11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
E. Oñate, J. Oliver and A. Huerta (Eds)

PERFORMANCE OF PROJECTION METHODS FOR LOW-REYNOLDS-NUMBER FLOWS

Fabricio S. Sousa[*], **Cassio M. Oishi**[†] **and Gustavo C. Buscaglia**[*]

[*] Department of Applied Mathematics and Statistics
Universidade de São Paulo (ICMC-USP), São Carlos, SP, Brazil.
fsimeoni@icmc.usp.br, gustavo.buscaglia@icmc.usp.br
[†] Department of Mathematics and Computer Science,
Universidade Estadual Paulista (UNESP), Presidente Prudente, SP, Brazil.
oishi@fct.unesp.br

Key words: Projection method; Navier-Stokes equations; Incompressible flow; Algebraic splitting; Low Reynolds number; Microfluidics

Abstract. There exists growing interest in modelling flows at millimetric and micrometric scales, characterised by low Reynolds numbers ($Re \ll 1$). In this work, we investigate the performance of projection methods (of the algebraic-splitting kind) for the computation of steady-state simple benchmark problems. The most popular approximate factorization methods are assessed, together with two so-called exact factorization methods. The results show that: (a) The error introduced by non-incremental schemes on the steady state solution is unacceptably large even in the simplest of flows. This is well-known for the basic first order scheme and motivated variants aiming at increased accuracy. Unfortunately, the variants studied either become unstable for the time steps of interest, or yield steady states with larger error than the basic first order scheme. (b) Incremental schemes have an optimal time step δt^* so as to reach the steady state with minimum computational effort. Taking $\delta t = \delta t^*$ the code reaches the steady state in not less than a few hundred time steps. Such a cost is significantly higher than that of solving the velocity-pressure coupled system, which can compute the steady state in one shot. (c) The main difficulty, however, is that if δt is chosen too large (in general δt^* is not known), then thousands or tens of thousands of time steps are required to reach the numerical steady state with incremental projection methods. The numerical solutions of these methods follow a time-step-dependent spurious transient which makes the computation of steady states prohibitively expensive.

1 INTRODUCTION

The Navier-Stokes equations modelling incompressible viscous Newtonian flows are

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{uu}) - \nu \nabla^2 \mathbf{u} + \nabla p = \mathbf{f} \quad \text{in} \quad [0,T] \times \Omega, \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in} \quad [0,T] \times \Omega, \tag{2}$$

where t is time, $\mathbf{u}$ is the velocity vector field, p is the pressure, ν is the kinematic viscosity of the fluid, $\mathbf{f}$ is a body-force, all defined in a domain $\Omega \subset \mathbb{R}^d$ ($d = 2$ or 3) for $t \in [0,T]$.

A significant fact in numerical solving the unsteady incompressible Navier-Stokes equations is that the system of equations (1)–(2) couple the velocity and pressure fields. The numerical solution of this coupled system is many times discarded because it leads to many computational difficulties, often related to its large size and non-linearities. Methods that use the decoupling of the velocity and pressure fields in (1) and (2) were originally considered by Chorin [1] and Temam [2], being known as "projection methods" or "fractional step methods". The basic idea is the use of the momentum equation (1) to compute a tentative velocity field, which does not generally satisfy the continuity equation (2). Then the intermediate velocity is projected to produce a solenoidal velocity and a gradient fields.

An interesting form of analysis of projection methods is the so called "algebraic splitting". In this method, the decoupling of the Navier-Stokes equations (1) and (2) can be seen as an factorization of the (coupled) discrete system of equations. The advantage of this approach is that it eliminates any artificial boundary conditions that should be imposed to pressure, therefore not influencing the splitting error. Approximate factorization methods were initially studied by Dukowicz & Dvinsky [5], having derived a second-order projection method. Soon after, Perot [6] presented a study using LU decomposition. According to [6], the construction of a high-order projection method via LU factorization can overcome some difficulties found in the standard analysis. Other studies on algebraic factorization were also conducted by Quarteroni and coworkers [7].

In order to eliminate completely the error introduced in the projection methods generated by approximate factorization, known as splitting error, exact factorizations were proposed in the literature. In [8], the authors presented a detailed study of approximate and exact factorizations for the construction of high-order projection methods. An important work describing an exact factorization method was presented by Chang et al. [9], where they based their method on the construction of divergence and gradient null spaces. Recently, Zhang et al. [10] introduced an exact factorization technique which is independent of time discretization methods and does not involve approximate inversion of matrices. The exact factorization schemes by Chang et al. [9] and Zhang et al. [10] were assessed mostly in moderate to high Reynolds numbers flows, not being put to the test for very low Reynolds numbers flows. In this work, we test popular algebraic splitting methods for low Reynolds flows, as happening in microfluidics applications.

2 OVERVIEW OF DISCRETIZATIONS

Our objective here is to describe pressure-segregation techniques for the numerical approximation of (1)-(2). These techniques are obtained by algebraic splitting of the fully coupled discretization, which in matrix form reads

$$\begin{bmatrix} \mathsf{A} & \mathsf{G} \\ \mathsf{D} & 0 \end{bmatrix} \begin{bmatrix} U^{n+1} \\ P^{n+\theta} \end{bmatrix} = \begin{bmatrix} r^n \\ 0 \end{bmatrix}, \tag{3}$$

$$\mathsf{A} = \delta t^{-1}\mathsf{I} - \theta\,\nu\,\mathsf{L}, \qquad r^n = \mathsf{N}U^n - \mathsf{T}(U^n) + b^n, \qquad \mathsf{N} = \delta t^{-1}\mathsf{I} + (1-\theta)\,\nu\,\mathsf{L} \tag{4}$$

where $U^n \in \mathbb{R}^{N+M}$ is the vector of velocity unknowns, assumed to be N for x-velocity components and M for y-velocity components, $P^{n+\theta} \in \mathbb{R}^Q$ is the vector of pressure unknowns and $\theta \in (0,1]$. In (4), $\mathsf{T}(U^n)$ represents the discrete operator of the nonlinear terms. Since we focus on low Reynolds numbers in this work, this term is approximated explicitly (and sometimes neglected altogether). Note that $\mathsf{A} \in \mathbb{R}^{(M+N)\times(M+N)}$, and $\mathsf{L} \in \mathbb{R}^{(M+N)\times(M+N)}$ is the sub-matrix corresponding to the discretization of the viscous operator, $\mathsf{G} \in \mathbb{R}^{(M+N)\times Q}$ corresponds to the discrete gradient operator and $\mathsf{D} \in \mathbb{R}^{Q\times(M+N)}$ is the discrete divergence operator. The vector $r^n \in \mathbb{R}^{(M+N)}$ in (4) contains all the quantities known at the current time level n, and additional information regarding the discretization of forces and boundary conditions, assumed contained in vector $b^n \in \mathbb{R}^{(M+N)}$.

In all the numerical examples the spatial discretization is performed with a Marker-and-Cell scheme, in which velocities (in x- and y-direction) are stored at the cell edges (positions $(i+\frac{1}{2}, j)$ and $(i, j+\frac{1}{2})$, respectively) while the pressure is stored at cell centers (positions (i,j)). This scheme is known to be div-stable, in the sense that convergence is attained without the appearance of spurious pressure modes. It is widely used and has been extensively documented in the literature.

A numerical approach that proposes the direct resolution of system (3) is said to be *coupled* or *monolithic*. There are some disadvantages in doing that, especially in terms of computational costs and matrix conditioning, which motivates the algebraic splitting of the system into subsystems of smaller size. Typically, matrix A is symmetric and positive, since L (upon enforcing the boundary conditions) and I are as well. The point of departure of algebraic splitting methods is an exact factorization of the matrix in (3) into two, three or four factors [8]. Considering the standard block LU factorization (see [5]), the system (3) can be rewritten as

$$\begin{bmatrix} \mathsf{A} & 0 \\ \mathsf{D} & -\mathsf{D}\mathsf{A}^{-1}\mathsf{G} \end{bmatrix} \begin{bmatrix} \mathsf{I} & \mathsf{A}^{-1}\mathsf{G} \\ 0 & \mathsf{I} \end{bmatrix} \begin{bmatrix} U^{n+1} \\ P^{n+\theta} \end{bmatrix} = \begin{bmatrix} r^n \\ 0 \end{bmatrix}. \tag{5}$$

Note that the system (5) can be solved in two stages:

$$\begin{bmatrix} \mathsf{A} & 0 \\ \mathsf{D} & -\mathsf{D}\mathsf{A}^{-1}\mathsf{G} \end{bmatrix} \begin{bmatrix} \tilde{U} \\ \tilde{P} \end{bmatrix} = \begin{bmatrix} r^n \\ 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \mathsf{I} & \mathsf{A}^{-1}\mathsf{G} \\ 0 & \mathsf{I} \end{bmatrix} \begin{bmatrix} U^{n+1} \\ P^{n+\theta} \end{bmatrix} = \begin{bmatrix} \tilde{U} \\ \tilde{P} \end{bmatrix}, \tag{6}$$

where the vectors $\tilde{U}$ and $\tilde{P}$ are often regarded as intermediate velocity and pressure, respectively. The systems appearing in (6) constitute an exact algebraic splitting method for the numerical solution of the Navier-Stokes equations. Unfortunately, it involves the matrix A^{-1} which is unaffordable to compute.

Algebraic splitting methods are approximations of the exact splitting above. These methods define matrices B_1 and B_2 which will approximate A^{-1} in its first and second occurrences, respectively. The numerical method is then defined by

$$\begin{bmatrix} \mathsf{A} & 0 \\ \mathsf{D} & -\mathsf{D}\mathsf{B}_1\mathsf{G} \end{bmatrix} \begin{bmatrix} \tilde{U} \\ \tilde{P} \end{bmatrix} = \begin{bmatrix} r^n - \gamma \mathsf{G} P^{n+\theta-1} \\ 0 \end{bmatrix} \tag{7}$$

$$\begin{bmatrix} \mathsf{I} & \mathsf{B}_2\mathsf{G} \\ 0 & \mathsf{I} \end{bmatrix} \begin{bmatrix} U^{n+1} \\ P^{n+\theta} \end{bmatrix} = \begin{bmatrix} \tilde{U} + \gamma\, \mathsf{B}_2\mathsf{G}\, P^{n+\theta-1} \\ \tilde{P} + \gamma P^{n+\theta-1} \end{bmatrix} \tag{8}$$

where $\gamma \in \{0,1\}$ is a parameter introduced to make $\tilde{P}$ "incremental". If $\gamma = 0$ then $\tilde{P} = P^{n+\theta}$ and the method is said to be "non-incremental". If $\gamma = 1$ then $\tilde{P} = P^{n+\theta} - P^{n+\theta-1}$ and the method is said to be "incremental".

The sequence of velocity and pressure solutions generated by the different algebraic splitting schemes will only coincide with the monolithic solution if $\mathsf{B}_1 = \mathsf{B}_2 = \mathsf{A}^{-1}$ (interestingly, the value of γ is irrelevant because it cancels out). In all other cases there appears a *splitting error*, both in velocity (i.e.; $U^n - U^n_{\text{coupled}}$) and in pressure (i.e.; $P^{n+\theta} - P^{n+\theta}_{\text{coupled}}$). The splitting error can be estimated, but this is quite involved, requiring lots of block matrix manipulations, and will not be detailed in this work for the sake of brevity.

Remark: Applying the discrete divergence D to the first equation of (8), and further eliminating $\tilde{U}$, one obtain $\mathsf{D}U^{n+1} = \mathsf{D}(\tilde{U} - \mathsf{B}_2\mathsf{G}\tilde{P}) = \mathsf{D}\,(\mathsf{B}_1 - \mathsf{B}_2)\,\mathsf{G}\tilde{P}$, which is exactly zero if both approximations to the inverse of A are the same. In this work, we consider only schemes where $\mathsf{B}_1 = \mathsf{B}_2 = \mathsf{B}$, exactly satisfying the (discrete) incompressibility equation.

The system of algebraic equations to be solved at each time step for the algebraic splitting methods considered here can thus be summarized as

$$\mathsf{A}\widetilde{U} = r^n - \gamma \mathsf{G} P^{n+\theta-1} \tag{9}$$

$$\mathsf{DBG}\left(P^{n+\theta} - \gamma P^{n+\theta-1}\right) = \mathsf{D}\widetilde{U} \tag{10}$$

$$U^{n+1} = \widetilde{U} - \mathsf{BG}\left(P^{n+\theta} - \gamma P^{n+\theta-1}\right). \tag{11}$$

In the next sections we recall several algebraic splitting methods of the type discussed above, which have their temporal accuracy well documented in the literature [7, 11, 12].

2.1 First-order projection scheme

From Eq. (4), one can expand A^{-1} as

$$\mathsf{A}^{-1} = \delta t \mathsf{I}_{\mathsf{u}} + \delta t^2 \theta\, \nu\, \mathsf{L} + \cdots = \delta t \mathsf{I}_{\mathsf{u}} + \sum_{i=1}^{\infty} \delta t^{i+1} \left(\theta\, \nu\, \mathsf{L}\right)^i, \tag{12}$$

a series that is known to be convergent if $\delta t < \|\theta\, \nu\, \mathsf{L}\|_2^{-1}$ is satisfied (see [13]). The first-order projection method defines $\gamma = 0$ and B as the first term of the expansion ($\mathsf{B} = \delta t \mathsf{I}_\mathsf{u}$), which corresponds to the algebraic system

$$\mathsf{A}\widetilde{U} = r^n; \qquad \delta t \mathsf{D}\mathsf{G}P^{n+\theta} = \mathsf{D}\widetilde{U}; \qquad U^{n+1} = \widetilde{U} - \delta t \mathsf{G}P^{n+\theta}. \tag{13}$$

As its name suggests, this method has splitting error of order $O(\delta t)$.

2.2 Incremental projection scheme

This method corresponds to taking $\gamma = 1$ and $\mathsf{B} = \delta t\, \mathsf{I}_\mathsf{u}$, and is thus the incremental version of the previous method. The equations that define the method are

$$\mathsf{A}\widetilde{U} = r^n - \mathsf{G}P^{n+\theta-1}; \quad \delta t \mathsf{D}\mathsf{G}\widetilde{P} = \mathsf{D}\widetilde{U}; \quad U^{n+1} = \widetilde{U} - \delta t \mathsf{G}\widetilde{P}; \quad P^{n+\theta} = P^{n+\theta-1} + \widetilde{P}. \tag{14}$$

Its splitting error is of order $O(\delta t^2)$, so that the scheme is second order accurate in time when $\theta = 0.5$ [3, 4].

2.3 Perot's second order approximation to A^{-1}

Another way to improve the order of accuracy in time of the first-order projection scheme [6] is to consider one more term in the expansion (12), i.e., $\mathsf{B} = \delta t \mathsf{I}_\mathsf{u} + \theta\, \nu\, \delta t^2\, \mathsf{L}$. One arrives at the following method (non-incremental version)

$$\mathsf{A}\widetilde{U} = r^n; \quad \mathsf{D}\left(\delta t \mathsf{I}_\mathsf{u} + \theta\nu\delta t^2 \mathsf{L}\right) \mathsf{G}P^{n+\theta} = \mathsf{D}\widetilde{U}; \quad U^{n+1} = \widetilde{U} - \left(\delta t \mathsf{I}_\mathsf{u} + \theta\nu\delta t^2 \mathsf{L}\right) \mathsf{G}P^{n+\theta}. \tag{15}$$

The splitting error of this method is $O(\delta t^2)$ when the time discretization employed for the coupled problem is second order in time [6].

2.4 Zhang's pseudo-exact factorization

Zhang et al. [10] introduced a pseudo-exact factorization technique which is based on a modified version of the system (3), given by

$$\begin{bmatrix} \mathsf{A} & \mathsf{A}\mathsf{G} \\ \mathsf{D} & 0 \end{bmatrix} \begin{bmatrix} U^{n+1} \\ \Phi^{n+\theta} \end{bmatrix} = \begin{bmatrix} r^n \\ 0 \end{bmatrix}, \tag{16}$$

where $\Phi^{n+\theta}$ is defined as a "numerical pressure" or "gauge pressure". In this case, systems (3) and (16) are equivalent if

$$\mathsf{A}\mathsf{G}\Phi^{n+\theta} = \mathsf{G}P^{n+\theta}. \tag{17}$$

The matrix in (16) can be exactly decomposed as

$$\begin{bmatrix} \mathsf{A} & \mathsf{A}\mathsf{G} \\ \mathsf{D} & 0 \end{bmatrix} = \begin{bmatrix} \mathsf{A} & 0 \\ \mathsf{D} & -\mathsf{D}\mathsf{G} \end{bmatrix} \begin{bmatrix} \mathsf{I} & \mathsf{G} \\ 0 & \mathsf{I} \end{bmatrix}, \tag{18}$$

resulting in the following segregated system

$$\mathsf{A}\widetilde{U} = r^n; \qquad \mathsf{D}\mathsf{G}\Phi^{n+\theta} = \mathsf{D}\widetilde{U}; \qquad U^{n+1} = \widetilde{U} - \mathsf{G}\Phi^{n+\theta}. \tag{19}$$

According to Zhang et al [10], the real pressure is obtained by taking the (discrete) divergence of Eq. (17), resulting in the following Poisson-like equation

$$\mathsf{DG}P^{n+\theta} = \mathsf{DAG}\Phi^{n+\theta}\ . \tag{20}$$

Notice that the additional discrete Poisson-like equation (20) can be used to compute pressure in a post-processing stage, since it is not used for the evolution of the other variables. The authors in [10] claim that this method is an "exact factorization method". However, since matrices G and D are not square, equation (17) is an overdetermined system for $P^{n+\theta}$, since there are more equations (i.e.; $M+N$) for the components of $P^{n+\theta}$ than the number of components itself (i.e.; Q). To understand the origin of the splitting error, let us write $\Phi^{n+\theta} = (\mathsf{DAG})^{-1}\mathsf{DG}P^{n+\theta}$, and by substitution into Eq. (19) one arrives at the equivalent system

$$\begin{bmatrix} \mathsf{A} & \mathsf{AG}(\mathsf{DAG})^{-1}\mathsf{DG} \\ \mathsf{D} & 0 \end{bmatrix} \begin{bmatrix} U^{n+1} \\ P^{n+\theta} \end{bmatrix} = \begin{bmatrix} r^n \\ 0 \end{bmatrix}, \tag{21}$$

which can be compared to the original coupled system (3). The splitting error of this method turns out to be first order in time. In fact, one can conclude that this pseudo-exact factorization method is a non-incremental scheme that corresponds to choosing $\mathsf{B} = \mathsf{G}(\mathsf{DAG})^{-1}\mathsf{D}$. An incremental version is readily obtained by taking this choice of B and $\gamma = 1$ in (9)-(11).

2.5 Exact factorization

In [9], an exact fractional step method for incompressible fluid flows is proposed. The scheme can be interpreted as a discrete streamfunction method; however the disadvantages of the streamfunction formulation are eliminated when the change of variables is applied to the already discrete problem. This method relies on the construction of null spaces for the discrete operators D and G, i.e., the computation of matrices R and C, such that $\mathsf{RG} = 0$ and $\mathsf{DC} = 0$, respectively. Let $V = M + N$ be the total number of velocity unknowns and Q the number of pressure unknowns. The null space matrices will have dimensions $\mathsf{R} \in \mathbb{R}^{(V-Q)\times V}$ and $\mathsf{C} \in \mathbb{R}^{V\times(V-Q)}$. The coupled system (3) is re-organized as follows

$$\begin{bmatrix} \mathsf{RA} & \mathsf{RG} \\ \mathsf{D} & 0 \end{bmatrix} \begin{bmatrix} \mathsf{C}S^{n+1} \\ P^{n+\theta} \end{bmatrix} = \begin{bmatrix} \mathsf{R}r^n \\ 0 \end{bmatrix}, \tag{22}$$

which is obtained by pre-multiplying the first row by R and defining the variable $S^{n+1} \in \mathbb{R}^{(V-Q)}$ such that $U^{n+1} = \mathsf{C}S^{n+1}$. Since RG is zero, we arrive at the following system of equations, which can be solved sequentially:

$$\mathsf{RAC}S^{n+1} = \mathsf{R}r^n; \qquad U^{n+1} = \mathsf{C}S^{n+1}; \qquad \mathsf{DG}P^{n+\theta} = \mathsf{D}r^n - \mathsf{DA}U^{n+1}. \tag{23}$$

Table 1: Methods used in the numerical tests in this work, and their expected temporal order of accuracy (using $\theta = 0.5$).

Ref.	Method	Order
M1	Monolithic	$\mathcal{O}(\delta t^2)$
M2	First order projection	$\mathcal{O}(\delta t)$
M3	Incremental projection	$\mathcal{O}(\delta t^2)$
M4	Perot's second order	$\mathcal{O}(\delta t^2)$
M5	Pseudo-exact factorization	$\mathcal{O}(\delta t)$
M6	Exact factorization	$\mathcal{O}(\delta t^2)$

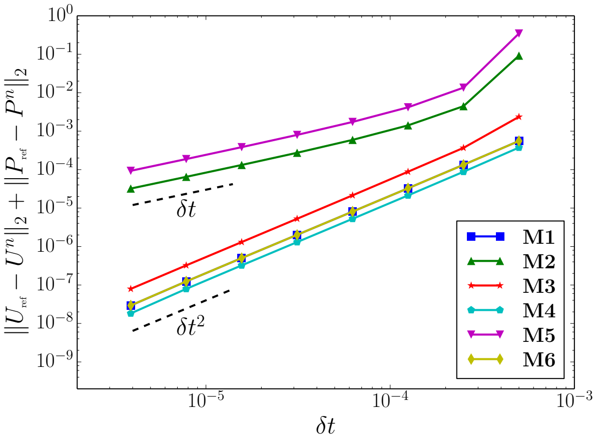

Figure 1: Time convergence for methods **M1-M6**.

The first equation is a Poisson-like equation, and the last equation is obtained by applying the divergence operator to the first row of (3). Note that this method requires the solution of one linear system for velocity, plus an optional Poisson-like problem to evaluate pressure at new time step $t_{n+\theta}$, if needed. A significant part of the cost lies, quite obviously, in the construction of the null space matrices R and C, which however have the good property of not changing with time.

This method is a truly exact factorization scheme: the original system is not modified by any approximation and the splitting error is in fact zero, as illustrated numerically later in this work. The procedure of finding the null spaces proposed in [9] is based on the discretization of the curl operator. Both null space matrices can be viewed as discrete curl operators: the curl of the gradient is zero ($\mathsf{RG} = 0$) and the divergence of the curl is zero ($\mathsf{DC} = 0$). Therefore S^{n+1} can be viewed as a discrete stream function associated with the velocity field U^{n+1} (the curl of the velocity).

3 NUMERICAL RESULTS

In this section, we report numerical results confirming the expected temporal convergence of the methods introduced before, and investigating their behavior to compute steady state solutions for low Reynolds flows.

3.1 Temporal convergence of algebraic splitting methods

In the previous sections several methods were described, that are summarized in Table 1. Formal analyses of the approximation error involved, with respect to the monolithic method (M1) were provided, and the expected orders of accuracy in time are also displayed in Table 1. To establish the actual performance of each method, let us test them in the well-known benchmark of freely decaying square vortices. The domain considered is the square $\Omega = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \times \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ such that the exact solution of the Navier-Stokes equations

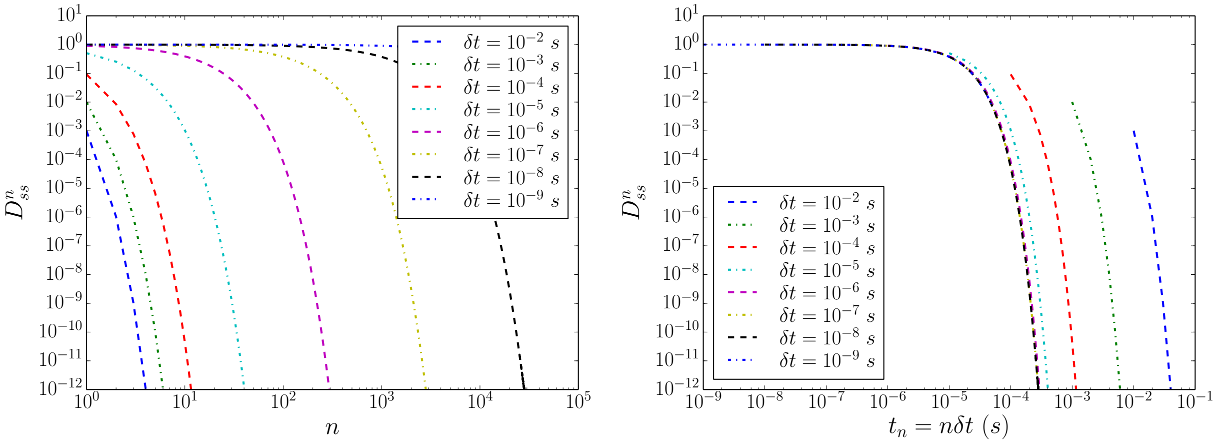

Figure 2: Evolution of D_{ss}^n for the monolithic method **M1**, for several δt; Left: As a function of the iteration number n; Right: As a function of time $t_n = n\,\delta t$ (s).

is given by

$$\mathbf{u}(x_1, x_2) = e^{-2\nu t}\left(-\cos(x_1)\sin(x_2),\ \sin(x_1)\cos(x_2)\right) \tag{24}$$

$$p(x_1, x_2) = -0.25\ e^{-4\nu t}\left(\cos(2x_1) + \cos(2x_2)\right) \tag{25}$$

For this problem, we choose $\nu = 100$, with unitary velocity and length scales, resulting in a Reynolds number of $Re = 10^{-2}$. The flow is initialized with the analytical solution computed at time $t = 0$. Since we are mainly concerned with temporal convergence, we compare the results with a reference solution, computed with the monolithic method (with the same initial condition) with $\delta t = 10^{-6}$. The problem was computed with all methods listed in Table 1, for time steps $\delta t_k = 2^{-k} \cdot 10^{-3}$, for $k = 1, 2, \ldots, 8$. To ensure second order time accuracy, a Crank-Nicolson scheme was employed ($\theta = 0.5$). In Fig. 1 the errors $\|U_{\text{ref}}(t) - U^n\|_2 + \|P_{\text{ref}}(t) - P^n\|_2$ at $t = 10^{-2}$ (with $n\delta t = t$) are plotted as functions of the non-dimensional δt. It becomes evident that the formal orders of convergence are therefore confirmed.

3.2 Numerical analysis of algebraic schemes for low-Re flows

In this paper, we are specially concerned in how most common algebraic splitting methods behave when applied to solve microflows, characterized mainly by a high viscosity, or a low Reynolds number. To illustrate the difficulties faced by those methods to compute those types of flows, we solve a simple closed loop channel, forcing the flow with a source term $\mathbf{f}$. The domain is $\Omega = [0, \ell] \times [0, \omega]$, with periodic boundary conditions in x_1 direction, and no slip boundaries at $x_2 = 0$ and $x_2 = \omega$. The source term is given by

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f(x_1) \\ 0 \end{bmatrix} \quad \text{with} \quad f(x_1) = \begin{cases} \frac{16W\nu}{\omega^2} & \text{if } x_1 < \frac{\ell}{2} \\ 0 & \text{if } x_1 \geq \frac{\ell}{2} \end{cases} \tag{26}$$

δt (s)	$\|U^* - U^\infty\|_2/\|U^\infty\|_2 + \|P^* - P^\infty\|_2/\|P^\infty\|_2$		
	M2	**M4**	**M5**
10^{-2}	1.4790	∞	7.4988
10^{-3}	1.4787	∞	7.4920
10^{-4}	1.4779	∞	7.4269
10^{-5}	1.4762	∞	6.9826
10^{-6}	1.3622	∞	$6.3877 \cdot 10^{-1}$
10^{-7}	$4.8151 \cdot 10^{-1}$	∞	$3.0071 \cdot 10^{-1}$
10^{-8}	$9.0683 \cdot 10^{-2}$	$5.4412 \cdot 10^{-3}$	$5.7038 \cdot 10^{-2}$
10^{-9}	$1.0058 \cdot 10^{-2}$	$3.4388 \cdot 10^{-5}$	$6.3322 \cdot 10^{-3}$

Table 2: Steady state errors for several δt, comparing non-incremental methods **M2**, **M4** and **M5**.

with W being a suitable velocity scale and ν the kinematic viscosity, in such a way that $Re = W\omega/\nu$. Therefore, the exact solution for this flow is given by

$$u_1^\infty = 4W\,\frac{x_2}{\omega}\left(1 - \frac{x_2}{\omega}\right), \quad u_2^\infty = 0, \quad \partial_{x_1} p^\infty = \begin{cases} \frac{8W\nu}{\omega^2} & \text{se } x_1 < \frac{\ell}{2} \\ -\frac{8W\nu}{\omega^2} & \text{se } x_1 > \frac{\ell}{2} \end{cases}, \quad \partial_{x_2} p^\infty = 0. \quad (27)$$

The numerical tests were performed with chosen parameters $W = 10^{-3}\ m/s$, $\omega = 10^{-5}\ m$, $\ell/\omega = 3$ and $\nu = 10^{-6}\ m^2/s$, which corresponds to water flowing through a 10 μm wide channel, resulting in $Re = 10^{-2}$. Regardless of the choice of spatial resolution, the steady state velocity and pressure fields are the solution of the non-linear system

$$\mathsf{T}(U^\infty) - \nu \mathsf{L} U^\infty + \mathsf{G} P^\infty = b \quad \text{and} \quad \mathsf{D} U^\infty = 0. \quad (28)$$

To compute the steady state solution with minimum cost, since $Re \ll 1$, a common practice is to choose a sufficiently large time step size as to avoid the initial transient, obtaining the final solution with few time steps. This procedure clearly requires a unconditionally stable scheme, to avoid stability issues. To establish a parameter, we plot the dynamics of the steady state error (or distance to steady state) defined as

$$D_{ss}^n = \|U^n - U^\infty\|_2/\|U^\infty\|_2 + \|P^n - P^\infty\|_2/\|P^\infty\|_2\ , \quad (29)$$

as a function of iteration count and simulation time, for the monolithic method (M1), shown in Fig. 2. These results were obtained with a 60×20 grid, for several values of δt. From there, one can see that (U^∞, P^∞) is achieved with error below 10^{-6}, in about 7 time steps for $\delta t = 10^{-4}$ s, and in 2 time steps for $\delta t = 10^{-2}$ s. Since the physical time to reach steady state is about $T_{ss} \simeq \omega^2/\nu = 10^{-4}$ s, any δt chosen smaller than T_{ss} will produce a solution that has to overcome this transient, increasing the number of iterations, as expected (see Fig. 2). Remember however that one should solve the coupled system (3) for each time step, an effort that splitting schemes try to avoid.

A fact already known is that implicit non-incremental schemes suffer from the dependency of the time step size in the computation of steady state solutions (see for instance

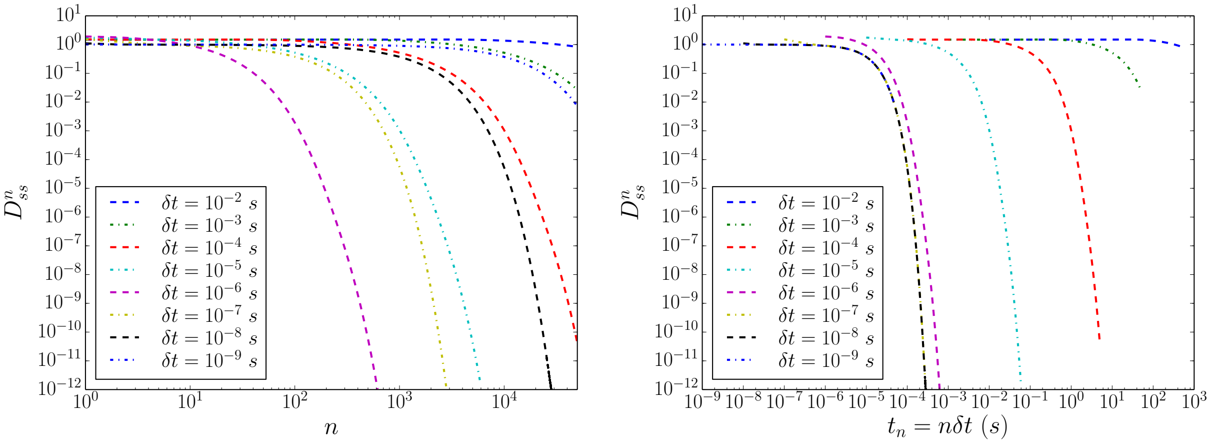

Figure 3: Evolution of D_{ss}^n for the incremental projection method **M3**, for several δt; Left: As a function of the iteration number n; Right: as a function of time $t_n = n\,\delta t$ (s).

[12]). This can be easily checked by eliminating $\tilde{U}$ from (13), resulting

$$\frac{U^{n+1} - U^n}{\delta t} + \mathsf{T}(U^n) - \nu \mathsf{L} U^{n+1} + \mathsf{G} P^{n+1} - \delta t \nu \ \mathsf{LG} P^{n+1} = b; \qquad \mathsf{D} U^{n+1} = 0, \tag{30}$$

whose stationary solution (U^*, P^*) such that $U^{n+1} = U^n = U^*$ and $P^{n+1} = P^n = P^*$, i.e.

$$\mathsf{T}(U^*) - \nu \mathsf{L} U^* + \mathsf{G} P^* - \delta t \nu \ \mathsf{LG} P^* = b; \qquad \mathsf{D} U^* = 0. \tag{31}$$

One can observe the term $-\delta t \nu \ \mathsf{LG} P^*$ which is dependent of δt, contaminating the numerical solution. The same holds for other non-incremental schemes, although the theoretical verification is omitted for the sake of brevity. Numerical computations show how this steady state error affect the final solution, producing approximations that differ from the expected steady state solution (U^∞, P^∞), with errors presented in Tab. 2. In this table, one can see that the solutions produced by the first order projection method **M2** are not acceptable for almost all time steps used for this problem. The same holds for the pseudo-exact factorization **M5**, that behaves just like **M2**. Perot's second order projection **M4** did not converged for $\delta t > 10^{-8}$, a result that is explained by the time step restriction imposed by the Taylor expansion performed in (12), which is estimated as $\delta t < \|\nu \mathsf{L}\|_2^{-1} \simeq 3.125 \cdot 10^{-8}$ s. Although the steady state errors for $\delta t \leq 10^{-8}$ s are relatively low, one must remember that the method should overcome the true transient, which is $T_{ss} \simeq 10^{-4}$ s, needing at least 10^4 time steps, an unacceptable number of iterations.

For incremental methods however, since pressure is treated differently, the computed steady state coincides with the true steady state (U^∞, P^∞), not presenting the steady error discussed previously. Indeed, the incremental projection method tested here is the most popular pressure segregation technique in the literature. However, for low Reynolds numbers, a phenomenon becomes apparent when computing the steady state solution by those methods: a "spurious transient" that hinders the convergence, increasing the number of time steps needed to attain (U^∞, P^∞) within reasonable accuracy.

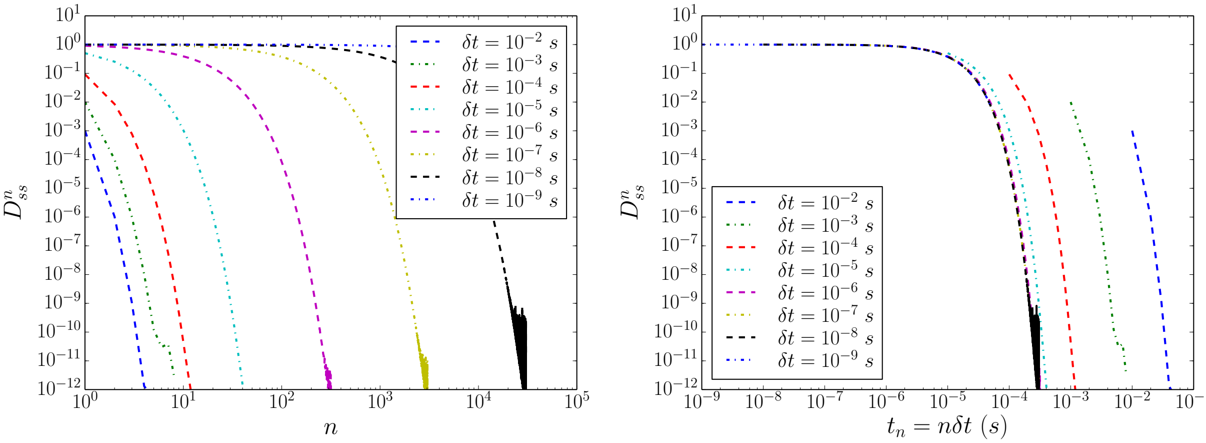

Figure 4: Evolution of D_{ss}^n for the exact factorization method **M6**, for several δt; Left: As a function of the iteration number n; Right: As a function of time $t_n = n\,\delta t$ (s).

In Fig. 3 we investigate the dynamics of the convergence to the steady state solution for the incremental projection method **M3**, by plotting D_{ss}^n (see Eq. (29)) as a function of both time step index and time. One can see that for time steps up to 10^{-6} s, the number of steps needed to attain steady state is about the same as the monolithic method, since the solution has to overcome the true transient. However, for large time steps, the number of time iterations increases, to the point that using $\delta t = 10^{-3}$ s leads to about 50k iterations to drop D_{ss}^n below 10^{-2}, taking a time greater than 10^2 s! Interestingly, there seems to be a optimal time step, which is around $\delta t = 10^{-6}$ s, leading to about two hundred iterations to drop D_{ss}^n below 10^{-6}. By taking computational cost in consideration, this number seems to be reasonable, however a wrong choice of time step can easily lead to thousands or tens of thousands iterations to obtain the same solution with same accuracy. This is specially worrying when a common practice of CFD practitioners is to choose a time step of the order of the CFL condition (that is this case would be $\delta t_{CFL} = 0.5 \cdot 10^{-3}$ s).

Finally, the exact factorization scheme presents no issues as seen before, behaving exactly as the monolithic method as can be observed in Fig 4. This result corroborates the exact splitting nature of this method, and its application to low-Reynolds flows.

4 CONCLUDING REMARKS

In this work we carefully tested most popular algebraic pressure segregation schemes, based on approximate or exact matrix factorizations. Through the numerical computation of a simple benchmark test, we showed that the most popular methods are inefficient to compute steady state solutions of low-Reynolds-number flows. In such flows, using large time steps (as compared to the stability limit of explicit schemes) is crucial to reach the steady state with reasonable computing effort. On one hand, the steady state error of non-incremental schemes is time-step dependent and happens to be unacceptably large for practical choices of δt. On the other hand, incremental schemes present a "spurious transient" that hinders the convergence and increases the number of time steps needed to attain steady state.

ACKNOWLEDGEMENTS

The authors thank the financial support from Brazilian agencies CNPq and FAPESP (grant 2013/07375-0).

REFERENCES

[1] A.J. Chorin. Numerical solution of the Navier-Stokes. *Math. Comput.*, 2:745–762, 1968.

[2] T. Temam. Sur l'approximation de la solution des equations de Navier-Stokes par la methode de pas fractionnaires (ii). *Arch. Ration. Mech. An.*, 33:377–385, 1969.

[3] D. L. Brown, R. Cortez, and M. L. Minion. Accurate projection methods for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 168:464–499, 2001.

[4] S. Armfield and R. Street. An analysis and comparison of the time accuracy of fractional-step methods for the Navier-Stokes equations on staggered grids. *Int. J. Numer. Meth. Fl.*, 38:255–282, 2002.

[5] J.K. Dukowicz and A. S. Dvinsky. Approximate factorization as a high order splitting for the implicit incompressible flow equations. *J. Comput. Phys.*, 102:336–347, 1992.

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

[7] A. Quarteroni, F. Saleri, and A. Veneziani. Factorization methods for the numerical approximation of Navier-Stokes equations. *Comput. Meth. Appl. M. Eng.*, 188:505–526, 2000.

[8] M. Lee, D. Oh, and Y.B Kim. Canonical fractional-step methods and consistent boundary conditions for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 168:73–100, 2001.

[9] W. Chang, F. Giraldo, and J.B. Perot. Analysis of an exact fractional-step method. *J. Comput. Phys.*, 180:183–199, 2002.

[10] K.K.Q. Zhang, W.J. Minkowycz, and F. Mashayek. Exact factorization technique for numerical simulations of incompressible Navier-Stokes flows. *Int. J. Heat Mass Tran.*, 49:535–545, 2006.

[11] M.Liu, Y.X. Ren and H. Zhang. A class of fully second order accurate projection methods for solving the incompressible NavierStokes eq. *J. Comput. Phys.*, 200:325–346, 2004.

[12] J.L. Guermond, P. Minev, and J. Shen. An overview of projection methods for incompressible flows. *Comput. Meth. Appl. M. Eng.*, 195:6011–6045, 2006.

[13] M.O. Henriksen and J. Holmen. Algebric splitting for incompressible Navier-Stokes equations. *J. Comput. Phys.*, 175:438–453, 2002.