A Numerical Solution of the Incompressible Navier-Stokes Equations

Shourya Mukherjee, Kunal Pathak, Dustin Maguire

November 28, 2023

1   Introduction

The study of the fluid dynamics of incompressible media has important applications in many biological fields, and there has been great interest in developing efficient ways to model them. In the realm of pharmaceutical research, incompressible fluid simulations offer a versatile platform for delving into the complex dynamics of drug dispersion within bodily fluids (Ferroni, De Gaetano, Cereda, & Boschetti, 2020), which provide insights into therapeutic outcomes and adverse effects. The development of drug carriers represents a significant stride in addressing challenges linked to the heterogeneous blood supply and irregular vascular structures within tumors, ultimately fostering improved drug distribution and therapeutic outcomes (Wikibooks, n.d.). Incompressible fluid simulation simulations are also used in the field of medicine, as they are used to design artificial organs and prosthetics, to develop cancer therapies (Mekheimer, Hasona, Abo-Elkhair, & Zaher, 2018), and to understand cardiovascular functions in the human body (Santhanakrishnan & Miller, 2011). In the domain of cardiovascular medicine, researchers leverage incompressible fluid simulations to scrutinize the effects of devices such as stents on blood flow patterns.

The motion of incompressible, viscous fluids is governed by the incompressible Navier-Stokes equations, which are a set of partial differential conservation laws. A common technique used to solve the incompressible Navier-Stokes equations is with the use of finite differences. Chorin first proposed a finite difference scheme to solve the equations by calculating auxillary velocity fields, and using a modified Dufort-Frankel over-relaxation scheme to calculate pressure and velocity in the next timestep (Chorin, 1968). He also demonstrated the power of finite difference methods by solving the Bernard convection problem in two dimensions. Since then, others have employed finite difference methods to solve the incompressible Navier-Stokes equation and offered improvements to Chorin's method.

Most notably, Peskin expanded on Chorin's method to model blood flow inside heart valves (Peskin, 1972), while also modeling fluid-structure interactions with an elastic boundary. Specifically, he used a projection method that reformulated the Navier-Stokes equations using finite differences and solved a tridiagonal linear system at each timestep. Like Chorin, Peskin also used a relaxation scheme to calculate pressure, but Peskin also employed an alternative method (Richtmyer, 1957) to solve the tridiagonal systems. He reported that the stability of the method was dependent on the Reynolds number. For more examples of papers that used finite difference methods to solve incompressible fluid problems for biological problems, see references in the paper by Nonino et al. (Nonino, Ballarin, Rozza, & Maday, 2022). Though these finite difference methods were already capable of solving computationally challenging problems in fluid dynamics, the main limitation of these methods was that the computational efficiency was limited by the speed at which the linear systems are solved. Thus, modern numerical solutions of the incompressible Navier-Stokes equations employ the Fast Fourier Transform (FFT) to increase efficiency.

However, regardless of the implementation of FFT, the Direct Numerical Solution, DNS, of the Navier-Stokes equations proved to be inefficient and impractical when attempting to model larger systems. Thus, other computational methods including Reynolds-Averaged Navier-Stokes, or RANS, and Large Eddy Simulations, or LES, were established to reduce computation time at the expense of the accuracy of the model. Regardless, these approximations are still widely used in many current applications in both academia and industry (Smits, 2009).

2 Theoretical Background

2.1 The Incompressible Navier-Stokes Equations

The Navier-Stokes equation is derived from momentum and mass conservation laws. The momentum equation may be simplified through the assumption of a Newtonian fluid, in which there is a linear relationship between the stress tensor τ_{ij} and the strain tensor ϵ_{ij} with the constant of proportionality being the dynamic viscosity μ. Under this assumption, τ_{ij} can be related to the components of the velocity and the pressure by

$$\tau_{ij} = -p\delta_{ij} + \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) + \delta_{ij}\lambda\nabla\cdot\mathbf{u}, \tag{1}$$

where

$$\delta_{ij} = \begin{cases} 0, i \neq j \\ 1, i = j \end{cases},$$

and

$$p = -\frac{1}{3}(\tau_{11} + \tau_{22}),$$

is the pressure, and μ is the dynamic viscosity, which is constant for Newtonian fluids (RIVAL, 2023). Thus, if we also assume the fluid is isotropic and disregard any heat flow and changes in energy, the momentum and mass conservation equations may be written as

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} = \mathbf{g} - \frac{1}{\rho}\nabla p + \nu\nabla^2\mathbf{u} \\ \nabla\cdot\mathbf{u} = 0 \end{aligned}, \tag{2}$$

where $\mathbf{u}$ is the velocity, p is the pressure, $\mathbf{g}$ is the external force density, and $\nu = \frac{\mu}{\rho}$ is the kinematic viscosity. These are the incompressible Navier-Stokes equations.

2.2 Finite Difference Operators

The fluid domain is discretized as a Cartesian grid, and the boundary is discretized as a curvilinear mesh. We use an adaptive mesh refinement scheme to increase the resolution of the mesh at regions of higher vorticity and regions near the boundary nodes.

In order to discretize the Navier-Stokes equations, we must discretize differential operators using the finite difference method. Let $\mathbf{e}_1$ and $\mathbf{e}_2$ be the basis vectors of the Cartesian grid, and let h be the grid size. Let us define the finite difference gradient operator as

$$\mathbf{D}^0 = (D_1^0, D_2^0), \text{ where} \tag{3}$$

$$(D_\alpha^0)\phi(\mathbf{x}) = \frac{\phi(\mathbf{x} + h\mathbf{e}_\alpha) - \phi(\mathbf{x} - h\mathbf{e}_\alpha)}{2h} \tag{4}$$

Let us define the forward and backward differential operators as

$$(D_\alpha^+)\phi(\mathbf{x}) = \frac{\phi(\mathbf{x} + h\mathbf{e}_\alpha) - \phi(\mathbf{x})}{h} \tag{5}$$

$$(D_\alpha^-)\phi(\mathbf{x}) = \frac{\phi(\mathbf{x}) - \phi(\mathbf{x} - h\mathbf{e}_\alpha)}{h} \tag{6}$$

Let us define the Laplacian operator as

$$(L\mathbf{u})(\mathbf{x}) = \sum_{\alpha=1}^{2} \frac{\mathbf{u}(\mathbf{x} + h\mathbf{e}_\alpha) + \mathbf{u}(\mathbf{x} - h\mathbf{e}_\alpha) - 2\mathbf{u}(\mathbf{x})}{h^2} \tag{7}$$

We can opt to handle the nonlinear (convective) term using a central difference approximation as

$$S(\mathbf{u})\phi = \frac{1}{2}\mathbf{u}\cdot\mathbf{D}^0\phi + \frac{1}{2}\mathbf{D}^0\cdot(\mathbf{u}\phi), \tag{8}$$

or we can use a biased difference approximation.

$$S(\mathbf{u})\phi = \mathbf{u} \cdot \mathbf{D}^{\pm}\phi, \tag{9}$$

where

$$u_\alpha D_\alpha^\pm = \begin{cases} u_\alpha D_\alpha^+, & u_\alpha < 0 \\ u_\alpha D_\alpha^-, & u_\alpha \geq 0 \end{cases}. \tag{10}$$

2.3   Discretized Navier-Stokes Equations

Using these differential operators, we can discretize the Navier-Stokes equations, denoting the velocity and pressure at the timestep t as $\mathbf{u}^t(\mathbf{x})$ and $p^t(\mathbf{x})$ respectively. We can obtain an implicit discretized Navier-Stokes equation for a time-step of Δt to be

$$\rho\left(\frac{\mathbf{u}^{t+\Delta t} - \mathbf{u}^t}{\Delta t} + S(\mathbf{u}^t)\mathbf{u}^t\right) = -\mathbf{D}^0 p^{t+\Delta t} + \mu L\mathbf{u}^{t+\Delta t} + \mathbf{f}^t \tag{11}$$

$$\mathbf{D}^0 \cdot \mathbf{u}^{t+\Delta t} = 0. \tag{12}$$

Equation 11 can be rearranged to yield

$$\left(1 - \frac{\mu\Delta t}{\rho}L\right)\mathbf{u}^{t+\Delta t} + \frac{\Delta t}{\rho}\mathbf{D}^0 p^{t+\Delta t} = \mathbf{u}^t - S(\mathbf{u}^t)\mathbf{u}^t\Delta t + \frac{\Delta t}{\rho}\mathbf{f}^t. \tag{13}$$

When we make a change of variables to $\mathbf{u} = \mathbf{u}^{t+\Delta t}$, $q = p^{t+\Delta t}$, and $\mathbf{w} = \mathbf{u}^t - S(\mathbf{u}^t)\mathbf{u}^t\Delta t + \frac{\Delta t}{\rho}\mathbf{f}^t$, then the discretized Navier-Stokes equations become

$$\left(1 - \frac{\mu\Delta t}{\rho}L\right)\mathbf{u} + \frac{\Delta t}{\rho}\mathbf{D}^0 q = \mathbf{w}. \tag{14}$$

$$\mathbf{D}^0 \cdot \mathbf{u}^{t+\Delta t} \tag{15}$$

2.4   Half-Stepping

In order to incorporate half-stepping, we can discretize the Navier-Stokes equation as follows:

$$\rho\left(2\frac{\mathbf{u}^{t+\Delta t/2} - \mathbf{u}^t}{\Delta t} + S(\mathbf{u}^t)\mathbf{u}^t\right) = -\mathbf{D}^0 \tilde{p}^{t+\Delta t/2} + \mu L\mathbf{u}^{t+\Delta t/2} + \mathbf{f}^{t+\Delta t/2} \tag{16}$$

$$\mathbf{D}^0 \cdot \mathbf{u}^{t+\Delta t/2} = 0 \tag{17}$$

Equation 16 can be rearranged to yield

$$\left(1 - \frac{\mu\Delta t}{2\rho}L\right)\mathbf{u}^{t+\Delta t/2} + \frac{\Delta t}{2\rho}\mathbf{D}^0 \tilde{p}^{t+\Delta t/2} = \mathbf{u}^t - \frac{\Delta t}{2}S(\mathbf{u}^t)\mathbf{u}^t + \frac{\Delta t}{2\rho}\mathbf{f}^{t+\Delta t/2}, \tag{18}$$

where $\tilde{p}$ is the preliminary pressure. Once these equations are solved for $\mathbf{u}^{t+\Delta t/2}$ and $\tilde{p}^{t+\Delta t/2}$, we can use these to solve the full step equations, which uses the same finite difference formulations of the differential operators, but also averages the velocities at times $t + \Delta t$ and t to calculate the Laplacian.

$$\rho\left(\frac{\mathbf{u}^{t+\Delta t} - \mathbf{u}^t}{\Delta t} + S(\mathbf{u}^{t+\Delta t/2})\mathbf{u}^{t+\Delta t/2}\right) = -\mathbf{D}^0 p^{t+\Delta t} + \mu L\left(\frac{\mathbf{u}^{t+\Delta t} + \mathbf{u}^t}{2}\right) + \mathbf{f}^{t+\Delta t/2} \tag{19}$$

$$\mathbf{D}^0 \cdot \mathbf{u}^{t+\Delta t} = 0 \tag{20}$$

Equation 19 can be rearranged to yield

$$\left(1 - \frac{\mu\Delta t}{2\rho}L\right)\mathbf{u}^{t+\Delta t} + \frac{\Delta t}{\rho}\mathbf{D}^0 p^{t+\Delta t} = \mathbf{u}^t - \Delta t S(\mathbf{u}^{t+\Delta t/2})\mathbf{u}^{t+\Delta t/2} + \frac{\Delta t\mu}{2\rho}L\mathbf{u}^t + \frac{\Delta t}{\rho}\mathbf{f}^{t+\Delta t/2}, \tag{21}$$

Now, for Equation 18, when we make the substitutions $\mathbf{u} = \mathbf{u}^{t+\Delta t/2}$, $q = \frac{1}{2}\tilde{p}^{t+\Delta t/2}$, and $\mathbf{w} = \mathbf{u}^t - \frac{\Delta t}{2}S(\mathbf{u}^t)(\mathbf{u}^t) + \frac{\Delta t}{2\rho}\mathbf{f}^{t+\Delta t/2}$, we can obtain equations of the form,

3

$$\left(1 - \frac{\Delta t \mu}{2\rho}\hat{L}\right)\mathbf{u} + \frac{\Delta t}{\rho}\mathbf{D}^0 q = \mathbf{w} \tag{22}$$

$$\mathbf{D}^0 \cdot \mathbf{u} = 0 \tag{23}$$

This same equation can be obtained with Equation 21 when we make the substitutions $\mathbf{u} = \mathbf{u}^{t+\Delta t}$, $q = p^{t+\Delta t/2}$, and $\mathbf{w} = \mathbf{u}^t - \Delta t S(\mathbf{u}^{t+\Delta t/2})\mathbf{u}^{t+\Delta t/2} + \frac{\Delta t}{\rho}\mathbf{f}^{t+\Delta t/2} + \frac{\Delta t \mu}{2\rho}L\mathbf{u}^t$.

2.5   Fourier Transforms

To solve the Navier-Stokes equations at the half step and the full step, we can use a 2D discrete Fourier transform to obtain an explicit equation for both $\mathbf{u}$ and p at the half-step $t + \Delta t/2$ and the full-step $t + \Delta t$. The Fourier transform of D_α^0 is

$$\hat{D}_\alpha^0(m_1, m_2) = \frac{i}{h}\sin\left(\frac{2\pi}{N}m_\alpha h\right), \tag{24}$$

and the Fourier transform of L is

$$\hat{L} = -\frac{4}{h^2}\sum_{\alpha=1}^{2}\sin^2\left(\frac{\pi}{N}m_\alpha h\right) \tag{25}$$

Using the Fourier transformed operators, we note that the resulting Fourier transformed Navier-Stokes equations are not coupled, allowing us to explicitly calculate $\mathbf{u}$. The Fourier transformed equation is

$$\left(1 - \frac{\Delta t \mu}{2\rho}\hat{L}\right)\hat{\mathbf{u}} + \frac{\Delta t}{\rho}\hat{\mathbf{D}}^0 \hat{q} = \hat{\mathbf{w}} \tag{26}$$

$$\hat{\mathbf{D}}^0 \cdot \hat{\mathbf{u}} = 0 \tag{27}$$

When we apply the $\hat{\mathbf{D}}^0$ operator on this equation and the incompressibility condition, we obtain explicit equations for $\hat{q}$, which can then be used to solve for $\hat{\mathbf{u}}$.

$$\hat{q} = \frac{\hat{\mathbf{D}}^0 \cdot \hat{\mathbf{w}}}{\frac{\Delta t}{\rho}\hat{\mathbf{D}}^0 \cdot \hat{\mathbf{D}}^0} \tag{28}$$

Using this calculated pressure, we find $\hat{\mathbf{u}}$ as

$$\hat{\mathbf{u}} = \frac{\hat{\mathbf{w}} - \frac{\Delta t}{\rho}\hat{\mathbf{D}}^0 \hat{q}}{1 - \frac{\Delta t \mu}{2\rho}\hat{L}} \tag{29}$$

This update rule applies to both the half-step and the full-step once the proper substitutions are made for $\hat{\mathbf{u}}$, $\hat{q}$, and $\hat{\mathbf{w}}$.

2.6   A Brief Background on Approximating Fluid Dynamics

DNS will theoretically have high accuracy over the same time interval. However, this is unreasonable due to the exceedingly expensive time complexity in directly calculating velocity and pressure at each time step. Thus, the RANS and LES equations aim to simplify this. The Navier-Stokes Equations can written in dimensionless form as

$$u_t + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \frac{1}{Re}\nabla^2 \mathbf{u} \tag{30}$$

where u_t is the derivative of u with respect to time, p is the pressure, and Re is the Reynolds number. Also, by decomposing the Navier-Stokes components into two parts we get the following:

$$u(x, t) = \bar{U}(x) + u'(x, t). \tag{31}$$

Note that $u(x, t)$ can also be expressed as a vector with takes the form

$$u(x,t) = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \cdot \begin{bmatrix} \bar{U} + u' \\ \bar{U} + v' \\ \bar{U} + w' \end{bmatrix}. \tag{32}$$

$\bar{U}(x)$ is the average fluid velocity at a certain point, which is independent of time. $u'(x, t)$ is therefore the fluctuating component of $u(x, t)$. A more rigorous definition of $\bar{U}$ is as follows:

$$\bar{U} = \lim_{T \to \infty} \int_{t_0}^{t_0+T} u(x,t)\, dt. \tag{33}$$

In the following derivation, we must rely on certain rules of averaging, which are as follows:

$$\bar{u'} = 0$$

$$\overline{u + v} = \bar{u} + \bar{v}$$

$$\bar{\bar{u}} = \bar{u}$$

$$\overline{\frac{\partial u}{\partial x}} = \frac{\partial \bar{u}}{\partial x}$$

(Smits, 2009).

2.6.1 Mass Continuity

Combining the vector dot product for $\mathbf{u}(\mathbf{x}, t)$ and Equation 2 we get

$$\frac{\partial \bar{U}}{\partial x} + \frac{\partial \bar{V}}{\partial y} + \frac{\partial \bar{W}}{\partial z} + \frac{\partial u'}{\partial x} + \frac{\partial v'}{\partial y} + \frac{\partial w'}{\partial z} = 0. \tag{34}$$

Taking the time average of the equation and utilizing the identities previously stated offers

$$\frac{\partial \bar{U}}{\partial x} + \frac{\partial \bar{V}}{\partial y} + \frac{\partial \bar{W}}{\partial z} = 0, \tag{35}$$

since the time average of the sum of the final three terms equal 0. Now that the first three terms must add to 0, we have

$$\frac{\partial u'}{\partial x} + \frac{\partial v'}{\partial y} + \frac{\partial w'}{\partial z} = 0. \tag{36}$$

Thus, the divergence of both the mean field and the fluctuating fields are 0 (Smits, 2009).

2.6.2 Momentum Equation Decomposition

For incompressible flows, and no body forces, start with the continuity and Navier-Stokes Equations with respect to the x-component of the momentum for brevity:

$$u_t + (u\frac{d}{dx} + v\frac{d}{dy} + w\frac{d}{dz})\mathbf{u} = -\frac{dp}{dx} + \frac{1}{Re}\nabla^2\mathbf{u} \tag{37}$$

where u_t denotes the first component of $\mathbf{u}$. Now we can perform a time average to the entire equation, utilizing the identities stated above to simplify the equation. First, we must expand by

$$\bar{U}_t + u'_t + \bar{U}\bar{U}_x + \bar{U}u'_x + u'\bar{U}_x + u'u'_x + \bar{V}\bar{U}_y + \bar{V}u'_y + v'\bar{U}_y + v'u'_y + \bar{W}\bar{U}_z + \bar{W}u'_z + w'\bar{U}_z + w'u'_z =$$
$$-\bar{p}_x - p'_x + \frac{1}{Re}\nabla^2\bar{U} + \frac{1}{Re}\nabla^2 u'. \tag{38}$$

Taking the time average of the equation and removing 0 terms gives the following:

$$\bar{U}\bar{U}_x + \bar{V}\bar{U}_y + \bar{W}\bar{U}_z + u'u'_x + v'u'_y + w'u'_z = -\bar{p}_x + \frac{1}{Re}\nabla^2\bar{U}. \tag{39}$$
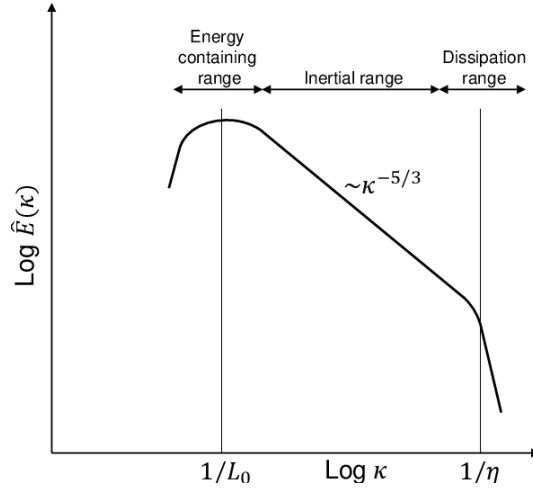
Figure 1: The Kolmogorov spectrum from (Nagy, 2018), a simplified version of the kinetic energy stored within the various eddy sizes across the entire domain with larger wave numbers signifying a smaller eddy.

While the fourth through sixth terms may seem to be 0, it is important to remember that these quantities signify the variance of the turbulence over time, which is not necessarily 0 and are the points in which turbulence is seen through this model. To further simplify the above equation we can evaluate using the chain rule:

$$\overline{u'u'_x} = \frac{\partial \overline{u'u'}}{\partial x} - \overline{u'\frac{\partial u'}{\partial x}} \tag{40}$$

$$\overline{v'u'_y} = \frac{\partial \overline{v'u'}}{\partial y} - \overline{u'\frac{\partial v'}{\partial y}} \tag{41}$$

$$\overline{w'u'_z} = \frac{\partial \overline{w'u'}}{\partial z} - \overline{w'\frac{\partial w'}{\partial z}}. \tag{42}$$

However, we know that

$$\overline{u'\frac{\partial u'}{\partial x}} + \overline{u'\frac{\partial v'}{\partial y}} + \overline{w'\frac{\partial w'}{\partial z}} = (u' \cdot \nabla)\mathbf{u} = u' \cdot 0 = 0. \tag{43}$$

Thus, our resulting equation for the x-component simplifies to

$$\bar{U}\bar{U}_x + \bar{V}\bar{U}_y + \bar{W}\bar{U}_z + \frac{\partial \overline{u'u'}}{\partial x} + \frac{\partial \overline{v'u'}}{\partial y} + \frac{\partial \overline{w'u'}}{\partial z} = -\bar{p}_x + \frac{1}{Re}\nabla^2\bar{U}. \tag{44}$$

The first three terms are mean value terms and are relatively easy to calculate. The issues arise when determining fourth through sixth terms, which are known as Reynolds Stresses. This occurs when taking their gradient with respect to space. The goal of turbulence modeling using RANS is to try to best approximate these quantities. RANS completes this step the fastest out of the other techniques by taking the integral of the turbulent energy cascade which is modeled in the figure below:

As a result of integrating this curve, we can set the effects of the smaller turbulent terms described in equation 44 to the constant area under the curve. This removes nearly all of the small eddy formation and its distribution across the fluid at the end, but it preprocesses the most expensive part of the calculation, therefore enabling it to converge to a fast solution.(Smits, 2009)

2.6.3 Large Eddy Simulations

RANS proves to be quite fast, however, its disregard for small scale turbulence makes it exceedingly inaccurate in many cases. LES aims to remediate this by applying the fluid onto a mesh of varying sizes which encapsulate the large eddies as well as some of the smaller ones.

The goal of the mesh is to divide the fluid into equal cells. If there is an eddy, it must be expressed as a circular vector field distributed across at least 4 adjacent cells. Thus, if there is an eddy that exists within

that threshold, then the mesh must be further subdivided such that the eddies that are too small are resolved. If the eddy is larger than 4 cells, then the mesh is capable of resolving that eddy, however, it comes at the cost of additional computational power. Therefore, the most important key is to maximize both the simulation accuracy and the grain of the mesh.

Referring back to Figure 1, a good LES model attempts to resolve at least 80% of the turbulent kinetic energy within the model. The other 20% must be included simply utilizing the same interpolation as seen in the RANS model.

By taking the integral length scale, or the length of an eddy at the average kinetic energy of all eddies within a given region in the domain, we can determine how to subdivide the grid. This is more formally denoted as

$$l_0 = \frac{\int_0^\infty k^{-1}E(k)d(k)}{\int_0^\infty E(k)d(k)} \tag{45}$$

where k is the wave number, $E(k)$ is the energy at that wave number, and $d(k)$ is the diameter of the eddy at that wave number. At regions with smaller l_0, the mesh should be smaller to account for the smaller average eddy size. However, if we keep a small mesh for larger l_0 values, the computation quickly increases, so it is important to move to a coarser grain when eddies are large.

Finally, to ensure that the cells encapsulate 80% of the turbulent kinetic energy, a mesh of 5 cells across the integral length scales typically resolves a sufficiently high turbulent kinetic energy and is usually a safe estimate. (Smits, 2009)

3 Methods

In this paper, we describe the results from three different types of fluid solvers for solving the incompressible Navier-Stokes equations. For all these methods, we use a L x L grid with a grid size of dx. The total number of grid points is N^2, where $N = L/dx$. We will denote the timestep size as dt.

3.1 Full Step Upwind FFT Scheme

For the first type of solver, we use a full step explicit Fast Fourier Transform (FFT) scheme (Peskin, 1997). The scheme goes as follows:

1. Initialize the N x N grid with values for $\mathbf{u} = (u, v)$, and p at t = 0.

2. Initialize the profile of the force over spatial coordinates and in time.

3. For each timestep,

    (a) Calculate $S(\mathbf{u}^t)\mathbf{u}^t = (\mathbf{u}^t \cdot \mathbf{D}^\pm)\mathbf{u}^t$ using the upwind discretization in Equation 10.
    (b) Calculate $\mathbf{w} = \mathbf{u}^t - S(\mathbf{u}^t)\mathbf{u}^t \Delta t + \frac{\Delta t}{\rho}\mathbf{f}^t$.
    (c) Take the discrete Fourier transform of $\mathbf{w}$ to obtain $\hat{\mathbf{w}}$.
    (d) Use $\hat{\mathbf{w}}$ to calculate $\hat{u}$, $\hat{v}$, and $\hat{q}$ at time $t + \Delta t$ using Equations 28 and 29.
    (e) Take the inverse Fourier transform of $\hat{u}$, $\hat{v}$

3.2 Half Step FFT Scheme without Upwind Discretization

We also used a method that incorporates half-stepping but also uses a central finite difference to calculate the nonlinear term $S(\mathbf{u}^t)\mathbf{u}^t$.

1. Initialize the N x N grid with values for $\mathbf{u} = (u, v)$, and p at t = 0.

2. Initialize the profile of the force over spatial coordinates and in time.

3. For each timestep,

    (a) For the half step, first calculate $S(\mathbf{u}^t)\mathbf{u}^t$ using Equation 8.
    (b) Calculate $\mathbf{w} = \mathbf{u}^t - \frac{\Delta t}{2}S(\mathbf{u}^t)(\mathbf{u}^t) + \frac{\Delta t}{2\rho}\mathbf{f}^{t+\Delta t/2}$.
    (c) Take the discrete Fourier transform of $\mathbf{w}$ to obtain $\hat{\mathbf{w}}$.

(d) Use $\hat{\mathbf{w}}$ to calculate $\hat{u}$, $\hat{v}$, and $\hat{q}$ at time $t + \Delta t$ using Equations 28 and 29.

(e) Take the inverse Fourier transform of $\hat{u}$, $\hat{v}$ to obtain $u^{t+\Delta t/2}$ and $v^{t+\Delta t/2}$.

(f) For the full step, first calculate $S(\mathbf{u}^{t+\Delta t})\mathbf{u}^{t+\Delta t}$ using Equation 8.

(g) Calculate $\mathbf{w} = \mathbf{u}^t - \Delta t S(\mathbf{u}^{t+\Delta t/2})\mathbf{u}^{t+\Delta t/2} + \frac{\Delta t}{\rho}\mathbf{f}^{t+\Delta t/2} + \frac{\Delta t \mu}{2\rho}L\mathbf{u}^t$

(h) Use $\hat{\mathbf{w}}$ to calculate $\hat{u}$, $\hat{v}$, and $\hat{q}$ at time $t + \Delta t$ using Equations 28 and 29.

(i) Take the inverse Fourier transform of $\hat{u}$, $\hat{v}$ to obtain $u^{t+\Delta t}$ and $v^{t+\Delta t}$.

3.3 Half Step FFT Scheme with Upwind Discretization

This scheme is identical to the Half Step FFT scheme described in the previous section, except for the calculation of $S(\mathbf{u})\mathbf{u}$ for which we use the upwind discretization in Equation 10. This is essentially the algorithm detailed in Section 3.1 but with half-stepping.

3.4 Force Profiles

We use 3 main types of force profiles in this paper: a constant force to quantify the propagation of roundoff error, a 3D Gaussian force distribution, and a 2D regularized Dirac delta. The constant force distribution is given by

$$\mathbf{f}_{i,j} = (1,0). \tag{46}$$

The Gaussian force distribution for the explosion is given by

$$\mathbf{f}_{i,j} = \left(A\exp\left[-(i-i_c)^2 - (j-j_c)^2 - (t-t_c)^2\right) \right], B\exp\left[-(i-i_c)^2 - (j-j_c)^2 - (t-t_c)^2\right) \right] \right) \tag{47}$$

For the regularized Dirac delta, we investigate the prescribed function recommended by Peskin for fluid structure interaction problems (Peskin, 2002).

$$\mathbf{f}_{i,j} = \frac{1}{h^2}\phi(i-i_c)\phi(j-j_c), \tag{48}$$

where

$$\phi(r) = \begin{cases} \frac{1}{4}(1+\cos\frac{\pi r}{2}), & |r| \leq 2 \\ 0, & \text{otherwise} \end{cases}. \tag{49}$$

3.5 Large Eddy Simulation Comparison

We also discuss ways to reduce computation time when modeling the Navier-Stokes equations through both RANS and LES. We outline a direct comparison taken between the DNS with an LES at scale. This involved first writing Python code that can run a DNS taking in the parameters shown below.

dt	Niter	N
0.2	50	400
0.1	100	400
0.05	200	400
0.025	400	400

We also used openFOAM, an open source computational fluid dynamics software used in many engineering and scientific capacities. Once downloading the package, we configured the periodic plane channel LES, listed under the pimpleFoam directory, and utilized the default values in the Smagorinsky model. Next, we ran the simulation across various time steps on the LES and DNS keeping the computer system and background processes constant.

4 Results

4.1 Constant Force

The full step (FS) solver has a much larger stability range. When we ran a timestep size $\Delta t = 0.2$, the full step solver ran for 350 iterations, while both the half step (HS) and the half step upwind (HSU) solvers failed to get past 8 iterations before encountering overflow errors. However, when we ran a timestep size $\Delta t = 0.02$, we observed notable differences between the different types of solvers. While the FS solver had a gradually decreasing continuous noise with increasing number of timesteps in v velocity, both the HS solvers seemed to have periodic noise in the v velocity. The solver with the central difference approximations (HS) did not see an overall gradually decreasing trend, while the other two did 2.
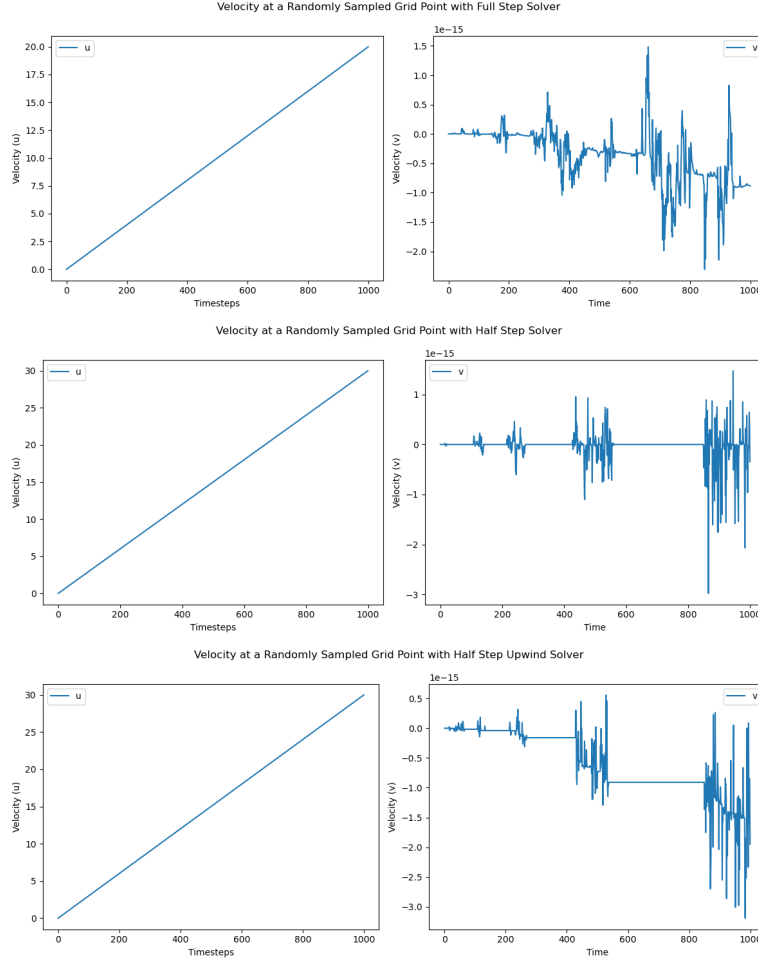

Figure 2: A collection of plots of x and y velocity (u and v respectively) as a function of the number of timesteps for the FS, HS, and HSU respectively.

4.2 Explosion Results (Gaussian and Dirac Delta)

For the Gaussian explosion, the HS and HSU solvers obtain similar solutions, while the full step solver fails to achieve a meaningful result. As shown in Figure 3, the explosion was initiated at timestep 2, and the result was sampled at a randomly chosen grid point. The velocity is initially zero until the explosion arrives at the point, and after some time, the velocity flattens out to reach a steady state. For the Dirac Delta explosion, as shown in Figure 4, the shape is clearly very different from the Gaussian explosion. Though, the FS solver once again produces an inconsistent solution compared to the HS solver.

The stability of the solution was highly dependent on the grid size dx and the timestep size dt. For the parameters, L = 0.5, dx = 0.1, and dt = 0.01 for instance, none of the three solvers were able to generate a stable solution for either force distribution.

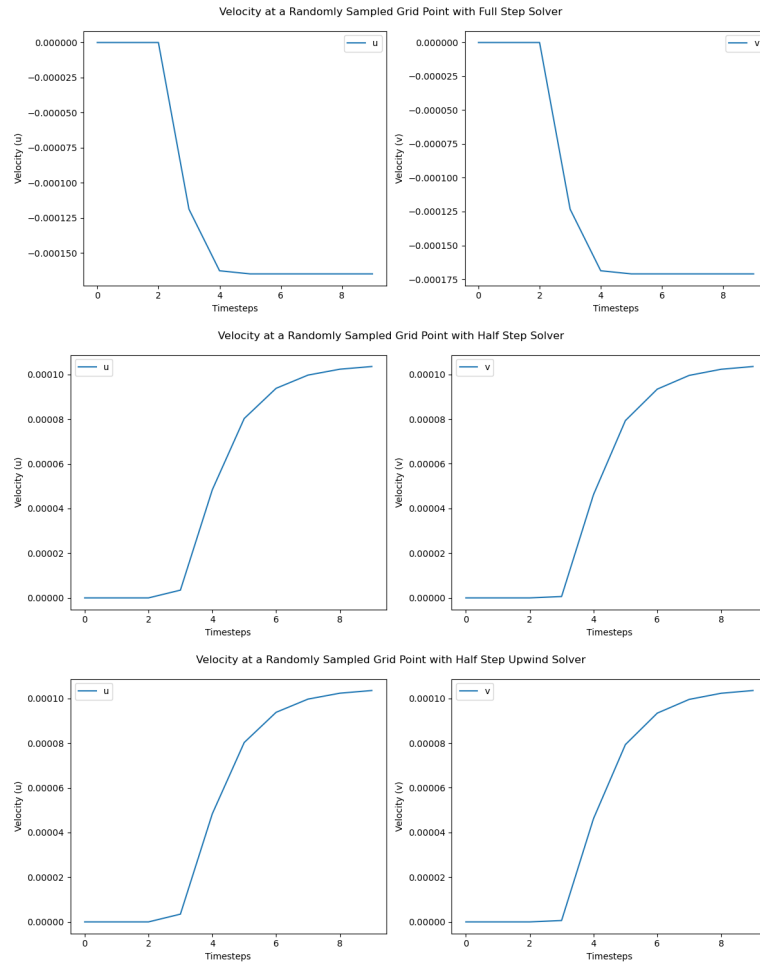Figure 3: Results for the full step solver and both half-step solvers on the explosion problem with L = 50, dx = 0.2, and dt = 0.01. The explosion occurs at timestep 2
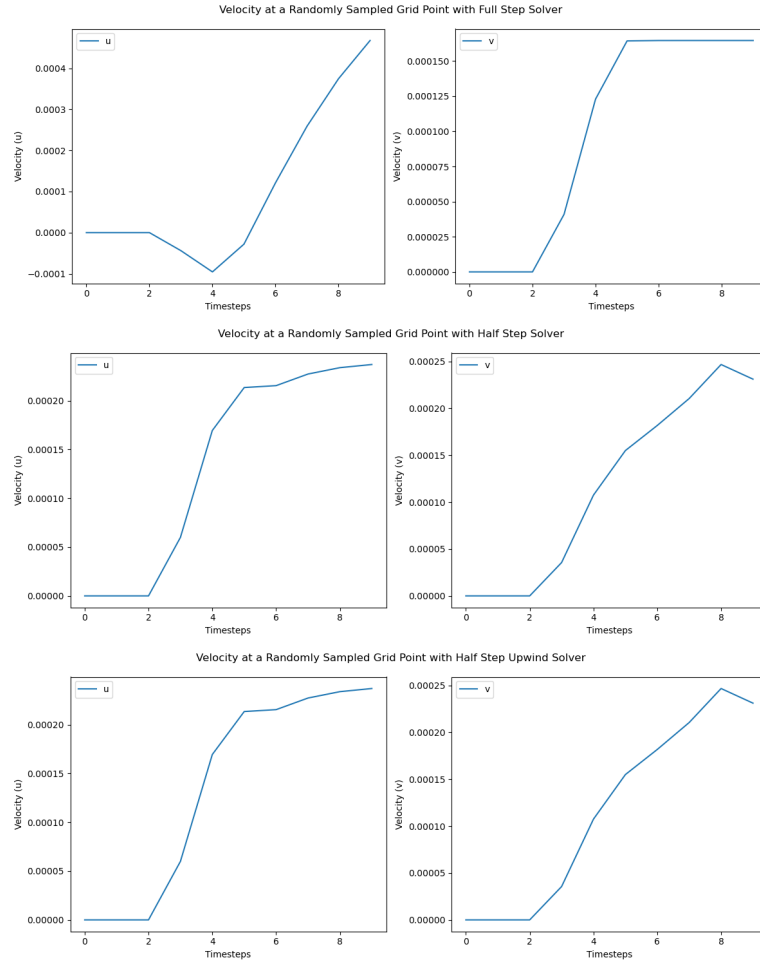
Figure 4: Results for the full step solver and both half-step solvers on the explosion problem with L = 50, dx = 0.2, and dt = 0.01. The explosion occurs at timestep 2

4.3 DNS vs LES Results

Here is our table for the results of running the DNS and LES on varying dt values:

Table 2: Time Spent to Compute Simulation DNS

dt (s)	Number of Iterations	Support Points	DNS result (s)
0.2	50	400	44.9
0.1	100	400	92.3
0.05	200	400	182.2
0.025	400	400	373.9

Both the DNS and LES methods had exponential time complexity, but LES had a much smaller rate of growth with decreasing timestep size. We fitted the data points to an exponential fit of the form

$$f(x) = ae^{-bx} + c, \tag{50}$$

where a, b, and c are parameters. The results are shown in Figure 5, and the results of the fit are given below.

11

Table 3: Time Spent to Compute Simulation LES

dt (s)	Number of Iterations	Support Points	LES result (s)
0.2	50	60000	13.1
0.1	100	60000	25.2
0.05	200	60000	49.0
0.025	400	60000	108.9

	a	b	c
hsu_3D	743.5938806634053	33.8671796117633	52.91721007728185
LES	245.96632953610074	39.135674499151285	16.0009187924904

Running both the DNS and LES at various time steps and plotting the points on a dt vs. Real-time values in seconds graph resulted in the following exponential function:
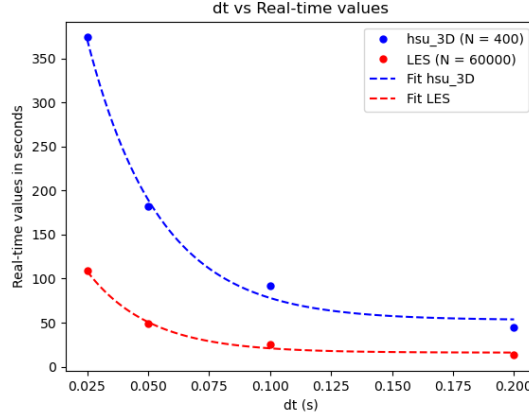

Figure 5: A plot of time taken for a time interval of 10 seconds using LES and a half step upwind finite difference solver. The half step upwind solver was run on a 3D grid with 400 support points. The LES was run on a 3D grid with 60000 support points.

To calculate the average factor difference across computation of the LES and DNS we used the following equation:

$$k = \sum \frac{DNS_i}{LES_i} \times \frac{1}{n} \tag{51}$$

where n is the number of trials and i is the $i'th$ trial of the solvers.

To calculate the variance, we adapted the general variance equation stating that

$$S^2 = \frac{\sum (\frac{DNS_i}{LES_i} - k)^2}{n - 1} \tag{52}$$

By entering our results into the previous equations, we get that the LES ran roughly $(3.6\pm0.02)\times$ faster than DNS with $150\times$ as many support points at each time step.

5   Conclusions and Discussion

The FS solver seemed to show some inconsistent results, which may warrant future investigation. However, the FS solver has a much larger stability range and is more robust in general. Although the HS and HSU solvers

have a much smaller stability range, their errors are significantly more controlled, taking the form of periodic oscillations. More work can be done to develop more efficient implementations of these solvers to better test their performance relative to each other.

LES is consistently faster than DNS by a multiple of around 3.6 even with 150 × more support points because it does not compute smaller, less significant, eddies. The general rule described earlier that states that LES attempts to resolve 80% of kinetic energy provided by the eddies in turbulent flow provides the LES with the capability of terminating less relevant eddy calculations. These calculations seem to have a significant effect on the time complexity of the DNS simulation even though it has a decreasing contribution to the net kinetic energy of the system. Likewise, in LES, the total energy is conserved by integrating the 20% that remains, which is inputted back into the average velocity of the system. As a result, while not completely accurate, the simulation upholds basic physical identities that enable it to be leveraged as a powerful and fast simulation tool.

Due to the nature of implementing a non-optimized DNS in Python3 and comparing it to a compiled openFoam simulation written in C, there are many sources of error that could reduce the accuracy of the results comparing the two models. This is primarily due to the fact that the same code written and compiled in C will take less time to run than the same interpreted code in Python3. Thus, the difference between LES and DNS in this experiment may be exasperated.

5.1 Future Goals in Research

The future of researching incompressible fluids in the medical field holds promising avenues. As computational capabilities advance, there is an increasing focus on refining simulation models to capture even more intricate aspects of fluid dynamics. Future research goals may include incorporating more realistic physiological conditions, such as temperature variations and the influence of different bodily tissues on fluid behavior.

Researchers in this field can aspire to contribute to the development of more sophisticated simulation techniques that bridge the gap between theoretical modeling and practical applications. Exploring novel drug delivery strategies, including the use of nanotechnology and targeted therapies, presents exciting opportunities for making meaningful contributions to the field.

Additionally, interdisciplinary collaboration between physicists and medical professionals is a key future goal. Fostering partnerships that bring together expertise from both fields is essential for translating research findings into tangible advancements in drug delivery systems and medical treatments.

Another key idea that should be explored further is comparing the exact difference in computational power of DNS, RANS, and LES by utilizing mathematical derivations of the time-complexities of each algorithm. In doing so, one can offer a comprehensible guide that can direct a given user to a specific solver given their use-case.

In conclusion, the future goals of researching incompressible fluids in the medical field offer exciting prospects. Through continued advancements in simulation techniques, computational power, exploration of innovative drug delivery strategies, and interdisciplinary collaboration, researchers can play a vital role in shaping the future of medical science.

6 Acknowledgements

References

Chorin, A. J. (1968). Numerical solution of the navier-stokes equations. *Mathematics of computation*, *22*(104), 745–762.

Ferroni, M., De Gaetano, F., Cereda, M. G., & Boschetti, F. (2020). A drug delivery analysis of large molecules in ocular vitreous chamber: Dependency on saccadic movements after intravitreal injection. *Medical Engineering & Physics*, *82*, 49–57.

Mekheimer, K. S., Hasona, W., Abo-Elkhair, R., & Zaher, A. (2018). Peristaltic blood flow with gold nanoparticles as a third grade nanofluid in catheter: Application of cancer therapy. *Physics Letters A*, *382*(2-3), 85–93.

Nagy, T. K. (2018). Reproducing the kolmogorov spectrum of turbulence with a hierarchical linear cascade model. *semanticscholar*.

Nonino, M., Ballarin, F., Rozza, G., & Maday, Y. (2022, Nov). Projection based semi-implicit partitioned reduced basis method for fluid-structure interaction problems. , *94*(1). doi: 10.1007/s10915-022-02049-6

Peskin, C. S. (1972, Oct). Flow patterns around heart valves: A numerical method. , *10*(2). doi: 10.1016/0021-9991(72)90065-4

Peskin, C. S. (1997, November). Fluid Dynamics of the Heart and its Valves. , 3INV.02.

Peskin, C. S. (2002). The immersed boundary method. *Acta numerica*, *11*, 479–517.

Richtmyer, R. D. (1957). *Difference methods for initial-value problems* (Vol. 4). Interscience Publishers.

RIVAL, D. E. (2023). Chapter 2: Fundamentals. In *Biological and bio-inspired fluid dynamics: Theory and application* (p. 31–32). SPRINGER NATURE.

Santhanakrishnan, A., & Miller, L. A. (2011). Fluid dynamics of heart development. *Cell biochemistry and biophysics*, *61*, 1–22.

Smits, A. J. (2009). *Viscous flows and turbulence* (Tech. Rep.). Princeton University.

Wikibooks. (n.d.). *Fluid mechanics/incompressible flow.* Retrieved from https://en.wikibooks.org/wiki/Fluid_Mechanics/Incompressible_Flow (Retrieved November 27, 2023)