TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

INTRODUCTION

Technological advances in recent years in the field of computer science for both hardware and software have contributed to the use of numerical simulations for the prediction of fluid flow. Numerical simulations can be combined with experimental methods to considerably reduce the time and cost of benchmark creation and the development and improvement costs of products. These benefits of numerical simulation software and new computer technologies on experimental methods have contributed to the development of new robust numerical methods in the field of "computational fluid dynamics" (CFD). Numerical simulations often provide accurate results more quickly and cheaply than experimental studies. The simulation of multiphase flows, which is a constituent of CFD, takes advantage of the computer technology benefits of both software and hardware by developing new numerical approaches to analyse these complex flows.

In physics and chemistry, the phase concept is used to specify the diverse states of a system. The phase concept or notion is also known by the term "state of matter". There are three states of matter: gas, liquid and solid. A system consists of a single phase if it is entirely homogeneous, both physically and chemically. Otherwise, each of the various homogeneous parts that may be continuous or dispersed within each other can be called a phase.

A multiphase flow is the simultaneous flow of several phases. The two-phase flow is a special case of a multiphase flow where there are only two phases. Two-phase models can be classified according to the natures of the phases in the flow:

- gas-solid flow;
- liquid-solid flow;
- liquid-liquid flow (immiscible liquids); and
- gas-liquid flow.

0.1 Problem

Multiphase flows are present all around us, and there are many examples that we can find both in nature and in technical and industrial processes. For instance, for the flow of gas and liquid mixtures, we have studies of clouds, raindrops, liquid aerosol injection, inkjet printing, the free surface flow of a liquid (river or sea), moulding, liquid sloshing in a tank, air and water mixing through porous soil in geothermal wells, gas and oil mixtures in petroleum extraction, bubble column reactors, steam generators, turbines, electrohydraulic dam design and the hydrodynamics of ships. Studies of mixtures of immiscible liquids of importance include water and oil and pollutant transport after a spill of oil in the ocean.

Phase changes may also result in a transition from a liquid flow to a mixed gas and liquid flow and vice versa. This phenomenon is found in the boiling process of a liquid in a boiler, the liquefaction of saturated vapour exiting a turbine to form a mixture of water and saturated vapour, and the cavitation of a hydraulic pump. Cavitation is an important phenomenon that must be taken into account in the design of hydraulic components and hydraulic systems such as pumps and valves. Cavitation is induced by a pressure drop that is lower than the pressure of the saturated vapour. In the case of pump cavitation, gas bubbles are formed, leading to a loss of efficiency that influences the operation of the device. When there is cavitation in a hydraulic pump, the implosion of the bubbles can cause the tearing of the material on the blades of the pump and thereby drastically damage the pump.

Spraying liquid into fine droplets in the air is another application widely used in the food industry to clean food or to dry a fluid injected by atomisation in hot air to recover it in a solid form, for example, for the production of milk or powdered juice. Liquid spraying can also be found in the injectors that supply fuel to a combustion chamber.

This flow example list is not exhaustive, and we could add many other cases of multiphase flow and phase transition in nature and in industrial processes.

Numerical methods allow us to avoid experimental methods in the study of multiphase flows, which are often difficult to implement. The pressure, velocity and temperature measurements can be obtained by average or local measurements, but the measuring instruments often have

an impact on the fluid flow or require physical access to the field flow. In the case of multiphase flows, the discontinuity of properties between the phases can make the data measurement difficult. For these reasons, computer simulations are very popular currently for the study of multiphase flows. Numerical methods allow us to reproduce and analyse physical phenomena to have a better understanding of them. Numerical simulations let us extrapolate a simulation to other cases after their validation using experimental data. Research on two-phase flows involves many difficulties because of the technical complexity of the phenomena being modelled and the poor resolution of these models using numerical methods.

The modelling of multiphase flow requires not only to solving the Navier-Stokes equations but also finding the position of the interface between the phases. This is not an easy task, not only because of the complexity of the discontinuity of the physical properties at the interface between the phases, resulting in the nonlinearity of the Navier-Stokes equations governing the flow field but also the difficulty of keeping the mass (area or volume) of each phase of the domain constant over time during the numerical simulation. Furthermore, in the case where the surface tension between the phases is taken into account, the level of complexity needed to solve the fluid flow equations in diphasic flow increases considerably. In two-phase flow, there is an interface between the two phases that distinctly separates the two phases into two subdomains. The difference between the fluid properties across the interface presents major difficulties in the resolution of this type of problem because it induces a velocity gradient jump (continuity of shear) and / or a jump in pressure (surface tension) on both sides of the interface. In this work, we use the finite element method for the finite discretisation of the problem. The solution of the equations in 2D is done by the eXtended Finite Element Method (XFEM) with the Crank-Nicholson scheme for time discretisation and a quadratic space discretisation by a grid of triangular elements. Because of the high Reynolds number in some cases, the Galerkin formulation becomes unstable. Stabilisation techniques such as Streamline-Upwind Petrov-Galerkin (SUPG) or Galerkin/Least-Squares (GLS) must be applied in this case to obtain a solution without oscillations.

The numerical solution of problems of immiscible phase flow can be classified into two basic approaches known as Eulerian (“interface capturing”) and Lagrangian (“interface tracking”).

In the Eulerian method, the properties are observed at a fixed point in space, while in the Lagrangian approach, the particles are tracked.
For example, an experimental Eulerian approach would entail taking, at a fixed point in a pipe, the measurement of the velocity with an anemometer or the measurement of the temperature with a thermocouple, giving the variation of the velocity or the temperature, respectively, at that fixed location over time. Among the Eulerian numerical methods ("interface capturing") that describe the interface implicitly by an auxiliary field defined in the entire domain, we can cite the level set method and the Volume of Fluid method (VOF). In contrast, in the case of a Lagrangian approach ("interface tracking"), the particles are followed. In this case, the velocity or the temperature of a particle is a function of time depending on its path. By analogy, numerical methods of monitoring the interface ("interface tracking") take into account the displacement of the interface by aligning elements that are at the interface during the simulation.

In two-phase flow, there is a discontinuity of the fluid properties through the interface. In the case of the interface-tracking method, this physical property discontinuity does not require additional treatment when the edges of all mesh elements located at the interface position are perfectly aligned with the interface. Furthermore, the interface-capturing method requires special treatment of the discontinuities across the interface because of the jump in the properties along the interface. The standard function forms of the finite element method are not able to correctly reproduce a discontinuous solution. In general, to overcome this problem by using the method of standard finite elements, there are two possible solutions that could be applied to a small band around the interface: refine the mesh on a narrow band around the interface or smooth the physical property discontinuity at the interface. These two methods can be used separately or combined together. However, the mesh refinement approach and the smoothing of physical properties at the interface may induce diffusion effects that could spread from the interface to the entire domain. To avoid the re-meshing and the diffusion effect at the interface, a new method called XFEM, which is based on the enrichment of the approximation space by the Partition of Unity Method (PUM) to take into account the discontinuities of the elements crossed by the interface, was introduced. The approximation space of a classical finite

element is enriched locally by functions that allow the representation of an exact approximation of the discontinuities in the elements cut by the interface. However, the enrichment increases the number of degrees of freedom of the enriched elements and thereby the cost in computation time.

The discretisation of the governing equations is done in time with the semi-implicit Crank-Nicolson scheme, while the spatial discretisation is performed with quadratic triangular elements. The stabilisation method Streamline-Upwind Petrov-Galerkin (SUPG) is used to control the stability problems incurred with the Galerkin formulation method in the case of advection problems.

0.2 Goal (objective)

The main objective of our research is to develop a numerical method that allows obtaining accurate numerical solutions of two-phase flow problems. In detail, we are interested in developing stabilised finite element methods for the level set equation, a correction method for the interface to conserve mass, and a coupling technique of these methods with a Navier-Stokes solver based on the eXtended Finite Element Method (XFEM).

0.3 Announcement of the methodology

The level set method was chosen to follow the evolution of the interface. The level set method, which is a Eulerian method, has the advantage of supporting topological changes such as the coalescence and separation of a phase, in contrast to the Lagrangian methods where an arbitrary criterion must be set to detect the coalescence or separation of a phase. However, the level set method has a drawback that could be significant in some cases. Although the topological changes are intrinsic to the level set method, the level set function initialised as a signed distance can move away from the signed distance property through the simulation depending on the difficulty or the duration of the problem simulation. Currently, the classic method used to maintain the level function as a signed distance function is to reset the level set function

after a certain number of steps selected in an ad hoc manner. This reinitialisation can be done in the whole domain or on a narrow band that is located on both sides of the interface.
We will propose methods for stabilising the level set equation to improve the accuracy of the numerical solution so that the level set function remains as close as possible to a distance function and the proposed methods reduce the use of reinitialisation.
On the other hand, the fact that the level set method does not keep the mass constant has led us to apply a method of correcting the level set function to improve the mass conservation by moving the level set field; this correction involves a small displacement of the interface.
The eXtended Finite Element Method (XFEM) is used instead of the conventional finite element method because of the discontinuity of the properties at the interface, which is best represented by shape functions with the enrichment of the space approximation by the eXtended Finite Element Method (XFEM).
The discretisation of the governing equations is done in time with the semi-implicit Crank-Nicolson scheme, while the spatial discretisation is performed with quadratic triangular elements. The stabilisation method Streamline-Upwind Petrov-Galerkin (SUPG) is used to control the stability problems incurred with the Galerkin formulation method in the case of advection problems. The strong coupling between the velocity and pressure fields and the level set field is achieved through an iterative algorithm for solving the Navier-Stokes equation followed by an algorithm for the resolution of the level set equation.

0.4 Plan of the thesis

The thesis has four chapters. The Introduction sets out the background work, describes the research objectives and sets out the thesis plan. CHAPTER 1 describes the state of the art in numerical methods for solving two-phase flows with the classical finite element method and the extended finite element method. This chapter shows the history, progress and usefulness of the research in the field of two-phase flows. We will highlight the areas of application of these methods, as well as the advantages and disadvantages of some techniques. CHAPTER 2 presents different methods for solving the transport problem of the level set function. The classical methods of solving the level set transport equation (section 2.1) and methods of

resetting the level set field are described (section 2.2). The various methods of resetting that are presented are the geometric method (2.2.1), the fast marching method (2.2.2) and the Eikonal reinitialisation method (2.2.3). CHAPTER 3 sets out the proposed stabilisation methods. CHAPTER 4 shows the resolution of the Navier-Stokes equations by the extended finite element method (XFEM) for the spatial discretisation and elaborates upon the mass conservation methods (section 0). The Conclusions succinctly summarise the results of the research and its lessons and outlines proposals and prospects for future research.

CHAPTER 1

LITERATURE REVIEW

1.1 The finite element method

The finite element first appeared in the 1950s in solid mechanics. The use of the finite element method is motivated by the need to solve problems that conventional methods such as strength of materials or machine components cannot resolve because of the complexity of the geometry or the structure. The finite element method was first applied to fluid mechanics in the 1970s. This delay in the use of the finite element method between solid mechanics problems and fluid mechanics problems is the result of, first, the strong coupling between the velocity and pressure and, second, the presence of a convective term in the Navier-Stokes equations, which implies a non-linearity. Furthermore, if the Reynolds number is very high, we could have turbulence phenomena. Among the early work on finite elements, we can include the work of Zienkiewicz and Cheung (Zienkiewicz et Cheung, 1965) in 1965, which presents the finite element method applied to the resolution of a quasi-harmonic partial derivative equation that one finds in conduction heat transfer problems and torsion of prismatic cross-section problems. In 1982, an algorithm based on the method of characteristics was proposed by Pironneau (Pironneau, 1982) to solve the transport-diffusion equation (convection-diffusion) and the Navier-Stokes equations, with a spatial discretisation made using Taylor-Hood elements. A new Streamline Upwind/Petrov-Galerkin (SUPG) formulation, which consists of adding diffusion or viscosity that acts only in the direction of the flow, was proposed (Brooks et Hughes, 1982). The SUPG method is more robust than the Galerkin method. A method called Galerkin/Least-Squares (GLS), which consists of adding the least squares form of the residual to the Galerkin method (Hughes, Franca et Hulbert, 1989), was presented, and it improves the stability while preserving the accuracy of the solution. Hughes et al. (Hughes, Franca et Balestra, 1986) presented a new Petrov-Galerkin formulation to solve the Stokes problem by introducing a stabilisation that avoids the use of Taylor-Hood mixed interpolations. This formulation provides a stable and convergent solution using a C^0 approximation of the same order for the velocity as well as the pressure in the case of the Stokes fluid flow problem. In 1998,

“Discontinuous Galerkin Method” (DGM) formulations were proposed to solve diffusion problems (Oden, Babuŝka et Baumann, 1998) and convection-diffusion problems (Baumann et Oden, 1999). This hp-discontinuous finite element method is conservative, the order of approximation of the elements is adjusted by an element and the stability of the method does not require the insertion of an artificial diffusion term.

1.2 Interface problem solving methods

The numerical methods that allow solving the interface transport problem are generally classified into different approaches that are classified as Eulerian methods, Lagrangian methods and Eulerian-Lagrangian methods. The Eulerian methods include the “front-tracking” methods and the “front-capturing” methods. “Front-tracking” is based on markers and can be divided into two groups: “volume-tracking” and “surface-tracking” methods. In the “volume-tracking” methods such as “marker and cell method” (MAC), markers are distributed in all phases of the domain, while with the “surface-tracking” methods, markers are distributed only on the interface. During the simulation, the markers can gather in certain areas, resulting in the need for the redistribution of markers in the domain for the “volume-tracking” methods or on the interface for the “surface-tracking” methods to preserve the precision of the numerical results. The MAC method, which consists of distributing the markers in the domain, was applied to dam break problems (Harlow et Welch, 1965) . The “surface-tracking” methods are generally more accurate than the “volume-tracking” methods to return the interface position. “Surface tracking” methods have been developed by Glimm et al. to simulate shock waves in compressible flows (Glimm, 1982; Glimm et McBryan, 1985), and more recently, Unverdi and Tryggvason developed techniques to solve the Navier-Stokes equations when they are applied to the case of bubbles with a surface tension effect (Unverdi et Tryggvason, 1992). In the Lagrangian approach, the mesh is entirely or partially mobile and is updated at each time step. Lagrangian methods offer accurate solutions, but they are difficult to implement in 3D, in particular in the case of complex deformations of the interface. In the Lagrangian-Eulerian methods, such as the “Arbitrary Lagrangian-Eulerian formulation” (ALE) (Hu, Patankar et Zhu, 2001), the nodes of the mesh can be moved with the fluid according to the Lagrangian

method or can be held stationary according to the Eulerian method. Therefore, some mesh nodes can be moved from an arbitrarily specified criterion to provide a continuous rezoning functionality. Because of this freedom in the movement of the mesh in the ALE formulation, some cases of larger distortions within the domain can be treated, whereas a purely Lagrangian method or purely Eulerian method would not provide as much precision in the results.

The methods of the "front-capturing" are commonly used to study two-phase flow because they represent natural topological changes without additional ad hoc parameter settings. The most common "front-capturing" techniques are the Volume Of Fluid method (VOF) (Hirt et Nichols, 1981; Noh et Woodward, 1976; Youngs, 1982) and the level set method (Osher et Sethian, 1988). These last two methods are based on solving a scalar transport equation. The scalar transported by the VOF method and the level set method are the volume fraction and the level set function, respectively. Among the methods used for VOF, we can cite the method of "Simple Line Interface Calculation" (SLIC) (Noh et Woodward, 1976) and the method of "Piece Linear Interface Calculation" (PLIC) (Rider et Kothe, 1998), which are techniques used to find the position of the interface in the VOF method where the unknown variable is the ratio of each phase in a cell or element. In the level set method, introduced by Osher and Sethian (Osher et Sethian, 1988), the transported scalar is defined as a continuous function corresponding to the signed distance from the interface. Because of numerical errors that occur during the resolution of the level set transport equation, several reinitialisation processes have been suggested to allow the level set function to keep its signed distance property (Adalsteinsson et Sethian, 1999; Chunming et al., 2010; Gross et Reusken, 2011; Hysing et Turek, 2005; Olsson et Kreiss, 2005; Olsson, Kreiss et Zahedi, 2007; Osher et Fedkiw, 2003; Qian, Zhang et Zhao, 2007; Sethian, 1996; Sethian, 1998; So, Hu et Adams, 2011; Sussman, Smereka et Osher, 1994; Zhao, 2005). The main difficulty when re-distancing the level set function or using the approach of the mass conservation is that these methods can slightly move the position of the interface according to their frequency of use (Gomes et Faugeras, 2000). In our proposed technique (Touré et Soulaïmani, 2012; 2016), we combine the level set function transport equation and the reinitialisation equation in one variational formulation. However, the level set method can cause quite significant mass loss in some cases. There are many ways to overcome this weakness, such as

- The application of a mass conservation method after the reinitialisation of the level set function;
- The coupling of the level set method with the VOF method that conserves the mass in an intrinsic way; or
- The global refinement of the mesh or the local refinement of the mesh in the vicinity of the interface.

Several studies were conducted to enforce the mass property conservation in the context of the application of the level set method (Ausas, Dari et Buscaglia, 2011; Desjardins, Moureau et Pitsch, 2008; Di Pietro, Lo Forte et Parolini, 2006; Doyeux et al., 2013; Kees et al., 2011; Kuzmin, 2014; Laadhari, Saramito et Misbah, 2010; Le Chenadec et Pitsch, 2013; Olsson et Kreiss, 2005; Olsson, Kreiss et Zahedi, 2007; Owkes et Desjardins, 2013; Smolianski, 2001; Sussman, 2003; Sussman et Puckett, 2000; van der Pijl et al., 2005; Wang, Simakhina et Sussman, 2012). In our two-phase flow studies, only one of the two phases is forced to preserve its global mass (Gross et Reusken, 2011; Smolianski, 2001).

1.3 The methods of solving the Navier-Stokes equations by the eXtended Finite Element Method (XFEM)

The Navier-Stokes equations for two-phase flow can be solved by the “eXtended Finite Element Method” (XFEM) (Chessa et Belytschko, 2003a) or the ghost fluid method (Ménard, Tanguy et Berlemont, 2007), which takes into account the discontinuities of the physical properties of elements cut by the interface. The XFEM is established based on the partition of unity method (Babuška et Melenk, 1997; Babuška, Caloz et Osborn, 1994; Melenk et Babuška, 1996), where the discontinuities of properties in the finite elements are treated through the enrichment of the shape functions by adding degrees of freedom.

We find the practical application of XFEM in fluid mechanics problems in many areas and past research (Chessa et Belytschko, 2003a; Chessa et Belytschko, 2003b; Fries, 2009; Fries et Belytschko, 2006; Groß et Reusken, 2007; Gross et Reusken, 2011; Reusken, 2008). Chessa and Belytschko proposed solving the Navier-Stokes equations for a two-phase flow with and without the effect of surface tension by the XFEM method using the “abs-enrichment”

formulation, which consists of introducing the absolute value of the level set function into the shape functions to reflect the jump in the velocity field at the interface (Chessa et Belytschko, 2003a; Chessa et Belytschko, 2003b). Gross and Reusken proposed the enrichment of the pressure approximation space using a Heaviside function (Groß et Reusken, 2007). Their XFEM method was applied to 3D two-phase flows with surface tension. An adaptive mesh and a mesh refinement at the interface are also presented. Fries and Belytschko (Fries et Belytschko, 2006) introduced the intrinsic XFEM, which is applied to two-phase flow (Fries, 2009). The intrinsic XFEM, unlike the extrinsic XFEM, does not require the addition of supplementary unknowns. Coppola-Owen and Codina (Coppola-Owen et Codina, 2005) introduced an enrichment function for discontinuous pressure gradients of two-phase flow, which are zero at the nodes of the cut elements whose gradient is constant on either side of the interface. Zlotnik and Díez (Zlotnik et Díez, 2009) generalised the "abs-enrichment" function according to Moës et al. (Moës et al., 2003) for multiphase flow problems by introducing a hierarchical collection of (n-1) level set functions for n phases in the domain. The approximation spaces of the velocity and the pressure are enriched for the numerical simulation of the Stokes problem. The coupling of the "abs-enrichment" method of Moës and the enrichment method using the Heaviside function were investigated for incompressible fluids (Legrain, Moës et Huerta, 2008).

Although the XFEM method ensures a better approximation than the standard finite element method for discontinuous solutions, we can face convergence problems with the iterative numerical methods. Indeed, XFEM may tend to poorly conditioned matrix systems, especially when the interface is very close to a node or when the ratio between the volumes of the two sides of the interface of the cut elements is very high (Fries et Belytschko, 2010). On the other hand, the "Generalised Finite Element Method" (GFEM) is a "Partition of Unity Method" (PUM), where the space of the Finite Element Method (FEM) is increased by using non-polynomial shape functions with compact support. Recently, Babuška and Banerjee presented a new method called the "Stable Generalised Finite Element Method" (Stable GFEM/XFEM) (Babuška et Banerjee, 2012). The Stable XFEM method was applied to two-phase flows (Sauerland et Fries, 2013). A preconditioner can be used so that the matrix systems are well-conditioned. Bechet et al. applied a preconditioner based on a local Cholesky decomposition

(Béchet et al., 2005), while Menk and Bordas (Menk et Bordas, 2011) used a preconditioner related to the domain decomposition procedure. Another method consists of moving the nodes of the cut elements in which the interface is very close to one of their nodes to obtain a proportional volume ratio of the two sides of the interface of the cut elements that can greatly improve the conditioning of the matrix system (Choi, 2011).

1.4 The coupling between the Navier-Stokes and the level set equation

The coupling between the velocity/pressure field and the level set field can be done by
- a resolution of a unique matrix system including the Navier-Stokes equations and the level set transport equation;
- a weak coupling of the Navier-Stokes equations whose unknowns are the couple of velocity and pressure obtained from the previous level set solution step and then doing an update of the level set function; or
- a strong coupling by performing iterations of the weak coupling method until the equilibrium (or balance) of the velocity, pressure and level set fields is achieved.

The weak coupling can lead to non-physical results, according to Akkerman et al. (Akkerman et al., 2011). The strong coupling is less expensive than the resolution of the Navier-Stokes equations and transports the level set function equation combined in a unique matrix system. Strong coupling was used in our case of a two-phase flow study.

CHAPTER 2

CLASSIC LEVEL SET METHODS

2.1 The level set method

The level set method was introduced in 1988 by Osher and Sethian (Osher et Sethian, 1988). The starting point of this method is the definition of a level set scalar function ϕ. The zero value of the level set function is the interface that is transported by the velocity field. The contours of the level set function initialized as a distance function may move away from the level set distance function due to the accumulation of numerical errors; hence, in classical reinitialisation methods the solution needs to be reset after a certain number of time steps.
The moving interface $\Gamma(t)$ is defined as the zero-level for the scalar function $\phi(\mathbf{x},t)$:

$$\Gamma(t)=\left\{x\in \mathrm{R}^3 \mid \phi(x,t)=0\right\}. \tag{2.1}$$

For example, in a two-phase flow, as illustrated in Figure 2.1, the entire domain $\Omega=\Omega_1\cup\Omega_2$ is divided into two subdomains Ω_1 and Ω_2 separated by the interface $\Gamma(t)$ according to the sign of the level set function:

$$\begin{cases} \phi(\mathbf{x},t)<0 \; if \; \mathbf{x}\in\Omega_1 \\ \phi(\mathbf{x},t)=0 \; if \; \mathbf{x}\in\Gamma \\ \phi(\mathbf{x},t)>0 \; if \; \mathbf{x}\in\Omega_2 \end{cases}. \tag{2.2}$$

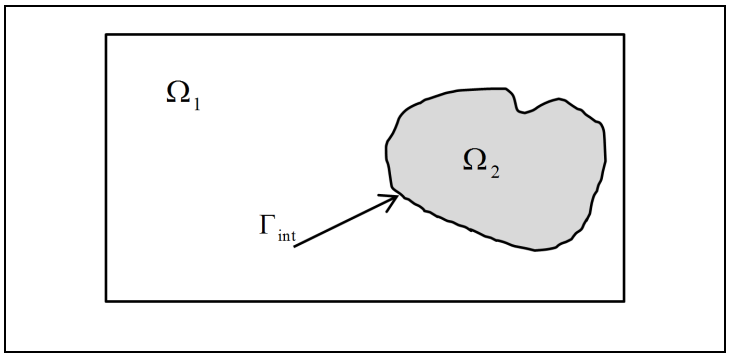


Figure 2.1: Two-phase domain.

Most often, the level set function is the signed distance function, which is defined by:

$$d_{signed}(\mathbf{x},t)=\begin{cases}-d(\mathbf{x},t) & \text{if } \mathbf{x}\in\Omega_1\\ 0 & \text{if } \mathbf{x}\in\Gamma\\ d(\mathbf{x},t) & \text{if } \mathbf{x}\in\Omega_2\end{cases} \tag{2.3}$$

with

$$d(\mathbf{x},t)=\min_{\forall \mathbf{x}_\Gamma\in\Gamma}\left(\left|\mathbf{x}-\mathbf{x}_\Gamma\right|\right) \tag{2.4}$$

The signed distance function has the following useful property:

$$\left|\nabla d_{signed}\right|=1 \tag{2.5}$$

which is an Eikonal equation. The level set function is initialized by the signed distance, $\phi(\mathbf{x},0)=d_{signed}(\mathbf{x},0)$, and its evolution is governed by the transport equation:

$$\frac{D\phi(\mathbf{x},t)}{Dt}=\frac{\partial\phi(\mathbf{x},t)}{\partial t}+\mathbf{u}\cdot\nabla\phi(\mathbf{x},t)=0 \tag{2.6}$$

or

$$\frac{\partial\phi}{\partial t}+F\left|\nabla\phi\right|=0 \tag{2.7}$$

where $F=u_n=\dfrac{\mathbf{u}\cdot\nabla\phi}{\left|\nabla\phi\right|}$ is the velocity component in the normal direction to the isocontours of ϕ. Equation (2.6) or (2.7) is called the level set equation (Osher et Sethian, 1988).

The unit (outward) normal vector is determined by $\hat{\mathbf{n}}=\dfrac{\nabla\phi}{\left|\nabla\phi\right|}$ and the mean curvature κ is defined as the divergence of $\hat{\mathbf{n}}$ so that $\kappa=\nabla\cdot\hat{\mathbf{n}}=\nabla\cdot\left(\dfrac{\nabla\phi}{\left|\nabla\phi\right|}\right)$. Since the interface is moving with a conservative flow field, the total area is conserved for both phases:

$$\begin{cases}\dfrac{D\Omega_1}{Dt}=0 \text{ if } \phi(\mathbf{x},t)<0\\ \dfrac{D\Omega_2}{Dt}=0 \text{ if } \phi(\mathbf{x},t)>0\end{cases} \tag{2.8}$$

These equations (2.8) are also known as the global mass conservation equations for each phase.

2.2 The reinitialisation methods

The numerical discretisation of the level set transport equation (2.6) does not necessarily conserve the signed distance function property (2.5). However, this latter property is essential to maintain good accuracy in the calculation of geometric quantities related to the interface and in the determination of the position of the interface. In the case of two-phase flows where the velocity fields can significantly stretch or tighten the level set, it is necessary to apply a constraint on the evolution of ϕ in such a way that $\left|\nabla \phi_{signed}\right|$ remains close to 1 without significantly moving the interface Γ.

2.2.1 The geometric reinitialisation method

The geometric reinitialisation is a "brute force search algorithm" and is described as follows: The elements crossed by the interface Γ are first found (Figure 2.2). We suppose that any element crossed by the interface has only two points with a zero level such that each point belongs to a different edge. Next, a piecewise linear approximation Γ^h of the interface is computed by joining these zero-level points, as illustrated by the blue segments in Figure 2.2. The nodes belonging to the crossed elements (cut elements) are labelled as "un-updated nodes" because their level set values are unchanged from the results computed by solving (2.6). For the other nodes (labelled as updated nodes), the level set values are corrected and set to the recomputed signed distance to the interface Γ^h. When this reinitialisation algorithm is used at each time step, it is expected to produce accurate results that can serve as references for comparison. Therefore, we do not expect or claim that the proposed stabilisation algorithms will perform better, but rather will produce acceptable results with simplified implementation.

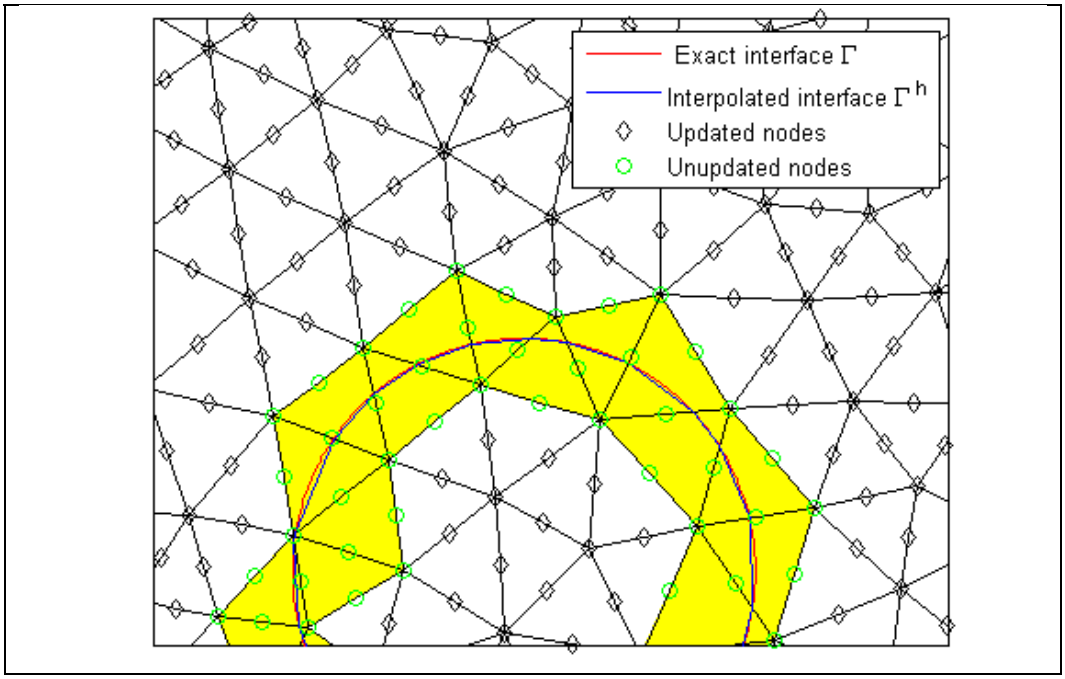


Figure 2.2: Interpolation of the interface.

2.2.2 The fast marching method ("The boundary value formulation")

The fast marching method (FMM) (Sethian, 1996; Sethian, 1998) is a popular technique for reinitialisation and describes an interface that only expands or only contracts. It is expressed with the boundary value formulation,

$$|\nabla T| F = 1 \tag{2.9}$$

where T is the arrival time function and $F(\mathbf{x})$ is the normal velocity u_n at the interface. This equation is an Eikonal equation and applies only when the front moves in one direction. The arrival function $T = T(\mathbf{x})$ gives the duration that the front starting from the interface takes to reach the point $\mathbf{x} = (x, y)$. This arrival function is the basic function of the fast marching method (FMM) (Sethian, 1996; Sethian, 1998). Furthermore, we have the relation $\phi(\mathbf{x}, t) = T(\mathbf{x}) - t$. If the front moves in both directions, that is to say, $F(\mathbf{x})$ can change sign,

then we must use the Eikonal reinitialisation method (also known as the "Initial Value Formulation") (see Section 2.2.3).

The proposed upwind scheme (Sethian, 1998) for the 3D spatial discretisation is

$$\begin{bmatrix} \max\left(D_{ijk}^{-x}T, -D_{ijk}^{+x}T, 0\right)^2 + \\ \max\left(D_{ijk}^{-y}T, -D_{ijk}^{+y}T, 0\right)^2 + \\ \max\left(D_{ijk}^{-z}T, -D_{ijk}^{+z}T, 0\right)^2 \end{bmatrix}^{1/2} = \frac{1}{F_{ijk}} \tag{2.10}$$

where D is the difference operator such as for example the forward difference operator is $D^{+x}\alpha = \frac{\alpha(x+\Delta x,t)-\alpha(x,t)}{\Delta x}$, the backward difference operator $D^{-x}\alpha = \frac{\alpha(x,t)-\alpha(x-\Delta x,t)}{\Delta x}$ and the centred difference operator is $D^{0x}\alpha = \frac{\alpha(x+\Delta x,t)-\alpha(x-\Delta x,t)}{2\Delta x}$.

In 2D, as illustrated in Figure 2.3, on a structured Cartesian grid of size element $\mathbf{h} = (\Delta x, \Delta y)$, we obtain the following spatial discretisation:

$$\begin{aligned} &\max\left(\frac{T_{i,j}-T_{i-1,j}}{\Delta x}, -\frac{T_{i+1,j}-T_{i,j}}{\Delta x}, 0\right)^2 + \\ &\max\left(\frac{T_{i,j}-T_{i,j-1}}{\Delta y}, -\frac{T_{i,j+1}-T_{i,j}}{\Delta y}, 0\right)^2 = \frac{1}{\left(F(x,y)\right)^2} \end{aligned} \tag{2.11}$$

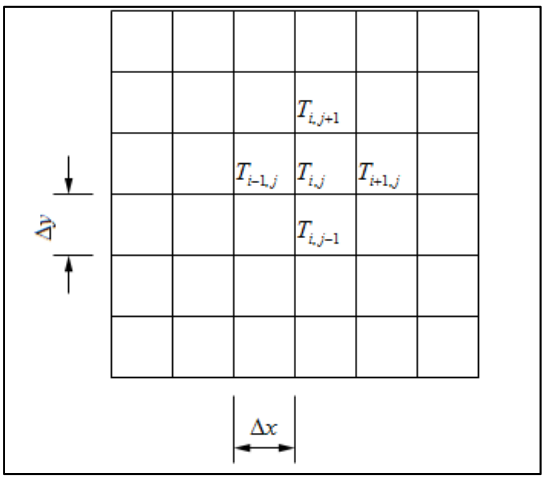


Figure 2.3: Example of 2-dimensional domain.

For this boundary value formulation, the normal velocity should be strictly positive ($F > 0$) or strictly negative ($F < 0$), regardless of the time. The FMM is useful when the interface can only move in one direction, while in the initial value formulation (see section 2.2.3), there is no limitation on the sign of the normal velocity F of the interface.

The goal of the method is to propagate a front from the interface along the outward normal direction by progressively computing the distance from the interface to the nodes of the mesh that are reached by the propagated front, as illustrated in Figure 2.4, where the front moves toward the southeast direction.

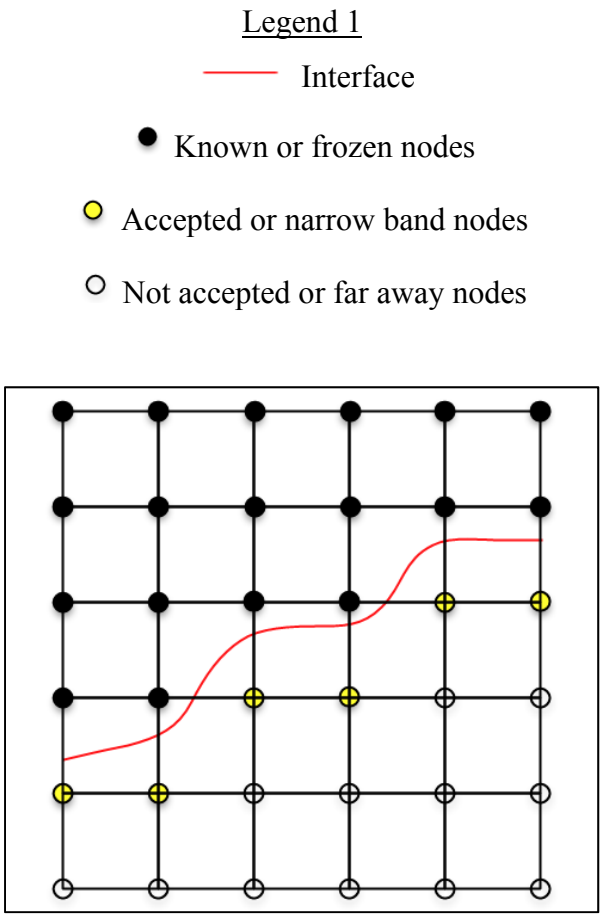


Figure 2.4: Configuration of the domain for the fast marching method (FMM).

The initialisation of the distance of the nodes is performed using, as a reference, the nodes of the elements that are cut or adjacent to the interface. The aim of the method is to determine the distance between the interface and the nodes that are adjacent to the nodes whose distance to the interface is already known. The front is moving along the outward normal direction with a normal velocity F .When the front crosses a new node of the mesh, it becomes part of the known nodes (or frozen nodes) set, and its signed distance will be computed. The distances of the closest nodes to the interface (accepted or narrow band nodes) that are on the opposite side of the known nodes are computed one after the other. The distances of the other nodes (not accepted or far away nodes) are temporary. They must be recalculated using the newly admitted nodes and their distance. The fast marching method (FMM) must perform quickly and recursively the following process.

Algorithm or procedure of reinitialisation by the fast marching method (FMM)

Initialisation step

1. Initialise the front by detecting the points on the interface. The nodes on the interface and upstream of the interface are labelled “Known nodes” (or “Frozen nodes”) ●, and their value of T is assigned to them. The nodes immediately adjacent to the “Known nodes” are labelled “Accepted nodes” (or “Narrow band nodes”) ○. The value of T of the “Accepted nodes” (or “Narrow band nodes”) is initialised by solving equation (2.11). All other nodes are labelled “Not accepted nodes” (or “Far away nodes”) ○, and they are on the downstream side of the “Accepted nodes” (or “Narrow band nodes”). The value T of the “Not accepted nodes” (or “Far away nodes”) is initialised numerically to $+\infty$.

Iterative loop stage

2. Start of the loop.

Determine the node p ("Trial node") of coordinates $\mathbf{x}_p$ such that the distance $T(\mathbf{x}_p) = \min_{\mathbf{x} \in Narrow\ Band} (T(\mathbf{x}))$ and then add the node to the Known (or Frozen) nodes and remove it from the "Accepted nodes" (or "Narrow band nodes").

3. Label all nodes that are adjacent to the node p to the "Accepted nodes" (or "Narrow band nodes") if they do not belong to the "Known nodes" (or "Frozen nodes"). If some of the neighbouring nodes that meet these last criteria are part of the "Not accepted nodes" (or "Far away nodes"), add them to "Trial nodes" and remove them from "Not accepted nodes" (or "Far away nodes").
4. Recalculate value T for all the neighbours of the node p by solving equation (2.11).
5. Return to the start of the loop at step 2.
   End of the loop

Sorting and efficiency of the algorithm

To speed up step 2 of the search algorithm, we use a complete binary tree including the nodes of the "Accepted nodes" (or "Narrow band nodes") and their temporary distance as shown in Figure 2.5 and
Table 2.1, where k is the position of the node and T is the nodal value.
The node p ("Trial node") that has the smallest distance is the root of the binary tree. When the node is removed from the tree, the other nodes are moved to the top of the tree to form a new tree.

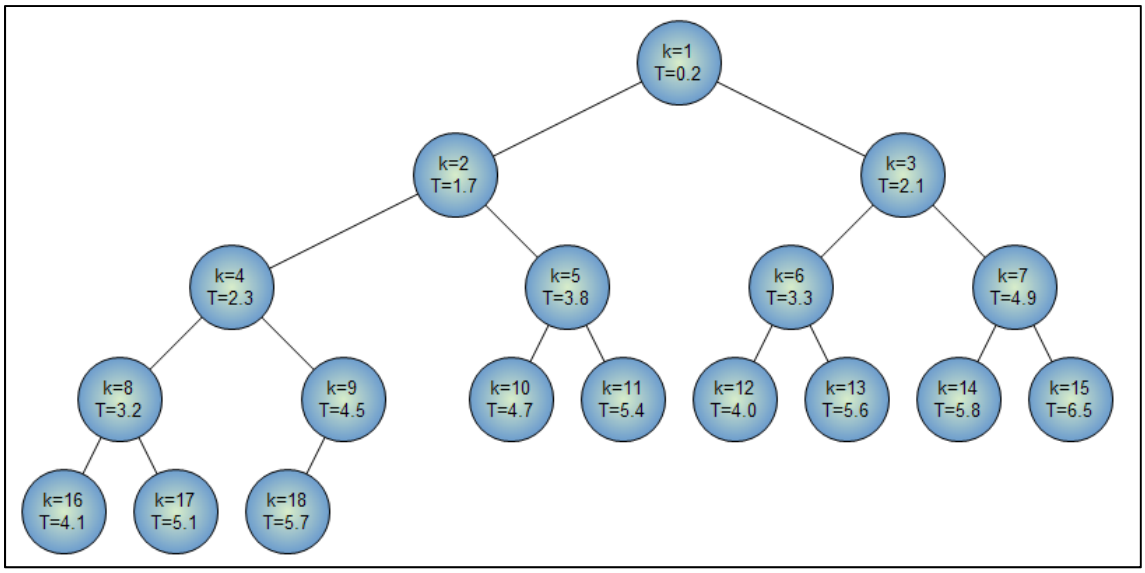


Figure 2.5: Arborescent structure of a binary tree.

Table 2.1: Structure of a binary tree in an array

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
T	0.2	1.7	2.1	2.3	3.8	3.3	4.9	3.2	4.5	4.7	5.4	4.0	5.6	5.8	6.5	4.1	5.1	5.7

Property of the binary tree implemented in an array

The access time to a node is $O(1)$. A balanced insertion or removal of a node in the tree spends $O(\log N)$ in time. The fast marching method calculates the distance of N nodes in $O(N \log N)$ in the worst case. For example, in a two- or three-dimensional mesh of n nodes in each direction, the total number of operations is $n^2 \log n$ or $n^3 \log n$, respectively.

2.2.3 The Eikonal reinitialisation method ("The initial value formulation")

The transport equation of the level set (2.6) is solved for a number of time steps using temporal discretisation and spatial discretisation. If the level set function is no longer a distance function because of error accumulation, a correction procedure is usually adopted, such as solving a

reinitialisation (or redistancing) equation as proposed by Sussman, Smereka and Osher (Sussman, Smereka et Osher, 1994),

$$\frac{\partial \varphi}{\partial \tilde{t}} + S(\varphi_0)\left(|\nabla \varphi| - 1\right) = 0 \tag{2.12}$$

where τ is a pseudo-time. Equation (2.12) is solved with the initial condition $\varphi(\mathbf{x},0) = \varphi_0 = \phi(\mathbf{x},t)$. At convergence, a corrected level set $\varphi(\mathbf{x},t)$ is obtained, satisfying $|\nabla \varphi| - 1 \approx 0$. The sign function $S(\varphi_0)$ is defined by

$$S(\varphi_0) = \begin{cases} 1 & si\ \varphi_0 > 0 \\ 0 & si\ \varphi_0 = 0 \\ -1 & si\ \varphi_0 < 0 \end{cases} \tag{2.13}$$

where $S(\varphi_0)$ can also be smoothed to $S(\varphi_0) = \frac{\varphi_0}{\sqrt{\varphi_0^2 + \varepsilon^2}}$, as proposed by Sussman et al. (Sussman, Smereka et Osher, 1994), or $S(\varphi_0) = \frac{\varphi_0}{\sqrt{{\varphi_0}^2 + |\nabla \varphi_0|^2 \varepsilon^2}}$, as suggested by Peng et al. (Peng et al., 1999).

The accuracy of this method depends on the time interval between the two reinitialisation steps. The time after which the level set solution is reinitialised should be chosen in a suitable manner. The reinitialisation process may move the zero level set contours from their initial position (Gomes et Faugeras, 2000). However, when the level set function $\phi(\mathbf{x},t)$ is too far from the signed distance function d_{signed}, the reinitialisation process may fail. It is correctly stated in Gross-Reusken (Gross et Reusken, 2011) that 'using this technique, one faces the following difficulties. Firstly, the method contains control parameters τ and ε, and there are no good practical criteria on how to select these. Secondly, the partial differential equation (2.12) is nonlinear and hyperbolic; accurate discretisation of this type of partial differential equation is rather difficult', and the stability and convergence depend on those parameters involved in the stabilised formulation.

In fact, we have implemented a finite element discretisation of the multiscale variational formulation proposed by Akkerman et al. (Akkerman et al., 2011; Kees et al., 2011), and we faced the same difficulties as those reported in Gross-Reusken (Gross et Reusken, 2011), namely, that the stability and convergence depend on the control parameters $\tilde{t}$ and ε and the parameters involved in the stabilised formulation. Akkerman et al. (Akkerman et al., 2011) proposed the formulation

$$\int_{\Omega} \eta^h \left(\frac{\partial \phi_d^h}{\partial \tilde{t}} + \mathbf{a} \cdot \nabla_x \phi_d^h - S_\varepsilon \left(\phi^h \right) \right) d\Omega + \int_{\Omega} \tau_{\phi_d} \mathbf{a} \cdot \nabla_x \eta^h \left(\frac{\partial \phi_d^h}{\partial \tilde{t}} + \mathbf{a} \cdot \nabla_x \phi_d^h - S_\varepsilon \left(\phi^h \right) \right) d\Omega = 0 \tag{2.14}$$

where

$$\mathbf{a} = S_\varepsilon \left(\phi^h \right) \frac{\nabla_x \phi_d^h}{\left| \nabla_x \phi_d^h \right|} \quad \text{and} \quad S_\varepsilon \left(\phi \right) = 2 H_\varepsilon \left(\phi \right) - 1 \tag{2.15}$$

H_ε is the regularised Heaviside function.

Akkerman et al. suggested to add a penalty term to equation (2.14) to maintain the interface at its position during the reinitialisation (Akkerman et al., 2011). The proposed penalty term is

$$+ \int_{\Omega} \eta^h \lambda_{penal} H_\varepsilon^{'} \left(\phi^h \right) \left(\phi_d^h - \phi^h \right) d\Omega \tag{2.16}$$

where λ_{penal} is a constant parameter.

The numerical tests

The following numerical tests were carried out to compare the accuracies of the proposed methods. First, in section 2.2.4, a numerical test was performed to evaluate the accuracy of the level set geometric reinitialisation and the fast marching method, described in section 2.2.1 and section 2.2.2, respectively, using disturbed level sets of a rectangular interface. In section 2.2.5, a test is performed to show the performance of the Eikonal reinitialisation previously presented in section 2.2.3. To analyse the solution accuracy, the following norms are defined:

$$E_1 = \int_{\Omega} \left| \phi(\mathbf{x},0) - \phi(\mathbf{x},T) \right| \, d\Omega \tag{2.17}$$

$$E_2 = \sqrt{\int_{\Omega} \left(\phi(\mathbf{x},0) - \phi(\mathbf{x},T) \right)^2 d\Omega} \tag{2.18}$$

$$E_3 = \frac{\sqrt{\int_{\Omega} \left(\phi(\mathbf{x},0) - \phi(\mathbf{x},T) \right)^2 d\Omega}}{\sqrt{\int_{\Omega} \phi^2(\mathbf{x},0) \, d\Omega}} \tag{2.19}$$

$$E_4 = \int_{\Omega} \left(\left| \nabla \phi(\mathbf{x},T) \right|^2 - 1 \right) d\Omega \tag{2.20}$$

where $\phi(\mathbf{x},0)$ is the finite element interpolation of the exact solution at the initial time. Furthermore, to assess the mass (or area) conservation of a phase, the following percentage is defined as

$$Mass(\%) = 100 \left(\frac{\text{area}(t)}{\text{area}(t=0)} \right) \tag{2.21}$$

2.2.4 The comparison of reinitialisation methods by the geometric method and the fast marching method

The interface Γ is defined as a rectangle. The nodal level set values are then obtained by computing the signed distance to the rectangular interface. The disturbed level set function is found by adding to the exact level set function a perturbing term, $\frac{r}{20} \sin\left(\frac{\pi d}{2r} \right) \sin(20\theta)$, where $r = 1$, $d = r - \sqrt{x^2 + y^2}$ and $\theta = \tan^{-1}\left(\frac{y}{x} \right)$.

The computational domain is a unit square. The grid convergence study is performed for the structured meshes ($N \times N$) 16×16, 32×32, 64×64, and 128×128. The spatial discretisation is obtained with quadratic triangular elements. Figures 2.6 (a-d) show the level sets for the 64×64 mesh: for the exact level sets solution, the disturbed level sets, the level sets after the geometric reinitialisation is applied to the disturbed level sets, and the level sets after the fast marching method is applied to the disturbed level sets. The error norms of the reinitialisation method are shown in Figure 2.7 to Figure 2.10. The mass error is approximately $\pm 2.5 \times 10^{-12}$ for the different meshes.

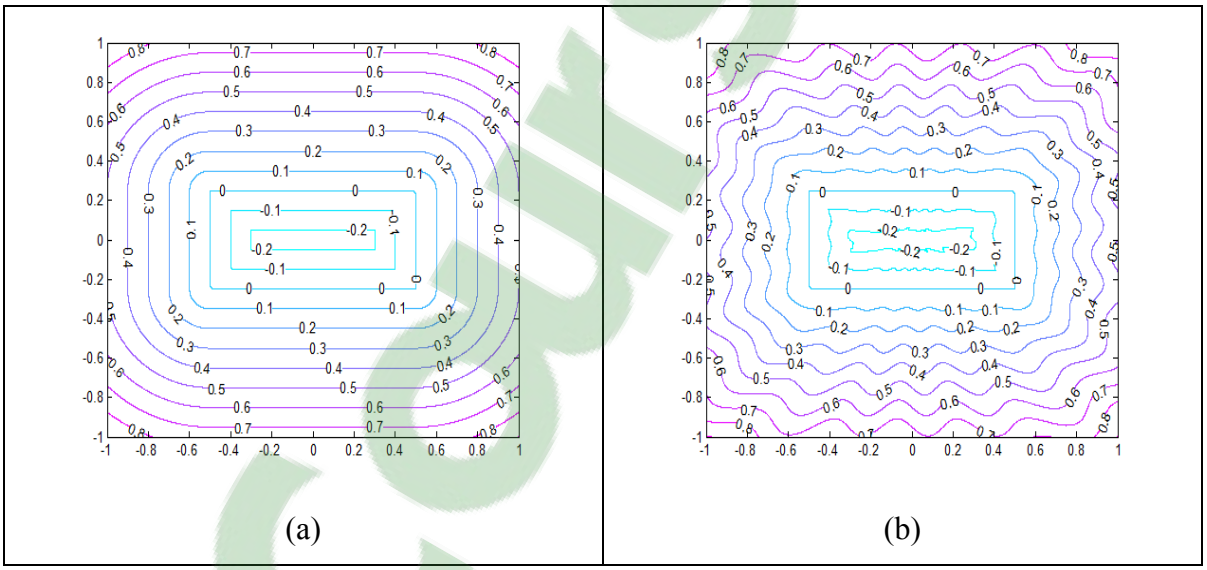
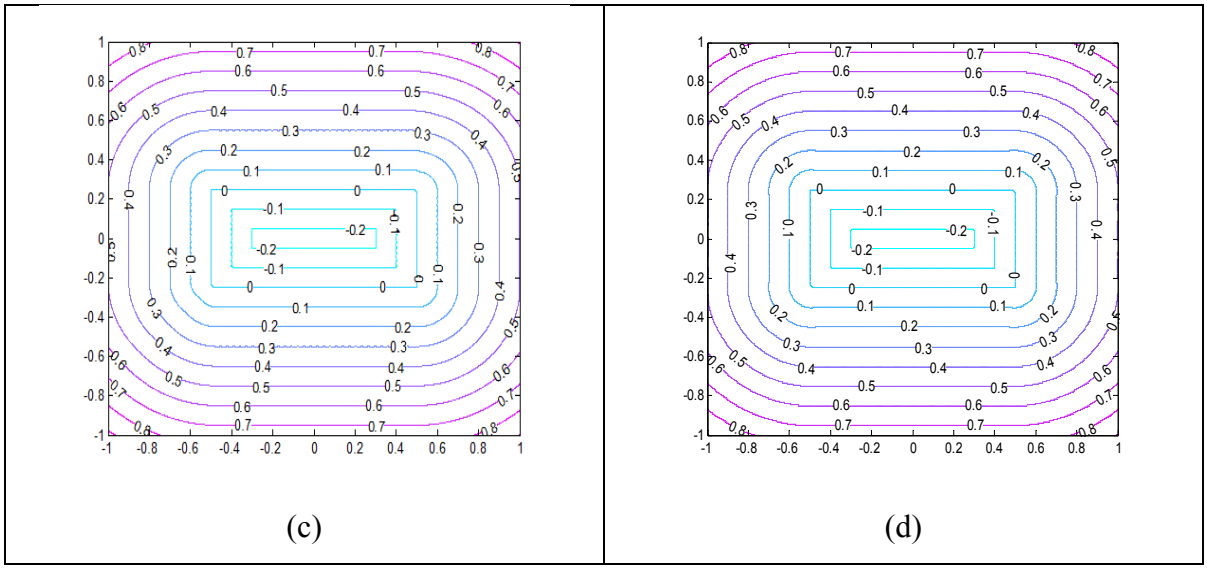


Figure 2.6: Exact level sets solution (a), disturbed level sets (b),
level sets after geometric reinitialisation (c),
and level sets after the FMM (d) for the 64×64 mesh

Figure 2.6 continuation



(c) (d)

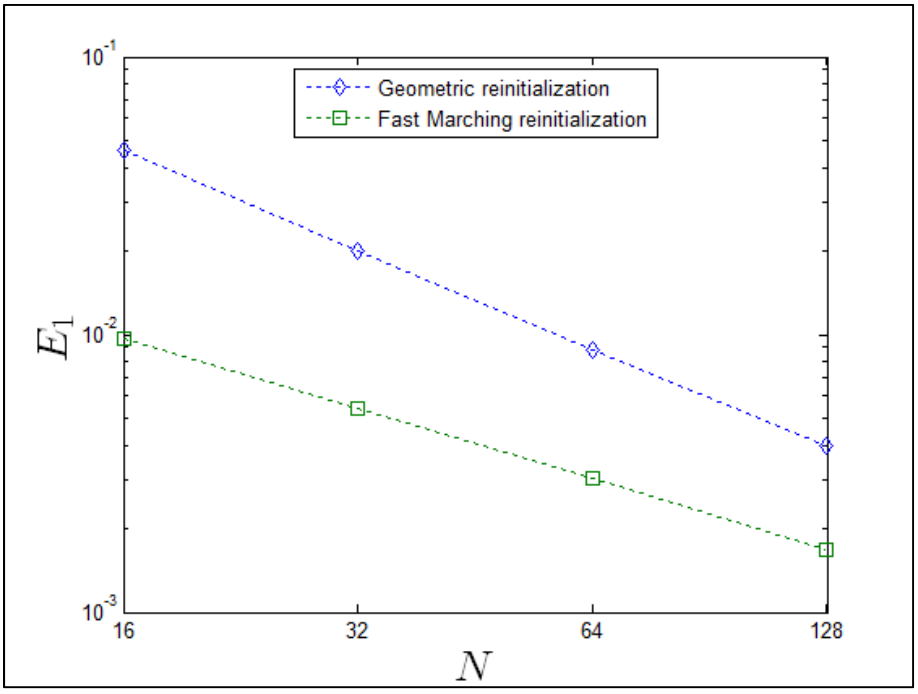


Figure 2.7: Error norm E_1 versus N

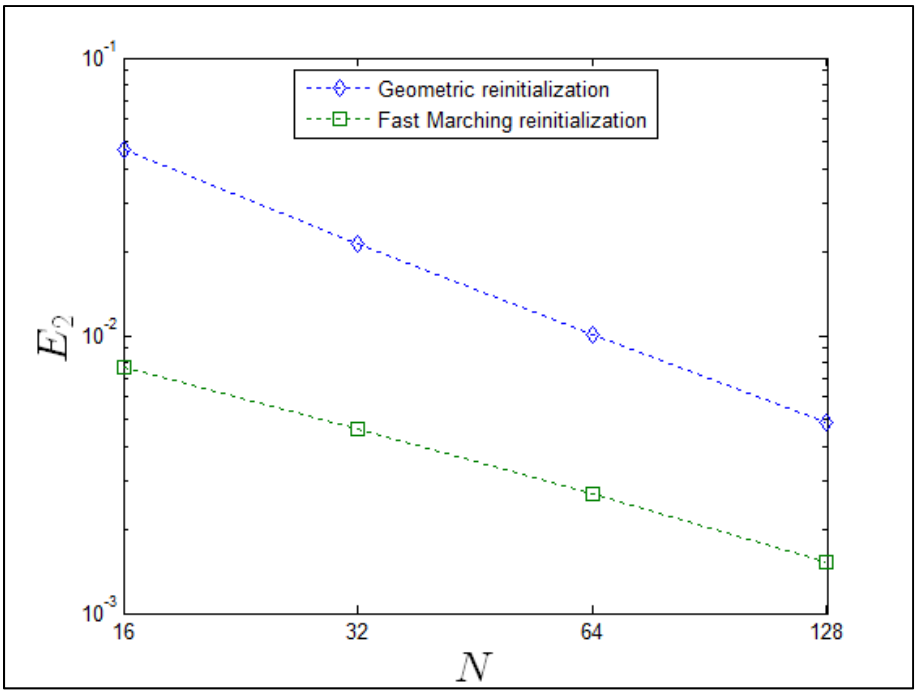


Figure 2.8: Error norm E_2 versus N

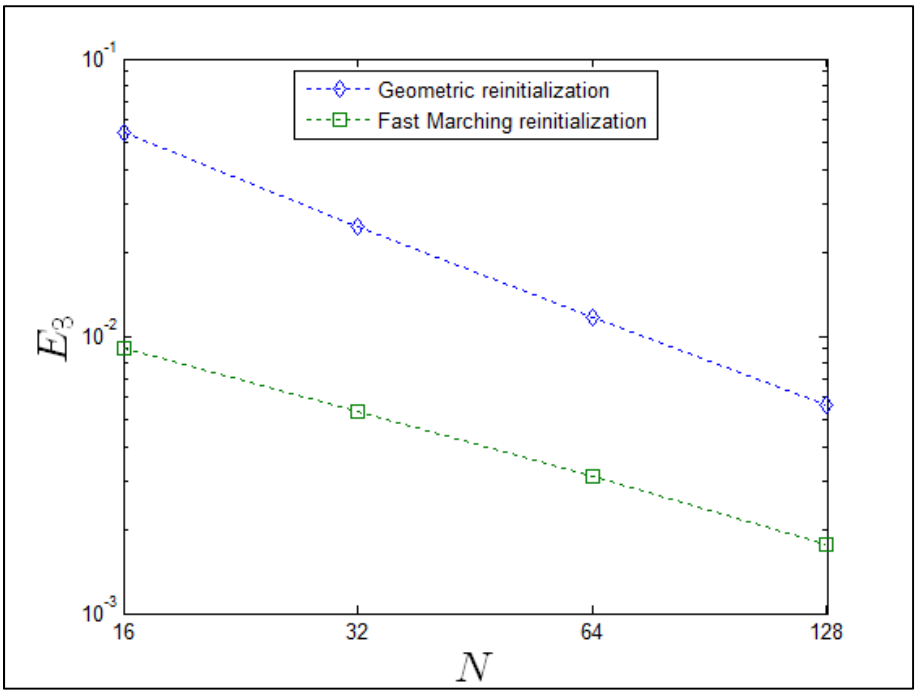


Figure 2.9: Error norm E_3 versus N

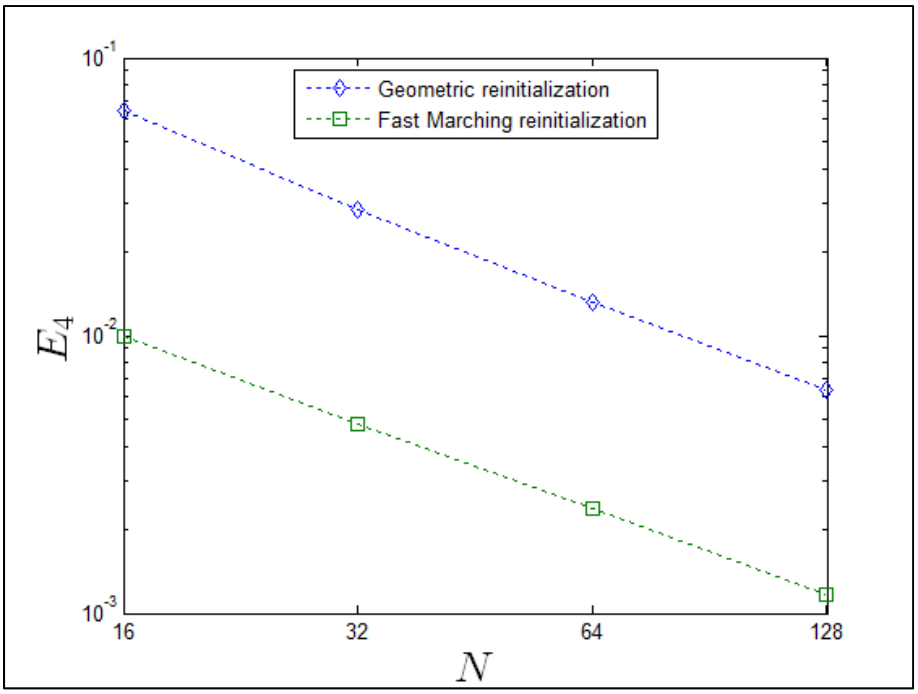


Figure 2.10: Error norm E_4 versus N

Figure 2.7 to Figure 2.10 show that the error norms converge linearly with the mesh size. The geometric reinitialisation thus has a good mass conservation property.

Figure 2.11 to Figure 2.14 show the error norms of the results of the 16×16 primary mesh with the effect of the refinement of the secondary mesh on which is projected the cubic interpolation of the nodal values of the level set before performing the fast marching method. The ratio between the dimension of the elements of the primary mesh sizes and that of the secondary mesh is represented by l .The results are even more accurate when the mesh on which we project the level set field is refined. However, the convergence of each error norm seems to be asymptotic.

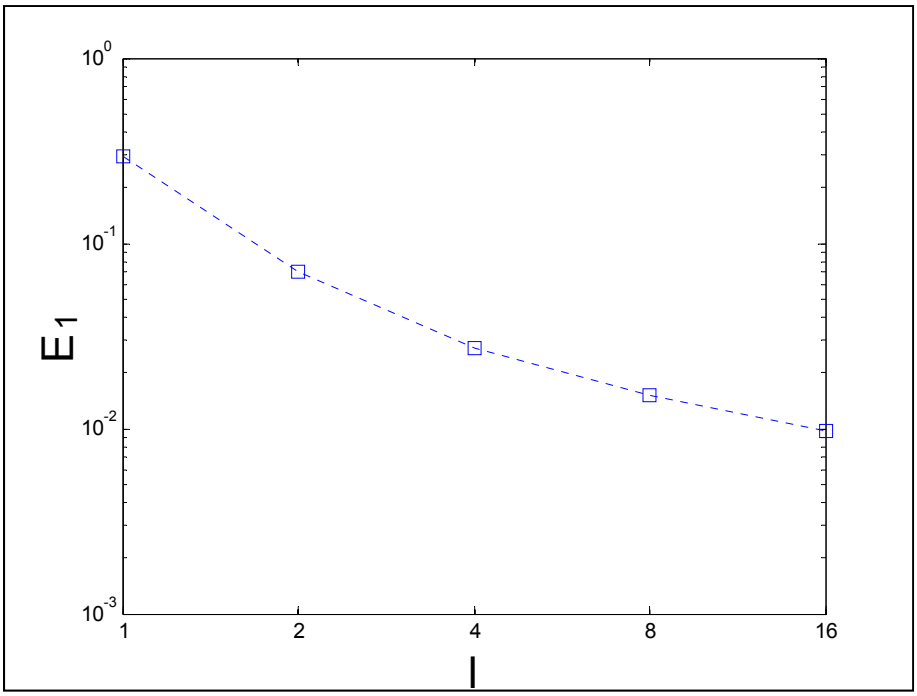


Figure 2.11: Error norm E_1 versus l

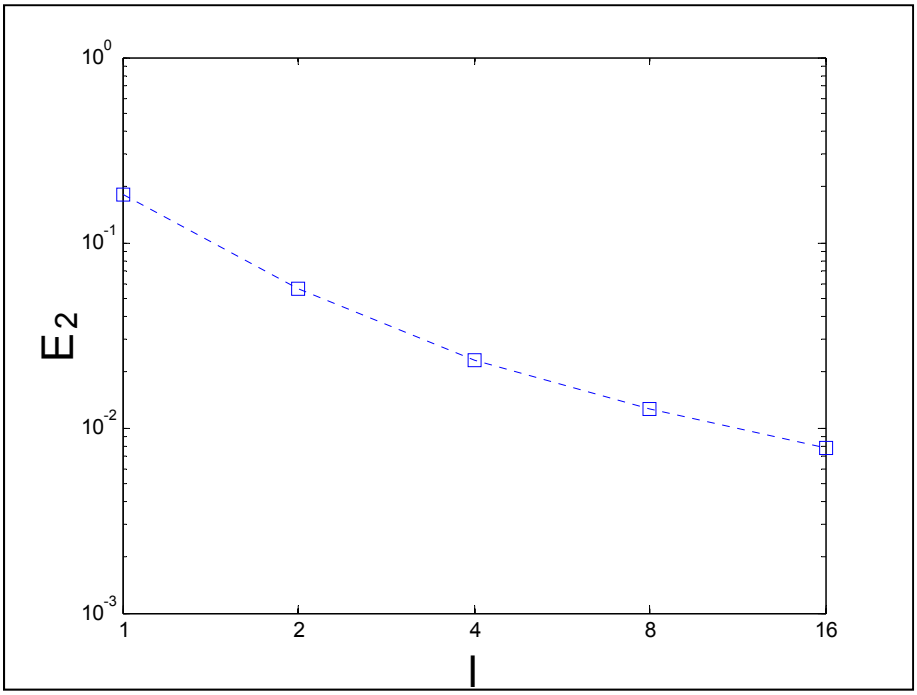


Figure 2.12: Error norm E_2 versus l

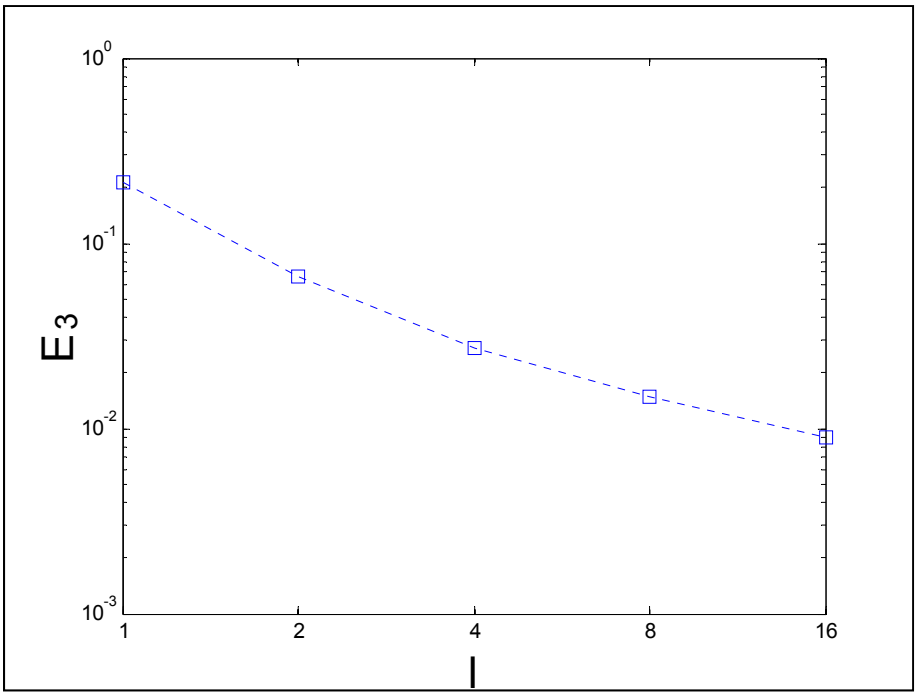


Figure 2.13: Error norm E_3 versus l

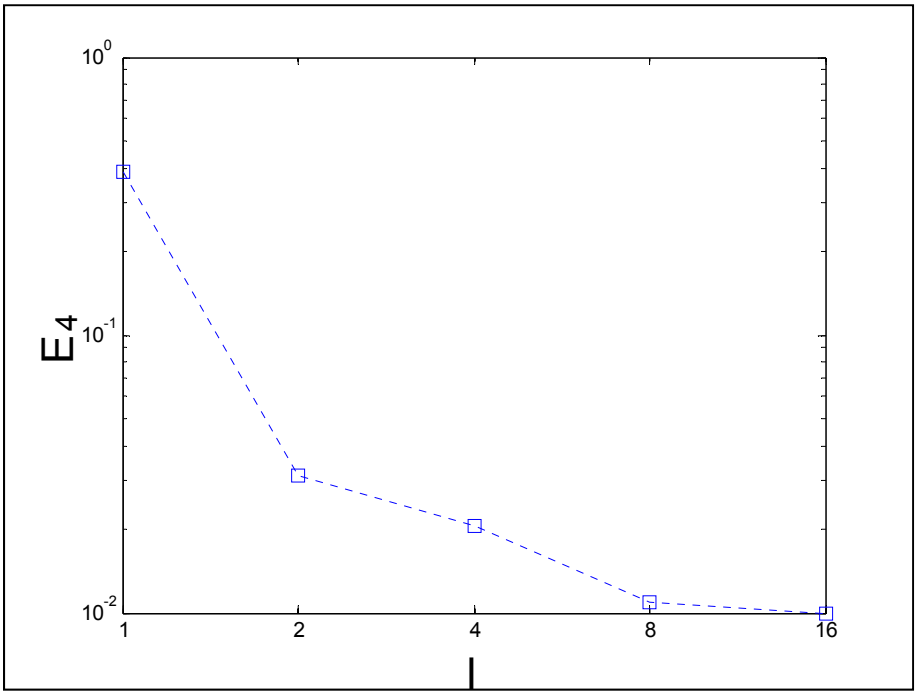


Figure 2.14: Error norm E_4 versus l

2.2.5 The reinitialisation of the level sets using the Eikonal method in the case of a vortex flow

We performed this test to evaluate the Eikonal method's sensitivity to a strong deforming interface separating two phases. This is a well-documented test case (Rider et Kothe, 1998) used to assess level set methods. The time-reversed vortex flow spins and stretches the fluids during the first half-period. The fluids then shrink and return to their original positions after one full period. A single vortex flow is defined by the stream function

$$\Psi_{single\ vortex} = \frac{1}{\pi} sin^2(\pi x) sin^2(\pi y) \tag{2.22}$$

The time-reversed vortex flow is defined by multiplying the single vortex flow stream function by $cos\left(\frac{\pi t}{T}\right)$, where T is the period, so that its stream function is

$$\Psi_{time-reversed\ vortex} = \frac{1}{\pi} sin^2(\pi x) sin^2(\pi y) cos\left(\frac{\pi t}{T}\right) \tag{2.23}$$

The relation between the velocity and the stream function is

$$u = -\frac{\partial \Psi}{\partial y} \text{ and } v = \frac{\partial \Psi}{\partial x} \tag{2.24}$$

where

$$u_{time-reversed\ vortex} = -2sin^2(\pi x) cos(\pi y) sin(\pi y) cos\left(\frac{\pi t}{T}\right) \tag{2.25}$$

and

$$v_{time-reversed\ vortex} = 2cos(\pi x) sin(\pi x) sin^2(\pi y) cos\left(\frac{\pi t}{T}\right) \tag{2.26}$$

A circle separates the unit square computational domain into two subdomains. The circle is centred at (0.5, 0.75), and its radius is 0.15. The computational time simulation is $T = 8s$. The flow reverses at $t = T/2$. The spatial discretisation is done with triangular elements of a 125×125 grid, resulting in 31,250 elements and 63K nodes. For the temporal discretisation,

the time step Δt is 0.008 s. The interface contours at the times $t = 0, T/4, T/2, 3T/4$ and T and the mass conservation with the Eikonal reinitialisation with the 125×125 grid are illustrated in Figure 2.15 and Figure 2.16, respectively.

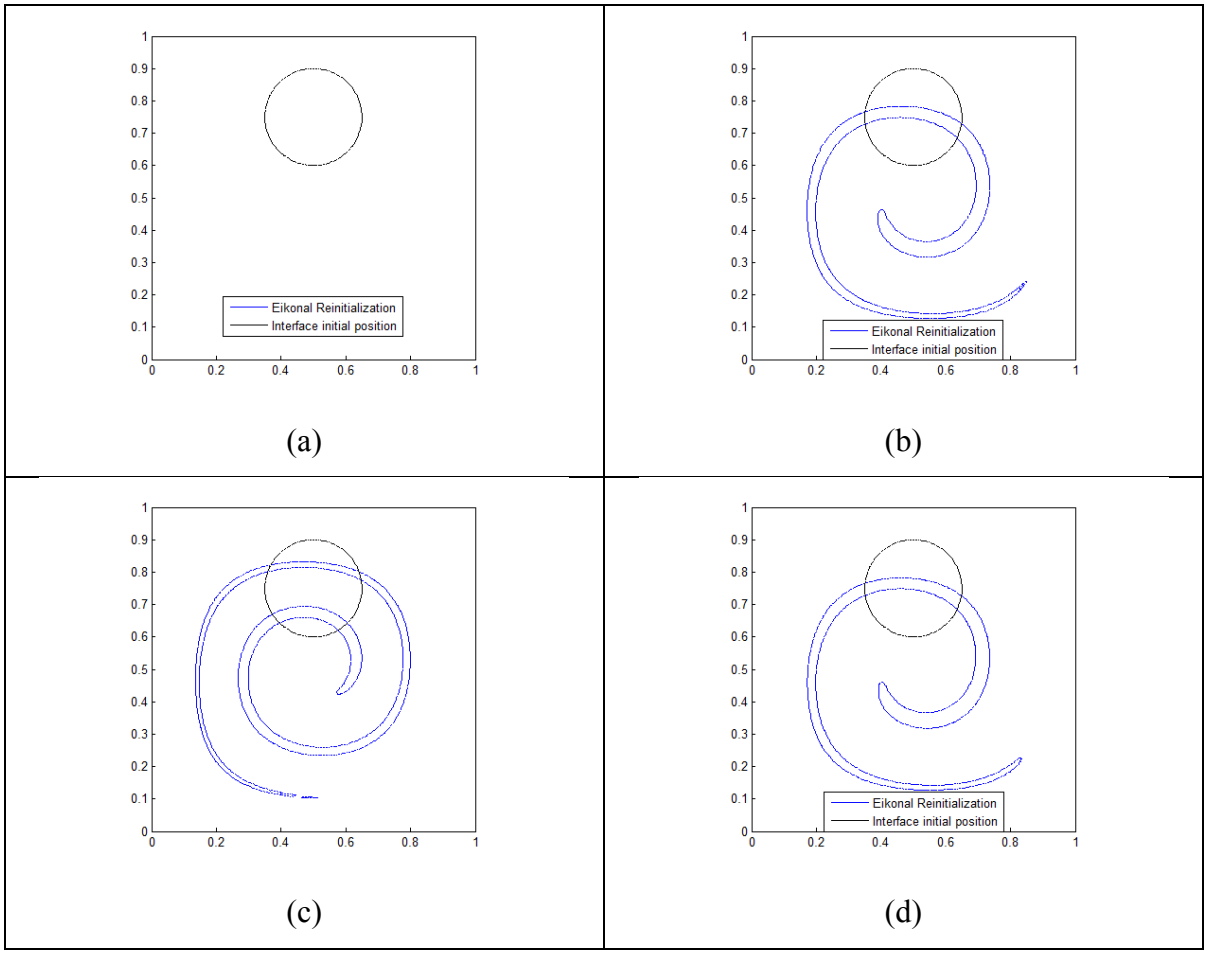
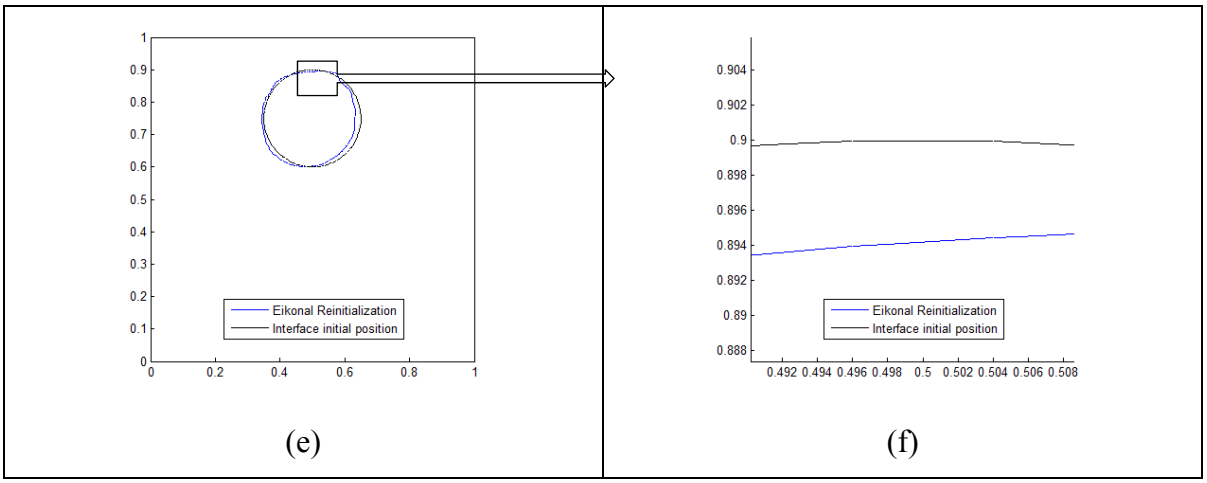


Figure 2.15: Time-reversed vortex flow. Interface positions at $t = 0(a), T/4(b), T/2(c), 3T/4(d)$ and $T(e)$ and with the 125×125 mesh using the Eikonal reinitialisation. Closer view of interface positions at T (f).

Figure 2.15 continuation



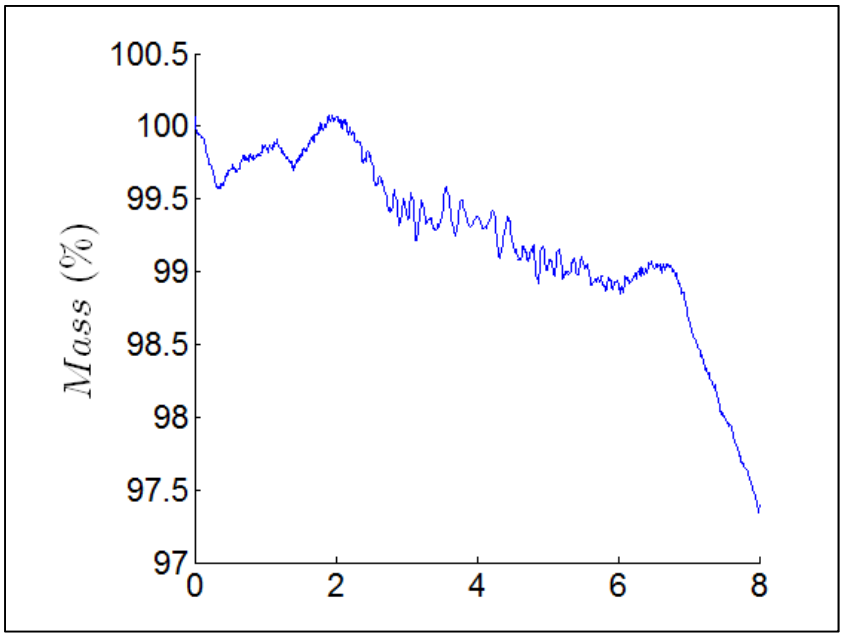


Figure 2.16: Time-reversed vortex flow. Percentage of disk area $Mass(\%)$ versus time using the Eikonal reinitialisation.

The Eikonal reinitialisation provides good results for a fine mesh. However, the mass loss increases quickly at the end of the time simulation between $13T/16$ and T.

CHAPTER 3

STABILIZED FINITE ELEMENT METHODS FOR SOLVING THE LEVEL SET EQUATION WITHOUT REINITIALISATION

Mamadou Kabirou Touré [a] and Azzeddine Soulaïmani [b]

[a, b] Department of Mechanical Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

This paper has been published (from page 37 to page 76)

in Computers & Mathematics with Applications, 2016

http://dx.doi.org/10.1016/j.camwa.2016.02.028

3.1 Introduction

New stabilized finite element methods are proposed for solving moving interface flow problems using the level set approach. The formulations enhance the interface resolution without the need to resort to the reinitialisation process. These are established by adding a perturbation term that depends on the local residual of the Eikonal equation to the SUPG variational formulation of the level set equation. These methods are numerically evaluated for well-known benchmark flow problems and compared with a modified variant of the penalty method of Li et al.(Li et al., 2005). The proposed stabilized finite element methods employing second-order time and space approximations are promising simple and accurate techniques for solving complex moving interface flows.

Moving interface problems are found in many engineering fields, such as fluid mechanics, solid mechanics and medical imaging. In fluid mechanics, free surface and multiphase flows are some examples of moving interface problems commonly encountered in nature and in industry. Robust and accurate numerical simulations of such flows are essential for economic and safety reasons. Significant progress has been achieved in numerical and experimental modeling of multiphase flows, such as flows of gas-liquid mixtures and flows of suspensions of particles or droplets (Prosperetti et Tryggvason, 2009).

Computational fluid dynamics (CFD) offers several numerical methods to solve multiphase flow problems. Numerical methods capable of simulating moving interface flows are usually classified in one of two approaches, Lagrangian or Eulerian methods. In the Lagrangian approach, the mesh is partially or totally mobile and is updated at each step. Lagrangian methods can be accurate but difficult to implement in 3D, especially in the case of complex interface deformations. Eulerian methods include the Front-tracking and the Front-capturing methods. Front-tracking methods use markers and can be divided into two groups: Volume tracking and Surface tracking. In Volume tracking methods, such as the Marker and cells method (MAC), markers are distributed in all parts of the domain, whereas with Surface tracking methods, the markers are spread only on the interface. During a simulation, marker points may gather in certain areas, making a redistribution of the markers in the domain or on the surface necessary to preserve accuracy. The MAC Volume tracking method with unconnected distributed marker points was applied on both a totally and a partially broken dam (Harlow et Welch, 1965). Surface-tracking methods are in general more accurate than Volume tracking methods in terms of interface position. Some Surface tracking methods were developed by Glimm et al. (Glimm, 1982; Glimm et McBryan, 1985) to track shocks in compressible flows, and more recently, Unverdi and Tryggvason developed ways to solve the Navier-Stokes equations when applied to bubbles with the surface tension effect (Unverdi et Tryggvason, 1992).

The Front-capturing methods are based on the evolution of a scalar function that is related to the interface position. Front-capturing methods, such as the Volume of fluid method (VOF) or the level set method with their variants, are easy to implement in coalescence or split interface problems. In the VOF method, the scalar function used for each cell is defined as the volume fraction of a phase. The VOF method is known to conserve the mass of the phases, but the main difficulty lies in the reconstruction of the interface approximated by the volume fraction values. Some of the first VOF methods were developed by Noh and Woodward (Noh et Woodward, 1976), also known as the simple line interface calculation (SLIC), and by Hirt and Nichols (Hirt et Nichols, 1981), but these two methods have the drawback of creating broken pieces of the interface. In (Youngs, 1982), the approximation of the interface uses straight lines whose orientation is determined for each cell by the volume fraction of the neighbouring cells.

Osher and Sethian's level set method (Osher et Sethian, 1988) was originally developed for the simulation of a phase change problem governed by a diffusion equation; numerous applications can be found, such as in image analysis (Sethian, 1998). The level set approach seeks to define a scalar function whose zero contour level is the moving interface. The level set function is in general defined as the signed distance to the interface. Solving a transport equation then allows the motion of the interface driven by a velocity field to be followed. The level set method handles complex topological changes; the calculation of geometrical characteristics such as the unit normal vector or the curvatures of the interface are particularly simple using the gradient of the level set function. However, the numerical errors that occur when solving the transport equation may lead to a loss of interface smoothness and a loss of mass, which are more or less important depending on both the total time simulation and the presence of shear and stretching zones in the flow field that expand or tighten the contours.

In the quest to enhance the numerical properties of the level set approach, a large number of publications have been produced in the last two decades. Various reinitialisation (or redistancing) algorithms have been proposed to smooth or to resharpen the contours so that the level set function remains a distance function to the interface (Adalsteinsson et Sethian, 1999; Chunming et al., 2010; Gross et Reusken, 2011; Hysing et Turek, 2005; Olsson et Kreiss, 2005; Olsson, Kreiss et Zahedi, 2007; Osher et Fedkiw, 2003; Qian, Zhang et Zhao, 2007; Sethian, 1996; Sethian, 1998; So, Hu et Adams, 2011; Sussman, Smereka et Osher, 1994; Zhao, 2005). The major difficulties in the reinitialisation process are the choice of the delay between two renitialisations and the method used for the reinitialisation, which may shift the interface position (Gomes et Faugeras, 2000). If the initial level set function is significantly different from a signed distance function, then the reinitialisation schemes may not be able to reinitialize the function to a signed distance function. Another issue with the level set approach is that mass conservation is not guaranteed at the discrete level. In recent years, several studies have been devoted to developing numerical methods and algorithms to enhance the mass conservation properties of the level set approach (Ausas, Dari et Buscaglia, 2011; Desjardins, Moureau et Pitsch, 2008; Di Pietro, Lo Forte et Parolini, 2006; Doyeux et al., 2013; Kuzmin, 2014; Laadhari, Saramito et Misbah, 2010; Le Chenadec et Pitsch, 2013; Olsson et Kreiss, 2005; Olsson, Kreiss et Zahedi, 2007; Owkes et Desjardins, 2013; Smolianski, 2001; Sussman,

2003; Sussman et Puckett, 2000; van der Pijl et al., 2005; Wang, Simakhina et Sussman, 2012). These can be classified into two groups: in the first, the level set method is combined with conservative discretization of the VOF method. In the second group, the level set evolution is constrained to conserve the mass of one phase globally or locally.

A variational level set formulation is described in (Li et al., 2005; Li et al., 2006), in which a level set transport equation is modified to include a nonlinear term that penalizes the deviation of the level set function from the signed distance. It is claimed that the penalty term not only eliminates the need for reinitialisation, it allows the use of a simpler and more efficient numerical scheme in the implementation than those used for conventional level set formulations. This method has been tested on some image segmentation problems, and we are interested in testing and validating this method with the goal of applying it to free surface flow and two-phase flow problems. However, the original method involves a penalty constant that needs to be tuned in accordance with the mesh size and the velocity field. Indeed, a small value of this coefficient may lead to irregularities in the level set contours over time. A large value would diffuse the contours and modify the zero level set position.

The first objective of this chapter is to propose a more general formulation for the penalty coefficient to limit the tuning process. The second objective is to propose stabilized variational formulations to solve the level set equation without reinitialisation. These stabilized methods are numerically evaluated and compared with the modified penalty method (Li et al., 2005) on the classical tests of a time-reversed vortex (Rider et Kothe, 1998), a rigid-body rotation of Zalesak's disk (Rider et Kothe, 1998; Zalesak, 1979) and the dam break problem (Hu et Sueyoshi, 2010; Martin et Moyce, 1952). These methods are numerically assessed in terms of accuracy, stability and mass conservation.

The chapter is organized as follows: In section 3.2, we briefly describe the variational penalty method of Li et al. (Chunming et al., 2010; Li et al., 2005). The stabilized variational methods are introduced in section 3.3, and the numerical tests are presented in section 3.4, followed by our conclusions.

3.2 The variational method by energy penalization

This method was introduced in the context of image segmentation (Li et al., 2005) to avoid the use of the reinitialisation step . A functional $\mathrm{P}(\phi)$ is assumed:

$$\mathrm{P}(\phi) = \int_{\Omega} \frac{1}{2}\left(|\nabla\phi| - 1\right)^2 d\Omega \tag{3.1}$$

and a total energy is defined by $\mathrm{E}(\phi)$:

$$\mathrm{E}(\phi) = \lambda_1 \mathrm{P}(\phi) + \mathrm{E}_m(\phi) \tag{3.2}$$

where λ_1 is a constant parameter that controls the deviation between the level set function ϕ and the signed distance function d_{signed}, and $\mathrm{E}_m(\phi)$ is a certain energy that would drive the motion of the zero level curve of ϕ .The objective is to minimize the total energy, which leads to solving the penalized level set equation:

$$\frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi - \nabla.\left(\lambda_1 K_1 \nabla\phi\right) = 0 \tag{3.3}$$

where

$$K_1 = 1 - \frac{1}{|\nabla\phi|} \tag{3.4}$$

Li et al. (Li et al., 2005) stated that equation (3.3) is a transport-diffusion equation when $K_1 > 0$, i.e. making ϕ smoother and thereby reducing the gradient $|\nabla\phi|$, and that it is an anti-diffusion transport equation when $K_1 < 0$ and therefore the gradient is increased. This method has been employed in imagery processing but has not been evaluated for CFD applications, where equation (3.3) is usually solved using the finite difference method.

Remark 1: In (Li et al., 2005), the penalty coefficient λ_1 is a constant to be determined to ensure stability for the explicit scheme used. A too-high value of this coefficient would significantly modify the interface dynamics. Note that this penalty coefficient has the dimension of

diffusivity. To take into account the effects of element size and flow speed, we propose a variable coefficient using the following formula, which is widely used in the context of stabilized methods:

$$\lambda_1 = \gamma(h^e)\frac{|\mathbf{u}|h^e}{2} \tag{3.5}$$

where h^e is the size of the element (defined as the minimal distance between the nodes), $|\mathbf{u}|$ is the velocity norm and $\gamma(h^e)$ is an empirical parameter. We found from extensive experiments that the optimal value of $\gamma(h^e)$ is proportional to the mesh size, so we adopt the simple relation: $\gamma(h^e) = \beta_1 h^e$ where β_1 is a constant close to unity; and therefore:

$$\lambda_1 = \beta_1 \frac{|\mathbf{u}|\left(h^e\right)^2}{2} \tag{3.6}$$

Remark 2: Although there is a penalty term for the constraint $|\nabla\phi|-1=0$, this Eikonal equation is not exactly satisfied because of the numerical discretizations.

3.3 Stabilized variational methods

3.3.1 A mixed variational method

One can establish a variational formulation where the level set function is constrained to conserve the distance property. Thus, the transport equation is solved along with the Eikonal constraint:

$$|\nabla\phi|-1=0 \quad \text{or} \quad |\nabla\phi|^2-1=0 \tag{3.7}$$

which are nonlinear equations. First a linearization of equation (3.7) is used. Let n, k and Δt denote the time, the iteration indices, and the time step, respectively. At each time instant the following equation is solved iteratively

$$\left|\nabla \phi^{n+1,k+1}\right| - 1 = 0 \tag{3.8}$$

Using Newton's algorithm, a value for $\Delta\phi$ is sought such that $\phi^{n+1,k+1} = \phi^{n+1,k} + \Delta\phi$ is the solution for the linear constraint:

$$\frac{\nabla \phi^{n+1,k} \cdot \nabla \Delta\phi}{\left|\nabla \phi^{n+1,k}\right|} = -\left(\left|\nabla \phi^{n+1,k}\right| - 1\right) \tag{3.9}$$

A mixed variational formulation can now be established by introducing a Lagrange multiplier $p^{n+1,k+1}$ corresponding to the linear constraint (3.9). Let ψ and q be two weight functions belonging to suitable spaces, and find $\left(\phi^{n+1,k+1}, p^{n+1,k+1}\right)$ such that:

$$\int_{\Omega} \left[\psi \left(\frac{\partial \phi^{n+1,k+1}}{\partial t} + \mathbf{u} \cdot \nabla \phi^{n+1,k+1} \right) + \left(\frac{\nabla \phi^{n+1,k} \cdot \nabla \psi}{\left|\nabla \phi^{n+1,k}\right|} \right) p^{n+1,k+1} \right] d\Omega = 0 \tag{3.10}$$

and

$$\int_{\Omega} q \left[\frac{\nabla \phi^{n+1,k} \cdot \nabla \Delta\phi}{\left|\nabla \phi^{n+1,k}\right|} + \left(\left|\nabla \phi^{n+1,k}\right| - 1\right) \right] d\Omega = 0 \tag{3.11}$$

Given $\phi^{n+1,k}$, let $\mathbf{w} = \dfrac{\nabla \phi^{n+1,k}}{\left|\nabla \phi^{n+1,k}\right|}$, $\phi^{*} = \phi^{n+1,k+1}$ and $p^{*} = p^{n+1,k+1}$. Then, at each iteration, a linear problem is solved:

$$\int_{\Omega} \left[\psi \left(\frac{\partial \phi^{*}}{\partial t} + \mathbf{u} \cdot \nabla \phi^{*} \right) + \left(\mathbf{w} \cdot \nabla \psi \right) p^{*} \right] d\Omega = 0 \tag{3.12}$$

and

$$\int_{\Omega} q \left(\mathbf{w} \cdot \nabla \phi^{*} \right) d\Omega = \int_{\Omega} q \left(\mathbf{w} \cdot \nabla \phi^{n+1,k} - \left(\left|\nabla \phi^{n+1,k}\right| - 1\right) \right) d\Omega \tag{3.13}$$

which corresponds to the Euler-Lagrange equations (Liu et al., 2011):

$$\begin{cases} \dfrac{\partial \phi^*}{\partial t} + \mathbf{u}\cdot\nabla\phi^* - \nabla\cdot\left(\mathbf{w}\, p^*\right) = 0 \\ \mathbf{w}\cdot\nabla\phi^* = g \end{cases} \tag{3.14}$$

with $g = \mathbf{w}\cdot\nabla\phi^{n+1,k} - \left(\left|\nabla\phi^{n+1,k}\right| - 1\right)$ and using homogenous boundary conditions on p^*. Clearly, the level set equation is modified by the introduction of the differential operator $-\nabla\cdot\left(\mathbf{w}\, p^*\right)$. Since the Lagrange multiplier is not necessarily zero, this term modifies the original transport equation by adding a diffusive or an anti-diffusive effect, depending on the sign of p^*. If the magnitude of p^* is relatively large, this may lead to an unphysical solution. Also, all equations in (3.14) are of hyperbolic type; accurate discretization of this type of partial differential equations is rather difficult. Indeed, this mixed variational formulation has been implemented in a finite element code using a quadratic interpolation for (ψ, ϕ^*) and a continuous linear (or discontinuous P_0) interpolation for (q, p^*) over triangular elements to satisfy the inf-sup condition. We performed extensive numerical tests and found that the results are not always satisfactory in terms of accuracy. Therefore, we conclude that solving the level set equations along with the Eikonal constraint in a strongly coupled manner is not a satisfactory method.

3.3.2 A new stabilized variational method

In order to have more control over the magnitude of p^*, we propose to use the penalty procedure at the discrete level (to simplify the notation, we omit the superscript *). Therefore, equation (3.13) is modified into:

$$\int_\Omega q\left(\left|\nabla\phi\right| - 1\right) d\Omega - \frac{1}{\lambda_2}\int_\Omega q\, p\, d\Omega = 0 \tag{3.15}$$

Using a constant interpolation per element Ω^e for (q, p) (i.e. the P0 interpolation), p^e is obtained:

$$p^e = \frac{\lambda_2}{\left|\Omega^e\right|} \int_{\Omega^e} q\left(\left|\nabla\phi\right|-1\right) d\Omega \tag{3.16}$$

where λ_2 is a small parameter defined by:

$$\lambda_2 = \beta_2 \frac{\left|\mathbf{u}\right|\left(h^e\right)^2}{2} \tag{3.17}$$

We can also rewrite $p^e = \lambda_2 \; K_2$, where $K_2 = -\frac{1}{\left|\Omega^e\right|} \int_{\Omega^e} q\left(\mathbf{w}.\nabla\phi \; - g\right) d\Omega$

Then, p^e of equation (3.16) will be inserted back into (3.12). This will be done for few iterations, and at convergence $K_2 \approx \frac{1}{\left|\Omega^e\right|} \int_{\Omega^e} \left(\left|\nabla\phi\right|-1\right) d\Omega$, which is the *P0* projection of the residual. In other words, K_2 is the element average of the Eikonal equation residual, whereas K_1 is related to the local residual $\left|\nabla\phi\right|-1$.

It is also simple to implement the *P1* or *P2* projection of $\left(\left|\nabla\phi\right|-1\right)$ using the following relations:

$$p^e = \sum_i N_i \; p_i \quad \text{and} \quad \sum_j M_{ij} p_j = \lambda_2 \int_{\Omega^e} N_i \left(\left|\nabla\phi\right|-1\right) d\Omega \tag{3.18}$$

with $M_{ij} = \int_{\Omega^e} N_i \, N_j \; d\Omega$ and N_i representing the linear or the quadratic shape functions (for *P1* and *P2* projection, respectively).

In summary, the proposed method modifies the level set equation into:

$$\frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi - \nabla\cdot\left(p\,\frac{\nabla\phi}{\left|\nabla\phi\right|}\right) = 0 \tag{3.19}$$

where p is the projection of the residual $\left(\left|\nabla\phi\right|-1\right)$ which is computed by solving (3.18).

A weak form corresponding to equation (3.3) or (3.19) using the stabilized Streamline-Upwind-Petrov-Galerkin (SUPG) method is:

$$\int_{\Omega}\left[(\psi+\tau\,\mathbf{u}\cdot\nabla\psi)\left(\frac{\partial\phi}{\partial t}+\mathbf{u}\cdot\nabla\phi\right)+p\,\frac{\nabla\phi}{|\nabla\phi|}\cdot\nabla\psi\right]d\Omega=0 \tag{3.20}$$

with $\tau=\alpha\dfrac{h^e}{2|\mathbf{u}|}$ where $0\le\alpha\le1$, and where ψ is a weighting function. Since we are looking for numerical solutions of (3.20), the trial and weighting functions belong to standard finite element spaces defined by continuous polynomials over triangles or quads in two-dimensions (for $d=2$).

For $\alpha=\beta_1=\beta_2=0$, equation (3.20) is the standard Galerkin formulation. For $\alpha=1$ and $\beta_1=\beta_2=0$, we find the stabilized SUPG formulation for the original level set equation. For $\alpha=0,\ \beta_1\neq0$ and $\beta_2=0$ we recover the energy penalization method of (Li et al., 2005). Finally, for $\alpha\neq0$, $\beta_1=0$ and $\beta_2\neq0$ we find the proposed stabilized formulation.

To solve (2.6) or (3.20), the initial solution is $\phi(x,0)=\phi_0$, in which ϕ_0 is an approximate signed distance to the interface $\Gamma(0)$ and should be such that $\{x\in\mathbb{R}^d:\phi_o(x)=0\}=\Gamma(0)$. To make the problem of the linear hyperbolic level set equation (2.6) well-posed, a boundary condition is needed at the inflow part of the domain $\partial\Omega_{in}=\{x\in\partial\Omega:\ \mathbf{u}\cdot\mathbf{n}<0\}$. As noted in (Gross et Reusken, 2011), there are no natural physics-based boundary conditions for ϕ at the inflow boundary; we are only interested in values of ϕ close to the interface. Note that for a closed domain with solid boundaries $\mathbf{u}\cdot\mathbf{n}=0$ along $\partial\Omega$, ϕ is defined up to a constant. An implicit condition is provided, however, by the variational formulation (3.20): $\int_{\Omega}\dfrac{\partial\phi}{\partial t}\,d\Omega=0$, since the function $\psi=1$ is in the weighting finite element space. Therefore, the constant is implicitly defined by $\int_{\Omega}\phi(x,t)\,d\Omega=\int_{\Omega}\phi(x,0)\,d\Omega$. This condition is often replaced by the more physical one which states the conservation of each phase volume. Let $V_1(\phi,t)$ be the volume of the subdomain Ω_1:

$$V_1(\phi,t) = \int_{\Omega_1} d\Omega \tag{3.21}$$

and then look for a constant δ such that: $\phi^{corrected} = \phi + \delta$, so that

$$V_1(\phi+\delta,t) = V_1(\phi,0) \tag{3.22}$$

Equation (3.22) is nonlinear with respect to δ, requiring an iterative numerical method. At convergence, the corrected level set $\phi^{corrected}$ is:

$$\phi^{corrected} = \phi + \delta \tag{3.23}$$

3.3.3 A Galerkin-Least-Squares (GLS) variational method

The last left-hand-side term of (3.20) can be interpreted as a perturbation of the original level set equation that depends on the Eikonal equation residual, and *not* a perturbation of the level set equation residual as traditionally used in the classical SUPG or GLS methods. In fact, the variational method by energy penalization can be set in the framework of the GLS variational formulation. The variational formulation (3.20) can be written as:

$$\int_{\Omega} \left[(\psi + \tau\, \mathbf{u}\cdot\nabla\psi)\left(\frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi\right) + \lambda_3\, K_1(\nabla\phi\cdot\nabla\psi) \right] d\Omega = 0 \tag{3.24}$$

where the stabilization parameter λ_3 is obtained by an analysis argument used in SUPG/GLS formulations. The term $\int_{\Omega} \lambda_3\, K_1(\nabla\phi\cdot\nabla\psi)\, d\Omega$ is in fact the Gâteaux derivative of the functional $\frac{1}{2}\int_{\Omega} \lambda_3 \left(|\nabla\phi| - 1\right)^2 d\Omega$ in the direction ψ (Duan, Ma et Zhang, 2008a; Duan, Ma et Zhang, 2008b). Therefore, the only difference between the proposed GLS and the variational method by energy penalization resides in the definition of the stabilization parameters.

To define the parameter λ_3 we perform the following derivations. Recall the unit vector $\mathbf{w} = \dfrac{\nabla\phi}{|\nabla\phi|}$ and $K_1 = \dfrac{|\nabla\phi|-1}{|\nabla\phi|}$, then

$$\int_\Omega \lambda_3 K_1 (\nabla\phi\cdot\nabla\psi)\, d\Omega = \int_\Omega \lambda_3 (\mathbf{w}\cdot\nabla\psi)\left(|\nabla\phi|-1\right)\, d\Omega \tag{3.25}$$

Suppose given the solutions ϕ^n at time step n and $\phi^{n+1,k}$ at time step $n+1$ and iteration k. In equation (3.25) we will use the following approximation: $\mathbf{w} \approx \mathbf{w}^k = \dfrac{\nabla\phi^{n+1,k}}{|\nabla\phi^{n+1,k}|}$. We look for $\phi^{n+1,k+1} = \phi^{n+1,k} + \Delta\phi$ such that:

$$\int_\Omega \lambda_3 (\mathbf{w}\cdot\nabla\psi)\left(|\nabla\phi^{n+1,k+1}|-1\right)\, d\Omega \approx \int_\Omega \lambda_3 \left(\mathbf{w}^k\cdot\nabla\psi\right)\left(\mathbf{w}^k\cdot\nabla\Delta\phi + r^k\right)\, d\Omega \tag{3.26}$$

with $r^k = |\nabla\phi^{n+1,k}|-1$.

The last integral clearly shows the diffusion operator in the direction $\mathbf{w}^k$; this suggests the design of the parameter λ_3 as $\lambda_3 \approx \gamma \dfrac{h^e}{2|\mathbf{w}^k|}$. Since the diffusion effect is desired to be very small around the interface, the parameter γ is designed as $\gamma = \beta_3 h^e \xi$ (with $\xi = 0$ for elements crossed by the interface and $\xi = 1$ elsewhere; β_3 has a dimension of 1/time). The coefficient ξ is introduced to minimize the numerical perturbations of the zero-level set. Finally, we will use the definition:

$$\lambda_3 = \beta_3 \frac{(h^e)^2}{2|\mathbf{w}^k|}\xi \tag{3.27}$$

Remark 3: To spatially discretize equations (3.20) and (3.24), triangular or quad elements are used with quadratic finite element approximations. The Crank-Nicolson semi-implicit scheme is used for time discretization (see Appendix). In the following numerical tests, the resulting nonlinear algebraic systems are solved using a quasi-Newton algorithm along with a direct algebraic solver.

3.4 Numerical tests

The following numerical tests were carried out to compare the accuracy of the proposed methods. First, in section 3.4.1, a numerical test is performed to evaluate the accuracy of the level set geometric reinitialisation described in section 2.2.1 using disturbed level sets of a rectangular interface. Then, in the subsequent two section, a velocity is imposed during the tests, utilizing a time-reversed vortex (section 3.4.2) and the rigid body rotation of Zalesak's disk (Rider et Kothe, 1998) (section 3.4.3). In section (3.4.4) an example is presented where the velocity is computed by solving a flow problem. We verified the convergence of the methods versus the mesh size. In order to analyze the solution accuracy, the following norms are defined:

$$E_1 = \int_\Omega \left|\phi(\mathbf{x},0) - \phi(\mathbf{x},T)\right| \, d\Omega \ ,$$

$$E_2 = \sqrt{\int_\Omega \left(\phi(\mathbf{x},0) - \phi(\mathbf{x},T)\right)^2 d\Omega} \ ,$$

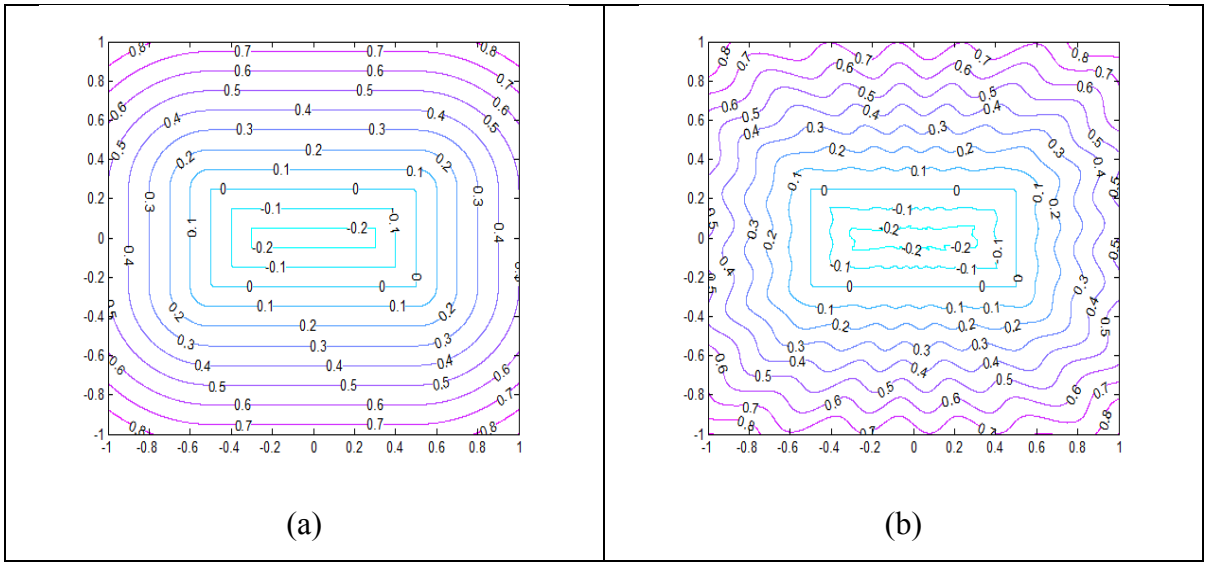
$$\text{and } E_3 = \frac{\sqrt{\int_\Omega \left(\phi(\mathbf{x},0) - \phi(\mathbf{x},T)\right)^2 d\Omega}}{\sqrt{\int_\Omega \phi^2(\mathbf{x},0)\, d\Omega}}$$

where $\phi(\mathbf{x},0)$ is the finite element interpolation of the exact solution at the initial time. Furthermore, to assess the mass (or area) conservation of a phase, the following percentage is defined as $Mass(\%) = 100\left(\frac{\text{area}(t)}{\text{area}(t=0)}\right)$.

3.4.1 Reinitialisation of disturbed level sets

The interface Γ is defined as a rectangle. The nodal level set values are then obtained by computing the signed distance to the rectangular interface. The disturbed level set function is found by adding to the exact level set function a perturbing term: $\frac{r}{20}\sin\left(\frac{\pi d}{2r}\right)\sin(20\theta)$, where $r=1$, $d=r-\sqrt{x^2+y^2}$ and $\theta=\tan^{-1}\left(\frac{y}{x}\right)$.

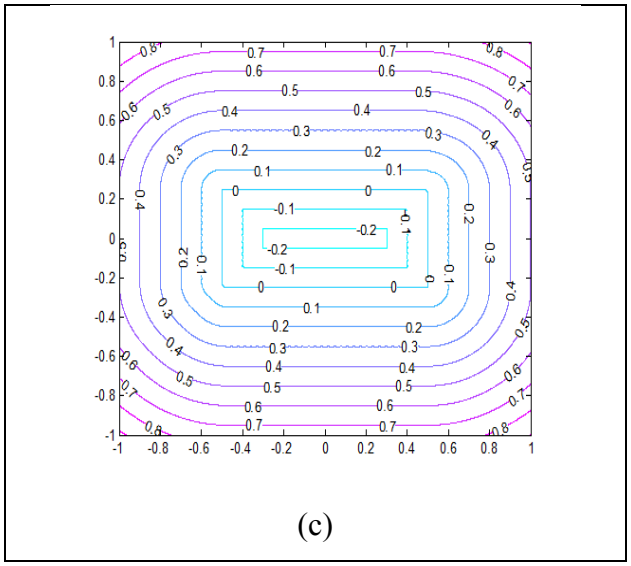
The computational domain is a unit square. The grid convergence study is performed for the structured meshes ($N\times N$): 16×16, 32×32, 64×64, and 128×128. The spatial discretization is obtained with quadratic triangular elements. Figures 3.1 (a-c) show the level sets for the 64×64 mesh: for the exact level sets solution, the disturbed level sets, and the level sets after the geometric reinitialisation is applied to the disturbed level sets. The error norms of the reinitialisation method are show in Figures 3.2 to 3.4. The mass error is around $\pm 2.5\times10^{-12}$ for the different meshes.



(a) (b)

Figure 3.1: Exact level sets solution (a), disturbed level sets (b), level sets after the geometric reinitialisation (c)

Figure 3.1 continuation



(c)

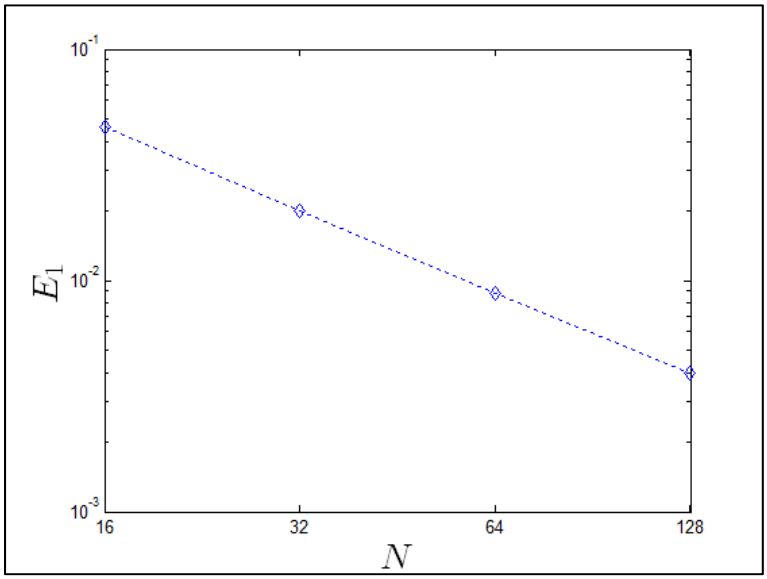


Figure 3.2: Error norm E_1 versus N

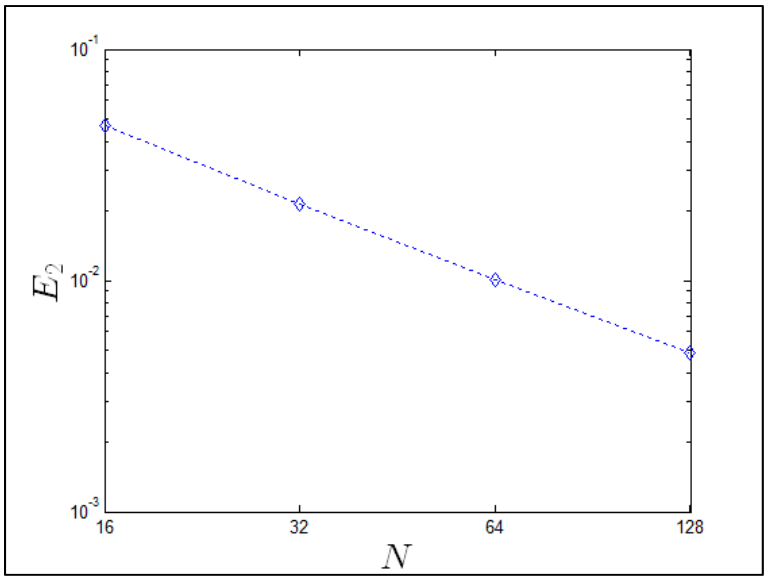


Figure 3.3: Error norm E_2 versus N

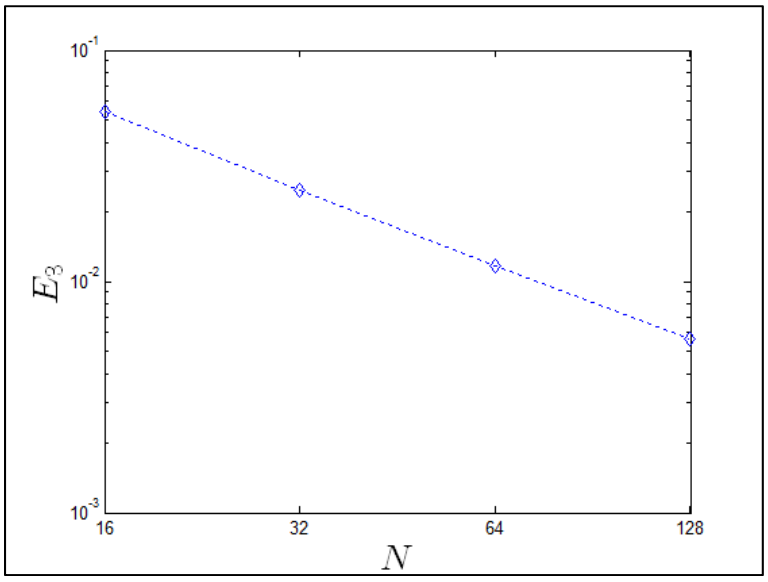


Figure 3.4: Error norm E_3 versus N

Figures 3.2 to 3.4 show that the error norms converge linearly with the mesh size. The geometric reinitialisation thus has a good mass conservation property.

3.4.2 Time-reversed vortex flow

We performed this test to evaluate the methods' sensitivity to a strong deforming interface separating two phases. This is a well-documented test case (Rider et Kothe, 1998) used to assess level set methods. The time-reversed vortex flow spins and stretches the fluids during the first half-period. The fluids then shrink and return to their original positions after one full period. A single vortex flow is defined by the stream function:

$$\Psi_{single\ vortex} = \frac{1}{\pi} sin^2(\pi x) sin^2(\pi y) \tag{3.28}$$

The time-reversed vortex flow is defined by multiplying the single vortex flow stream function by $cos\left(\frac{\pi t}{T}\right)$, where T is the period, so that its stream function is:

$$\Psi_{time-reversed\ vortex} = \frac{1}{\pi} sin^2(\pi x) sin^2(\pi y) cos\left(\frac{\pi t}{T}\right) \tag{3.29}$$

The relation between the velocity and the stream function is:

$$u = -\frac{\partial \Psi}{\partial y} \text{ and } v = \frac{\partial \Psi}{\partial x} \tag{3.30}$$

then,

$$u_{time-reversed\ vortex} = -2sin^2(\pi x) cos(\pi y) sin(\pi y) cos\left(\frac{\pi t}{T}\right) \tag{3.31}$$

and

$$v_{time-reversed\ vortex} = 2cos(\pi x) sin(\pi x) sin^2(\pi y) cos\left(\frac{\pi t}{T}\right) \tag{3.32}$$

A circle separates the unit square computational domain into two subdomains. The circle is centered at (0.5, 0.75), and its radius is 0.15.

3.4.2.1 Comparisons between the penalty and the P_0 projection method

In the following tests, we use unstructured meshes with triangular quadratic elements. An example of such a mesh with the interface at the initial instant is shown in Figure 3.5.

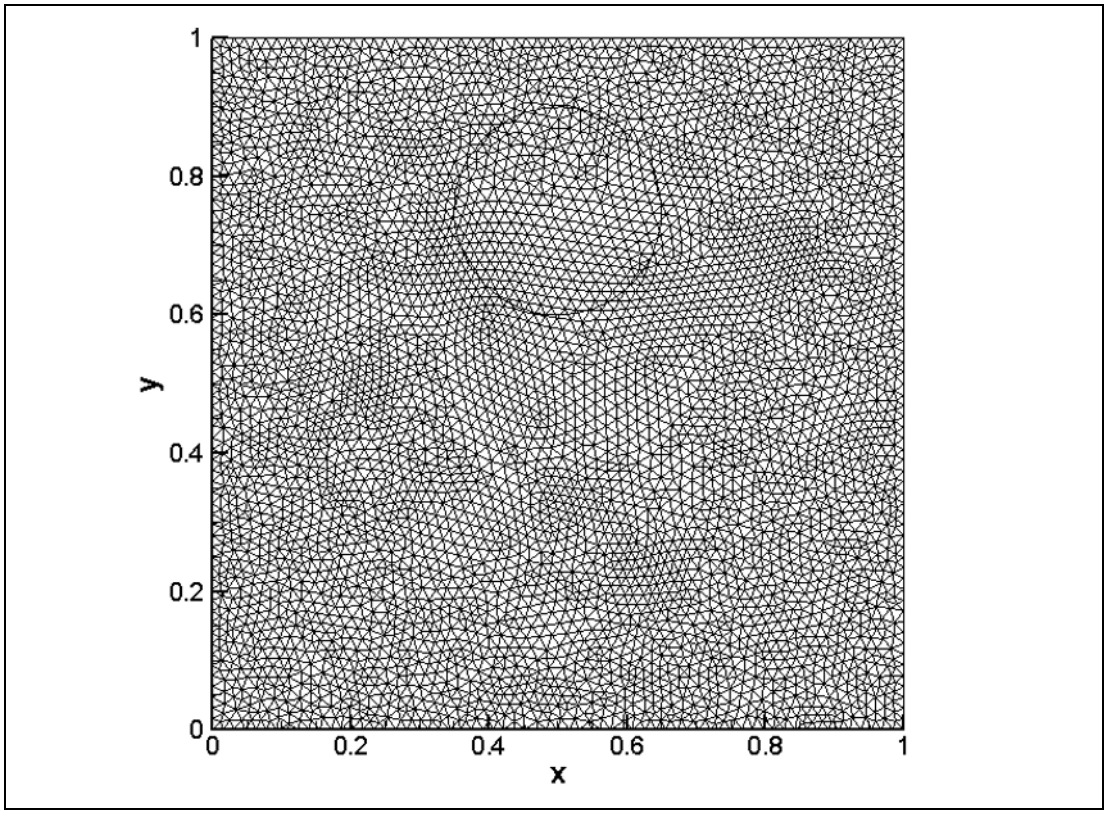


Figure 3.5: Time-reversed vortex flow. Mesh of 65×65 and initial interface at t = 0

The four meshes considered are listed in Table 3.1, with the number of nodes and number of elements for each mesh.

Table 3.1: Time-reversed vortex flow.
Number of nodes and number of elements.

Mesh ($N \times N$)	Number of nodes	Number of elements
65×65	19,185	9,464
85×85	33,977	16,820
105×105	52,473	26,028
125×125	58,609	29,108

The simulation was carried out for a period $T = 8$. The flow reverses after the half-period. First, a sensitivity analysis was conducted by varying the β_i $(i = 1, 2)$ parameter, using the 85×85 mesh and with the time step fixed to $\Delta t = 8 \times 10^{-3}$. The results of the disk mass percentage at $t = T$ are shown in Figure 3.6.

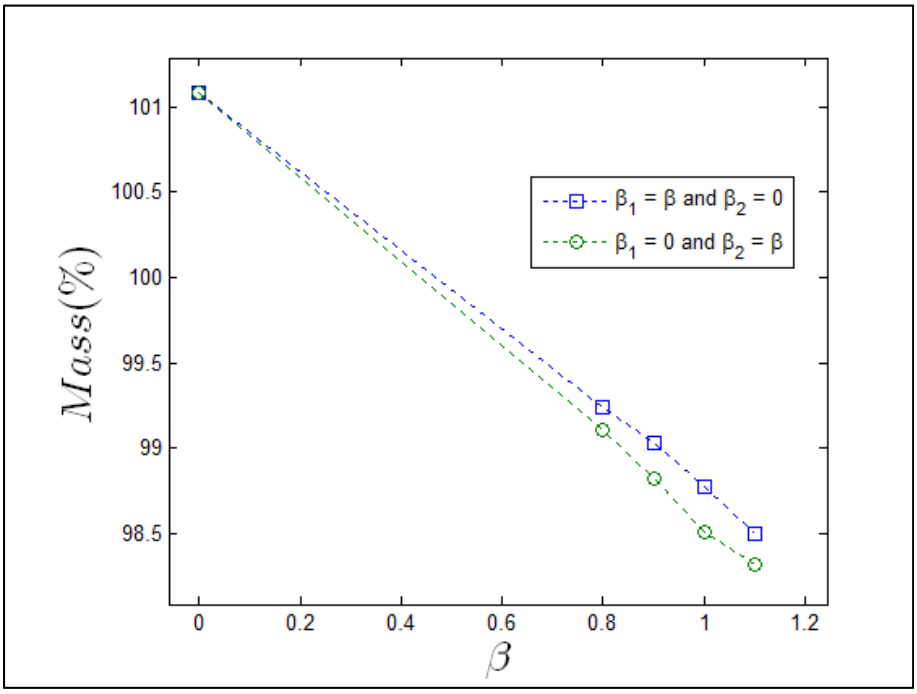


Figure 3.6: Time-reversed vortex flow.
Percentage of the disk mass at $t = T$ for $T = 8$.

The interface contours obtained for $t = 0, T/4, T/2, 3T/4$, and T using the 85×85 mesh with the proposed stabilized finite element methods and the geometric reinitialisation at each time step are shown in Figure 3.7. The contours correspond to the parameters enumerated in Legend 2.

$\beta_1 = 0$ and $\beta_2 = 0.8$

$\beta_1 = 0$ and $\beta_2 = 1$

geometric reinitialisation at each time step

Initial interface

Legend 2

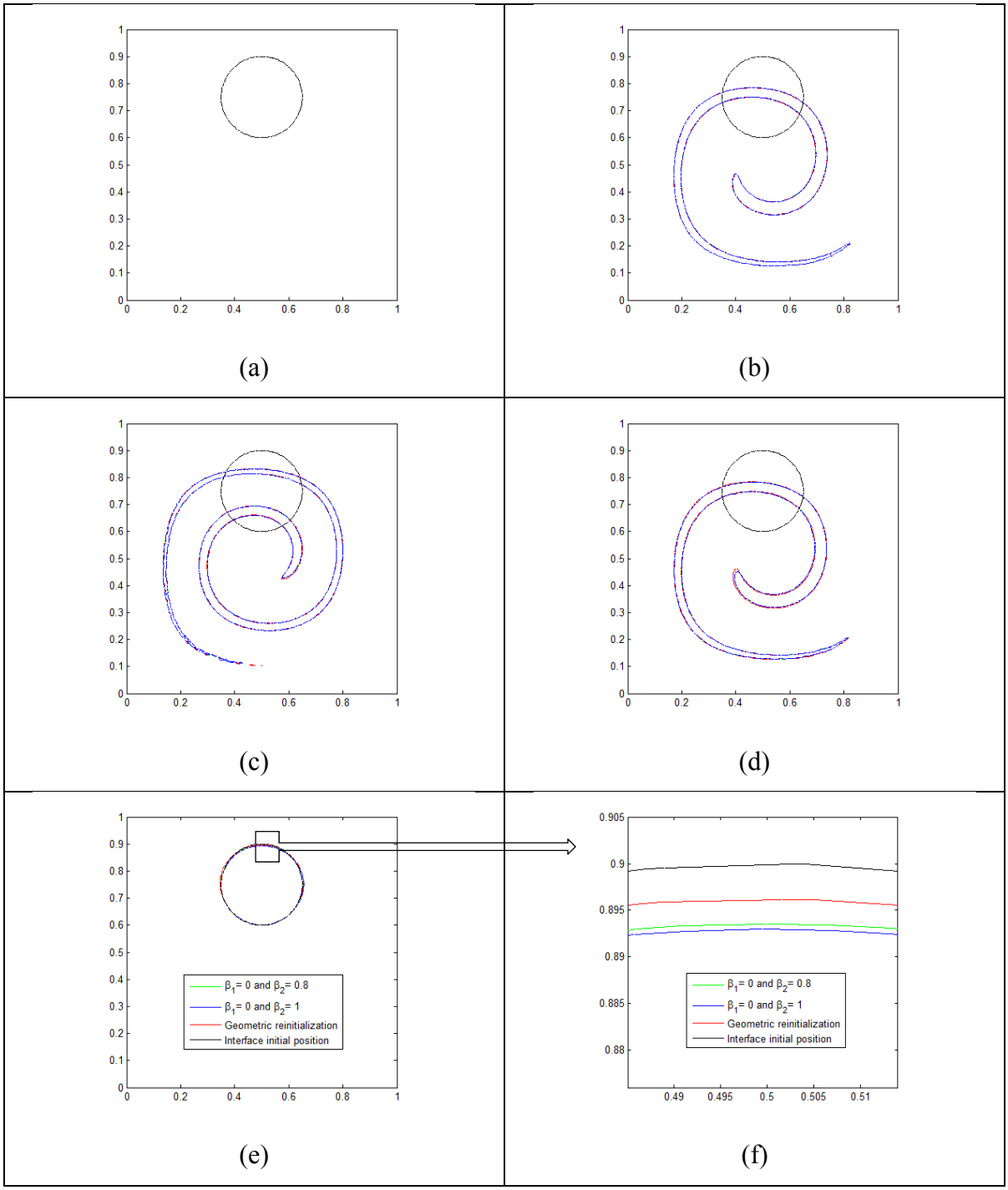


Figure 3.7: Time-reversed vortex flow. Interface positions at $t = 0\,(a), T/4\,(b), T/2\,(c), 3T/4\,(d)$ and $T\,(e)$ and with the 85×85 mesh. Closer view of interface positions at T (f).

In Figure 3.7, the computed interfaces look almost superposed, with a slight shift from the exact contour.

The mass conservation is analyzed using the stabilized and the geometric reinitialisation (taken as the reference method) methods for the four meshes. The values of the parameters used are enumerated in Legend 3.

Case 1: $\beta_1 = 1$ and $\beta_2 = 0$

Case 2: $\beta_1 = 0$ and $\beta_2 = 1$

Case 3: $\beta_1 = 0$ and $\beta_2 = 0$ with geometric renitialization at each time step

Legend 3

The time evolution of the disk mass is shown in Figures 3.8 to 3.10. In all cases, the mesh refinement effect is clearly shown; the mass (%) values are closer to 100% for the 125×125 fine mesh.

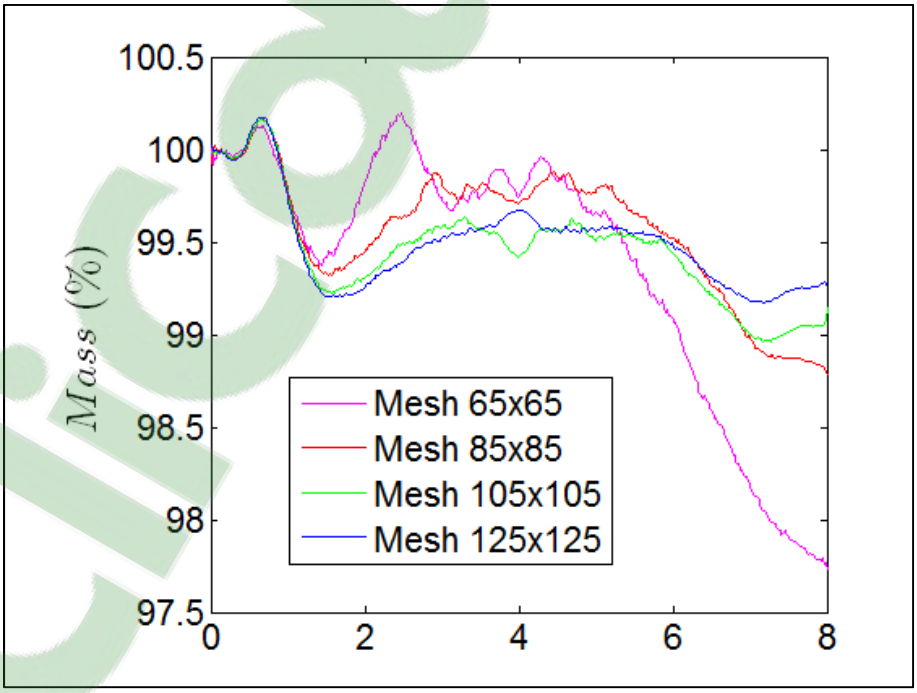


Figure 3.8: Time-reversed vortex flow. Percentage of the disk area $Mass(\%)$ versus time for case 1: $\beta_1 = 1$ and $\beta_2 = 0$.

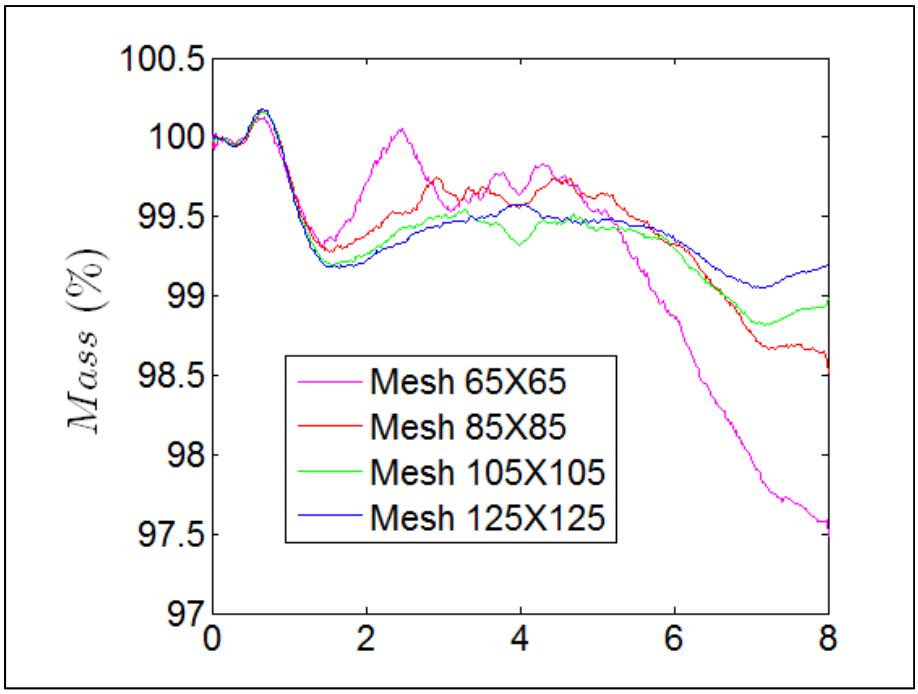


Figure 3.9: Time-reversed vortex flow. Percentage of the disk area $Mass(\%)$ versus time for case 2: $\beta_1 = 0$ and $\beta_2 = 1$.

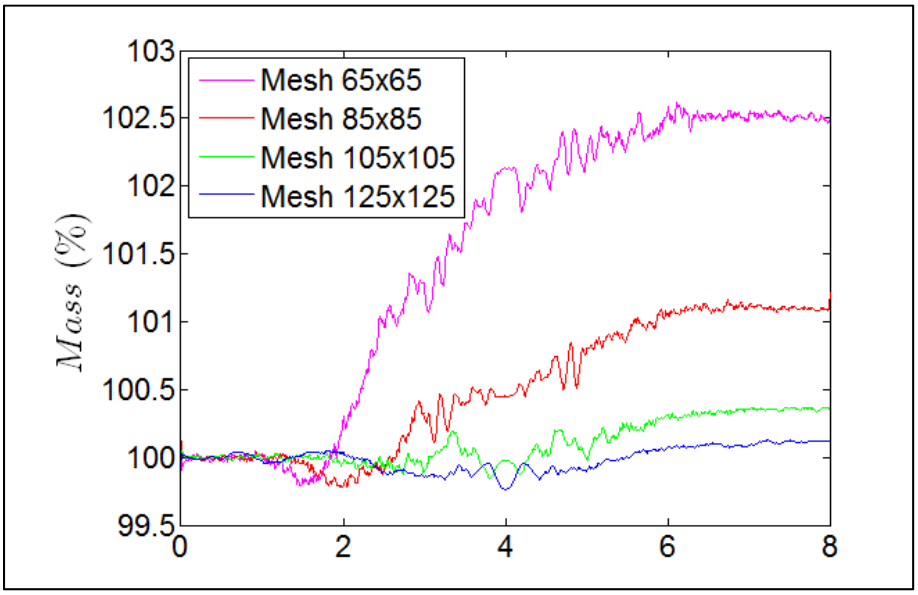


Figure 3.10: Time-reversed vortex flow. Percentage of the disk area $Mass(\%)$ versus time for case 3: $\beta_1 = 0$ and $\beta_2 = 0$ with geometric reinitialisation at each time step.

The stabilization methods and the geometric reinitialisation method at each time step deliver similar mass conservation results for the coarse and medium meshes. Mesh convergence can also be observed with mesh size refinement.

3.4.2.2 Comparisons between SUPG, stabilized projections and GLS methods

In this sub-section, the proposed methods are evaluated using the same test problem. Structured meshes with quadratic (nine-node) elements are used in order to perform a consistent and simple mesh refinement procedure and to obtain the order of convergence using the norm E_2. The meshes are constructed by decomposing each axis into N quadratic elements. The time step is fixed to $\Delta t = 8\times10^{-3}$. Figure 3.11 and

Table 3.2 show the masses for the projection methods compared to those obtained with the SUPG method. It is clear that stabilized methods improve the mass conservation for coarse meshes, with a slight advantage for the P_1 projection. Figure 3.12 shows the E_2 norms; these are increased compared to when using SUPG. For the SUPG method combined with Crank-Nicholson time discretization, the mathematical convergence analysis (Burman, 2010; Reusken et Loch, 2011) gives the bound: $E_2 \leq c\,T\,(h^p + \Delta t^2)$ with $p = k + \frac{1}{2}$, where k is the finite element interpolation order (in the present case $k = 2$) and $h = 1/N$ the mesh size. The numerical tests show that p is close to 2.45 for the SUPG method and 2.39 for the other methods. It can be concluded that the stabilized methods improve the numerical interface positions that are crucial for moving interface problems, but they may have some effect on the global error over the entire domain.

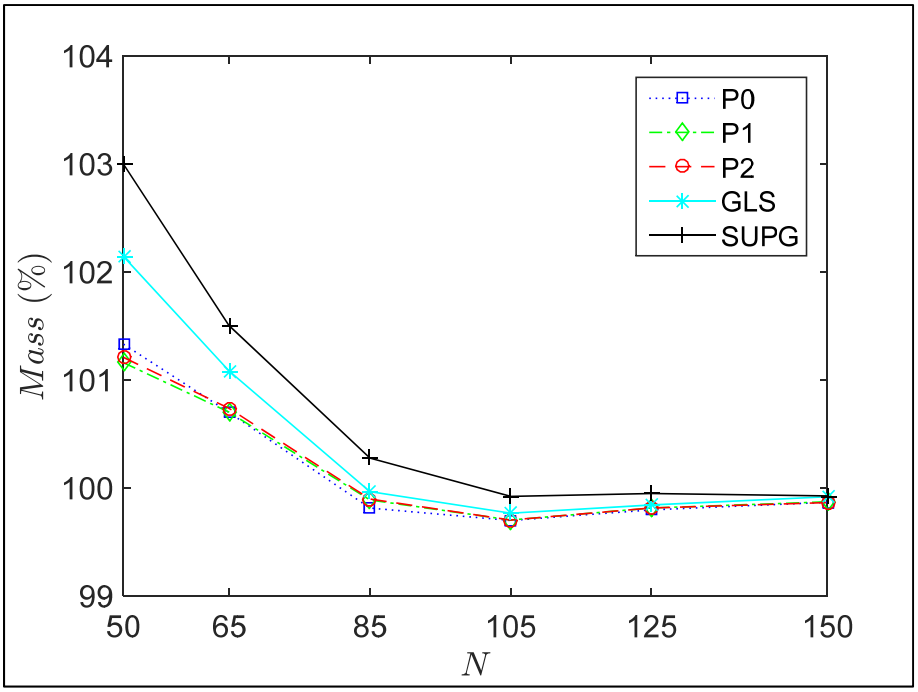


Figure 3.11: Time-reversed vortex flow. Percentage of the disk area $Mass(\%)$ versus N

Table 3.2: Mass conservation at $t = T$

Mass (%)	50×50	65×65	85×85	105×105	125×125	150×150
P0	101.3347	100.7017	99.8164	99.7017	99.7999	99.8697
P1	101.1630	100.7041	99.8914	99.7056	99.8174	99.8743
P2	101.2122	100.7397	99.9003	99.7056	99.8174	99.8672
GLS	102.147	101.086	99.9685	99.7684	99.8452	99.9213
SUPG	103.0011	101.5056	100.2795	99.9245	99.9515	99.929

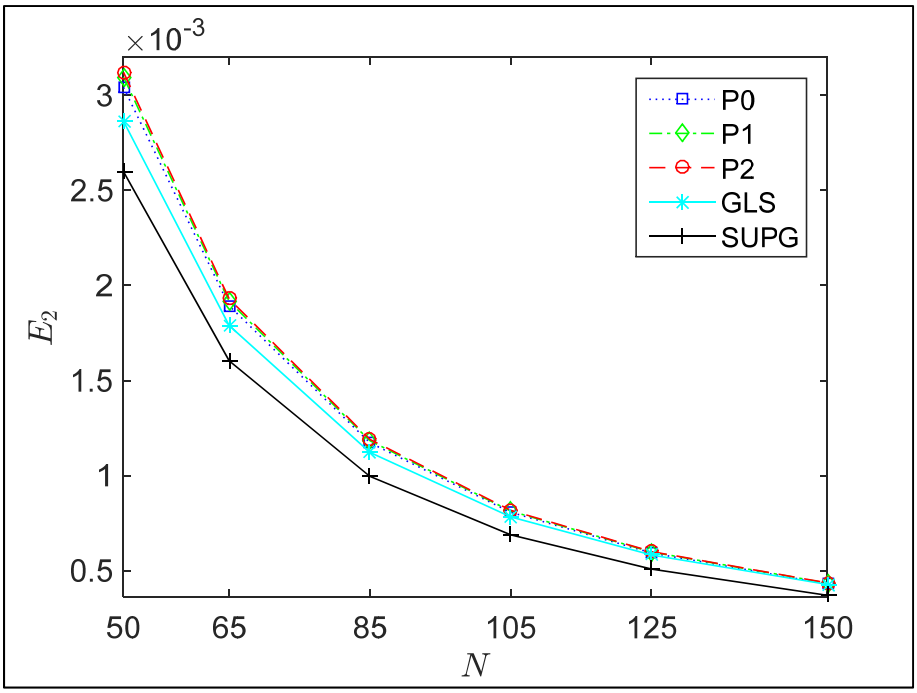


Figure 3.12: Time-reversed vortex flow. Error norm E_2 versus N

3.4.3 Rigid body rotation of Zalesak's disk

This well-documented test case (Rider et Kothe, 1998; Zalesak, 1979) was performed to examine the errors induced by a rotating flow field on an undeforming interface constituted by a circle with a small slot. The computational domain is a disk with a unit radius chosen to avoid any numerical problem that might be introduced by the inflow boundary conditions. Indeed, the imposed flow field has a zero normal velocity. The interface is a slotted circle centered at (0.5, 0.75) and its radius is 0.15. The slot has a height of 0.25 and a width of 0.05. The velocity components are given as:

$$u = \frac{2\pi}{T}(0.5 - y) \text{ and } v = \frac{2\pi}{T}(x - 0.5) \tag{3.33}$$

with $T = 6.28$. Four meshes are again considered. The number of nodes and the number of elements are listed in Table 3.3. The number of nodes and elements used is quite large in order

to accurately reproduce the sharp slot corners. The mesh and the interface at the initial instant are shown in Figure 3.13 for the 65×65 coarse mesh.

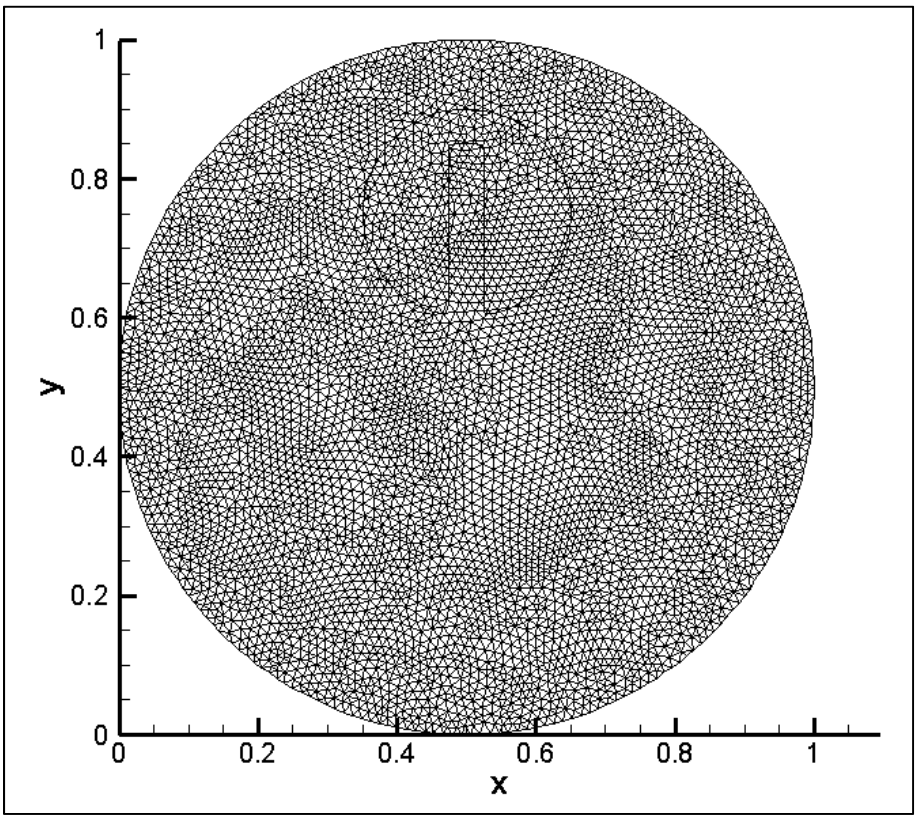


Figure 3.13: Rigid body rotation of Zalesak's disk.
Coarse mesh and initial interface at t = 0.

Table 3.3: Rigid body rotation of Zalesak's disk.
Number of nodes and number of elements for each mesh.

Mesh ($N\times N$)	Number of nodes	Number of elements
65×65	16,449	8,122
85×85	26,475	13,104
105×105	40,141	19,906
125×125	57,493	28,550

The first simulations are carried out for one time period, for a time step $\Delta t = 6.28\times10^{-3}$ using the 85×85 mesh. A sensitivity analysis is carried out by varying the parameters β_i, enumerated in Legend 4:

Case 1: $\beta_1 = \beta$ and $\beta_2 = 0$

Case 2: $\beta_1 = 0$ and $\beta_2 = \beta$

Legend 4

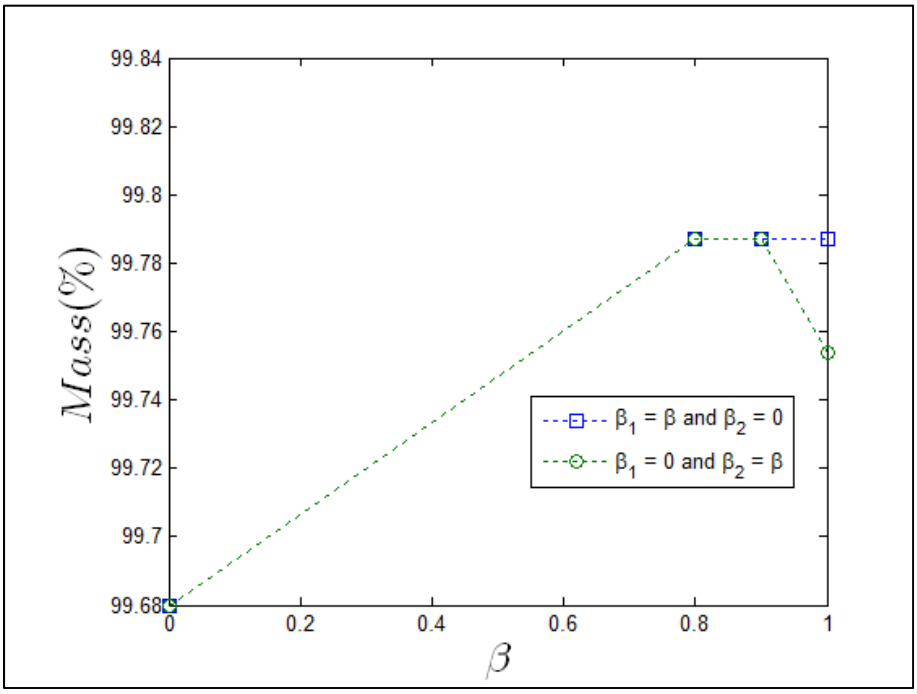


Figure 3.14: Rigid body rotation of Zalesak's disk.
Percentage of the disk mass at $t = T$.

The results shown in Figure 3.14 indicate that the best results for the stabilized and penalty methods correspond to an optimal value of β_1 and β_2 close to one.

The interface positions obtained for $t = 0, \frac{T}{4}, \frac{T}{2}, \frac{3T}{4}$, and T with the 85×85 mesh using the new stabilized finite element method and the geometric reinitialisation are shown in Figure 3.15(a, b).

$\beta_1 = 0$ and $\beta_2 = 0.8$

$\beta_1 = 0$ and $\beta_2 = 1$

geometric reinitialisation at each time step

Initial interface

Legend 5

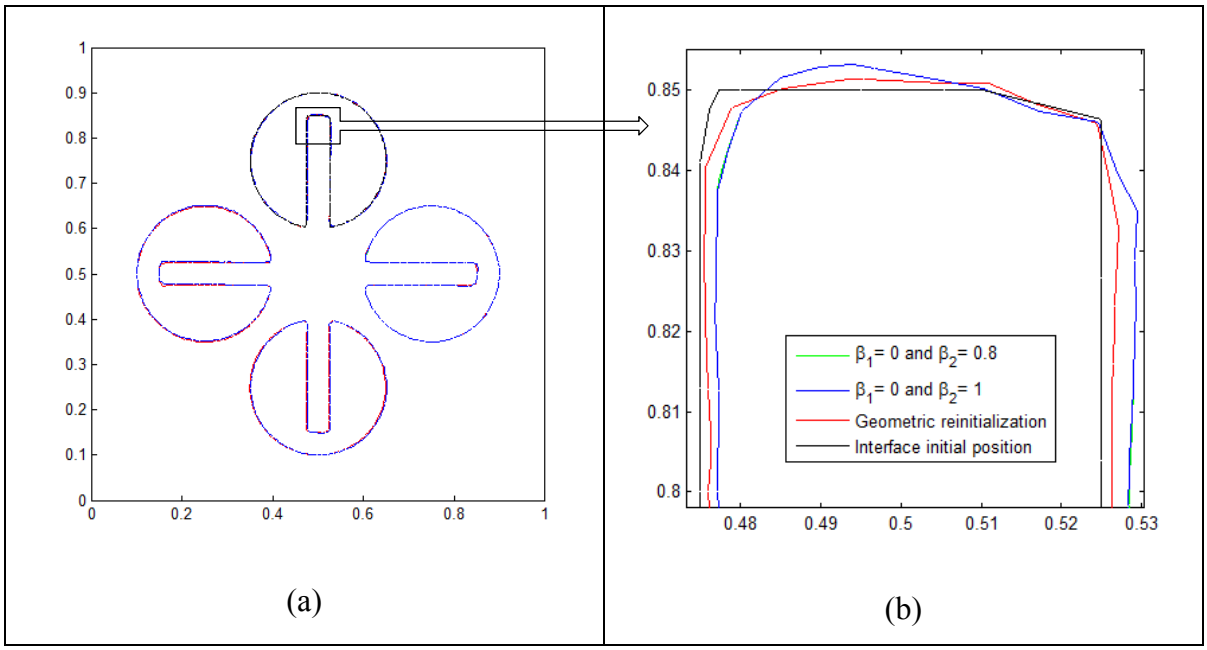


Figure 3.15: Rigid body rotation of Zalesak's disk. Interface positions at $t = 0, T/4, T/2, 3T/4$, and T for the stabilized formulation and for with the 85×85 mesh (a). Closer view of interface positions at T (b).

In Figures 3.15(a, b), the computed interfaces appear slightly shifted to the right from the exact contour, and the corners are rounded. The results of the disk's mass obtained at the final time for the different formulations are illustrated in Figure 3.16.

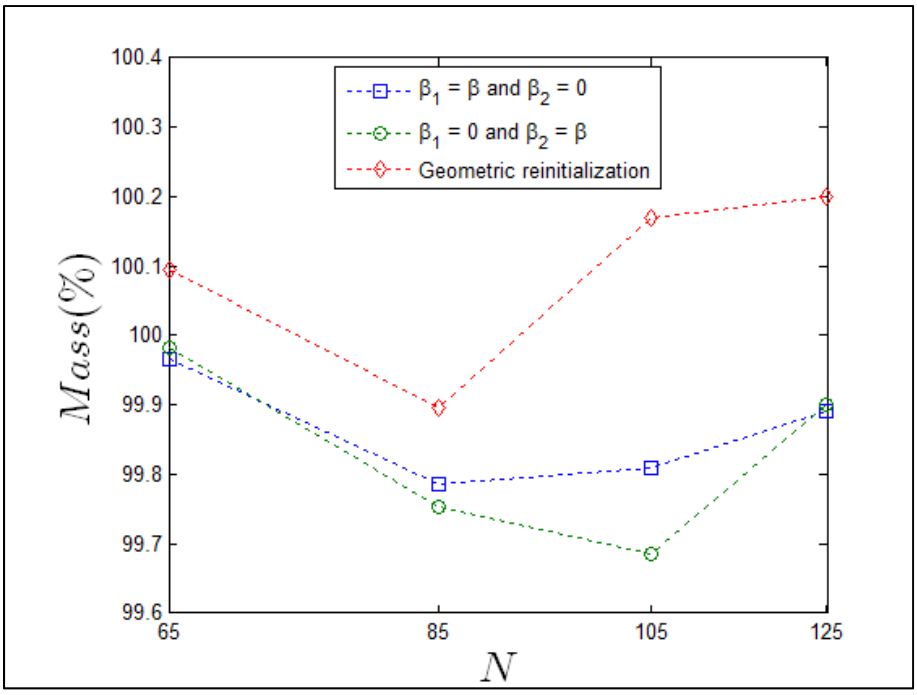


Figure 3.16: Rigid body rotation of Zalesak's disk. Percentage of the disk area $Mass(\%)$ at $t = T$ for $T = 6.28$.

Figure 3.16 shows that the mass fluctuations are of the same order when using either the stabilized methods or the geometric renitialisation method.

3.4.4 Dam-break problem

In this test case, the proposed methods are used to solve a flow problem where the velocity field is computed by solving the Navier-Stokes equations for a two-phase flow. The Navier-Stokes equations and the numerical formulation are briefly presented in the following sub-sections.

3.4.4.1 The Navier-Stokes equations for two-phase flows

The conservation of momentum and mass are described by the Navier-Stokes equations for two-immiscible incompressible fluids:

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u}\right)+\nabla\tilde{p}-\nabla\cdot\boldsymbol{\sigma}(\mathbf{u})=\rho\,\mathbf{g} \tag{3.34}$$

$$\nabla\cdot\mathbf{u}=0 \tag{3.35}$$

in a fixed space-time domain $\Omega\times[0,T], \Omega\subset\mathbb{R}^2$ where ρ is the density, $\mathbf{u}$ is the velocity, $\tilde{p}$ is the pressure, $\boldsymbol{\sigma}(\mathbf{u})=\mu\left(\frac{\nabla\mathbf{u}+(\nabla\mathbf{u})^T}{2}\right)$ where μ is the dynamic viscosity coefficient, and $\rho\,\mathbf{g}$ is the body force. It is assumed that fluid 1 occupies the domain Ω_1 and fluid 2 occupies Ω_2 such that the material interface is $\Gamma=\partial\Omega_1\cap\partial\Omega_2$. The density and the dynamic viscosity are discontinuous across the interface:

$$\rho(\mathbf{x},t)=\begin{cases}\rho_1 & \text{for } \mathbf{x}\in\Omega_1(t)\\ \rho_2 & \text{for } \mathbf{x}\in\Omega_2(t)\end{cases} \tag{3.36}$$

and

$$\mu(\mathbf{x},t)=\begin{cases}\mu_1 & \text{for } \mathbf{x}\in\Omega_1(t)\\ \mu_2 & \text{for } \mathbf{x}\in\Omega_2(t)\end{cases} \tag{3.37}$$

Since the surface tension effects are neglected, the interface conditions are:

$$\llbracket\mathbf{u}\rrbracket=0 \quad\text{and}\quad \llbracket-\tilde{p}\,\mathbf{I}+\boldsymbol{\sigma}\mathbf{n}\rrbracket\cdot\mathbf{n}=0 \tag{3.38}$$

where $\mathbf{n}$ is the unit normal at the interface Γ pointing from Ω_1 to Ω_2, $\mathbf{I}$ is the identity tensor, and $\llbracket f\rrbracket=f_{|\Omega_1\cap\Gamma}-f_{|\Omega_2\cap\Gamma}$ is the jump across the interface of a function f.

3.4.4.2 Stabilized variational formulation of the Navier-Stokes equations

The stabilized variational formulation of the Navier-Stokes equations adopted here is briefly described (Rasthofer et al., 2011; Schott et al., 2014). Assuming suitable finite dimensional

spaces for $(\mathbf{u}, p)$ as well as for the weighting functions $(\mathbf{v}, q)$, a variational formulation of equations (3.34) and (3.35) is given as: find $(\mathbf{u}, p)$ such that for all $(\mathbf{v}, q)$:

$$\left(\mathbf{v}, \rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right)\right) - (\nabla\cdot\mathbf{v}, \tilde{p}) + (\nabla\cdot\mathbf{v}, \boldsymbol{\sigma}) + \mathrm{d}(\mathbf{v}, \mathbf{u}) = (\mathbf{v}, \rho\,\mathbf{g}) + (\mathbf{v}, \mathbf{s})_{\partial\Omega} \tag{3.39}$$

$$(q, \nabla\cdot\mathbf{u}) = 0 \tag{3.40}$$

where $(.,.) = (.,.)_{\Omega}$ and $(.,.)_{\partial\Omega}$ denote the L_2 inner product on Ω and $\partial\Omega$ respectively, and $\mathbf{s} = \boldsymbol{\sigma}\cdot\mathbf{n}$. The term $\mathrm{d}(\mathbf{v}, \mathbf{u})$ is added to the standard Galerkin weak form to stabilize the flow field in case of high Reynolds numbers and is given as:

$$\mathrm{d}(\mathbf{v}, \mathbf{u}) = (\mathbf{u}\cdot\nabla\mathbf{v}, \tau_M \Re(\mathbf{u}, \tilde{p})) + (\nabla\cdot\mathbf{v}, \tau_C \nabla\cdot\mathbf{u}) \tag{3.41}$$

with

$$\Re(\mathbf{u}, \tilde{p}) = \rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) + \nabla\tilde{p} - \nabla\cdot\boldsymbol{\sigma} - \rho\,\mathbf{g} \tag{3.42}$$

Note that there is no stabilization term added to the continuity equation since the finite elements approximations that are chosen respect the inf-sup stability condition. In particular, we will use the Taylor-Hood P_2-P_1 element that will be enriched by suitable functions to handle discontinuities inside the elements. The stabilization parameters τ_M and τ_C are given by:

$$\tau_M = \frac{h^e}{2|\mathbf{u}|} \text{ and } \tau_C = \frac{\left(\rho h^e |\mathbf{u}|\right)}{2}, \tag{3.43}$$

where h^e is the element size, defined as the minimum distance between two nodes.

Because of the viscosity and density differences between the phases across the interface, the velocity has a discontinuous gradient and the pressure is discontinuous. When the mesh is not aligned with the interface, the classical polynomial finite element approximations may not resolve such discontinuities accurately. The extended finite element (XFEM) approximations can account for these discontinuities by enriching the approximation spaces (Gross et Reusken, 2011). Therefore, the pressure-discrete space is enriched with the shifted sign enrichment, leading to the following approximation for the pressure:

$$\tilde{p}(\mathbf{x},t)^{XFEM} = \sum_{i\in I} N_i^p(\mathbf{x})\,\tilde{p}_i(t) + \sum_{i\in I^*} N_i^p(\mathbf{x})\,(\psi(\mathbf{x},t) - \psi_i(\mathbf{x},t))\,a_i(t) \tag{3.44}$$

where $N_i^p(\mathbf{x})$ is the standard linear shape function for node i, I is the set of all nodes in the domain Ω, $\tilde{p}_i$ are the nodal values of the pressure $\tilde{p}$, a_i are the additional XFEM degrees of freedom, I^* is the nodal subset of the enrichment and $\psi(\mathbf{x},t)$ is the global enrichment function defined as:

$$\psi\ (x,t) = \operatorname{sign}\left(\phi(x,t)\right) = \begin{cases} -1 & \text{if } \phi(\mathbf{x},t) < 0 \\ 0 & \text{if } \phi(\mathbf{x},t) = 0 \\ 1 & \text{if } \phi(\mathbf{x},t) > 0 \end{cases} \tag{3.45}$$

3.4.4.3 Numerical results

The dam-break problem is a challenging test for numerical methods, as it encompasses large interface deformations. The entire domain is a rectangle with the dimensions of 0.584 m by 0.365 m. The initial interface separates the domain into two regions. The lower left side subdomain is a rectangle with dimensions $0.146\,\text{m} \times 0.292\,\text{m}$, filled with water with a density of 998.2 kg/m^3 and a dynamic viscosity of 0.001Pa·s. The upper fluid has a density of 1.204 kg/m^3 and a dynamic viscosity of $1.75\ 10^{-5}$ Pa·s. The slip condition is applied to all boundaries. The simulation is conducted for a total time of 1 s with a time step Δt of $5\ 10^{-4} s$. The unstructured mesh has 1,804 quadratic triangular elements and 3,723 nodes. The mesh and the interface at the initial instant are shown in Figure 3.17. At the initial time $t = 0$, the water column is defined such that its front position z is a and its height η is $2a$ with $a = 0.146\,\text{m}$. The simulations presented here correspond to the experimental tests carried out by Martin and Moyce (Martin et Moyce, 1952) and Hu and Sueyoshi (Hu et Sueyoshi, 2010).

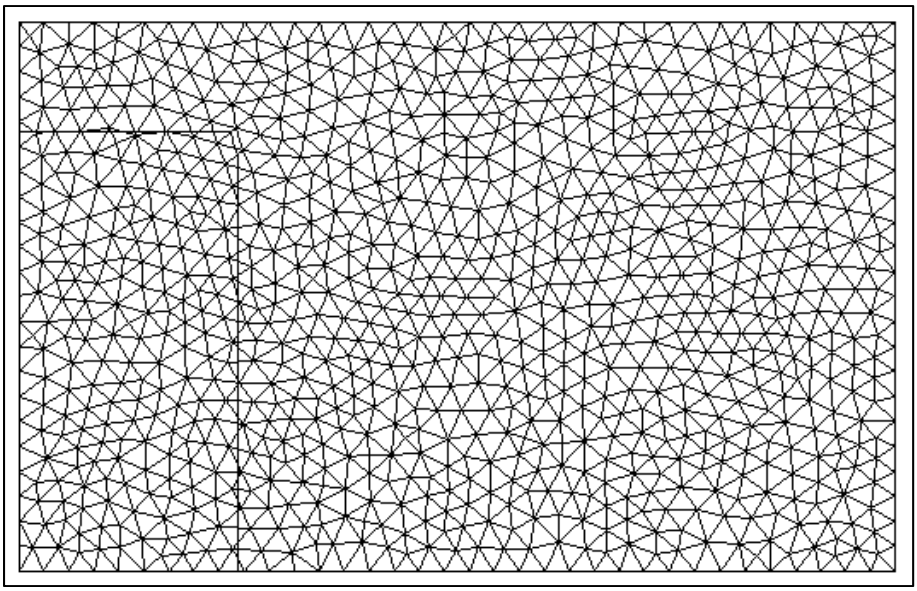


Figure 3.17: Dam break. Mesh and initial interface at t = 0 s.

The flow can be briefly described as follows: in a first stage from the initial time up to $t = 0.275\ s$ the water column collapses, maintaining a compact domain, and then violently hits the right wall. The fluid then rises along the wall and, under the effect of gravity, collapses again, creating a breaking wave. This wave propagates to the left of the field by trapping an air bubble, and continues on its way to hit the left wall and bounce back. Experimental data comprises the positions of the fluid front on the bed and on the left vertical wall, as well as pictures of the flow at certain times in the experiment.

Figures 3.18-3.20 show the interface positions at different times for the SUPG-geometric reinitialisation method and the P_1 stabilized projection method. Around time $t = 0.5\ s$ some oscillations appear close to the upper wall. With the SUPG-geometric reinitialisation method these oscillations degenerate and the code stops running, while it continues for the projection-stabilized methods. The results for the different projection methods are presented in Figures 3.21 to 3.25.

Comparisons with the experimental data for the front positions (z) on the bed and for the water height (η) on the left wall are shown in Figure 3.26 (a) and Figure 3.26 (b). Our numerical experiments show a solution for the surge front velocity that is faster than the results

found in Martin and Moyce (Martin et Moyce, 1952). This disparity is common for most interface-tracking methods (Lee, Dolbow et Mucha, 2014). Some discrepancies are observed between the results obtained with different projection methods when the wave breaks up. For all methods, the mass is well conserved (Figure 3.25).

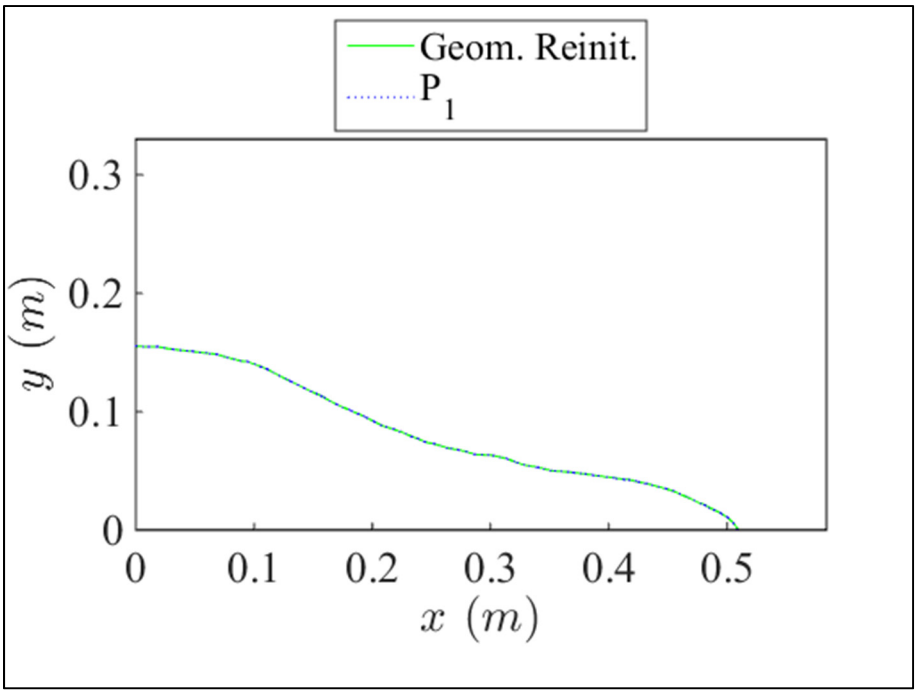


Figure 3.18: Dam break interface position at t = 0.23s
for the geometric reinitialisation and the P_1 projection.

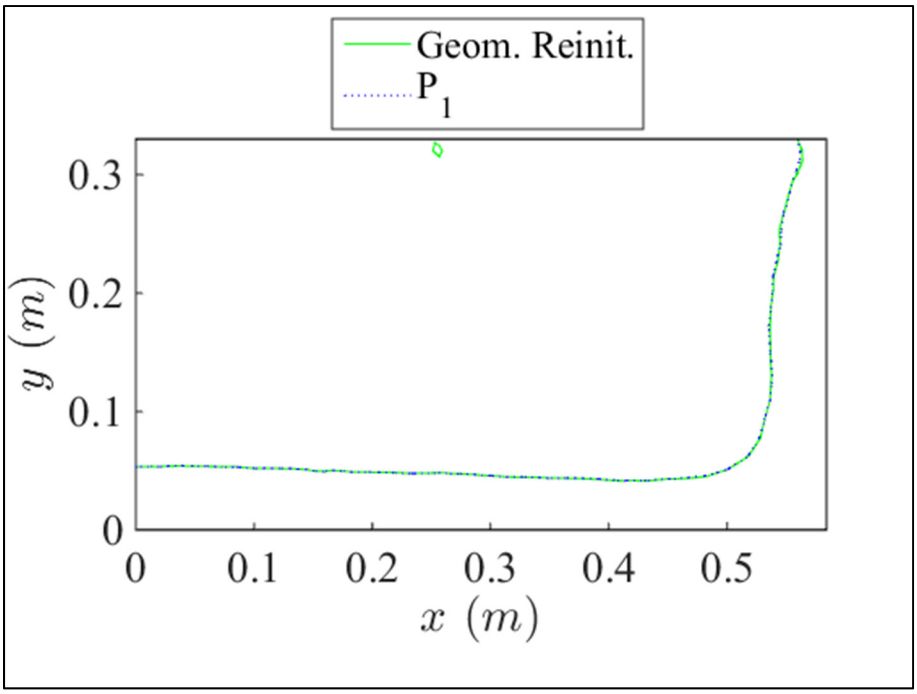


Figure 3.19: Dam break interface position at t = 0.5 s
for the geometric reinitialisation and the P_1 projection.

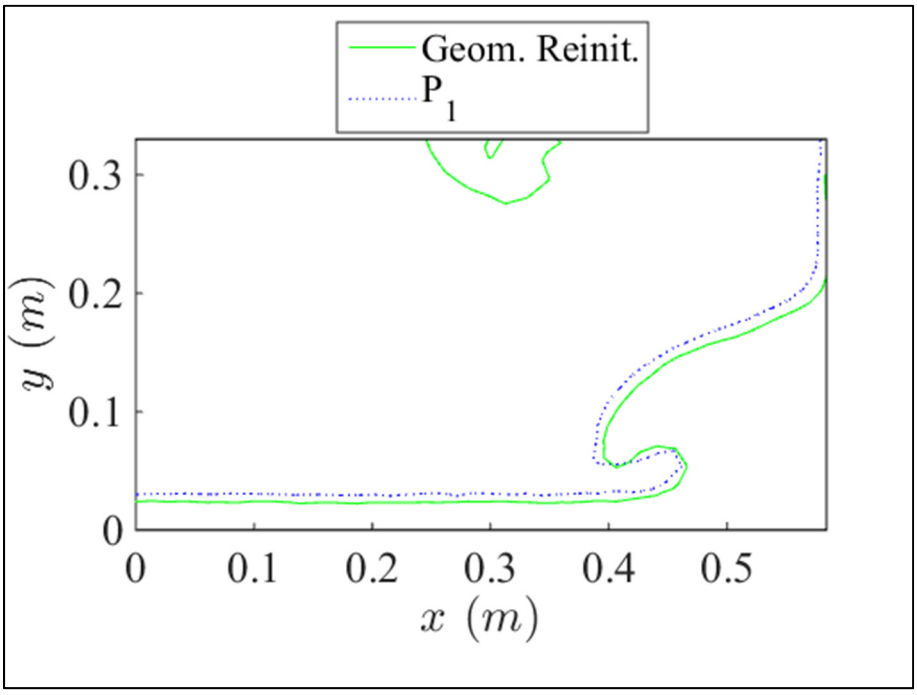


Figure 3.20: Dam break interface position at t = 0.75s
for the geometric reinitialisation and the P_1 projection.

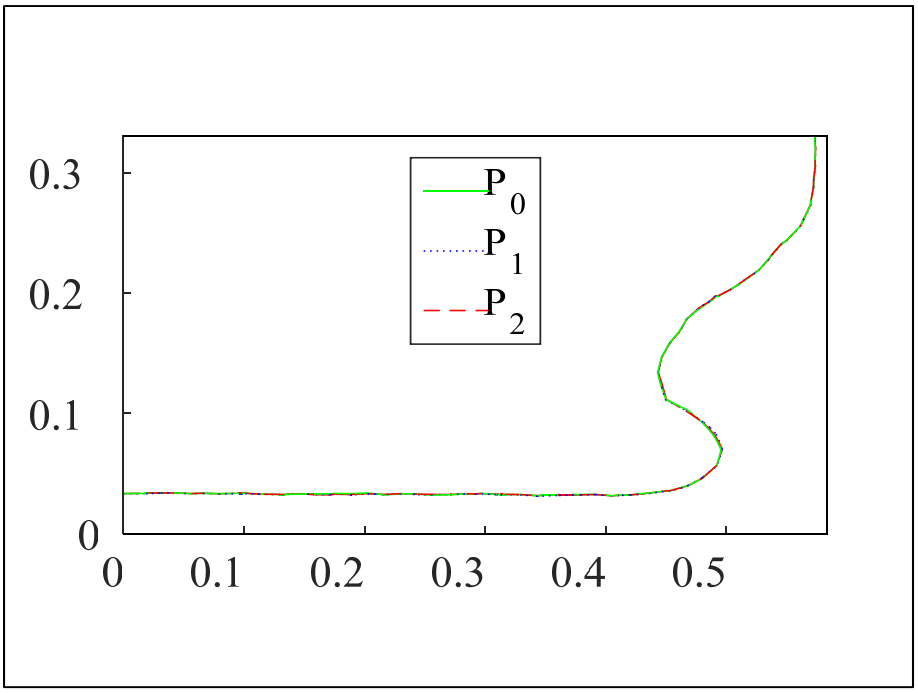


Figure 3.21: Dam break interface position at t = 0.7s
for the P_1, P_2 and P_3 projections.

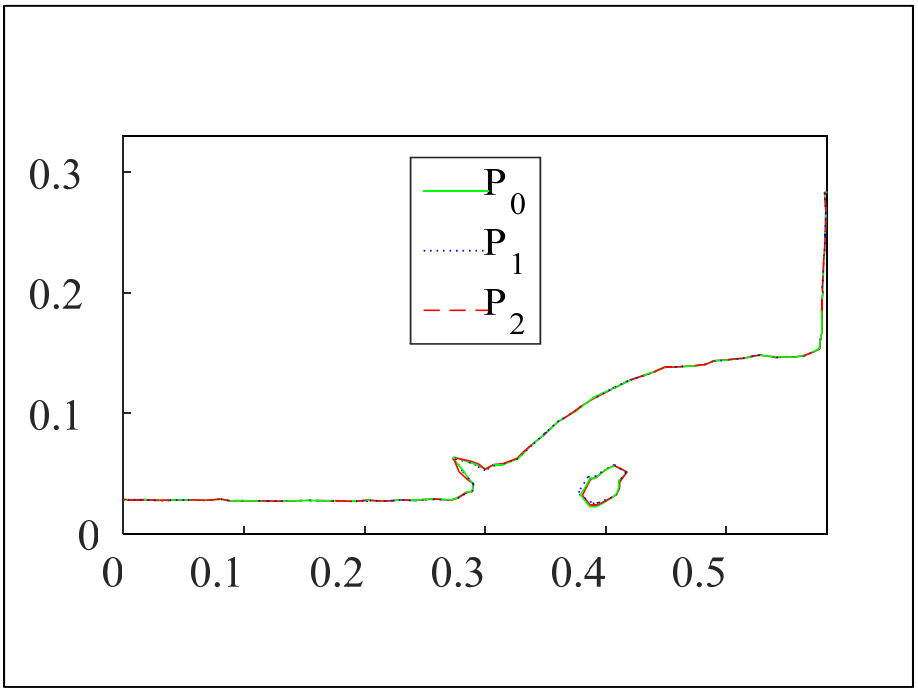


Figure 3.22: Dam break interface position at t = 0.8s
for the P_1, P_2 and P_3 projections.

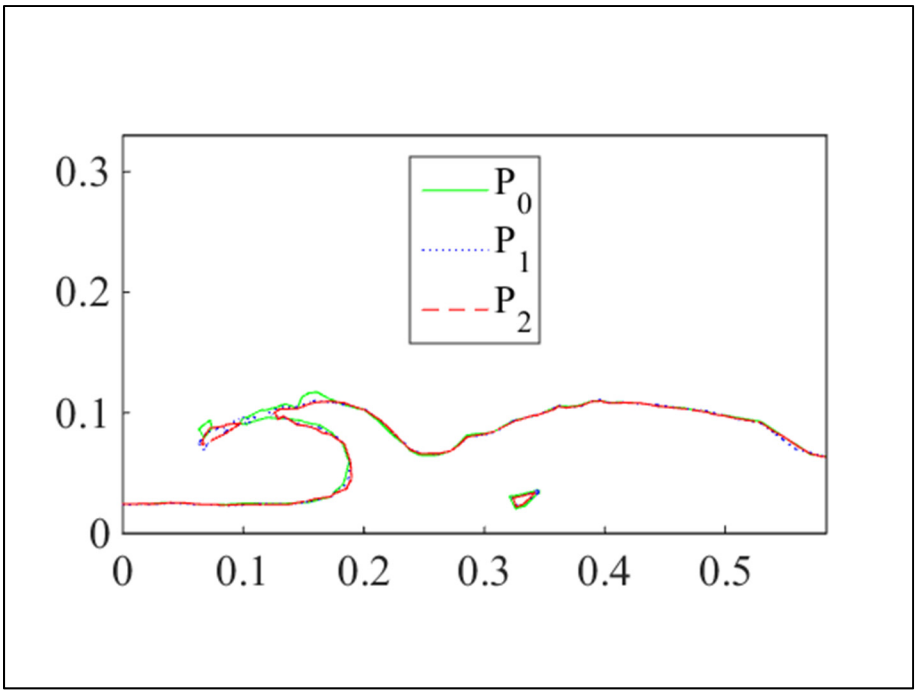


Figure 3.23: Dam break interface position at t = 0.9s
for the P_1, P_2 and P_3 projections.

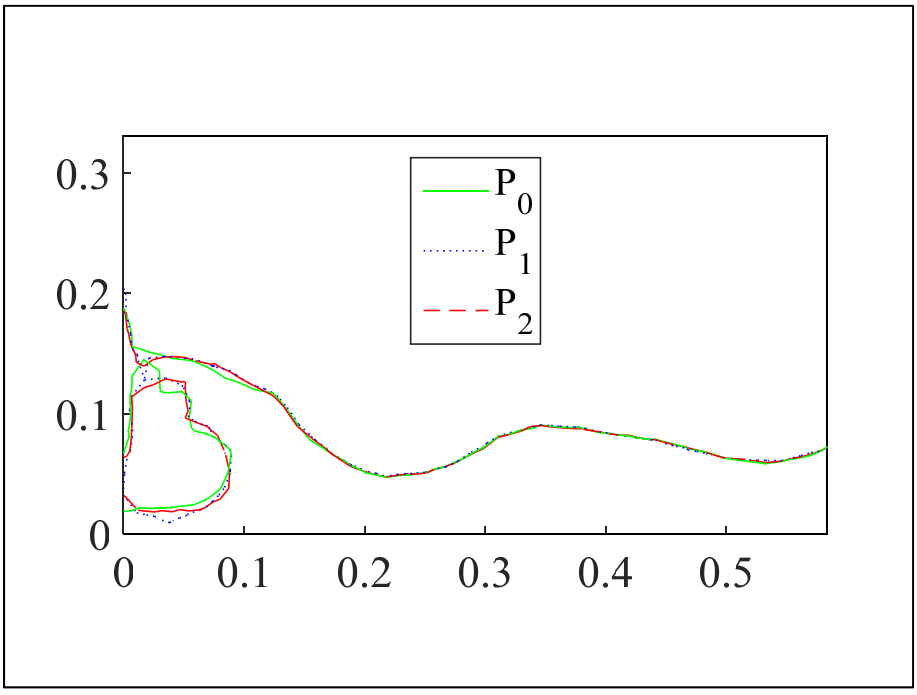


Figure 3.24: Dam break interface position at t = 1.0s
for the P_1, P_2 and P_3 projections.

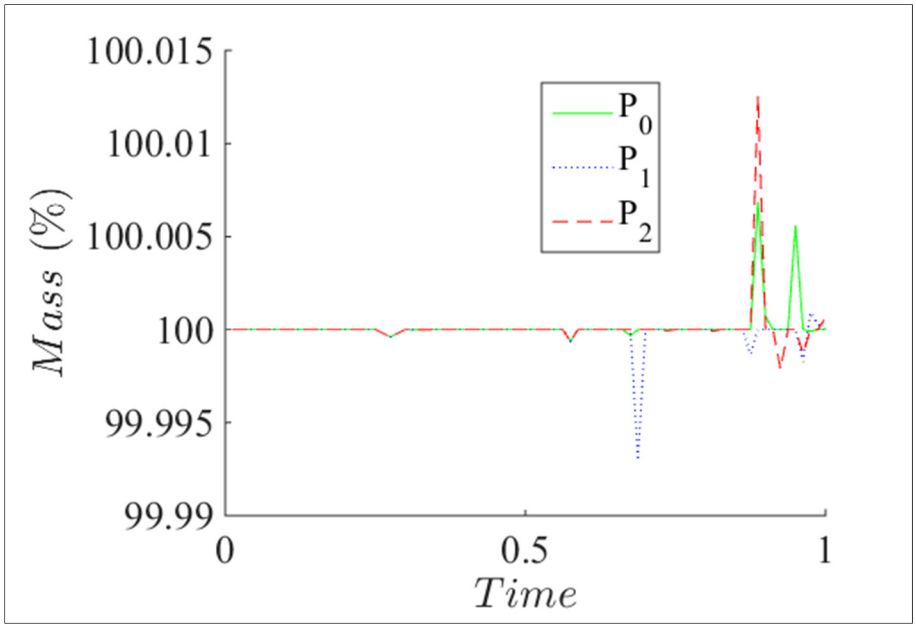


Figure 3.25: Dam break. Evolution of the lower subdomain $Mass(\%)$.

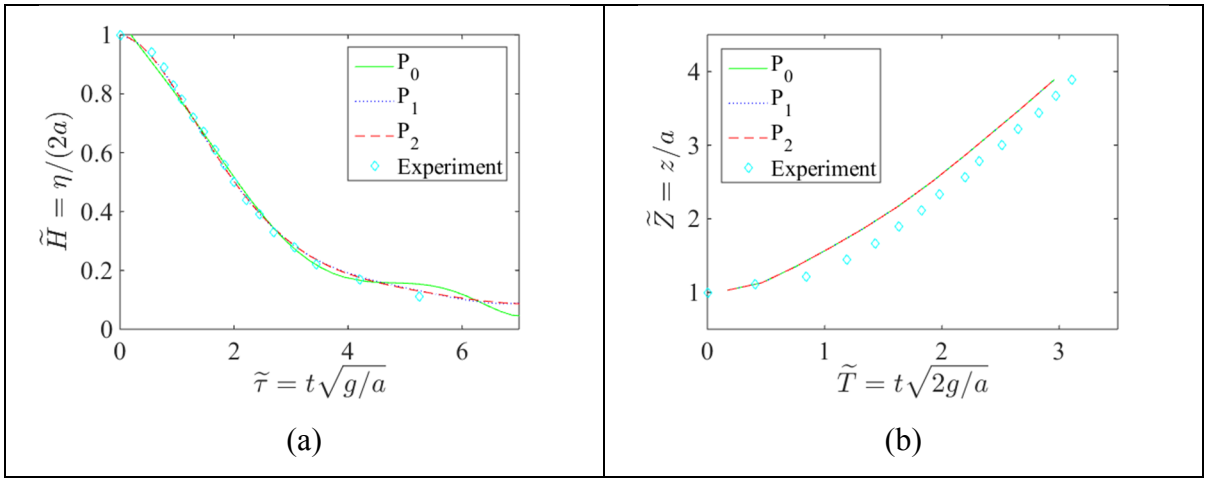


Figure 3.26: Dam break. Position of the water column front on the bed (a) and height of the water column on the left boundary (b).

3.5 Conclusions

In this work, new stabilized variational formulations are proposed for solving the level set equation without recourse to the reinitialisation process. These formulations are obtained by adding new terms to the basic SUPG formulation which depend on the residual of the Eikonal

equation. These methods were compared to a modified variant of the penalty method proposed by Li et al. (Li et al., 2005) and to the brute-force geometric reinitialisation used at every time step. The discretization was done in time using a second-order implicit scheme and in space with quadratic finite element approximations. The numerical comparisons were made using the standard benchmark tests of a time-reversed vortex, a rigid body motion of Zalesak's disk and the dam break problem. The proposed stabilized methods improve the numerical behavior of the basic SUPG method. They are effective at capturing strongly deformed interfaces and deliver acceptable mass conservation. These methods are relatively easy to implement in existing finite element solvers in 2D as well as in 3D.

APPENDIX

The time discretization and solution algorithm for (3.20) is detailed in the following.

For any function $f(t)$ we will use the standard notation: $f(t = n\,\Delta t) = f^n$ and $f^{n+\frac{1}{2}} = \dfrac{f^n + f^{n+1}}{2}$ with Δt the time step and n a positive integer.

Applying the Crank-Nicholson algorithm, (3.30) becomes:

$$\int_\Omega \left[\left(\psi + (\tau \mathbf{u})^{n+\frac{1}{2}} \cdot \nabla \psi \right) \left(\frac{\phi^{n+1} - \phi^n}{\Delta t} + \mathbf{u}^{n+\frac{1}{2}} \cdot \nabla \phi^{n+\frac{1}{2}} \right) + p^{n+\frac{1}{2}} \left(\frac{\nabla \phi}{|\nabla \phi|}^{n+\frac{1}{2}} \cdot \nabla \psi \right) \right] d\Omega = 0 \qquad \text{(A.1)}$$

Equation (A.1) is nonlinear in terms of ϕ^{n+1}. It is solved using an iterative quasi-Newton algorithm. Let $\phi^{n+1,m}$ be the solution at iteration m, we look for the correction $\Delta\phi$ such that $\phi^{n+1,m+1} = \phi^{n+1,m} + \Delta\phi$ is the solution of a linearized version of (A.1). To use the quasi-Newton algorithm, one needs the variation of $\left(\dfrac{p}{|\nabla \phi|} \right)^{n+1}$ with respect to $\phi^{n+1,m}$. We use the approximation: $\Delta \left(\dfrac{p}{|\nabla \phi|} \right)^{n+1} \approx -p^{n+1,m} \dfrac{\nabla \phi^{n+1,m} \cdot \nabla \Delta \phi}{|\nabla \phi^{n+1,m}|^{\frac{3}{2}}}$.

This iterative process converges in a few iterations, provided that the level set is oscillation-free.

The general solution algorithm for coupling the Navier-Stokes equations with the level set is summarized in the following:

- Initialization given $\left(\mathbf{u}(\mathbf{x},0), \tilde{p}(\mathbf{x},0), \phi(\mathbf{x},0)\right)$
- Loop over time steps:

  For $n = 1 : Nsteps$

  - Solve the Navier-Stokes equations (3.39) and (3.40) to get $\left(\mathbf{u}^n, \tilde{p}^n\right)$;
  - Solve the level set equation (3.20) or (3.24) to get ϕ^n;
  - Solve the level set correction equations (3.22-3.23) to get $\phi^{n,corrected}$; and set $\phi^n \leftarrow \phi^{n,corrected}$

  End

CHAPTER 4

STABILIZED FINITE ELEMENT METHODS FOR SOLVING THE LEVEL SET EQUATION WITH MASS CONSERVATION

Mamadou Kabirou Touré [a], Adil Fahsi [b] and Azzeddine Soulaïmani [c]

[a, b, c] Department of Mechanical Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

This paper has been published (from page 77 to page 105 and from page 114 to 115)

in *International Journal of Computational Fluid Dynamics*, 2016

http://dx.doi.org/10.1080/10618562.2016.1155703

4.1 Introduction

Finite element methods are studied for solving moving interface flow problems using the level set approach and a stabilized variational formulation proposed in ((Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016)), coupled with a level set correction method. The level set correction is intended to enhance the mass conservation satisfaction property. The stabilized variational formulation ((Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016)) constrains the level set function to remain close to the signed distance function, while the mass conservation is a correction step which enforces the mass balance. The eXtended Finite Element Method (XFEM) is used to take into account the discontinuities of the properties within an element. XFEM is applied to solve the Navier-Stokes equations for two-phase flows. The numerical methods are numerically evaluated on several test cases such as time-reversed vortex flow, a rigid-body rotation of Zalesak's disk, sloshing flow in a tank, a dam-break over a bed, and a rising bubble subjected to buoyancy. The numerical results show the importance of satisfying global mass conservation to accurately capture the interface position.

Two-phase flow problems are commonly encountered in nature and in industrial processes. Three methodologies are generally found in the literature to numerically solve such flows, namely the Lagrangian methods, where the mesh is moving (Johnson et Tezduyar, 1997), the

Eulerian methods where the mesh is fixed (Hyman, 1984), and the Eulerian-Lagrangian approach combining the Lagrangian and the Eulerian methods (Hu, Patankar et Zhu, 2001). The Eulerian methods can be divided into two main groups, Front tracking methods and Front-capturing methods. The Front tracking methods include the Volume tracking and the Surface tracking methods (Glimm, 1982; Glimm et McBryan, 1985; Harlow et Welch, 1965; Unverdi et Tryggvason, 1992). Front-capturing methods are commonly used to study two-phase flows, as they naturally account for topological changes. The most common Front-capturing techniques are the Volume Of Fluid (VOF) (Hirt et Nichols, 1981; Noh et Woodward, 1976; Youngs, 1982) and the Level Set methods. Both methods rely on the resolution of a scalar transport equation. For example, the Simple Line Interface Calculation (SLIC) (Noh et Woodward, 1976) and the Piece Linear Interface Calculation (PLIC) (Rider et Kothe, 1998) methods are used to recover the interface in the VOF method. In the level set method introduced by Osher and Sethian (1988), the scalar to be transported is defined as a continuous function corresponding to the signed distance from the interface. Because of the numerical errors that occur when the transport equation of the level set function is solved, several renitialisation processes have been suggested to correct the level set functions so that it remains a distance function (Adalsteinsson et Sethian, 1999; Chunming et al., 2010; Gross et Reusken, 2011; Hysing et Turek, 2005; Olsson et Kreiss, 2005; Olsson, Kreiss et Zahedi, 2007; Osher et Fedkiw, 2003; Qian, Zhang et Zhao, 2007; Sethian, 1996; Sethian, 1998; So, Hu et Adams, 2011; Sussman, Smereka et Osher, 1994; Zhao, 2005). The main problem with the renitialisation process of the level set function is that it might shift the interface according to the frequency of its usage (Gomes et Faugeras, 2000). In the technique proposed in ((Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016)), the transport equation of the level set function is blended with the reinitialisation equation into one single formulation. However, the level set method could lead to significant mass errors. There are many solutions to resolve this defect, such as adding a mass conservation constraint, coupling the level set method to the VOF method, or refining the mesh. Several studies have investigated how to enforce the mass conservation property of the level set method (Ausas, Dari et Buscaglia, 2011; Desjardins, Moureau et Pitsch, 2008; Di Pietro, Lo Forte et Parolini, 2006; Doyeux et al., 2013; Kees et al., 2011; Kuzmin, 2014; Laadhari, Saramito et Misbah, 2010; Le Chenadec et Pitsch, 2013;

Olsson et Kreiss, 2005; Olsson, Kreiss et Zahedi, 2007; Owkes et Desjardins, 2013; Smolianski, 2001; Sussman, 2003; Sussman et Puckett, 2000; van der Pijl et al., 2005; Wang, Simakhina et Sussman, 2012). In this chapter, one of the two phases is constrained to preserve its mass globally (Gross et Reusken, 2011; Smolianski, 2001). The Navier-Stokes equations are discretized in space using a stabilized eXtended Finite Element Method (XFEM) (Chessa et Belytschko, 2003a; Dolbow, Moës et Belytschko, 2001; Fries, 2009; Fries et Belytschko, 2006; 2010; Groß et Reusken, 2007; Menk et Bordas, 2011; Moës et Belytschko, 2002; Rasthofer et al., 2011; Reusken, 2008; Schott et al., 2014) which accounts for solution discontinuities that occur across the interface.

In this study, the stabilized level set variational formulation ((Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016)) and a mass conservation correction algorithm are coupled with a Navier-Stokes solver based on an XFEM formulation (Fahsi, 2016) to solve two-phase flows in two-dimensions. The level set and correction methods are first evaluated in well-established test cases: a time-reversed vortex flow (Rider et Kothe, 1998), a rigid body rotation of Zalesak's disk (Rider et Kothe, 1998; Zalesak, 1979) and a sloshing flow in a cavity (Ubbink, 1997). The numerical evaluations then continue with two-phase flow problems: a collapse of a water column (Martin et Moyce, 1952; Ubbink, 1997), a collapse of a water column with an obstacle (Ubbink, 1997) and a rising bubble including surface tension (Hysing et al., 2009).

The chapter is organized as follows: The stabilized formulations for Navier-Stokes-the level set equations; the XFEM discretization and the mass conservation algorithm are presented in section 4.3. In section 4.4, various flow problems and their numerical results obtained with the different methods are introduced and numerically evaluated. The last section contains the concluding comments.

4.2 Navier-Stokes equations

The conservation of momentum and mass are described for two immiscible-incompressible fluids by the Navier-Stokes equations:

$$\rho(\mathbf{x},t)\left(\frac{\partial \mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u}\right)+\nabla p-\nabla\cdot\boldsymbol{\sigma}(\mathbf{u})=\rho(\mathbf{x},t)\mathbf{g} \tag{4.1}$$

$$\nabla\cdot\mathbf{u}=0 \tag{4.2}$$

in a fixed space-time domain $\Omega\times[0,T], \Omega\subset\mathbb{R}^2$ where ρ is the density, $\mathbf{u}$ is the velocity, p is the pressure, μ is the dynamic viscosity coefficient, $\boldsymbol{\sigma}(\mathbf{u})=\mu(\mathbf{x},t)\left(\frac{\nabla\mathbf{u}+(\nabla\mathbf{u})^T}{2}\right)$ and $\rho\ \mathbf{g}$ is the body force. It is assumed that fluid 1 occupies domain Ω_1 and fluid 2 occupies Ω_2 such that the material interface is $\Gamma_{\text{int}}=\partial\Omega_1\cap\partial\Omega_2$ (Figure 2.1). The density and the dynamic viscosity are discontinuous across the interface:

$$\rho(\mathbf{x},t)=\begin{cases}\rho_1 & \text{for } \mathbf{x}\in\Omega_1(t)\\ \rho_2 & \text{for } \mathbf{x}\in\Omega_2(t)\end{cases} \tag{4.3}$$

and

$$\mu(\mathbf{x},t)=\begin{cases}\mu_1 & \text{for } \mathbf{x}\in\Omega_1(t)\\ \mu_2 & \text{for } \mathbf{x}\in\Omega_2(t)\end{cases} \tag{4.4}$$

Surface tension effects are taken into account by the following interface conditions:

$$[\![\mathbf{u}]\!]=0 \ \text{ and } \ \left[\!\!\left[-p\,\mathbf{I}+\mu(x,t)\left(\frac{\nabla\mathbf{u}+(\nabla\mathbf{u})^T}{2}\right)\right]\!\!\right]\cdot\mathbf{n}=-\gamma\kappa\mathbf{n} \tag{4.5}$$

where $\mathbf{n}$ is the unit normal at the interface Γ_{int} pointing from Ω_1 to Ω_2, $[\![f]\!]=f_{|\Omega_1\cap\Gamma_{\text{int}}}-f_{|\Omega_2\cap\Gamma_{\text{int}}}$ is the jump across the interface of a function f, γ is the surface tension coefficient, κ is the curvature of the interface Γ_{int} and $\mathbf{I}$ is the identity tensor. Two approaches are often used to handle the tension force, either to rewrite it as a volume force:

$$\boldsymbol{f}_{st}=-\gamma\kappa\mathbf{n}\,\delta(\mathbf{x},\Gamma_{\text{int}}) \tag{4.6}$$

where $\delta(\mathbf{x},\Gamma_{\text{int}})$ is the Dirac distribution to the interface, or to introduce the Laplace-Beltrami operator (Bänsch, 2001; Ganesan, 2006). The first approach is adopted in this work.

4.3 Stabilized variational formulations

4.3.1 The flow field

The stabilized variational formulation of the Navier-Stokes equations adopted here is briefly described in the following (Rasthofer et al., 2011; Schott et al., 2014). Assuming suitable finite dimensional spaces for $(\mathbf{u}, p)$ as well as for the weighting functions $(\mathbf{v}, q)$, a variational formulation of equations (4.1) and (4.2) is given as: find $(\mathbf{u}, p)$ such that for all $(\mathbf{v}, q)$:

$$\left(\mathbf{v}, \rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right)\right) - (\nabla\cdot\mathbf{v}, p) + (\nabla\mathbf{v}, \sigma(\mathbf{u})) + \mathrm{d}(\mathbf{v}, \mathbf{u}) = (\mathbf{v}, \rho\mathbf{g}) - (\mathbf{v}, \mathbf{f}_{st}) + (\mathbf{v}, \mathbf{h})_{\partial\Omega} \tag{4.7}$$

$$(q, \nabla\cdot\mathbf{u}) = 0 \tag{4.8}$$

where $(.,.) = (.,.)_{\Omega}$ and $(.,.)_{\partial\Omega}$ denote the L_2 inner product on Ω and $\partial\Omega$ respectively, and $\mathbf{h} = \boldsymbol{\sigma}\cdot\mathbf{n}$. The term $\mathrm{d}(\mathbf{v}, \mathbf{u})$ is added to the standard Galerkin weak form to stabilize the flow field in case of high Reynolds numbers, is given as:

$$\mathrm{d}(\mathbf{v}, \mathbf{u}) = (\mathbf{u}\cdot\nabla\mathbf{v}, \tau_M \Re(\mathbf{u})) + (\nabla\cdot\mathbf{v}, \tau_C \nabla\cdot\mathbf{u})$$

with

$$\Re(\mathbf{u}) = \rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) + \nabla p - \nabla\cdot\sigma(\mathbf{u}) - \rho\mathbf{g} + \mathbf{f}_{st}$$

Note that there is no stabilization term added to the continuity equation since the finite element approximations that are chosen respect the inf-sup stability condition. In particular, we will use the Taylor-Hood P_2-P_1 element enriched by suitable functions to handle discontinuities inside the elements. The stabilization parameters τ_M and τ_C are given by:

$$\tau_M = \frac{h^e}{2\|\mathbf{u}\|} \text{ and } \tau_C = \frac{\left(\rho h^e \|\mathbf{u}\|\right)}{2},$$

where h^e is the element size, defined as the minimum distance between two nodes.

4.3.2 Extended Finite Element Method (XFEM)

Because of the viscosity and density differences between the phases across the interface, the velocity has a discontinuous gradient and the pressure is discontinuous (for non-zero surface tension) or has a discontinuous gradient (for zero surface tension). When the mesh is not aligned with the interface, the classical polynomial finite element approximations may not resolve such discontinuities accurately. The extended finite element (XFEM) approximations can account for these discontinuities by enriching the approximation spaces. The XFEM incorporates enrichment functions that reproduce the discontinuity inside the cut with additional degrees of freedom. However, it is fundamental that the enrichment functions be chosen judiciously (Sauerland et Fries, 2011). The works of Coppola and Owen (2009) and Sauerland et al (2013) show that it is not advantageous to enrich the velocity field in case of driven gravity flows, as serious instability problems may be encountered. Therefore, only the pressure space is enriched with the shifted sign enrichment, leading to the following approximation for the pressure:

$$p(\mathbf{x},t)^{XFEM} = \sum_{i\in I} N_i^p(\mathbf{x})\, p_i(t) + \sum_{i\in I^*} N_i^p(\mathbf{x}) \left(\psi_{sign}(\mathbf{x},t) - \psi_{sign}(\mathbf{x}_i,t)\right) a_i \tag{4.9}$$

where $N_i^p(\mathbf{x})$ is the standard linear shape function for node i, I is the set of all nodes in the domain Ω, p_i are the nodal values of the pressure p, a_i are the additional XFEM degrees of freedom, I^* is the nodal subset of the enrichment and $\psi(\mathbf{x},t)$ is the global enrichment function defined as:

$$\psi_{sign}(x,t) = \operatorname{sign}\left(\phi(x,t)\right) = \begin{cases} -1 & \text{if } \phi(\mathbf{x},t) < 0 \\ 0 & \text{if } \phi(\mathbf{x},t) = 0 \\ 1 & \text{if } \phi(\mathbf{x},t) > 0 \end{cases} \tag{4.10}$$

The approximation for the velocity space is given with classical finite element methods (FE) as:

$$\mathbf{u}(\mathbf{x},t) = \sum_{i\in I} N_i^{\mathbf{u}}(\mathbf{x})\, \mathbf{u}_i(t) \tag{4.11}$$

where $N_i^u(\mathbf{x})$ is the standard quadratic FE shape function for node i and $\mathbf{u}_i$ are the nodal

values.

Remark: In general, if we start with a stable velocity-pressure finite element pair, such as the Taylor-Hood element, it is not guaranteed that the XFEM spaces will still be stable after enrichment. We have performed extensive numerical experiments on the choice of velocity and pressure enrichments, and we have concluded that the enrichment of the velocity often leads to instabilities, while the pressure enrichment for the Taylor-Hood element is stable (provided that some care is taken for the cut elements where the interface is very close to an enriched node, see (Gross et Reusken, 2011). This conclusion is also shared by other researchers (Sauerland et Fries, 2011). To our knowledge, there is no proof for the inf-sup stability of the (P_2-P_1) pressure-enriched element, but only partial theoretical results are reported in (Gross et Reusken, 2011). However, in (Schott et al., 2014) a hybrid Nitsche-XFEM formulation is developed where jump enrichments for both velocity and pressure are applied. In this formulation, stabilization terms are introduced to prevent convective, inf-sup, and incompressibility instabilities. A mathematical analysis of the inf-sup stability is provided in (Massing et al., 2014).

4.3.3 The level set field

We have developed a stabilized variational method ((Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016)) for the level set equation without staggered re-initialization. The proposed method transforms the level set equation (2.6) into:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi - \nabla \cdot \left(\lambda \, K \, \nabla \phi \right) = 0 \tag{4.12}$$

where λ is a parameter defined by $\lambda = \beta \dfrac{\|\mathbf{u}\| \left(h^e \right)^2}{2 \left| \nabla \phi \right|}$ and $K = \dfrac{1}{\left| \Omega^e \right|} \int\limits_{\Omega^e} \left(\left| \nabla \phi \right| - 1 \right) d\Omega$.

Li et al. (Li et al., 2005) stated that equation (4.12) is a transport-diffusion equation when $K > 0$, i.e. making ϕ smoother and therefore reducing the gradient $\left| \nabla \phi \right|$, and that it is an anti-diffusion-transport equation when $K < 0$ with an increases in the gradient. This method is commonly used in imagery processing but has not been evaluated for computational fluid

dynamics (CFD) applications. Equation (4.12) is usually solved using the finite difference method.

The parameter β is a parameter close to unity according to a sensitivity analysis ((Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016)). The weak form of the formulation (4.12) using the stabilized Streamline-Upwind-Petrov-Galerkin (SUPG) method reads: find ϕ such that for all weighting function ψ

$$\int_{\Omega}\left[(\psi+\tau_{\phi}\,\mathbf{u}\cdot\nabla\psi)\left(\frac{\partial\phi}{\partial t}+\mathbf{u}\cdot\nabla\phi\right)+\lambda\,K\left(\nabla\phi\cdot\nabla\psi\right)\right]d\Omega=0 \tag{4.13}$$

with $\tau_{\phi}=\alpha\dfrac{h^{e}}{2\|\mathbf{u}\|}$ where $0\le\alpha\le 1$.

For $\alpha=\beta=0$, equation (4.13) would designate the standard Galerkin formulation. For $\alpha=1$ and $\beta=0$, we obtain the stabilized SUPG formulation. For $\alpha\neq 0$ and $\beta\neq 0$, equation (4.13) will lead to the stabilized variational formulation introduced in (Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016). The approximation for the level set function is given by:

$$\phi(\mathbf{x},t)=\sum_{i\in I}N_{i}^{\phi}(\mathbf{x})\,\phi_{i}(t) \tag{4.14}$$

where $N_{i}^{\phi}(\mathbf{x})$ are the quadratic Lagrange polynomial on the triangle, and ϕ_i are the nodal values of the level set function ϕ.

To solve (4.13) the initial solution is $\phi(x,0)=\phi_0$, in which ϕ_0 is an approximate signed distance to the interface $\Gamma(0)$ and should be configured such that $\left\{x\in\mathbb{R}^{2}:\ \phi_o(x)=0\right\}=\Gamma(0)$. To make the problem of the linear hyperbolic level set equation (4.13) well-posed, a boundary condition is needed at the inflow part of the domain $\partial\Omega_{in}=\left\{x\in\partial\Omega:\ \mathbf{u}\cdot\mathbf{n}<0\right\}$. As noted in (Gross et Reusken, 2011), there are no natural physics-based boundary conditions for ϕ at the inflow boundary; we are only interested in values of ϕ that are close to the interface. For a closed domain with a solid boundary $\partial\Omega$, we have $\mathbf{u}\cdot\mathbf{n}=0$ along $\partial\Omega$; ϕ is then defined up to a constant. An implicit condition is provided, however, by the variational formulation (4.13), i.e.

$\int_{\Omega} \frac{\partial \phi}{\partial t}\, d\Omega = 0$, since the function $\psi = 1$ is in the weighting finite element space. Therefore, the constant is implicitly defined by $\int_{\Omega} \phi(x,t)\, d\Omega = \int_{\Omega} \phi(x,0)\, d\Omega$.

4.3.4 Mass conservation

The proposed variational method does not accurately maintain the total mass, as a loss of mass (or gain) is usually observed in numerical calculations. This is due to the numerical errors accumulated after several steps, and to the size of the elements that may be too large to precisely represent the topology of the interface. The mass conservation can be enforced by different methods, such as combining the level set method with the VOF method (Di Pietro, Lo Forte et Parolini, 2006; Olsson et Kreiss, 2005; Olsson, Kreiss et Zahedi, 2007; Sussman, 2003; van der Pijl et al., 2005) or by shifting the zero level set while keeping the shape of the interface with a level-set correction step (Smolianski, 2001) or by using a mass conservation constraint (Gross et Reusken, 2011).

For a closed domain, the total mass is obviously conserved. However, the volumes of each phase are not necessary conserved; therefore, a simple correction of the level set function can be adopted. Let $V_1(\phi,t)$ be the volume of the subdomain Ω_1:

$$V_1(\phi,t) = \int_{\Omega_1} d\Omega \tag{4.15}$$

We need to find an unknown constant δ such as

$$V_1(\phi+\delta,t) = V_1(\phi,0) \tag{4.16}$$

Equation (4.16) is nonlinear with respect to δ, requiring an iterative numerical method; at convergence, the mass-corrected level set $\phi^{corrected}$ is set as:

$$\phi^{corrected} = \phi + \delta \tag{4.17}$$

In the case of an open domain, with a nonzero net mass flow, the conservation equation (4.16) should be modified accordingly; however, the numerical procedure can still be used to correct the level.

Cranck-Nicholson's algorithm is used to discretize the Navier-Stokes and level set equations in time. The general solution algorithm is summarized in the following:

- Set $\left(\mathbf{u}^n, p^n, \phi^n\right)$ with $\left(\mathbf{u}(\mathbf{x},0), p(\mathbf{x},0), \phi(\mathbf{x},0)\right)$
- Loop over time steps:
  For $n = 1 : Nsteps$
  - Solve the Navier-Stokes equations (4.7) and (4.8) to get $\left(\mathbf{u}^n, p^n\right)$
  - Solve the level set equation (4.13) to get ϕ^n
  - Solve the mass correction equations (4.16) and (4.17) to get $\phi^{n,corrected}$

  End For

Newton's algorithm is used to solve the Navier-Stokes, the level set and the mass correction equations.

4.4 Numerical tests

The following numerical tests are performed to study the accuracy of the proposed methods. In the first tests, the time-reversed vortex (Rider et Kothe, 1998) and the rigid body rotation of Zalesak's disk (Rider et Kothe, 1998), the velocity is imposed. In the second series of tests, the velocity field is computed by solving the incompressible Navier-Stokes equations for the sloshing flow in a cavity, for the collapse of a water column (Martin et Moyce, 1952; Ubbink, 1997) and for the collapse of a water column with an obstacle (Ubbink, 1997). The Navier-Stokes equations are solved using the triangular P_2-P_1 element with pressure enrichment for the cut elements. For the test problems considered (time-reversed vortex and rigid body rotation of Zalesak's disk), the solution is periodic. To analyse the accuracy of the interface location and of the mass conservation of the phases, the following norms and relative errors are specified:

$$E_1 = \sqrt{\int_\Omega \left(\phi(\mathbf{x},0)_{exact} - \phi(\mathbf{x},T)\right)^2 d\Omega} \tag{4.18}$$

$$E_2 = \sqrt{\int_\Omega \left(\phi(\mathbf{x},0)_{exact} - \phi(\mathbf{x},T)\right)^2 d\Omega} \Big/ \sqrt{\int_\Omega \phi^2(\mathbf{x},0)_{exact}\, d\Omega} \tag{4.19}$$

$$Mass(\%)=100\left(\frac{\text{area}(t)}{\text{area}(t=0)}\right) \tag{4.20}$$

where $\phi(\mathbf{x},0)_{exact}$ is the exact solution at the initial time (or at $t=T$) and $\phi(\mathbf{x},T)$ is the numerical solution computed at $t=T$. For the Navier-Stokes problems (described in subsections 4.4.3, 4.4.4 and 4.4.5), only the mass conservation is evaluated because the exact interface is unknown.

4.4.1 Time-reversed vortex flow

This is a well-known and documented case (Rider et Kothe, 1998) to evaluate level set methods. This test problem was first introduced in (Bell, Colella et Glaz, 1989) to test the ability of the numerical scheme to resolve thin filaments. These filaments occur when a disk is submitted to a stretching flow field defined by the single vortex stream function:

$$\Psi_{single\ vortex}=\frac{1}{\pi}sin^2(\pi x)sin^2(\pi y) \tag{4.21}$$

The time-reversed vortex flow stream function is obtained by multiplying the single vortex stream function (4.21) by $cos\left(\frac{\pi t}{T}\right)$ where T is the period. The time-reversed vortex flow stream function is thus:

$$\Psi_{time-reversed\ vortex}=\frac{1}{\pi}sin^2(\pi x)sin^2(\pi y)cos\left(\frac{\pi t}{T}\right) \tag{4.22}$$

The velocity is derived from the stream function:

$$u=-\frac{\partial\Psi}{\partial y}\ \text{and}\ v=\frac{\partial\Psi}{\partial x}\ , \tag{4.23}$$

so that,

$$u_{time-reversed\ vortex}=-2sin^2(\pi x)cos(\pi y)sin(\pi y)cos\left(\frac{\pi t}{T}\right) \tag{4.24}$$

and

$$v_{time-reversed\ vortex} = 2cos(\pi x)sin(\pi x)sin^2(\pi y)cos\left(\frac{\pi t}{T}\right) \tag{4.25}$$

The computational domain is a unit square. One of the phases of the domain is a disk centred on the coordinates (0.5, 0.75); this disk has a radius of 0.15. The computational time simulation is $T = 8s$. The flow reverses at $t = T/2$. The spatial discretization is done with triangular elements of a 125×125 grid resulting in 31,250 elements and 63K nodes. For the temporal discretization, the time step Δt is 0.008 s. The results obtained with the same mesh using the SUPG method $(\alpha = 1, \beta = 0)$ and the stabilized method $(\alpha = 1, \beta = 1)$ with and without mass conservation constraint are shown in Figure 4.1(a) to Figure 4.1(f). In Figure 4.1(e), the disk does not return perfectly to its original position for the three methods; its final interface position is slightly altered for each of the methods. The disk mass loss $Mass(\%)$ results are shown in Figure 4.2 and Table 4.1. The trailing edge of the spiral originated from the disk phase is discontinuous, as shown in Figures 4.1(b), 4.1(c) and 4.1(d), corresponding to the instants $T/4$, $T/2$, and $3T/4$, respectively.

The stabilized method with mass conservation clearly offers the best results for the mass loss $Mass(\%)$. The error norms for the different cases are presented in Table 4.2, where the global errors are shown to be minimal when the SUPG method is used. However, they are reduced for the stabilized method if the mass conservation correction step is used.

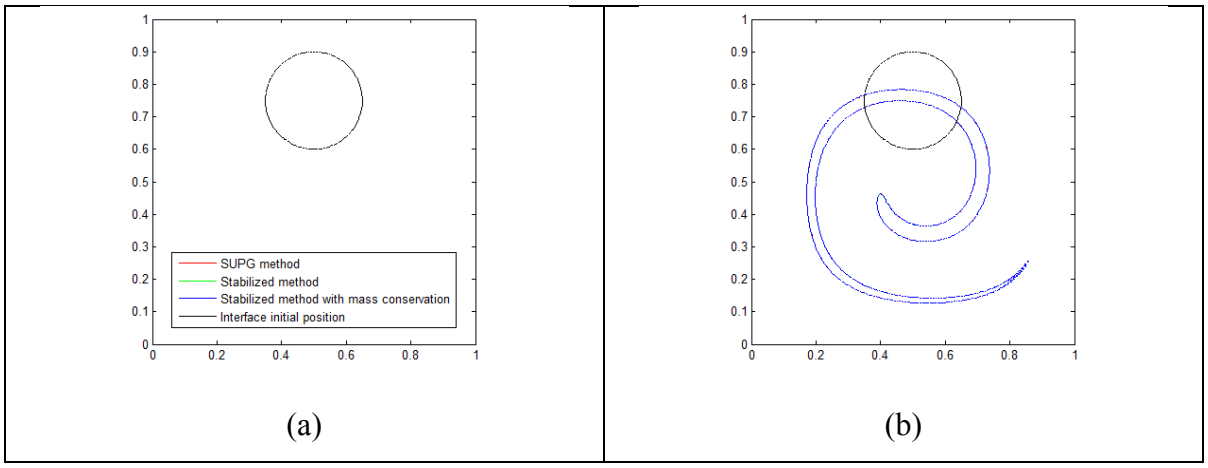
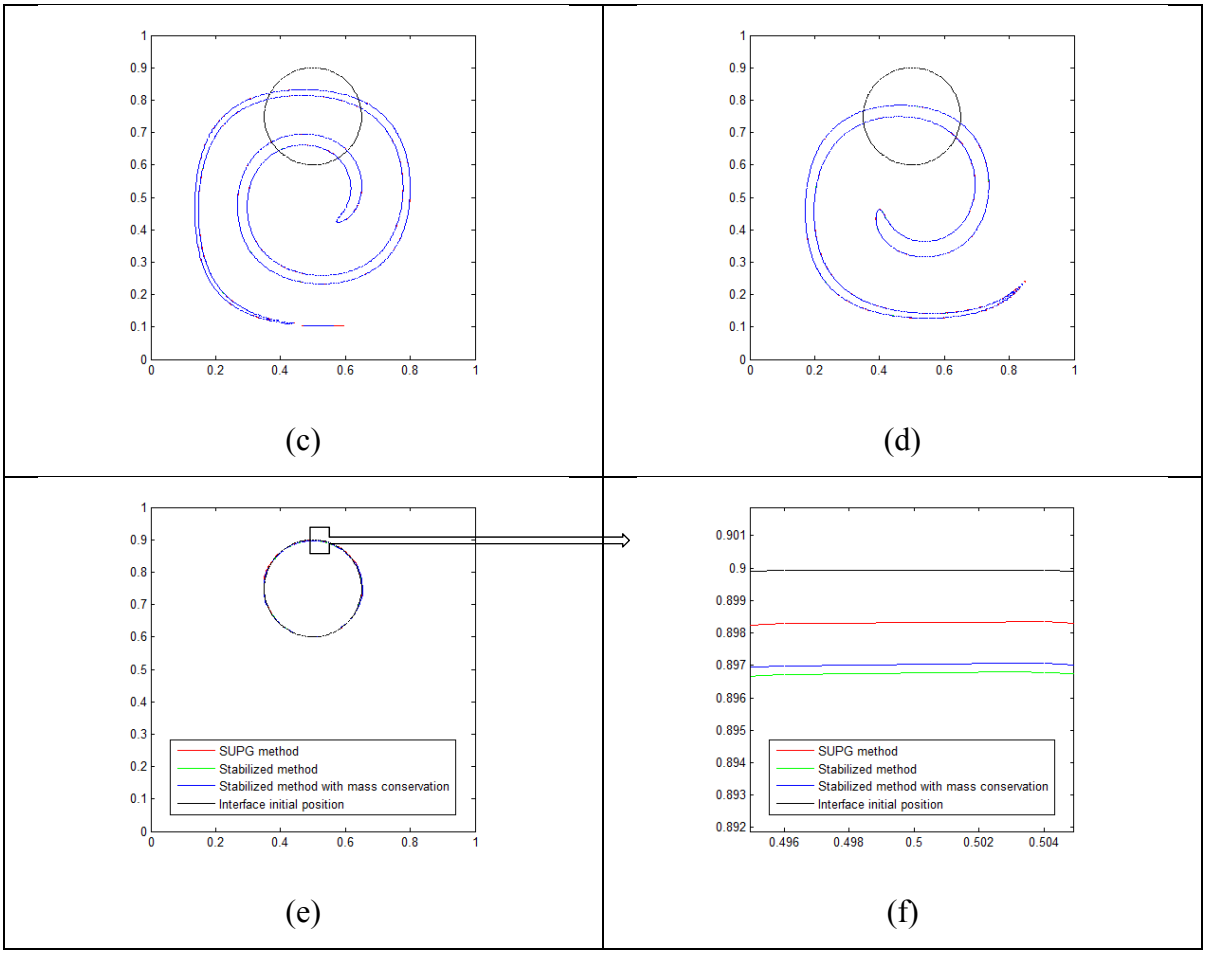


Figure 4.1: Time-reversed vortex flow. Interface positions at $t = 0(a), T/4(b), T/2(c), 3T/4(d)$ and $T(e)$. Closer view of the interface at $T(f)$

Figure 4.1 continuation



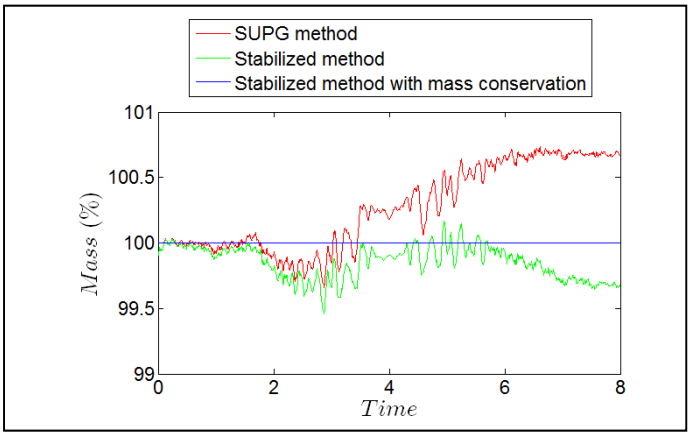


Figure 4.2: Time-reversed vortex flow. Percentage of disk area $Mass(\%)$ versus time.

Table 4.1: Time-reversed vortex flow.
Percentage of the disk mass loss at $t = T$ for $T = 8$.

Method	Mass (%)
Case 1: $\alpha = 1$ and $\beta = 0$ SUPG method	100.666
Case 2: $\alpha = 1$ and $\beta = 1$ Stabilized method	99.666
Case 3: $\alpha = 1$ and $\beta = 1$ Stabilized method with mass conservation	100.000

Table 4.2: Time-reversed vortex flow.
Error norms E_1 and E_2 at $t = T$.

Method	E_1	E_2
Case 1: $\alpha = 1$ and $\beta = 0$ SUPG method	1.194E-3	3.440E-3
Case 2: $\alpha = 1$ and $\beta = 1$ Stabilized method	2.098E-3	6.044E-3
Case 3: $\alpha = 1$ and $\beta = 1$ Stabilized method with mass conservation	1.294E-3	3.729E-3

4.4.2 Rigid body rotation of Zalesak's disk

This test is a well-known and highly documented application of the level set method. The aim of this test is to examine the mass loss and the error norms due to a non-deforming interface and a rotating flow field. The domain is a circle of a unit diameter. One of the phases is a disk of 0.15 radius and its centre is located at (0.5, 0.75). The disk has a small vertical slot which has a height of 0.25 and a width of 0.05. The slot makes the test difficult because of the mass loss/gain at its corners. The x and y components' flow field velocity are:

$$u = \frac{\pi}{314}(0.5 - y) \text{ and } v = \frac{\pi}{314}(x - 0.5) \tag{4.26}$$

The spatial discretization is done with 28,732 triangular elements and 57,857 nodes. The simulation is carried out with a period $T = 6.28$ and a time step $\Delta t = 6.28\ 10^{-3}$ s. The results obtained for the different methods are shown in Figures 4.3(a) and 4.3(b). At the end of the

simulation, the sharp edges are smoothed. A dense mesh would help to preserve the sharpness of the corners, but at a cost of an increase in the computational time. The values of $Mass(\%)$ for the different cases are shown in Figure 4.4 and Table 4.3. The values of the error norms are presented in Table 4.4 for the different methods. The error norms are almost equivalent for all three methods.

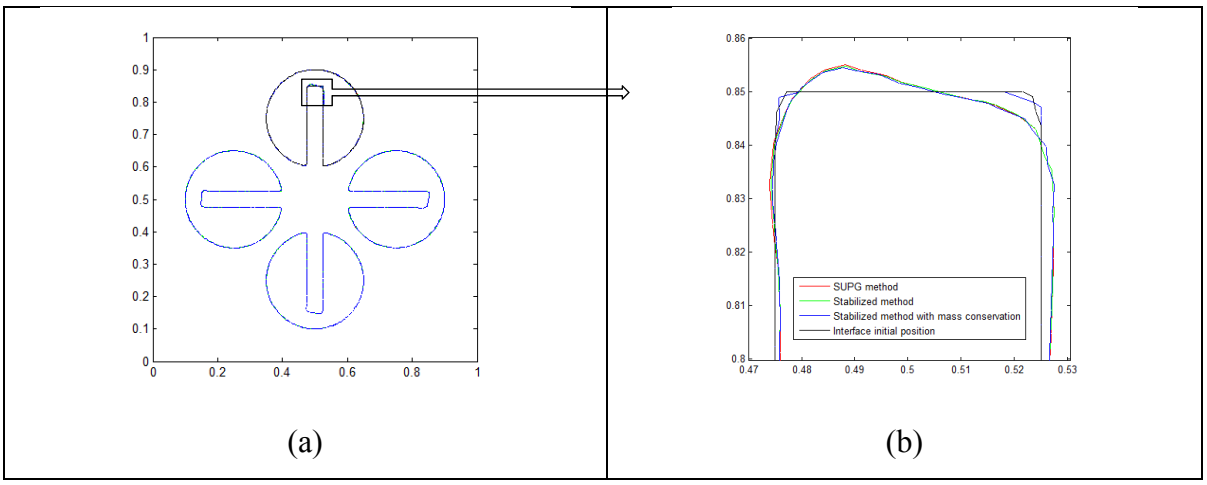


Figure 4.3: Rigid body rotation of Zalesak's disk. Interface positions at $t = 0, T/4, T/2, 3T/4$ and T (a). Closer view of the interface at T (b)

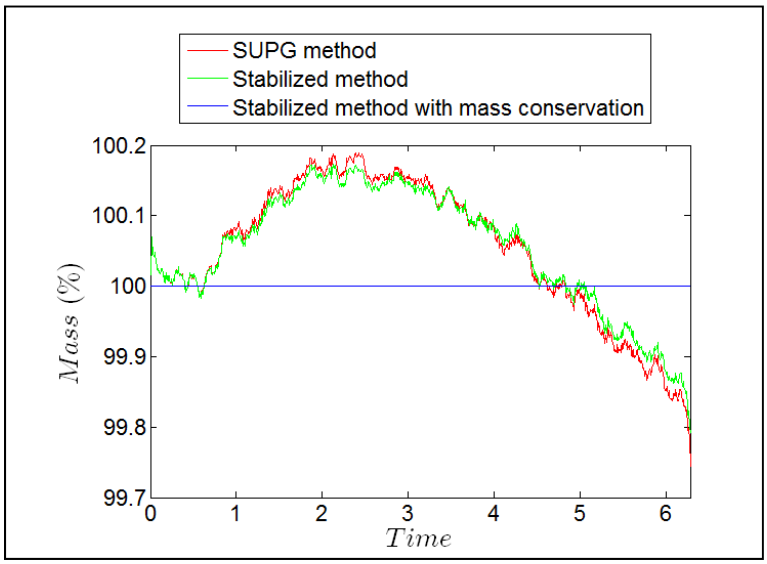


Figure 4.4: Rigid body rotation of Zalesak's disk. Evolution history of the $Mass(\%)$.

Table 4.3: Rigid body rotation of Zalesak's disk.
Percentage of the disk mass loss at $t = T$ for $T = 6.28$.

Method	Mass (%)
Case 1: $\alpha = 1$ and $\beta = 0$ SUPG method	99.743
Case 2: $\alpha = 1$ and $\beta = 1$ Stabilized method	99.790
Case 3: $\alpha = 1$ and $\beta = 1$ Stabilized method with mass conservation	100.000

Table 4.4: Rigid body rotation of Zalesak's disk.
Error norms E_1 and E_2 at $t = T$.

Method	E_1	E_2
Case 1: $\alpha = 1$ and $\beta = 0$ SUPG method	4.639E-4	1.733E-3
Case 2: $\alpha = 1$ and $\beta = 1$ Stabilized method	4.619E-4	1.725E-3
Case 3: $\alpha = 1$ and $\beta = 1$ Stabilized method with mass conservation	4.721E-4	1.763E-3

Table 4.4 shows that the error norms are almost equivalent for all of the methods, because of the refinement of the mesh. The sharp corners of the slot are smoothed with the simulation.

4.4.3 Sloshing flow in a cavity

This test is performed in order to assess the ability of the proposed methods to simulate the large sloshing motion of a liquid-air interface contained in a closed tank. The Navier-Stokes equations are solved for a two-phase flow. A body force is imposed with $\mathbf{g} = (g_x, -\mathrm{g})$ where $g_x = 0.25\, g\, \sin\left(2\,\pi\, t/T\right), g = 9.81\,\mathrm{m/s}$ and the period T is equal to 2 s. The exact solution of the interface position is unknown, therefore only the mass conservation is verified. The entire domain is a rectangle of dimensions 0.584 m by 0.365 m. The lower subdomain has a height of 0.2 m, and is filled with a fluid phase of density 998 kg/m^3 and dynamic viscosity of 0.1 Pa·s. The upper subdomain has dimensions of 0.584 m by 0.165 m, filled with a fluid phase

of density 1.24 kg/m^3 and dynamic viscosity of 0.01 Pa·s. The slip condition is applied to the right, left and bottom solid boundaries. The top boundary is freely open to the atmosphere. The simulation is performed for a total time of 3s with a time step Δt of 0.005 s. The mesh has 1,804 elements and 3,723 nodes. The mesh and the interface at the initial instant are shown in Figure 4.5.

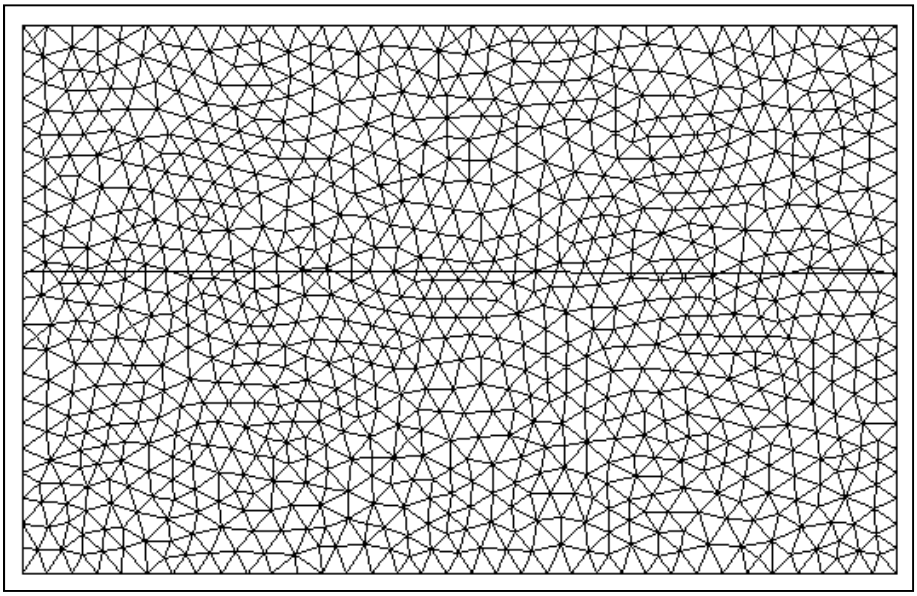


Figure 4.5: Sloshing flow in a cavity. Initial interface at t = 0 s

The results of the interface position, with the mass loss obtained for the stabilized variational methods with and without a mass conservation constraint are shown in Figure 4.6(a) to Figure 4.6(g). The values of the parameters are represented as explained in Legend 6:

Case 1: ——— $\beta_1 = 0$ and $\beta_2 = 1$

Case 2: - - - - - - - $\beta_1 = 0$ and $\beta_2 = 1$ with the mass conservation

Legend 6

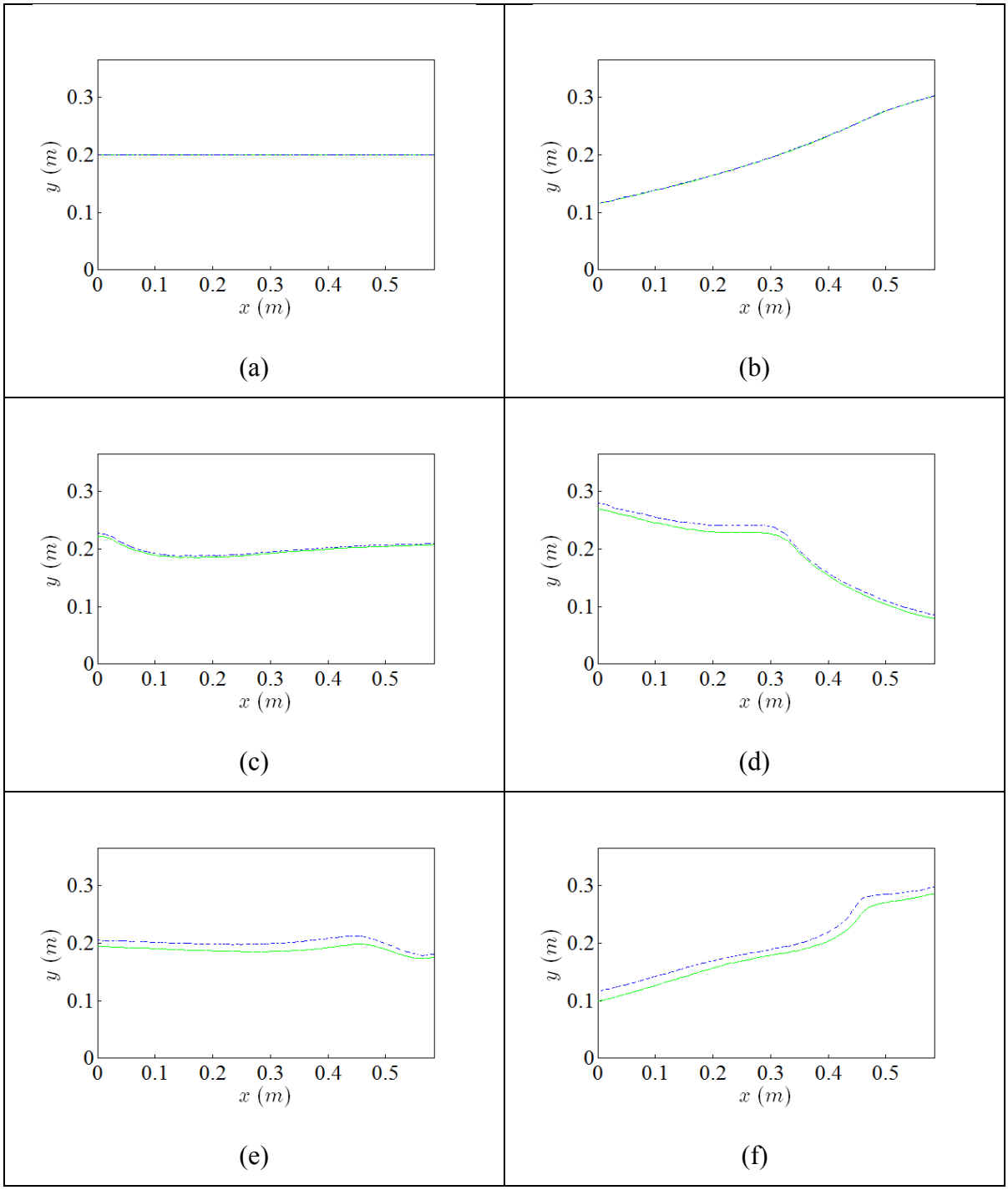
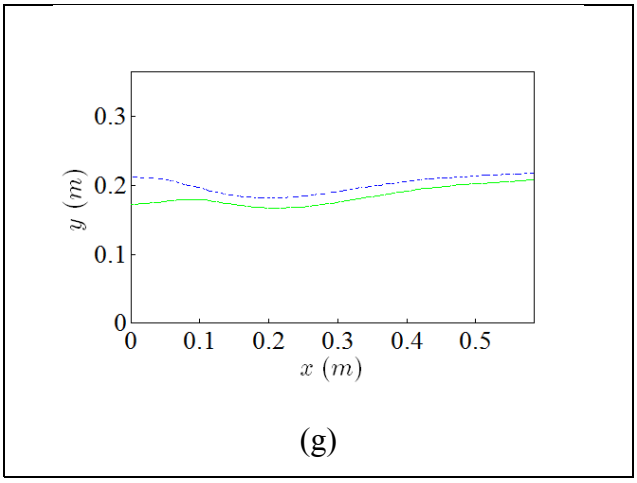


Figure 4.6: Sloshing flow in a cavity. Interface positions at $t = 0s\,(a), 0.5s\,(b), 1s\,(c), 1.5s\,(d), 2s\,(e), 2.5s\,(f)$ and $3s\,(g)$.

Figure 4.6 continuation



(g)

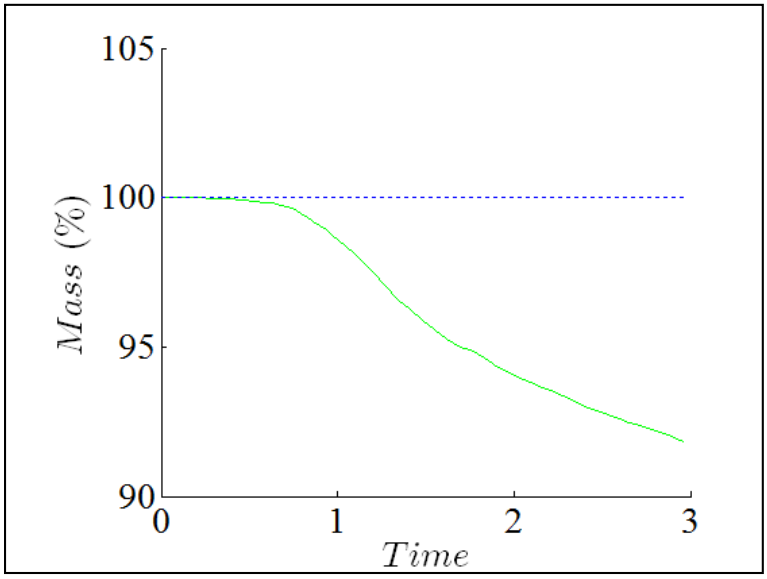
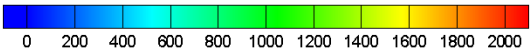


Figure 4.7: Sloshing flow in a cavity.
The mass loss of the lower subdomain.

In Figure 4.6(a) to Figure 4.6(g), the stabilized methods give (virtually) the same wave forms both with and without mass conservation constraint. However, the mass loss after 3s of the stabilized variational method is 8.17 % (as shown in Figure 4.7), which is unacceptable. When a mass conservation constraint is applied to the stabilized variational method, the mass is

conserved up to the sixth digit. The results of the zero level set, the pressure field and the velocity vectors of the stabilized variational method with mass conservation are presented in Figure 4.8(a) to Figure 4.8(f). The colour map of the pressure contours is shown in Legend 7. The velocity vectors in Figure 4.8(a) to Figure 4.8(f) show strong moving vortices. The strength of the vortices is higher for the lighter phase.



Legend 7: Pressure

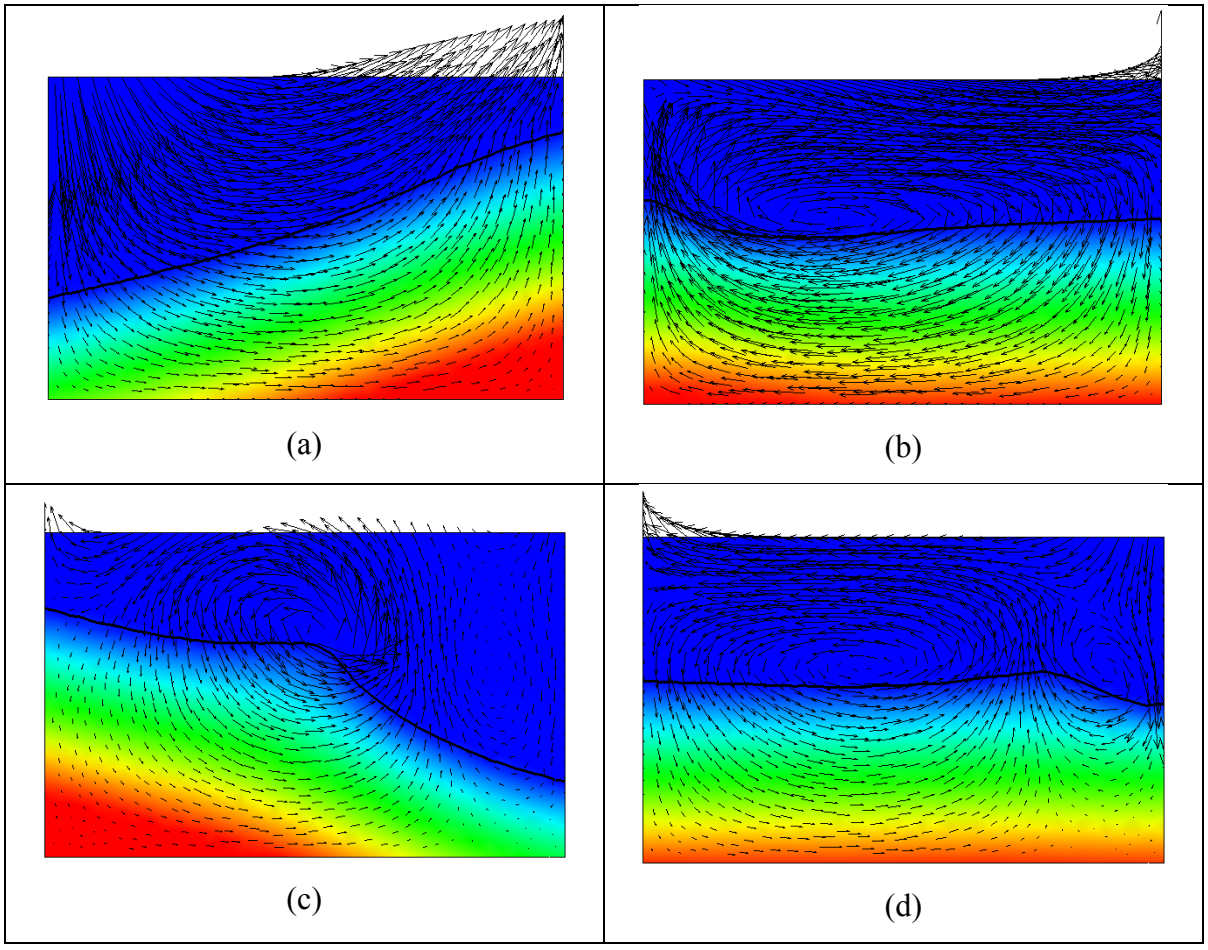
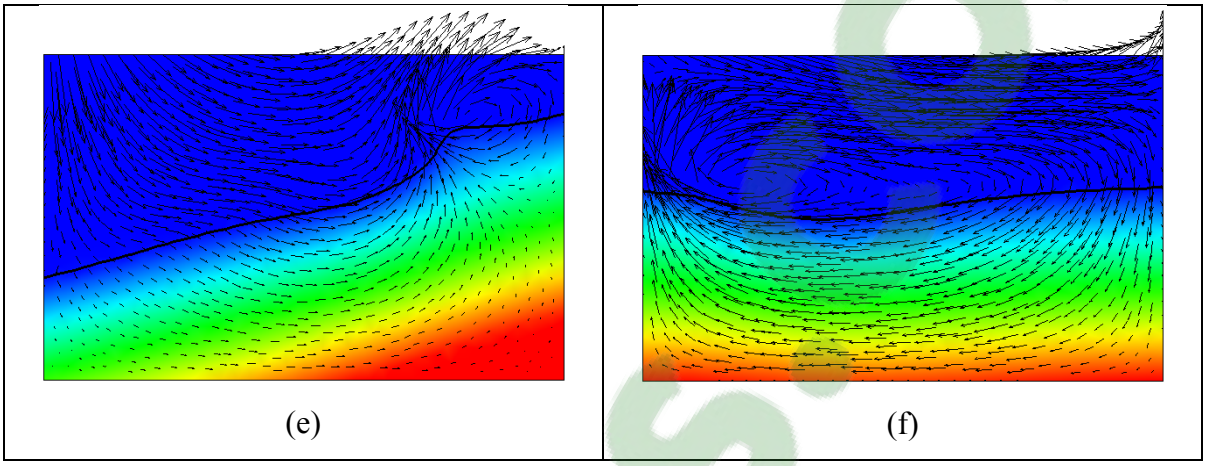


Figure 4.8: Sloshing flow in a cavity, with the pressure at $t = 0.5s(a), 1s\ (b), 1.5s\ (c), 2s\ (d), 2.5s\ (e)$ and $3s\ (f)$.

Figure 4.8 continuation



(e) (f)

4.4.4 Dam break flow

The dam-break problem is a challenging test for numerical methods, as it encompasses large interface deformations. It is a well-known problem, widely used to validate free-surface solvers since it has experimental results ((Martin et Moyce, 1952) and (Hu et Sueyoshi, 2010)). A column of water is initially sustained by a dam which is suddenly removed. The water collapses under the influence of gravity ($g = 9.81\,\mathrm{m/s^2}$), and flows downward until it hits the opposite wall. The computational domain is a rectangle with dimensions of 0.584 m by 0.365 m. The initial interface separates the domain into two regions. The lower and left side subdomain is a rectangle of $0.146\,\mathrm{m} \times 0.292\,\mathrm{m}$, filled with water with a density of 998.2 kg/m^3 and a dynamic viscosity of 0.1Pa·s. The upper fluid has density 1.204 kg/m^3 and a dynamic viscosity of 0.01Pa·s. The slip condition is applied to all boundaries. The simulation is conducted for a total time of 1 s with a time step Δt of $10^{-3} s$. The unstructured mesh has 1,804 elements and 3,723 nodes. The mesh and the interface at the initial instant are shown in Figure 4.9. At the initial time $t = 0$, the water column is defined such that its front position z is a and its height η is $2a$ with $a = 0.146\,\mathrm{m}$.

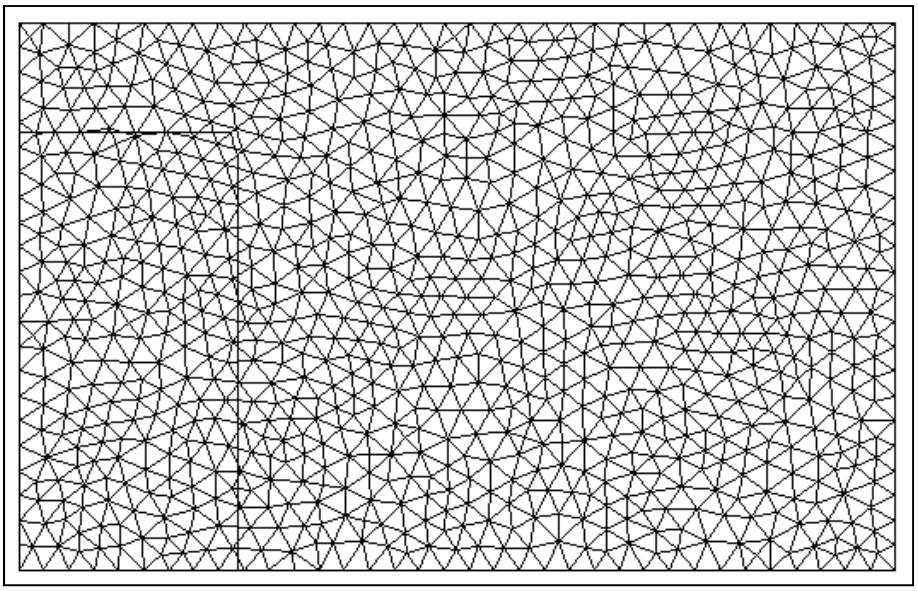


Figure 4.9: Dam-break. Mesh and initial interface at t = 0 s.

The flow can be briefly described as follows: in a first stage from the initial time up to $t = 0.275\ s$ the water column collapses, maintaining a compact domain, and then it violently hits the right wall. The fluid then rises along the wall and, under the effect of gravity, collapses again, creating a breaking wave. This wave propagates to the left of the field by trapping an air bubble, and continues on its way to hit the left wall and bounce back. Experimental data comprises the positions of the fluid front on the bed and on the left vertical wall, as well as pictures of the flow at certain times in the experiment.

The comparison is conducted between the SUPG methods with and without mass conservation constraint. The values of the parameters used are represented as described in Legend 8. Figure 4.10(a) to Figure 4.10(f) illustrate the interface at different times.

Case 1: —— Stabilized method $(\alpha = 1, \beta = 1)$

Case 2: - - - - - - - Stabilized method $(\alpha = 1, \beta = 1)$ with the mass conservation restraint

Legend 8

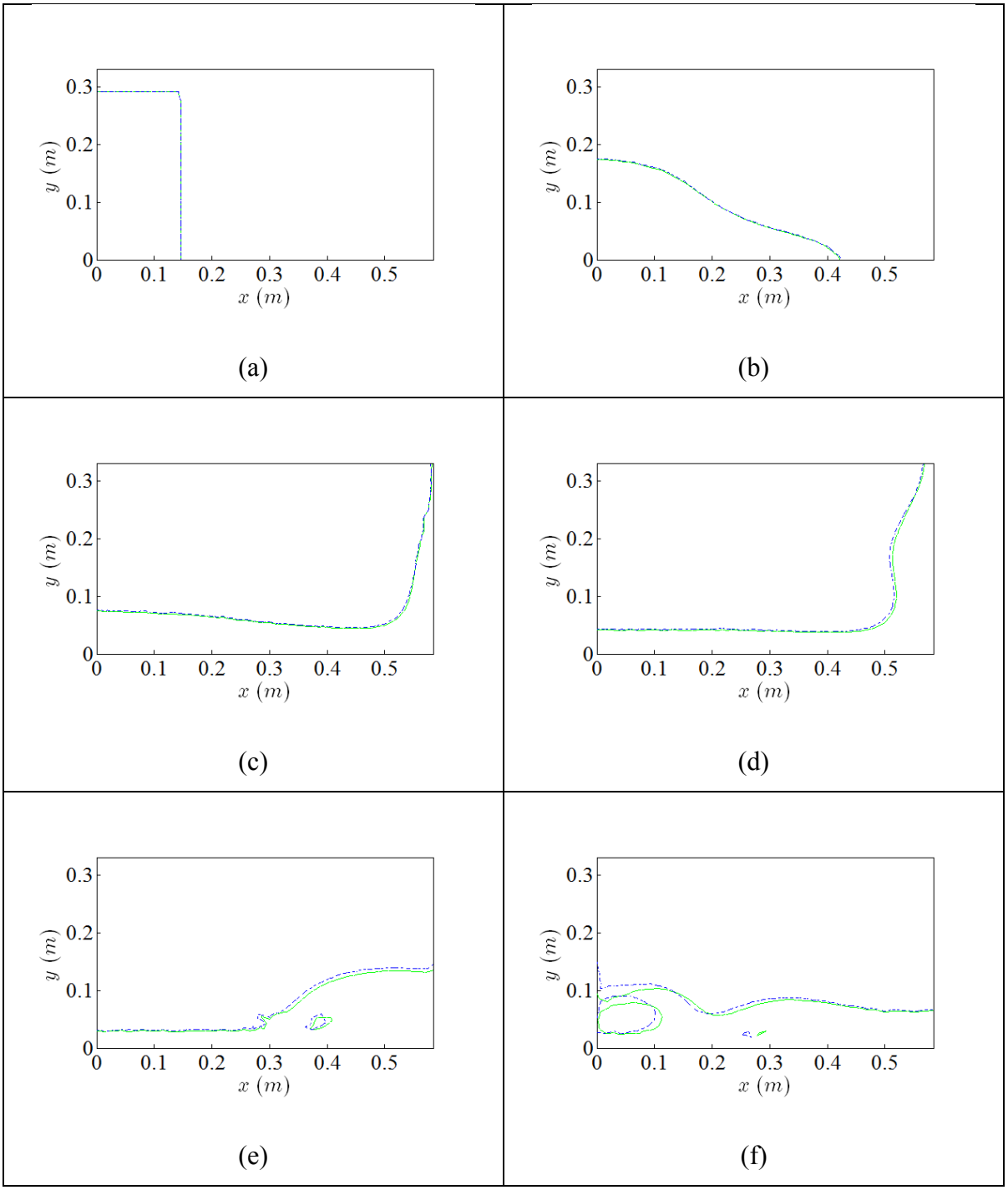


Figure 4.10: Dam-break. Interface positions at
$t = 0s\,(a), 0.2s\,(b), 0.4s\,(c), 0.6s\,(d), 0.8s\,(e)$ and $1s\,(f)$.

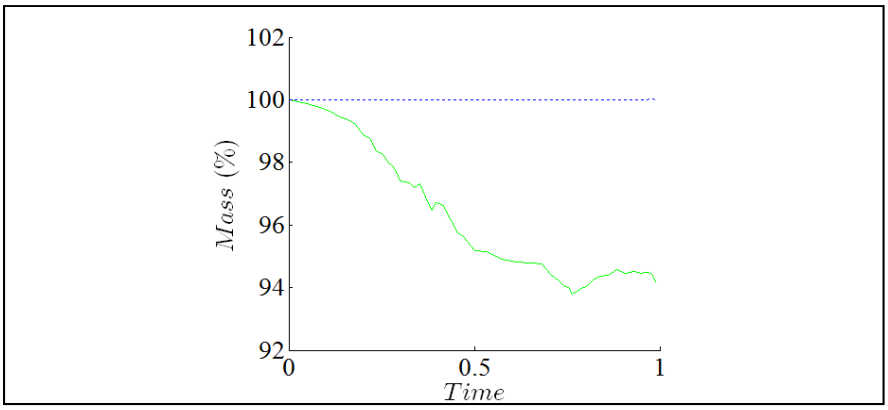


Figure 4.11: Dam-break. The mass loss of the lower subdomain.

The results of the non-dimensional quantities $\tilde{Z} = z/a$ and $\tilde{H} = \eta/(2\,a)$ versus $\tilde{T} = t/\sqrt{2g/a}$ and $\tilde{\tau} = t/\sqrt{g/a}$, respectively, are shown in Figures 12(a,b) for the two methods and the experimental data of Martin and Moyce (Martin et Moyce, 1952). The non-dimensional quantities are defined with the following variables: z is the front position of the water column on the bed, η is the height position of the water column on the left boundary, g is the acceleration due to gravity, t is the time and $a = 0.146$ m.

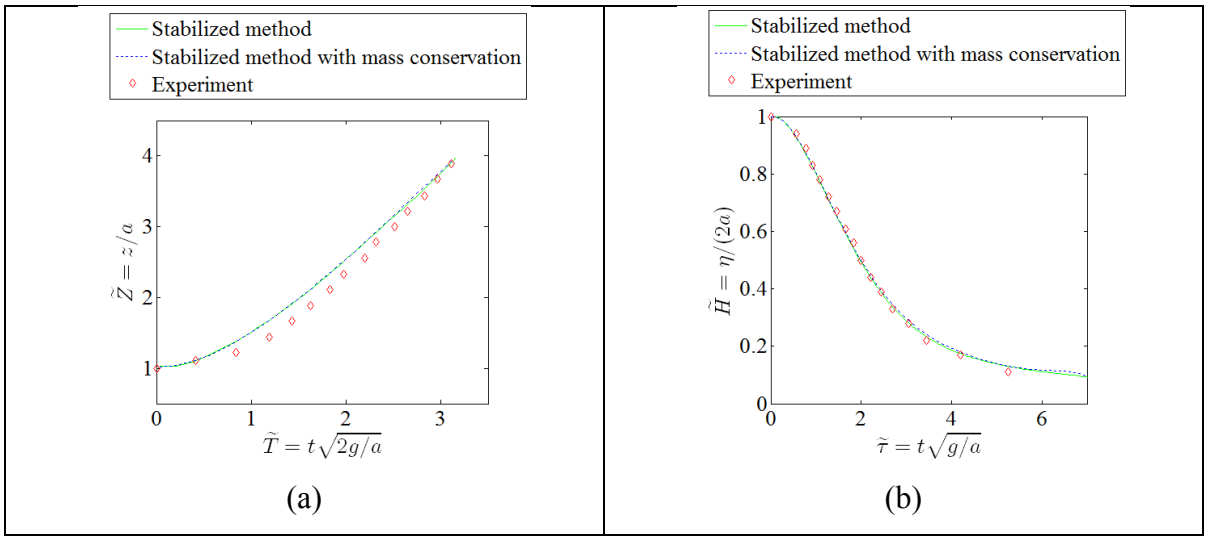
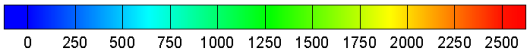


Figure 4.12: Dam-break. Position of the water column front on the bed (a) and the height of the water column on the left boundary (b).

The zero level set, the pressure field and the velocity vectors of the stabilized variational method with the mass conservation results are shown in Figure 4.13(a) to Figure 4.13(e). The colour map of the pressure contours is given in Legend 9.



Legend 9: Pressure

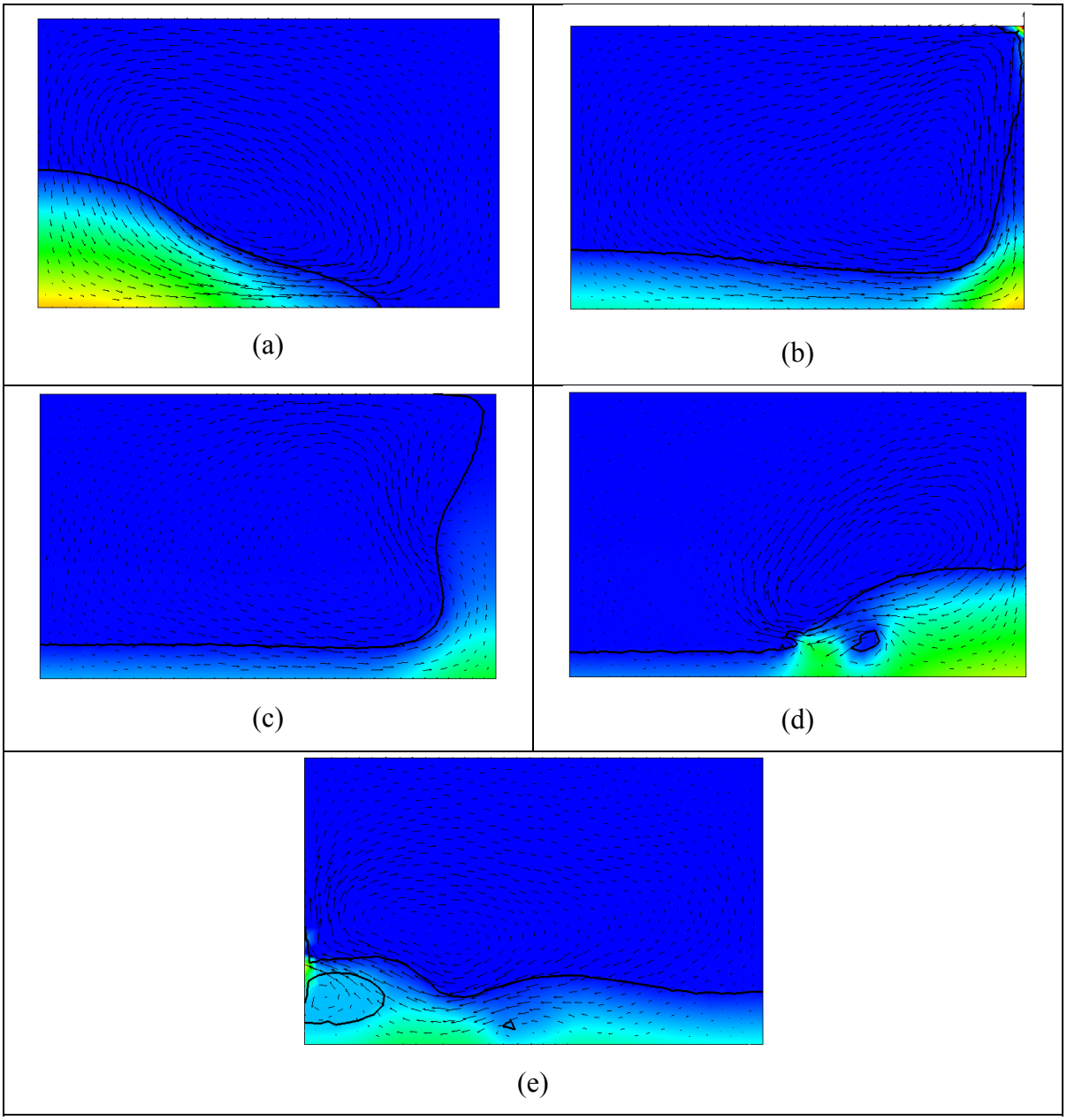


Figure 4.13: Dam break. The pressure at $t = 0.2s\,(a), 0.4s\,(b), 0.6s\,(c), 0.8s\,(d)$ and $1s\,(e)$.

It is clearly shown that the best mass conservation results are obtained with the mass conservation method. The stabilized method without mass conservation seems to diffuse the water phase (Figure 4.11). In Figure 4.12(b), the position of the water height on the left ($\tilde{H}$) is in qualitatively good agreement with the data from the experiment ((Martin et Moyce, 1952). However, there is a small offset of the water front position on the bed ($\tilde{Z}$) between the computed results and the experimental data (Martin et Moyce, 1952), as shown in Figure 4.12(a) for the range of $\tilde{T}$ between 0.84 and 1.83. This disparity is common for most interface-tracking methods (Lee, Dolbow et Mucha, 2014). Some discrepancies of the interface positions are also observed between the two stabilized methods when the wave breaks up (Figures 4.10 (e) and (f)), showing that the mass conservation constraint has some effect.

4.4.5 Rising bubble

The rising bubble benchmark test (Hysing et al., 2009) was introduced to assess the accuracy of numerical methods for two-phase flows more quantitatively, taking into account surface tension. The results of a first test case obtained with our methods are compared with the TP2D results (Hysing et al., 2009) which are known to be very accurate. The TP2D code (short for Transport Phenomena in 2D) is an incompressible flow solver to treat immiscible fluids with the level set method. More details on the solver can be found in (Hysing, 2007; Turek, 1997; 1999). The computational domain is a rectangle of dimensions 1m by 2m. At the initial time, a circular bubble with a radius of $r_0 = 0.25$ m is centred at (0.5, 0.5). The liquid which surrounds the bubble has a density of $\rho_1 = 1000$ kg/m^3 and a viscosity of $\mu_1 = 10$ Pa·s, and it occupies the subdomain Ω_1. The fluid bubble has a density of $\rho_2 = 100$ kg/m^3 and a viscosity of $\mu_2 = 1$ Pa·s, and it fills the subdomain Ω_2. The surface tension coefficient γ between the two phases is 24.5 N/m. The gravitational acceleration g is 9.81 m/s^2. The simulation is carried out for a period of 3 s with a time step Δt of 0.004 s. The mesh consists of 3,574 triangular elements. The initial condition of the rising bubble implies a Reynolds number $\mathrm{Re} = \dfrac{\rho_1 \sqrt{g} \left(2r_0\right)^{3/2}}{\mu_1} = 35$, an Eötvös number $Eo = \dfrac{4\rho_1 g r_0^2}{\gamma} = 10$, a density ration $\dfrac{\rho_1}{\rho_2} = 10$, and

a dynamic viscosity ratio $\frac{\mu_1}{\mu_2}=10$. The following quantities are defined to track the movement of the bubble: the centre of mass coordinates $\mathbf{X}_c=(x_c,y_c)=\frac{\int_{\Omega_2}\mathbf{x}\,dx}{\int_{\Omega_2}1\,dx}$ and the rising velocity $\mathbf{U}_c=(u_c,v_c)=\frac{\int_{\Omega_2}\mathbf{u}\,dx}{\int_{\Omega_2}1\,dx}$.

The problem is solved using our stabilized XFEM and level set method with the mass constraint. The application of the level set method to capture the interface allows for directly computing the curvature as $\kappa=\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right)$. Since the gradient is discontinuous across the elements, a continuous gradient $\mathbf{G}_h$ is constructed using an L_2 projection over the P2-finite element space. For all spatial directions of $i=1,2$, we solve the linear system for the components $(\mathbf{G}_h)_i$:

$$(q,(\mathbf{G}_h)_i)=(q,(\nabla\phi)_i) \tag{4.27}$$

where q are weighting functions belonging to the P2-finite element space. The continuous gradient is then given as a standard finite element interpolation using the P_2 polynomials N_j^{ϕ}: $\mathbf{G}_h=\sum_{i=1}^{nn}N_j^{\phi}\mathbf{G}_j$, where $\mathbf{G}_j$ are the nodal gradients and $nn=6$, the number of element nodes. The curvature is then computed at the integration points when evaluating the integrals in the variational formulation.

The centre of mass ordinate y_c, the rise velocity v_c, and the evolution of $Mass(\%)$ in time of the bubble are illustrated in Figure 4.14(a) to Figure 4.14(c). The interface Γ_{int} at $t=3s$ is shown in Figure 4.14(d).

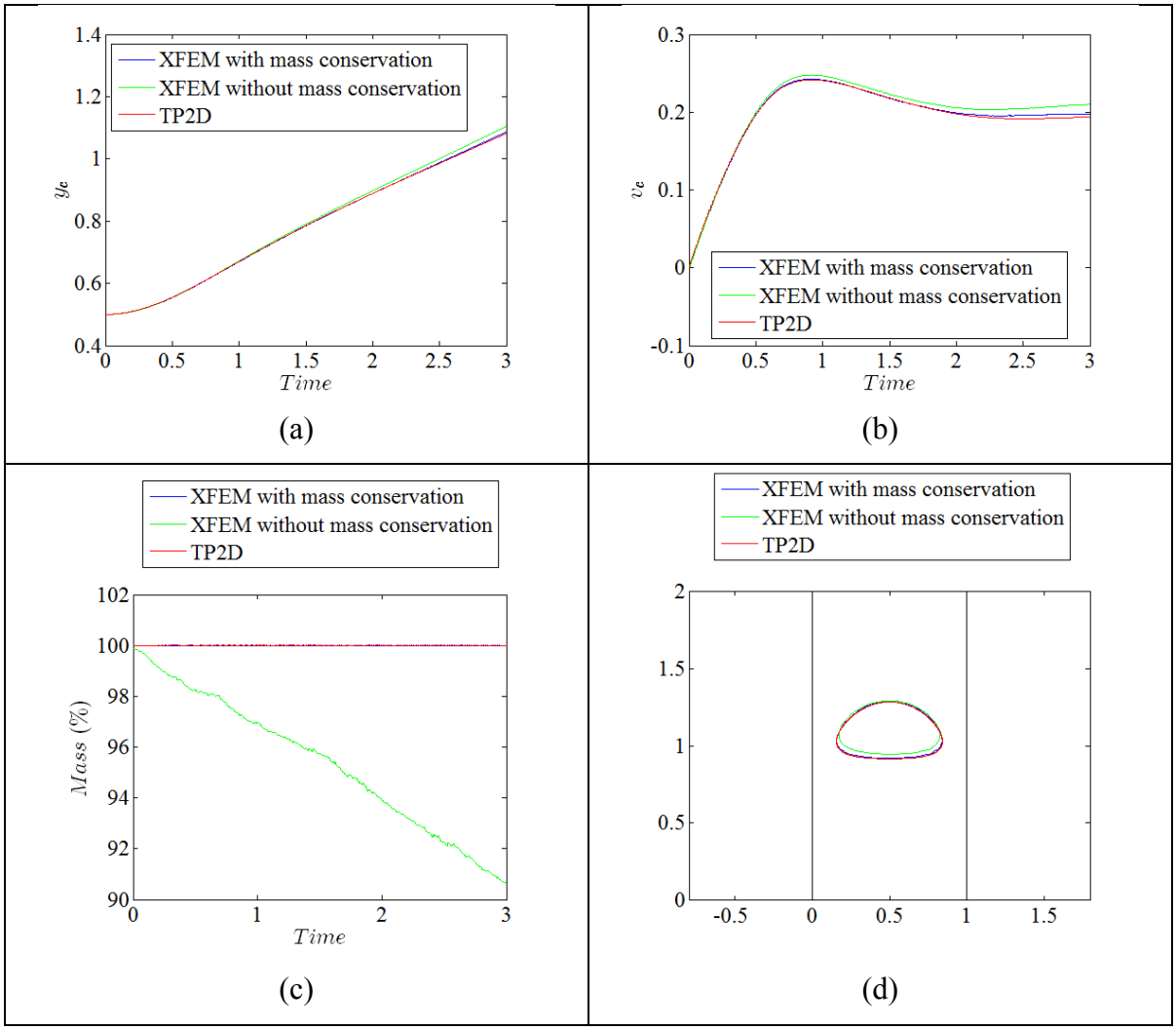


Figure 4.14: Rising bubble. Centre of mass ordinate y_c (a), rise velocity v_c (b), *Mass*(%) (c), and theinterface at t = 3s (d).

Figures 4.14(a-d) show that the proposed XFEM-level set methods deliver accurate results when they are combined with the mass conservation algorithm. For example, without the mass conservation algorithm the mass loss is about 9.32% (Figure 4.14 (c)). Moreover, our results obtained with the stabilized methods and the mass conservation are close to the TP2D results (Hysing et al., 2009) in terms of the position of the centre of mass, the rate of rise of the position of the interface and the conservation of mass, as shown in Table 4.5.

Table 4.5: Rising bubble. Centre of mass y_c and rise velocity v_c.

Method	Centre of mass y_c	Rise velocity v_c
TP2D	1.081 m	0.194 m/s
Stabilized method with mass conservation	1.086 m	0.198 m/s
Stabilized method without mass conservation	1.105 m	0.210 m/s

4.4.6 Dam break flow over an obstacle

The setting of this test is similar to that of the dam break test. The dimensions are the same as for the dam break test, but an obstacle is placed on the bottom surface, which adds some complications. The fluid properties and the time step are the same as in the dam break test. The mesh is unstructured and has 1,704 elements and 3,523 nodes. The total time of the simulation is 1 s, and the time step Δt is 0.001 s. The mesh and the interface at the initial instant are shown in Figure 4.15.

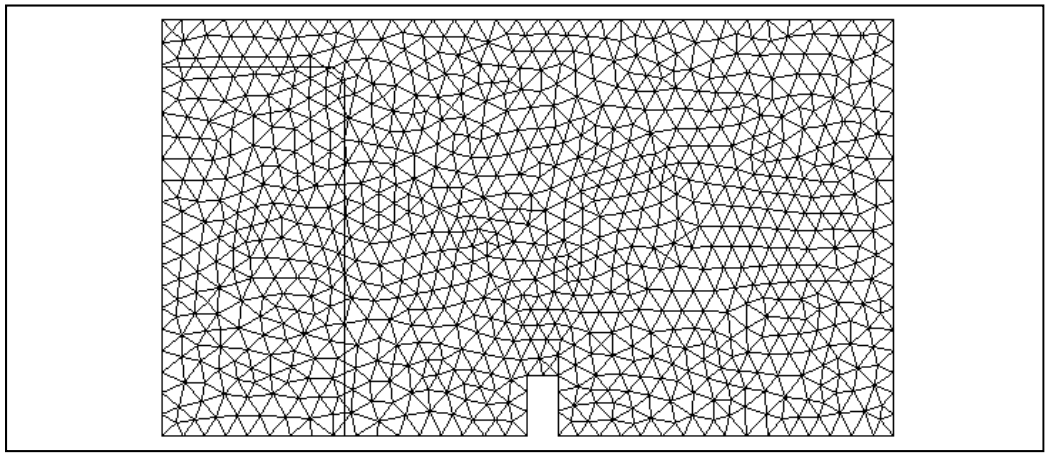


Figure 4.15: Dam break over an obstacle. Mesh and initial interface at t = 0 s.

In this test, we compare the results given by the stabilised variational method to those obtained with the stabilised variational method coupled with the mass conservation constraint applied every 5 time steps. The values of the parameters used and their indicators are detailed in Legend 10. Figure 4.16(a) to Figure 4.16(f) show the interface at different instants. There are

large interface deformations with encapsulated air cavities as the wave rolls over the obstacle and hits the wall.

Case 1: $\beta_1 = 0$ and $\beta_2 = 1$

Case 2: $\beta_1 = 0$ and $\beta_2 = 1$ with mass conservation

Legend 10

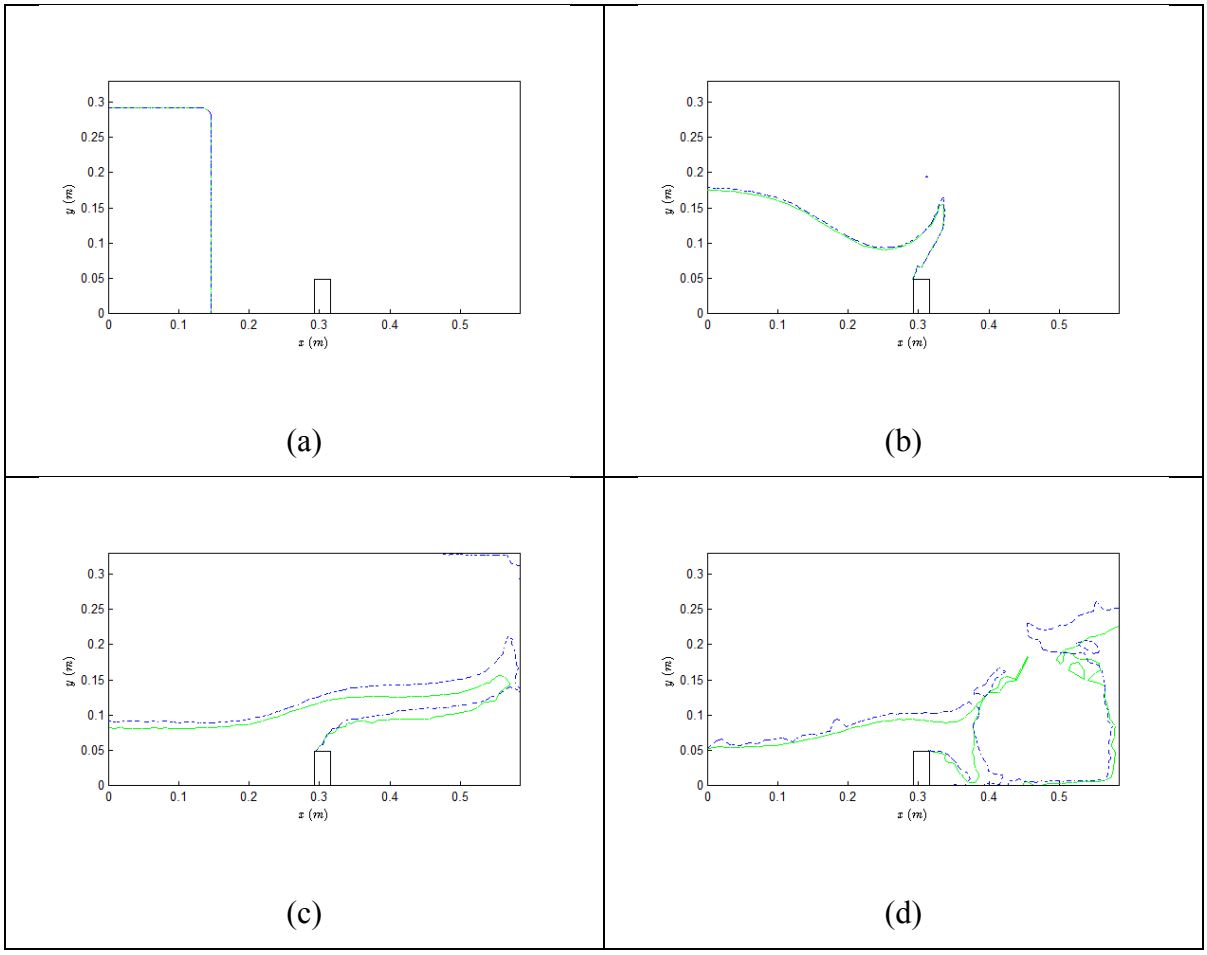
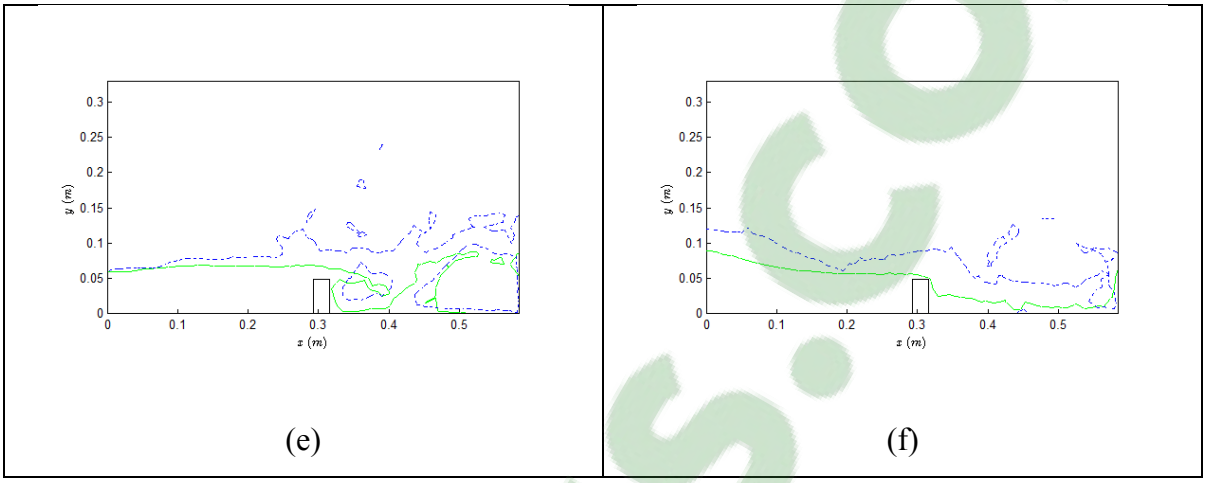


Figure 4.16: Dam break with an obstacle. Interface positions at $t = 0s\,(a), 0.2s\,(b), 0.4s\,(c), 0.6s\,(d), 0.8s\,(e)$ and $1s\,(f)$.

Figure 4.16 continuation



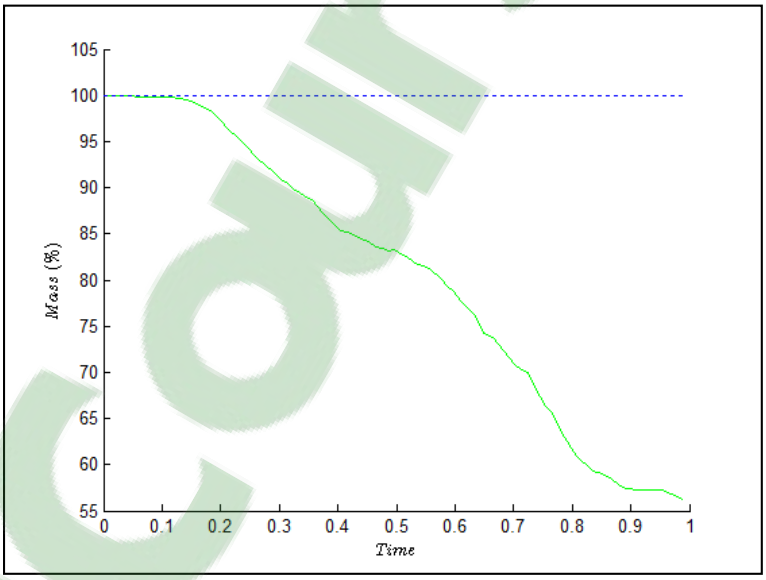
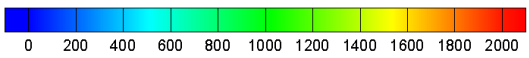


Figure 4.17: Dam break with an obstacle.
The mass loss of the lower subdomain.

The zero level set, the pressure field and the velocity vectors of the stabilised variational method with the mass conservation results are shown in Figure 4.18(a) to Figure 4.18(e). The colour map of the pressure contours is given in Legend 11.



Legend 11: Pressure

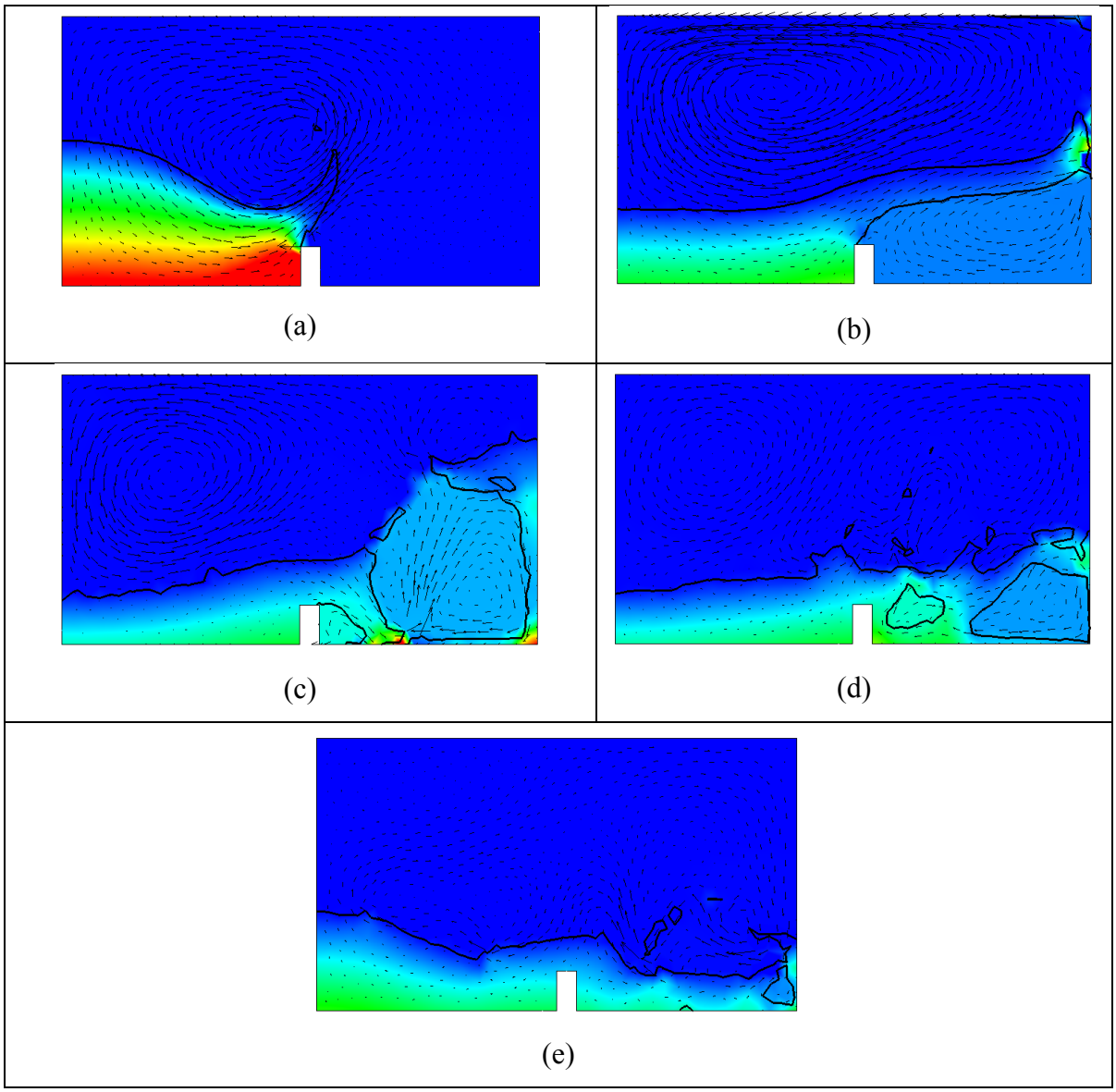


Figure 4.18: Dam break with an obstacle. The pressure at $t = 0.2s\,(a), 0.4s\,(b), 0.6s\,(c), 0.8s\,(d)$ and $1s\,(e)$.

In this case, the mass loss observed without mass conservation is more than 40%, as shown in Figure 4.17. Our results show that mass conservation is imperative to attain a realistic solution.

4.4.7 Rayleigh-Taylor instabilities

In this test, the Rayleigh-Taylor instability with a high Reynolds number is studied. This example can be retrieved from (Marchandise et Remacle, 2006; Rasthofer et al., 2011). There are two fluids in a rectangular domain $\Omega=[-L,L]\times[-H,H]$ of length $2L=1\,\mathrm{m}$ and height $2H=4\,\mathrm{m}$. The origin of the coordinate system is at the centre of the rectangular area. The heavier fluid is placed above the lighter fluid. The interface between the fluids initially has a sinusoidal form, as the interface is defined by $y=a\cos(\omega x)$ where $a=0.05$ is the amplitude and $\omega=2\pi$.

The upper fluid has a density $\rho_1=1.5\ \mathrm{kg/m^3}$ and a dynamic viscosity of $\mu_1=0.0033\ \mathrm{kg/(m\cdot s)}$, while the lower fluid has a density $\rho_2=1\ \mathrm{kg/m^3}$ and a dynamic viscosity of $\mu_2=0.0022\ \mathrm{kg/(m\cdot s)}$. The surface tension γ is taken into account for this example, and $\gamma=0.06$. The gravitational acceleration vector is $\mathbf{g}=\left(g_x,g_y\right)^T=\left(0,-10\ \mathrm{m/s^2}\right)^T$

.

The Reynolds number is $\mathrm{Re}=\dfrac{\sqrt{|g_y|HL}\rho_1}{\mu_1}=\dfrac{\sqrt{|g_y|HL}\rho_2}{\mu_2}=1000$, and the Atwood number is

$A=\dfrac{\rho_1-\rho_2}{\rho_1+\rho_2}=0.2$.

At the initial time, the velocity field is considered null. The velocity boundary conditions of the domain are sliding conditions. The mesh consists of 4132 triangular elements, and the time step is 0.005 s. Figure 4.19(a-j) shows the change of the interface at different times with the mass conservation. Figure 4.20(a-j) shows the velocity field, the velocity vectors and the interface at several times with the mass conservation. The mass loss is presented in Figure 4.22.

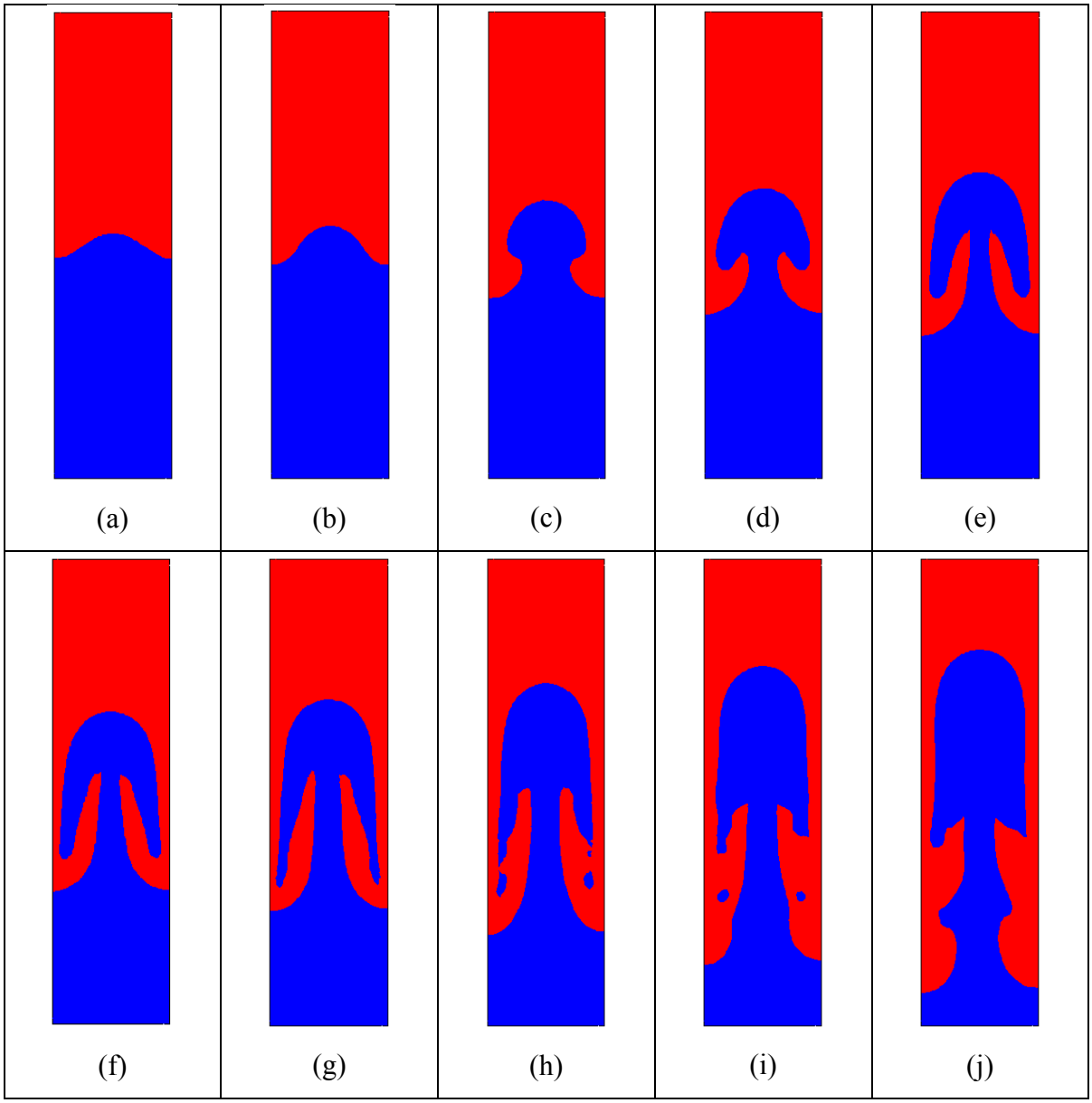


Figure 4.19: Rayleigh-Taylor instability. Interface position at
$t = 0.25s\,(a), 0.7s\,(b), 1.25s\,(c), 1.5s\,(d), 1.85s\,(e),$
$2s\,(f), 2.25s\,(g), 2.5s\,(h), 2.75s\,(i)$ and $3s\,(j)$
with the mass conservation

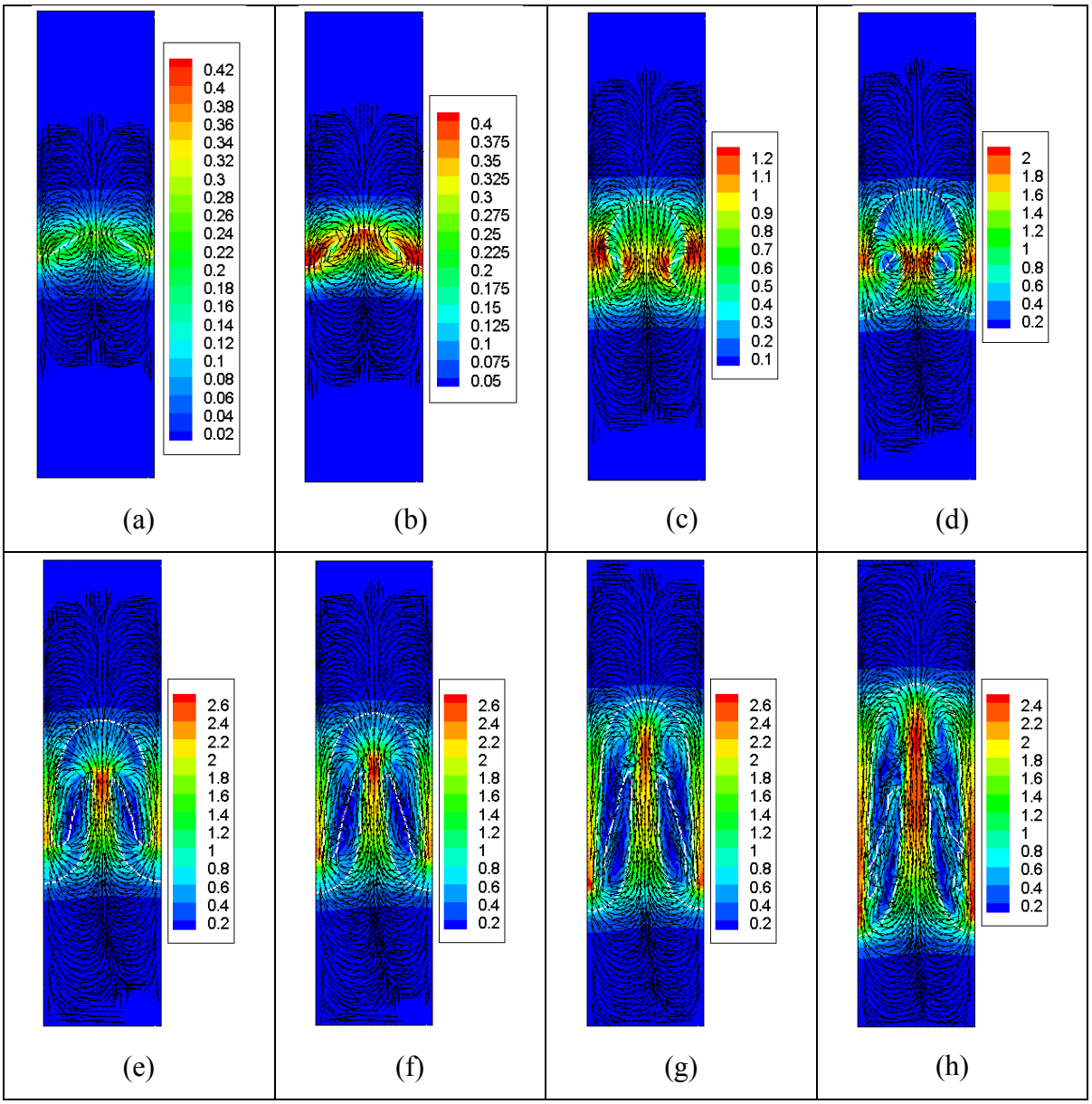
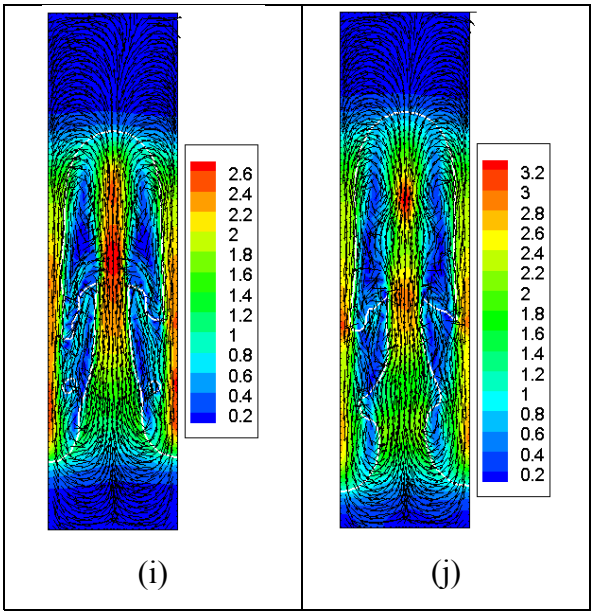


Figure 4.20: Rayleigh-Taylor instability.
Velocity field, velocity vector and interface position at
$t = 0.25s\,(a), 0.7s\,(b), 1.25s\,(c), 1.5s\,(d), 1.85s\,(e),$
$2s\,(f), 2.25s\,(g), 2.5s\,(h), 2.75s\,(i)$ and $3s\,(j)$
with the mass conservation

Figure 4.20 continuation



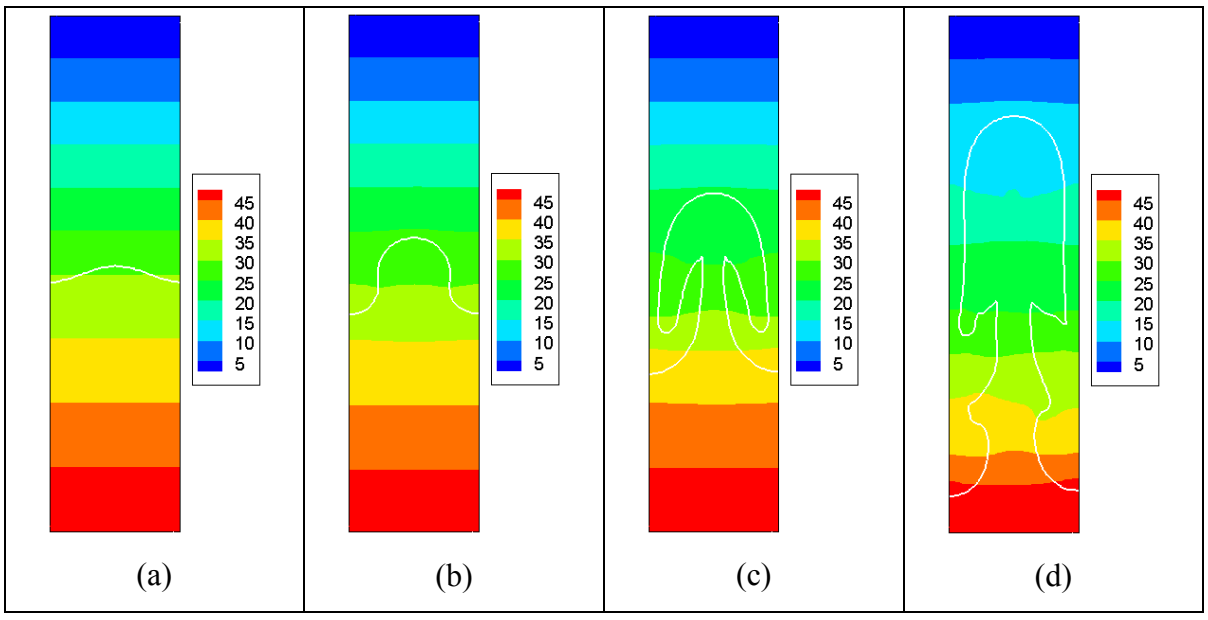


Figure 4.21: Rayleigh-Taylor instability.
Pressure field and interface position
at $t = 0.25s\,(a), 1s\,(b), 1.85s\,(c)$, and $3s\,(d)$
with the mass conservation

Case 1: $\beta_1 = 0$ and $\beta_2 = 1$

Case 2: $\beta_1 = 0$ and $\beta_2 = 1$ with mass conservation

Legend 12

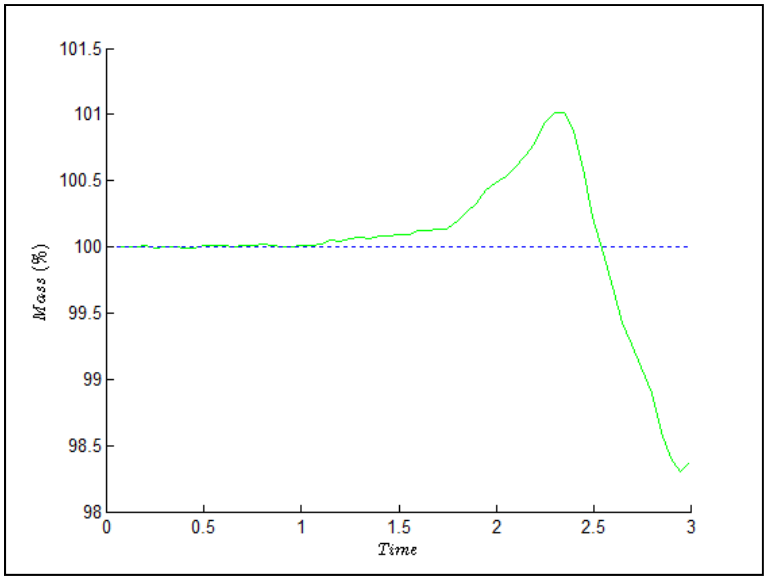


Figure 4.22: Rayleigh-Taylor instability.
The mass loss of the lower subdomain.

In Figure 4.19(a-j), the heavy fluid falls on the side walls and rolls in two vortexes, while the lighter fluid goes through the centre, filling the void left by the heavy fluid. A droplet detachment of the lighter fluid at approximately t = 2.45s can be observed. Figure 4.21(a-d) illustrates the pressure field and the interface position at different times. In Figure 4.20(a-j), the speed increases from 0 to 3 m/s between the initial time and the time t = 3 s. In Figure 4.22, the mass at T= 3 s is 98.3% with the stabilised method without mass conservation, while with the mass conservation method, it is 100%.

4.5 Conclusions

In this work, finite element methods are proposed and numerically evaluated for solving some moving interface problems using the level set approach. The variational formulation proposed in ((Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016)) is used to constrain the level set function to stay close to a distance function during the time computation. A mass

conservation approach was coupled with the variational formulation in order to evaluate its effect. The discretization was done over time with a second-order implicit scheme, and in space with extended finite element approximations over triangles. The following test problems were considered: a time-reversed vortex, a rigid body motion of Zalesak's disk, sloshing flow in a tank, a dam-break flow and a rising bubble. The mass and the interface positions are accurately resolved. For the dam-break flow, the stabilized variational method without mass conservation is not sufficiently accurate. However, when it is combined with the mass conservation approach, stable and accurate results are obtained. The rising bubble results are in good agreement with those obtained by the TP2D code (Hysing et al., 2009). In summary, the stabilized finite element method with the mass conservation algorithm delivers a robust and accurate approach for solving moving interface problems. This method is relatively easy to implement in existing finite element solvers in 2D as well as in 3D. As a possible improvement, the stabilized level set formulation and the mass correction method could be applied locally to a narrow-band defined by the cut elements. The displacement of the interface δ in equation (4.17) can be made proportional to the normal velocity at the interface given by: $u_n = \dfrac{\mathbf{u}\cdot\nabla\phi}{|\nabla\phi|}$ (Löhner, Yang et Oñate, 2006). Once the zero level (i.e. the interface) has been moved, the other level sets of the narrow band must be corrected accordingly.

CONCLUSIONS

In the first part of our research, new variational stabilised formulations are proposed to solve the level set transport equation without using the reinitialisation method. These formulations are obtained by adding new terms that depend on the residual of the Eikonal equation to the basic SUPG formulation of the level set equation. These methods were compared with a modified variant or alternative of the penalty method proposed by Li et al. (Li et al., 2005) and the geometric reinitialisation through a "brute force search algorithm". The temporal discretisation was made using a semi-implicit Crank-Nicolson scheme, while the space discretisation is carried out by approximation using quadratic elements. The numerical comparisons were made using standard validation tests such as a periodic reversed vortex and a rotational movement of the Zalesak disk, which is a rigid body. The proposed stabilisation methods improve the numerical behaviour of the SUPG method. They allow the effective of capture the interfaces between the phases, which are weakly or strongly deformed, and provide at the same time a retention of the acceptable mass. These methods are relatively easy to implement in existing finite element solvers in 2D and in 3D.

In the second part of our research, the variational formulation proposed in our previous articles ((Touré et Soulaïmani, 2012) and (Touré et Soulaïmani, 2016)) is used to constrain the level set function to remain the closest to a distance function throughout the numerical simulation. A mass conservation method was coupled with the variational formulation to assess its accuracy and stability. The temporal discretisation was performed with a semi-implicit Crank-Nicolson scheme, and the space discretisation is performed using approximations by quadratic finite elements. The following test problems were investigated: a periodic reversed vortex, a rotational movement of a rigid body of Zalesak disk flow, sloshing in a tank, a dam break flow without an obstacle, a dam break flow with an obstacle and Rayleigh-Taylor instabilities. In the periodic reversed vortex and the rigid body motion of a Zalesak disk, the comparison is performed between the initial position and the final position of the interface; the latter should remain the same as the initial position of the interface. Despite the observed oscillations of the mass loss curve, the results are acceptable. In the case of the periodic reversed vortex and the rigid body motion of the Zalesak disk, a sufficiently refined mesh would effectively allow

capturing the sharp edges of interfaces and providing satisfactory mass conservation and error norms. The problem of dam break flow is very arduous to resolve using the level set method. This case of flow induces not only large deformations of the interface but also random and abrupt changing velocity and pressure fields at the time. For this test case, the stabilised variational method without mass conservation lacks sufficient robustness and accuracy. However, when it is combined with the approach of the conservation of mass, we obtain stable and accurate results. Indeed, stabilised finite element methods for the level set of second-order temporal and spatial approximations combined with the mass conservation method provide a robust and precise approach to solve problems with a moving interface.

RECOMENDATIONS

During the development of the proposed method, we encountered several difficulties that were successfully overcome. The further development of the solver is to improve the performance and the extension of the boundaries of application subjects by modelling phenomena that are more complex. We offer the following ideas that should be integrated to improve and significantly expand the application areas of the developed code:

- the extension of the proposed methods by developing the code in 3D;
- the improvement of the geometric reinitialisation with a quadratic approximation of the interface instead of the linear approximation that was used in our research;
- the optimisation of the algorithm for computing the signed distance during the geometric reinitialisation;
- the application of mass conservation by locally moving the interface in a manner proportional to the normal velocity and then by reinitialising the level set in the rest of the domain;
- the parallelisation of code to increase the performance of the computing time and to accelerate the resolution to reach a level of high-performance computing (HPC);
- the integration of the energy equation that takes into account the heat transfer problems; and
- the modelling of phase change to simulate phenomena such as cavitation and atomisation.

APPENDIX

MASS CONSERVATION METHOD

The variational method proposed does not keep the mass of each phase in a precise manner. A loss (or gain) of weight of one of the phases is observed during the simulation, as noted in previous sections (Touré et Soulaïmani, 2016). In the sections below, we present other test assessments of the mass conservation algorithm.

I-Test: Time-reversed vortex flow using the fast marching method with/without the conservation of mass

A numerical test is performed with the SUPG method with the same conditions as defined in section 2.2.5. The interface contours obtained for the time-reversed vortex flow at $t = 0, T/4, T/2, 3T/4,$ and T and the mass loss for the fast marching method with and without the mass conservation method with a mesh of 125×125 are illustrated in Figure-A 1 (a-f) to Figure-A 3.

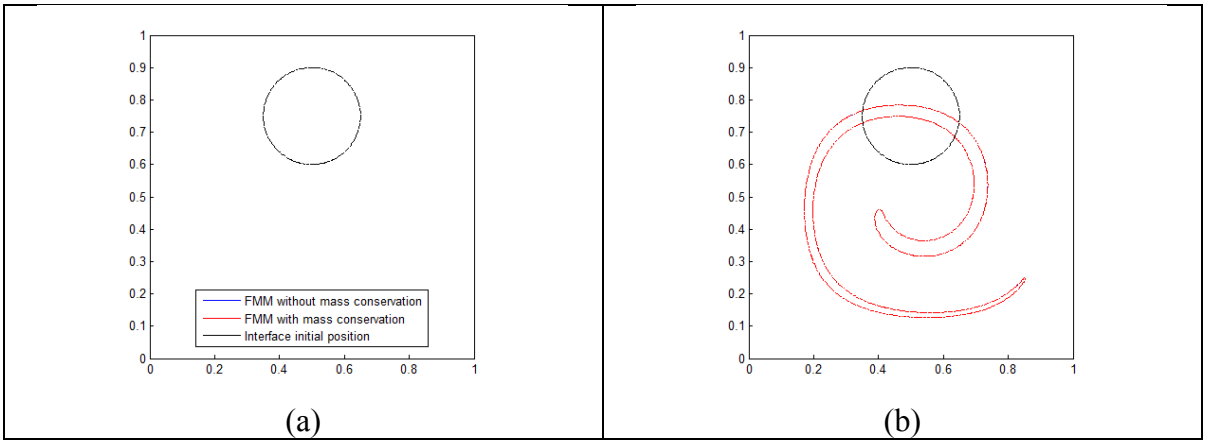
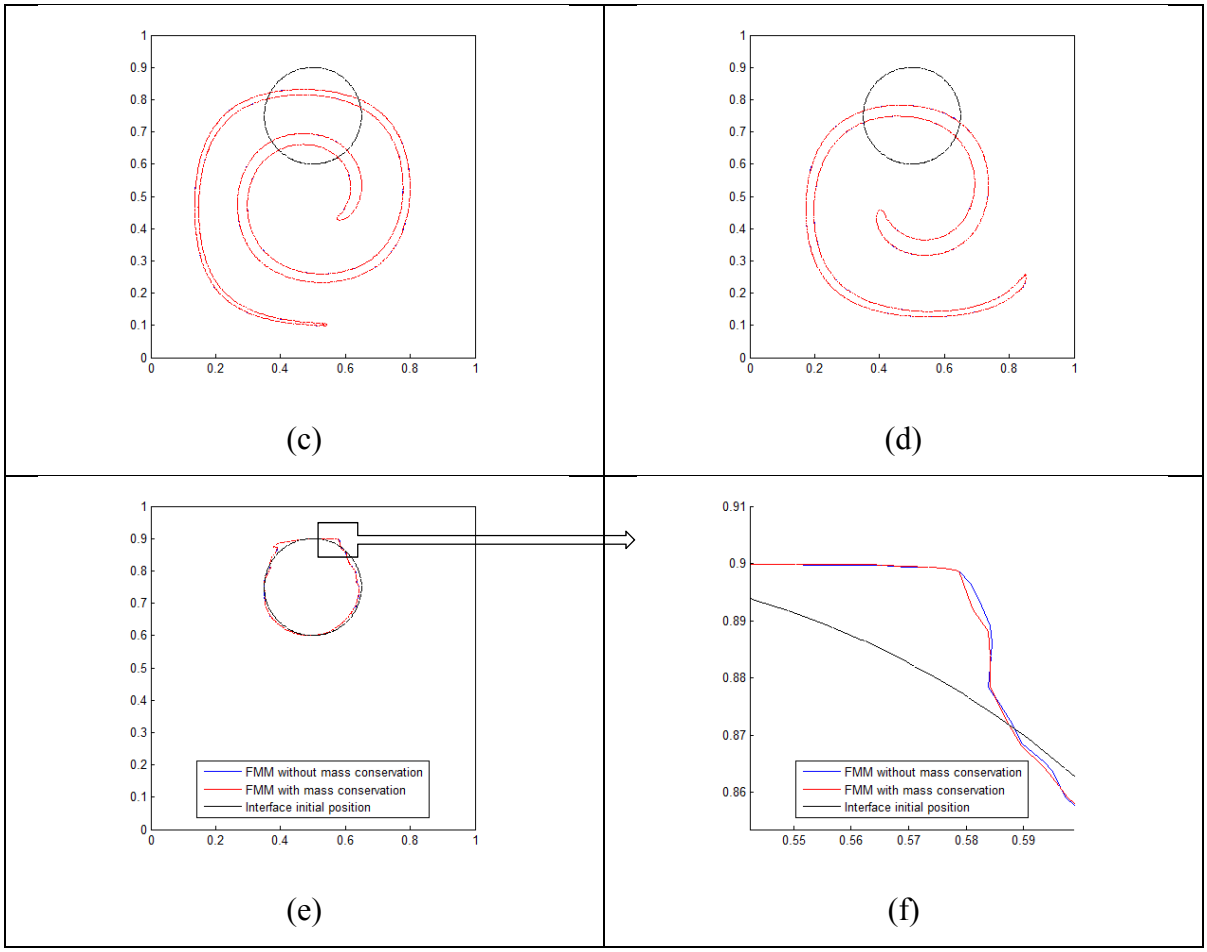


Figure-A 1: Time-reversed vortex flow. Interface positions at $t = 0\,(a), T/4\,(b), T/2\,(c), 3T/4\,(d)$ and $T\,(e)$ and with the 125×125 mesh. Closer view of interface positions at T (f).

Figure-A 1 continuation



The mass conservation is analysed by comparing the fast marching method with or without mass conservation of four different meshes. The evolution of the mass of the disk is shown in Figure-A 2 and Figure-A 3.

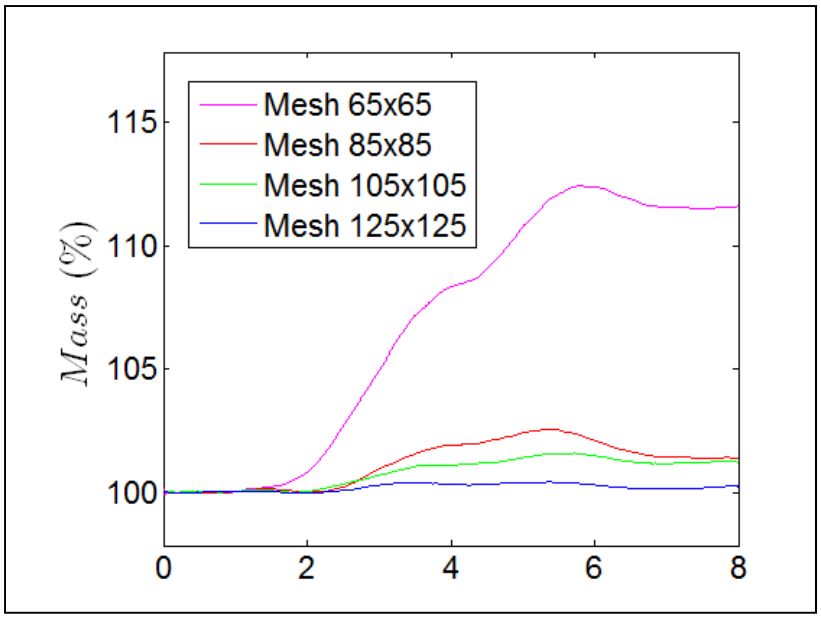


Figure-A 2: Time-reversed vortex flow. Percentage of disk area $Mass(\%)$ versus time using the FMM without the mass conservation.

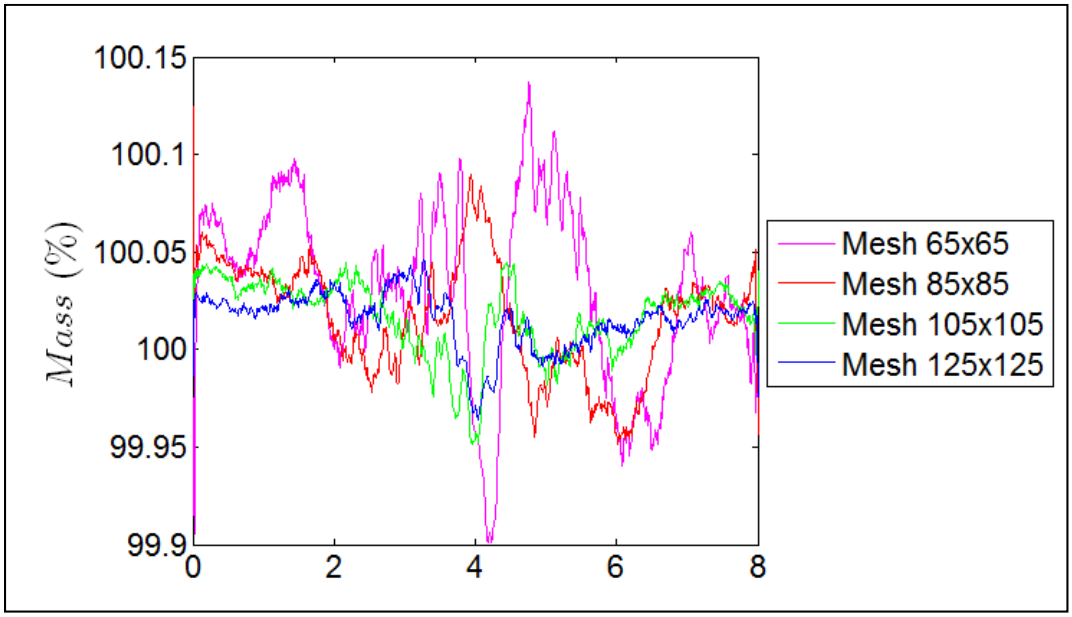


Figure-A 3: Time-reversed vortex flow. Percentage of disk area $Mass(\%)$ versus time using the FMM with the mass conservation.

In Figure-A 1(a-f), the interfaces for the FMM without mass conservation and the FMM with mass conservation method are overlapped. The mass conservation method improves the results for all of the different meshes. For the coarse 65×65 mesh, there is a mass conservation improvement of 10% using the FMM with mass conservation rather than the FMM without mass conservation. Hence, the mass conservation is suitable for a method that does not require fine mesh.

II-Test: Time-reversed flow through a convergent-divergent channel using the geometric reinitialalisation with/without the mass conservation

The aim of the test is to assess the accuracy of the proposed methods with an analytical benchmark of the advection of a bubble through a contracted channel (Vincent, Lakehal et Friess, 2004). The imposed velocity is a solenoidal velocity field that is given by

$$u = \left(1 - \left(\frac{y}{f(x)}\right)^2\right)\frac{1}{f(x)} \tag{A.1}$$

and

$$v = \left(1 - \left(\frac{y}{f(x)}\right)^2\right)\frac{y}{(f(x))^2}\frac{\mathrm{d}f(x)}{\mathrm{d}x} \tag{A.2}$$

where

$$f(x) = 1 - a\exp\left[-\frac{1}{2}\left(\frac{x}{b}\right)^2\right] \tag{A.3}$$

with $a = 0.75$ and $b = 0.5$. The interface positions obtained for $t = 0, t_1 = 1.787109375, t_2 = 2.958984375,$ and $2t_2$ with the 128×64 (10,478 triangular elements) unstructured mesh using the SUPG method, the geometric reinitialisation, and the mass conservation are shown in Figure-A 4 (a-d). The flow is reversed between t_2 and $2t_2$, such that

$$u = -\left(1 - \left(\frac{y}{f(x)}\right)^2\right)\frac{1}{f(x)} \tag{A.4}$$

and

$$v = \left(1 - \left(\frac{y}{f(x)}\right)^2\right) \frac{y}{\left(f(x)\right)^2} \frac{\mathrm{d}f(x)}{\mathrm{d}x} \tag{A.5}$$

where

$$f(x) = 1 - a \exp\left[-\frac{1}{2}\left(\frac{x}{b}\right)^2\right] \tag{A.6}$$

The black dots in Figure-A 4(a-d) represent the positions of the marker points listed in table 24.1 of (Vincent, Lakehal et Friess, 2004). The mass loss is illustrated in Figure-A 8 and Table-A 1, and the error norms are presented in Table-A 2.

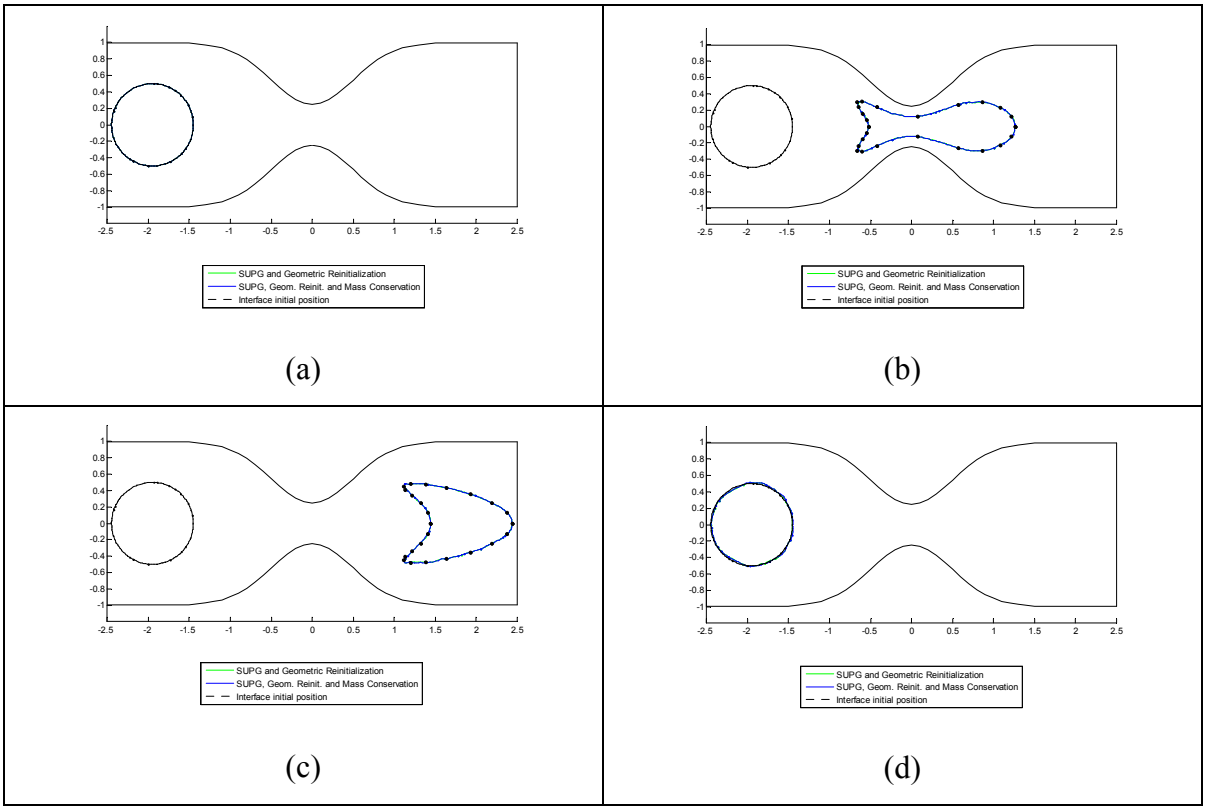


Figure-A 4: Convergent-divergent channel. Interface positions at $t = 0$(a), t_1(b), t_2(c), and $2t_2$(d).

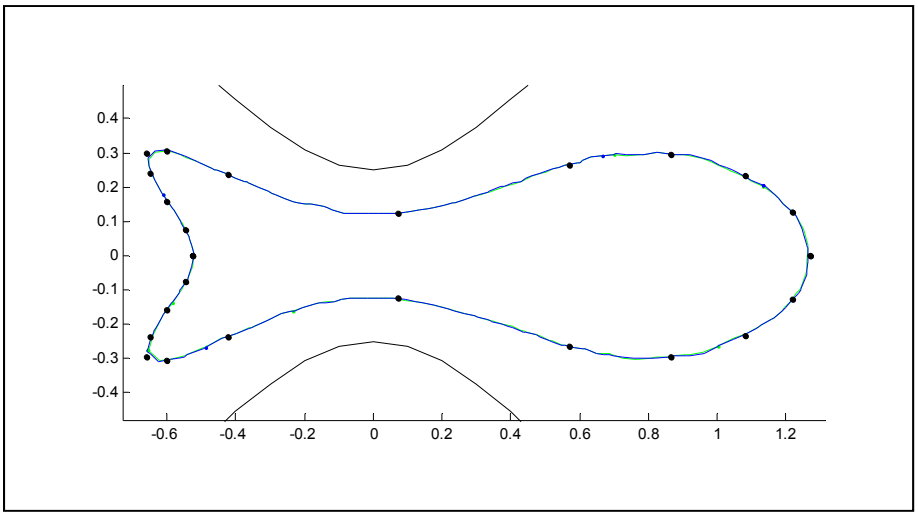


Figure-A 5: Convergent-divergent channel. Closer view at t_1.

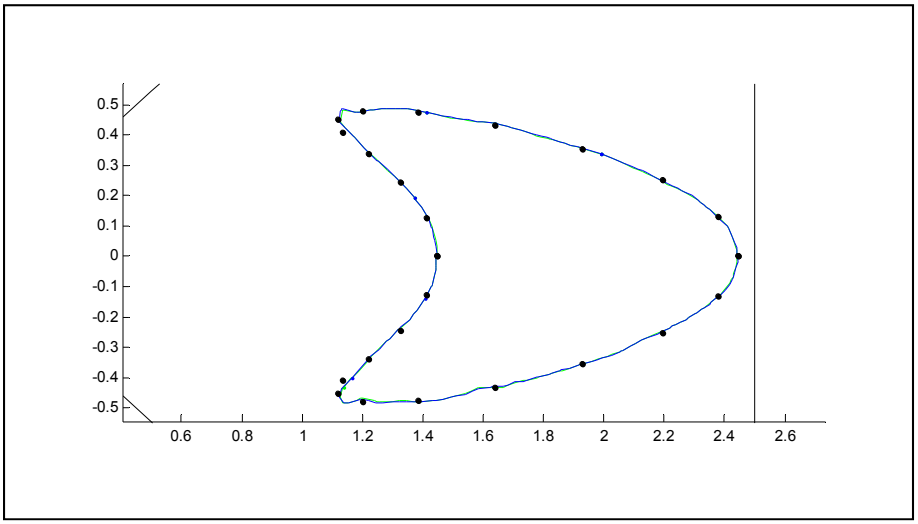


Figure-A 6: Convergent-divergent channel. Closer view at t_2.

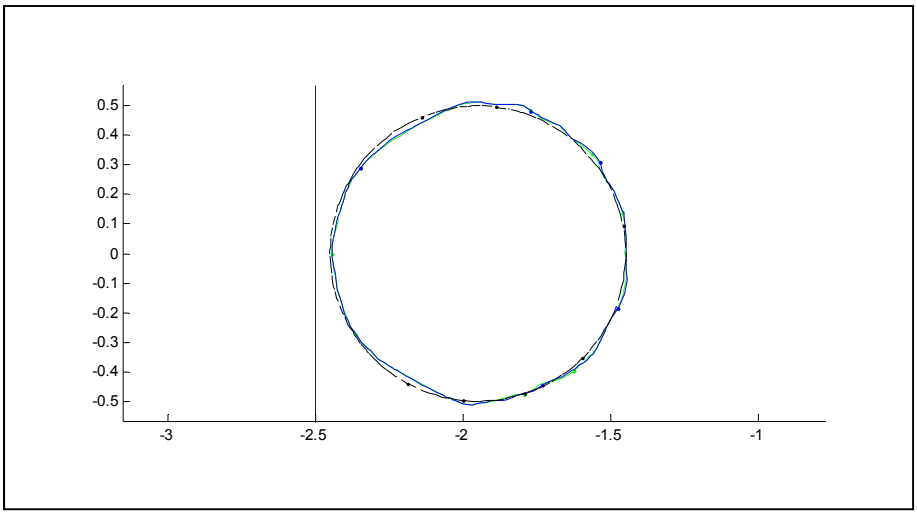


Figure-A 7: Convergent-divergent channel. Closer view at $2t_2$.

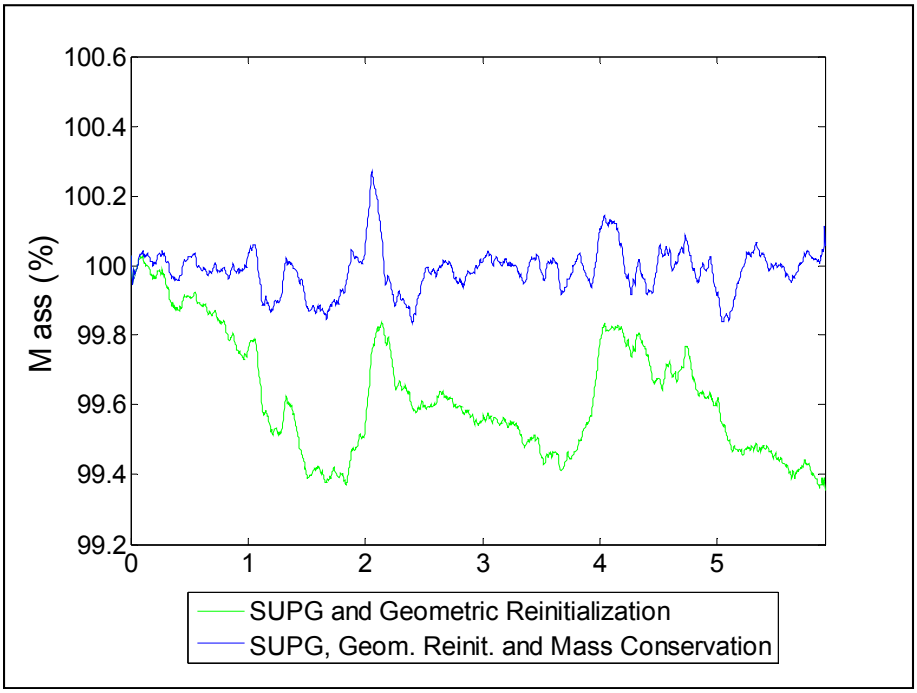


Figure-A 8: Convergent-divergent channel. Percentage of the disk area $Mass(\%)$ versus time with the stabilised finite element method, the geometric reinitialisation, and the mass conservation.

Table-A 1: Convergent-divergent channel. Percentage of the disk area $Mass(\%)$ versus time with the stabilised finite element method, the geometric reinitialisation, and the mass conservation.

Method	Mass (%) at t_2	Mass (%) at $2t_2$
Case 1: $\alpha = 1, \beta_1 = 0$ and $\beta_2 = 0$ SUPG method with geometric reinitialisation	99.62	99.35
Case 2: $\alpha = 1, \beta_1 = 0$ and $\beta_2 = 0$ SUPG method with geometric reinitialisation every 5 time steps and mass conservation	99.96	100.06

Table-A 2: Convergent-divergent channel. Error norms E_1, E_2, E_3 and E_4 at $2t_2$

Method (à t = à 5.91796875s)	E_1	E_2	E_3	E_4
Case 1: $\alpha = 1, \beta_1 = 0$ and $\beta_2 = 0$ SUPG method with geometric reinitialisation	6.37E-02	2.95E-02	4.78E-03	5.95E-03
Case 2: $\alpha = 1, \beta_1 = 0$ and $\beta_2 = 0$ SUPG method with geometric reinitialisation every 5 time steps and mass conservation	6.09E-02	2.78E-02	4.51E-03	6.16E-03

The interface positions of Figure-A 4(a) to Figure-A 4(d) and Figure-A 5 to Figure-A 7 show that our results are in good agreement with the positions of the marker points listed in table 24.1 of (Vincent, Lakehal et Friess, 2004). The mass loss is minimised with the mass conservation in Table-A 1. In this test, the numerical errors are similar whether or not the mass conservation is used. The mass conservation does not significantly shift the interface position according to the results of Table-A 2.

BIBLIOGRAPHY

Adalsteinsson, D., et J. A. Sethian. 1999. « The fast construction of extension velocities in level set methods ». *Journal of Computational Physics,* vol. 148, nº 1, p. 2-22.

Akkerman, I., Y. Bazilevs, D. J. Benson, M. W. Farthing et C. E. Kees. 2011. « Free-Surface Flow and Fluid-Object Interaction Modeling With Emphasis on Ship Hydrodynamics ». *Journal of Applied Mechanics,* vol. 79, nº 1, p. 010905-010905.

Ausas, R. F., E. A. Dari et G. C. Buscaglia. 2011. « A geometric mass-preserving redistancing scheme for the level set function ». *International Journal for Numerical Methods in Fluids,* vol. 65, nº 8, p. 989-1010.

Babuška, I., et U. Banerjee. 2012. « Stable Generalized Finite Element Method (SGFEM) ». *Computer Methods in Applied Mechanics and Engineering,* vol. 201–204, nº 0, p. 91-111.

Babuška, I., et J. M. Melenk. 1997. « The partition of unity method ». *International Journal for Numerical Methods in Engineering,* vol. 40, nº 4, p. 727-758.

Babuška, Ivo, Gabriel Caloz et John E. Osborn. 1994. « Special finite element methods for a class of second order elliptic problems with rough coefficients ». *SIAM J. Numer. Anal.,* vol. 31, nº 4, p. 945-981.

Bänsch, Eberhard. 2001. « Finite element discretization of the Navier–Stokes equations with a free capillary surface ». *Numerische Mathematik,* vol. 88, nº 2, p. 203-235.

Baumann, Carlos Erik, et J. Tinsley Oden. 1999. « A discontinuous hp finite element method for convection—diffusion problems ». *Computer Methods in Applied Mechanics and Engineering,* vol. 175, nº 3–4, p. 311-341.

Béchet, E., H. Minnebo, N. Moës et B. Burgardt. 2005. « Improved implementation and robustness study of the X-FEM for stress analysis around cracks ». *International Journal for Numerical Methods in Engineering,* vol. 64, nº 8, p. 1033-1056.

Bell, John B., Phillip Colella et Harland M. Glaz. 1989. « A second-order projection method for the incompressible navier-stokes equations ». *Journal of Computational Physics,* vol. 85, nº 2, p. 257-283.

Brooks, Alexander N., et Thomas J. R. Hughes. 1982. « Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the

incompressible Navier-Stokes equations ». *Computer Methods in Applied Mechanics and Engineering,* vol. 32, nº 1–3, p. 199-259.

Burman, Erik. 2010. « Consistent SUPG-method for transient transport problems: Stability and convergence ». *Computer Methods in Applied Mechanics and Engineering,* vol. 199, nº 17–20, p. 1114-1123.

Chessa, J., et T. Belytschko. 2003a. « An Extended Finite Element Method for Two-Phase Fluids ». *Journal of Applied Mechanics,* vol. 70, nº 1, p. 10-17.

Chessa, Jack, et Ted Belytschko. 2003b. « An enriched finite element method and level sets for axisymmetric two-phase flow with surface tension ». *International Journal for Numerical Methods in Engineering,* vol. 58, nº 13, p. 2041-2064.

Choi, Young Joon. 2011. « Modeling Particulate Complex Flows using XFEM ». Eindhoven, Netherlands, Technische Universiteit Eindhoven, 169 p. < http://repository.tue.nl/716491 >.

Chunming, Li, Xu Chenyang, Gui Changfeng et M. D. Fox. 2010. « Distance Regularized Level Set Evolution and Its Application to Image Segmentation ». *IEEE Transactions on Image Processing,* vol. 19, nº 12, p. 3243-54.

Coppola-Owen, A. H., et R. Codina. 2005. « Improving Eulerian two-phase flow finite element approximation with discontinuous gradient pressure shape functions ». *International Journal for Numerical Methods in Fluids,* vol. 49, nº 12, p. 1287-1304.

Coppola-Owen, H. 2009. « A Finite Element Model for Free Surface and Two Fluid Flows on Fixed Meshes ». PhD thesis. Universitat Politecnica de Catalunya.

Desjardins, O., V. Moureau et H. Pitsch. 2008. « An accurate conservative level set/ghost fluid method for simulating turbulent atomization ». *Journal of Computational Physics,* vol. 227, nº 18, p. 8395-8416.

Di Pietro, D. A., S. Lo Forte et N. Parolini. 2006. « Mass preserving finite element implementations of the level set method ». *Applied Numerical Mathematics,* vol. 56, nº 9, p. 1179-1195.

Dolbow, John, Nicolas Moës et Ted Belytschko. 2001. « An extended finite element method for modeling crack growth with frictional contact ». *Computer Methods in Applied Mechanics and Engineering,* vol. 190, nº 51–52, p. 6825-6846.

Doyeux, V., Y. Guyot, V. Chabannes, C. Prud'homme et M. Ismail. 2013. « Simulation of two-fluid flows using a finite element/level set method. Application to bubbles and vesicle dynamics ». *Journal of Computational and Applied Mathematics,* vol. 246, p. 251-259.

Duan, Xian-Bao, Yi-Chen Ma et Rui Zhang. 2008a. « Shape-topology optimization for Navier–Stokes problem using variational level set method ». *Journal of Computational and Applied Mathematics,* vol. 222, nº 2, p. 487-499.

Duan, Xianbao, Yichen Ma et Rui Zhang. 2008b. « Optimal shape control of fluid flow using variational level set method ». *Physics Letters A,* vol. 372, nº 9, p. 1374-1379.

Fahsi, Adil. 2016. « An Extended Finite Element-level Set Method for Simulating Two-phase and Free-surface Flows ». Ph.D. Thesis. Montréal, École de technologie supérieure.

Fries, Thomas-Peter. 2009. « The intrinsic XFEM for two-fluid flows ». *International Journal for Numerical Methods in Fluids,* vol. 60, nº 4, p. 437-471.

Fries, Thomas-Peter, et Ted Belytschko. 2006. « The intrinsic XFEM: a method for arbitrary discontinuities without additional unknowns ». *International Journal for Numerical Methods in Engineering,* vol. 68, nº 13, p. 1358-1385.

Fries, Thomas-Peter, et Ted Belytschko. 2010. « The extended/generalized finite element method: An overview of the method and its applications ». *International Journal for Numerical Methods in Engineering,* vol. 84, nº 3, p. 253-304.

Ganesan, Sashikumaar. 2006. « Finite element methods on moving meshes for free surface and interface flows ». Magdeburg, Germany, Otto-von-Guericke-Universit, Magdeburg, Germany, 122 p.

Glimm, J. 1982. « Tracking of interfaces for fluid flow: Accurate methods for piecewise smooth problems ». In *Transonic, shock, and multidimensional flows: Advances in scientific computing; Proceedings of the Symposium, New York*. (Madison, WI, May 13-15, 1981), Academic Press Vol. A83-29926 12-34, p. 259-287. New York: Meyer, R. E.

Glimm, James, et Oliver A. McBryan. 1985. « A computational model for interfaces ». *Advances in Applied Mathematics,* vol. 6, nº 4, p. 422-435.

Gomes, José, et Olivier Faugeras. 2000. « Reconciling Distance Functions and Level Sets ». *Journal of Visual Communication and Image Representation,* vol. 11, nº 2, p. 209-223.

Groß, Sven, et Arnold Reusken. 2007. « An extended pressure finite element space for two-phase incompressible flows with surface tension ». *Journal of Computational Physics,* vol. 224, nº 1, p. 40-58.

Gross, Sven, et Arnold Reusken. 2011. *Numerical Methods for Two-phase Incompressible Flows*, 40. Coll. « Springer Series in Computational Mathematics ». Springer Berlin Heidelberg.

Harlow, F. H., et J. E. Welch. 1965. « Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface ». *Physics of Fluids,* vol. 8, nº 12, p. 2182-2189.

Hirt, C. W., et B. D. Nichols. 1981. « Volume of fluid (VOF) method for the dynamics of free boundaries ». *Journal of Computational Physics,* vol. 39, nº 1, p. 201-25.

Hu, Changhong, et Makoto Sueyoshi. 2010. « Numerical simulation and experiment on dam break problem ». *Journal of Marine Science and Application,* vol. 9, nº 2, p. 109-114.

Hu, Howard H., N. A. Patankar et M. Y. Zhu. 2001. « Direct Numerical Simulations of Fluid–Solid Systems Using the Arbitrary Lagrangian–Eulerian Technique ». *Journal of Computational Physics,* vol. 169, nº 2, p. 427-462.

Hughes, Thomas J. R., Leopoldo P. Franca et Marc Balestra. 1986. « A new finite element formulation for computational fluid dynamics: V. Circumventing the babuška-brezzi condition: a stable Petrov-Galerkin formulation of the stokes problem accommodating equal-order interpolations ». *Computer Methods in Applied Mechanics and Engineering,* vol. 59, nº 1, p. 85-99.

Hughes, Thomas J. R., Leopoldo P. Franca et Gregory M. Hulbert. 1989. « A new finite element formulation for computational fluid dynamics: VIII. The galerkin/least-squares method for advective-diffusive equations ». *Computer Methods in Applied Mechanics and Engineering,* vol. 73, nº 2, p. 173-189.

Hyman, James M. 1984. « Numerical methods for tracking interfaces ». *Physica D: Nonlinear Phenomena,* vol. 12, nº 1–3, p. 396-407.

Hysing, S. 2007. « Performance in Two-Phase Flow Simulations: Finite Elements and the Level Set Method. ». Dortmund, Dortmund University.

Hysing, S. R., et S. Turek. 2005. *The eikonal equation: numerical efficiency vs. algorithmic complexity on quadrilateral grids*. Coll. « Algoritmy 2005: 17th Conference on Scientific Computing, Proceedings ». Bratislava: Slovak Univ Technology, Bratislava, 22-31 p.

Hysing, S., S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan et L. Tobiska. 2009. « Quantitative benchmark computations of two-dimensional bubble dynamics ». *International Journal for Numerical Methods in Fluids,* vol. 60, nº 11, p. 1259-1288.

Johnson, A. A., et T. E. Tezduyar. 1997. « 3D Simulation of fluid-particle interactions with the number of particles reaching 100 ». *Computer Methods in Applied Mechanics and Engineering,* vol. 145, nº 3–4, p. 301-321.

Kees, C. E., I. Akkerman, M. W. Farthing et Y. Bazilevs. 2011. « A conservative level set method suitable for variable-order approximations and unstructured meshes ». *Journal of Computational Physics,* vol. 230, nº 12, p. 4536-4558.

Kuzmin, Dmitri. 2014. « An optimization-based approach to enforcing mass conservation in level set methods ». *Journal of Computational and Applied Mathematics,* vol. 258, p. 78-86.

Laadhari, Aymen, Pierre Saramito et Chaouqi Misbah. 2010. « Improving the mass conservation of the level set method in a finite element context ». *Comptes Rendus Mathematique,* vol. 348, nº 9–10, p. 535-540.

Le Chenadec, V., et H. Pitsch. 2013. « A 3D Unsplit Forward/Backward Volume-of-Fluid Approach and Coupling to the Level Set Method ». *Journal of Computational Physics,* vol. 233, p. 10-33.

Lee, Curtis, John Dolbow et Peter J. Mucha. 2014. « A narrow-band gradient-augmented level set method for multiphase incompressible flow ». *Journal of Computational Physics,* vol. 273, p. 12-37.

Legrain, G., N. Moës et A. Huerta. 2008. « Stability of incompressible formulations enriched with X-FEM ». *Computer Methods in Applied Mechanics and Engineering,* vol. 197, nº 21–24, p. 1835-1849.

Li, Chunming, Chenyang Xu, Changfeng Gui et Martin D. Fox. 2005. « Level set evolution without re-initialization: A new variational formulation ». In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, June 20, 2005 - June 25, 2005*. (San Diego, CA, United states) Vol. I, p. 430-436. Coll. « Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005 »: Institute of Electrical and Electronics Engineers Computer Society. < http://dx.doi.org/10.1109/CVPR.2005.213
http://ieeexplore.ieee.org/ielx5/9901/31472/01467299.pdf?tp=&arnumber=1467299&isnumber=31472 >.

Li, Chunming, Chenyang Xu, Kishori M. Konwar et Martin D. Fox. 2006. « Fast distance preserving level set evolution for medical image segmentation ». In *9th International Conference on Control, Automation, Robotics and Vision, 2006, ICARCV '06, December 5, 2006 - December 8, 2006*. (Singapore, Singapore). Coll. « 9th International Conference on Control, Automation, Robotics and Vision, 2006, ICARCV '06 »: Inst. of Elec. and Elec. Eng. Computer Society. < http://dx.doi.org/10.1109/ICARCV.2006.345357
http://ieeexplore.ieee.org/ielx5/4149990/4126184/04150341.pdf?tp=&arnumber=4150341&isnumber=4126184 >.

Liu, C. X., F. F. Dong, S. F. Zhu, D. X. Kong et K. F. Liu. 2011. « New Variational Formulations for Level Set Evolution Without Reinitialization with Applications to Image Segmentation ». *Journal of Mathematical Imaging and Vision,* vol. 41, nº 3, p. 194-209.

Löhner, Rainald, Chi Yang et Eugenio Oñate. 2006. « On the simulation of flows with violent free surface motion ». *Computer Methods in Applied Mechanics and Engineering,* vol. 195, nº 41–43, p. 5597-5620.

Marchandise, Emilie, et Jean-François Remacle. 2006. « A stabilized finite element method using a discontinuous level set approach for solving two phase incompressible flows ». *Journal of Computational Physics,* vol. 219, nº 2, p. 780-800.

Martin, J. C., et W. J. Moyce. 1952. « Part IV. An Experimental Study of the Collapse of Liquid Columns on a Rigid Horizontal Plane ». *Philosophical Transactions of the Royal Society of London, Series A, Mathematical and Physical Sciences,* vol. 244, p. 13.

Massing, André, Mats G. Larson, Anders Logg et Marie E. Rognes. 2014. « A Stabilized Nitsche Fictitious Domain Method for the Stokes Problem ». *Journal of Scientific Computing,* vol. 61, nº 3, p. 604-628.

Melenk, J. M., et I. Babuška. 1996. « The partition of unity finite element method: Basic theory and applications ». *Computer Methods in Applied Mechanics and Engineering,* vol. 139, nº 1–4, p. 289-314.

Ménard, T., S. Tanguy et A. Berlemont. 2007. « Coupling level set/VOF/ghost fluid methods: Validation and application to 3D simulation of the primary break-up of a liquid jet ». *International Journal of Multiphase Flow,* vol. 33, nº 5, p. 510-524.

Menk, Alexander, et Stéphane P. A. Bordas. 2011. « A robust preconditioning technique for the extended finite element method ». *International Journal for Numerical Methods in Engineering,* vol. 85, nº 13, p. 1609-1632.

Moës, N., M. Cloirec, P. Cartraud et J. F. Remacle. 2003. « A computational approach to handle complex microstructure geometries ». *Computer Methods in Applied Mechanics and Engineering,* vol. 192, nº 28–30, p. 3163-3177.

Moës, Nicolas, et Ted Belytschko. 2002. « Extended finite element method for cohesive crack growth ». *Engineering Fracture Mechanics,* vol. 69, nº 7, p. 813-833.

Noh, W. F., et P. Woodward. 1976. « SLIC (simple line interface calculation) [for multifluid dynamics] ». In *Proceedings of the 5th International Conference on Numerical Methods in Fluid Dynamics, 28 June-3 July 1976*. (Berlin, West Germany), p. 330-40.

Coll. « Proceedings of the 5th International Conference on Numerical Methods in Fluid Dynamics »: Springer-Verlag.

Oden, J. Tinsley, Ivo Babuŝka et Carlos Erik Baumann. 1998. « A DiscontinuoushpFinite Element Method for Diffusion Problems ». *Journal of Computational Physics,* vol. 146, nº 2, p. 491-519.

Olsson, E., et G. Kreiss. 2005. « A conservative level set method for two phase flow ». *Journal of Computational Physics,* vol. 210, nº 1, p. 225-246.

Olsson, E., G. Kreiss et S. Zahedi. 2007. « A conservative level set method for two phase flow II ». *Journal of Computational Physics,* vol. 225, nº 1, p. 785-807.

Osher, Stanley, et James A. Sethian. 1988. « Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations ». *Journal of Computational Physics,* vol. 79, nº 1, p. 12-49.

Osher, Stanly, et Ronald Fedkiw. 2003. *Level set methods and dynamic implicit surfaces*, 153. Coll. « Applied Mathematical Sciences ». New York: Springer, 273 p.

Owkes, Mark, et Olivier Desjardins. 2013. « A discontinuous Galerkin conservative level set scheme for interface capturing in multiphase flows ». *Journal of Computational Physics,* vol. 249, p. 275-302.

Peng, Danping, Barry Merriman, Stanley Osher, Hongkai Zhao et Myungjoo Kang. 1999. « A PDE-Based Fast Local Level Set Method ». *Journal of Computational Physics,* vol. 155, nº 2, p. 410-438.

Pironneau, O. 1982. « On the transport-diffusion algorithm and its applications to the Navier-Stokes equations ». *Numerische Mathematik,* vol. 38, nº 3, p. 309-332.

Prosperetti, A., et G. Tryggvason. 2009. *Computational methods for multiphase flows*. Coll. « International Journal of Multiphase Flow ». New York: Cambridge University Press, 488 p.

Qian, Jianliang, Yong-Tao Zhang et Hong-Kai Zhao. 2007. « Fast sweeping methods for eikonal equations on triangular meshes ». *SIAM Journal on Numerical Analysis,* vol. 45, nº 1, p. 83-107.

Rasthofer, U., F. Henke, W. A. Wall et V. Gravemeier. 2011. « An extended residual-based variational multiscale method for two-phase flow including surface tension ». *Computer Methods in Applied Mechanics and Engineering,* vol. 200, nº 21–22, p. 1866-1876.

Reusken, Arnold. 2008. « Analysis of an extended pressure finite element space for two-phase incompressible flows ». *Computing and Visualization in Science,* vol. 11, nº 4-6, p. 293-305.

Reusken, Arnold, et Eva Loch. 2011. *On the accuracy of the Level Set SUPG method for approximating interfaces*. Aachen. < http://publications.rwth-aachen.de/record/47338 >.

Rider, William J., et Douglas B. Kothe. 1998. « Reconstructing Volume Tracking ». *Journal of Computational Physics,* vol. 141, nº 2, p. 112-152.

Sauerland, Henning. 2013. « An XFEM based sharp interface approach for two-phase and free-surface flows ». PhD thesis. Aachen University, 141 p.

Sauerland, Henning, et Thomas-Peter Fries. 2011. « The extended finite element method for two-phase and free-surface flows: A systematic study ». *J. Comput. Phys.,* vol. 230, nº 9, p. 3369-3390.

Sauerland, Henning, et Thomas-Peter Fries. 2013. « The stable XFEM for two-phase flows ». *Computers & Fluids,* vol. 87, nº 0, p. 41-49.

Schott, B., U. Rasthofer, V. Gravemeier et W. A. Wall. 2014. « A face-oriented stabilized Nitsche-type extended variational multiscale method for incompressible two-phase flow ». *International Journal for Numerical Methods in Engineering,* vol. 104, nº 7, p. 721-748.

Sethian, J. A. 1996. « Theory, algorithms, and applications of level set methods for propagating interfaces ». *Acta Numerica,* vol. 5, p. 309-95.

Sethian, J.A. 1998. *Level Set Methods and Fast Marching Methods – Evolving Interfaces in Computational Geometry, Fluid Mechanics*. Coll. « Computer Vision, and Materials Science ». Cambridge, UK: Cambridge University Press.

Smolianski, Anton. 2001. « Numerical Modeling of Two-Fluid Interfacial Flows ». Jyväskylä, University of Jyväskylä, Jyväskylä, Filand, 109 p. < http://user.math.uzh.ch/smolianski/pdf/phd_thesis.pdf >.

So, K. K., X. Y. Hu et N. A. Adams. 2011. « Anti-diffusion method for interface steepening in two-phase incompressible flow ». *Journal of Computational Physics,* vol. 230, nº 13, p. 5155-5177.

Sussman, M. 2003. « A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles ». *Journal of Computational Physics,* vol. 187, nº 1, p. 110-136.

Sussman, M., et E. G. Puckett. 2000. « A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows ». *Journal of Computational Physics,* vol. 162, nº 2, p. 301-337.

Sussman, Mark, Peter Smereka et Stanley Osher. 1994. « A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow ». *Journal of Computational Physics,* vol. 114, nº 1, p. 146-159.

Touré, Mamadou Kabirou, et Azzeddine Soulaïmani. 2012. « A level set method without re-initialization ». In *Conference of the CFD Society of Canada*. (Canmore, Alberta), p. 7.

Touré, Mamadou Kabirou, et Azzeddine Soulaïmani. 2016. « Stabilized finite element methods for solving the level set equation without renitialization ». *Computers & Mathematics with Applications*.

Turek, Stefan. 1997. « On discrete projection methods for the incompressible Navier-Stokes equations: an algorithmical approach ». *Computer Methods in Applied Mechanics and Engineering,* vol. 143, nº 3–4, p. 271-288.

Turek, Stefan. 1999. *Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach*. Coll. « Lecture Notes in Computational Science and Engineering », 6. Springer-Verlag Berlin Heidelberg.

Ubbink, Onno. 1997. « Numerical Prediction of Two-Fluid Systems with Sharp Interfaces ». PhD Thesis. Imperial College of Science, Technology & Medicine, London University, 69 p.

Unverdi, S. O., et G. Tryggvason. 1992. « A front-tracking method for viscous incompressible multi-fluid flows ». *Journal of Computational Physics,* vol. 100, nº 1, p. 25-37.

van der Pijl, S. P., A. Segal, C. Vuik et P. Wesseling. 2005. « A mass-conserving Level-Set method for modelling of multi-phase flows ». *International Journal for Numerical Methods in Fluids,* vol. 47, nº 4, p. 339-361.

Vincent, S., D. Lakehal et H. Friess. 2004. « Test-case N# 27: interface tracking based on an imposed velocity field in a convergent-divergent channel (PN) ». vol. 16, nº 1-3, p. 165-170.

Wang, Y., S. Simakhina et M. Sussman. 2012. « A hybrid level set-volume constraint method for incompressible two-phase flow ». *Journal of Computational Physics,* vol. 231, nº 19, p. 6438-6471.

Youngs, D. L. 1982. « Time dependent multimaterial flow with large fluid distortion ». *Numerical Methods for Fluid Dynamics* (New York). p. 27.

Zalesak, Steven T. 1979. « Fully multidimensional flux-corrected transport algorithms for fluids ». *Journal of Computational Physics,* vol. 31, nº 3, p. 335-362.

Zhao, H. K. 2005. « A fast sweeping method for Eikonal equations ». *Mathematics of Computation,* vol. 74, nº 250, p. 603-627.

Zienkiewicz, O.C., et Y.K. Cheung. 1965. « Finite elements in the solution of field problems ». *The Engineer,* vol. 507-10 edition.

Zlotnik, Sergio, et Pedro Díez. 2009. « Hierarchical X-FEM for n-phase flow ». *Computer Methods in Applied Mechanics and Engineering,* vol. 198, nº 30–32, p. 2329-2338.