Algebraic fractional step schemes with spectral methods for the incompressible Navier-Stokes equations *

Paola Gervasio[†], Fausto Saleri[§], Alessandro Veneziani[‖]

Abstract

The numerical investigation of a recent family of algebraic fractional-step methods for the solution of the incompressible time-dependent Navier-Stokes equations is presented. These methods are improved versions of the *Yosida* method proposed in [29] and [28] and one of them (the *Yosida4* method) is proposed in this paper for the first time. They rely on an approximate *LU* block factorization of the matrix obtained after the discretization in time and space of the Navier-Stokes system, yielding a splitting in the velocity and pressure computation. In this paper we analyze the numerical performances of these schemes when the space discretization is carried out with a spectral element method, with the aim of investigating the impact of the splitting on the global accuracy of the computation.

1 Introduction

One of the most known techniques for an efficient solution of the incompressible Navier-Stokes equations consists in using fractional-step methods of differential or algebraic type. In the former, the splitting is based either on physical considerations (see for example [12]), or on the Helmholtz decomposition principle. These methods are called *projection methods* and the most famous one is the Chorin-Temam scheme ([5], [36]). The accuracy of projection methods depends strongly on the boundary conditions chosen for the differential subproblems in which the original problem is split. In the last three decades many papers have been devoted to the study of high accurate differential fractional-step schemes, see for example [23, 22, 8, 38, 15, 14, 35, 3, 17, 18, 19] and the references therein.

On the other hand, algebraic fractional-step methods are based on an algebraic decomposition of the matrix arising from the full discretization (in both space and time) of the Navier-Stokes equations. Such decomposition (or *splitting*) could be performed either by a sum of simpler matrices (see for example the methods described by Yanenko [41] and Marchuk [24]) or a product of block-triangular matrices. In this perspective, Perot [26] revisited the Chorin-Temam method as an approximate (or *inexact*) block *LU* factorization of the matrix arising from the fully

---

[†]Department of Mathematics, University of Brescia, 25133 Brescia (Italy). `gervasio@ing.unibs.it`

[§]MOX, Department of Mathematics, Politecnico di Milano, 20133 Milano (Italy). `fausto.saleri@polimi.it`

[‖]MOX, Department of Mathematics, Politecnico di Milano, 20133 Milano (Italy). `alessandro.veneziani@polimi.it`

discretized equations. Following this approach the boundary conditions were incorporated in the discretized operator and no boundary conditions have to be selected.

The paper of Perot was followed by various works in which different formulations of the Chorin-Temam method were proposed and investigated (see [6, 29, 28, 40, 20]).

The interpretation of the Chorin-Temam scheme from an algebraic point of view gave rise to the investigation of new families of (algebraic) fractional-step schemes, with good accuracy and stability properties and which do not have a differential counterpart (see [29], [28], [20]). To set up new algebraic methods, the idea is to choose appropriately the inexact factorization of the matrix (say A) arising from the full discretized equations.

The *Yosida* method, proposed in [29, 28], is an algebraic fractional-step scheme, which differs from the algebraic version of the Chorin-Temam (ACT) method basically for fulfilling the discrete momentum equation, while the Chorin-Temam method guarantees the conservation of the mass. These two schemes differ from one another also in accuracy and stability properties and they have been investigated when coupled to finite element methods, also as preconditioners (see also [34], [40]). In [10] and [34] a more accurate scheme (here called *Yosida3*), based on the idea of adding a final correction step for the pressure computation to Yosida scheme is proposed.

In this paper we present a new (and computationally feasible) modification (called *Yosida4*) of the Yosida scheme, featuring better accuracy properties. Actually, when the Yosida3 method is combined with a third order approximation scheme in time, the discrete $L^2(H^1)$-error on the velocity behaves like Δt^3 and the discrete $L^2(L^2)$-error on the pressure like $\Delta t^{5/2}$ for vanishing Δt, while, if the Yosida4 method is combined with a fourth order approximation scheme in time, the resulting discrete $L^2(H^1)$-error on the velocity behaves like Δt^4 and the discrete $L^2(L^2)$-error on the pressure like $\Delta t^{7/2}$. An extensive theoretical analysis of the schemes, corroborating the numerical results presented here, will be carried out elsewhere (see [11]).

The second aim of the paper is to validate the accuracy of the Yosida schemes when conformal $\mathbb{P}_N - \mathbb{P}_{N-2}$ spectral elements are chosen for the space discretization ([1]).

Finally we will compare Yosida, Yosida3 and Yosida4 schemes from a computational point of view.

An outline of this paper is as follows. In Section 2 we state the terms of the problem: we briefly recall the notations of Navier-Stokes equations for incompressible flows, the Backward Differentiation formulas (BDF) used for approximation in time and the spectral element methods. In Section 3 we briefly recall the Yosida scheme and we show the numerical results obtained using it jointly with spectral methods. In Section 4, starting for the strategy for improving Yosida schemes introduce in [34] and based on a further step for the pressure computation, we derive the new *Yosida4* scheme. We prove the fourth order in time dependence of the splitting error. Then, we illustrate several numerical results obtained by applying these schemes to spectral elements approximation. Some conclusions are drawn in Section 5.

2 Definitions and settings

We consider the Navier-Stokes equations for Newtonian incompressible fluids in the primal velocity-pressure formulation. For any open bounded domain $\Omega \subset \mathbb{R}^2$ with a Lipschitz boundary, and a positive T, given an external field $\mathbf{f} \in [L^2(0,T;L^2(\Omega))]$, a boundary data $\mathbf{g} \in [L^2(0,T;H^{1/2}(\partial\Omega))]^2$ and an initial datum $\mathbf{u}_0 \in [H^1(\Omega)]^2$ such that $\nabla \cdot \mathbf{u}_0 = 0$, we look for

the velocity field $\mathbf{u} \in [L^2(0,T;H^1(\Omega))]^2$ and the pressure field $p \in L^2(0,T;L_0^2(\Omega))$ solutions of

$$
\begin{cases}
\dfrac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \times (0,T) \\
\nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \times (0,T) \\
\mathbf{u} = \mathbf{g} & \text{on } \partial\Omega \times (0,T) \\
\mathbf{u} = \mathbf{u}_0 & \text{in } \Omega \times \{0\},
\end{cases}
\tag{1}
$$

where $\nu > 0$ is the kinematic viscosity.

Problem (1) admits a unique solution if suitable smallness assumptions on the data, with respect to the viscosity ν, are assumed (see [37]). It is worth to mention that all methods and the analysis developed hereafter can be applied to different kinds of boundary conditions as well.

We approximate in time the Navier-Stokes system (1) by a *Backward Differentiation Formula* of order p (*BDFp*) and we linearize the convective term by an extrapolation formula of the same order of the *BDF* used.

Given $\Delta t \in (0,T)$, we set $t^0 = 0$ and $t^n = t^0 + n\Delta t$ with $n = 1, ..., N_T$ and $N_T = \left[\frac{T}{\Delta t}\right]$. Given $\mathbf{u}^n$, for $n \geq n_0$ (depending on the *BDF* scheme used), we look for the solution $(\mathbf{u}^{n+1}, p^{n+1})$ of the system

$$
\begin{cases}
\dfrac{\alpha}{\Delta t}\mathbf{u}^{n+1} - \nu \Delta \mathbf{u}^{n+1} + (\mathbf{u}^* \cdot \nabla)\mathbf{u}^{n+1} + \nabla p^{n+1} = \tilde{\mathbf{f}}^{n+1} & \text{in } \Omega \\
\nabla \cdot \mathbf{u}^{n+1} = 0 & \text{in } \Omega \\
\mathbf{u}^{n+1} = \mathbf{g}^{n+1} & \text{on } \partial\Omega,
\end{cases}
\tag{2}
$$

where

$$
\mathbf{u}^* = \begin{cases}
\mathbf{u}^n & \text{when using BDF1} \\
2\mathbf{u}^n - \mathbf{u}^{n-1} & \text{when using BDF2} \\
3\mathbf{u}^n - 3\mathbf{u}^{n-1} + \mathbf{u}^{n-2} & \text{when using BDF3} \\
4\mathbf{u}^n - 6\mathbf{u}^{n-1} + 4\mathbf{u}^{n-2} - \mathbf{u}^{n-3} & \text{when using BDF4}
\end{cases}
\tag{3}
$$

and

$$
\begin{aligned}
& n_0 = 1, \quad \alpha = 1, \quad \tilde{\mathbf{f}}^{n+1} = \mathbf{f}^{n+1} + \frac{1}{\Delta t}\mathbf{u}^n && \text{for BDF1} \\
& n_0 = 2, \quad \alpha = \frac{3}{2}, \quad \tilde{\mathbf{f}}^{n+1} = \mathbf{f}^{n+1} + \frac{1}{\Delta t}\left(2\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1}\right) && \text{for BDF2} \\
& n_0 = 3, \quad \alpha = \frac{11}{6}, \quad \tilde{\mathbf{f}}^{n+1} = \mathbf{f}^{n+1} + \frac{1}{\Delta t}\left(3\mathbf{u}^n - \frac{3}{2}\mathbf{u}^{n-1} + \frac{1}{3}\mathbf{u}^{n-2}\right) && \text{for BDF3} \\
& n_0 = 4, \quad \alpha = \frac{25}{12}, \quad \tilde{\mathbf{f}}^{n+1} = \mathbf{f}^{n+1} + \frac{1}{\Delta t}\left(4\mathbf{u}^n - 3\mathbf{u}^{n-1} + \frac{4}{3}\mathbf{u}^{n-2} - \frac{1}{4}\mathbf{u}^{n-3}\right) && \text{for BDF4.}
\end{aligned}
\tag{4}
$$

Moreover we chose conforming spectral elements with numerical integration to approximate with respect to the space variables. In order to fulfill the inf-sup condition for the discretized problem we use the $\mathbb{P}_N - \mathbb{P}_{N-2}$ scheme with staggered grids ([1]), according to which local polynomials of degree N in each space variable are used to approximate the velocity field and local polynomials of degree $N-2$ in each space variable are used to approximate the pressure.

By this choice, the inf-sup condition is satisfied with a constant which depends on the parameter N ([2]).

We introduce a conformal, regular and quasi-uniform (see, e.g., [21]) partition $\mathcal{T}_h$ of Ω in N_e quadrilaterals T_k such that

$$\overline{\Omega} = \bigcup_{k=1}^{N_e} \overline{T}_k, \tag{5}$$

with

$$h = \max_{T_k \in \mathcal{T}_h} h_k, \qquad h_k = \mathrm{diam}(T_k), \quad k = 1, .., N_e. \tag{6}$$

Let $\mathbb{P}_N(T_k)$ be the set of algebraic polynomials, defined on T_k, of degree less than or equal to N in each direction, and set

$$\mathbb{P}_{\mathcal{H}}(\Omega) = \{v_{\mathcal{H}} \in \mathcal{C}^0(\overline{\Omega}) : \ v_{N,k} := v_{\mathcal{H}|T_k} \in \mathbb{P}_N(T_k), \ \forall T_k \in \mathcal{T}_h\}. \tag{7}$$

The subscript $\mathcal{H}$ represents the discretization level and it stands for the couple (h, N). The definition (7) states that the space $\mathbb{P}_{\mathcal{H}}(\Omega)$ is the space of global continuous functions on $\overline{\Omega}$, which are polynomials of degree N, with respect to each space variable, on every quadrilateral of the mesh. The functions $u_{\mathcal{H}}$, $v_{\mathcal{H}}, ...$ will denote generic elements of the space $\mathbb{P}_{\mathcal{H}}(\Omega)$. Given $u_{\mathcal{H}}$, $v_{\mathcal{H}} \in \mathbb{P}_{\mathcal{H}}(\Omega)$, we set

$$(u_{\mathcal{H}}, v_{\mathcal{H}})_{\mathcal{H}, \Omega} = \sum_{k=1}^{N_e} (u_{N,k}, v_{N,k})_{N, T_k}, \tag{8}$$

where $(\cdot, \cdot)_{N, T_k}$ denotes the discrete inner product in $L^2(T_k)$, based on the Gauss-Lobatto Legendre (GLL) quadrature formulas ([4]). In each element T_k of the partition we define a local GLL grid of $(N + 1)^2$ points and a local Gauss Legendre (GL) grid of $(N - 1)^2$ points. The last grid is staggered with respect to the former one and it is internal to T_k. Therefore, we denote by N_v (resp. N_p) the total number of GLL (resp. GL) grid points in Ω.

We define the finite dimensional spectral element spaces:

$$\mathbf{V}_{\mathcal{H}} = [\mathbb{P}_{\mathcal{H}}(\Omega)]^2, \quad \mathbf{V}_{\mathcal{H}}^0 = [\mathbb{P}_{\mathcal{H}}(\Omega) \cap H_0^1(\Omega)]^2$$
$$Q_{\mathcal{H}} = \{q_{\mathcal{H}} \in L^2(\Omega) : \ q_{N,k} := q_{\mathcal{H}|T_k} \in \mathbb{P}_{N-2}(T_k), \ \forall T_k \in \mathcal{T}_h\}, \tag{9}$$

so that the finite dimensional counterpart of (2) reads: for $n \geq n_0$, given $\mathbf{u}_{\mathcal{H}}^* \in \mathbf{V}_{\mathcal{H}}$, look for the solution $(\mathbf{u}_{\mathcal{H}}^{n+1}, p_{\mathcal{H}}^{n+1}) \in \mathbf{V}_{\mathcal{H}} \times Q_{\mathcal{H}}$, with: $\mathbf{u}_{\mathcal{H}}^{n+1}(\mathbf{x}) = \mathbf{g}^{n+1}(\mathbf{x})$ for any $\mathbf{x}$ of the GLL mesh which belongs to $\partial\Omega$, $p_{\mathcal{H}}^{n+1}$ known in one point of the GL mesh, such that

$$\begin{cases} \dfrac{\alpha}{\Delta t}(\mathbf{u}_{\mathcal{H}}^{n+1}, \mathbf{v}_{\mathcal{H}})_{\mathcal{H}, \Omega} + \nu(\nabla\mathbf{u}_{\mathcal{H}}^{n+1}, \nabla\mathbf{v}_{\mathcal{H}})_{\mathcal{H}, \Omega} + ((\mathbf{u}_{\mathcal{H}}^* \cdot \nabla)\mathbf{u}_{\mathcal{H}}^{n+1}, \mathbf{v}_{\mathcal{H}})_{\mathcal{H}, \Omega} \\ \qquad - (p_{\mathcal{H}}^{n+1}, \nabla \cdot \mathbf{v}_{\mathcal{H}})_{\mathcal{H}, \Omega} = (\tilde{\mathbf{f}}^{n+1}, \mathbf{v}_{\mathcal{H}})_{\mathcal{H}, \Omega} \qquad\qquad \forall\mathbf{v}_{\mathcal{H}} \in \mathbf{V}_{\mathcal{H}}^0 \\ (\nabla \cdot \mathbf{u}_{\mathcal{H}}^{n+1}, q_{\mathcal{H}})_{\mathcal{H}, \Omega} = 0 \qquad\qquad\qquad\qquad\qquad \forall q_{\mathcal{H}} \in Q_{\mathcal{H}}. \end{cases} \tag{10}$$

We denote by $\mathbf{U}^n \in \mathbb{R}^{2N_v}$ the array of the velocity grid function evaluated on the GLL mesh at time t^n, and by $\mathbf{P}^n \in \mathbb{R}^{N_p}$ the array of the pressure grid function evaluated on the GL mesh at time t^n.

Denoting by M the mass matrix, K the stiffness matrix, and B (resp. $N(\mathbf{U}^*)$) the matrix related to the discretization of $-\nabla\cdot$ (resp. of the convective term), we rewrite system (10) as

$$\begin{cases} \dfrac{\alpha}{\Delta t}M\mathbf{U}^{n+1} + \nu K\mathbf{U}^{n+1} + N(\mathbf{U}^*)\mathbf{U}^{n+1} + B^T\mathbf{P}^{n+1} = \widetilde{\mathbf{F}}_1^{n+1} \\ B\mathbf{U}^{n+1} = \mathbf{0}, \end{cases} \qquad (11)$$

where $\widetilde{\mathbf{F}}_1^{n+1}$ is the array generated by the term $(\widetilde{\mathbf{f}}^{n+1}, \mathbf{v}_{\mathcal{H}})_{\mathcal{H},\Omega}$.

We set $C = \dfrac{\alpha}{\Delta t}M + \nu K + N(\mathbf{U}^*)$. The right hand side is modified accordingly, taking into account the contributions that the boundary nodes give to internal nodes. This step generates a right hand side $[\mathbf{F}_1^{n+1}, \mathbf{F}_2^{n+1}]^t$ that is non zero also for the second equation of (11).

At each time level the system has the following matrix form:

$$\begin{aligned} C\mathbf{U}^{n+1} + B^T\mathbf{P}^{n+1} &= \mathbf{F}_1^{n+1} && (12) \\ B\mathbf{U}^{n+1} &= \mathbf{F}_2^{n+1} && (13) \end{aligned}$$

or

$$A\mathbf{W}^{n+1} = \mathbf{F}^{n+1}, \qquad (14)$$

with

$$A = \begin{bmatrix} C & B^T \\ B & 0 \end{bmatrix}, \qquad \mathbf{W}^{n+1} = \begin{bmatrix} \mathbf{U}^{n+1} \\ \mathbf{P}^{n+1} \end{bmatrix}, \qquad \mathbf{F}^{n+1} = \begin{bmatrix} \mathbf{F}_1^{n+1} \\ \mathbf{F}_2^{n+1} \end{bmatrix}.$$

From now on, for the sake of simplicity, we will drop the index $n+1$ from all the vectors.

To solve system (14) one could use a global approach such as a preconditioned Krylov method with either algebraic or differential preconditioners or again with Schwarz type domain decomposition preconditioners ([25]) or a pressure Schur complement approach ([9]).

Alternatively, system (14) can be solved by a block LU factorization with

$$L = \begin{bmatrix} C & 0 \\ B & -BC^{-1}B^T \end{bmatrix}, \qquad U = \begin{bmatrix} I & C^{-1}B^T \\ 0 & I \end{bmatrix}. \qquad (15)$$

The matrix

$$\Sigma := -BC^{-1}B^T \qquad (16)$$

is the so called *pressure Schur complement* matrix.

Solving system (14) through the block LU factorization (15) consists in finding the solution of the following subsystems:

$$\begin{aligned} L-\text{step}: \ \text{find } \widetilde{\mathbf{U}}, \ \widetilde{\mathbf{P}}: \quad & \begin{cases} C\widetilde{\mathbf{U}} = \mathbf{F}_1 \\ \Sigma\widetilde{\mathbf{P}} = \mathbf{F}_2 - B\widetilde{\mathbf{U}}, \end{cases} \\[2mm] U-\text{step}: \ \text{find } \mathbf{U}, \ \mathbf{P}: \quad & \begin{cases} \mathbf{P} = \widetilde{\mathbf{P}} \\ C(\widetilde{\mathbf{U}} - \mathbf{U}) = B^T\mathbf{P}. \end{cases} \end{aligned} \qquad (17)$$

This is sometimes called the *pressure matrix method* (see e.g. [30]). It is worthwhile noting that, when a semi-implicit treatment of the convective term is considered, matrix C is time-dependent and the construction of matrix Σ at each time step is quite expensive. An effective way for reducing computational costs is to suitably approximate Σ: this is the basic idea underlying to the Yosida scheme (and to the algebraic reformulation of the Chorin-Temam method proposed by Perot see [26], [29]) that we are going to introduce.

3   The Yosida scheme

The *Yosida* scheme it is characterized by replacing the factors L and U of A, given in (15) with an *inexact LU factorization* of the form $\hat{A} = \hat{L} U$, where

$$\hat{L} = \begin{bmatrix} C & 0 \\ B & -BHB^T \end{bmatrix} \qquad \text{and} \qquad H := \frac{\Delta t}{\alpha} M^{-1}. \tag{18}$$

At each time-step, system (14) is replaced by

$$\hat{A}\widehat{\mathbf{W}} = \widehat{\mathbf{F}} \quad \text{with} \quad \widehat{\mathbf{W}} = \begin{bmatrix} \widehat{\mathbf{U}} \\ \widehat{\mathbf{P}} \end{bmatrix}, \tag{19}$$

where $\widehat{\mathbf{F}}$ is the right hand side which takes into account the approximate solutions of previous steps. Matrix

$$S := -BHB^T \tag{20}$$

is an approximation of the Schur complement Σ and it can be obtained by a zero-order truncation of the Neumann expansion of C^{-1} in Σ (see [6, 39, 40]).

The computational convenience in solving (19) at a generic time step, relies to the fact that we have actually to solve the following (smaller) subsystems:

$$\begin{aligned} \hat{L} - \text{step}: \ \text{find } \widetilde{\mathbf{U}}, \ \widetilde{\mathbf{P}}: \quad & \begin{cases} C\widetilde{\mathbf{U}} = \widehat{\mathbf{F}}_1 \\ S\widetilde{\mathbf{P}} = \widehat{\mathbf{F}}_2 - B\widetilde{\mathbf{U}}, \end{cases} \\[2mm] U - \text{step}: \ \text{find } \widehat{\mathbf{U}}, \ \widehat{\mathbf{P}}: \quad & \begin{cases} \widehat{\mathbf{P}} = \widetilde{\mathbf{P}} \\ C(\widetilde{\mathbf{U}} - \widehat{\mathbf{U}}) = B^T \widehat{\mathbf{P}}. \end{cases} \end{aligned} \tag{21}$$

Remark 3.1 (Computational cost of Yosida scheme) At each time step we have to solve two linear systems whose matrix is C and we can make use of preconditioned Krylov methods (such as Bi-CGStab or GMRES) with incomplete LU preconditioner ([33]) or finite-element preconditioner ([31, 7]), or again Schwarz domain decomposition methods. Moreover we have to solve a linear system whose matrix is $S = -BHB^T$. If the *inf-sup* condition is satisfied, then the matrix B is a full-rank matrix and S is symmetric, negative definite. Then, for 2D implementations, we can resort to a Cholesky factorization obtained by a suitable QR splitting of $H^{1/2}B^T$ (see [40]), while for 3D implementations we can refer to either preconditioned conjugate gradient algorithm or multigrid scheme (see [39]). Observe that S can be factorized once at all at the beginning of the time loop since it is time independent.

The computational convenience with respect to the *pressure matrix method* is evident by the fact that the matrix S is considerably easier to solve than Σ, especially when M is a diagonal matrix (as it is in spectral methods with numerical integration). Moreover the two steps (21) have to be solved once at each time level, while other iterative schemes, such as the Uzawa one, require the iterative solution of systems in C and in the pressure mass matrix. On the other hand, while both pressure matrix and Uzawa method compute the solution of the *unsplit* discretized problem, Yosida method computes an approximate solution affected by the splitting error. When the time discretization is obtained with a first order scheme, it has been proved

in [28] that the first order accuracy is maintained by the split solution. For higher order time discretizations we need to resort to more accurate splitting schemes such as the ones that we are going to consider in the next Sections.

Finally, we quote that a comparison of the Yosida scheme with other splitting schemes (Chorin-Temam projection method or its algebraic reinterpretation), which introduce a splitting error as well, can be found in [29].

3.1   Numerical results

In this section we illustrate the performances of the Yosida scheme (21) coupled with a spectral element space discretization and we do a comparison with a classical unsplit method, namely a Bi-CGStab solver on system (14), preconditioned with an incomplete LU factorization. The comparison will be done from both accuracy and computational points of view. Since systems (14) and (17) are equivalent, the unsplit approach will be referred to also with the term *exact LU factorization* in contrast with the Yosida method that will be named *inexact LU factorization*.

Following the same notation of Section 2, we denote by $(\hat{\mathbf{u}}_{\mathcal{H}}^n, p_{\mathcal{H}}^n) \in \mathbf{V}_{\mathcal{H}} \times Q_{\mathcal{H}}$ the numerical solution at time t^n obtained by the Yosida scheme. We define the velocity and pressure errors in $L^2(H^1)-$ and $L^2(L^2)-$ discrete norms respectively:

$$
E_{\mathbf{u}} := \left(\Delta t \sum_{n=0}^{N_T} \|\mathbf{u}(t_n) - \mathbf{w}_{\mathcal{H}}^n\|_{H^1(\Omega)}^2 \right)^{1/2}, \qquad E_p := \left(\Delta t \sum_{n=0}^{N_T} \|p(t_n) - q_{\mathcal{H}}^n\|_{L^2(\Omega)}^2 \right)^{1/2}, \quad (22)
$$

where either $\mathbf{w}_{\mathcal{H}}^n = \hat{\mathbf{u}}_{\mathcal{H}}^n$, $q_{\mathcal{H}}^n = \hat{p}_{\mathcal{H}}^n$ when we use the Yosida method or $\mathbf{w}_{\mathcal{H}}^n = \mathbf{u}_{\mathcal{H}}^n$, $q_{\mathcal{H}}^n = p_{\mathcal{H}}^n$ when we use the unsplit approach.

Here and in the sequel we will consider two different test cases with a given exact solution. As a first test case, we consider the computational domain $\Omega = (-1, 1)^2$ and $t \in [0, 1]$, while the forcing term, the boundary conditions and the initial conditions are set in such a way that the exact solution is

$$
\begin{aligned}
\mathbf{u}(x, y, t) &= [\sin(x) \sin(y + t), \cos(x) \cos(y + t)]^T \\
p(x, y, t) &= \cos(x) \sin(y + t).
\end{aligned}
\tag{23}
$$

On the same computational domain we will consider a second test solution where the dependence with respect to the space and time variables is factorized, that is:

$$
\begin{aligned}
\mathbf{u}(x, y, t) &= [e^{(x+y)}, -e^{(x+y)}]^T \sin(2\pi t) \\
p(x, y, t) &= -(x^2 + y^2) \sin(2\pi t).
\end{aligned}
\tag{24}
$$

For BDF schemes with order greater than 1 we need further initial data that in our cases will be provided by the exact solution. For general cases, initial data could be provided by suitable explicit schemes (e.g. Runge Kutta) of the same order of the BDF used.

Accuracy. In Figures 1 and 2 we report the errors (22) for both velocity and pressure for different values of the time-step Δt. The curves marked with *BDF1*, *BDF2*, and *BDF3* refer to the solution of system (14) obtained with first, second and third order *BDF* scheme (respectively), by solving the linear system by a global preconditioned Bi-CGStab.

The curves marked with *BDF1+Yosida*, *BDF2+Yosida* and *BDF3+Yosida* refer to the solution
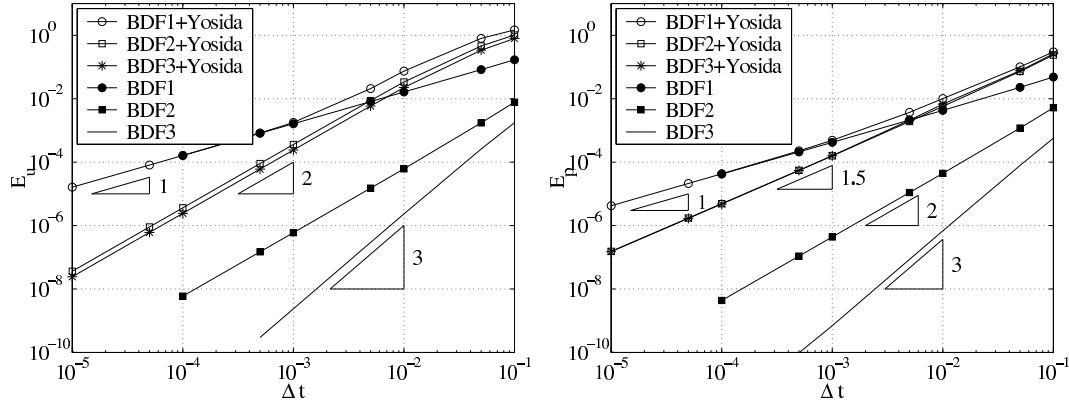
Figure 1: The errors (22) for the exact solution (23). $\nu = 10^{-3}$, $N = 16$ and $N_e = 1$.

of system (14) obtained with first, second and third order (respectively) *BDF* scheme and the Yosida method (21).

First of all we compare the convergence lines of *BDF1* and *BDF1+Yosida* and we note that for $\Delta t \leq 10^{-3}$ the two errors coincide. This means that for small time-steps (i.e. $\Delta t \leq 10^{-3}$) the Yosida method introduces an error (the so called *splitting error*) that is $o(\Delta t)$ for both velocity and pressure. The splitting error affects the computations with a BDF2 time discretization on the pressure: the comparison of BDF2 and BDF2+Yosida curves shows that the velocity error of the Yosida scheme is higher than in the unsplit solution, even if the accuracy is still of second order. The pressure is more affected by the splitting error and, in particular, a BDF2 time discretization coupled to the Yosida scheme exhibits an order of accuracy equal to 3/2. This can be theoretically justified (see Remark (4.1)). Finally, we note that we don't gain any advantage from using the Yosida scheme joined with *BDF3*.

In conclusion we can write the global velocity error in time as the sum of two errors, the first one due to the BDFp scheme and the second one due to the Yosida scheme:

$$err(\Delta t) = c_{BDFp} \, \Delta t^p + c_{Yos}\Delta t^2, \tag{25}$$

where c_{BDFp} and c_{Yos} are two positive constants independent of Δt. When $p = 1$ the Yosida scheme introduces a splitting error that is smaller than the BDF1 error. When $p = 2$ the Yosida scheme introduces a splitting error of the same order of BDF2, so that BDF2+Yosida is second order accurate, even if the error of BDF2+Yosida is considerably greater than the error of the pure BDF2.

If we want to use an algebraic fractional step method with a BDF scheme of order $p \geq 2$ and, at the same time, we want to preserve the accuracy obtained with the pure BDFp scheme, we have to introduce more accurate inexact *LU* factorizations which produce smaller splitting errors. Two different more accurate inexact *LU* factorizations will be described in the next Section.

Remark 3.2 In the context of differential splitting schemes such as the Chorin-Temam projection method (see e.g. [18], [27], [32]), an important issue is the prescription of pressure boundary conditions. Actually, in solving a differential problem for the pressure, unphysical boundary conditions are prescribed, which can induce boundary layers on the behavior of the pressure error.
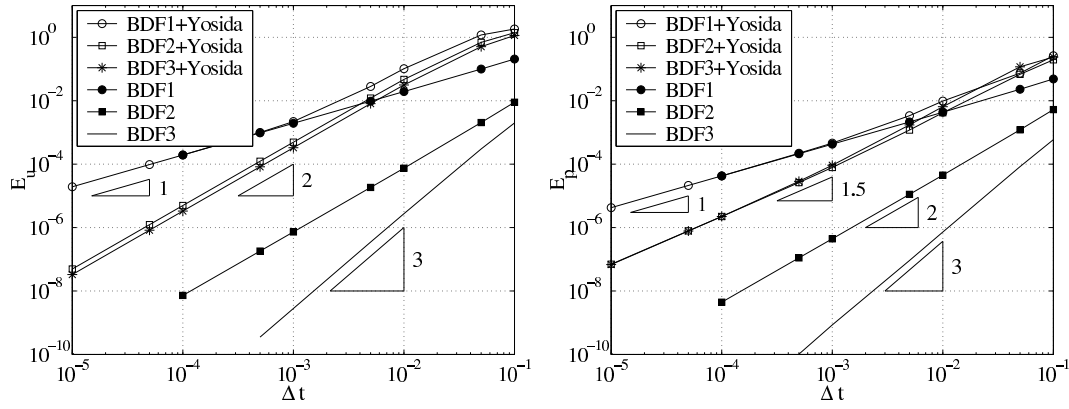
Figure 2: The errors (22) for the exact solution (23). $\nu = 10^{-5}$, $N = 16$ and $N_e = 1$.

In the context of algebraic fractional-step schemes no auxiliary boundary conditions for the pressure are needed since the splitting is performed on the algebraic problem, which already include all the (physical) boundary conditions and this usually reflects in a better behavior of the pressure solution on the boundary (see e.g. [29]).

This circumstance is confirmed in Figure 3 where we show the pressure error at $T = 1$ for the solution (23) and we can see that no boundary layers occur either if the uniqueness of the pressure is forced by fixing the pressure in one point of the computational domain (on the left) or by setting a null-average condition on the pressure (on the right).

Another relevant point concerning the accuracy of the scheme refers to the mass conservation equation. As we have already pointed out, the consistency error of the Yosida splitting affects only the continuity equation. We investigate the time dependency of this error, that means that we consider the behavior of the divergence of the velocity in time. In Figure 4 we show the behavior of $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}^n\|_{L^\infty(\Omega)}$, versus time, for three different values of the time step (on the left for the test case featuring the exact solution (23) and in the middle for (24)). On the right of Figure 4 the norm $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}^n\|_{L^\infty(L^\infty(\Omega))}$ versus the time step Δt is plotted. We observe that the behavior of $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}^n\|_{L^\infty(\Omega)}$ depends on the considered test problem, and in particular on how the velocity field depends on the time.
The picture on the right highlights that the norm $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}\|_{L^\infty(L^\infty(\Omega))} \leq c\Delta t^2$, which confirms the theoretical results of [28] and similar results in the context of finite elements discretization.

Computational cost. In Tables 1-3 the CPU-time (in seconds), needed to execute 100 time steps, is shown for both unsplit approach (marked with *BDF1* and *BDF2*) and inexact *LU* factorization (marked with *BDF1+Yosida* and *BDF2+Yosida*). The squared computational domain is subdivided in $N_e = n_e \times n_e$ equal squared elements and N denotes the polynomial degree in each direction, on each element. With one spectral element and a high polynomial degree N, the Yosida scheme is about ten times faster than the unsplit solver, in the other cases it is even more convenient in terms of CPU times. All programs were run on an Intel Pentium 4 processor with a frequency of 2.8 GigaHertz under an IEEE754 standard mode operation.
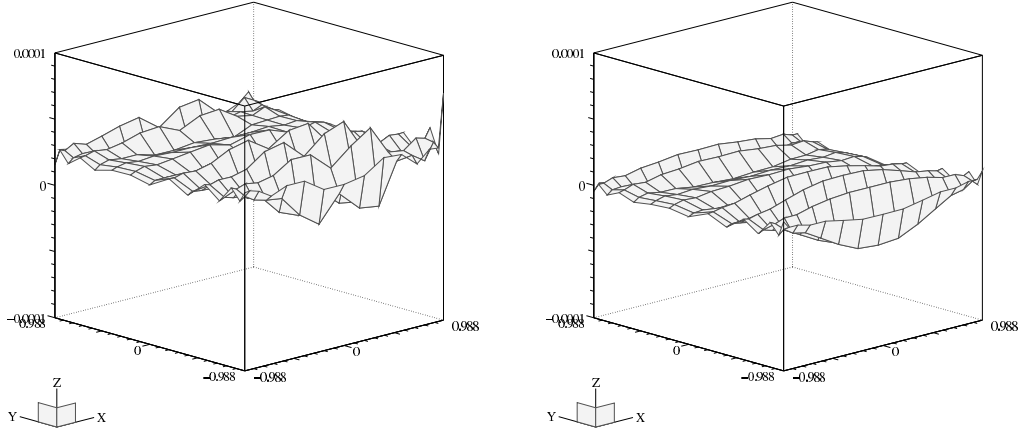
Figure 3: The error field $p_{\mathcal{H}} - p$ on the Gauss-Legendre grid at $T = 1$ for the test case featuring the exact solution (23), $\nu = 10^{-5}$. The numerical solution is computed with the scheme *BDF2+Yosida*, $\Delta t = 10^{-3}$, $N = 16$ and $N_e = 1$. On the left the numerical solution is obtained by fixing the pressure in one node of the computational domain, while on the right the numerical solution is obtained by fixing the average of the pressure.
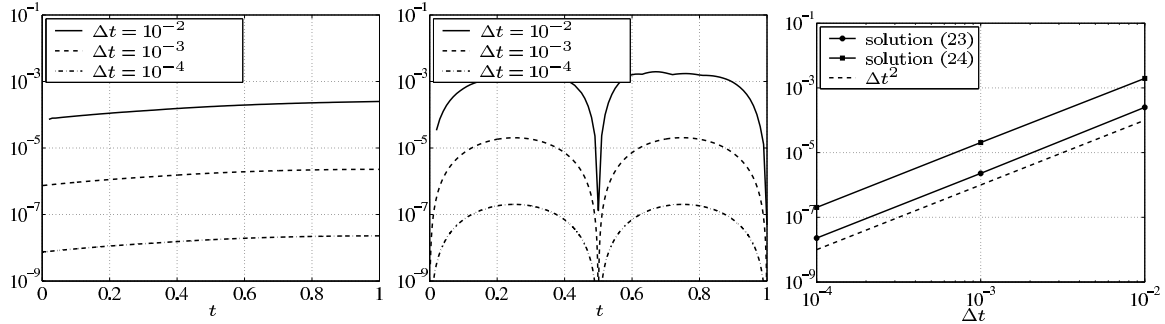

Figure 4: The behavior of $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}^n\|_{L^\infty(\Omega)}$ versus time (on the left for the test case featuring the exact solution (23), in the middle for (24)) and the behavior of $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}\|_{L^\infty(L^\infty(\Omega))}$ versus the time step (right). The discretization parameters are $N = 16$ and $N_e = 1$, $\nu = 10^{-5}$.

	$N = 16$	$N = 24$
BDF1	33.03	374.81
BDF1+Yosida	3.61	46.97
BDF2	29.34	323.03
BDF2+Yosida	3.91	46.26

Table 1: CPUtime (sec) for 100 time-steps, $\Delta t = 10^{-4}$, $\nu = 10^{-5}$, $N_e = 1$.

	$N_e = 16$	$N_e = 36$	$N_e = 64$	$N_e = 100$
BDF1	18.42	55.21	107.47	181.16
BDF1+Yosida	1.41	3.68	7.77	14.31
BDF2	29.79	61.60	135.10	241.10
BDF2+Yosida	1.31	3.58	7.53	13.91

Table 2: CPUtime (sec) for 100 time-steps, $\Delta t = 10^{-4}$, $\nu = 10^{-5}$, $N = 6$.

	$N = 4$	$N = 6$	$N = 8$	$N = 10$
BDF1	22.42	7.18	8.65	21.90
BDF1+Yosida	0.07	0.27	0.79	1.89
BDF2	23.20	7.56	10.25	22.85
BDF2+Yosida	0.08	0.27	0.74	1.80

Table 3: CPUtime (sec) for 100 time-steps, $\Delta t = 10^{-4}$, $\nu = 10^{-5}$, $N_e = 4$.

4   Improved Yosida methods: pressure correction strategy

We begin by noting that at each time-step, the final pressure $\widehat{\mathbf{P}}$ of the Yosida scheme coincides with the pressure computed at the $\hat{L}$−step, that is $\widehat{\mathbf{P}} = \widetilde{\mathbf{P}}$, and it induces a difference in the treatment of velocity and pressure deteriorating the accuracy of the scheme (see [38], [34]). In [34] a strategy for improving the accuracy by performing a correction step for the pressure is proposed. This corresponds to modify matrix U in (15) with a new matrix $\hat{U}$ featuring a matrix Q. More precisely we define:

$$\hat{U} = \left[\begin{array}{cc} I & C^{-1}B^T \\ 0 & Q \end{array} \right], \tag{26}$$

so that the matrix approximating A is

$$\hat{A} = \hat{L}\hat{U} = \left[\begin{array}{cc} C & B^T \\ B & -\Sigma + SQ \end{array} \right]. \tag{27}$$

Therefore, system (21) is replaced by

$$\begin{aligned} \hat{L} - \text{step}: \ \text{find } \widetilde{\mathbf{U}}, \ \widetilde{\mathbf{P}}: \ & \left\{ \begin{array}{l} C\widetilde{\mathbf{U}} = \widehat{\mathbf{F}}_1 \\ S\widetilde{\mathbf{P}} = \widehat{\mathbf{F}}_2 - B\widetilde{\mathbf{U}}, \end{array} \right. \\ \hat{U} - \text{step}: \ \text{find } \widehat{\mathbf{U}}, \ \widehat{\mathbf{P}}: \ & \left\{ \begin{array}{l} Q\widehat{\mathbf{P}} = \widetilde{\mathbf{P}} \\ C(\widetilde{\mathbf{U}} - \widehat{\mathbf{U}}) = B^T\widehat{\mathbf{P}}. \end{array} \right. \end{aligned} \tag{28}$$

A first possible choice for Q is given in [34], in this Section we will reconsider this choice of Q and the numerical results will show that the resulting scheme (which will be named *improved Yosida of order three*, briefly *Yosida3*) is third order accurate in time for the velocity and of order 5/2 for the pressure, provided it is associated to a third order *BDF* scheme. Moreover we will introduce a new choice of Q for which the resulting scheme (named *improved Yosida of*

order four, briefly *Yosida4*) is fourth order accurate in time for the velocity and of order 7/2 for the pressure, provided it is associated to a fourth order *BDF* scheme.

Observe that if we choose $Q = I$ we recover the Yosida scheme described in the previous section.

In general, matrix Q is chosen in such a way that

$$\|\Sigma - SQ\| \le c\Delta t^p, \qquad \text{with } p \ge 3, \tag{29}$$

and where $\|\cdot\|$ is the 2-matrix norm.

Setting $D = BHCHB^T$, if we choose

$$Q^{-1} = I - S^{-1}\hat{D} = -S^{-1}BHCHB^T = -S^{-1}D \tag{30}$$

we obtain the Yosida3 scheme introduced in [34].

Remark 4.1 It is worthwhile pointing out that the idea of introducing a pressure correction step has been introduced at first in the framework of differential splitting (projection) schemes (see [13], [38]). In this context, it has been recently analyzed by Guermond and Shen (see e.g. [19], [17] and [16] for an overview). We would like to stress the fact that the loss of 1/2 in the order of accuracy for the pressure, that we observe in our numerical experiments, is theoretically justified by the fact that the operator $\Sigma - SQ$, acting on the pressure and introduced in (27) even for $Q = I$, has a coercivity constant which depends on Δt in a linear way [11]. This fact has been verified also at the differential level, as it results by the rigorous analysis (limited to the order 2 for the velocity and 1/2 for the pressure) carried out in [19] (Theorem 4.1).

Yosida4 scheme.   Let us now introduce a new possible definition of Q.

We set

$$L = H^{-1} + C$$
$$W_1 = -BHLHB^T$$
$$W_2 = -B(HL)^2HB^T,$$

under the assumption that $\rho(S^{-1}(W_1 + W_2)) < 1$ ($\rho(\cdot)$ denotes the spectral radius of a matrix) we can exploit the Neumann expansion of C^{-1}, so that

$$\Sigma = -B(I + H)^{-1}HB^T = S - W_1 - W_2 + \Delta t^4 Z, \tag{31}$$

where Z denotes a generic matrix of the same order of Q independent of Δt. By their definition, if c denotes a positive constant, it holds that $\|W_1\| \lesssim c\Delta t^2$ and $\|W_2\| \lesssim c\Delta t^3$. A natural candidate for yielding with SQ a fourth order approximation of Σ is to set therefore

$$Q = I - S^{-1}(W_1 + W_2). \tag{32}$$

This is again an unfeasible solution from the computational viewpoint. However, we could try to find a suitable approximation of Q^{-1} starting from (32). Exploiting again the Neumann expansion we have:

$$Q^{-1} = I + S^{-1}(W_1 + W_2) + \left(S^{-1}(W_1 + W_2)\right)^2 + \left(S^{-1}(W_1 + W_2)\right)^3 + \dots$$

and, if we consider the time dependence of each factor on Δt, we can rewrite:

$$Q^{-1} = I + S^{-1}(W_1 + W_2) + S^{-1}W_1 S^{-1}W_1 + \Delta t^3 Z. \tag{33}$$

Now, by setting again $D = BHCHB^T$, we have

$$W_1 = D - S$$

and neglecting the terms that behave like Δt^3 we obtain

$$Q^{-1} = -S^{-1}D + (S^{-1}D)^2 + S^{-1}B(HC)^2 HB^T. \tag{34}$$

By using this matrix in (28) we obtain the *Yosida4* method.

Proposition 4.1 *If Q is defined as in (34) and $\rho(S^{-1}(W_1 + W_2)) < 1$, there exists a positive constant c independent of Δt such that*

$$\|\Sigma - SQ\| \le c\Delta t^4.$$

Proof. Let us compute SQ explicitly. By exploiting again the Neumann expansion we have:

$$SQ = S\left(I - S^{-1}(W_1 + W_2) + S^{-1}(W_1 + W_2)S^{-1}(W_1 + W_2) + \ldots\right) =$$
$$S - W_1 - W_2 - W_1 S^{-1}W_1 + W_1 S^{-1}W_1 + \Delta t^4 Z,$$

where again, Z denotes a generic matrix independent of Δt. By comparing this expression with the Neumann expansion (31) the thesis holds. $\circ$

Since in the Yosida pressure corrected schemes it holds $B\mathbf{U} = (\Sigma - SQ)\mathbf{P}$, the immediate corollary of the previous Proposition is that the discrete divergence of the velocity field computed by the Yosida4 scheme behaves like Δt^4 when Δt vanishes.

This will be verified by numerical experiments.
A detailed analysis of both accuracy and stability of the Yosida4 method is carried out in [11].

Remark 4.2 The computational effort required by either Yosida3 or Yosida4 methods increases mildly with respect to the cost of the Yosida method. As a matter of fact, at each time-step of Yosida3 (or Yosida4) we have to solve an additional linear system of type $Q\widehat{\mathbf{P}} = \tilde{\mathbf{P}}$. This means to solve the system

$$S\widehat{\mathbf{P}} = -D\tilde{\mathbf{P}} \tag{35}$$

in the Yosida3 scheme, and to do the following steps

$$
\begin{array}{ll}
\text{compute} & \mathbf{P}_D = -D\tilde{\mathbf{P}} \\
\text{solve} & S\mathbf{P}_S = \mathbf{P}_D \\
\text{solve} & S\widehat{\mathbf{P}} = -\mathbf{P}_D - D\mathbf{P}_S + B(HC)^2 HB^T\tilde{\mathbf{P}}
\end{array}
$$

in the Yosida4 scheme.
We remind that (see Remark 3.1), matrix S can be factorized before the time loop and then, in order to implement Yosida3, only two triangular systems have to be solved more, as well as to compute the matrix-vector product $D\tilde{\mathbf{P}}$. Otherwise, to implement Yosida4, four triangular systems have to be solved more, as well as to compute two matrix-vector products involving

matrix D.

From Tables 4-6 we observe that, when the number of elements $N_e \leq 36$, the cost of both Yosida3 and Yosida4 is comparable with that of Yosida, while for higher N_e, each iteration of Yosida3 is half as much more expensive than Yosida, and each iteration of Yosida4 is twice the cost of an iteration of Yosida.

In summary, the computation of both $\widehat{\mathbf{U}}$ and $\widehat{\mathbf{P}}$ through the improved Yosida schemes can be done with the following algorithm:

Algorithm 4.1 (A step of the improved Yosida algorithm) *For two given vectors* $\mathbf{F}_1$ *and* $\mathbf{F}_2$

$$
\begin{aligned}
i) \quad & solve \quad && C\widetilde{\mathbf{U}} = \mathbf{F}_1 \\
& solve \quad && S\widetilde{\mathbf{P}} = \mathbf{F}_2 - B\widetilde{\mathbf{U}} \\
& compute \quad && \mathbf{P}_D = -D\widetilde{\mathbf{P}} \\
& solve \quad && S\widehat{\mathbf{P}} = \mathbf{P}_D \\[6pt]
ii) \quad & compute \quad && \mathbf{P}_D = -\mathbf{P}_D - D\widehat{\mathbf{P}} + B(HC)^2 H B^T \widetilde{\mathbf{P}} \\
& solve \quad && S\widehat{\mathbf{P}} = \mathbf{P}_D \\[6pt]
iii) \quad & solve \quad && C(\widetilde{\mathbf{U}} - \widehat{\mathbf{U}}) = B^T \widehat{\mathbf{P}}.
\end{aligned}
$$

Observe that, when the implementation of the scheme is considered, the Yosida3 scheme is immediately obtained by dropping step ii). This could be of some interest in devising a general purpose code featuring different schemes.

4.1   Numerical results

In this Section we presents numerical results about the schemes Yosida3 and Yosida4 on the test cases given by the exact solutions (23) and (24).

Accuracy. In Figure 5 we report the errors (22) versus the time-step Δt. We have chosen the fourth order *BDF4* scheme for the temporal approximation and the fourth order extrapolation formula for the non linear term since our aim is to capture the *splitting error* introduced by the inexact *LU* factorizations. The Yosida scheme introduces an error on the velocity of order two with respect to the time-step (confirming the numerical results reported in Section 3), Yosida3 an error of order three and Yosida4 an error of order four. The errors on the pressure are scaled of half order with respect to the errors on the velocity.

In Figure 6, the errors (22) for the exact solution (24) are shown for the schemes *BDF3+Yosida3, BDF3+Yosida4, BDF3* and *BDF4* versus the time-step, with viscosity $\nu = 10^{-5}$.

The error on the velocity is third order accurate with respect to Δt for both *BDF3* and *BDF3+Yosida3*, while it is fourth order accurate for *BDF4*. The order of accuracy on the pressure is three for *BDF3*, four for *BDF4* and 5/2 for *BDF3+Yosida3*.

In Figure 7 the errors (22) are shown for the schemes *BDF2+Yosida, BDF3+Yosida3, BDF4+Yosida4* versus the time-step. A discretization in $N_e = 10 \times 10$ equal spectral elements is considered with $N = 6$.

When Δt is small enough, the schemes *BDF2+Yosida, BDF3+Yosida3* and *BDF4+Yosida4* are of second, third and fourth order accurate in time, respectively.
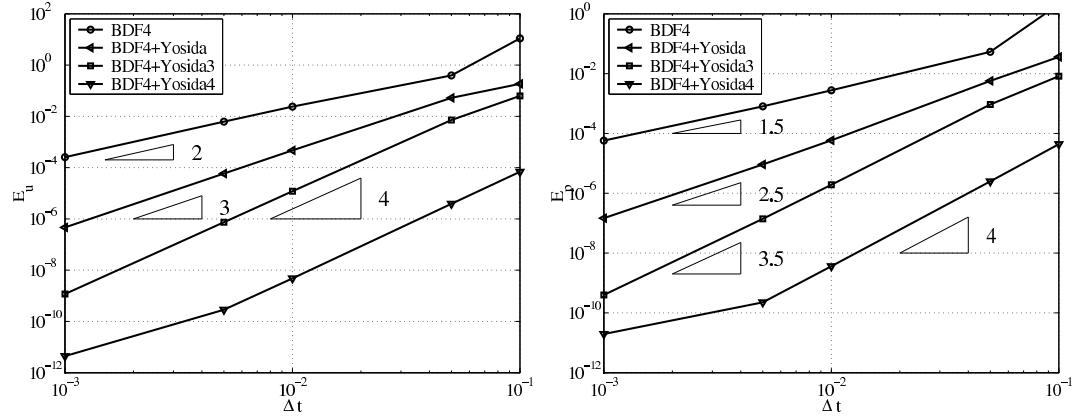
Figure 5: Errors (22) for the test case (23) with *BDF4* (unsplit approach), *BDF4+Yosida*, and the improved Yosida schemes *BDF4+Yosida3*, *BDF4+Yosida4*. The viscosity is $\nu = 10^{-5}$, the space discretization corresponds to $N_e = 1$ and $N = 16$.
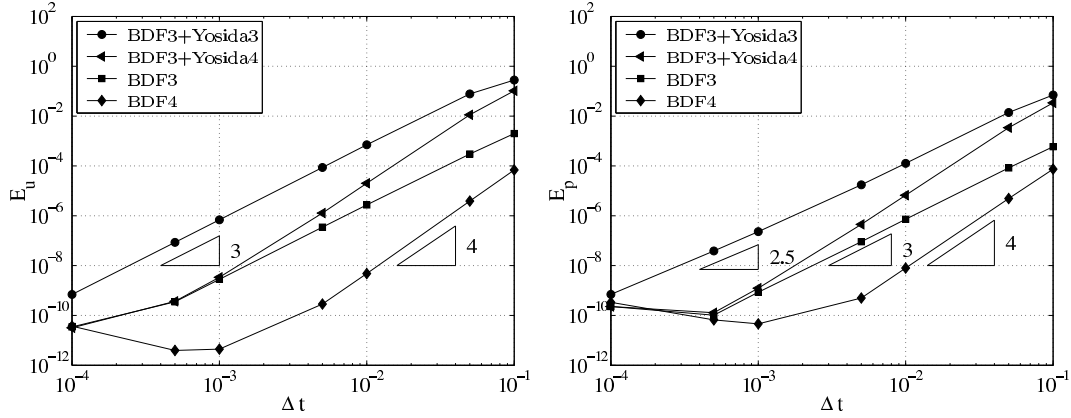

Figure 6: Errors (22) for the exact solution (23) with *BDF3+Yosida3*, *BDF3+Yosida4*, *BDF3* and *BDF4*. The viscosity is $\nu = 10^{-5}$. Spectral elements with $N = 16$ and $N_e = 1$ are considered.
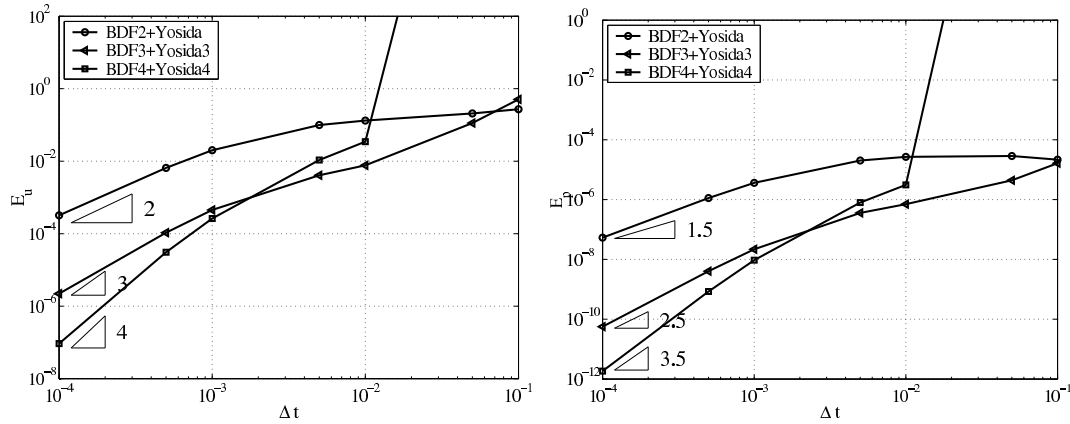
Figure 7: Errors (22) at $T = 1$ on the test case (24) with *BDF2+Yosida, BDF3+Yosida3* and *BDF4+Yosida4*. The viscosity is $\nu = 10^{-1}$. $N_e = 10 \times 10$ equal spectral elements are considered with $N = 6$.

In Figure 8 we report the error field $p_{\mathcal{H}} - p$ at $T = 1$ for both *BDF3+Yosida3* and *BDF4+Yosida4* when the viscosity is $\nu = 10^{-5}$. We observe that no boundary layers on the pressure are introduced by the improved Yosida schemes. In [32], Rannacher supposed that the boundary layer on the pressure error produced by the Chorin-Temam differential fractional-step scheme is of the same order of $\sqrt{\nu \Delta t}$. We have carried out a simulation with viscosity $\nu = 10^{-1}$ and, as we can see in Figure 9, also in this case the *BDF4+Yosida4* produces an error on the pressure without boundary layers. Analogous results have been obtained with both BDF2+Yosida and BDF3+Yosida3 schemes.

It is worthwhile to note that, when $\mathbb{P}_N - \mathbb{P}_{N-2}$ spectral elements are considered, the nodes of the Gauss Legendre grid, on which the discrete pressure is defined, are internal to the domain Ω and the pressure is discontinuous on the interfaces between two adjacent elements.

In Figure 10 (left) we show the behavior of $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}^n\|_{L^\infty(\Omega)}$, versus the time t for three different values of the time step for the Yosida3 scheme; the picture in the middle shows the behavior of the same norm for the Yosida4 scheme and, finally, the picture on the right shows the behavior of the norm $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}\|_{L^\infty(L^\infty(\Omega))}$ versus the time step Δt. The oscillations in the case of Yosida4 (middle) are possible due to instability reasons. The picture on the right highlights that the norm $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}\|_{L^\infty(L^\infty(\Omega))} \leq c\Delta t^3$ for Yosida3 and $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}\|_{L^\infty(L^\infty(\Omega))} \leq c\Delta t^4$ for Yosida4.

Stability. We present now some numerical simulations aiming at investigating the stability properties of the Yosida4 scheme. In Figure 7 we observe that *BDF4+Yosida4* requires a stability condition on the time-step more restrictive of those of both *BDF2+Yosida* and *BDF3+Yosida3*. Surely, *BDF4* has a stability region smaller than both *BDF2* and *BDF3*, but also the inexact LU factorization introduces a more restrictive stability condition on the time-step. Moreover, also the extrapolation formula can induce a stability restriction [34, 11]. In order to highlight the stability bound required by the inexact LU factorizations, we consider the time-dependent Stokes problem and the scheme *BDF2*, which is unconditionally stable. Then, we compare *BDF2+Yosida, BDF2+Yosida3* and *BDF2+Yosida4* and in Figure 11 we show the errors (22) for the exact solution (24) on the time interval $(0, T) = (0, 100)$. We observe that, when the viscosity is $\nu = 10^{-1}$ (first column), both Yosida3 and Yosida4 needs a stability bound, which
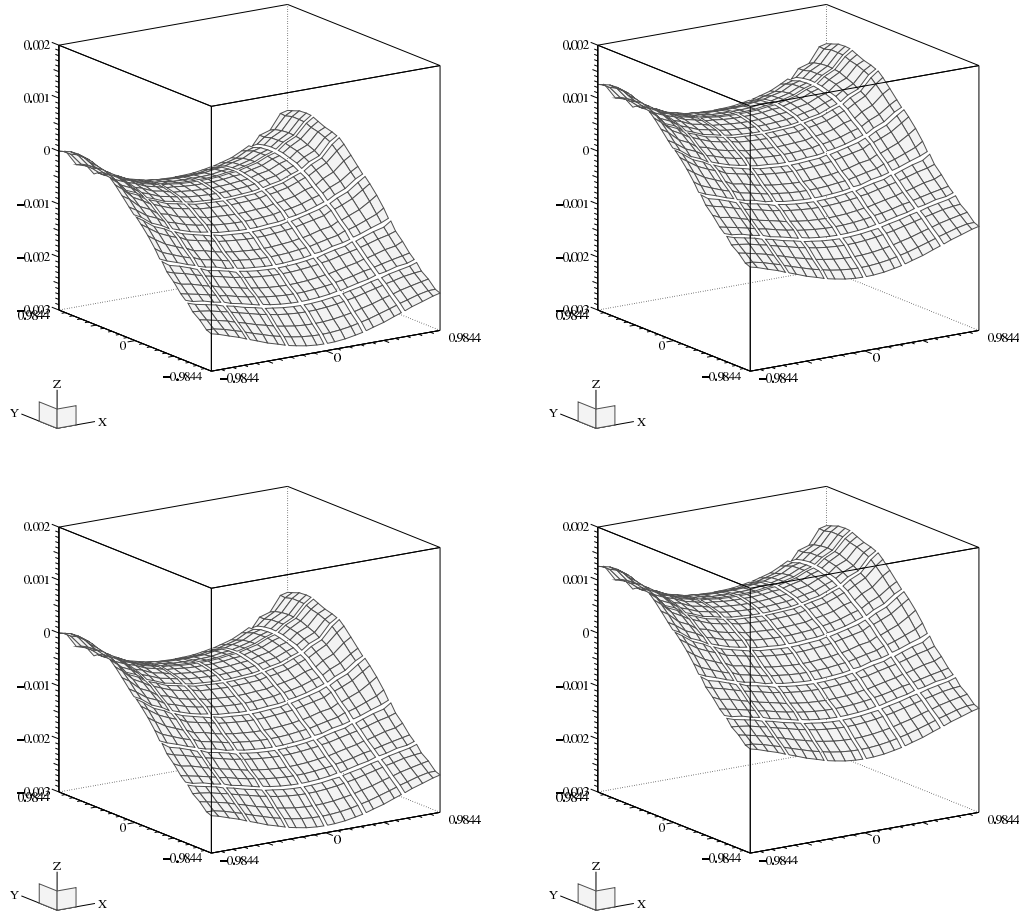
Figure 8: The error field $p_{\mathcal{H}} - p$ for the test case featuring the solution (23) at $T = 1$ for *BDF3+Yosida3* (top) and *BDF4+Yosida4* (bottom). The viscosity is $\nu = 10^{-5}$. $N_e = 6 \times 6$ equal spectral elements are considered with $N = 6$, $\Delta t = 10^{-3}$. On the left the numerical solution is obtained by fixing the pressure in one node of the mesh, on the right the numerical solution is obtained by fixing the average of the pressure.
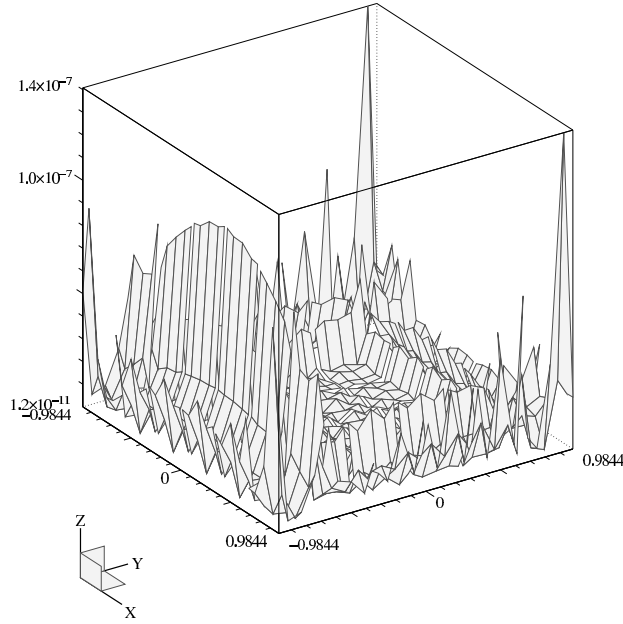
Figure 9: The error field $|p_{\mathcal{H}} - p|$ for the test case featuring the solution (23) at $T = 1$ for *BDF4+Yosida4*. The viscosity is $\nu = 10^{-1}$. $N_e = 6 \times 6$ equal spectral elements are considered with $N = 6$, $\Delta t = 10^{-3}$. The numerical solution is obtained by fixing the average of the pressure. The error peaks near to the corners of the domain (which is Lipschitz, but not more regular) are due to analytical reasons and they do not depend on the fractional step scheme.
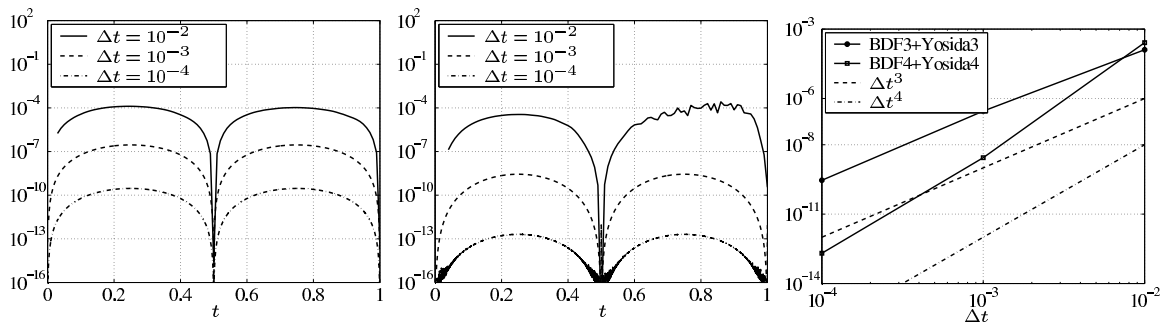

Figure 10: The behavior of $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}^n\|_{L^\infty(\Omega)}$ versus time for the exact solution (24) (on the left for Yosida3 and at middle for Yosida4) and the behavior of $\|\nabla \cdot \hat{\mathbf{u}}_{\mathcal{H}}\|_{L^\infty(L^\infty(\Omega))}$ versus the time step (right). The discretization parameters are $N = 16$ and $N_e = 1$, $\nu = 10^{-5}$.

	$N = 16$	$N = 24$
BDF2+Yosida	3.91	46.26
BDF3+Yosida3	3.89	46.65
BDF4+Yosida4	3.80	47.32

Table 4: CPUtime (sec) for 100 time-steps, $\Delta t = 10^{-4}$, $\nu = 10^{-5}$, $N_e = 1$.

	$N_e = 16$	$N_e = 36$	$N_e = 64$	$N_e = 100$
BDF2+Yosida	1.31	3.58	7.53	13.91
BDF3+Yosida3	1.52	4.12	9.32	18.19
BDF4+Yosida4	1.67	4.73	11.23	22.60

Table 5: CPUtime (sec) for 100 time-steps, $\Delta t = 10^{-4}$, $\nu = 10^{-5}$, $N = 6$.

depends mildly on the spectral elements size H. Otherwise, when the viscosity is smaller (second and third columns), both Yosida and its variants require the same stability condition ($\Delta t < 0.5$), independently of both the mesh size and the Yosida scheme.

Computational cost. In Table 4-6 the CPU-time (in seconds) needed to execute 100 time steps for *BDF2+Yosida*, *BDF3+Yosida3* and *BDF4+Yosida4* are shown. These results show that, when the number of elements is small, the computational cost of both Yosida3 and Yosida4 is comparable with that of Yosida, whereas, when the number of elements N_e grows up, the time needed by Yosida3 tends to be about half as much again the time needed by Yosida, while the time needed by Yosida4 tends to be about twice the time needed by Yosida.

5   Conclusions

We have considered the algebraic fractional-step Yosida scheme and two improved reformulations (Yosida3 and Yosida4) for the numerical solution of incompressible Navier-Stokes equations coupled to the spectral elements discretization. Yosida and Yosida3 have been previously introduced and experimented in the context of finite elements discretization. Yosida4 has been proposed in this paper for the first time.

Numerical results show that the high accuracy in time of the above schemes combines quite well with the high accuracy of spectral methods. The pressure corrected Yosida schemes exhibit significant accuracy improvements with respect to the original one. Moreover, as for all the algebraic fractional step schemes, the pressure solution is not affected by error boundary layers.

	$N = 4$	$N = 6$	$N = 8$	$N = 10$
BDF2+Yosida	0.08	0.27	0.74	1.80
BDF3+Yosida3	0.07	0.28	0.86	1.94
BDF4+Yosida4	0.07	0.31	0.88	2.11

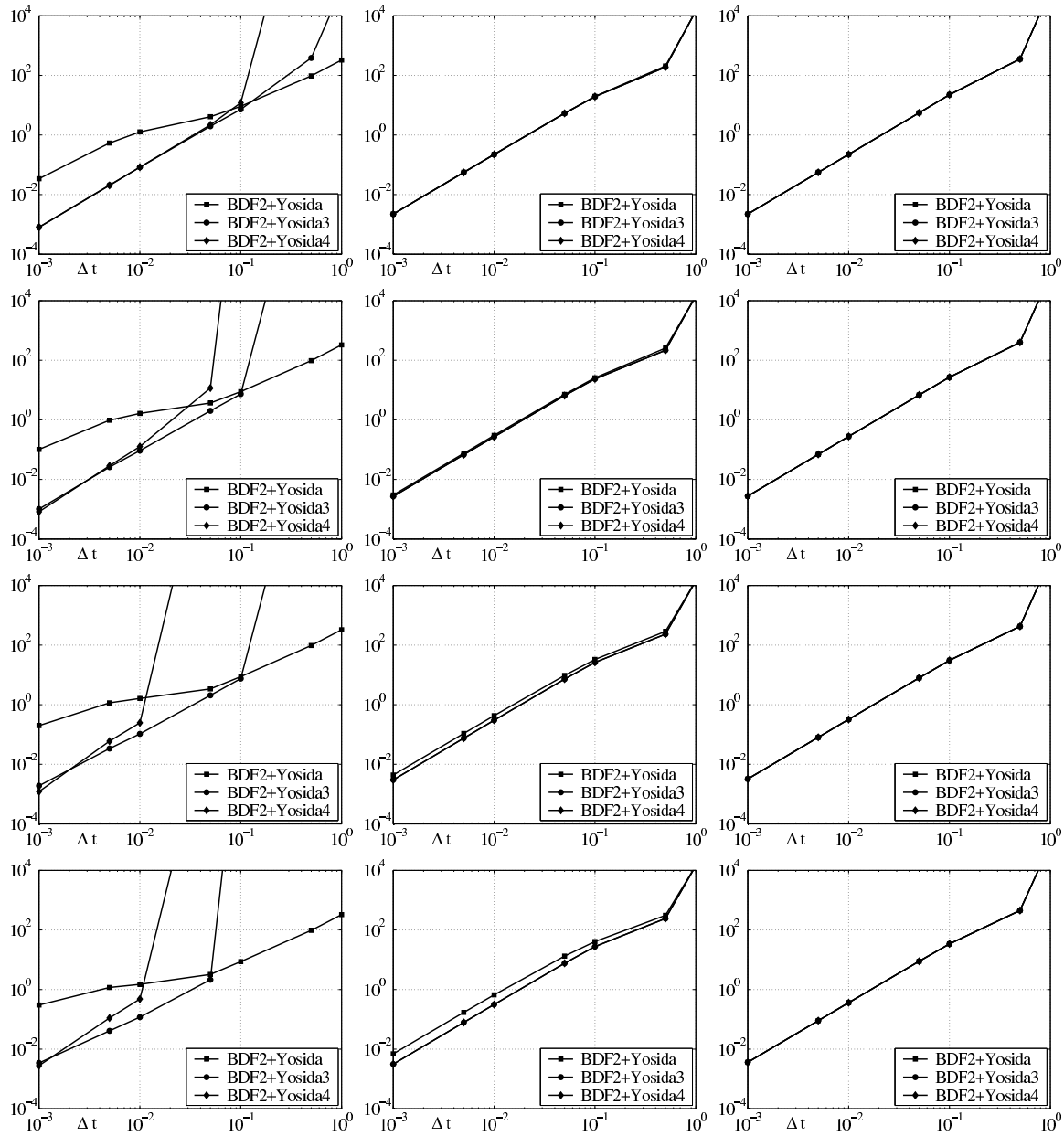Table 6: CPUtime (sec) for 100 time-steps, $\Delta t = 10^{-4}$, $\nu = 10^{-5}$, $N_e = 4$.

Figure 11: The error $E_{\mathbf{u}}$ defined in (22) on the exact solution (24) for the generalized Stokes problem and the schemes *BDF2+Yosida*, *BDF2+Yosida3* and *BDF2+Yosida4*. The viscosity is $\nu = 10^{-1}$ on the left, $\nu = 10^{-3}$ in the middle and $\nu = 10^{-5}$ on the right. $N_e = n_e \times n_e$ equal spectral elements are considered with $N = 6$; $n_e = 4$ at top, $n_e = 6$ in the second line, $n_e = 8$ in the third line and $n_e = 10$ in the last line.

As it has been already pointed out in [34] for the finite element discretization, stability could be a problem, since stability bounds of the time discretization scheme can be reduced by the splitting. An extensive theoretical analysis of the properties of these schemes is therefore in order.

Acknowledgments

The authors thank prof. A. Quarteroni for fruitful discussions during the preparation of this report.

References

[1] C. Bernardi and Y. Maday. *Approximations Spectrales de Problèmes aux Limites Elliptiques.* Springer Verlag, Paris, 1992.

[2] F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers. *R.A.I.R.O. Anal. Numér.*, 8 (R2):129–151, 1974.

[3] D.L. Brown, R. Cortez, and M.L. Minion. Accurate projection methods for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 168(2):464–499, 2001.

[4] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods in Fluid Dynamics.* Springer Verlag, Berlin, 1988.

[5] A.J. Chorin. Numerical solutions of the Navier-Stokes equations. *Math. Comput.*, 22:745–762, 1968.

[6] W. Couzy. *Spectral Element Discretization of the Unsteady Navier-Stokes Equations and its Iterative Solution on Parallel Computers.* PhD thesis, EPFL, Lausanne, 1995.

[7] M.O. Deville and E.H. Mund. Finite element preconditioning for pseudospectral solutions of elliptic problems. *SIAM J. Sci. Stat. Comput.*, 11:311–342, 1990.

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

[9] P. F. Fischer and E. M. Rønquist. Spectral element methods for large scale parallel Navier-Stokes calculations. *Comput. Methods Appl. Mech. Engrg.*, 116(1-4):69–76, 1994. ICOSA-HOM'92 (Montpellier, 1992).

[10] A. Gauthier, F. Saleri, and A. Veneziani. A fast preconditioner for the incompressible navier stokes equations. *Computing and Visualization in Science*, 6(2-3):105–112, 2004.

[11] P. Gervasio and F. Saleri. Analysis of high order algebraic fractional step schemes for spectral methods. In preparation, 2004.

[12] R. Glowinski. Splitting methods for the numerical solution of the incompressible Navier-Stokes equations. In *Vistas in applied mathematics*, Transl. Ser. Math. Engrg., pages 57–95. Optimization Software, New York, 1986.

[13] K. Goda. A multistep technique with implicit difference schemes for calculating two- or threedimensional cavity flows. *J. Comput. Phys*, 30:76–95, 1979.

[14] J.-L. Guermond and L. Quartapelle. On stability and convergence of projection methods based on pressure Poisson equation. *Internat. J. Numer. Methods Fluids*, 26(9):1039–1053, 1998.

[15] J.-L. Guermond and L. Quartapelle. On the approximation of the unsteady Navier-Stokes equations by finite element projection methods. *Numer. Math.*, 80(2):207–238, 1998.

[16] J.L. Guermond, P. Minev, and J. Shen. An overview of projection methods for incompressible flows. to appear in Comp. Meth. Appl. Mech. Eng., 2005.

[17] J.L. Guermond and J. Shen. A new class of truly consistent splitting schemes for incompressible flows. *J. Comput. Phys.*, 192(1):262–276, 2003.

[18] J.L. Guermond and J. Shen. Velocity-correction projection methods for incompressible flows. *SIAM J. Numer. Anal.*, 41(1):112–134, 2003.

[19] J.L. Guermond and J. Shen. On the error estimates for the rotational pressure-correction projection methods. *Math. Comp.*, 73(248):1719–1737, 2004.

[20] M.O. Henriksen and J. Holmen. Algebraic splitting for incompressible Navier-Stokes equations. *J. Comput. Phys.*, 175(2):438–453, 2002.

[21] C. Johnson. *Numerical Solution of Partial Differential Equations by the Finite Element Method.* Cambridge University Press, 1987.

[22] G.E. Karniadakis, M. Israeli, and S.A. Orszag. High order splitting methods for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 97:414–443, 1991.

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

[24] G.I. Marchuk. Splitting and Alternating Direction Methods. In P.G. Ciarlet and J.L. Lions, editors, *Handbook of Numerical Analysis*, volume 1, pages 197–462. Elsevier Science Publishers B.V. (North Holland), 1990.

[25] L.F. Pavarino and O.B. Widlund. Balancing Neumann-Neumann methods for incompressible Stokes equations. *Comm. Pure Appl. Math.*, 55(3):302–335, 2002.

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

[27] A. Prohl. *Projection and Quasi-Compressibility Methods for Solving the Incompressible Navier-Stokes Equations.* Springer-Verlag, 1997.

[28] A. Quarteroni, F. Saleri, and A. Veneziani. Analysis of the Yosida method for the incompressible Navier-Stokes equations. *J. Math. Pures Appl. (9)*, 78(5):473–503, 1999.

[29] A. Quarteroni, F. Saleri, and A. Veneziani. Factorization methods for the numerical approximation of Navier-Stokes equations. *Comput. Methods Appl. Mech. Engrg.*, 188(1-3):505–526, 2000.

[30] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations.* Springer Verlag, Heidelberg, 1994.

[31] A. Quarteroni and E. Zampieri. Finite element preconditioning for Legendre spectral collocation approximation to elliptic equations and systems. *SIAM J. Numer. Anal.*, 29:917–936, 1992.

[32] R. Rannacher. On Chorin's projection method for the incompressible Navier- Stokes equations. In R. Rautmann J.G. Heywood, K. Masuda and S.A. Slonnikov, editors, *The Navier-Stokes Equations II: Theory and Numerical Methods*, pages 167–183. Springer-Verlag, Berlin, 1992.

[33] Y. Saad. *Iterative methods for sparse linear systems.* Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.

[34] F. Saleri and A. Veneziani. Pressure-correction algebraic splitting methods for the incompressible navier-stokes equations. Technical report, MOX, Department of Mathematics, Politecnico di Milano, 2003. Accepted for publication on SIAM Journal on Numerical Analysis.

[35] J.C. Strikwerda and Y.S. Lee. The accuracy of the fractional step method. *SIAM J. Numer. Anal.*, 37(1):37–47, 1999.

[36] R. Temam. Sur l'approximation de la solution des équations de Navier-Stokes par la méthode de pas fractionaires (II). *Arch. Rat. Mech. Anal.*, 33:377–385, 1969.

[37] R. Temam. *Navier-Stokes equations and nonlinear functional analysis*, volume 41 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1983.

[38] L.J.P. Timmermans, P.D. Minev, and F.N. van de Vosse. An approximate projection scheme for incompressible flow using spectral elements. *Int. J. Numer. Methods Fluids*, 22(7):673–688, 1996.

[39] S. Turek. On discrete projection methods for the incompressible Navier-Stokes equations: an algorithmical approach. *Comput. Methods Appl. Mech. Engrg.*, 143(3-4):271–288, 1997.

[40] A. Veneziani. Block factorized preconditioners for high-order accurate in time approximation of the Navier-Stokes equations. *Numer. Methods Partial Differential Equations*, 19(4):487–510, 2003.

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