Swansea University
Prifysgol Abertawe

Civil and Computational Engineering
School Of Engineering

**Master Thesis**

# A Computational Framework for Sloshing in Liquid Storage Tank: Theory and Application

**Pavan Kumar Sriram – 530144**
MSc Computational Mechanics
June 2010

**Professor Djordje Peric**
Master Thesis Supervisor
Civil Engineering Programme Director

Dissertation submitted to Swansea University in Partial fulfillment for the Degree of Master of Science in Computational Mechanics

# Declaration

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

**STATEMENT 1**

This dissertation is being submitted in partial fulfillment of the requirements for the degree of MSc.

**STATEMENT 2**

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

**STATEMENT 3**

I hereby give consent for my dissertation, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations. These declarations should be accompanied by your signature and date of signature.

Signature Date

Pavan Kumar Sriram

# Acknowledgements

I am very much indebted to Djordje Peric for having accepted to be my thesis director. He accepted me in his group and gave me the opportunity to work on very interesting and various subjects in analysis and numerical simulation fluid flow problems. I have learned a lot from his suggestions and remarks, as well in research or in teaching.

I am most grateful to Dr W. G. Dettmer for his scientific support. I particularly appreciate his availability and his numerous good advices and indications. It is also a pleasure to thank Maartje for her kind and frequent support.

Thankful to Dr. Antonio J. Gil being for all the help and support provided during our Masters course. Thanks also Anne Davies for the guidance and support with administartive work. I thank the European Union for its fnancial support through Erasmus Mundus Scholarship.

At this point, I also want to thank all the present and former members of the team of MSc Computational Mechanics. For a couple of years, we have shared many interesting discussions about subject or other aspects of a Masters student's life. In particular, thanks for the very good atmosphere inside our group. Finally I would like to thank all my friends, inside and outside the Swansea University, who contributed from far or near to making this period so rich and interesting.

I am most thankful to European Commission for sponsoring me with Erasmus Mundus Scholarship and also to International Center for Numerical Methods in Engineering (CIMNE), Barcelona for making my Masters course an enriching experience.

I owe my loving thanks to my family and friends. They have lost a lot due to my research abroad. Without their encouragement and understanding it would have been impossible for me to finish this work.

# Abstract

Mathematical and numerical aspects of free surface flows are investigated.

On one hand, the mathematical analysis of some free surface flows is considered. A model problem in one space dimension is first investigated. The Burgers equation with diffusion has to be solved on a space interval with one free extremity. This extremity is unknown and moves in time. The main work is concerned with the simulation of the incompressible Newtonian fluid flow problem. The space discretisation is based on the stabilized velocity-pressure finite element method. The movement and the deformation of the domain are accounted for by employing the arbitrary Lagrangian-Eulerian (ALE) description of the fluid kinematics. The time discretisation is carried out by using implicit, explicit and semi implicit scheme. The stability and the convergence of time splitting scheme are investigated. A partitioned solution procedure is developed based on the Newton-Raphson methodology which incorporates full linearization of the overall incremental problem. Accuracy and stability of the solutions are demonstrated in example for which the analytical solutions are known. In the example, the Burger's equation analogue to 1-D fluid flows is solved without and with FE mesh motion, to show that the mesh motion practically does not affect the solutions. All solutions presented show that the proposed algorithm is sufficiently accurate and stable. Since the algorithm is implicit, high accuracy of results can be achieved with a relatively large time step. A numerical example is provided to demonstrate the efficiency of the methodology by modeling large amplitude sloshing in a rectangular tank.

# Table of Contents

# Chapter 1

## Introduction

Flows with free surface find an important place in many engineering applications [1-16]. Free surface flows find large number of industrial interests like, liquid sloshing in LNG tankers, chemical and food industry, diesel injectors, atomization, droplet-wall interaction, cavitation, Ink-jets and similar devices and involved complex

## 1.1 Motivation

Numerical methods for solving free surface problems are of great importance in many engineering applications. Free surfaces are abundant in nature and are often can be applied to civil, mechanical and chemical engineering. Problems with free surfaces appear in sloshing in liquid storage tanks [1, 2, 3], fluid-structure interactions [4, 5, 6], blood flows in moving arteries [7], immiscible multi-fluids problems [8, 9, 10], motion of glaciers [11], viscoelastic flows [12, 13], mould filling [14, 15, 16] and many other domains.

Free surface flows are investigated here, with particular emphasis on the process of sloshing in liquid storage tanks. Liquid storage tanks are important components of lifeline and industrial facilities. They are critical elements in municipal water supply and fire fighting systems, and in many industrial facilities for storage of water, oil, chemicals and liquefied natural gas. Behavior of large tanks during seismic events has implications far beyond the mere economic value of the tanks and their contents. If, for instance, a water tank collapses under earthquakes, loss of public water supply can have serious consequences. Similarly, failure of tanks storing combustible materials, under earthquakes, can lead to extensive uncontrolled fires. Many researchers have investigated the dynamic behavior of liquid storage tanks both theoretically and experimentally. Investigations have been conducted to seek possible improvements in the design of such tanks to resist earthquakes.

Liquid storage tanks can be found in many configurations: elevated, ground-based, and underground. Steel ground-based tanks consist essentially of a steel wall that resists outward liquid pressure, a thin flat bottom plate that prevents liquid from leaking out, and a thin roof plate that protects contents from the atmosphere. It is common to classify such tanks in two categories depending on support conditions: anchored and unanchored tanks.

Anchored tanks must be connected to large foundations to prevent the uplift in the event of earthquake occurrence. However, improperly detailed anchors may damage the shell under seismic loading resulting in a ripped tank bottom. Hence, it is common, particularly for large size tanks, to support the shell on a ring wall foundation without anchor bolts and to support the bottom plate on a compacted soil though, sometimes, ring walls are omitted. Based on the orientation of the axis of symmetry, anchored tanks are either horizontal or vertical. Circular vertical tanks made of carbon steel are more numerous than any other type because they are efficient in resisting liquid hydrostatic pressure mostly by membrane stresses, simple in design, and easy in construction.

## 1.2 Damage to Storage tanks in earthquakes

Research on seismic response and behavior of liquid storage tanks is a matter of special importance, not only because of the economic factors, but also because of the consequences that result from failing tanks. Without an assured water supply, uncontrolled fires may cause enormously more damage than the earthquake itself, as it happened 1906 in the great San Francisco (USA) or 1995 in the earthquake of Kobe (Japan).

Spillage of toxic chemicals or liquefied gases from the damaged tanks can lead to disastrous effects in populated areas. The seeping of oil into the ground can ruin the ground water, so happened 1978 in Japan. Failure of tanks containing high inflammable products can lead to extensive fires, as occurred following the Nigata and Alaska of 1964, or the earthquake in Turkey on 17.8.1999 when over 17100 people died.

It becomes very important to study the liquid storage tank, when these liquid storage tanks are subjected to earthquakes, they suffer significant damage It becomes important to study and understand the behavior of tank under these conditions and various failures modes. The major damages observed in the liquid storage tanks are *"Elephant foot" and "Diamond shape" buckling,. "Uplifting of tank, "Damage and collapse of tank roofs".* The seismic design standards have been revised several times to improve the performance. Because of cost, ground supported liquid storage tanks are often not fixed to their foundation, even in seismic areas. In this work we will study and understand the effect of sloshing on the tank. Tanks that are not provided with sufficient freeboard can be damaged by the sloshing waves.

In the present study, we model one dimensional free surface problem using burgers equation and understand the problem in moving domain framework. The goal is to examine the formulations of the non-linear liquid sloshing behavior for a one dimensional and two dimensional problems. Sloshing waves of high amplitude often cause damage to the roofs of tanks and render them temporarily unserviceable. As a consequence, liquid spillage over the roof may either result in fires or in the loss of water supply used in putting out fires. It can be seen that the most commonly reported damage to above ground prestressed concrete tanks is due to sloshing. The cause of this damage is mainly due to the liquid sloshing of oil storage tanks excited by long-period strong ground motion.

Figure 1: Damaged Liquid Storage tank (A) due to sloshing, Ref Carleton University

Figure 2: Damaged Liquid Storage tank (B) due to sloshing Ref: Carleton University

The main objective of the thesis is concerned with the simulation of free surface flows with a focus to develop 1 dimensional and 2 dimensional problems of incompressible Newtonian fluid flow (sloshing of liquid storage tanks, were in the surface tension presence is neglected). The aimed application is the representation of sloshing in liquid storage tank. To reach this goal, it is first necessary to choose the right models and develop stable and efficient enough algorithm.

## 1.3 Outline

Thus, the outline of this dissertation is as follows:

This thesis consists of two different parts. In a first part, a problem related to this model is investigated from a theoretical point of view. The second part focuses on the simulation of such free surface flows. Many methods exist in the literature to treat free surface problems. References appearing in this work, and especially in this introduction, give only a non-exhaustive list of examples.

The first part of this thesis (*chapters 1 and 2*) is devoted to theoretical studies related to free surface flows.

In *chapter1,* main motivation behind this work is explained and it explains why there is need to study the effect of sloshing on liquid storage tanks this is done mathematically by using the one dimensional burgers equation with the viscous term. The study of burgers equation with a free extremity is carried out in a fixed and a moving domain framework.

In *chapter 2*, a one-dimensional model problem for the velocity u is first investigated. This model is a one-dimensional simplification of the free surface problem. It consists of Burgers' equation with an additional diffusion term in a space interval with one free extremity. This extremity is unknown and moves in time. Surface tension effects are not taken into account. A zero force boundary condition is thus enforced on the free extremity of the interval. Other results for the Burgers equation with a free surface can be found in [18, 19] for instance.

The fluid flow considered is governed by the incompressible Burgers equation and modelled by employing stabilized equal order velocity–pressure finite elements and for the temporal discretisation the explicit, implicit and semi-implicit method is employed. The resulting nonlinear equations are solved by means of a novel partitioned solution procedure, which is based on the Newton–Raphson methodology and incorporates full linearisation of the overall incremental problem. In this chapter the motion of the fluid is assumed to be fixed under dirichlet boundary conditions, which fix the value of the field on the surface.

In *chapter 3*, the motion of the fluid domain is accounted for by an Arbitrary Lagrangian–Eulerian (ALE) [20-35] strategy. The domain of computation is stretched and re-meshed at each time step. For flows in complex topological domains, re-meshing can be difficult since the deformation of the liquid domain is large. The solution of free surface flow problems may be regarded as a first step towards the modeling of fluid-structure interactions problem, where the coupling is the more complex, including also the equations governing solid deformations.

The second part of this thesis (*chapters 4 and 5*) is devoted to study of variation of viscosities in unsteady flow and numerical example related to free surface flows.

In *chapter 4*, the study of shock formation by varying the viscosity values provided by the time evolution of a sinusoidal wave profile under fixed Cartesian grid and a domain accounted for by an ALE strategy is carried out and we can carry out a instability study by adding others terms to the Burgers' equation. In *chapter 5*, an example of two-dimensional and axisymmetric free surface flows have been used for the verification of the computational strategy. The numerical results obtained for a selection of problems are presented. The example considers two-dimensional small and large amplitude sloshing of a fluid in a rectangular tank.

# Chapter 2

## Burgers Equation: Fixed Mesh Framework

A one-dimensional simplified model of a free surface problem is considered in this chapter, namely the Burgers equation with an additional diffusion. This problem is a simplification in one space dimension of the Navier-Stokes equations with a free surface encountered in [36, 37]. Space and time discretisation are investigated (see also [38-44] for instance). Here the Cartesian grid movement is fixed and the numerical results are obtained for the same.

## 2.1 Computational Method

In this chapter the computational method for fluid flow is explained. First, the mathematical equations are stated that describe the flow including the motion of the free surface. Secondly the discretisation in space and the governing equations are discussed. Finally, the temporal discretisation is explained. Phenomenon involving free surface are found extensively in nature and in most engineering applications in many interdisciplinary fields. Due to the complexity of the equations involved in the motion of a fluid, it is only possible to obtain analytical solutions for a few simple problems, often by using simplifications. The numerical treatment of free surface problems is highly interesting but complex because the computational domain is continuously moving. Although the underlying physics of free surface flows is understood, the numerical treatment differs with respect to handling the moving boundaries at the free surface. Numerical simulations of the viscous free surface flows are well established for the velocity formulation and the results have been obtained using various numerical schemes such as the finite difference method (FDM) [45,46], the finite element method (FEM) [47-52] and the finite volume method (FVM) [53].

## 2.2 Mathematical Model

Burgers equation is a one-dimensional case of the Navier-Stokes momentum equation, when the pressure gradient and the forcing term are neglected, and is given as follows.

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} - \mu\frac{\partial^2 u}{\partial x^2} = 0 \qquad \forall \quad 0 \le x \le L \tag{1}$$

Equation (1) is parabolic when the viscous term is included. If the viscous term is neglected, the remaining equation is composed of the unsteady term and a nonlinear convection term. This results in a hyperbolic equation. Dirichlet conditions will be applied on the boundaries, that is, $x=0$ and $x=L$ such that the value of the quantity $u$ is known. The above equation can be viewed as a nonlinear wave equation where each point on the wave front can propagate with different speed.

A consequence of the changing wave speed is the coalescence of characteristics and the formation of discontinuous solution similar to shock waves in fluid mechanics. These discontinuities are studied with the simple one dimensional model in chapter 3.

The first step for a finite element space discretisation it to introduce a weight function $w(x)$ that is arbitrary along the domain, but zero at the boundary with Dirichlet conditions ($w(0) = w(L) = 0$). Then, the weak form of the burgers equation becomes:

$$\int_{\Omega} w \cdot \frac{\partial u}{\partial t} dx + \int_{\Omega} w \cdot u \frac{\partial u}{\partial x} dx - \int_{\Omega} w \cdot \mu \frac{\partial^2 u}{\partial x^2} dx = 0 \tag{2}$$

Applying integration by parts on the second order derivative term, taking the constants out of the integral and knowing that the weight function is zero at the boundaries, Eq. (2) may be rewritten as:

$$\int_{\Omega} w \cdot \frac{\partial u}{\partial t} dx + \int_{\Omega} w \cdot u \cdot \frac{\partial u}{\partial x} dx + \mu \int_{\Omega} \frac{\partial w}{\partial x} \cdot \frac{\partial u}{\partial x} dx = 0 \tag{3}$$

Equation (3) is the weak form of the unsteady Burgers equation modeling the incompressible fluid flow free surface problem. We use a stabilized equal order velocity-pressure finite element formulation adapted to a fixed domain. The stabilization technique employed was introduced by Hughes and co-workers [38-40] and then further developed by Tezduyar and others [41-44]. It enhances stability of the velocity field in the advection dominated regions of the domain and at the same time enables the use of computationally convenient equal order finite elements for spaces for velocity and pressure fields. The formulation used in the work is referred to as Standard Galerkin and Petrov-Galerkin methods [5]. Such methods have become standard in Eulerian finite element formulations and have been applied to various problems in fluid mechanics [38]. A review of a variety of stabilization techniques may be found in [39]

## 2.3 Discretization in Space using FEM Technique

*2.3.1 Standard Galerkin Method*

The Standard Galerkin method is the most widely used method for the weight function in the finite element method. The reason lays in its simplicity, where the weight function *w* is replaced by the shape function $N_i$, interpolation function. The weight function and its derivative may be written as:

$$w(x) = \sum_i^n N_i w_i \qquad \text{and} \qquad \frac{dw}{dx} = \sum_i^n \frac{dN_i}{dx} w_i \tag{4}$$

Different shape functions can be chosen for the one-dimensional finite element problem but, in this work it will be adopt only linear shape functions. Applying the Galerkin method in Eq. (3) the following expression is obtained:

$$\int_\Omega u_j \frac{\partial N_i}{\partial t} dx + \int_\Omega N_1 N_i u_1 \frac{\partial u}{\partial x} u_j dx + \int_\Omega N_2 N_i u_2 \frac{\partial u}{\partial x} u_j dx + \mu \int_\Omega \frac{\partial N_i}{\partial x} \cdot \frac{\partial N_j}{\partial x} u_j dx = 0 \tag{5}$$

For a two-noded element in the reference space with linear interpolation of the shape functions may be written as:

$$N_1 = \frac{1}{2}(1-\xi) \qquad \text{and} \qquad N_2 = \frac{1}{2}(1+\xi) \tag{6}$$

with the mapping from the current element to the reference element as:

$$dx = \frac{1}{2}(x_2 - x_1) = \frac{h}{2} d\xi \tag{7}$$

Substituting equations (6) and (7) into Eq.(5), integrating over the reference element size ($-1 \leq x \leq 1$) and rewriting in a matrix form, the above equation becomes:

$$\mathbf{A}^e = \frac{h}{6}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} ; \quad \mathbf{C}^e = \frac{u}{2}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{K}^e = \frac{\mu}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{8}$$

By assembling together contributions from all elements we find the matrix equation

$$\mathbf{A}\frac{\partial u}{\partial t} + [\mathbf{C} + \mathbf{K}]u = 0 \tag{9}$$

### *2.3.2 Petrov-Galerkin Approximation*

Different from the standard Galerkin method, the Petrov-Galerkin uses a weight function different from the shape function. The most used Petrov-Galerkin weight function is the discontinuous function defined as:

$$w(x) = \sum_{i}^{n} N_i(x) + \alpha \frac{h}{2} \frac{dN_i(x)}{dx} \frac{u}{|u|} \tag{10}$$

Substituting Eq.(10) into the weak form of the convection-diffusion problem in Eq.(3) the following equation is obtained:

$$\sum_{i}^{n}\sum_{j}^{n} \int_{\Omega^e} \left[ u \left( N_i(x) + \alpha \frac{h}{2} \frac{dN_i(x)}{dx} \right) \frac{dN_j}{dx} + \mu \frac{dN_i}{dx} \frac{dN_j}{dx} \right] u_j + \int_{\Omega^e} \left( N_i(x) + \alpha \frac{h}{2} \frac{dN_i(x)}{dt} \right) dx = 0 \tag{11}$$

$$\begin{aligned} &\int_{\Omega} u_j \frac{\partial N_i}{\partial t} dx + \int_{\Omega} \left( N_i + \alpha \frac{h}{2} \cdot \frac{\partial N_i}{\partial x} \right) N_1 u_1 \frac{\partial u}{\partial x} u_j dx + \int_{\Omega} \left( N_i + \alpha \frac{h}{2} \cdot \frac{\partial N_i}{\partial x} \right) N_2 u_2 \frac{\partial u}{\partial x} u_j dx \\ &+ \mu \int_{\Omega} \left( \frac{\partial N_i}{\partial x} + \alpha \frac{h}{2} \cdot \frac{\partial^2 N_i}{\partial x^2} \right) \frac{\partial N_j}{\partial x} u_j dx = 0 \end{aligned} \tag{12}$$

Once again the result of the quantity variable $u$ in each node may be obtained by solving the system of equations. It is possible to see that the diffusion matrix was not changed when implementing the Petrov-Galerkin method. Since it was implemented a linear shape function, the second derivative of the shape function is zero and, therefore, the value of the derivative $dN_i/dx$ is the same for both Galerkin and Petrov-Galerkin methods.

We can see that with $\alpha$ = 0, the standard galerkin approximation is recovered. An optimal value α of may be obtained as well as the limit of α for which oscillations don't occur. The optimal value was used throughout this work and may be defined as:

$$\alpha_{opt} = \coth|Pe| - \frac{1}{|Pe|} \tag{13}$$

It is important to notice that, since the constant $\alpha$ cancels for the internal nodes during the assembly of the elements, the only terms affected by the constant $\alpha$ are the non-diagonal terms in the convective matrix. The same result is obtained when introducing the balancing diffusion technique. The results obtained with the code written in Matlab, comparing the artificial diffusion and the Petrov-Galerkin stabilization methods with the standard Galerkin finite element method will be shown in the next section.

## 2.4 Temporal Discretization

For unsteady processes the physical quantities in addition to the spatial dependence also depend on the time *t*. In many practical applications the processes under consideration are unsteady and thus require for their numerical simulation the solution of time- dependent model equations.

### *2.4.1 Explicit Method*

Here time discretisation is carried out using the explicit Euler method [56], which is obtained by approximating the time derivative at time level, $t_n$ by means of a forward differencing scheme:

$$\frac{\partial u}{\partial t}(t_n) \approx \frac{u^{n+1} - u^n}{\Delta t_n} = \ell(u^n) \tag{14}$$

This corresponds to an approximation of the time derivative of the components $u_i$ of $u$ at the time $t_n$ by means of the slope of the straight line through the points $u_i^n$ and $u_i^{n+1}$.

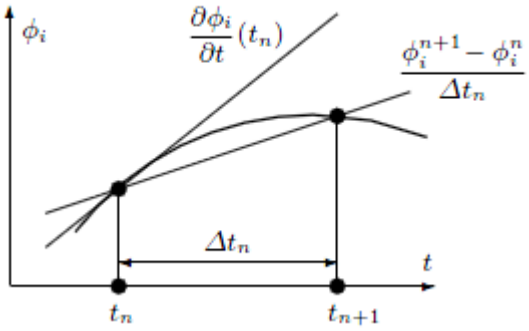



Figure 2.1: Approximation of time derivative with explicit Euler method [56]

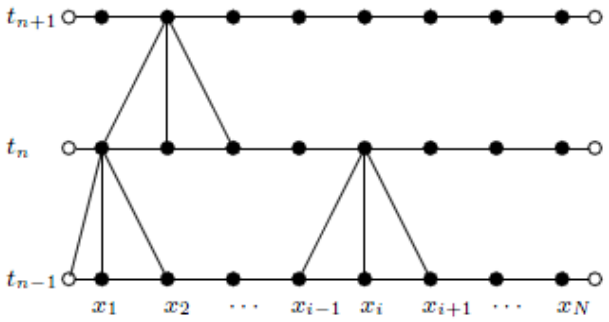



Figure 2.2: Procedure and flow of information for explicit Euler method [56]

### 2.4.2 *Implicit Method*

Approximating the time derivative at time $t_{n+1}$ by a first order backward difference formula results in the implicit Euler method [56].

$$\frac{\partial u}{\partial t}(t_{n+1}) \approx \frac{u^{n+1}-u^{n}}{\Delta t_n} = \ell(u^{n+1}) \tag{15}$$

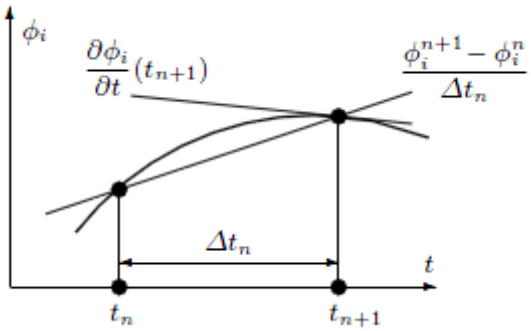



Figure 1.3: Approximation of time derivative with implicit Euler method [56]

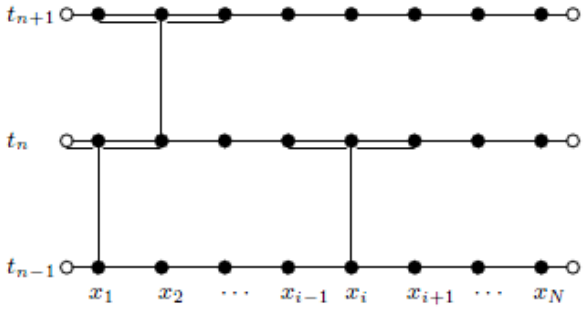



Figure 2.4: Procedure and flow of information for implicit Euler method [56]

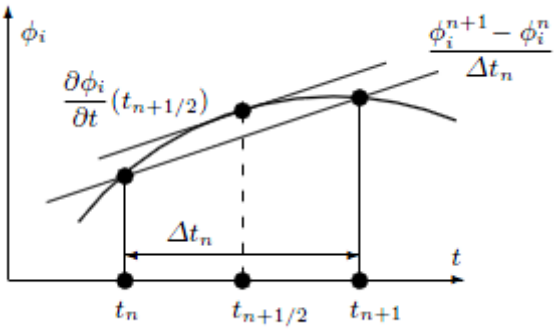



Figure 2.5: Approximation of time derivative with semi implicit crank nicholson method [56]

We have integrated the explicit and implicit Euler methods as well as the Crank-Nicolson method into a single code in a simple way by introducing a control parameter $\theta$

$$\frac{u^{n+1}-u^{n}}{\Delta t_n} = \theta\ell(u^{n+1}) + (1-\theta)\ell(u^{n}) \tag{16}$$

This approach in the literature is often called *θ-method*. For $\theta = 0$ and $\theta = 1$ the explicit and implicit Euler methods, respectively, result. $\theta = 1/2$ gives the Crank-Nicolson method [56]. Valid time discretisation is also obtained for all other values of $\theta$ in the interval [0, 1].

## 2.5 Newton Raphson Procedure

The burgers equation is non-linear the equation and has to be solved using a suitable iteration procedure. There is a variety of different methods through which the numerical solution of non-linear equations can be obtained. In practice, the most commonly employed iteration procedure is the Newton-Raphson method, which we briefly present below.

To incorporate full linearization of the problem and to achieve quadratic convergence of the solution for all unknowns the Newton-Raphson method is employed. The Newton-Raphson iteration procedure can be described as follows. Let us consider a differentiable function $f(x)$ be a on some closed interval and we want to find the value of x for which $f(x)=0$. If we have an approximation to the root of the function, say $x^n$, then by taking a linear approximation of Taylor series expansion of $f$ about $x^n$ (i.e. the first two terms) we get the tangent line to $f(x^n)$ at the point $x^n$ which reads

$$y = f(x^n) + (x - x^n) f'(x^n) \tag{17}$$

The point where this tangent line crosses the $x$ axis is denoted by $x^{n+1}$ and represents an improved approximation to the root

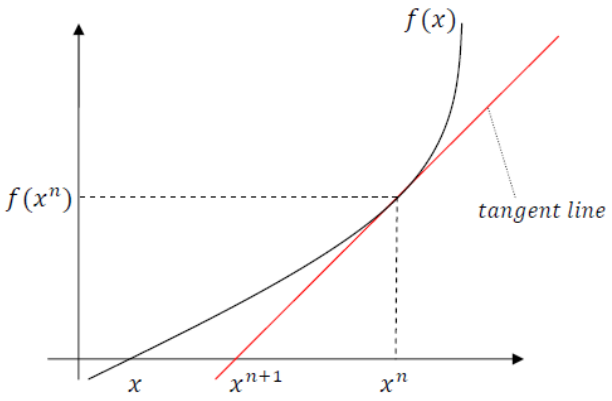



Figure 2.6: Newton Raphson Method

The process is repeated until the method converges, i.e. a desired approximation to x is achieved. This enables us to express the method, mathematically, as follows

$$\begin{aligned} x^{n+1} &= x^n + \Delta x \\ f'(x^n)\Delta x &= -f(x^n) \end{aligned} \tag{18}$$

The Newton-Raphson method converges very quickly if $f'(x^n) \neq 0$ holds and an approximation $x^n$ are reasonably close to the root. The process is terminated when the solution approximations become identical up to the specified number of decimal places. In order to generalize the algorithm for the case when we are dealing with a *n*-dimensional vector function $\mathbf{F}(x)$ we have to employ a *Jacobian* matrix of $\mathbf{F}$. The Jacobian matrix of $\mathbf{F}$ consists of first order partial derivatives of the vector function $\mathbf{F}(\boldsymbol{x})$ and can be expressed as follows

$$\boldsymbol{F}'(\boldsymbol{x}) = \left[\frac{\partial f_i}{\partial x_j}\right] = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix} \tag{19}$$

We, define the Newton Raphson algorithm as follows:

$$\begin{aligned} \boldsymbol{x}^{n+1} &= \boldsymbol{x}^n + \Delta \boldsymbol{x}, \\ \boldsymbol{F}'(\boldsymbol{x}^n)\Delta \boldsymbol{x} &= -\boldsymbol{F}(\boldsymbol{x}^n). \end{aligned} \tag{20}$$

The termination point of an iteration process is determined through an established criteria and the level of accuracy required.

Newton- Raphson Algorithm in steps: For a given time step

1. Predict initial free surface velocity
2. Free surface solver
3. Compute residual
4. Check for convergence
5. Compute linearization and solve system of equations
6. Update the free surface velocity
7. Go to 2

The above solution procedure is repeated until the prescribed number of time steps is reached.

## 2.6 Numerical Results

In order to validate our algorithm for the numerical simulation of the free surface flow in 1-D, we have considered burgers equation in fixed frame work. We study the time evolution of the fluid flow.

An analytical solution for this problem may be found as:

$$u_{exact} = \frac{e^{ux/k} - e^{ul/k}}{1 - e^{uL/k}} \tag{21}$$

The numerical analysis was carried out for Standard Galerkin and Petrov Galerkin method discretisation in space and implicit and explicit method in time and each one of the methods described for three cases of Peclet number. First two case with Pe < 1 and another case with Pe > 1.

The finite element code with the discretisation in space using Galerkin method and in time using backward Euler, Crank Nicholson and forward Euler was first run for a viscosity $\mu = 0.5$ and number of elements $n = 20$ resulting in Pe = 0.5. For the second case it was used a kinematic viscosity $\mu = 0.25$ and number of elements $n=20$ resulting in Pe = 1. For the third case it was used a viscosity $\mu = 0.125$ and number of elements $n=20$ resulting in Pe = 2. The results obtained for Galerkin finite element method for cases 1, 2 and 3 are shown in Fig. below.

*Standard Galerkin in space and Backward Euler in time, Pe = 0.5*

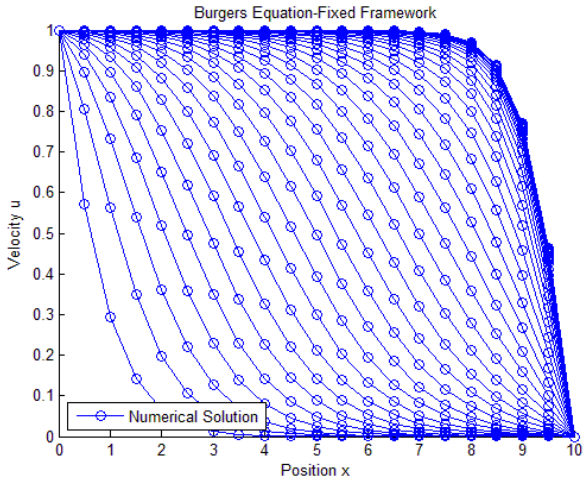



Figure 2.7: Velocity v/s Position, Numerical Soln, Pe=0.5

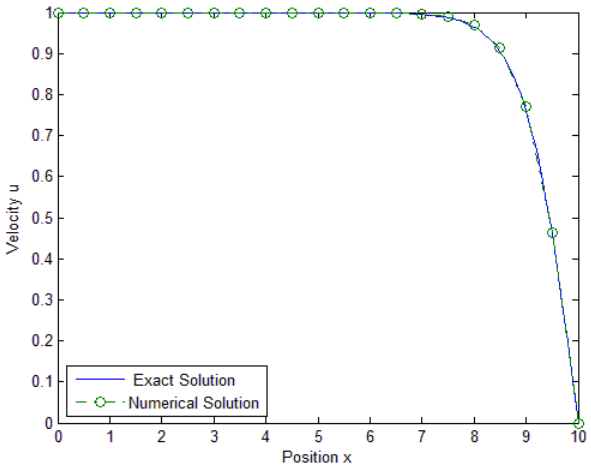



Figure 2.8: Comparison of Numerical with Exact Solution, Pe=0.5

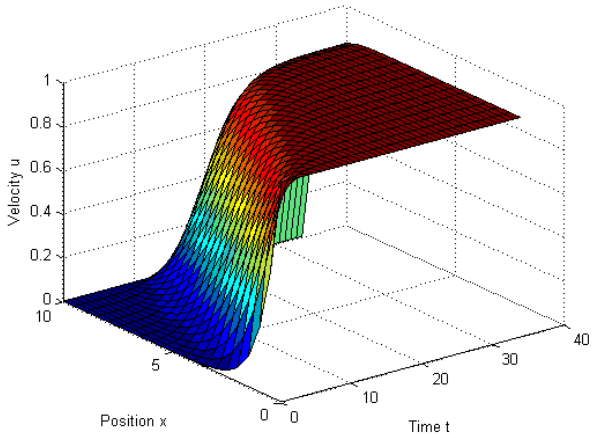



Figure 2.9: Surface Plot, Pe=0.5

*Standard Galerkin in space and Backward Euler in time, Pe = 1*

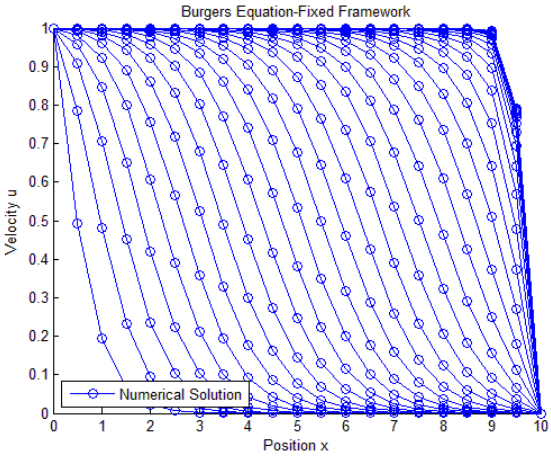



Figure 2.10: Velocity v/s Position, Numerical Soln, Pe=1

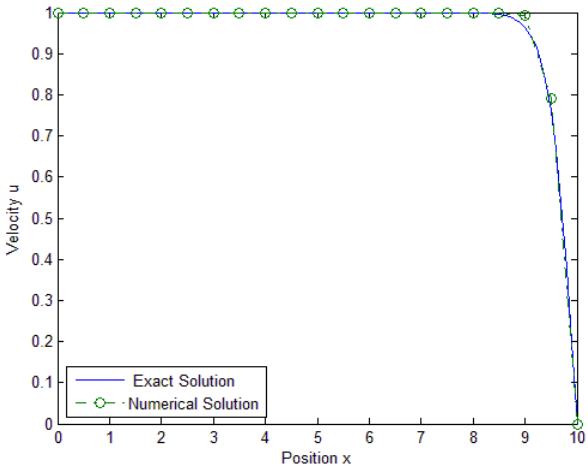



Figure 2.11: Comparison of Numerical with Exact Solution, Pe=1

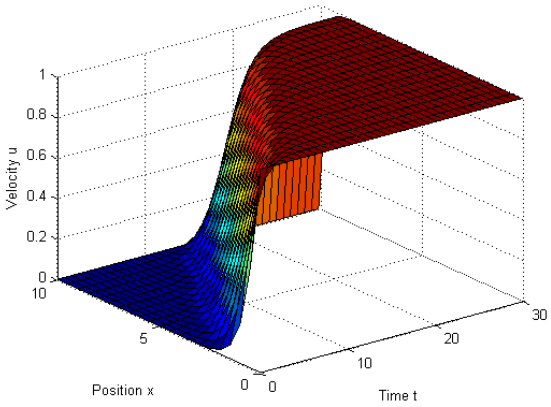



Figure 2.12: Surface Plot, Pe=1

*Standard Galerkin in space and Backward Euler in time, Pe = 2*

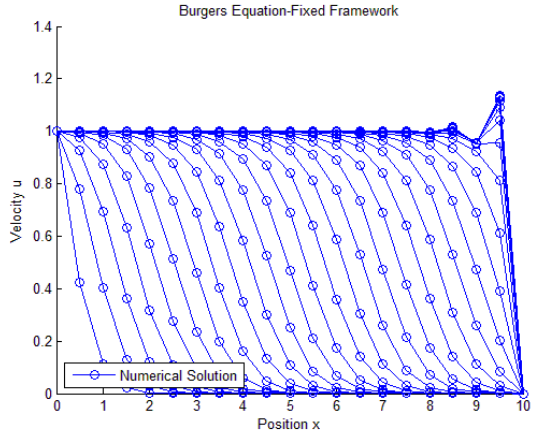



Figure 2.13: Velocity v/s Position, Numerical Soln, Pe =2

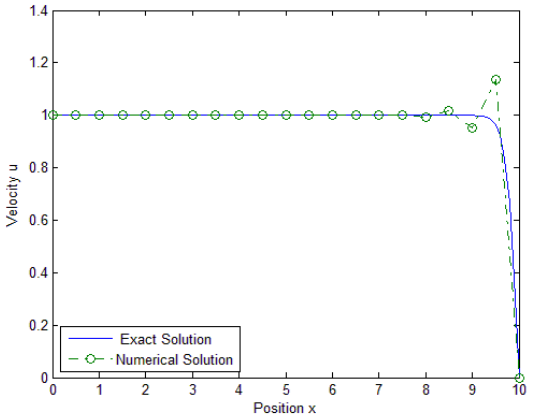



Figure 2.14: Comparison of Numerical with Exact Solution, Pe=2

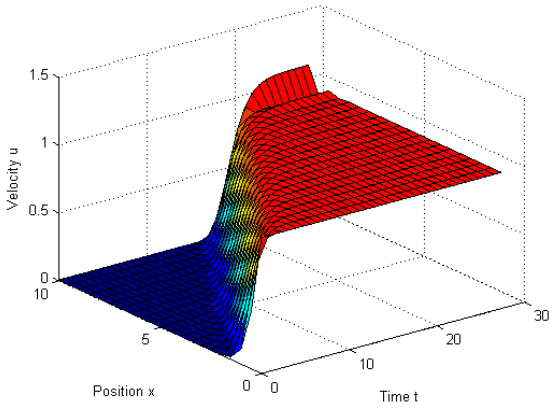



Figure 2.15: Surface Plot, Pe=2

*Standard Galerkin in space and Crank Nicholson in time, Pe = 0.5*

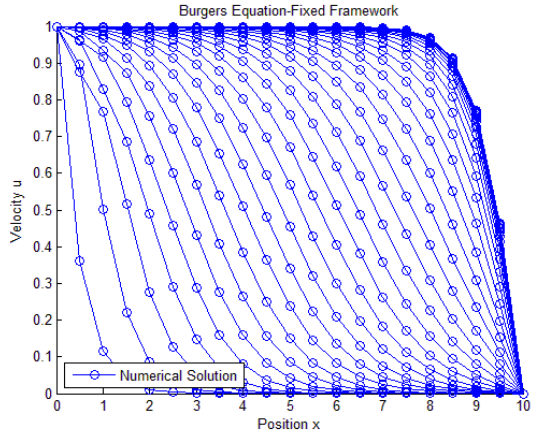



Figure 2.16: Velocity v/s Position, Numerical Soln, Pe=0.5

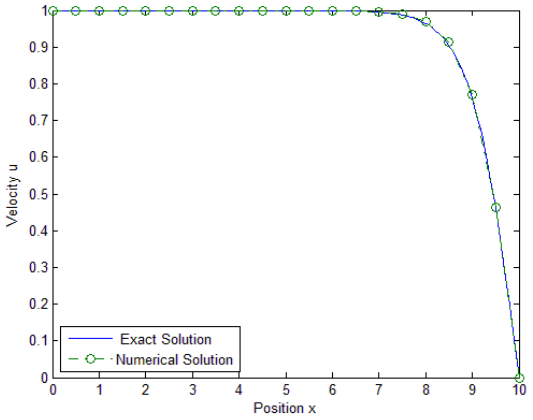



Figure 2.17: Comparison of Numerical with Exact Solution, Pe=0.5

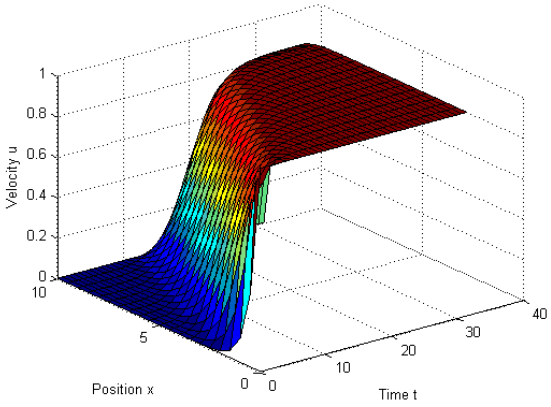



Figure 2.18: Surface Plot, Pe=0.5

*Standard Galerkin in space and Crank Nicholson in time, Pe = 1*

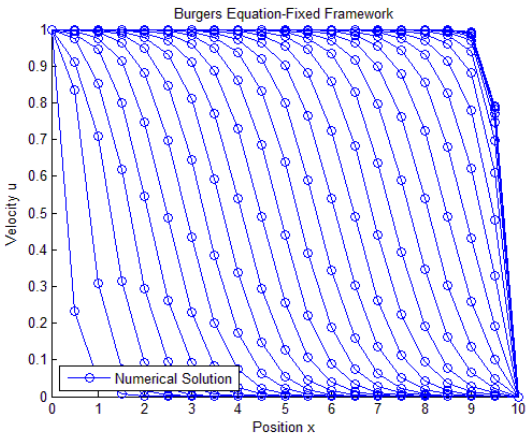



Figure 2.19: Velocity v/s Position, Numerical Soln, Pe=1

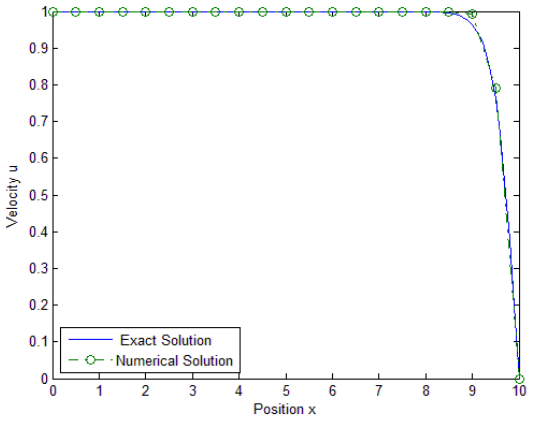



Figure 2.20: Comparison of Numerical with Exact Solution, Pe=1

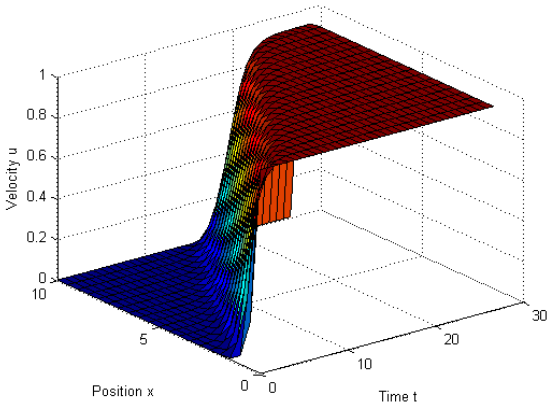



Figure 2.21: Surface Plot, Pe=1

*Standard Galerkin in space and Crank Nicholson in time, Pe = 2*

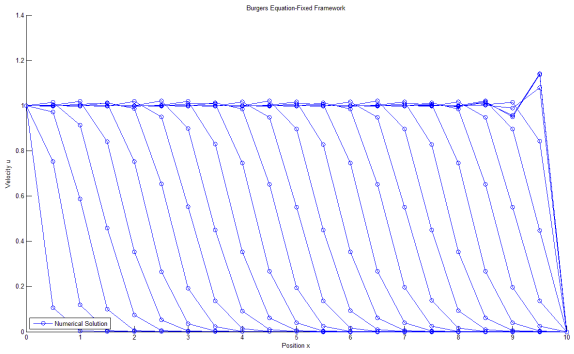



Figure 2.22: Velocity v/s Position, Numerical Soln, Pe=2

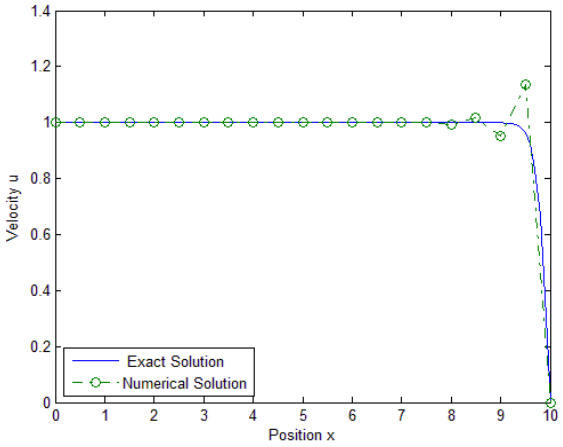



Figure 2.23: Comparison of Numerical with Exact Solution, Pe=2

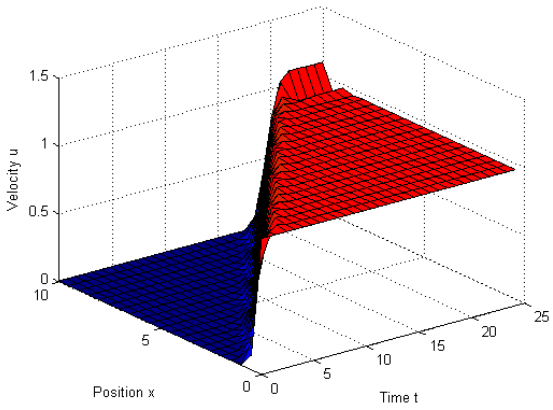



Figure 2.24: Surface Plot, Pe=2

*Standard Galerkin in space and Forward Euler in time, Pe = 2*

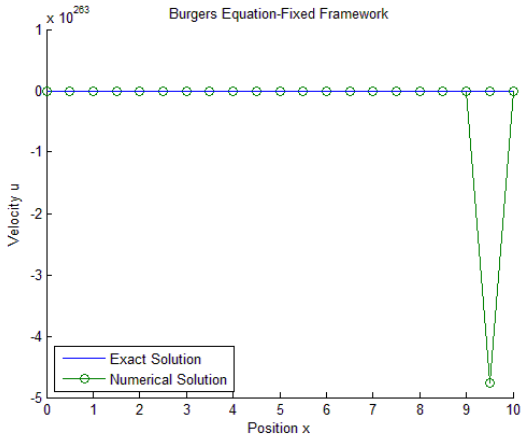



Figure 2.25: Velocity v/s Position, Numerical Soln, Pe=2

*Petrov Galerkin in space and Backward Euler in time, Pe = 0.5*

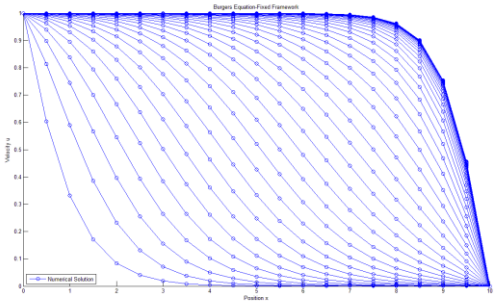


Figure 2.26: Velocity v/s Position, Numerical Soln, Pe=0.5

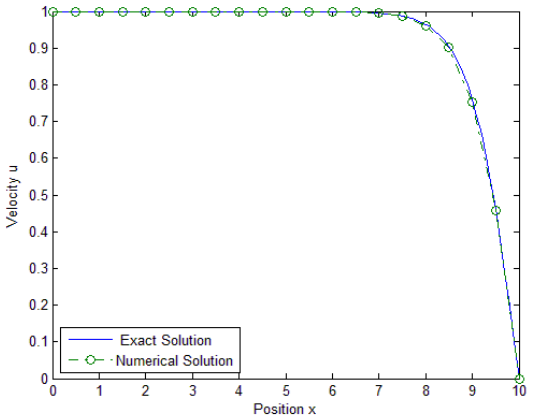



Figure 2.27: Comparison of Numerical with Exact Solution, Pe=0.5

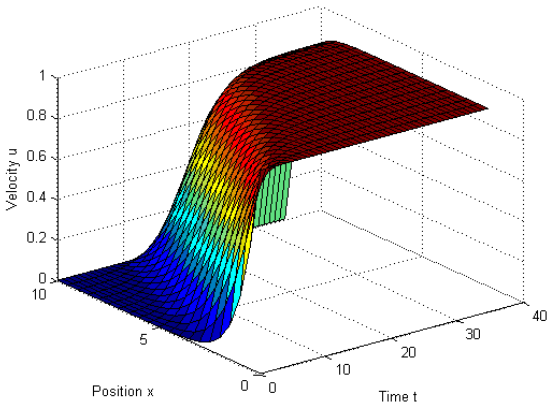



Figure 2.27: Surface Plot, Pe=0.5

*Petrov Galerkin in space and Backward Euler in time, Pe = 1*

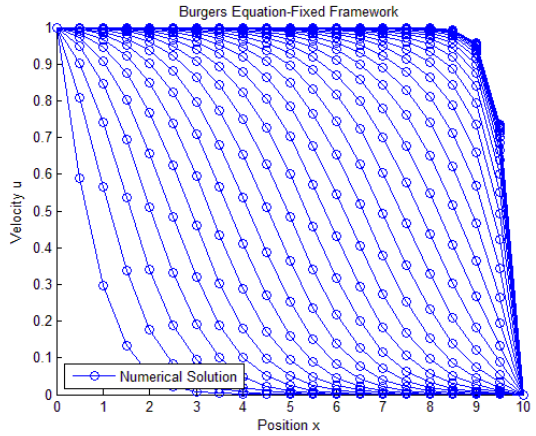



Figure 2.29: Velocity v/s Position, Numerical Soln, Pe=1

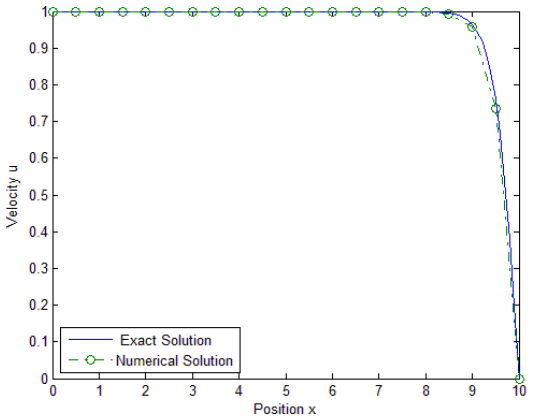



Figure 2.30: Comparison of Numerical with Exact Solution, Pe=1

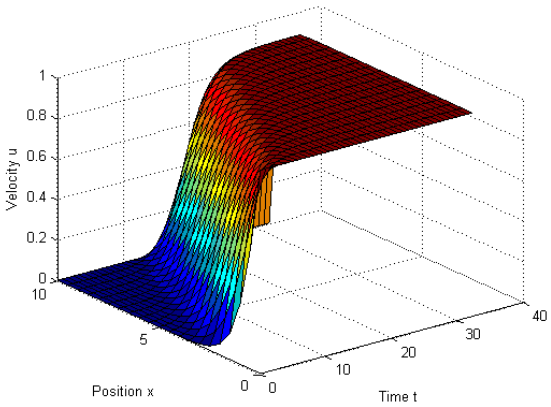



Figure 2.31: Surface Plot, Pe=1

*Petrov Galerkin in space and Backward Euler in time, Pe = 2*

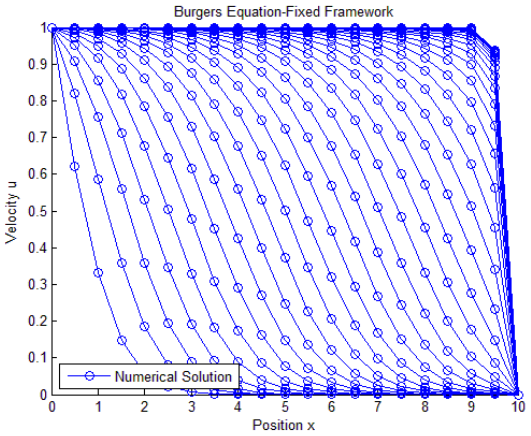



Figure 2.32: Velocity v/s Position, Numerical Soln, Pe=2

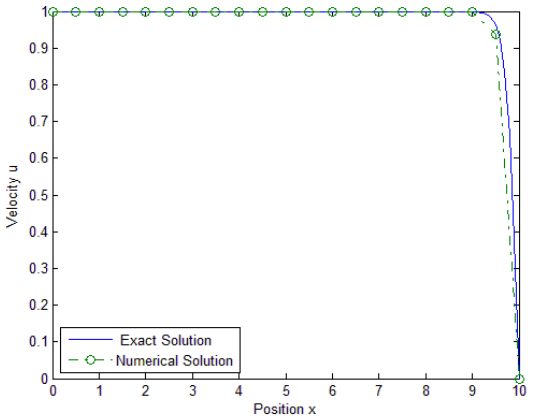



Figure 2.33: Comparison of Numerical with Exact Solution, Pe=2

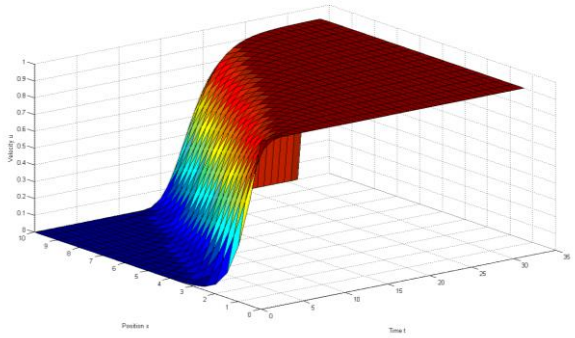


Figure 2.33: Surface Plot, Pe=2

*Petrov Galerkin in space and Crank Nicholson in time, Pe = 0.5*

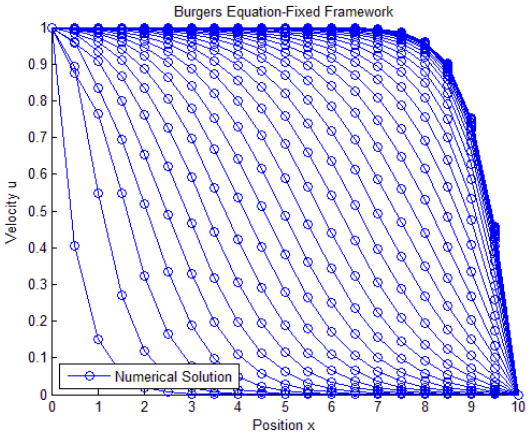



Figure 2.34: Velocity v/s Position, Numerical Soln, Pe=0.5

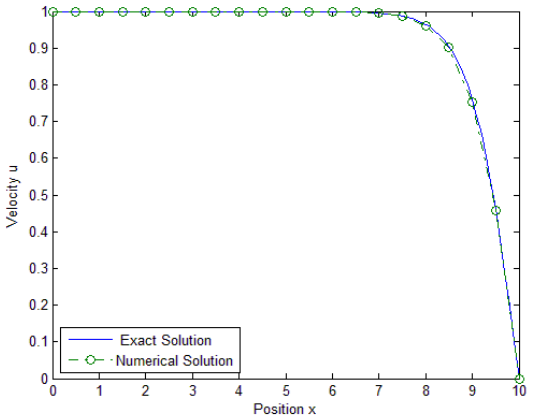



Figure 2.35: Comparison of Numerical with Exact Solution, Pe=0.5

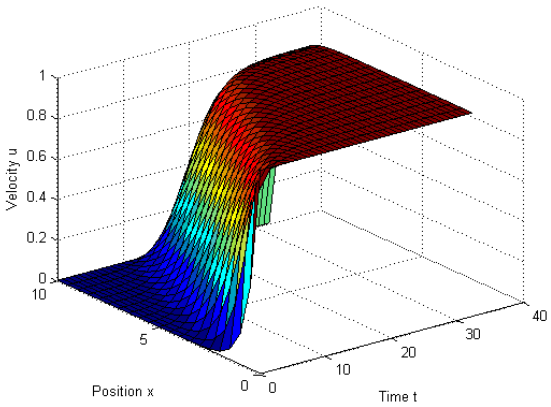



Figure 2.36: Surface Plot, Pe=0.5

*Petrov Galerkin in space and Crank Nicholson in time, Pe = 1*

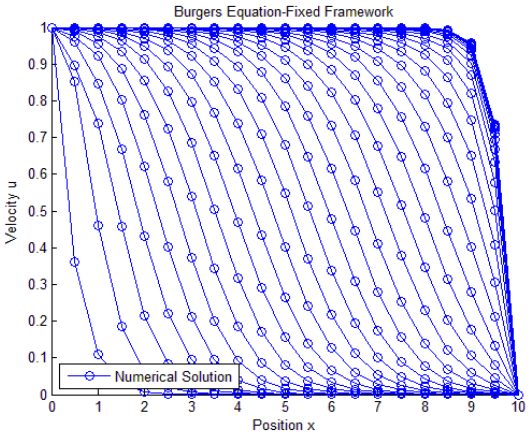



Figure 2.37: Velocity v/s Position, Numerical Soln, Pe=1

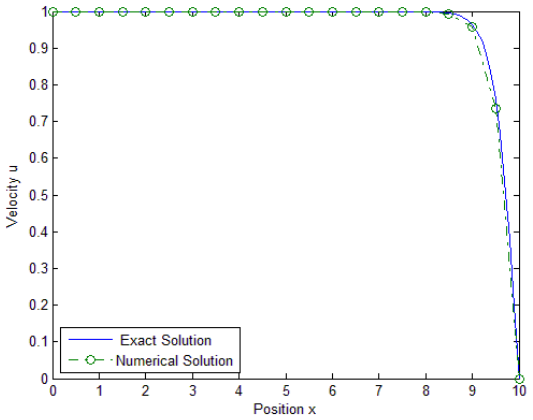



Figure 2.38: Comparison of Numerical with Exact Solution, Pe=1

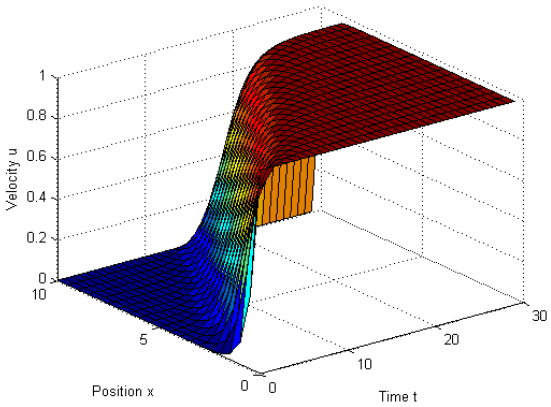



Figure 2.39: Surface Plot, Pe=1

*Petrov Galerkin in space and Crank Nicholson in time, Pe = 2*

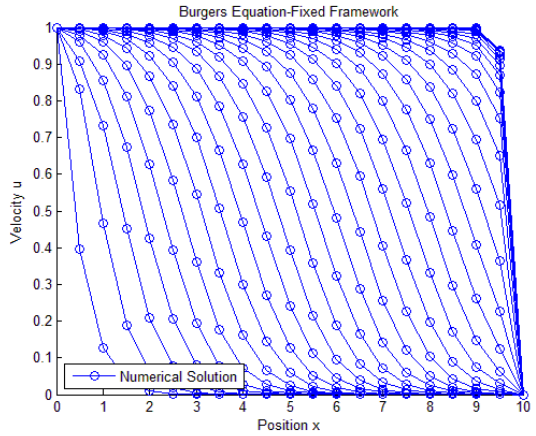



Figure 2.40: Velocity v/s Position, Numerical Soln, Pe=2

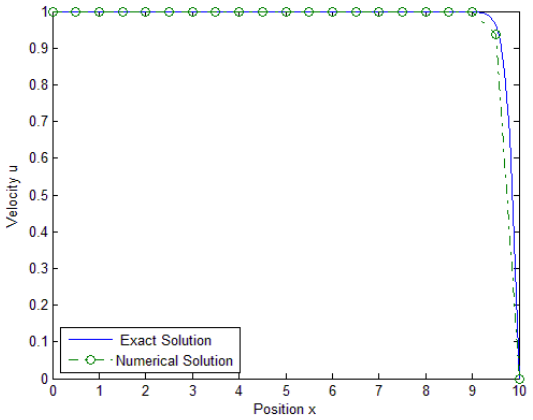



Figure 2.41: Comparison of Numerical with Exact Solution, Pe=2

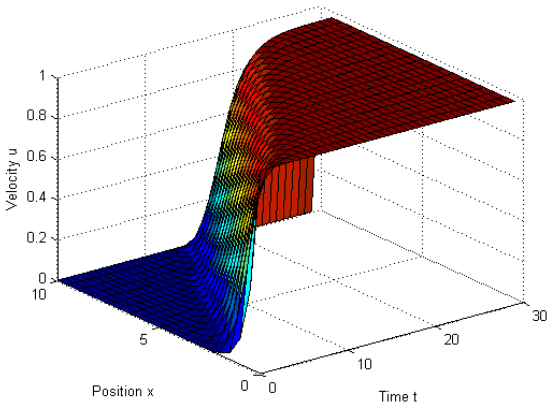



Figure 2.42: Surface Plot, Pe=2

*Convergence of the method*

It important to understand the convergence of the Newton-Raphson scheme to see that the implementation of the method is employed very well and the solution converges quickly. The Newton- Raphson method is well known by exhibiting the quadratic convergence if the initial approximation lies in a close vicinity of the exact solution.

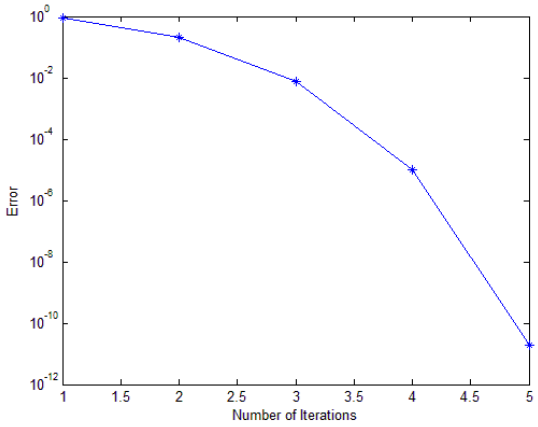



Figure 2.43: Convergence Plot: Newton Raphson

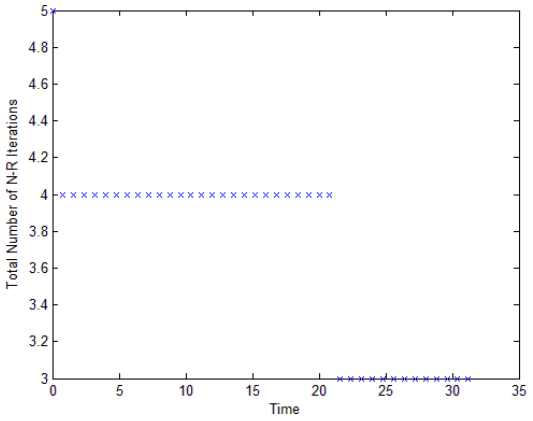



Figure 2.44: Total Number of Iterations v/s Time

*Stability*

We have observed that, the Forward Euler method is unconditionally unstable; this scheme will not produce any solution at any circumstances. Thus, it is essential is avoid such schemes. The instability is developed by the negative dissipation added by the discretisation. In comparison with the Backward Euler and Crank Nicholson methods are more accurate and also unconditionally stable.

*Discussion*

We can see a distinct stabilization when we increase the Pectlet number beyond one. The biggest advantage of Petrov-Galerkin method over traditional Galerkin is that the method will give stable solutions even for Peclet number greater than 1. We can see clearly the oscillation and hence the unstable solution when the Pe number is higher than one for Galerkin method.

The poor performance of Galerkin finite element method for convection dominated transport has a remedy by the use of Petrov-Galerkin method, where the weighting functions are a combination of global interpolation function and their derivatives. This modification biases the resulting equations in a particular direction and gives us overall stability.

# Chapter 3

## Burgers Equation: Moving Mesh Framework

The work in this chapter is concerned with the simulation of incompressible Newtonian fluid flow problems. The computational framework is based on the stabilized velocity finite element method. The movement and deformation of the computational domain are accounted by employing the arbitrary Lagrangian- Eulerian (ALE) [69] strategy. A partitioned procedure is based on the Newton- Raphson methodology is incorporated for full linearization of the problem.

## 3.1 Incompressible Newtonian fluid flow on a moving domain

The algorithms of continuum mechanics make use of three distinct types of description of motion: the Lagrangian description, the Eulerian description and the ALE description. *Lagrangian algorithms,* in which each individual node of the computational mesh follows the associated material particle during motion, are mainly used in structural mechanics. The Lagrangian description allows easy tracking of free surfaces and interfaces between different materials. Its weakness is its inability to follow large distortions of the computational domain without recourse to frequent remeshing operations. *Eulerian algorithms* are widely used in fluid mechanics. Here, the computational mesh is fixed and the fluid moves with respect to the grid. The Eulerian formulation facilitates the treatment of large distortions in the fluid motion and is indispensable for the simulation of turbulent flows. Its handicap is the difficulty to follow free surfaces and interfaces between different materials or different media (e.g., fluid-fluid and fluid-solid interfaces).

*ALE algorithms* are particularly useful in flow problems involving large distortions in the presence of mobile and deforming boundaries. Typical examples are problems describing the interaction between a fluid and a flexible structure and the simulation of metal forming processes. The key idea in the ALE formulation is the introduction of a computational mesh which can move with a velocity independent of the velocity of the material particles. With this additional freedom with respect to the Eulerian and Lagrangian descriptions, the ALE method succeeds to a certain extend in minimizing the problems encountered in the classical kinematical descriptions, while combining at best their respective advantages.

ALE methods were first proposed in the finite difference context where original developments were made, among others, by Noh [69], Trulio [69] and Hirt, Amsden and Cook [69]; The method was subsequently adopted in the finite element context and early applications are to be found in the work of Donea, Fasoli-Stella and Giuliani [69], Belytschko, Kennedy and Schoeberle [69], Belytschko and Kennedy [69] and Hughes, Liu and Zimmermann [69].

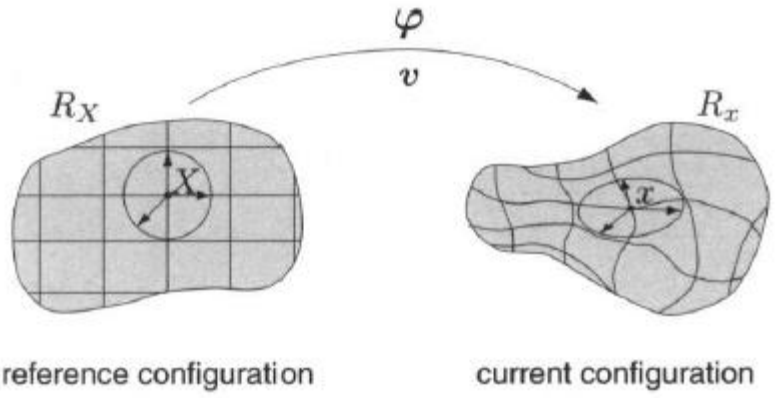



Figure 3.1: Lagrangian description of motion [69]

## 3.2 ALE Kinematics

In the ALE description of motion, neither the material $R_X$ nor the spatial $R_x$ configuration is taken as the reference. Thus, a third domain is needed: the referential configuration $R_{\mathbf{X}}$ where reference coordinates **x** are introduced to identify the grid points.

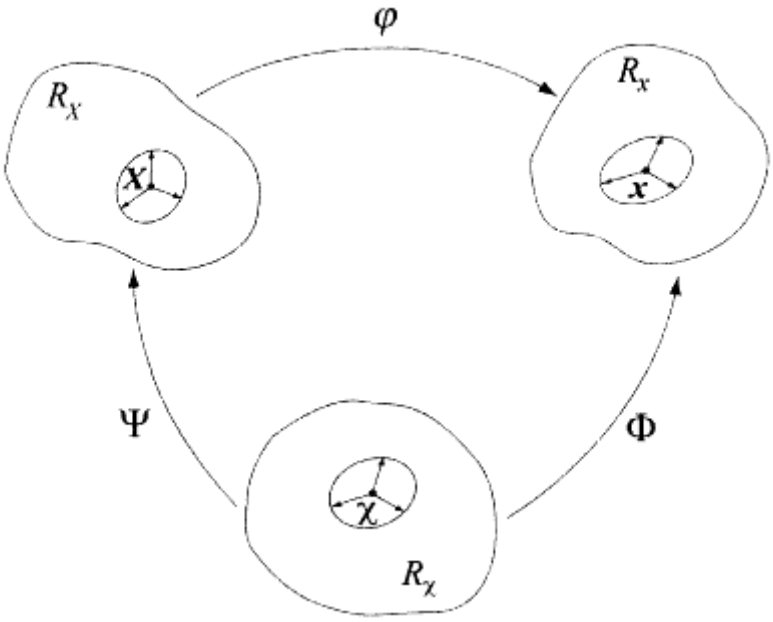



Figure 3.2: The motion of the ALE computational mesh is independent of the material motion [69]

An arbitrary Lagrangian–Eulerian (ALE) description is used to account for the deformation of the fluid domain which arises from the displacement and deformation of the solid structure. Some of the first researchers to demonstrate the potential of this approach are, among others, Hirt et al. [20], Hughes et al. [21], Donea [22], Ramaswamy and Kawahara [23,24], Huerta and Liu [25], Soulaimani et al. [26]. More recent publications are e.g. [27–30]. A related strategy based on the space–time finite element formulation on moving domains has been developed by Tezduyar et al. [31,32], Masud and Hughes [34] and Hansbo [35]. In all the above strategies the movement of the fluid finite element mesh is governed by an appropriate algorithm, thus maintaining a good mesh quality despite substantial deformation of the fluid domain.

## 3.3 Mathematical Model

Burgers equation is a one-dimensional case with ALE of the Navier-Stokes momentum equation, when the pressure gradient and the forcing term are neglected, and is given as follows.

$$\frac{\partial u}{\partial t} + (u - \hat{v})\frac{\partial u}{\partial x} - \mu \frac{\partial^2 u}{\partial x^2} = 0 \qquad \forall \quad 0 \le x \le L \tag{22}$$

The velocity difference $u - \hat{v}$ is denoted as the convective velocity. In the framework of the finite element method, the moving reference frame is identified with the finite element mesh. The Eulerian or Lagrangian representations of the material time derivative of u are easily recovered by setting $\hat{v} = 0$ or $\hat{v} = u$, respectively.

## 3.4 Finite element formulation

The formulation used in this work to discretise in space is similar to the one employed in equation (5, 12, 16), but has been extended to incorporate modifications required in the ALE framework.

For Standard Galerkin Method:

$$\int_\Omega u_j \frac{\partial N_i}{\partial t} dx + \int_\Omega N_1 N_i (u_1 - \hat{v}) \frac{\partial u}{\partial x} u_j dx + \int_\Omega N_2 N_i (u_2 - \hat{v}) \frac{\partial u}{\partial x} u_j dx$$
$$+ \mu \int_\Omega \frac{\partial N_i}{\partial x} \cdot \frac{\partial N_j}{\partial x} u_j dx = 0$$

(23)

For Petrov Galerkin Method:

$$\int_\Omega u_j \frac{\partial N_i}{\partial t} dx + \int_\Omega \left( N_i + \alpha \frac{h}{2} \cdot \frac{\partial N_i}{\partial x} \right) N_1 (u_1 - \hat{v}) \frac{\partial u}{\partial x} u_j dx$$
$$+ \int_\Omega \left( N_i + \alpha \frac{h}{2} \cdot \frac{\partial N_i}{\partial x} \right) N_2 (u_2 - \hat{v}) \frac{\partial u}{\partial x} u_j dx$$
$$+ \mu \int_\Omega \left( \frac{\partial N_i}{\partial x} + \alpha \frac{h}{2} \cdot \frac{\partial^2 N_i}{\partial x^2} \right) \frac{\partial N_j}{\partial x} u_j dx = 0 \qquad (24)$$

## 3.5 Temporal Discretization

We have provided with comparison of explicit and implicit time intergartion schemes with respect to the Newtonian free surface flow. The generic formulation called as the *θ-method* is employed and numerical results are obtained for $\theta = 0$ and $\theta = 1$ (the explicit and implicit Euler methods), and $\theta = 1/2$ (the Crank-Nicolson method). This is same as the equation (16) used for time discretisation of the burgers equation in a fixed domain.

$$\frac{u^{n+1} - u^n}{\Delta t_n} = \theta \ell(u^{n+1}) + (1-\theta) \ell(u^n)$$

## 3.6 Mesh Update

In this method the mesh is neither connected to the material as in the case of Lagrangian nor fixed to the spatial coordinates system as in the case of Eulerian method. Here the mesh is prescibed in a arbitary manner by defining the mesh velocity.

Due to this the remapping of the state variables becomes necessary and is one of the important step along with the computation of mesh velocity in ALE approch. The work of Ramaswamy and Kawahara [23] describes the detailed applicaion of ALE for free surface flow using FEM. A simple midpoint scheme is carried out in this work for the mesh update.

$$u_{n+1} = \frac{1}{\Delta t}(x_{n+1} - x_n) - v_n \tag{25}$$

## 3.7 Numerical Results

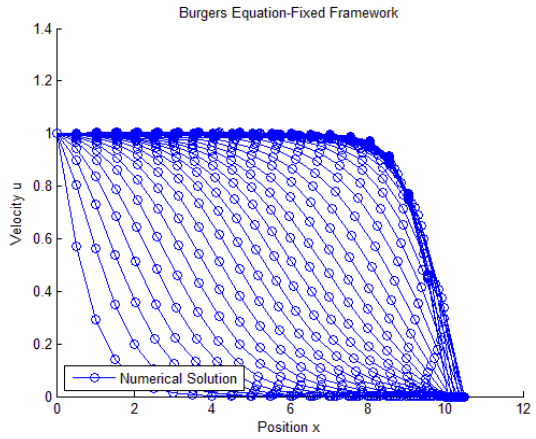
In order to numerically solve fluid flow problems with displacements of the boundaries, one of the most attractive approaches is the so-called arbitrary Lagrangian–Eulerian (ALE) formulation. It is specifically suitable for the fluid–structure interaction problems because the moving mesh of the fluid domain can be attached to the structural boundary and the compatibility between the fluid and the solid domains can be conveniently handled. Various computational algorithms have been introduced within the ALE formulation, ranging from the explicit to the fully implicit algorithms. Here, we overview some of the typical algorithms and briefly point out their specific features relevant to our study.

One important observation, which can be made here is that, the Burger's equation analogue to 1-D fluid flows which is solved without and with FE mesh motion, shows that the mesh motion practically does not affect the solutions. The numerical model is developed for simulation of incompressible viscous free surface flow.

As in the case of Burgers equation in fixed mesh framework, here also we carried numerical analysis with Standard Galerkin and Petrov Galerkin method discretisation in space and implicit and explicit method in time and each one of the methods described for three cases of Peclet number. First two case with Pe < 1 and another case with Pe > 1.

The ALE formulation is employed and we make a initial guess of the mesh velocity. The analysis is performed with the same three cases used in fixed frame work of Burgers equation with a kinematic viscosity $\mu = 0.5, 0.25, 0.125$ and number of elements $n = 20$ resulting in Pe = 0.5, 1, 2. The results obtained for Galerkin finite element method for cases 1, 2 and 3 are shown in Fig. below.

*Standard Galerkin in space and Backward Euler in time, Pe = 0.5*




3.3: Velocity v/s Position, Numerical Soln, Pe=0.5

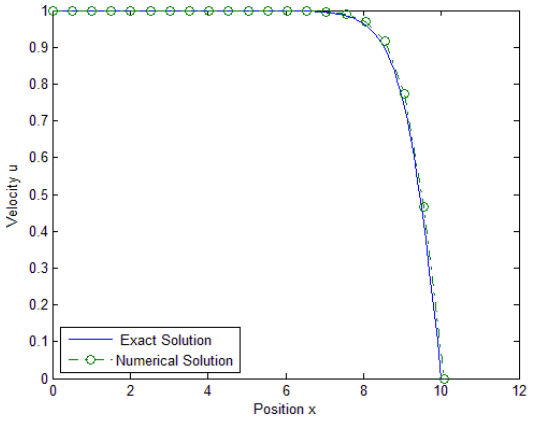



Figure 3.4: Comparison of Numerical with Exact Solution, Pe=0.5

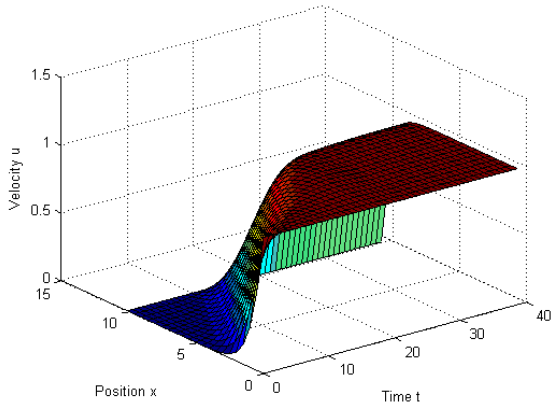



Figure 3.5: Surface Plot, Pe=0.5

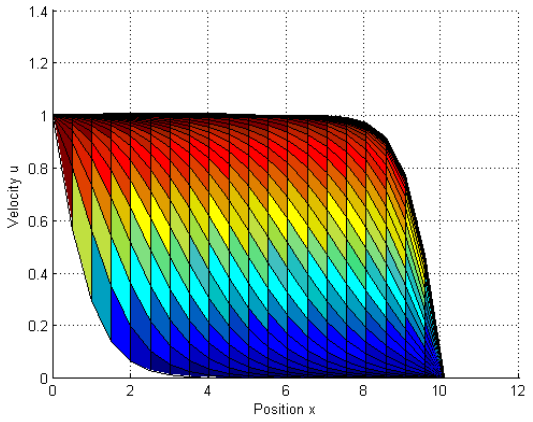
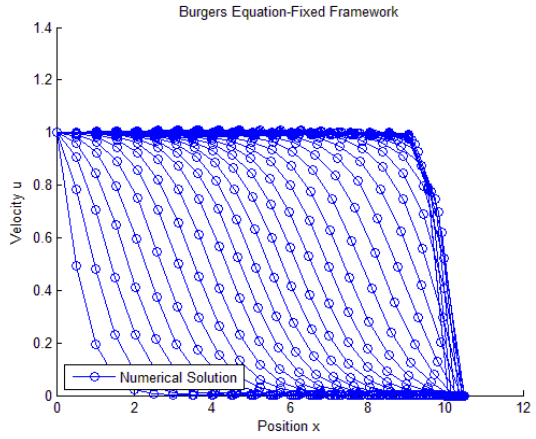



Figure 3.6: Surface Plot XZ view, Pe=0.5

*Standard Galerkin in space and Backward Euler in time, Pe = 1*




3.7: Velocity v/s Position, Numerical Soln, Pe=1

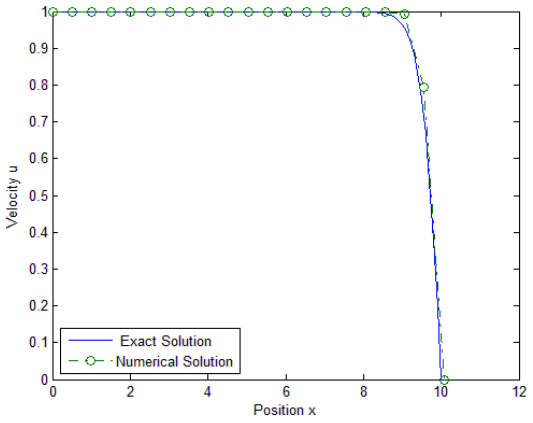



Figure 3.8: Comparison of Numerical with Exact Solution, Pe=1

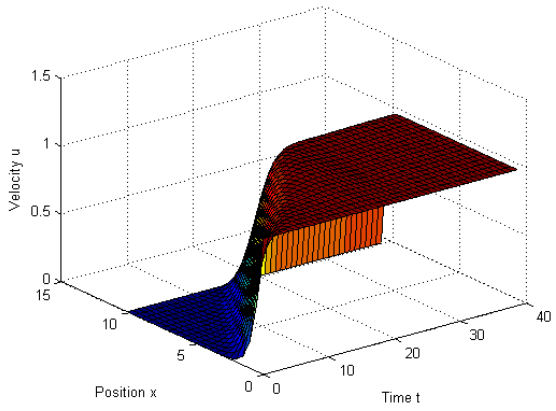



Figure 3.9: Surface Plot, Pe=1

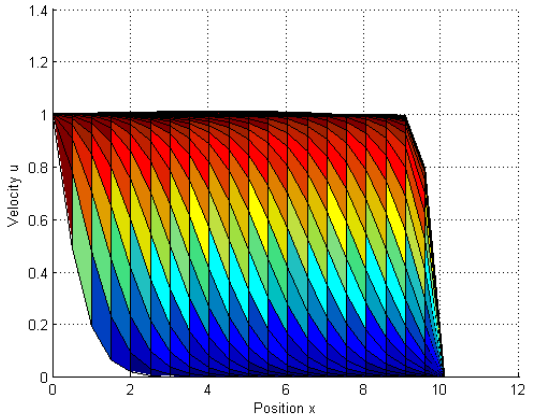
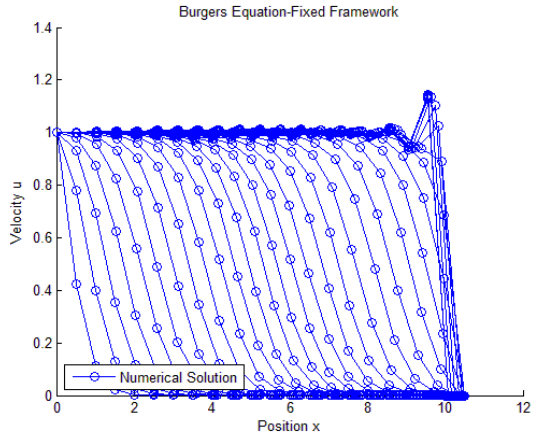



Figure 3.10: Surface Plot XZ view, Pe=1

*Standard Galerkin in space and Backward Euler in time, Pe = 2*




3.11: Velocity v/s Position, Numerical Soln, Pe=2

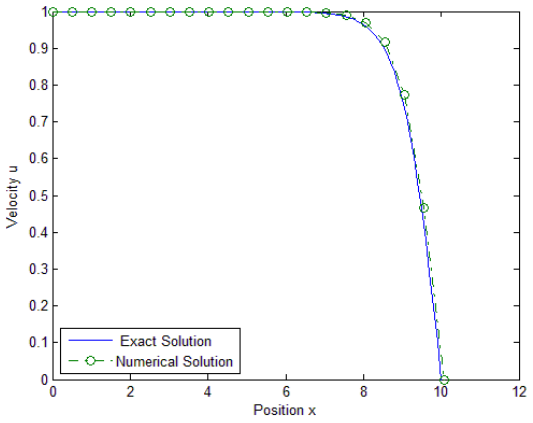



Figure 3.12: Comparison of Numerical with Exact Solution, Pe=2

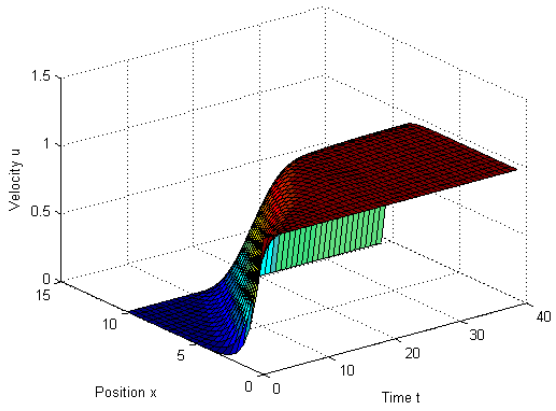



Figure 3.13: Surface Plot, Pe=2

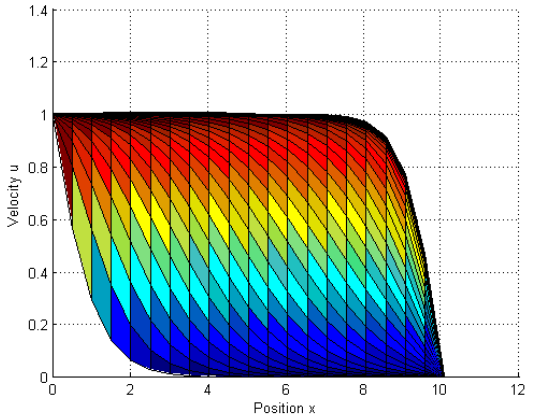
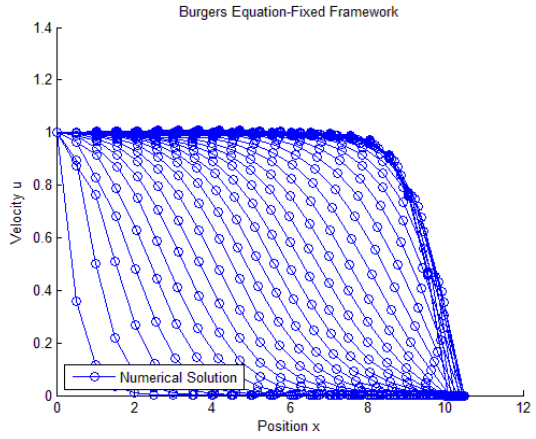



Figure 3.14: Surface Plot XZ view, Pe=2

*Standard Galerkin in space and Crank Nicholson in time, Pe = 0.5*




3.15: Velocity v/s Position, Numerical Soln, Pe=0.5

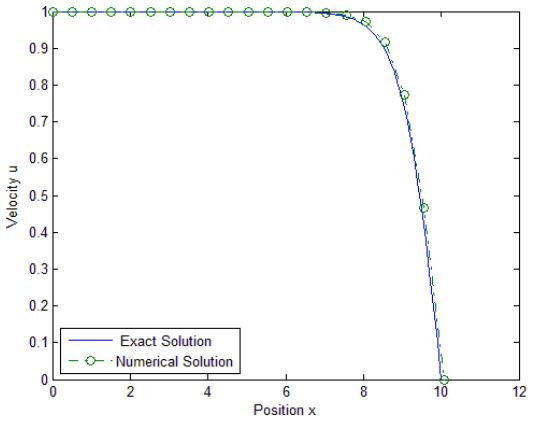



Figure 3.16: Comparison of Numerical with Exact Solution, Pe=0.5

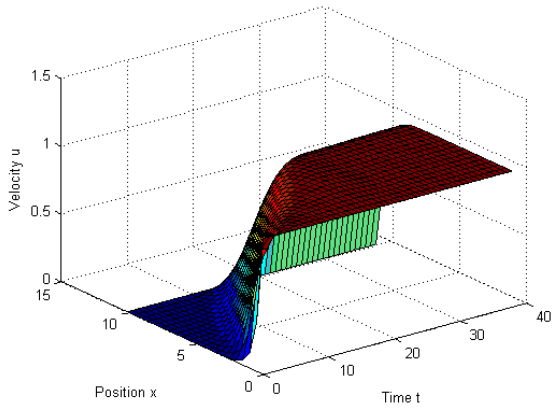



Figure 3.17: Surface Plot, Pe=0.5

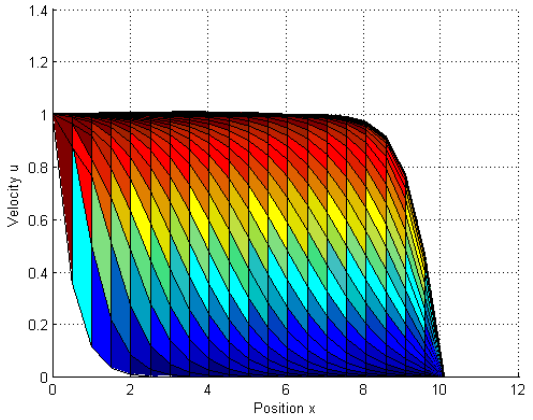
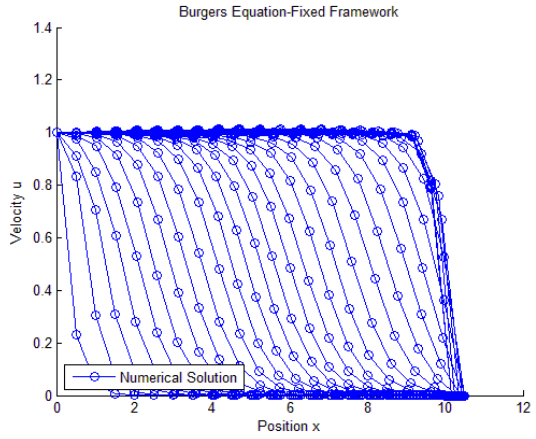



Figure 3.18: Surface Plot XZ view, Pe=0.5

*Standard Galerkin in space and Crank Nicholson in time, Pe = 1*




3.19: Velocity v/s Position, Numerical Soln, Pe=1

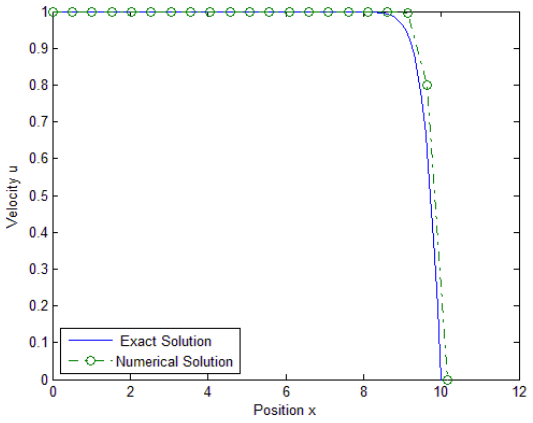



Figure 3.20: Comparison of Numerical with Exact Solution, Pe=1

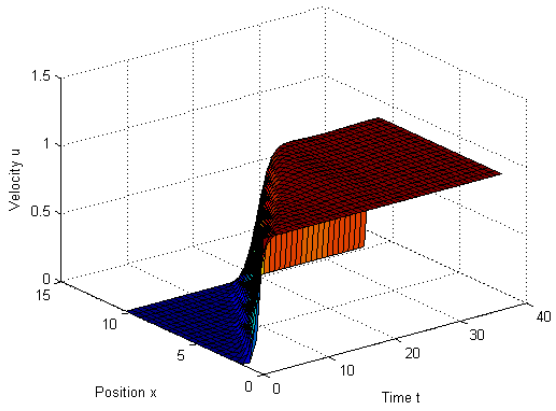



Figure 3.21: Surface Plot, Pe=1

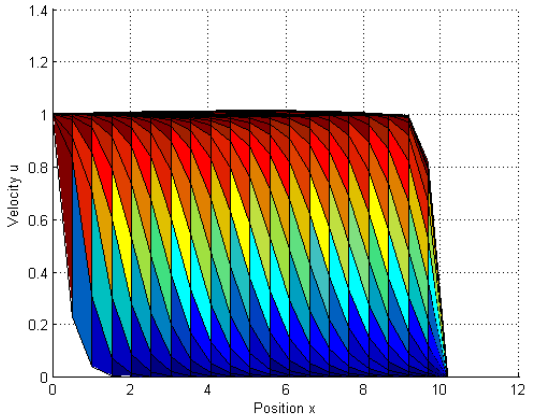
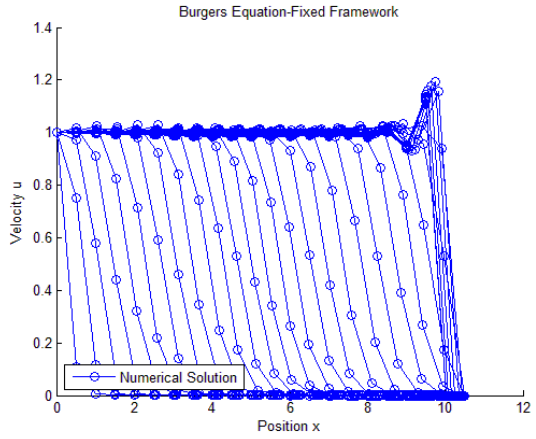



Figure 3.22: Surface Plot XZ view, Pe=1

*Standard Galerkin in space and Crank Nicholson in time, Pe = 2*




3.23: Velocity v/s Position, Numerical Soln, Pe=2

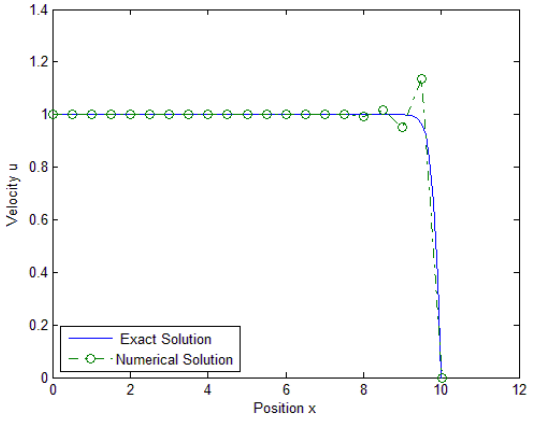



Figure 3.24: Comparison of Numerical with Exact Solution, Pe=2

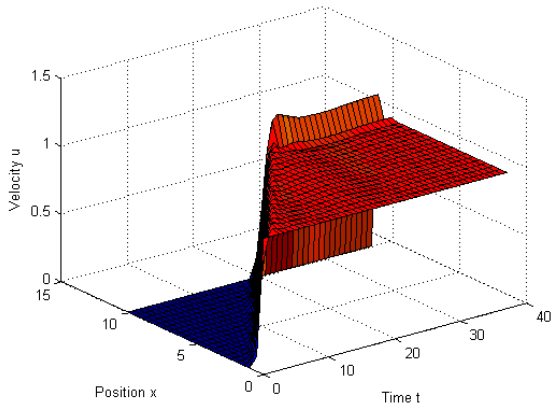



Figure 3.25: Surface Plot, Pe=2

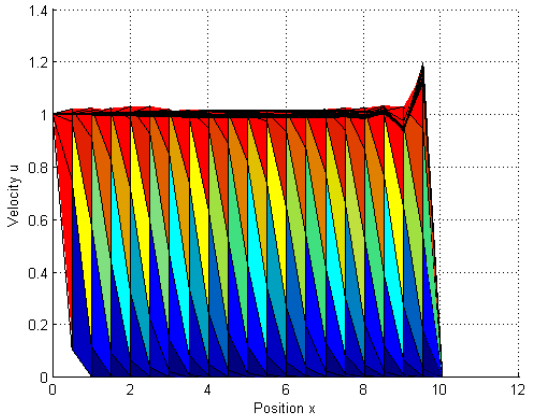
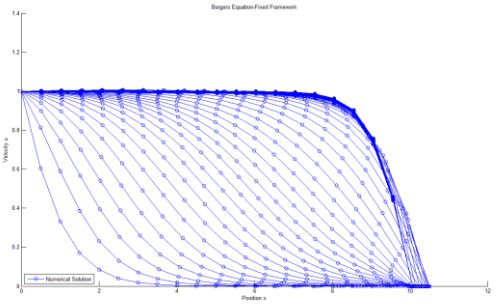



Figure 3.26: Surface Plot XZ view, Pe=2

*Petrov Galerkin in space and Backward Euler in time, Pe = 0.5*



3.27: Velocity v/s Position, Numerical Soln, Pe=0.5

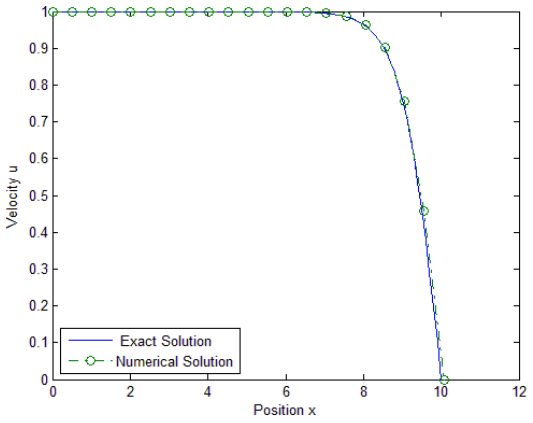



Figure 3.28: Comparison of Numerical with Exact Solution, Pe=0.5

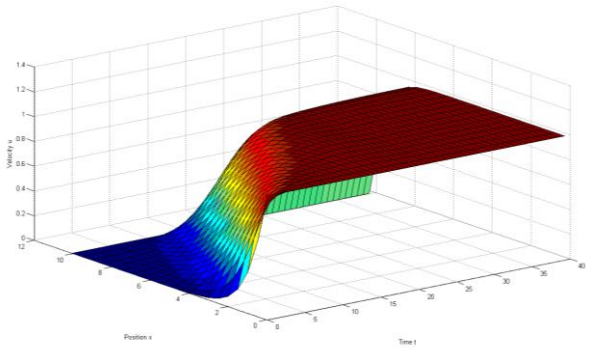


Figure 3.29: Surface Plot, Pe=0.5

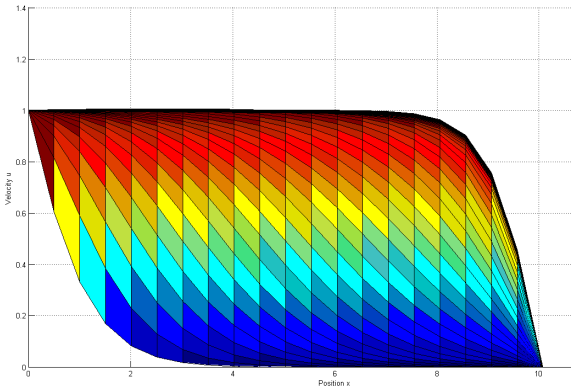
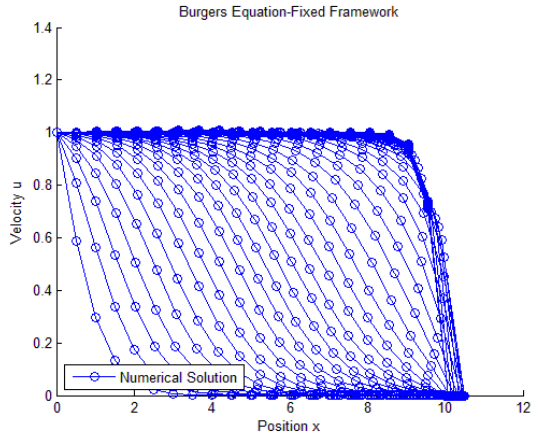


Figure 3.30: Surface Plot XZ view, Pe=0.5

*Petrov Galerkin in space and Backward Euler in time, Pe = 1*




3.31: Velocity v/s Position, Numerical Soln, Pe=1

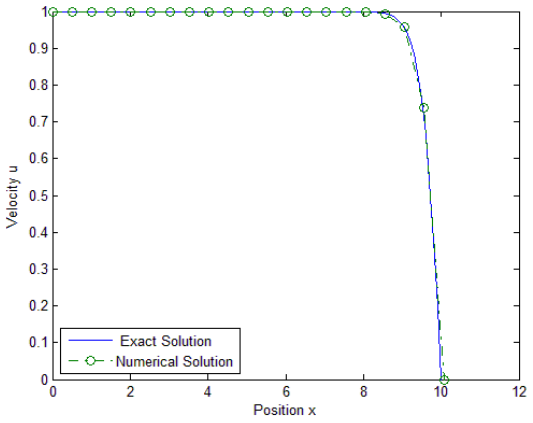



Figure 3.32: Comparison of Numerical with Exact Solution, Pe=1

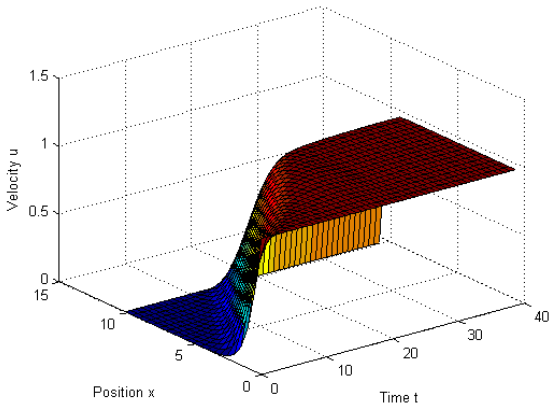



Figure 3.33: Surface Plot, Pe=1

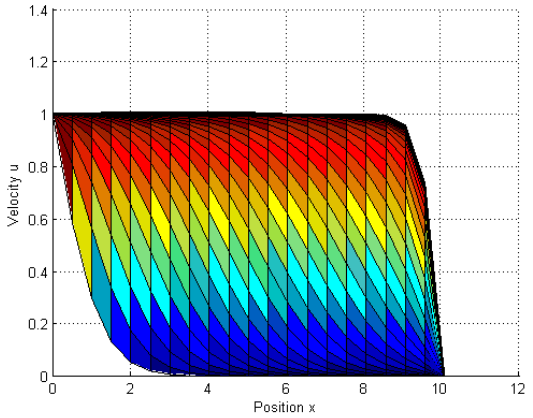
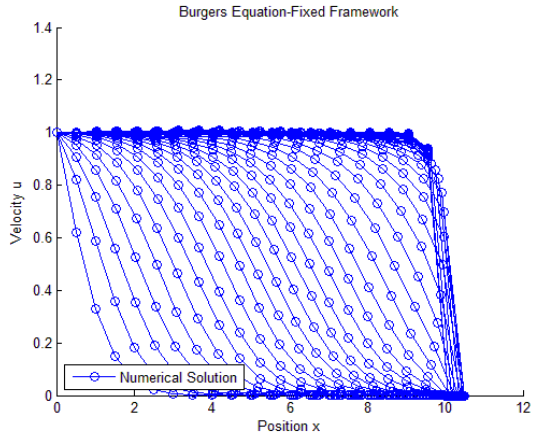



Figure 3.34: Surface Plot XZ view, Pe=1

*Petrov Galerkin in space and Backward Euler in time, Pe = 2*




3.35: Velocity v/s Position, Numerical Soln, Pe=2

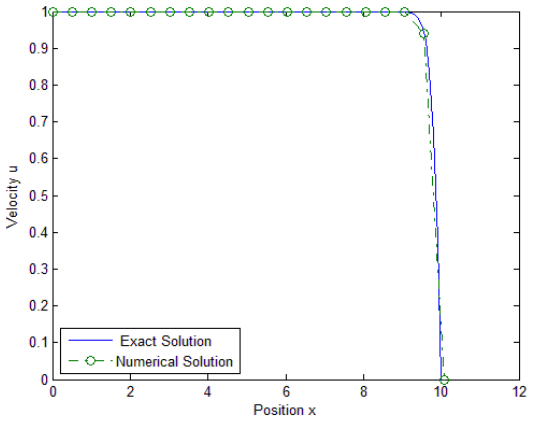



Figure 3.36: Comparison of Numerical with Exact Solution, Pe=2

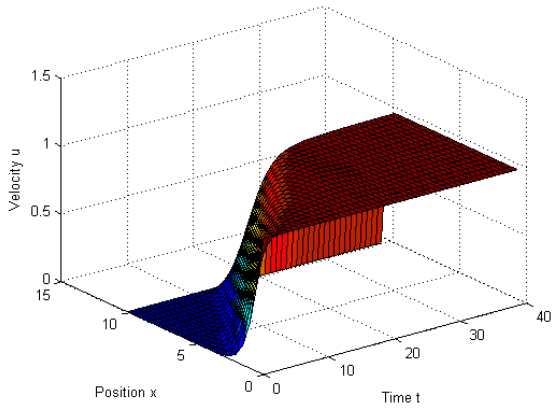



Figure 3.37: Surface Plot, Pe=2

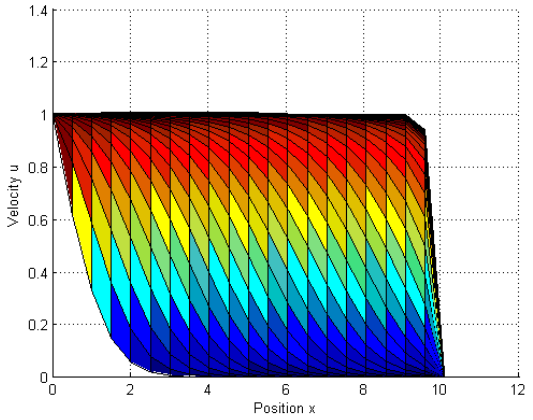
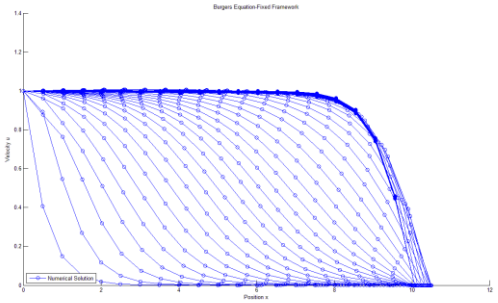



Figure 3.38: Surface Plot XZ view, Pe=2

*Petrov Galerkin in space and Crank Nicholson in time, Pe = 0.5*



3.39: Velocity v/s Position, Numerical Soln, Pe=0.5

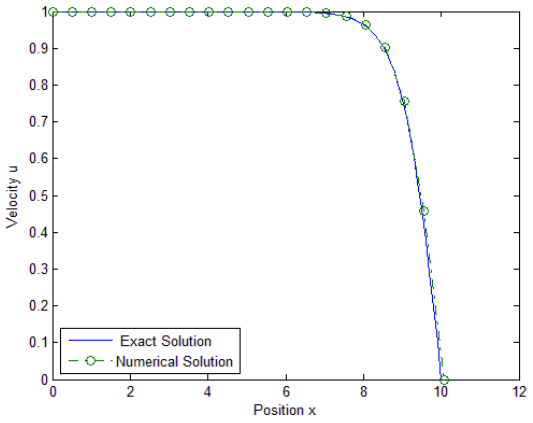



Figure 3.40: Comparison of Numerical with Exact Solution, Pe=0.5

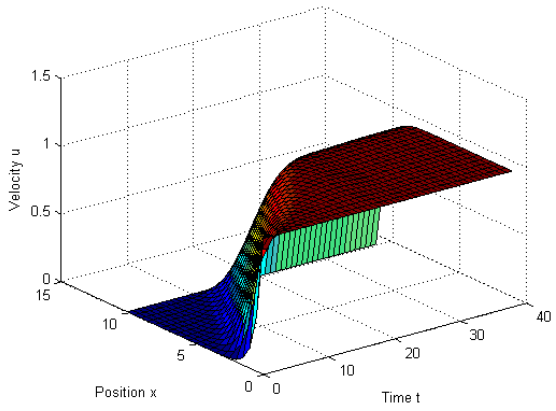



Figure 3.41: Surface Plot, Pe=0.5

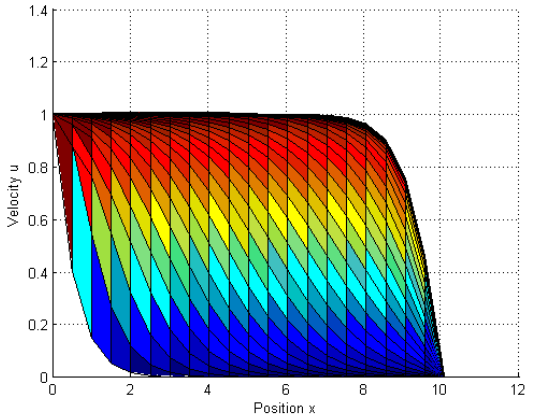
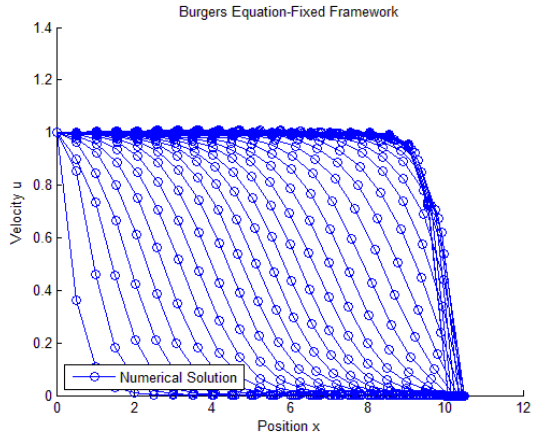



Figure 3.42: Surface Plot XZ view, Pe=0.5

*Petrov Galerkin in space and Crank Nicholson in time, Pe = 1*




3.43: Velocity v/s Position, Numerical Soln, Pe=1

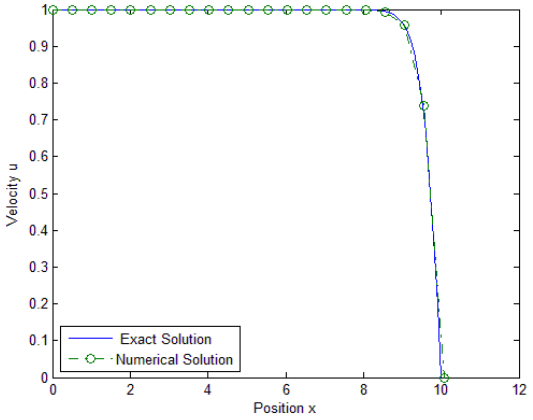



Figure 3.44: Comparison of Numerical with Exact Solution, Pe=1

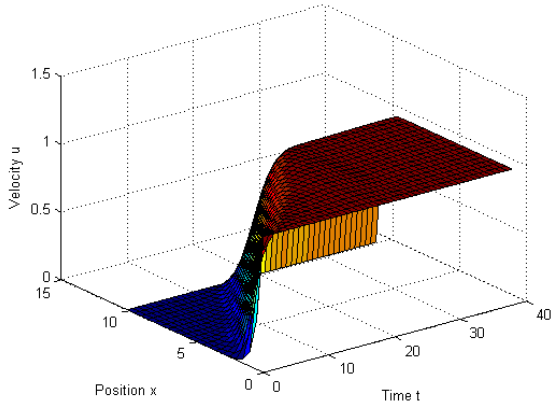



Figure 3.45: Surface Plot, Pe=1

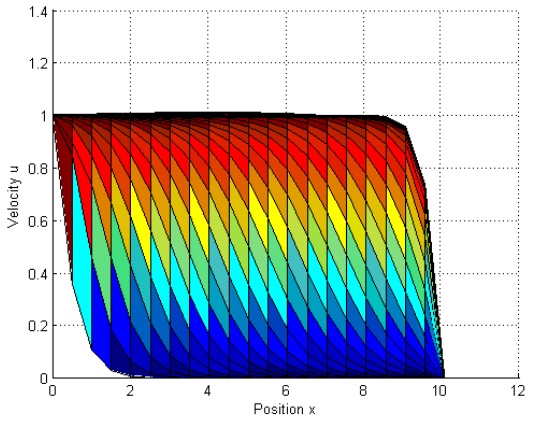
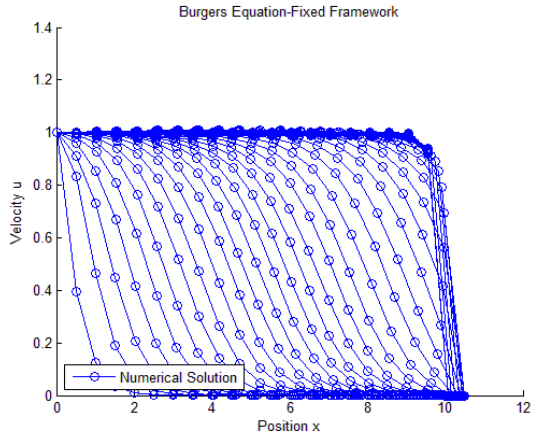



Figure 3.46: Surface Plot XZ view, Pe=1

*Petrov Galerkin in space and Crank Nicholson in time, Pe = 2*




3.47: Velocity v/s Position, Numerical Soln, Pe=2

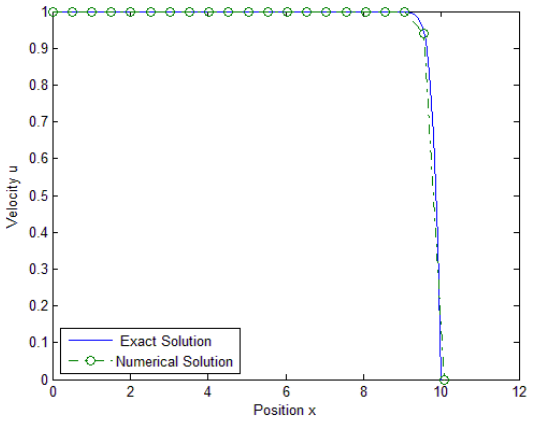



Figure 3.48: Comparison of Numerical with Exact Solution, Pe=2

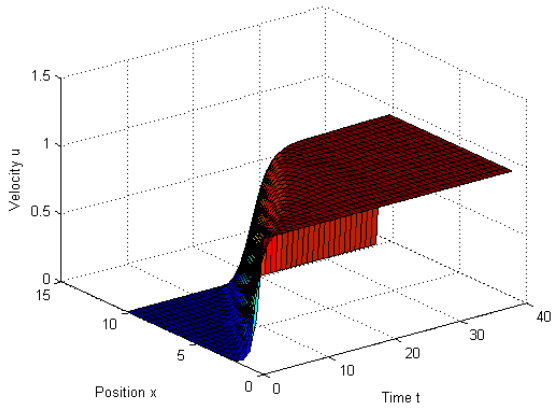



Figure 3.49: Surface Plot, Pe=2

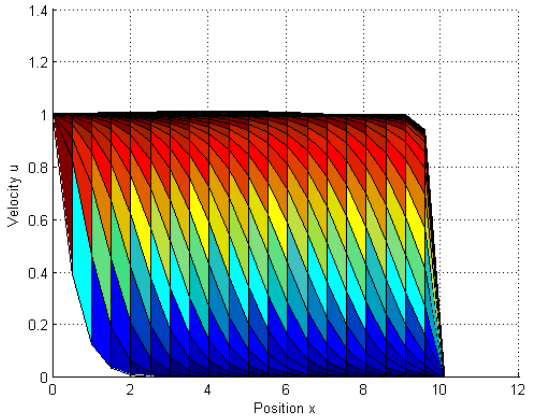



Figure 3.50: Surface Plot XZ view, Pe=2

*Convergence of the method*

We can deduce that the Petrov-Galerkin method also exhibits quadratic convergence, which is easily observed by looking at the cases considered above. Furthermore, we can conclude from the results presented in this section that the Petrov-Galerkin scheme with the Newton-Raphson procedure provides desired, numerical, stability regardless of the number of elements used in finite element discretisation. The oscillations are eliminated and numerical accuracy is improved.

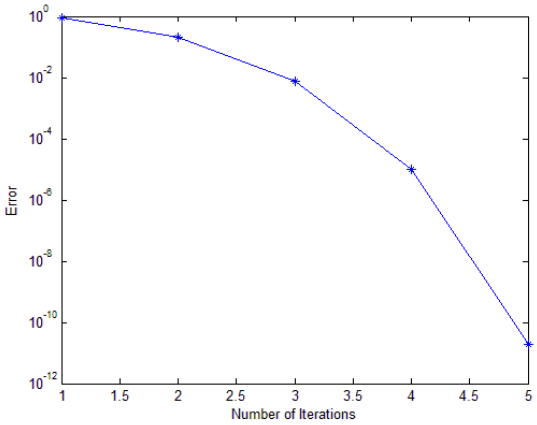



Figure 3.51: Convergence Plot: Newton Raphson

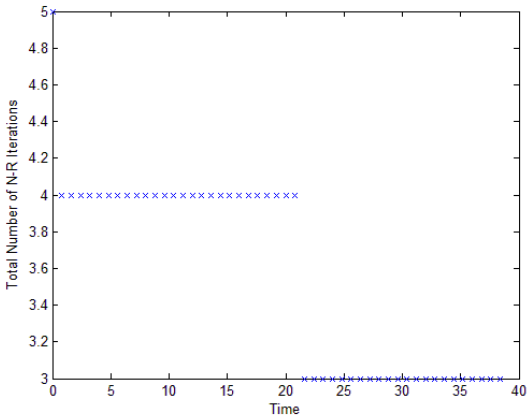



Figure 3.52: Total Number of Iterations v/s Time

*Stability*

As found in the case of fixed mesh framework, here also we have observed that, the Forward Euler method is unconditionally unstable; this scheme will not produce any solution at any circumstances. Thus, it is essential is avoid such schemes. The instability is developed by the negative dissipation added by the discretisation. In comparison with the Backward Euler and Crank Nicholson methods are more accurate and also unconditionally stable.

*Discussion*

An implicit algorithm for incompressible fluid flow solution using the arbitrary Lagrangian–Eulerian (ALE) formulation is employed to investigate solution accuracy and stability of the 1-D Burgers equation. Accuracy and stability of the solutions are demonstrated in example for which the analytical solutions are known. In the example, the Burger's equation analogue to 1-D fluid flows is solved without and with FE mesh motion, to show that the mesh motion practically does not affect the solutions and this also is the set as a benchmark problem for study of free surface flow problem. All solutions presented show that the proposed algorithm is sufficiently accurate and stable. Since the algorithm is implicit, high accuracy of results can be achieved with a relatively large time step. As observed in the case of fixed mesh, the solution is not stable for Pe > 1 for Galerkin method and we use the Petrov Galerkin method to obtain stable results.

# Chapter 4

## Dynamics of Liquid Sloshing

A shock/hydraulic jump in a free surface can have a major impact and have a high impact on the structure containing them. It becomes interesting to study the unsteady flows with shock formation which is provided by the time evolution of a sinusoidal wave. Understand the effect of dissipative term on the shock formation. Carry out an instability study to undertand the physical behavior of the burgers equation.

## 4.1 Shock Formation

As discussed in the introduction chapter the storage liquid tanks are usually mounted on the ground in two different ways: unanchored and anchored. Large-sized unanchored tanks with flat bases usually experience different kinds of damage under the action of ground motions. The best known damages are the elephant foot bulge, which takes the form of buckling at the bottom part of the tank, and cracks at the corner of the bottom plate-shell. Both classes are related to uplifting of the bottom plate and thus involve strong nonlinearity due to the associated large displacement and the separation between the bottom plate and foundation for unanchored tanks. The failure and damage of liquid tanks under earthquake excitations have been studied by many civil engineers, Shibata, et al., [57, 58] Shepherd, [59], Hanson, [60], Haroun and Housner, [61-63] Niwa and Clough, [67,68], Haroun and Mourad, [65], and Hatano and Konno [66]. The main area of study was to numerically estimate the hydrodynamic pressure.

When an impulsive acceleration acts on a liquid, this can result in impact hydrodynamic pressure of the free surface on the tank walls. This can also occur during maneuvering or docking of spacecraft in an essentially low gravity field. The methods for estimating these impacts and the hydrodynamic pressure are not well developed yet and they are mainly identified through experimental studies. When the hydraulic jumps or traveling waves are present, it leads to extremely high impact pressures on the tank walls. A hydraulic jump/shock [55] may occur in a liquid container undergoing oscillatory motion if the liquid height is relatively shallow and the
excitation frequency is close to the natural frequency of the free surface.

The hydraulic jump/ shock could create localized high impact pressures on the container walls, which has a direct effect on the container dynamics and may result in structural damage. This hydraulic jump is a nonlinear phenomenon, analogous to the shock wave appearing in one-dimensional gas flow under similar resonance conditions. This movement will have an impact on the stability of the storage tank.

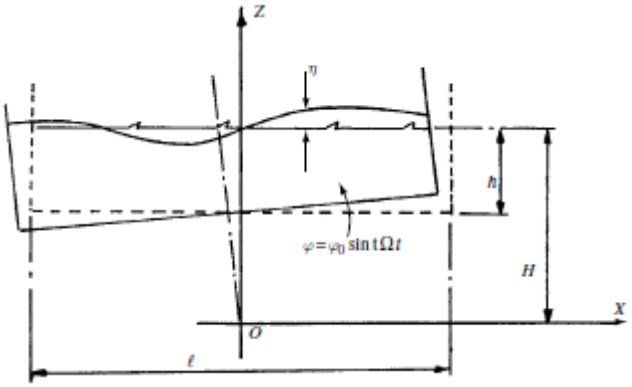



Figure 4.1, Rectangular container partially filled with liquid under pitching excitation [55].

When a tank is subjected to sinusoidal excitation of a liquid tank, a hydrodynamic jump may be formed in the neighborhood of the resonance frequency. The jump usually moves back and forth between the walls of the tank. Verhagen and WijnGaarden [55] employed a theory for one-dimensional gas flow to study the fluid oscillations experiencing a hydraulic jump in a rectangular tank. This theory was originally developed by Betchov [55], Chu and Ying [55], and Chester [55]. Figure shows a schematic diagram of a rectangular container of width $\ell$ filled with liquid to a level h. The container is allowed to oscillate about axis Y, through O, with a small pitching angle $\varphi = \varphi \sin \Omega t$. The breadth of the container in the Y-direction is large enough for the flow to be described as two-dimensional. The undisturbed liquid free surface is located at Z = H, while the surface elevation measured from this level is denoted by $\eta$

## 4.2 Physical Behaviour of Burgers Equation

If the viscous term is dropped from the Burgers' equation the nonlinearity allows discontinuous solutions to develop. The way that this can occur is illustrated schematically in the following figure [54]. A wave is convecting from left to right and solutions for succesive times are indicated. Points on the wave with larger values of u convect faster and consequently overtake parts of the wave convecting with smaller values of u. As seen the top of the wave is moving faster than the root and as a result the top travels much faster. This leads to shock formation. Hence it becomes necessary to postulate a shock accross which u change discontinuously to have a unique solution and so a physically result.

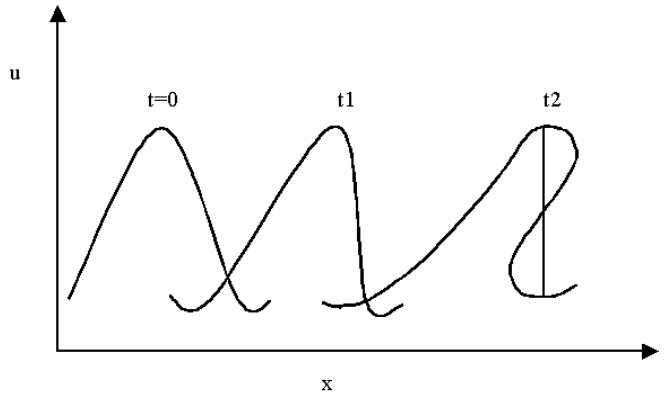



Figure 4..2.1, Shock wave in inviscid flow- a physical representattion [69].

The comparable wave development for the 'viscous' Burgers' equation is shown in the following figure. The effect of the viscous term is twofold. First , it reduces the amplitude of the wave for increasing time. Second, it prevents multivalued solutions from developping.

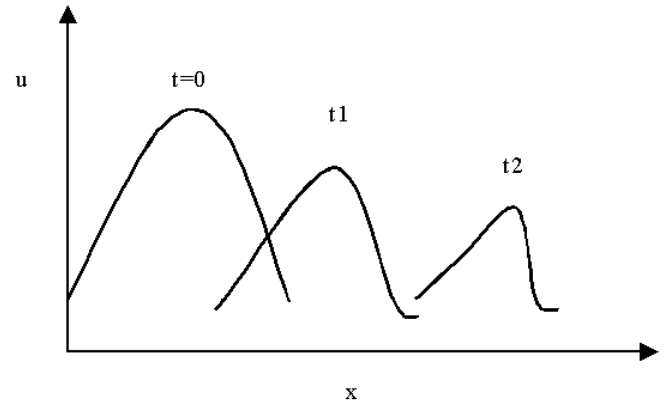



Figure 4.2.2, Viscous flow [69].

The above features make Burgers' equation a very suitable model for testing computational algorithms for flows where severe gradients or shocks are anticipated.

## 4.3 Discretization Methods

We have choose to solve the Burgers' equation using a finite-element technique for the spatial discretisation and for temporal we have used the explicit scheme, which is easy to program but fails to give a correct solution when the viscosity is too low. Indeed stability conditions need to be respected. To avoid unstable solutions, we have programmed an implicit Euler method and also an implicit Crank-Nicholson method [45-52].

## 4.4 Numerical Results

At the beginning of the simulation we disturb the flow with a flow $u = u_0 \sin(2\pi x / l)$, we can observe that that the fastest fluid catches up with the slowest one so that to create a velocity break. This phenomenon is called shock. The equation is discretized in space using Galerkin and in Time with Crank Nicholson method.

It becomes very important to study these phenomena, since this can lead to many disasters. One such problem is sloshing in tank and here with the numerical results we try to understand the effect of the viscous term in non-linear Burgers equation.

Below figures gives the evolution of the solution. Each point propagates with a different speed and leads to the formation of a shock. This shock is then damped owing to the dissipative term. We have obtained the numerical results for a sinusoidal wave profile, with two different amplitude cases and by varying the viscous term for the same.

The shock formation process is seen for four different viscosity values ranging from 0.1 to 0.0001. If the viscosity is too low 0.0001, a wiggle appears and eventually causes the solution to blow up. For lower viscosities, we may have to you some other numerical schemes must be used. By increasing the viscosity, the expected shock fades. We can carry out a instability study by adding others terms to the Burgers' equation. The numerical simulation is carried out with 100 elements.

*Shock formation process with amplitude 0.3*

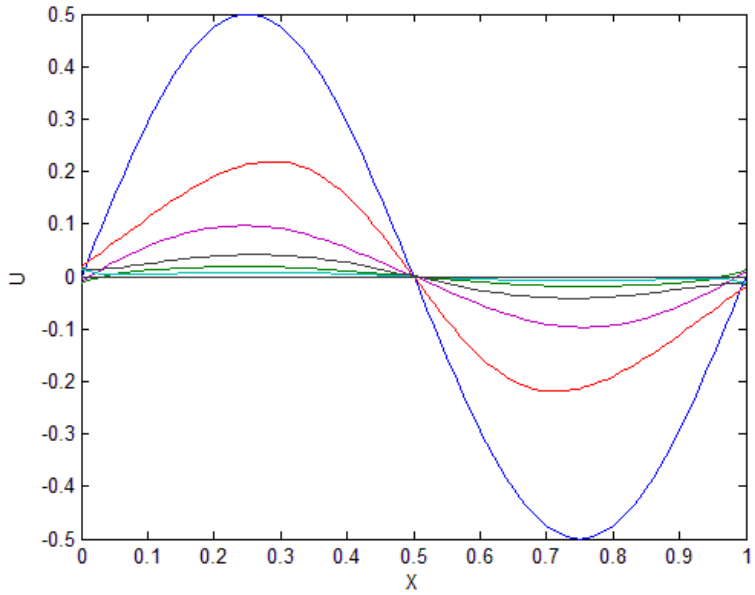



Figure 4.3: Velocity v/s Position, Viscosity=0.1

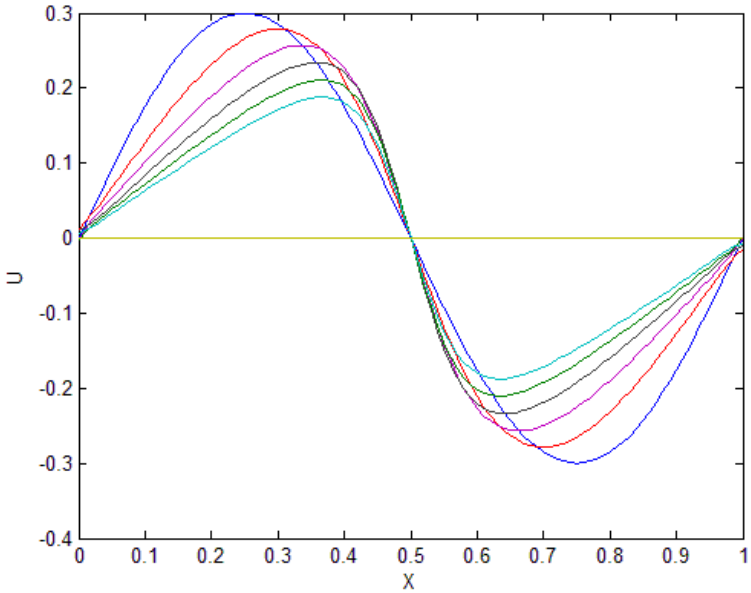



Figure 4.4: Velocity v/s Position, Viscosity=0.01

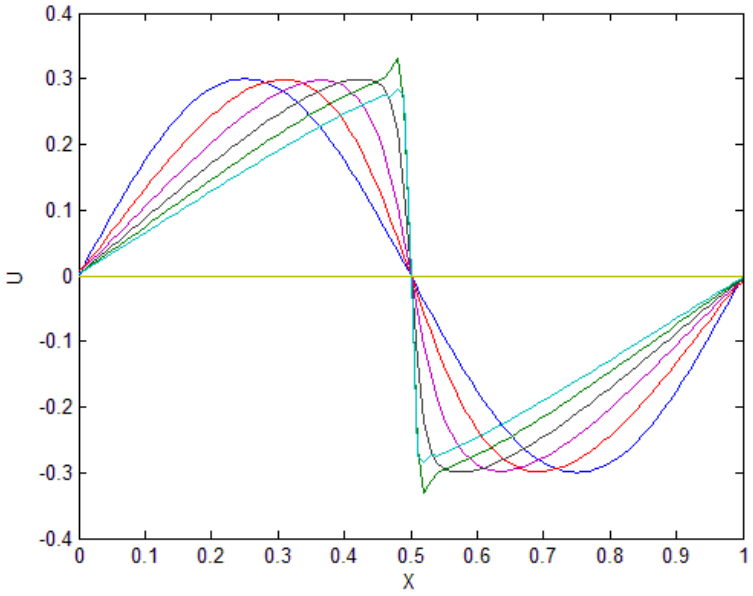



Figure 4.5: Velocity v/s Position, Viscosity=0.001

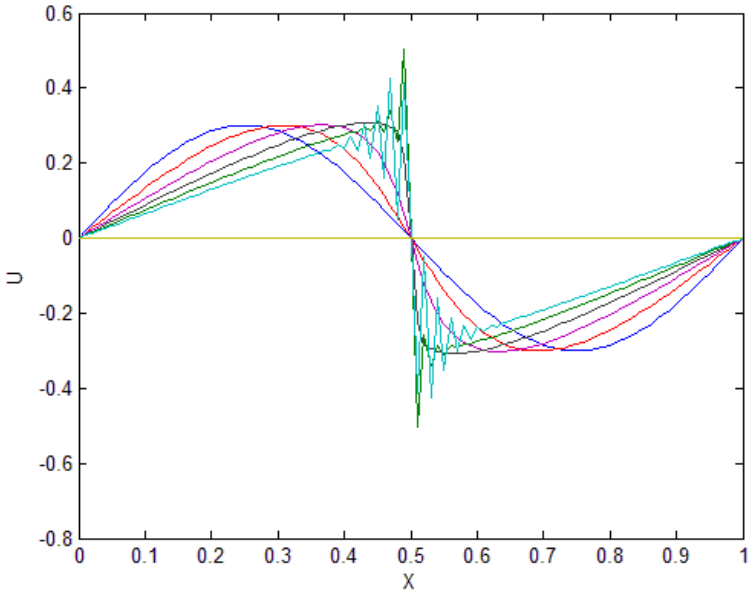



Figure 4.6: Velocity v/s Position, Viscosity=0.0001

In the following above figure viscosity 0.0001 $m^2/s$: at the top and the bottom of the shock wiggles appears.

*Shock formation process with amplitude 0.5,*

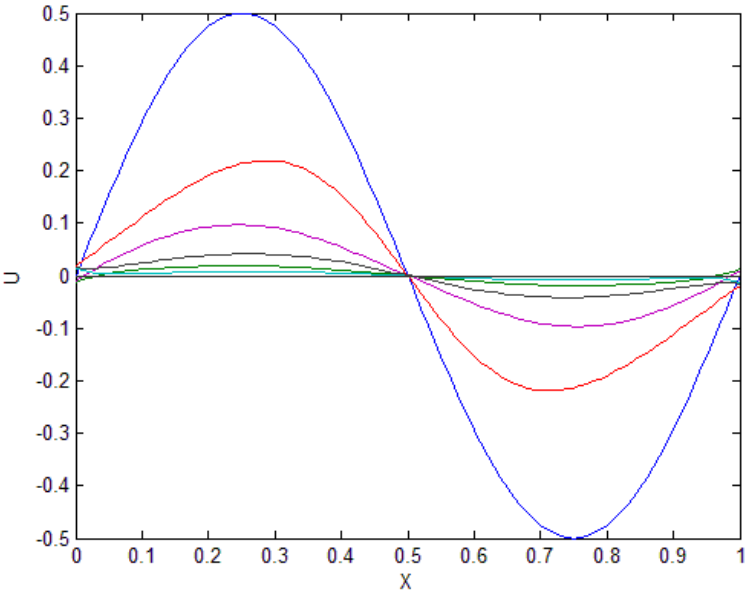



Figure 4.7: Velocity v/s Position, Viscosity=0.1

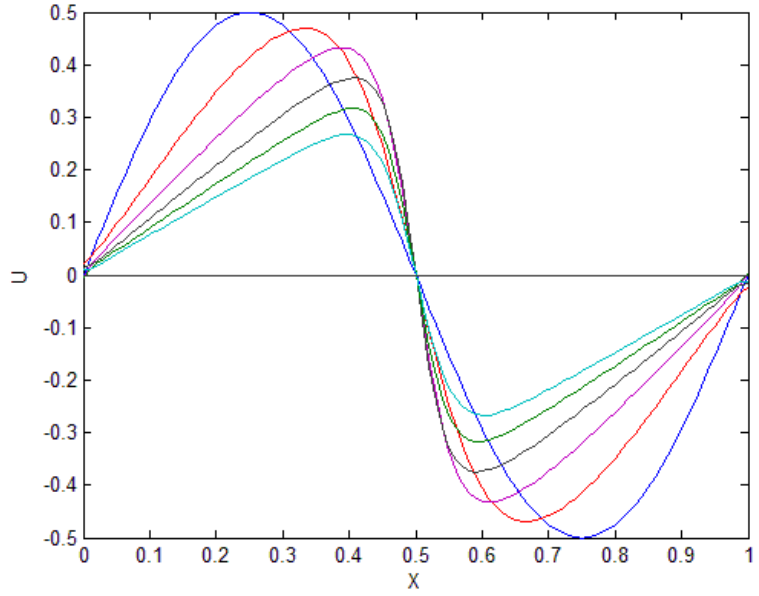



Figure 4.8: Velocity v/s Position, Viscosity=0.01

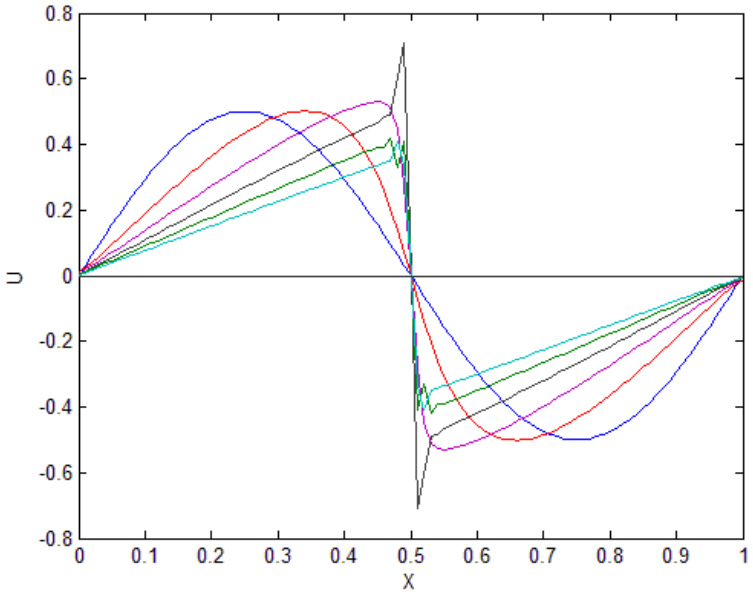



Figure 4.9: Velocity v/s Position, Viscosity=0.001

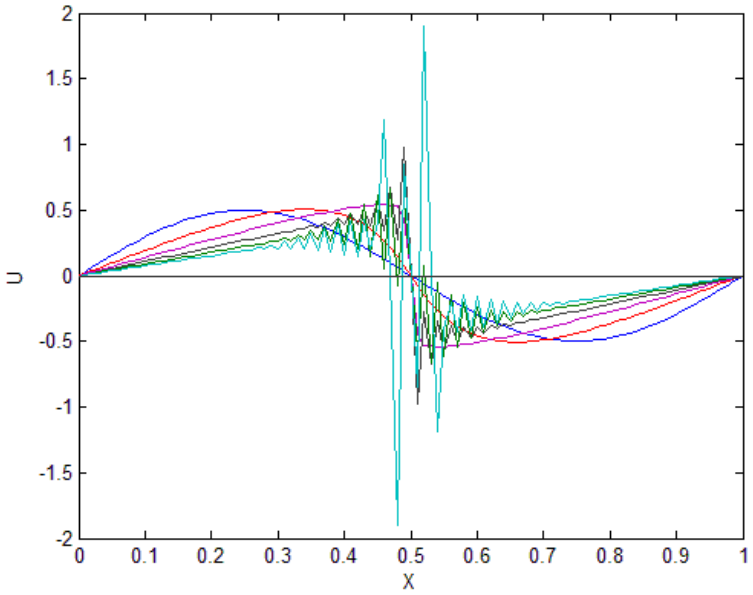



Figure 4.10: Velocity v/s Position, Viscosity=0.0001

*Discussion*

The shock starts to appear as the viscosity values lend lower. So the viscous term in Burgers equation has a major role to play in the understanding of the shock formation.

The Implicit Crank-Nicholson scheme proves to be efficient to solve burgers' equations. However for too low viscosities others schemes must be used. This work shows that instabilities can have numerical and physical reasons. Paying attention to both problems is then required.

The Burgers' equation allows understanding the process of shocks formation. The effect of the viscous term is to reduce the amplitude of the wave for increasing time and then can prevent shocks from forming. This simple model helps us understand the concept of behavior of the liquid under excitation.

# Chapter 5

## 2-D Sloshing in Rectangular Tank

The example considers small and large sloshing of a fluid in a two dimensional rectangular tank. The study is carried out with the formulation of the problem and then the numerical results are obtained for the same.

## 5.1 Introduction

Many applications in fluid mechanics have problems in which the fluids are not contained in a fixed domain. The area which is covered by the fluid may change continuously. Then we need to evaluate the free surface part of the problem, and one can speak about sloshing in tank as one of the example for free surface flow. Analytical approaches used to predict the dynamic response of liquid-filled tanks have difficulties in handling some special types of tank geometry. Numerical methods are useful when the container walls are not vertical and straight or when the liquid wave heights are large. The main computational methods which are used in practice today are (1) the finite difference (FD) [45, 46], (2) the finite element (FE) [47-52], (3) finite volume (FV) [53] and (4) the boundary element (BE). The BE method is convenient for studying liquid sloshing behavior because both the FD and FE methods require very long computational time and a large amount of input data. By accuracy of the results are very important and generally FE method results agreed fairly with the experimental measurements compared to the other FD and BE and even FV is very commonly used in practice for the sloshing dynamics today.

In this chapter, a computational framework is presented for modeling of a free surface fluid flow. The main area of focus is on the incompressible Newtonian fluid flow at relatively low Reynolds numbers. The motion of the fluid domain is determined by the motion of the free surface. For the modeling of the incompressible fluid flow we use a stabilized equal order velocity–pressure finite element formulation adapted to a moving domain. The stabilization technique employed is same as in the mentioned and discussed in the previous chapters. The formulation used in this work is referred to as the Galerkin/least-squares stabilization technique, which under certain conditions becomes equivalent to the combined streamline-upwind/- and pressure-stabilizing/Petrov–Galerkin method [32-35].

An arbitrary Lagrangian–Eulerian (ALE) [69] description allows the mesh in the interior of the domain to move independently of the fluid. Thus it is possible to maintain reasonably shaped meshes and to describe the boundaries accurately at the same time. In all the above strategies the movement of the finite element mesh is governed by an appropriate algorithm, thus maintaining a good mesh quality despite substantial deformation of the fluid domain.

As discussed in detailed in the chapter 2, the velocity difference $u-\hat{v}$ is denoted as the convective velocity. Here the grid points can be moved independently of the fluid motion. This strategy allows us to study the problem even when there is substantial deformation of the fluid domain.

The integration in time is carried out by using the generalized α method this method belongs to the class of discrete and implicit single step integration schemes. For linear problems the scheme can be shown to be second order accurate and unconditionally stable. Furthermore, the generalized α method was originally developed for the second order problems arising in solid dynamics by Chung and Hulbert [70], and later adapted to the first order problems typically encountered in Eulerian fluid dynamics by Jansen et al. [71]. For a detailed study of the generalized α method in the context of the stabilized Eulerian finite element formulation we refer to [71].

Finally to compute the final set of unknowns at the current time instant we need to develop an efficient solution procedure. Here a Newton-Raphson procedure [4], which incorporates the full linearization of the incremental problem and hence exhibits asymptotically quadratic convergence of the solution for all unknowns.

## 5.2 Governing Equation

We can formulate the momentum conservation law and continuity equation for the incompressible flow in the referential description as

$$\rho(\dot{u}+(\nabla_{\hat{x}}u)(u-\hat{v})-f)-\nabla_{\hat{x}}\cdot\sigma=0 \quad \forall(\hat{x},t)\in\Omega\times I \tag{26}$$

$$\nabla_{\hat{x}}\cdot u=0 \ \forall(\hat{x},t)\in\Omega\times I, \tag{27}$$

where ρ, $f$ and r $\sigma$ represent, respectively, the fluid density, the volume force vector and the Cauchy stress tensor. The time interval is bound within $I$ = [0, T]. Now the constitutive equation gets restricted to,

$$\sigma=-p\boldsymbol{I}+2\mu\nabla^{s}_{\hat{x}}u \tag{28}$$

where $\boldsymbol{I}$ is the second order identity tensor,$\boldsymbol{\mu}$ is the fluid viscosity and $\boldsymbol{p}$ is the pressure gradient. The boundary conditions are given as below:

$$u-g=0,\forall(\hat{x},t)\in\Gamma_{g}\times I, \tag{29}$$

$$\sigma_{\hat{n}}-h=0,\forall(\hat{x},t)\in\Gamma_{h}\times I, \tag{30}$$

$$\sigma_{\hat{n}}=0,\forall(\hat{x},t)\in\Gamma_{free}\times I, \tag{31}$$

$$(u-\hat{v})\cdot\hat{n}=0,\forall(\hat{x},t)\in\Gamma_{free}\times I, \tag{32}$$

$g$, $h$ and $\hat{n}$ are the prescribed velocity and the traction vectors and the current outward normal unit vector of the boundary, which is given by positions $\hat{x}$ of the boundary. For the free surface $\Gamma_{free}$ the velocity $\hat{v}$ of the reference frame and the position $\hat{x}$ need to staisfy Eq. (32), which will ensure that no fluid particles flow across this part of the boundary $\Gamma$.

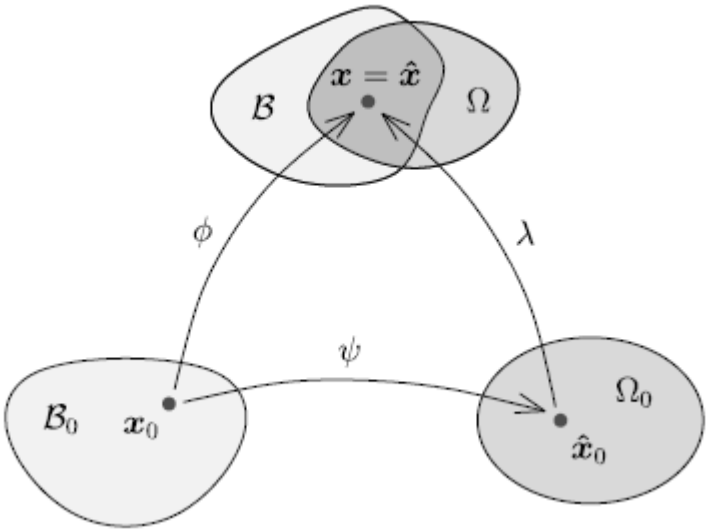



Figure 5.1, Mappings and configurations in ALE description [4].

## 5.3 Finite element formulation

A stabilized velocity-pressure finite element formulation of the problem is described below

$$G_{Gal}(u^h,p^h;\eta^h,q^h)+G_{Stab}(u^h,p^h;\eta^h,q^h)=0 \tag{33}$$

This is the variational form of the standard Galerkin terms, to which a stabilized term of the momentum equation is added.

$$\begin{aligned}G_{Gal}(u^h,p^h;\eta^h,q^h)=&\int_{\Omega^h}((\rho\eta^h\cdot(\dot{u}^h+(\nabla_{\hat{x}}u^h)(u^h-\hat{v}^h)-f)+\\&\nabla_{\hat{x}}\eta^h:\sigma(u^h,p^h)+q^h(\nabla_{\hat{x}}\eta^h\cdot u^h))dv-\int_{\Gamma^h{}_h}\eta^h\cdot h^h da\end{aligned} \tag{34}$$

$$G_{Stab}(u^h, p^h; \eta^h, q^h) = \sum_{e=1}^{n_{el}} \int_{\Omega^e} [\tau_u \rho(\nabla_{\hat{x}} \eta^h)(u^h - \hat{v}^h) + \tau_p \nabla_{\hat{x}} q^h] \cdot$$
$$[\rho(\dot{u}^h + \nabla_{\hat{x}} u^h)(u^h - \hat{v}^h) - f)(u^h - \hat{v}^h) + \nabla_{\hat{x}} p^h] dv \tag{35}$$

In this above Eq. (35), there is no viscous term. To implement the methodology we shall follow the example [4] and employ two parameters, here denoted as $\tau_u$ and $\tau_p$. We treat both the stabilisation parameter seperately and these are defined below:

$$\tau = \frac{h_e}{2\left\|u^e - \hat{v}^e\right\|\rho} z, z = \frac{\beta_1}{\sqrt{1 + \left(\frac{\beta_1}{\beta_2 \mathrm{Re}^e}\right)^2}}, \mathrm{Re}^e = \frac{\left\|u^e - \hat{v}^e\right\| h^e \rho}{2\mu}, \tag{36}$$

The $\tau_u$ and $\tau_p$ .are constant within every element and hence the stabilization terms are discountinuous across the inter element boundaries, which in turn explains the summation of intergralas in Eq.(36)

## 5.4 Motion of the finite element mesh

We can see that the motion in mesh is arbitary, except for the outline. On the free surface Eq.(32) has to be satsisfied, but still there may be some tangential movement of the boundary nodes undertermined. On the non free surface we can prescribe the motion of the nodes normal to the current configuration of the boundary. We can also note that some regions may not need to get adapted to the new geometry and we can have fixed boundary in space. Here we can set the mesh velocity as zero and the flow problem with be Eulerian.

*5.4.1 Motion of the Internal Nodes*

We use following two techniques [4], for the motion of the internal nodes:

*Pseudo-elastic technique:* Here the mesh is assumed to represent an elastic solid body. A standard Lagrangian finite element technique typically employed in solid mechanics can then be used to adapt mesh to the new geometry of the domain. For small distortions of the geometry the linear elastic model is sufficient. In the presence of large deformations of the fluid domain a hyperelastic model may be more suitable.

*Optimization of mesh quality:* The mesh can be moved in such a way that its quality, with respect to a certain criteria, is optimal at all times. We have chosen a criteria which satisfies the following condition and is the version used in [4].

$$W = \sum_{e=1}^{n_{el}} \left( \frac{r_{out}^e}{r_{in}^e} \right) \Rightarrow MIN, \tag{37}$$

The quantities $r_{in}^e$ and $r_{out}^e$ denote the inner and the outer radii of a triangular or tetrahedral finite element. This methodology renders acceptable meshes even for much distorted geometries. Both methods can be fully linearised and thus enable the employment of the Newton–Raphson procedure to solve for the new nodal positions.

### *5.4.2 Motion of nodes on the free surface*

*Lagrangian description*: The most straightforward approach to satisfy (32) is the purely Lagrangian description of the free surface, But this often very quickly leads to distortion of the surface mesh.

*Optimization of surface mesh quality:* This is the approach adopted in this work. Similar to the methodology for the internal nodes, the surface mesh may be moved such that certain quality criteria are met.

*Combination with internal node algorithm:* The third option to determine the motion of the boundary nodes simply consists in treating (32) as a boundary condition for the update algorithm of the internal nodes. This, however, requires a rather tedious elimination procedure in the mesh solver of the surface degrees of freedom and has not been further investigated in this work.

## 5.5 Integration in time

Here we have suggested the generalized $\alpha$ method [70] for the two dimentional sloshing problem. This method has proved it is very efficient and robust alternative to the more expensive time finite element method.

The generalized $\alpha$ method is given by [71]:

$$\begin{aligned} u^h_{n+1} &= u^h_n + \Delta t(1-\gamma)\dot{u}^h_n + \Delta t\gamma\dot{u}^h_{n+1}, \\ \dot{u}^h_{n+\alpha_m} &= (1-\alpha_m)\dot{u}^h_n + \alpha_m\dot{u}^h_{n+1}, \\ u^h_{n+\alpha_f} &= (1-\alpha_f)u^h_n + \alpha_f u^h_{n+1}, \end{aligned} \tag{38}$$

$$\bar{G}_{Gal}(u^h, p^h; \eta^h, q^h) + \bar{G}_{Stab}(u^h, p^h; \eta^h, q^h) = 0 \tag{39}$$

The above is the semi discrete form of the finite element formulation. Where $\gamma, \alpha_m$ and $\alpha_f$ are the integration parameters. The integration parameters are reduced to one independent control variable as follows:

$$\gamma = \frac{1}{2} + \alpha_m - \alpha_f, \alpha_m = \frac{1}{2}\frac{3-\rho_\infty}{1+\rho_\infty}, \alpha_f = \frac{1}{1+\rho_\infty}, \tag{40}$$

Foe detailed study of this method we can refer [72].

## 5.6 Mesh Update

The configuration $\hat{x}^h_n$ and the velocity field $\hat{v}^h_n$ at the discrete time instants $t_n$, n = 0, 1, 2. . . $N_{times}$ are introduced. In this work, $\hat{x}^h_n$ and $\hat{v}^h_n$ are related by a simple generalized midpoint scheme

$$\hat{v}^h_{n+1} = \frac{1}{\Delta t\hat{\gamma}}(\hat{x}^h_{n+1} - \hat{x}^h_n) - \frac{1-\hat{\gamma}}{\hat{\gamma}}\hat{v}^h_n, \tag{41}$$

where $\hat{\gamma}$ is the intergration parameter. The following expressions are then employed in the weak
form (39).

$$\begin{aligned} \hat{x}^h_{n+\alpha_f} &= (1-\alpha_f)\hat{x}^h_n + \alpha_f\hat{x}^h_{n+1}, \\ \hat{v}^h_{n+\alpha_f} &= (1-\alpha_f)\hat{v}^h_n + \alpha_f\hat{v}^h_{n+1}, \end{aligned} \tag{42}$$

## 5.7 Solution Algorithm

We can use one of the following startegy for obtaining the soultion for the sloshing problem

*Monolithic approach:* A solution procedure is performed on a complete system in terms of all unknowns. The strongly coupled problem is normally solved by applying an iterative procedure until the desired accuracy is achieved. This methodology yields a very large system matrix, which also tends to be badly conditioned.

*Partitioned solution procedure:* The three components of the problem are solved sequentially, thereby communicating intermediate results between them. We introduce the following terminology: Note that each of the solvers fluid, free surface and mesh requires the application of a nonlinear solution procedure, such as the Newton–Raphson method, to overcome the nonlinearities of the problem. The considerable difficulty posed by the partitioned methodology consists in finding an appropriate sequence of the three solvers and in passing the appropriate information between them. The resulting lack of accuracy often adversely affects the stability and leads to a restriction of the method to small time steps. Such staggered solution procedures may be improved by interfield iteration. However, the convergence of the residuals tends to be poor especially for large systems.

*Mixed strategies:* The degree of coupling of the problem may be reduced by various combinations of implicit and explicit time integration schemes applied to different parts of the solution domain. Such approaches may again adversely affect the stability and impose severe restrictions on the time step size, especially for problems with large deformation of the free surface.

In this work, a iterative partitioned solution [4] methodology is followed, which enables the computation of the solution of the discretized problem up to the desired accuracy. The algorithm represents, in fact, an application of the Newton–Raphson procedure, and thus, it exhibits quadratic rate of asymptotic convergence.

Iterative partitioned solution procedure

1. Predict the free suface velocity, initial velocity and pressure at n+1 time step on the basis of the know solution at time step n.

2. Free surface solver
3. Mesh solver
4. Fluid solver: compute residual
5. Check for the tolerance and then exit
6. Fluid solver: compute linearization, solve system, update for free surface velocity, initial velocity amd the pressue.
7. Go to 2

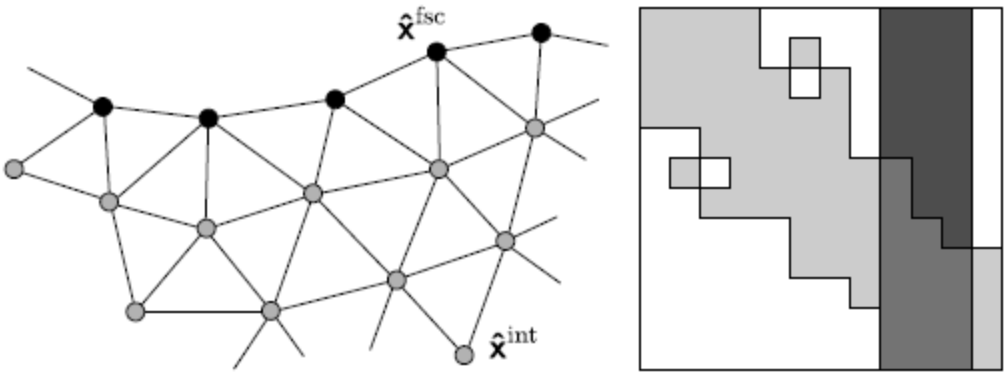



Figure 5.2, Finite elemenet mesh with free surface and fluid stiffness matrix [4].

## 5.8 Large amplitude sloshing

A rectangular container is considered. It is subjected to a periodic displacement in x-direction. This is achieved by treating the horizontal motion of the finite element nodes at the vertical boundaries of the container in a lagrangian manner. The x- velocity of these nodes is prescribed as $u = A\omega \sin(\omega t)$. The horizontal position of the container oscialltes with the amplitude A and the frequency $f = \omega / (2\pi)$.

Initiallt, the fluid is at rest and in equilibrium. The motion of the mesh is based on the pseudo-elastic methodology. We have plotted the figure for the configuration of the mesh and pressure isolines at different time instants. We also have plotted a graph for $y$ v/s $\omega$.

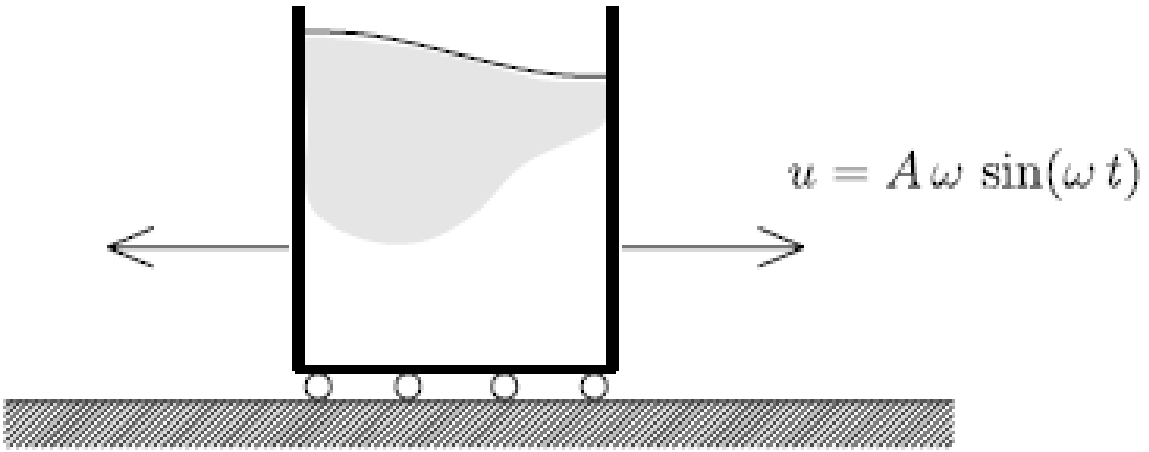



Figure 5.3, Large Amplitude Sloshing, Problem description [4].

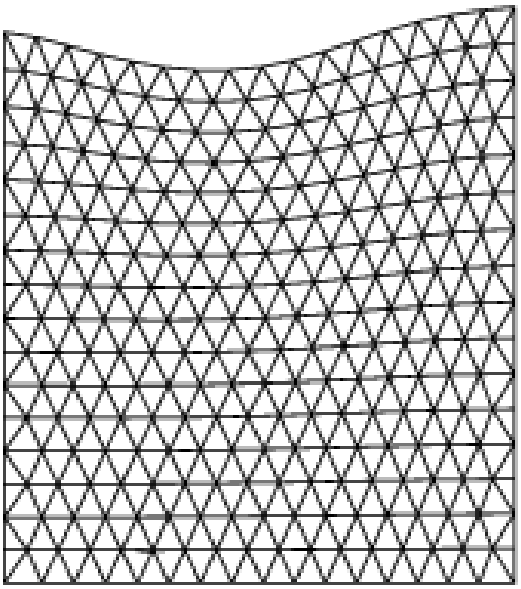


Figure 5.4 :Mesh at time instant 96.3 s, $\omega$ =1.5

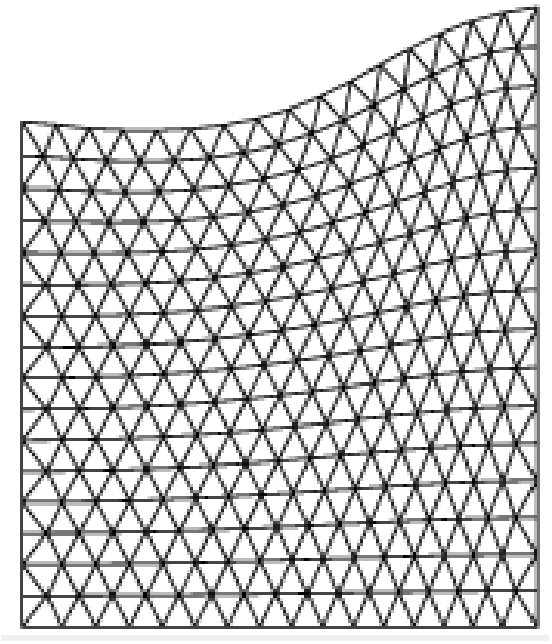


Figure 5.5 :Mesh at time instant 96.168 s, $\omega$ =1.5

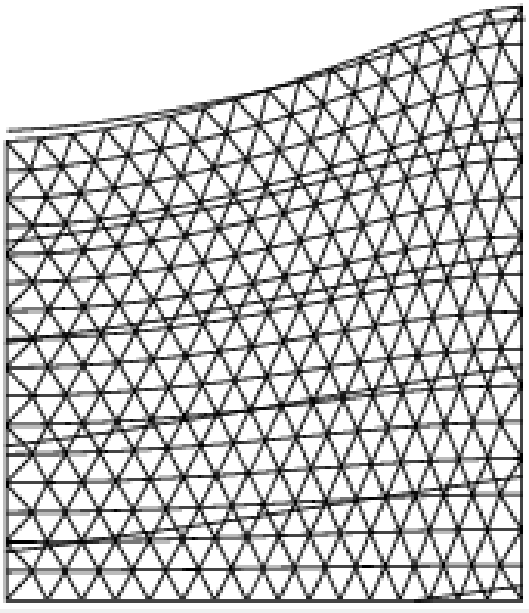


Figure 5.6: Mesh, Pressure Isoline at time instant 6.4817 s, $\omega$ =1.5

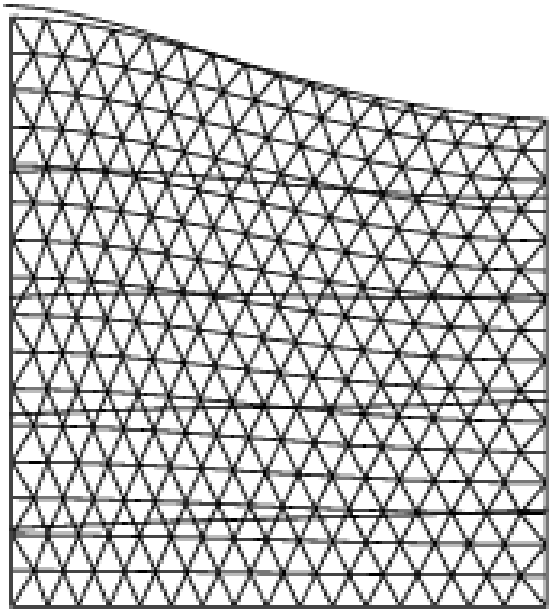


Figure 5.7: Mesh, Pressure Isoline at time instant 5.4235 s, $\omega$ =1.5

Large amplitude sloshing,: The graph displays the amplitudes of the bubble and the spike for the different frequencies. A lock in phenomena is clearly captured at frequency =1.5 for the material density of water. Another for the LNG we can see that the phenomena is occurring close to 1.6.

For density of water

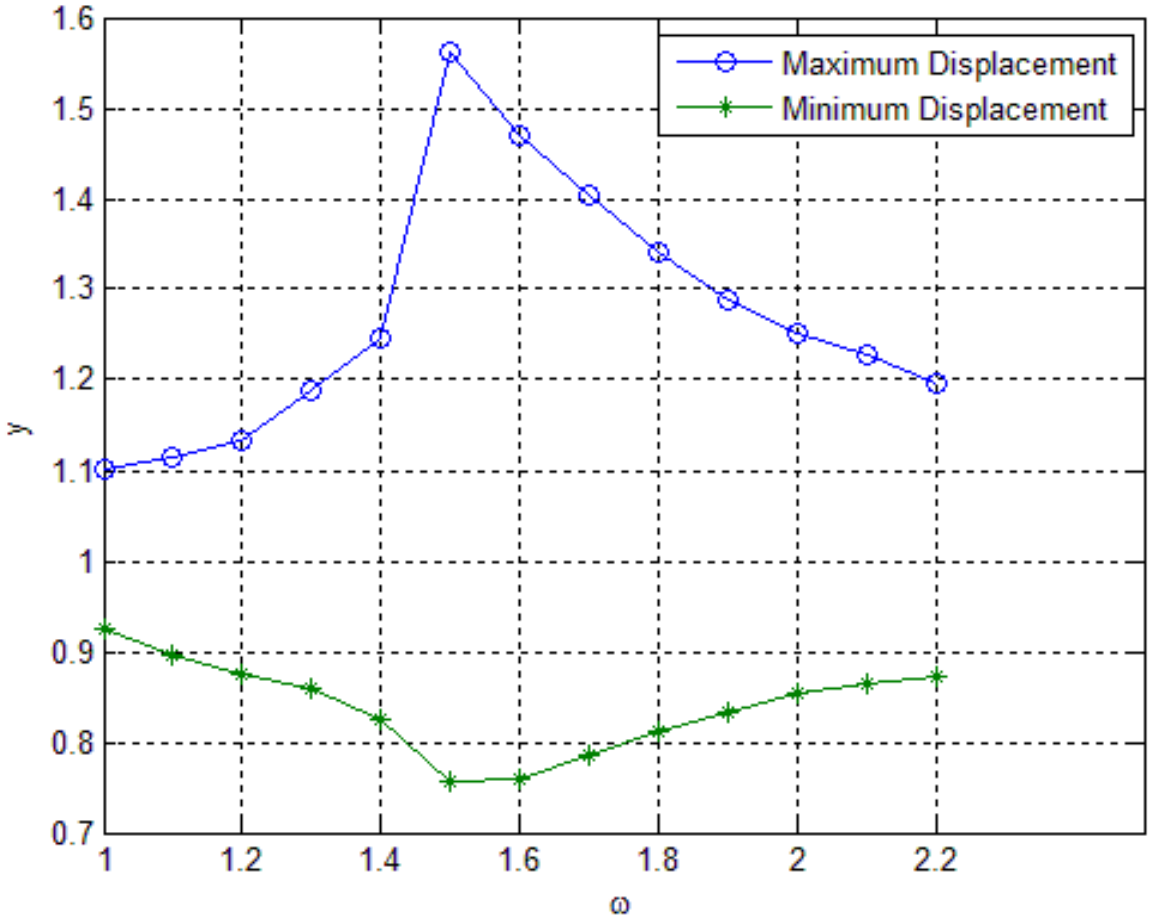



Figure 5.8: Displacement v/s Velocity plot

For density of LNG

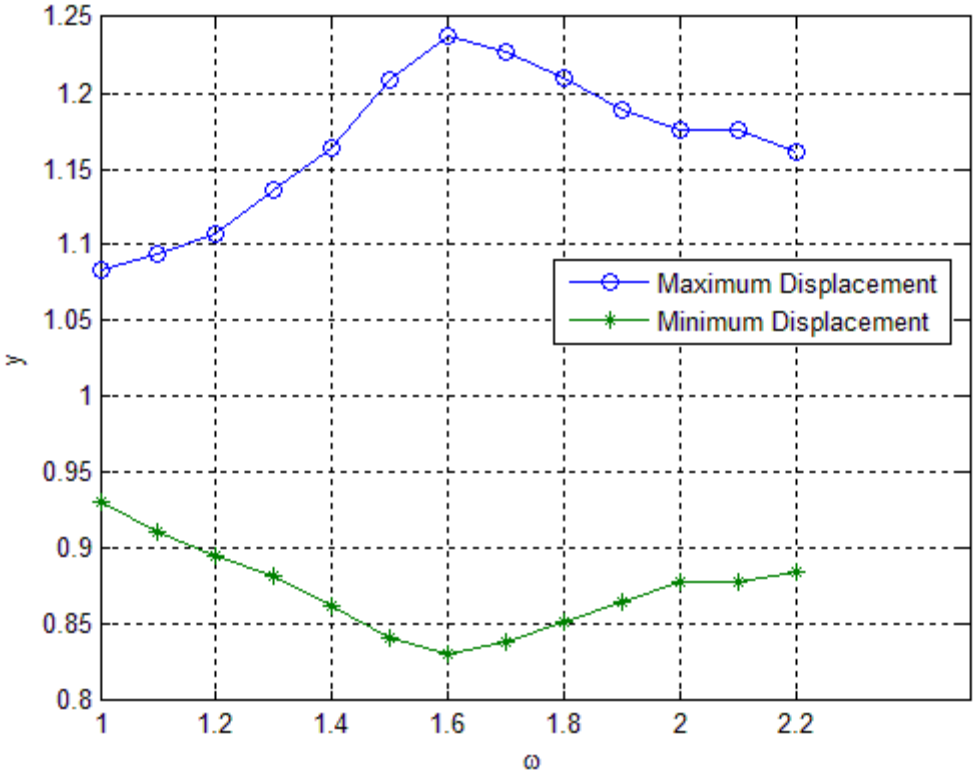



Figure 5.9: Displacement v/s Velocity plot

*Discussion*

In order to validate our algorithm for the numerical simulation of free surface flows, we have considered a classical two-dimensional space sloshing problem with small deformations. A uniform flow field is subjected on its free surface to an external traction. We study the time evolution of the free surface and of the fluid flow. The sloshing problem consists in looking at the free oscillations of a liquid contained in a two-dimensional tank of unit width and height.

# Chapter 6

# Conclusion and Further Research

## 6.1 Conclusions

The preceding chapters have presented a 1-D model for Burger equation in fixed and moving mesh framework to study the free surface flow problem with emphasis on sloshing in liquid storage tanks.

We have developed a numerical model to simulate incompressible viscous free surface flow (sloshing problem). The model is based on Burgers equation for fluid flow equations. The effective numerical algorithm has been developed which combines the Finite element method and the Finite difference methods to solve the governing equations at the boundary and the interior of the computational domain using an ALE scheme. The numerical results show that the formulation gives accurate results.

- The numerical solutions obtained for fixed mesh framework using burgers equation are accurate and show good agreement with the analytical solution.

- A partitioned solution procedure is developed based on the Newton-Raphson methodology which incorporates full linearization of the overall incremental problem. The Newton method has been proven as the most powerful method given its quadratic convergence. As a matter of fact, this method should be preferred.

- The Burger's equation analogue to 1-D fluid flows is solved without and with FE mesh motion, to show that the mesh motion practically does not affect the solutions. All solutions presented show that the proposed algorithm is sufficiently accurate and stable.

- The implicit algorithm showed that high accuracy of results can be achieved with a relatively large time step. The implicit schemes (Backward Euler and Crank Nicholson) where unconditionally unstable in comparison with the forward Euler scheme which showed lot of oscillations and instability.

- A numerical example is provided to demonstrate the efficiency of the methodology by modeling large amplitude sloshing in a rectangular tank. This gives evidence of accuracy, robustness and efficiency of the computational strategy.

- The effect of viscous term on Burgers equation was analyzed and the process of shock formation gave an insight into the effect of sloshing with large amplitudes.

## 6.2 Recommendations for further research

- Extend the solution algorithm to incorporate the fluid structure interactions.

- Develop various models for different types of Liquid Storage Tanks and carry out analysis for the same.

- Consider the effect of surface tension into the current model.

- Compare the results obtained with external available software packages like Ansys etc

# Appendices

# List of Symbols and Abbreviations

| *Abbreviation* | *Description* |
|---|---|
| LNG | Liquefied Nitrogen Gas |
| ALE | Arbitrary Lagrangian Eulerian |
| FEM | Finite Element Method |
| FDM | Finite Difference Method |
| FVM | Finite Volume Method |
| u | Solution variable |
| μ | Dynamic Viscosity |
| $w(x)$ | Weighting function |
| $N_i$ | Shape function |
| ξ | Reference element dimension |
| h (or) l | element length |
| Pe | Peclet number |
| α | Constant in the Petrov-Galerkin Weight |
| t | Time |
| x | Space dimension |
| f(x) | Differentiable function |
| L | Maximum length of the domain |
| ν | Kinematic viscosity |
| a | Convective velocity |
| F | Jacobian matrix |
| Δt | Time step |
| Φ | Pitching angle |

| | |
|---|---|
| $\eta$ | Surface elevation |
| BE | Boundary Element |
| $\sigma$ | Cauchy stress |
| p | Normal pressure |
| I | Identity tensor |
| $\rho$ | Density |
| $\rho_\infty$ | Free stream density |
| Re | Reynolds number |
| $\gamma$ | Integration parameter |
| $\alpha_m$ | Integration parameter |
| $\alpha_f$ | Integration parameter |
| $\hat{v}_n^h$ | Velocity field |
| $t_n$ | time instance |
| $\hat{x}_n^h$ | Mesh configuration |
| $\Gamma$ | Boundary |
| $\hat{n}$ | Outward normal unit vector |
| $G_{Gal}$ | Function of Galerkin terms |
| $G_{Stab}$ | Function of Stabilization terms |

# Bibliography


[1] S.J. Lee, M.H. Kim, D.H. Lee, J.W. Kim and Y.H. Kim, The effects of LNG-tank sloshing on the global motions of LNG carriers. *Ocean Engineering Volume 34, Issue 1, 10-20, 2007.*

[2] Chen, Y.G., Price, W.G. and Temarel, P, Numerical simulation of liquid sloshing in LNG tanks using a compressible two-fluid flow model. *In, Proceedings of the 19th International Offshore and Polar Engineering. Mountain View, USA, The International Society of Offshore and Polar Engineers (ISOPE), 221-230, 2009.*

[3] Lee, S.J., M.H. Kim, D.H. Lee, and Y.S. Shin (2007), The effects of tank sloshing on LNG-ship responses, *Proc. 26th Int. Offshore Mech. and Arctic Eng. Conference, San Diego, USA.*

[4] W.G. Dettmer, D. Peric, A Computational framework for free surface fluid flows accounting for surface tensions. *Comp methods in applied mechanics and Engrg,195, 3038-3071, 2004.*

[5] W.G. Dettmer, D. Peric, A Computational framework for fluid-structure interactions. Finite element formulation and application *Comput Methods in Appl mechanics and Engrg, 195 5754–5779, 2006.*

[6] W.G. Dettmer, D. Peric, A Computational framework for fluid-rigid body. Finite element formulation and application *Comp Methods in Appl mechanics and Engrg,195, 1633–1666, 2005.*

[7] B. Desjardins, M. J. Esteban, C. Grandmont, and P. Le Tallec, Weak Solutions for a Fluid-Elastic Structure Interaction Model. *Rev. Mat. Comput., 14(2):523-538, 2001.*

[8] C. Grandmont and Y. Maday, Existence for an Unsteady Fluid-Structure Interaction Problem. *M2AN, 34(3):609-636, 2000.*

[9] M. Kang, R. P. Fedkiw, and X.-D. Liu, A Boundary Condition Capturing Method for Multiphase Incompressible Flow. *Journal of Scientific Computing, 15:323-360, 2000.*

[10] J. Li and Y. Renardy, Numerical Study of Flows of Two Immiscible Liquids at Low Reynolds Number. *SIAM Rev., 42(3):417-439, 2000.*

[11] A.-K. Tornberg and B. Engquist, Interface Tracking in Multiphase Flows. *Multifield Problems in Solid and Fluid Mechanics, to appear, 2000.*

[12] M. Picasso, J. Rappaz, A. Reist, M. Funk, and H. Blatter, Numerical Simulation of the Motion of a Two Dimensional Glacier. *Int. J. Num. Meth. Engrg, submitted, 2003.*

[13] A. Bonito, M. Picasso, and M. Laso, Numerical Simulation of 3D Viscoelastic Flows with Free Surfaces. *In J. Non-Newtonian Fluid Mechanics. XIIIth Workshop on Numerical Methods for non-Newtonian Flows, 2003.*

[14] M. J. Shelley, F.-R. Tian, and K. Wlodarski, Hele-Shaw Flow and Pattern Formation in a Time-Dependent Gap. *Nonlinearity, 10:1471-1495, 1997.*

[15] R. W. Lewis, S. E. Navti, and C. Taylor, A Mixed Lagragian-Eulerian Approach to Modelling Fluid Flow During Mould Filling. *Int. J. Numer. Meth. Fluids, 25:931 - 952, 1997.*

[16] J. C. Martin and W. J. Moyce, An Experimental Study of the Collapse of Liquid Columns on a Rigid Horizontal Plate. *Philos. Trans. Roy. Soc. London Ser., A 244:312-324, 1952.*

[17] M. Schmid and F. Klein, Einuflder Wandreibung auf das Fullverhalten Dunner Platten. *Preprint, Steinbeis Transferzentrum, Fachhochschule Aachen, 1996.*

[18] M. Bertsch, D. Hilhorst, and Cl. Schmidt-Lainfle, The Well-Posedness of a Free Boundary Problem for Burgers' Equation. *Nonlinear Analysis, Theory, Methods and Applications, 23(9):1211-1224, 1994.*

[19] D. Errate, M. J. Esteban, and Y. Maday, Couplage Fluide-Structure. Un Modele Simplifie en Dimension 1. *C.R. Acad. Sci. Paris, 318, Serie 1:275-281, 1994.*

[20] C.W. Hirt, A.A. Amsden, J.L. Cook, An arbitrary Lagrangian–Eulerian computing method for all flow speeds, J. Comput. Phys. 14, 227–253, 1974.

[21] T.J.R. Hughes, W.K. Liu, T.K. Zimmermann, Lagrangian–Eulerian finite element formulation for incompressible viscous flows, *Comput. Methods Appl. Mech. Engrg. 29,329–349, 1981.*

[22] J. Donea, Arbitrary Lagrangian–Eulerian finite element methods, in: T. Belytschko, T.J.R. Hughes (Eds.). *Computational Methods for Transient Analysis, Elsevier, Amsterdam, 1983, pp. 473–516.*

[23] B. Ramaswamy. M. Kawahara, Arbitrary Lagrangian–Eulerian finite element method for unsteady, convective, incompressible viscous free surface fluid flow. .*Int. J. Numer. Methods Fluids 7,1053–1075, 1987.*

[24] B. Ramaswamy, Numerical simulation of unsteady viscous free surface flow. *J. Comput. Phys. 90, 396–430, 1990.*

[25] A. Huerta, W.K. Liu, Viscous flow with large free surface motion. *Comput. Methods Appl. Mech. Engrg. 69, 277–324, 1988.*

[26] A. Soulaimani, M. Fortin, G. Dhatt, Y. Ouellet, Finite element simulation of two- and three-dimensional free surface flows. *Comput. Methods Appl. Mech. Engrg.86 265–296, 1991.*

[27] P.A. Sackinger, P.R. Schunk, R.R. Rao, A Newton–Raphson pseudo-solid domain mapping technique for free and moving boundary problems: a finite element implementation. *J. Comput. Phys. 125, 83–103, 1996.*

[28] A. Soulaimani, Y. Saad, An arbitrary Lagrangian–Eulerian finite element method for solving three-dimensional free surface flows. *Comput. Methods Appl. Mech. Engrg. 162 79–106, 1998.*

[29] H. Braess, P. Wriggers, Arbitrary Lagrangian Eulerian finite element analysis of free surface flow. *Comput. Methods Appl. Mech. Engrg. 190, 95–109, 2000.*

[30] T. Belytschko, W.K. Liu, B. Moran, Nonlinear Finite Elements for Continua and Structures. *John Wiley & Sons, Chichester, UK, 2000.*

[31] T.E. Tezduyar, M. Behr, J. Liou, A new strategy for finite element computations involving moving boundaries and interfaces—The deforming- spatial-domain/space–time-procedure: I. The concept and the preliminary numerical tests. *Comput. Methods Appl. Mech. Engrg. 94, 339–351, 1992.*

[32] T.E. Tezduyar, M. Behr, S. Mittal, J. Liou, A new strategy for finite element computations involving moving boundaries and interfaces—The deforming-spatial-domain/space–time-procedure: II. Computation of free-surface flows, two-liquid flows, and flows with drifting cylinders. *Comput.Methods Appl. Mech. Engrg. 94,353–371, 1992.*

[33] W.G. Dettmer, P.H. Saksono, D. Peric´, On a finite element formulation for incompressible Newtonian fluid flows on moving domains in the presence of surface tension. *Commun. Numer. Methods Engrg. 19 , 659–668, 2003.*

[34] A. Masud, T.J.R. Hughes, A space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations for moving domain problems. *Comput. Methods Appl. Mech. Engrg. 146, 91–126, 1997.*

[35] P. Hansbo, A Crank–Nicolson type space–time finite element method for computing on moving meshes. *J. Comput. Phys. 159, 274–289, 2000.*

[36] V. Maronnier, M. Picasso, and J. Rappaz, Numerical Simulation of Free Surface Flows. *J. Comp. Phys., 155:439-455, 1999.*

[37] V. Maronnier, M. Picasso, and J. Rappaz, Numerical Simulation of Three Dimensional Free Surface Flows. *Int. J. Num. Meth. Fluids, 42(7):697-716, 2003.*

[38] A.N. Brooks, T.J.R. Hughes, Streamline-upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Comput. Methods Appl. Mech. Engrg. 32,199–259, 1982.*

[39] T.J.R. Hughes, L.P. Franca, G.M. Hulbert, A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective–diffusive equations.*Comput. Methods Appl. Mech. Engrg. 73,173–189,1989.*

[40] T.J.R. Hughes, L.P. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuska–Brezzi condition: a stable Petrov–Galerkin formulation of the Stokes problem accommodating equal-order interpolations. *Comput. Methods Appl. Mech.Engrg. 59, 85–99, 1986.*

[41] T.E. Tezduyar, S. Mittal, S.E. Ray, R. Shih, Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity–pressure elements. *Comput. Methods Appl. Mech. Engrg. 95,221–242, 1992.*

[42] J.-J. Droux, T.J.R. Hughes, A boundary integral modification of the Galerkin/least-squares formulation for the Stokes problem. *Comput. Methods Appl. Mech. Engrg. 113 ,173–182, 1994.*

[43] K.E. Jansen, S.S. Collis, F. Shakib, A better consistency for low-order stabilized finite element methods.*Comput. Methods Appl. Mech. Engrg. 174,153–170,1999.*

[44] T.E. Tezduyar, Y. Osawa, Finite element stabilization parameters computed from element matrices and vectors. *Comput. Methods Appl. Mech.Engrg. 190,411–430,2000.*

[45] D.H. Zhang, A.T. Chwang, Numerical study of nonlinear shallow water waves produced by a submerged moving disturbance in viscous flow.*Phys. Fluids 8,147,1996.*

[46] D.H. Zhang, A.T. Chwang, On solitary waves forced by underwater moving objects. *J. Fluid Mech. 389, 119,1999.*

[47] H.G. Choi, A study on segregated finite element algorithms for the Navier–Stokes equations, *Ph.D. Thesis, Seoul National University, 1996.*

[48] D.L. Young, Y.S. Liu, Finite element analysis of free surface flows, in: W.L. Hogarth, B.J. Noye (Eds.). *Computational Techniques and Applications, Hemisphere Pub, Washington, DC, USA, p. 461,1990.*

[49] A. Masud, T.J.R. Hughes, A space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations for moving domain problems. *Comput. Meth. Appl. Mech. Eng. 146, 91,1997.*

[50] S.E. Navti, K. Ravindran, R.W. Lewis, D.C. Taylor, Finite element modeling of surface tension effects using a Lagrangian– Eulerian kinematic description. *Comput. Meth. Appl. Mech. Eng. 147, 41, 1997.*

[51] X. Cai, H.P. Langtangen, B.F. Nielsen, A. Tveito, A finite element method for fully nonlinear water waves. *J. Comput. Phys. 143,544, 1998.*

[52] S.R. Idelsohn, E. Onate, C. Sacco, Finite element solution of free-surface ship-wave problems, Int. J. Numer. Meth. Eng. 45 (1999) 503. problems, *J. Comput. Phys. 154 , 497, 1999.*

[53] B.T. Helenbrook, L. Martinelli, C.K. Law, A numerical method for solving incompressible flow problems with a surface of discontinuity. *J. Comput. Phys. 148, 366, 1999.*

[54] J.C. Tannehill, D.A. Anderson, and R.H. Pletcher, Computational Fluid Mechanics and Heat Transfer, *Taylor and Francis, 1997.*
[55] Raouf. A. Ibrahim, Liquid Sloshing Dynamics: Theory and Applications. *Cambridge University Press Taylor and Francis, 2005.*
[56] Michael Schäfer, Introduction to Numerical Methods, *Springer Berlin Heidelberg New York, 2006.*
[57] Shibata H., Sato H., and Shigeta T, A seismic design of machine structure, *Proc. 3 WCEE II,552–562, 1975.*
[58] Shibata H., Shinozaki Y., Kobayashi N., and Mieda T., A study of the liquid slosh response In horizontal cylindrical tanks, *ASME Proc. Pressure Vess. Piping Conf., PVP-108, 137–142, 1986.*
[59] Shepherd R. (1969), Earthquake resistant design of petroleum storage tanks, *Proc. 2nd Austral.Conf. Mech. Struct. and Mat., Adelaide, 8.1–8.11, 1969.*
[60] Hanson R. D. , Behavior of liquid storage tanks: the great Alaska earthquake of 1964,*National Academy Sci. Washington D. C., 331–339, 1973.*
[61] Haroun M. A. and Housner G. W. , Earthquake response of deformable liquid storage tanks, *ASME J. Appl. Mech. 48, 411–418,1981.*
[62] Haroun M. A. and Housner G. W. , Dynamic interaction of liquid storage tanks and foundation soil, *Proc. 2nd ASCE/EMD Specialty Conf. Dynamic Response of Structures, Atlanta, Georgia, 1981.*
[63] Haroun M. A. and Housner G. W. ,Complications in free vibration analysis of tanks, *ASCE J. Eng. Mech. Div. 108(EM5), 801–818, 1982.*
[64] Haroun M. A. and Mourad S. A. , Buckling behavior of liquid filled shells under lateral seismic shear, in Flow–Structure Vibration and Sloshing, *ASME Pressure Vess. Piping Conf.,PVP- 191, 11–17, 1990.*
[65] Haroun M. A., Mourad S. A., and Pence P. W. ,Vibration suppression through liquid oscillations, *Mechanics, 1991.*
[66] Hatano T. and Konno H. , Numerical solution of hydrodynamic pressures during earthquakes on arch dam, *Trans. JSCE 131, 19–23, 1966.*
[67] Clough D. P. and Niwa A., Static tilt tests of a tall cylindrical liquid storage tank, *Rept. No UCB/EERC-79/06, UCA, Berkeley, 1979.*
[68] Clough R. W., Niwa A., and Clough D. P., Experimental seismic study of cylindrical tanks,*ASCE J. Struct. Division 105(ST12), 1979.*
[69] Jean Donea and Antonio Huerta, Finite Element Methods for Flow Problems, *John Wiley & Sons, Ltd, 2003.*
[70] J. Chung, G.M. Hulbert, A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-a method, *J. Appl. Mech. 60 , 371–375, 1993.*
[71] K.E. Jansen, C.H. Whiting, G.M. Hulbert, A generalized-a method for integrating the filtered Navier–Stokes equations with a stabilized finite element method, *Comput. Methods Appl. Mech. Engrg. 190 ,305–319, 2000.*
[72] W.G. Dettmer, D. Peric´, An analysis of the time integration algorithms for the finite element solutions of incompressible Navier–Stokes equations based on a stabilised formulation, *Comput. Methods Appl. Mech. Engrg. 192 ,1177–1226, 2003.*