Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 217 (2006) 502–518
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A finite element penalty–projection method for incompressible flows

M. Jobelin [a], C. Lapuerta [a], J.-C. Latché [a,*], Ph. Angot [b], B. Piar [a]

[a] *Direction de la Prévention des Accidents Majeurs, Institut de Radioprotection et de Sûreté Nucléaire (IRSN), BP3-13115 St. Paul-lez-Durance Cedex, France*
[b] *Laboratoire d'Analyse, Topologie et Probabilités (LATP), 39 rue F. Joliot Curie, F-13453 Marseille Cedex 13, France*

Received 15 February 2005; received in revised form 22 July 2005; accepted 9 January 2006
Available online 28 February 2006

Abstract

The penalty–projection method for the solution of Navier–Stokes equations may be viewed as a projection scheme where an augmentation term is added in the first stage, namely the solution of the momentum balance equation, to constrain the divergence of the predicted velocity field. After a presentation of the scheme in the time semi-discrete formulation, then in fully discrete form for a finite element discretization, we assess its behaviour against a set of benchmark tests, including in particular prescribed velocity and open boundary conditions. The results demonstrate that the augmentation always produces beneficial effects. As soon as the augmentation parameter takes a significant value, the projection method splitting error is reduced, pressure boundary layers are suppressed and the loss of spatial convergence of the incremental projection scheme in case of open boundary conditions does not occur anymore. For high values of the augmentation parameter, the results of coupled solvers are recovered. Consequently, in contrast with standard penalty methods, there is no need for a dependence of the augmentation parameter with the time step, and this latter can be kept to reasonable values, to avoid to degrade too severely the conditioning of the linear operator associated to the velocity prediction step.

Keywords: Incompressible flows; Navier–Stokes; Projection method; Penalty method; Finite elements

1. Introduction

Since the pioneering work of Chorin and Temam [5,30] in the late sixties, pressure correction methods have gained a lot of popularity for the solution of transient incompressible flow problems. Indeed, schemes of this type have proved to be extremely efficient, essentially because, at each time step, they reduce the solution of a saddle-point type problem to a cascade of decoupled elliptic equations for the velocity and the pressure. This feature makes them particularly attractive for industrial applications, as for instance nuclear safety studies, which are the context of this work.

* Corresponding author.
E-mail address: jean-claude.latche@irsn.fr (J.-C. Latché).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2006.01.019

The principle of pressure correction methods is to perform the advance in time in two steps: in a first stage, the momentum balance is solved to obtain an intermediate (or predicted) velocity field; then this intermediate velocity is projected on a space of solenoidal vector fields. This process introduces a numerical error, often called the "splitting error", which magnitude may be expected at least heuristically to be linked to the magnitude of the divergence of the predicted velocity. Hence the basic approach followed here is to constrain the divergence of the intermediate velocity field by adding in the first step of the scheme an augmentation term, of the same form as in augmented Lagrangian methods [7]. This idea has been already suggested in the literature, first by Shen [25] and then, independently and with some additional variants, by Caltagirone and Breil [4]. In particular, a different projection step is proposed (called by the authors "vector projection step"), which is not used in the present work. In the paper by Shen, this novel family of numerical schemes was introduced as an improvement of both pressure correction and penalty methods (see [27] and references herein), and received the name of penalty–projection method. The numerical scheme presented here falls in this category, but is different from the original one, by changes in the algorithm itself, together with the fact that the augmentation parameter is totally independent from the time step Δt, whereas it was varying as Δt^{-2} in [25]. This point is discussed in the last remark of Section 2. To the best of our knowledge, no in-depth numerical study of any penalty–projection scheme is available in the literature; this is the general purpose of the present paper, together with a derivation of the method, dealing with general boundary conditions and including some implementation issues.

From our experience of the use of augmented Lagrangian methods in the finite element framework, it appears preferable to build the augmentation terms from the algebraic formulation of the discrete equations. This is due to the fact that the constraint *in the continuous sense* is usually not satisfied by the discrete solution: for the problem at hand, namely the solution of incompressible Navier–Stokes equations with a finite element method, the divergence constraint is only imposed in a weak sense. Consequently, an augmentation term built with the continuous expression of the constraint does not vanish for the solution of the discrete problem, and its presence may severely impact the results (see Section 3 for an explanation). However, to avoid an unnecessary restriction of the presentation to the finite element context, we choose to first carry out an introduction of the penalty–projection scheme in a space-continuous formulation. This is the goal of Section 2. The reader will keep in mind that, for practical implementation, the correct formulation of the method is given in the following section (Section 3), where the formulation of the scheme in the finite element context is detailed. In the last part of the paper, we compare the penalty–projection scheme to reference algorithms, namely the classical Euler semi-implicit method and frequently encountered projection schemes, for a set of benchmark tests involving a wide range of situations. In particular, besides prescribed velocity boundary conditions, we also consider the more challenging case (for pressure correction methods) of open boundary conditions.

2. The time semi-discrete algorithm

We are interested in the solution of the Navier–Stokes equations, governing an incompressible and isothermal flow of a Newtonian fluid, which read:

$$\varrho\left[\frac{\partial u}{\partial t}+(u\cdot\nabla)u\right]=-\nabla p+\nabla\cdot\tau(u)+\varrho g \quad \text{in } [0,T]\times\Omega \tag{1a}$$

$$\nabla\cdot u=0 \quad \text{in } [0,T]\times\Omega \tag{1b}$$

$$u=u_{\Gamma_D} \quad \text{on } [0,T]\times\Gamma_D \tag{1c}$$

$$-pn+\tau(u)\cdot n=f_N \quad \text{on } [0,T]\times\Gamma_N \tag{1d}$$

$$u=u_0 \quad \text{in } \{0\}\times\Omega \tag{1e}$$

where u stands for the fluid velocity of initial value u_0, p for the pressure, g for the external body forces, and ϱ for the fluid density, supposed to be a positive constant. The computational domain Ω is an open bounded connected subset of $\mathbb{R}^d$ with $d=2$ or $d=3$. We suppose that the boundary Γ of Ω is partitioned in two subsets Γ_D and Γ_N, of outward normal vector n. On Γ_D, the velocity is set to the value u_{Γ_D}; the force per surface unit exerted at each point of the boundary Γ_N is given and equal to f_N.

The tensor τ stands for the viscous part of the stress tensor, the divergence of which is given by one of the following expressions:

$$\nabla \cdot \tau(u) = \mu \Delta u \tag{2a}$$

or

$$\nabla \cdot \tau(u) = \nabla \cdot \mu(\nabla u + \nabla u^{\mathrm{T}}) \tag{2b}$$

where μ stands for the dynamic viscosity of the fluid. As Eq. (2a) is physically meaningful only if the parameter μ is a constant, this relation will be used only in this restrictive case in the rest of the paper.

Let $0 = t^0 < t^1 < \cdots < t^N = T$ be a partition of the time interval of computation $[0, T]$, which we suppose uniform for the sake of simplicity. We denote by Δt the time step, i.e., the constant difference between two successive times t^n and t^{n+1}, $0 \leqslant n \leqslant N-1$. A semi-implicit semi-discretization of the system of equations (1) with respect to the time variable is given by:

$$\varrho\left[\frac{Du^{n+1}}{\Delta t} + (u^{\star,n+1} \cdot \nabla)u^{n+1}\right] = -\nabla p^{n+1} + \nabla \cdot \tau(u^{n+1}) + \varrho g \quad \text{in } \Omega \tag{3a}$$

$$\nabla \cdot u^{n+1} = 0 \quad \text{in } \Omega \tag{3b}$$

$$u^{n+1} = u_{\Gamma_{\mathrm{D}}}^{n+1} \quad \text{on } \Gamma_{\mathrm{D}} \tag{3c}$$

$$-p^{n+1} n + \tau(u^{n+1}) \cdot n = f_{\mathrm{N}}^{n+1} \quad \text{on } \Gamma_{\mathrm{N}} \tag{3d}$$

where $u_{\Gamma_{\mathrm{D}}}^{n+1} = u_{\Gamma_{\mathrm{D}}}(t^{n+1})$, $f_{\mathrm{N}}{}^{n+1} = f_{\mathrm{N}}(t^{n+1})$, u^n and p^n stand for an approximation of, respectively, the velocity u and the pressure p at $t = t^n$, $u^{\star,n+1}$ is an extrapolation of the velocity at t^{n+1} and $Du^{n+1}/\Delta t$ provides an approximation of the time derivative of the velocity at t^{n+1} by a backward differentiation formula, which takes the general form:

$$Du^{n+1} \stackrel{\text{def}}{=} \beta_q u^{n+1} - \sum_{j=0}^{q-1} \beta_j u^{n-j}$$

In practice, the following choices are commonly used:

$$Du^{n+1} = u^{n+1} - u^n, \quad u^{\star,n+1} = u^n \tag{4}$$

$$Du^{n+1} = \frac{3}{2}u^{n+1} - 2u^n + \frac{1}{2}u^{n-1}, \quad u^{\star,n+1} = 2u^n - u^{n-1} \tag{5}$$

The first choice (4) leads to a first order scheme, the second one (5) to a formally second order numerical method.

Due to the saddle-point nature of the system (3), the solution is CPU-time consuming, which makes decoupled fractional step strategies attractive. The penalty–projection method belongs to this family of schemes. Classically, the first step consists in the solution of the momentum balance equation using a beginning-of-step value for the pressure. The specific feature of the proposed scheme lies in the addition at this stage of an augmentation term built from the divergence constraint. This yields the following elliptic problem for an intermediate velocity field at time t^{n+1}, $\tilde{u}^{n+1}$:

$$\varrho\left[\frac{\beta_q \tilde{u}^{n+1} - \sum_{j=0}^{q-1}\beta_j u^{n-j}}{\Delta t} + (u^{\star,n+1} \cdot \nabla)\tilde{u}^{n+1}\right] - r\nabla(\nabla \cdot \tilde{u}^{n+1}) = -\nabla p^n + \nabla \cdot \tau(\tilde{u}^{n+1}) + \varrho g \quad \text{in } \Omega \tag{6a}$$

$$\tilde{u}^{n+1} = u_{\Gamma_{\mathrm{D}}}^{n+1} \quad \text{on } \Gamma_{\mathrm{D}} \tag{6b}$$

$$-p^n n + \tau(\tilde{u}^{n+1}) \cdot n = f_{\mathrm{N}}^{n+1} \quad \text{on } \Gamma_{\mathrm{N}} \tag{6c}$$

where r is an augmentation parameter to be defined.

Let H be the following affine variety of divergence free vector fields:

$$H = \{v \in [L^2(\Omega)]^d, \quad \nabla \cdot v = 0, \quad v \cdot n = u_{\Gamma_{\mathrm{D}}}^{n+1} \cdot n \quad \text{on } \Gamma_{\mathrm{D}}\}$$

The second step of a projection method is usually chosen to realize the L^2 orthogonal projection of $\tilde{u}^{n+1}$ onto H, which takes the general form:

$$\beta_q \varrho \frac{u^{n+1} - \tilde{u}^{n+1}}{\Delta t} + \nabla \phi = 0 \quad \text{in } \Omega \tag{7a}$$

$$\nabla \cdot u^{n+1} = 0 \quad \text{in } \Omega \tag{7b}$$

For computational efficiency reason, this Darcy system is then reformulated by taking the divergence of the first relation to obtain a Poisson problem for ϕ, which must be supplemented by boundary conditions on Γ_D and Γ_N. The first one follows from the definition of H:

$$u^{n+1} \cdot n = \tilde{u}^{n+1} \cdot n = u^{n+1}_{\Gamma_D} \cdot n \quad \text{on } \Gamma_D$$

and, consequently:

$$\nabla \phi \cdot n = 0 \quad \text{on } \Gamma_D$$

The second one is derived from the L^2-orthogonality condition for the projection onto H, which reads:

$$\int_\Omega (u^{n+1} - \tilde{u}^{n+1}) \cdot (u^{n+1} - v) = 0 \quad \forall v \in H$$

By the relation (7a), integrating by parts and using the definition of H yields:

$$0 = \int_\Omega \nabla \phi \cdot (u^{n+1} - v) = \int_\Gamma \phi (u^{n+1} - v) \cdot n - \int_\Omega \phi \nabla \cdot (u^{n+1} - v) = \int_{\Gamma_N} \phi (u^{n+1} - v) \cdot n \quad \forall v \in H$$

which is satisfied if the following Dirichlet boundary condition for ϕ holds:

$$\phi = 0 \quad \text{on } \Gamma_N$$

Finally, ϕ is then the solution of the following elliptic problem:

$$\Delta \phi = \frac{\beta_q \varrho}{\Delta t} \nabla \cdot \tilde{u}^{n+1} \quad \text{in } \Omega \tag{8a}$$

$$\nabla \phi \cdot n = 0 \quad \text{on } \Gamma_D \tag{8b}$$

$$\phi = 0 \quad \text{on } \Gamma_N \tag{8c}$$

The end-of-step velocity is computed afterwards by the relation (7a):

$$u^{n+1} = \tilde{u}^{n+1} - \frac{\Delta t}{\beta_q \varrho} \nabla \phi \tag{9}$$

Finally, to obtain an expression for an approximation of the pressure at time t^{n+1}, we add the relations (6a) and (7a) to recover a discretization of the momentum balance equation at time t^{n+1}:

$$\varrho \left[\frac{\beta_q u^{n+1} - \sum_{j=0}^{q-1} \beta_j u^{n-j}}{\Delta t} + (u^{\star,n+1} \cdot \nabla) \tilde{u}^{n+1} \right] = -\nabla (p^n - r \nabla \cdot \tilde{u}^{n+1} + \phi) + \nabla \cdot \tau(\tilde{u}^{n+1}) + \varrho g \quad \text{in } \Omega \tag{10}$$

This suggests the following expression for p^{n+1}:

$$p^{n+1} = p^n - r \nabla \cdot \tilde{u}^{n+1} + \phi \tag{11}$$

To summarize, the algorithm corresponding to one time step of the penalty–projection method consists in solving the sequence of Eqs. (6) and (8) and computing the end-of step velocity and pressure by (9) and (11), respectively. For the reader's convenience, these four steps are gathered at the beginning of Section 4 (Eq. (21)), together with a synthetic presentation of some of the most popular projection schemes.

Remark (*Construction of a rotational penalty–projection method*). Following ideas of [31,17], it is possible to build what is termed in [17] as a "rotational pressure correction method". We suppose temporarily that we are in the case where the viscosity is constant and $\nabla \cdot \tau(\tilde{u}^{n+1})$ is equal to $\mu \Delta \tilde{u}^{n+1}$.

Using the identity $\Delta\tilde{u}^{n+1} = -\nabla\wedge(\nabla\wedge\tilde{u}^{n+1}) + \nabla(\nabla\cdot\tilde{u}^{n+1})$, Eq. (10) reads equivalently:

$$\varrho\left[\frac{\beta_q u^{n+1} - \sum_{j=0}^{q-1}\beta_j u^{n-j}}{\Delta t} + (u^{\star,n+1}\cdot\nabla)\tilde{u}^{n+1}\right] = -\nabla(p^n - r\nabla\cdot\tilde{u}^{n+1} + \phi) - \mu\nabla\wedge(\nabla\wedge\tilde{u}^{n+1}) + \mu\nabla(\nabla\cdot\tilde{u}^{n+1}) + \varrho g \tag{12}$$

Using the fact that the rotational of a gradient vanishes, the relation (7a) yields:

$$\nabla\wedge\tilde{u}^{n+1} = \nabla\wedge u^{n+1}$$

As the end-of-step velocity u^{n+1} is divergence free, we then get:

$$\nabla\wedge(\nabla\wedge\tilde{u}^{n+1}) = \nabla\wedge(\nabla\wedge u^{n+1}) = -\Delta u^{n+1}$$

and the time-discrete momentum balance equation finally reads:

$$\varrho\left[\frac{\beta_q u^{n+1} - \sum_{j=0}^{q-1}\beta_j u^{n-j}}{\Delta t} + (u^{\star,n+1}\cdot\nabla)\tilde{u}^{n+1}\right] = -\nabla(p^n - (r+\mu)\nabla\cdot\tilde{u}^{n+1} + \phi) + \mu\Delta u^{n+1} + \varrho g \tag{13}$$

We then have recovered an expression closer to the semi-implicit time discretization, with an additional term in the pressure increment:

$$p^{n+1} = p^n - (r+\mu)\nabla\cdot\tilde{u}^{n+1} + \phi$$

This variant will be named hereafter the rotational penalty–projection scheme.

Remark (*From penalty methods to penalty–projection methods*). The penalty scheme for the Navier–Stokes equations takes, in its simplest version, the following form:

$$\varrho\left[\frac{Du^{n+1}}{\Delta t} + (u^{\star,n+1}\cdot\nabla)u^{n+1}\right] = -\nabla p^{n+1} + \nabla\cdot\tau(u^{n+1}) + \varrho g$$
$$\nabla\cdot u^{n+1} + \epsilon p^{n+1} = 0$$

where the positive penalty parameter ϵ is taken far smaller than 1. Since the pressure can be explicitly expressed as a function of the velocity divergence by the second equation, it can be eliminated from the system to yield:

$$\varrho\left[\frac{Du^{n+1}}{\Delta t} + (u^{\star,n+1}\cdot\nabla)u^{n+1}\right] - \frac{1}{\epsilon}\nabla(\nabla\cdot u^{n+1}) = \nabla\cdot\tau(u^{n+1}) + \varrho g \tag{14}$$

which introduces an augmentation-like term (with $r = 1/\epsilon$) in the momentum balance equation.

Remark (*Comparison of the present scheme and the penalty–projection method proposed by Shen* [25]). As quoted in the introduction, the first penalty–projection scheme has been introduced by Shen in 1992 [25, Section 6]. The main difference between this scheme and the method presented here lies in the integration of the pressure in the algorithm. Indeed, Shen builds an auxiliary sequence ψ^n, $0 \leqslant n \leqslant N-1$, by the following induction relation:

$$\psi^{n+1} = \psi^n + \phi$$

where ϕ is given by the system (7). The quantity ψ^n is used in the prediction step (6) instead of p^n, and an approximation of the pressure is computed only as a "post-processing" of the outcomes of the computation, by two possible relations in the spirit of (11), the most simple one reading:

$$p^{n+1} = \psi^{n+1} - r\nabla\cdot\tilde{u}^{n+1}$$

In addition, in [25], the augmentation parameter is varying with the time step, and the author shows that a second order Crank–Nicholson discretization of the momentum balance equation combined with an adjustment of r as $r \simeq \Delta t^{-2}$ leads to a second order scheme. We adopt here a different point of view, in trying to assess the properties of the method when the augmentation parameter is kept constant (and within reasonable values), to avoid to excessively degrade the conditioning of the linear system associated to the velocity prediction step.

3. A finite element implementation

We now turn to the finite element implementation of the penalty–projection method. As mentioned in Section 1, due to the fact that the discrete solution is only "discretely-divergence-free", it seems preferable to build the augmentation term at the algebraic level, to avoid a non-intentional perturbation of the solution growing with the augmentation parameter. To this purpose, we must proceed as follows: first, we discretize in space the time-discrete equations of the scheme keeping them free of any augmentation term, then this latter is added afterwards.

Let a finite element discretization of the velocity and pressure be given, or, equivalently, let two Lagrange finite element spaces V_h and M_h included respectively in $[H^1(\Omega)]^d$ and $L^2(\Omega)$ be chosen. In addition, the pressure approximation space is required to be included in $H^1(\Omega)$, to allow a conforming approximation of the Poisson's step.

This discrete variational formulation associated to the coupled problem (3) at the $(n+1)$th time step is then routinely converted into an algebraic system of the form:

$$\begin{cases} \dfrac{\beta_q}{\Delta t}\mathbf{M}\mathbf{U}_{\mathrm{F}} + \mathbf{A}\mathbf{U}_{\mathrm{F}} + \mathbf{B}^{\mathrm{T}}\mathbf{P} = \mathbf{F} \\ \mathbf{B}\mathbf{U}_{\mathrm{F}} = \mathbf{G} \end{cases} \tag{15}$$

where $\mathbf{U}_{\mathrm{F}}$ and $\mathbf{P}$ are vectors gathering the degrees of freedom of, respectively, the velocity and the pressure at time t^{n+1}, and the right-hand members $\mathbf{F}$ and $\mathbf{G}$ represent the effects of the forcing term and non-homogeneous Neuman boundary conditions (for $\mathbf{F}$) together with the effects of the discrete lifting associated to non-homogeneous velocity Dirichlet boundary conditions (for $\mathbf{F}$ and $\mathbf{G}$).

The augmentation term is built by pre-multiplying the discrete divergence constraint by a scaling (symmetrical definite positive) matrix, denoted $\mathbf{M}_{\mathbf{p}}^{-1}$ because a typical choice for this operator is the inverse of the lumped pressure mass matrix, then by the matrix $\mathbf{B}^{\mathrm{T}}$ to obtain a positive symmetrical operator:

$$\text{augmentation term} = r\mathbf{B}^{\mathrm{T}}\mathbf{M}_{\mathbf{p}}^{-1}(\mathbf{B}\mathbf{U}_{\mathrm{F}} - \mathbf{G})$$

where r is the (positive and constant) augmentation parameter. As in the time semi-discrete case, the first step of the penalty–projection method is obtained from the momentum balance equation by taking the pressure at the beginning of the time step and adding the augmentation term:

$$\frac{\beta_q}{\Delta t}\mathbf{M}\widetilde{\mathbf{U}}_{\mathrm{F}} + \mathbf{A}\widetilde{\mathbf{U}}_{\mathrm{F}} + r\mathbf{B}^{\mathrm{T}}\mathbf{M}_{\mathbf{p}}^{-1}\mathbf{B}\widetilde{\mathbf{U}}_{\mathrm{F}} + \mathbf{B}^{\mathrm{T}}\mathbf{P}_{\mathrm{exp}} = \mathbf{F} + r\mathbf{B}^{\mathrm{T}}\mathbf{M}_{\mathbf{p}}^{-1}\mathbf{G} \tag{16}$$

where $\widetilde{\mathbf{U}}_{\mathrm{F}}$ and $\mathbf{P}_{\mathrm{exp}}$ are vectors gathering the degrees of freedom of, respectively, the predicted velocity and the explicit pressure, that is the pressure at t^n.

The algebraic analogue of the projection step (7) reads:

$$\begin{cases} \dfrac{\beta_q}{\Delta t}\mathbf{M}(\mathbf{U}_{\mathrm{F}} - \widetilde{\mathbf{U}}_{\mathrm{F}}) + \mathbf{B}^{\mathrm{T}}\mathbf{\Phi} = 0 \\ \mathbf{B}\mathbf{U}_{\mathrm{F}} = \mathbf{G} \end{cases} \tag{17}$$

where $\mathbf{\Phi}$ is the vector associated to the projector ϕ. Solving the first relation for $\mathbf{U}_{\mathrm{F}}$ and substituting the obtained expression in the second equation yields:

$$\mathbf{B}\mathbf{M}^{-1}\mathbf{B}^{\mathrm{T}}\mathbf{\Phi} = \frac{\beta_q}{\Delta t}(\mathbf{B}\widetilde{\mathbf{U}}_{\mathrm{F}} - \mathbf{G})$$

As, in the general case, the velocity mass matrix is not diagonal, handling the operator $\mathbf{B}\mathbf{M}^{-1}\mathbf{B}^{\mathrm{T}}$ appears to be rather inefficient. Relation (8) suggests to replace it by:

$$\mathbf{L}\mathbf{\Phi} = \frac{\beta_q \varrho}{\Delta t}(\mathbf{B}\widetilde{\mathbf{U}}_{\mathrm{F}} - \mathbf{G}) \tag{18}$$

where $\mathbf{L}$ is the discrete operator associated to a Poisson problem with Dirichlet boundary conditions on Γ_{N} and natural Neumann conditions on Γ_{D}. To avoid some complexity in the assembling process (for instance, eliminating in the linear systems the degrees of freedom of ϕ associated with the pressure nodes located on Γ_{N}

would lead to vectors of different dimension for $\mathbf{\Phi}$ and $\mathbf{P}$), the Dirichlet boundary conditions may be simply imposed by penalization.

Finally, the end of step pressure is obtained by following the same technique than in the time semi-discrete case, that is building a discrete momentum equation by adding relation (16) to the first equation of (17) and comparing to the semi-implicit discretization of the momentum balance (15). This process yields:

$$\mathbf{P} = \mathbf{P}_{\text{exp}} + \mathbf{\Phi} + r\mathbf{M}_{\mathbf{p}}^{-1}(\mathbf{B}\widetilde{\mathbf{U}}_{\text{F}} - \mathbf{G}) \tag{19}$$

To sum up the developments of this section, the algorithm for performing a time step of the penalty–projection method is obtained by gathering Eqs. (16) and (18), the first relation of (17) and (19):

$$\begin{cases} \left[\dfrac{\beta_q}{\Delta t}\mathbf{M} + \mathbf{A} + r\mathbf{B}^{\text{T}}\mathbf{M}_{\mathbf{p}}^{-1}\mathbf{B}\right]\widetilde{\mathbf{U}}_{\text{F}} + \mathbf{B}^{\text{T}}\mathbf{P}_{\text{exp}} = \mathbf{F} + r\mathbf{B}^{\text{T}}\mathbf{M}_{\mathbf{p}}^{-1}\mathbf{G} \\ \mathbf{L}\mathbf{\Phi} = \dfrac{\beta_q \varrho}{\Delta t}(\mathbf{B}\widetilde{\mathbf{U}}_{\text{F}} - \mathbf{G}) \\ \mathbf{U}_{\text{F}} = \widetilde{\mathbf{U}}_{\text{F}} - \dfrac{\Delta t}{\beta_q}\mathbf{M}^{-1}\mathbf{B}^{\text{T}}\mathbf{\Phi} \\ \mathbf{P} = \mathbf{P}_{\text{exp}} + \mathbf{\Phi} + r\mathbf{M}_{\mathbf{p}}^{-1}(\mathbf{B}\widetilde{\mathbf{U}}_{\text{F}} - \mathbf{G}) \end{cases} \tag{20}$$

Remark (*On the derivation of the augmentation term*). A direct finite element space-discretization of the time semi-discrete equations of the preceding section leads to an augmentation term which appears in the variational form of the momentum balance equation as:

$$r\int_{\Omega} \nabla \cdot u_h \nabla \cdot v$$

As the divergence of the velocity test functions v does not lie in the pressure discrete space, this term does not vanish by the continuity equation for the discrete solution. Consequently, using this formulation leads to a dependence of the solution on the augmentation parameter. For the particular discretization used here (Taylor–Hood element), an excessive smearing of the solution is then observed at high values of r.

4. Numerical experiments

The objective of the present section is to perform a comparison between the projection method presented here and various pressure correction schemes widely used in the literature for the solution of unstationary incompressible flow problems. In the time-discrete setting (which, as explained in the preceding section, is somewhat incorrect for the augmentation terms), the methods considered here can be recast under the form of a four stages fractional step scheme as follows:

$$\begin{cases} \varrho\left[\dfrac{\beta_q\tilde{u}^{n+1} - \sum_{j=0}^{q-1}\beta_j u^{n-j}}{\Delta t} + (u^{\star,n+1}\cdot\nabla)\tilde{u}^{n+1}\right] - r_1\nabla(\nabla\cdot\tilde{u}^{n+1}) = -\nabla p^n + \nabla\cdot\tau(\tilde{u}^{n+1}) + \varrho g \quad \text{in } \Omega \\ \tilde{u}^{n+1} = u_{\Gamma_{\text{D}}}^{n+1} \quad \text{on } \Gamma_{\text{D}} \\ -p^n n + \tau(\tilde{u}^{n+1})\cdot n = f_{\text{N}}^{n+1} \quad \text{on } \Gamma_{\text{N}} \end{cases} \tag{21a}$$

$$\begin{cases} \Delta\phi = \dfrac{\beta_q\ \varrho}{\Delta t}\nabla\cdot\tilde{u}^{n+1} \quad \text{in } \Omega \\ \nabla\phi\cdot n = 0 \quad \text{on } \Gamma_{\text{D}} \\ \phi = 0 \quad \text{on } \Gamma_{\text{N}} \end{cases} \tag{21b}$$

$$u^{n+1} = \tilde{u}^{n+1} - \frac{\Delta t}{\beta_q\varrho}\nabla\phi \tag{21c}$$

$$p^{n+1} = p^n + \alpha\phi - r_2\nabla\cdot\tilde{u}^{n+1} \tag{21d}$$

where r_1, r_2 and α are parameters to be specified.

The penalty–projection method is obtained by the choice of parameters $r_1 = r_2 = r$, $\alpha = 1$. For the rotational penalty–projection variant, values of the parameters are $r_1 = r$, $r_2 = r + \mu$, $\alpha = 1$.

The case $r_1 = r_2 = 0$ and $\alpha = 1$ corresponds to the so-called incremental projection method, which is probably today the most popular one. To the best of our knowledge, the origin of this scheme can be traced back to Goda [9], and, since that time, this method has received a considerable attention: see in particular [33] for the extension to (formally) second order in time, [24,26,28] for the analysis in the time semi-discrete case and [13,12,14,15] for the analysis in the fully discrete case. The main identified drawback of this scheme is that, as the parameter r_2 is zero, the pressure inherits the boundary conditions of ϕ, i.e., spurious homogeneous Neumann boundary conditions on no-slip boundaries and homogeneous Dirichlet conditions on open boundaries.

The choice $r_1 = 0$, $r_2 = \mu$ and $\alpha = 1$ leads to the method introduced in [31] and analysed in [17,15]. Following these last two papers, we will refer it as the rotational pressure correction scheme.

As detailed at the end of Section 2, making the particular choice $r_1 = \Delta t^{-2}$ and $r_2 = 0$ yields a method which will give the same results for the velocity as the scheme studied in [25, Section 6]. In this case, a post-processing will be necessary to recover an accurate approximation of the pressure.

Finally, taking $r_1 = r_2 = r$ and $\alpha = 0$, one recovers a scheme which presents some similarities with the so-called vectorial projection method presented in the finite volume framework in [4] (see [1] for a discussion on this topic). The reason why the parameter α is set to zero in the latter work is that the authors make use of a different projection step, which does not provide a natural pressure increment. On the opposite, with the projection step employed here, to include the variable ϕ in the pressure correction seems a reasonable option, and the case $\alpha = 0$ is not considered here.

Table 1 gathers the choices of parameters corresponding to the various methods considered in numerical experiments. Of course, this choice of schemes does not pretend to be exhaustive (see, e.g. [20,2] for alternative pressure correction schemes).

The present section is organized as follows. First, we examine the accuracy of the method on a standard Navier–Stokes benchmark, namely the computation of Taylor–Green vortices. In a second time, we focus on the behaviour of the pressure near the boundaries for a Stokes flow with Dirichlet boundary conditions. Third, we deal with a case with open boundary conditions. Finally, we turn to a more complex situation, the two-dimensional flow past a cylinder.

All the simulations presented here are performed with a formally second order in time scheme, corresponding to Eq. (5), i.e., a second order backward differentiation formula for the approximation of the time derivative and a second order Richardson extrapolation for the estimation of the advection field. This scheme is initialized with a first time step performed with a standard backward Euler differentiation (relation (4)). The pressure at $t = 0$ is obtained by interpolation of the solution when available (i.e., in the first three cases) or set to zero. The problem is discretized in space using $\mathbf{P}_2$–P_1 finite elements (the so-called Taylor–Hood mixed finite element, see, e.g. [8], [6, Chapter 5], [22, Chapter 9]). The solution of the coupled system obtained when using the semi-implicit scheme is performed by a standard augmented Lagrangian technique (see [7]).

The practical implementation has been performed using the software object-oriented component library PELICANS, developed at IRSN [21].

4.1. Taylor–Green vortices

As a first benchmark for the proposed numerical method, we use a particular periodic flow widely used in the literature (e.g. [20]), which enjoys the property to obey the Navier–Stokes equations with a zero forcing

Table 1
Choice of parameters in (20) for the schemes used in the numerical experiments

Incremental projection method (standard form)	$r_1 = 0$, $r_2 = 0$, $\alpha = 1$
Incremental projection method (rotational form)	$r_1 = 0$, $r_2 = \mu$, $\alpha = 1$
Penalty–projection method (standard form)	$r_1 = r$, $r_2 = r$, $\alpha = 1$
Penalty–projection method (rotational form)	$r_1 = r$, $r_2 = r + \mu$, $\alpha = 1$

term. It is a particular case of a class of flows which seems to have been first introduced by Taylor and Green [29]. The velocity and pressure fields read:

$$u(x,y,t) = \begin{bmatrix} -\cos(2\pi x) & \sin(2\pi y) \\ \sin(2\pi x) & \cos(2\pi y) \end{bmatrix} \exp(-8\pi^2 \mu t)$$

$$p(x,y,t) = -\frac{(\cos(4\pi x) + \cos(4\pi y))}{4} \exp(-16\pi^2 \mu t)$$

The chosen computational domain is $\Omega = [1/8,5/8] \times [1/8,5/8]$ and the velocity is imposed on the whole boundary. Of course, initial and boundary conditions are taken in such a way to match the analytical solution. The density is set to $\varrho = 1$ and the viscosity to $\mu = 0.01$.

The meshes used in this study are obtained by cutting the domain in an $n \times n$ regular grid and splitting each squared cell along its diagonals to obtain four simplicial elements.

Figs. 1–3 show the difference between the numerical solution and the analytical one at $t = 1$, measured in $[L^2(\Omega)]^d$ and $[H^1(\Omega)]^d$ norms for the velocity and in $L^2(\Omega)$ norm for the pressure. These curves are drawn for the 80×80 mesh.

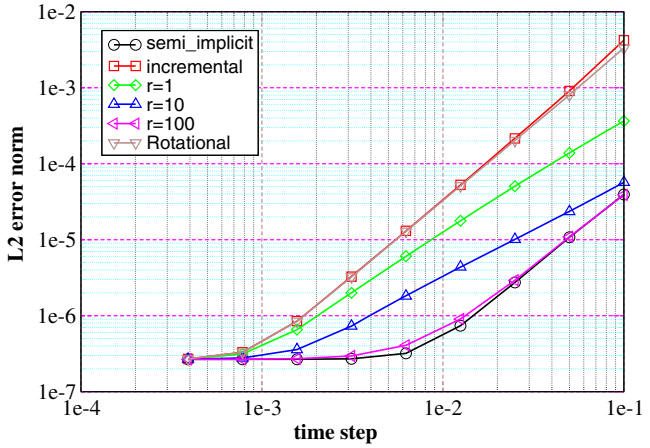

Fig. 1. Taylor–Green vortex-L^2 norm of the error for the velocity as a function of the time step, for the incremental, rotational, penalty projection ($r = 1$, $r = 10$, $r = 100$) schemes and for the semi-implicit scheme.

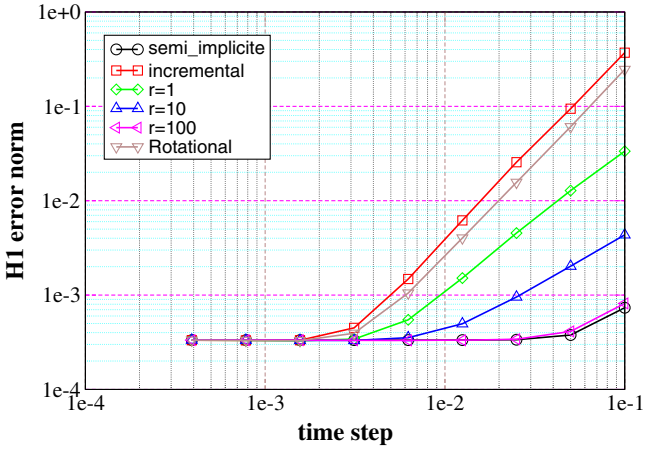

Fig. 2. Taylor–Green vortex-H^1 norm of the error for the velocity as a function of the time step, for the incremental, rotational, penalty–projection ($r = 1$, $r = 10$, $r = 100$) schemes and for the semi-implicit scheme.

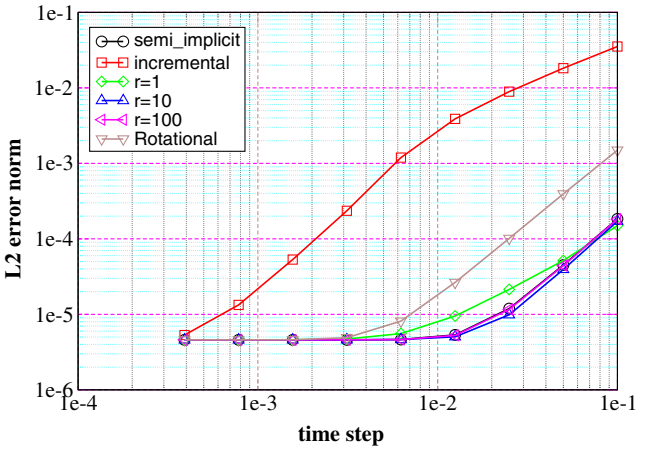

Fig. 3. Taylor–Green vortex-L^2 norm of the error for the pressure as a function of the time step, for the incremental, rotational, penalty–projection ($r = 1$, $r = 10$, $r = 100$) schemes and for the semi-implicit scheme.

Table 2
Difference between the exact solution and the numerical solution on the time-convergence plateau and between the exact solution and its interpolate (values obtained at $t = 1$)

	20×20	40×40	80×80
$\|\|u - u_h\|\|_{[L^2(\Omega)]^d}$	1.09×10^{-4}	1.34×10^{-5}	1.66×10^{-6}
$\|\|u - u_i\|\|_{[L^2(\Omega)]^d}$	2.07×10^{-5}	2.59×10^{-6}	3.23×10^{-7}
$\|\|u - u_h\|\|_{[H^1(\Omega)]^d}$	3.75×10^{-3}	9.34×10^{-4}	2.32×10^{-4}
$\|\|u - u_i\|\|_{[H^1(\Omega)]^d}$	9.13×10^{-4}	2.28×10^{-4}	5.71×10^{-5}
$\|\|p - p_h\|\|_{L^2(\Omega)}$	2.86×10^{-3}	7.14×10^{-4}	1.78×10^{-4}
$\|\|p - p_i\|\|_{L^2(\Omega)}$	5.04×10^{-3}	1.26×10^{-3}	3.15×10^{-4}

All the curves show an error decrease with the time step for large values of this latter, then a plateau is observed, which corresponds to the space discretization error. Values of the error on the plateau are gathered, for various meshes, in Table 2, together with the interpolation error (difference between the analytical solution and the element of the approximation space obtained by interpolating the solution at each node). One can check than an optimal spatial convergence is found in each case.

As far as the time convergence is concerned, the first observation is the outstanding importance of the splitting error associated to the standard projection schemes at large time steps; for the incremental projection method, for instance, the error is about two decades higher than for the semi-implicit scheme, and this ratio even increases for the pressure when the time step decreases, since these schemes does not exhibit the same convergence rate.

The rates of convergence globally agree with theoretical studies. For the velocity, a second order convergence is observed in the $[L^2(\Omega)]^d$ norm for the semi-implicit scheme (see [18] for an in-depth study of the Crank–Nicholson time stepping method), the incremental projection [14] and the rotational projection method [17]. The pressure is first order accurate for the incremental projection method and second order accurate for the rotational projection scheme and the semi-implicit scheme, which is slightly better than theoretical bounds: a 3/2 order convergence for the rotational projection scheme is proven in [17] (and seems to be optimal for general computational domains), a second order convergence for the pressure in a weaker norm than the $L^2(\Omega)$ one is given in [23] (see remark below).

Concerning the penalty–projection scheme, the first point is that any penalization leads to a decrease of the error. Keeping a constant value of the augmentation parameter r when the time-step decreases, the second order in time convergence is lost, but the error remains bounded from above by the incremental projection one. Second, as can be inferred from the literature of penalty methods [27], the splitting error vanishes when r raises. However, keeping this parameter within relatively moderate bounds seems to be sufficient ($r = 100$ for the greater value, having in mind the presence of the inverse of the pressure mass matrix as scaling operator in the definition of the penalty term). Finally, the results compare with the rotational projection scheme as follows.

For the velocity, the results with $r = \mu$ are superimposed (the corresponding curves for the penalty–projection scheme have been omitted in Figs. 1 and 2); for this test, where the viscosity is rather small ($\mu = 0.01$), the difference with the incremental projection scheme is hardly visible. For the pressure, the rotational scheme is clearly better than the penalty–projection scheme with $r = \mu$, but the opposite conclusion holds for larger values of the augmentation parameter. The same behaviour has been confirmed in all the experiments presented in this paper.

Remark (*Convergence of the incremental projection scheme*).

1. The evolution of the pressure error as a function of the time step for the incremental projection scheme shows a linear decrease at large time steps, then, for smaller values of the time step, the convergence seems to become faster. As pointed out in [16, Section 7], this phenomenon has the following explanation. Higher order convergence results for the pressure are known to hold in weaker norms [12, Section 4.4] than the $L^2(\Omega)$ one. As in finite dimensional spaces, all the norms are equivalent, the $L^2(\Omega)$ norm of the error is bounded by the weaker norm multiplied by a coefficient depending on the meshing. For large values of the time step, the L^2 estimate yields the sharpest bound and a first order convergence rate is observed; then, when the time step falls below a threshold value (decreasing with the mesh size), the bound in the weaker norm becomes sharper and a faster decrease of the error occurs.
2. The pressure convergence is known to be limited by the prescription of spurious boundary conditions to the pressure increment in the projection step. This can be checked here by the following simple numerical experiment. Let us solve the same problem on a different computational domain $\Omega = [1/4, 5/4] \times [1/4, 5/4]$, such that (by chance) the exact pressure satisfies homogeneous Neumann boundary conditions over the whole boundary. Then a second order convergence rate is recovered.

4.2. A Stokes flow with Dirichlet boundary conditions

The incremental projection method is known to be plagued by the presence of boundary layers in the pressure solution, which results from artificial homogeneous Neumann boundary conditions prescribed to the pressure increment on the part of the boundary where the velocity is prescribed. This phenomenon is highlighted in the setting used here, where a discrete pressure Laplacian operator with Neumann boundary conditions is explicitly built; however, similar problems also affect projection methods derived via an inexact factorization of the algebraic discrete equations (e.g. [10,11,22,32] for a presentation of such methods, [16, Section 6] and [32, pp. 53–54], [22] for a discussion on this topic). The aim of this section is to assess the behaviour of the penalty–projection scheme with respect to this issue.

As the behaviour of the pressure error in the L^∞ norm is strongly affected by the regularity of the domain [17,16], we chose for Ω a circle, of diameter 1. We only solve an unstationary Stokes problem (with $\mu = 1$, $\nabla \cdot \tau(u) = \mu \Delta u$), and we adjust the forcing term in such a way that the velocity and pressure fields read:

$$u(x, y, t) = \begin{bmatrix} \sin(x + t) & \sin(y + t) \\ \cos(x + t) & \cos(y + t) \end{bmatrix}$$

$$p(x, y, t) = \cos(x - y + t)$$

This test case is the same as studied in [17,16], which allows a verification of our computations, as methods considered in these papers are part of the set of schemes tested in the present work.

We use for this test an unstructured mesh of 3600 elements.

The distribution of the pressure error for the considered schemes at $t = 1$ is shown in Fig. 4. The time step used for these computations is $\Delta t = 0.0125$.

As expected, a boundary layer is observed for the pressure error with the incremental scheme. This error measured in the L^∞ norm is, for this scheme, almost two orders of magnitude greater than for the implicit method.

This phenomenon disappears for the rotational variants of both the incremental projection scheme and the penalty–projection scheme, and for the standard penalty–projection method for high values of the augmentation parameter. The spatial distribution of the pressure error then becomes strongly influenced by the geometrical structure of the meshing. In particular, peaks appear along a diameter which has been used as a symmetry axis for the mesh generation.

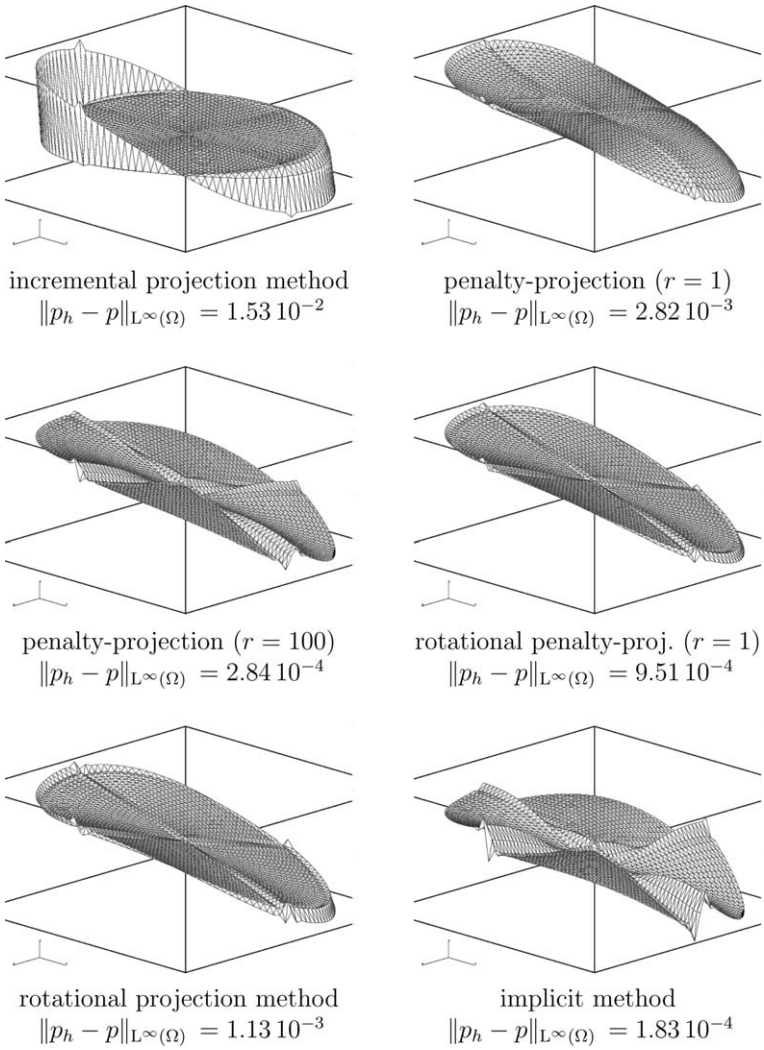

Fig. 4. Stokes flow with prescribed velocity boundary conditions – distribution of the pressure error for the incremental, rotational, penalty–projection ($r = 1$, $r = 10$, $r = 100$) schemes and for the implicit scheme.

For the rotational projection scheme, the error is still 10 times higher than for the implicit scheme. Equivalent results are obtained with the penalty–projection scheme for r ranging between 1 and 10, then, for higher values of r, the error becomes smaller.

4.3. A Stokes flow with open boundary conditions

If the prescription of spurious Neumann boundary conditions for the pressure on Dirichlet parts of the boundary is an unquestionable drawback of the incremental projection method, things even go worse when open boundaries are to be dealt with. Indeed, homogeneous Dirichlet boundary conditions have to be artificially enforced to the pressure increment on that part of the boundary. This is done either explicitly (when a pressure Poisson problem is built at the continuous level) or implicitly (within the algebraic splitting approach, see [16, Section 6] and [32, pp. 53–54] for a related discussion). Since the pressure approximation space is consequently modified, one may fear to lose even the spatial accuracy of the scheme; both theoretical and experimental evidences that this indeed occurs are given in [15].

As before, the penalty–projection scheme and the rotational projection scheme share the property that the spurious boundary conditions are imposed only to the intermediate variable ϕ and not (by induction) to the

pressure itself. Hence we can hope that these schemes enjoy significantly better convergence properties. The goal of the present section is to check this issue.

As in the preceding section, we choose a test case already used in the literature [15,16]. It consists in an unstationary Stokes problem (with $\mu = 1$, $\nabla \cdot \tau(u) = \mu \Delta u$), with a forcing term and initial and boundary conditions corresponding to the following expression for the velocity and pressure fields:

$$u(x,y,t) = \begin{bmatrix} \sin(x) & \sin(y+t) \\ \cos(x) & \cos(y+t) \end{bmatrix}$$

$$p(x,y,t) = \cos(x)\sin(y+t)$$

The computational domain is $\Omega = [0,1] \times [0,1]$ and the velocity is prescribed on its boundary, except for the part included in the y-axis, where homogeneous natural Neumann conditions are imposed:

$$\mu \nabla u \cdot n - pn = 0$$

The meshes are obtained by cutting along its diagonals in four simplices each square of a $n \times n$ regular grid.

Figs. 5–7 show, for the 80×80 meshing, the difference between the numerical solution and the analytical one at $t = 1$, measured in $[L^2(\Omega)]^d$ and $[H^1(\Omega)]^d$ norms for the velocity and in $L^2(\Omega)$ norm for the pressure.

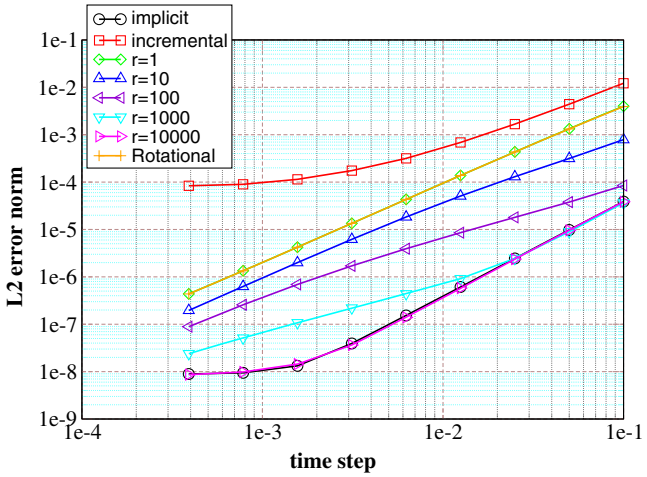

Fig. 5. Stokes flow with open boundary conditions – L^2 norm of the error for the velocity as a function of the time step, for the incremental, rotational, penalty–projection ($r = 1$, $r = 10$, $r = 100$) schemes and for the implicit scheme.

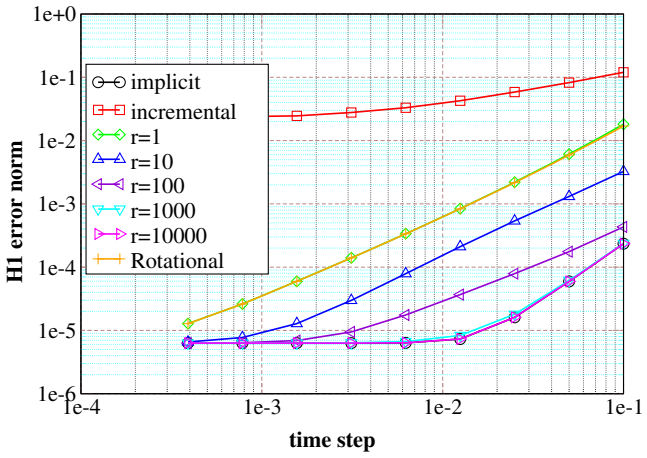

Fig. 6. Stokes flow with open boundary conditions – H^1 norm of the error for the velocity as a function of the time step, for the incremental, rotational, penalty–projection ($r = 1$, $r = 10$, $r = 100$) schemes and for the implicit scheme.

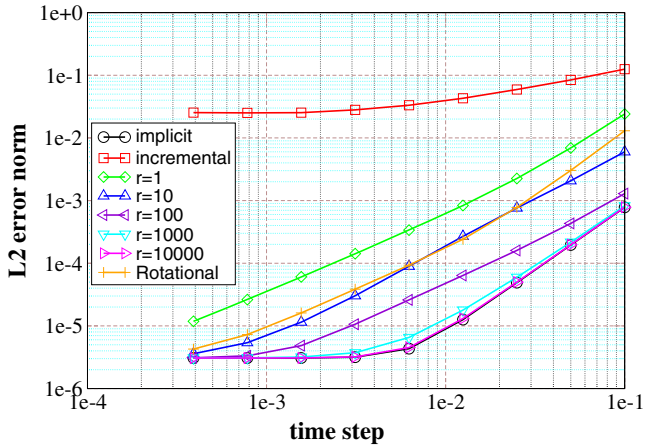

Fig. 7. Stokes flow with open boundary conditions – L^2 norm of the error for the pressure as a function of the time step, for the incremental, rotational, penalty–projection ($r = 1$, $r = 10$, $r = 100$) schemes and for the implicit scheme.

As expected, the incremental projection method converges for small time steps to a solution which differs from the plateau obtained with the other schemes, and corresponds to a much greater error for both velocity and pressure. Errors obtained at small time steps are gathered for various meshings in Table 3 for the incremental projection scheme and in Table 4 for the other schemes. A space convergence order of 1/2 is observed for the incremental projection method, which is in agreement with the bound proven in [15], whereas an optimal convergence rate is obtained for the other schemes.

A second order convergence with respect to the time step is observed for the implicit scheme for both the velocity and the pressure. The results of the rotational projection scheme conform to the numerical experiments reported in [15]: the convergence rate is respectively *1.65* and *1* for the velocity in $[L^2(\Omega)]^d$ and $[H^1(\Omega)]^d$ norm, and lies between 1 and 3/2 for the pressure in $L^2(\Omega)$ norm. As for the preceding numerical experiments, the penalty–projection method with $r = \mu$ (=1) gives for the velocity the same results as the rotational projection scheme, then the results come closer and closer to the implicit scheme when the augmentation

Table 3
Difference between the exact solution and the numerical solution on the time-convergence plateau, for the incremental projection method, and difference between the exact solution and its interpolate (values obtained at $t = 1$)

	20×20	40×40	80×80
$\|\|u - u_h\|\|_{[L^2(\Omega)]^d}$	7.76×10^{-4}	3.24×10^{-4}	1.46×10^{-4}
$\|\|u - u_i\|\|_{[L^2(\Omega)]^d}$	9.66×10^{-7}	1.21×10^{-7}	1.51×10^{-8}
$\|\|u - u_h\|\|_{[H^1(\Omega)]^d}$	3.64×10^{-2}	2.59×10^{-2}	1.84×10^{-2}
$\|\|u - u_i\|\|_{[H^1(\Omega)]^d}$	8.39×10^{-5}	2.10×10^{-5}	5.26×10^{-6}
$\|\|p - p_h\|\|_{L^2(\Omega)}$	6.14×10^{-2}	4.34×10^{-2}	3.07×10^{-2}
$\|\|p - p_i\|\|_{L^2(\Omega)}$	2.19×10^{-4}	5.47×10^{-5}	1.37×10^{-5}

Table 4
Difference between the exact solution and the numerical solution on the time-convergence plateau, for schemes other than the incremental projection method, and difference between the exact solution and its interpolate (values obtained at $t = 1$)

	20×20	40×40	80×80
$\|\|u - u_h\|\|_{[L^2(\Omega)]^d}$	9.12×10^{-7}	1.14×10^{-7}	1.55×10^{-8}
$\|\|u - u_i\|\|_{[L^2(\Omega)]^d}$	9.66×10^{-7}	1.21×10^{-7}	1.51×10^{-8}
$\|\|u - u_h\|\|_{[H^1(\Omega)]^d}$	7.77×10^{-5}	1.94×10^{-5}	4.85×10^{-6}
$\|\|u - u_i\|\|_{[H^1(\Omega)]^d}$	8.39×10^{-5}	2.10×10^{-5}	5.26×10^{-6}
$\|\|p - p_h\|\|_{L^2(\Omega)}$	6.01×10^{-5}	1.51×10^{-5}	3.76×10^{-6}
$\|\|p - p_i\|\|_{L^2(\Omega)}$	2.19×10^{-4}	5.47×10^{-5}	1.37×10^{-5}

parameter increases. For the pressure, a higher value ($r = 10$) is necessary to recover the accuracy of the rotational projection method, this point being corrected by the use of the rotational penalty–projection variant.

4.4. Flow past a cylinder

In this section, we are concerned by a more physically meaningful problem, namely the two-dimensional, incompressible, laminar and unsteady flow of a viscous fluid around a solid circular cylinder, in the layout proposed in [19].

We use a rectangular domain $\Omega = [0,20] \times [0,5]$ with a solid obstacle represented by a disk C. Its center lies at the point $(5,2.5)$ and its diameter is equal to 1.

The meshing used for this test consists in roughly 4200 elements, which leads to about 8500 degrees of freedom for each component of the velocity, 2200 for the pressure. The velocity is prescribed to zero on the boundary of C and to the value $(1,0)$ at the inflow boundary of the computational domain. We impose a perfect slip condition to the velocity at the top and bottom of Ω and natural Neumann conditions at the exit.

The density is set to $\varrho = 1$ and the viscosity to $\mu = 0.01$, which leads to a cylinder diameter-based Reynolds number of 100, at which the flow is known to be unsteady, exhibiting the periodic shedding of vortices downstream the cylinder. This phenomenon is triggered in our computation by introducing a perturbation affecting the inlet velocity during one time unit, starting at time $t = 15$.

Results are clearly sensitive to the boundary conditions prescribed at the top and bottom of the computational domain, which shows than an accurate simulation of the physics of the flow would require a more appropriate treatment of the boundary conditions (either enlarging the computational domain, either using more sophisticated modelling of artificial boundaries as proposed in [3]). However, they are in reasonable agreement with the literature: drag coefficient varying in time from 1.69 to 1.72 and vortex shedding period of 4.92 time units.

Fig. 8 shows the evolution versus the augmentation parameter r of the difference between the velocity fields obtained at time $t = 60$ by the semi-implicit coupled scheme and the penalty–projection one, measured in the $[L^2(\Omega)]^d$ norm. When r is large, the magnitude of the splitting error varies approximately as $1/r$. Moreover, we can see that it decreases with the time step, with an apparent order lying between 1 and 2. This behaviour can also be inferred from the variation of the total error as a function of the time step observed in the other test cases.

5. Discussion

The numerical experiments reported in this paper show that the penalty–projection method yields a considerable gain in accuracy compared to incremental projection schemes, implemented as well in standard form as

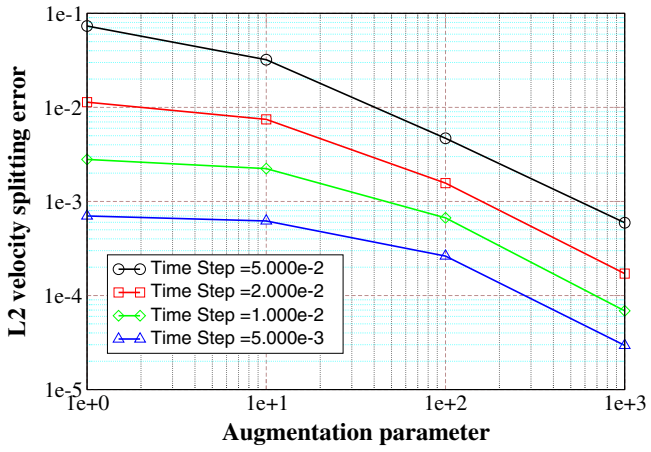

Fig. 8. Flow past a cylinder – L^2 norm of the splitting error for the velocity as a function of the augmentation parameter r, for the penalty–projection method.

in the rotational form (see [16] for a review). The splitting error is reduced as soon as the augmentation parameter r takes a significant value, and decreases down to negligible values as r is increased. Results of the semi-implicit method are thus recovered in the latter case. The pressure layers suffered by the incremental projection scheme in the vicinity of the prescribed velocity boundaries are suppressed. Finally, the loss of spatial convergence of the standard incremental projection scheme in case of open boundary conditions does not occur anymore.

As a consequence, we observe that the scheme is convergent, or, more precisely speaking, enjoys convergence properties at least equivalent to the standard incremental projection method, whatever the augmentation parameter may be. This implies, in particular, that the rate of convergence for the velocity is bounded from above by the second order estimate valid for the incremental scheme with Dirichlet boundary conditions (even if the penalty–projection scheme is not second order), as soon as a formally second-order scheme is employed. We confirm this desirable feature by a theoretical analysis in energy norms in [1].

As classically observed when making use of penalty or augmented Lagrangian methods, the price to pay for this gain in accuracy is an ill-conditioning of the linear system associated to the prediction step, which may reinforce the importance of keeping a reasonable value for the augmentation parameter. Note also that the decoupling of the velocity components in the prediction step, when the divergence of the stress tensor can be expressed as the velocity Laplacian (i.e., for constant viscosity and particular boundary conditions), is lost. However, even if the penalty–projection scheme is more time-consuming than the considered incremental projection methods for a given time-step, it has been observed in our tests to be cheaper to yield a given accuracy. Note nevertheless that this fact is reported here as a first information, but clearly needs additional assessment, with respect to at least two aspects. First, a numerical efficiency study must include a careful discussion concerning the preconditioning of the linear algebraic operators, specially for the velocity prediction step, in which the convergence of Krylov's subspaces methods highly depends on this issue; this point was disregarded in the present work. Second, the behaviour of linear solvers appears to be highly problem-dependent. For instance, at small time step and for the ILU preconditioned GMRES algorithm, taking a Richardson's time extrapolation of the velocity as starting point in the prediction stage, convergence was obtained within less than twenty iterations for the Taylor–Green vortices test, as it took a few hundred iterations for the computation of the flow past a cylinder. A careful design of a numerical efficiency study should then deal with this issue, which has not been the case for the choice of the tests investigated here. With this respect, note that the interest of the penalty–projection scheme should grow for non-isothermal applications which involve temperature-dependent physical properties (and even, for some industrial applications, properties tabulated with respect to the temperature), for which the part of the CPU time devoted to the assembling of the discrete operators becomes important. In any case, testing the present scheme in a multi-grid framework seems appealing; this is the subject for a further work.

Acknowledgments

The work of the first two authors has been partly supported by the Région Provence-Alpes-Côte d'Azur. It has been followed by the PRINCIPIA and OPTIFLOW companies.

References

[1] Ph. Angot, M. Jobelin, J.-C. Latché, On two variants of the penalty–projection method, Mathematical Modelling and Numerical Analysis (submitted for publication).
[2] John B. Bell, Philip Colella, Harland M. Glaz, A second-order projection method for the incompressible Navier–Stokes equations, Journal of Computational Physics 85 (1989) 257–283.
[3] C.H. Bruneau, P. Fabrie, Effective downstream boundary conditions for incompressible Navier–Stokes equations, International Journal for Numerical Methods in Fluids 19 (1994) 693–705.
[4] Jean-Paul Caltagirone, Jérôme Breil, Sur une méthode de projection vectorielle pour la résolution des équations de Navier–Stokes, Comptes-Rendus de l'Académie des Sciences, Paris – Série II 327 (1999) 1179–1184.
[5] Alexandre Joel Chorin, Numerical solution of the Navier–Stokes equations, Mathematics of Computation 22 (1968) 745–762.
[6] Alexandre Ern, Jean-Luc Guermond, Eléments finis: théorie, applications, mise en œuvre, Mathématiques & Applications, vol. 36, Springer, Germany, 2002.
[7] M. Fortin, R. Glowinski, Méthodes de Lagrangien Augmenté, Dunod, Paris, 1982.

[8] Vivette Girault, Pierre-Arnaud Raviart, Finite element methods for Navier–Stokes equations, Theory and Algorithms, Springer Series in Computational Mathematics, vol. 5, Springer-Verlag, Germany, 1986.
[9] Katuhiko Goda, A multistep technique with implicit difference schemes for calculating two- or three-dimensional cavity flows, Journal of Computational Physics 30 (1979) 76–95.
[10] Philip M. Gresho, On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 1: Theory, International Journal for Numerical Methods in Fluids 11 (1990) 587–620.
[11] Philip M. Gresho, On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 2: Implementation, International Journal for Numerical Methods in Fluids 11 (1990) 621–659.
[12] J.-L. Guermond, L. Quartapelle, On the approximation of the unsteady Navier–Stokes equations by finite element projection methods, Numerische Mathematik 80 (1998) 207–238.
[13] Jean-Luc Guermond, Some implementations of projection methods for Navier–Stokes equations, Mathematical Modelling and Numerical Analysis 30 (5) (1996) 637–667.
[14] Jean-Luc Guermond, Un résultat de convergence d'ordre deux en temps pour l'approximation des équations de Navier–Stokes par une technique de projection incrémentale, Mathematical Modelling and Numerical Analysis 33 (1) (1999) 169–189.
[15] J.L. Guermond, P. Minev, Jie Shen, Error analysis of pressure-correction schemes for the Navier–Stokes equations with open boundary conditions, SIAM Journal on Numerical Analysis 43 (1) (2005) 239–258.
[16] J.L. Guermond, P. Minev, Jie Shen, An overview of projection methods for incompressible flows, International Journal on Numerical Methods in Engineering (submitted for publication).
[17] J.L. Guermond, Jie Shen, On the error estimates for the rotational pressure-correction projection methods, Mathematics of Computation 73 (248) (2004) 1719–1737.
[18] John G. Heywood, Rolf Rannacher, Finite element approximation of the nonstationary Navier–Stokes problem. Part IV: Error analysis for second-order time discretization, SIAM Journal on Numerical Analysis 27 (2) (1990) 353–384.
[19] Khodor Khadra, Philippe Angot, Sacha Parneix, Jean-Paul Caltagirone, Fictitious domain approach for numerical modelling of Navier–Stokes equations, International Journal for Numerical Methods in Fluids 34 (2000) 651–684.
[20] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, Journal of Computational Physics 59 (1985) 308–323.
[21] Bruno Piar, PELICANS: Un outil d'implémentation de solveurs d'équations aux dérivées partielles, Note Technique 2004/33, IRSN, 2004 (in French).
[22] A. Quarteroni, A. Valli, Numerical Approximation of Partial Differential Equations, Springer, Germany, 1994.
[23] Rolf Rannacher, Finite element methods for the incompressible Navier–Stokes equations, in: Fundamental Directions in Mathematical Fluid Mechanics, Birkhauser, Basel, 2000, pp. 191–293.
[24] Jie Shen, On error estimates of projection methods for Navier–Stokes equations: first-order schemes, SIAM Journal on Numerical Analysis 29 (1) (1992) 57–77.
[25] Jie Shen, On error estimates of some higher order projection and penalty–projection methods for Navier–Stokes equations, Numerische Mathematik 62 (1992) 49–73.
[26] Jie Shen, Remarks on the pressure estimates for the projection methods, Numerische Mathematik 67 (1994) 513–520.
[27] Jie Shen, On error estimates of the penalty method for unsteady Navier–Stokes equations, SIAM Journal on Numerical Analysis 32 (2) (1995) 386–403.
[28] Jie Shen, On error estimates of projection methods for Navier–Stokes equations: second-order schemes, Mathematics of Computation 65 (215) (1996) 1039–1065.
[29] G.I. Taylor, B.A. Green, Mechanism of the production of small eddies from large ones, Proceedings of the Royal Society of London A 158 (1935) 499–521.
[30] R. Temam, Sur l'approximation de la solution des Équations de Navier–Stokes par la méthode des pas fractionnaires (II), Archive for Rational Mechanics and Analysis 33 (1969) 377–385.
[31] L.J.P. Timmermans, P.D. Minev, F.N. Van de Vosse, An approximate projection scheme for incompressible flow using spectral elements, International Journal for Numerical Methods in Fluids 22 (1996) 673–688.
[32] Stefan Turek, Efficient Solvers for Incompressible Flow Problems: An Algorithmic Approach in View of Computational Aspects, Springer, Germany, 1999.
[33] J. Van Kan, A second-order accurate pressure-correction scheme for viscous incompressible flow, SIAM Journal on Scientific and Statistical Computing 7 (3) (1986) 870–891.