

Navier-Stokes solvers for incompressible single-and two-phase flows

Mohamed El Ouafa, Vincent Stéphane, Vincent Le Chenadec

▶ **To cite this version:**

Mohamed El Ouafa, Vincent Stéphane, Vincent Le Chenadec. Navier-Stokes solvers for incompressible single-and two-phase flows. 2020. hal-02892344

HAL Id: hal-02892344
https://hal.science/hal-02892344

Preprint submitted on 7 Jul 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Navier-Stokes solvers for incompressible single- and two-phase flows

Mohamed El Ouafa*, Vincent Stephane*, Vincent Le Chenadec*

* MSME, Univ. Gustave Eiffel, UMR CNRS 8208, Univ. Paris-Est Créteil,
5 bd Descartes, 77454 Marne-la-Vallée, France,
email: mohamed.elouafa@u-pem.fr

July 7, 2020

Abstract

The presented work is dedicated to the mathematical and numerical modeling of unsteady single- and two-phase flows using finite volume and penalty methods. Two classes of Navier-Stokes solvers are considered in order to compare their accuracy and robustness, as well as to highlight their limitations. Exact (or monolythic) solvers such as the Augmented Lagrangian and the Fully Coupled methods address the saddle-point structure on the pressure-velocity couple of the discretized system by means of a penalization term or even directly, whereas the approximate (or segregated) solvers such as the Standard Projection method rely on operator splitting to break the problem down into decoupled systems. The objective is to compare all approaches in the context of two-phase flows at high viscosity and density ratios. To characterize the interface location, a volume of fluid (VOF) approach is used based on a Piecewise Linear Interface Construction (PLIC). Various 2D simulations are performed on single- and two-phase flows to characterize the behavior and performances of the various solvers.

Keywords—
Monolythic solvers, Augmented Lagrangian, two-phase flows, saddle point, projection method

1 Introduction

The Navier-Stokes equations play a crucial role in several industrial and environmental applications such as weather forecast, combustion and propulsion, civil engineering, aerodynamics and hydrodynamics to name a few. However, their resolution causes many problems and remains a challenging task for mathematicians as well as engineers. In the context of incompressible two-phase flows, the difficulties are two-fold: on the one hand, the velocity-pressure coupling induced by the incompressibility constraint, which must hold at every instant, gives rise to a saddle-point problem for which dedicated solvers must be designed and, on the other hand, large density and viscosity ratios in the presence of large interface distortions yield ill-conditioned linear systems. However, solving ill-conditioned linear systems is common in many engineering problems (the reader is referred to [8, 6] for more information).

The cures to the pressure-velocity coupling challenge fall along one of the two following strategies. On the one hand, segregated methods, introduced by Chorin [7] and Temam [33], approximate the original system using time splitting, thereby resulting in two decoupled equations: one to update the velocity field and the other the pressure field. This raises a number of issues, such as the appearance of numerical artifacts as a consequence of the splitting error, or the need for pressure boundary conditions that do not exist in the original problem. On the other hand, coupled methods solve both fields (velocity and pressure) simultaneously, hence preserving the consistency of the discretized system with the continuous equations (as opposed to segregated methods, which introduce a splitting error that is inconsistent with the continuous system). The system to be solved in coupled methods is known as a saddle point system on the pressure-velocity couple. This saddle point system can in turn be solved either directly (this strategy will be referred to as the Fully Coupled (FC) method) or by means of a penalty method (Augmented Lagrangian (AL)). The Augmented Lagrangian (AL) method was proposed initially by Fortin and Glowinski [13] primarily for single-phase Stokes flows. This approach was recently applied to two-phase flows by Vincent and coworkers [21, 35]. This is an iterative method, where each iteration requires the

solution of the velocity equation modified to include an additional term that penalizes the incompressibility constraint. A subsequent update of the pressure field is performed explicitly. Fully Coupled (FC) methods on the other hand are purely algebraic, using specific block preconditionners so as to improve the spectral properties of the original saddle point system. In recent years, significant efforts have been dedicated to this approach, and several techniques have been proposed in the context of single-phase flows, in the Stokes or steady regimes primarily, using finite element methods [23, 10].

In the context of two-phase flow modeling at high density and viscosity ratios, several techniques have been investigated to tackle the induced stiffness. For projection methods (PR), Guermond and Salgado [17] have studied a new fractional time-step technique for variable density flows, which consist in extracting the density from the Poisson equation by penalizing the divergence of the velocity. The validity of this method is demonstrated for Rayleigh-Taylor instability at small density ratio. More recently, Dodd and Ferrante [9] have proposed a fast pressure-correction method for simulating incompressible two-phase flows with large density and viscosity ratios. The method is based on the splitting of the pressure gradient into two terms, one with a variable density term which is explicitly treated and the other one with a constant density term treated implicitly. The results obtained for a capillary wave show that this method is able to treat two-phase flow problems with density and viscosity ratios up to 10^5. Nevertheless, the explicit treatment of the velocity prediction imposes a restriction on the time-step size to maintain numerical stability. Another approach followed by Caltagirone and Breil [5] relies on a vector projection of the velocity field, using only a velocity correction and Hodge-Helmholtz decomposition. Three-dimensional tests show that this method is very effective for simulating two-phase flows with very large ratios of density and viscosity ($\rho_2/\rho_1 > 1000$ and $\mu_2/\mu_1 > 1000$). However, the solution of large sparse linear systems on all velocity components is complex, and as in the scalar projection method, it also has a time spliting error.

Regarding the Augmented Lagrangian method, Vincent et al. [35] have proposed the Algebraic Adaptive Augmented Lagrangian (AAAL) for simulating two-phase flows. As opposed to the Standard Augmented Lagrangian (SAL), where the penalty parameter is constant, AAAL automates the parameter selection locally by analysing the matrix arising from the discretization of the penalized velocity equation. The robustness of this method has been demonstrated by simulating the free fall of a dense cylinder, with large density and viscosity ratios (10^5 and 10^{10}, respectively). Although this method is very effective for two-phase flows with large contrasts, it is computationally expensive, because the linear systems to be solved are ill-conditioned and the only preconditioner identified by the authors as sufficiently robust (ILU [30]) does not scale well in parallel implementation with distributed memory. For the Fully Coupled method, studies on two-phase flows have been documented as well. In the case of steady Stokes motions, Cai et al. [25] have investigated the rise of a bubble at high contrasts ($r_\rho = r_\mu = 10^3$). For unsteady Navier-Stokes equations, Bootland et al. [2] have proposed a new preconditionner (termed Pressure Convection-Diffusion, or PCD). Numerical results for a dynamic dam-break problem (air-water) showed that the PCD can be an effective approach for two-phase flows. A distinctive feature in the aforementioned Fully Coupled methods is that they all rely on specific block-triangular preconditionners.

The main objective of the present work is to further investigate Fully Coupled methods, the robustness of which can ultimately unlock new applications such as the simulation of two-phase flows at large density and viscosity ratios. The challenge effectively lies at the algebraic level, and the poor conditioning of the linear systems must be addressed in order to challenge the widespread use of segregated methods. In this context, the main objective of the current study is to review robust parallel preconditioners able to cope with these systems, and compare their performance with well-established alternatives. The paper is organized as follows: in Sec. 2 the Naviers-Stokes equations are formulated for two-phase incompressible flows, and the temporal and spatial discretizations are presented. The solution strategies used in our simulations are delineated in Sec. 3, with a particular emphasis for the block preconditionners used on the saddle point systems. Sec. 4 is dedicated to numerical results for single- and two-phase flows. This section also provides detailed comparisons with respect to various documented solvers for incompressible flows.

2 Governing equations

The one-fluid formulation, developped by Kataoka [22] for multiphase flows, is considered. In its incompressible form, the one-fluid model consists of the Navier-Stokes equation augmented with the capillary effect, supplemented by the divergence-free condition and a transport equation for a color function, denoted C, used to characterize the interface location and the fluid properties (mass density and viscosity). In the proposed work, C will be chosen as the volume fraction occupied by a reference phase. The

originality of the one-fluid model is that it takes implicitly into account the jumps relations across the interface induced by surface tension, which revertes to the classical single phase Navier-Stokes equations as soon as $C = 1$ or 0 everywhere.

Let $\Omega \subset \mathbb{R}^d$ be an open bounded domain, with continuous boundary $\Gamma = \partial\Omega$ and $\mathbf{n}$ the unit normal to Γ. The one-fluid model equations governing the multiphase flow motion read:

$$\frac{\partial \rho(C)\mathbf{u}}{\partial t} + \nabla \cdot (\rho(C)\mathbf{u} \otimes \mathbf{u}) = -\nabla p + \nabla \cdot [\mu(C)(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \rho(C)\mathbf{g} + \mathbf{F}_s, \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2}$$

$$\frac{\partial C}{\partial t} + \mathbf{u} \cdot \nabla C = 0. \tag{3}$$

Eq. 1 represents the conservation of linear momentum, Eq. 2 the conservation of volume (the phases are immiscible and incompressible) and Eq. 3 the advection of color function C. In Eqs. 1, 2 and 3, $\mathbf{u}$ is the velocity field, p the pressure field, t the time, $\mathbf{g}$ the gravity field, ρ and μ are the mixture density and dynamic viscosity, respectively, which depend on C, and $\mathbf{F}_s$ is the capillary term, modeled in this study by the continuum surface tension force model [3]. The mixture density and viscosity are defined according to the following constitutive laws (arithmetic average):

$$\rho(C) = \rho_1 C + (1 - C)\rho_2, \tag{4}$$

$$\mu(C) = \mu_1 C + (1 - C)\mu_2. \tag{5}$$

where ρ_1, ρ_2, μ_1 and μ_2 are the densities and the viscosities of fluid 1 and 2. By mixture density or viscosity, it is understood that the aforementioned material properties are phase-wise constant. C (and ρ and μ, therefore) vary spatially in the interface vicinity only, that is in the cells or control volumes cut by the interface.

The current study focuses on solution techniques for the incompressible two-phase Navier-Stokes equations 1 and 2. The solution of the advection equation 3 is performed with by the Volume-of-Fluid (VOF) method, using the Piecewise Linear Interface Construction (PLIC) technique [36]. The spatial discretization of the Navier-Stokes equations 1 and 2 is performed on a staggered mesh according to the MAC scheme of Harlow and Welch [18]. This mesh guarantees the consistency of the differential operators such as the divergence and the gradient, it also avoids oscillations on the pressure field. A centered approximation of the convective term alleviates any numerical viscosity. The resulting semi-discrete system is advanced in time implicitly by means of a second order integration scheme (Gear, Eq. 6):

$$\left(\frac{\partial \mathbf{u}}{\partial t}\right)^{n+1} = \frac{\frac{3}{2}\mathbf{u}^{n+1} - 2\mathbf{u}^n + \frac{1}{2}\mathbf{u}^{n-1}}{\Delta t}, \tag{6}$$

and the convective terms are linearized using Adams-Bashforth extrapolation (Eq. 7):

$$\mathbf{u}^{n+1} \otimes \mathbf{u}^{n+1} \approx \left(2\mathbf{u}^n - \mathbf{u}^{n-1}\right) \otimes \mathbf{u}^{n+1}. \tag{7}$$

Here, n refers to a time index associated to physical time $n\Delta t$ and Δt is the assumed constant time step of each solved iteration.

2.1 Physical boundary conditions

To complete the one-fluid formulation, the conditions for the velocity field $\mathbf{u}$ at the boundary $\Gamma = \partial\Omega$ must be specified. As an implicit Fully Coupled (FC) resolution is targeted, a penalty method is considered [28]. This method allows to treat different types of boundary conditions, such as Dirichlet and Neumann conditions, in a natural way. Numerically, this method is based on a control term $\overline{\overline{B}}_{penu}\left(f(\mathbf{u}) - \mathbf{u}_\infty\right)$, which is added to the momuntum equation:

$$NS(\mathbf{u}^{n+1}) + \overline{\overline{B}}_{penu}\left(f(\mathbf{u}^{n+1}) - \mathbf{u}_\infty\right) = 0 \tag{8}$$

where

$$\overline{\overline{B}}_{penu} = \begin{pmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{pmatrix} \tag{9}$$

is a tensor field, whose diagonal components tend to infinity along the boundary Γ and are identically zero inside the fluid domain Ω. Here, $f(\mathbf{u}^{n+1})$ (Eq. 10 and Eq. 11) is a discrete function of u^{n+1} and v^{n+1}, which is written as a linear combination of resolved velocities $u_{i,j}^{n+1}$, $v_{i,j}^{n+1}$ and their neighbors:

$$f(u) = a_0 u_{i,j}^{n+1} + a_1 u_{i-1,j}^{n+1} + a_2 u_{i+1,j}^{n+1} + a_3 u_{i,j-1}^{n+1} + a_4 u_{i,j+1}^{n+1}, \tag{10}$$

$$f(v) = a_0 v_{i,j}^{n+1} + a_1 v_{i-1,j}^{n+1} + a_2 v_{i+1,j}^{n+1} + a_v u_{i,j-1}^{n+1} + a_4 v_{i,j+1}^{n+1}. \tag{11}$$

The treatment of the Dirichlet boundary conditions, applied to the left boundary for exemple (see Fig. 1), is controlled by the coefficients a_i which must be fixed at the following values: $a_0 = a_2 = \frac{1}{2}$ and $a_1 = a_3 = a_4 = 0$. Thus, the discrete momuntum equation on the left boundary becomes:

$$NS(u_{i,j}^{n+1}) + \alpha_u \left(\frac{1}{2} u_{0,j}^{n+1} + \frac{1}{2} u_{1,j}^{n+1} - u_{\infty \frac{1}{2},j} \right) = 0. \tag{12}$$

To impose homogeneous Neumann boundary conditions, the coefficients have to be taken as $a_0 = -1$, $a_2 = 1$ and $a_1 = a_3 = a_4 = 0$. The resulting equation in discrete form is:

$$NS(u_{i,j}^{n+1}) + \alpha_u \left(-u_{0,j}^{n+1} + u_{1,j}^{n+1} - u_{\infty \frac{1}{2},j} \right) = 0. \tag{13}$$

with $u_{\infty \frac{1}{2},j} = 0$.

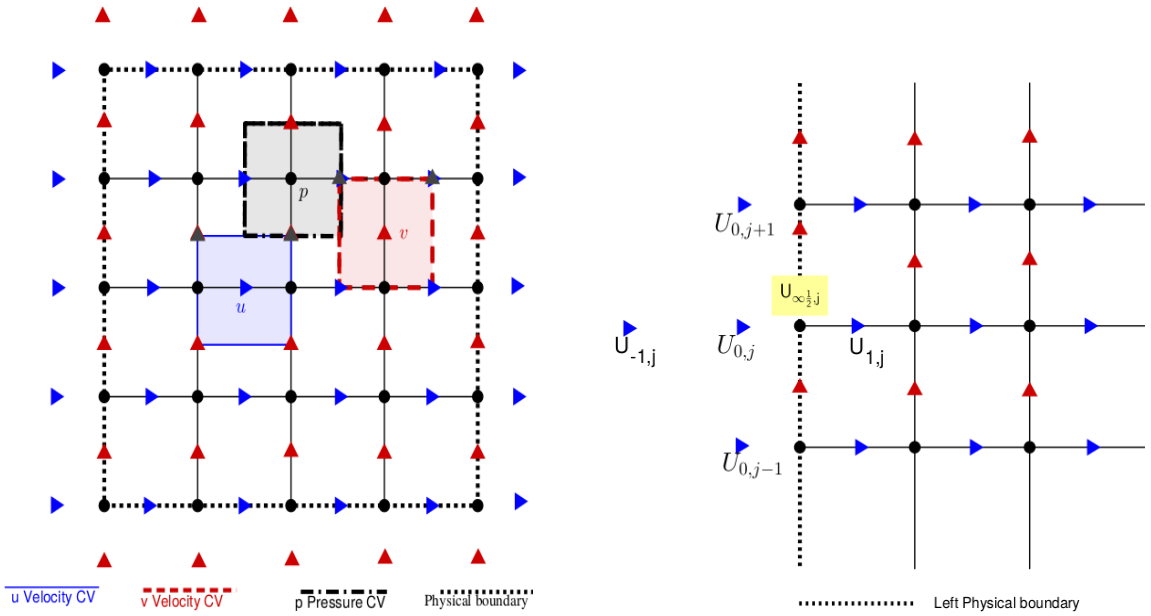


Figure 1: Left: staggered pressure and velocity unknowns in the global domain and right, zoom on a left boundary in 2D

The time-discretized system of the Fully Coupled formulation, referred to as the penalized one-fluid model, finally reads:

$$\frac{\rho(C^{n+1})}{\Delta t} \left(\frac{3}{2} \mathbf{u}^{n+1} - 2\mathbf{u}^n + \frac{1}{2} \mathbf{u}^{n-1} \right) + \nabla \cdot \left(\rho(C^{n+1}) \left(2\mathbf{u}^n - \mathbf{u}^{n-1} \right) \otimes \mathbf{u}^{n+1} \right)$$
$$+ \overline{\overline{B}}_{penu} \left(f(\mathbf{u}^{n+1}) - \mathbf{u}_\infty \right) = -\nabla p^{n+1} + \nabla \cdot [\mu(C^{n+1})(\nabla \mathbf{u}^{n+1} + \left(\nabla \mathbf{u}^{n+1} \right)^T)] + \rho(C^{n+1}) \mathbf{g} + \mathbf{F}_s, \tag{14}$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0, \tag{15}$$

$$\frac{C^{n+1} - C^n}{\Delta t} + \mathbf{u}^n \cdot \nabla C^n = 0. \tag{16}$$

3 Numerical Methods

As mentioned in the introduction, different approaches have been used to solve the Navier-stokes equations. In the following, the numerical methods utilized for solving the motion equations in the context of two-phase flow modeling are presented.

3.1 Projection method

The litterature on projection methods is vast and many variants and formulations exist. In the present work, the projection method proposed by Goda [15] is considered. It keeps the pressure gradient in the prediction step. As opposed to the original work of Goda, where the prediction step is performed explicitly, an implicit resolution is chosen here so as to overcome the time-step restrictions to guarantee numerical stability. The projection method essentially decouples velocity and pressure fields, and the resulting algorithm consists of two steps: first, a velocity prediction is performed, where $\mathbf{u}^*$ is computed as the solution of a decoupled momentum equation.

$$\frac{\rho(C^{n+1})\left(\frac{3}{2}\mathbf{u}^* - 2\mathbf{u}^n + \frac{1}{2}\mathbf{u}^{n-1}\right)}{\Delta t} + \nabla\cdot\left(\rho(C^{n+1})\left(2\mathbf{u}^n - \mathbf{u}^{n-1}\right)\otimes\mathbf{u}^*\right) + \overline{\overline{B}}_{penu}\left(f(\mathbf{u}^*) - \mathbf{u}_\infty\right) = -\nabla p^n + \nabla\cdot\left[\mu(C^{n+1})(\nabla\mathbf{u}^* + \nabla\mathbf{u}^{*T})\right] + \rho(C^{n+1})\mathbf{g} + \mathbf{F}_s \quad (17)$$

The predicted velocity however does not satisfy the divergence-free condition Eq. 15. In the second step, therefore, the intermediate velocity is corrected by projecting it on a divergence-free subspace. This step is performed by computing a pressure increment p^* as the solution of the following Helmholtz equation, with homogeneous Neumann boundary conditions:

$$\nabla\cdot\left(\frac{\Delta t}{\rho(C)}\nabla p^*\right) + \alpha_p\left(f(p^*) - p_\infty\right) = \nabla\cdot\mathbf{u}^* \quad (18)$$

where the penalisation term α_p tends to infinity as Γ is approached, and zero in Ω. The velocity and pressure updates are performed as follows:

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\Delta t}{\rho(C)}\nabla p^*, \quad (19)$$

$$p^{n+1} = p^n + p^*. \quad (20)$$

The appeal for the projection method stems from its simplicity and ease of implementation. However, this approach suffers from well-known limitations: the splitting operator introduces an error that reduces the temporal accuracy of the numerical solution, and introduces the need for appropriate boundary conditions for the pressure field in the projection step which do not exist in the original (coupled) system. Indeed, the inconsistency of the boundary conditions induces numerical errors in the pressure field, giving rise to artificial boundary layers as described in [16]. Additionally, when multiphase flows are simulated with large density ratios, the challenging task is to solve the matrix system arising from the disretization of the variable coefficient Helmholtz Eq. 18, which become ill-conditionned due to discontinuities in the Laplacian operator $\nabla\cdot\left(\frac{\Delta t}{\rho(C)}\nabla p^*\right)$.

3.2 Augmented Lagrangian method

To alleaviate the difficulties related to the projection method, an exact method called Augmented Lagrangian (AL) has been proposed. Originally developped by Fortin and Glowinski [13] for single-phase Stokes flows, this method has been extended recently for two-phase flows [21, 35] to take advantage of its robustness and efficiency. The principle of this method is that the conservation equation of mass is not used in its original form, but rather transformed into an explicit pressure Eq. 22 that will be coupled with the velocity. The resolution consists in finding a saddle point on the couple $\left(\mathbf{u}^{n+1}, p^{n+1}\right)$. Starting with $\mathbf{u}^{*,0} = \mathbf{u}^n$ and $p^{*,0} = p^n$, the solution is given by an Uzawa algorithm [21]: while $|\nabla\cdot\mathbf{u}^{*,m}| > \epsilon$, solve

$$\frac{\rho(C)\left(\frac{3}{2}\mathbf{u}^{*,m} - 2\mathbf{u}^{*,0} + \frac{1}{2}\mathbf{u}^{*,-1}\right)}{\Delta t} + \nabla\cdot\left(\rho(C)\left(2\mathbf{u}^{*,0} - \mathbf{u}^{*,-1}\right)\otimes\mathbf{u}^{*,m}\right) + \overline{\overline{B}}_{penu}\left(f(\mathbf{u}^{*,m}) - \mathbf{u}_\infty\right) = -\nabla p^{*,0} + r\nabla\left(\nabla\cdot\mathbf{u}^{*,m}\right) + \nabla\cdot\left[\mu(C)(\nabla\mathbf{u}^{*,m} + \nabla\mathbf{u}^{(*,m)T})\right] + \rho\mathbf{g} + \mathbf{F}_s, \quad (21)$$

$$p^{*,m} = p^{*,0} - r\nabla\cdot\mathbf{u}^{*,m} \quad (22)$$

where ϵ is the divergence threshold and m is the iteration index. So for $m = 1, 2\cdots n$, the iterative process simultaneously corrects the pressure and the velocity until convergence to the solution $\left(\mathbf{u}^{n+1}, p^{n+1}\right)$. In practice, one iterative step of the Augmented Lagrangian (AL) with a suitable value of the parameter r

suffices to obtain good solution. A proper choice of parameter r should guarantee that the incompressiblity constraint Eq. 15 is satisfied, while simultaneously ensuring that the magnitude of the Augmented Lagrangian (AL) term compares with the largest of the terms in the Navier-Stokes equations. Taking very low values of r, the importance is given to the Navier-Stokes equations whereas when r is very large, the velocity field will be divergence-free but will not satisfy the Navier-Stokes equations. In practice, values of r between 10 to 1000 times larger than the other dimensionless terms yield satisfactory results. The reader is referred to recent work on the automation of its computation for more information [34].

The Augmented Lagrangian (AL) method has many advantages. On the one hand, it avoids imposing boundary conditions on the pressure field. On the other hand, using the Augmented Lagrangian (AL) method, two-phase flows at high density and viscosity ratio ($\frac{\rho_2}{\rho_1} \ll 1$ or $\frac{\rho_2}{\rho_1} \gg 1$ as well as $\frac{\mu_2}{\mu_1} \ll 1$ or $\frac{\mu_2}{\mu_1} \gg 1$) can be simulated, as opposed to the ranges achievable with the Standard Projection (PR) method. However, the disadvantage of this method is its numerical expense, as a result of the effect of parameter r. Indeed, the simulation of two-phase flows with high density and viscosity ratios, together with large values of the parameter r, leads to ill-conditioned linear systems. In addition, it has to be highlighted that the numerical solution strongly depends on the choice of the parameter r, itself largely sensitive to problem parameters such as the density and viscosity ratios, and grid sizes.

3.3 Fully Coupled method

To date, the Fully Coupled (FC) method, although widely used in the finite element community in the context of stationary and single-phase flow simulations, has not found much interest from the finite difference and finite volume communities. The objective of this study is to investigate Fully Coupled approaches in the context of unsteady two-phase flow simulated by finite volumes and penalty methods. They require the solution of systems of the following form:

$$\begin{pmatrix} \frac{1}{\Delta t}\, M_u^{(\rho)} + N_u^{(\rho)} + \ L_u^{(\mu)} & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ 0 \end{pmatrix} \tag{23}$$

or $A\mathbf{x} = \mathbf{b}$. Eq. 23 represents a non-symmetric saddle-point problem, where the matrix B_u is the discrete negative divergence and B_p^T represents to the discrete pressure gradient operator, $M_u^{(\rho)}$ is the velocity mass matrix, $L_u^{(\mu)}$ is the discrete velocity Laplacian, $N_u^{(\rho)}$ denotes the convective matrix and $\mathbf{f}$ represents the right-hand side vector.

To solve this sparse linear system, direct methods such as Gaussian elimination are very robust with respect to various problem but do not scale well with problem size and are impossible to use in three space dimensions. In the context of large-scale systems, methods of choice include iterative solution techniques, such as the generalized minimal residual (GMRES [31]), algebric or geometric multigrid methods [32] and the conjugate gradient and its variants [20]. In this study, the BiCGStab(2) algorithm [20] was found to perform slightly bettter than $GMRES$ (10% on average) for both Fully Coupled (FC) and augmented Lagrangian (AL) systems, and was therefore preferred in spite of the favorable monotonous convergence property of GMRES that BiCGStab(2) does not share.

Because the performances of the iterative methods depends on the conditioning of the system, the use of an efficient preconditioning proves necessary to accelerate the convergence of the iterative solver and to avoid possible numerical instabilities. The preconditioning can be considered as a transformation of the original system $A\mathbf{x} = \mathbf{b}$ into an equivalent one $P^{-1}A\mathbf{x} = P^{-1}\mathbf{b}$, where the preconditioning matrix P is an approximation of A ($P \approx A$) whose inverse can be efficiently applied.

In order to build a suitable preconditionner P, we consider the block LDU decomposition of the original matrix A, given by:

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} = \begin{pmatrix} I_u & 0 \\ B_u F_u^{-1} & I_p \end{pmatrix} \begin{pmatrix} F_u & 0 \\ 0 & S_u \end{pmatrix} \begin{pmatrix} I_u & F_u^{-1} B_p^T \\ 0 & I_p \end{pmatrix} \tag{24}$$

Here $S_u = -B_u F_u^{-1} B_p^T$ is the Schur complement of the pressure block and $F_u = \frac{1}{\Delta t}\, M_u^{(\rho)} + N_u^{(\rho)} + \ L_u^{(\mu)}$ is the convection-diffusion-reaction operator. In this work, one of the most popular block triangular preconditioner for saddle point problems is used, by taking the DU product of Eq. 24 :

$$P = \begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix} \tag{25}$$

Thus, the application of the preconditioner of Eq. 25 entails solving the following system of equations by backward substitution:

$$\begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix} \begin{pmatrix} \mathbf{z}_u \\ z_p \end{pmatrix} = \begin{pmatrix} \mathbf{r}_u \\ r_p \end{pmatrix} \tag{26}$$

Algorithm Triangular preconditionner

1: Solve $S_p z_p = r_p$.
2: Update $\mathbf{r}_u \longleftarrow \mathbf{r}_u - B_p^T z_p$.
3: Solve $F_u \mathbf{z}_u = \mathbf{r}_u$.

where $\begin{pmatrix} \mathbf{r}_u \\ r_p \end{pmatrix} = \mathbf{b}$ and $\mathbf{p}$ and $\begin{pmatrix} \mathbf{r}_u \\ r_p \end{pmatrix} = \mathbf{b}, \mathbf{v}^k, \mathbf{s}^k, \mathbf{t}^k$ and $\mathbf{w}^k$ are a sequence of vectors generated at each BiCGSTAB(2) (for further details, see Appendix A).

When the triangular preconditionner P^{-1} is applied to A, the preconditionned system $P^{-1}A$ has a single eigenvalue $\lambda = 1$ of multiplicity $n_u + n_p$. The characteristic polynomial of $P^{-1}A$ has degree 2, which guarantees that a Krylov subspace solver such as GMRES converges in two iterations. Indeed, by considering the eigenvalue problem for the preconditionned system $P^{-1}A$, this values can easily be obtained from the following generalized eigenvalue problem:

$$\begin{pmatrix} F_u & B_p^T \\ B_u & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} = \lambda \begin{pmatrix} F_u & B_p^T \\ 0 & S_p \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} \tag{27}$$

From the first line of equation of 27 we obtain $(1-\lambda)(F_u\mathbf{u} + B_p^T p) = 0$, admitting two possible solutions. The first is eignvalue $\lambda = 1$ of multiplicity n_u, and the second corresponds to $\mathbf{u} = -F_u^{-1}B_p^T p$. Then, the substitution of $\mathbf{u}$ in the second row of 27 leads to $-B_u F_u^{-1} B_p^T p = \lambda S_p p$, wich gives the second eigenvalue $\lambda = 1$ of multiplicity n_p.

Likewise, if instead of the upper triangular preconditioner $P = DU$, the diagonal block preconditioner D is used, then it merges that $\lambda = 1$ is an eigenvalue of multiplicity at least $n_u - n_p$ for any eigenvector $\left[\mathbf{u}^T, \mathbf{0}^T\right]^T$ with $B\mathbf{u} = 0$, while $\lambda = \frac{1}{2} \pm \frac{\sqrt{5}}{2}$ of multiplicity n_p as described in [10]. As the convergence depends on the eigenvalues λ of the preconditioned system (the smaller the eigenvalue the faster the convergence), the block triangular preconditionner is expected to perform better and is therefore selected for this work.

This approach has however one major drawback, which is that the inversion of S_p and F_u can not be performed in realistic computations. They are indeed more expensive than solving the saddle point 23 by direct methods. As a result, they must be approximated by matrices $\hat{S}_p$ and $\hat{F}_u$.

Finding an effective approximation for the Schur complement S is not obvious. In recent years, a variety of preconditioners for incompressible flow problems have been introduced. These include the Cahouet-Chabard preconditioner [4] for homogeneous Stokes flows. This preconditionner has been extended [25] to two-phase flows in the case of steady Stokes flow. The Augmented Lagrangian formulation can also be used as a preconditioner, for example in the case of the Oseen problem [24] or the steady Stokes equations with variable viscosity [24]. Recently for the resolution of two-phase flows, Bootland and coworkers [2] have proposed two preconditioners called the pressure convection diffusion (PCD) preconditioner and the least squares commutator (LSC) preconditioner. For these preconditioners, the results show that the PCD preconditioner is more efficient than the LSC preconditioner. In this work, the pressure convection diffusion (PCD) preconditioning with suitable scaling is therefore chosen, the aforementioned study showing it to be the most effective at solving unsteady two-phase flows [2]. This preconditionner is given by:

$$\hat{S}_{PCD}^{-1} = (M_p^{1/\mu})^{-1} + (A_p^{1/\rho})^{-1}(\ N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)}\ )(M_p^{(1)})^{-1} \tag{28}$$

where $M_p^{1/\mu}$ is the diagonal pressure mass matrix scaled by the inverse of the viscosity $1/\mu$, $M_p^{(1)}$ is the diagonal standard pressure mass matrix, $N_p^{(1)}$ represents the standard convective matrix in the pressure space and $A_p^{1/\rho}$ is the scaled Laplacian, which corresponds to the discretisation of the term $\nabla \cdot \left(\frac{1}{\rho}\nabla P\right)$.

In order to approximate the inverse of the Schur complement $\hat{S}_{PCD}^{-1}$, the SMG (Semi Coarsening Multigrid) solver of the HYPRE library [12] is used, where the action of $A_p^{(1/\rho)}$ is required, as the multigrid solver is more appropriate for solving elliptic equations with variable coefficients. Then, the inverse of two diagonal mass matrix are solved, namely $(M_p^{(1/\mu)})^{-1}$ and $(M_p^{(1)})^{-1}$, by applying a rescaling to suitable vectors and a matrix-vector product for the operator $F_p^{(1)} = N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)}$. As described in the Work of Elman and al [11], the use of this preconditionner requires appropriate boundary conditions, which must be compatible with the physical boundary conditions applied to the velocity $\mathbf{u}$. Indeed, the

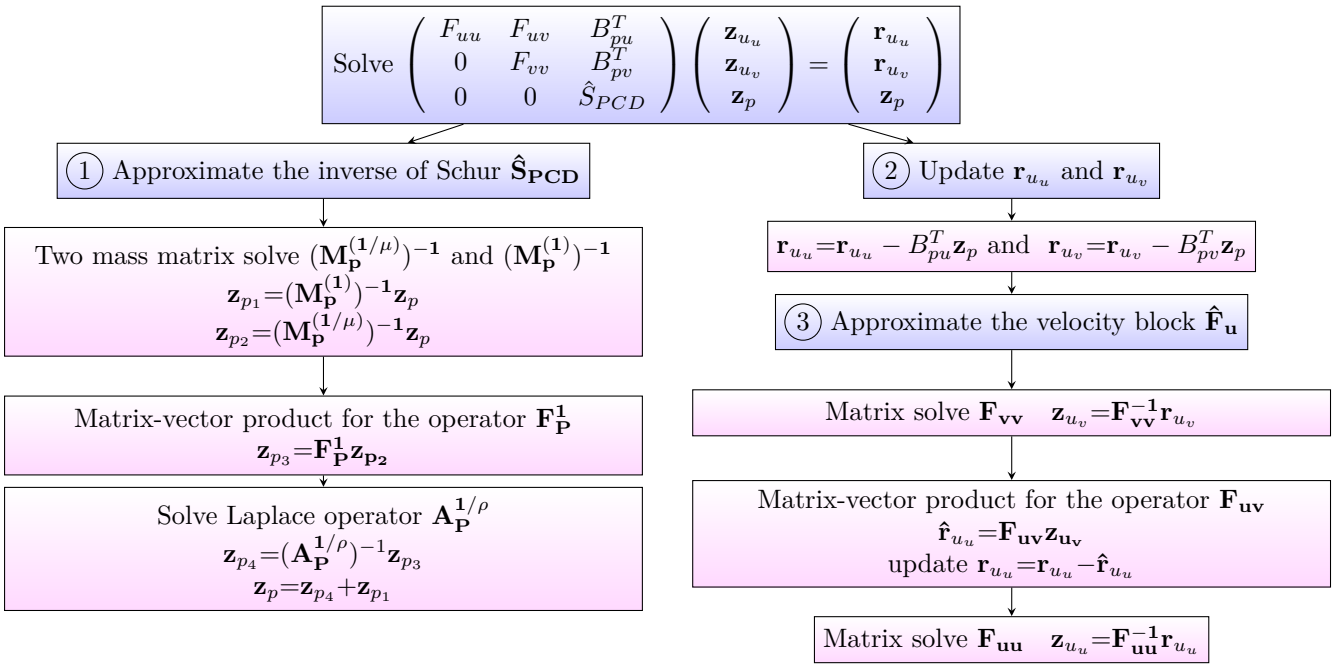


Figure 2: Graphical representation of the application of the block triangular preconditioner

choice of Dirichlet boundary conditions for the velocity $\mathbf{u}$, implies the application of Neumann boundary conditions for the discrete operators $A_p^{(1/\rho)}$ and $N_p^{(1)} + \frac{1}{\Delta t} M_p^{(1)}$. On the other hand, the use of inflow or outflow boundary conditions respectively for the velocity $\mathbf{u}$, results the use of Robin conditions in the case of inflow and Dirichlet or Neumann boundary (it depends on the Reynolds number) conditions in the case of outflow for the discrete operators of the Schur approximation $\hat{S}_{PCD}$.

Thus, to build a robust and efficient preconditionning P, a good approximation $\hat{F}_u$ is required. Many approximations of the velocity block F_u have been documented, for the most part based on the use of taylored multigrid implementations. These include the product splitting, block Jacobi, block Gauss-Seidel, block diagonal and triangular preconditioner (see [1] for more details). In the current study, the block Gauss-Seidel preconditioner

$$\hat{F}_u = \begin{pmatrix} F_{uu} & F_{uv} \\ 0 & F_{vv} \end{pmatrix}$$

was found to perform satisfactorily. This preconditioner requires the solution of two linear systems, F_{vv} and F_{uu}, which are both performed using the forementioned SMG. The algorithm applied to block triangular preconditionner is summarized in the graphical representation, shown in Fig. 2.

4 Numerical Results

4.1 Single phase flows: driven cavity

A first benchmark problem is considered, *i.e.* the driven cavity for viscous incompressible fluid flow, in a two-dimensional square domain of side $L = 1m$. In this configuration, all boundaries are walls, the top side moves at velocity $u_0 = -1m \cdot s^{-1}$, while the other three sides are no slip walls. For two Reynolds numbers $Re = 100$ and 1000, the results obtained with the Fully Coupled solver are compared with those proposed by Botella and Peyret [27]. Estimating the convergence rate is achieved by using a Richardson extrapolation (for further details, see Apprendix B). Concerning the numerical simulations, several meshes are used on uniform Cartesian grids, with time step constant and equal to $10^{-3}s$. As only the steady flow is considered here, the iterations are terminated when the stationnary criterion

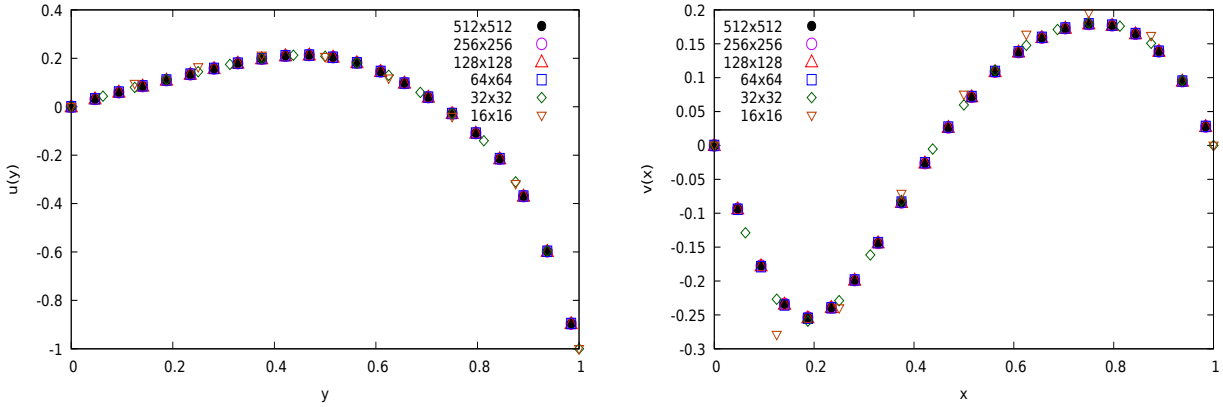
$$\max\left(\left\|\mathbf{u}^{n+1}-\mathbf{u}^n\right\|, \left\|\mathbf{v}^{n+1}-\mathbf{v}^n\right\|\right) \leq \epsilon \max\left(\left\|\mathbf{u}^{n+1}\right\|, \left\|\mathbf{v}^{n+1}\right\|\right) \tag{29}$$

is satisfied ($\epsilon = 10^{-6}$). The Reynolds number is defined as:

$$Re = \frac{\rho u_0 L}{\mu} \tag{30}$$

where ρ is the density, μ is the dynamic viscosity, L is the length of the cavity, and u_0 is the reference velocity of the flow.

Figs. 3a and 3b show the velocity plots along the middle-line of the cavity, obtained by the Fully Coupled method for $Re = 100$ with various uniform Cartesian meshes. It is clear from these figures that the results are qualitatively similar starting from the mesh 32×32 for both velocities components u and v. Additionally, Table 1 includes the numerical values corresponding to the maximum of the velocity u on the vertical centerline denoted u_{max}, the maximum of the velocity v on the horizontal centerline represents by v_{max}, the vorticity ω_{center}, the velocity u_{center} and v_{center} at the center of the cavity, compared to benchmark solution. As can be seen in Table 1, all computational results are in good agreement with those of Botella and Peyret [27], and the difference do not exceed 0.006%. The estimated accuracy for u_{center}, v_{center}, ω_{center}, u_{max} and v_{max} at the finest mesh are 2.00, 1.94, 2.05, 2.05 and 2.04, as expected given the second order accuracy of the spatially centered scheme.



(a) u-velocity on the vertical centerline at $x = 0.5$

(b) u-velocity on the horizontal centerline at $y = 0.5$

Figure 3: Lid driven cavity at $Re = 100$ - (comparison of the horizontal and vertical velocity along the centerline through the center of the cavity to results of Botella and Peyret [27])

N	u_{center}	order	v_{center}	order	ω_{center}	order	u_{max}	order	v_{max}	order
64	0.20876	1.39	0.05804	3.03	1.17746	5.01	0.21375	-	0.17991	3.72
128	0.20904	1.41	0.05767	2.34	1.17461	5.97	0.21395	1.65	0.17956	2.09
256	0.20912	1.85	0.05757	1.89	1.17446	1.29	0.21402	1.56	0.17959	2.10
512	0.20914	1.98	0.05754	1.82	1.17440	1.47	0.21403	2.53	0.17957	1.91
1024	0.20914	2.00	0.05753	1.94	1.17439	2.05	0.21403	2.05	0.17957	2.04
*	-	-	-	-	1.17439	-	0.21403	-	0.179560	-
Ref [27]	-	-	-	-	1.17441	-	0.21404	-	0.179572	-

Table 1: Lid driven cavity at $Re = 100$ - comparison of some characteristic values along the middle-line to reference values of Botella and Peyret [27] -* is the Richardson interpolated value.

A second test is considered for $Re = 1000$, for which the flow dynamics are largely dominated by inertial forces. The objective of this test is to investigate the ability of the solver to capture the different structures such as primary, right secondary and left secondary vortex, and to evaluate the vorticity at these locations. Again, the numerical results are compared to the reference solution of Botella and Peyret [27], keeping the same numerical parameters as in the previous study.

At $Re = 1000$, the stream function of the flow is illustrated in Fig. 4. The existence of three distinct vortices is visible. The primary vortex denoted PV is large and centered in the cavity, the other secondary vortices represented by LSV and RSV are small and located near the left and right bottom corner of the cavity, respectively. In Fig. 5, the velocity u extracted from the vertical centerline and the velocity v along the horizontal centerline are presented. The numerical results show good agreement with [27], which validates the accuracy of the solver. In addition, table 2, 3 and 4 summarize the values of the vorticity at the center of each vortex and their positions. All results exhibit second order accuracy in the asymptotic convergence zone. For the positions of the vortex, the difference is always smaller than 0.11%, and for the vorticity, the largest discrepancy is of the order of 0.06%.

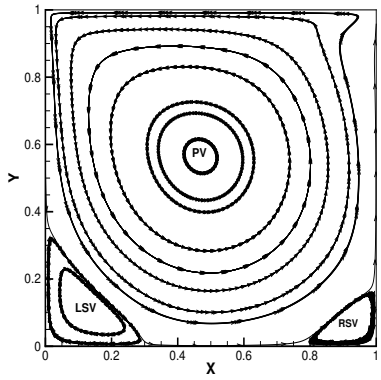
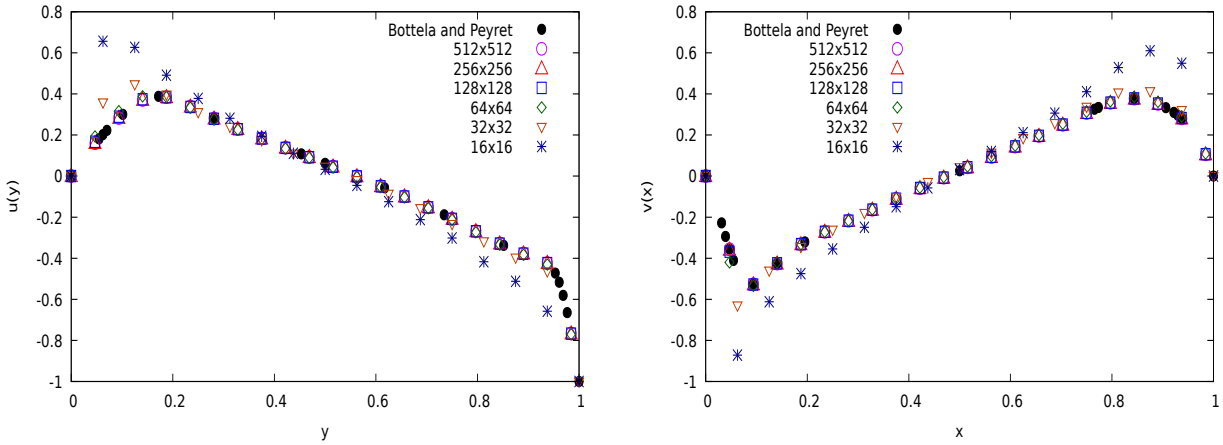


Figure 4: Stream function contours of primary and secondary vortex, obtained at $Re = 1000$



(a) u-velocity on the vertical centerline at $x = 0.5$

(b) v-velocity on the horizontal centerline at $y = 0.5$

Figure 5: Lid driven cavity at $Re = 1000$ - (comparison of the horizontal and vertical velocity along the centerline through the center of the cavity to reference results of Botella and Peyret [27])

N	x	order	y	order	ω	order
32	0.460674		0.545429		2.16344	
64	0.467474		0.560757		2.07752	
128	0.468837	1.31	0.564257	2.13	2.06684	3.01
256	0.469113	2.30	0.565009	2.21	2.06693	-
512	0.469185	1.93	0.565183	2.11	2.06737	2.29
Extra	0.469179	-	0.56518	-	2.067369	-
Ref [27]	0.4692	-	0.5652	-	2.067753	-

Table 2: Lid driven cavity at $Re = 1000$ - Richardson extrapolation for the primary vortex: (x,y) the location of the center of the vortex, and ω is the value of the vorticity at this location

4.2 Single phase flows: Green-Taylor vortex

The second test case is the unsteady Green-Taylor vortex, used as a benchmark to validate numerical codes. This problem is defined in two dimensions on a square domain $\Omega = (0, 1) \times (0, 1)$, with periodic boundary conditions. In this case, the analytical solution has the following form with $\nu = \frac{\mu}{\rho}$:

$$u(x, y, t) = V_0 \cos(2\pi x) \sin(2\pi y) e^{-8\pi^2 \nu t} \tag{31}$$

$$v(x, y, t) = -V_0 \sin(2\pi x) \cos(2\pi y) e^{-8\pi^2 \nu t} \tag{32}$$

N	x	order	y	order	ω	order
32	0.08940		0.08617		-1.50012	
64	0.12906		0.10731		-1.23281	
128	0.13438	2.89	0.11081	2.59	-1.13287	1.41
256	0.13556	2.16	0.11157	2.20	-1.11431	2.42
512	0.13586	1.96	0.11174	2.12	-1.11053	2.30
Extra	0.13584	-	0.11173	-	-1.1105301	-
Ref [27]	0.1360	-	0.1118	-	-1.109789	-

Table 3: Lid driven cavity at $Re = 1000$ - Richardson extrapolation for the left secondary vortex: (x,y) the location of the center of the vortex, and ω is the value of the vorticity at this location

N	x	order	y	order	ω	order
32	0.9073		0.0985712		-0.736348	
64	0.9152		0.0808984		-0.401598	
128	0.916427	2.68	0.0786425	2.96	-0.361584	3.06
256	0.916657	2.41	0.07821	2.38	-0.354138	2.42
512	0.916709	2.14	0.0781233	2.31	-0.352588	2.26
Extra	0.916708	-	0.078116	-	-0.3525867	-
Ref [27]	0.9167	-	0.0781	-	-0.3522861	-

Table 4: Lid driven cavity at $Re = 1000$ - Richardson extrapolation for the right secondary vortex: (x,y) the location of the center of the vortex, and ω is the value of the vorticity at this location

Using equations (31)-(32), two interesting physical quantities can be evaluated: the kinetic energy E_k and the enstrophy ζ by:

$$E_k = \int_0^1 \int_0^1 \rho \frac{(u^2+v^2)}{2} \mathrm{d}x\mathrm{d}y = \frac{\rho\ V_0^2\ F^2(t)}{4} \tag{33}$$

$$\zeta = \int_0^1 \int_0^1 \rho \frac{\omega^2}{2} \mathrm{d}x\mathrm{d}y = 2\pi^2 \rho\ V_0^2\ F^2(t) \tag{34}$$

Where $\omega = \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right)$ is the vorticity in two dimensions and $F(t) = e^{-8\pi^2 \nu\ t}$.

The following discrete analogs of the kinetic energy E_k and enstrophy ζ are used:

$$E_k \simeq \sum_{\Omega_p} \frac{\rho}{2}\ (u_p^2 + v_p^2)\ \Omega_p \tag{35}$$

$$\zeta \simeq \sum_{\Omega_p} \frac{\rho}{2}\ \omega_p^2\ \Omega_p \tag{36}$$

The squared velocity components u_p^2 et v_p^2 are obtained by the following interpolations:

$$u_p^2 = \frac{1}{2}\ (u_{i,j}^2 + u_{i+1,j}^2) \tag{37}$$

$$v_p^2 = \frac{1}{2}\ (v_{i,j}^2 + v_{i,j+1}^2) \tag{38}$$

The vorticity is defined as the rotational of the velocity ($\vec{\omega} = \nabla \times \vec{u}$). In 2D, the vorticity is reduced to a scalar that will be expressed at the viscosity node (see Fig. 6):

$$\omega(i,j) = \frac{v(i,j) - v(i-1,j)}{\Delta x} - \frac{u(i,j) - u(i,j-1)}{\Delta y}. \tag{39}$$

The square of vorticity ω_p^2 is obtained by the mean of the squares of the four vorticities that have been just expressed. The following equation is obtained:

$$\omega_p^2 = \frac{\omega^2(i,j) + \omega^2(i+1,j) + \omega^2(i,j+1) + \omega^2(i+1,j+1)}{4} \tag{40}$$

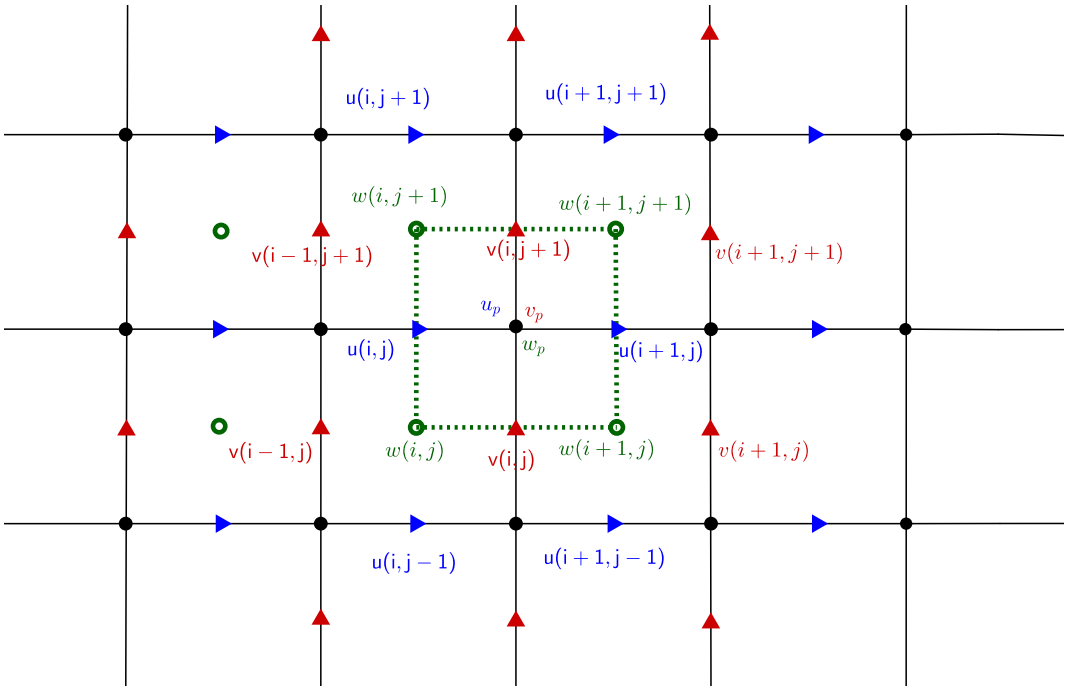


Figure 6: Calcul of the kinetic energy E_k and the enstrophy ζ

In order to quantify the efficiency and robustness of the methods that were presented previously, relative errors on the kinetic energy E_k and enstrophy ζ are calculated in two case: viscous and inviscid flows, respectively. The numerical results on the discrete mesh are confronted to the analytic values obtained from (33)-(34). So, let q_t^{exact} be the exact quantity, q_t the numerical quantiy and nt the number of time steps. the error on q_t is estimated by the following expression:

$$Err_2^{rel}\left(q_t\right) = \left(\frac{\sum_{k=1}^{nt} \left\| q_t^{exact} - q_t \right\|^2}{\sum_{k=1}^{nt} \left\| q_t^{exact} \right\|^2} \right)^{1/2} \tag{41}$$

For this case, three solvers are used: the Fully Coupled with PCD, the Standard Augmented Lagrangian (with $r = 10^6$ in the inviscid case and $r = 10^2$ in the viscous case) and the Standard Projection. Five meshes are simulated, ranging from 16×16 to 256×256, with a residual of $\epsilon = 10^{-12}$ for the BiCGSTAB(2). Concerning the time derivatives, a constant time step $\Delta t = 10^{-3}s$ is chosen and 2000 time steps are computed, corresponding to 0.3 secondes of the flow motion. The fluid properties are $1Kg \cdot m^{-3}$ for the density and $10^{-3}Pa \cdot s$ and $0Pa \cdot s$ for the dynamic viscosity in viscous and inviscid regimes respectively (Figs. 8 and 9). The characteristic velocity V_0 is chosen equal to $1m \cdot s^{-1}$.
Before comparing the three solvers more effectively, the effet of the Augmented Lagrangian parameter is considered in the Fig. 7, for r values in the range 10^4 and 10^7, and for each mesh size. It can be observed in this figure that a value $r = 10^6$ of the Augmented Lagrangian parameter appears to be a good choice in order to conserve the kinetic energy E_k to the order of machine precision. Note that beyond this value, the cost of calculating with this method becomes very high because the use of an iterative solver is difficult when the parameter r is very large.

The L_2 error of the enstrophy ζ and kinetic energy E_k are shown in Figs. 8a and 8b, respectively for the inviscid flow. There is no dissipation phenomon ($\mu = 0Pa \cdot s$). For the enstrophy ζ, as the approximations of the derivatives are second order, all solvers have second order accuracy, except the Standard Projection with $\Delta t = 10^{-3}s$ which cannot give a desired accuracy. Concerning the kinetic energy E_k, the results show that the Fully Coupled and the Augmented Lagrangian are better than the Standard Projection, because they conserve the kinetic energy E_k to the order of machine accuracy. Therefore, the impact of the time step Δt on the Standard Projection is clearly observed due to splitting operators that introduces numerical dissipation. Indeed, when the time step is divided by 10, the error on the kinetic energy error is divided by 100. The same case was simulated on physical times up to 20 seconds, by keeping the same numerical parameters as in the case of 2 seconds. In this case, all solvers continue to converge to machine precision, except the Standard Projection that fails after 6 secondes due to the accumulation of errors, even for small time steps of the order $\Delta t = 10^{-6}s$. In addition, it has been shown

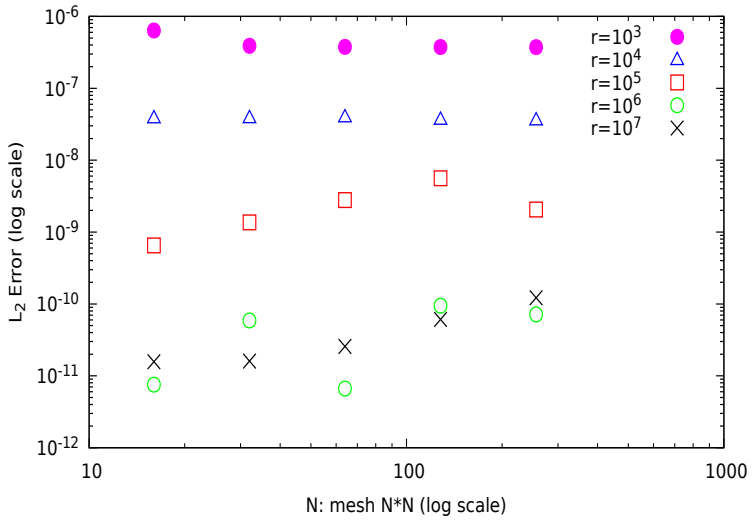
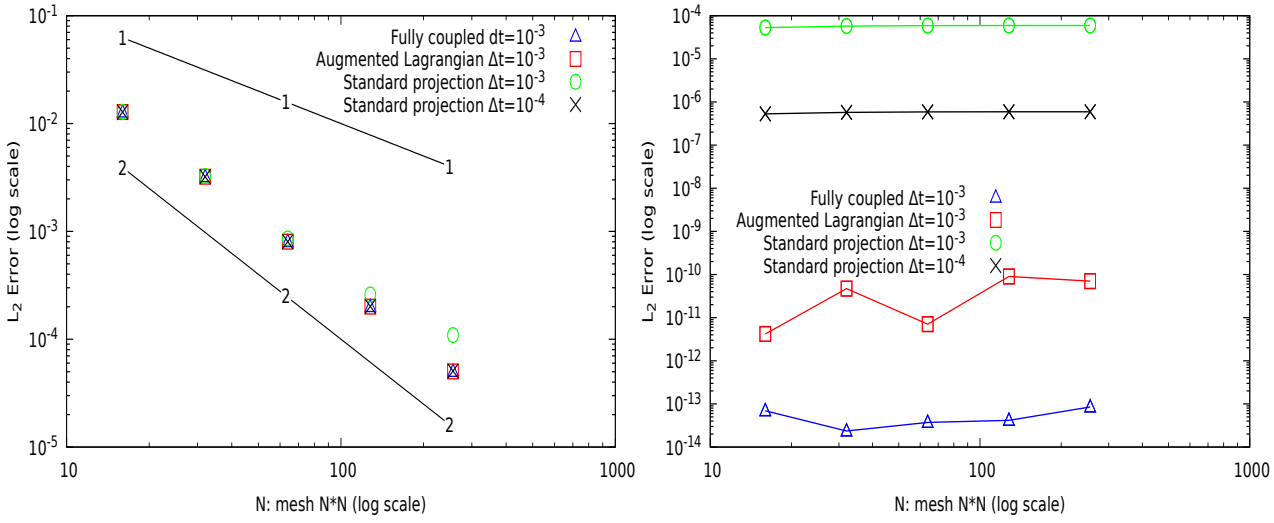


Figure 7: Relative L_2 error of the kinetic energy E_k for the inviscid Taylor-Green test case with different Lagrangian parameters r



(a) Enstrophy ζ

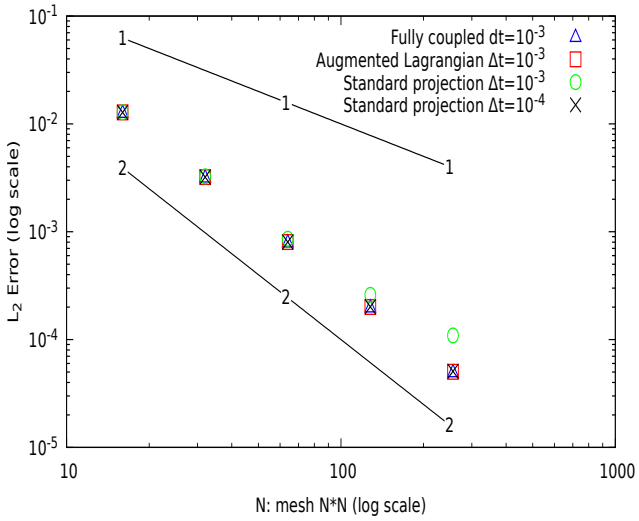
(b) Kinetic energy E_k

Figure 8: Kinetic energy E_k and enstrophy ζ conservations and accuracy of the different methods in the inviscid Taylor-Green test case

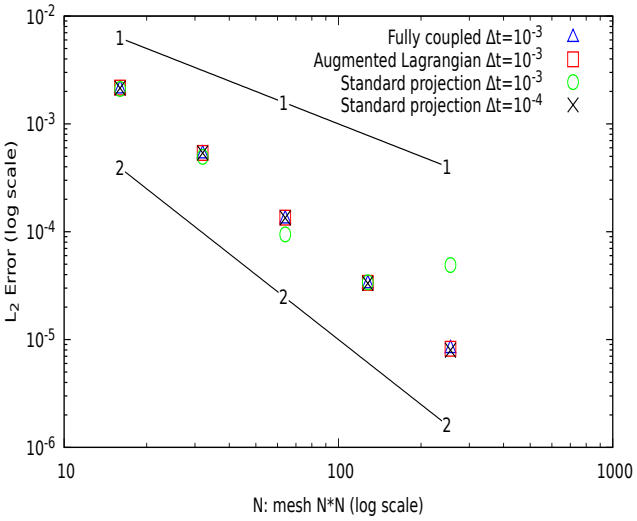
that skew-symmetric scheme preserve energy within machine accuracy on fine meshes [14], and the other methods that can achieve machine precision are spectral methods [29]. A centered scheme with either Fully Coupled or Augmented Lagrangian techniques discretely conserves kinetic energy also, Which is an interesting results of this work.

Result for the viscous case with $\mu = 10^{-3} Pa \cdot s$ are presented in Fig. 9. For the enstrophy ζ and the kinetic energy E_k, it is easily noticeable that the Fully Coupled and Augmented Lagrangian converge with second order accuracy. The effect of the time step is visible for the Standard Projection. Indeed, results with $\Delta t = 10^{-4} s$ are noticeably better than those obtained with $\Delta t = 10^{-3} s$. It can be concluded that the Standard Projection is not able to give good results with large time steps, because the splitting introduces spurious dissipation, and therefore leads to significant numerical errors. Moreover, the effet of numerical dissipation disappears for the Standard Projection, when small time steps are chosen, at the cost of increased computational time.

Fig. 10 reports the CPU time (in second) for the three different solvers of Navier-Stokes in the viscous and inviscid cases, on the finest mesh 256×256. Obtaining the same quality of numerical solutions



(a) Enstrophy ζ



(b) Kinetic energy E_k

Figure 9: Kinetic energy E_k and Enstrophy ζ conservations and accuracy of the different methods in the viscous Taylor-Green test case

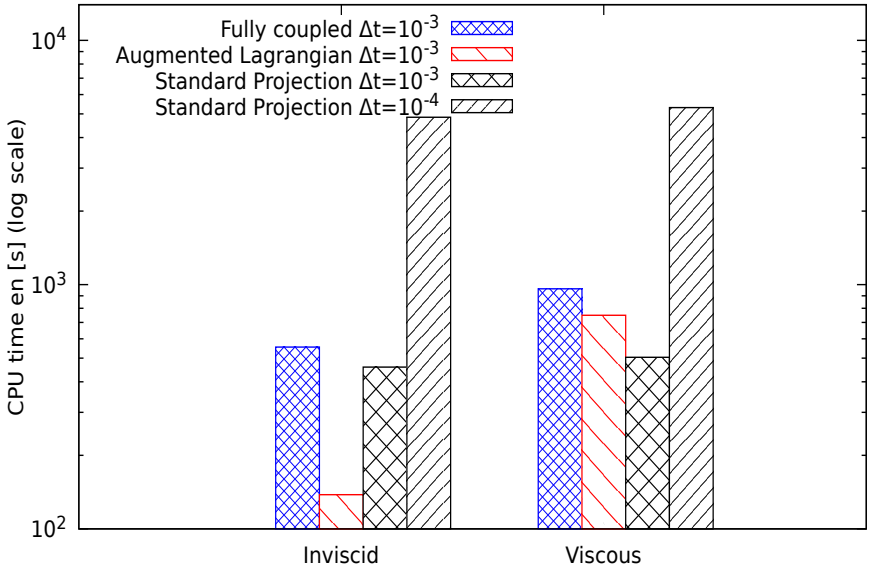


Figure 10: CPU time(s) of Navier-Stokes solvers for Taylor-Green test case (fine mesh 256×256)

whatever the studied case (viscous and inviscid), it is seen that the Standard Projection method is the most expensive in CPU time compared to the two other methods. We also observe for both cases, that the Augmented Lagrangian method is faster than the Fully Coupled method. This can be explained by the fact that the two solvers need one iteration to converge to the exact solution, and that the size of the system to be solved in the case of the Fully Coupled method is larger than that of the Augmented Lagrangian. In addition, the results show that the Fully Coupled method saves 84% (inviscid case) and 63%(viscous case) of the CPU time compared to the Standard Projection, while consuming more than 75% (inviscid case) and 22%(viscous case) of the CPU time compared to the Augmented Lagrangian.

4.3 Two-phase flows: rising buble

The rise of a two-dimensional bubble is studied numerically using the Fully Coupled approaches with the PCD preconditioner, the Adaptative Augmented Lagrangian with $r = 10^3$ and the Standard Projection method. Two configurations are studied : the first corresponds to a numerical benchmark proposed in

[19]. Initially, a bubble of diameter $D = 0.5m$ is centered in a rectangular $2m$ high and $1m$ wide domain. Dirichlet and symmetric boundary conditions are used for the horizontal and the vertical boundaries respectively. The bubble is driven up by a constant gravity $g = 0.98m \cdot s^{-2}$ in the negative y direction, while the deformation of the bubble is counterbalanced by the surface tension, whose coefficient σ is set to $1.98N \cdot m^{-1}$. The first case ($\rho_1/\rho_2 = 10^2, \mu_1/\mu_2 = 10^1$) was simulated, but in this case all the methods give the same results as those of Hysing

The second configuration is characterized by high density and viscosity ratios, respectively ($\rho_1/\rho_2 = 10^3$ and $\mu_1/\mu_2 = 10^2$). The main interest of this case is to compare the time-to-solution of the three methods. To provide detailed comparisons, three time steps are chosen : $\Delta t = 10^{-3}s$, $5 \cdot 10^{-3}s$ and $10^{-2}s$ while target residual is keept constant ($\epsilon = 10^{-7}$ for the BiCGSTAB(2) solver).

In the following, the temporal evolution of two quantities will be considered in order to compare qualitatively the different methods.

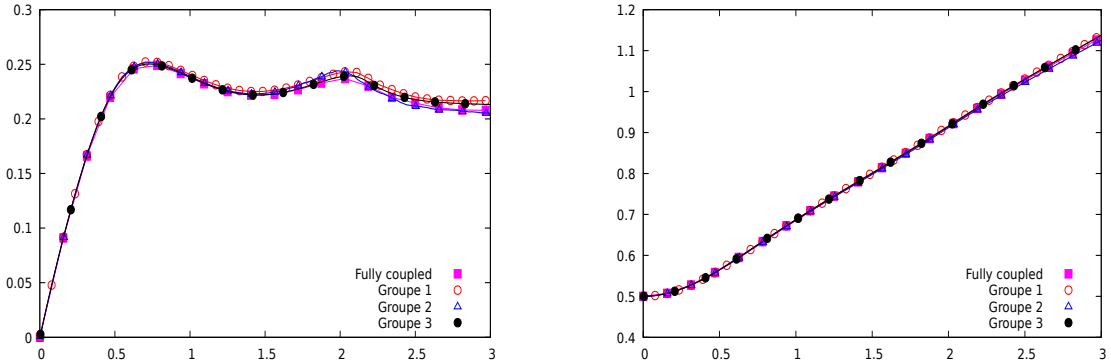
- Center of mass Y_c

$$Y_c = \frac{\int_{\Omega_2} y \, \mathrm{d}x}{\int_{\Omega_2} 1 \, \mathrm{d}x}, \tag{42}$$

- Vertical velocity $\mathbf{U}_c$

$$\mathbf{U}_c = \frac{\int_{\Omega_2} \mathbf{U} \, \mathrm{d}x}{\int_{\Omega_2} 1 \, \mathrm{d}x}. \tag{43}$$

Here Y_c denotes the center of mass of the bubble, $\mathbf{U}_c$ denotes the velocity at the center of mass of the bubble, and Ω_2 represents the subdomain occupied by the bubble.



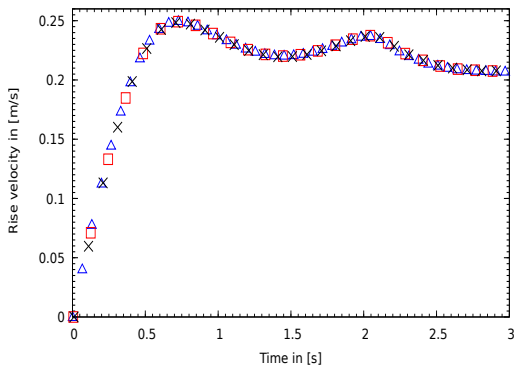
(a) rise velocity

(b) center of mass

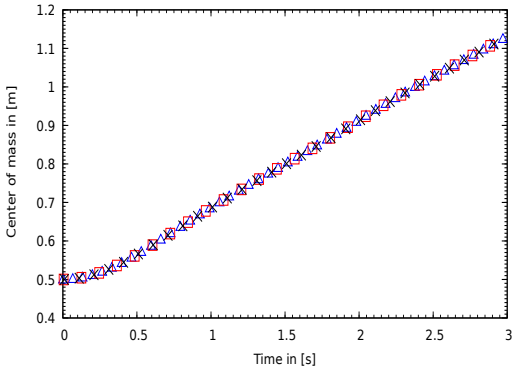
Figure 11: Vertical velocity and center of mass for test case 2 $\left(\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2\right)$. Comparisons betwen the Fully Coupled method and all groups of Hysing and al. [19]

Before comparing the three solver, the solutions computed on the finest grid h=1/320 with the Fully coupled method are compared with the corresponding results of the codes TP2D(groupe1), FreeLIFE(groupe2) and MooNMD(groupe3) given by Hysing and al. [19]. As can be seen in figure. 11b, the estimate of the center of mass by the Fully coupled method is in good agreement with the references values. As regards to the rise velocity of the bubble, the solutions seem to deviate from each other starting at $t = 1 \cdot s$. At long times ($t = 2 \cdot 5s$), the solution of the Fully coupled method and the reference of group 2 seem to converge although it is difficult to discriminate between methods in this case.

Figs. 12, 13 and 14 show the vertical velocity and the center of mass of the bubble, obtained on the fine mesh 320×640 with FC, ALL and PR method. The solutions provide results in good agreement with those of Hysing and al. [19] even if they are not presented here. Also, the simulations show that the results obtained by the Fully Coupled and the Augmented Lagrangian methods are insensitive to the time step. Indeed, for these two methods the rise velocity and the center of mass profiles are qualitatively similar regardless of the time step. This is in contrast with the projection method, for which large time steps clearly induce significant errors on the rise velocity. Fig. 15 compares the shape of the bubble at $t = 3s$ for all three methods using different time steps. It can be observed from these figures that the Fully Coupled and the Augmented Lagrangian methods are more accurate than the projection method. Indeed, in the resolved regions of the bubble, the overall shape of the bubble is converged, and the

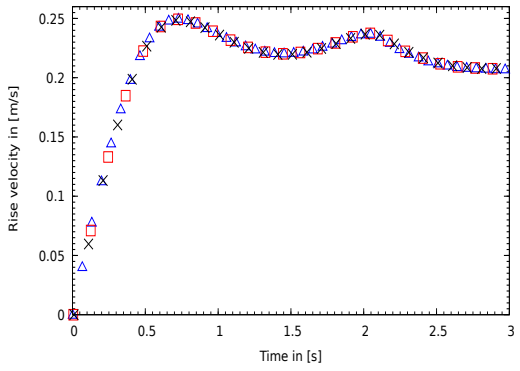


(a) rise velocity

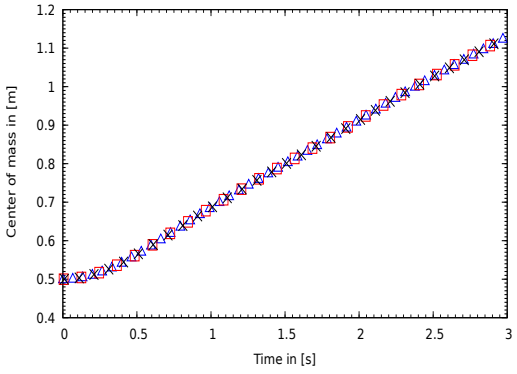


(b) center of mass

Figure 12: Vertical velocity and center of mass for test case 2 $\left(\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2\right)$ obtained by the Fully Coupled method with several time steps. $\Delta t = 10^{-3}s$ (represented by triangles), $\Delta t = 5 \cdot 10^{-3}s$ (represented by squares) and $\Delta t = 10^{-2}s$ (represented by crosses)

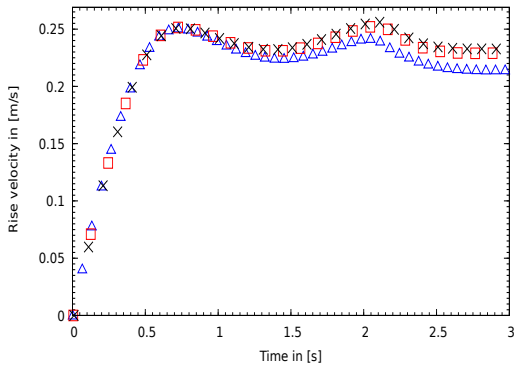


(a) rise velocity

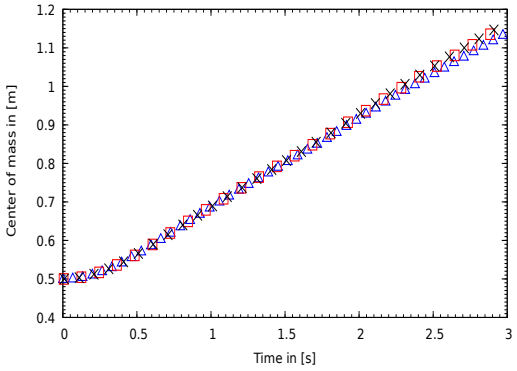


(b) center of mass

Figure 13: Vertical velocity and center of mass for test case 2 $\left(\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2\right)$ obtained by the Adaptative Augmented Lagrangian ($r = 10^3$) with several time step. $\Delta t = 10^{-3}s$ (represented by triangles), $\Delta t = 5 \cdot 10^{-3}s$ (represented by squares) and $\Delta t = 10^{-2}s$ (represented by crosses)



(a) rise velocity

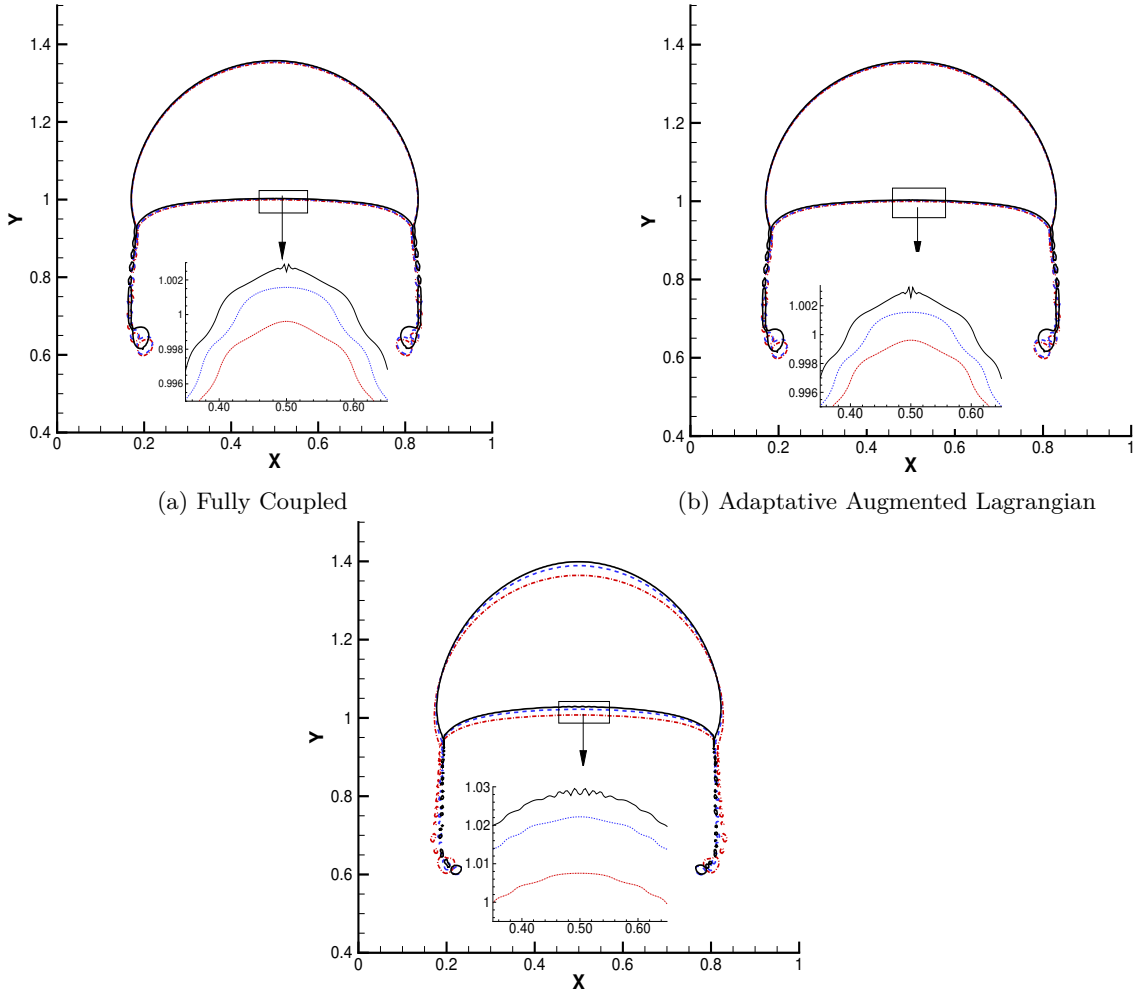


(b) center of mass

Figure 14: Vertical velocity and center of mass for test case 2 $\left(\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2\right)$ obtained by the Standard Projection with several time step. $\Delta t = 10^{-3}s$ (represented by triangles), $\Delta t = 5 \cdot 10^{-3}s$ (represented by squares) and $\Delta t = 10^{-2}s$ (represented by crosses)

filamentary regions resemble the reference results [19]. The solutions were not precisely compared with those of Hysing, because even in the case of Hysing there is a significant dispersion between the methods, none of which can serve as a reference solution. The overall shape of the bubble however is similar to that

of Hysing, whereas for the projection method, a large difference is noticed when the time step is changed, in the solved regions but also on the point of break up. For the second case ($\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2$), we conclude that the Fully Coupled and the Augmented Lagrangian methods are able to give the best results with large step time, which is not possible for the projection method.



(a) Fully Coupled

(b) Adaptative Augmented Lagrangian

(c) Standard Projection

Figure 15: Bubble shapes at time $t = 3s$ for test case 2 $\left(\rho_1/\rho_2 = 10^3, \mu_1/\mu_2 = 10^2\right)$ with several time steps. $\Delta t = 10^{-3}s$ (represented by dash-dotted line), $\Delta t = 5 \cdot 10^{-3}s$ (represented by dashed lines) and $\Delta t = 10^{-2}s$ (represented by solid lines)

In the following, the CPU times necessary for each methods to simulate the rising bubble are considered. These CPU times are shown in Fig. 16. For the three time steps, the Augmented Lagrangian method is the slowest to converge. On the other hand, the difference between the Fully Coupled and the Standard Projection method is significant, as well for the small time step $\Delta t = 10^{-3}s$ the Fully Coupled method saves 80% of the CPU time compared to the Augmented Lagrangian, while it saves 57% more time than the Standard Projection. These large differences can be explained by the fact that the Augmented Lagrangian and Standard Projection take a lot of iterations to converge. As large density and viscosity ratio are considered, the use of a very large value of the parameter r is required to satisfy the incompressibility constraint. This leads to ill-conditioned linear system, hence the high cost of the Augmented Lagrangian method. For the Standard Projection, the cost of the computation is related to the resolution of the Helmholtz's equation with variable coefficients.

4.4 Two-phase flow: free fall of a dense square

The present test case aims at checking the ability of the Fully Coupled method to simulate multiphase flows with high density and viscosity ratios, compared to other methods like the Augmented Lagrangian and the Standard Projection. The two-dimensional free fall of a dense square is tested. Initially, a rigid

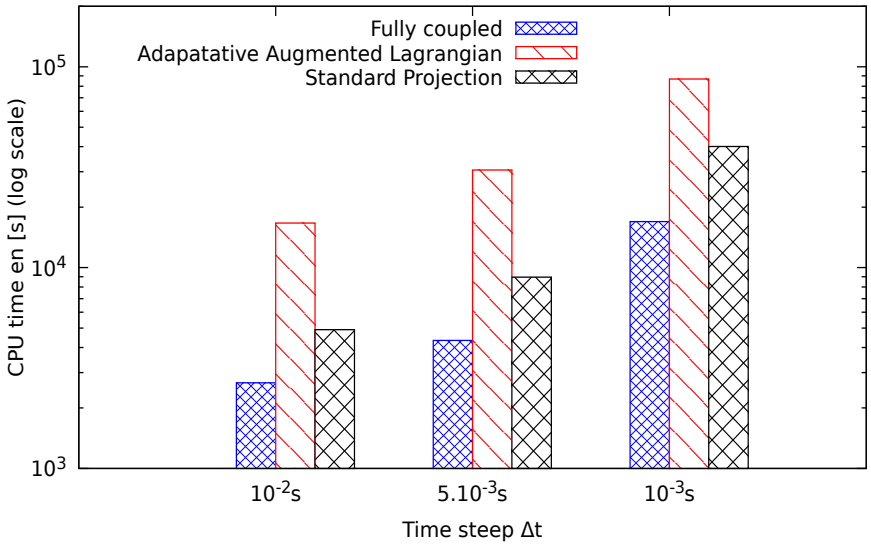


Figure 16: CPU time(s) of Navier-Stokes solvers for rising bubble test case 2 (fine mesh 320×640)

square of side $L = 0.00125m$, density $\rho_2 = 10^7 kg \cdot m^{-3}$ and dynamic viscosity $\mu_2 = 10^5 Pa \cdot s$ is released without velocity in a quiescent gas. Gravity is set at $g = 9.81m \cdot s^{-2}$ while the surface tension coefficient σ is set to zero. The square is centered at $(0.05m, 0.175m)$ in a rectangular cavity full of gas whose density and viscosity are $\rho_1 = 1kg \cdot m^{-3}$ and $\mu_1 = 1Pa \cdot s$ respectively. The dimensions of the cavity are $0.2m$ high and $0.1m$ wide. The interest of this case lies in the possibility of comparing numerical solutions (eq. 43) to an exact reference value for the falling velocity $V_c = -gt$, obtained analytically by neglecting the effect of the gas on the solid. Considering analytical solutions as the appropriate reference solution, the errors committed in this test are evaluated using the definition 41. Where q_t^{exact} is the falling velocity.

The case is simulated on four Cartesian grids (64×128, 128×256, 256×512 and 512×1024), with a constant time step $\Delta t = 5.10^{-5}s$ and a residual of $\epsilon = 10^{-7}$ for the BiCGSTAB(2). Figs. 17 and 18 illustrate the simulation results for the four grids at time $t = 0.15s$ after 3000 temporal iterations with the AAL and FC methods. The corresponding vorticity magnitude and isoline $C = 0.5$ of the color function at the interface are presented. It can be seen that the Fully Coupled method and the Augmented Lagrangian approaches give qualitatively similar results for the same time step ($\Delta t = 5.10^{-5}s$). On the contrary, the Standard Projection method requires a smaller time step ($\Delta t = 5.10^{-6}s$) to obtain a comparable result. So in spite of using this small time step, the simulation with the projection on a 128×256 grid fails to converge after a few thousand iterations because the correction step was not able to reduce the divergence of the predicted velocity.

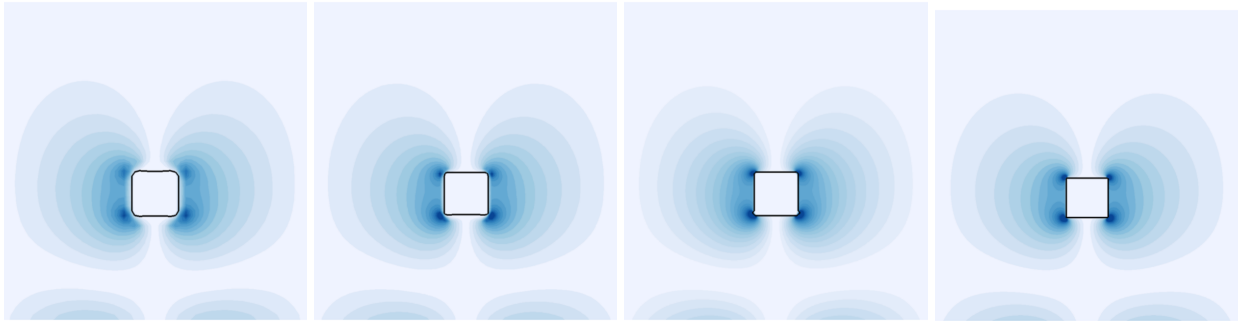


Figure 17: Adaptative Augmented Lagrangian method ($r = 10^3$): vorticity magnitude and isoline ($C = 0.5$) for the free fall of a dense cylinder with 64×128, 128×256, 256×512 and 512×1024 meshes

Fig. 19 presents in logarithmic scale, a comparison of the errors involved by the FC and AAL solvers on the calculation of the vertical velocity at $t = 0.15s$, as well as the corresponding CPU time. As shown in Fig. 19, the Fully Coupled method gives low relative errors, it also saves 25% of the CPU time compared

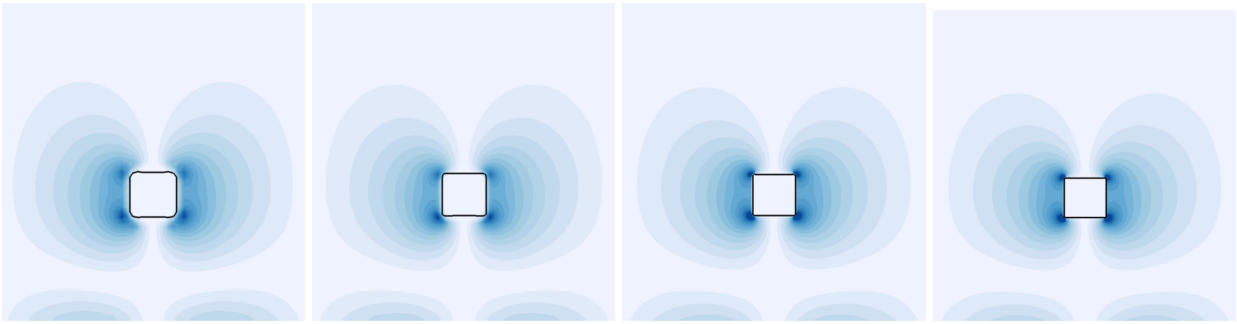


Figure 18: Fully Coupled method: vorticity magnitude and isoline ($C = 0.5$) for the free fall of a dense cylinder with 64×128, 128×256, 256×512 and 512×1024 meshes

to the Adaptative Augmented Lagrangian on the fines mesh 512×1024. Nevertheless, the solutions obtained are always better with the Fully Coupled method than with the Augmented Lagrangian, even if there is a saturation of the BiCG-Stab(2) when the number of iterations increases. However, it has been seen that Standard Projection methods require a time step 10 times smaller than other methods to achieve low levels of error. This method cannot converge on the mesh 128×256.

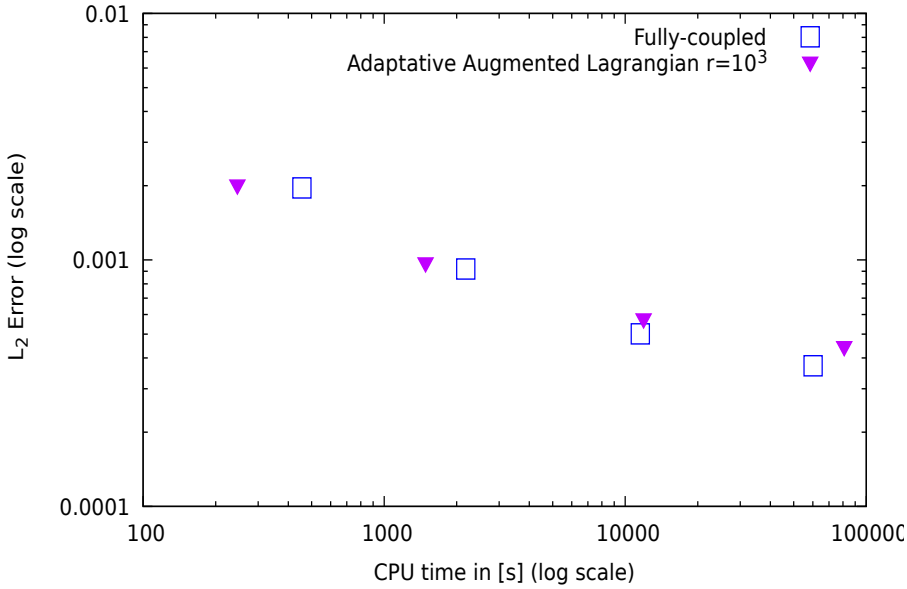


Figure 19: CPU time(s) and error (%) of the vertical velocity for free fall of a dense cylindre

5 Conclusion

Three methods were investigated to solve the Navier-stokes equations for incompressible single- and two-phase flows at large density and viscosity ratios. In the case of the singe-phase, the results obtained for lid-driven cavity case by the Fully coupled method agree well with those of the literature, and exhibit second order accuracy. The unsteady Green-Taylor vortex cases has proven the ability of the Fully coupled and Augmented Lagrangian methods to preserve energy within machine accuracy, whereas the Standard Projection methods introduces numerical dissipation. As regard to two-phase flows, all methods were validated on two benchmark: the rise of a two-dimensional bubble and the free fall of a dense object. These benchmarks highlighted the fact that the Fully Coupled and Adaptative Augmented Lagrangian methods are more efficient to solve two-phase flows (with larger time step Δt). On the contrary, the Standard Projection (SP) method has difficulties for large time steps, due to the accumulation of errors introduced by the pressure-velocity splitting. Large viscosity ratios however introduce a strong coupling between the velocity components, and future work will therefore focus on the preconditioning of the velocity block.

6 ACKNOWLEDGMENTS

The authors are grateful for access to the computational facilities of the French CINES (National computing center for higher education) and CCRT (National computing center of CEA) under project number A0032B06115. We are also grateful for I-SITE FUTURE, UPE and PIA under funding ANR-16-IDEX-0003.

A The BiCGSTAB(2) method

The biconjugate gradient stabilized method (BiCGSTAB(2)) is an iterative method introduced by Van Der Vorst [20] to solve efficiently large, sparse and nonsymmetric linear systems. Contrary to other BiCG variants such as the conjugate gradient squared algorithm (CGS) and the biconjugate gradient (BiCG), this specific method combines two BiCG (called even and odd) with a Generalized Conjugate Residual (GCR) algorithm. This solver is deemed robust and very stable, while reaching convergence quickly in spite of the higher computational cost of each iteration.

In practice, for more difficult problems, it is essential to combine the BiCGSTAB(2) algorithm with a suitable preconditioner P of eq. 25 to accelerate the convergence of the iterative solver and to avoid possible numerical instabilities. Instead of solving $Ax = b$, we prefer to solve $P^{-1}Ax = P^{-1}b$, for which $cond(P^{-1}A) << cond(A)$. Therefore, in the BiCGSTAB(2) algorithm, every matrix-vector product $Ax = p$ will be followed by the calculation $P^{-1}y = p$, the same transformation must be applied to the second member $P^{-1}b = b$. Then, to solve the $Ax = b$ system by giving an initial guess $x^{(0)}$, BiCGSTAB(2) proceeds as below:

Algorithm Preconditionned BiCGSTAB(2) (**solve** $Ax = b$ **with the tolerance** ϵ**)**

1. **Initializations**

$$
\begin{aligned}
&x^{(0)} \in \mathbb{R}^N \\
&b = P^{-1}\, b \\
&p = A\, x^{(0)} \\
&q = P^{-1}\, p \\
&r^{(0)} = b - q \\
&\tilde{r}^{(0)} = p_0 = r^{(0)} \\
&\rho_0^{(0)} = \omega_2^{(0)} = 1 \ , \ \alpha^{(0)} = 0
\end{aligned}
$$

2. **For** $k = 1, 2, 3, \ldots$ **until** convergence **do**

$$\rho_0^{k-1} = -\omega_2^{k-1}\rho_0^{k-1}$$

Even step

$$
\begin{aligned}
&\rho_1^k = (\tilde{r}^{(0)}, r^{k-1}) \\
&\beta^k = \alpha^{k-1}\rho_1^k/\rho_0^{k-1},\ \rho_0^k = \rho_1^k \\
&p = A\, u^{k-1} \\
&v^k = P^{-1}\, p
\end{aligned}
$$

$\gamma^k = (\tilde{r}^{(0)}, v^k)$, $\alpha^k = \rho_0^k / \gamma^k$
$r^{k+1} = r^k - \alpha^k v^k$
$p = A\, r^k$
$s^k = P^{-1}\, p$
$x^{k+1} = x^k + \alpha^k u^k$

Odd step

$\rho_1^k = (\tilde{r}^{(0)}, s^k)$, $\beta^k = \alpha^k \rho_1^k / \rho_0^k$
$v^k = s^k - \beta^k v^k$
$p = A\, v^k$
$w^k = P^{-1}\, p$
$\gamma^k = (\tilde{r}^{(0)}, w^k)$, $\alpha^k = \rho_0^k / \gamma^k$
$u^k = r^k - \beta^k u^k$
$r^k = r^k - \alpha^k v^k$
$s^k = s^k + \alpha^k w^k$
$p = A\, s^k$
$t^k = P^{-1}\, p$

GCR part

$\omega_1^k = (r^k, s^k)$, $\mu^k = (s^k, s^k)$, $\nu^k = (s^k, t^k)$
$\tau^k = (t^k, t^k)$, $\omega_2^k = (r^k, t^k)$, $\tau^k = \tau^k - \frac{\nu^k \nu^k}{\mu^k}$
$\omega_2^k = (\omega_2^k - \nu^k \omega_1^k / \mu^k) / \tau^k$, $\omega_1^k = (\omega_1^k - \nu^k \omega_2^k / \nu^k) / \mu^k$
$x^k = x^k + \omega_1^k r^k + \omega_2^k s^k + \alpha^k u^k$
$r^k = r^k - \omega_1^k s^k - \omega_2^k t^k$
$u^k = u^k - \omega_1^k v^k - \omega_2^k w^k$

3. **test conveergence**
If residual $\leq \epsilon$ **exit**
end if
end for

B The Richardson extrapolation

In order to check the convergence of the numerical solutions, The Richardson extrapolation [26] is used. It allows evaluating the convergence order α of the numerical method, but also obtaining an approximation f^{ex} of the exact solution f, without the need for an analytical solution to the problem. For this purpose, we choose generally (at least) three meshes (with size grids noted h_1, h_2 and h_3 such as $h_1/h_2 = h_2/h_3$) in the convergence region.

The Richardson extrapolation can be a useful method for extracting the convergence order and values extrapolated to the higher order. So, let be f^{ex} the extrapolated quantity, f_{h_i} the numerical solution on the grid of size h_i, α the convergence order and C_α is a coeficient depending on α. In practice, using three grids with $h_1/h_2 = h_2/h_3$, the value of the convergence order is given as follows:

$$\alpha = \frac{\ln\left[\left(f_{h_1} - f_{h_2}\right) / \left(f_{h_2} - f_{h_3}\right)\right]}{\ln\left(h_1/h_2\right)}. \tag{44}$$

By using Eq. 44, the extrapolated quantity f^{ex} is given by the folllowing:

$$f^{ex} = f_{h_3} - C_\alpha h_3^\alpha \tag{45}$$

where

$$C_\alpha = \frac{f_{h_2} - f_{h_3}}{f_{h_2}^\alpha - f_{h_3}^\alpha}. \tag{46}$$

References

[1] M. Benzi and B. Uçar. Block triangular preconditioners for m-matrices and markov chains. *Electronic Transactions on Numerical Analysis. Volume*, 26:209–227, 01 2007.

[2] N. Bootland, A. Bentley, C. Kees, and A. Wathen. Preconditioners for Two-Phase Incompressible Navier–Stokes Flow. *SIAM Journal on Scientific Computing*, 41(4):B843–B869, jan 2019.

[3] J.U Brackbill, D.B Kothe, and C. Zemach. A continuum method for modeling surface tension. *Journal of Computational Physics*, 100(2):335 – 354, 1992.

[4] J. Cahouet and J.-P. Chabard. Some fast 3d finite element solvers for the generalized stokes problem. *International Journal for Numerical Methods in Fluids*, 8(8):869–895, 1988.

[5] J-P. Caltagirone and J. Breil. Sur une méthode de projection vectorielle pour la résolution des équations de navier-stokes. *Comptes Rendus de l'Académie des Sciences - Series IIB - Mechanics-Physics-Astronomy*, 327(11):1179 – 1184, 1999.

[6] Ku. Cheng-Yu. A novel method for solving ill-conditioned systems of linear equations with extreme physical property contrasts. *Computer Modeling in Engineering and Sciences*, 96, 12 2013.

[7] A-J. Chorin. Numerical solution of the navier-stokes equations. *Mathematics of Computation*, 22(104):745–762, 1968.

[8] Lee. Long Craig C. Douglas and Y. Man-Chung. On solving ill conditioned linear systems. *Procedia Computer Science*, 80:941 – 950, 2016. International Conference on Computational Science 2016, ICCS 2016, 6-8 June 2016, San Diego, California, USA.

[9] M-S. Dodd and A. Ferrante. fast pressure-correction method for incompressible two-fluid flows. *Journal of Computational Physics*, 273:416 – 434, 2014.

[10] S. David E. Howard and J-W. Andrew. Finite elements and fast iterative solvers: With applications in incompressible fluid dynamics. 01 2006.

[11] H. Elman and S-R. Tuminar. Boundary conditions in approximate commutator preconditioners for the navier-stokes equations. *Sci. Comput.J. Comput. Appl. Math.SIAM J. Sci. Comput.SIAM J. Sci. Comput*, 35:257–280, 01 2009.

[12] D-R. Falgout and U-M. Yang. hypre: A library of high performance preconditioners. In Peter M. A. Sloot, Alfons G. Hoekstra, C. J. Kenneth Tan, and Jack J. Dongarra, editors, *Computational Science — ICCS 2002*, pages 632–641, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[13] M. Fortin and R. Glowinski. Augmented lagrangian methods: Applications to the numerical solution of boundary-value problems. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 65(12):622–622, 1985.

[14] D. Fuster. An energy preserving formulation for the simulation of multiphase turbulent flows. *Journal of Computational Physics*, 235:114–128, 2013.

[15] K. Goda. A multistep physics technique with implicit difference schemes for calculating two- or three-dimensional cavity flows. *Journal of computational physics 30*, 1979.

[16] J-L. Guermond. Remarques sur les méthodes de projection pour l'approximation des équations de Navier-Stokes. *Numer. Math.*, 67(4):465–473, 1994.

[17] J-L. Guermond and A. Salgado. A splitting method for incompressible flows with variable density based on a pressure poisson equation. *Journal of Computational Physics*, 228:2834–2846, 05 2009.

[18] F-H. Harlow and J. Eddie Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids*, 8(12):2182–2189, 1965.

[19] S. Turek D. Kuzmin N. Parolini E. Burman S. Ganesan Hysing, S. and L. Tobiska. Quantitative benchmark computations of two-dimensional bubble dynamics. *International Journal for Numerical Methods in Fluids*, 60:1259 – 1288, 08 2009.

[20] D-C. Sorensen J-J. Dongarra, L-S. Duff and Henk A. Vander Vorst. *Numerical Linear Algebra for High Performance Computers*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998.

[21] P. Lubin-N. Randrianarivelo J-P. Caltagirone, S. Vincent. An adaptative augmented lagrangian method for three-dimensional multi-material flows. *Computers and Fluids 33*, 2004.

[22] I. Kataoka. Local instant formulation of two-phase flow. *International Journal of Multiphase Flow*, 12(5):745 – 758, 1986.

[23] G-H. Golub M. Benzi and J. Liesen. Numerical solution of saddle point problems. *Acta Numerica, 14, pp*, 2005.

[24] M.Benzi and M. Olshanskii. An augmented lagrangian-based approach to the oseen problem. *SIAM Journal on Scientific Computing*, 28:2095–2113, 01 2006.

[25] C. Mingchao, B. John A. Nonaka, and and A. Donev B-E. Griffith. Efficient variable-coefficient finite-volume stokes solvers. *Communications in Computational Physics*, 16(5):1263–1297, 2014.

[26] M. Marc N. Xavier, G. Stéphane and G. Stéphane. Benchmark solution for a three-dimensional mixed-convection flow, part 2: Analysis of richardson extrapolation in the presence of a singularity. *Numerical Heat Transfer*, pages 346–369, 11 2011.

[27] B. Olivier and P. Roger. Benchmark spectral results on the lid-driven cavity flow. *Computers and Fluids*, 27:421–433, 05 1998.

[28] C-H. Bruneau P. Angot and P. Fabrie. A penalization method to take into account obstacles in viscous flows. *Numerische Mathematik*, 81:497–520, 02 1999.

[29] Wai-Sun D. Gottlieb O. Schilling and L. Jameson S. Chi-Wang, Don. Numerical convergence study of nearly incompressible, inviscid taylor–green vortex flow. *Journal of Scientific Computing*, 24(1):1–27, Jul 2005.

[30] Y. Saad. Ilut: A dual threshold incomplete lu factorization. *Numerical Linear Algebra with Applications*, 1(4):387–402, 1994.

[31] Y. Saad and M. Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.

[32] K. Stüben. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics*, 128(1):281 – 309, 2001. Numerical Analysis 2000. Vol. VII: Partial Differential Equations.

[33] R. Témam. Sur l'approximation de la solution des équations de navier-stokes par la méthode des pas fractionnaires (i). *Archive for Rational Mechanics and Analysis*, 32(2):135–153, Jan 1969.

[34] S. Vincent, A. Sarthou, J.P. Caltagirone, F. Sonilhac, P. Février, C. Mignot, and G. Pianet. Adaptive augmented lagrangian techniques for simulating unsteady multiphase flows. 07 2007.

[35] S. Vincent, A. Sarthou, J.P. Caltagirone, F. Sonilhac, P. Février, C. Mignot, and G. Pianet. Augmented lagrangian and penalty methods for the simulation of two-phase flows interacting with moving solids. application to hydroplaning flows interacting with real tire tread patterns. *Journal of Computational Physics*, 230(4):956–983, 2011.

[36] D. L. Youngs. *Time-Dependent Multi-material Flow with Large Fluid Distortion*, volume 24. 01 1982.