MT5846: Advanced Computational Techniques

Project One

Name:Jim YANG
Matriculation number:230018398

Date: March 2024

Abstract

This report investigates the behavior of viscous fluids confined between two parallel plates, an essential study in fluid dynamics. It centers on the velocity evolution of the fluid, influenced by kinematic viscosity, and governed by a simplified Partial Differential Equation (PDE) that disregards edge effects due to the plates' infinite extent. A general form of Heat Equation will be applied to solve [1]. This report focuses on the study of this system using different Numerical methods within a fixed reaction time. Employing numerical methods such as FTCS, DuFort-Frankel, Laasonen, and Crank-Nicolson, the study critically analyzes stability, accuracy, and computational efficiency. Each method's performance is evaluated against various timestep sizes, revealing that while smaller timesteps yield more accurate results, larger timesteps, although computationally more feasible, introduce significant errors. The report highlights the Laasonen implicit method for its stability and accuracy across a range of timestep sizes, making it preferable for modeling fluid dynamics in practical scenarios where a balance between computational speed and solution accuracy is required.

KEYWORDS: Viscous fluids; Numerical Methods; Stability analysis; Error analysis

Contents

1 Introduction

This report explores the dynamics of a viscous fluid constrained between two parallel plates, a fundamental problem in fluid dynamics. The focus is on modeling the fluid's velocity evolution over time, governed by a Partial Differential Equation (PDE) that reflects the kinematic viscosity's impact. This scenario is idealized by assuming infinite plate extension, eliminating edge effects and allowing a focus on the fluid's behavior due to an instantaneous lower plate velocity change.

The approach involves applying numerical methods to solve the PDE, specifically targeting the challenges and intricacies of the FTCS explicit method, DuFort-Frankel explicit method, Laasonen implicit method, and Crank-Nicolson implicit method. Each method presents unique advantages and limitations, particularly regarding stability and accuracy, which will be thoroughly investigated. The project not only aims to understand the fluid's behavior under specified initial and boundary conditions but also to evaluate the numerical methods' efficacy and reliability in simulating complex physical systems.

By constructing the given function, the analytical results can be found by solving the function under different conditions. The function is given [2], and the analytical solutions should be considered as accurate enough. The errors will be found by calculating the difference between the values of the numerical solutions and the values of the analytical solutions

Our analysis pays particular attention to the Laasonen method, where we embark on a detailed analysis based on different time steps to see how the numerical solution changed. The exploration extends to identifying potential limitations and strengths unique to the Laasonen method, as well as the other numerical techniques under consideration.

2 Model

This investigation explores the behavior of a viscous fluid confined between two endlessly extended parallel plates, simplified from the complex Navier-Stokes equations. The simplification is grounded on several assumptions: the fluid's in-compressibility, the flow's bi-dimensionality and steadiness, and the disregard of any end effects due to the plates' infinite length. This approach allows for a focused examination of fluid dynamics within a constrained geometry, shedding light on the fundamental principles that govern fluid movement in such settings.

This results in a partial differential equation (PDE) that models the fluid velocity u, as a function of time t and the vertical position y, as follows:

$$\frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial y^2} \tag{1}$$

where ν is the kinematic viscosity of the fluid. This PDE is subject to initial and boundary conditions that reflect the scenario where the fluid is initially at rest, and the bottom plate is instantaneously accelerated to a velocity U_0, while the top plate remains stationary.

An illustrative figure can significantly enhance the understanding of the setup and the fluid dynamics involved. The figure should depict:
- The configuration of the parallel plates, with the bottom plate moving at a constant velocity U_0 and the top plate stationary.
- The developing velocity profile of the fluid between the plates, from the initial state to the final state,

highlighting how the fluid velocity varies with the vertical distance y from the moving plate.

3 Numerical approach

This section outlines the numerical methods employed to simulate the dynamics of a viscous fluid confined between two parallel plates, as dictated by the given partial differential equation (PDE). The study leverages explicit and implicit finite difference methods (FDM) to approximate the solution, including the Forward-Time Central-Space (FTCS) method, the DuFort-Frankel method, the Laasonen implicit method, and the Crank-Nicolson method. These approaches are chosen for their ability to balance accuracy, stability, and computational efficiency in solving time-dependent PDEs.

3.1 Initial and Boundary Conditions

The simulation initiates with the fluid at rest, leading to the following initial condition:

$$u(y, 0) = 0, \quad 0 < y \leq h \tag{2}$$

and employs constant Dirichlet boundary conditions to model the movement of the bottom plate and the stationary top plate:

$$u(0, t) = U_0, \quad t > 0, \tag{3}$$
$$u(h, t) = 0, \quad t > 0. \tag{4}$$

where U_0 is the velocity of the bottom plate, and h is the distance between the plates.

3.2 Grid Discretization and Numerical Parameters

The computational domain is discretized with a grid size $\Delta y = 0.001$ m, resulting in $j_m = 40$ spatial points, and Δt varies per the method used, reflecting the trade-off between computational stability and accuracy. The choice of Δy and Δt ensures sufficient resolution to capture the fluid's velocity profile evolution while maintaining numerical stability as per the Von Neumann stability analysis.

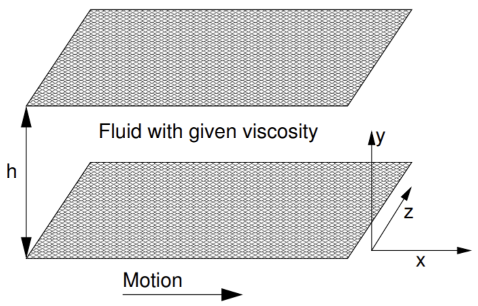

Figure 1: Parallel Plate Simulation

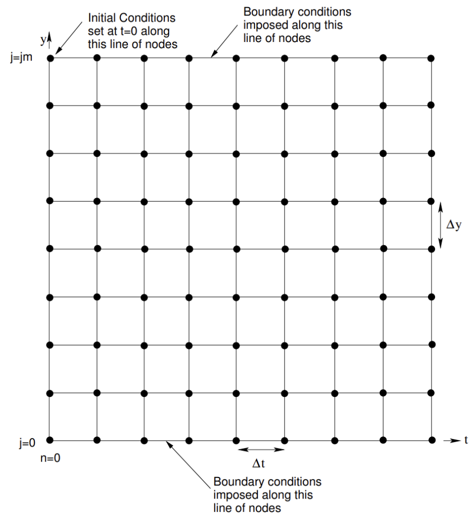

Figure 2: Computational Grid

3.3 Numerical Methods

This study employs four numerical methods to solve the partial differential equation describing the fluid flow between two parallel plates. Each method is selected for its unique advantages in terms of stability, accuracy, and computational efficiency.

3.3.1 Forward-Time Central-Space (FTCS) Method

The Forward-Time Central-Space (FTCS) method is an explicit scheme, characterized by its simplicity and straightforward implementation. It approximates the time derivative with a forward difference and the spatial derivative with a central difference, resulting in the following finite difference equation:

$$u_j^{n+1} = u_j^n + \frac{\Delta t}{\Delta y^2}\nu(u_{j+1}^n - 2u_j^n + u_{j-1}^n) \tag{5}$$

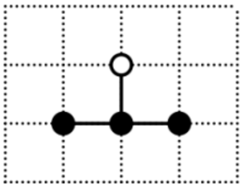

Figure 3: Stencil for FTCS Method

3.3.2 DuFort-Frankel Method

The DuFort-Frankel method is employed to enhance numerical stability for explicit schemes. It is particularly useful for addressing numerical dispersion. The method uses a staggered time-stepping strategy, represented by:

$$u_j^{n+1} = \frac{(2\Delta t\nu/\Delta y^2)(u_{j+1}^n + u_{j-1}^n) + u_j^{n-1}}{1 + 2(\Delta t\nu/\Delta y^2)} \tag{6}$$

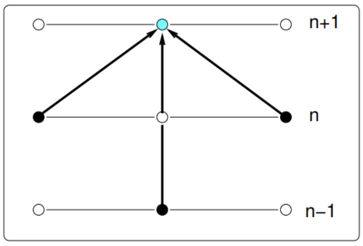

Figure 4: Stencil for DuFort-Frankel Method

3.3.3 Laasonen Implicit Method

The Laasonen implicit method is known for its unconditional stability, making it suitable for simulations over longer time scales. It approximates the spatial derivative using a backward difference, leading to the equation:

$$u_j^{n+1} = u_j^n + \frac{\Delta t}{\Delta y^2}\nu(u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}) \tag{7}$$

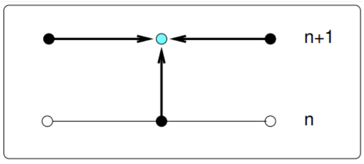

Figure 5: Stencil for Laasonen Implicit Method

3.3.4 Crank-Nicolson Method

The Crank-Nicolson method offers a balanced approach between accuracy and stability by averaging the implicit and explicit methods. It is formulated as:

$$u_j^{n+1} = u_j^n + \frac{\Delta t}{2\Delta y^2}\nu \left[(u_{j+1}^n - 2u_j^n + u_{j-1}^n) + (u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1})\right] \tag{8}$$

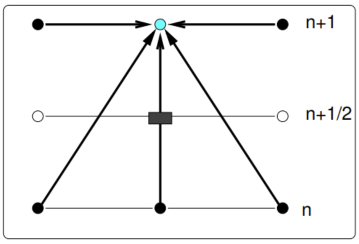

Figure 6: Stencil for Crank-Nicolson Method

*Those stencil graphs were taken from the notes [3].

3.4 Stability analysis

The numerical methods' convergence is mainly determined by stability and consistency. Here, only the stability analysis is considered. The consistency of the 4 given methods had been shown [4]. A general result of this part states the FTCS is conditional stable and the other 3 methods are unconditionally stable, where the stable condition is related to the value of parameters.

Von Neumann analysis is used to assess the stability of numerical methods by examining how errors propagate through iterations. It involves analyzing the growth factor of errors in the discrete Fourier modes of the numerical solution. If the growth factor for all possible modes is less than or equal to one, the method is considered stable, as it indicates that errors will not amplify uncontrollably over time. This criterion provides a mathematical basis to predict the behavior of numerical errors, ensuring the reliability of simulations. The first order Taylor expansions have been omitted since they are simple, while the second order expansion will be only needed for the DuFort-Frankel method, which will be shown.

In this analysis, the parameter and the transform term can be written as:

$$\alpha = \frac{\nu \Delta t}{(\Delta x)^2} > 0 \tag{9}$$

$$u_i^n = \xi^n e^{I\theta i}, u_{i+1}^{n+1} = \xi^{n+1} e^{I\theta(i+1)}, u_{i-1}^{n-1} = \xi^{n-1} e^{I\theta(i-1)} \tag{10}$$

where $\xi^n e^{I\theta i}$ attains a transform from Fourier series, I is the imaginary unit, $\theta \in [-\pi, \pi]$.
For any stable numerical method, it requires that, for any n:

$$\xi^{n+1} \leq \xi^n \tag{11}$$

This will be the main idea to determine the stability of the given methods.

3.4.1 FTCS explicit method

For FTCS explicit method described in Equation 5, substitute the transformed term described in Eqn. 10 into the method's equation:

$$\xi^{n+1}e^{I\theta i} = \xi^n e^{I\theta i} + \alpha[\xi^n e^{I\theta(i-1)} - 2\xi^n e^{I\theta i} + \xi^n e^{I\theta(i+1)}]$$

Therefore,

$$\begin{aligned} \xi^{n+1} &= \xi^n[1 + \alpha(e^{-I\theta i} + e^{I\theta i} - 2)] \\ &= \xi^n[1 + \alpha(2cos(\theta) - 2)] \end{aligned}$$

Using the stable condition in 11, it requires that

$$|1 + 2\alpha(cos(\theta) - 1)| \leq 1$$

where

$$\begin{aligned} cos(\theta) &\in [-1, 1] \\ \alpha &\in (0, \infty) \end{aligned}$$

Therefore,

$$2\alpha(cos(\theta) - 1) \in [-4\alpha, 0]$$

$$|1 - 4\alpha| \leq 1$$

Thus, the stable condition for FTCS can be determined, this method is stable when the value of α such that:

$$\alpha \leq \frac{1}{2} \tag{12}$$

As mentioned above, this method is conditional stable method, where the parameter's value should satisfy the condition here. The other methods' stability will be discussed below while parts of the calculation steps will be simplified.

3.4.2 DuFort-Frankel Method

The stability analysis for this method is more complicated than others, from Eqn. 6, the transformed term can be rewritten as:

$$(1 + 2\alpha)\xi^{n+1}e^{I\theta i} = 2\alpha\xi^n(e^{I\theta(i+1)} + e^{I\theta(i-1)}) + (1 - 2\alpha)\xi^{n-1}e^{I\theta i}$$

Therefore,

$$\begin{aligned} \xi(1 + 2\alpha) &= 2\alpha(e^{I\theta} + e^{-I\theta}) + (1 - 2\alpha)\xi^{-1} \\ &= 4\alpha cos(\theta) + (1 - 2\alpha)\xi^{-1} \end{aligned}$$

Here, denoting ξ as G, treat it as a variable, then a quadratic function can be found:

$$G(1-2\alpha) = 4\alpha cos(\theta) + (1-2\alpha)\frac{1}{G}$$
$$(1+2\alpha)G^2 - 4\alpha cos(\theta)G - (1-2\alpha) = 0 \tag{13}$$

The solution of this function can be found by:

$$G = \frac{4\alpha cos(\theta) \pm \sqrt{16\alpha^2 cos^2(\theta) + 4(1+2\alpha)(1-2\alpha)}}{2(1+2\alpha)}$$
$$= \frac{2\alpha cos(\theta) \pm \sqrt{1-4\alpha^2 sin^2(\theta)}}{1+2\alpha}$$

G may be real or complex depending if the sign of the square root argument is positive or negative.We will get a negative argument if

$$\Delta := 1 - 4\alpha^2 sin^2(\theta) < 0$$

Stability depends on having$|G| <= 1$. If G is complex, we must have $G^*G <= 1$. Using the equation just derived for G, we obtain the following stability requirement for complex G.

$$G^*G = \frac{4\alpha^2 cos^2(\theta) + 1 - 4\alpha^2 sin^2(\theta)}{(1+2\alpha)^2}$$
$$= \frac{1 + 4\alpha^2[1 - 2sin^2(\theta)]}{(1+2\alpha)^2}$$
$$\leq 1$$

The $cos^2(\theta)$ term ranges from 0 to 1,

$$\text{When } cos^2(\theta) = 1, G^*G \leq 1$$
$$\text{When } cos^2(\theta) = 0, G^*G \leq 1$$

When $\Delta \geq 0$, the growth factor is real. The argument of the square root is one minus a positive number. This positive number must be less than 1 to maintain a real solution. Thus the square root in the real case will always be less than or equal to 1. The limiting condition will be when the square root term is +1. In this case the growth factor inequality becomes:

$$G = \frac{1 + 2\alpha cos(\theta)}{1+2\alpha}$$
$$\leq 1$$

We see that this will be satisfied for any value of α since the cos term is always less than one. Thus, the stability analysis shows that any value of α produces a growth factor whose magnitude is less than or equal to one. We can conclude that this method is unconditionally stable.

3.4.3 Laasonen Implicit Method

After transforming the term in Eqn. 7:

$$\xi^{n+1}e^{I\theta i} = \xi^n e^{I\theta i} + \alpha\xi^{n+1}[e^{I\theta(i-1)} - 2e^{I\theta i} + e^{I\theta(i+1)}]$$

Then,

$$\xi^{n+1} = \xi^n + \alpha\xi^{n+1}(e^{-I\theta i} + e^{I\theta i} - 2)$$
$$= \xi^n + \xi^{n+1}[2\alpha(cos(\theta) - 1)]$$

Therefore,

$$\xi^{n+1}[1 + 2\alpha(1 - cos(\theta))] = \xi^n$$

where

$$1 + 2\alpha(1 - cos(\theta)) \geq 1$$

Thus, for any value of $\alpha > 0$, the condition (11) always holds in this method. This method is unconditionally stable.

3.4.4 Crank-Nicolson Method

After transforming the term in Eqn.8:

$$\xi^{n+1}e^{I\theta i} = \xi^n e^{I\theta i} + \frac{\alpha}{2}[\xi^n(e^{I\theta(i-1)} - 2e^{I\theta i} + e^{I\theta(i+1)}) + \xi^{n+1}(e^{I\theta(i-1)} - 2e^{I\theta i} + e^{I\theta(i+1)})]$$

Therefore,

$$\begin{aligned} \xi^{n+1} &= \xi^n + \frac{\alpha}{2}(\xi^{n+1} + \xi^n)(2cos(\theta) - 2) \\ &= \xi^n + \alpha(\xi^{n+1} + \xi^n)(cos(\theta) - 1) \end{aligned}$$

Since

$$cos(\theta) - 1 \leq 0$$

For any value of $\alpha > 0$, the condition (11) always holds in this method. This method is unconditionally stable.

4 Results

The results from the four numerical methods under different simulation parameters reveal distinctive behaviors. In Case 1, with $\Delta t = 0.002$ and $n_m = 540$, all methods produce stable and consistent velocity profiles, demonstrating their efficacy under fine temporal resolution.

For Case 2, the increased timestep to $\Delta t = 0.0024$ and reduced iterations to $n_m = 450$ lead to FTCS method instability. Under other different value of the parameters, the DuFort-Frankel, Laasonen, and Crank-Nicolson methods remain stable, with the implicit Laasonen and Crank-Nicolson methods confirming their appropriateness for scenarios that require larger timesteps for computational efficiency. The exact value of the parameters can be found in a table [2], it can also be found in the Appendix Table 1.

A further study of Laasonen method based on different timesteps shows that the accuracy of the numerical method could be strongly effected by the value of timesteps. When the timesteps become larger and larger, the error will also become larger and larger in this method. The final distribution graphs will be shown below from Figure 7 to Figure 14.

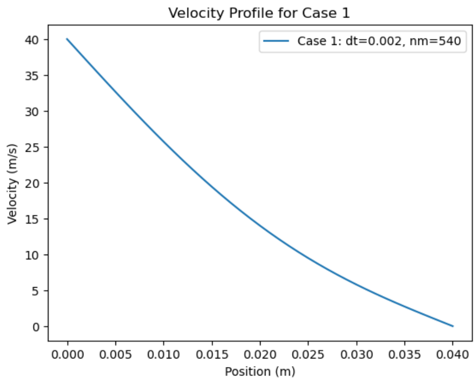

Figure 7: FTCS result Case 1

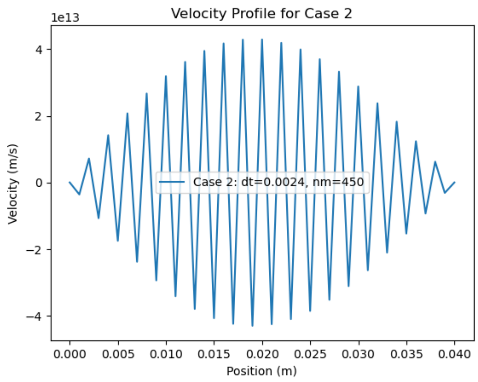

Figure 8: FTCS result Case 2 (Unstable)

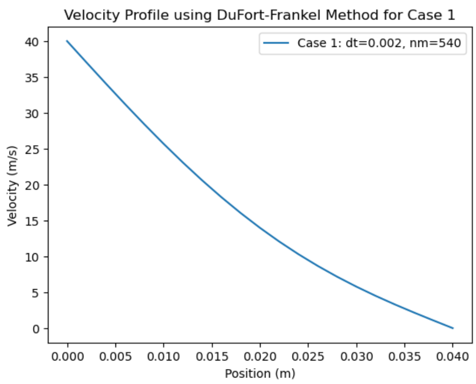

Figure 9: DuFort-Frankel result Case 1

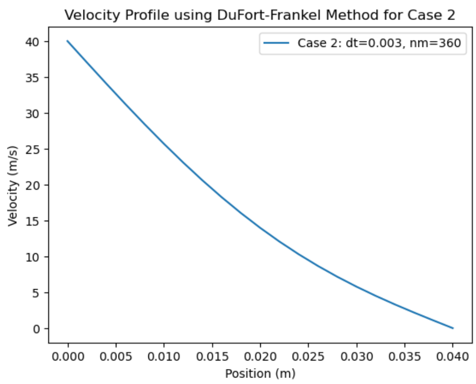

Figure 10: DuFort-Frankel result Case 2

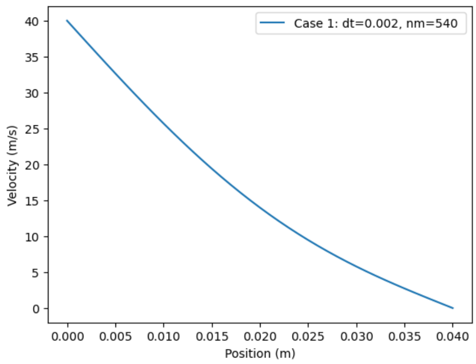

Figure 11: Laasonen result Case 1

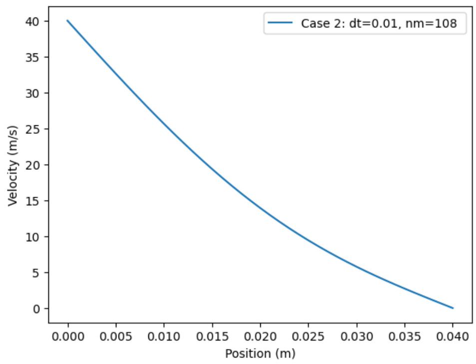

Figure 12: Laasonen result Case 2

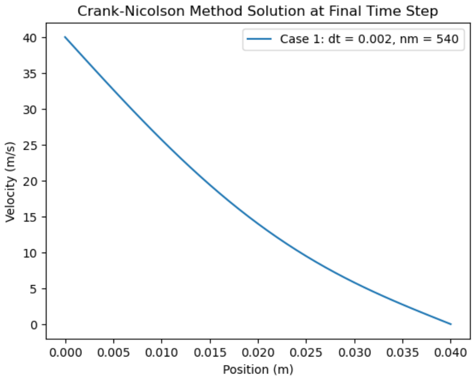

Figure 13: Crank-Nicolson result Case 1

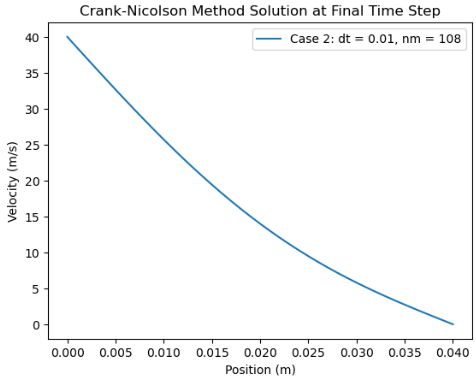

Figure 14: Crank-Nicolson result Case 2

4.1 Errors

As mentioned above, the analytical solution should be given as an accurate enough solution, the form of the analytical solution will be found by solving a complimentary error functions [2]:

$$\begin{aligned} u &= U_0 \left(\sum_{n=0}^{\infty} \text{erfc}\left[2n\eta_1 + \eta\right] - \sum_{n=0}^{\infty} \text{erfc}\left[2(n+1)\eta_1 - \eta\right] \right) \\ &= U_0 \left\{ \text{erfc}(\eta) - \text{erfc}(2\eta_1 - \eta) + \text{erfc}(2\eta_1 + \eta) - \text{erfc}(4\eta_1 - \eta) + \text{erfc}(4\eta_1 + \eta) - \ldots \right\} \end{aligned}$$

where

$$\eta = \frac{y}{2\sqrt{\nu t}}, \quad \eta_1 = \frac{h}{2\sqrt{\nu t}}.$$

Here the value of the n should be determined, analytically speaking, the solution will become more accurate as the value of n become larger. As n trends to infinity, the solution of this function can be seem as a totally accurate solution. In the summation, the contribution of each subsequent term to the sum becomes smaller and smaller. Usually after n reaches a certain value, such as 100, the impact of subsequent terms on the numerical solution is negligible [5]. Therefore, in this report, the value of n would be taken as 100.

The error distribution of the four numerical methods under two different parameter settings when compared to the analytical solution indicates a varying degree of accuracy, where $t = 1.08$ and $t = 0.18$. The FTCS method, under the first parameter set, shows a symmetric error distribution, while the second parameter set reveals fluctuations in error. The DuFort-Frankel method's error plots exhibit more significant fluctuations across both parameter sets, particularly with the second set of parameters. The Laasonen method's error distribution charts display relatively minimal errors, especially under larger timestep sizes. The Crank-Nicolson method offers a middle ground, with smooth and symmetrical error distributions under both parameter sets.

These visualizations reflect that the choice of parameters significantly impacts the accuracy of the numerical solutions, and different numerical methods have varying sensitivities to changes in parameters. The FTCS and DuFort-Frankel methods are more sensitive to increases in timestep size, showing larger error fluctuations. In contrast, the Laasonen and Crank-Nicolson methods demonstrate robustness to timestep variations, maintaining smaller errors even at larger timesteps. The graphs of the error distribution will be listed from Figure 15 to Figure 22.

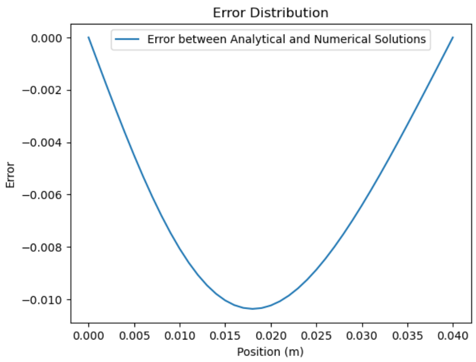

Figure 15: FTCS Error Distribution 1

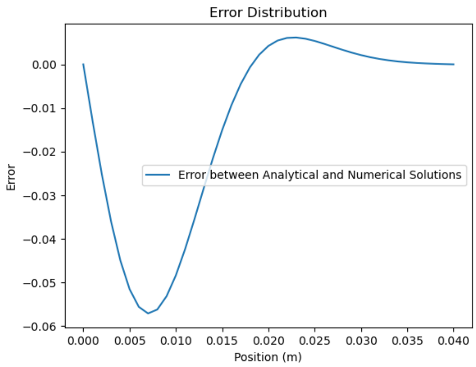

Figure 16: FTCS Error Distribution 2

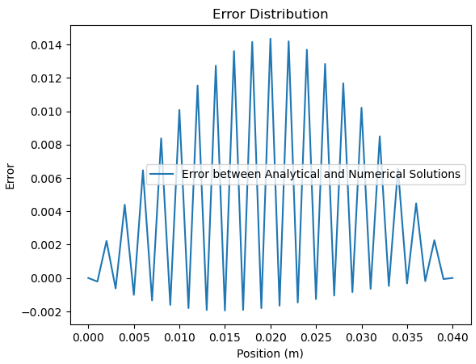

Figure 17: DF Error Distribution 1

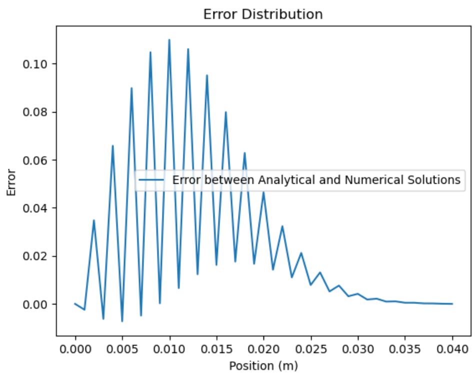

Figure 18: DF Error Distribution 2

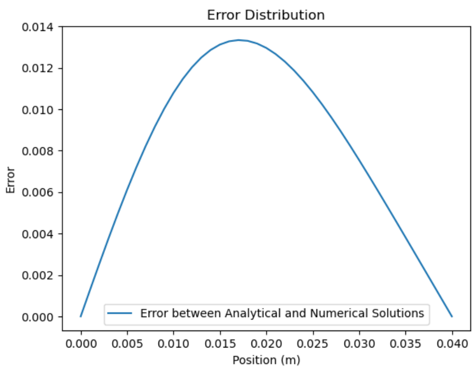

Figure 19: Laasonen Error Distribution 1

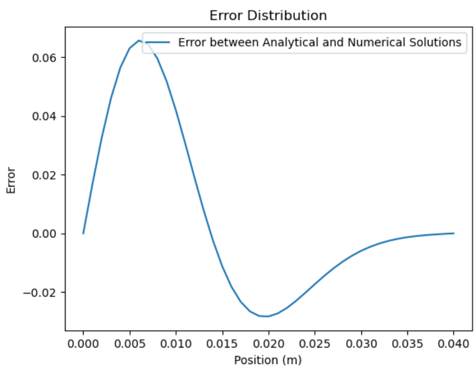

Figure 20: Laasonen Error Distribution 2

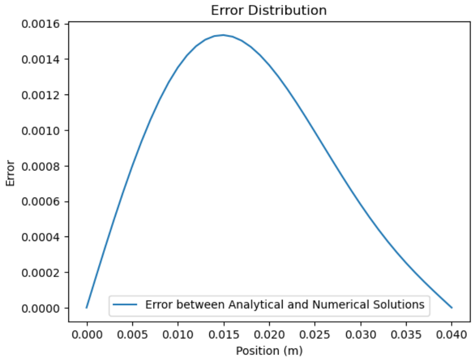

Figure 21: CN Error Distribution 1

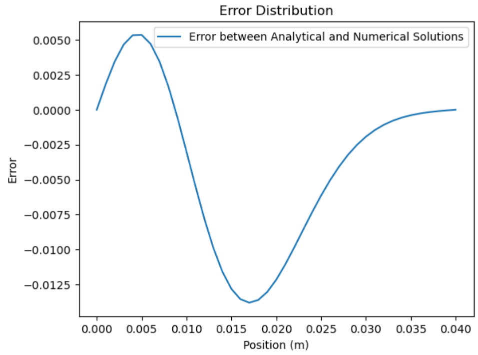

Figure 22: CN Error Distribution 2

4.2 Time step in the Laasonen method

The provided graph illustrates the impact of varying the time step (Δt) on the error magnitude when using the Laasonen implicit method. As Δt increases, there is a clear trend of increasing error.

With the smallest time step ($\Delta t = 0.005$, $n_m = 200$), the error is minimal and the curve is almost flat, indicating a high degree of accuracy in the numerical solution. This suggests that the Laasonen method, with a sufficiently small time step, can provide results that closely approximate the analytical solution. As the time step is increased to $\Delta t = 0.01$ with $n_m = 100$, the error starts to grow, but remains relatively low. This indicates that the method still performs well, but the larger time step begins to introduce more noticeable inaccuracies. Further increasing the time step to $\Delta t = 0.1$ with $n_m = 10$, the error becomes more pronounced, with the error curve peaking higher. This significant increase in error suggests that such a time step is too large to capture the nuances of the problem accurately. Finally, with a very large time step ($\Delta t = 0.2$, $n_m = 5$), the error is substantial, with the curve peaking sharply. This level of error could render the numerical solution unreliable for practical purposes.

In conclusion, the choice of time step is critical in the Laasonen method. Smaller time steps lead to more accurate solutions with lower error, while larger time steps, although computationally less demanding, result in higher errors and can potentially compromise the integrity of the solution. The balance between computational efficiency and accuracy must be carefully managed, especially in problems where precision is crucial.

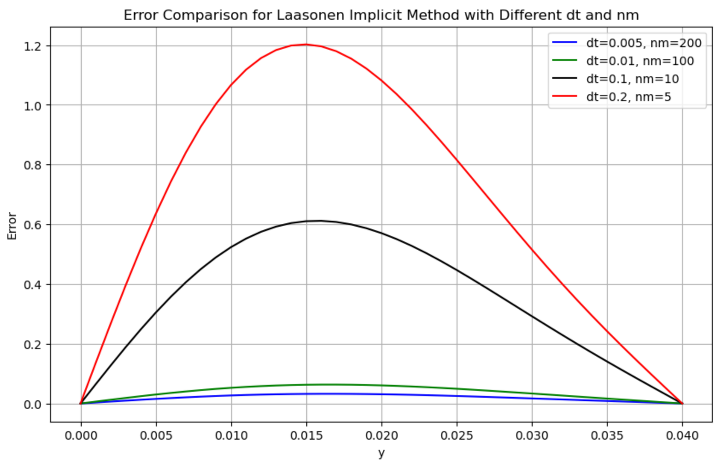

Figure 23: Error distribution under different timesteps

5 Discussion and Conclusion

This study extensively evaluated the performance of four numerical methods—FTCS, DuFort-Frankel, Laasonen, and Crank-Nicolson—in simulating the behavior of viscous fluids between two parallel plates. The FTCS method demonstrated conditional stability, making it suitable for scenarios with small time steps. Conversely, the DuFort-Frankel method, despite its complexity, proved unconditionally stable under various conditions, offering a reliable choice for larger time steps. The Laasonen and Crank-Nicolson methods, both implicit, showcased exceptional stability and accuracy across all tested parameters, making them ideal for long-term simulations where computational efficiency and precision are paramount.

Considering the trade-offs between computational demand and accuracy, the choice of method largely depends on the specific requirements of the study. For research prioritizing accuracy over computational speed, the Laasonen and Crank-Nicolson methods emerge as superior choices due to their robustness and stability. In contrast, for studies where computational efficiency is crucial, the DuFort-Frankel method offers a viable alternative, especially in scenarios that can tolerate the complexities of its implementation.

All four methods are suitable under specific conditions, with the FTCS method requiring careful parameter selection. The other three methods offer more flexibility, allowing for use without significant restrictions. However, computational capacity must always be considered, particularly with unconditionally stable methods where increasing step size can accelerate convergence. Additionally, when addressing complex, multi-dimensional problems, the complexity of implementing implicit methods and the effort required to optimize the code are crucial considerations.

Moreover, the error plays an important role when solving a problem in reality. This report has already shown that the error distributions are different between the four given methods. Especially for the DuFort-Frankel method, it showed a oscillated distribution. These differences between the distributions could be caused by the different computational process of the methods. As the graphs showed above, all the methods leaded to small errors, it can be believed that all the methods we used in this report are accurate enough. There are still improvements for accuracy, for instance, reducing the timesteps [6] or using a more accurate numerical method[7].

In essence, this study highlights the critical role of method selection in computational research, focusing on how the goals of a study should guide the choice between speed and precision. The balance between computational speed and result accuracy is emphasized, suggesting that the decision should align with the specific objectives and constraints of the research at hand.

Appendix

Table 1: Comparison of Numerical Method Parameters for Different Test Cases

Method	Δt (s)	n_m
FTCS explicit method		
Case 1	0.002	540
Case 2	0.0024	450
DuFort-Frankel explicit method		
Case 1	0.002	540
Case 2	0.003	360
Laasonen implicit method		
Case 1	0.002	540
Case 2	0.01	108
Crank-Nicolson implicit method		
Case 1	0.002	540
Case 2	0.01	108

References

[1] Wikipedia. Heat equation. https://en.wikipedia.org/wiki/Heat$_equationf, none.[no]$.

[2] University of St.Andrew MT5846: Advanced Computational Techniques. ACT project 1. https://moody.st-andrews.ac.uk/moodle/pluginfile.php/1636229/mod$_resource/content/6/Project1.pdf, 2024.[no$

[3] University of St.Andrew MT5846: Advanced Computational Techniques. ACT lecture notes 7. https://moody.st-andrews.ac.uk/moodle/pluginfile.php/1636191/mod$_resource/content/7/Lecture[no]$.

[4] University of St.Andrew MT5846: Advanced Computational Techniques. ACT lecture notes. https://moody.st-andrews.ac.uk/moodle/course/view.php?id=17008, 2024. [no].

[5] JimYang. Explain why n can be taken as 100. https://github.com/JimYoung0/A-help-file-for-the-report, 2024. [no].

[6] MR Todd, PM O'dell, and GJ Hirasaki. Methods for increased accuracy in numerical reservoir simulators. *Society of Petroleum Engineers Journal*, 12(06):515–530, 1972.

[7] Martin Kronbichler, Timo Heister, and Wolfgang Bangerth. High accuracy mantle convection simulation through modern numerical methods. *Geophysical Journal International*, 191(1):12–29, 2012.