A PARALLEL FINITE ELEMENT METHOD FOR TWO-PHASE FLOWS

Michele Giordano*, Aldo Bonfiglioli* and Vinicio Magi*

*Università degli Studi della Basilicata, DIFA
Via dell’Ateneo Lucano, Potenza, 85100, Italy
†CASPUR,
Via dei Tizii 6/b, Roma, 00185, Italy

e-mail: mggiordano@unibas.it, bonfiglioli@unibas.it, magi@unibas.it

Key words: Finite Element Methods, SUPG, Two-Phase Flows, Parallel Implementation

Abstract. A parallel finite element model for incompressible laminar two-phase flows is presented. A two-fluid model, describing the laminar non-equilibrium flow of two incompressible phases, is discretized by means of a properly designed Streamline Upwind Petrov-Galerkin (SUPG) finite element procedure. Such a procedure is consistent with a continuous pressure equation. The design and the implementation of the algorithm are presented together with its validation throughout a comparison with simulations available in literature.

1 INTRODUCTION

The aim of this paper is to present some results on the application of a finite element methodology to the two-phase flows. Both, the mathematical modeling of the flow and the numerical discretization adopted are considered to be interesting for the reader, so that they are discussed in details.

Over the last years, an increasing interest has characterized the research of mathematical models that may describe the behaviour of two-phase flows [1, 2, 3]. In a microscopic sense, a two-phase flow system consists of a number of single-phase regions bounded by moving interfaces. Therefore, mathematically it is possible to formulate a two-phase flow problem by considering a field subdivided into single-phase regions where the Navier-Stokes (NS) equations hold for each sub-region. At the phase interfaces, appropriate jump and boundary conditions are used to match the solutions of these differential equations. This type of formulation is called a local instant formulation [4]. For most practical two-phase problems, the mathematical difficulties encountered by using this formulation are insurmountable. The difficulties arise from the existence of deformable moving interfaces with their motion being unknown, and the existence of the fluctuations of variables due
to turbulences and to the motions of the interfaces. The first effect causes complicated coupling between the field equations of each phase and the interfacial conditions, whereas the second effect introduces statistical characteristics originating from the instability of the NS equations. Since these difficulties exist in almost all two-phase flow systems, an application of the local instant formulation to obtain a solution is very limited.

A method of averaging the local instant formulation is followed in practice. The averaging procedure is basically low pass filtering that eliminates unwanted high frequency signals from local instant fluctuations. Only the macroscopic and statistical effects appear in the averaged equations. Instead of considering local instant transfers at the interface, collective interactions between the phases are considered in a macroscopic two-phase flow formulation. Averaging procedures can be classified into three main groups: Eulerian, Lagrangian and Boltzmann statistical averages. In the Eulerian average, the time and space coordinates are taken as the independent variables and the averaging is done with respect to these variables. Lagrangian averages are directly related to the Lagrangian description of mechanics. The particle coordinate replaces the space coordinate, respect to the Eulerian approach, and averages are performed along particle trajectories. The Boltzmann statistical averaging is based on the introduction of a probability density function, described by the Boltzmann transport equation, which plays a weighting role in the integration of the instantaneous values to obtain the averages of the dependent variables.

Averaging results in macroscopic field equations for the mass, momentum and energy of each phase with coupling between the phases through appropriate terms. Closure relations are needed to model these interface transport terms.

Here the incompressibility constraint is considered for both phases, such that the energy conservation equations are not solved. Moreover a two-fluid model is used: this means that an Eulerian approach for both phases is adopted, and the tracking of individuals particles is avoided, but transport properties for the dispersed phase are defined. This approach is considered to be appropriate for the most general and detailed description of two-phase flows and represents a fair compromise between the complexity of the model and its suitability to the physics of the problem. Being an averaged model, it describes the two-phase system from a macroscopic point of view, allowing a more engineering oriented analysis. On the other hand, as said before, the averaging process leads to the loss of the information related to the heterogeneous structure of the flow and to a blurring of the material properties. In particular, the two-fluid model describes the flow as the combination of two interpenetrating continua, where the interactions between the two phases are described through *ad hoc* built forces and parameters.

Here, two-phase steady flows are solved by means of a finite element numerical approach. The discretization is based on the SUPG method [5, 6, 7] which consists in a classical Galerkin finite element method enriched with stabilization terms acting in the upwind direction. In particular, the system of equations is solved as an advective-diffusive set of equations written in primitive variables, and over the triangulated domain piecewise linear functions are used (the so-called P1 functions), both for the basis and the weighting
functions. In the design of the stabilizing parameters, a common criterion has been used, i.e. the numerical discretization adopted to be consistent to the analogous finite element approach for incompressible single-phase flows. Hence, an extension of the SUPG method used for solving the single-phase NS equations is applied to the two-phase flow equations, and similarly to the single-phase flow the problem is written in terms of the primitive variables. Since the system of equations is not symmetric, a straightforward direction to be followed has not been traced in the SUPG frame, and the idea of maintaining such a consistency greatly helps in the design of the intrinsic time scales. In order to achieve a fair level of stabilization, a discontinuity capturing operator is introduced in order to prevent oscillations in boundary and sharp layer regions.

Due to the big amount of computations needed to perform a two-phase flow simulation, a parallelization strategy has been pursued. Through the use of the MPI protocol and the free PETSc library [8], the code has been transposed from its first serial to its definitive parallel version.

In the following, the governing equations describing the two-phase flow and the stabilized finite element approach are given. Then, the parallelization process is shown, and some test cases to validate the proposed methodology are shown. Finally, conclusions are drawn.

2 TWO PHASE FLOW MODEL

The governing equations of the two-phase flow are presented in this section. In particular, the two-fluid model adopted in this paper is discussed and displayed in its main features.

2.1 Basic Concepts

The chosen model is an Eulerian-Eulerian one [1], since both phases are treated as continua. Moreover, a pressure equilibrium between the phases is assumed, leading to the following relation:

\[ p_l = p_g = p \]  \hspace{1cm} (1)

where \( p \) is the pressure, and the subscripts \( l \) and \( g \) are referred to as the primary and the secondary phases, respectively.

Since the phases are considered as interpenetrating, non-mixing continua, each of them occupies a well-defined volume of space. The volume \( V_q \) of the generic phase \( q \) is defined by

\[ V_q = \int_\Omega \alpha_q d\Omega \]  \hspace{1cm} (2)

where \( \alpha_q \) is the phasic volume fraction, representing the volume fraction occupied by the phase \( q \) in the whole domain \( \Omega \). The summation of the phase volumes must recover the whole domain, so that the volume fractions of the phases are coupled by the following
relation:

\[ \alpha_g + \alpha_l = 1 \]  \hspace{1cm} (3)

Equation (3) represents the fundamental constitutive law for the volume fractions.

### 2.2 Mass Conservation

The two-fluid model must guarantee the conservation of mass. This is achieved writing the conservation equations for each phase. Referring to the continuity equation of the generic phase \( q \), with the hypothesis of incompressibility, one can write:

\[ \frac{\partial \alpha_q}{\partial t} + \nabla \cdot (\alpha_q \bar{u}_q) = 0 \]  \hspace{1cm} (4)

where \( \rho_q \) and \( \bar{u}_q \) are the density and the velocity of the phase, respectively.

### 2.3 Momentum Conservation

The two-phase flow model must guarantee the conservation of momentum as well. The momentum balance for the generic phase \( q \), without body forces, yields

\[ \frac{\partial \bar{u}_q}{\partial t} + \bar{u}_q \cdot \nabla \bar{u}_q = \tilde{F}_{p,q} + \tilde{F}_{v,q} + \tilde{F}_{d,q} \]  \hspace{1cm} (5)

where \( \tilde{F}_{p,q} \) is representative of the pressure contribution, \( \tilde{F}_{v,q} \) of the viscous force, and \( \tilde{F}_{d,q} \) of the drag force, i.e. the interaction between the phases. The pressure and the viscous force contribution are given by

\[ \tilde{F}_{p,q} = -\frac{1}{\rho_q} \nabla p \quad \tilde{F}_{v,q} = \frac{1}{\rho_q \alpha_q} \nabla \cdot (\alpha_q \overline{\tau}_q) \]  \hspace{1cm} (6)

where the stress tensor is written as

\[ \overline{\tau}_q = \mu_q (\nabla \bar{u}_q + \nabla \bar{u}_q^T) \]  \hspace{1cm} (7)

where \( \mu_q \) is the shear viscosity and the superscript \( T \) denotes the transpose of \( \nabla \bar{u}_q \). Referring to the primary phase \( l \) and to the secondary phase \( g \), the drag force [1, 9] can be written as

\[ \tilde{F}_{d,l} = \frac{3}{4} \frac{C_D}{d_g \rho_l} \alpha_l \rho_l |\bar{u}_g - \bar{u}_l| (\bar{u}_g - \bar{u}_l) \]  \hspace{1cm} (8)

\[ \tilde{F}_{d,g} = \frac{3}{4} \frac{C_D}{d_g \rho_g} \alpha_g \rho_l |\bar{u}_l - \bar{u}_g| (\bar{u}_l - \bar{u}_g) \]  \hspace{1cm} (9)

where \( C_D \) is the drag coefficient, \( d_g \) is the diameter of the bubbles of the secondary phase and \( \rho \) is the mixture density defined as

\[ \rho = \alpha_l \rho_l + \alpha_g \rho_g \]  \hspace{1cm} (10)
2.4 System of Equations

Referring to the primary phase as \( l \) and to the secondary phase as \( g \), the set of unknowns is given by the primitive variables

\[
U = \begin{pmatrix} p & u_l & v_l & \alpha_g & u_g & v_g \end{pmatrix}^T
\]  

Hence, the final system of equations consists of six equations: since it appears logic to calculate velocities and volume fractions from the momentum and the continuity equations, respectively, the fundamental point is the assignment of the equation for the pressure. The chosen equation, obtained from the conservation of the total volume \( (\sum_{q=l,g} \alpha_q = 1) \) applied to the mass conservation equations, reads

\[
\sum_{q=l,g} \nabla \cdot (\alpha_q \vec{u}_q) = 0
\]

In this equation, the artificial compressibility formulation is used, i.e. a pressure time derivative is introduced to enhance robustness for steady state simulations [10]. Hence, the re-arranged equation for the pressure computation is

\[
\frac{1}{\rho c^2} \frac{\partial p}{\partial t} + \nabla \cdot ((1 - \alpha_g) \vec{u}_l) + \nabla \cdot (\alpha_g \vec{u}_g) = 0
\]

where \( c \) is a reference velocity and \( \rho \) is the mixture density.

3 FINITE ELEMENT APPROACH

The discretization of the governing equations is done through a Galerkin finite element approach with \( P1 \) basis functions for all the variables [11]. This means piecewise linear functions are used both for the basis and the weighting functions and the domain is discretized into triangles. In addition, a Petrov-Galerkin stabilization is employed.

3.1 Galerkin Formulation

The governing equations can be written as a system of equations in terms of the chosen set of variables \( U \)

\[
\mathcal{L}(U) = A_i U_{,i} + F_{\text{adv}}^{i;i} - F_{\text{diff}}^{i;i} = S
\]

where \( F_{\text{adv}}^{i;i} = A_i U_{,i} \) is the advective contribution, \( F_{\text{diff}}^{i;i} \) is the diffusive (pressure and viscous term) contribution, and \( S \) represents the source term.

Introducing the trial solution space \( \mathcal{V}_h \), and the weighting solution space \( \mathcal{W}_h \), the weak form of (14) reads as find \( U \in \mathcal{V}_h \) such that \( \forall \mathcal{W} \in \mathcal{W}_h \)

\[
\int_{\Omega} (\nabla \cdot A_{i} U_{,t} + \nabla \cdot F_{\text{adv}}^{i;i} + \nabla \cdot F_{\text{diff}}^{i;i} - \mathcal{W} \cdot S) \, d\Omega - \int_{\Gamma} (\nabla \cdot F_{\text{diff}}^{i;i}) n_t \, d\Gamma = 0
\]

where \( \Omega \) is the spatial domain, \( \Gamma \) represents its boundaries and \( n_t \) is a component of the outer unit normal to the boundary.
3.2 SUPG Formulation

The stabilization is obtained following a Streamline Upwind Petrov-Galerkin procedure applied to advection-diffusion systems. The new integral equation is

$$
\int_{\Omega} \left( W \cdot A_0 \mathbf{U}_t + W \cdot \mathbf{F}^{adv}_{i,i} + W \cdot \mathbf{F}^{diff}_{i,i} - W \cdot S \right) d\Omega
- \int_{\Gamma} (W \cdot \mathbf{F}^{diff}_{i,i}) n_i d\Gamma + ST_{supg} = 0
$$

(16)

where

$$
ST_{supg} = \sum_{e=1}^{n_{el}} A_i^T W_{i,j} \mathbf{\tau}(\mathcal{L}(\mathbf{U}) - \mathbf{S}) d\Omega
$$

(17)

In the above formula \(\Omega_e\) is one of the \(n_{el}\) elements in which the domain is divided, and \(\mathbf{\tau}\) represents the matrix of the intrinsic time scales of the stabilizing operator.

The choice of \(\mathbf{\tau}\) is cumbersome. In this work a simple diagonal matrix is used

$$
\mathbf{\tau} = \text{diag}(\tau_p, \tau_{ml}, \tau_{ml}, \tau_{\alpha}, \tau_{mg}, \tau_{mg})
$$

(18)

The values of the \(\tau\)’s are obtained by applying the one-dimensional theory to each equation. Specifically, the values available in the literature [6, 7, 11] are considered and re-arranged for this methodology.

In particular, the time scale for the secondary-phase momentum equation is chosen as:

$$
\tau_{mg} = \frac{\alpha_g h_g}{2 |\vec{u}_g|} \zeta \left( Re^{hy} \right)
$$

(19)

where \(h\) is the element length in the direction of the local flow, \(\vec{u}_g\) the local advection velocity, \(Re^{hy}\) the element Reynolds number based on the element length and \(\zeta\) a function of \(Re^{hy}\). Their definitions are

$$
Re^{hy} = \frac{h_g |\vec{u}_g|}{2 \nu_g}
$$

(20)

$$
\zeta = \max \left[ 0, \min \left( \frac{Re^{hy}}{3}, 1 \right) \right]
$$

(21)

$$
h_g = \frac{2}{\sum_{i=1}^{3} \frac{|\vec{u}_g|}{|\vec{u}_g| \cdot \nabla w_i \vec{u}_g} \left| w_i \vec{u}_g \right|}
$$

(22)

where the summation is performed for all the nodes of the element and \(w_i \vec{u}_g\) are the weighting functions associated to the secondary phase velocity.

Similarly, the time scale for the primary-phase momentum equation is chosen as:

$$
\tau_{ml} = \frac{(1 - \alpha_g) h_i}{2 |\vec{u}_i|} \zeta \left( Re^{hi} \right)
$$

(23)
with

\[ R^{h_1} = \frac{h_1 ||\tilde{u}_i||}{2\nu} \]  
(24)

\[ h_1 = \frac{2}{\sum_{i=1}^{3} \frac{||\tilde{u}_i|| \cdot \nabla w_i^i}{||\tilde{u}_i||^2}} \]  
(25)

In this work, the evaluation of \( \tau_p \) has not been considered, since zero matrix entries correspond to the time scale related to the pressure equation residual in the SUPG framework. Finally, the expression of \( \tau_\alpha \) will be given in the next subsection.

### 3.3 Stabilization of the Continuity Equation

The key point of the proposed finite element approach concerns with the stabilization of the continuity equation.

Let us consider Eq.(4) written in terms of \( \alpha_g \)

\[ \frac{\partial \alpha_g}{\partial t} + \tilde{u}_g \cdot \nabla \alpha_g + (\nabla \cdot \tilde{u}_g) \alpha_g = 0 \]  
(26)

which is a scalar advection-reaction equation, where \( \tilde{u}_g \) is the advection velocity and \( \nabla \cdot \tilde{u}_g \) the reaction source. Then, the choice of the time scale \( \tau_\alpha \) follows the analogous choice of the scalar advection-reaction equation given in ref. [12]

\[ \tau_\alpha = \frac{C_1}{2||\tilde{u}_g|| + C_2|\nabla \cdot \tilde{u}_g|} \]  
(27)

where \( C_1 \) and \( C_2 \) are constants set through numerical tests. Suggested values, to prevent adding too much numerical viscosity, are \( C_1 = 0.5 \) and \( C_2 = 12 \).

Moreover, in order to prevent oscillations in the boundary layer region, a discontinuity capturing operator [6] is introduced for the continuity equation. The scalar weighting function referred to \( \alpha_g \) is increased in the following way

\[ \tilde{u}_i^\alpha = w_i^\alpha + \tau_\alpha \tilde{u}_g \cdot \nabla w_i^\alpha + \tau_{\alpha_2} \tilde{u}_{g||} \cdot \nabla w_i^\alpha \]  
(28)

with the last term representing the contribution coming from the discontinuity capturing operator. The definitions for \( \tilde{u}_{g||} \) and \( \tau_{\alpha_2} \) are

\[ \tilde{u}_{g||} = \begin{cases} 
\frac{(\tilde{u}_g \cdot \nabla \alpha_g)}{||\nabla \alpha_g||^2} \nabla \alpha_g & \text{if } \nabla \alpha_g \neq 0 \\
0 & \text{if } \nabla \alpha_g = 0
\end{cases} \]  
(29)

and

\[ \tau_{\alpha_2} = \max(0, \tau_{\alpha||} - \tau_\alpha) \]  
(30)
with
\[ \tau_{\alpha||} = \frac{C_1}{2 \| \tilde{u}_{\alpha||} \| + C_2 | \nabla \cdot \tilde{u}_{\alpha||} |} \]  
(31)

From their definitions, \( \tilde{u}_{\alpha||} \) is the component of \( \tilde{u}_{\alpha||} \) in the direction of the \( \alpha \)-gradient, and \( \tau_{\alpha||} \) is the extra-amount of the time scale in the \( \tilde{u}_{\alpha||} \) direction. In the above formulas, \( h_{\alpha||} \) is computed from \( \tilde{u}_{\alpha||} \) and it is equal to
\[ h_{\alpha||} = \frac{2}{\sum_{i=1}^{3} \frac{||\tilde{u}_{\alpha||}||}{||\tilde{u}_{\alpha||}||} \cdot | \nabla w_i^{\alpha||} |} \]  
(32)

4 REMARKS ON THE STABILIZATION METHOD

In this section, further details of the stabilization technique are given. The main idea is to prove it acts consistently with a PSPG-SUPG stabilization criterion used for the incompressible NS equations, in the case P1 linear interpolation functions and Galerkin approach are employed, so successfully circumventing the so-called Babuška-Brezzi (BB) condition [5, 13, 14].

4.1 Navier-Stokes Equations

The traditional stabilization method for a Galerkin finite element discretization of the incompressible NS equations is the so-called PSPG (Petrov-Galerkin Pressure Stabilization) stabilization for the continuity equation and SUPG stabilization for the momentum equation.

This method consists by adding the following stabilizing terms to the integral formulation of the continuity and momentum equations
\[ S_{PSPG} = \sum_e \int_{\Omega_e} \left( \tau_{PSPG} \nabla w_i^p \right) \cdot \tau_i^{h_{\text{mom}}} d\Omega_e \]  
(33)
\[ S_{SUPG} = \sum_e \int_{\Omega_e} \left( \tau_{SUPG} \tilde{u}_i \cdot \nabla w_i^d \right) \tau_i^{h_{\text{mom}}} d\Omega_e \]  
(34)

where \( \tau_{\text{mom}} \) is the residual of the discrete momentum equation, \( \tau_{PSPG} \) and \( \tau_{SUPG} \) are the appropriate PSPG and SUPG time scales, and \( w_i^p \) and \( w_i^d \) are the weighting functions associated to the pressure and the velocity.

The same kind of stabilization is obtained when the SUPG formula (16) for a system of equations is applied to the NS equations. This is done by writing the equations in terms of the primitive variable and choosing the following \( \tau \) matrix
\[ \tau = \text{diag}(0, \tau_m, \tau_m) \]  
(35)
with $\tau_{PSGP} = \tau_{SUPG} = \tau_m$.

Infact, if we write the NS equations in the primitive variables $U$

$$U = (p \; u \; v)^T,$$

(36)

the corresponding Euler Jacobian matrices $A_x$ and $A_y$ are

$$A_x = \begin{pmatrix}
0 & 0 & 0 \\
0 & u & 0 \\
0 & 0 & u
\end{pmatrix},
A_y = \begin{pmatrix}
0 & 0 & 0 \\
0 & v & 0 \\
0 & 0 & v
\end{pmatrix}. \quad (37)
$$

Since the stabilizing terms are given by equation (17):

$$ST_{supg} = \sum_{e=1}^{n_e} \int_{\Omega_e} A_i^T W_{i,\tau} \cdot \tau (U - S)d\Omega; \quad (38)$$

it comes out that, with the above definition of the $\tau$ matrix,

$$ST_{supg} = S_{PSGP} + S_{SUPG}. \quad (39)$$

4.2 Two-Fluid Model

Similarly, also for the two-fluid model the SUPG term depends from the Euler Jacobian matrices $A_i$, with $F_{adv,i} = A_i U_{i,\tau}$. The two matrices $A_x$ and $A_y$ specifically are

$$A_x = \begin{pmatrix}
0 & 1 - \alpha_g & 0 & u_g - u_l & \alpha_g & 0 \\
0 & u_l & 0 & 0 & 0 & 0 \\
0 & 0 & u_l & 0 & 0 & 0 \\
0 & 0 & 0 & u_g & \alpha_g & 0 \\
0 & 0 & 0 & 0 & u_g & 0 \\
0 & 0 & 0 & 0 & 0 & u_g
\end{pmatrix},
A_y = \begin{pmatrix}
0 & 0 & 1 - \alpha_g & v_g - v_l & 0 & \alpha_g \\
0 & v_l & 0 & 0 & 0 & 0 \\
0 & 0 & v_l & 0 & 0 & 0 \\
0 & 0 & 0 & v_g & \alpha_g & 0 \\
0 & 0 & 0 & 0 & v_g & 0 \\
0 & 0 & 0 & 0 & 0 & v_g
\end{pmatrix}. \quad (40)$$

Introducing $A_x$ and $A_y$ in Eq. (17), the stabilizing terms to be added to each equation are obtained.

In particular, the stabilizing term for the momentum equation of the generic phase $q$ is equal to

$$S_{mq} = \sum_{e} \int_{\Omega_e} \left( \tau_{mq} \hat{v}^h_{q,j} \cdot \nabla \hat{w}^h_{i,j} \right) \tilde{r}^h_{mq} d\Omega; \quad (41)$$

where $\tilde{r}^h_{mq}$ is the residual of the discrete $q$-phase momentum equation. It is clear that the obtained stabilization term has the same expression as the one used in the momentum equation of the NS equations.

As far as the pressure equation (13) concerns, the stabilizing term is

$$S_p = \sum_{e} \int_{\Omega_e} (1 - \alpha^h_g) \left( \tau_{ml} \nabla w^p_{i,j} \right) \cdot \tilde{r}^h_{ml} d\Omega + \sum_{e} \int_{\Omega_e} \alpha^h_g \left( \tau_{mg} \nabla w^p_{i,j} \right) \cdot \tilde{r}^h_{mg} d\Omega.$$
\[ + \sum_e \int_{\Omega_e} \tau_\alpha (\bar{u}_g^h - \bar{u}_l^h) \cdot \nabla w_p^h r_\alpha^h d\Omega_e \]  

where \( \tilde{r}_{ml}^h \) is the residual of the discrete momentum equation of the primary phase, \( \tilde{r}_m^h \) the residual of the discrete momentum equation of the secondary phase, and \( \tilde{r}_\alpha^h \) is the residual of the continuity equation for \( \alpha_g \). When the flow reduces to single-phase flow, \( \alpha_g = 0 \) and its residual vanishes, so that the stabilizing term is equal to the PSPG term in equation (33).

5 PARALLELIZATION STRATEGY

Parallelization of flow solvers offers the possibility of achieving computational speeds that are many times faster than the single-processor computation. To enhance this goal, an efficient parallel algorithm has to be implemented. In the following some details on the parallel algorithm through the use of PETSc are shown, and the evaluation of the parallelization based on the actual speed up is presented.

5.1 Use of PETSc

Starting from the mesh, already partitioned in a certain number of subdomain, each processor reads the informations corresponding to the partition of the domain assigned, and in particular the local connectivity. So, it has to be intended that most of computation runs at a local level, while only when needed communication between the processes is used. Great attention must be paid in the introduction of the mapping relating the local indexing of the nodes to the global one, since it makes possible to share informations from different processes, and in the treatment of the ghost nodes (Fig. 1-a). Both instances rely in the introduction of the so-called ghost vector (Fig. 1-b), defined in PETSc. This vector is defined such that it has a local and a global representation. In the local representation, the vector is made of a certain number of positions corresponding to the local internal nodes, and of a number of positions corresponding to those nodes, the ghost nodes, belonging, at least, to one element of the local subdomain. All the ghost positions are located at the end of the vector. In the global representation, each local position is translated into a global one through an appropriate mapping, and to each ghost node position corresponds only one global position. Thanks to the introduction of these ghost vectors, the communication between different processes is made possible by switching between local and global representation of the vector and between local and global mapping. Similarly, it is possible to introduce a local and a global representation for the matrices.

The entire parallellization process can be split into three main computational components:

- construction of residual vector and mass matrix defining the problem, by means of ghost vector and mapping;
- preconditioning of the linear system;
solution of the preconditioned system.

Note the last two items are handled internally by the PETSc library.

5.2 Speed Up

The parallel performance of the present method is measured here in terms of _speed up_ (computational time of $N_{\text{proc}}$ processors divided by that of one processor) and _efficiency_ (speed up divided by $N_{\text{proc}}$).

In order to determine these parameters the test case described in Subsection 6.1 has been computed on a Beowulf HP Linux Cluster at the University of Basilicata. A crucial point in the determination of the speed up is the choice of the computational mesh, since it has to guarantee that a large number of nodes has to be assigned to each processor, respect to the number of ghost nodes. This is to ensure that the computation time is significantly larger than the time spent in the communication between processes.

The case considered is two-dimensional with a grid made of 67168 elements and 33965 nodes, which is enough refined for effective parallelization on a large number of processors. The computation has been performed till full convergence, starting from a single processor to 16 nodes.

The speed up obtained for the different number of processors is shown, as plot and table, in Fig. 2 (a) and (c). The results relative to 2 and 4 processors are reasonable, but they start to tail off for larger numbers, as easily predictable. On Fig. 2-(b) the speed up with $N_{\text{proc}}$ relative to $N_{\text{proc}}/2$ processors is reported: it shows how, increasing the number of processors, the relative parallelization benefit decreases.
6 NUMERICAL RESULTS

In this section, the results of the numerical simulation of the two-phase, laminar, steady flows in a channel and in a T-junction, already done with a serial code in [16], but here done with a parallel version, are presented.

6.1 Two-Phase Flow in a Channel

The first test case concerns with the two-dimensional two-phase laminar flow in a channel. Nevertheless this example presents a relatively simple geometry, the presence of the two phases renders the flow much more complex than the single phase one. In the present computations, no interactions between the two phases are considered (i.e. zero drag force).

The simulation is done by giving the velocity at the inlet and setting uniform pressure at the outlet. The pressure at the inlet comes from the computation. The other boundaries are treated as no-slip walls. The chosen inlet velocity profile is set to be uniform. The geometry is shown in Fig. 3. The Reynolds number is defined as

\[ Re = \frac{ud}{\nu} \]  

(43)

where \( u \) is the inlet velocity and \( \nu \) is the kinematic viscosity. Table 1 shows the physical properties of the two phases.
Table 1: Two-phase flow in a channel - physical properties.

<table>
<thead>
<tr>
<th></th>
<th>( Re_1 = 100 )</th>
<th>( Re_2 = 100 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reynolds Number</td>
<td>( Re_1 = 100 )</td>
<td>( Re_2 = 100 )</td>
</tr>
<tr>
<td>Kinematic Viscosity</td>
<td>( \mu_1 = 0.01 )</td>
<td>( \mu_2 = 0.005 )</td>
</tr>
<tr>
<td>Density</td>
<td>( \rho_1 = 1.0 )</td>
<td>( \rho_2 = 0.5 )</td>
</tr>
<tr>
<td>Inlet Volume Fraction</td>
<td>( \alpha_1 = 0.5 )</td>
<td>( \alpha_2 = 0.5 )</td>
</tr>
</tbody>
</table>

Different grids are considered: the coarsest grid consists of 1699 nodes and 3236 elements, and the finest one consists of 33965 nodes and 67168 elements. The grids shows a boundary layer refinement.

The comparison with the results given in ref. [17] is done for the horizontal velocity profiles of both phases along the center-line of the channel. The results, as seen in Fig. 4(a), show a quite good agreement with the reference solution. In Fig. 4(b) the dependency of the solution from the grid is also shown. Further comparisons are made for the pressure and the volume fraction. Figure 5(a) shows the static pressure along the centerline of the channel for the two different solutions. Regarding the volume fraction, the comparison is made through the values obtained along a vertical cut at \( x = 2.5d \). The results are shown in Fig. 5(b). The contour plots of the horizontal velocity for both the phases, and of the volume fraction of the secondary phase along the channel are presented (Figures 6-7), in order to assure a fair level of stabilization has been reached in the solution.

6.2 Two-Phase Flow in a T-Junction

The aim of this test case is to demonstrate the correctness of the implementation of the model and its accuracy for relatively complex flow patterns due to phase separation and mixing. The solution obtained is again compared with the one presented in [17]. The geometry of the problem is shown in Fig. 8.

The simulation is carried out setting the same uniform velocity \( u \) at the two inlets and uniform pressure at the outlet. The remaining boundaries are treated as no-slip walls.
Figure 4: Two-phase flow in a channel - comparison for horizontal velocity profiles along $y=0.5d$ between [17] and the present solution (a), and with different grids (b).

Figure 5: Two-phase flow in a channel - comparison between the present and the reference solution [17] for the pressure profiles along $y = 0.5d$ (a), and for the primary phase volume fraction profiles along $x = 2.5d$ (b).
Figure 6: Two-phase flow in a channel - contour plot for the secondary phase volume fraction.

Figure 7: Two-phase flow in a channel - contour plot for the primary (a) and the secondary (b) phase x-velocity.
Again the Reynolds number is defined as

\[ Re = \frac{ud}{\nu} \]  \hspace{1cm} (44)

On Tab. 2 the values of the physical properties of both phases are given.

<table>
<thead>
<tr>
<th></th>
<th>( Re_1 = 100 )</th>
<th>( Re_2 = 75 )</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Reynolds Number</strong></td>
<td>( Re_1 = 100 )</td>
<td>( Re_2 = 75 )</td>
</tr>
<tr>
<td><strong>Kinematic Viscosity</strong></td>
<td>( \mu_1 = 0.01 )</td>
<td>( \mu_2 = 0.0066 )</td>
</tr>
<tr>
<td><strong>Density</strong></td>
<td>( \rho_1 = 1.0 )</td>
<td>( \rho_2 = 0.5 )</td>
</tr>
<tr>
<td><strong>Inlet Volume Fraction</strong></td>
<td>( \alpha_1 = 0.5 )</td>
<td>( \alpha_2 = 0.5 )</td>
</tr>
</tbody>
</table>

Table 2: Two-phase flow in a T-junction - physical properties.

Several unstructured grids are considered for the simulation. The coarsest grid consists of 1676 nodes and 3158 elements, and the finest one consists of 9981 nodes and 19468 elements. The grids present a boundary layer refinement, to better reproduce the behaviour of the flow near the corners. In order to reproduce the results given in [17], the same drag force parameters are used. In particular

\[ C_D = 1 \quad \text{and} \quad d_g = 0.1 \]  \hspace{1cm} (45)

The comparison with the reference results [17] is carried out by plotting the horizontal and vertical velocity profiles along the center-line of the main channel of the T-junction. The results are shown in Fig. 9(a). The velocity profiles are not referred to a specific phase. This is because the phases have the same components of the velocity, due to the presence of the drag force. Good agreement is achieved with the reference solution. In Fig. 9(b) the grid dependency is also shown. As far as the volume fraction concerns, the comparison is carried out at two different locations: \( x = 3.5d \) and \( x = 6.5d \). Figures 10(a,b) show the results of [17] and the present solution. The agreement between the two
Figure 9: Two-phase flow in a T-junction - comparison for the velocity profiles along $y=0.5d$ between [17] and the present solution (a), and with different grids (b).

Figure 10: Two-phase flow in a T-junction - primary phase volume fraction profiles along $x=3.5d$ and $x=6.5d$: [17] (a) and present solution (b).
solutions is less good respect to the one obtained for the velocity profiles. Nevertheless, the maximum value is still comparable. The major differences are located especially near the walls.

Further plots are presented for the contour line of the pressure and the volume fraction along the all T-junction (see Figures 11-12). Regarding the volume fraction plot, the secondary phase accumulates near the lower wall after the junction, and the primary phase at the middle of the channel and at the top wall (see Figure 12). The secondary phase, having lower inertia, can better follow the geometry of the T-junction, and that is why it accumulates in the bottom wall.

7 CONCLUSIONS

In this paper a parallel Streamline Upwind Petrov-Galerkin finite element formulation for two phase-flows has been shown.

The mathematical formulation of the two-phase flow model has been given, together with the strategy to achieve the discretization and the stabilization of the set of the resulting equations. The system of equations has been discretized as an advective-diffusive set of equations and a SUPG stabilization has been performed by considering the advective matrix contributions. The parallelization issues and the advantage in term of speed up have been presented. The accuracy of the method has been proved by comparing the results, in the case of multiple meshes, for the channel and the T-junction flow problems with the ones given in ref. [17].

The results highlight that the choice of this approach is very promising to simulate two-phase flows. The results also suggest a better stabilization is needed in the boundary layer region, thus leading to the improvement of the discontinuity capturing operator acting in this region. Then, the parallel implementation suggests the computation of future more complex problems to test the stability and efficiency of the adopted algorithm.

REFERENCES

Figure 11: Two-phase flow in a T-junction - contour plot for the pressure.

Figure 12: Two-phase flow in a T-junction - contour plot for the secondary phase volume fraction.


