

DEVELOPMENT OF A LOCALLY ANALYTIC PROLONGATION OPERATOR IN THE TWO-GRID, THREE-LEVEL METHOD FOR THE NAVIER-STOKES EQUATIONS

Tony W. H. Sheu and R. K. Lin

Department of Engineering Science and Ocean Engineering, National Taiwan University, Taipei, Taiwan, Republic of China

In this article, two three-level methods employing the same prolongation operator are proposed for efficiently solving the incompressible Navier-Stokes equations in a two-grid system. Each method involves solving one smaller system of nonlinear equations in the coarse mesh. The chosen Newton- or Oseen-type linearized momentum equations along with a correction step are solved only once on the fine mesh. Within the three-level framework, the locally analytic prolongation operator needed to bridge the convergent Navier-Stokes solutions obtained at the coarse mesh and the interpolated velocities at the fine mesh is developed to improve the prediction quality. To increase prediction accuracy, the linearized momentum equations are discretized within the alternating direction implicit context using our previously developed nodally exact convection-diffusion-reaction finite-difference scheme. Two proposed three-level methods are rigorously assessed in terms of simulated accuracy, nonlinear convergence rate, and elapsed CPU time.

1. INTRODUCTION

Numerical simulation of incompressible viscous flow equations often encounters stability problems for velocities in the approximation of multidimensional advective terms [1]. Moreover, many upwind advection schemes can introduce false diffusion error [2]. Therefore, splitting the equations becomes an attractive means to dispense with crosswind diffusion error without sacrificing scheme stability [3]. Another trivial benefit of applying the operator splitting technique is that solutions can be obtained efficiently within the one-dimensional framework. When simulating the incompressible Navier-Stokes equations in collocated grids, it is necessary to eliminate the node-to-node oscillatory pressure solutions arising from the decoupling velocity and pressure fields [4–6]. This motivated the calculation of incompressible flow solutions on nonstaggered grids for ease of programming [7].

Received 6 January 2006; accepted 17 February 2006.

This work was supported by the National Science Council of the Republic of China under Grant NSC 94-2611-E-002-021.

Address correspondence to Tony W. H. Sheu, Department of Engineering Science and Ocean Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, 106 Taiwan, Republic of China. E-mail: twhsheu@ntu.edu.tw

NOMENCLATURE

\mathbf{f}	body force per unit volume	Re	Reynolds number ($\equiv \rho u_\infty L / \mu$)
K	diffusion coefficient defined in Eq. (16)	u_{iid}	characteristic velocity
$K_0(m)$	modified Bessel function defined in Eq. (27)	u_∞	reference velocity
L	characteristic length	γ	Euler's constant
\mathbf{n}	unit outward normal vector	ξ	delta function
		ρ	fluid density

Linearization of incompressible flow equations is another major bottleneck. Inappropriate linearization of the convective terms may retard solution convergence and sometimes can even cause the divergence of the solution. Computational expense is therefore considerable. Among earlier proposed methods, the multilevel method has gained some attraction to avoid time-consuming calculation of the nonlinear Navier-Stokes equations [8–10]. To accelerate the nonlinear convergence, we are motivated to apply the multilevel method, which solves the differential equations on continuously refined mesh points. For example, in a two-level method, initially the nonlinear differential system is solved in the coarse mesh, followed by carrying out a computationally more expensive calculation for the same differential system of equations in the fine mesh. To communicate the two solutions obtained at the coarse and fine meshes, a prolongation step is needed to approximate \mathbf{u}^* shown in $(\mathbf{u}^* \cdot \nabla)\mathbf{u}$ in the fine mesh, where \mathbf{u}^* is the convergent velocity vector obtained from the full Navier-Stokes equations in the coarse mesh. To the authors' knowledge, there exist some multilevel methods that have been successfully applied to solve semilinear systems of elliptic equations [11], scalar partial differential equations [12], nonlinear reaction-diffusion equations [13], and the Navier-Stokes equations [14].

In multilevel Navier-Stokes flow analysis, the Newton equation, which is implemented either in a fixed mesh [15] or in several successive meshes [16], has been referred to. For a high Reynolds number problem, use of the Newton linearization method can result in an asymmetric matrix that is highly indefinite [1]. The matrix asymmetry and indefiniteness, which are the two major difficulties in solving the matrix equation, arise from the convective term and the reaction (or production) term, respectively. To resolve the indefiniteness problem, Layton and Lenferink [9] neglected the reaction term shown in the Newton linearized equation [8]. Hence, the Oseen-type Navier-Stokes algebraic system becomes relatively easier to be solved and, in fact, has been extensively employed in the past. One can refer to the work of Layton and Tobiska [14] for more information about this class of methods. Since the computational efficiency of multilevel Navier-Stokes methods depends strongly on the linearization method chosen, the defect-correction step can be integrated into the modified Picard method (or Oseen two-step method plus correction step) [10] to improve the convergence [14] for a computation carried out in the coarse mesh.

The remainder of this article is organized as follows. In Section 2, the momentum equations in primitive-variable form are solved along with the pressure Poisson equation. This is followed by a presentation of the three-level methods employed. In Section 4, the underlying convection-diffusion-reaction (CDR) spatial discretization scheme is applied alternatively in each spatial direction based on the alternating

direction implicit (ADI) method when solving the momentum equations. To improve the solution accuracy in the fine mesh, we develop in Section 5 an analytic prolongation operator so as to communicate the simulated coarse-mesh Navier-Stokes solutions with those needed in the fine mesh. In Section 6, the model is validated by solving a problem which is amenable to exact solution. Also, the lid-driven cavity problem is investigated at different Reynolds numbers. Finally, some conclusions are drawn in Section 7.

2. WORKING EQUATIONS

The viscous incompressible flow motion governed by the following continuity equation and momentum equations is investigated at the Reynolds number $Re(\equiv \rho u_\infty L/\mu)$, where μ is denoted as the fluid viscosity:

$$\nabla \cdot \mathbf{u} = 0 \tag{1}$$

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

All lengths are normalized by L , the velocity components by u_∞ , the time by L/u_∞ , and the pressure by ρu_∞^2 , where ρ denotes the fluid density. The chosen primitive variables \mathbf{u} and p are sought subject to the boundary velocities. It is noted that specification of boundary condition for p is not permitted for the differential system given by (1)–(2) [17–19].

To preserve the incompressibility condition, momentum conservation equations can be solved together with the divergence-free constraint condition [or continuity equation (1)]. This coupled method cannot, however, result in a well-conditioned matrix and, in turn, may yield a poor matrix eigenvalue spectrum. For a large-sized problem, it is very difficult to obtain (\mathbf{u}, p) solutions from (1)–(2) using either a computationally less expensive iterative solver [20] or a memory-demanding direct solver. These two drawbacks make the use of the coupled formulation to solve Eqs. (1)–(2) less attractive. One strategy to circumvent this difficulty is to apply the well-known pressure Poisson equation (PPE) approach [2, 21]. By applying a curl operator on the momentum equations, the following pressure Poisson equation can be derived in lieu of the divergence-free equation (1):

$$\nabla^2 p = \nabla \cdot \left[\frac{1}{Re} \nabla^2 \mathbf{u} - (\mathbf{u} \cdot \nabla)\mathbf{u} + \mathbf{f} \right] \tag{3}$$

For the purpose of closure, analysis of Eq. (3) needs to be supplemented with the rigorous integral-type pressure boundary condition derived in [17]. In the current study, such a theoretical integral pressure boundary condition is not implemented, for ease of computation. Instead, the following Neumann-type pressure boundary condition for Eq. (3) is adopted [22]:

$$\frac{\partial p}{\partial n} = \left[\frac{1}{Re} \nabla^2 \mathbf{u} - (\mathbf{u} \cdot \nabla)\mathbf{u} + \mathbf{f} \right] \cdot \mathbf{n} \tag{4}$$

In the above equation, \mathbf{n} denotes the unit outward normal vector to the boundary. Note that Eq. (4) is legitimate only when the discrete divergence-free condition is satisfied.

Subjected to the velocity boundary conditions, Eqs. (2)–(4) are discretized in a domain in which the velocity and pressure unknowns are stored at the same point. Within this nonstaggered-mesh framework, the pressure gradient term must be appropriately approximated to avoid checkerboard pressure oscillations. The approach chosen to suppress the notorious even–odd solution pattern is to take the nodal value of p_j into account while discretizing ∇p at the interior node j . Our approximation of $\nabla p|_j$ involves solving the following implicit equation for F_j ($F_j = hp_j$, where h denotes the uniform mesh size) [23]:

$$\alpha F_{j+1} + \beta F_j + \gamma F_{j-1} = c_1(p_{j+2} - p_{j+1}) + c_2(p_{j+1} - p_j) + c_3(p_j - p_{j-1}) + c_4(p_{j-1} - p_{j-2}) \quad (5)$$

The seven coefficients are determined by expanding $F_{j\pm 1}$ in Taylor series with respect to F_j and $p_{j\pm 1}$, $p_{j\pm 2}$ with respect to p_j . This is followed by substituting the resulting expansions into Eq. (5) to derive a set of algebraic equations for α , β , γ , c_1 , c_2 , c_3 , and c_4 . Note that it is legitimate to set $\alpha = \gamma$ owing to the elliptic nature of Eq. (3) for p . Setting $\alpha = \gamma$, the rest of the coefficients shown in Eq. (5) are determined as $\beta = 3/5$, $\gamma = 1/5$, $c_1 = 1/60$, $c_2 = 29/60$, $c_3 = 29/60$, and $c_4 = 1/60$. The readers may refer to the other compact schemes for the first-derivative terms discussed in [24,25].

3. TWO-GRID, THREE-LEVEL METHODS

3.1. Linearization of Momentum Equations

In this study, the Newton-Raphson method is chosen to linearize the convective terms shown in Eq. (2). For linearization, a product term, say ST , at the current iteration $k + 1$ is expanded with respect to that evaluated at the previous iteration k to yield the first-order-accurate expression for $S^{k+1} T^{k+1} \approx S^{k+1} T^k + S^k T^{k+1} - S^k T^k + \dots + \text{H.O.T.}$, where H.O.T. = higher-order terms [26]. Thanks to this approximation, the nonlinear terms $(u^2)_x^{k+1}$ and $(uv)_y^{k+1}$ shown in the x - and y -momentum equations are linearized as

$$(u^2)_x^{k+1} = u_x^{k+1} u^k + u^{k+1} u_x^k + u_x^k u^{k+1} + u^k u_x^{k+1} - u_x^k u^k - u^k u_x^k \quad (6)$$

$$(uv)_y^{k+1} = u_y^{k+1} v^k + u^{k+1} v_y^k + u_y^k v^{k+1} + u^k v_y^{k+1} - u_y^k v^k - u^k v_y^k \quad (7)$$

Substitution of the above two equations into Eq. (2) enables us to derive the linearized x - and y -momentum equations, respectively, as shown below:

$$u^k u_x^{k+1} + v^k u_y^{k+1} - \frac{1}{\text{Re}} \nabla^2 u^{k+1} + \underline{u_x^k u^{k+1}} = -p_x^{k+1} + \underline{u^k u_x^k + v^k u_y^k - u_y^k v^{k+1}} \quad (8)$$

$$u^k v_x^{k+1} + v^k v_y^{k+1} - \frac{1}{\text{Re}} \nabla^2 v^{k+1} + \underline{v_y^k v^{k+1}} = -p_y^{k+1} + \underline{u^k v_x^k + v^k v_y^k - v_x^k u^{k+1}} \quad (9)$$

The underlined terms denote the extra corrections to the classical linearization method.

The convective terms in the linearized equations (8)–(9) are responsible for the presence of matrix asymmetry, which increases memory storage requirements, and for the decreasing diagonal dominance, which can destabilize the discrete system. Numerical instability of this kind can be, in theory, resolved by applying any upwinding scheme for the convective terms. The indefinite nature of the reaction term $\nabla \mathbf{u}^k \cdot \mathbf{u}^{k+1}$ shown in Eqs. (8)–(9) can cause another type of numerical instability to occur [1]. This is the reason for omitting the potentially destabilizing positive-valued linear production (or reaction) term shown in the Newton linearized equation. We can therefore consider the resulting Oseen linearization method [10] as the approximated Newton linearization method.

3.2. Two Investigated Three-Level Methods

The level of difficulty in solving the Navier-Stokes equations increases with decreasing mesh size because of the increasingly reduced spectrum for eigenvalues. To accelerate convergence, a multilevel method, which involves calculations at the coarse and fine grid levels, has been proposed [8, 9]. For example, in a two-level method, the convergent solutions for Eqs. (1) and (3) are solved from a comparatively conditioned matrix system on the coarse mesh. This is followed by solving the linearized Navier-Stokes equations only once in the fine mesh. Two-level methods differ from each other solely in the chosen (linear) equations which are solved in the fine mesh.

The Navier-Stokes equations, namely, $(\mathbf{u}_H \cdot \nabla)\mathbf{u}_H - (1/\text{Re})\nabla^2 \mathbf{u}_H^2 = -\nabla p_H + \mathbf{f}$, are initially solved in a coarse mesh of width H until the convergent solutions (\mathbf{u}_H, p_H) are obtained. This is followed by exporting the converged \mathbf{u}_H solution to the corresponding nodes in the fine mesh of grid width h . Depending on the chosen Newton- or Oseen-type linearization method, in the second step the solutions for (\mathbf{u}_h, p_h) in the fine mesh are sought from Eqs. (3)–(4), and the corresponding linearized momentum equations are given below:

Newton equation for \mathbf{u}_h^N [8]

$$(\mathbf{u}_H \cdot \nabla)\mathbf{u}_h^N - \frac{1}{\text{Re}}\nabla^2 \mathbf{u}_h^N + (\mathbf{u}_h^N \cdot \nabla)\mathbf{u}_H = -\nabla p_h^N + (\mathbf{u}_H \cdot \nabla)\mathbf{u}_H + \mathbf{f} \quad (10)$$

Oseen equation for \mathbf{u}_h^O [9]

$$(\mathbf{u}_H \cdot \nabla)\mathbf{u}_h^O - \frac{1}{\text{Re}}\nabla^2 \mathbf{u}_h^O = -\nabla p_h^O + \mathbf{f} \quad (11)$$

The above Newton- and Oseen-type linearized equations are then solved only once in the fine mesh.

Although the solutions for (\mathbf{u}_H, p_H) can be more easily obtained in the coarse mesh, the simulated solution accuracy may be less than satisfactory. However, a defect correction can be implemented in the coarse mesh to improve the accuracy. Define $\mathbf{u}_h^N = \mathbf{u}_h^O + \mathbf{e}_H^{\text{OC}}$ and $p_h^N = p_h^O + \varepsilon_H^{\text{OC}}$ and then substitute them into Eq. (11).

The equation for the correction term \mathbf{e}_H^{OC} can be derived accordingly as [11, 14]

$$\nabla \cdot \mathbf{e}_H^{\text{OC}} = 0 \quad (12)$$

$$(\mathbf{u}_H \cdot \nabla) \mathbf{e}_H^{\text{OC}} - \frac{1}{\text{Re}} \nabla^2 \mathbf{e}_H^{\text{OC}} + (\mathbf{e}_H^{\text{OC}} \cdot \nabla) \mathbf{u}_H = -\nabla \varepsilon_H^{\text{OC}} + [(\mathbf{u}_H - \mathbf{u}^0) \cdot \nabla] \mathbf{u}_H + \mathbf{f} \quad (13)$$

Similarly, by the definitions of $\mathbf{u}_h = \mathbf{u}^{\text{N}} + \mathbf{e}_H$ and $p_h = p^{\text{N}} + \varepsilon_H$, the following Newton defect-correction equation for \mathbf{e}_H^{NC} can be derived from Eq. (10) as [9, 14]

$$\nabla \cdot \mathbf{e}_H^{\text{NC}} = 0 \quad (14)$$

$$(\mathbf{u}_H \cdot \nabla) \mathbf{e}_H^{\text{NC}} - \frac{1}{\text{Re}} \nabla^2 \mathbf{e}_H^{\text{NC}} + (\mathbf{e}_H^{\text{NC}} \cdot \nabla) \mathbf{u}_H = -\nabla \varepsilon_H^{\text{NC}} + [(\mathbf{u}_H - \mathbf{u}^{\text{N}}) \cdot \nabla] (\mathbf{u}^{\text{N}} - \mathbf{u}_H) \quad (15)$$

For clarity, the equations for the Newton, Oseen, modified Picard (Oseen two-step method plus correction step), and modified Newton (Newton two-step method plus correction step) methods are summarized in the Appendix.

4. DISCRETIZATION OF EQUATIONS IN THE COLLOCATED MESH

As Eqs. (8)–(9) reveal, the Newton linearized Navier-Stokes equations take the following convection-diffusion-reaction (CDR) form:

$$A\phi_x + B\phi_y - K\nabla^2\phi + C\phi = F \quad (16)$$

The presence of the convection and production terms can simultaneously destabilize the scheme stability. Moreover, while solving the above two-dimensional transport equation, the simulated result is susceptible to false diffusion error [2]. Because of these three numerical difficulties, we resort to the operator splitting technique of Peaceman and Rachford [27] to obtain a solution alternatively along each spatial direction.

$$A\phi_x^* - K\phi_{xx}^* + C\phi^* = F_1 \quad (17)$$

$$B\phi_y^{n+1} - K\phi_{yy}^{n+1} + C\phi^{n+1} = F_2 \quad (18)$$

The source terms shown in the above two equations are expressed as $F_1 = F^* - B\phi_y^n + K\phi_{yy}^n$ and $F_2 = F^{n+1} - A\phi_x^* + K\phi_{xx}^*$, respectively.

Within the spatial splitting framework, many well-established one-dimensional discretization schemes can be applied. To obtain an accurate solution from the two-dimensional equation (16), an accurate CDR scheme has been developed for the one-dimensional model equation given by $\bar{u}\phi_x - \bar{k}\phi_{xx} + \bar{c}\phi = \bar{f}$ [3]. The general solution for this model equation is given by $\phi = a_1 e^{\lambda_1 x} + a_2 e^{\lambda_2 x} + \bar{f}/\bar{c}$, where $\lambda_1 = (\bar{u}^* + \sqrt{\bar{u}^2 + 4\bar{c}\bar{k}})/2\bar{k}$ and $\lambda_2 = (\bar{u}^* - \sqrt{\bar{u}^2 + 4\bar{c}\bar{k}})/2\bar{k}$. At an interior node i ,

the discretized equation is assumed to take the following form:

$$\left(-\frac{\bar{u}}{2h} - \frac{m}{h^2} + \frac{\bar{c}}{6}\right)\phi_{i-1} + 2\left(\frac{m}{h^2} + \frac{\bar{c}}{3}\right)\phi_i + \left(\frac{\bar{u}}{2h} - \frac{m}{h^2} + \frac{\bar{c}}{6}\right)\phi_{i+1} = \bar{f} \quad (19)$$

Substitution of the exact solutions for $\phi_i = a_1 e^{\lambda_1 x_i} + a_2 e^{\lambda_2 x_i} + \bar{f}/\bar{c}$, $\phi_{i+1} = a_1 e^{\lambda_1 h} e^{\lambda_1 x_i} + a_2 e^{\lambda_2 h} e^{\lambda_2 x_i} + \bar{f}/\bar{c}$, and $\phi_{i-1} = a_1 e^{-\lambda_1 h} e^{\lambda_1 x_i} + a_2 e^{-\lambda_2 h} e^{\lambda_2 x_i} + \bar{f}/\bar{c}$ into the Eq. (19) enables us to derive m as follows [5]:

$$m = h^2 \left[\frac{(\bar{c}/3) + (\bar{c}/6) \cosh(\bar{\lambda}_1) \cosh(\bar{\lambda}_2) + (\bar{u}/2h) \sinh(\bar{\lambda}_1) \cosh(\bar{\lambda}_2)}{\cosh(\bar{\lambda}_1) \cosh(\bar{\lambda}_2) - 1} \right] \quad (20)$$

where

$$(\bar{\lambda}_1, \bar{\lambda}_2) = \left(\frac{\bar{u}h}{2k}, \sqrt{\left(\frac{\bar{u}h}{2k}\right)^2 + \frac{\bar{c}h^2}{k}} \right)$$

5. DERIVATION OF THE LOCALLY ANALYTIC PROLONGATION OPERATOR

The method for calculating the solution (\mathbf{u}_H, p_H) in the coarse mesh remains unchanged for each multilevel method investigated. In the fine mesh, the linearized equations are solved only once. Therefore, the quality of the multilevel method employed depends mainly on the way chosen to map the convergent values of \mathbf{u}_H obtained at the nodes “□” (see schematic in Figure 1) in the coarse mesh to those in the fine mesh. In this study, an attempt is made to develop an analytic prolongation operator so as to be able to communicate effectively the solutions obtained at nodes with different levels of grid resolution.

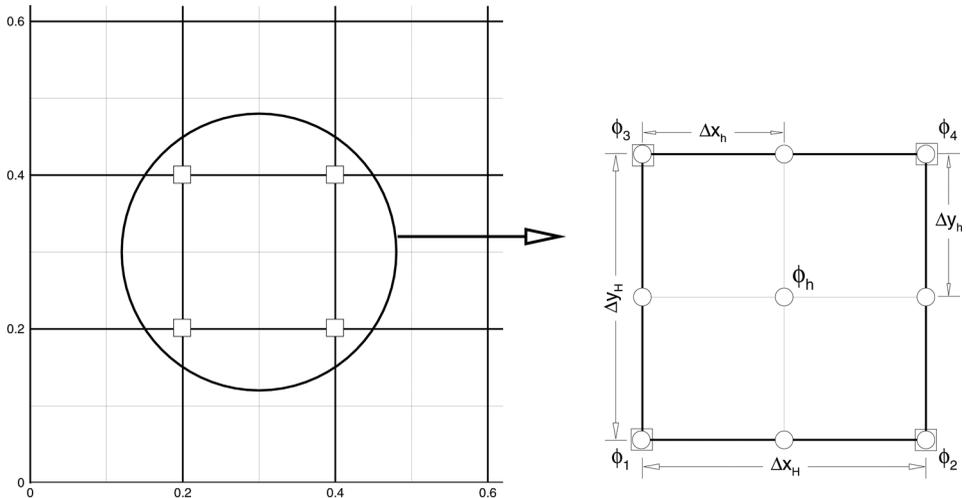


Figure 1. Schematic of the coarse and fine mesh systems (○, fine mesh; □, coarse mesh).

The prolongation operator, which is the key element of the current study, is derived in detail as shown below. Consider the following constant-coefficient equation in a simply connected domain V :

$$a\phi_x + b\phi_y - k\nabla^2\phi + c\phi = 0 \quad (21)$$

The convective velocity components a and b represent the two constants along the x and y directions, respectively. Two coefficients k and c shown above are known as the diffusion coefficient and the reaction coefficient, respectively. For the sake of simplicity, Eq. (21) is solved subject to the condition $\phi = g$ on the boundary ∂V . Equation (21) can be rewritten as

$$\nabla^2\phi = 2\bar{A}\phi_x + 2\bar{B}\phi_y + d \quad (22)$$

where $\bar{A} = a/2k$, $\bar{B} = b/2k$, and $d = c/2k$. By virtue of the mapping equation

$$\phi(x, y) = e^{\bar{A}x + \bar{B}y} f(x, y) \quad (23)$$

Eq. (21) for ϕ can be transformed to the following Helmholtz equation for f :

$$\nabla^2 f - \bar{k}^2 f = 0 \quad (24)$$

where $\bar{k}^2 = \bar{A}^2 + \bar{B}^2 + d$. How the above Helmholtz equation for f is discretized turns out to be the key issue in the numerical simulation of the CDR equation for ϕ .

Derivation of the prolongation operator is followed by introducing the scalar function G , which is governed by the following inhomogeneous Helmholtz equation:

$$\nabla^2 G - \bar{k}^2 G = \delta(\underline{x} - \underline{\xi}) \quad (25)$$

In the above equation, $\underline{\xi}$ represents the delta function. The reason for conducting the transformation given by Eq. (23) is that the resulting auxiliary equation (25) is amenable to the exact solution given by

$$G = \frac{1}{2\pi} K_0(\bar{k}|\underline{x} - \underline{\xi}|) \quad (26)$$

Here, the modified Bessel function $K_0(m)$ is expressed as

$$\begin{aligned} K_0(m) = & -\left[\ln\left(\frac{m}{2}\right) + \gamma\right] \left(1 + \frac{m^2}{2^2} + \frac{m^4}{2^2 4^2} + \frac{m^6}{2^2 4^2 6^2} + \dots\right) + \frac{m^2}{2^2} \\ & + \frac{m^4}{2^2 4^2} \left(1 + \frac{1}{2}\right) + \frac{m^6}{2^2 4^2 6^2} \left(1 + \frac{1}{2} + \frac{1}{3}\right) + \dots \end{aligned} \quad (27)$$

The Euler's constant $\gamma [\equiv \lim 1 + (1/2) + (1/3) + \dots + (1/n) - \ln(n)]$ shown above is equal to 0.5772156649. By performing $[f \cdot (25) - G \cdot (24)]$, the following equation

is derived:

$$f(\nabla^2 G - \bar{k}^2 G) - G(\nabla^2 f - \bar{k}^2 f) = f \delta(\underline{x} - \underline{\xi}) \tag{28}$$

Integration of Eq. (28) in a volume $V(\xi)$ enables us to derive the following equation:

$$\int_{V(\xi)} [f \nabla^2 G - G \nabla^2 f] dV = \varepsilon(\underline{x}) f(\underline{x}) \tag{29}$$

where

$$\varepsilon(\underline{x}) = \begin{cases} 1 & \underline{x} \in V \\ 0 & \text{otherwise} \end{cases} \tag{30}$$

By conducting an integration by parts on Eq. (29), the equivalent equation along $S(\equiv \partial V)$, which is the boundary of $V(\xi)$ having an outward normal direction of n , is derived as

$$\int_S \left(f \frac{\partial G}{\partial n} - G \frac{\partial f}{\partial n} \right) dS = \varepsilon(\underline{x}) f(\underline{x}) \tag{31}$$

In a single element with its centroid located at (i,j) , f is approximated by the bi-quadratic polynomials N_i to render $f = \sum_{k=1}^9 N_k f_k$. By employing the expression of G as defined in Eq. (26), along with $\partial G/\partial n$, and substituting f into Eq. (31), the implicit nine-point stencil equation at node 5 in the schematic in Figure 2 is derived as

$$A_1 f_1 + A_2 f_2 + A_3 f_3 + A_4 f_4 + A_6 f_6 + A_7 f_7 + A_8 f_8 + A_9 f_9 = f_5 \tag{32}$$

The coefficients A_i ($i = 1 \sim 4$ and $6 \sim 9$) shown above take the following form:

$$A_i = \frac{\int_S [-N_i(\partial G/\partial n) + G(\partial N_i/\partial n)] dS}{\int_S [N_5(\partial G/\partial n) - G(\partial N_5/\partial n)] dS} \tag{33}$$

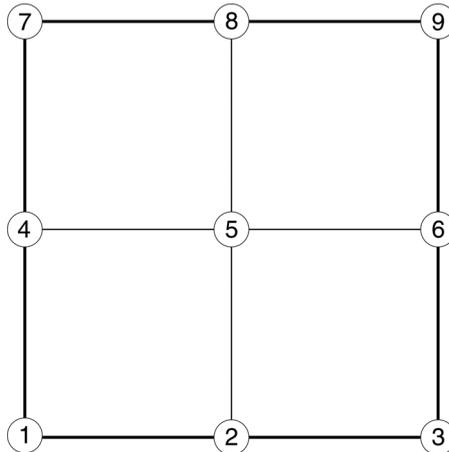


Figure 2. Schematic of the stencil points 1-9.

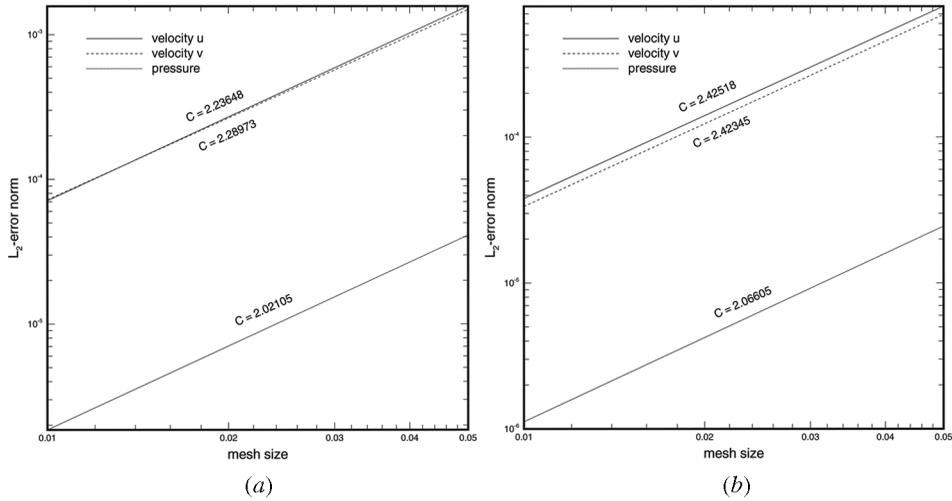


Figure 3. Plots of the rates of convergence C based on the computed L_2 -error norms obtained at different mesh sizes using the three-level methods investigated: (a) modified Picard method; (b) modified Newton method.

By integrating Eq. (31) in each cell from the left to the right end and then from the bottom to the top, the global matrix equation for f is constructed.

6. NUMERICAL RESULTS

6.1. Validation and Assessment Studies

The classical Kovasznay flow problem [28], which is amenable to the following analytic solution, is investigated to justify the proposed three-level methods:

$$u = 1 - e^{\lambda x} \cos(2\pi y) \quad (34)$$

$$v = \frac{\lambda}{2\pi} e^{\lambda x} \sin(2\pi y) \quad (35)$$

$$p = c_1 + \frac{1}{2}(1 - e^{2\lambda x}) \quad (c_1 \text{ is a constant}) \quad (36)$$

Table 1. Computed L_2 -error norms using (a) direct method, (b) modified Picard method, and (c) modified Newton method for the validation problem given by (34)–(36)

Mesh point	$\ u_{\text{ext}} - u_{\text{sol}}\ $			$\ v_{\text{ext}} - v_{\text{sol}}\ $			$\ p_{\text{ext}} - p_{\text{sol}}\ $		
	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)
21 × 21	7.65E-04	1.34E-03	1.06E-03	6.70E-04	1.32E-03	1.06E-03	2.30E-05	4.72E-05	4.52E-05
41 × 41	2.22E-04	7.02E-04	6.03E-04	1.95E-04	7.21E-04	6.03E-04	6.88E-06	2.18E-05	8.98E-06
61 × 61	1.01E-04	5.01E-04	3.52E-04	8.96E-05	4.87E-04	3.52E-04	3.19E-06	7.81E-06	5.34E-06
81 × 81	5.79E-05	9.01E-05	7.97E-05	5.07E-05	8.81E-05	7.97E-05	1.79E-06	3.85E-06	3.25E-06
101 × 101	3.68E-05	6.03E-05	5.30E-05	3.24E-05	5.59E-05	5.30E-05	1.00E-06	2.12E-06	1.79E-06

Table 2. Required CPU time (s) and needed number of iterations N for solving the nonlinear momentum equations using (a) direct method, (b) modified Picard method, and (c) modified Newton method^a

Mesh point	(a)		(b)		(c)	
	CPU	N	CPU	N	CPU	N
21 × 21	188.86	163	28.96	25	28.96	25
41 × 41	226.93	205	42.96	33	42.96	33
61 × 61	546.10	481	93.09	82	93.09	82
81 × 81	1498.64	1308	255.51	223	255.51	223
101 × 101	2269.31	1974	390.86	340	390.86	340

^aNote that the needed grid numbers in each spatial direction for (b) and (c) are half of that used in (a).

where

$$\lambda = \frac{\text{Re}}{2} - \left(\frac{\text{Re}^2}{4} + 4\pi^2 \right)^{1/2}$$

The solutions are calculated at several uniform meshes to obtain the rates of convergence. It is clearly seen from the simulated L_2 -error norms shown in Figure 3 that the computed and the exact solutions agree very well. Moreover, the rates of convergence C plotted in Figures 3a and 3b for the two three-level methods investigated are greater than 2 for both the velocities and pressure.

For the sake of completeness, the computational performance of the two three-level methods investigated is assessed in comparison with that of the one-level method (or direct method), which is implemented in the finest mesh using the methods described in Section 2 for ∇p and in Section 4 for ϕ_x and ϕ_y . The present assessment study is made in terms of the predicted L_2 -error norms, the elapsed CPU seconds needed to reach the user's specified tolerance (10^{-12} set for the nonlinear

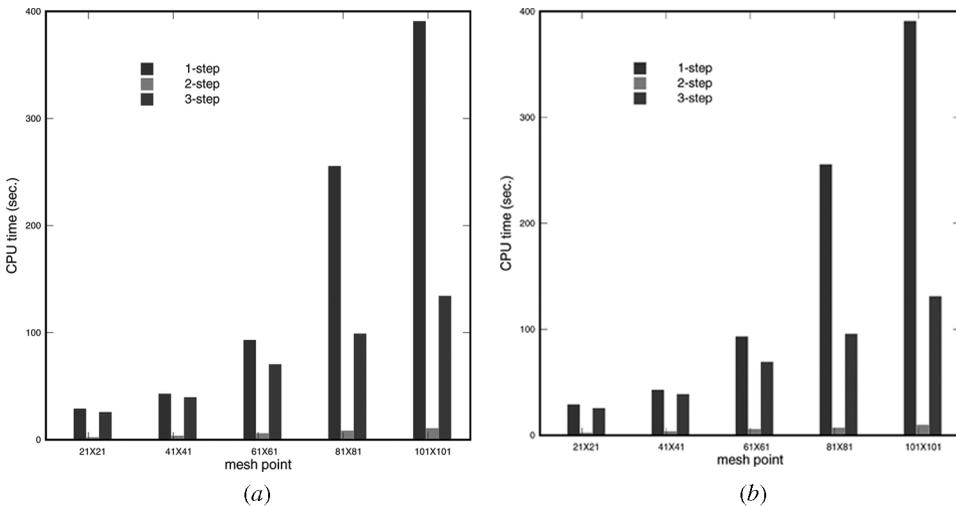


Figure 4. Comparison of the CPU time required (s) for the three steps involved, in the two three-level methods investigated, at different mesh points: (a) modified Picard method; (b) modified Newton method.

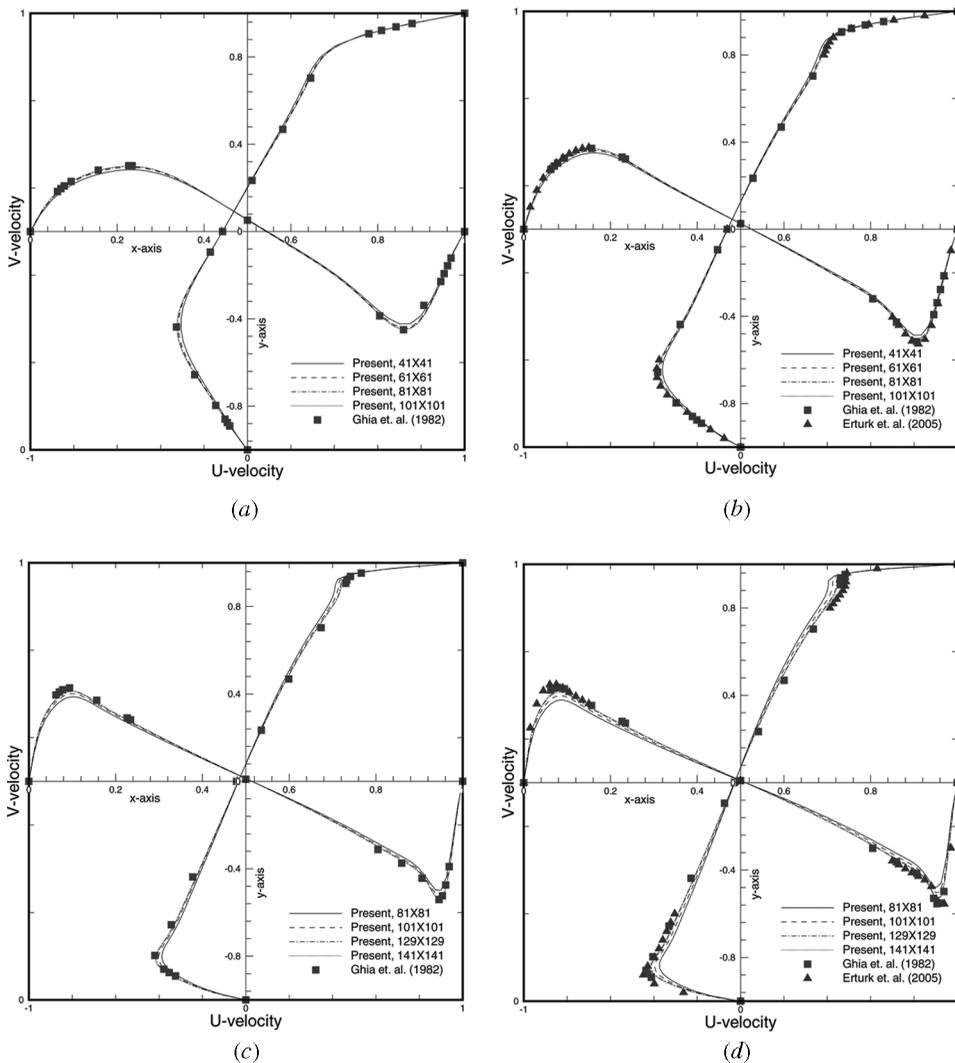


Figure 5. Simulated velocity profiles for $u(x, 0.5)$ and $v(0.5, y)$ at various Reynolds numbers: (a) $Re = 400$; (b) $Re = 1,000$; (c) $Re = 3,200$; (d) $Re = 5,000$.

iteration), and, of course, the total number of nonlinear iterations. In the comparison, the solutions for \mathbf{u} and p have been calculated on five uniformly discretized domains with 21^2 , 41^2 , 61^2 , 81^2 , and 101^2 mesh points. As Table 1 shows, the accuracy of the simulated three-level Navier-Stokes solutions is slightly less than that of the solution obtained from the one-level method. The accompanying negligibly increased L_2 -error norms, however, can result in a considerable saving of CPU time, as is clearly demonstrated in Table 2. The total number of nonlinear iterations N is also seen to be considerably reduced. The expense to obtain the solution from the tri-diagonal matrix

is also decreased because the matrix size has been reduced for the calculations carried out in the coarse mesh. From Table 2 it is observed that the slightly improved accuracy, which is even indistinguishable, using the one-level method is at the expense of CPU time and the number of iterations needed. The advantage of employing the three-level method we use now becomes apparent.

In employing the multilevel method, it is interesting to note the CPU time consumed in the step accounting for the nonlinear Navier-Stokes calculation in the coarse mesh, the prolongation step, and the step accounting for the linearized Navier-Stokes calculation in the fine mesh. To show this, the CPU times for the two three-level methods carried out at different mesh sizes are plotted in Figure 4. The present simulated results show that much of the CPU time is consumed in the

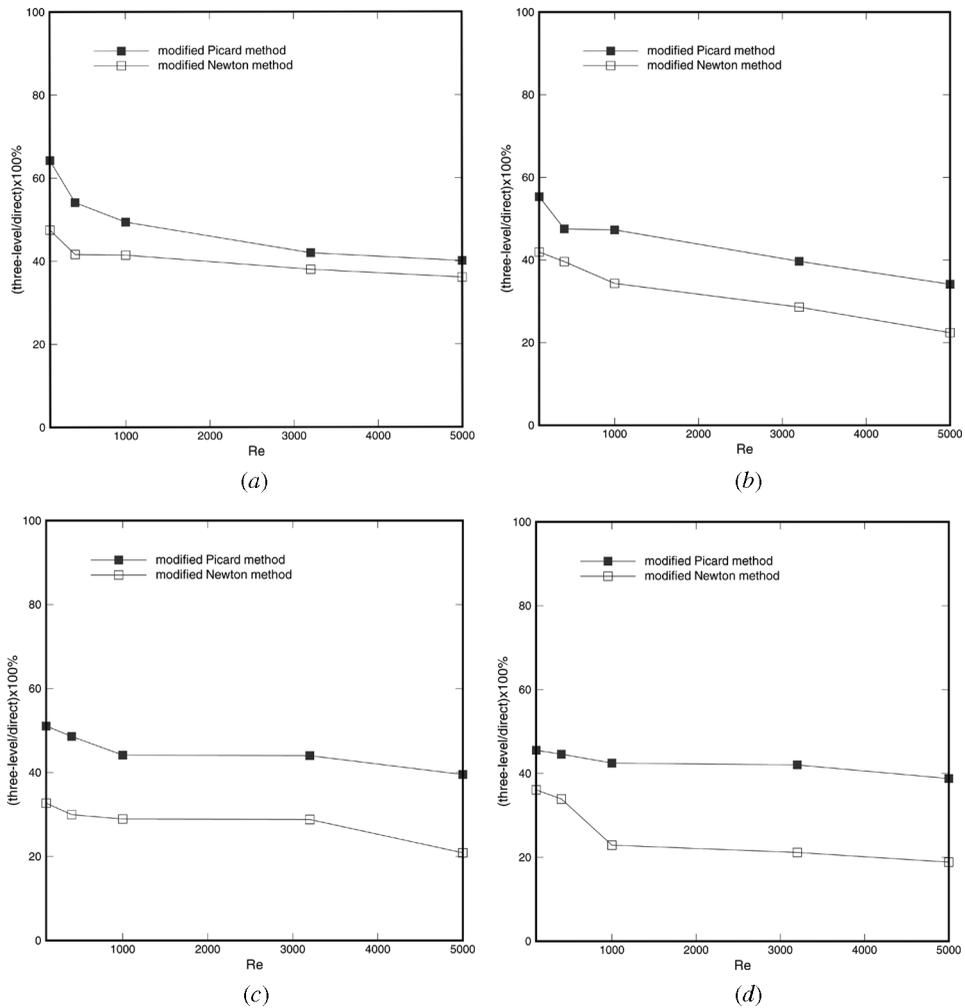


Figure 6. Comparison of the ratios of the CPU times for the two proposed three-level methods against the single-grid method: (a) 41²; (b) 81²; (c) 101²; (d) 129².

nonlinear calculation, which involves fewer mesh points. This demonstrates the importance of accelerating the Navier-Stokes flow calculation in the coarse mesh.

6.2. Lid-Driven Cavity Flow Analysis

The flow driven by a constant upper lid velocity $u_{\text{lid}} (=1)$ in a square cavity is also investigated. With $L (=1)$ as the characteristic length, u_{lid} as the characteristic velocity, and ν as the fluid viscosity, the lid-driven cavity flow problem is investigated at $Re = 400, 1,000, 3,200,$ and $5,000$. In each case, the mesh needs to be continuously refined so as to obtain a grid-independent solution. The simulated velocity profiles $u(0.5, y)$ and $v(x, 0.5)$ plotted at the mid-plane in Figure 5 compare excellently with the benchmark solutions of Ghia [29] and Erturk [30]. The applicability of the proposed scheme is thus confirmed. To demonstrate the efficiency of the proposed three-level methods, the ratios of the CPU time for the modified Picard and modified Newton methods performed at different mesh points are plotted in Figure 6. For completeness, the CPU times needed for computations carried out at different Re values in a single grid are plotted in Figure 7.

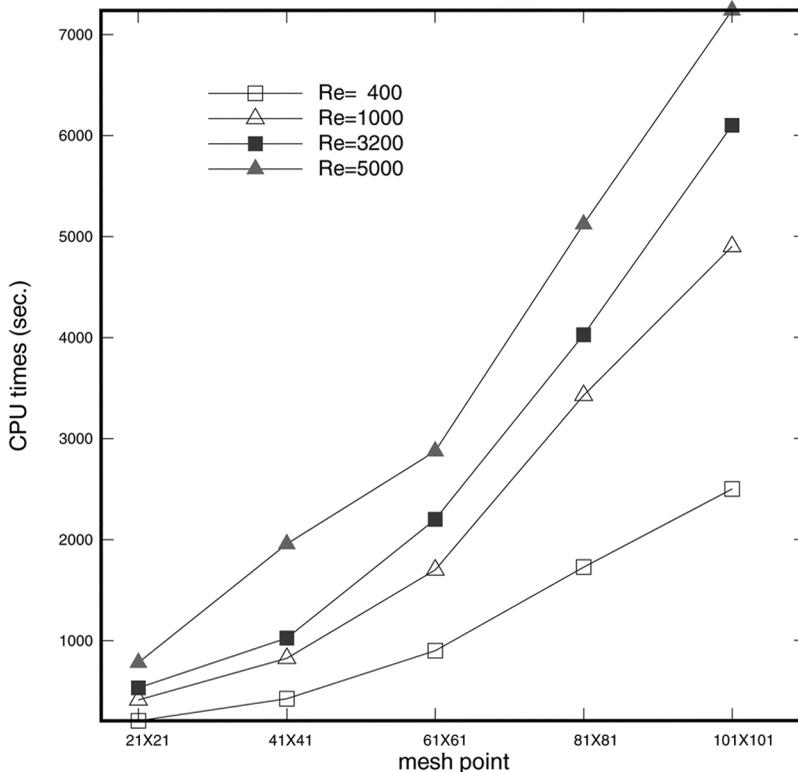


Figure 7. Comparison of the CPU time required (s) for the calculations carried out in the single grid at various mesh points at various Reynolds numbers.

7. CONCLUDING REMARKS

The three-level Navier-Stokes methods employed for solving the incompressible flow equations are featured with the theoretically derived prolongation operator to communicate the solutions obtained at two different mesh levels. Besides improving the prediction quality in the fine mesh using the proposed prolongation operator, another distinct feature of the proposed method is the transformation of the convection-diffusion differential equation into its convection-diffusion-reaction counterpart. For the sake of computational efficiency, the ADI solution algorithm of Peaceman and Rachford is adopted so that the nodally exact one-dimensional convection-diffusion-reaction scheme can be applied. Good agreement between the simulated and analytical solutions is clearly demonstrated for the test problems conducted at different mesh sizes. The present study clearly shows that a slightly engaged deterioration in the prediction accuracy is accompanied by a considerable reduction of CPU time and the number of nonlinear iterations. It can be concluded that the larger the problem, the greater will be the saving of computational time.

REFERENCES

1. S. Turek, A Comparative Study of Time-Stepping Techniques for the Incompressible Navier-Stokes Equations: From Fully Implicit Non-linear Schemes to Semi-implicit Methods, *Int. J. Numer. Meth. Fluids*, vol. 22, pp. 987–1011, 1996.
2. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, New York, 1980.
3. T. W. H. Sheu, S. K. Wang, and R. K. Lin, An Implicit Scheme for Solving the Convection-Diffusion-Reaction Equation in Two Dimensions, *J. Comput. Phys.*, vol. 164, pp. 123–142, 2000.
4. F. H. Harlow and J. E. Welch, Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface, *Phys. Fluids*, vol. 8, pp. 2182–2189, 1965.
5. P. H. Chiu, T. W. H. Sheu, and R. K. Lin, Development of a Dispersion-Relation-Preserving Upwinding Scheme for Incompressible Navier-Stokes Equations on Nonstaggered Grids, *Numer. Heat Transfer B*, vol. 48, no. 6, pp. 543–569, 2005.
6. A. W. Date, Solution of Navier-Stokes Equations on Nonstaggered Grid at All Speeds, *Numer. Heat Transfer B*, vol. 33, no. 4, pp. 451–467, 1998.
7. C. N. Rhie and W. L. Chow, Numerical Study of the Turbulent Flow past an Airfoil with Trailing Edge Separation, *AIAA J.*, vol. 21, pp. 1525–1532, 1983.
8. W. Layton, A Two-Level Discretization Method for the Navier-Stokes Equations, *Comput. Math. Appl.*, vol. 26, pp. 33–38, 1993.
9. W. Layton and W. Lenferink, Two-Level Picard and Modified Picard Methods for the Navier-Stokes Equations, *Appl. Math. Comput.*, vol. 69, no. 2–3, pp. 263–274, 1995.
10. J. Volker, Residual a Posteriori Error Estimates for Two-Level Finite-Element Methods for the Navier-Stokes Equations, *Appl. Numer. Math.*, vol. 37, pp. 503–518, 2001.
11. J. Xu, A Novel Two-Grid Method for Semilinear Elliptic Equations, *SIAM J. Sci. Comput.*, vol. 15, no. 1, pp. 231–237, 1994.
12. J. Xu, Two-Grid Finite Element Discretizations for Nonlinear P.D.E.'s, *SIAM J. Numer. Anal.*, vol. 33, no. 5, pp. 1759–1777, 1996.
13. L. Wu and M. B. Allen, A Two-Grid Method for Mixed Finite-Element Solution of Reaction-Diffusion Equations, *Numer. Meth. Part. Diff. Eqns.*, vol. 15, no. 3, pp. 317–332, 1999.
14. W. Layton and L. Tobiska, A Two-Level Method with Backtracking for the Navier-Stokes Equations, *SIAM J. Numer. Anal.*, vol. 35, no. 5, pp. 2035–2054, 1998.

15. V. Girault and P. A. Raviart, *Finite Element Methods for the Navier-Stokes Equations. Theory & Algorithms*, Springer-Verlag, Berlin, 1986.
16. C. Lin, J. Liu, and S. McCormick, Multilevel Adaptive Methods for Incompressible Flow in Grooved Channels, *J. Comput. Appl. Math.*, vol. 38, pp. 283–295, 1991.
17. L. Quartapelle and M. Napolitano, Integral Conditions for the Pressure in the Computation of Incompressible Viscous Flows, *J. Comput. Phys.*, vol. 64, pp. 340–348, 1986.
18. S. F. Tsai and T. W. H. Sheu, Finite-Element Analysis of Incompressible Navier-Stokes Equations Involving Exit Pressure Boundary Conditions, *Numer. Heat Transfer B*, vol. 39, no. 5, pp. 479–507, 2001.
19. T. W. H. Sheu, S. F. Tsai, and M. M. T. Wang, Discussion of Numerical Deficiency of Applying a Partially Weighted Upwind Finite-Element Model to Incompressible Navier-Stokes Equations, *Numer. Heat Transfer B*, vol. 32, no. 2, pp. 197–214, 1997.
20. M. M. T. Wang and T. W. H. Sheu, An Element-by-Element BICGSTAB Iterative Method for Three-Dimensional Steady Navier-Stokes Equations, *J. Comput. Appl. Math.*, vol. 79, pp. 147–165, 1997.
21. P. M. Gresho and R. L. Sani, On Pressure Boundary Conditions for the Incompressible Navier-Stokes Equations, *Int. J. Numer. Meth. Fluids*, vol. 7, pp. 1111–1145, 1987.
22. T. W. H. Sheu, M. M. T. Wang, and S. F. Tsai, Pressure Boundary Conditions for a Segregated Approach to Solving Incompressible Navier-Stokes Equations, *Numer. Heat Transfer B*, vol. 34, pp. 457–467, 1998.
23. T. W. H. Sheu and R. K. Lin, An Incompressible Navier-Stokes Model Implemented on Non-staggered Grids, *Numer. Heat Transfer B*, vol. 44, no. 3, pp. 277–294, 2003.
24. A. O. Demuren, R. V. Wilson, and M. Carpenter, Higher-Order Compact Schemes for Numerical Simulation of Incompressible Flows, Part I: Theoretical Development, *Numer. Heat Transfer B*, vol. 39, pp. 207–230, 2001.
25. A. O. Demuren, R. V. Wilson, and M. Carpenter, Higher-Order Compact Schemes for Numerical Simulation of Incompressible Flows, Part II: Applications, *Numer. Heat Transfer B*, vol. 39, pp. 231–255, 2001.
26. T. W. H. Sheu and R. K. Lin, Newton Linearization on the Incompressible Navier-Stokes Equations, *Int. J. Numer. Meth. Fluids*, vol. 44, pp. 297–312, 2004.
27. D. W. Peaceman and H. H. Rachford, The Numerical Solution of Parabolic and Elliptic Differential Equations, *J. SIAM*, vol. 3, pp. 28–41, 1955.
28. L. I. G. Kovasznay, Laminar Flow behind a Two Dimensional Grid, *Proc. Camb. Phil. Soc.*, vol. 44, pp. 58–62, 1948.
29. U. Ghia, K. N. Ghia, and C. T. Shin, High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method, *J. Comput. Phys.*, vol. 48, pp. 387–411, 1982.
30. E. Erturk, T. C. Corke, and C. Gökcöl, Numerical Solution of 2-D Steady Incompressible Driven Cavity Flow at High Reynolds Numbers, *Int. J. Numer. Meth. Fluids*, vol. 48, pp. 747–774, 2005.

APPENDIX

One-Level Method (Direct Method Implemented in a Single Grid)

$$(\mathbf{u}_H \cdot \nabla) \mathbf{u}_H = -\nabla p_H + \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_H + \mathbf{f} \quad (\text{A.1})$$

Two-Grid, Two-Level Methods

Newton method for \mathbf{u}_h^N

$$\text{1-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{u}_H = -\nabla p_H + \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_H + \mathbf{f} \quad (\text{A.2})$$

$$\begin{aligned} \text{2-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{u}_h^N - \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_h^N + (\mathbf{u}_h^N \cdot \nabla) \mathbf{u}_H \\ = -\nabla p_h^N + (\mathbf{u}_H \cdot \nabla) \mathbf{u}_H + \mathbf{f} \end{aligned} \quad (\text{A.3})$$

Oseen method for \mathbf{u}_h^O

$$\text{1-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{u}_H = -\nabla p_H + \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_H + \mathbf{f} \quad (\text{A.4})$$

$$\text{2-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{u}_h^O - \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_h^O = -\nabla p_h^O + \mathbf{f} \quad (\text{A.5})$$

Two-Grid, Three-Level Methods

Modified Picard method (Oseen two-step method plus correction step)

$$\text{1-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{u}_H = -\nabla p_H + \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_H + \mathbf{f} \quad (\text{A.6})$$

$$\text{2-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{u}_h^O - \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_h^O = -\nabla p_h^O + \mathbf{f} \quad (\text{A.7})$$

$$\begin{aligned} \text{3-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{e}_H^{\text{OC}} - \frac{1}{\text{Re}} \nabla^2 \mathbf{e}_H^{\text{OC}} + (\mathbf{e}_H^{\text{OC}} \cdot \nabla) \mathbf{u}_H \\ = -\nabla \varepsilon_H^{\text{OC}} + \mathbf{f} + [(\mathbf{u}_H - \mathbf{u}^O) \cdot \nabla] \mathbf{u}_H \end{aligned} \quad (\text{A.8})$$

Modified Newton method (Newton two-step method plus correction step)

$$\text{1-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{u}_H = -\nabla p_H + \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_H + \mathbf{f} \quad (\text{A.9})$$

$$\begin{aligned} \text{2-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{u}_h^N - \frac{1}{\text{Re}} \nabla^2 \mathbf{u}_h^N + (\mathbf{u}_h^N \cdot \nabla) \mathbf{u}_H \\ = -\nabla p_h^N + (\mathbf{u}_H \cdot \nabla) \mathbf{u}_H + \mathbf{f} \end{aligned} \quad (\text{A.10})$$

$$\begin{aligned} \text{3-step: } (\mathbf{u}_H \cdot \nabla) \mathbf{e}_H^{\text{NC}} - \frac{1}{\text{Re}} \nabla^2 \mathbf{e}_H^{\text{NC}} + (\mathbf{e}_H^{\text{NC}} \cdot \nabla) \mathbf{u}_H \\ = -\nabla \varepsilon_H^{\text{NC}} + \mathbf{f} + [(\mathbf{u}_H - \mathbf{u}^N) \cdot \nabla] (\mathbf{u}^N - \mathbf{u}_H) \end{aligned} \quad (\text{A.11})$$