Weak (variational) formulation of Navier-Stokes equations and the corresponding finite element implementation

Mojtaba Barzegari

1 Navier-Stokes equations

In its general form, the Navier-Stokes equations describing the flow of an incompressible fluid with constant density ρ in the domain $\Omega \subset \mathbb{R}^d$ (with d being the dimension, so 2 or 3) can be written as:

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial t} - \nabla \cdot [\nu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + (\mathbf{u}.\nabla)\mathbf{u} + \nabla \mathbf{p} = \mathbf{f}, & x \in \Omega, t > 0, \\ \nabla \cdot \mathbf{u} = 0, & x \in \Omega, t > 0, \end{cases} \tag{1}$$

in which $\mathbf{u}$ is the fluid velocity, $\mathbf{p}$ is the pressure (which is actually pressure divided by the density), $\nu = \frac{\mu}{\rho}$ is the kinematic viscosity (with μ being the dynamic viscosity), and $\mathbf{f}$ is a force term. The equations are conservation of linear momentum and conservation of mass (also called continuity equation), respectively. When ν is constant, the diffusion term in Eq. 1 can be simplified as:

$$\mathrm{div}[\nu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] = \nu(\Delta \mathbf{u} + \nabla \mathrm{div} \mathbf{u}) = \nu \Delta \mathbf{u}, \tag{2}$$

which turns Eq. 1 into the following form:

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\, \mathbf{u} + \nabla p = \mathbf{f}, & x \in \Omega, t > 0, \\ \nabla \cdot \mathbf{u} = 0, & x \in \Omega, t > 0, \end{cases} \tag{3}$$

Eq. 3 satisfies the incompressibility condition $\nabla \cdot \mathbf{u} = 0$ and needs proper initial and boundary conditions to be well-posed. The initial condition can be defined as:

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \qquad \forall \mathbf{x} \,\epsilon\, \mathbf{\Omega}, \tag{4}$$

where $\mathbf{u}_0$ is a divergence-free velocity field. Various types of boundary conditions can be applied, but the ones we deal with in this chapter are described here. If $\partial\Omega$ is the boundary of Ω, it can be split into 3 distinct boundaries $\partial\Omega = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$ each of which with a different type. On Γ_1, the inlet can be defined as a Dirichlet boundary condition for the velocity for a given velocity profile $\mathbf{g}$:

$$\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma_1 \tag{5}$$

On Γ_2, a wall boundary no-slip condition can be considered:

$$\mathbf{u} = 0 \quad \text{on } \Gamma_2 \tag{6}$$

On Γ_3, for the outlet condition, a homogeneous Neumann condition on velocity and a zero pressure condition can be defined like:

$$\frac{\partial \mathbf{u}}{\partial n} = 0, \quad \mathbf{p} = 0, \quad \text{on } \Gamma_3 \tag{7}$$

with n being the normal direction on the boundary $\partial\Omega$. Broadly speaking, these boundaries can be grouped into 2 sets: $\Gamma_D = \Gamma_1 \cup \Gamma_2$ and $\Gamma_N = \Gamma_3$ for boundaries with Dirichlet and Neumann conditions, respectively.

The Navier-Stokes equations can be written componentwise for individual components of the flow vector field in the Cartesian coordinates. Denoting $u_i, i = 1, \ldots, d$ (with $d = 2$ in 2D and $d = 3$ in 3D), Eq. 3 can be presented as:

$$\begin{cases} \dfrac{\partial u_i}{\partial t} - \nu \Delta u_i + \displaystyle\sum_{j=1}^{d} u_j \dfrac{\partial u_i}{\partial x_j} + \dfrac{\partial p}{\partial x_i} = f_i, \qquad i = 1, \ldots, d, \\ \displaystyle\sum_{j=1}^{d} \dfrac{\partial u_j}{\partial x_j} = 0. \end{cases} \tag{8}$$

2 Weak formulation of the Navier-Stokes equations

For deriving the weak formulation, the first equation of 3 is multiplied by a test function v defined on a proper function space V in which the test functions vanish on the Dirichlet boundary:

$$V = [\mathbf{H}^1_{\Gamma_D}(\Omega)]^d = \{\mathbf{V} \in [\mathbf{H}^1(\Omega)]^d : \mathbf{v}|\Gamma_D = \mathbf{0}\}. \tag{9}$$

yielding to:

$$\int\limits_{\Omega} \frac{\partial \mathbf{u}}{\partial t}.\mathbf{v}\ d\omega - \int\limits_{\Omega} \nu \triangle \mathbf{u}.\mathbf{v} d\omega + \int\limits_{\Omega} [(\mathbf{u}.\nabla)\mathbf{u}].\mathbf{v} d\omega + \int\limits_{\Omega} \nabla p.\mathbf{v} d\omega = \int\limits_{\Omega} \mathbf{f}.\mathbf{v} d\omega. \tag{10}$$

Applying Green's divergence theory results in:

$$-\int_{\Omega} \nu \Delta \mathbf{u} \cdot \mathbf{v} d\omega = \int_{\Omega} \nu \nabla \mathbf{u} \cdot \nabla \mathbf{v} d\omega - \int_{\partial\Omega} \nu \frac{\partial \mathbf{u}}{\partial \mathbf{n}} \cdot \mathbf{v} d\gamma \tag{11}$$

and

$$\int_{\Omega} \nabla p \cdot \mathbf{v} d\omega = -\int_{\Omega} p \nabla \cdot \mathbf{v} d\omega + \int_{\partial\Omega} p \mathbf{v} \cdot \mathbf{n} d\gamma \tag{12}$$

Substituting Eqs. 11 and 12 into Eq, 10 yields to:

$$\begin{aligned} \int_{\Omega} \frac{\partial \mathbf{u}}{\partial t} \cdot \mathbf{v} d\omega + \int_{\Omega} \nu \nabla \mathbf{u} \cdot \nabla \mathbf{v} d\omega + \int_{\Omega} [(\mathbf{u} \cdot \nabla)\mathbf{u}] \cdot \mathbf{v} d\omega - \int_{\Omega} p \nabla \cdot \mathbf{v} d\omega \\ = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} d\omega + \int_{\partial\Omega} \left(\nu \frac{\partial \mathbf{u}}{\partial \mathbf{n}} - p\mathbf{n} \right) \cdot \mathbf{v} d\gamma \quad \forall \mathbf{v} \in V. \end{aligned} \tag{13}$$

The last term of Eq. 13 is expressed in accordance to the defined Neumann boundary condition, which vanishes on Γ_3 due to the defined condition in the current study (Eq. 7). Moreover, this term vanishes on the Dirichlet boundaries due to the properties of the function space V (Eq. 9).

Similarly, the second equation of 3 is multiplied by a test function q belonging to the function space Q, called the pressure space:

$$Q = \mathbf{L}^2_0(\Omega) = \{p \in L^2(\Omega) : \int\limits_{\Omega} p\ d\omega = 0\}, \tag{14}$$

resulting in:

$$\int\limits_{\Omega} q \nabla \cdot \mathbf{u}\ d\omega = 0 \qquad \forall q \in Q. \tag{15}$$

Eqs. 13 and 15 are so called weak forms of the Navier-Stokes equations.

3 Stokes equations

For viscous flow, where the Reynolds number is less than 1 ($Re = \frac{|\mathbf{U}|L}{\nu}$, with L and $\mathbf{U}$ being the representative length and velocity of the domain), the convection term of the Navier-Stokes equations can be neglected, simplifying Eq. 3 to:

$$\begin{cases} \alpha\mathbf{u} - \nu\Delta\mathbf{u} + \nabla p = f & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega, \end{cases} \tag{16}$$

with α being a positive coefficient. Eq. 16 can be used to model laminar flow in low Reynolds regimes and is simpler to handle than Eq. 3 from the numerical computing perspective. The weak formulation of the Stokes equation can be derived by following the approach taken for the Navier-Stokes equations in Section 2. The final form of the weak formulation is:

$$\begin{cases} \int\limits_\Omega (\alpha\mathbf{u}.\mathbf{v} + \nu\nabla\mathbf{u}.\nabla\mathbf{v})\, d\omega - \int\limits_\Omega p\nabla \cdot \mathbf{v}\, d\omega = \int\limits_\Omega \mathbf{f}.\mathbf{v}\, d\omega & \forall\mathbf{v} \in V, \\ \int\limits_\Omega q\nabla \cdot \mathbf{u}\, d\omega = 0 & \forall q \in Q, \end{cases} \tag{17}$$

Eq. 17 can be written in the standard finite element variational form by defining 2 bilinear terms $a : V \times V \mapsto \mathbb{R}$ and $b : V \times Q \mapsto \mathbb{R}$:

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) &= \int\limits_\Omega (\alpha\mathbf{u}.\mathbf{v} + \nu\nabla\mathbf{u}.\nabla\mathbf{v})\, d\omega, \\ b(\mathbf{u}, \mathbf{q}) &= -\int\limits_\Omega q\nabla \cdot \mathbf{u}\, d\omega, \end{aligned} \tag{18}$$

which causes the variational problem of the Stokes equation becomes to find $(\mathbf{u}, p) \in V \times Q$ such that

$$\begin{cases} a(\mathbf{u}, \mathbf{v}) + \mathbf{b}(\mathbf{v}, \mathbf{p}) = (\mathbf{f}, \mathbf{v}) & \forall\mathbf{v} \in V, \\ b(\mathbf{u}, \mathbf{q}) = 0 & \forall\mathbf{q} \in Q, \end{cases} \tag{19}$$

in which

$$(\mathbf{f}, \mathbf{v}) = \sum_{i=1}^{d} \int_\Omega f_i v_i d\omega. \tag{20}$$

4 Implementation

Numerical implementation of the Stokes (Eq. 16) and Navier-Stokes (Eq. 16) equations can be tricky due to the presence of specific sources of instability, which highly depends on the type of studied fluid regime. Various numerical models have been presented for dealing with these equations, some of which are commonly used in CFD applications, such as the Newton-Raphson approximation of Navier-Stokes equations and the Chorin's projection method.

In order to increase the stability and avoid problems in the mathematical analysis of the numerical models (e.g., V-ellipticity property), a pseudo-compressibility assumption can be added to the continuity equation. The pseudo-compressible approximation appears as a pressure term εp with ε being a very small coefficient, resulting in the following equation as the final form of the Navier-Stokes equations that we consider in this study:

$$\begin{cases} \dfrac{\partial\mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + (\mathbf{u} \cdot \nabla)\,\mathbf{u} + \nabla p = \mathbf{f}, \\ \nabla \cdot \mathbf{u} + \varepsilon p = 0. \end{cases} \tag{21}$$

Similarly, the Stokes equation can be written as:

$$\begin{cases} \alpha\mathbf{u} - \nu\Delta\mathbf{u} + \nabla p = f, \\ \nabla\cdot\mathbf{u} + \varepsilon p = 0. \end{cases} \tag{22}$$

Another challenging part is to approximate the convection terms in the equations. One of the best approaches to do so is to take advantage of the method of characteristics, in which the characteristics curves of a PDE are used to convert it to an ODE, resulting in a simpler solution. By using the method of characteristics for the convection term and a backward Euler discretization for the temporal term, the weak form of the Navier-Stokes and continuity equations (Eq. 21) can be rewritten as:

$$\begin{aligned} &\int_\Omega \frac{\mathbf{u}^{n+1} - \mathbf{u}^n \circ X^n}{\Delta t}\cdot\mathbf{v}d\omega + \nu\int_\Omega \nabla\mathbf{u}^{n+1}\cdot\nabla\mathbf{v}d\omega - \int_\Omega p^{n+1}\nabla\cdot\mathbf{v}d\omega = \int_\Omega \boldsymbol{f}\cdot\mathbf{v}d\omega \\ &\int_\Omega \nabla\cdot\mathbf{u}^{n+1}qd\omega + \varepsilon\int_\Omega p^{n+1}qd\omega = 0 \end{aligned} \tag{23}$$

in which $(\mathbf{u}^{n+1}, p^{n+1})$ are the unknowns to be computed from the known state $\mathbf{u}^n$ coming from the previous time step or the initial condition. In Eq. 23, the term $\mathbf{u}^{n+1} - \mathbf{u}^n \circ X^n$ is corresponding to the convection term being approximated using the method of characteristics.

The weak form of the Stokes equations stays almost the same as Eq. 17 (because it doesn't contain transient and convection terms) but needs a slight modification to add the pseudo-compressible terms from Eq. 22:

$$\begin{aligned} &\int\limits_\Omega (\alpha\mathbf{u}.\mathbf{v} + \nu\nabla\mathbf{u}.\nabla\mathbf{v})\,d\omega - \int\limits_\Omega p\nabla\cdot\mathbf{v}\;d\omega = \int\limits_\Omega \mathbf{f}.\mathbf{v}\;d\omega, \\ &\int\limits_\Omega q\nabla\cdot\mathbf{u}\;d\omega + \varepsilon\int_\Omega pqd\omega = 0. \end{aligned} \tag{24}$$

The model was implemented in the open-source PDE solver FreeFEM using P1 elements for the pressure and P2 elements for the velocity state variables. Eqs. 23 and 24 can be easily implemented in FreeFEM thanks to the built-in support of the method of characteristics via the `convect` function.

5 Preconditioning and parallelizing the computation

The solution of the Stokes and Navier-Stokes equations using the finite element method in 3D is a computationally intensive process, and as a result, taking advantage of high-performance techniques to reduce the simulation time becomes crucial in real-world applications. Preconditioning the system and parallelizing the simulation by partitioning the mesh and distributing the partitions among available computing nodes in a parallel computing setup are great solutions to this challenge.

In this implementation, the METIS graph partitioner and HPDDM package were used to partition the computational mesh and distribute the load over the available resources. For preconditioning and improving the solution time of the derived equations, various preconditioners and iterative or direct solvers available in the PETSc toolkit were tested to find the most suitable combination.

While exact factorization preconditioners (such as LU) are easy to implement and use for fluid flow applications, they show bad memory scaling profiles in large-scale problems, meaning that memory usage increases exponentially with the problem size, making it almost impossible to use them for 3D cases. A better solution for this class of problems is to take advantage of the FieldSplit preconditioner in the PETSc toolkit, which allows solving the derived linear system of equations using the block matrices technique. In this technique, the matrices are divided into smaller blocks, and separate preconditioners or solvers can be assigned to each block (each field). These blocks arise naturally from the underlying physics or numerical discretization of the problem, such as velocity and pressure in fluid flow applications.

For matrices with an arbitrary number of blocks, three different “block” algorithms are available in the PETSc toolkit: block Jacobi (`additive`), block Gauss-Seidel (`multiplicative`), and symmetric block Gauss-Seidel (`symmetric_multiplicative`), which can be set by passing the desired one to the `pc_fieldsplit_type` flag. For two blocks, like the one in fluid flow problems with velocity and pressure as the blocks, another family of solvers based on Schur complements can be used.

In the current study, the FieldSplit preconditioner with Schur complement approximation was used on two blocks for velocity and pressure. A GMRES KSP type was employed to solve the preconditioned system with an iterative solver. An Algebraic Multigrid (AMG) preconditioner was used for the velocity block, and a Jacobi preconditioner was assigned to the pressure block. The result of this configuration, as well as the request for appropriate monitoring tools, can be written as follows:

```
-ksp_monitor -ksp_converged_reason -ksp_type fgmres
-pc_type fieldsplit -pc_fieldsplit_type schur
-pc_fieldsplit_schur_fact_type full
-fieldsplit_velocity_pc_type gamg
-fieldsplit_velocity_ksp_type preonly
-fieldsplit_pressure_pc_type jacobi
-fieldsplit_pressure_ksp_max_it 5
```

to pass to PETSc while solving the equations.