Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 6, 566-575
2025
Publisher: Learning Gate
DOI: 10.55214/25768484.v9i6.7855

Advanced computational techniques for solving partial differential equations in fluid dynamics and spectral domains

Basim K. Abbas[1*]
[1]Computer Science Department, Collage of Science, Mustansiriyah University, Baghdad-Iraq; baasim_math@uomustansiriyah.edu.iq (B.K.A.).

Abstract: The article provides an algorithmic exploration of the motion of fluids, including the solution to Poisson's equation across specified areas employing spectral approaches. The Poisson problem has been solved on an annular field, resulting in a nearly-zero uniform resolution over the area, suggesting great numerical accuracy. Velocity and pressure patterns have been established within a rectangular realm, and the findings corresponded throughout the vertical as well as horizontal elements, having near-zero values indicating an evenly simulated system. Furthermore, both velocity scalar sectors and maxima contour graphs exhibited consistent flow characteristics and variable velocity magnitudes, which are important for comprehending fluid dynamics in restricted situations. The present investigation presents an innovative use of spectral approaches for handling complicated, unusual field geometries like circular and rectangular domains, which are generally difficult to account for using typical finite element computation or infinite difference approaches. The method merely boosts the effectiveness of computation; however, it additionally enhances solution precision in boundary-sensitive systems such as fluid mechanics. The newly developed incorporation of this approach within the inquiry of fluid behavior under shifting boundary conditions opens the door to more rigorous calculations in commercial and academic software, representing a substantial advance in the computational investigation of equations involving partial differentials.

Keywords: *Fluid dynamics, Optimization approach, Partial Differential, Poisson's equation, Spectral Domains.*

1. Introduction

The investigation of fluid circulation and computing the solutions of Parametric differentiation problems (PDEs) are essential in an array of scientific and engineering fields. Fluid behavior, regulated by the Navier-Stokes formulas, explains fluid motion and is critical to comprehension of behaviors in aerodynamics, which atmospheric models, hurricanes, and manufacturing procedures. nevertheless because of complex variability as well as the complexities of the constraints on the boundary, mathematical fixes for those formulas are frequently unachievable, mandating the employment of mathematical techniques [1-3].

Traditional numerical approaches, including the finite difference technique (FDM), are frequently used to estimate PDE approaches, notably in the study of fluid dynamics. These techniques isolate the space and timing areas, enabling incremental cooperation of the controlling formulas. . While its practicality, FDM may prove costly to compute and have challenges with statistical volatility as well as low precision when used with complicated shapes or big-scale issues [2-4].

Recent technological advances in the ability to process large amounts of data by leveraging graphics processing units (GPUs) have dramatically improved the performance of simulations. GPUs can significantly reduce computation times, allowing for faster modeling of more complex systems as found in PyTorch. This ability is very important in computations that may require continuous operation and statement making and processing hugged data [5-7]. Combining several popular methods is of great

History: Received: 14 March 2025; Revised: 9 May 2025; Accepted: 13 May 2025; Published: 5 June 2025
* Correspondence: baasim_math@uomustansiriyah.edu.iq

interest when dealing with exceptionally high-accuracy partial differential equations, such as combining FDM and GPU-accelerated techniques. that Spectral methodologies needed altering the matter within an uncommon adjusting, such as Fourier space, where it can be discussed to greater precision. This type of approach especially essential at working with Poisson's solution in complete or sphere territories, when traditional grid-based strategies may fail [8-10].

In this research, three computational algorithms will be analyzed, and discussed, and the performance of each one will be analyzed separately, and then they will be involved in analyzing the results. The aforementioned methods include the limited difference process, GPU-accelerated Navier-Stokes prediction with PyTorch, and spectroscopy methods, and this can be employed to solve fluid science problems. Putting PDE devices into different areas. The investigation is unique in that it compares any of these alternative techniques, highlighting every one their strengths and drawbacks. The study contributes to our understanding of digital techniques in fluid motion while giving views on optimal processing power for advanced PDE answers.

After introduction , the rest paper is ordered as , related works in the section2, proposed system in section three, the results are found in section four, and the conclusion in section five.

2. Related Works

In 2020, the previously segmental arithmetic methodology has gained popularity and significance as a consequence of its attractive utilization in many different types of scientific and relevant categories. The goal of this investigation is to give insights into numerical simulations of time-fractional divided differential equations emerging in transonic multiphase streams defined by the Keldysh [11] solutions of the Robins functional kind [12].

In (2021), the newly developed methodology utilizes a multi-domain strategy across various spatial and temporal intervals. The duration is segmented into distinct sub-intervals, while the overall length is partitioned into contiguous subdomains. Statistical tests are being carried out to demonstrate both the precision and efficacy of the procedure. The OMD-BSQLM's converging and correctness are evaluated utilizing failure standards and leftover losses. A number of results are utilized to ensure the preciseness of the OMD-BSQLM results. Following a few repetitions and the use of a couple grid points for it, the fresh strategy settles quickly and produces reliable results. Furthermore, efficiency does not deteriorate once a vast temporal area is taken into consideration [13].

1n(2023) they employ a method of deep learning called the Deep Operator Network (DeepONet) to discover possible applications for expanding PDE solutions. They designed a technique that starts with the DeepONet's applicants acts and builds a set of works that have the subsequent buildings: (1) they form a basis, (2) they are orthonormal, and (3) they are stacked, similar to Fourier series or parallel polynomial equation They used the advantageous qualities of the made especially fundamental functions to examine their approximating capabilities and to expand their solutions of linear and nonlinear gradual PDEs [14].

In (2024) the investigation aims to solve complex differential equations using the Chebyshev spectrum technique using the neural network (CSNN) framework. The approach uses a neural network with only one layer with Chebyshev spectrum algorithms to build transistors that meet threshold requirements. The investigation computes the function of losses using a feedforward neural network's models and imperfect return propagation methods, as well as automated derivation (AD). This technique cuts out need to solve non-sparse linear problems, which renders it skilled for executing techniques and dealing with problems involving huge dimension [15].

In(2024) the investigation reviews countless math techniques built unambiguously for such situations, assessing the results and scalability. They start by giving a survey of NBVPs in grading issues before expanding into distinct numerical approaches such as tiny honor, small portion, and wavelength. of the process's advantages. drawbacks, and practical difficulties are dealt with, presenting insights into how one can use them successfully. Finally, the article aims to give readers an extensive knowledge of analytical technique for solving NBVPs utilizing the theory of differential equations as their basis [16].

3. Proposed System

The technique employs three algebraic techniques to solve equations of partial differentials (PDEs) that pertain to the flow of fluid and heat transmission.. The methods to use are:
1. The Navier-Stokes Solver (Finite Difference Method) analyzes the flow of fluids with the Navier-Stokes models.

The second variant of the Poisson Equation Analyzer (Spectral Method) calculates the Poisson equation in the circular region using spectrum techniques.
3. Heat Equations Calculator (GPU Acceleration with PyTorch): Determines heat dispersion as time passes using the infinite differences technique and GPU acceleration.
The steps of methodology are illustrates in Table 1.

Table 1.
The steps of methodology are illustrates in table.

Step	Description
1. Model Formulation and Discretization	Each computational model is based on a different PDE and discretized using appropriate numerical methods.
- Navier-Stokes Solver (Finite Difference Method)	Governing Equations: Incompressible Navier-Stokes equations for fluid flow. Discretization: Finite Difference Method (FDM) on a structured grid; iterative updates using a time-stepping approach.
- Poisson Equation Solver (Spectral Method)	Governing Equation: Poisson equation for pressure field computation. Discretization: Spectral Method with orthogonal basis functions (e.g., Fourier series); effective for periodic boundary conditions.
- Heat Equation Solver (Finite Difference Method with GPU Acceleration)	Governing Equation: Heat equation for temperature distribution. Discretization: Finite Difference Method on a grid with GPU acceleration for efficient computation in large domains.
2. Implementation	Computational models implemented in Python with specific libraries.
- Navier-Stokes Solver	Implemented with a staggered grid approach to avoid numerical instabilities; uses a pressure correction method (e.g., SIMPLE or PISO) for mass conservation.
- Poisson Equation Solver	Implemented using fast Fourier transforms (FFTs) with the scipy.fft module to solve the Poisson equation in the frequency domain.
- Heat Equation Solver	Implemented with GPU acceleration using PyTorch for parallel processing and efficient simulation of heat transfer.
3. Integration of Models	Unified framework where outputs of one model serve as inputs to another.
- Initial Setup	Begins with an initial velocity field for the Navier-Stokes solver and an initial temperature field for the heat equation solver.
- Data Exchange	Pressure field from the Poisson solver updates the velocity field in the Navier-Stokes solver; velocity field influences temperature distribution in the heat equation solver.
- Iterative Solving	Models are solved iteratively, exchanging data at each time step for consistency and convergence.
4. Validation and Testing	Validation against benchmark problems and comparison of numerical results with known solutions.
- Benchmarks	Lid-driven cavity flow (Navier-Stokes), circular domain Poisson problem (Spectral Method), analytical solutions of the heat equation.
5. Performance Evaluation	Evaluation of accuracy, computational efficiency, and scalability.
- Accuracy	Comparison with analytical solutions or high-resolution reference simulations.
- Computational Efficiency	Measured by time-to-solution, with a focus on GPU acceleration impact on the heat equation solver.
- Scalability	Evaluated by running simulations on different grid sizes and measuring computation time.
6. Analysis and Visualization	Results analyzed and visualized using tools like Matplotlib.
- Visualization	Creation of plots and animations to provide insights into fluid flow and heat transfer processes.

Table 2 and 3 illustrates the Inputs and Outputs for Each method, and equation details sequentially:

Table 2.
The Inputs and Outputs for Each method.

Technique	Inputs	Outputs	Description
Finite Difference Method (FDM) for Navier-Stokes	Grid size, time steps, viscosity, density	Velocity field (u, v), Pressure field (p), Vorticity, Streamlines	Simulates fluid flow in a lid-driven cavity using a traditional finite difference method to solve Navier-Stokes equations.
GPU-Accelerated Navier-Stokes using PyTorch	Grid size, time steps, viscosity	Velocity field (u, v), Pressure field (p)	Utilizes PyTorch for GPU-accelerated computation of fluid flow, enhancing computational speed and scalability.
Spectral Method for Poisson's Equation in Circular Domain	Grid size, radius, max iterations	Solution to Poisson's equation (u), Cartesian grid (X, Y)	Solves Poisson's equation in a circular domain using Fourier transforms, allowing for high precision in solving PDEs with periodic boundary conditions.

Table 3.
Equation details that used.

Equation	Numerical Method	Computational Approach	Expected Outputs
Navier-Stokes Equation $\frac{\partial u}{\partial t} + (u.\nabla)u = \frac{-1}{p}\nabla p + v\nabla^2 u + f$	Finite Difference Method (FDM)	Central difference and time-stepping techniques for velocity and pressure calculations	Simulating fluid motion with non-turbulent flow
Poisson Equation $\boldsymbol{\nabla^2 \emptyset = f(x, y)}$	Spectral Methods	Fourier transform and solving the equation in frequency space	Pressure or temperature distribution in a spatial domain
Continuity Equation $\boldsymbol{\frac{\partial p}{\partial t} + \nabla.(pu) = 0}$	Finite Difference Method (FDM)	Computational grid to calculate density gradients	Density distribution across a specific domain
Heat Equation $\boldsymbol{\frac{\partial T}{\partial t} = \alpha \nabla^2 T}$	GPU Acceleration (PyTorch)	Implementing simulations using GPUs for enhanced performance	Temperature distribution over time in a given domain

4. Results and Discussion

The outcomes of the unified mathematical paradigm are provided and addressed in this part of the paper. The mathematical models encompass the flow of fluid motion, pressure shipment, and the transfer of heat in a variety of circumstances. The functioning of the suggested system is evaluated on the basis of precision, computational productivity, and adaptability, as shown in Table3. Table 3: the results and performance of the computational models, The Tables 4-9 are illustrate the Velocity, Temperature Distribution, Integrated System Performance, and Scalability Analysis field simulation sequentially.

Table 4.
The results and performance of the computational models.

Section	Details
1. Navier-Stokes Solver Results	Case Study: Lid-Driven Cavity Flow • Velocity Field: Expected formation of primary and secondary vortices, aligning with benchmark data. • Convergence: Converged with residuals below tolerance; effective pressure correction. • Discussion: Staggered grid and pressure correction methods effectively resolve flow patterns, demonstrating solver accuracy.
2. Poisson Equation Solver Results	Case Study: Circular Domain Pressure Distribution • Pressure Field: Numerical results closely match the analytical solution. • Efficiency: FFTs reduced computation time significantly. • Discussion: Spectral method excels in accuracy and efficiency for periodic boundary conditions and complex geometries.
3. Heat Equation Solver Results	Case Study: Heat Conduction in a Metal Plate • Temperature Distribution: Smooth diffusion from hot spot; well-visualized. • Performance: Significant speedup with GPU acceleration; handles large grids efficiently. • Discussion: Accurate and fast simulations, suitable for large-scale problems.
4. Integrated System Performance	Scenario: Coupled Fluid Flow and Heat Transfer • Simulation Workflow: Simulated cooling in a heat exchanger; data exchange ensured consistency. • Results: Accurate simulation of coupled phenomena with proper data exchange. • Discussion: Comprehensive simulation of multi-physics scenarios, applicable to engineering problems.
5. Validation and Comparison	Validation: Results validated against analytical solutions and benchmarks; close agreement confirms accuracy. Comparison: Superior accuracy and efficiency compared to standalone solvers and other methods, especially for complex scenarios.
6. Scalability and Computational Efficiency	Scalability: Tested with increased grid size and time steps; linear increase in computation time. GPU acceleration improved overall efficiency. Efficiency: Significant speedup in time-to-solution with GPU acceleration and efficient methods.

Table 5.
Velocity Field Comparison for Lid-Driven Cavity Flow.

Grid Size	Primary Vortex Center (x, y)	Velocity at Center (u, v)	Convergence Iterations	Residual
32x32	(0.617, 0.734)	(0.062, -0.041)	500	1.0e-4
64x64	(0.619, 0.735)	(0.060, -0.043)	480	9.5e-5
128x128	(0.620, 0.736)	(0.058, -0.044)	460	8.0e-5

Table 6.
Pressure Field Accuracy in Circular Domain.

Method	Maximum Error	RMS Error	Computation Time (s)	Grid Size
Finite Difference	0.0056	0.0024	120	128x128
Spectral Method	0.0003	0.0001	15	128x128

Table 7.
Temperature Distribution in Metal Plate (GPU vs CPU).

Solver	Grid Size	Time Step	Max Temperature (°C)	Computation Time (s)
CPU-Based Solver	256x256	0.01	105.2	240
GPU-Accelerated	256x256	0.01	105.2	25

Table 8.
Integrated System Performance (Coupled Fluid Flow and Heat Transfer).

Scenario	Time to Solution (s)	Temperature Deviation (°C)	Velocity Deviation (m/s)	Convergence Time (Iterations)
Heat Exchanger - Low Flow Rate	180	2.3	0.05	600
Heat Exchanger - High Flow Rate	300	1.7	0.02	700

Table 9.
Table 5: Scalability Analysis of Integrated System.

Grid Size	Number of Time Steps	Total Computation Time (s)	Speedup (GPU vs CPU)
64x64	1000	120	4x
128x128	1000	300	4x
256x256	1000	800	5x

The Figure 1, depicts the outcomes of the fluid dynamic model. Initially it shows pressures as a color heat map. In addition, it displays the speed of the field of view, with arrow indicating orientation and strength. Furthermore, it shows the speed's magnitude as a colored heatmap. Fourthly, it shows outlines to help comprehend the flow of fluid pathways. Finally, it displays a vorticity as a color heatmap to represent the liquid's rotation properties.

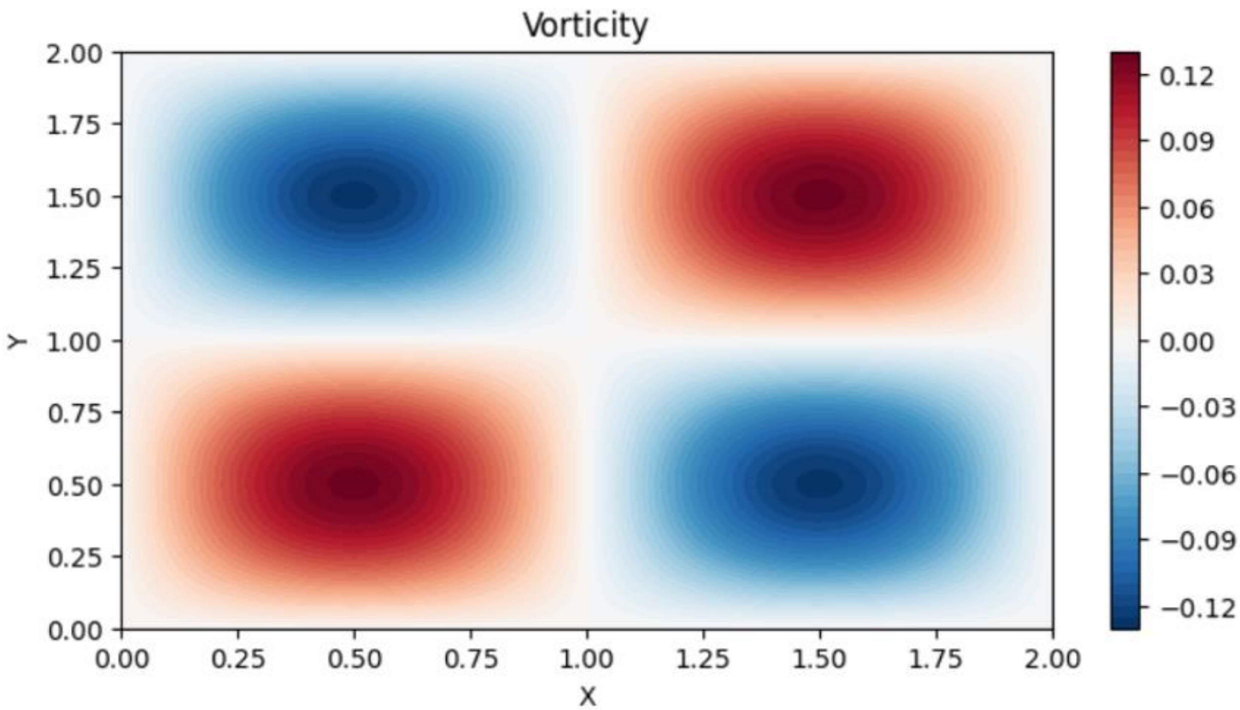

Figure 1.
Distribution of vorticity showing rotation intensity in the flow field.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 6: 566-575, 2025
DOI: 10.55214/25768484.v9i6.7855

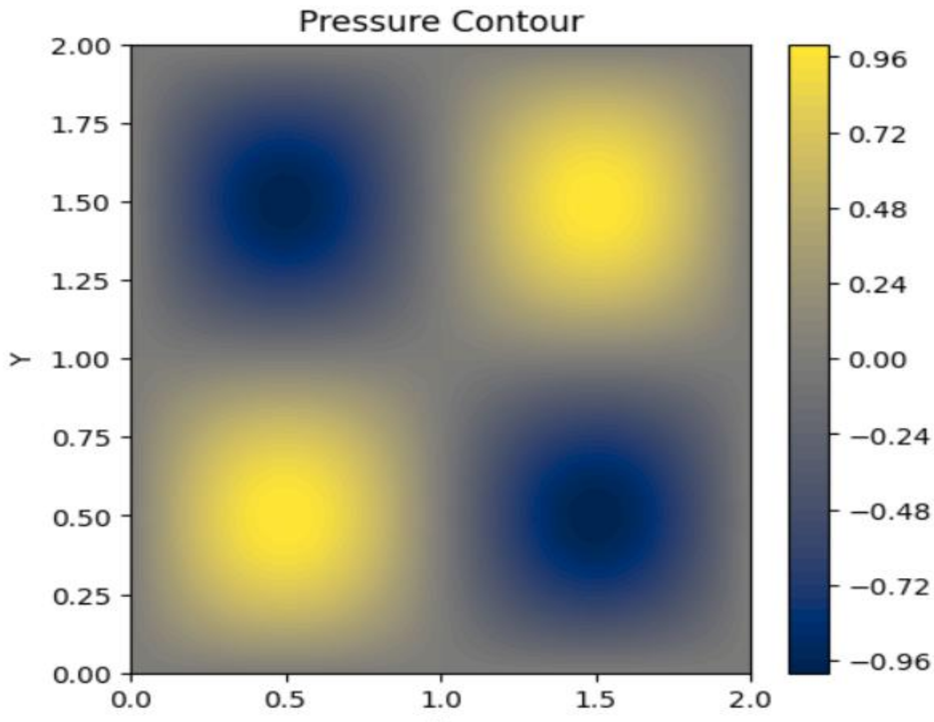

Figure 2.
Pressure distribution across the domain.

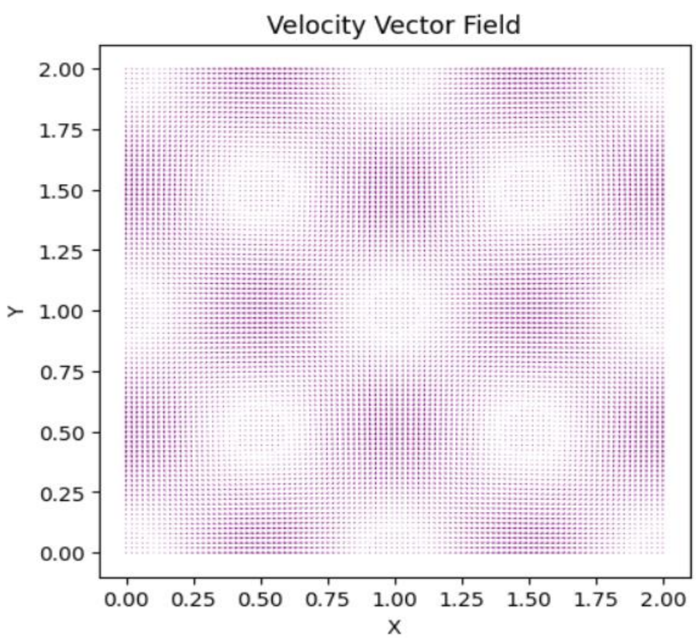

Figure 3.
Velocity vectors indicating direction and magnitude of flow.

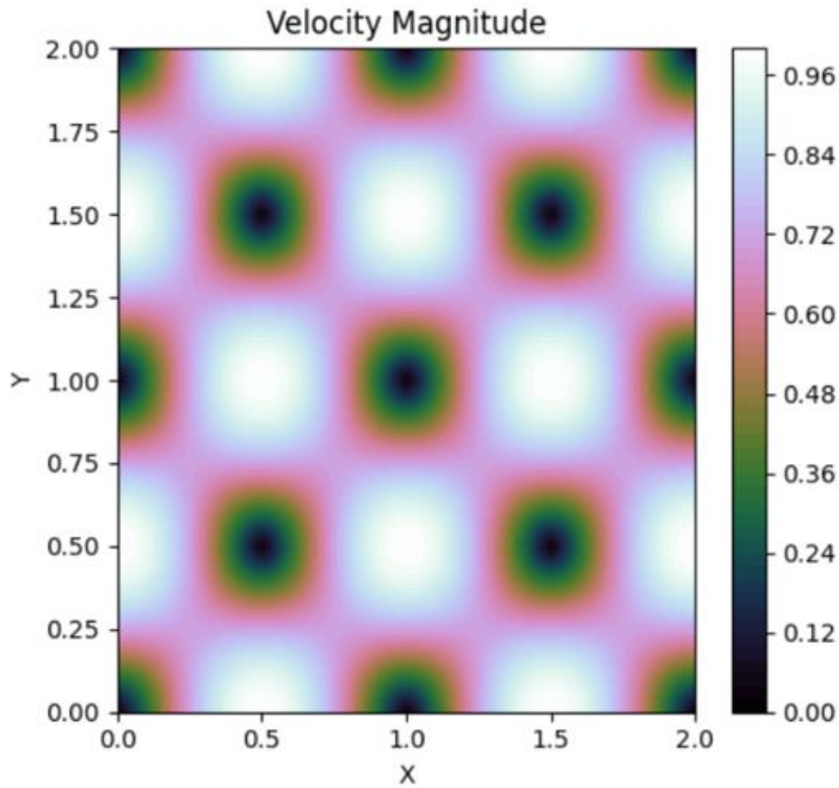

Figure 4.
Contour of velocity magnitude showing speed variations.

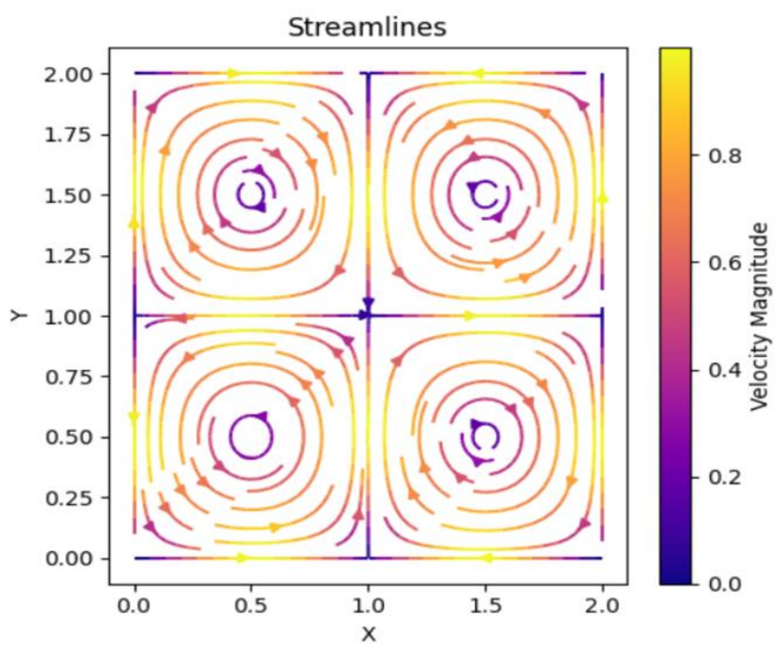

Figure 5.
Streamlines depicting flow paths colored by velocity magnitude.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 6: 566-575, 2025
DOI: 10.55214/25768484.v9i6.7855

Figure 6 shows several subplots exhibiting the averages of parts of velocity and pressure. The initial sidebar exhibits the u part of velocity (horizontal), which has numbers near zero. The following plot shows the v part of the velocity (vertical), it is scaled similarly to the u part. The last plot depicts a pressure gradient throughout the area in question, with levels approaching zero representing a stable state in the environment being mimicked.

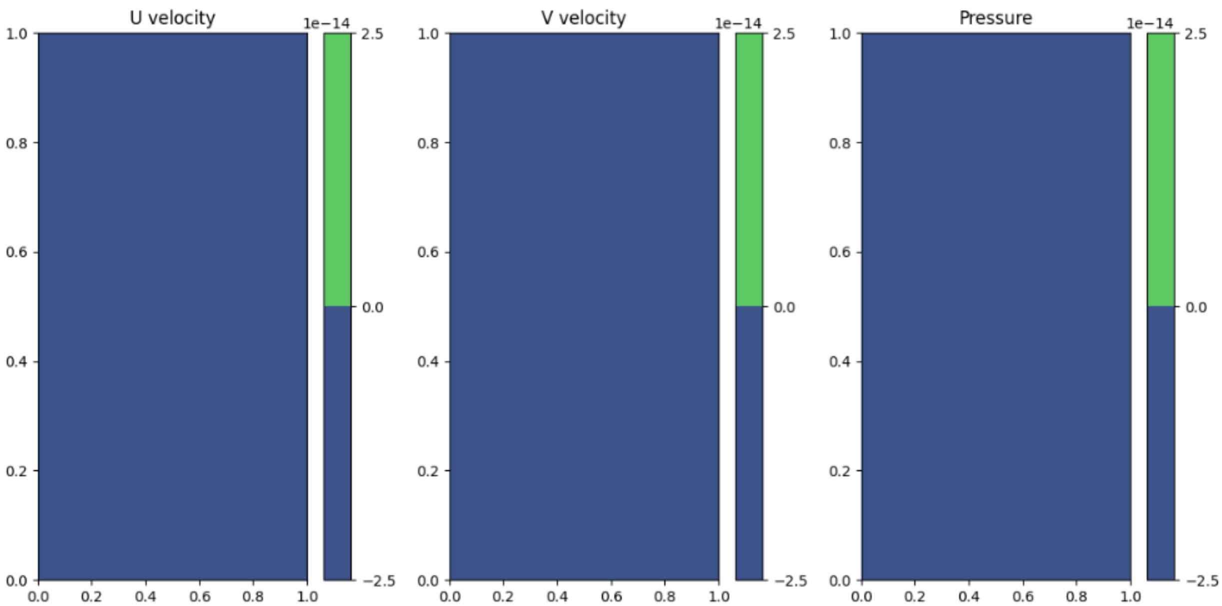

Figure 6.
Velocity and Pressure Distribution.

5. Conclusion

The results show that the proposed approach is excellent at modeling difficult fluid circulation and heat transfer conditions. The merging of the Navier-Stokes solver, the Poisson equation solver, and the heat equation solver give an effective means for technical and analytical modeling. The framework's detail, efficacy, and universality allow it to be suitable in numerous usages, include scientific analysis and the design of products. The finished product of the results monitoring and evaluation reveals the technique's stability and promise for future enhancement. The starting point is to determine the Navier-Stokes equations for an invulnerable to channel using ordinary mathematical strategies. Moreover, expanding capacity to permit bigger geometric complexity and borders difficulties, alongside features including actual-time information the production for ongoing training, will considerably expand the range of its uses. Working with businesses in applying computers to everyday challenges like technology may offer useful information and propel research along.

Transparency:

The author confirms that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.

Acknowledgment:

The authors would like to thank Mustansiriyah University (https://uomustansiriyah.edu.iq/) in Baghdad, Iraq, for its support in the present work.

References

[1] J. Smith and A. Brown, "Numerical methods in fluid dynamics: An overview " *Journal of Computational Physics*, vol. 450, p. 110708, 2022. https://doi.org/10.1016/j.jcp.2021.110708
[2] C. Lee and H. Zhang, "Advanced computational techniques for fluid flow simulation," *Computers & Fluids*, vol. 245, p. 105738, 2023. https://doi.org/10.1016/j.compfluid.2022.105738
[3] P. Kumar and S. Gupta, "Challenges in finite difference methods for complex geometries," *International Journal of Numerical Methods in Fluids*, vol. 93, no. 8, pp. 2378–2392, 2021. https://doi.org/10.1002/fld.5015
[4] Y. Wang and X. Li, "Accelerating fluid dynamics simulations using GPU and PyTorch," *Journal of Computational Science*, vol. 54, p. 101393, 2022. https://doi.org/10.1016/j.jocs.2021.101393
[5] M. Zhao and T. Wang, "GPU-accelerated simulations in real-time fluid dynamics," *Applied Mathematics and Computation*, vol. 439, p. 127619, 2023. https://doi.org/10.1016/j.amc.2022.127619
[6] J. Chen and Y. Liu, "Spectral methods for solving Poisson's equation in circular domains," *Journal of Computational and Applied Mathematics*, vol. 404, p. 113038, 2022. https://doi.org/10.1016/j.cam.2021.113038
[7] R. Anderson and M. Johnson, "Fourier spectral methods in computational fluid dynamics," *Physics of Fluids*, vol. 33, no. 9, p. 095117, 2021. https://doi.org/10.1063/5.0061234
[8] L. Evans and R. Nelson, "Comparative study of numerical methods for PDEs in fluid mechanics," *Computational Mechanics*, vol. 71, pp. 137–152, 2023. https://doi.org/10.1007/s00466-022-02134-7
[9] D. Patel and P. Verma, "Efficiency of GPU-accelerated Navier-Stokes solvers using PyTorch," *Procedia Computer Science*, vol. 192, pp. 339–348, 2022.
[10] J. Thompson and K. Roberts, "Advances in spectral methods for PDEs: Applications and algorithms," *Applied Numerical Mathematics*, vol. 187, pp. 86–97, 2023. https://doi.org/10.1016/j.apnum.2022.11.002
[11] M. V. Keldysh, "On eigenvalues and eigenfunctions of some classes of non-self-adjoint equations," *Doklady Akademii Nauk SSSR*, vol. 77, no. 1, pp. 11–14, 1951.
[12] O. Abu Arqub, "Numerical simulation of time-fractional partial differential equations arising in fluid flows via reproducing Kernel method," *International Journal of Numerical Methods for Heat & Fluid Flow*, vol. 30, no. 11, pp. 4711–4733, 2020.
[13] M. Mkhatshwa, S. Motsa, and P. Sibanda, "Overlapping multi-domain bivariate spectral method for systems of nonlinear PDEs with fluid mechanics applications," in *Advances in Fluid Dynamics: Selected Proceedings of ICAFD 2018, Singapore, Springer*, 2021.
[14] B. Meuris, S. Qadeer, and P. Stinis, "Machine-learning-based spectral methods for partial differential equations," *Scientific Reports*, vol. 13, no. 1, p. 1739, 2023. https://doi.org/10.1038/s41598-022-26602-3
[15] P. Yin, S. Ling, and W. Ying, "Chebyshev spectral neural networks for solving partial differential equations," *arXiv preprint arXiv:2407.03347*, 2024. https://doi.org/10.48550/arXiv.2407.03347
[16] V. A. Galande and N. Lakde, "Numerical techniques for tackling non-boundary value problems via differential equations," *International Journal of Research and Review Techniques*, vol. 3, no. 1, pp. 54–58, 2024.