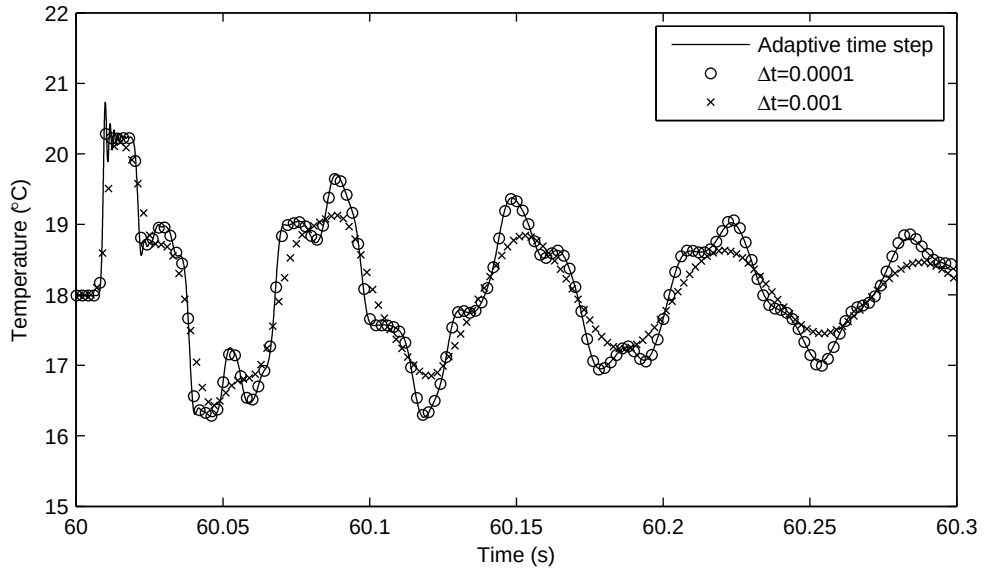



**Figure 5.21:** The solution of the temperature at the junction (node 3) of the pipe at the onset of the fast transient due to the instantaneous valve closure as calculated with the adaptive and constant time step methods.

adaptive time step algorithm are sufficiently accurate.

Another observation can be made at the very end of the transient. Figure 5.22 illustrates the velocity at the middle of pipe 1 near the very end of the transient. The

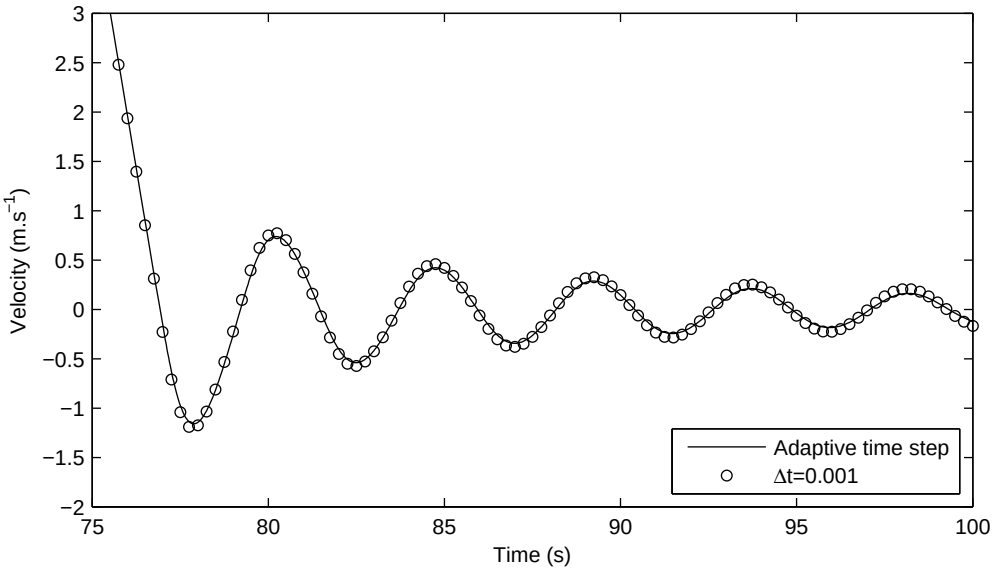



**Figure 5.22:** The solution of the velocity at the middle of pipe 1 at the very end of the transient.

velocity oscillated slowly until the fluid eventually came to rest.

Figure 5.23 illustrates the time step sizes used throughout the entire transient as determined by the adaptive time step algorithm for the second test problem. Initially, the time step was once again small, corresponding to the initial fast wave phenomena. Then in the neighbourhood of 1200 steps, the step size was increased, which corresponded to the slow transients. For this second test problem, however, two distinct regions exist. The first part corresponds to the faster storage effect of tank 1 and the second part to the slower storage effect of tank 2 which can be seen in Figures 5.18 and 5.19. Figure 5.24 provides a more detailed view of this region. At approximately 3200 steps, the time step was reduced to provide accurate integration of the second fast transient associated with the valve closure at $t = 60$ s. Finally at roughly 7200 steps, the transient was again dominated by slower storage effects, this time due to the mixing and stabilisation of the fluid in the tanks and pipes until the end of the transient was reached. The oscillating time step selection observed in Figure 5.23 near the end of the transient is associated with the oscillating solution illustrated in Figure 5.22. The adaptive time step algorithm automatically detects and treats all phenomena correctly with appropriate step sizes.

The computational costs associated with the constant and adaptive time step meth-

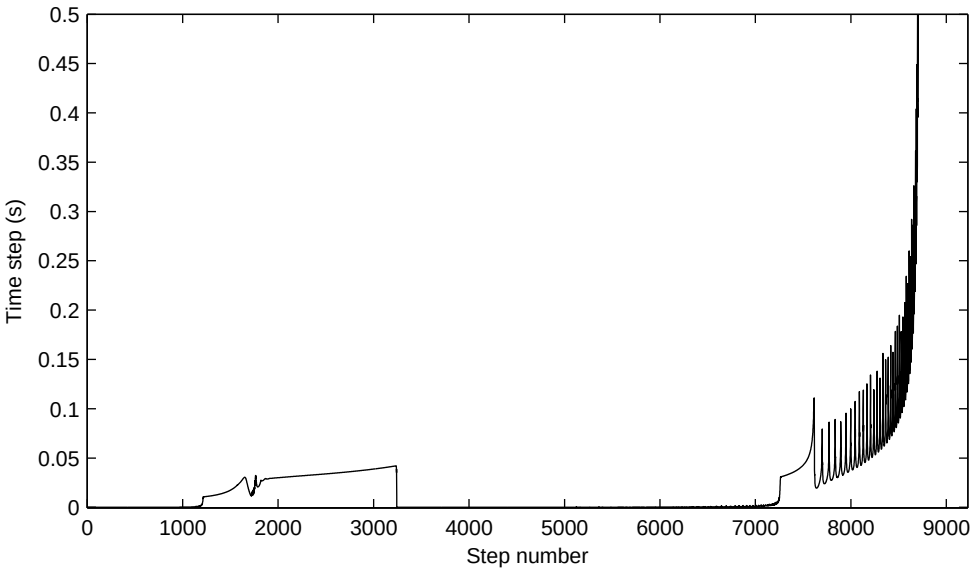



**Figure 5.23:** Time step size vs. step number for the entire transient.

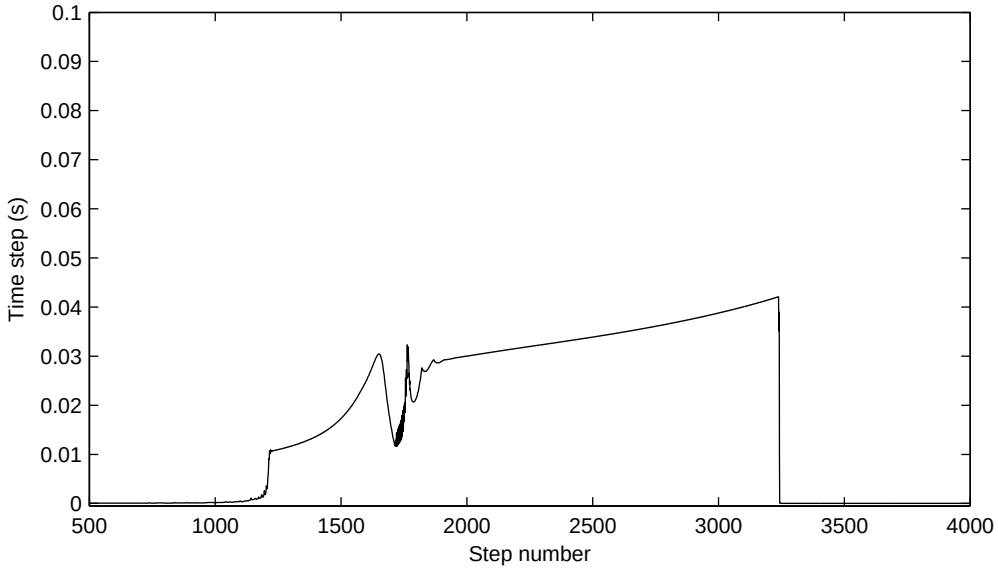



**Figure 5.24:** Time step size vs. step number for the slow transient. Two distinct regions are present, the first one due to the faster transient (tank 2) and the second one due to the slower transient (tank 1).

ods are given in Table 5.6. The transient was solved up to $t = 80$ s. The computational time for the adaptive time algorithm was 12.016 s. The two constant time step methods $\Delta t = 0.0001$ s and $\Delta t = 0.001$ s required 1211.281 s and 121.128 s respectively. The adaptive time step method is more than 100 times more efficient than the $\Delta t = 0.0001$ s step method and more than 10 times more efficient than the $\Delta t = 0.001$ s step method.

**Table 5.6:** Computational cost for adaptive and fixed time step methods.

| Method | Time steps | Rejected steps | Computational time (s) |
|---|---|---|---|
| Adaptive time step | 7776 | 49 | 12.016 |
| $\Delta t = 0.0001$ | 800000 | – | 1211.281 |
| $\Delta t = 0.001$ | 80000 | – | 121.128 |

The total time steps required by the adaptive algorithm is $7776 + 49 = 7825$ effective implicit time steps. The two constant time step methods $\Delta t = 0.0001$ s and $\Delta t = 0.001$ s required 800000 and 80000 steps respectively. From this perspective, the adaptive time step method is again more than 100 times more efficient than the $\Delta t = 0.0001$ s step method and more than 10 times more efficient than the $\Delta t = 0.001$ s step method. Once again these factors are only slightly more than the factors based on computational time, again indicating that the additional function evaluations required for error control do not contribute significantly to the overall expense of the implicit method. The constant time step of $\Delta t = 0.001$ s only provided sufficiently accurate integration of the slow transient, and is not suitable for the fast transient. Once again, significant savings can be observed by using the the adaptive time step algorithm in order to obtain an accurate solution.

## 5.4 Conclusion

This chapter discussed an adaptive time step algorithm which can be applied efficiently with conventional methods such as PISO and SIMPLE algorithms. Inexpensive error estimates were given which are valid for any first order integration scheme. In this paper, the $\alpha$-scheme was used with a value of $\alpha = 0.7$.

It was demonstrated the adaptive time step algorithm provides efficient and accurate integration of pipe networks consisting of different flow phenomena with vastly different time scales. Two test problems were presented. The first was a single pipe and tank where both fast and slow transients were present. The second test problem was a simple branching pipe network consisting of two tanks and three pipes. This problem introduced slow transients with different time scales. The adaptive time step algorithm was found to correctly detect the presence and advent of all transient phenomena and

automatically select appropriate time steps for accurate integration.

For all problems the performance of the adaptive time step algorithm was compared with that of constant time step methods. For the first test problem the adaptive time step method was found to be more than 25 times more efficient than a constant time step method of $\Delta t = 0.0001$ for approximately similar accuracy. For the second test problem the adaptive time step method was found to be more than 100 times more efficient than a constant time step method of $\Delta t = 0.0001$ for approximately similar accuracy. It can hence be concluded that adaptive time stepping is an efficient and essential component of the simulation of transient flow in pipe networks consisting of different transient phenomena.

# Chapter 6

# Fully adaptive algorithm

## 6.1 Introduction

In the previous chapter, an adaptive time step algorithm was introduced that can be used with any conventional first order methods for the simulation of transient flow in pipe networks. A single implicit method was used throughout that chapter. Since typical pipe network problems consist of vastly different time scales, as demonstrated in the previous chapter, stiffness is always encountered. This chapter focusses on the next logical step, i.e. the automatic selection of appropriate algorithms in conjunction with adaptive time stepping. For brevity, such an algorithm is referred to as a fully adaptive algorithm.

## 6.2 Automatic algorithm selection

During the non-stiff part of the transient the time step of conditionally stable methods, e.g. fully explicit or semi-implicit methods, is usually determined by accuracy considerations rather than stability. There is therefore generally no good reason not to use one of these methods since they are computationally less expensive than implicit schemes in the presence of fast wave phenomena. However, once stiffness is encountered these methods become driven by stability, i.e. the step size lies near the border of the stability domain, and an implicit method becomes more suitable. The onset of stiffness takes place when the fast wave phenomena are damped out and the transient is dominated by much slower effects such as storage phenomena.

Automatic switching can be done in a straightforward manner if the potential next time step $\Delta t$ based on accuracy and maximum allowable time step $\Delta t_s$ based on stability for explicit or semi-implicit methods are known. The semi-implicit as well as explicit methods usually have sharp stability criteria which can be used effectively in such a procedure.

### 6.2.1 Choice of methods

For the purpose of demonstration, the methods used here are the non-iterative static pressure method based on the $\alpha$-scheme as well as the semi-implicit method which only treats acoustic terms implicitly (the method and its stability criterion are derived in Appendix A). Both are done on a staggered grid and incorporate the non-conservative static pressure formulation of conservation of momentum. The automatic switching procedure, however, is not limited to any specific method and can therefore also be applied to the non-iterative total pressure formulation that was derived in Chapter 3.

A reason for the choice of static pressure methods stems from the fact that a semi-implicit method with sharp stability criteria based on total pressures is not available at present, and no attempt is made in this chapter to derive such a method. In addition, it is preferred that the different time integration schemes use the same formulation of all conservation laws.

Explicit methods with sharp stability criteria are usually formulated on a collocated grid as opposed to the staggered grid of the aforementioned methods, and this will require additional work associated with interpolation between staggered and collocated grid arrangements. Therefore explicit methods are not included in this study, although it will result in unquestionable additional savings if included in the algorithm. Nevertheless, the two methods under consideration are adequate for this study.

### 6.2.2 Switching procedure

The basic procedure of the algorithm selection is outlined in Figure 6.1. At the current time step, the adaptive time step method of Chapter 5 is applied with *method*, which can be either the semi-implicit or implicit methods where the potential step size $\Delta t$ at the next time time step is also obtained. Next, the maximum time step $\Delta t_s$ of the semi-implicit method for stable integration is calculated based on the newly obtained

1. Apply the adaptive time step algorithm of Chapter 5 at the current time step with $method$ and obtain the step size $\Delta t$ for the next time step.

2. Compute the maximum step size for the semi-implicit method $\Delta t_s$.

3. If $(method = semi{-}implicit)$ then

   If $(\Delta t > fac_1 \cdot \Delta t_s)$ then

   $method = implicit$

   Else if $(\Delta t > \Delta t_s)$ then

   $\Delta t = \Delta t_s$

   End if

   Else if $(method = implicit)$ then

   If $(\Delta t < fac_2 \cdot \Delta t_s)$ then

   $method = semi{-}implicit$

   End if

   End if

4. Return to step 1

**Figure 6.1:** Automatic algorithm switching.

velocity, i.e $\Delta t_s = 0.8 \cdot \Delta x/|u|_{max}$. A safety factor of 0.8 was introduced to ensure that the stability condition is satisfied, i.e. $\Delta t \leq \Delta x/|u|_{max}$.

If the current method is the semi-implicit method, the algorithm switches to the implicit method for the next time step if $\Delta t > fac_1 \cdot \Delta t_s$. Here $fac_1 > 1$ is a factor introduced to prevent immediate switching from the semi-implicit method to the implicit method. One reason for its introduction is to prevent an unwanted back-and-forth switching between algorithms when the algorithms have significantly different error estimates. Another reason is to remain with the semi-implicit method since this method is approximately three times faster than the implicit method, and $fac_1$ can be

set in the neighbourhood of 2. For this study, unless otherwise noted, $fac_1$ was set to 1.1. If $\Delta t_s < \Delta t \leq fac_1 \cdot \Delta t_s$ then $\Delta t$ is set to $\Delta t = \Delta t_s$.

If the current method is the implicit method, the algorithm switches to the semi-implicit method for the next time step if $\Delta t < fac_2 \cdot \Delta t_s$. Here $fac_2 < 1$ is a factor similar to $fac_1$ to prevent immediate switching from the implicit method to the semi-implicit method. For the implicit method, it is usually desired to switch as soon as possible to the semi-implicit method from a computational cost perspective, and a value close to 1 is desired. Unless otherwise stated, $fac_2$ was set to 0.9 throughout.

## 6.3 Test problems

In this section the test problems of Chapter 5 are once again used to demonstrate the additional benefits of a fully adaptive algorithm as applied to transient flow in pipe networks. Results are compared with constant step as well as adaptive time step methods.

### 6.3.1 A single tank and pipe

For convenience, the first test problem under consideration is once again illustrated in Figure 6.2. Initially, the flow is at rest, with initial conditions other than ambient conditions, which are summarised in Table 6.1. The pipe and tank characteristics are summarised in Table 6.2. The fluid is air, with $\gamma = 1.4$ and $R = 287$ J·kg$^{-1}$·K$^{-1}$. A spatial increment of $\Delta x = 0.1$ was used. For this problem, however, $\alpha = 1$ was used for the implicit method. As the semi-implicit method treats terms either fully explicit or fully implicit, this leads to error estimates of similar magnitudes which is preferred

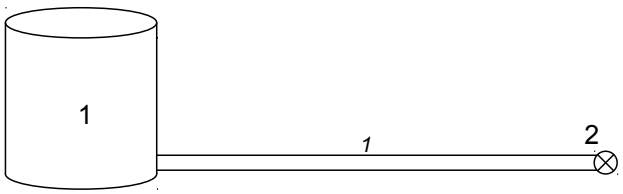



**Figure 6.2:** A single tank and pipe test problem.

**Table 6.1:** Initial conditions for the problem in Figure 6.2.

| | Pressure (kPa) | Temperature (°C) |
|---|---|---|
| Pipe and tank | 120 | 30 |
| Ambient | 100 | 20 |

**Table 6.2:** Pipe and tank characteristics for the problem in Figure 6.2.

**(a)** Tank

| Tank | Volume ($m^2$) |
|---|---|
| 1 | 10 |

**(b)** Pipe

| Pipe | Diameter (m) | Length (m) | Friction factor |
|---|---|---|---|
| 1 | 0.02 | 5 | 0.02 |

for the switching procedure. For both adaptive time step algorithms, tolerances of $Atol = Rtol = 1 \times 10^{-4}$ were chosen as before.

The boundary conditions are again the same as in Chapter 5. At $t = 0$, the valve was opened instantaneously which induced rapid transients associated with wave phenomena for a short duration. Wave phenomena were damped out by friction and the transient consisted of the slow storage effects. At $t = 60$, the valve was then closed again instantaneously which again induced rapid wave phenomena until the fluid came to rest.

Figure 6.3 illustrates the velocity at the exit of the pipe due to the instantaneous valve opening at $t = 0$. Results are given that were obtained with a constant time step, adaptive time step with the semi-implicit method, adaptive time step with the implicit method as well as the fully adaptive algorithm. The fully adaptive method started off with the semi-implicit method and switched to the fully implicit method at $t = 0.092$, indicated in Figure 6.3. The switching procedure took place as expected, i.e. the method switched to implicit once the majority of the wave effects were damped out. A constant step size of $\Delta t = 0.001$ was also used which did not capture wave phenomena accurately, as seen in Figure 6.3. The results obtained with both adaptive time step methods and the fully adaptive methods are identical.

The pressure and temperature in the tank during the entire transient are illustrated in Figures 6.4 and 6.5 respectively. Results are given for a constant time step as well as the fully adaptive algorithm. The time step of $\Delta t = 0.001$ was used which gives

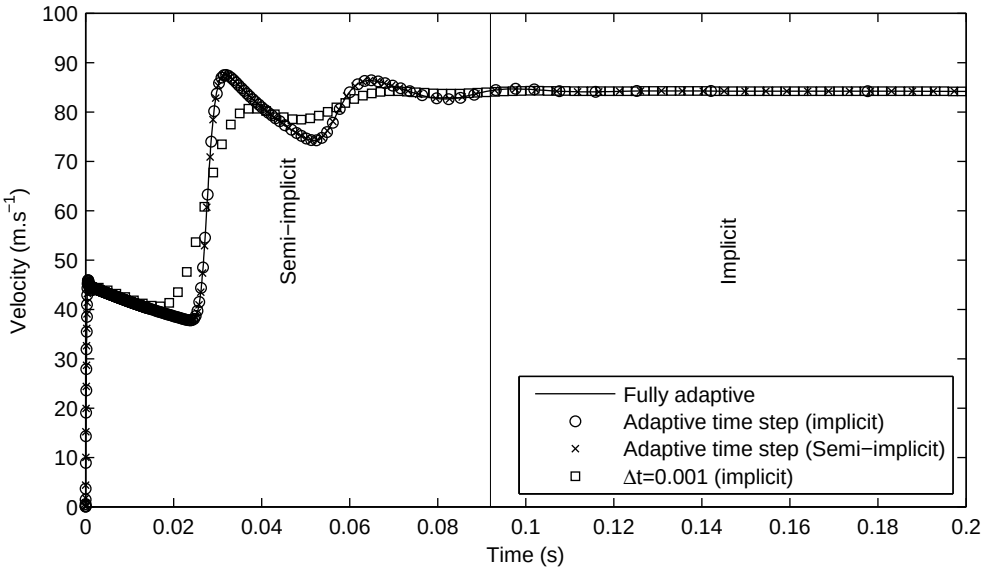



**Figure 6.3:** The solution of the velocity at the exit of the pipe at the onset of the fast transient due to the instantaneous valve opening as calculated with various methods.

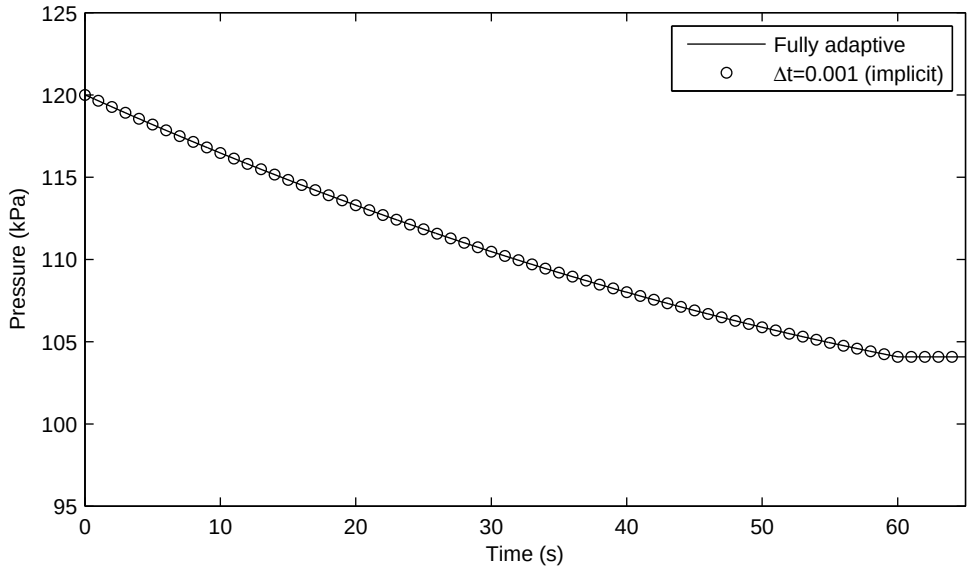



**Figure 6.4:** The solution of the pressure in the tank as calculated with the adaptive and a constant time step methods.

sufficient accuracy for the slow transient.

At $t = 60$ when the valve was closed instantaneously the second fast transient took place. The pressure and temperature at the middle of the pipe during the fast transient are illustrated in Figures 6.6 and 6.7 respectively. Results obtained with various implicit methods as well as the fully adaptive algorithm are given. The fully adaptive algorithm

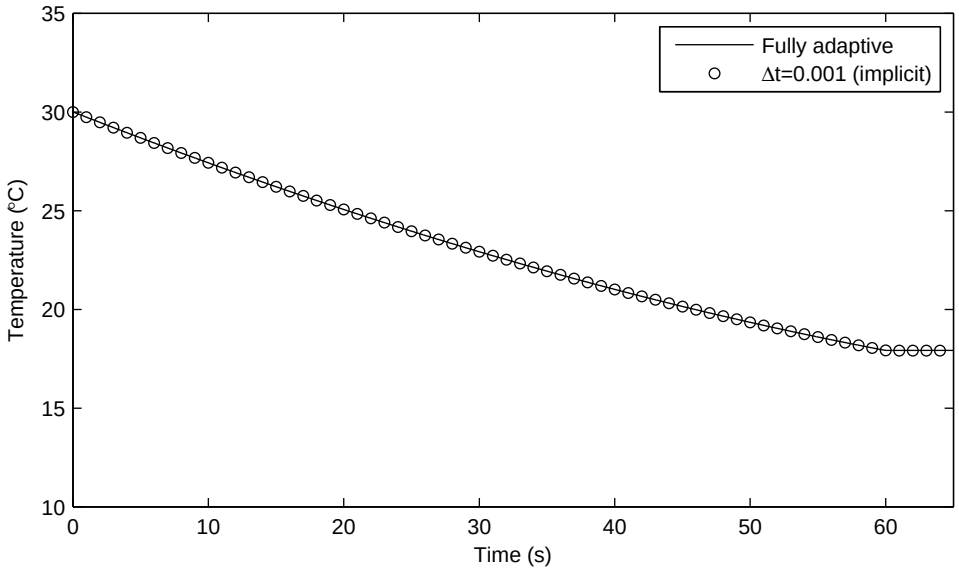



**Figure 6.5:** The solution of the temperature in the tank as calculated with the adaptive and a constant time step methods.

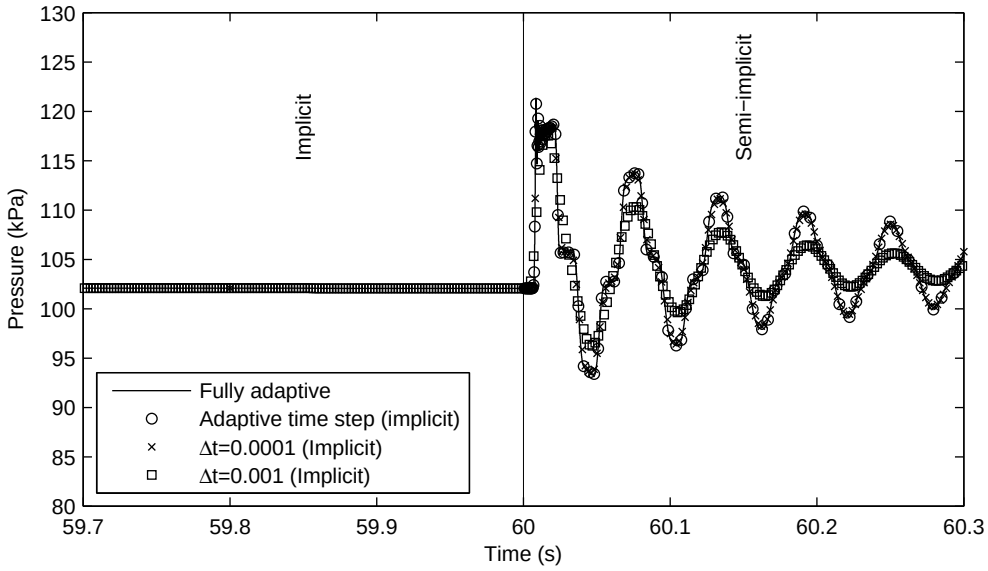



**Figure 6.6:** The solution of the pressure at the middle of the pipe at the onset of the fast transient due to the instantaneous valve closure as calculated with various methods.

switched from the implicit to the semi-implicit method at $t = 60$ and the algorithm selection is illustrated in Figures 6.6 and 6.7. The step size of $\Delta t = 0.001$ provides insufficient accuracy, while all the other methods provide acceptable accuracy.

Figure 6.8 illustrates the time step sizes used throughout the entire transient as determined by the fully adaptive algorithm. The regions where the semi-implicit and

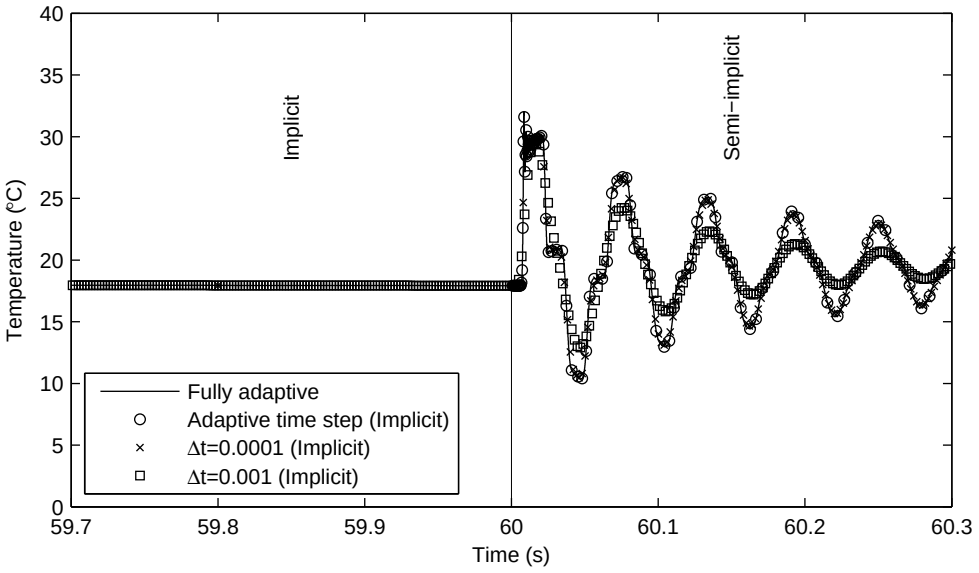



**Figure 6.7:** The solution of the temperature at the middle of the pipe at the onset of the fast transient due to the instantaneous valve closure as calculated with various methods.

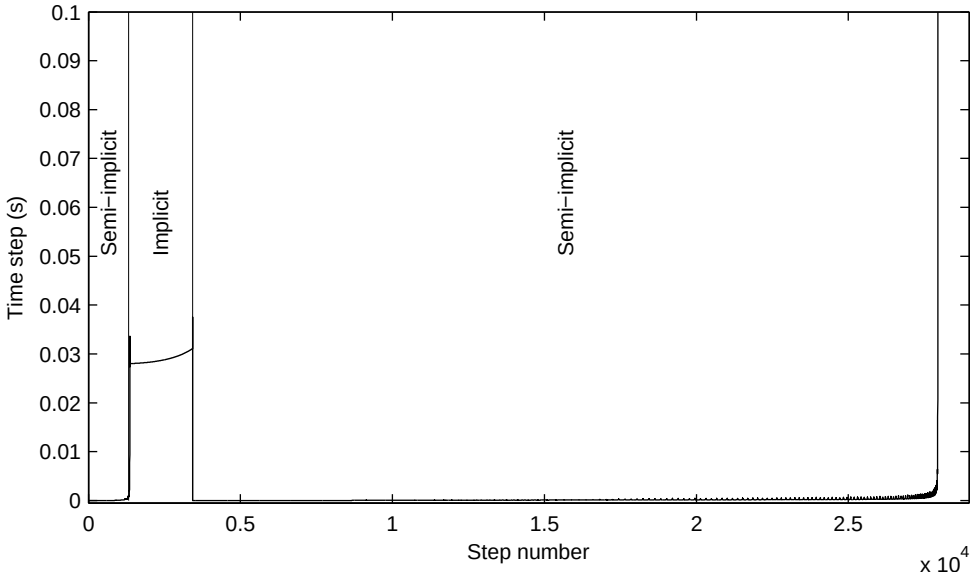



**Figure 6.8:** Time step size vs. step number for the entire transient as employed in the fully adaptive algorithm.

implicit methods were used are also given. The first region where the semi-implicit method was used corresponds to the initial fast transient. The second region, where the algorithm switched to the implicit method, corresponds to the slow transient associated with storage. The third region, corresponding to the semi-implicit method, is the second fast transient associated with the valve closure at $t = 60$.

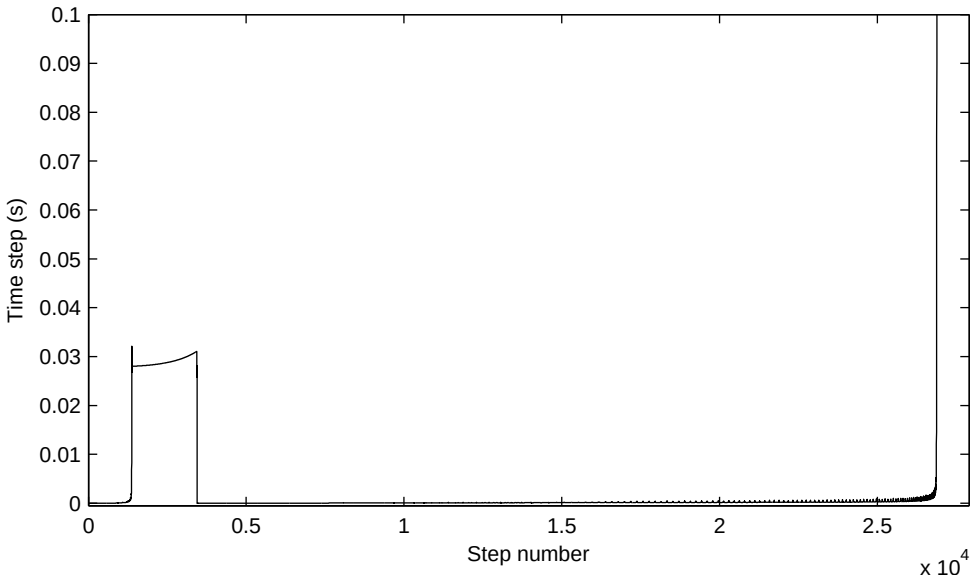



**Figure 6.9:** Time step size vs. step number for the entire transient as employed in the implicit adaptive time algorithm.

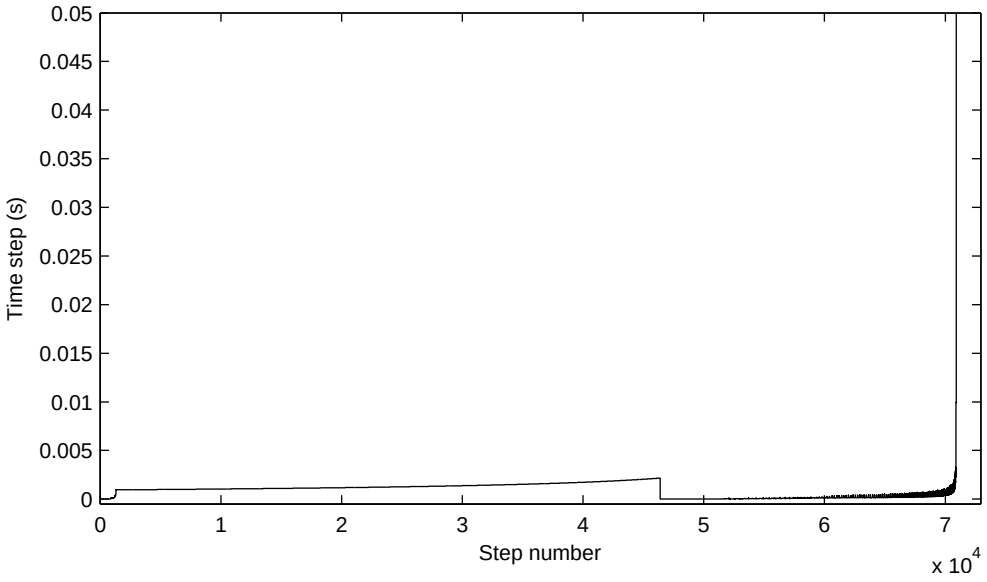



**Figure 6.10:** Time step size vs. step number for the entire transient as employed in the semi-implicit adaptive time algorithm.

Figure 6.9 illustrates the time step sizes used throughout the entire transient as determined by the implicit adaptive time algorithm. The result is nearly identical to that of the fully adaptive algorithm. However, the implicit method is used throughout and the computational cost is higher. Figure 6.10 illustrates the time step sizes used throughout the entire transient as determined by the semi-implicit adaptive time algo-

**Table 6.3:** Computational cost for various methods for the first test problem.

**(a)** Summary and comparison of various methods

| Method | Time steps | Rejected steps | Comput. time (s) |
|---|---|---|---|
| Fully adaptive | 27952 | 151 | 4.344 |
| Adaptive time step (semi-implicit) | 70921 | 151 | 10.156 |
| Adaptive time step (implicit) | 26887 | 152 | 8.281 |
| $\Delta t = 0.0001$ (semi-implicit) | 650000 | – | 83.203 |
| $\Delta t = 0.001$ (semi-implicit) | 65000 | – | 8.32 |
| $\Delta t = 0.001$ (implicit) | 65000 | – | 18.89 |

**(b)** Summary of the fully adaptive algorithm

| Method | Time steps | Rejected steps |
|---|---|---|
| Semi-implicit | 25848 | 149 |
| Implicit | 2104 | 2 |
| Total | 27952 | 151 |

rithm. As can be seen, during the slow transient, the method selected a much smaller time step than the implicit method. This is since the method is driven by stability in this region, whereas a much larger time step would have been adequate for an accurate solution. As seen in the figure, the semi-implicit method required approximately three times the number of time steps of the implicit method. However, the implicit method is more expensive than the semi-implicit method. This is where the fully adaptive method excels.

The computational costs associated with various methods are given in Table 6.3. The transient was solved up to $t = 65$ s, approximately the end of the transient. All simulations were done on a 2.67 GHz Intel® Core™2 Duo processor while a generic direct matrix inversion was adopted. The computational time of the fully adaptive algorithm was 4.344 s. The computational times of the adaptive time step methods for the semi-implicit and fully implicit methods are 10.156 s and 8.281 s respectively. A constant time step of $\Delta t = 0.0001$, which gave acceptable accuracy for the entire solution, required 83.203 s with the semi-implicit method. A constant time step of

$\Delta t = 0.001$, which gave unacceptable accuracy, required 8.32 s with the semi-implicit and 18.89 with the implicit method. The fully adaptive method is nearly twice as fast as the second fastest method which gave accurate results, i.e. the implicit method with adaptive time steps. The gain of using both semi-implicit and implicit methods in a fully adaptive algorithm is clearly apparent.

### 6.3.2 A simple branching network with tanks

The second test problem under consideration is also the one used in Chapter 5 and is shown in Figure 6.11. Initially, the flow is at rest, with initial conditions other than ambient conditions, which are summarised in Table 6.4. The pipe and tank characteristics are summarised in Table 6.5. The fluid is air, with $\gamma = 1.4$ and $R = 287$ J·kg$^{-1}$·K$^{-1}$. A spatial increment of $\Delta x = 0.1$ was used for all pipes. $\alpha = 1$ was again used throughout for the implicit method. For the adaptive time step algorithm, tolerances of $Atol = Rtol = 1 \times 10^{-4}$ were chosen as previously.

The same boundary conditions were again used as in Chapter 5. At $t = 0$, the valve was opened instantaneously which induced rapid transients associated with wave phenomena for a short duration. Wave phenomena were damped out eventually and the transient consisted of the slow storage effects associated with the volumes of the tanks and pipes. At $t = 60$, the valve was closed instantaneously which again induced

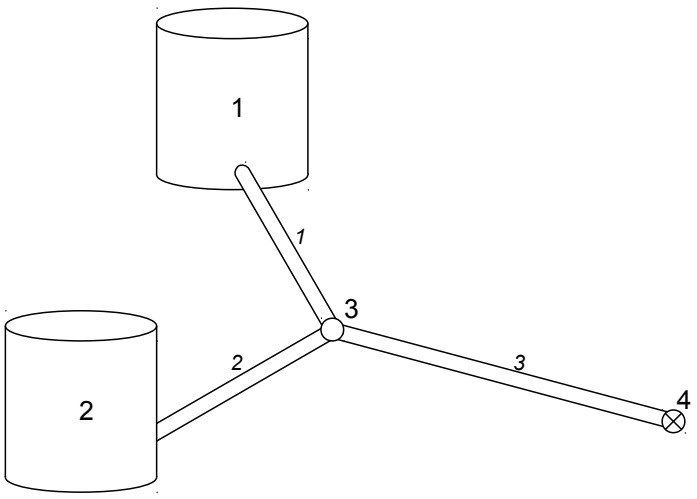



**Figure 6.11:** A simple branching network consisting of two tanks and three pipes.

**Table 6.4:** Initial conditions for the problem in Figure 6.11.

| | Pressure (kPa) | Temperature (°C) |
|---|---|---|
| Pipes and tanks | 120 | 30 |
| Ambient | 100 | 20 |

**Table 6.5:** Pipe and tank characteristics for the problem in Figure 6.11.

**(a)** Tanks

| Tank | Volume ($m^3$) |
|---|---|
| 1 | 10 |
| 2 | 5 |

**(b)** Pipes

| Pipe | Diameter (m) | Length (m) | Friction factor |
|---|---|---|---|
| 1 | 0.02 | 5 | 0.02 |
| 2 | 0.04 | 2 | 0.02 |
| 3 | 0.05 | 3 | 0.02 |

rapid wave phenomena. At the end of the wave phenomena, the fluid stabilised until a uniform state was reached.

Figure 6.12 illustrates the velocity at the exit of the pipe due to the instantaneous valve opening at $t = 0$. Results given are again obtained with a constant time step method, adaptive time step with the semi-implicit method, adaptive time step with the implicit method as well as the fully adaptive algorithm. The fully adaptive method started off with the semi-implicit method and switched to the fully implicit method at $t = 0.107$ as indicated in Figure 6.12. The results obtained with both adaptive time step methods and the fully adaptive methods are identical.

The pressure and temperature in the tanks during the entire transient are illustrated in Figures 6.13 and 6.14 respectively. The faster transient associated with tank 2 and slower transient associated with tank 1 can be clearly seen once again. At $t = 60$ the two tanks stabilised until a uniform pressure was reached. Results are again given for the fully adaptive algorithm as well as constant time step of $\Delta t = 0.001$.

The pressure and temperature at the junction (node 3) during the onset of the fast transient at $t = 60$ s are illustrated in Figures 6.15 and 6.16 respectively. Results given are obtained with various implicit methods as well as the fully adaptive algorithm. The methods selected by the fully adaptive algorithm are once again illustrated and exactly as expected. The step size of $\Delta t = 0.001$ provides insufficient accuracy while the results

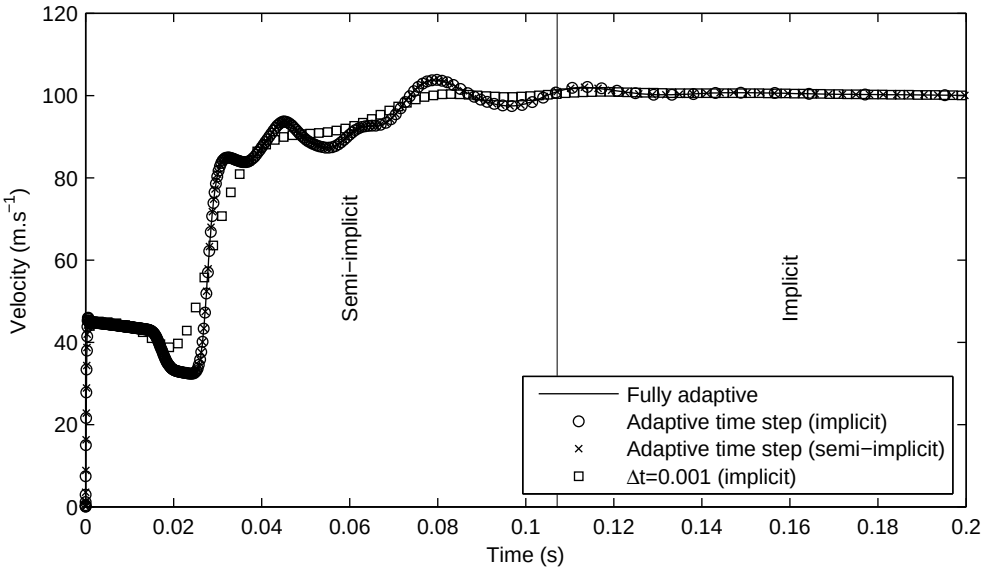



**Figure 6.12:** The solution of the velocity at the exit of the pipe at the onset of the fast transient due to the instantaneous valve opening as calculated with various methods.

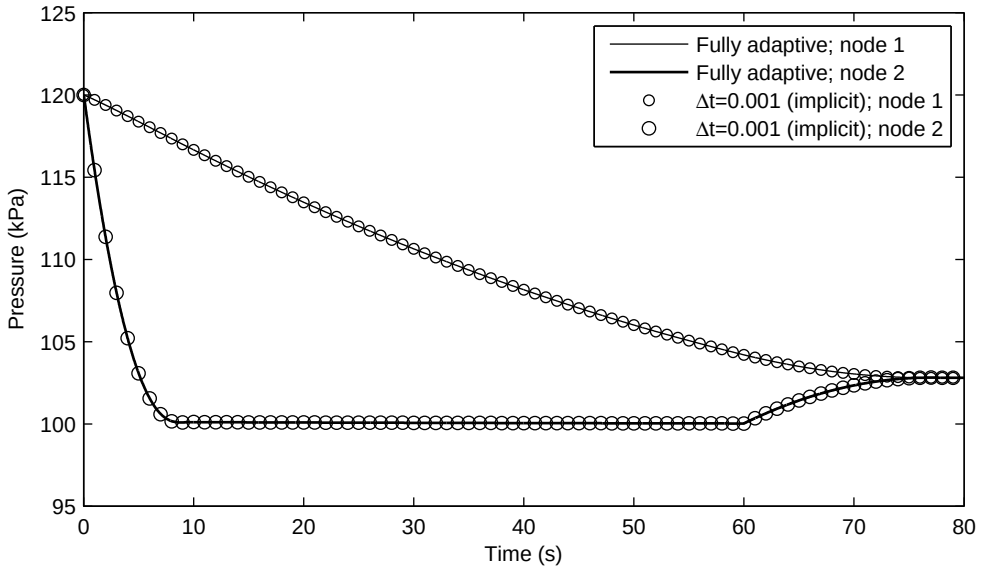



**Figure 6.13:** The solution of the pressure in the tanks as calculated with various methods.

of $\Delta t = 0.0001$ and the adaptive time step algorithms are sufficiently accurate.

A somewhat larger picture of the switching procedure is given in Figure 6.17. Figure 6.17 illustrates the velocity at the middle of pipe 1 from $t = 50$ s onwards. During the slow transient, the fully adaptive algorithm switched to the implicit method, which was illustrated in Figure 6.12. At the onset of the second fast transient, i.e. the instantaneous valve closure at $t = 60$ s, the algorithm switched to the computationally less ex-

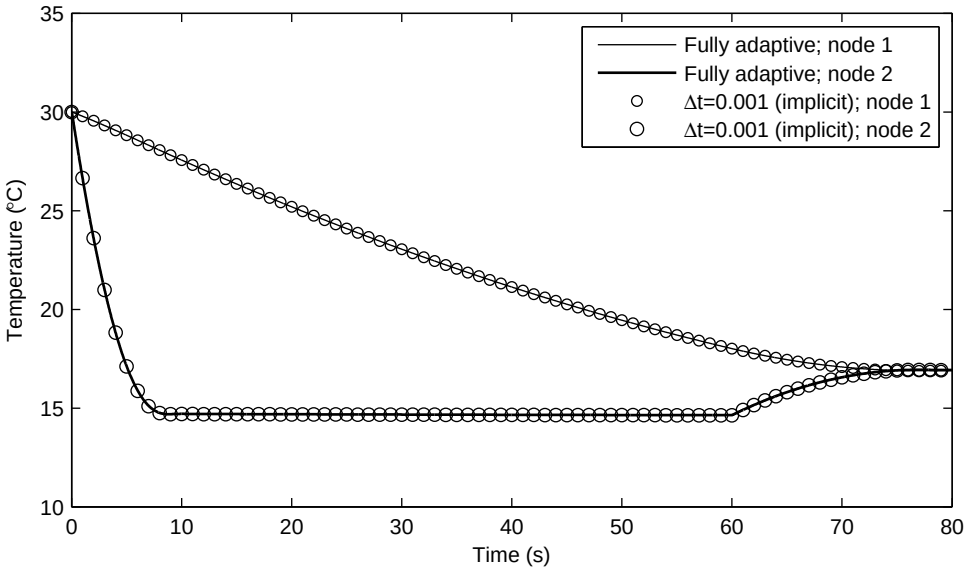



**Figure 6.14:** The solution of the temperature in the tanks as calculated with various methods.

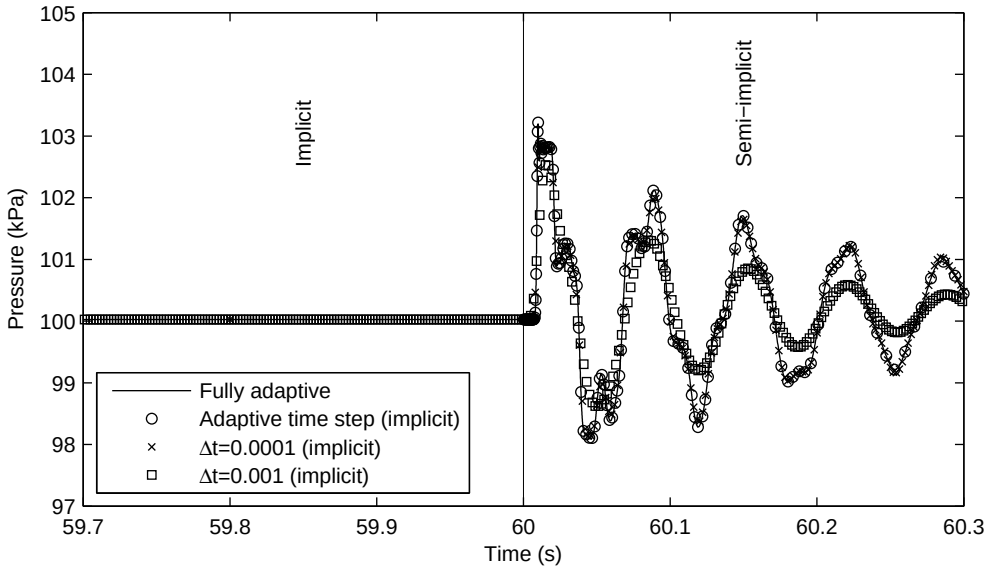



**Figure 6.15:** The solution of the pressure at the junction (node 3) of the pipe at the onset of the fast transient due to the instantaneous valve closure as calculated with various methods.

pensive semi-implicit method. Once the fast transient subsided, the algorithm switched back to the implicit method. Finally, at $t = 76.571$ s, the algorithm switched to the semi-implicit method. Although this is a slow transient, the semi-implicit method is more efficient at this stage, since the very low velocities allow a very large time step

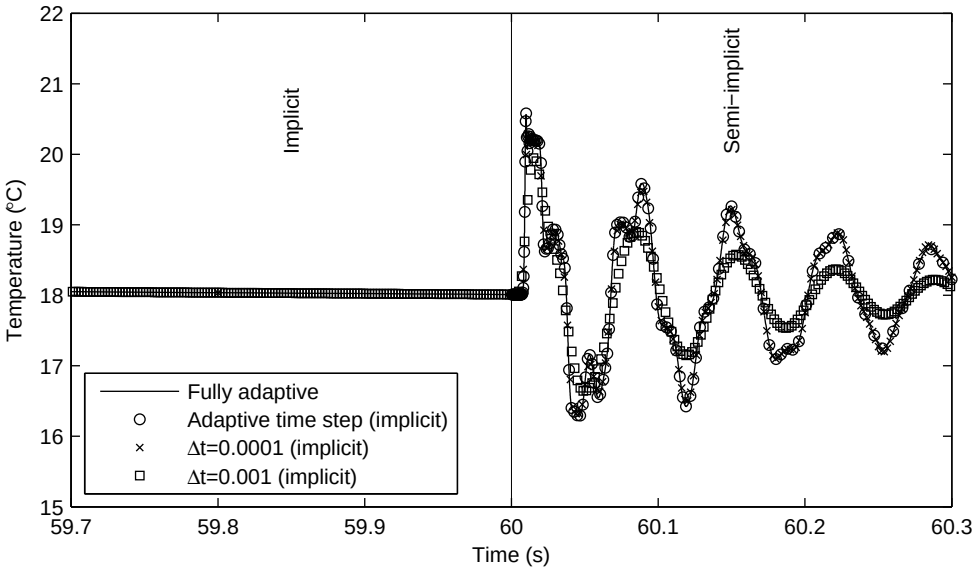



**Figure 6.16:** The solution of the temperature at the junction (node 3) of the pipe at the onset of the fast transient due to the instantaneous valve closure as calculated with various methods.

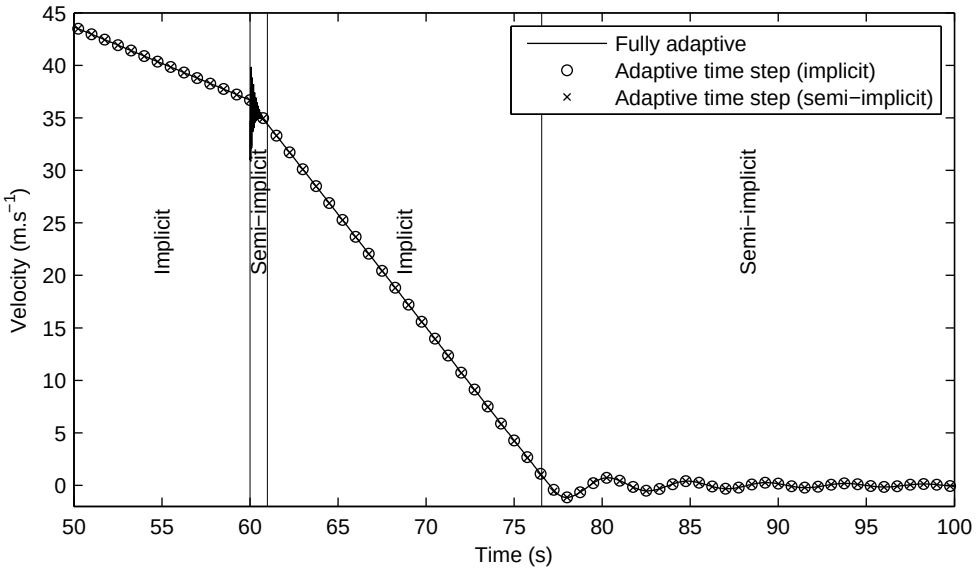



**Figure 6.17:** The solution of the velocity at the middle of pipe 1 from $t = 50$ s onwards. The methods selected for the time integration is illustrated.

from a stability point of view, and the semi-implicit method is driven by accuracy and is hence the method of choice.

Figure 6.18 illustrates the time step sizes used throughout the entire transient as determined by the fully adaptive algorithm. The regions where the semi-implicit and

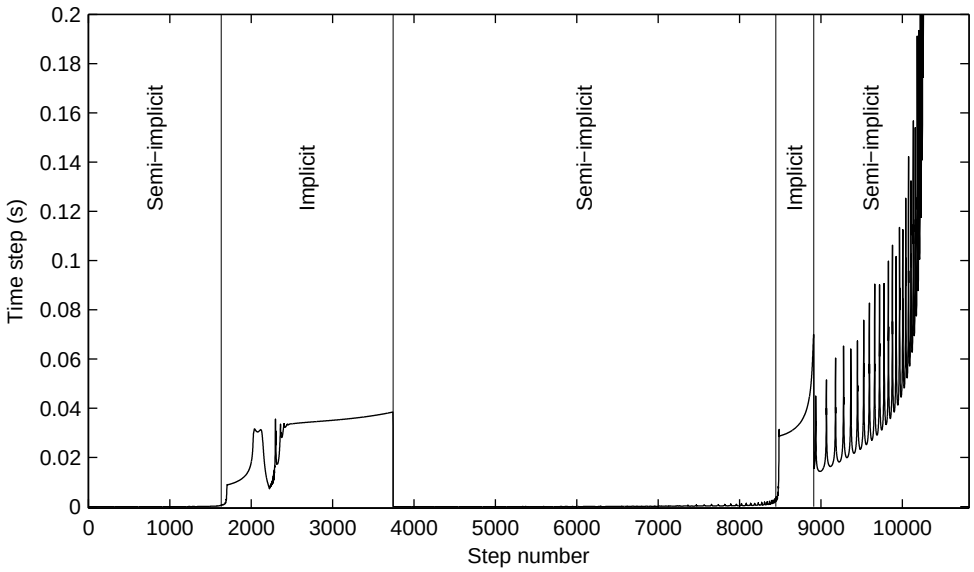



**Figure 6.18:** Time step size vs. step number for the entire transient

implicit methods were used are again given and is self explanatory from the previous discussion.

The computational costs are given in Table 6.6 for the various methods. The transient was solved up to $t = 80$ s. The computational time of the fully adaptive algorithm was 7.5 s. The computational times of the adaptive time step methods for the semi-implicit and fully implicit methods were 31.282 s and 13.343 s respectively. A constant time step of $\Delta t = 0.0001$ required 423.375 s with the semi-implicit method. A constant time step of $\Delta t = 0.001$, which gave unacceptable accuracy, required 42.338 s with the semi-implicit and 115.406 with the implicit method. The gain of using both semi-implicit and implicit methods in a fully adaptive algorithm is apparent. The fully adaptive method is again nearly twice as fast as the second fastest method which gave accurate results, which is again the implicit method with adaptive time steps.

## 6.4 Conclusion

This chapter built on the previous chapter by introducing an algorithm which, in addition to time step selection, selects the most efficient algorithm for time integration. Once a time step size is determined by the adaptive time step algorithm, an appropriate algorithm can be selected in a straightforward manner if sharp stability criteria exist.

The fully adaptive algorithm in this chapter utilises the fully implicit method based

**Table 6.6:** Computational cost for various methods for the second test problem.

**(a)** Summary and comparison of various methods

| Method | Time steps | Rejected steps | Comput. time (s) |
|---|---|---|---|
| Fully adaptive | 9099 | 45 | 7.5 |
| Adaptive time step (semi-implicit) | 56066 | 34 | 31.282 |
| Adaptive time step (implicit) | 9046 | 46 | 13.343 |
| $\Delta t = 0.0001$ (semi-implicit) | 800000 | – | 423.375 |
| $\Delta t = 0.001$ (semi-implicit) | 80000 | – | 42.338 |
| $\Delta t = 0.001$ (implicit) | 80000 | – | 115.406 |

**(b)** Summary of the fully adaptive algorithm

| Method | Time steps | Rejected steps |
|---|---|---|
| Semi-implicit | 2592 | 9 |
| Implicit | 6507 | 36 |
| Total | 9099 | 45 |

on the $\alpha$-scheme as well as the semi-implicit method, a method which only treats acoustic terms implicitly and is subjected to stability criteria based on kinematic terms. Both methods are based on static pressure formulations, although not limited to these methods. Adding explicit methods to the available methods of the fully adaptive algorithm will give rise to additional savings. However, for the sake of simplicity, this was not done.

The two test problems of Chapter 5 were once again used for demonstration purposes. The fully adaptive algorithm correctly selects the appropriate method as determined by accuracy considerations. By automatically selecting appropriate methods for the different transient phenomena resulted in significant savings in computational cost. For both test problems, the fully adaptive algorithm reduced the cost by nearly half of that of the most efficient adaptive time step method for the implicit method. It can be concluded that a fully adaptive algorithm, i.e. an automatic algorithm selection routine in addition to adaptive time stepping, greatly improves the efficiency of the accurate simulation of transient flow in pipe networks.

# Chapter 7

# Conclusion and recommendations

## 7.1 Overview

This study focused on two aspects encountered in the field of transient simulation of flow in pipelines and networks. The first focus was on the different formulations of conservation of momentum, i.e. formulations based on static or total pressures. The second focus was on the adaptive control of numerical methods for efficient temporal integration.

Static and total pressure formulations of conservation of momentum have been widely applied in pipe network simulation models. The total pressure formulations are convenient in that they eliminate the difficulties associated with the calculation of the convective terms and pipe junctions are treated in a straightforward manner based on total pressure losses. However, the different formulations of total pressure for compressible and incompressible flow require different formulations of the momentum conservation equation. This thesis proposed and applied a unified total pressure formulation and associated numerical methods. Comparisons were done with traditional static pressure based methods.

The different time scales that are involved in the simulation of transient flow in pipe networks are problematic when conventional fixed time step methods are used for time integration. Time scales associated with acoustic and kinematic wave phenomena as well as storage effects differ greatly. To accommodate all these effects efficiently, adaptive algorithms were developed and presented. Comparisons were done with conventional fixed time step methods.

## 7.2 Conclusions

### Total pressure method

A unified 1D total pressure formulation for conservation of momentum was presented which is valid for all homogeneous fluids and convenient for the simulation of pipe networks. A non-iterative total pressure solution algorithm was presented for this formulation. Consistency was demonstrated with test problems such as selected Riemann problems and a sudden valve closure problem from the literature. The non-iterative method was found to be sufficiently stable for subsonic flow conditions. The convergence towards a steady state solution of the total pressure method was superior to that of static pressure methods. It can be concluded that methods based on total pressure formulations are convenient in network simulation codes for general application and a generic formulation for all fluids can be implemented.

### Adaptive algorithms

This thesis presented a computationally inexpensive adaptive time step algorithm which can be readily used in conjunction with all the common first order methods used in pipe networks. The $\alpha$-scheme was used, and two test problems were selected to demonstrate the efficiency and savings obtained with this procedure. The adaptive time step algorithm correctly selected appropriate time steps for all phenomena and significant computational savings are observed for accurate integration.

In addition to the adaptive time stepping, a procedure was implemented which automatically selects the appropriate time-wise integration method. The resulting algorithm is a fully adaptive algorithm, which utilises the fully implicit method based on the $\alpha$-scheme as well as a semi-implicit method, a method which only treats acoustic terms implicitly. The two test problems were once again used to demonstrate the efficiency and savings. The fully adaptive algorithm correctly selects appropriate methods for all phenomena and significantly further computational savings are observed.

## 7.3 Recommendations

1. Total pressure method

(a) Although the total pressure formulation has several advantages, possible issues need to be addressed. A potential problem can arise associated with accurate shock capturing especially at supersonic speeds, which is attributed to the non-conservative formulation. Although the presence of high speed supersonic shocks seems limited for transient flow in pipe networks, a thorough study needs to be undertaken to investigate the importance of this aspect.

(b) A similar problem will arise in the resolution of stationary shocks. An example is a converging-diverging nozzle. For this case, the total pressure formulation will predict a constant total pressure throughout, hence failing to capture the stationary shock correctly. This is in contrast with a conservative method, which will converge to the correct shock location. The implementation of shock relations in conjunction with the total pressure method may remedy this, although this reduces the robustness of this method. An investigation into the importance and presence of such phenomena in pipe networks and the treatment thereof is needed. For typical pipe networks, however, stationary shocks such as these are unlikely and this aspect was beyond the scope of this study.

(c) A more thorough integration of pipe junctions and losses as well as other components with the total pressure method (and static pressure methods, if preferred) need to be addressed.

2. Comparison of methods

(a) In order to do a more realistic and complete comparison of the computational efficiency of methods, iterative and sparse matrix techniques have to be employed. In particular, in the case of FIBS, generic dense methods for matrix computations and storage were used which were significantly more expensive than other methods, and no attempt was made to reach a general conclusion.

3. Adaptive algorithms

(a) In addition to the implicit and semi-implicit methods, the inclusion of fully explicit methods will improve the efficiency of the fully adaptive algorithm,

especially in the vicinity of wave phenomena. Classical explicit methods based on a collocated grid will, however, require interpolation to and from a staggered grid.

(b) The extension to higher order methods may be valuable. This will not be a trivial task in the case of the semi-implicit method which is attributed to its hybrid nature. Furthermore, if the PISO based non-iterative method is used in conjunction with higher order methods, this will require additional iterations as opposed to the minimalistic version used in this thesis.

(c) The error tolerances used in the adaptive time step algorithm can be based on spatial errors such that the temporal and spatial errors are of equal magnitude. This is to ensure that the numerical solution is not dominated by temporal discretisation errors, nor result in a temporal integration which is unnecessarily more accurate than the spatial discretisation. In addition, the implementation of spatial adaptivity will further improve efficiency of the adaptive algorithm.

(d) Iterative and sparse matrix techniques will be more efficient in conjunction with the adaptive time step algorithms.

## 7.4 Acknowledgement

This work is based upon research supported by the South African Research Chairs Initiative of the Department of Science and Technology and National Research Foundation.

**Disclaimer** Any opinion, findings and conclusions or recommendations expressed in this material are those of the author(s) and therefore the NRF and DST do not accept any liability with regard thereto.

# Bibliography


Abbaspour, M. & Chapman, K. S. (2008), 'Nonisothermal transient flow in natural gas pipeline', *Journal of Applied Mechanics* **75**.

Afshar, M. H. & Rohani, M. (2008), Exploring the versatility of the implicit method of characteristics (MOC) for transient simulation of pipeline systems, *in* 'Twelfth International Water Technology Conference, IWTC12 2008, Alexandria, Egypt', pp. 703–715.

Altman, T. & Boulos, P. F. (1995), 'Convergence of newton method in nonlinear network analysis', *Mathematical and Computer Modelling* **21**(4), 35–41.

Arfaie, M. & Anderson, A. (1991), 'Implicit finite-differences for unsteady pipe flow', *Mathematical engineering in industry* **3**(2), 133–151.

Banda, M. K., Herty, M. & Klar, A. (2005), 'Gas flow in pipeline networks', *AIMS* .

Bassett, M. D., Winterbone, D. E. & Pearson, R. J. (2001), 'Calculation of steady flow pressure loss coefficients for pipe junctions', **215**, 861–881.

Behbahani-Nejad, M. & Bagheri, A. (2008), A matlab simulink library for tranient flow simulation of gas networks, *in* 'Proceedings of World Academy of Science, Engineering and Technology', Vol. 33, pp. 153–159.

Benson, R. S., Woollatt, D. & Woods, W. A. (1963-4), 'Unsteady flow in simple branch system', *Proc. Instn Mech. Engrs* **178**(10), 24–49.

Berzins, M. (1995), 'Temporal error control for convection-dominated equations in two space dimensions', *SIAM J. Sci. Comput.* **16**, 558–581.

Bingharn, J. F. & Blair, G. P. (1985), 'An improved branched pipe model for multicylinder automotive engine calculations', *Proc. Instn Mech. Engrs* **199**(D1).

Botha, F. (2003), An explicit method for the analysis of transient compressible flow in pipe networks, Master's thesis, School of Mechanical and Materials Engineering, Potchefstroom University for Christian Higher Education.

Casulli, V. & Greenspan, D. (1984), 'Pressure method for the numerical solution of transient, compressible fluid flows', *International Journal for Numerical Methods in Fluids* **4**, 1001–1012.

Chaczykowski, M. (2009), 'Transient flow in natural gas pipelines - the effect of pipeline thermal model', *Applied Mathematical Modelling* **34**, 1051–1067.

Chae, K. S., Lee, K. T., Hwangc, C. J. & Lee, D. J. (2006), 'Formulation and validation of boundary conditions at a branched junction for nonlinear waves', *Journal of Sound and Vibration* **295**, 13–27.

Chang, S. (2002), A program development for unsteady gas flow analysis in complex pipe networks, Technical report, R&D Center, Korea Gas Corporation.

Chaudhry, M. H. (1979), *Applied Hydraulic Transients*, Van Nostrand, New York.

Chua, T. S. & Dew, P. M. (1984), 'The design of a variable-step integrator for the simulation of gas transmission networks', *International journal for numerical methods in engineering* **20**, 1797–1813.

Corberan, J. M. (1992), 'A new constant pressure model for $N$-branch junctions', *Journal of Automobile Engineering, Proceedings of the Institution of Mechanical Engineers, Part D* **206**, 117–123.

Cross, H. (1936), Analysis of flow in networks of conduits or conductors, Technical report, University of Illinois. Engineering Experimental Station. Bulletin no. 286.

Dukhovnaya, Y. & Adewumi, M. A. (2000), 'Simulation of non-isothermal transients in gas/condesate pipelines using tvd scheme', *Powder Technology* **112**, 163–171.

Emara-Shabaik, H. E. (2004), 'Simulation of transient flow in pipelines for computer-based operations monitoring', *International Journal for Numerical Methods in Fluids* **44**, 257–275.

Engl, G. (1996), 'The modeling and numerical simulation of gas flow networks', *Numerische Mathematik* **72**, 349–366.

Fletcher, C. A. J. (1991), *Computational Techniques for Fluid Dynamics*, Vol. 1, second edition edn, Springer-Verlag.

Garland, W. J. (1998), 'Nuclear reactor process systems: Thermal hydraulic analysis', Department of Engineering Physics, McMaster University, Canada.

Gresho, P. M., Lee, R. L. & Sani, R. L. (1980), On the time-dependent solution of the incompressible Navier-Stokes equations in two and three dimensions, *in* 'Recent Advances in Numerical Methods in Fluids', Pineridge Press.

Greyvenstein, G. P. (2002), 'An implicit method for the analysis of transient flows in pipe networks', *International Journal for Numerical Methods in Engineering* **53**, 1127–1143.

Greyvenstein, G. P. & Laurie, D. P. (1994), 'A segregated CFD approach to pipe network analysis', *International Journal for Numerical Methods in Engineering* **37**, 3685–3705.

Guy, J. J. (1967), 'Computation of unsteady gas flow in pipe networks', *Institute of Chemical Engineers Symposium Series* (23), 139–145.

Hairer, E., Norsett, S. & Wanner, G. (1987), *Solving Ordinary Differential Equations I - Nonstiff Problems*, Springer, Berlin.

Hairer, E. & Wanner, G. (1991), *Solving Ordinary Differential Equations II - Stiff and Differential Algebraic Problems*, Springer, Berlin.

Herran-Gonzalez, A., De La Cruz, J. M., Andres-Toro, B. D. & Risco-Martin, J. L. (2009), 'Modeling and simulation of a gas distribution pipeline network', *Applied Mathematical Modelling* **33**, 1584–1600.

Hong, S. W. & Kim, C. (2009), 'A new finite volume method on junction coupling and boundary treatment for flow network system analyses', *International Journal for Numerical Methods in Fluids* **65**, 707–742.

Hornëll, K. & Lötstedt, P. (2001), 'Time step selection for shock problems', *Communications in Numerical Methods in Engineering* **17**, 477–484.

Islam, M. R. & Chaudhry, M. H. (1998), 'Modeling of constituent transport in unsteady flows in pipe networks', *Journal of Hydraulic Engineering* pp. 1115–1124.

Issa, R. I. (1986), 'Solution of te implicitly discretised fluid flow equaitons by operator-splitting', *Journal of Computational Physics* **62**, 40–65.

Issa, R. I. & Spalding, D. B. (1972), 'Unsteady one-dimensional compressible frictional flow with heat transfer', *Mechanical Engineering Science* **14**(6).

John, V. & Rang, J. (2010), 'Adaptive time step control for the incompressible Navier-Stokes equations', *Computation methods in applied mechanics and engineering* **199**, 514–524.

Kaps, P., Poon, S. W. H. & Bui, T. D. (1985), 'Rosenbrock methods for stiff odes: A comparison of richardson extrapolation and embedding technique', *Computing* **34**, 17–40.

Kay, D. A., Gresho, P. M., Griffiths, D. F. & Silvester, D. J. (2010), 'Adaptive time-stepping for incompressible flow part ii: Navier-Stokes equations', *SIAM J. Sci. Comput.* **32**(1), 111–128.

Kelkar, K. M. & Patankar, S. V. (2003), Analysis and design of liquid-cooling systems using flow network modeling (FNM), *in* 'Proceedings of IPACK03'.

Kesavan, H. K. & Chandrashekar, M. (1972), 'Graph-theoretical models for pipe network analysis', *Journal of Hydraulic Engineering* **98**.

Kim, Y. I. (2008), Advanced Numerical and Experimental Transient Modelling of Water and Gas Pipeline Flows Incorporating Distributed and Local Effects, PhD thesis, School of Civil, Environmental and Mining Engineering, The University of Adelaide, Australia.

Kiuchi, T. (1994), 'An implicit method for transient gas flows in pipe networks', *The international journal of heat and fluid flow* **15**, 378–383.

Kwatra, N., Su, J., Grétarsson, J. T. & Fedkiw, R. (2009), 'A method for avoiding the acoustic time step restriction in compressible flow', *Journal of Computational Physics* **228**, 4146–4161.

Laney, C. B. (1998), *Computational Gasdynamics*, Cambridge University Press.

Leung, R. (1997), Dynamic simulation of an utility boiler's natural-circulation circuit, Master's thesis, Department of Chemical and Materials Engineering, Edmonton, Alberta.

Leveque, R. J. (2002), *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press.

Lewandowski, A. (2000), New numerical methods for transient modeling of gas pipeline networks, Technical report, American Management Systems, Inc, Birmingham, Alabama.

Majumdar, A. (1999), Generalized fluid system simulation program (gfssp) version 3.0, Technical report, Sverdrup Technology Inc, MSFC Group.

Majumdar, A. K. & Ravindran, S. S. (2010), 'Fast, nonlinear network flow solvers for fluid and thermal transient analysis', *International Journal of Numerical Methods for Heat and Fluid Flow* **20**(6), 617–637.

Nicolet, C. (2007), Hydroacoustic modelling and numerical simulation of unsteady operation of hydroelectric systems, PhD thesis, À La Faculté Sciences Et Techniques De L'Ingénieur, École Polytechnique Fédérale De Lausanne.

Osiadacz, A. J. & Cahczykowski, M. (2001), 'Comparison of isothermal and non-isothermal pipeline gas flow models', *Chemical Engineering Journal* **81**, 41–51.

Prosperetti, A. & Tryggvason, G., eds (2007), *Computational Methods for Multiphase Flow*, Cambridge University Press.

Shampine, L. F., Gladwell, I. & Thompson, S. (2003), *Solving ODEs with MATLAB*, Cambridge University Press.

Städtke, H. (2006), *Gasdynamic Aspects of Two-phase Flow*, Wiley-CVH.

Suwan, K. & Anderson, A. (1992), 'Method of lines applied to hyperbolic fluid transient equations', *International Journal for Numerical Methods in Engineering* **33**, 1501–1511.

Tao, W. Q. & Ti, H. C. (1998), 'Transient analysis of gas pipeline network', *Chemical Engineering Journal;* **69**, 47–52.

Tentis, E., Margaris, D. & Papinikas, D. (2003), 'Transient gas flow simulation using an adaptive method of lines', *C. R. Mecanique* **331**, 481–487.

Thompson, K. W. (1987), 'Time dependent boundary conditions for hyperbolic systems', *Journal of Computational Physics* **68**, 1–24.

Thorley, A. R. D. & Tiley, C. H. (1987), 'Unsteady and transient flow of compressible fluids in pipelines–a review of theoretical and some experimental studies', *The International Journal of Heat and Fluid Flow* **8**(1).

Valli, A. M. P., Carey, G. F. & Coutinho, A. L. G. A. (2005), 'Control strategies for timestep selection in finite element simulation of incompressible flows and coupled reaction-convection-diffusion processes', *International Journal for Numerical Methods in Fluids* **47**, 201–231.

van der Heul, D. R., Vuik, C. & Wesseling, P. (2000), Efficient computation of flow with cavitation by compressible pressure correction, *in* 'European Congress on Computational Methods in Applied Sciences and Engineering'.

van der Heul, D., Vuik, C. & Wesseling, P. (2001), 'Stability analysis of segregated solution methods for compressible flow', *Applied Numerical Mathematics* **38**, 257–274.

van der Heul, D., Vuik, C. & Wesseling, P. (2003), 'A conservative pressure-correction method for flow at all speeds', *Computers & Fluids* **32**, 1113–1132.

Van Ravenswaay, J. P. (1998), An implicit method for transient pipe network analysis, Master's thesis, School of Mechanical and Materials Engineering, Potchefstroom University for Christian Higher Education.

Vande Wouwer, A., Saucez, P. & Schiesser, W. E., eds (2001), *Adaptive Method of Lines*, Chapman & Hall/CRC.

Walter, H., Ponweiser, K. & Linzer, W. (2000), The flow problem in heated tube-header-structures, *in* 'IMACS Symposium on Mathematical Modelling', Vol. 1.

Wesseling, P. (2001), *Principles of Computational Fluid Dynamics*, Springer Series in Computational Mathematics, Springer, Berlin Heidelberg.

William-Louis, M. J. P., Ould-El-Hadrami, A. & Tournier, C. (1998), 'On the calculation of the unsteady compressible flow through an N-branch junction', *Proc. Instn Mech. Engrs* **212**, 49–56.

Wood, D. J. & Rayes, A. G. (1981), 'Reliability of algorithms for pipe network analysis', *Journal of Hy* **107**, 1145–1161.

Wylie, E. B. & Streeter, V. L. (1993), *Fluid Transients in Systems*, Upper Saddle River, NJ, Prentice Hall.

Yoon, S. Y. & Yabe, T. (1999), 'The unified simulation for incompressible and compressible flow by the predictor-corrector scheme based on the CIP method', *Computer Physics Communications* **119**, 149–158.

Zucker, R. D. & Biblarz, O. (2002), *Fundamentals of Gas Dynamics*, John Wiley & Sons, Inc.

# Appendix A

# Additional methods

## A.1 Semi-implicit method

In this section, a semi-implicit temporal discretised method will be derived. The method is based on the implicit treatment of acoustic terms only, which results in an efficient solution algorithm. Many variations can be found in the literature based on this idea, see for example Prosperetti & Tryggvason (2007) and Wesseling (2001).

### Identification of acoustic terms

The terms in the Euler equations associated with acoustic waves are identified below and highlighted with an over-bar:

**Continuity**

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho \bar{u} A)}{A \partial x} = 0. \tag{A.1}$$

**Conservation of momentum**

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u \bar{u} A)}{A \partial x} + \frac{\partial \bar{p}}{\partial x} = -\frac{f \rho u |u|}{2D}. \tag{A.2}$$

**Conservation of energy**

$$\frac{\partial(\rho e_0)}{\partial t} + \frac{\partial(\rho e_0 \bar{u} A)}{A \partial x} + \frac{\partial(\bar{p} \bar{u} A)}{A \partial x} = \dot{q}. \tag{A.3}$$

With these terms identified, a numerical method can be developed which is semi-implicit by nature, i.e. which handles kinematic phenomena explicitly, while handling

acoustic phenomena implicitly. In applications where acoustic phenomena are unimportant, this is the ideal approach. If acoustic phenomena are of importance, explicit compressible schemes remain the methods of choice.

To improve the efficiency of the numerical algorithm, the momentum equation can be written in nonconservative form:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{1}{\rho}\frac{\partial \bar{p}}{\partial x} = -\frac{fu|u|}{2D}, \tag{A.4}$$

with the acoustic terms once again highlighted with an over-bar. The main advantage of this form as opposed to the form of (A.2) is that only pressure has to be treated implicitly, as opposed to pressure and convected momentum. This leads to a more efficient semi-implicit solution algorithm. If acoustic terms are treated implicitly, while kinematic terms are treated explicitly, a stability criterion exists based on the material Courant number:

$$\max|u|\frac{\Delta t}{\Delta x} \leq 1. \tag{A.5}$$

**A semi-implicit algorithm**

In this section, a semi-implicit algorithm is presented. The method is derived for a general equation of state, with pressure and internal energy chosen as input variables. Furthermore, for simplicity, kinetic energy is neglected in the derivation. The method, however, is not limited to this assumption.

Consider the semi-discrete form of the continuity equation:

$$\mathcal{V}_i\frac{\partial \rho_i}{\partial t} + (\rho u A)_{i+1/2} - (\rho u A)_{i-1/2} = 0. \tag{A.6}$$

For $\rho = \rho(p, e)$, this can be rewritten as

$$\mathcal{V}_i\left(\frac{\partial \rho_i}{\partial p_i}\frac{\partial p_i}{\partial t} + \frac{\partial \rho_i}{\partial e_i}\frac{\partial e_i}{\partial t}\right) + (\rho u A)_{i+1/2} - (\rho u A)_{i-1/2} = 0. \tag{A.7}$$

Similarly, consider the semi-discrete energy equation:

$$\mathcal{V}_i\frac{\partial\left(\rho_i e_i\right)}{\partial t} + \left[\left(\rho e + p\right)uA\right]_{i+1/2} - \left[\left(\rho e + p\right)uA\right]_{i-1/2} = \dot{Q}_i. \tag{A.8}$$

Rewrite this in a similar fashion for $\rho = \rho(p, e)$:

$$\mathcal{V}_i\left[\left(e_i\frac{\partial \rho_i}{\partial e_i} + \rho_i\right)\frac{\partial e_i}{\partial t} + e_i\frac{\partial \rho_i}{\partial p_i}\frac{\partial p_i}{\partial t}\right] + \left[\left(\rho e + p\right)uA\right]_{i+1/2} - \left[\left(\rho e + p\right)uA\right]_{i-1/2} = \dot{Q}_i. \tag{A.9}$$

Next eliminate $\frac{\partial e}{\partial t}$ by combining (A.7) and (A.9), which leads to

$$\mathcal{V}_i \frac{\partial \rho_i}{\partial p_i}\frac{\partial p_i}{\partial t} + \left[\left(\frac{e_i}{\rho_i}\frac{\partial \rho_i}{\partial e_i} + 1\right)\rho_{i+1/2} - \frac{1}{\rho_i}\frac{\partial \rho_i}{\partial e_i}\left(\rho_{i+1/2}e_{i+1/2} + \overline{p}_{i+1/2}\right)\right]\overline{u}_{i+1/2}A_{i+1/2}$$
$$- \left[\left(\frac{e_i}{\rho_i}\frac{\partial \rho_i}{\partial e_i} + 1\right)\rho_{i-1/2} - \left(\frac{1}{\rho_i}\frac{\partial \rho_i}{\partial e_i}\right)\left(\rho_{i-1/2}e_{i-1/2} + \overline{p}_{i-1/2}\right)\right]\overline{u}_{i-1/2}A_{i-1/2} = -\frac{1}{\rho_i}\frac{\partial \rho_i}{\partial e_i}Q_i, \tag{A.10}$$

where the over bar once again represents the acoustic terms. This equation is general and suitable for all flow speeds. It can be simplified for various cases. In case of perfect gas flows, $\frac{e_i}{\rho_i}\frac{\partial \rho_i}{\partial e_i} + 1 = 0$, while for constant density, the equation reduces to the constraint $(\rho u A)_{i+1/2} - (\rho u A)_{i-1/2} = 0$.

Although the pressure needs to be treated implicitly in this equation from a formal characteristic analysis, experience has shown that adequate results are obtained with an explicit, upwind treatment of pressure. This greatly reduces the complexity of the resulting algebraic equations. With the substitutions

$$a_{i,i+1/2} = \left(\frac{e_i}{\rho_i}\frac{\partial \rho_i}{\partial e_i} + 1\right)\rho_{i+1/2} - \frac{1}{\rho_i}\frac{\partial \rho_i}{\partial e_i}\rho_{i+1/2}h_{i+1/2},$$
$$a_{i,i-1/2} = \left(\frac{e_i}{\rho_i}\frac{\partial \rho_i}{\partial e_i} + 1\right)\rho_{i-1/2} - \frac{1}{\rho_i}\frac{\partial \rho_i}{\partial e_i}\rho_{i-1/2}h_{i-1/2},$$
$$b_i = \frac{\mathcal{V}_i}{\Delta t}\frac{\partial \rho_i}{\partial p_i}$$

and

$$d_i - \frac{1}{\rho_i}\frac{\partial \rho_i}{\partial e_i}Q_i$$

(A.10) can be rewritten as

$$b_i \Delta t \frac{\partial p_i}{\partial t} + \left(a_{i,i+1/2}u_{i+1/2} - a_{i,i-1/2}u_{i-1/2}\right) = d_i. \tag{A.11}$$

A semi-implicit temporal discretisation leads to

$$b_i \Delta t \frac{p_i^{n+1} - p_i^n}{\Delta t} + \left(a^n_{i,i+1/2}u^{n+1}_{i+1/2} - a^n_{i,i-1/2}u^{n+1}_{i-1/2}\right) = d_i^{n+1}. \tag{A.12}$$

where all scalars at cell faces have been discretised using the upwind formulation. Of course, other methods, such as a central scheme or the MUSCL strategy, can be implemented as well.

The discretised momentum equation (A.4) can be written as

$$\frac{u^{n+1}_{i+1/2} - u^n_{i+1/2}}{\Delta t} + u^n_{i+1/2}\frac{u^n_{i+1} - u^n_i}{\Delta x} + \frac{1}{\rho^n_{i+1/2}}\frac{p^{n+1}_{i+1} - p^{n+1}_i}{\Delta x} = -\frac{f^n_{i+1/2}|u^n_{i+1/2}|u^{n+1}_{i+1/2}}{2D_{i+1/2}}, \tag{A.13}$$

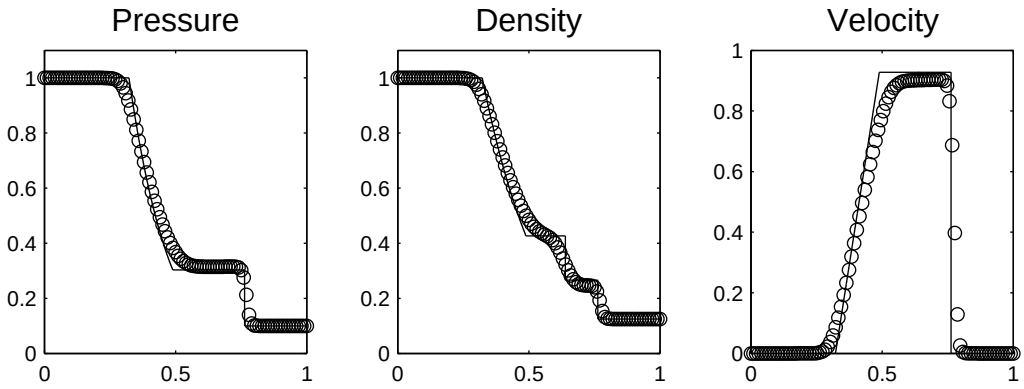



**Figure A.1:** Sod's shock tube problem for the semi-implicit method.

where $\rho_{i+1/2}$ can be obtained from a centred approximation and an upwind formulation can be used for the convected velocities $u_i$ and $u_{i+1}$. Now, $u_{i+1/2}^{n+1}$ can be rewritten explicitly in terms of pressure:

$$u_{i+1/2}^{n+1} = \left(1 + \frac{\Delta t f_{i+1/2}^n |u_{i+1/2}^n|}{2D_{i+1/2}}\right)^{-1} \left[\frac{\lambda}{\rho_{i+1/2}^n}(p_{i+1}^{n+1} - p_i^{n+1}) + u_{i+1/2}^n - \lambda u_{i+1/2}^n (u_{i+1}^n - u_i^n)\right], \tag{A.14}$$

which can be written as

$$u_{i+1/2}^{n+1} = c_{i+1/2}^1 (p_{i+1}^{n+1} - p_i^{n+1}) + c_{i+1/2}^0. \tag{A.15}$$

Similarly, for $u_{i-1/2}$ it follows that

$$u_{i-1/2}^{n+1} = c_{i-1/2}^1 (p_i^{n+1} - p_{i-1}^{n+1}) + c_{i-1/2}^0. \tag{A.16}$$

Substitution of (A.15) and (A.16) in (A.12) leads to the final tridiagonal set of equations

$$\begin{aligned}\left(b_i + a_{i,i+1/2} c_{i+1/2}^1 + a_{i,i-1/2} c_{i-1/2}^1\right) p_i^{n+1} - a_{i,i+1/2} c_{i+1/2}^1 p_{i+1}^{n+1} - a_{i,i-1/2} c_{i-1/2}^1 p_{i-1}^{n+1} \\ = b_i p_i^n + d_i^{n+1} - a_{i,i+1/2} c_{i+1/2}^0 - a_{i,i-1/2} c_{i-1/2}^0.\end{aligned} \tag{A.17}$$

With the pressure known, velocity can be calculated from conservation of momentum. For compressible flow, internal energy can be updated from continuity, while for incompressible flows, conservation of energy can be used. For validation purposes, the solution to Sod's shock tube problem as obtained with this method is given in Figure A.1.

## A.2 FIBS

The method described in this section was originally proposed by Garland (1998). The method is based on the following ODE scheme. Consider the ODE

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(\mathbf{y}). \tag{A.18}$$

The $\alpha$-scheme to solve the above equation is once again given by

$$\frac{\Delta\mathbf{y}}{\Delta t} = (1-\alpha)\mathbf{f}^n + \alpha\mathbf{f}^{n+1}. \tag{A.19}$$

Furthermore, linearization of $\mathbf{f}^{n+1}$ about the previous time denoted by $n$ gives

$$\mathbf{f}^{n+1} = \mathbf{f}^n + \mathbf{J}\Delta\mathbf{y}, \tag{A.20}$$

with $J$ the Jacobian matrix. Substitution of (A.20) into (A.19) leads to

$$\Delta\mathbf{y} = \Delta t(\mathbf{f}^n + \alpha\mathbf{J}\Delta\mathbf{y}), \tag{A.21}$$

or

$$(\mathbf{I} - \alpha\Delta t\mathbf{J})\Delta\mathbf{y} = \Delta t\mathbf{f}^n. \tag{A.22}$$

**Governing equations** The simplified conservation laws used in the original method will be presented next.

**Continuity**

$$\frac{dM_i}{dt} + \sum_j (Ws)_j = 0. \tag{A.23}$$

Here, $M_i$ is the mass of control volume $i$, and $W_j$ the mass flow rates at the control volume faces $j$.

**Conservation of energy**

$$\frac{dH_i}{dt} + \sum_j \left(\frac{HWs}{M}\right)_j = \dot{Q}_i. \tag{A.24}$$

In this equation, $H$ is the enthalpy defined by $H = hM$. The assumption of neglecting the pressure contribution to internal energy has been made (Garland 1998).

**Conservation of momentum**

$$\frac{dW_j}{dt} + \frac{A_j}{L}\Delta p = -k_j|W_j|W_j, \tag{A.25}$$

with the omission of convected momentum.

The semi-discrete equations for continuity, momentum and energy are given below respectively in matrix notation.

$$\frac{d\mathbf{M}}{dt} = \mathbf{f}_M(\mathbf{W}) \tag{A.26}$$

$$\frac{d\mathbf{W}}{dt} = \mathbf{f}_W(\mathbf{W}, \mathbf{p}) \tag{A.27}$$

$$\frac{d\mathbf{H}}{dt} = \mathbf{f}_H(\mathbf{M}, \mathbf{W}, \mathbf{H}). \tag{A.28}$$

Applying scheme (A.21) to these equations give

$$\Delta\mathbf{M} = \Delta t(\alpha_M^W \mathbf{A}_M^W \Delta\mathbf{W} + \mathbf{f}_M^n) \tag{A.29}$$

$$\Delta\mathbf{W} = \Delta t(\alpha_W^p \mathbf{A}_W^p \Delta\mathbf{p} + \alpha_W^W \mathbf{A}_W^W \Delta\mathbf{W} + \mathbf{f}_W^n) \tag{A.30}$$

$$\Delta\mathbf{H} = \Delta t(\alpha_H^M \mathbf{A}_H^M \Delta\mathbf{M} + \alpha_H^W \mathbf{A}_H^W \Delta\mathbf{W} + \alpha_H^H \mathbf{A}_H^H \Delta\mathbf{H} + \mathbf{f}_H^n). \tag{A.31}$$

Here, $\mathbf{A}$ is an associated Jacobian matrix. For example, $\mathbf{A}_M^W$ is the Jacobian matrix of the mass equation $\mathbf{M}$ with respect to mass flow rate $\mathbf{W}$.

**Equation of state** The approach of FIBS is to obtain expressions for $\Delta p$ and $\Delta T$ in terms of mass $\Delta M$ and enthalpy $\Delta H$ through linearisation and the aid of thermodynamic relations:

$$\Delta p = \left.\frac{\partial p}{\partial M}\right|^n \Delta M + \left.\frac{\partial p}{\partial H}\right|^n \Delta H \tag{A.32}$$

$$\Delta T = \left.\frac{\partial T}{\partial M}\right|^n \Delta M + \left.\frac{\partial T}{\partial H}\right|^n \Delta H. \tag{A.33}$$

If the thermodynamic relations can be explicitly expressed as $p = p(M, H)$ and $T = T(M, H)$, the above linearised equations are straightforward to compute. If the equation of state cannot be expressed as $p = p(M, H)$ and $T = T(M, H)$, but instead $M = M(p, T)$ and $H = H(p, T)$ for example, the partial derivatives $\frac{\partial p}{\partial M}$ and $\frac{\partial p}{\partial H}$ can be obtained through inversion of the Jacobian matrix, as described in Garland (1998).

For $N$ nodes, the linearised equations for pressure and temperature take the form

$$\Delta p_i = C^M_{p,i}\Delta M_i + C^H_{p,i}\Delta H_i, \qquad 1 \le i \le N \tag{A.34}$$

$$\Delta T_i = C^M_{T,i}\Delta M_i + C^H_{T,i}\Delta H_i, \qquad 1 \le i \le N. \tag{A.35}$$

This can be written in matrix notation:

$$\Delta \mathbf{p} = \mathbf{C}^M_p \Delta \mathbf{M} + \mathbf{C}^H_p \Delta \mathbf{H} \tag{A.36}$$

$$\Delta \mathbf{T} = \mathbf{C}^M_T \Delta \mathbf{M} + \mathbf{C}^H_T \Delta \mathbf{H}. \tag{A.37}$$

where $\mathbf{C}^M_p = diag\{C^M_{p,1}, C^M_{p,2}, \ldots, C^M_{p,N}\}$ and $\mathbf{C}^H_p = diag\{C^H_{p,1}, C^H_{p,2}, \ldots, C^H_{p,N}\}$, and $\mathbf{C}^M_T = diag\{C^M_{T,1}, C^M_{T,2}, \ldots, C^M_{T,N}\}$ and $\mathbf{C}^H_T = diag\{C^H_{T,1}, C^H_{T,2}, \ldots, C^H_{T,N}\}$.

**Reduction of equations** In Garland's original method, the above set of equations are reduced to give a system of equations in $\Delta\mathbf{W}$ only. This procedure will now be described.

Substitution of (A.29) in (A.31) leads to

$$\Delta \mathbf{H} = \Delta t[\alpha^M_H \Delta t \mathbf{A}^M_H(\alpha^W_M \mathbf{A}^W_M \Delta \mathbf{W} + \mathbf{f}^n_M) + \alpha^W_H \mathbf{A}^W_H \Delta \mathbf{W} + \alpha^H_H \mathbf{A}^H_H \Delta \mathbf{H} + \mathbf{f}^n_H].$$

which can be rewritten as:

$$(\mathbf{I} - \alpha^H_H \Delta t \mathbf{A}^H_H)\Delta \mathbf{H} = \Delta t(\alpha^M_H \alpha^W_M \Delta t \mathbf{A}^M_H \mathbf{A}^W_M + \alpha^W_H \mathbf{A}^W_H)\Delta \mathbf{W} + \Delta t(\alpha^M_H \Delta t \mathbf{A}^M_H \mathbf{f}^n_M + \mathbf{f}^n_H).$$

Solving for $\Delta\mathbf{H}$ gives

$$\begin{aligned}\Delta \mathbf{H} = {} & \Delta t(\mathbf{I} - \alpha^H_H \Delta t \mathbf{A}^H_H)^{-1}(\alpha^M_H \alpha^W_M \Delta t \mathbf{A}^M_H \mathbf{A}^W_M + \alpha^W_H \mathbf{A}^W_H)\Delta \mathbf{W} \\ & + \Delta t(\mathbf{I} - \alpha^H_H \Delta t \mathbf{A}^H_H)^{-1}(\alpha^M_H \Delta t \mathbf{A}^M_H \mathbf{f}^n_M + \mathbf{f}^n_H).\end{aligned} \tag{A.38}$$

which is in terms of only $\Delta\mathbf{W}$.

By substituting (A.29) and (A.38) into (A.36), $\Delta\mathbf{p}$ can be expressed in terms of mass flow rate $W$:

$$\begin{aligned}\Delta \mathbf{p} = {} & \Delta t \mathbf{C}^M_p(\alpha^W_M \mathbf{A}^W_M \Delta \mathbf{W} + \mathbf{f}^n_M) \\ & + \Delta t \mathbf{C}^H_p(\mathbf{I} - \alpha^H_H \Delta t \mathbf{A}^H_H)^{-1}(\alpha^M_H \alpha^W_M \Delta t \mathbf{A}^M_H \mathbf{A}^W_M + \alpha^W_H \mathbf{A}^W_H)\Delta \mathbf{W} \\ & + \Delta t \mathbf{C}^H_p(\mathbf{I} - \alpha^H_H \Delta t \mathbf{A}^H_H)^{-1}(\alpha^M_H \Delta t \mathbf{A}^M_H \mathbf{f}^n_M + \mathbf{f}^n_H).\end{aligned} \tag{A.39}$$

Defining

$$\mathbf{A}^W_p = \Delta t[\alpha^W_M \mathbf{C}^M_p \mathbf{A}^W_M + \mathbf{C}^H_p(\mathbf{I} - \alpha^H_H \Delta t \mathbf{A}^H_H)^{-1}(\alpha^M_H \alpha^W_M \Delta t \mathbf{A}^M_H \mathbf{A}^W_M + \alpha^W_H \mathbf{A}^W_H)]$$

and

$$\mathbf{A}_p^0 = \Delta t[\mathbf{C}_p^M \mathbf{f}_M^n + \mathbf{C}_p^H (\mathbf{I} - \alpha_H^H \Delta t \mathbf{A}_H^H)^{-1} (\alpha_H^M \Delta t \mathbf{A}_H^M \mathbf{f}_M^n + \mathbf{f}_H^n)],$$

(A.39) can be rewritten as

$$\Delta \mathbf{p} = \mathbf{A}_p^W \Delta \mathbf{W} + \mathbf{A}_p^0. \tag{A.40}$$

Finally, substitution of (A.40) into (A.30) leads to:

$$\Delta \mathbf{W} = \Delta t(\alpha_W^p \mathbf{A}_W^p \mathbf{A}_p^W \Delta \mathbf{W} + \alpha_W^p \mathbf{A}_W^p \mathbf{A}_p^0 + \alpha_W^W \mathbf{A}_W^W)\Delta \mathbf{W} + \Delta t \mathbf{f}_W^n.$$

and upon rearrangement gives the final set of equations:

$$[\mathbf{I} - \Delta t(\alpha_W^p \mathbf{A}_W^p \mathbf{A}_p^W + \alpha_W^W \mathbf{A}_W^W)]\Delta \mathbf{W} = \Delta t(\alpha_W^p \mathbf{A}_W^p \mathbf{A}_p^0 + \mathbf{f}_W^n). \tag{A.41}$$

This is of the form $\mathbf{Ay} = \mathbf{b}$ and can be solved by conventional means for $\Delta \mathbf{W}$. With $\Delta \mathbf{W}$ known, $\Delta \mathbf{M}$ can be solved for directly with (A.29) and $\Delta \mathbf{H}$ with (A.31). Pressure and temperature changes can be solved with (A.36) and (A.37) respectively.

**Drifting phenomena and alternative form of FIBS** Garland (1998) described the drifting of the pressure and temperature when updated with the expressions

$$\Delta p = \left.\frac{\partial p}{\partial M}\right|^n \Delta M + \left.\frac{\partial p}{\partial H}\right|^n \Delta H \tag{A.42}$$

$$\Delta T = \left.\frac{\partial T}{\partial M}\right|^n \Delta M + \left.\frac{\partial T}{\partial H}\right|^n \Delta H. \tag{A.43}$$

which leads to stationary solutions not satisfying the equation of state. Garland further described correcting techniques in overcoming the drifting. In this section, an alternative treatment of the equation of state is given.

If the thermodynamic relations can be explicitly expressed as $p = p(M, H)$ and $T = T(M, H)$, the pressure and temperature updates can be updated explicitly without any drifting. If the equation of state cannot be expressed as $p = p(M, H)$ and $T = T(M, H)$, but instead $M = M(p, T)$ and $H = H(p, T)$ for example, then the equation of state can be solved for using a non-linear solver such as Newton's method. For example, in solving $p$ and $T$ at temporal index $n + 1$, Newton's method gives

$$p^{k+1} = p^k + \left.\frac{\partial p}{\partial M}\right|^k [M^{n+1} - M(p^k, T^k)] + \left.\frac{\partial p}{\partial H}\right|^k [H^{n+1} - H(p^k, T^k)] \tag{A.44}$$

$$T^{k+1} = T^k + \left.\frac{\partial T}{\partial M}\right|^k [M^{n+1} - M(p^k, T^k)] + \left.\frac{\partial T}{\partial H}\right|^k [H^{n+1} - H(p^k, T^k)]. \tag{A.45}$$

Here $k$ is the iteration counter. For a single iteration with initial guess $p = p^n$ and $T = T^n$ this reduces to

$$p^{n+1} = p^n + \left.\frac{\partial p}{\partial M}\right|^n [M^{n+1} - M(p^n, T^n)] + \left.\frac{\partial p}{\partial H}\right|^n [H^{n+1} - H(p^n, T^n)] \tag{A.46}$$

$$T^{n+1} = T^n + \left.\frac{\partial T}{\partial M}\right|^n [M^{n+1} - M(p^n, T^n)] + \left.\frac{\partial T}{\partial H}\right|^n [H^{n+1} - H(p^n, T^n)]. \tag{A.47}$$

This is nearly similar to the formulations from FIBS ((A.42) and (A.43)), but without drifting.