Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 219 (2006) 976–991
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Runge–Kutta–Chebyshev projection method ☆

Zheming Zheng, Linda Petzold *

Department of Mechanical Engineering, University of California Santa Barbara, Santa Barbara, CA 93106, USA

Received 17 May 2005; received in revised form 11 June 2006; accepted 12 July 2006
Available online 24 August 2006

Abstract

In this paper a fully explicit, stabilized projection method called the Runge–Kutta–Chebyshev (RKC) projection method is presented for the solution of incompressible Navier–Stokes systems. This method preserves the extended stability property of the RKC method for solving ODEs, and it requires only one projection per step. An additional projection on the time derivative of the velocity is performed whenever a second-order approximation for the pressure is desired. We demonstrate both by numerical experiments and by order analysis that the method is second order accurate in time for both the velocity and the pressure. Being explicit, the RKC projection method is easy to implement and to parallelize. Hence it is an attractive candidate for the solution of large-scale, moderately stiff, diffusion-like problems.

Keywords: Projection methods; Incompressible flow; Incompressible Navier–Stokes equations; Extended stability regions; Runge–Kutta–Chebyshev method

1. Introduction

Projection methods [10,9,8,4,5,15,16,11] have been widely used in the solution of incompressible Navier–Stokes equations, written in the nondimensionalized form:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla P = -(\mathbf{u}\cdot\nabla)\mathbf{u} + \frac{1}{Re}\nabla^2\mathbf{u}, \tag{1a}$$

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

with boundary conditions

$$\mathbf{u}|_\Gamma = \mathbf{u}_b, \tag{2}$$

where $\mathbf{u}$ is the velocity, P is the pressure and Re is the Reynolds number.

☆ This work was supported in part by the National Science Foundation under NSF awards NSF/ITR ACI-0086061, NSF/CCF-0428912 and NSF/CTS-0205584.

* Corresponding author. Tel.: +1 805 893 5362; fax: +1 805 893 5435.
E-mail addresses: zhengzhm@engineering.ucsb.edu (Z. Zheng), petzold@engineering.ucsb.edu (L. Petzold).

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.07.005

To solve this problem, projection methods use a fractional step approach, in which an intermediate velocity is obtained by solving the momentum (1a) without regard to the incompressibility constraint (1b), and then a projection of the intermediate velocity onto the divergence-free space is performed to obtain the corrected velocity that satisfies the incompressibility constraint. The pressure is computed in the projection step. In solving the incompressible Navier–Stokes equations with projection methods, much of the difficulty lies in the pressure update. The pressure does not evolve according to a differential equation. Rather, its value is determined by enforcing the incompressibility constraint. It has been observed that while the velocity can be reliably computed to second order accuracy in time, the pressure is typically only first order accurate in time [9].

If the incompressible Navier–Stokes equations are semi-discretized in space, they become a differential-algebraic equation (DAE) system. The mathematical structure of this DAE system is referred to as Hessenberg index 2 [3]. In the DAE context, $\mathbf{u}$ is the differential variable and P is the algebraic variable. The pressure P is further determined to be index 2, where the number of differentiations needed to determine the time derivative of P as a function of $\mathbf{u}$, P and t, is called the index of the DAE. The numerical solution of index 2 DAE systems can be challenging [7].

Often the momentum equation (1a) is solved implicitly in projection methods, due to the stiffness introduced by the viscous term. However we are motivated to develop an explicit projection method by solving the momentum equation explicitly with the use of a special-purpose explicit Runge–Kutta method, the Runge–Kutta–Chebyshev (RKC) method [19,18,20], due to its enhanced stability properties. The RKC method was first proposed by Van der Houwen and Sommeijer [19]. It was designed for the solution of moderately stiff ordinary differential equation (ODE) systems. This method exploits some remarkable properties of a family of explicit Runge–Kutta formulas of the Chebyshev type. This Runge–Kutta method uses the first two stages to achieve second order accuracy. The remainder of the stages are used to enlarge the stability region. It has the property of being stable while retaining a good accuracy using a minimum number of stages. The RKC method has been used in the solution of parabolic partial differential equations discretized by the method of lines [18,20].

The explicit projection method we propose in this paper is called the Runge–Kutta–Chebyshev projection (RKCP) method. In the RKCP method, the momentum equation is solved explicitly by the RKC method. One projection per step, regardless of the number of stages used in the RKC method, is performed at the last stage of the RKC method. An additional projection on the time derivative of the velocity, i.e., the acceleration, is performed to recover the second-order temporal accuracy of the pressure, when it is desired. Because the RKC method was designed for the solution of moderately stiff ODE systems [19,18,20], the RKCP method is particularly well-suited for viscous dominated flows (for example microfluidics [25]).

This paper is organized as follows. A brief review of projection methods and of the RKC method is given in Section 2. In Section 3 the RKCP method is presented, followed in Section 4 by an analysis of the method's stability and order of accuracy for the time discretization, and a discussion of issues related to the additional projection. Numerical examples are presented in Section 5 which verify the order of convergence and the enhanced stability properties. Conclusions are given in Section 6.

2. Background

2.1. Projection methods

First proposed by Chorin [10], projection methods have attracted much research interest [4,5,15,16,11]. A general projection procedure is given in [8], based on the following second-order, time-discrete semi-implicit form of Eq. (1)

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t}+\nabla P^{n+\frac{1}{2}}=-[(\mathbf{u}\cdot\nabla)\mathbf{u}]^{n+\frac{1}{2}}+\frac{1}{2Re}\nabla^2(\mathbf{u}^{n+1}+\mathbf{u}^n), \tag{3a}$$

$$\nabla\cdot\mathbf{u}^{n+1}=0 \tag{3b}$$

with boundary conditions

$$\mathbf{u}^{n+1}|_{\Gamma}=\mathbf{u}_b^{n+1}. \tag{4}$$

The term $[(\mathbf{u}\cdot\nabla)\mathbf{u}]^{n+\frac{1}{2}}$ in Eq. (3) is a second-order approximation to the convective derivative term at time level $t^{n+\frac{1}{2}}$ which is usually computed explicitly.

The numerical scheme consists of the following three steps.

Step 1: Solve for the intermediate velocity u*

$$\frac{\mathbf{u}^{*}-\mathbf{u}^{n}}{\Delta t}+\nabla q=-[(\mathbf{u}\cdot\nabla)\mathbf{u}]^{n+\frac{1}{2}}+\frac{1}{2Re}\nabla^{2}(\mathbf{u}^{*}+\mathbf{u}^{n}), \tag{5}$$

$$B(\mathbf{u}^{*})=0, \tag{6}$$

where q is an approximation to $P^{n+\frac{1}{2}}$ and $B(\mathbf{u}^{*})$ is the boundary condition for $\mathbf{u}^{*}$.
Step 2: Perform the projection

$$\mathbf{u}^{n+1}=\mathbf{u}^{*}-\Delta t\nabla\phi^{n+1}, \tag{7}$$

$$\nabla\cdot\mathbf{u}^{n+1}=0. \tag{8}$$

Step 3: Update the pressure

$$P^{n+\frac{1}{2}}=q+L(\phi^{n+1}). \tag{9}$$

The difference between the methods lies in the choice of the pressure approximation q, the boundary condition $B(\mathbf{u}^{*})$, and the pressure-updating function $L(\phi^{n+1})$. A formula is given in [24,8] which yields a second order (in time) pressure update:

$$P^{n+\frac{1}{2}}=q+\phi^{n+1}-\frac{\Delta t}{2Re}\nabla^{2}\phi^{n+1}. \tag{10}$$

This formula can be obtained by replacing $\mathbf{u}^{*}$ in Eq. (5) with Eq. (7) and subtracting Eq. (3a).

Many of the existing projection methods can be cast in the above framework, including Bell et al. [4,5], Kim and Moin [15], Van Kan [16], Botella [6], E and Liu [11], etc.

The projection method proposed by Bell et al. [4,5] can be recovered with the choices $q=P^{n-\frac{1}{2}}$ and $B(\mathbf{u}^{*})=(\mathbf{u}^{*}-\mathbf{u}_{b}^{n+1})|_{\Gamma}=0$. The advection term is computed using a Godunov type method. The projection step is performed by solving an elliptic problem for ϕ^{n+1} with the boundary condition $\mathbf{n}\cdot\nabla\phi^{n+1}=0$. The pressure update, $\nabla P^{n+\frac{1}{2}}=\nabla P^{n-\frac{1}{2}}+\nabla\phi^{n+1}$, is only first-order.

Kim and Moin [15] choose $q=0$ in Eq. (5), so that the pressure does not appear. This method is referred to as the pressure-free method. The boundary condition for $\mathbf{u}^{*}$ is chosen as $\mathbf{u}^{*}|_{\Gamma}=(\mathbf{u}^{n+1}+\Delta t\nabla\phi^{n})|_{\Gamma}$. A second order accurate velocity can be obtained with this method. The pressure can be computed with the formula $P^{n+\frac{1}{2}}=\phi^{n+1}-\frac{\Delta t}{2Re}\nabla^{2}\phi^{n+1}$, which is second order accurate in time.

Van Kan [16] chooses $q=P^{n}$ and solves the implicit momentum equation with the ADI method. The pressure is updated as $\nabla P^{n+1}=\nabla P^{n}+2\nabla\phi^{n+1}$. This method is second order accurate for the velocity.

E and Liu [11] proposed the gauge method. By using an extrapolation in time, all variables can be computed to second order accuracy.

Recently Liu et al. [17] proposed a continuous projection method. Based on a local truncation error (LTE) analysis, they obtain a sufficient condition for the continuous projection methods to be temporally second order accurate. Second order convergence for both the velocity and the pressure is achieved.

Brown et al. [8] surveyed some of the projection methods and speculated on the effects of boundary conditions and different pressure-update formulas to the order of accuracy. Because performing the projection exactly on a cell-centered grid can cause numerical difficulties, approximate projection methods [2,1] are often used instead. In these methods, the incompressibility constraint is only approximately satisfied. However, approximate projection is highly sensitive to the grid structure and the method used and may be susceptible to instabilities [14].

All of the above methods solve the momentum equation (1a) implicitly, making the overall method implicit. An explicit projection method has been proposed by Fernandez-Feria and Sanmiguel-Rojas [12], but their method can only work for pressure-driven flow problems where Dirichlet boundary conditions are specified for the pressure on part of the boundary.

2.2. Runge–Kutta–Chebyshev method

The RKC method is an explicit Runge–Kutta method for solving moderately stiff ODE systems $y'(t) = f(t,y)$. The first two stages of this Runge–Kutta method are used to obtain second order consistency. The remaining stages are used to extend the region of absolute stability along the negative direction of the real axis of the complex plane. In the context of fluid flow problems, the RKC method is particularly well-suited to viscous dominated flows, where the eigenvalues of the discretized Laplacian operator extend along the negative real axis.

The typical stability region of the RKC method is a long strip along the negative real axis. The length of the segment, β, is a quadratic function of s, the number of stages [20], $\beta(s) = c(s)s^2$, where $c(s)$ is a nearly constant function of s. Therefore, the RKC method is stable when the stability condition, $\Delta t\rho\left(\frac{\partial f}{\partial y}\right) \leqslant \beta(s)$ for a given step size Δt and number of stages s, is satisfied [20,21], where $\frac{\partial f}{\partial y}$ is the Jacobian matrix of the ODE system and $\rho(A)$ is the spectral radius of matrix A.

The RKC method consists of the following steps:

$$\begin{aligned}
&\mathbf{Y}_0 = \mathbf{u}_n,\\
&\mathbf{Y}_1 = \mathbf{Y}_0 + \tilde{\mu}_1\Delta t\mathbf{F}_0,\\
&\mathbf{Y}_j = (1-\mu_j-\nu_j)\mathbf{Y}_0 + \mu_j\mathbf{Y}_{j-1} + \nu_j\mathbf{Y}_{j-2} + \tilde{\mu}_j\Delta t\mathbf{F}_{j-1} + \tilde{\gamma}_j\Delta t\mathbf{F}_0, \quad j = 2,\ldots,s,\\
&\mathbf{u}_{n+1} = \mathbf{Y}_s,
\end{aligned} \tag{11}$$

where $\mathbf{u}_n$ and $\mathbf{u}_{n+1}$ are the numerical solutions at t_n and t_{n+1} respectively, $\mathbf{F}_i = f(t_i,\mathbf{Y}_i)$, Δt is the time step, s is the number of stages and the coefficients $\mu_j, \tilde{\mu}_j, \nu_j, \tilde{\gamma}_j$ are determined for accuracy and stability of the method. Using the recursive property of Chebyshev polynomials, all of these coefficients can be given as analytical expressions for any number of stages. For more details, the reader is referred to [19,20,18].

The RKC method is especially efficient for diffusion-like problems (e.g., viscous dominated flow problems) which are moderately stiff. It cannot handle extremely stiff problems. Recently, an implicit–explicit (IMEX) extension of the explicit RKC method has been proposed by Verwer, Sommeijer and Hundsdorfer [22,23]. This IMEX scheme is designed for diffusion-reaction problems, where the diffusion terms are treated explicitly and the highly stiff reaction terms implicitly.

3. RKC projection method

We are motivated by the above two methods to develop an explicit RKC projection method for solving the semi-discretized incompressible Navier–Stokes equations. It is an index 2 DAE system. The velocities are differential variables and the pressure is an index 2 algebraic variable in such a DAE system.

We write the spatially discretized incompressible Navier–Stokes equations as

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = -\nabla_h P - (\mathbf{u}\cdot\nabla_h)\mathbf{u} + \frac{1}{Re}\nabla_h^2\mathbf{u}, \tag{12a}$$

$$0 = \nabla_h\cdot\mathbf{u} \tag{12b}$$

with boundary conditions

$$\mathbf{u}|_\Gamma = \mathbf{u}_b$$

and initial conditions

$$\mathbf{u}(\mathbf{x}, t=0) = \mathbf{u}_0.$$

∇_h, $\nabla_h\cdot$ and ∇_h^2 in the above equations denote the discretized gradient operator, the discretized divergence operator and the discretized Laplacian operator respectively. The standard central differencing is used for the pressure gradient and the viscous term and central differencing of the conservative form is used for the convection term.

The following is the numerical scheme for the RKCP method. It returns the solution at t_{n+1}, $\mathbf{u}_{n+1}$ and P_{n+1}, given the solution at t_n, $\mathbf{u}_n$ and P_n. In the scheme, s is the number of stages and $\mu_j, \tilde{\mu}_j, \nu_j, \tilde{\gamma}_j$ are the coefficients of the RKC method:

$$\begin{aligned}
&\mathbf{Y}_0 = \mathbf{u}_n,\\
&\mathbf{F}_0 = -(\mathbf{Y}_0\cdot\nabla_h)\mathbf{Y}_0 + \frac{1}{Re}\nabla_h^2\mathbf{Y}_0 - \nabla_h P_n,\\
&\mathbf{Y}_1 = \mathbf{Y}_0 + \tilde{\mu}_1\Delta t\mathbf{F}_0,\\
&\mathbf{F}_{j-1} = -(\mathbf{Y}_{j-1}\cdot\nabla_h)\mathbf{Y}_{j-1} + \frac{1}{Re}\nabla_h^2\mathbf{Y}_{j-1} - \nabla_h P_n,\\
&\mathbf{Y}_j = (1-\mu_j-v_j)\mathbf{Y}_0 + \mu_j\mathbf{Y}_{j-1} + v_j\mathbf{Y}_{j-2} + \tilde{\mu}_j\Delta t\mathbf{F}_{j-1} + \tilde{\gamma}_j\Delta t\mathbf{F}_0 \quad (j = 2,3,\ldots,s).\\
&\text{Solve } \nabla_h^2\phi_1 = \nabla_h\cdot\mathbf{Y}_s \text{ with } \left.\frac{\partial\phi_1}{\partial\mathbf{n}}\right|_\Gamma = 0 \text{ for } \phi_1.\\
&\text{Update } \mathbf{u}_{n+1} = \mathbf{Y}_s - \nabla_h\phi_1 \text{ with boundary conditions } \mathbf{u}_{n+1}|_\Gamma = \mathbf{u}_b^{n+1}.\\
&\text{Update } P_{n+1} = P_n + \frac{2\phi_1}{\Delta t}.
\end{aligned} \tag{13}$$

If a second-order approximation of the pressure is desired at time t_m, an additional projection on the acceleration $\mathbf{F} = \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}$ is performed:

$$\begin{aligned}
&\mathbf{F}_m = -(\mathbf{u}_m\cdot\nabla_h)\mathbf{u}_m + \frac{1}{Re}\nabla_h^2\mathbf{u}_m - \nabla_h P_m.\\
&\text{Solve } \nabla_h^2\phi_2 = \nabla_h\cdot\mathbf{F}_m \text{ with } \left.\frac{\partial\phi_2}{\partial\mathbf{n}}\right|_\Gamma = 0 \text{ for } \phi_2.\\
&\text{Output } \widehat{P}_m = P_m + \phi_2.
\end{aligned} \tag{14}$$

The additional projection is motivated by the fact that the acceleration is also divergence-free. It is a hidden constraint for the acceleration, which is the time derivative of the incompressibility constraint for the velocity. It is important to note that this additional projection, which brings the pressure to the same local order of accuracy as the velocity, is a post-processing step for output only. Using it to update the pressure in the next step can ruin the extended stability properties of the RKCP method.

4. Analysis of the RKCP method

4.1. Temporal order analysis

In this section we will show that the RKCP method is second order accurate in time for both the velocity and the pressure. We consider one time step of the RKCP method from t_n to t_{n+1}. The target is to solve the Eqs. (12) from t_n to t_{n+1}. Let $\Delta t = t_{n+1} - t_n$. Suppose we have the exact solutions as the initial conditions at time step t_n, i.e., $\mathbf{u}_n = \mathbf{u}(t_n)$, $P_n = P(t_n)$, where $\mathbf{u}(t_n)$ and $P(t_n)$ are the exact solutions to the spatially discretized system (12). We have these constraints satisfied:

$$\nabla_h\cdot\mathbf{u}_n = 0, \tag{15}$$

$$\nabla_h\cdot\mathbf{F}_n = 0, \tag{16}$$

where $\mathbf{F}_n = \left.\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\right|_{t_n} = -(\mathbf{u}_n\cdot\nabla_h)\mathbf{u}_n + \frac{1}{Re}\nabla_h^2\mathbf{u}_n - \nabla_h P_n$.

In the first step of the RKCP method, the RKC method is used to solve the momentum equation (17) without regard to the incompressibility constraint to obtain an intermediate velocity $\mathbf{u}^*_{n+1}$.

$$\frac{\mathrm{d}\mathbf{u}^*}{\mathrm{d}t} = -(\mathbf{u}^*\cdot\nabla_h)\mathbf{u}^* + \frac{1}{Re}\nabla_h^2\mathbf{u}^* - \nabla_h q \tag{17}$$

with boundary conditions

$$\mathbf{u}^*|_\Gamma = \mathbf{u}_b$$

and initial conditions

$$\mathbf{u}^*|_{t_n} = \mathbf{u}_n,$$

where q is an approximation of the pressure and is a constant in Eq. (17). We choose $q = P_n$. Suppose $\mathbf{u}^*_{n+1}$ is the numerical solution of Eq. (17) at t_{n+1} obtained by the RKC method, and $\mathbf{u}^*(t_{n+1})$ is the exact solution of Eq. (17) at t_{n+1}. Because the RKC method is second order accurate [20], we have

$$\mathbf{u}^*_{n+1} - \mathbf{u}^*(t_{n+1}) = \mathrm{O}(\Delta t^3). \tag{18}$$

Following the methodology in [13], we obtain

$$\mathbf{u}^*(t_{n+1}) - \mathbf{u}(t_{n+1}) = \frac{\Delta t^2}{2}\nabla_h \frac{\mathrm{d}P}{\mathrm{d}t}\bigg|_{t_n} + \mathrm{O}(\Delta t^3). \tag{19}$$

Since $\mathbf{u}(t)$ is the exact solution of Eq. (12), we have

$$\nabla_h \cdot \mathbf{u}^*_{n+1} = \frac{\Delta t^2}{2}\nabla_h^2 \frac{\mathrm{d}P}{\mathrm{d}t}\bigg|_{t_n} + \mathrm{O}(\Delta t^3). \tag{20}$$

The projection on $\mathbf{u}^*_{n+1}$ is given by:

$$\mathbf{u}_{n+1} = \mathbf{u}^*_{n+1} - \nabla_h \phi_1, \tag{21}$$

$$\nabla_h^2 \phi_1 = \nabla_h \cdot \mathbf{u}^*_{n+1} \tag{22}$$

with boundary conditions

$$\frac{\partial \phi_1}{\partial \mathbf{n}}\bigg|_{\Gamma} = 0.$$

The combination of Eqs. (20) and (22) yields

$$\nabla_h^2 \phi_1 = \frac{\Delta t^2}{2}\nabla_h^2 \frac{\mathrm{d}P}{\mathrm{d}t}\bigg|_{t_n} + \mathrm{O}(\Delta t^3), \tag{23}$$

from which follows that

$$\phi_1 = \frac{\Delta t^2}{2}\frac{\mathrm{d}P}{\mathrm{d}t}\bigg|_{t_n} + \mathrm{O}(\Delta t^3). \tag{24}$$

Finally, it follows from Eqs. (19), (21) and (24) that

$$\mathbf{u}_{n+1} - \mathbf{u}(t_{n+1}) = \mathrm{O}(\Delta t^3). \tag{25}$$

Thus, the RKCP method is locally second order accurate in time for the velocity.

The pressure,

$$P_{n+1} = P_n + \frac{2\phi_1}{\Delta t} = P_n + \Delta t \frac{\mathrm{d}P}{\mathrm{d}t}\bigg|_{t_n} + \mathrm{O}(\Delta t^2) \tag{26}$$

is locally first order accurate in time to $P(t_{n+1})$.

The above analysis actually applies to any second order method to solve the ODE (17) plus the projection.

To obtain a second order accurate approximation of the pressure, an additional projection is required. This projection is done only when a second-order approximation to the pressure is needed for output purposes. It is not necessary for maintaining second order in the velocity. The additional projection at t_m is given by:

$$\mathbf{F}_m = -(\mathbf{u}_m \cdot \nabla_h)\mathbf{u}_m + \frac{1}{Re}\nabla_h^2 \mathbf{u}_m - \nabla_h P_m, \tag{27}$$

$$\nabla_h^2 \phi_2 = \nabla_h \cdot \mathbf{F}_m \tag{28}$$

with boundary conditions

$$\frac{\partial \phi_2}{\partial \mathbf{n}}\bigg|_{\Gamma} = 0.$$

Suppose $\mathbf{u}_m - \mathbf{u}(t_m) = \mathrm{O}(\Delta t^k)$. Then Eq. (27) can be written as

$$\begin{aligned}\mathbf{F}_m &= -(\mathbf{u}(t_m)\cdot\nabla_h)\mathbf{u}(t_m) + \frac{1}{Re}\nabla_h^2\mathbf{u}(t_m) + \mathrm{O}(\Delta t^k) - \nabla_h P(t_m) + \nabla_h P(t_m) - \nabla_h P_m \\ &= \left.\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\right|_{t_m} + \nabla_h P(t_m) - \nabla_h P_m + \mathrm{O}(\Delta t^k). \qquad (29)\end{aligned}$$

Its divergence is given by

$$\nabla_h\cdot\mathbf{F}_m = \nabla_h^2 P(t_m) - \nabla_h^2 P_m + \mathrm{O}(\Delta t^k). \qquad (30)$$

Inserting the above result into Eq. (28) gives

$$\nabla_h^2\phi_2 = \nabla_h^2 P(t_m) - \nabla_h^2 P_m + \mathrm{O}(\Delta t^k). \qquad (31)$$

It follows that

$$\phi_2 = P(t_m) - P_m + \mathrm{O}(\Delta t^k). \qquad (32)$$

After the additional projection, the acceleration is updated by $\mathbf{F}_m = \mathbf{F}_m - \nabla_h\phi_2$ and the pressure is updated by

$$P_m = P_m + \phi_2 = P(t_m) + \mathrm{O}(\Delta t^k). \qquad (33)$$

Now the pressure has the same local order of accuracy as the velocity.

4.2. Error propagation

In this section we look at how the error grows globally. For ease of notation, we replace the divergence and gradient operators, $\nabla_h\cdot$ and ∇_h, with the matrices D and G here. Because of the zero Neumann boundary conditions used for ϕ_1 and ϕ_2, the Laplacian matrix $L = DG$ is singular. We modify the last row of the matrix L so that it is nonsingular. This matrix is denoted by L_m. It is argued in Section 4.3 that replacing L with L_m does not change the solution of $G\phi$.

Neglecting the convection term, we can write the DAE (12) in the matrix form:

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = A\mathbf{u} - GP, \qquad (34a)$$

$$0 = D\mathbf{u} \qquad (34b)$$

with boundary conditions

$$\mathbf{u}|_\Gamma = \mathbf{u}_b$$

and initial conditions

$$\mathbf{u}(\mathbf{x}, t=0) = \mathbf{u}_0.$$

The RKCP scheme (13) is written as follows in the matrix form:

$$\begin{aligned}&\mathbf{Y}_0 = \mathbf{u}_n,\\ &\mathbf{F}_0 = A\mathbf{Y}_0 - GP_n,\\ &\mathbf{Y}_1 = \mathbf{Y}_0 + \tilde{\mu}_1\Delta t\mathbf{F}_0,\\ &\mathbf{F}_{j-1} = A\mathbf{Y}_{j-1} - GP_n,\\ &\mathbf{Y}_j = (1-\mu_j-v_j)\mathbf{Y}_0 + \mu_j\mathbf{Y}_{j-1} + v_j\mathbf{Y}_{j-2} + \tilde{\mu}_j\Delta t\mathbf{F}_{j-1} + \tilde{\gamma}_j\Delta t\mathbf{F}_0 \quad (j = 2,3,\ldots,s).\\ &\text{Solve } L_m\phi_1 = D\mathbf{Y}_s \text{ with } \left.\frac{\partial\phi_1}{\partial\mathbf{n}}\right|_\Gamma = 0 \text{ for } \phi_1.\\ &\text{Update } \mathbf{u}_{n+1} = \mathbf{Y}_s - G\phi_1 \text{ with boundary conditions } \mathbf{u}_{n+1}|_\Gamma = \mathbf{u}_b^{n+1}.\\ &\text{Update } P_{n+1} = P_n + \frac{2\phi_1}{\Delta t}.\end{aligned} \qquad (35)$$

The projection on $\mathbf{u}^*_{n+1}$, Eqs. (21) and (22), is written in the matrix form as

$$\mathbf{u}_{n+1} = \mathbf{u}^*_{n+1} - G\phi_1 = (I - GL_m^{-1}D)\mathbf{u}^*_{n+1} = (I - Q)\mathbf{u}^*_{n+1}, \tag{36}$$

where I is the identity matrix and $Q = GL_m^{-1}D$. As argued in Section 4.3, $I - Q$ is a discrete projection onto the divergence-free space, satisfying $(I - Q)G = 0$ and $D(I - Q) = 0$.

We note that one step of the RKC method applied for $y'(t) = \lambda y$ can be written in the form

$$y_{n+1} = P_s(z)y_n, \tag{37}$$

where $z = \Delta t\lambda$ and P_s is a polynomial of degree s, s being the number of stages. $P_s(z)$ can be written as

$$P_s(z) = 1 + z + \frac{z^2}{2} + \sum_{k=3}^{s} C_k z^k, \tag{38}$$

where C_k are the coefficients of the polynomial. The first three terms of Eq. (38) are used for retaining a second order accuracy and the other terms of Eq. (38) are used for stabilization of the method.

Applying the RKC method to the equation

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = A\mathbf{u} - GP_n \tag{39}$$

for one step yields

$$\mathbf{u}^*_{n+1} = P_s(\Delta tA)\mathbf{u}_n + [I - P_s(\Delta tA)]A^{-1}GP_n. \tag{40}$$

Inserting this expression into Eq. (36) gives

$$\mathbf{u}_{n+1} = (I - Q)P_s(\Delta tA)\mathbf{u}_n + (I - Q)[I - P_s(\Delta tA)]A^{-1}GP_n. \tag{41}$$

Since

$$P_s(\Delta tA) = I + \Delta tA + \frac{\Delta t^2}{2}A^2 + \sum_{k=3}^{s} C_k \Delta t^k A^k \tag{42}$$

and

$$[I - P_s(\Delta tA)]A^{-1} = -\Delta t - \frac{\Delta t^2}{2}A - \sum_{k=3}^{s} C_k \Delta t^k A^{k-1}, \tag{43}$$

we have

$$\mathbf{u}_{n+1} = (I - Q)P_s(\Delta tA)\mathbf{u}_n - \Delta t[(I - Q) + \mathrm{O}(\Delta t)]GP_n. \tag{44}$$

Using the fact that $\mathbf{u}_n$ satisfies the incompressibility constraint, from which it follows that $D\mathbf{u}_n = 0$ and $Q\mathbf{u}_n = 0$, and the relationship $(I - Q)G = 0$, we have

$$\mathbf{u}_{n+1} = (I + \mathrm{O}(\Delta t))\mathbf{u}_n + \mathrm{O}(\Delta t^2)GP_n. \tag{45}$$

For the pressure, we note that

$$P_{n+1} = P_n + \frac{2\phi_1}{\Delta t} \tag{46}$$

and

$$\phi_1 = L_m^{-1}D\mathbf{u}^*_{n+1} = L_m^{-1}D[P_s(\Delta tA)\mathbf{u}_n + [I - P_s(\Delta tA)]A^{-1}GP_n]. \tag{47}$$

Thus P_{n+1} is given by

$$P_{n+1} = P_n + \frac{2\phi_1}{\Delta t} = P_n + \frac{2}{\Delta t}L_m^{-1}D[P_s(\Delta tA)\mathbf{u}_n + [I - P_s(\Delta tA)]A^{-1}GP_n]. \tag{48}$$

Using the relationships above, we obtain

$$P_{n+1} = [2L_m^{-1}DA + \mathrm{O}(\Delta t)]\mathbf{u}_n + P_n + [-2L_m^{-1}D + \mathrm{O}(\Delta t)]GP_n. \tag{49}$$

Multiplying by G, we obtain the recurrence relation for GP_{n+1}

$$GP_{n+1} = [2QA + \mathrm{O}(\Delta t)]\mathbf{u}_n + [I - 2Q + \mathrm{O}(\Delta t)]GP_n, \tag{50}$$

and, since $(I - Q)G = 0$ and $QG = G$,

$$GP_{n+1} = [2QA + \mathrm{O}(\Delta t)]\mathbf{u}_n + [-I + \mathrm{O}(\Delta t)]GP_n. \tag{51}$$

Collecting the results from Eqs. (45) and (51), we can write

$$\begin{pmatrix} \mathbf{u}_{n+1} \\ GP_{n+1} \end{pmatrix} = \begin{bmatrix} I + \mathrm{O}(\Delta t) & \mathrm{O}(\Delta t^2) \\ 2QA + \mathrm{O}(\Delta t) & -I + \mathrm{O}(\Delta t) \end{bmatrix} \begin{pmatrix} \mathbf{u}_n \\ GP_n \end{pmatrix}. \tag{52}$$

Clearly, the eigenvalues of this error propagation matrix are all of magnitude $1 + \mathrm{O}(\Delta t)$. On the interface of the block matrices, a Jordan-like matrix of the form

$$\begin{bmatrix} 1 + \mathrm{O}(\Delta t) & \mathrm{O}(\Delta t^2) \\ \mathrm{O}(1) & -1 + \mathrm{O}(\Delta t) \end{bmatrix}$$

applies. We note that the eigenvalues above and below the interface are different. Thus all eigenvalues are simple. Hence by the standard theory for numerical ODEs (e.g. [3]), the method is 0-stable in the velocities $\mathbf{u}_n$ and in the pressure gradient GP_n. Global second order accuracy in $\mathbf{u}_n$ follows from the results of the previous subsection.

4.3. Discussion

Here we comment on the boundary conditions, the implementation of the projection, and the additional projection. With a staggered grid, boundary conditions are not required for the pressure, but we do need boundary conditions for the intermediate velocity $\mathbf{u}^*$. It is natural and easy to use the same boundary conditions for $\mathbf{u}$. Thus we obtain the boundary conditions for ϕ_1: $\frac{\partial \phi_1}{\partial n}\big|_\Gamma = 0$, and similarly for ϕ_2. These may not be the optimal boundary conditions, but they cause pollution only within a numerical boundary layer of the domain [13].

The projection amounts to solving the Poisson equation with zero Neumann boundary conditions,

$$\begin{aligned} &\nabla^2 \phi = f, \\ &\frac{\partial \phi}{\partial n}\bigg|_\Gamma = 0. \end{aligned} \tag{53}$$

In practice we use the fast Fourier transform (FFT) technique to solve this equation efficiently. The solution of Eq. (53) is not unique (this corresponds to the singularity of the Laplacian matrix $L = DG$). One solution plus any constant still satisfies the equation. Thus an additional constraint is required to make the solution unique. We let

$$\sum \phi_{i,j} = 0. \tag{54}$$

This condition eliminates the singularity of the Laplacian matrix $L = DG$. The last row of the Poisson equation is replaced by Eq. (54). The resulting Laplacian matrix is L_m, which is invertible. It can be argued that Eqs. (55) and (56) have the same solution for $G\phi$, i.e., $G\phi^a = G\phi^b$:

$$L\phi^a = D\mathbf{u}, \tag{55}$$

$$L_m\phi^b = D\mathbf{u}. \tag{56}$$

The solvability condition for Eq. (55) is that the right hand side, $D\mathbf{u}$, is orthogonal to the null space of L^{T}. This is automatically satisfied since the null space of L^{T} is the constant field d and $D^{\mathrm{T}}d = 0$, implying that the inner product $(d, D\mathbf{u})$ is 0. Therefore the solution of Eq. (55) exists but is not unique. The solution ϕ^a plus a constant still satisfies the equation. By solving Eq. (56) instead of Eq. (55), a unique solution, ϕ^b, is obtained. ϕ^a and ϕ^b can differ by a constant, but $G\phi^a = G\phi^b$ since the gradient of a constant is 0. Thus the operator $I - Q = I - GL_m^{-1}D$ is a discrete projection onto the divergence-free fields, with the properties $(I - Q)G = 0$ and $D(I - Q) = 0$.

In the RKCP method, the projection for the intermediate velocity is the fulfillment of the incompressibility constraint Eq. (15). The additional projection for the acceleration is the fulfillment of the hidden constraint Eq. (16), which is the time derivative of Eq. (15). The additional projection brings the pressure to the same order of temporal accuracy as the velocity. Without the additional projection, the second order temporal accuracy for the pressure may not be achieved. However, the velocity is second order accurate with or without the additional projection.

The additional projection has a close relationship to the expression (10). The hidden constraint (16) has been used in deriving the pressure update formula (10). The difference of Eqs. (3a) and (5) with the use of Eq. (7) gives

$$\nabla P^{n+\frac{1}{2}} = \nabla q + \nabla \phi^{n+1} - \frac{\Delta t}{2Re}\nabla^2 \nabla \phi^{n+1}. \tag{57}$$

Since the discretized gradient operator is not invertible, we take the divergence of Eq. (57),

$$\nabla^2 P^{n+\frac{1}{2}} = \nabla^2 q + \nabla^2 \phi^{n+1} - \frac{\Delta t}{2Re}\nabla^2 \nabla^2 \phi^{n+1}, \tag{58}$$

which is the difference of the divergence of the discretized momentum equations (3a) and (5). In doing that, the hidden constraint, i.e., that the divergence of the acceleration is zero, has been used. Eq. (58) can be solved to obtain the pressure update formula (10) by inverting the discretized Laplacian operator. Therefore, the last term of (10) comes from the divergence of the momentum equation, which implies the fulfillment of the hidden constraint. It is equivalent to the additional projection in the RKCP method.

We note that in general the pressure should be updated to second order only for output purposes. The second order pressure obtained by the additional projection should not be used to update the velocities, because this can destroy the absolute stability properties of the method for the viscous term. In the framework of projection methods, the pressure appears as an explicit variable. The extra term which brings the pressure to a higher order usually imposes an additional stability constraint beyond the stability constraints of the numerical ODE algorithm. A full analysis of the stability issue will be forthcoming in a future paper.

The additional projection is also used in the consistent initialization of the pressure. Assuming that we begin with a consistent (divergence-free) initial velocity and an initial guess for the pressure (e.g., 0), the consistent initial condition for the pressure is obtained by the additional projection (14).

Although we have presented the additional projection in the context of the explicit RKC method for solving the momentum equation, it actually works with any (explicit or implicit) ODE solver for solving the momentum equation. The analysis in Section 4.1 applies to any second order method to solve the momentum equation plus the projection. With the additional projection idea, higher order methods can be easily constructed. As long as the higher order velocity is computed, the additional projection will bring the pressure to the same order as the velocity.

5. Numerical experiments

Following the analysis of [17], the numerical solution obtained by a method which is rth order accurate in time and sth order accurate in space can be expressed as

$$\eta^n_{i,j} = \eta(x_i, y_j, t^n) + \alpha^n_{i,j}(\Delta t)^r + \beta^n_{i,j}(\Delta x)^s + \epsilon, \tag{59}$$

where $\eta^n_{i,j}$ is the numerical solution at (x_i, y_j, t^n), $\eta(x_i, y_j, t^n)$ is the exact solution, $\alpha^n_{i,j}(\Delta t)^r$ represents the error associated with the temporal discretization, $\beta^n_{i,j}(\Delta x)^s$ represents the error associated with the spatial discretization and ϵ is the round-off error. To check the temporal convergence, we first compute a reference solution which is obtained using a very small time step ($\Delta t \ll 1$) so that the term $\alpha^n_{i,j}(\Delta t)^r$ can be ignored. The temporally accurate reference solution is approximately

$$\tilde{\eta}^n_{i,j} = \eta(x_i, y_j, t^n) + \beta^n_{i,j}(\Delta x)^s + \epsilon. \tag{60}$$

Then $\|\eta^n - \tilde{\eta}^n\| \sim (\Delta t)^r$. A plot of $\log(\|\eta^n - \tilde{\eta}^n\|)$ vs. $\log(\Delta t)$ gives the information of the order r. Similarly, for the spatial convergence, we compute the reference solution using a very small Δx. The spatially accurate reference solution is approximately

$$\hat{\eta}^n_{i,j} = \eta(x_i, y_j, t^n) + \alpha^n_{i,j}(\Delta t)^r + \epsilon. \tag{61}$$

Thus we have $\|\eta^n - \hat{\eta}^n\| \sim (\Delta x)^s$. A plot of $\log(\|\eta^n - \hat{\eta}^n\|)$ vs. $\log(\Delta x)$ yields the order of spatial convergence.

5.1. Test 1: Forced flow

This test problem is a forced flow problem in two dimensions taken from [11]. It has the following exact solution of the Navier–Stokes equation:

$$\begin{aligned} u(x,y,t) &= -\cos(t)\sin^2(\pi x)\sin(2\pi y), \\ v(x,y,t) &= \cos(t)\sin(2\pi x)\sin^2(\pi y), \\ P(x,y,t) &= -\frac{\sin(t)}{4}(2+\cos(\pi x))(2+\cos(\pi y)) + \frac{\pi^2}{2}\cos(t)(\cos(\pi x)+\cos(\pi y)+\cos(\pi x)\cos(\pi y b)) \end{aligned} \tag{62}$$

with appropriate forcing terms added to the incompressible Navier–Stokes equations (1) to ensure that (62) is the exact solution.

We solved this problem in the domain $[0,1]\times[0,1]$. Dirichlet boundary conditions were applied for both u and v.

First we checked the spatial convergence in Fig. 1. The plot was obtained by comparing the numerical solution with the reference solution at $t = 1$, with Reynolds number $Re = 100$ and time step $\Delta t = 1.0\times 10^{-4}$. The reference solution, $\mathbf{u}_0$ and P_0, was obtained with $\Delta x = \Delta y = \frac{1}{512}$ (512×512 grid). Second order spatial convergence was observed in Fig. 1.

The temporal convergence is examined in Fig. 2. At $t = 1$ with Reynolds number $Re = 100$ and $\Delta x = \Delta y = \frac{1}{128}$, the numerical solutions for a range of time steps were compared with the reference solution, which was obtained with $\Delta t = 1.0\times 10^{-5}$. An additional projection was performed for the pressure when outputting it at $t = 1$. It can be seen from the figure that both the velocity and the pressure are second order convergent. Without the additional projection, order reduction is observed on the pressure as shown in Fig. 3.

Next we constructed a test to see how the method behaves with different Reynolds numbers. At 128×128 grids, fixing the time step to $\Delta t = 0.01$, we computed the difference between the numerical solution and the exact solution, and counted the number of stages used in the RKCP method (the number of stages is determined adaptively in the code by the algorithm described in [18]). These are plotted in Figs. 4 and 5. The error

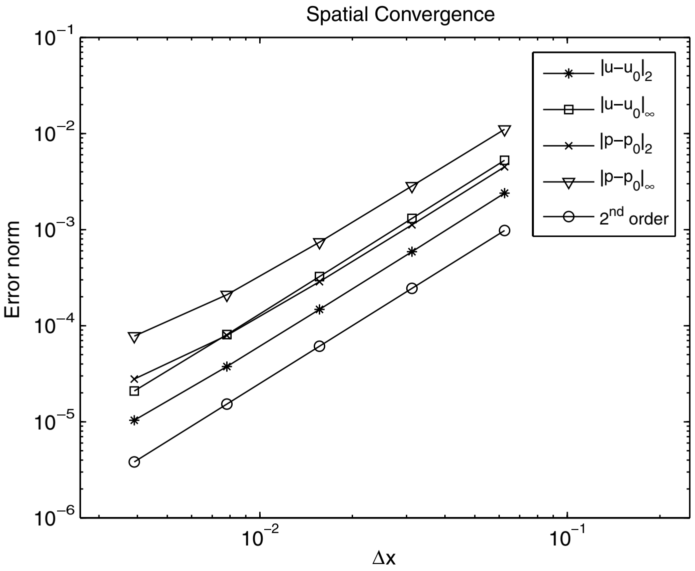

Fig. 1. Spatial convergence in the forced flow problem with $Re = 100$ and $\Delta t = 1.0\times 10^{-4}$.

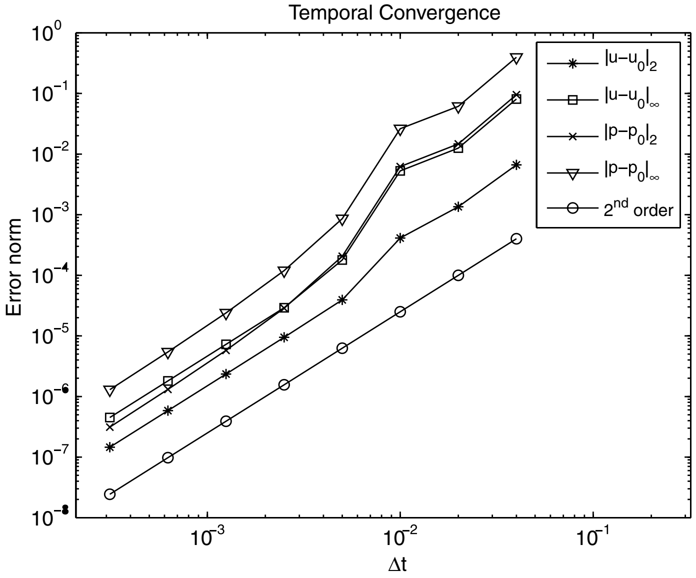

Fig. 2. Temporal convergence in the forced flow problem at 128 × 128 grids and $Re = 100$.

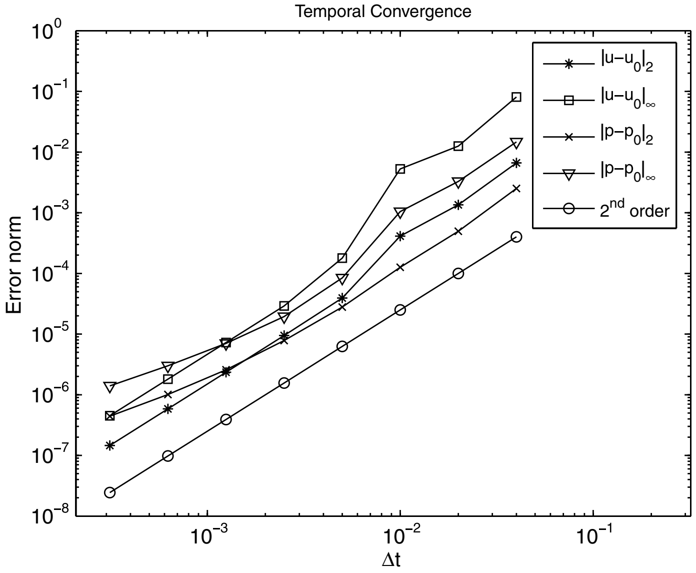

Fig. 3. Temporal convergence in the forced flow problem at 128 × 128 grids and $Re = 100$, without the last additional projection.

plotted in Fig. 5 is the 2-norm of the difference between the exact solution and the numerical solution. We can see that for moderately stiff problems (moderate Reynolds number) the RKCP method uses a small number of stages and the error is controlled well, indicating that the method is practically efficient.

When checking the absolute stability properties of the RKCP method with respect to how they compare with those of RKC, we neglected the convection term and solved the Stokes equations. We did this because in practice, most of the stiffness will come from the viscous term, and because the analysis of stability region for RKC in [19,18,20] was done only for diffusion problems. In this test, we found the maximal stable time step-size by numerical experiment for each number of stages. The results are plotted in Fig. 6. In Fig. 6, the RKCP method with m projections per step is the RKCP method with projection at each stage of the RKC method, which has the same stability region as the RKC method (this is straightforward to show using

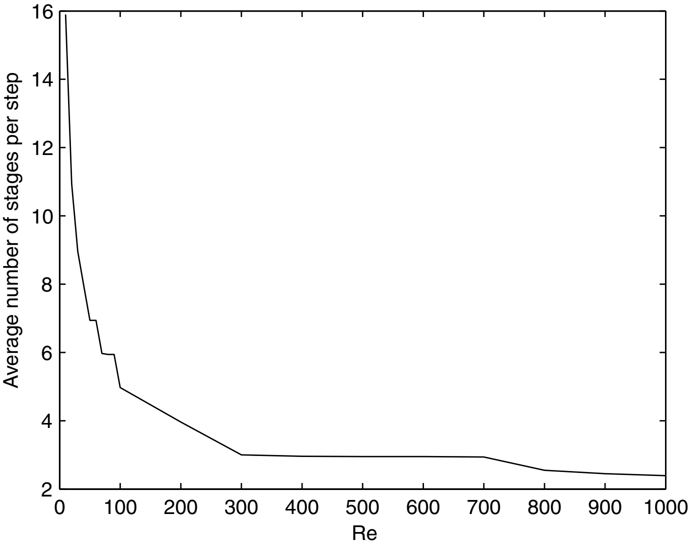

Fig. 4. Average number of stages vs. different Reynolds numbers in the forced flow problem at 128 × 128 grids and $\Delta t = 1.0 \times 10^{-2}$.

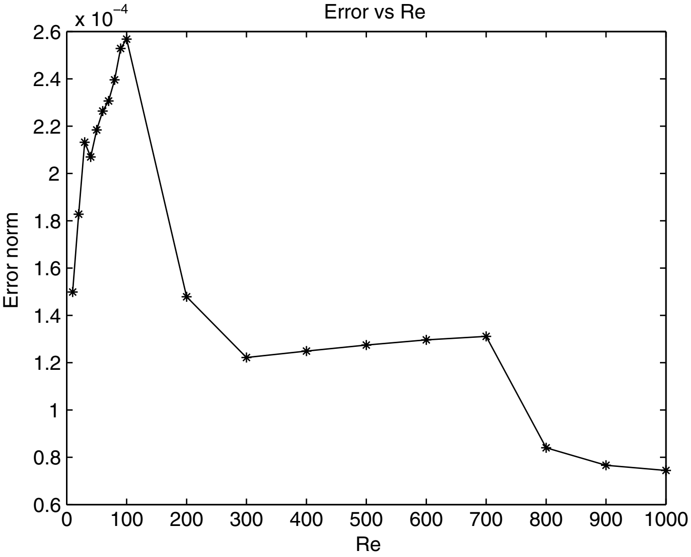

Fig. 5. Error of solution vs. different Reynolds numbers in the forced flow problem at 128 × 128 grids and $\Delta t = 1.0 \times 10^{-2}$.

methodology from DAE theory). By comparing the stability of these two methods, we can see how the stability properties of the RKC method are preserved in the RKCP method. Fig. 6 shows that the enhanced stability properties of the RKC method are indeed preserved in the RKCP method. And as we know already for RKC, the maximal stable step-size for RKCP increases quadratically with the number of stages.

5.2. Test 2: Driven cavity

In this test we implemented the RKCP method on the driven cavity problem. The computational domain is $[0,1] \times [0,1]$. Zero Dirichlet boundary conditions were applied for the velocities, except that on the top boundary $y = 1$, the x-direction velocity u was set to 1. The initial condition was set to 0, meaning that the fluid is initially still. Although the solution eventually reaches a steady state, we wanted to test the time accuracy on

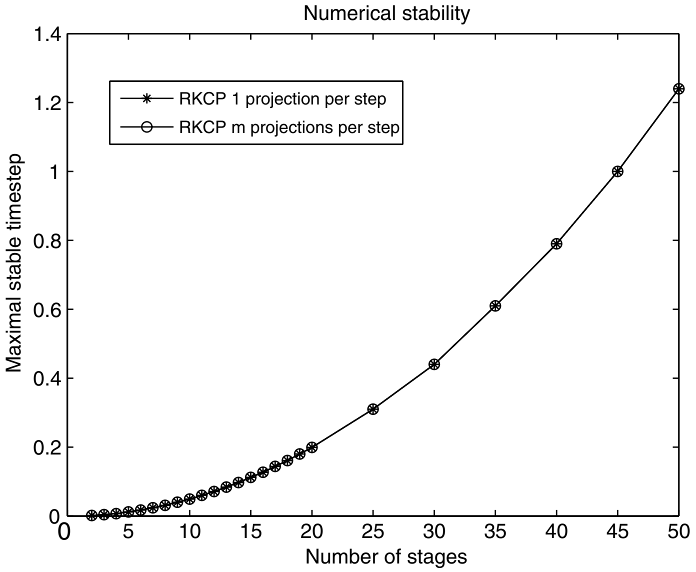

Fig. 6. Maximal stable step-size vs. number of stages for the Stokes equations at 128 × 128 grids and $Re = 100$ (the stability region of the RKC method is equivalent to that of the RKCP method with m projections per step).

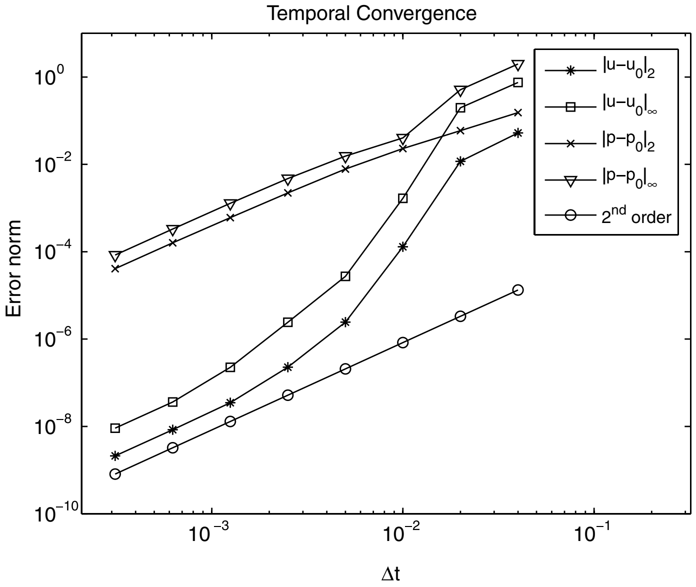

Fig. 7. Temporal convergence in the driven cavity problem at 128 × 128 grids and $Re = 100$.

the unsteady part. Thus we used the solution at $t = 1$ to test the order of accuracy in time. The test is done for $Re = 100$ at 128 × 128 grids. Second order convergence was observed as shown in Fig. 7, where the reference solution, $\mathbf{u}_0$ and P_0, was obtained with $\Delta t = 1.0 \times 10^{-5}$.

5.3. Additional test: absolute stability

Fig. 6 shows the stability of the RKCP method applied to the Stokes equation, where the convection term is neglected. We are interested in solving viscous dominated Navier–Stokes equations. Thus we used the RKCP method to solve the Navier–Stokes equations for the forced flow problem with Reynolds number $Re = 5$ on a

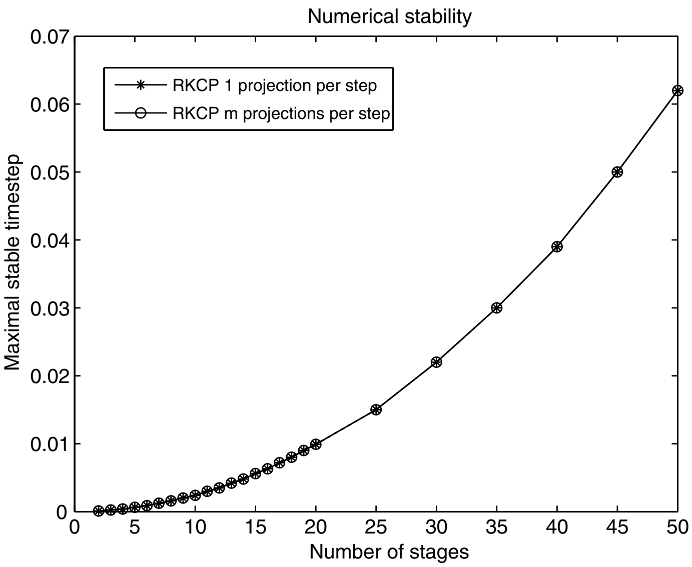

Fig. 8. Maximal stable step-size vs. number of stages for the Navier–Stokes equations at 128 × 128 grids and $Re = 5$ (the stability region of the RKC method is equivalent to that of the RKCP method with m projections per step).

128 × 128 grid. Similar to Fig. 6, we found the maximal stable time step-size by numerical experiment for each number of stages. The stability properties of the RKCP method for solving Navier–Stokes equations are shown in Fig. 8. In Fig. 8, the RKCP method with m projections per step has the same stability region as the RKC method. When solving the driven cavity problem with the same Reynolds number and the same spatial grid, nearly the same plot as Fig. 8 was obtained. We arrived at the conclusion that the enhanced stability properties of the RKC method are preserved in the RKCP method for solving viscous dominated Navier–Stokes equations.

6. Conclusions

In this paper a fully explicit, stabilized projection method called the Runge–Kutta–Chebyshev projection (RKCP) method for the solution of the incompressible Navier–Stokes equations was presented. Numerical experiments show that the RKCP method preserves the extended stability property of the RKC method and it requires only one projection per step, independent of the number of Runge–Kutta stages. An additional projection is performed whenever a second-order approximation for the pressure is desired. We showed that the RKCP method is second order accurate in time for both the velocity and the pressure. An order analysis and a convergence analysis were presented. Being explicit, the RKC projection method is easy to implement and to parallelize, hence it is a promising candidate for the solution of large-scale problems, especially moderately stiff, diffusion-like problems.

Acknowledgments

The authors thank Bernd Simeon for many insightful discussions while writing this paper, and Jan Verwer for his constructive comments.

References

[1] A.S. Almgren, J.B. Bell, W.Y. Crutchfield, Approximate projection methods: Part I. Inviscid analysis, SIAM J. Sci. Comput. 22 (2000) 1139.

[2] A.S. Almgren, J.B. Bell, W.G. Szymczak, A numerical method for the incompressible Navier–Stokes equations based on an approximate projection, SIAM J. Sci. Comput. 17 (1996) 358.

[3] U.M. Ascher, L.R. Petzold, Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations, Society for Industrial and Applied Mathematics, 1998.
[4] 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.
[5] J.B. Bell, P. Colella, L.H. Howell, An efficient second-order projection method for viscous incompressible flow, in: Proceedings of the 10th AIAA Computational Fluid Dynamics Conference, AIAA, 1991, p. 360.
[6] O. Botella, On the solution of the Navier–Stokes equations using Chebyshev projection schemes with third-order accuracy in time, Comp. Fluids 26 (1997) 107.
[7] K.E. Brenan, S.L. Campbell, L.R. Petzold, Numerical Solution of Initial-value Problems in Differential-Algebraic Equations, Society for Industrial and Applied Mathematics, 1996.
[8] D.L. Brown, R. Cortez, M.L. Minion, Accurate projection methods for the incompressible Navier–Stokes equations, J. Comput. Phys. 168 (2001) 464.
[9] A.J. Chorin, On the convergence of discrete approximation to the Navier–Stokes equations, Math. Comput. 23 (1969) 341.
[10] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comput. 22 (1997) 745.
[11] W. E, J.G. Liu, Gauge method for viscous incompressible flows, Comm. Math. Sci. 1 (2003) 317.
[12] R. Fernandez-Feria, E. Sanmiguel-Rojas, An explicit projection method for solving incompressible flows driven by a pressure difference, Comp. Fluids 33 (2004) 463.
[13] P.M. Gresho, On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 1: theory, Int. J. Numer. Fluids 11 (1990) 587.
[14] R.D. Guy, A.L. Fogelson, Stability of approximate projection methods on cell-centered grids, J. Comput. Phys. 203 (2005) 517.
[15] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, J. Comput. Phys. 59 (1985) 308.
[16] J.V. Kan, A second-order accurate pressure-correction scheme for viscous incompressible flow, SIAM J. Sci. Stat. Comput. 7 (1986) 871.
[17] M. Liu, Y. Ren, H. Zhang, A class of fully second order accurate projection methods for solving the incompressible Navier–Stokes equations, J. Comput. Phys. 200 (2004) 325.
[18] B.P. Sommeijer, L.F. Shampine, J.G. Verwer, RKC: an explicit solver for parabolic PDEs, J. Comp. Appl. Math. 88 (1997) 315.
[19] P.J. van der Houwen, B.P. Sommeijer, On the internal stability of explicit, *m*-stage Runge–Kutta methods for large *m*-values, Z. Angew. Math. Mech. 60 (1980) 479.
[20] J.G. Verwer, Explicit Runge–Kutta methods for parabolic differential equations, Appl. Numer. Math. 22 (1996) 359.
[21] J.G. Verwer, W.H. Hundsdorfer, B.P. Sommeijer, Convergence properties of the Runge–Kutta–Chebyshev method, Numer. Math. 57 (1990) 157.
[22] J.G. Verwer, B.P. Sommeijer, An implicit–explicit Runge–Kutta–Chebyshev scheme for diffusion-reaction equations, SIAM J. Sci. Comput. 25 (2004) 1824.
[23] J.G. Verwer, B.P. Sommeijer, W. Hundsdorfer, RKC time-stepping for advection–diffusion-reaction problems, J. Comput. Phys. 201 (2004) 61.
[24] R. Verzicco, P. Orlandi, A finite-difference scheme for three-dimensional incompressible flows in cylindrical coordinates, J. Comput. Phys. 123 (1996) 402.
[25] J. Waltz, Microfluidics simulation using adaptive unstructured grids, Int. J. Numer. Meth. Fluids 46 (2004) 939.