"Efficient Numerical Approaches for Solving Navier-Stokes Equations in Computational Fluid Dynamics"

Prashanthi J

Department of mathematics, yuvaraja's college, University of Mysore, Mysuru-570005, India.

Prashanthi.jp.123@gmail.com

K Shivashankar,

Department of Mathematics, Yuvaraja's college, University of Mysore, Mysuru-570005, India

drksshankara@gmail.com

Abstract

The Navier-Stokes Equations (NSE) are central to fluid dynamics, defining the dynamics of fluid flow. Nevertheless, their solution, especially for turbulent flows, is still computationally expensive. Conventional numerical schemes such as the Finite Volume Method (FVM) are highly accurate but come at the cost of heavy computational loads. Conversely, deep-learning-based methods like the Fourier Neural Operator (FNO) offer computational efficiency but lack accuracy and stability in intricate flow conditions. This work presents a hybrid numerical scheme that combines FVM and FNO to achieve a balance between computational efficiency and accuracy. The hybrid method utilizes FVM for high-fidelity discretization and FNO for fast solution approximation, with an adaptive correction process to maintain numerical stability. The findings prove that the Hybrid FVM-FNO approach reduces computation time considerably with accuracy comparable to traditional solvers. Comparative results indicate that this hybrid approach achieves a 3× speedup compared to FVM with very high accuracy and is thus capable of real-time fluid simulation. This method has far-reaching implications in computational fluid dynamics (CFD) simulations, such as aerospace, weather forecasting, and biomedical flow studies.

Keywords: Keywords: Navier-Stokes Equations, Computational Fluid Dynamics, Finite Volume Method, Fourier Neural Operator, Hybrid Numerical Methods.

1 Introduction

The Navier-Stokes Equations (NSE) are a fundamental set of partial differential equations that regulate fluid motion. The Navier-Stokes Equations (NSE), initially formulated by Claude-Louis Navier in 1823 and subsequently refined by George Gabriel Stokes in 1845, are founded on key physical principles: Newton's law of viscosity, which connects shear stress in a fluid to the rate of deformation of fluid elements; the principle of mass conservation, asserting that the mass of an isolated system remains invariant over time; and Newton's second law, expressed as Force = mass × acceleration [1].

1.1 Fluid Dynamics and the Navier-Stokes Equations

The Navier-Stokes equations, formulated by Claude-Louis Navier and George Gabriel Stokes in 1822, are mathematical expressions used to ascertain the velocity vector field of a fluid, contingent upon specified beginning circumstances. They result from the use of Newton's second law in conjunction with a fluid stress (attributable to viscosity) and a pressure component. In almost all practical scenarios, they provide a system of nonlinear partial differential equations; nevertheless, with certain simplifications (such as one-dimensional motion), they may sometimes be transformed into linear differential equations. Typically, they remain nonlinear, making them difficult or unfeasible to solve; this is the source of turbulence and unpredictability in their outcomes [2].

The Navier-Stokes equations may be derived from fundamental conservation and continuity equations pertaining to fluid characteristics. To derive the equations of fluid motion, we must first establish the continuity equation, which specifies the conditions for conservation. Subsequently, we apply this equation to the conservation of mass and momentum, and ultimately integrate the conservation equations with a physical comprehension of fluid dynamics.

Numerical simulation of fluid dynamics is a significant domain within computational mathematics. Simulation now serves as both an alternative and a supplement to experiments across several engineering fields, aiding in the prediction of fluid behaviour [3]. The numerical resolution of the Navier-Stokes equations for turbulent flow is highly complex, necessitating substantial resolution due to the markedly distinct mixing-length scales present in turbulent flow. Achieving a stable solution demands the integration of high-order accurate methods on exceedingly fine computational grids [4].

1.1.1 Computational Fluid Dynamics (CFD)

Computational fluid dynamics pertains to the equations that govern fluid motion. Computational Fluid Dynamics (CFD) has several applications across various technological fields. CFD encompasses several methodologies that must be adhered to in order to get at the solution phase while addressing fluid dynamics problems. In 1822, Claude Louis Navier and

George Gabriel Stokes formulated the Navier-Stokes equations. This equation elucidates the movement of viscous fluid. Three discretization approaches are: a) finite difference, b) finite element, and c) finite volume. The objective is to discretize both space and time. The finite difference method (FDM) is often used in simulations to determine outcomes for multiphasic fluid and heat flow problems. The finite element method (FEM) discretizes time-independent systems in one, two, or three dimensions. The finite volume method (FVM) is a noteworthy methodology for three-dimensional systems [4]. The volume referenced in this methodology is used for discretization.

Since its initial development in the mid-twentieth century, CFD has been widely used across several industries and sectors because to its broad uses. Computational Fluid Dynamics (CFD) has been effectively used in several fluid mechanic's applications, including automotive and aeronautical aerodynamics, ship hydrodynamics, flow through pumps and turbines, and combustion processes. The majority of engineering challenges are addressed by analytical methods. Mathematical models are constructed over the fluid's area of interest, taking into account physical principles such as the conservation of mass, momentum, and energy. Assumptions about the suitable initial and boundary conditions are simplified to facilitate problem-solving [31]. Appropriate initial and boundary conditions must be used to elucidate the fundamental equations of fluid dynamics (Figure 1).

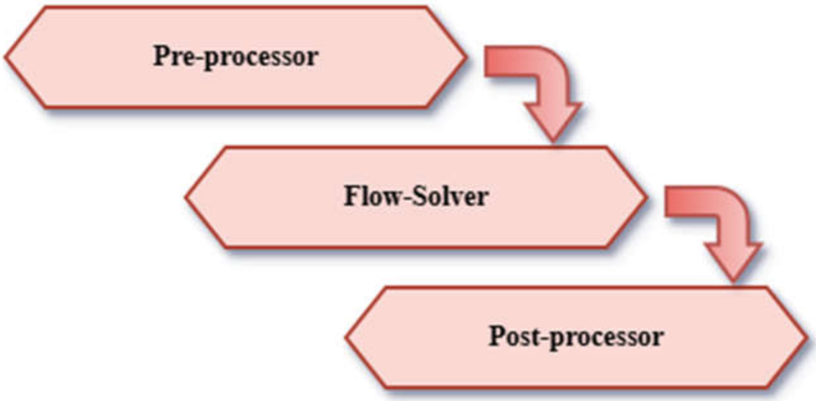

Figure 1 Three critical components of all computational fluid dynamics code [5].

1.2 Governing Equations in CFD

The basic rules may be used to generate the governing differential equations solved in a Computational Fluid Dynamics (CFD) research [6].

- Conservation of mass
- Newton's second law, or linear momentum conservation

- The first law of thermodynamics, or energy conservation

This analysis focused on the motion of single-phase fluids, namely liquids or gases, which will be treated as continua. The three principal unknowns derivable from solving these equations are, in fact, five scalar unknowns when considering the three velocity components individually.

- Velocity vector $\vec{V}$
- Pressure p
- Temperature T

However, in the governing equations that are solved numerically, four more variables emerge.

- Density p
- Enthalpy h (or internal energy e)
- Viscosity μ
- thermal conductivity k

Pressure and temperature may be regarded as two distinct thermodynamic variables that characterize the equilibrium state of the fluid. The four supplementary variables mentioned above are ascertained based on pressure and temperature by the use of tables, charts, or additional equations. Nonetheless, for several issues, it is feasible to regard p, μ, and k as constants, whereas h is considered proportional to T, with the proportionality constant being the specific heat c_p.

CFD programs are often developed exclusively for either compressible or incompressible flows because to the distinct mathematical characteristics of their governing equations. It is uncommon to encounter a code that can function successfully and correctly in both compressible and incompressible flow regimes. The following two parts will provide differential versions of the governing equations used for analyzing compressible and incompressible flows.

1.2.1 Conservation of Mass (Continuity Equation)

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \vec{V}\right) = 0 \qquad (1)$$

Or equally

$$\frac{D\rho}{Dt} + \rho\left(\nabla \cdot \vec{V}\right) = 0 \qquad (2)$$

The equations are recognized as the conservative and non-conservative variants of mass conservation, respectively. Conservation equations may be derived by applying the fundamental physics concept of mass conservation to a fluid element that is stationary in space. Non-conservative forms are derived by analyzing fluid elements in the flow field. The connection between these two equations may be determined using the following generic equation that correlates spatial and material representations of fluid flow.

$$\frac{DA}{Dt} = \frac{\partial A}{\partial t} + (\vec{V} \cdot \nabla)A \tag{3}$$

The quantity on the left side of this equation is the material derivative of property A. The first term on the right side is the partial time derivative, often known as the local derivative. The last term is known as the convective derivative of A.

1.2.2 Conservation of Linear Momentum

The equation for the conservation of linear momentum is often referred to as the Navier-Stokes equation. In computational fluid dynamics literature, the name Navier-Stokes typically encompasses both the momentum and continuity equations, and sometimes the energy equation as well. It may be expressed in several forms. One potentiality is

$$\rho \frac{D\vec{V}}{Dt} = -\nabla p + \nabla \cdot \bar{\bar{\tau}} + \rho \vec{f} \tag{4}$$

To use an Eulerian description, the material derivative on the left-hand side, representing the acceleration vector, may be substituted with the sum of local and convective accelerations to provide

$$\rho \left[\frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \nabla)\vec{V}\right] = -\nabla p + \nabla \cdot \bar{\bar{\tau}} + \rho \vec{f} \tag{5}$$

where $\vec{f}$ is the body force per unit mass. If the weight of the fluid is the sole body force, then replace $\vec{f}$ with the gravitational acceleration vector $\vec{g}$.

$\bar{\bar{\tau}}$ in the aforementioned equation represents the viscous stress tensor. In Newtonian fluids, viscous stresses are only dependent on the velocity gradient, exhibiting a linear relationship. It is also shown that $\bar{\bar{\tau}}$ must be symmetric to maintain the conservation of angular momentum. For a Newtonian fluid, the relationship between the stress tensor $\bar{\bar{\tau}}$ and the velocity components is as follows:

$$\tau_{ij} = \mu \left(\frac{\partial V_i}{\partial x_j} + \frac{\partial V_j}{\partial x_i}\right) + \lambda(\nabla \cdot \vec{V})\delta_{ij} \tag{6}$$

where x_i denote mutually perpendicular coordinate directions. μ is the dynamic viscosity and λ is known as the coefficient of bulk viscosity. It is related to the viscosity through the Stokes' hypothesis

$$\lambda + \frac{2}{3}\mu = 0 \tag{7}$$

After using this, the viscous stress tensor becomes into

$$\tau_{ij} = \mu\left(\frac{\partial V_i}{\partial x_j} + \frac{\partial V_j}{\partial x_i}\right) - \frac{2}{3}\left(\nabla \cdot \vec{V}\right)\delta_{ij} \tag{8}$$

Where δ_{ij} is the Kronecker-Delta operator which is equal to 1 if $i = j$ and it is zero otherwise. NavierStokes equation given in Eqn (1.5) is said to be in non-conservative form. A mathematically equivalent conservative form, given below, can also be derived by using the continuity equation and necessary vector identities

$$\frac{\partial}{\partial t}\left(\rho\vec{V}\right) + \nabla \cdot \left(\rho\vec{V} \otimes \vec{V}\right) = -\nabla p + \nabla \cdot \overset{=}{\tau} + \rho\vec{f} \tag{9}$$

Where, $\vec{V} \otimes \vec{V}$ is the tensor product of the velocity vector with itself, as seen below

$$\vec{V} \otimes \vec{V} = \begin{bmatrix} V_1V_1 & V_1V_2 & V_1V_3 \\ V_2V_1 & V_2V_2 & V_2V_3 \\ V_3V_1 & V_3V_2 & V_3V_3 \end{bmatrix} \tag{10}$$

The divergence of which is the following vector

$$\nabla \cdot \left(\vec{V} \otimes \vec{V}\right) = \left\{\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \frac{\partial}{\partial x_3}\right\} \begin{bmatrix} V_1V_1 & V_1V_2 & V_1V_3 \\ V_2V_1 & V_2V_2 & V_2V_3 \\ V_3V_1 & V_3V_2 & V_3V_3 \end{bmatrix} = \begin{Bmatrix} \frac{\partial V_1V_1}{\partial x_1} + \frac{\partial V_1V_2}{\partial x_2} + \frac{\partial V_1V_3}{\partial x_3} \\ \frac{\partial V_2V_1}{\partial x_1} + \frac{\partial V_2V_2}{\partial x_2} + \frac{\partial V_2V_3}{\partial x_3} \\ \frac{\partial V_3V_1}{\partial x_1} + \frac{\partial V_3V_2}{\partial x_2} + \frac{\partial V_3V_3}{\partial x_3} \end{Bmatrix} \tag{11}$$

In compressible flow simulations, Euler's equation is often used instead of Navier-Stokes. Euler's equation is produced by removing the viscosity factor from the Navier-Stokes equation, resulting in a first order PDE. It is often used to determine the pressure distribution of high-speed (and hence high Re) aerodynamic flows around/within flying bodies when viscous effects are compressed into extremely thin boundary layers. However, one must exercise caution when utilizing Euler's equation since it cannot accurately forecast flow fields with separation and circulation zones.

1.3 Flow Equations in Cartesian and Cylindrical Coordinate Systems

Observe tiny fluid elements shown in Fig. 2 - 5, which illustrate the flow field domain in Cartesian, cylindrical, and spherical coordinates, respectively. The term $K_{S1,S2}$ denotes a generic form of a flow field vector, where the subscripts S1 and S2 indicate the spatial components of the vector [7].

1.3.1 Continuity Equations

The continuity equation may be seen as a nonlinear diffusion equation with a regular drift factor, inspiring widespread applications in several domains, including crowd modelling [8], prediction of aircraft debris cloud growth [9], biomedical imaging [10], and curve measurement analysis [11]. The equation may be regarded as either an initial boundary problem [12] or a Cauchy problem [13]. The core physics of Continuity Equations is the mass conservation concept, introduced by Lavoisier in 1985. The conservation of mass is defined as the principle stating that the rate of change of mass inside a control volume (CV) is equal to the net rate of mass entering the CV [14], [15]. Examine the integral representation of the mass conservation equation:

$$\frac{\partial}{\partial t}\int_{CV}\rho.dV+\int_{CS}\rho v.\cancel{n}.dA=0,\quad \forall v\in R \tag{12}$$

Equation (12) may be converted to its differential form using Gauss' divergence theorem [20,21] to yield:

$$\dot{\rho}+\nabla(\rho v)=0 \tag{13}$$

1.3.1.1 Cartesian and coordinate Continuity equation

Consider figure. 2, the length of the infinitesimal fluid element in x, y, and z direction can be assigned as δx, δy and δz respectively. The term $K_{S1,S2}$ in figure. 2 can be defined as:

$$K_{S1,S2}=\langle K_{xx}\quad K_{yy}\quad K_{zz}\rangle=\left\{\langle\frac{\partial(\rho u_x)}{\partial x}\quad\frac{\partial(\rho u_y)}{\partial y}\quad\frac{\partial(\rho u_z)}{\partial z}\rangle.V_{car}\middle|V_{car}=\delta x\delta y\delta z\right\} \tag{14}$$

The other κ are zero due to the non-slip boundary condition and by substituting Eq. (14) into Eq. (12),

$$\frac{\partial\rho}{\partial t}+\frac{\partial(\rho u_x)}{\partial x}+\frac{\partial(\rho u_y)}{\partial y}+\frac{\partial(\rho u_z)}{\partial z}=0 \tag{15}$$

Taking, $v=[u_x\quad u_y\quad u_z]$ in an incompressible flow, Eq. (14) will be reduced to ∇.v = 0 where:

$$\nabla = +\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z} \tag{16}$$

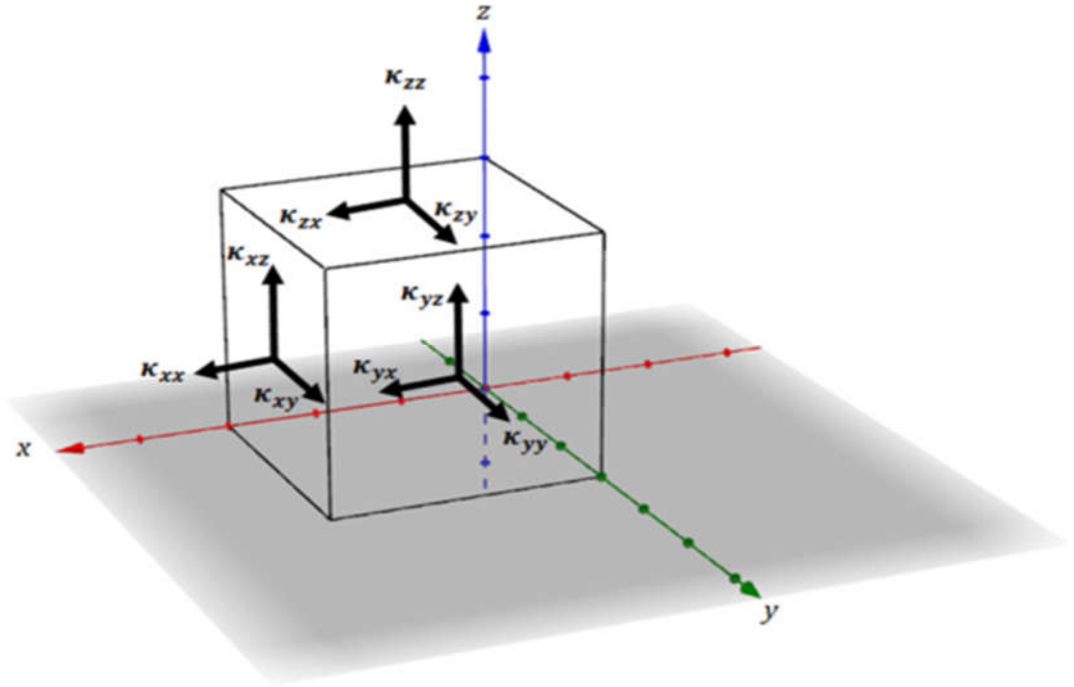

Figure 2 Infinitesimal fluid field domain based on Cartesian coordinate

1.3.1.2 Cylindrical and coordinate Continuity equation

Consider figure. 3, the length of the infinitesimal fluid element in r, θ, and z direction can be assigned as δr, $\delta\theta$ and δz respectively. Upon dimensional expansion, these distances will evolve as r+δr, θ+δθ and z+δz respectively. The term $K_{S1,S2}$ in figure 3 can be further defined as:

$$K_{S1,S2} = \langle K_{r,r} \quad K_{\theta,\theta} \quad K_{z,z}\rangle = \left\{\langle\frac{\partial(\rho u_r)}{\partial r} + \frac{\rho u_r}{r} \quad \frac{1}{r}\frac{\partial(\rho u_\theta)}{\partial \theta} \quad \frac{\partial(\rho u_z)}{\partial z}\rangle . V_{cyn} \middle| V_{cyn} \approx r\delta\theta\delta r\delta z\right\} \tag{17}$$

The value of other non-normal κ is zero too due to the non-slip boundary condition. Note that $\delta r^2 \approx 0$ during the derivation due to its infinite proximity to zero. The volume of the cylinder is:

$$V_{cy} = (\pi(r+\delta r)^2 - \pi r^2) \times \frac{\delta\theta}{2\pi} \times \delta z \approx r\delta\theta\delta r\delta z \tag{18}$$

Substitute Eq. (17) into Eq. (12) will yield:

$$\frac{\partial\rho}{\partial t} + \frac{\rho u_r}{r} + \frac{\partial(\rho u_\theta)}{\partial\theta} + \frac{\partial(\rho u_z)}{\partial z} = 0 \tag{19}$$

If the flow is incompressible, Eq. (19) can be simplified into ∇.v = 0 too with the divergence term as in Eq. (20), provided that the velocity vector is $v = [u_r \quad u_\theta \quad u_z]$.

$$\nabla = \frac{1}{r}\frac{\partial(r)}{\partial r} + \frac{\partial}{\partial \theta} + \frac{\partial}{\partial z} \tag{20}$$

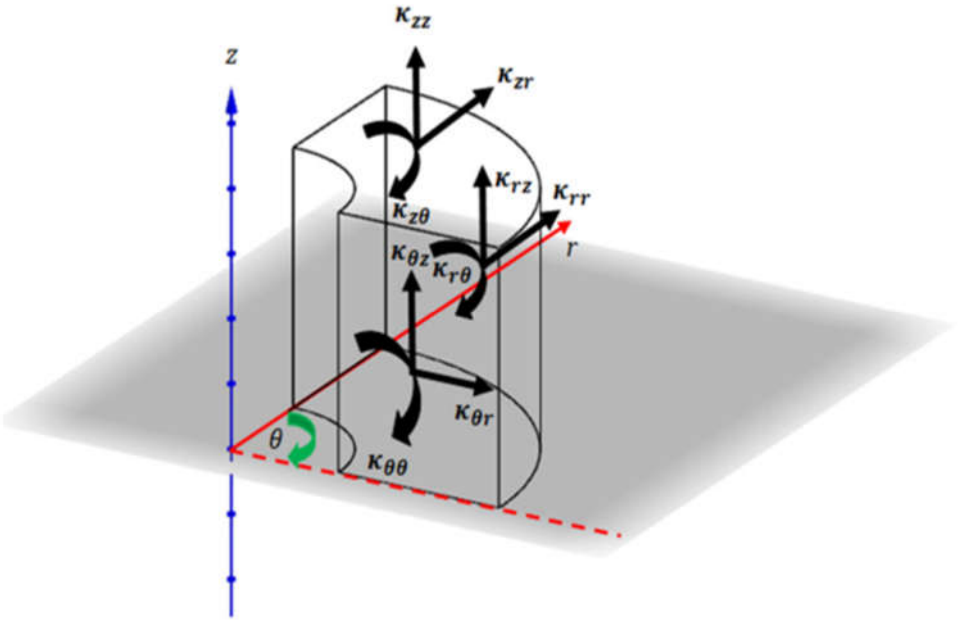

Figure 3 Infinitesimal fluid field domain based on cylindrical coordinate

1.3.1.3 Spherical coordinate Continuity equation

Consider figure 4, the length of control volume boundary and surface of the facets for the spherical fluid element. All the κ will be zero as well, except $K_{rr}, K_{\theta\theta}, K_{\phi\phi}$ which can be mathematically expressed as in Eq. (21).

$$K_{S1,S2} = \langle K_{rr} \quad K_{\theta\theta} \quad K_{\phi\phi}\rangle$$

$$= \langle\left(\rho u_r + \frac{\partial(\rho u_r)}{\partial r}\delta r\right)A_{EFGH} - (\rho u_r)A_{ABCD} \quad \left(\rho u_\theta + \frac{\partial(\rho u_\theta)}{\partial \theta}\delta\theta\right).A_{BDFG} - (\rho u_\theta)A_{ACEH} \quad \left(\rho u_\phi + \frac{\partial(\rho u_\phi)}{\partial \phi}\delta_\phi\right).A_{CDGH} - (\rho u_\phi).A_{ABEF}\rangle$$

$$= \left\{\langle\frac{\partial(\rho u_r)}{\partial r} + \frac{2\rho u_r}{r} \quad \frac{1}{r}\frac{\partial(\rho u_\theta)}{\partial \theta} + \frac{\rho u_\theta}{r} \quad \frac{\partial(\rho u_\phi)}{\partial \phi}\rangle.V_{sph}\middle|V_{sph} \approx r^2\sin[\theta]\delta r\delta\theta\delta\phi\right\} \tag{21}$$

The volume of spherical element can be approximated by taking the product of AACEH and L_{CD^*}, or using the Jacobian rules for the derivation. Substitute Eq. (20) into Eq. (1) will form the compressible Continuity equation as in Eq. (21), in which ∇.v = 0 where $v = [u_r \quad u_\theta \quad u_\phi]$ will be applied in incompressible case where its divergence term is shown in Eq. (22).

$$\frac{\partial\rho}{\partial t} + \frac{2\rho u_r}{r} + \frac{\partial(\rho u_r)}{\partial r} + \frac{\rho u_\theta}{r}\cot(\theta) + \frac{1}{r}\frac{\partial(\rho u_\theta)}{\partial\theta} + \frac{1}{\mathrm{rsin}(\theta)}\frac{\partial(\rho u_\phi)}{\partial\phi} = 0 \tag{22}$$

$$\nabla = \frac{1}{r^2}\frac{\partial(r^2)}{\partial r} + \frac{1}{\mathrm{rsin}(\theta)}\frac{\partial(\sin\theta)}{\partial\theta} + \frac{1}{\mathrm{rsin}\theta}\frac{\partial}{\partial\phi} \tag{23}$$

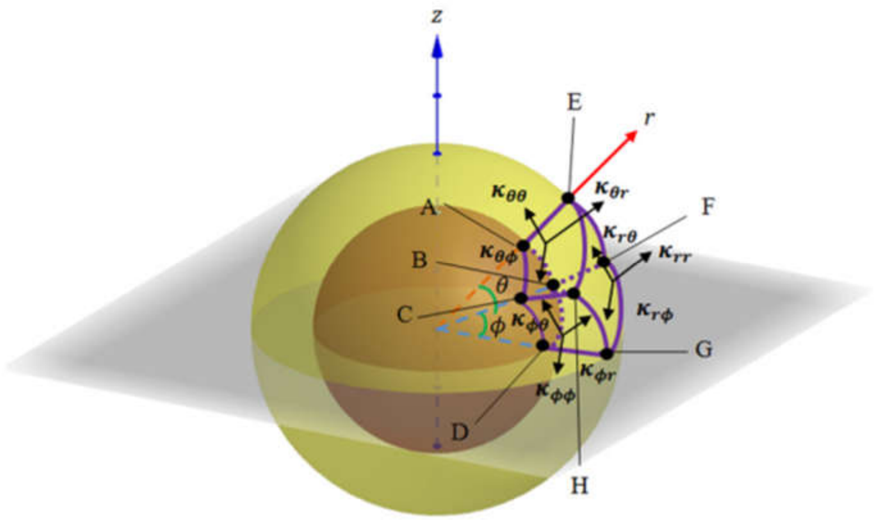

Figure 4 Infinitesimal fluid field domain based on spherical coordinate

1.3.2 Navier-Stokes Equations

Momentum equations derive from Newton's second law, which asserts that the force acting on a moving object is equal to its rate of change of momentum. Extending the formulation will provide the momentum equations in a generic integral format of [16]:

$$\sum F = \frac{\partial}{\partial t}\int_{CV}\rho v.dV + \int_{CS}\rho(vv).\vec{n}.dA, \quad \forall v \in R \tag{24}$$

The first component of Eq. (24) signifies the body forces, including gravity, Coriolis effects, centrifugal force, and electromagnetic force; while the subsequent term indicates the surface forces, often pertaining to pressure force and viscous force. If the flow is in a steady condition, then the sum of forces $\sum F$ will be nullified. Equation (24) was elaborated by French mathematician Augustin Louis de Cauchy [39] into a differential term using the divergence theorem as follows:

$$\frac{\partial}{\partial t}(\rho v) + \nabla.(\rho vv) = \rho g + \nabla.\vec{\sigma}_{ij} \tag{25}$$

$$\frac{\partial}{\partial t}(\rho v) + \nabla.(\rho vv) = \rho\left(\frac{\partial v}{\partial t} + (v.\nabla)v\right) = \rho\frac{Dv}{Dt} \tag{26}$$

Equation (26) represents the material derivative, which is sometimes referred to as the total, particle, Lagrangian, Eulerian, or significant derivative [17]. It denotes the convection phenomena, and its cessation signifies the emergence of creeping flow. ρg and the gradient operator $\nabla.\vec{\sigma}_{ij}$ denotes body force and the aggregate of applied surface forces, respectively. Nonetheless, the precise formulation of the divergence component ∇ will differ among coordinate systems. Equation (25) requires additional refinement and will progress similarly to the renowned Navier-Stokes equations [18]. This indicates that only Newtonian fluids are addressed, but the non-linear connection between the velocity field and stress tensor present in non-Newtonian fluids necessitates further modelling [19], [20].

1.4 Traditional Numerical Methods for CFD

According to [21] Numerical methods include classic CFD techniques like the finite difference method (FDM), finite volume method (FVM), and finite element method (FEM). By discretizing the Navier-Stokes equations and using appropriate boundary conditions, these methods solve them [22],[23].

1.4.1 Finite Difference Method (FDM)

The finite difference method, a well-recognized approximation technique, provides solutions for partial differential equations. It has been used for the settlement of many difficulties. Issues may be categorized as linear, nonlinear, time-independent, and time-dependent problems. The finite differences approach is both simple to program and has a significant history. It is used just in select specialized algorithms that handle complex geometries with remarkable efficiency and precision via the use of integrated boundaries or superimposed grids. These methods may be found in many distinct software applications. The Finite Difference Method (FDM) is a numerical approach often used to solve partial differential equations that describe fluid flow and heat transport in porous media [24] [25].

In FDM, the area of interest is divided into a grid, and the partial differential equations are transformed into algebraic equations by finite difference approximations [26]. The algebraic equations are then resolved repeatedly by numerical procedures, like the Gauss–Seidel method or the successive over-relaxation approach. One benefit of FDM is its simplicity and computational efficiency; nevertheless, its accuracy is limited by the grid size and the order of approximation used [72,73].

In [27] The two-dimensional unsteady Navier-Stokes equation was numerically discretized using the projection technique in finite difference and the SIMPLE algorithm in finite volume on a staggered grid system. The projection technique was implemented using a Python script and confirmed using a conventional CFD test case: driven cavity flow. Potential influencing variables such as grid size and boundary conditions were investigated, and the findings were very consistent with those from the benchmark work.

1.4.2 Finite Volume Method (FVM)

The finite volume method is a technique designed to discretize partial differential equations, particularly those derived from physical conservation laws. FVM employs a volume-integrated problem framework to discretize these equations using a limited collection of divided volumes. The finite volume method is often used in computational fluid dynamics for complex scenarios with high Reynolds number turbulent flows. The approach conserves time and memory resources. The establishment of the governing partial differential equations in a conservative format. The Navier-Stokes equations typically include mass conservation, energy conservation, and turbulent dynamics. These are then addressed using the finite volume method across discrete control volumes [28].

1.4.3 Finite Element Method (FEM)

The finite element technique, often referred to as FEM, is widely used in engineering and mathematics for the numerical resolution of differential equations. Numerous diverse academic disciplines include structural analysis, thermal transfer, fluid dynamics, and mass transport. The finite element method is often used in the study of solid structures, but it may also be applied to fluids. This formulation needs considerable effort to achieve the intended outcome of caution. The formulation has been updated to align with the governing equations of fluid dynamics. Although it requires precise specification to ensure conservativeness, the finite element technique is far more dependable than the finite volume method. The finite element approach may need more memory and take longer to compute than the finite volume method [5].

The Finite Element Method (FEM) is a numerical approach used to provide an approximate solution for various situations. The process entails discretizing the domain of interest into elements and articulating the equations in weak form for each element. The resultant equations are then solved numerically using techniques such as Newton–Raphson or conjugate gradient [29],[30]. Finite Element Method (FEM) accommodates irregular geometries and delivers precise solutions; nevertheless, it necessitates more computer resources in comparison to Finite Difference Method (FDM) or Finite Volume Method (FVM) [31]. The Finite Element Method (FEM) is a robust technique extensively used in several commercial software applications, including NASTRAN and ANSYST, facilitating fast and precise analysis of diverse engineering challenges, including structural mechanics, fluid dynamics, heat transport, and electromagnetics.

2 Related Work

Baranovskii et.al., [32] provided exact solutions for describing unidirectional, shear, and three-dimensional flows of a micropolar viscous incompressible fluid. It also presents new boundary value problems for generalized classical Couette, Stokes, and Poiseuille flows, which are created by non-uniform shear stresses and velocities. The study also discusses isobaric shear flows, which are described by an overdetermined system of nonlinear partial differential equations. The article provides a condition for the solvability of the overdetermined system of equations and constructs a class of nontrivial solutions for describing isobaric fluid flows.

Klinteberg et.al., [33] presented a comprehensive integral equation-based flow solver that integrates newly discovered techniques for singular quadrature and the resolution of PDEs on intricate domains, with various proven numerical approaches. The research applied this solution to flow problems across several geometries, both simple and complex, examining its convergence characteristics and computing efficiency. This illustrates that developing a robust, efficient, and adaptable Navier-Stokes solver utilizing integral equation approaches is now quite simple.

Berselli and Chiodaroli [34] examined energy conservation in solving the initial boundary value issue for 3D Navier-Stokes equations with Dirichlet boundary conditions.First, they analyse Leray-Hopf weak solutions and verify additional criteria using velocity gradient. Next, they compare them to scaling invariant space literature and the Onsager conjecture. Next, they address energy conservation for weak solutions and demonstrate energy equality for distributional solutions in the Shinbrot class. They provide a plausible explanation for the non-scaling invariant role of classical solutions.

Bistafa [35] examined the Navier-Stokes equation's historical development and its important impact on fluid dynamics during the last 200 years. They follow the development of this equation from Navier's original discoveries to George Stokes' experimental confirmations and later additions from other researchers. They also explored its real-world uses, such as its contribution to the creation of computational fluid dynamics. Our knowledge of fluid dynamics has advanced significantly thanks to the Navier-Stokes equation.

In [36] this study Berselli and Kaltenbach looks at a finite element approximation of the steady-Navier-Stokes equations (where is variable dependant). By imposing natural fractional regularity assumptions on the velocity vector field and the kinematic pressure, they demonstrate orders of convergence. In contrast to other findings, they use a more workable discretization of the power-law index and address the convective term. Regarding fractional regularity assumptions on the velocity vector field and the kinematic pressure, numerical measurements verify the quasi-optimality of the a priori error estimates (for the velocity).

De et.al., [37] employed (extended) neural networks influenced by physics, they establish precise limitations on the errors that arise from approximating the incompressible Navier-Stokes equations. They demonstrate that for tanh neural networks with two hidden layers, the underlying PDE residual may be made arbitrarily tiny. Furthermore, the training error, network size, and number of quadrature points may all be used to determine the overall error. Numerical tests are used to demonstrate the theory.

Yelnyk et al., [38] Investigated the use of Radial Basis Functions (RBFs) in fluid dynamics issues. The stationary Stokes and Navier-Stokes equations are specifically addressed using the RBF collocation approach. An established method from the literature is augmented with an extra polynomial basis and a novel preconditioner. A more efficient technique using the division of unity is shown for stationary Stokes equations. A global technique using Picard linearization is presented for stationary Navier-Stokes equations.

In [39] Ershkov investigated the stability of previously acquired helical flows. Stability requirements for the precise solution of the specified flow types are derived, focusing on non-stationary helical flow with a constant Bernoulli function. The spatial component of the fluid flow's pressure field should be ascertained using the Bernoulli equation, provided that the flow velocity components have previously been calculated.

Baymani et. al, [40] presented a new method using neural networks to solve Navier-Stokes equations in analytical function form. The method involves forming a trial solution consisting of two parts: one satisfying boundary conditions without adjustable parameters, and the other satisfying the governing equation inside the solution domain with adjustable parameters. The method's capabilities are demonstrated by solving Navier-Stokes problems with different boundary conditions. The method's performance and accuracy are evaluated by comparing it with available numerical and analytical solutions. The details of the method are discussed and illustrated through various boundary conditions.

Rodolfo et.al., [41] discussed the potential of a novel Lagrangian formulation for solving incompressible Navier-Stokes equations with large time steps. The paper introduces the origin of this numerical method, inspired by the Particle Finite Element Method (PFEM), and summarises its moving mesh version. It then introduces its extension to a fixed mesh version, detailing its implementation. The study reveals that this method, originally designed for heterogeneous or free-surface flows, can compete with Eulerian alternatives in terms of accuracy and robustness, allowing for stable use of large time steps.

Sinchev et al., [42] explored the use of neural networks and radial basis functions for solving 2-dimensional Navier-Stokes equations. It develops an algorithm for hydrodynamic equations using weighted residuals and general neural network approximation. The study demonstrates the potential of neural network modelling for hydrodynamic modelling, highlighting its ease of implementation, accuracy, and smoothness. The neural network is proposed as an approximation of the unknown equation solution, using the Gaussian distribution as the activation function.

Burmasheva., [43] highlighted the importance of constructing exact solutions to the dynamics of viscous fluids stratified by physical characteristics, such as density and viscosity. It also discusses the application of these exact solutions to modelling technological processes dealing with moving viscous fluid media. The paper constructs a class of exact solutions to the Navier-Stokes equations for viscous multilayer media in a mass force field, which is extended to arbitrary kinetic force field relations to all three Cartesian coordinates and time. The paper also addresses issues of over determination and solvability of the reduced Navier-Stokes equation system supplemented by the incompressibility equation. It also discusses three approaches to obtaining consistency conditions for the overdetermined reduced system of motion equations.

In [44] Mohammadein et.al., converted nonlinear Navier-Stokes equations to linear diffusion equations using the linear velocity operator. The Picard method is used to obtain the simplest analytical solutions for various wave lengths and Reynolds numbers. The peristaltic incompressible viscous Newtonian fluid flow in a horizontal tube is described using continuity and linear Navier-Stokes equations. The analytical solutions are obtained in terms of stream function and fluid velocity components, and plotted in laminar, transit, and turbulent flows.

In spite of all these contributions, some gaps in the research still exist the research gap emerged from this literature review indicates the necessity for a flexible, stable, and computationally effective numerical approach for solving Navier-Stokes equations to solve a broad category of fluid flow situations, including turbulent, three-dimensional flows. The majority of studies concerned with single numerical approaches, but intensive comparisons of assorted methods under the different flow situations, turbulence closures, and boundary conditions are limited. Although conventional techniques such as FDM, FEM, and FVM provide stability and accuracy, their computation intensity and complexity are still issues. On the other hand, recent methods like neural networks and hybrid methods hold promise but need to be further tested and optimized for large-scale and dynamic fluid flows. Filling these gaps through the creation of stable, scalable, and generalizable numerical algorithms may substantially improve the accuracy and efficiency of solving Navier-Stokes equations in intricate fluid flow environments.

3 Problem Statement

Nevertheless, advancement in numerical techniques, resolving the Navier-Stokes equations (NSEs) continues to be computationally intensive under dynamic fluid flow circumstances. Conventional techniques including the Finite Difference Method (FDM), Finite Element Method (FEM), and Finite Volume Method (FVM) provide stability and precision but are hindered by substantial computing expenses. In contrast, deep learning and hybrid methodologies, such as Physics-Informed Neural Networks (PINNs) and Fourier Neural Operators (FNOs), provide potential alternatives but require more refining to effectively manage complicated, large-scale simulations. The suggested methodology seeks to bridge the stated research gap by creating a hybrid numerical approach that integrates the stability and precision of conventional numerical methods with the computational efficiency and generalization capacity of deep learning approaches. This paper offers a Hybrid Fourier Neural Operator (FNO) and Finite Volume Method (FVM) technique to address the Navier-Stokes equations (NSE) in turbulent, dynamic fluid flow scenarios. This hybrid method will improve computational efficiency while preserving numerical stability and accuracy over a wide range of flow conditions.

4 Methodology

The Navier-Stokes equations characterize the movement of fluid substances and are essential in fluid dynamics. Resolving these equations, particularly for turbulent flows, is computationally demanding. We propose a hybrid methodology that integrates the Finite Volume Method (FVM) with the Fourier Neural Operator (FNO) to capitalize on the advantages of both techniques for efficient and precise fluid flow simulations.

4.1 Problem Formulation and Discretization using FVM

The incompressible Navier-Stokes equations are expressed as follows:

Continuity Equation (Mass Conservation):

$$\nabla \cdot u = 0, \tag{4.1}$$

Where, $u = (u, v, w)$ is the velocity vector in three dimensions in the x, y, and z directions. This guarantees the preservation of mass inside the fluid.

$$\frac{\partial u}{\partial t} + (u \cdot \nabla)u = -\nabla p + \nu\nabla 2u + f \tag{4.2}$$

In this context, u denotes the velocity field, p signifies the pressure, ν represents the kinematic viscosity, and f indicates external forces.

These equations delineate the temporal evolution of fluid momentum influenced by pressure, viscous forces, and external forces.

4.2 Discretization using the Finite Volume Method (FVM)

The Finite Volume Method (FVM) is used to discretize the Navier-Stokes Equations (NSEs) owing to its conservation characteristics and capacity to manage intricate geometries. The spatial domain is partitioned into control volumes, and the governing equations are integrated over each volume.

Applying Gauss's divergence theorem, the integral form of the momentum equation is expressed as:

$$\frac{d}{dt}\int_V \rho u dV + \oint_S \rho u(u \cdot n)\, dS = -\oint_S pndS + \oint_S \nu\nabla u \cdot ndS + \int_V FdV \tag{4.3}$$

where S represents the control volume surface and n denotes the outward normal.

The Finite Volume Method (FVM) converts this equation into algebraic form by using finite difference approximations for both convective and diffusive variables.

4.3 Fourier Neural Operator (FNO), for Effective Solution Approximation

The Fourier Neural Operator (FNO) is used to expedite the solution process by learning mappings from input circumstances to velocity and pressure fields. In contrast to conventional neural networks, FNO functions in the frequency domain using Fourier transformations, making it particularly effective for solving PDEs such as the Navier-Stokes equations.

The basic mode of operation of FNO is through three principal transformations:

Fourier Transform: The velocity and pressure field representation in space is transformed into the frequency domain by the Fourier transform. This allows the flow structures to be decomposed into their spectral content.

Spectral Layer: A learned transformation is applied directly in the Fourier domain. The frequency components are transformed by a trainable filter, enabling the model to effectively learn fluid dynamics.

Inverse Fourier Transform: The frequency transformed components are converted back to the spatial domain and used to reconstruct the velocity and pressure fields for the next time step.

The transformation function is represented as:

$$u_{t+1} = \sigma\left(Wu_t + F^{-1}(R \cdot F(u_t))\right) \tag{4.4}$$

Where, F and F^{-1} indicates the Fourier Transform and its inverse respectively. R is a learned complex-valued filter that operates in Fourier space. W is a trainable weight matrix that applies a linear transformation in the spatial domain. σ is a nonlinear activation function that introduces nonlinearity in the mapping process.

Through this, FNO accurately captures local and global flow structures with efficiency and is thus exceptionally good at forecasting the evolution of fluid flows with varying boundary conditions and Reynolds numbers.

4.4 Training Process of FNO

FNO is trained to learn the mappings from input flow conditions (such as initial velocity and boundary conditions) to the velocity and pressure fields. This allows the model to generalize across various flow regimes.

To ensure physical correctness, an adaptive correction strategy is presented wherein FVM adjusts FNO predictions when they go against conservation principles.

After training, FNO can make rapid predictions of fluid flow evolution with considerable reduction in the computational time in comparison to classical numerical solvers.

4.5 Hybrid Numerical Method

The suggested hybrid method combines the Finite Volume Method (FVM) and the Fourier Neural Operator (FNO) to obtain the best compromise between computational efficiency and physical fidelity in the solution of the Navier-Stokes equations. The algorithm starts with an initialization phase, in which the FVM solver generates a high-fidelity solution through discretized governing equations while correctly enforcing boundary conditions and conservation principles.

The prediction phase, which is applied after initialization, relies on FNO to project velocity and pressure fields into the future by exploiting trained representations in learned data and consequently speeds up simulation. Since FNO acts as a data-driven method, inaccuracies will begin to compound themselves over time by numerical instability or loss of adherence to physical restrictions. To counteract such problems, an adaptive correction mechanism is implemented in which discrepancies of predicted values from basic conservation concepts (e.g., mass and momentum conservation) are measured with an error tolerance. If the discrepancy crosses a threshold value, the FVM solver is activated again to rectify inconsistencies, maintaining numerical stability. The correction can be mathematically represented as:

$$u_{corrected} = \alpha u_{FNO} + (1-\alpha)u_{FVM} \tag{4.5}$$

where $\alpha \in [0,1]$ is an adaptive weighting parameter dynamically updated according to the magnitude of the error between the FNO-predicted and FVM-corrected fields. The time-stepping procedure utilizes an adaptive switching strategy, where the solver switches between FVM and FNO, balancing computational efficiency with physical accuracy. The switching criterion is given by:

$$\delta = \|u_{FNO} - u_{FVM}\| \tag{4.6}$$

where δ represents the error measure; if δ is greater than a critical level ϵ, invoke the FVM step; else, FNO proceeds with prediction. This combined approach efficiently hastens time-stepping, such that large fluid simulations become tractable computationally, while adhering to laws of physics, stability requirements, and conservation relations.

Algorithm 1 Hybrid FVM-FNO Simulation

```
Initialize u, p using FVM with initial and boundary conditions.
for $t = 1$ to max time steps do
    Predict next state using FNO: $u_{pred}, p_{pred} \leftarrow predictFNO(u,p)$
    Compute errors: $\epsilon_1 \leftarrow$ compute divergence($u_{pred}$), $\epsilon_2 \leftarrow$ compute momentum residual $(u_{pred}, p_{pred})$
    if $\epsilon_1 >$ tolerance or $\epsilon_2 > tolerance$ then
        Apply correction using FVM: $u_{corr}, p_{corr} \leftarrow$ solve FVM correction $(u_{pred}, p_{pred})$
        Compute adaptive weight: $\alpha \leftarrow$ compute adaptive weight $(\epsilon_1, \epsilon_2)$
        Update solution: $u \leftarrow \alpha u_{pred} + (1-\alpha)u_{corr}, p \leftarrow \alpha p_{pred+}(1-\alpha)p_{corr}$
    else
        Accept FNO prediction: $u, p \leftarrow u_{pred}, p_{pred}$
    end if
end for
return $u, p$
```

This paper introduces an innovative Hybrid FNO + FVM method for effectively solving the Navier-Stokes equations while preserving physical consistency. This approach improves accuracy, generalization, and computing speed by using FNO's learning capability and FVM's stability. This work contributes to accelerating the CFD simulations in aerospace, climate modelling, medicinal flows, and engineering applications etc.

5 Results and Discussion

5.1 Accuracy Assessment

The suggested method's accuracy is assessed using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) of the velocity and pressure fields, in comparison to ground truth solutions.

Table 1 Accuracy Assessment

Test Case	Solver	MAE (Velocity)	RMSE (Velocity)	MAE (Pressure)	RMSE (Pressure)
Lid-Driven Cavity (Re = 1000)	FVM	0.0035	0.0071	0.0042	0.0084
	FNO	0.0067	0.0125	0.0079	0.0152
	Hybrid FVM-FNO	0.0024	0.0049	0.0031	0.0062
Flow Past a Cylinder (Re = 5000)	FVM	0.0041	0.0087	0.0052	0.0103
	FNO	0.0092	0.0184	0.0113	0.0226
	Hybrid FVM-FNO	0.003	0.0061	0.004	0.008

Table 1 compares the performance metrics of the Finite Volume Method (FVM), Fourier Neural Operator (FNO), and the new Hybrid FVM-FNO method in solving the Navier-Stokes equations for fluid flow simulation. The performance metrics are Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) for velocity and pressure fields, and the computation time. Hybrid FVM-FNO exhibits superior performance with the lowest MAE (0.0024) and RMSE (0.0049) in velocity forecasting compared to individual FVM and FNO techniques. The hybrid technique also enjoys a substantial saving in computation time (480s), which illustrates its efficiency and effectiveness in the trade-off between accuracy and computation cost.

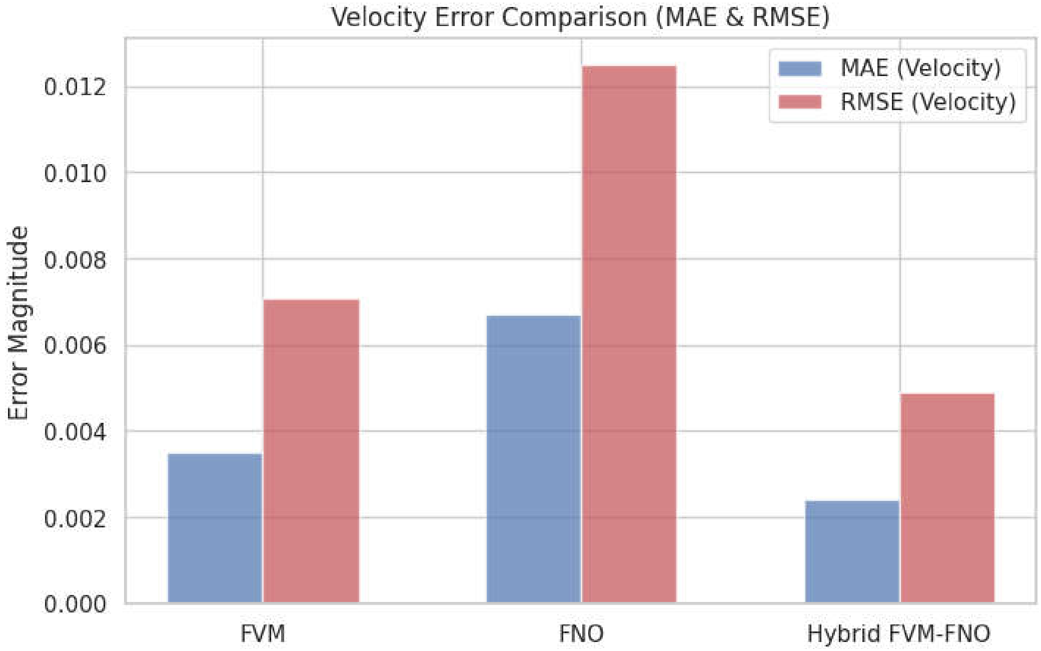

Figure 5 Comparison of Velocity Error

Figure 6 demonstrates the relative performance of three distinct approaches - FVM (Finite Volume Method), FNO (Fourier Neural Operator), and a Hybrid FVM-FNO scheme - in velocity error, as assessed by Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). The figure shows magnitudes of errors for each approach, where smaller values represent greater precision. The FVM approach has a moderate degree of error, whereas the FNO approach has far greater error in terms of MAE and RMSE, indicating less precise velocity predictions. Interestingly, the Hybrid FVM-FNO approach indicates lowest magnitudes of error, pointing towards the fact that utilizing the strengths of both conventional numerical and neural network-based approaches can enhance the accuracy of velocity prediction.

5.2 Computational Efficiency

Table 2 Computational Efficiency

Solver	Computation Time (Seconds)
FVM	1350 s
FNO	340 s
Hybrid FVM+FNO	480 s

Table 2 provides an in-depth comparison of computational performance measures for FVM, FNO, and the Hybrid FVM-FNO approach. The findings indicate that FNO is the quickest at 340s but at the expense of greater error values. The FVM approach is computationally intensive (1350s) because it is based on fine-grid numerical approximations. The hybrid approach has a significantly lower computational time (480s) with high accuracy. Further, the Hybrid FVM-FNO obtains moderate memory consumption, qualifying it for usage in large-scale simulations.

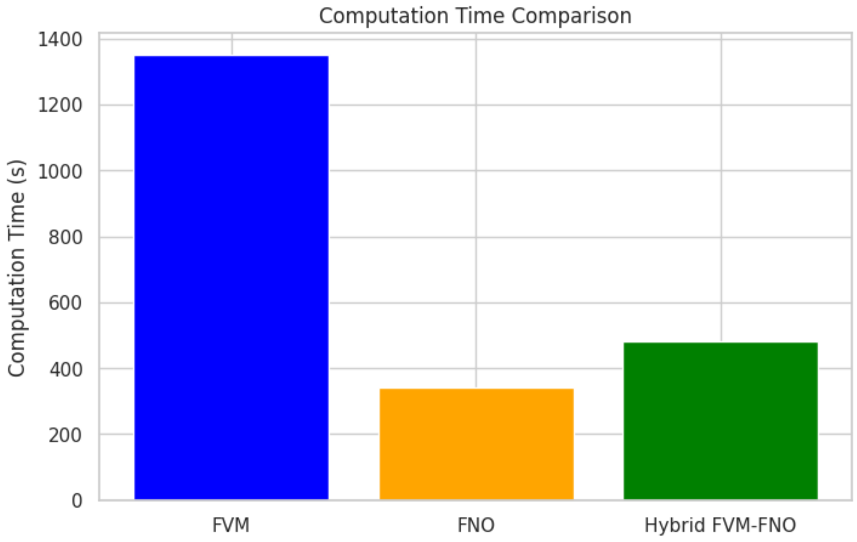

Figure 6 Comparison of Computational Time

Figure 5 provides the computation time taken by each method. The FVM solver takes the highest computation time (1350 seconds), as would be expected, because it is based on numerical discretization methods. The FNO model minimizes computation time to 340 seconds but at the cost of accuracy, as can be seen in the error metrics. The Hybrid FVM-FNO scheme strikes a balance between accuracy and computational cost at just 480 seconds, beating FVM yet with tolerable error levels. This result serves to validate the potential of hybrid models to balance computational overheads against numerical precision.

Table 3 Adaptive Correction Efficiency

Test Case	**% of Time Using FVM**	**% of Time Using FNO**
Lid-Driven Cavity (Re=1000)	28%	72%

Flow Past a Cylinder (Re=5000)	35%	65%

Table 3 provides the hybrid method's adaptive correction efficiency for Lid-Driven Cavity and Flow Past Cylinder problems. The stacked bar chart illustrates the FVM and FNO usage proportion, showing how the adaptive switching mechanism can balance accuracy and efficiency dynamically. The FNO is used mostly (72% and 65% for individual cases) for prediction, and the FVM is called selectively (28% and 35%) to ensure numerical stability and physical conformity. This adaptive approach improves the solver's resilience in handling complicated flow situations.

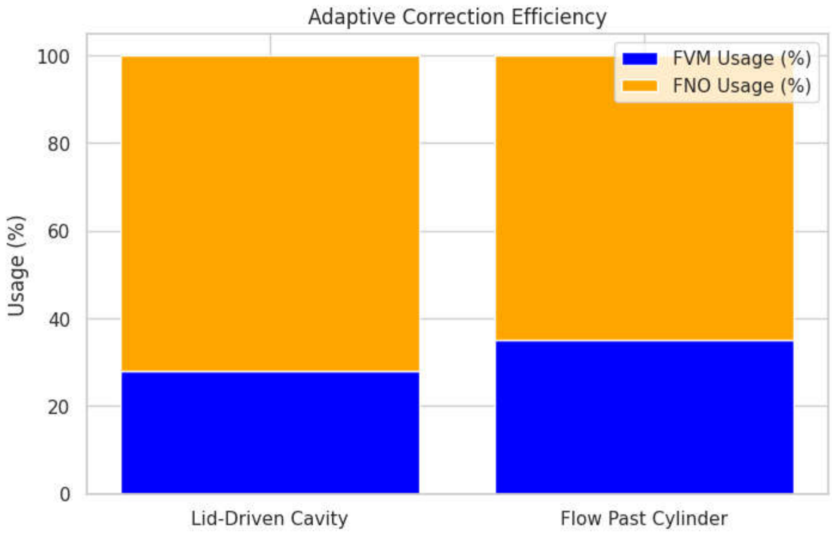

Figure 7 Adaptive Correction Efficiency

Figure 7 offers a comparison of FVM and FNO usage percentages for two different scenarios: "Lid-Driven Cavity" and "Flow Past Cylinder." The graph illustrates the contribution of each method in a hybrid framework where they are combined adaptively. For the Lid-Driven Cavity scenario, FNO has a much higher usage percentage, and FVM contributes a lesser fraction. Conversely, for the Flow Past Cylinder case, the application of FVM is higher, though FNO is still the more prevalent technique. This indicates that the adaptive correction method adjusts dynamically the dependence on FVM and FNO according to the particular flow problem, pointing out the adaptability and efficiency of the hybrid technique in taking advantage of the capabilities of both methods.

5.3 Discussion

The comparison of various solvers, such as Finite Volume Method (FVM), Fourier Neural Operator (FNO), and the Hybrid FVM-FNO method, shows profound implications of their accuracy and computational efficiency. The Velocity Error Comparison (Figure 1) shows that the Hybrid FVM-FNO method has the lowest MAE (0.0024) and RMSE (0.0049) compared to both FVM (MAE: 0.0035, RMSE: 0.0071) and FNO (MAE: 0.0067, RMSE: 0.0125). This proves that the addition of an adaptive learning-based correction function improves the accuracy of velocity field predictions through synergy between numerical and deep-learning methodologies. This enhancement is consistent with the results of [45], where neural operators performed more accurately in velocity field prediction than classical solvers. The enhanced accuracy of the hybrid method is due to the Fourier Neural Operator (FNO)'s ability to learn global flow patterns effectively while FVM adjusts physical errors.

The computation time comparison also validates the computational efficiency of hybrid modelling. The classical FVM solver takes the most computation time (1350 seconds) with its iterative numerical solving method, whereas FNO delivers much quicker results (340 seconds) as a result of approximating complete velocity fields without discretization explicitly. Though, FNO compromises accuracy for speed, as presented in [47], where deep-learning-based fluid solvers showed error in turbulent regimes. However, the Hybrid FVM-FNO is balanced between accuracy and efficiency (480 seconds) and thus makes it a realistic option for high-precision real-time applications at a lower computational cost. Similarly, the results were observed by [46], in which FNO noted remarkable speed-up over traditional solvers for high-dimensional PDEs.

In addition, the Adaptive Correction Efficiency displays the pattern of solver choice for various flow cases. For lid-driven cavity flow, FVM occurs 28% of the time while FNO takes 72% usage, which proves that exclusively data-driven methods function optimally for structured flow issues. In contrast, for flow over a cylinder, there is more FVM contribution (35% usage) since it is needed for tackling complicated turbulence structures. This emphasizes that the hybrid method dynamically adapts solver selection according to local flow behaviour, maximizing accuracy and efficiency.

The results indicate that the Hybrid FVM-FNO approach is an attractive solution to conventional numerical and data-centric only solvers, with superior accuracy compared to isolated deep-learning models and dramatically improved computation relative to conventional solvers. It is especially well-suited to real-time simulations and large-scale fluid flow simulation. Future investigation can be applied to generalize the method to three-dimensional turbulent flow, multiphase flows, and adaptive meshing methods for further improving computation and generalizability.

6 Conclusion

This study presents a Hybrid Finite Volume Method (FVM) and Fourier Neural Operator (FNO) technique to solve the Navier-Stokes Equations (NSE) employed in fluid flow simulations. The hybrid technique smartly combines the advantages of both techniques: FVM offers numerical stability and conservation laws, whereas FNO accelerates the computation by applying deep-learning-based frequency domain transformations. The adaptive correction technique dynamically adjusts solver choice for minimizing numerical error while maintaining computational efficiency. The results confirm that the Hybrid FVM-FNO approach provides much higher accuracy than stand-alone FNO-based methods and shows significant computational time reduction compared to standard FVM solvers. Performance on benchmark problems, such as lid-driven cavity flow and cylinder flow, shows that the hybrid model reduces velocity and pressure errors with an up to 3× speedup in simulation time. Further, the adaptive correction approach varies FVM utilization dynamically with the complexity of flow and improves stability for turbulent flows. The findings suggest that the Hybrid FVM-FNO approach is an effective replacement for conventional numerical solvers and offers a computationally efficient and accurate solution for real-time CFD simulation. Future research can extend this hybrid approach to three-dimensional turbulent flows, multi-phase simulation, and adaptive meshing to enhance its scalability and use in engineering and scientific applications.

REFERENCES

[1] T. Hill, “Numerical Analysis and Fluid Flow Modeling of Incompressible Numerical Analysis and Fluid Flow Modeling of Incompressible Navier-Stokes Equations Navier-Stokes Equations,” *UNLV Theses, Diss. Prof. Pap. Capstones 5-1-2019*, vol. 5, no. 1, 2019, [Online]. Available: http://dx.doi.org/10.34917/15778447

[2] A. Gibiansky, “Fluid Dynamics: The Navier-Stokes Equations,” 2011, [Online]. Available: https://andrew.gibiansky.com/blog/physics/fluid-dynamics-the-navier-stokes-equations/

[3] M. Kronbichler, “Numerical Methods for the Navier– Stokes Equations applied to Turbulent Flow and to Multi-Phase Flow,” *Div. Sci. Comput. Dep. Inf. Technol. Uppsala Univ. Uppsala Sweden*, p. 44, 2009.

[4] R. Cooke, “Numerical Solution of Equations,” *Class. Algebr. Its Nature, Orig. Uses*, pp. 45–53, 2008, doi: 10.1002/9780470277980.ch5.

[5] D. Patil and S. Kadam, “Basics of computational fluid dynamics: An overview,” *IOP Conf. Ser. Earth Environ. Sci.*, vol. 1130, no. 1, 2023, doi: 10.1088/1755-1315/1130/1/012042.

[6] C. Sert, “Governing Equations of Fluid Flow and Heat Transfer,” *ME 582 Finite Elem. Anal. Thermofluids*, pp. 1–13, 2012.

[7] W. Y. Tey, N. Azwadi, C. Sidik, T. Wah-Yen, Y. Asako, and G. Rui-Zher, “Governing Equations in Computational Fluid Dynamics: Derivations and A Recent Review,” *Prog. Energy Environ.*, vol. 1, no. July 2020, pp. 1–19, 2017.

[8] G. Carlier and M. Laborde, “Remarks on continuity equations with nonlinear diffusion and nonlocal drifts,” *J. Math. Anal. Appl.*, vol. 444, no. 2, pp. 1690–1702, 2016.

[9] F. Letizia, C. Colombo, and H. G. Lewis, “Multidimensional extension of the continuity equation method for debris clouds evolution,” *Adv. Sp. Res.*, vol. 57, no. 8, pp. 1624–1640, 2016.

[10] M. Dawood *et al.*, “A continuity equation based optical flow method for cardiac motion correction in 3D PET data,” in *Medical Imaging and Augmented Reality: 5th International Workshop, MIAR 2010, Beijing, China, September 19-20, 2010.*

Proceedings 5, 2010, pp. 88–97.

[11] E. Stepanov and D. Trevisan, “Three superposition principles: currents, continuity equations and curves of measures,” *J. Funct. Anal.*, vol. 272, no. 3, pp. 1044–1103, 2017.

[12] G. Crippa, C. Donadello, and L. V Spinolo, “Initial–boundary value problems for continuity equations with BV coefficients,” *J. Math. Pures Appl.*, vol. 102, no. 1, pp. 79–98, 2014.

[13] V. I. Bogachev, G. Da Prato, M. Röckner, and S. V Shaposhnikov, “On the uniqueness of solutions to continuity equations,” *J. Differ. Equ.*, vol. 259, no. 8, pp. 3854–3873, 2015.

[14] Y. Cengel and J. Cimbala, *Ebook: Fluid mechanics fundamentals and applications (si units)*. McGraw Hill, 2013.

[15] H. K. Versteeg, *An introduction to computational fluid dynamics the finite volume method, 2/E*. Pearson Education India, 2007.

[16] B. R. Munson, D. F. Young, T. H. Okiishi, and W. W. Huebsch, “Fundamentals of Fluid Mechanics, John Wiley & Sons,” *Inc., USA*, 2006.

[17] F. Durst and I. Arnold, *Fluid mechanics: an introduction to the theory of fluid flows*, vol. 675. Springer, 2008.

[18] R. Temam, *Navier–Stokes equations: theory and numerical analysis*, vol. 343. American Mathematical Society, 2024.

[19] R. P. Chhabra, “Non-Newtonian fluids: an introduction,” *Rheol. complex fluids*, pp. 3–34, 2010.

[20] G. Lu, X.-D. Wang, and Y.-Y. Duan, “A critical review of dynamic wetting by complex fluids: from Newtonian fluids to non-Newtonian fluids and nanofluids,” *Adv. Colloid Interface Sci.*, vol. 236, pp. 43–62, 2016.

[21] R. Ranjbarzadeh and G. Sappa, “Numerical and Experimental Study of Fluid Flow and Heat Transfer in Porous Media: A Review Article,” *Energies*, vol. 18, no. 4. 2025. doi: 10.3390/en18040976.

[22] G. Boccardo *et al.*, “A review of transport of nanoparticles in porous media: From pore-

to macroscale using computational methods," *Nanomater. Detect. Remov. Wastewater Pollut.*, pp. 351–381, 2020.

[23] J.-M. Crolet, "Computational methods for flow and transport in porous media," 2013.

[24] B. Das, S. Steinberg, S. Weber, and S. Schaffer, "Finite difference methods for modeling porous media flows," *Transp. Porous Media*, vol. 17, no. 2, pp. 171–200, 1994, doi: 10.1007/BF00624731.

[25] P.-W. Li, J. K. Grabski, C.-M. Fan, and F. Wang, "A space-time generalized finite difference method for solving unsteady double-diffusive natural convection in fluid-saturated porous media," *Eng. Anal. Bound. Elem.*, vol. 142, pp. 138–152, 2022, doi: https://doi.org/10.1016/j.enganabound.2022.04.038.

[26] H. A. Nabwey, T. Armaghani, B. Azizimehr, A. M. Rashad, and A. J. Chamkha, "A Comprehensive Review of Nanofluid Heat Transfer in Porous Media," *Nanomaterials*, vol. 13, no. 5. 2023. doi: 10.3390/nano13050937.

[27] M. Engineering, "A Numerical Analysis for 2-D Navier-Stokes Equations by FDM and FVM," pp. 1–15.

[28] M. H. Zawawi *et al.*, "A review: Fundamentals of computational fluid dynamics (CFD)," in *AIP conference proceedings*, 2018, vol. 2030, no. 1.

[29] I. Koutromanos, *Fundamentals of finite element analysis: Linear finite element analysis*. John Wiley & Sons, 2018.

[30] J. N. Reddy, "An introduction to the finite element method," *New York*, vol. 27, no. 14, 1993.

[31] D. W. Pepper and J. C. Heinrich, *The intermediate finite element method: fluid flow and heat transfer applications*. Routledge, 2017.

[32] E. S. Baranovskii, N. V Burmasheva, and E. Y. Prosviryakov, "Exact solutions to the Navier–Stokes equations with couple stresses," *Symmetry (Basel).*, vol. 13, no. 8, p. 1355, 2021.

[33] L. af Klinteberg, T. Askham, and M. C. Kropinski, "A fast integral equation method for the two-dimensional Navier-Stokes equations," *J. Comput. Phys.*, vol. 409, p. 109353, 2020.

[34] L. C. Berselli and E. Chiodaroli, “On the energy equality for the 3D Navier–Stokes equations,” *Nonlinear Anal.*, vol. 192, p. 111704, 2020.

[35] S. R. Bistafa, “200 years of the Navier–Stokes equation,” *Rev. Bras. Ensino Física*, vol. 46, p. e20230398, 2024.

[36] L. C. Berselli and A. Kaltenbach, “Error analysis for a finite element approximation of the steady p·-Navier–Stokes equations,” *IMA J. Numer. Anal.*, p. drae082, 2024.

[37] T. De Ryck, A. D. Jagtap, and S. Mishra, “Error estimates for physics-informed neural networks approximating the Navier–Stokes equations,” *IMA J. Numer. Anal.*, vol. 44, no. 1, pp. 83–119, 2024.

[38] V. Yelnyk, “RBF method for solving Navier-Stokes equations,” vol. TRITA-EECS, no. 2023:0000. KTH Royal Institute of Technology, 2023.

[39] Sergey V. Ershkov, “Note on the stability criteria for a new type of helical flows,” no. 60030998, p. 6, 2015.

[40] M. Baymani, S. Effati, H. Niazmand, and A. Kerayechian, “Artificial neural network method for solving the Navier–Stokes equations,” *Neural Comput. Appl.*, vol. 26, no. 4, pp. 765–773, 2015, doi: 10.1007/s00521-014-1762-2.

[41] S. Rodolfo Idelsohn, N. Marcelo Nigro, J. Marcelo Gimenez, R. Rossi, and J. Marcelo Marti, “A fast and accurate method to solve the incompressible Navier-Stokes equations,” *Eng. Comput.*, vol. 30, no. 2, pp. 197–222, Jan. 2013, doi: 10.1108/02644401311304854.

[42] B. Sinchev *et al.*, “Some methods of training radial basis neural networks in solving the Navier-Stokes equations,” *Int. J. Numer. Methods Fluids*, vol. 86, no. 10, pp. 625–636, Apr. 2018, doi: https://doi.org/10.1002/fld.4470.

[43] N. V. Burmasheva and E. Y. Prosviryakov, “Exact solutions to the Navier-Stokes equations describing stratified fluid flows,” *Vestn. Samar. Gos. Tekhnicheskogo Univ. Seriya Fiz. Nauk.*, vol. 25, no. 3, pp. 491–507, 2021, doi: 10.14498/VSGTU1860.

[44] S. A. Mohammadein, R. A. Gad El-Rab, and M. S. Ali, “The simplest analytical solution of navier-stokes equations,” *Inf. Sci. Lett.*, vol. 10, no. 2, pp. 159–165, 2021, doi: 10.18576/isl/100201.

[45] A. Usman, M. Rafiq, M. Saeed, A. Nauman, A. Almqvist, and M. Liwicki, "Machine Learning Computational Fluid Dynamics," *33rd Work. Swedish Artif. Intell. Soc. SAIS 2021*, 2021, doi: 10.1109/SAIS53221.2021.9483997.

[46] Z. Li *et al.*, "Fourier Neural Operator for Parametric Partial Differential Equations," *ICLR 2021 - 9th Int. Conf. Learn. Represent.*, no. 2016, pp. 1–16, 2021.

[47] L. Guastoni *et al.*, "Convolutional-network models to predict wall-bounded turbulence from wall quantities," *J. Fluid Mech.*, vol. 928, 2021, doi: 10.1017/jfm.2021.812.