JOURNAL OF MECHANICAL ENGINEERING AND SCIENCES
ISSN: 2289-4659 e-ISSN: 2231-8380
VOLUME 19, ISSUE 2, 2025, 10594 - 10603
DOI: https://doi.org/10.15282/jmes.19.2.2025.2.0830

RESEARCH ARTICLE

Spectral-based numerical solution method of the incompressible Navier-Stokes equation in a stretched coordinate system

K. A. Khalid[1], K. Osman[2], K. M. Isa[3], A. M. M. Ismail[3], N. Ahmad[3], A. A. M. Yusof[3]*

[1] Vestigo Petroleum Sdn Bhd, 50450 Kuala Lumpur, Malaysia
[2] Faculty of Mechanical Engineering, Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia
[3] School of Mechanical Engineering, College of Engineering, Universiti Teknologi MARA, UiTM, Johor Branch, Masai, Malaysia

ABSTRACT - The numerical solution of the Navier-Stokes equation is widely used in computational fluid dynamics to address engineering problems related to fluids. The method can deal with nonlinear coupling between velocity and pressure fields. In this study, a robust numerical approach integrates the pseudospectral method with a splitting algorithm and a pressure correction technique to solve the two-dimensional incompressible Navier-Stokes equation effectively. This study aims to confirm the accuracy of the suggested method by comparing its results with benchmark two-dimensional solutions and to assess the computational efficiency of the developed algorithm. The pseudospectral method is typically applied to simpler problems involving Ordinary Differential Equations and Partial Differential Equations. Spatial discretization was performed using cardinal Chebyshev basic functions, which offer spectral accuracy and apply to non-periodic domains. The problem was divided into smaller, more manageable pseudospectral grids using a splitting algorithm and combined with a nonlinear term formulation. The combination technique allows for the decoupling of velocity and pressure computations. The numerical scheme was implemented in MATLAB and evaluated for Reynolds numbers (Re) of 100, 400, and 1000. The simulation results indicate a strong correlation between the predicted velocity profile and vortex formation, aligning closely with established benchmark data. The percentage differences of the compared data were 0.02% for U-Velocity and 2.11% for V-Velocity at Re = 100, respectively. The predicted flow pattern and centre location of the vortices matched closely with the reference value, confirming the method's accuracy. Additionally, the approach exhibited rapid convergence and computational efficiency. In conclusion, the proposed technique successfully demonstrates the capability to solve the incompressible flow problem and provides accurate results up to a Reynolds number of 1000. A stability analysis described in this study indicates that the proposed method remains applicable at higher Reynolds numbers.

ARTICLE HISTORY

Received	:	13th Nov. 2024
Revised	:	13th May 2025
Accepted	:	09th June 2025
Published	:	30th June 2025

KEYWORDS

Incompressible Navier-Stokes
Splitting method
Pressure correction approach
Pseudospectral gridding method

1. INTRODUCTION

The initial derivations of the Navier–Stokes (N-S) equation were presented in two memoirs by Claude-Louis Navier between 1822 and 1836 [1]. The N–S equation, foundational to fluid dynamics, governs the behavior of liquids and gases and underpins diverse applications in physics, engineering, and biomedical sciences [2-4]. Resolving the N-S equation requires various essential techniques, especially for incompressible flows. Integrating splitting and pseudospectral methods for solving the incompressible N-S equation is an accepted approach in computational fluid dynamics. Splitting methods originated in the 1960s [5]. Janenko [6] further advanced the technique, which was noted for its efficiency in decoupling the pressure and velocity components of the N-S equation. Splitting is a time-marching strategy that substitutes simultaneous processes with sequential stages to enhance efficiency and simplify complex equations [7, 8]. The decoupling facilitates sequential resolution of the equations, thereby decreasing computational complexity. Initial developments, exemplified by Roger Temam's research [9], utilized splitting methods for the N-S equation but achieved only first-order accuracy, necessitating subsequent enhancements.

The momentum equation can be applied to both linear and nonlinear applications. The linear part is solved with a linear operator, while the nonlinear part is treated independently. Splitting methods are classified into three categories: pressure correction schemes, velocity correction schemes, and consistent splitting schemes. Researchers, such as J. van Kan [10], have utilized advanced pressure correction schemes derived from Chorin's work [5], with a focus on enhancing pressure updates and improving time-stepping accuracy [10]. Meanwhile, velocity correction schemes were developed by Karniadakis et al. [11]. The method achieved second-order accuracy and was effectively implemented in benchmark problems, such as channel and cavity flows. Consistent splitting schemes preserve accuracy and stability by ensuring the alignment of pressure and velocity fields during computations. Osman [12] utilised the velocity correction scheme to investigate bifurcations in three-dimensional lid-driven cavities. In contrast, Ngali [13] employed the same approach for

***CORRESPONDING AUTHOR | A. A. M. Yusof |** abaziz86@uitm.edu.my

two-dimensional cavity flows, attaining precise results at low Reynolds numbers yet facing difficulties at elevated Reynolds numbers.

Several studies have been reported to clarify the adaptability of splitting methods while emphasising the constraints, particularly in the context of high Reynolds number flows. A thin boundary layer must be depicted to properly exhibit the flow transition from laminar to turbulent at high Reynolds numbers [14]. Additionally, pseudospectral methods are discussed, particularly their application in solving partial differential equations (PDEs). For periodic boundary conditions, including the N-S Stokes equations in fluid dynamics, pseudospectral methods offer viable numerical approaches for PDEs. These methods require minimal computational resources to solve the nonlinear Schrödinger equation (NLSE) [15]. According to their simplicity and optimisations, the NLSE within a conservative framework can clarify various dynamics, including modulation instability and fundamental and higher-order solitons [16]. Furthermore, pseudospectral methods offer high accuracy with fewer grid points by employing basis functions, such as Chebyshev polynomials, for solution approximation [17]. Benchmark studies, including the work of Botella and Peyret on the 2D-driven cavity problem, show the efficacy of pseudospectral methods [18]. Hence, the present study utilised the pseudospectral approach in a reduced and sequential N-S equation. Splitting and pseudospectral methods were integrated to solve a reduced and sequential version of the N–S equations. This research presents a novel approach that strikes a balance between computational efficiency and high accuracy. This contrasts with traditional methods that may excel in one aspect but not in both. The integrated approach is particularly advantageous for simulations that require precise results without incurring prohibitive computational costs.

Thus, the study was conducted with two main objectives. The first objective was to evaluate the accuracy of the solution by comparing it with benchmark studies based on two-dimensional approaches. The second objective was to assess the computational efficiency of the developed algorithm. By decomposing the problem into smaller components, the approach seeks to minimise errors associated with time discretisation and enhance the stability and accuracy of simulations, particularly under high Reynolds number flow conditions.

2. MATERIALS AND METHODS

2.1 Solution Strategy

The pseudo-spectral method was applied to simpler Ordinary Differential Equations (ODEs) and Partial Differential Equations (PDEs), including the Helmholtz and Poisson equations. These tests involved evaluating different gridding approaches using Chebyshev polynomial bases. The cardinal Chebyshev basis was straightforward and easier to implement, yielding good numerical results [19]. In this study, the FORTRAN code developed by Osman et al. was adapted [12] and subsequently translated into MATLAB code for further analysis. Furthermore, an effort was made to refine the pressure boundary condition according to the formulation by Karniadakis et al. [11] to augment accuracy and stability. However, the modifications offered minor improvements. The study then implemented the splitting algorithm proposed by Guermond and Quartapelle [20] . This method was adapted to the pseudospectral grid in the earlier stages. An additional enhancement involved modifying the nonlinear component of the incompressible Navier-Stokes equations, following Karniadakis et al. [11], which was also included. An extensive evaluation was conducted, focusing on diffusive stability criteria and Courant–Friedrichs–Lewy (CFL) conditions at low resolutions across various Reynolds numbers, specifically Re = 100, 400, and 1000. The results were compared to a well-established 2D lid-driven cavity problem, which showed excellent agreement with the data provided by Ghia et al. [21] and Botella and Peyret [18]. This work presents the unique integration of Guermond and the Quartapelle splitting algorithm with modified nonlinear terms from Karniadakis et al., which utilises the cardinal Chebyshev basis for spatial discretisation, distinguishing it from previous approaches in the field.

2.2 Splitting Method via Pressure Correction

A schematic diagram for the boundary conditions for the incompressible Navier-Stokes equation in lid-driven cavity flow is shown in Figure 1. Despite its simple geometry, this problem captures the complexity of fluid physics flow, exhibiting multiple counter-rotating, recirculating regions that appear in the center and corners of the cavity, depending on the Reynolds number of the flow. The setup served as an ideal model for benchmarking the numerical algorithm developed in the study. In the simulation, the V-velocity was set to zero through the domain, while the U-velocity was set to unity on the lid and zero through the setup boundaries. The result provided the flow of the incompressible fluid within a square cavity driven by the lid moving at a constant speed.

The study was conducted using various mesh grid elements, including 17×17 (289 cells), 21×21 (441 cells), 31×31 (961 cells), and 65×65 (4225 cells). For cases involving a Reynolds number of 1000, the mesh grid element ranged from 9×9 to 65×65 to compare the peak velocity value and numerical efficiency for a single time step. The number grid can be increased to improve numerical accuracy, but this comes at the cost of computational efficiency. In particular, the time required to solve the result increases exponentially with increasing N+1 grid elements. In this study, 1^{st-} and 2nd-order accuracy schemes were applied to the mesh grid to improve solution accuracy and investigate the influence of the discretization order on the results.

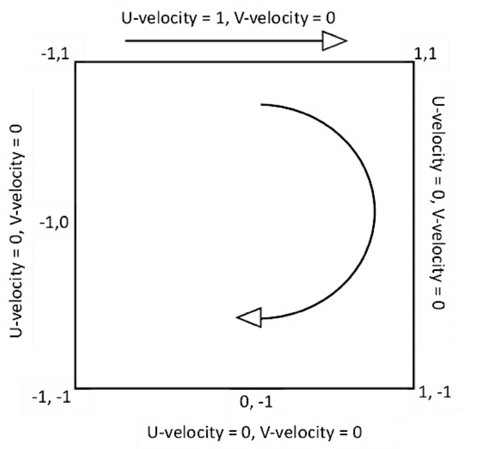

Figure 1. 2D schematic diagram of the lid-driven cavity problem

This study assumes that the flow is two-dimensional, incompressible (constant density and viscosity are applied), exhibits Newtonian fluid behavior, is unsteady, and has no heat transfer and body force effects, as shown in Eq. (1).

$$\frac{\partial \bar{v}}{\partial t} = -\nabla \bar{p} + \frac{1}{Re}\nabla^2 \bar{v} - \bar{v}.\nabla \bar{v} \tag{1}$$

where, $\bar{v}$ is the flow velocity vector, $\bar{p}$ is the pressure field, Re is the Reynolds number defined as $Re = \frac{Ul}{v}$, and U = speed of the wall, l = length of the wall, and v = dynamic viscosity of the fluid inside the cavity.

An equivalent form of the nonlinear terms was used by Karniadakis et al. [5] (Eq. (2)) and adopted by the current work instead of Eq. (1).

$$N(\bar{v}) = \bar{v}.\nabla \bar{v} = \frac{1}{2}[(\bar{v}.\nabla \bar{v}) + (\nabla \bar{v}.\overline{v)}] \tag{2}$$

Applying a temporal integration over a time step, Eq. (3) was obtained as follows:

$$\int_{t_n}^{t_{n+1}} \frac{\partial \bar{v}}{\partial t} dt = -\int_{t_n}^{t_{n+1}} (\nabla \overline{p)} dt - \int_{t_n}^{t_{n+1}} (N(\bar{v}))dt + \int_{t_n}^{t_{n+1}} (L(\bar{v}))dt \tag{3}$$

For all splitting methods, the schemes used to evaluate and split the linear, nonlinear, and pressure terms, as well as the order in which they are performed, differentiate one method from another. The first substep involves solving for an intermediate velocity by extrapolating or "projecting" pressure and velocity from the current and previous values for the next time step. The next substep involved correcting for the pressure. Again, from Eq. (3), the pressure term was solved from the previously solved velocities. After pressure at the next time step, p^{k+1} was solved from Eq. (4), and substep one was repeated for the next step.

A combined explicit-implicit scheme was used for temporal integration. In this study, the second-order Backwards Differentiation Formula (BDF2) was utilised. Let *k be* denoted as the time step index, where $k \geq 0$. The velocity at the current time step was represented as v^k, with v^{k-1} and v^{k+1} denoting the velocities at the previous and next time step, respectively.

$$\frac{\partial^2 (p^{k+1} - p^k)}{\partial x^2} + \frac{\partial^2 (p^{k+1} - p^k)}{\partial y^2} = \frac{3}{2\Delta t}\left(\frac{\partial U^{k+1}}{\partial x} + \frac{\partial V^{k+1}}{\partial y}\right) \tag{4}$$

$$AV = \left[\frac{3I}{2} - \frac{\Delta t}{Re}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\right] V^{k+1} \tag{5}$$

$$A_p P = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)(p^{k+1} - p^k) \tag{6}$$

$$b_p = \frac{3}{2\Delta t}\left(\frac{\partial U^{k+1}}{\partial x} + \frac{\partial V^{k+1}}{\partial y}\right) \tag{7}$$

The boundary conditions, as in Figure 1 for the left and right walls of the driven cavity (Eq. (8)), were:

$$\frac{\partial (p^{k+1} - p^k)}{\partial x} = 0 \tag{8}$$

The boundary conditions for the top lid and bottom wall of the driven cavity (Eq. (9)) were:

$$\frac{\partial(p^{k+1}-p^{k})}{\partial y}=0 \tag{9}$$

For each time step, the forms of the left-hand side of the matrix (Eqs. (4), (5), and (6)) remain constant and can be declared early in the algorithm.

2.3 Algorithm and Coding using MATLAB

Numerical methods for the incompressible Navier-Stokes equation are validated against the unsteady two-dimensional (2-D) driven cavity flow. This study utilizes this problem, which retains the complexity of fluid physics, as a test bed for the algorithm, assuming two-dimensional, incompressible, Newtonian fluid behavior and no heat transfer. MATLAB was selected for its advantages in rapid prototyping and debugging. It enhances work efficiency by utilising built-in functions that handle overdetermined and ill-conditioned matrices, which are standard in pseudospectral implementations. These functions leverage LAPACK libraries and utilize direct back-substitution or linear iterative solvers, depending on the matrix size, thereby reducing the need for custom subroutines for operations such as matrix multiplication, vector conversion, and transposition.

To create the coordinate system for the driven cavity problem, Chebyshev grid points were generated in both the x- and y-directions. MATLAB's built-in meshgrid function was employed to generate arrays of coordinates, and the colon operator was used to vectorize these arrays, where N refers to the grid element. The MATLAB code implementation would follow this structured approach as follows.

```
x = cos(pi*(0:N)/N); y = x; [gridx,gridy]= meshgrid(x,y);
xx = gridx(:);yy = gridy(:)
```

The identity matrix was used to represent the discretized forms of the velocity in the x-direction, U, the velocity in the y-direction, V, and pressure, P. MATLAB's built-in 'eye' function can be used to generate the identity matrix. In the example Matlab code below, the variable I2 was used to represent U, V, and P, respectively.

```
I = eye(N+1); I2 = eye((N+1) ^2)
```

The cardinal Chebyshev derivative matrix is created using a well-known MATLAB function, "cheb.m" by Trefethen [22]. Other researchers, such as Johnston and Liu [19], have documented the use of this concise function in their respective research.

```
function [D,x] = cheb(N)
if N == 0,D = 0; x=1;
x = cos(pi*(0:N)/N)';
c = [2; ones(N-1,1); 2].*(-1).^(0:N)';
X = repmat(x,1,N+1);
dX = X-X';
D=(c*(1./c)^' )./(dX+(eye(N+1)))
D=D-diag(sum(D^' ))
```

The following example shows how this can be implemented in MATLAB. Higher derivatives that are needed are generated by taking the power of the derivative matrix. Note that the derivative matrices for the x and y directions are the same.

```
Dx = cheb(N);dxx=dx^2;dy=dx;dyy=dxx;
```

The MATLAB built-in function "kron" is used to generate the partial derivative matrices for evaluating the Kronecker products of the cardinal Chebyshev basis and its derivatives. The actual example code used in MATLAB is provided below. Dx, Dxx, Dy, and Dyy refer to the first partial derivative matrix concerning x, the second partial derivative concerning x, the first partial derivative concerning y, and the second partial derivative concerning y, respectively.

```
Dx = kron(dx,I);Dxx=kron(dxx,I);Dy=kron(I,dx);Dyy=kron(I,dxx);
```

Matrix multiplication between the generated partial derivatives and the velocity or pressure vector was used to take the derivative of a velocity or pressure value vector. To enforce boundary conditions within the algorithm, MATLAB's built-in "find" function is also used extensively to identify the row index required to be replaced with the boundary condition expression. Next, for the discretised form of the splitting equations, the left-hand side matrix A of the velocity equations can be represented in discretised form using the matrices described next. Rows that correspond to the grid points at the wall and lid are also replaced with the identity matrix to represent the Dirichlet boundary conditions.

```
a = 1.5*I2-(dt/Re).*(Dyy+Dxx);
k = find(abs(yy)==1|abs(xx)==1);a(k,:)=I2(k,:);
```

Similarly, the left-hand side matrix A of the pressure equation is discretised below. Rows that correspond to the Neumann boundary condition in Eq. (8) and Eq. (9) are replaced.

```
aPressure = Dxx+Dyy;
k = find(yy = = 1); aPressure(k,:) = Dy(k,:);
k = find(yy = = -1); aPressure(k,:) = Dy(k,:);
k=find(abs(xx)==1);aPressure(k,:)=Dx(k,:);
```

The right-hand side of the U velocity in the x direction was evaluated in this study. Suppose let P, P1, and P0 be vectors of pressure values at time step t. In that case, t-1 and t-2 are the respective time steps, U and U0 represent the U velocity at time steps k and k-1, Us is the intermediate U velocity, and NLx represents the nonlinear terms. The right-hand side of the U velocity equation is evaluated in MATLAB as follows.

```
gradPx = (Dx*(7*P-5*P1+P0))/3;
Us = 2*U-U0;
NLx = -0.5*(Us.*(Dx*Us)+ Vs.*(Dy*Us)+Dx*(Us.*Us)*(Us.*Vs));
bU=(NLx-gradPx)*dt+2*U-0.5*U0;
```

Dirichlet boundary conditions are enforced at the walls and the lid of the driven cavity by replacing the rows corresponding to the respective grid points.

```
k1 = find(abs(xx)==1);
k2 = find(yy==1 & abs(xx)≅1);
k3 = find( ==-1);
bU(k1)= 0;
bU (k2)= 1;
bU(k3) = 0;
```

Similarly, for the V velocity in the y direction, let P, P1, and P0 vector of pressure values at time step t, t-1 and t-2, respectively, V and V0 as the V velocity at time step t and t-1 respectively, Vs as the intermediate U velocity and NLy as the nonlinear terms. It can then be represented in a discretized form using the following MATLAB code.

```
gradPy = (Dy*(7*P-5*P1+P0))/3;
Vs = 2*V - V0;
NLy = -0.5*(Us.*(Dx*Vs) + Vs.*(Dy*Vs) + Dx*(Us.*Vs) + Dy(Vs*Vs));
bV = (NLy - gradPy)*dt + 2*V - 0.5*V0;
```

Dirichlet boundary conditions are enforced at the walls and the lid of the driven cavity by replacing the rows corresponding to the respective grid points.

```
k4 = find(abs(yy)= =1 | abs(xx)= =1);
bV(k4) = 0;
```

The right-hand side of the pressure Equation (7) can be evaluated in the same manner as previously described. Let U and V variables be the solved velocities at the next step, t+1. Thus, the right-hand side can be written as.

```
bPressure = (Dx*U + Dy*V)/dt;
```

Neumann boundary conditions are enforced at the walls and the lid of the driven cavity by replacing the rows corresponding to the respective grid points.

```
K = find(yy==-1);
bPressure(k) = 0;
k = find(yy==1);
bPressure(k) = 0;
k = find(xx==-1);
bPressure(k) = 0;
k = find(xx==1);
bPressure(k) = 0;
```

3. RESULTS AND DISCUSSION

Figure 2(a) shows the horizontal velocity u at the centre vertical line, and Figure 2(b) shows the vertical velocity v at the centre horizontal line, respectively. Reynolds's number of 100 was used to compare with the results from Ghia et al. [21]. The present study's results align very well with the benchmark, with a 0.02% difference for the U-velocity and a 2.1% difference in V-Velocity at the centerline for Re = 100, indicating a high level of accuracy in the present simulation method.

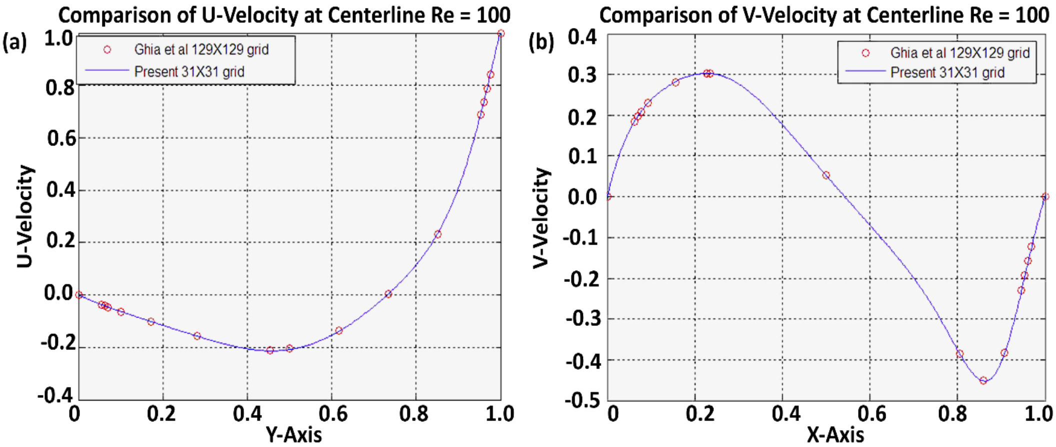

Figure 2. (a) Comparison of U-Velocity along the centre vertical line and (b) V-Velocity along the centre horizontal line for Re = 100

Figure 3 illustrates the steady-state flow pattern in the lid-driven cavity for increasing Reynolds numbers. As the Reynolds number increases, the primary vortex is shifted closer to the centre of the cavity, as in Figure 3(a). At the Reynolds number of 100, the secondary vortices begin to form in the corner, indicating the onset of a complex flow pattern. The vortices continue to develop as the Reynolds number increases. The vortex in the bottom-right corner was more prominent than the one in the bottom-left for all three cases.

Spectral methods achieved exponential convergence rates for smooth problems, meaning that the error decreases exponentially with an increase in the number of basis functions or grid points. Compared to finite difference or finite element methods, the method allows for high accuracy with relatively few grid points. The number of grid elements can be increased at the cost of computational efficiency to increase accuracy. Often, for the spectral method, the time required to solve the full banded matrices increases exponentially with increasing N+1 grid elements for the numerical analysis. Additionally, increasing the number of grid elements results in larger system matrices, which are often dense and can be ill-conditioned. Solving these systems requires more computational resources, and the time complexity can grow rapidly with N. In summary, while spectral methods offer superior accuracy for smooth problems, their computational cost increases significantly with the number of grid elements.

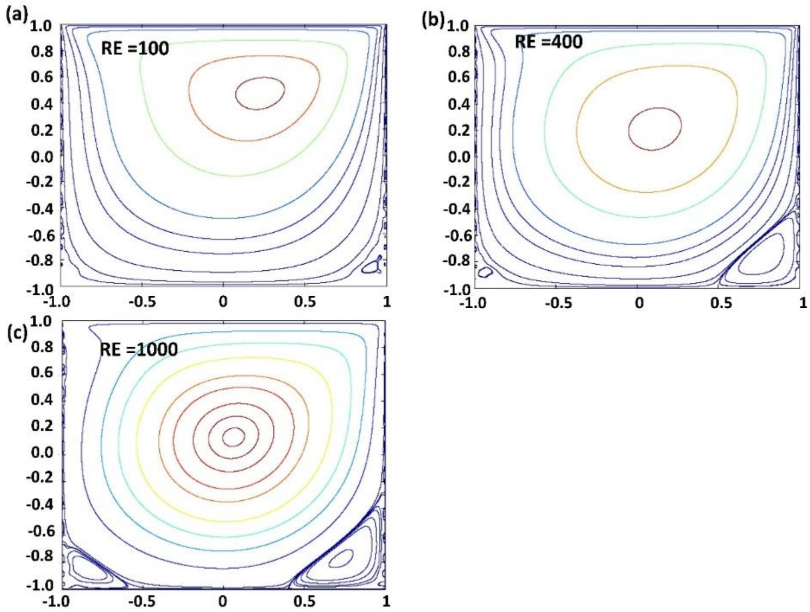

Figure 3. Streamlines of the driven cavity flow at: (a) Re = 100 using 31 × 31 grid elements, (b) Re = 400 using 31 × 31 grid elements, and (c) Re = 1000 using 31×31 grid elements

Table 1 compares the peak velocities at different resolutions for a Reynolds number of 100. The results show that even with the coarse 17 ×17 grid elements, reasonable 1st-order accuracy can be obtained. This indicates that coarse grids can still capture essential flow characteristics effectively for certain flow conditions, such as laminar or low-Reynolds-number flows. Increasing the grid element naturally improves the accuracy of the solution, and at 31×31 elements, a 2nd-order accuracy was achieved. This demonstrates the expected improvement in solution precision with finer discretisation. Doubling the grid elements to 65 × 65 improves the solution slightly. The results in Table 1 suggest that conducting grid convergence studies is essential to determine the optimal grid resolution that achieves sufficient accuracy without incurring unnecessary computational expense. Flow turbulence is expected to increase the cavity for higher Reynolds numbers, with secondary vortices playing a more important role in the cavity's flow behaviour. Hence, to capture these effects, finer grid resolutions are necessary.

Table 1. Extrema of the velocities through the centerline of the cavity at Re = 100

Ref.	Grid Element	u_{max}	y_{max}	v_{max}	x_{max}	v_{min}	x_{min}
Present	17 × 17	-0.21227	0.4585	0.17887	0.2387	-0.25019	0.8161
	21 × 21	-0.21246	0.4567	0.17823	0.2417	-0.25301	0.8113
	31 × 31	-0.21357	0.4563	0.17923	0.2403	-0.25363	0.8108
	65 × 65	-0.21397	0.4585	0.17947	0.2375	-0.25369	0.8104
Ghia et al. [21]	129 × 129	-0.21090	0.4531	0.17860	0.2344	-0.2521	0.8125
Botella [18]	97 × 97	-0.21404	0.4581	0.17957	0.2370	-0.25380	0.8104

Table 2 compares the peak velocities at different resolutions for a Reynolds number of 1000. Here, it can be seen that accuracy is affected in the coarse 17 × 17 and 21 × 21 grid elements. Nevertheless, increasing the grid element naturally improves the accuracy of the solution, and after 31 × 31 grid elements, a high accuracy for the velocity was achieved.

Table 2. Peak velocities at the centerline of the cavity for Re = 1000

Ref.	Grid Element	u_{max}	y_{max}	v_{max}	x_{max}	v_{min}	x_{min}
Present	17 × 17	-0.33433	0.1940	0.31660	0.1887	-0.45115	0.9001
	21 × 21	-0.37129	0.1798	0.36051	0.1649	-0.49989	0.9066
	31 × 31	-0.38553	0.1732	0.37406	0.1604	-0.52286	0.9090
Ghia et al. [21]	129 × 129	-0.38289	0.1719	0.37095	0.1563	-0.51550	0.9063
Botella [18]	97 × 97	-0.38857	0.1717	0.37694	0.1578	-0.52707	0.9092

Numerical efficiency is measured in terms of the time required to compute a solution for a single time step, k. Table 3 lists the computational time required to compute a single time step for any given number of grid elements. The trade-off for achieving higher accuracy is minimal for grid elements below 39 × 39. However, beyond that, the computational time required becomes quite expensive.

Table 3. Time required to compute a single time step for N+1 grid elements

Grid elements used, N+1	Time measured per time step, k (s)
9 × 9	0.01
11 × 11	0.01
17 × 17	0.05
21 × 12	0.14
31 × 31	0.60
39 × 39	4.50
51 × 51	22.0
55 × 55	40.0
61 × 61	66.0
65 × 65	87.0

Table 4 presents the time required for the study to reach a steady-state solution, which varies with different Reynolds numbers. While solving the flow on a 31×31 grid is relatively quick and manageable, increasing the grid resolution significantly raises computational demands. Simulations on finer grids can become time-consuming and may exceed the capabilities of standard workstations. This underscores the importance of balancing grid resolution with available computational resources when conducting fluid dynamics simulations. Most computational time for each time step (approximately 75% of the total) is dedicated to solving the implicit velocity and pressure equations. Spectral problem matrices are typically ill-conditioned from the outset, with singularities occurring at the bottom corners of the driven cavity, resulting in an overdetermined system of linear equations.

Table 4. The estimated computational time required for a steady-state solution

Reynolds Number	Simulation Timing Required to Achieve Steady State (s)	Time Required to Obtain a Steady State Solution t = 0.1s		
		17 × 17	31 × 31	65 × 65
100	35	3 minutes	35 minutes	3.5 days
400	80	7 minutes	1.3 hours	8 days
1000	160	13 minutes	2.6 hours	16 days

Several attempts were made to achieve a steady-state solution using coarse grids, ranging from 17 × 17 to 31 × 31 elements, at higher Reynolds numbers above 2000. However, these attempts were unsuccessful, even with minimal time steps of 0.0001 s. This led to further investigation into related studies on the stability of splitting algorithms. A particularly insightful study by Kress and Lotstedt (2006) [23] provides a detailed mathematical analysis of a velocity correction variant. It examines the impact of various time-stepping schemes, including the Adams-Bashforth/Moulton family, the Euler method, and the BDF method. Their analysis demonstrated that the stability and accuracy of a solution can be predicted. Specifically, the CFL condition, $\xi = \Delta s/\Delta t$, can be correlated with the diffusive stability constraint, $\theta = Re(\Delta s)^2$, to identify regions of stability and accuracy. This allows for estimating the necessary time step and grid size at a given Reynolds number to ensure a stable and accurate solution.

Figure 4 presents the stability region, which is divided into two zones: an 'unstable region' represented by the top shaded area in red and a 'stable region' shown as the bottom shaded area in blue. A reference line indicating second-order accuracy was drawn based on the selected test cases and the highlighted condition required for the accurate solution. The yellow stars on the chart indicate that, for a Reynolds number of 400, a time step of 0.01 s with a grid size of 31 × 31 is practical. It is worth noting that increasing the Reynolds number while keeping the grid size and time step constant shifts a point on the chart to the right, moving it into the high-accuracy region. Thus, with a 31 × 31 grid and a 0.01-time step, calculations for Re < 400 are expected to achieve second-order accuracy.

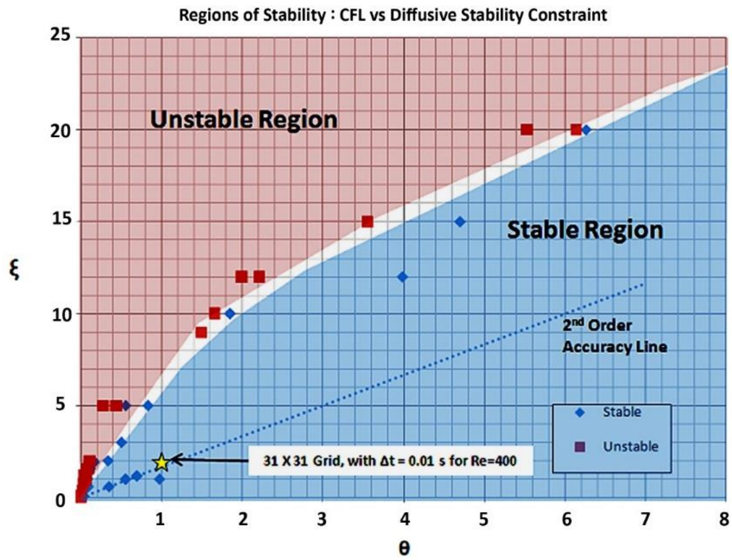

Figure 4. Regions of stability for the present algorithm

The results from the current algorithm were compared with those from a velocity correction splitting method using Adams-Bashforth and Crank-Nicholson schemes, as well as pseudospectral discretization. According to Kress and Lotstedt [23], Adams-Bashforth schemes may have more restrictive stability regions than BDF schemes. To test this, the velocity correction method was examined to determine whether it can produce stable solutions at low Reynolds numbers and with more significant time steps compared to the current algorithm in this study. The result also highlighted the robustness of the current algorithm, which employs the BDF scheme. A low-resolution grid should yield highly accurate results based on the CFL and diffusive stability constraint at low Reynolds numbers.

The current study establishes a robust foundation for solving the incompressible Navier-Stokes equation by integrating a spectral-based numerical method. Future work could investigate extending the current approach to three-dimensional (3D) applications. In the case of a 3D model, the complexity of the velocity field and dynamic pressure increases significantly. The pseudospectral method, combined with the splitting algorithm and pressure correction, holds the efficiency in solving such higher-dimensional problems. In addition, the 3D approach can be adapted for turbulence flow by integrating a turbulence model, such as Large-Eddy Simulation or Direct Numerical Simulation, which provides a more realistic and complex representation of flow phenomena. To further leverage the applied method, implementing the current process on adaptive mesh refinement or hybrid CPU-GPU is also a good approach to enhance resolution in the localized flow region. The suggested approach is to ensure that it is both relevant and competitive in addressing the challenge in fluid dynamics analysis.

4. CONCLUSIONS

The algorithm's performance was validated against the benchmark results of Ghia et al. for a Reynolds number (Re) of 100, demonstrating second-order spatial accuracy and excellent agreement with established data. Notably, the method effectively captured complex flow phenomena, including the formation of primary and secondary vortices within the cavity. A significant contribution of this study is the development of a stability prediction method, which assesses the algorithm's robustness across varying Reynolds numbers. While the algorithm remains stable and accurate at higher Reynolds numbers, it does face increased computational demands on finer grids due to the intensive calculations required for solving implicit equations. Nonetheless, the method's ability to maintain stability and accuracy under these conditions underscores its potential for simulating complex incompressible flows. The algorithm's robust framework and demonstrated capabilities position it as a strong candidate for further development and application across a range of complex fluid dynamics problems. Future research could focus on extending its applicability to three-dimensional and turbulent flows, integrating it with hybrid and machine learning techniques, and adapting it for use in multiphysics simulations. In summary, the integration of splitting methods, advanced pressure correction schemes, and pseudospectral gridding positions this algorithm as a promising tool for accurately and efficiently solving the incompressible Navier-Stokes equations in two-dimensional lid-driven cavity flows.

ACKNOWLEDGEMENT

The authors would like to sincerely thank all individuals who have contributed directly or indirectly to the study's success, especially members from UTM and UiTM. We are also grateful to Mr Zamani for his assistance and valuable insights into the splitting algorithm, which greatly enhanced the technical aspect of the project. The support and collaboration from colleagues and the lab members are also deeply appreciated. This research is supported by UTM Fundamental Research Grant No. 21H62.

CONFLICT OF INTEREST

The authors declare that they have no conflicts of interest.

AUTHORS CONTRIBUTION

K.A. Khalid (Methodology; Data curation; Writing - original draft; Resources)

K. Osman (Conceptualization; Software; Funding acquisition; Supervision)

K.M. Isa (Writing - original draft; Validation; Writing - review & editing)

A.M.M. Ismail (Analyse the data; Parameters study; Proofread manuscript)

N. Ahmad (Performing analysis; Proofreading manuscript)

A.A.M. Yusof (Project administration; Visualisation; Conceptualization; Writing - review & editing)

AVAILABILITY OF DATA AND MATERIALS

The data supporting this study’s findings are available on request from the corresponding author.

ETHICS STATEMENT

There are no human or animal subjects involved in this article and informed consent is not applicable

REFERENCE

[1] S. R. Bistafa, “On the development of the Navier-Stokes equation by Navier,” *Revista Brasileira de Ensino de Física,* vol. 40, no. 2, p. e2603, 2018.

[2] C. De Michele, F. Capuano, G. Coppola, “Fast-projection methods for the incompressible Navier–Stokes equations,” *Fluids,* vol. 5, no. 4, p. 222, 2020.

[3] M. Kavitha, L. Abhang, M. Kumar, M. Balamurugan, M. Mallesh, “Advanced finite element methods for solving fluid dynamics problems in engineering applications,” *Metallurgical and Materials Engineering,* vol. 31, no. 3, pp. 201-209, 2025.

[4] W. Cao, X. Shan, S. Tang, W. Ouyang, W. Zhang, “Solving parametric high-Reynolds-number wall-bounded turbulence around airfoils governed by Reynolds-averaged Navier–Stokes equations using time-stepping-oriented neural network,” *Physics of Fluids,* vol. 37, no. 1, p. 1, 2025.

[5] A. J. Chorin, "Numerical solution of the Navier-Stokes equations," *Mathematics of Computation,* vol. 22, no. 104, pp. 745-762, 1968.

[6] N. N. Janenko. The Method of Fractional Steps: The Solution Problems of Mathematical Physics in Several Variables. 1st Ed. Berlin: Springer Science & Business Media, 2012.

[7] J. P. Boyd. Chebyshev and Fourier Spectral Methods. 2nd Ed. New York: Dover Publications, Inc., 2001.

[8] S. Blanes, F. Casas, A. Murua, "Splitting methods for differential equations," *Acta Numerica,* vol. 1, pp. 1-160, 2024.

[9] R. Temam. Navier–Stokes Equations: Theory and Numerical Analysis. Volume 343. Rhode Island: AMS Chelsea Publishing, 2024.

[10] J. Van Kan, "A second-order accurate pressure-correction scheme for viscous incompressible flow," *SIAM Journal on Scientific and Statistical Computing,* vol. 7, no. 3, pp. 870-891, 1986.

[11] G. E. Karniadakis, M. Israeli, S. A. Orszag, "High-order splitting methods for the incompressible Navier-Stokes equations," *Journal of Computational Physics,* vol. 97, no. 2, pp. 414-443, 1991.

[12] K. B. Osman, "Multiple steady solutions and bifurcations in the symmetric driven cavity," *Doctoral Dissertations*, University of New Hampshire, 2004.

[13] M. Z. Ngali, "Solution to navier-stokes equation in stretched coordinate system," *Master Thesis,* Universiti Teknologi Malaysia, 2005.

[14] W. Kim, H. Choi, "Immersed boundary methods for fluid-structure interaction: A review," *International Journal of Heat and Fluid Flow,* vol. 75, pp. 301-309, 2019.

[15] H. Ibarra-Villalon, O. Pottiez, A. Gómez-Vieyra, J. Lauterio-Cruz, "Comparative study of finite difference methods and pseudo-spectral methods for solving the nonlinear Schrödinger equation in optical fiber," *Physica Scripta,* vol. 98, no. 6, p. 065514, 2023.

[16] H. Ibarra-Villalon, O. Pottiez, A. Gómez-Vieyra, Y. Bracamontes-Rodriguez, J. Lauterio-Cruz, "Numerical study of the fundamental fiber soliton propagation," *Revista Mexicana de Física E,* vol. 17, no. 12, pp. 191-200, 2020.

[17] S. A. Mathias, G. C. Sander, "Pseudospectral methods provide fast and accurate solutions for the horizontal infiltration equaetion," *Journal of Hydrology,* vol. 598, p. 126407, 2021.

[18] O. Botella,R. Peyret, "Benchmark spectral results on the lid-driven cavity flow," *Computers & Fluids,* vol. 27, no. 4, pp. 421-433, 1998.

[19] H. Johnston, J.-G. Liu, "Accurate, stable and efficient Navier–Stokes solvers based on explicit treatment of the pressure term," *Journal of Computational Physics,* vol. 199, no. 1, pp. 221-259, 2004.

[20] J.-L. Guermond, L. Quartapelle, "Calculation of incompressible viscous flows by an unconditionally stable projection FEM," *Journal of Computational Physics,* vol. 132, no. 1, pp. 12-33, 1997.

[21] U. Ghia, K. N. Ghia, C. Shin, "High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method," *Journal of Computational Physics,* vol. 48, no. 3, pp. 387-411, 1982.

[22] L. N. Trefethen. Spectral Methods in MATLAB. 1st Ed. United States: SIAM Publications Library, 2000.

[23] W. Kress, P. Lötstedt, "Time step restrictions using semi-explicit methods for the incompressible Navier–Stokes equations," *Computer Methods in Applied Mechanics and Engineering,* vol. 195, no. 33-36, pp. 4433-4447, 2006.