Journal of Computational Mathematics, Vol.20, No.1, 2002, 35–56.

ON DISCRETE PROJECTION AND NUMERICAL BOUNDARY CONDITIONS FOR THE NUMERICAL SOLUTION OF THE UNSTEADY INCOMPRESSIBLE NAVIER-STOKES EQUATIONS[*1)]

Lan Chieh Huang
(*ICMSEC, Academy of Mathematics and System Sciences, Chinese Academy of Sciences, Beijing 100080, China*)

Abstract

The unsteady incompressible Navier-Stokes equations are discretized in space and studied on the fixed mesh as a system of differential algebraic equations. With discrete projection defined, the local errors of Crank Nicholson schemes with three projection methods are derived in a straightforward manner. Then the approximate factorization of relevant matrices are used to study the time accuracy with more detail, especially at points adjacent to the boundary. The effects of numerical boundary conditions for the auxiliary velocity and the discrete pressure Poisson equation on the time accuracy are also investigated. Results of numerical experiments with an analytic example confirm the conclusions of our analysis.

Key words: Differential algebraic equations, Discrete projection, Numerical boundary conditions.

1. Introduction

Let us consider the unsteady incompressible Navier–Stokes equations (INSE)

$$\frac{\partial \boldsymbol{w}}{\partial t} + (\boldsymbol{w} \cdot grad)\boldsymbol{w} + grad\ p = \frac{1}{\mathsf{Re}} div\ grad\ \boldsymbol{w} + \mathbf{f} \tag{1.1}$$

$$div\ \boldsymbol{w} = 0 \tag{1.2}$$

on a two–dimensional rectangular region Ω with boundary $\partial\Omega$. Here $\boldsymbol{w} = (u, v)^T$ is the velocity vector; p is the pressure; and $\mathbf{f}$ a known vector function of $x, y,$ and t. The initial condition is given as

$$\boldsymbol{w}|_{t=0} = \mathbf{w}^0 \quad \text{on } \Omega \tag{1.3}$$

satisfying (1.2). We are concerned mainly with the solid wall boundary condition

$$\boldsymbol{w} = \mathbf{w}_B \quad \text{on } \partial\Omega \quad \text{satisfying } \oint_{\partial\Omega} \mathsf{w}_{Bn} ds = 0 \tag{1.4}$$

The difficulty in the numerical solution of the above problem lies in that (1.1) and (1.2) are partial differential equations with constraint; i.e., the system of equations is not entirely evolutionary. The projection methods of Temam [23], Chorin [2], and van Kan [25] have been

* Received June 2, 1999.

[1)] Project 19772056 supported by the National Natural Science Foundation of China and the Laboratory of Scientific and Engineering Computing of the Institute of Computational Mathematics, the Chinese Academy of Sciences.

widely used and have proven to be most efficient for this type of problems. However, it is not always well understood and many of its problems need further investigation and rigorous analysis. Historically, Temam [23] gave the convergence proofs for the proposed methods with full discretization, but only Chorin [2] gave the error estimate for his proposed method with full discretization. Then with recent interest, great progress has been made in the study of errors of projection methods; for example: the works of E and Liu [4, 5], Orszag, Israeli, and Deville [16], Shen [20, 21], Rannacher [19], and Hou and Wetton [9, 26]. Significant as these mathematical papers are, many analyses are done for the Stokes equations or for the semi-discrete INSE (space continuous), and some with explicit approximation for convection. It is the author's opinion that much work is required before the error estimation of a projection method for a fully discretized INSE become easily accessible to the computational community.

It has been the author's attempt to contribute in this direction using simple mathematical tools familiar to the computational fluid dynamics community. With spatial discretization on a *fixed* mesh, the INSE become a system of differential algebraic equations (DAE), for which the local errors of a numerical method can be quite different from its counterpart for the ordinary differential equation, see [7] for example. Further errors are introduced with the projection method for the system derived from INSE. The authors [11, 12] defined discrete projection with the minimum requirement as that needed for the projection step in numerical solution of the system from INSE. With the projection operators, the derivation of local errors of the velocity and the pressure gradient for projection methods becomes straightforward with Taylor series. This paper studies in particular the fully implicit (for convection and viscosity) Crank Nicholson (CN) schemes, mainly CN2 to be described below, and three projection methods: pressure correction (PC) studied thoroughly by van Kan in [25], pressure (PR) of the earlier projection papers and the present version here of Kim and Moin [15], and component-consistent pressure correction (CCPC) proposed by the authors in [11] for its approximate preservation of component-consistency under projection. This version has been used by Bell, Colella, and Glaz with a different projection procedure in [1], and interpreted as the present version by [4].

The global errors on the fixed mesh follow from the local errors, as for the general DAE [11], with correct interpretation of the assumption that the right hand side functions have bounded derivatives in some closed region of our interest; but convergence for a finite time interval is almost trivial, as it is for the ordinary differential equations, and gives little information to problems of INSE as partial differential equations. To gain some insight into this type of problems, the local errors are analyzed with approximate factorization (AF) of relevant matrices as Yanenko [27], Perot [17, 18], and the author [10], with special attention to the discrete approximation on points adjacent to the boundary. From our analysis we can see, for example, the reason why an improvement in the numerical boundary condition (NBC) for the auxiliary velocity over just (1.4) can lead to an increase of an order in the accuracy of the velocity, e.g. the Kim and Moin method in [15]. Also several NBCs for the auxiliary velocity and the discrete pressure Poisson equation frequently stated in literature will be investigated and clarified in terms of discrete projection.

In Section 2, the discrete projection will be stated and two Crank Nicholson (CN) schemes for the DAE formed from INSE will be given. Three projection methods based on these schemes: PC, PR, and CCPC will be described and their local errors of the velocity and the pressure gradient will be briefly derived in Section 3. Then these errors will be studied more carefully with the AF method in Section 4. Several NBCs frequently stated in literature will be summarized in terms of discrete projection in Section 5. Finally, in Section 6, the results of numerical experiment with an analytic example, on the staggered mesh for simplicity, will be given. These results confirm the conclusions of our analysis.

Here a word on the notation of this paper is in order: boldface ($\boldsymbol{Z}$) denotes a "double"

vector with components of vector ($\boldsymbol{z} = (z^1, z^2)^T$) on discrete points of the computational region, underline ($\underline{Z}$) denotes a matrix or a linear operator, sans serif (Z, z) denotes known quantities, and caligraphic ($\mathcal{Z}$) denotes a "triple" vector with ($\underline{\mathcal{Z}}$) the associated matrix; other symbols are as defined.

2. From INSE to DAE CN Schemes

Let the computational region be covered by a mesh with interior velocity points I, boundary points B, and interior pressure points I_0. For the staggered mesh, I consists of I_u and I_v, with $N_2 = N_u + N_v$ number of points, where the former are the interior u points and the latter the interior v points; similarly for B. I_0 consists of M number of points. Upon spatial discretization, (1.1), (1.2), and (1.4) can be written as

$$\frac{d\boldsymbol{w}}{dt} + A(\boldsymbol{w}) + Gp = \mathbf{f}, \quad \text{on } I \tag{2.1}$$

$$\frac{d\boldsymbol{w}}{dt} - \mathbf{w}'_B = 0, \quad \text{on } B \tag{2.2}$$

$$D\boldsymbol{w} = 0, \quad \text{on } I_0 \tag{2.3}$$

where A is a nonlinear operator with $A(\boldsymbol{w})$ approximating the convection and viscosity terms; G and D are linear operators approximating respectively *grad* and *div*. On a fixed mesh, A is a nonlinear vector function; G and D are linear functions; the independent variables are the $\boldsymbol{w}$'s and the p's on the neighboring points. Then (2.1)–(2.3) form DAE

$$\frac{d\boldsymbol{W}}{dt} + \boldsymbol{F}(\boldsymbol{W}, t) + \underline{G}P = 0 \tag{2.4a}$$

$$\underline{D}\boldsymbol{W} = 0 \tag{2.4b}$$

The components of vector $\boldsymbol{W} = (\boldsymbol{W}_I, \boldsymbol{W}_B)^T$ are the components of $\boldsymbol{w}$'s on I and B points; The components of vector P are the p's on I_0; and $\boldsymbol{F}$ is a nonlinear vector function, with components $A(\boldsymbol{w}) - \mathbf{f}$ on I and $-\mathbf{w}'_B$ on B. In general

$$\underline{G}P = \begin{pmatrix} \underline{G_I} \\ 0 \end{pmatrix} P, \quad \underline{D}\boldsymbol{W} = (\underline{D_I}, \underline{D_B}) \begin{pmatrix} \boldsymbol{W}_I \\ \boldsymbol{W}_B \end{pmatrix}$$

where $\underline{G_I}$ is a $N_2 \times M$ matrix, i.e. Gp on the boundary is not explicitly involved; and where $\underline{D_I}$ is a $M \times N_2$ matrix. (2.4) leads to

$$\underline{DG}P = \underline{D_I G_I} P = -\underline{D}\boldsymbol{F}(\boldsymbol{W}, t) \tag{2.5}$$

In terms of DAE, this is the consistency condition between the components of the solution vector of (2.4), $\boldsymbol{W}$ and P.

We state here the discrete projection theorem of [12]. It's proof is almost trivial, but it can be applied to many practical situations, being based only on the condition for the solution of the discrete Poisson equation in the projection method used.

Discrete Projection Theorem. *Any vector $\boldsymbol{V}$ of a vector space $\mathcal{V}$ can be uniquely decomposed into*

$$\boldsymbol{V} = \boldsymbol{U} + \underline{G}\Phi \tag{2.6}$$

where $\underline{D}\boldsymbol{U} = 0$, if the system of linear algebraic equations

$$\underline{DG}\Phi = \underline{D}\boldsymbol{V} \tag{2.7}$$

has a solution Φ, with $\underline{G}\Phi$ unique.

Indeed, $\underline{D}$(2.6) yields (2.7); from the solution of the latter, $\underline{G}\Phi$ is obtained and hence $\boldsymbol{U} = \boldsymbol{V} - \underline{G}\Phi$; thus the existence of the decomposition. From the uniqueness of $\underline{G}\Phi$, results the uniqueness of the decomposition.

From the unique decomposition, we define projection $\underline{P} : \mathcal{V} \to \mathcal{V}$ such that $\underline{P}\boldsymbol{V} = \boldsymbol{U}$, and $\underline{Q} = \underline{I} - \underline{P}$. So

$$\begin{array}{rclcrcl} \underline{P}\boldsymbol{U} & = & \boldsymbol{U}, & & \underline{P}(\underline{G}\Phi) & = & 0 \\ \underline{Q}\boldsymbol{U} & = & 0, & & \underline{Q}(\underline{G}\Phi) & = & \underline{G}\Phi \end{array} \qquad (\mathbf{P})$$

Let $\mathcal{D}$ denote the subspace of $\mathcal{V}$ of $\boldsymbol{U}$ with $\underline{D}\boldsymbol{U} = 0$, and $\mathcal{G}$ the subspace of $\mathcal{V}$ of form $\underline{G}\Phi$. $\underline{P}$ is the projection of $\mathcal{V}$ on $\mathcal{D}$ along $\mathcal{G}$; and $\underline{Q}$ is the projection of $\mathcal{V}$ on $\mathcal{G}$ along $\mathcal{D}$, see [8]. Now let $\boldsymbol{V}$ be $-\boldsymbol{F}(\boldsymbol{W}, t)$, $\boldsymbol{U}$ be $\frac{d\boldsymbol{W}}{dt}$, and $\underline{G}\Phi$ be $\underline{G}P$, then (2.7) becomes (2.5). When $\boldsymbol{W}$ and t are known, (2.5) is a system of linear algebraic equations; it is also called the discrete Poisson equation (DPE) of the pressure, with $\underline{DG}$ the discrete Laplacian. For the following projection methods, the DPEs (3.5c), (3.6c), and (3.7c) are also of this form. In Section 5, the validity of the assumption of the Discrete Projection Theorem will be discussed. Here we simply assume the existence of solution P, with $\underline{G}P$ unique, then the discrete projection holds. $\underline{P}$ (2.4) and $\underline{Q}$ (2.4) gives respectively, using (**P**),

$$\frac{d\boldsymbol{W}}{dt} + \underline{P}\boldsymbol{F}(\boldsymbol{W}, t) = 0 \qquad (2.8)$$

$$\underline{G}P = -\underline{Q}\boldsymbol{F}(\boldsymbol{W}, t) \qquad (2.9)$$

The CN time approximation of DAE (2.4) can be written as CN1:

$$\frac{\boldsymbol{W}^{n+1} - \boldsymbol{W}^n}{\Delta t} + \boldsymbol{H}(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}, t^n, t^{n+1}) + \frac{1}{2}(\underline{G}P^n + \underline{G}P^{n+1}) = 0 \qquad (2.10a)$$

$$\underline{D}\boldsymbol{W}^{n+1} = 0 \qquad (2.10b)$$

where $\boldsymbol{H}$ approximates $A(\boldsymbol{w}) - \mathbf{f}$ on I and $-\frac{1}{\Delta t}\big(\mathbf{w}_B(t^{n+1}) - \mathbf{w}_B(t^n)\big)$ on B. If $\boldsymbol{W}^n = \mathbf{W}_B(t^n)$ on B, then

$$\boldsymbol{W}^{n+1} = \mathbf{W}_B(t^{n+1}) \qquad (2.11)$$

The CN time approximation of DAE (2.4) can also be written as CN2:

$$\frac{\boldsymbol{W}^{n+1} - \boldsymbol{W}^n}{\Delta t} + \boldsymbol{H}(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}, t^n, t^{n+1}) + \underline{G}P^{n+\frac{1}{2}} = 0 \qquad (2.12a)$$

$$\underline{D}\boldsymbol{W}^{n+1} = 0 \qquad (2.12b)$$

First we assume that $\boldsymbol{H}(\boldsymbol{W}_1, \boldsymbol{W}_2, t_1, t_2)$ is sufficiently differentiable "in the closed region under consideration", as generally done. Denote the Jacobian $\frac{\partial \boldsymbol{H}}{\partial \boldsymbol{W}_1}$ as $\boldsymbol{H}_{\boldsymbol{W}_1}$, $\frac{\partial \boldsymbol{H}}{\partial \boldsymbol{W}_2}$ as $\boldsymbol{H}_{\boldsymbol{W}_2}$; and vector $\frac{\partial \boldsymbol{H}}{\partial t_1}$ as $\boldsymbol{H}_{t_1}$, $\frac{\partial \boldsymbol{H}}{\partial t_2}$ as $\boldsymbol{H}_{t_2}$. Then we assume that

$$\begin{array}{rcll} \boldsymbol{H}(\boldsymbol{W}, \boldsymbol{W}, t, t) & = & \boldsymbol{F}(\boldsymbol{W}, t) & \text{and} \\ \boldsymbol{H}_{\boldsymbol{W}_1} = \boldsymbol{H}_{\boldsymbol{W}_2} = \boldsymbol{H}_{\boldsymbol{W}} & = & \frac{1}{2}\boldsymbol{F}_{\boldsymbol{W}} & \text{when } \boldsymbol{W}_1 = \boldsymbol{W}_2 = \boldsymbol{W} \\ \boldsymbol{H}_{t_1} = \boldsymbol{H}_{t_2} = \boldsymbol{H}_t & = & \frac{1}{2}\boldsymbol{F}_t & \text{when } t_1 = t_2 = t \end{array} \qquad (\mathbf{A})$$

For the Hessians and the various "second derivatives" of $\boldsymbol{H}$, we consider the following cases, for which the "mixed derivatives" in $\boldsymbol{W}$ and t are 0:

Case 1. $\boldsymbol{F}(\boldsymbol{W}, t) = \boldsymbol{A}(\boldsymbol{W}) + \mathbf{F}(t)$ and $\boldsymbol{H}(\boldsymbol{W}_1, \boldsymbol{W}_2, t_1, t_2) = \frac{1}{2}\big(\boldsymbol{A}(\boldsymbol{W}_1) + \boldsymbol{A}(\boldsymbol{W}_2) + \mathbf{F}(t_1) + \mathbf{F}(t_2)\big)$. We have

$$\boldsymbol{H}_{\boldsymbol{W}_2}(\boldsymbol{W}_1, \boldsymbol{W}_2, t_1, t_2) = \frac{1}{2}\boldsymbol{A}_{\boldsymbol{W}}(\boldsymbol{W}_2), \qquad \boldsymbol{H}_{t_2}(\boldsymbol{W}_1, \boldsymbol{W}_2, t_1, t_2) = \frac{1}{2}\mathbf{F}'(t_2)$$

$$\boldsymbol{H}_{\boldsymbol{W}_2\boldsymbol{W}_2}(\boldsymbol{W}_1, \boldsymbol{W}_2, t_1, t_2) = \frac{1}{2}\boldsymbol{A}_{\boldsymbol{W}\boldsymbol{W}}(\boldsymbol{W}_2), \quad \boldsymbol{H}_{t_2t_2}(\boldsymbol{W}_1, \boldsymbol{W}_2, t_1, t_2) = \frac{1}{2}\mathbf{F}''(t_2)$$

Case 2. $\boldsymbol{F}(\boldsymbol{W},t)=\boldsymbol{A}(\boldsymbol{W})+\mathsf{F}(t)$ but $\boldsymbol{H}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\boldsymbol{A}(\frac{1}{2}\boldsymbol{W}_1+\frac{1}{2}\boldsymbol{W}_2)+\mathsf{F}(\frac{1}{2}t_1+\frac{1}{2}t_2)$. We have

$$\begin{aligned}
&\boldsymbol{H}_{\boldsymbol{W}_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{2}\boldsymbol{A}_{\boldsymbol{W}}(\frac{1}{2}\boldsymbol{W}_1+\frac{1}{2}\boldsymbol{W}_2), && \boldsymbol{H}_{t_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{2}\mathsf{F}'(\frac{1}{2}t_1+\frac{1}{2}t_2)\\
&\boldsymbol{H}_{\boldsymbol{W}_2\boldsymbol{W}_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{4}\boldsymbol{A}_{\boldsymbol{W}\boldsymbol{W}}(\frac{1}{2}\boldsymbol{W}_1+\frac{1}{2}\boldsymbol{W}_2), && \boldsymbol{H}_{t_2t_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{4}\mathsf{F}''(\frac{1}{2}t_1+\frac{1}{2}t_2)
\end{aligned}$$

When $\boldsymbol{F}(\boldsymbol{W},t)=\underline{L}\boldsymbol{W}+\mathsf{F}(t)$, $\underline{L}$ a linear funtion, then (2.4a) becomes

$$\frac{d\boldsymbol{W}}{dt}+\underline{L}\boldsymbol{W}+\mathsf{F}(t)+\underline{G}P=0 \tag{2.13}$$

and $\boldsymbol{F}_{\boldsymbol{W}}(\boldsymbol{W},t)=\underline{L},\boldsymbol{F}_t(\boldsymbol{W},t)=\mathsf{F}'(t),\boldsymbol{F}_{\boldsymbol{W}\boldsymbol{W}}(\boldsymbol{W},t)=0$, and $\boldsymbol{F}_{tt}(\boldsymbol{W},t)=\mathsf{F}''(t)$. If $\underline{P}$ and $\underline{Q}$ commute with $\underline{L}$, then in place of (2.8) and (2.9), we have

$$\frac{d\boldsymbol{W}}{dt}+\underline{L}\boldsymbol{W}+\underline{P}\mathsf{F}(t)=0 \tag{2.14}$$

$$\underline{G}P=-\underline{Q}\mathsf{F}(t) \tag{2.15}$$

Then, parallel to Case 1 and Case 2, we consider respectively:

Case 1'. $\boldsymbol{H}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{2}\big(\underline{L}\boldsymbol{W}_1+\underline{L}\boldsymbol{W}_2\big)+\frac{1}{2}\big(\mathsf{F}(t_1)+\mathsf{F}(t_2)\big)$. We have

$$\begin{aligned}
&\boldsymbol{H}_{\boldsymbol{W}_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{2}\underline{L}, && \boldsymbol{H}_{t_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{2}\mathsf{F}'(t_2)\\
&\boldsymbol{H}_{\boldsymbol{W}_2\boldsymbol{W}_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=0, && \boldsymbol{H}_{t_2t_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{2}\mathsf{F}''(t_2)
\end{aligned}$$

Case 2'. $\boldsymbol{H}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\underline{L}(\frac{1}{2}\boldsymbol{W}_1+\frac{1}{2}\boldsymbol{W}_2)+\mathsf{F}(\frac{1}{2}t_1+\frac{1}{2}t_2)$. We have

$$\begin{aligned}
&\boldsymbol{H}_{\boldsymbol{W}_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{2}\underline{L}, && \boldsymbol{H}_{t_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{2}\mathsf{F}'(\tfrac{1}{2}t_1+\tfrac{1}{2}t_2)\\
&\boldsymbol{H}_{\boldsymbol{W}_2\boldsymbol{W}_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=0, && \boldsymbol{H}_{t_2t_2}(\boldsymbol{W}_1,\boldsymbol{W}_2,t_1,t_2)=\frac{1}{4}\mathsf{F}''(\tfrac{1}{2}t_1+\tfrac{1}{2}t_2)
\end{aligned}$$

3. Local Errors of CN Projection Methods

We start by discussing the local error of the basic CN1 and CN2 schemes. For CN1 and Case 1, it is known that the local errors for both $\boldsymbol{W}$ and $\underline{G}P$ are $O(\Delta t^3)$ (i.e. second order global error), e.g. see [7]–Lobatto IIIA method for DAE with index 2. Here for DAE (2.4), let $\boldsymbol{W}(t^n)$ and $\underline{G}P(t^n)$ be the exact solution at t^n. From (2.8) and (2.9) we have

$$\begin{aligned}
&\boldsymbol{W}'(t^n)=-\underline{P}\boldsymbol{F}\big(\boldsymbol{W}(t^n),t^n\big), && \boldsymbol{W}''(t^n)=-\underline{P}\big(\boldsymbol{F}_{\boldsymbol{W}}\big(\boldsymbol{W}(t^n),t^n\big)\boldsymbol{W}'(t^n)+\boldsymbol{F}_t\big(\boldsymbol{W}(t^n),t^n\big)\big)\\
&\underline{G}P(t^n)=-\underline{Q}\boldsymbol{F}\big(\boldsymbol{W}(t^n),t^n\big), && \underline{G}P'(t^n)=-\underline{Q}\big(\boldsymbol{F}_{\boldsymbol{W}}\big(\boldsymbol{W}(t^n),t^n\big)\boldsymbol{W}'(t^n)+\boldsymbol{F}_t\big(\boldsymbol{W}(t^n),t^n\big)\big)\\
&\underline{G}P''(t^n)=-\underline{Q}\Big(\boldsymbol{F}_{\boldsymbol{W}\boldsymbol{W}}\big(\boldsymbol{W}(t^n),t^n\big)\big(\boldsymbol{W}'(t^n)\big)^2+\boldsymbol{F}_{\boldsymbol{W}}\big(\boldsymbol{W}(t^n),t^n\big)\boldsymbol{W}''(t^n)+\boldsymbol{F}_{tt}\big(\boldsymbol{W}(t^n),t^n\big)\\
&\qquad\qquad +2\boldsymbol{F}_{\boldsymbol{W}t}\big(\boldsymbol{W}(t^n),t^n\big)\boldsymbol{W}'(t^n)\Big)
\end{aligned} \tag{3.1}$$

For CN1, let the numerical solution $\boldsymbol{W}^n=\boldsymbol{W}(t^n)$ and $\underline{G}P^n=\underline{G}P(t^n)$; $\boldsymbol{W}^{n+1}$ and $\underline{G}P^{n+1}$ are functions of Δt. Expand $\boldsymbol{W}(t^n+\tau)$ and $\underline{G}P(t^n+\tau)$ by Taylor series at $\tau=0$ and expand $\boldsymbol{W}^{n+1}(\Delta t)$ and $\underline{G}P^{n+1}(\Delta t)$ by Taylor series at $\Delta t=0$. We then see that when

$$\begin{aligned}
&\boldsymbol{W}^{n+1}(0)=\boldsymbol{W}(t^n), && \dot{\boldsymbol{W}}^{n+1}(0)=\boldsymbol{W}'(t^n), && \ddot{\boldsymbol{W}}^{n+1}(0)=\boldsymbol{W}''(t^n)\\
&\underline{G}P^{n+1}(0)=\underline{G}P(t^n), && \underline{G}\dot{P}^{n+1}(0)=\underline{G}P'(t^n) && \underline{G}\ddot{P}^{n+1}(0)=\underline{G}P''(t^n)
\end{aligned} \tag{3.2}$$

(with the top dot denoting differentiation with respect to Δt) the local errors of $\boldsymbol{W}$ and $\underline{G}P$ are $O(\Delta t^3)$.

Now, $\underline{P}$(2.10a) yields

$$\begin{aligned}
\boldsymbol{W}^{n+1}(\Delta t) &= \boldsymbol{W}^n - \Delta t \underline{P}\boldsymbol{H}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)\\
\Rightarrow \boldsymbol{W}^{n+1}(0) &= \boldsymbol{W}^n = \boldsymbol{W}(t^n)\\
\dot{\boldsymbol{W}}^{n+1}(\Delta t) &= -\underline{P}\Big(\boldsymbol{H}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big) + \Delta t\dot{\boldsymbol{H}}\Big)\\
\Rightarrow \dot{\boldsymbol{W}}^{n+1}(0) &= -\underline{P}\boldsymbol{H}\big(\boldsymbol{W}^n, \boldsymbol{W}^n, t^n, t^n\big) = -\underline{P}\boldsymbol{F}\big(\boldsymbol{W}(t^n), t^n\big) = \boldsymbol{W}'(t^n)\\
\ddot{\boldsymbol{W}}^{n+1}(\Delta t) &= -\underline{P}\Big(2\dot{\boldsymbol{H}} + \Delta t\ddot{\boldsymbol{H}}\Big)\\
\Rightarrow \ddot{\boldsymbol{W}}^{n+1}(0) &= -\underline{P}\Big(2\boldsymbol{H}_{\boldsymbol{W}_2}\big(\boldsymbol{W}^n, \boldsymbol{W}^n, t^n, t^n\big)\dot{\boldsymbol{W}}^{n+1}(0) + 2\boldsymbol{H}_{t_2}\big(\boldsymbol{W}^n, \boldsymbol{W}^n, t^n, t^n\big)\Big)\\
&= -\underline{P}\Big(\boldsymbol{F}_{\boldsymbol{W}}\big(\boldsymbol{W}(t^n), t^n\big)\boldsymbol{W}'(t^n) + \boldsymbol{F}_t\big(\boldsymbol{W}(t^n), t^n\big)\Big) = \boldsymbol{W}''(t^n)
\end{aligned} \tag{3.3}$$

where $\dot{\boldsymbol{H}} = \boldsymbol{H}_{\boldsymbol{W}_2}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)\dot{\boldsymbol{W}}^{n+1}(\Delta t) + \boldsymbol{H}_{t_2}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)$. Up to now, only Assumption (**A**) has been used, and the content of $\ddot{\boldsymbol{H}}$ has not mattered, as long as it is bounded as $\Delta t \to 0$. However, $\underline{Q}$(2.10a) results in

$$\begin{aligned}
\underline{G}P^{n+1}(\Delta t) &= -\underline{G}P^n - 2\underline{Q}\boldsymbol{H}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)\\
\Rightarrow \underline{G}P^{n+1}(0) &= -\underline{G}P^n - 2\underline{Q}\boldsymbol{F}\big(\boldsymbol{W}(t^n), t^n\big) = \underline{G}P(t^n)\\
\underline{G}\dot{P}^{n+1}(\Delta t) &= -2\underline{Q}\dot{\boldsymbol{H}}\\
\Rightarrow \underline{G}\dot{P}^{n+1}(0) &= -\underline{Q}\Big(\boldsymbol{F}_{\boldsymbol{W}}\big(\boldsymbol{W}(t^n), t^n\big)\boldsymbol{W}'(t^n) + \boldsymbol{F}_t\big(\boldsymbol{W}(t^n), t^n\big)\Big) = \underline{G}P'(t^n)\\
\underline{G}\ddot{P}^{n+1}(\Delta t) &= -2\underline{Q}\ddot{\boldsymbol{H}} = -2\underline{Q}\Big(\boldsymbol{H}_{\boldsymbol{W}_2\boldsymbol{W}_2}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)\big(\dot{\boldsymbol{W}}^{n+1}(\Delta t)\big)^2\\
&\quad + \boldsymbol{H}_{\boldsymbol{W}_2}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)\ddot{\boldsymbol{W}}^{n+1}(\Delta t)\\
&\quad + \boldsymbol{H}_{t_2t_2}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)\\
&\quad + 2\boldsymbol{H}_{\boldsymbol{W}_2t_2}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)\dot{\boldsymbol{W}}^{n+1}(\Delta t)\Big)
\end{aligned} \tag{3.4}$$

We note first that under Assumption (**A**), the local error for $\boldsymbol{W}$ is $O(\Delta t^3)$ and for $\underline{G}P$ is $O(\Delta t^2)$. We also note that an additional $O(\Delta t^2)$ term in the first equation of CN1 will not effect this result. This is due to the fact that $\boldsymbol{W}^{n+1}(\Delta t)$, $\dot{\boldsymbol{W}}^{n+1}(\Delta t)$, and $\ddot{\boldsymbol{W}}^{n+1}(\Delta t)$ will have respectively an $O(\Delta t^3)$, $O(\Delta t^2)$, and $O(\Delta t)$ term; while $\underline{G}P^{n+1}(\Delta t)$ and $\underline{G}\dot{P}^{n+1}(\Delta t)$ will have respectively an $O(\Delta t^2)$ and $O(\Delta t)$ term; all reducing to 0 as $\Delta t \to 0$. Now we observe the last equation of (3.4), and see that the content of $\ddot{\boldsymbol{H}}$ is relevant to the order of accuracy of $\underline{G}P$. For Case 1,

$$\begin{aligned}
\underline{G}\ddot{P}^{n+1}(0) &= -\underline{Q}\Big(\boldsymbol{F}_{\boldsymbol{W}\boldsymbol{W}}\big(\boldsymbol{W}(t^n), t^n\big)\big(\boldsymbol{W}'(t^n)\big)^2 + \boldsymbol{F}_{\boldsymbol{W}}\big(\boldsymbol{W}(t^n), t^n\big)\boldsymbol{W}''(t^n) + \boldsymbol{F}_{tt}\big(\boldsymbol{W}(t^n), t^n\big)\Big)\\
&= \underline{G}P''(t^n)
\end{aligned}$$

and the local errors for both $\boldsymbol{W}$ and $\underline{G}P$ are $O(\Delta t^3)$; this is in agreement with the known result.

For CN2, under Assumption (**A**), $\underline{P}$(2.12a) yields precisely (3.3), so the local error for $\boldsymbol{W}$ is again $O(\Delta t^3)$. $\underline{Q}$(2.12a) yields $\underline{G}P^{n+\frac{1}{2}}(\Delta t) = \underline{Q}\boldsymbol{H}\big(\boldsymbol{W}^n, \boldsymbol{W}^{n+1}(\Delta t), t^n, t^n+\Delta t\big)$. Proceeding as (3.4) and comparing the results with the terms in the expansion of $\underline{G}P(t^n+\tau)$, $\tau = \Delta t/2$, we see that the local error for $\underline{G}P$ is again $O(\Delta t^2)$. However for Case 2', it can be shown that the local error for $\underline{G}P$ is $O(\Delta t^3)$; for details, see [13]. From the practical point of view, the local error of $\underline{G}P$ itself is not important; since once $\boldsymbol{W}$ is known, $\underline{G}P$ can always be found from the component consistency condition (2.5).

Now we consider the following three projection methods: CN1+PC (based on CN1 and the

pressure correction projection method as [25]):

$$\frac{\widetilde{\boldsymbol{W}}^{n+1}-\boldsymbol{W}^{n}}{\Delta t}+\boldsymbol{H}\big(\boldsymbol{W}^{n},\widetilde{\boldsymbol{W}}^{n+1}(\Delta t),t^{n},t^{n}+\Delta t\big)+\underline{G}P^{n}=0 \tag{3.5a}$$

$$\frac{\boldsymbol{W}^{n+1}-\widetilde{\boldsymbol{W}}^{n+1}}{\Delta t}+\frac{1}{2}\underline{G}\Phi=0\quad\text{where }\Phi=P^{n+1}-P^{n} \tag{3.5b}$$

$$\underline{D}\boldsymbol{W}^{n+1}=0\quad\Longrightarrow\quad\underline{DG}\Phi=\frac{2}{\Delta t}\,\underline{D}\widetilde{\boldsymbol{W}}^{n+1} \tag{3.5c}$$

CN2+PR (based on CN2 and the pressure projection method as [15]):

$$\frac{\widetilde{\boldsymbol{W}}^{n+1}-\boldsymbol{W}^{n}}{\Delta t}+\boldsymbol{H}\big(\boldsymbol{W}^{n},\widetilde{\boldsymbol{W}}^{n+1}(\Delta t),t^{n},t^{n}+\Delta t\big)=0 \tag{3.6a}$$

$$\frac{\boldsymbol{W}^{n+1}-\widetilde{\boldsymbol{W}}^{n+1}}{\Delta t}+\underline{G}P^{n+\frac{1}{2}}=0 \tag{3.6b}$$

$$\underline{D}\boldsymbol{W}^{n+1}=0\quad\Longrightarrow\quad\underline{DG}P^{n+\frac{1}{2}}=\frac{1}{\Delta t}\,\underline{D}\widetilde{\boldsymbol{W}}^{n+1} \tag{3.6c}$$

CN2+CCPC (based on CN2 and the component-consistent pressure projection method as [11]):

$$\frac{\widetilde{\boldsymbol{W}}^{n+1}-\boldsymbol{W}^{n}}{\Delta t}+\boldsymbol{H}\big(\boldsymbol{W}^{n},\widetilde{\boldsymbol{W}}^{n+1}(\Delta t),t^{n},t^{n}+\Delta t\big)+\underline{G}P^{n-\frac{1}{2}}=0 \tag{3.7a}$$

$$\frac{\boldsymbol{W}^{n+1}-\widetilde{\boldsymbol{W}}^{n+1}}{\Delta t}+\underline{G}\Phi=0\quad\text{where }\Phi=P^{n+\frac{1}{2}}-P^{n-\frac{1}{2}} \tag{3.7b}$$

$$\underline{D}\boldsymbol{W}^{n+1}=0\quad\Longrightarrow\quad\underline{DG}\Phi=\frac{1}{\Delta t}\,\underline{D}\widetilde{\boldsymbol{W}}^{n+1} \tag{3.7c}$$

for $n=0$, $\underline{G}P^{-\frac{1}{2}}$ can be just $\underline{G}P(t^0)$. Note that for all the above projection methods, on the B points

$$\frac{\widetilde{\boldsymbol{W}}_B^{n+1}-\boldsymbol{W}_B^{n}}{\Delta t}-\frac{\mathbf{W}_B(t^{n+1})-\mathbf{W}_B(t^{n})}{\Delta t}=0\implies\widetilde{\boldsymbol{W}}_B^{n+1}=\mathbf{W}_B(t^{n+1}) \tag{3.8a}$$

$$\frac{\boldsymbol{W}_B^{n+1}-\widetilde{\boldsymbol{W}}_B^{n+1}}{\Delta t}=0\implies\boldsymbol{W}_B^{n+1}=\widetilde{\boldsymbol{W}}_B^{n+1}=\mathbf{W}_B(t^{n+1}) \tag{3.8b}$$

thus satisfying (2.11).

The method of derivation of local errors for the projection methods is essentially the same as for the basic schemes, and for CN2+CCPC it is given in detail in [11]. Briefly, $\widetilde{\boldsymbol{W}}^{n+1}(0)$, $\dot{\widetilde{\boldsymbol{W}}}^{n+1}(0)$, and $\ddot{\widetilde{\boldsymbol{W}}}^{n+1}(0)$ are derived first from the (a) equations, then $\boldsymbol{W}^{n+1}(0)$, $\dot{\boldsymbol{W}}^{n+1}(0)$, and $\ddot{\boldsymbol{W}}^{n+1}(0)$, are obtained with $\boldsymbol{W}^{n+1}=\underline{P}\widetilde{\boldsymbol{W}}^{n+1}$, which results from applying $\underline{P}$ to the (b) equations. The expression for $\underline{G}P^{n+1}$ or $\underline{G}P^{n+\frac{1}{2}}$ are found by adding the (a) and (b) equations and applying $\underline{Q}$. For details, see [13]. Here we simply state the results:
For CN1+PC, the local $O(\Delta t^3)$ and $O(\Delta t^2)$ errors of $\boldsymbol{W}$ and $\underline{G}P$ respectively are retained, with no improvement for $\underline{G}P$ even in Case 1.
For CN2+PR, $\widetilde{\boldsymbol{W}}^{n+1}(0)=\boldsymbol{W}(t^n)$, $\dot{\widetilde{\boldsymbol{W}}}^{n+1}(0)=-\boldsymbol{F}\big(\boldsymbol{W}(t^n),t^n\big)\big)\neq\boldsymbol{W}'(t^n)$, and $\ddot{\widetilde{\boldsymbol{W}}}^{n+1}(0)=-\Big(\boldsymbol{F}_{\boldsymbol{W}}\big(\boldsymbol{W}(t^n),t^n\big)\dot{\widetilde{\boldsymbol{W}}}^{n+1}(0)+\boldsymbol{F}_t\big(\boldsymbol{W}(t^n),t^n\big)\Big)$. The local error of $\boldsymbol{W}$ is only $O(\Delta t^2)$ in general. But if $\boldsymbol{F}(\boldsymbol{W},t)=\underline{L}\boldsymbol{W}+\mathbf{F}(t)$ and $\underline{P}$ and $\underline{Q}$ commute with $\underline{L}$, then with

$$\begin{aligned}&\boldsymbol{W}'(t^n)=-\underline{L}\boldsymbol{W}(t^n)-\underline{P}\mathbf{F}(t^n),\quad\boldsymbol{W}''(t^n)=-\underline{L}\boldsymbol{W}'(t^n)-\underline{P}\mathbf{F}'(t^n)\\&\underline{G}P(t^n)=-\underline{Q}\mathbf{F}(t^n),\quad\underline{G}P'(t^n)=-\underline{Q}\mathbf{F}'(t^n),\quad\underline{G}P''(t^n)=-\underline{Q}\mathbf{F}''(t^n)\end{aligned} \tag{3.9}$$

we get $\ddot{\boldsymbol{W}}^{n+1}(0) = -\underline{L}\boldsymbol{W}'(t^n) - \underline{P}\mathsf{F}'(t^n) = \boldsymbol{W}''(t^n)$. and so the local error of $\boldsymbol{W}$ is $O(\Delta t^3)$. However, the local error of $\underline{G}P$ is only $O(\Delta t)$. We note here that if $\boldsymbol{F}(\boldsymbol{W},t) = \boldsymbol{N}(\boldsymbol{W}) + \underline{L}\boldsymbol{W} + \mathsf{F}(t)$, where $\boldsymbol{N}$ is nonlinear, and $\boldsymbol{N}(\mathsf{W}^{n+1/2})$ in (3.6a) is a known function with $\mathsf{W}^{n+1/2}(\Delta t) = \boldsymbol{W}(t^n) + \frac{\Delta t}{2}\boldsymbol{W}'(t^n) + O(\Delta t^2)$, then this term can be regarded as an $\mathsf{F}(t^n + \frac{\Delta t}{2}) = \mathsf{F}(\frac{1}{2}t^n + \frac{1}{2}t^{n+1})$ satisfying Assumption (**A**) and hence the local error of $\boldsymbol{W}$ is $O(\Delta t^3)$. That is, projection dealing with only the linear part of $\boldsymbol{H}$ does not impair the accuracy of $\boldsymbol{W}$.

For CN2+CCPC, $\widetilde{\boldsymbol{W}}^{n+1}(0) = \boldsymbol{W}(t^n)$, $\dot{\widetilde{\boldsymbol{W}}}^{n+1}(0) = -\boldsymbol{F}\big(\boldsymbol{W}(t^n),t^n\big) - \underline{G}P(t^n) = \boldsymbol{W}'(t^n)$, and $\ddot{\widetilde{\boldsymbol{W}}}^{n+1}(0) = -\Big(\boldsymbol{F}_{\boldsymbol{W}}\big(\boldsymbol{W}(t^n),t^n\big)\boldsymbol{W}'(t^n) + \boldsymbol{F}_t\big(\boldsymbol{W}(t^n),t^n\big)\Big) + 2\underline{G}P'(t^n) \Rightarrow \ddot{\boldsymbol{W}}^{n+1}(0) = \boldsymbol{W}''(t^n)$. Hence it has local error $O(\Delta t^3)$ and $O(\Delta t^2)$ respectively for $\boldsymbol{W}$ and $\underline{G}P$.

4. Local Errors of Projection Methods with the AF Method

In this section we study the local errors of projection methods with more details using the AF of the relevant matrices, see [17, 10]. Assuming the local errors of basic finite difference schemes, the AF errors are simply the projection errors. Let $\boldsymbol{H}$ be linearized as

$$\boldsymbol{H}(\boldsymbol{W}^n,\boldsymbol{W}^{n+1},t^n,t^{n+1}) = \boldsymbol{H}(\boldsymbol{W}^n,\boldsymbol{W}^n,t^n,t^{n+1}) + \boldsymbol{H}_{\boldsymbol{W}_2}(\boldsymbol{W}^n,\boldsymbol{W}^n,t^n,t^{n+1})\dot{\Delta}\boldsymbol{W} + O\big((\dot{\Delta}\boldsymbol{W})^2\big)$$

where $\dot{\Delta}\boldsymbol{W} = \boldsymbol{W}^{n+1} - \boldsymbol{W}^n (= O(\Delta t))$. Let $\boldsymbol{H}(\boldsymbol{W}^n,\boldsymbol{W}^n,t^n,t^{n+1})$ be denoted by $\boldsymbol{H}^n$ and $\boldsymbol{H}_{\boldsymbol{W}_2}(\boldsymbol{W}^n,\boldsymbol{W}^n,t^n,t^{n+1})$ be denoted by $\underline{C}^n$. We direct our attention towards the CN2 scheme, (2.12) becomes, upon neglecting the $O(\Delta t^2)$ linearization error term in (2.12a) and multiplying by Δt,

$$[\underline{I} + \Delta t\underline{C}^n]\boldsymbol{W}^{n+1} + \Delta t\underline{G}P^{n+\frac{1}{2}} = [\underline{I} + \Delta t\underline{C}^n]\boldsymbol{W}^n - \Delta t\boldsymbol{H}^n \tag{4.1a}$$

$$\underline{D}\boldsymbol{W}^{n+1} = 0 \tag{4.1b}$$

or

$$[\underline{I} + \Delta t\underline{C}^n]\dot{\Delta}\boldsymbol{W} + \Delta t\underline{G}\dot{\Delta}P = -\Delta t(\boldsymbol{H}^n + \underline{G}P^{n-\frac{1}{2}}) \tag{4.2a}$$

$$\underline{D}\dot{\Delta}\boldsymbol{W} = 0 \tag{4.2b}$$

in which $\dot{\Delta}P = P^{n+\frac{1}{2}} - P^{n-\frac{1}{2}}$. The local errors of CN2 are not affected by the above linearization error under Assumption (**A**). (4.2) can be written as

$$\begin{bmatrix}\underline{I} & 0\\ 0 & 0\end{bmatrix}\begin{pmatrix}\dot{\Delta}\boldsymbol{W}\\ \dot{\Delta}P\end{pmatrix} + \Delta t\begin{bmatrix}\underline{C}^n & 0\\ 0 & 0\end{bmatrix}\begin{pmatrix}\dot{\Delta}\boldsymbol{W}\\ \dot{\Delta}P\end{pmatrix} + \Delta t\begin{bmatrix}0 & \underline{G}\\ \underline{D} & 0\end{bmatrix}\begin{pmatrix}\dot{\Delta}\boldsymbol{W}\\ \dot{\Delta}P\end{pmatrix} = \begin{pmatrix}RHS\\ 0\end{pmatrix} \tag{4.3}$$

where RHS stand for the right hand side of (4.2a). This can be written as

$$[\underline{\mathcal{E}} + \Delta t\underline{\mathcal{R}}^n + \Delta t\underline{\mathcal{S}}]\dot{\Delta}\mathcal{U} = \mathcal{RHS} \tag{4.4}$$

where $\underline{\mathcal{E}} = \left[\begin{smallmatrix}\underline{I} & 0\\ 0 & 0\end{smallmatrix}\right], \underline{\mathcal{R}}^n = \left[\begin{smallmatrix}\underline{C}^n & 0\\ 0 & 0\end{smallmatrix}\right], \underline{\mathcal{S}} = \left[\begin{smallmatrix}0 & \underline{G}\\ \underline{D} & 0\end{smallmatrix}\right]$ and $\dot{\Delta}\mathcal{U} = \left(\begin{smallmatrix}\dot{\Delta}\boldsymbol{W}\\ \dot{\Delta}P\end{smallmatrix}\right), \mathcal{RHS} = \left(\begin{smallmatrix}RHS\\ 0\end{smallmatrix}\right)$. With AF of its matrix, equation (4.4) is approximated by

$$[\underline{\mathcal{I}} + \Delta t\underline{\mathcal{R}}^n][\underline{\mathcal{E}} + \Delta t\underline{\mathcal{S}}]\dot{\Delta}\mathcal{U} = \mathcal{RHS} \tag{4.5}$$

where $\underline{\mathcal{I}} = \left[\begin{smallmatrix}\underline{I} & 0\\ 0 & \underline{I}\end{smallmatrix}\right]$. From this we get the following fractional step method:

$$\begin{aligned}&[\underline{\mathcal{I}} + \Delta t\underline{\mathcal{R}}^n]\dot{\Delta}\widetilde{\mathcal{U}} = \mathcal{RHS}\\ &[\underline{\mathcal{E}} + \Delta t\underline{\mathcal{S}}]\dot{\Delta}\mathcal{U} = \dot{\Delta}\widetilde{\mathcal{U}}\end{aligned} \tag{4.6}$$

where $\dot{\Delta}\widetilde{\mathcal{U}} = \left(\begin{smallmatrix}\dot{\Delta}\widetilde{\boldsymbol{W}}\\ \dot{\Delta}\widetilde{P}\end{smallmatrix}\right)$ with $\dot{\Delta}\widetilde{\boldsymbol{W}} = \widetilde{\boldsymbol{W}}^{n+1} - \boldsymbol{W}^n$, $\dot{\Delta}\widetilde{P}$ similarly. These give respectively

$$\begin{aligned}&[\underline{I} + \Delta t\underline{C}^n]\dot{\Delta}\widetilde{\boldsymbol{W}} = RHS, \quad \dot{\Delta}\widetilde{P} = 0\\ &\dot{\Delta}\boldsymbol{W} + \Delta t\underline{G}\dot{\Delta}P = \dot{\Delta}\widetilde{\boldsymbol{W}}, \quad \Delta t\underline{D}\dot{\Delta}\boldsymbol{W} = \dot{\Delta}\widetilde{P}\end{aligned}$$

which is just

$$\begin{aligned}\frac{\widetilde{\boldsymbol{W}}^{n+1}-\boldsymbol{W}^n}{\Delta t}+\underline{C}^n(\widetilde{\boldsymbol{W}}^{n+1}-\boldsymbol{W}^n)&=-(\boldsymbol{H}^n+\underline{G}P^{n-\frac{1}{2}})\\ \frac{\boldsymbol{W}^{n+1}-\widetilde{\boldsymbol{W}}^{n+1}}{\Delta t}+\underline{G}(P^{n+\frac{1}{2}}-P^{n-\frac{1}{2}})&=0\\ \underline{D}\boldsymbol{W}^{n+1}&=0\end{aligned}\tag{4.7}$$

which is the linearized version of CN2+CCPC.

From (4.5) we know that the AF error is $\Delta t^2\underline{\mathcal{R}}^n\underline{\mathcal{S}}\dot{\Delta}\mathcal{U}=\Delta t^2\left(\begin{smallmatrix}\underline{C}^n\underline{G}\dot{\Delta}P\\ 0\end{smallmatrix}\right)$, this adds an $O(\Delta t^3)$ error to (4.2a), which differs from Δt(2.10a) by an $\Delta t O(\Delta t^2)$ term. No change in the local errors are expected, i.e. for (4.7) $\boldsymbol{W}$ has an $O(\Delta t^3)$ local error and $\underline{G}P$ has an $O(\Delta t^2)$ local error, in agreement with the result of the last section.

However, if the variable is taken to be $\mathcal{U}$ and not $\dot{\Delta}\mathcal{U}$, then the error of AF is $\Delta t^2\underline{\mathcal{R}}^n\underline{\mathcal{S}}\mathcal{U}=\Delta t^2\left(\begin{smallmatrix}\underline{C}^n\underline{G}P\\ 0\end{smallmatrix}\right)$, which adds an $O(\Delta t^2)$ error to (4.1a) which differs from Δt(2.10a) by an $\Delta t O(\Delta t^2)$ term. An $O(\Delta t)$ drop in the local errors is expected, i.e. $\boldsymbol{W}$ has an $O(\Delta t^2)$ local error and $\underline{G}P$ has an $O(\Delta t)$ local error for the linearized version of CN2+PR, derived from (4.1) as the linearized version of CN2+CCPC from (4.2). These local errors are also in agreement with the result of the last section.

Now we study the AF error with more details, especially for points adjacent to the boundary. We first follow the analysis of [27] for NBCs of fractional step methods for parabolic equations. For simplicity, suppose $C^n=-\frac{1}{2}L=-\frac{1}{2}\frac{1}{\Delta^2}(\delta_x^2+\delta_y^2)$ ($\Delta=\Delta x=\Delta y$) corresponds to the element of $\underline{C}^n$ on the inner interior points (verses those adjacent to the boundary), where $\delta_x^2 f=f_{j-1}-2f_j+f_{j+1}$, $\delta_y^2 f$ similarly. Then

$$LG\phi=\frac{1}{\Delta^2}\big((G\phi)_{j-1k}-2(G\phi)_{jk}+(G\phi)_{j+1k}+\delta_y^2(G\phi)_{jk}\big)\tag{4.8}$$

On the boundary points there is no $LG\phi$ since its equation is simply $\dot{\Delta}\boldsymbol{W}=\mathbf{W}_B(t^{n+1})-\mathbf{W}(t^n)$, no problem is expected. We see from (3.8) also that the equations on the boundary points are exact. But on points adjacent to the boundary, e.g. the left boundary,

$$LG\phi=\frac{1}{\Delta^2}\big(\quad -2(G\phi)_{jk}+(G\phi)_{j+1k}+\delta_y^2(G\phi)_{jk}\big)\tag{4.9}$$

because $G\phi$ on the boundary is not explicitly involved, or $G\phi=0$ in our formulation. On the fixed mesh, both (4.8) and (4.9) are $O(\Delta t)$ terms with $\phi=\dot{\Delta}p$. However, from the partial differential equation point of view, on the inner interior points as $\Delta\to 0$, $\frac{\delta_x^2 f}{\Delta^2}\to\frac{\partial^2 f}{\partial x^2}$, thus $\frac{\delta_x^2\phi}{\Delta^2}=O(\Delta t)$, and the added error $\Delta t^2 LG\phi=O(\Delta t^3)$; while on the points adjacent to the boundary, this is not so.

Suppose on the boundary $G\phi=\mathsf{G}_B$ is known, let the boundary condition be written as

$$\frac{\boldsymbol{W}^{n+1}-\boldsymbol{W}^n}{\Delta t}+\underline{G_B}\Phi=\frac{\mathbf{W}_B(t^{n+1})-\mathbf{W}_B(t^n)}{\Delta t}+\mathbf{G}_B$$

with $\underline{G_B}$ and Φ, or $\underline{G_B}\Phi$, suitably defined, and $\mathbf{G}_B$ the vector with components G_B. Change (3.8) to

$$\frac{\widetilde{\boldsymbol{W}}_B^{n+1}-\boldsymbol{W}_B^n}{\Delta t}=\frac{\mathbf{W}_B(t^{n+1})-\mathbf{W}_B(t^n)}{\Delta t}+\mathbf{G}_B\implies\widetilde{\boldsymbol{W}}_B^{n+1}=\mathbf{W}_B(t^{n+1})+\Delta t\mathbf{G}_B\tag{4.10a}$$

$$\frac{\boldsymbol{W}_B^{n+1}-\widetilde{\boldsymbol{W}}_B^{n+1}}{\Delta t}+\underline{G_B}\Phi=0\implies\boldsymbol{W}_B^{n+1}=\mathbf{W}_B(t^{n+1})\tag{4.10b}$$

We see that the boundary condition remains exact after projection. We see also that on points adjacent to the boundary

$$LG\phi = \frac{1}{\Delta^2}\big((G_B\phi)_{j-1k} - 2(G\phi)_{jk} + (G\phi)_{j+1k} + \delta_y^2(G\phi)_{jk}\big) \tag{4.11}$$

Thus the added projection error is the same as that of the inner interior points, i.e. $O(\Delta t^3)$.

The above discussion with $\phi = \dot{\Delta} p$ holds for (4.7), the linearized version of CN2+CCPC. For the linearized version of CN2+PR, (4.8) and (4.9) on the fixed mesh are $O(1)$ terms with $\phi = p$, and the added projection error is $O(\Delta t^2)$. But as the mesh size decreases, this is no longer true for (4.9). Again, if (4.10) is employed as the NBC, (4.11) for $\phi = p$ is $O(1)$ and the added projection error is $O(\Delta t^2)$ on all interior points.

Now we come back to the fixed mesh and study the case of $\underline{C}^n = -\frac{1}{2}\underline{L}$ and

$$\underline{LG}\Phi = \underline{G}\Psi \quad \text{for some } \Psi \tag{4.12}$$

then (4.1a) with added PR projection error can be written as

$$\frac{\boldsymbol{W}^{n+1} - \boldsymbol{W}^n}{\Delta t} + \boldsymbol{H}^n - \frac{1}{2}\underline{L}(\boldsymbol{W}^{n+1} - \boldsymbol{W}^n) + \underline{G}P^{n+\frac{1}{2}} - \frac{1}{2}\Delta t\underline{G}\Psi = 0 \tag{4.13}$$

Applying $\underline{\mathrm{P}}$, we obtain

$$\frac{\boldsymbol{W}^{n+1} - \boldsymbol{W}^n}{\Delta t} + \underline{P}\big(-\frac{1}{2}\boldsymbol{H}^n + \underline{L}(\boldsymbol{W}^{n+1} - \boldsymbol{W}^n)\big) = 0$$

from which we can derive as (3.3) that the local error of $\boldsymbol{W}$ is $O(\Delta t^3)$. Thus when (4.12) approximates elementwise $(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})\begin{pmatrix}\frac{\partial}{\partial x}\\ \frac{\partial}{\partial y}\end{pmatrix}\phi = \begin{pmatrix}\frac{\partial}{\partial x}\\ \frac{\partial}{\partial y}\end{pmatrix}(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})\phi = \begin{pmatrix}\frac{\partial}{\partial x}\\ \frac{\partial}{\partial y}\end{pmatrix}\psi$, for example, we expect the accuracy of $\boldsymbol{W}$ to increase approximately one order. And this commutativity of operators will hold only with (4.11), i.e. the use of NBC (4.10). Such is also the case for the Kim and Moin method [15] with projection dealing with only the linear part of $\boldsymbol{H}$.

5. Numerical Boundary Conditions

In this section we discuss further the NBCs for the auxiliary velocity and DPE, with the CN2+CCPC method as a specific example. For simplicity, we consider a left boundary point L on a staggered mesh, see Fig. 1. On r, a, and b, (3.7b) are respectively

$$\begin{aligned}
&\frac{u_r^{n+1} - \tilde{u}_r}{\Delta t} + \frac{\delta\phi}{\Delta x}\Big|_r = 0\\
&\frac{v_a^{n+1} - \tilde{v}_a}{\Delta t} + \frac{\delta\phi}{\Delta y}\Big|_a = 0\\
&\frac{v_b^{n+1} - \tilde{v}_b}{\Delta t} + \frac{\delta\phi}{\Delta y}\Big|_b = 0
\end{aligned} \tag{5.1}$$

here we have neglected the superscript $n+1$ for the auxiliary velocity. We list several NBCs discussed frequently in literature:
(1). Velocity NBC for $\tilde{\boldsymbol{w}}$ and DPE, i.e. (3.8). On L, (3.8) is

$$\frac{\tilde{u}_L - \mathtt{u}_L(t^{n+1})}{\Delta t} = 0 \tag{5.2a}$$

$$\frac{u_L^{n+1} - \tilde{u}_L}{\Delta t} = 0 \tag{5.2b}$$

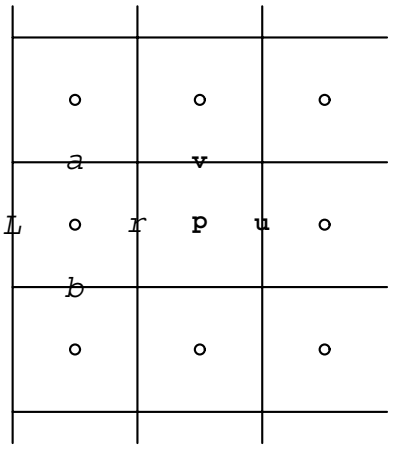

Figure 1. Staggered Mesh Left Boundary

and the DPE formed with (5.1) and (5.2b) is

$$\begin{aligned} &\frac{1}{\Delta t}\left(\frac{u_r^{n+1}-u_L^{n+1}}{\Delta x}-\frac{\tilde{u}_r-\tilde{u}_L}{\Delta x}\right)+\frac{1}{\Delta x}\left[\frac{\delta\phi}{\Delta x}\Big|_r\quad\right] \\ + \quad &\frac{1}{\Delta t}\left(\frac{v_a^{n+1}-v_b^{n+1}}{\Delta y}-\frac{\tilde{v}_a-\tilde{v}_b}{\Delta y}\right)+\frac{1}{\Delta y}\left[\frac{\delta\phi}{\Delta y}\Big|_a-\frac{\delta\phi}{\Delta y}\Big|_b\right]=0 \end{aligned} \tag{5.3}$$

i.e.

$$\frac{1}{\Delta x}\left[\frac{\delta\phi}{\Delta x}\Big|_r\quad\right]+\frac{1}{\Delta y}\left[\frac{\delta\phi}{\Delta y}\Big|_a-\frac{\delta\phi}{\Delta y}\Big|_b\right]=\frac{1}{\Delta t}\left(\frac{\tilde{u}_r-\mathsf{u}_L(t^{n+1})}{\Delta x}-\frac{\tilde{v}_a-\tilde{v}_b}{\Delta y}\right) \tag{5.4}$$

where (5.2a) has been used for $\tilde{u}_L$. It is seen that $\frac{\delta\phi}{\Delta x}$'s on the boundary point L do not appear; and it can be regarded as zero, comparing (3.8b) and (3.7b). Because the equations for interior points and for boundary points are different, $\tilde{\boldsymbol{w}}$ and Gp can have numerical boundary layers, especially if Re is large, i.e. there is not enough viscosity to smooth out the numerical solutions. However, $\boldsymbol{w}$ itself, a projection of $\tilde{\boldsymbol{w}}$ on $\mathcal{D}$ along $\mathcal{G}$, does not have numerical boundary layers, see discussions in [4, 24, 19]. The right hand side of the DPE is of discrete divergence form and satisfies $\sum_{I_0} rhs = 0$, its solvability condition. This is because the sum reduces to a sum of the normal component of the velocity on the boundary, which approximates the integral in (1.4) and thus is approximately 0. We consider the mesh to be sufficiently fine such that the solvability condition is guaranteed. The solvability condition can be derived in the following way: Let Ψ_0 denote the base vector of the null space of the discrete Laplacian $\underline{DG}$ in the DPE; it is known that $\Psi_0 = (1, 1, \cdots, 1)^T$. The constraint condition for the DPE to have a solution is $\Psi_0 \cdot R = 0$ where R is the vector with components rhs. It is also clear from Ψ_0 that solution ϕ is unique up to a constant, hence $\underline{G}\phi$ is unique.

(2). Improved NBC for $\tilde{\boldsymbol{w}}$ and DPE, i.e. (4.10).

On L, (4.10) is

$$\frac{\tilde{u}_L-\mathsf{u}_L(t^{n+1})}{\Delta t}-\mathsf{G}_L^x=0 \tag{5.5a}$$

$$\frac{u_L^{n+1}-\tilde{u}_L}{\Delta t}+G_L^x\phi=0 \quad \text{where } G_L^x\phi=\mathsf{G}_L^x \tag{5.5b}$$

The DPE formed with (5.1) and (5.5b) is

$$\frac{1}{\Delta x}\left[\frac{\delta\phi}{\Delta x}\Big|_r-G_L^x\phi\right]+\frac{1}{\Delta y}\left[\frac{\delta\phi}{\Delta y}\Big|_a-\frac{\delta\phi}{\Delta y}\Big|_b\right]=\frac{1}{\Delta t}\left(\frac{\tilde{u}_r-\tilde{u}_L}{\Delta x}-\frac{\tilde{v}_a-\tilde{v}_b}{\Delta y}\right)$$

Substituting (5.5a) into the above equation results again in (5.4), see also [6]. It is seen that if the NBC for $\tilde{\boldsymbol{w}}_B$ used in the computation of $\tilde{\boldsymbol{w}}$ is the same as the relation used in the formation of the DPE , then the DPE is just (5.4). Because the equations for interior points and for boundary points are the same (compare (4.10b) with (3.7b)), no numerical boundary layers are expected for $\tilde{\boldsymbol{w}}$ and $\underline{G}\phi$.

(3). Improved NBC for $\tilde{\boldsymbol{w}}$.

For $\tilde{u}_L$ we have (5.5a). If (5.2b) were used for the formation of DPE, we would get (5.3), but using (5.5a) for $\tilde{u}_L$ results in

$$\frac{1}{\Delta x}\left[\frac{\delta\phi}{\Delta x}\Big|_r\quad\right]+\frac{1}{\Delta y}\left[\frac{\delta\phi}{\Delta y}\Big|_a-\frac{\delta\phi}{\Delta y}\Big|_b\right]=\frac{1}{\Delta t}\left(\frac{\tilde{u}_r-\mathsf{u}_L(t^{n+1})}{\Delta x}-\frac{\tilde{v}_a-\tilde{v}_b}{\Delta y}\right)-\frac{1}{\Delta x}\mathsf{G}_L^x$$

and the solvability condition would involve not only w_{nB} but also $\frac{\partial p}{\partial n}\big|_B$ on the boundary (see below). However, we can simply use $\boldsymbol{w}_B^{n+1} = \mathbf{w}_B(t^{n+1})$, see [10], here it is

$$\frac{u_L^{n+1}-\mathsf{u}_L(t^{n+1})}{\Delta t}=0 \tag{5.6}$$

DPE formed with (5.1) and (5.6) is again (5.4). We will call this type of boundary conditions KMNBC, implied by the method of [15] as interpreted by [4], in which the boundary condition for ϕ (or p) is the homogeneous Neumann condition. We see that the NBC for $\tilde{\boldsymbol{w}}$ and the DPE formed in (2) and (3) come down to the same thing.

(4). Velocity NBC for $\tilde{\boldsymbol{w}}$, but improved NBC for DPE.

That is, use (5.2a) for the computation of the auxiliary velocity, but (5.5b) in the formation of DPE, this yields

$$\frac{1}{\Delta x}\left[\frac{\delta\phi}{\Delta x}\Big|_r \quad\right] + \frac{1}{\Delta y}\left[\frac{\delta\phi}{\Delta y}\Big|_a - \frac{\delta\phi}{\Delta y}\Big|_b\right] = \frac{1}{\Delta t}\left(\frac{\tilde{u}_r - \mathsf{u}_L(t^{n+1})}{\Delta x} - \frac{\tilde{v}_a - \tilde{v}_b}{\Delta y}\right) + \frac{1}{\Delta x}\mathsf{G}_L^x \tag{5.7}$$

Thus the solvabililty condition involves not only w_{nB} but also $\frac{\partial p}{\partial n}\big|_B$ on the boundary. We will call this type of boundary conditions OIDNBC, again from the interpretation in [4], of the method of [16]. In this method, $\frac{\partial p}{\partial n}\big|_B$ is given for the solution of the DPE, and one of the $\tilde{\boldsymbol{w}}$ and $\boldsymbol{w}^{n+1}$ satisfies the boundary condition, while the other satisfies the divergence-free condition. In the present version, $\tilde{\boldsymbol{w}}$ satisfies the boundary condition, and $\boldsymbol{w}^{n+1}$ satisfies the divergence-free condition. On L, (5.2a) and (5.5b) implies $u_L^{n+1} = \mathsf{u}_L(t^{n+1}) - \Delta t \mathsf{G}_L^x$, which means in general an $O(\Delta t)$ error in $\boldsymbol{w}_B^{n+1}$. The $\frac{\partial p}{\partial n}\big|_B$ and the constraint condition involved are discussed in [16]; e.g. for the two-dimensional Stokes equations with time-independent boundary conditions, $\frac{\partial p}{\partial n} = -\frac{1}{\mathsf{Re}}\boldsymbol{n}\cdot\left(\frac{\partial \omega_z}{\partial y}, -\frac{\partial \omega_z}{\partial x}\right)^T$ where $\omega_z = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$, which satisfies $\oint_{\partial\Omega}\frac{\partial p}{\partial n}\big|_B ds = 0$; with appropriate finite difference approximation on a sufficiently fine mesh, the solution of DPE is guaranteed.

We summerize the projection methods associated with the NBCs discussed above in terms of the Discrete Projection Theorem of Section 2. Here (2.6) and (2.7) become (neglecting factor Δt)

$$\widetilde{\boldsymbol{W}} = \boldsymbol{W} + \underline{G}\Phi \tag{5.8}$$

$$\underline{D}\underline{G}\Phi = \underline{D}\widetilde{\boldsymbol{W}} \tag{5.9}$$

Since the NBC for determining $\widetilde{\boldsymbol{W}}_I$ can be different from that used in the formation of DPE, we will use $\mathbf{A}_B$ to denote the left hand side of (5.8) on B for projection. We have

(1). Velocity NBC:

For the computation of $\widetilde{\boldsymbol{W}}_I$, $\widetilde{\boldsymbol{W}}_B = \mathbf{W}_B$; for projection $\mathbf{A}_B = \mathbf{W}_B$, so the decomposition (5.8) and (5.9) are respectively

$$\begin{pmatrix} \widetilde{\boldsymbol{W}}_I \\ \mathbf{W}_B \end{pmatrix} = \begin{pmatrix} \boldsymbol{W}_I \\ \mathbf{W}_B \end{pmatrix} + \begin{pmatrix} \underline{G}_I\Phi \\ 0 \end{pmatrix} \tag{5.10}$$

$$\underline{D}_I\widetilde{\boldsymbol{W}}_I + \underline{D}_B\mathbf{W}_B = 0 + \underline{D}_I\underline{G}\Phi \tag{5.11}$$

(2). Improved NBC:

For the computation of $\widetilde{\boldsymbol{W}}_I$, $\widetilde{\boldsymbol{W}}_B = \mathbf{W}_B + \mathbf{G}_B$; for projection $\mathbf{A}_B = \mathbf{W}_B + \mathbf{G}_B$, so

$$\begin{pmatrix} \widetilde{\boldsymbol{W}}_I \\ \mathbf{W}_B + \mathbf{G}_B \end{pmatrix} = \begin{pmatrix} \boldsymbol{W}_I \\ \mathbf{W}_B \end{pmatrix} + \begin{pmatrix} \underline{G}_I\Phi \\ \mathbf{G}_B \end{pmatrix} \tag{5.12}$$

$$\underline{D}_I\widetilde{\boldsymbol{W}}_I + \underline{D}_B\mathbf{W}_B = 0 + \underline{D}_I\underline{G}\Phi \tag{5.13}$$

(3). KMNBC:

For the computation of $\widetilde{\boldsymbol{W}}_I$, $\widetilde{\boldsymbol{W}}_B = \mathbf{W}_B + \mathbf{G}_B$; for projection $\mathbf{A}_B = \mathbf{W}_B$, so

$$\begin{pmatrix} \widetilde{\boldsymbol{W}}_I \\ \mathbf{W}_B \end{pmatrix} = \begin{pmatrix} \boldsymbol{W}_I \\ \mathbf{W}_B \end{pmatrix} + \begin{pmatrix} \underline{G}_I\Phi \\ 0 \end{pmatrix} \tag{5.14}$$

$$\underline{D}_I\widetilde{\boldsymbol{W}}_I + \underline{D}_B\mathbf{W}_B = 0 + \underline{D}_I\underline{G}\Phi \tag{5.15}$$

(4). OIDNBC:
For the computation of $\widetilde{\boldsymbol{W}}_I$, $\widetilde{\boldsymbol{W}}_B = \mathbf{W}_B$; for projection $\mathbf{A}_B = \mathbf{W}_B$, but $\mathbf{G}_B$ is specified meeting certain requirements, so

$$\begin{pmatrix} \widetilde{\boldsymbol{W}}_I \\ \mathbf{W}_B \end{pmatrix} = \begin{pmatrix} \boldsymbol{W}_I \\ \boldsymbol{W}_B \end{pmatrix} + \begin{pmatrix} \underline{G}_I \Phi \\ \mathbf{G}_B \end{pmatrix} \tag{5.16}$$

$$\underline{D}_I \widetilde{\boldsymbol{W}}_I + \underline{D}_B \mathbf{W}_B = 0 + \underline{D}_I \underline{G} \Phi + \underline{D}_B \mathbf{G}_B \tag{5.17}$$

in which $\boldsymbol{W}_B$ is to be determined.

6. Numerical Experiments

Consider INSE with exact solution

$$u = e^t \ \sin x \ \cos y \quad v = -e^t \ \cos x \ \sin y \quad p = e^t \ \sin x \ \sin y \tag{6.1}$$

on a square $0 \le x \le \pi, 0 \le y \le \pi$. The initial and boundary values for u and v (and $\frac{\partial p}{\partial n}$), as well as the corresponding nonhomogeneous terms in the momentum equations, are taken from (6.1); here $\mathsf{R}e = 1$ unless stated otherwise. Note that the boundary values are time-dependent with $w_n = 0$ and $\frac{\partial p}{\partial n} \neq 0$.

In the numerical experiments, uniform staggered meshes were used for its simplicity, with fineness 32×32, 64×64, and 128×128. Computations were carried out for $0 \le t \le 1$, with total number of steps $N = 10$, 20, 40, 80, 160, and 320. Note that the numerical boundary condition for $\frac{\partial p}{\partial n}$ were exact in this numerical experiment, so as not to complicate matters with further approximation and the possible related stability problems.

For the INSE, the following computation were performed:

CN2+CCPC method	with UVNBC	result in	Table Ia
as above	KMNBC		Table Ib
CN2+PR method	UVNBC		Table IIa
as above	KMNBC		Table IIb

Since u and v, $\frac{\partial p}{\partial x}$ and $\frac{\partial p}{\partial y}$ have symmetries and the numerical solution showed the same symmetries, only: adum $=\max_{(j,k)\in Iu+Bu} |u^N_{jk} - u_e(x_j, y_k, 1)|$, dul2 $=\big(\sum_{(j,k)\in Iu+Bu} |u^N_{jk} - u_e(x_j, y_k, 1)|^2 \Delta x_j \Delta y\big)^{1/2} = \|u^N - u_e\|$, and adpxm$=\max_{(j,k)\in Iu} \big|\big(\frac{\delta p}{\Delta x}\big)^{N-1/2}_{jk} - \frac{\partial p_e}{\partial x}(x_j, y_k, 1 - \frac{\Delta t}{2})\big|$ are shown in the tables of results; here u_e and $\frac{\partial p_e}{\partial x}$ are of the exact solution, and $\Delta x_j = \frac{\Delta x}{2}$ at the left and right boundary. But the exact solution is for the space continuous problem and no exact solution is available for the space discrete problem, i.e. for the DAE. So, as [25], the number $\kappa = \|u^N(\Delta t) - u^N(2\Delta t)\| / \|u^N(\frac{\Delta t}{2}) - u^N(\Delta t)\|$ is used to show the order of accuracy on the fixed mesh. For first and second order accuracy, i.e. global error $O(\Delta t)$ and $O(\Delta t^2)$, $\kappa \to 2$ and 4 respectively. Double precision was used throughout the computation to get the desired number of significant figures for this number. The DPEs were solved using the direct solvers from FISHPACK, see [22], with the same precision, which produced $\max_{(j,k)\in I_0} |(D\boldsymbol{w})_{jk}| < 10^{-12}$.

Before examining the table of results, we note that as we are dealing with partial differential equations, various errors of various terms of the equations interact, especially $\max|u|\frac{\Delta t}{\Delta}$ ($\max|v|\frac{\Delta t}{\Delta}$) and $\frac{1}{\mathsf{R}e}\frac{\Delta t}{\Delta^2}$ influence the accuracy of results. These aspects should be taken into consideration, though they are difficult to account for specifically. In the tables of results, we see first that for a particular mesh, errors adum, dul2, and adpxm decrease with the decrease of Δt only up to a certain point. But from κ we can infer the global error of $\boldsymbol{w}$ on the fixed mesh. For CN2+CCPC, we see from Table I that it is $O(\Delta t^2)$. From the partial differential equation point of view, as Δt and $\Delta \to 0$, we expect $O(\Delta t^2 + \Delta^2)$ global error for $\boldsymbol{w}$, i.e. as Δt and Δ decrease by a factor of 2, the errors decrease by a factor of 4, and from Table Ia, ratios $r(32, 64) =$

dul2($\Delta t = 0.1,\ 32 \times 32$ mesh) / dul2($\Delta t = 0.05,\ 64 \times 64$ mesh) $= 0.6808 \times 10^{-2}/0.1920 \times 10^{-2}$ $= 3.546$, $r(64,128)$= dul2($\Delta t = 0.05,\ 64 \times 64$ mesh) / dul2($\Delta t = 0.025,\ 128 \times 128$mesh) $= 0.1920 \times 10^{-2}/0.5064 \times 10^{-3} = 3.791$. From Table Ib, these ratios are $r(32,64) = 3.736$ and $r(64,128) = 3.889$. Comparing Table Ia with Table Ib, we see little difference in the errors of $\boldsymbol{w}$; but for large Δt, adpxms for KMNBC are much smaller than those for UVNBC. The difference in Gp can be easily observed from Fig. 2, in which the solid lines represent the profiles of $u_e(x,y,1)$ and $\frac{\partial p_e}{\partial x}(x,y,1-\frac{\Delta t}{2})$ for $0 \le x \le \frac{\pi}{2}$ at $y = \frac{\pi}{4}$, and the dots (small dots) represent the corresponding u^N and $(\frac{\delta p}{\delta x})^{N-1/2}$ (and $\tilde{u}$). There is no appreciable numerical boundary layer in $\frac{\delta p}{\delta x}$ with the KMNBC. However, accuracy of this Gp may not be important for practical computation, as once $\boldsymbol{w}$ is known, an accurate p can be obtained from the component-consistency condition (2.5). The $(\frac{\delta p}{\delta x})^N$ (with $\frac{\partial p_e}{\partial x}(x,y,1)$) in Fig. 2(b) was obtained in this way; there is no numerical boundary layer. The corresponding sequence of κ for $\frac{\delta p}{\delta x}^N$ on the 64×64 mesh for $\Delta t = 0.1$ to 0.003125 is 6.994, 6.154, 4.596, 4.150.

The computational results for the CN2+PR method for INSE are shown in Table II and Fig. 3. In Table II, we see first that the errors adum and dul2 are much larger than those in Table I, we infer from κ that the global error of $\boldsymbol{w}$ is $O(\Delta t)$ on the fixed mesh. Then as Δt and Δ decrease by a factor of 2, we have for UVNBC $r(32,64) = 1.681$ and $r(64,128) = 1.819$, and for KMNBC $r(32,64) = 1.865$ and $r(64,128) = 1.926$. In Table IIa and Table IIb, we again see little difference in the global errors of $\boldsymbol{w}$. As for the global errors of Gp, we state only that they are much larger than those of Table I, and that again adpxm for KMNBC are much smaller than those for UVNBC. Also compare Fig. 3 with Fig. 2.

In summary, results of the numerical experiments on the INSE show that for the fixed mesh, the global error of $\boldsymbol{w}$ for the CN2+CCPC method is $O(\Delta t^2)$, while it is one order less, i.e. $O(\Delta t)$, for the CN2+PR method. The global error of Gp for the former is also of higher order than that for the latter. These results are in agreement with the local errors derived under Assumption (**A**) in Section 3.

In order to verify the errors derived for the linear case in Section 3. and to study the effect of NBC on the numerical solutions, the following computations were performed for the Stokes equations:

CN2+PR method with UVNBC KMNBC OIDNBC

and periodic BC for comparison. Computations with UVNBC and KMNBC were carried out on the 64×64 mesh, and with OIDNBC and the periodic boundary condition on the 128×128 mesh for the $0 \le x \le 2\pi, 0 \le y \le 2\pi$ square region, i.e. with the same Δ. The computational results are shown in Table III and Fig.4–7, in which v^N and $(\frac{\delta p}{\Delta y})^{N-1/2}$ (and $\tilde{v}$) are also shown, again for $0 \le x \le \frac{\pi}{2}$ at $y = \frac{\pi}{4}$.

The most noticable thing in Table III is that with KMNBC, $\kappa \simeq 4$; while for INSE, with the same method and the same NBC, $\kappa \simeq 2$ as shown in Table IIb. Also seen in Table III is that the behavior of errors of $\boldsymbol{w}$ and Gp for KMNBC is very much like that for the periodic boundary condition. We remark that Fig. 6 for KMNBC is almost identical to the corresponding figure for the periodic BC (not shown). These results are in agreement with the analysis of Section 3 for the linear case with commutativity of $\underline{P}$, $\underline{Q}$ with $\underline{L}$; and in accordance with the discussion at the end of Section 4 for the linear case with KMNBC (4.10). It seems that in the linear case, NBC (4.10) "makes" the formulation of the problem periodic; compare (3.6b) for the interior points with (4.10b) (with P in place of Φ) for the boundary points. With periodicity, the commutativity of operators becomes likely. However, in practical computation $\frac{\partial p}{\partial n}\big|_B$ is not available and needs approximation; the accuracy and stability involved are not studied in this paper.

The projection errors are dependent on $\mathsf{R}e$, as seen from the AF errors $-\frac{1}{2}\frac{1}{\mathsf{R}e}\Delta t^2 \underline{LG}P$ of Section 4, which decreases with increasing $\mathsf{R}e$. With UVNBC, there are "numerical boundary layers" in P, in agreement with the analyses in the literature cited in the Introduction. This phenomenon is illustrated in Fig. 4 ($\mathsf{R}e = 1$) and Fig. 5 ($\mathsf{R}e = 100$).

Now we come to the OIDNBC as discussed in Section 5. It is seen that for CN2+PR with this NBC, $\kappa \simeq 2$, i.e. the global error of $\boldsymbol{w}$ is $O(\Delta t)$, This is to be expected by the $O(\Delta t)$ error in $\boldsymbol{w}_B^{n+1}$ ($\boldsymbol{w}_B^{n+1} = \mathbf{w}_B(t^{n+1}) - \Delta t \mathsf{G}_B$). With no boundary layer in $\boldsymbol{w}_B^{n+1}$, this error is spread into the interior, as shown in Fig. 7. This situation prevails for $\mathsf{R}e = 100$. The original version of the corresponding algorithm in [16] was also tested, resulting also in $\kappa \simeq 2$. However, with the improved pressure boundary condition and higher order time differencing and time splitting of [16], higher order accurate numerical results have been obtained, see [14].

7. Concluding Remarks

The INSE with spatial discretization on a fixed mesh is regarded as a system of DAE. The local errors of projection methods for the system can be easily derived with the projection operators defined by the discrete projection theorem. In this paper, local errors for two CN schemes with PC, PR, and CCPC projection methods are studied. It is seen that local errors of the velocity and the pressure gradient can be of different orders, and that these orders can depend on the linearity and commutativity of certain operators, as also seen from similar results in the literature.

Since time accuracy for the spatially discretized INSE on fixed meshes does not give the whole picture of a numerical method for the INSE as partial differential equations, the projection errors are also investigated as AF errors of relevant matrices. Special attention is paid to the NBCs for the auxiliary velocity and the DPE; it is seen that the KMNBC may change the order of accuracy of the entire method. Also, the NBCs are summarized in the context of discrete projections. Results of numerical experiment with an analytic example confirm many of the above conclusions. It is seen that time accuracy for the spatially discretized INSE as DAE is also significant for the INSE as partial differential equations.

Among the methods studied in this paper, the author recommends the CN+CCPC method for its accuracy and for its approximate preservation of component-consistency under projection. Also, the UVNBC is the simplest, and suffices for practical purposes; since once the velocity with a certain accuracy is known, another solution of pressure from the component-consistency condition will produce pressure with the same accuracy. Mostly, the author hopes to contribute to the understanding of the numerical solution for the INSE as partial differential equations with constraint. It is shown here that with simple tools familiar to the computational fluid dynamic community, many aspects of a general method can be analyzed. The conclusions, while not completely rigorous, can give insight to the many perplexities of the projection methods for the INSE.

CN2+	Δt	adum	dul2	κ	adpxm
CCPC	0.1	0.5511E-2	0.6808E-2		0.6438E-0
UVNBC	0.05	0.1970E-2	0.2556E-2	3.455	0.2754E-0
32x32	0.025	0.9646E-3	0.1345E-2	3.799	0.7965E-1
	0.0125	0.7168E-3	0.1036E-2	3.921	0.1963E-1
	0.00625	0.6585E-3	0.9592E-3	4.015	0.2078E-1
	0.003125	0.6439E-3	0.9401E-3		0.2110E-1
	0.1	0.5097E-2	0.6207E-2		0.1020E+1
	0.05	0.1537E-2	0.1920E-2	3.420	0.6483E-0
64x64	0.025	0.5089E-3	0.6687E-3	3.775	0.2767E-0
	0.0125	0.2432E-3	0.3412E-3	3.913	0.8218E-1
	0.00625	0.1797E-3	0.2595E-3	4.011	0.2114E-1
	0.003125	0.1646E-3	0.2395E-3		0.5224E-2
	0.1	0.4994E-2	0.6061E-2		0.1291E+1
	0.05	0.1432E-2	0.1768E-2	3.406	0.1040E+1
128x128	0.025	0.4010E-3	0.5064E-3	3.754	0.6547E-0
	0.0125	0.1294E-3	0.1711E-3	3.900	0.2779E-0
	0.00625	0.6118E-4	0.8605E-4	4.006	0.8260E-1
	0.003125	0.4515E-4	0.6526E-4		0.2159E-1

Table Ia. INSE with CN2+CCPC and UVNBC

CN2+	Δt	adum	dul2	κ	adpxm
CCPC	0.1	0.5813E-2	0.7550E-2		0.8164E-1
KMNBC	0.05	0.2030E-2	0.2722E-2	3.587	0.2088E-1
32x32	0.025	0.9812E-3	0.1388E-2	3.837	0.1532E-1
	0.0125	0.7217E-3	0.1048E-2	3.932	0.1961E-1
	0.00625	0.6597E-3	0.9620E-3	4.016	0.2077E-1
	0.003125	0.6443E-3	0.9409E-3		0.2110E-1
	0.1	0.5324E-2	0.6792E-2		0.1091E-0
	0.05	0.1574E-2	0.2021E-2	3.582	0.3938E-1
64x64	0.025	0.5167E-3	0.6904E-3	3.836	0.9887E-2
	0.0125	0.2453E-3	0.3467E-3	3.931	0.3891E-2
	0.00625	0.1802E-3	0.2609E-3	4.015	0.4947E-2
	0.003125	0.1648E-3	0.2399E-3		0.5223E-2
	0.1	0.5204E-2	0.6600E-2		0.1276E-0
	0.05	0.1462E-2	0.1848E-2	3.578	0.5637E-1
128x128	0.025	0.4056E-3	0.5197E-3	3.835	0.1940E-1
	0.0125	0.1303E-3	0.1739E-3	3.931	0.4683E-2
	0.00625	0.6144E-4	0.8675E-4	4.015	0.9808E-3
	0.003125	0.4523E-4	0.6544E-4		0.1241E-2

Table Ib. INSE with CN2+CCPC and KMNBC

CN2+	Δt	adum	dul2	κ	adpxm
PR	0.1	0.5137E-1	0.6771E-1		0.1767E+1
UVNBC	0.05	0.3073E-1	0.4067E-1	1.508	0.1497E+1
32x32	0.025	0.1701E-1	0.2270E-1	1.738	0.1179E+1
	0.0125	0.9178E-2	0.1236E-1	1.857	0.8535E-0
	0.00625	0.5004E-2	0.6796E-2	1.922	0.5635E-0
	0.003125	0.2841E-2	0.3900E-2		0.3398E-0
	0.1	0.5098E-1	0.6734E-1		0.2130E+1
	0.05	0.3043E-1	0.4028E-1	1.504	0.1981E+1
64x64	0.025	0.1669E-1	0.2226E-1	1.733	0.1759E+1
	0.0125	0.8811E-2	0.1185E-1	1.850	0.1479E+1
	0.00625	0.4593E-2	0.6221E-2	1.913	0.1164E+1
	0.003125	0.2404E-2	0.3280E-2		0.8441E-1
	0.1	0.5094E-1	0.6725E-1		0.2345E+1
	0.05	0.3034E-1	0.4019E-1	1.503	0.2289E+1
128x128	0.025	0.1661E-1	0.2215E-1	1.731	0.2169E+1
	0.0125	0.8722E-2	0.1173E-1	1.848	0.1991E+1
	0.00625	0.4494E-2	0.6087E-2	1.910	0.1758E+1
	0.003125	0.2299E-2	0.3133E-2		0.1476E+1

Table IIa. INSE with CN2+PR and UVNBC

CN2+	Δt	adum	dul2	κ	adpxm
PR	0.1	0.6128E-1	0.8890E-1		0.3664E-0
KMNBC	0.05	0.3437E-1	0.4851E-1	1.773	0.1977E-0
32x32	0.025	0.1834E-1	0.2571E-1	1.881	0.9578E-1
	0.0125	0.9704E-2	0.1357E-1	1.938	0.4347E-1
	0.00625	0.5220E-2	0.7316E-2	1.969	0.2140E-1
	0.003125	0.2936E-2	0.4137E-2		0.1160E-1
	0.1	0.6117E-1	0.8781E-1		0.3768E-0
	0.05	0.3382E-1	0.4767E-1	1.770	0.2115E-0
64x64	0.025	0.1784E-1	0.2495E-1	1.879	0.1108E-0
	0.0125	0.9222E-2	0.1285E-1	1.937	0.5493E-1
	0.00625	0.4745E-2	0.6608E-2	1.968	0.2545E-1
	0.003125	0.2464E-2	0.3436E-2		0.1200E-1
	0.1	0.6100E-1	0.8749E-1		0.3816E-0
	0.05	0.3367E-1	0.4744E-1	1.768	0.2150E-0
128x128	0.025	0.1771E-1	0.2475E-1	1.878	0.1145E-0
	0.0125	0.9098E-2	0.1267E-1	1.937	0.5878E-1
	0.00625	0.4626E-2	0.6432E-2	1.968	0.2934E-1
	0.003125	0.2347E-2	0.3262E-2		0.1417E-1

Table IIb. INSE with CN2+PR and KMNBC

CN2+PR	Δt	adum	dul2	κ	adpxm
	0.1	0.1221E-1	0.8983E-2		0.2112E+1
UVNBC	0.05	0.5378E-2	0.3753E-2	2.342	0.1972E+1
64x64	0.025	0.2208E-2	0.1511E-2	2.437	0.1755E+1
	0.0125	0.8673E-3	0.6146E-3	2.558	0.1479E+1
	0.00625	0.3333E-3	0.3132E-3	2.648	0.1164E+1
	0.003125	0.1613E-3	0.2447E-3		0.8447 -0
	0.1	0.2017E-2	0.2994E-2		0.2032E-0
KMNBC	0.05	0.3830E-3	0.5664E-3	4.043	0.1134E-0
64x64	0.025	0.2826E-4	0.4159E-4	4.052	0.6071E-1
	0.0125	0.1264E-3	0.1846E-3	4.061	0.3168E-1
	0.00625	0.1515E-3	0.2209E-3	4.090	0.1627E-1
	0.003125	0.1578E-3	0.2289E-3		0.8287E-2
	0.1	0.4310E-2	0.1354E-1		0.2403E-0
periodic	0.05	0.8188E-3	0.2568E-2	3.996	0.1277E-0
128x128	0.025	0.5775E-4	0.1786E-3	3.995	0.6585E-1
	0.0125	0.2764E-3	0.8655E-3	4.004	0.3344E-1
	0.00625	0.3311E-3	0.1037E-2	3.974	0.1685E-1
	0.003125	0.3448E-3	0.1080E-2		0.8462E-2
	0.1	0.2577E-0	0.1050E+1		0.3584E-0
OIDNBC	0.05	0.1322E-0	0.5431E-0	1.902	0.4812E-0
128x128	0.025	0.6696E-1	0.2764E-0	1.949	0.2565E-0
	0.0125	0.3371E-1	0.1395E-0	1.973	0.1336E-0
	0.00625	0.1691E-1	0.7005E-1	1.986	0.6849E-1
	0.003125	0.8470E-2	0.2511E-1		0.3476E-1

Table III. The Stokes Equations with CN2+PR

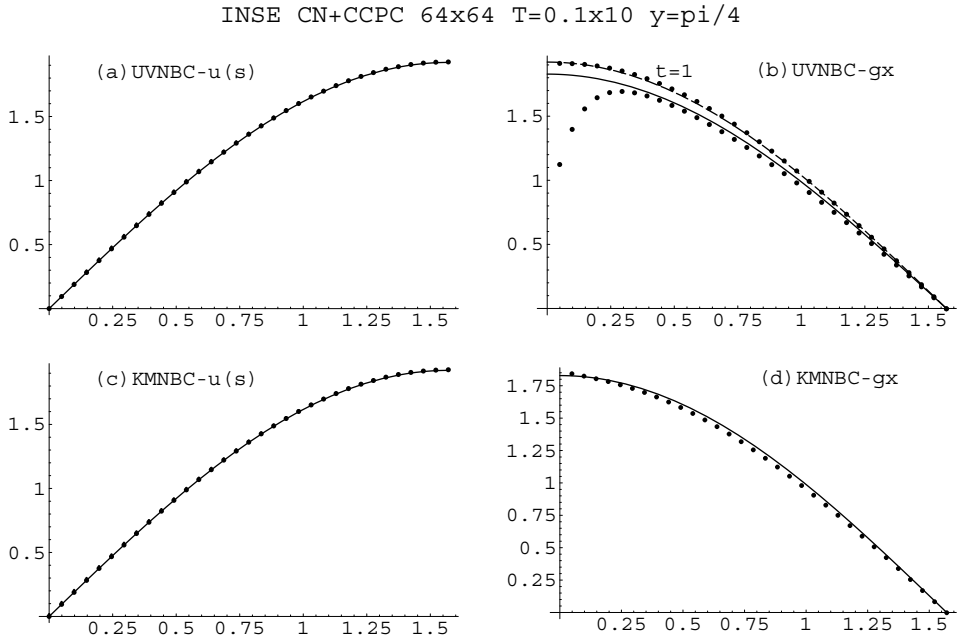

Figure 2

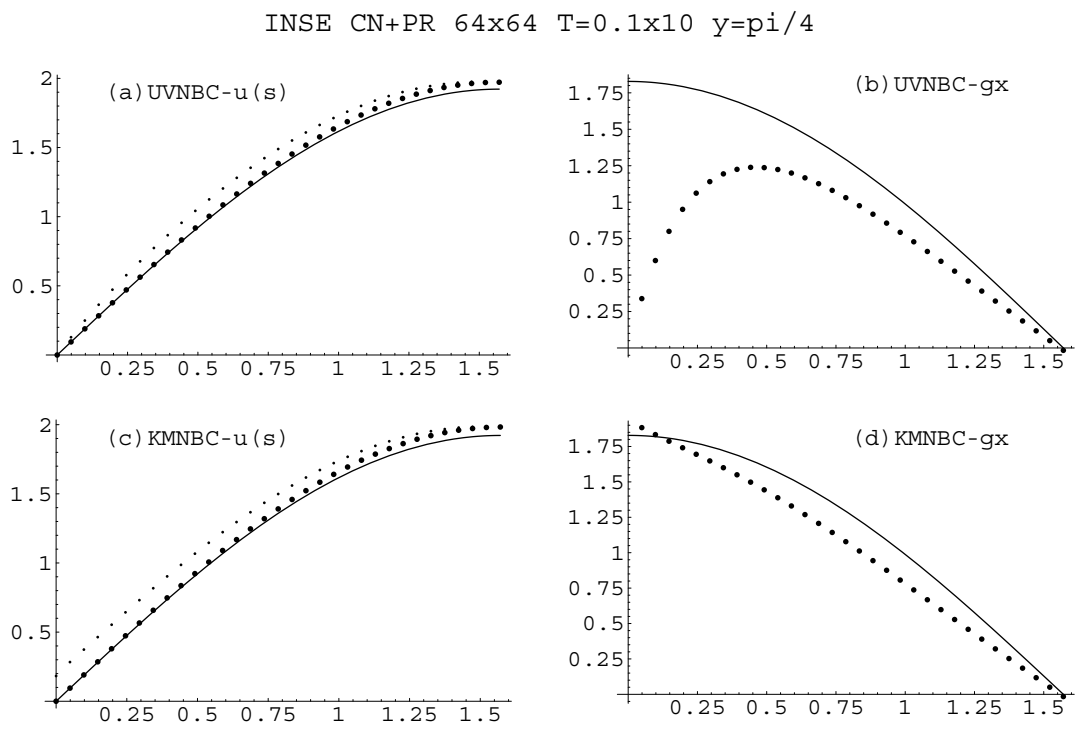

Figure 3

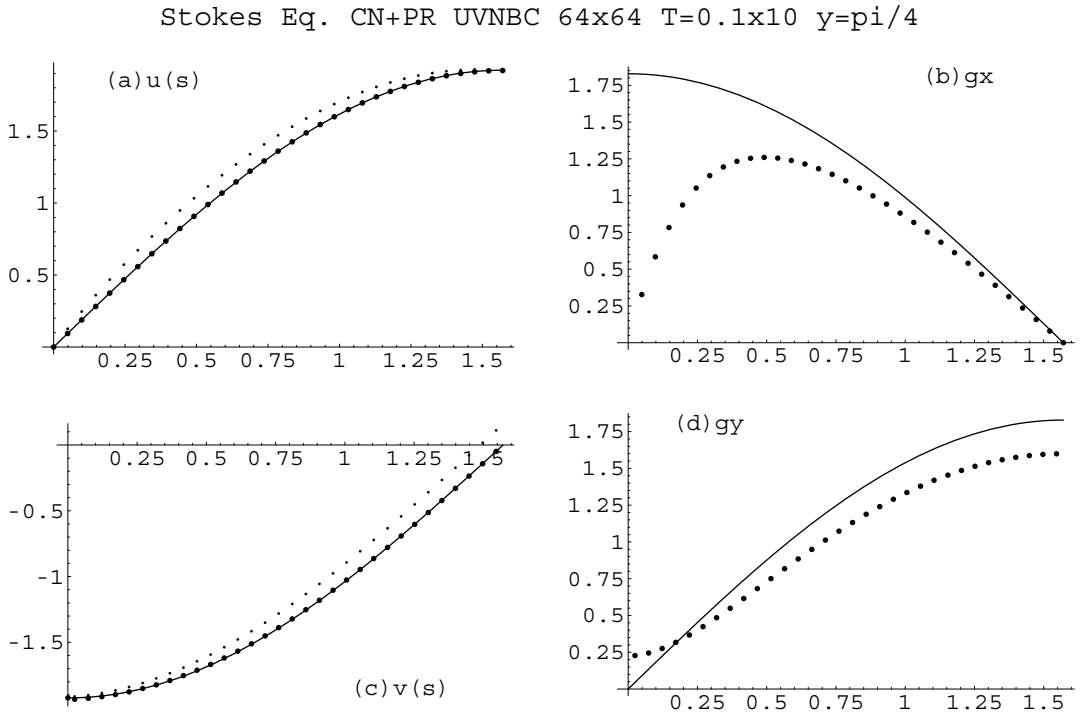

Figure 4

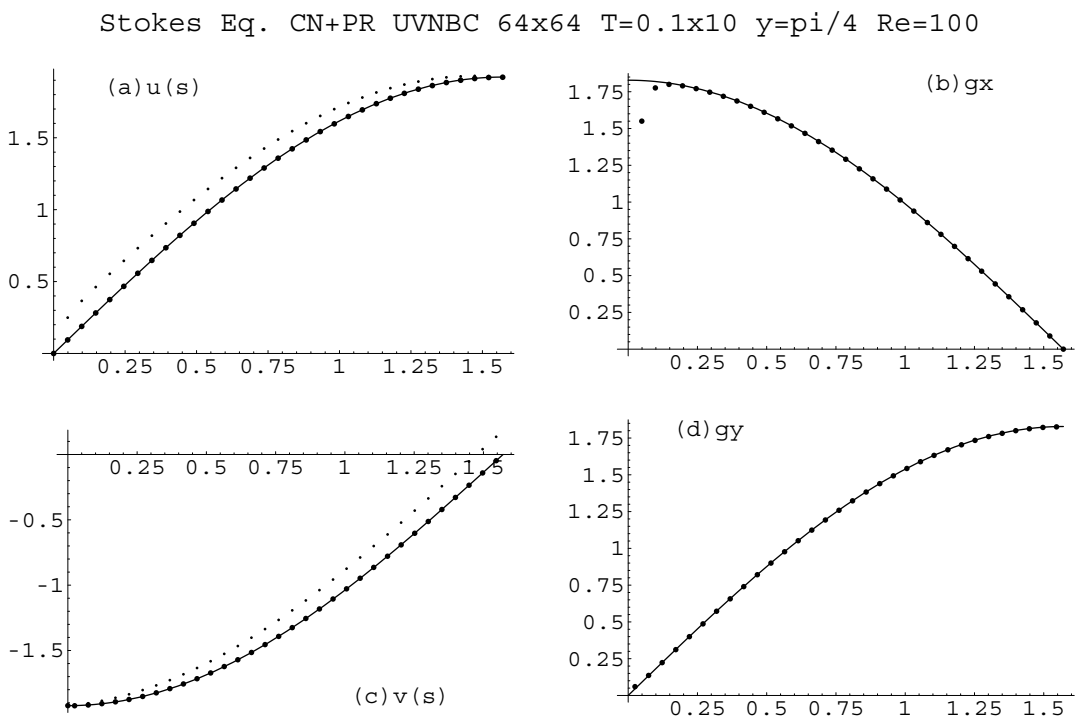

Figure 5

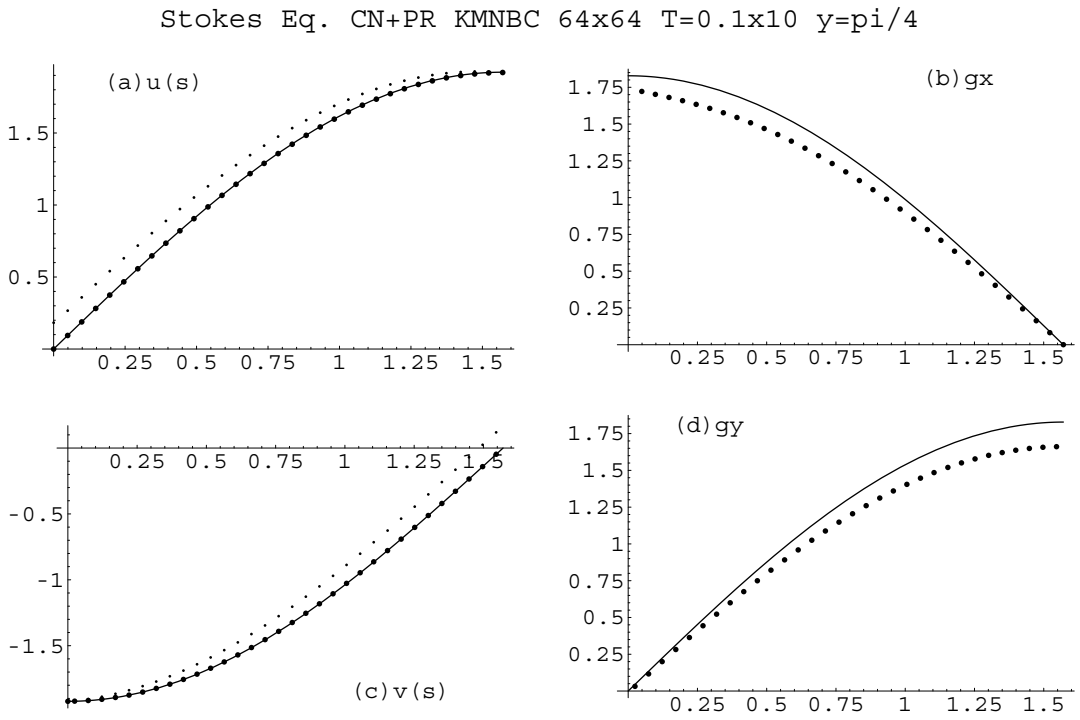

Figure 6

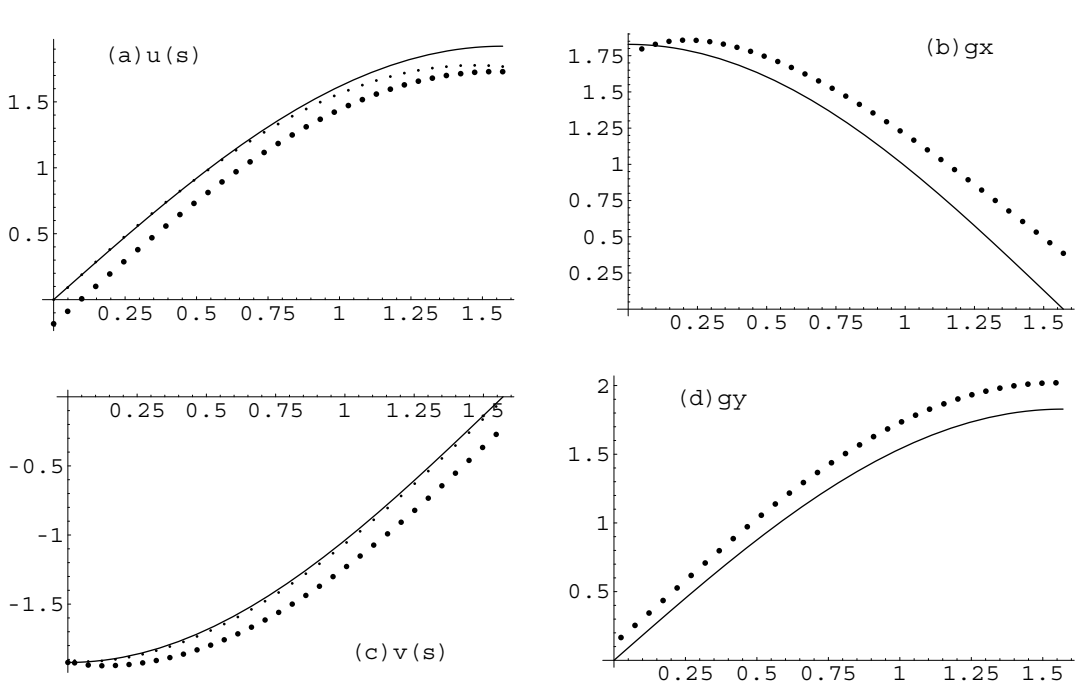

Figure 7

References

[1] J.B. Bell, P. Colella, H.M. Glaz, A second-order projection method for the incompressible Navier–Stokes equations, *J. Comput. Phys.*, **85** (1989), 257–283.

[2] A.J. Chorin, Numerical solution of Navier–Stokes equations, *Math. Comp.*, **22** (1968), 745–762.

[3] A.J. Chorin, On the convergence of discrete approximations to the Navier-Stokes equations, *Math. Comp.*, **23** (1969), 341-353.

[4] W.N. E, J.G. Liu, Projection method I: convergence and numerical boundary layers, *SIAM J. Num. Anal.*, **32** (1995), 1017–1057.

[5] W.N. E, J.G. Liu, Projection method II: Godunov-Ryabenki analysis, *SIAM J. Num. Anal.*, **33** (1996), 1597–1621.

[6] C.P. Easton, Homogeneous boundary conditions for pressure in the MAC method, *J. Comput. Math.*, **9** (1972), 375–379.

[7] E. Hairer, C. Lubich, M. Roche, The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Method, Lecture Notes in Mathematics 1049, Springer-Verlag New York, 1989.

[8] K. Hoffman, R. Kunze, Linear Algebra, Prentice–Hall, Inc., New Jersey, 1971.

[9] T.Y. Hou, B.R. Wetton, Second order convergence of a projection scheme for imcompressible Navier-Stokes equations with boundaries, *SIAM J. Numer. Anal.*, **30** (1993), 609–629.

[10] L.C. Huang, On boundary treatment for numerical solution of the imcompressible Navier-Stokes equations with finite difference methods, *J. Comput. Math.*, **14** (1996), 135–142.

[11] L.C. Huang, Y.D. Wu, The component–consistent pressure correction projection method for the incompressible Navier–Stokes equations, *Computer and Math. with Applic.*, **31** (1996), 1–21.

[12] L.C. Huang, The numerical solution of the unsteady incompressible Navier–Stokes equations on the curvilinear half–staggered mesh, *J. Comput. Math.*, **18** (2000), 521–540.

[13] L.C. Huang, On discrete projection and numerical boundary conditions for the numerical solution of the unsteady incompressible Navier–Stokes equations, Research Report ICM-99-60, 1999.

[14] G.E. Karniadakis, M. Israeli, S.A. Orszag, Higher order splitting methods for the incompressible Navier–Stokes equations, *J. Comput. Phys.,* **97** (1991), 414–443.

[15] J. Kim, P. Moin, Application of a fractional step method to incompressible Navier-Stokes equations, *J. Comput. Phys.,* **59** (1985), 308–323.

[16] S.A. Orszag, M. Israeli, M.O. Deville, Boundary conditions for incompressible flow, *J. Sci. Comput.,* **1** (1996), 75–111.

[17] J.B. Perot, An analysis of the fractional step method, *J. Comput. Phys.,* **108** (1993), 51–58.

[18] J.B. Perot, Letter to the editor: Comments on fractional step method, *J. Comput. Phys.,* **121** (1995), 190–191.

[19] R. Rannacher, On Chorin's projection method for the incompressible Navier– Stokes equations, in Navier–Stokes Equations, Theory and Numerical Methods, edited by J.G. Heywood et al., Springer–Verlag, Berlin, 1992.

[20] J. Shen, On error estimates of projection methods for Navier–Stokes equations: first order schemes, *SIAM J. Numer. Anal.,* **29** (1992), 57–77.

[21] J. Shen, On error estimates of projection methods for Navier–Stokes equations: second order schemes, *Math. Comp.,* **65** (1996), 1039–1065.

[22] U. Schumann, R.A. Sweet, A direct method for the solution of Poisson's equation with Neumann boundary conditions on a staggered grid of arbitrary size, *J. Comput. Phys.,* **20** (1979), 171-181.

[23] R. Temam, Navier-Stokes Equations, Theory and Numerical Analysis, Elsevier Science Pub. B.V., New York, 1984.

[24] R. Temam, Remark on the pressure boundary condition for the projection method, *Theoretical and Computational Fluid Dynamics,* **3** (1991), 181–184.

[25] J. van Kan, A second–order accurate pressure-correction scheme for viscous incompressible flow, *SIAM J. Sci. Stat. Comput.,* **7** (1986), 870–891.

[26] B.R. Wetton, Error analysis for Chorin's original fully discrete projection method and regularizations in space and time, *SIAM J. Numer. Anal.,* **34** (1997), 1683–1697.

[27] N.N. Yanenko, The Method of Fractional Steps, Springer-Verlag New York, 1971.