COMPUTATIONAL & APPLIED MATHEMATICS

Volume 24, N. 3, pp. 417–438, 2005
Copyright © 2005 SBMAC
ISSN 0101-8205
www.scielo.br/cam

Time and Space parallelization of the Navier-Stokes equations

ISIDORO I. ALBARREAL NÚÑEZ[1], M. CARMEN CALZADA CANALEJO[2], JOSÉ LUIS CRUZ SOTO[2], ENRIQUE FERNÁNDEZ CARA[1], JOSÉ R. GALO SÁNCHEZ[2] and MERCEDES MARÍN BELTRÁN[2]

[1]Departamento E.D.A.N., Universidad de Sevilla, Aptdo. 1160, E-41080 Sevilla
[2]Departamento de Informática y Análisis Numérico, Universidad Córdoba,
Campus de Rabanales, Ed. C2-3, E-14071 Córdoba
E-mails: iignacio@us.es / cara@us.es / ma1canam@uco.es / jlcruz@uco.es /
merche@uco.es / ma1gasaj@uco.es.

Abstract. In this paper, we will be mainly concerned with a parallel algorithm (in time and space) which is used to solve the incompressible Navier-Stokes problem. This relies on two main ideas: (a) a splitting of the main differential operator which permits to consider independently the most important difficulties (nonlinearity and incompressibility) and (b) the approximation of the resulting stationary problems by a family of second-order one-dimensional linear systems. The same strategy can be applied to two-dimensional and three-dimensional problems and involves the same level of difficulty. It can be also useful for the solution of other more complicate systems like Boussinesq or turbulence models. The behavior of the method is illustrated with some numerical experiments.

Mathematical subject classification: 65M06, 35A35, 68W10.

Key words: Navier-Stokes equations, numerical solution, parallel algorithms.

1 Introduction

We will consider here a numerical method for solving the incompressible, time-dependent, Navier-Stokes equations. These equations can be used to model the

#622/04. Received: 9/XII/04. Accepted: 5/IV/05.

behavior of a homogeneous, incompressible, viscous newtonian fluid. When we impose Dirichlet conditions on the velocity field, the problem reads

$$
\left\{
\begin{array}{lll}
\dfrac{\partial \mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p = \mathbf{f}(\mathbf{x},t) & \text{in} & Q = \Omega \times (0,T), \\
\nabla\cdot\mathbf{u} = 0 & \text{in} & \Omega\times(0,T), \\
\mathbf{u}(\mathbf{x},t) = \mathbf{0} & \text{on} & \partial\Omega\times(0,T), \\
\mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(x) & \text{in} & \Omega.
\end{array}
\right. \tag{1}
$$

Here, $\Omega \subset \mathbb{R}^d$ is a bounded regular domain ($d = 2$ or 3), $\mathbf{u} = \mathbf{u}(\mathbf{x},t)$ is the velocity field, $p = p(\mathbf{x},t)$ is the pressure, $\nu > 0$ is the kinematic viscosity (a positive constant) and $\mathbf{f} = \mathbf{f}(\mathbf{x},t)$ is the density function of a field of external forces. For simplicity, we have assumed in (1) that the fluid has unit mass density.

Concerning the solution to (1), it is well known that we can only expect to get numerical approximations. However, it is also well understood nowadays that this is a very difficult task.

A good strategy seems to be the use of *parallel computers*. Of course, in order to optimize their efficiency, one has to design appropriate algorithms (in general terms, the next generation of processors is expected to multiply the speed of computation by a factor 10; at the same time, new forthcoming parallel algorithms are expected to produce an increase of a factor 100, see [22]). However, up to now, parallelization has been performed almost always at the lowest level, when the task has been reduced to the solution of finite-dimensional linear systems with probably many unknowns.

The goal of this work is to propose a different method which relies on parallelization at the highest possible level and tries to reduce as much as possible the computer time by using a large number of processors. It will be seen that this method leads to difficulties essentially of the same kind in the 2D and 3D settings. The seminal ideas for this approach can be found in [19].

In this paper, we will only consider low or moderate Reynolds numbers (respectively up to 4000 and 1000 in 2D and 3D problems). Recall that the Reynolds number of (1) is given by $Re = UL/\nu$, where U and L are characteristic values of the velocity field modulus and the length, respectively. For higher Re, we would need more subtle arguments and methods. The design of appropriate techniques, similar to those in this paper, for the numerical solution of large Reynolds number problems (1) will be the subject of future work.

As usual, the approximation of (1) is performed in two steps. We first discretize in the time variable and, then, we solve numerically the resulting stationary problems by introducing a spatial approximation.

At both steps, we can use a plenty of methods. Among all them, let us mention *viscosity splitting methods* and in particular θ*-scheme fractional schemes* for the approximation in time and *finite element* and *finite difference methods* for the approximation in space. A detailed analysis of the behavior of these and many other methods can be found in [15].

Our interest has focused on the design and analysis of numerical schemes relying on two main ideas: (a) to split or separate in parallel the most important difficulties (nonlinearity and incompressibility) and (b) to approximate the resulting stationary problems by a (large) family of second-order, completely independent, one-dimensional linear systems.

At the 1D level, it will be then easy an adequate to apply finite difference techniques to produce good approximations. In this way, the solution strategy will make possible a very high level or parallelization.

The research described in this paper is a small part of a much larger project concerning parallelization and nonlinear partial differential systems. Up to now, this has led to some publications and PhD Theses. See for instance [1]–[3], [5], [6], [8]–[12]. However, the numerical techniques we present below can be useful for solving many different problems: linear and semilinear elliptic and parabolic systems with nonlinear boundary conditions, Boussinesq systems, one-equation and two-equation turbulence models, fluid-solid interaction models, fully nonlinear equations of the Monge-Ampère kind, etc.

2 The algorithm

Before recalling the formulation of the algorithm, let us introduce some notation:

- $J(\Omega) = \{\varphi \in C_0^\infty(\Omega)^d \, : \, \nabla \cdot \varphi = 0 \text{ in } \Omega\}$; H (resp. V) is the closure of $J(\Omega)$ in the space $L^2(\Omega)^d$ (resp. $H_0^1(\Omega)^d$). Thus, H (resp. V) is a Hilbert space for the scalar product of $L^2(\Omega)^d$ (resp. $H_0^1(\Omega)^d$), which will be denoted by $(\cdot\,,\cdot)$ (resp. $((\cdot\,,\cdot))$). The associated norm will be denoted by $|\cdot|$ (resp. $\|\cdot\|$).

- V' is the dual space of V; $\langle\cdot\,,\cdot\rangle$ denotes the duality pairing between V' and V.
- We also introduce the trilinear forms $b(\cdot\,,\cdot\,,\cdot)$ and $\hat{b}(\cdot\,,\cdot\,,\cdot)$, with
$$b(\mathbf{u},\mathbf{v},\mathbf{w}) = \int_\Omega u_i\, D_i v_j\, w_j\, dx, \quad \hat{b}(u,v,w) = \frac{1}{2}\,(b(\mathbf{u},\mathbf{v},\mathbf{w}) - b(\mathbf{u},\mathbf{w},\mathbf{v}))$$
for any $\mathbf{u},\mathbf{v},\mathbf{w} \in H^1(\Omega)^d$ (here, the usual summation convention is used).

The following properties of V and H are well known:

$$V = \{\mathbf{v} \in H_0^1(\Omega)^d : \nabla\cdot\mathbf{v} = 0 \text{ in } \Omega\},$$
$$H = \{\mathbf{v} \in L^2(\Omega)^d : \nabla\cdot\mathbf{v} = 0 \text{ in } \Omega, \quad \mathbf{v}\cdot\mathbf{n} = 0 \text{ on } \Gamma\},$$
$$V \hookrightarrow H \hookrightarrow V', \quad \text{where the embeddings are dense and compact.}$$

We can now give a rigorous formulation of the unsteady Navier-Stokes problem in $\Omega\times(0,T)$:

$$\begin{cases} \mathbf{u} \in L^2(0,T;V)\cap L^\infty(0,T;H), \\ \langle \dfrac{\partial\mathbf{u}}{\partial t}(t),\mathbf{v}\rangle + \nu((\mathbf{u}(t),\mathbf{v})) + b(\mathbf{u}(t),\mathbf{u}(t),\mathbf{v}) = \langle\mathbf{f}(t),\mathbf{v}\rangle \\ \quad \text{a.e. in } (0,T), \quad \forall\mathbf{v}\in V, \\ \mathbf{u}(0) = \mathbf{u}_0\,. \end{cases} \tag{2}$$

In (2), $\mathbf{u}_0 \in H$ and $\mathbf{f} \in L^2(0,T;L^2(\Omega)^d)$. It is well known that (2) possesses at least one solution which is furthermore unique if $d = 2$. If $\mathbf{u}$ is a solution, then $\mathbf{u}$ solves, together with some scalar distribution p, the Navier-Stokes equations (1) (for instance, see [18]). One also has

$$\mathbf{u} = 0 \quad \text{on} \quad \partial\Omega\times(0,T)$$

and $\mathbf{u}|_{t=0} = \mathbf{u}_0$ in an appropriate sense.

Notice that

$$b(\mathbf{u},\mathbf{v},\mathbf{w}) = \hat{b}(\mathbf{u},\mathbf{v},\mathbf{w}), \quad \forall\mathbf{u}\in V, \quad \forall\mathbf{v},\mathbf{w}\in H_0^1(\Omega)^d.$$

Also, $\hat{b}(\mathbf{u}, \mathbf{v}, \mathbf{v}) = 0$ for all $\mathbf{v} \in H_0^1(\Omega)^d$ (even when div $\mathbf{u} \neq 0$). Consequently, the variational evolution equation in (2) can also be written in terms of $\hat{b}(\cdot\,, \cdot\,, \cdot)$ and this gives the following equivalent formulation:

$$\begin{cases} \mathbf{u} \in L^2(0, T; V) \cap L^\infty(0, T; H), \\ \langle \dfrac{\partial \mathbf{u}}{\partial t}(t), \mathbf{v} \rangle + \nu((\mathbf{u}(t), \mathbf{v})) + \hat{b}(\mathbf{u}(t), \mathbf{u}(t), \mathbf{v}) = \langle \mathbf{f}(t), \mathbf{v} \rangle, \\ \quad \text{a.e. in } (0, T), \quad \forall \mathbf{v} \in V, \\ \mathbf{u}(0) = \mathbf{u}_0\,. \end{cases} \tag{3}$$

We are now going to indicate how to approximate in time. Let us divide the interval $[0, T]$ in M subintervals of length k $(k = T/M)$ and let us assume that the parameters $\sigma \in (0, 1]$, $\theta, \mu \in [0, 1]$ and $a, b > 0$ are given.

We first put

$$\mathbf{u}^0 = \mathbf{u}_0\,. \tag{4}$$

Then, for given $m \geq 0$ and $\mathbf{u}^m \in H_0^1(\Omega)^d$ (an approximation of $\mathbf{u}$ at time $t_m = mk$), we compute $\mathbf{u}^{m+a}$, $\mathbf{u}^{m+b}$ and then $\mathbf{u}^{m+1}$ as follows. We first solve in parallel the elliptic systems

PROBLEM (BP) (Burgers)

$$\begin{cases} u^{m+a} \in H_0^1(\Omega)^d, \\ \dfrac{1}{ak}(\mathbf{u}^{m+a} - \mathbf{u}^m, \mathbf{w}) + \nu((\sigma \mathbf{u}^{m+a} + (1-\sigma)\mathbf{u}^m, \mathbf{w})) \\ \quad + \dfrac{2\theta}{a}\hat{b}(\mathbf{u}^*, \mathbf{u}^{**}, \mathbf{w}) = \dfrac{2\mu}{a}(\mathbf{f}^{m+a}, \mathbf{w}), \quad \forall \mathbf{w} \in H_0^1(\Omega)^d. \end{cases} \tag{5}$$

and

PROBLEM (SP) (Stokes)

$$\begin{cases} \mathbf{u}^{m+b} \in V, \\ \dfrac{1}{bk}(\mathbf{u}^{m+b} - \mathbf{u}^m, \mathbf{v}) + \nu((\sigma \mathbf{u}^m + (1-\sigma)\mathbf{u}^{m+b}, \mathbf{v})) \\ \quad = \dfrac{2(1-\mu)}{b}(\mathbf{f}^{m+b}, \mathbf{v}) - \dfrac{2(1-\theta)}{b}\hat{b}(\mathbf{u}^m, \mathbf{u}^m, \mathbf{v}), \quad \forall \mathbf{v} \in V. \end{cases} \tag{6}$$

Then, we set

$$\mathbf{u}^{m+1} = \frac{1}{2}(\mathbf{u}^{m+a} + \mathbf{u}^{m+b}). \tag{7}$$

In (5) and (6), $\mathbf{f}^{m+a}$ y $\mathbf{f}^{m+b}$ are appropriate approximations of $\mathbf{f}$. For instance, we can make the following choice:

$$\mathbf{f}^{m+j}(x) = \frac{1}{jk}\int_{mk}^{(m+j)k} \mathbf{f}(x,t)\,dt, \qquad j = a, b.$$

In (5), several different definitions of $\mathbf{u}^*$ and $\mathbf{u}^{**}$ are possible. Thus, it seems natural to put

$$\mathbf{u}^{**} = \alpha\mathbf{u}^{m+a} + (1-\alpha)\mathbf{u}^m$$

for some α. Actually, the choice of $\mathbf{u}^{**}$ is crucial when one tries to establish "a priori" estimates of the numerical solutions. On the other hand, the particular $\mathbf{u}^*$ we use determines the degree of linearity we conserve in (5).

Using more or less standard arguments, we can deduce existence and uniqueness results for (5) and (6), at least when ν is not too small (see for instance [17] and [18]).

In the previous works [5] and [6], we have presented theoretical and numerical results obtained for some parallel schemes of the kind (4)–(7). There, parallelization was performed only at the time approximation level and the stationary problems **(BP)** and **(SP)** were solved with finite element techniques. In this work, we are going to extend the parallelization procedure to all the variables.

To this end, we will apply *simultaneous directions implicit* (SDI) techniques to the previous stationary problems. Thus, let us denote by W_h a finite dimensional Hilbert space determined by a second-order finite difference approximation of $H_0^1(\Omega)^d$ (h is a parameter that allows to identify the mesh; of course, we pass from the finite-dimensional to the infinite-dimensional problems by letting $h \to 0$). Let $V_h \in W_h$ be the subspace formed by the functions in W_h with vanishing *discrete divergence* (see [21] for several possible W_h and V_h and the associated definitions of the discrete divergence).

Then, the spatial approximation of (4)–(7) is the following:

First, $\mathbf{u}_h^0$ is the orthogonal projection of $\mathbf{u}_0$ on W_h for the L^2 scalar product, i.e.

$$(\mathbf{u}_h^0, \mathbf{v}_h) = (\mathbf{u}_0, \mathbf{v}_h), \quad \forall \mathbf{v}_h \in W_h\,, \quad \mathbf{u}_h^0 \in W_h\,. \tag{8}$$

Then, for any given $m \geq 0$ and $\mathbf{u}_h^m \in W_h$, we compute $\mathbf{u}_h^{m+a}$, $\mathbf{u}_h^{m+b}$ and then $\mathbf{u}_h^{m+1}$ as follows. We first solve in parallel two independent problems:

$$\left\{\begin{aligned} &\frac{1}{ak}(\mathbf{u}_h^{m+a} - \mathbf{u}_h^m, \mathbf{w}_h) + \nu((\sigma \mathbf{u}_h^{m+a} + (1-\sigma)\mathbf{u}_h^m, \mathbf{w}_h))_h \\ &\quad + \frac{2\theta}{a}\hat{b}_h(\mathbf{u}_h^*, \mathbf{u}_h^{**}, \mathbf{w}_h) = \frac{2\mu}{a}(\mathbf{f}^{m+a}, \mathbf{w}_h), \quad \forall \mathbf{w}_h \in W_h\,. \end{aligned}\right. \tag{9}$$

$$\left\{\begin{aligned} &\frac{1}{bk}(\mathbf{u}_h^{m+b} - \mathbf{u}_h^m, \mathbf{v}_h) + \nu((\sigma \mathbf{u}_h^{m} + (1-\sigma)\mathbf{u}_h^{m+b}, \mathbf{v}_h))_h \\ &= \frac{2(1-\mu)}{b}(\mathbf{f}^{m+b}, \mathbf{v}_h) - \frac{2(1-\theta)}{b}\hat{b}(\mathbf{u}_h^m, \mathbf{u}_h^m, \mathbf{v}_h), \quad \forall \mathbf{v}_h \in V_h\,. \end{aligned}\right. \tag{10}$$

Then, we put

$$\mathbf{u}_h^{m+1} = \frac{1}{2}(\mathbf{u}_h^{m+a} + \mathbf{u}_h^{m+b}). \tag{11}$$

Again, we have several possibilities for the choice of $\mathbf{u}_h^*$ and $\mathbf{u}_h^{**}$. For instance, we can take

$$\mathbf{u}_h^{**} = \sigma \mathbf{u}_h^{m+a} + (1-\sigma)\mathbf{u}_h^m\,,$$

$\mathbf{u}_h^* = \mathbf{u}_h^m$ or $\mathbf{u}_h^* = \mathbf{u}_h^{m+a}$, etc.

The existence and uniqueness of a solution $\mathbf{u}_h^{m+b}$ to (10) is an immediate consequence of Lax–Milgram's lemma. The existence of a solution $\mathbf{u}_h^{m+a}$ to (9) is easily implied by Brouwer's fixed point theorem (see for instance [18]).

In [6] and [8], we have deduced convergence and stability results for the completely discretized scheme (in time and space). In some particular situations, we have also deduced error estimates, cf. [2] and [1].

3 A convergence result

We recall in this section a convergence-stability result for the previous numerical method. In the sequel, C denotes a generic positive constant only depending on the data Ω, T, ν, $\mathbf{u}_0$ and $\mathbf{f}$ and, possibly, the parameters σ, θ and μ.

There exist *optimal* quantities d_0, $S(h)$ and $S_1(h)$ such that

$$\frac{1}{d_0}|\mathbf{w}_h| \leq \|\mathbf{w}_h\| \leq S(h)|\mathbf{w}_h|, \qquad \forall \mathbf{w}_h \in W_h \tag{12}$$

and

$$|\hat{b}_h(\mathbf{v}_h, \mathbf{v}_h, \mathbf{w}_h)| \leq S_1(h)|\mathbf{v}_h|^2\|\mathbf{w}_h\|, \qquad \forall \mathbf{v}_h, \mathbf{w}_h \in W_h\,. \tag{13}$$

More precisely, we have $d_0 = 2\ell$ where ℓ is the smallest size of Ω in the directions $x_1, \ldots, x_d$,

$$S(h) = 2\left(\sum_{i=1}^{d} h_i^{-2}\right)^{1/2} \quad \text{and} \quad S_1(h) = \begin{cases} 3\sqrt{2}\,S(h) & \text{if } d = 2 \\ 2\cdot 3^{3/2}\,S^{3/2}(h) & \text{if } d = 3 \end{cases} \tag{14}$$

(see [21]).

For each time step k and each $h > 0$, we introduce the functions u_{kh}, v_{kh}, w_{kh}, z_{kh}, $\tilde{u}_{kh}$, $\tilde{v}_{kh}$ and $\tilde{w}_{kh}$, given as follows:

$$\begin{cases} u_{kh}, v_{kh}, w_{kh}, z_{kh} : [0,T] \mapsto W_h \quad \text{are piecewise constant, with} \\ u_{kh}(t) = u_h^m\,,\ v_{kh}(t) = u_h^{m+a}\,,\ w_{kh}(t) = u_h^{m+b}, \end{cases}$$

$$\begin{cases} \tilde{u}_{kh}, \tilde{v}_{kh}, \tilde{w}_{kh} : [0,T] \mapsto W_h \quad \text{are continuous and piecewise linear, with} \\ \tilde{u}_{kh}(mk) = u_h^m\,,\ \tilde{v}_{kh}(mk) = u_h^{m+a}\,,\ \tilde{w}_{kh}(mk) = u_h^{m+b}\,. \end{cases}$$

Theorem 1. *Assume that $\sigma \in (\frac{1}{2}, 1]$, $a + b = 2$, $u_h^{**} = \sigma u_h^{m+a} + (1-\sigma)u_h^{m+b}$ and, for instance, $u_h^* = u_h^m$ in* (9). *There exist constants K_0 and K_1, only depending on Ω, $|u_0|$, $\|f\|_{L^2(0,T;H)}$, ν and σ, such that, whenever*

$$kS(h)^2 \leq K_0, \quad kS_1(h)^2 \leq K_1, \tag{15}$$

we have:

1. *There exist subsequences $u_{k'h'}, \ldots, \tilde{w}_{k'h'}$ that converge strongly in the space $L^2(0,T; L^2(\Omega)^d)$, weakly in $L^2(0,T; H_0^1(\Omega)^d)$ and also weakly-$*$ in $L^\infty(0,T; L^2(\Omega)^d)$ to the same function u.*

2. *The limit of any such subsequence is a solution of* (2).

3. *Consequently, when* $d = 2$*, the whole sequences* $u_{kh}, \dots, \tilde{w}_{kh}$ *converge (in the above sense) to the unique solution of* (2).

4. *Finally, if* $d = 2$ *and* k *and* h *satisfy*

$$kS(h)^2 \to 0, \quad kS_1(h)^2 \to 0, \tag{16}$$

we also have strong convergence of the whole sequences in the space $L^2(0, T; H_0^1(\Omega)^d)$.

The proof of this result is given in [2]. Notice that (15) can be viewed as a *stability condition*. It means in practice that, for any small h, k cannot be too large.

Remark 1. From the proof of this result we see that, in order to get stability with restrictions as weak as possible, it seems preferable to take $\theta = 1$, which is equivalent to leave the whole nonlinear term in **(BP)**. On the other hand, as expectable, we see that the choice of μ has no influence on (15). A more detailed analysis shows that the best parameters conserving stability and low computational cost are those satisfying

$$\frac{2}{17} < a < \frac{2}{5}; \quad \frac{k}{h_0^2} < \frac{17}{2^9 \nu} \text{ if } d = 2; \quad \frac{k}{h_0^2} < \frac{17}{3 \cdot 2^8 \nu} \text{ if } d = 3 \tag{17}$$

(see [9]), where $h_0 = \min_{1 \le i \le d} h_i$. A remarkable fact is that, for small ν (the most interesting situation from the realistic viewpoint), the stability requirements (17) become weak.

4 The numerical solution in practice

After time discretization, we must solve independent stationary problems of two kinds:

- Burgers-like problems **(BP)** that can be linear or not, depending on the definition of $\mathbf{u}^*$.
- Generalized (linear) Stokes problems **(SP)**.

We are now going to indicate the way these problems are solved in practice. The main idea is to reduce the task to the solution of a family of Poisson problems. Then, as already said, we will apply SDI techniques. As a result, we will only find (many) independent 1D differential problems.

4.1 The numerical solution of Burgers problems

The goal is to solve numerically a system of the kind

$$\begin{cases} \alpha\mathbf{u} - \mu_1\Delta\mathbf{u} + 2\theta(\mathbf{u}^*\cdot\nabla)(\sigma\mathbf{u} + (1-\sigma)\mathbf{v}) & \\ \qquad\qquad +\theta(\mathrm{div}\mathbf{u}^*)((1-\sigma)\mathbf{v} + \sigma\mathbf{u}) = \mathbf{F} & \text{in } \Omega, \\ \mathbf{u} = \mathbf{0} & \text{on } \Gamma, \end{cases} \tag{18}$$

where

$$\begin{aligned} \mathbf{u} &= \mathbf{u}^{m+a}, \qquad \mathbf{v} = \mathbf{u}^m, \\ \mu_1 &= \nu a\sigma\ , \qquad \alpha = 1/k, \\ \mathbf{F} &= \mathbf{F}(\mathbf{v}) = 2\mu\mathbf{f}^{m+a} + \nu a(1-\sigma)\Delta\mathbf{v} + \alpha\mathbf{v} - \nabla p^m. \end{aligned}$$

When $\mathbf{u}^* = \mathbf{v}$, we find a linear elliptic system. Contrarily, when $\mathbf{u}^* = \mathbf{u}$, (18) is nonlinear. In both cases, (18) is solved applying an iterative fixed point algorithm leading to standard Poisson equations completed with Dirichlet conditions.

To this end, let us rewrite (18) in the form

$$\begin{cases} \alpha\mathbf{u} - \mu_1\Delta\mathbf{u} = \mathbf{G}(\mathbf{u},\mathbf{v}) & \text{in } \Omega, \\ \mathbf{u} = \mathbf{0} & \text{on } \Gamma, \end{cases} \tag{19}$$

with

$$\mathbf{G}(\mathbf{u},\mathbf{v}) = \mathbf{F} - 2\theta(\mathbf{u}^*\cdot\nabla)(\sigma\mathbf{u} + (1-\sigma)\mathbf{v}) - \theta(\mathrm{div}\mathbf{u}^*)((1-\sigma)\mathbf{v} + \sigma\mathbf{u}).$$

We first take $\mathbf{u}^0 = \mathbf{v}$ (in practice, the velocity field we know from the previous time step). Then, for any $n \geq 0$, we compute the solution $\mathbf{u}^{n+1}$ to the linear system

$$\begin{cases} \alpha\mathbf{u}^{n+1} - \mu_1\Delta\mathbf{u}^{n+1} = \mathbf{G}(\mathbf{u}^n,\mathbf{v}) & \text{in } \Omega, \\ \mathbf{u}^{n+1} = \mathbf{0} & \text{on } \Gamma \end{cases} \tag{20}$$

and we iterate until the desired precision is reached.

Observe that (20) is a set of d independent scalar Poisson-Dirichlet problems, for which the unknowns are u_i for $i = 1, \dots, d$. Consequently, they can be solved in parallel. To each of these problems, SDI techniques will be applied (see subsection 4.3). Thus, we see that at least conceptually (18) reduces to a family of 1D differential problems, many of them independent, all them leading to similar numerical difficulties.

When (18) is nonlinear and the considered Reynolds number is large, the previous fixed point argument does not suffice. In that case, more sophisticated methods are required based for instance on least square reformulations and Newton-like or conjugate gradient algorithms (see [14] for a complete analysis). However, for the low or moderate Reynolds numbers considered in this paper, it is sufficient to argue as before. In fact, the numerical experiments show that good convergence is attained after very few iterates (see [8] for more details).

4.2 The solution of the generalized Stokes problems

Now, we deal with the linear problem

$$\begin{cases} \alpha\mathbf{u} - \mu_2\Delta\mathbf{u} + \nabla p^* = \mathbf{F} & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega, \\ \mathbf{u} = \mathbf{0} & \text{on } \Gamma, \\ \displaystyle\int_\Omega p^*\,dx = 0, & \end{cases} \tag{21}$$

where

$$\begin{aligned} \mathbf{u} &= \mathbf{u}^{m+b}, \\ \mu_2 &= b\nu(1-\sigma), \qquad \alpha = 1/k, \\ \mathbf{F} &= 2(1-\mu)\mathbf{f}^{m+b} + b\nu\sigma\Delta\mathbf{u^m} - 2(1-\theta)(\mathbf{u^m}\cdot\nabla)\mathbf{u^m} \\ &= -(1-\theta)(\mathrm{divu^m})\mathbf{u^m} + \alpha\mathbf{u^m}. \end{aligned}$$

This generalized Stokes problem has been solved using a conjugate gradient algorithm adapted from the methods in [7]. Its complete description has been given in [9].

This procedure reduces (again) the task to the solution of Poisson-Dirichlet problems of the kind (22) (see below). In practice, in order to improve its behavior, we have to incorporate preconditioners. For more details, see [7] and [8].

Remark 2. At each time step, when we solve **(SP)**, we get in particular a numerical approximation to the pressure $p(t_{m+1})$.

4.3 The solution to the Poisson-Dirichlet problems with SDI methods

We include here a brief description of the parallel method originally proposed in [19] and later used and analyzed in [8] and [11] to solve the previous Poisson-Dirichlet problems.

Thus, let us consider the system

$$\begin{cases} Lu = -\Delta u + \alpha u = f(x) & \text{in } \Omega, \\ u = h(x) & \text{on } \Gamma, \end{cases} \tag{22}$$

where f and h are given. Let us write L in the form $L = L_1 + \cdots + L_d$, where

$$L_n = -\frac{\partial^2}{\partial x_n^2} + \frac{\alpha}{d} I \quad \text{for } 1 \leq n \leq d.$$

We start with an arbitrary U^0 satisfying

$$U^0 = h(x) \quad \text{on } \Gamma,$$

Then, for any $m \geq 0$, U^{m+1} is found from U^m as follows. First, we compute $U^{m+1,1}, \ldots, U^{m+1,d}$ by solving in parallel the "one-dimensional" problems

$$(I + \tau L_n)\, U^{m+1,n} = (I - \tau \sum_{j \neq n} L_j) U^m + \tau f, \tag{23}$$

completed with appropriate boundary conditions deduced from (22). Secondly, we set

$$U^{m+1} = \frac{\omega}{d} \sum_{n=1}^{N} U^{m+1,n} + (1 - \omega) U^m. \tag{24}$$

Here, τ and ω (that can depend on m) are parameters that must be determined in order to improve convergence properties. For instance, when $d = 2$, the iterative method (23)–(24) leads to the following problems:

$$\left(I + \tau \left(\frac{\alpha}{2} I - \frac{\partial^2}{\partial x_1^2}\right)\right) U^{m+1,1} = \left(I - \tau \left(\frac{\alpha}{2} I - \frac{\partial^2}{\partial x_2^2}\right)\right) U^m + \tau f, \tag{25}$$

$$\left(I+\tau\left(\frac{\alpha}{2}I-\frac{\partial^2}{\partial x_2^2}\right)\right)U^{m+1,2}=\left(I-\tau\left(\frac{\alpha}{2}I-\frac{\partial^2}{\partial x_1^2}\right)\right)U^{m}+\tau f, \qquad (26)$$

completed with Dirichlet (two-point) boundary conditions.

We see that, for each fixed x_2, the unknown function

$$U^{m+1,1}(\cdot, x_2)$$

solves in (25) an ordinary differential equation (in (25), x_2 is a parameter). In practice, it seems reasonable to fix a finite set of x_2 values and, then, solve numerically the corresponding problems (25). In this way, we will obtain approximations to the values of the unknown $U^{m+1,1}$ in a finite set of grid points (x_1^ℓ, x_2^j).

Of course, similar things can be said for (26). Since both unknowns $U^{m+1,1}$ and $U^{m+1,2}$ have to be used in (23) for the computation of U^{m+1}, it is desirable to use the same coordinates x_1^ℓ and x_2^j in (25) and (26). Accordingly, we are led to use rectangular grids.

It is important to emphasize that the level of difficulty is not increased in the case $d = 3$ since, at the end, the task is reduced to the numerical solution of (many) one-dimensional problems like (25) and (26).

Several slight generalizations of (23)–(24) and a convergence analysis have been given in [11].

5 The behavior of the SDI method

We are now going to illustrate the behavior of the SDI method when it is applied to the following test problems:

- The two-dimensional problem (22) in $\Omega = (0,1)\times(0,1)$ with $\alpha = 0$, $\beta = 1$ and $f \equiv 2$. The exact solution is

$$u = \sinh(\pi x_1)\ \sin(\pi x_2) + x_1(1-x_1).$$

- A similar three-dimensional problem in $\Omega = (0,1)\times(0,1)\times(0,1)$, with $\alpha = 0$, $\beta = 1$ and $f \equiv 4$. Now, the exact solution is

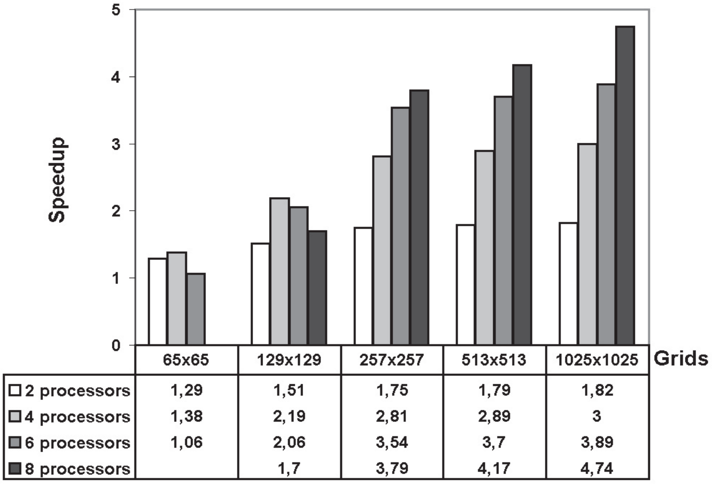
$$u = \sinh(\pi x_1)\ \sin(\pi x_2) + \sinh(\pi x_1)\ \sin(\pi x_3) + x_1(1-x_1) + x_3(1-x_3)\,.$$

Of course, in both cases the imposed boundary conditions are those fitted by u.

The previous algorithm has been implemented in a SGI Origin 2000 computer with 8 processors, using the parallel computing model of OpenMP. In order to measure the performance of the parallel algorithm, let us introduce two parameters: the **speed-up** S_p, defined by

$$S_p = \frac{\text{Resolution time with 1 processor}}{\text{Resolution time with } p \text{ processors}},$$

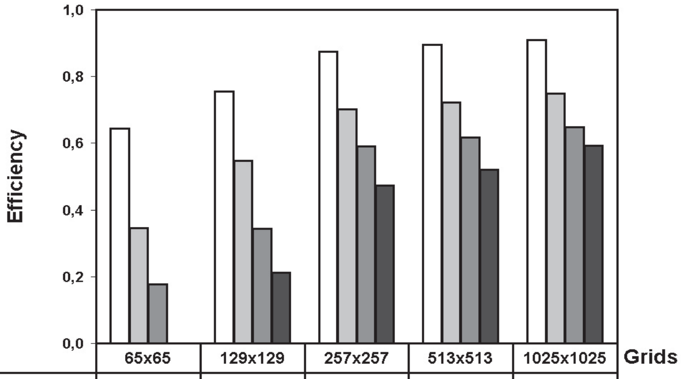
and the **efficiency** $\eta = S_p/p$. We have obtained results for different meshes and for 2, 4, 6 and 8 processors. In Fig. 1–4, the associated speed-up's and efficiencies are shown.



	65x65	129x129	257x257	513x513	1025x1025
□ 2 processors	1,29	1,51	1,75	1,79	1,82
▒ 4 processors	1,38	2,19	2,81	2,89	3
▓ 6 processors	1,06	2,06	3,54	3,7	3,89
■ 8 processors		1,7	3,79	4,17	4,74

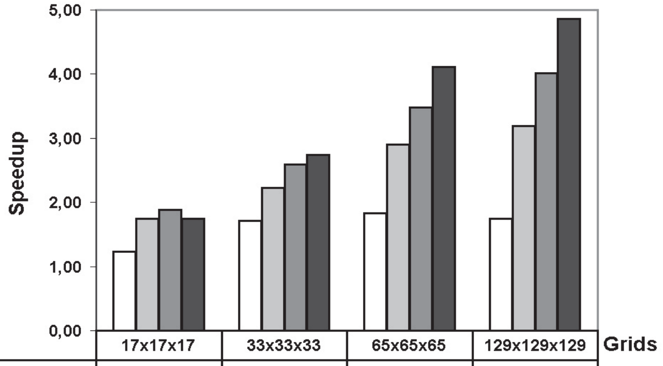
Figure 1 – The speed-up (2D test).

The observed behavior is very similar in dimensions 2 and 3. For coarse grids, parallelization does not improve the speed-up. In fact, in the case of a 65×65 mesh, the results when using 8 processors are worse than those provided by a sequential method. This can be justified because, in this case, the cost of initializing the processors is, probably, greater than the benefit (the computational work of each processor is too small). On the contrary, when the number of nodes



	65x65	129x129	257x257	513x513	1025x1025
□ 2 processors	0,65	0,76	0,88	0,90	0,91
▨ 4 processors	0,35	0,55	0,70	0,72	0,75
▩ 6 processors	0,18	0,34	0,59	0,62	0,65
■ 8 processors		0,21	0,47	0,52	0,59

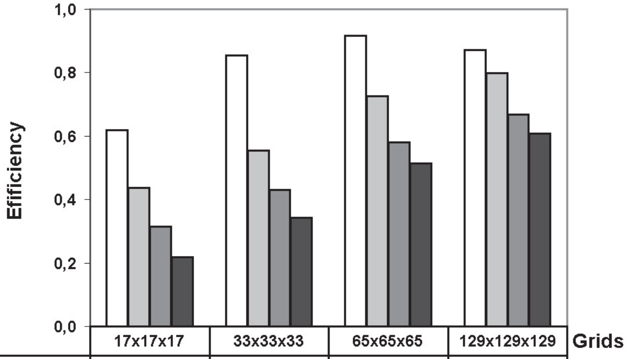
Figure 2 – The efficiency (2D test).



	17x17x17	33x33x33	65x65x65	129x129x129
□ 2 processors	1,24	1,71	1,83	1,74
▨ 4 processors	1,74	2,22	2,90	3,19
▩ 6 processors	1,89	2,59	3,48	4,01
■ 8 processors	1,74	2,74	4,11	4,86

Figure 3 – The speed-up (3D test).

is high, the speed-up and the efficiency increase, and we obtain an efficiency of 0.6 for 8 processors, that we think reasonable. The results are analogous for other tests in non rectangular domains. For more details, see [8].



	17x17x17	33x33x33	65x65x65	129x129x129
□ 2 processors	0,62	0,85	0,92	0,87
▨ 4 processors	0,44	0,56	0,73	0,80
▨ 6 processors	0,31	0,43	0,58	0,67
■ 8 processors	0,22	0,34	0,51	0,61

Figure 4 – The efficiency (3D test).

6 Two-dimensional numerical experiments

We are now going to present some numerical results concerning the Navier-Stokes problem (1). We begin with the so called *2D no-flow test*. This is related to the motion of a viscous 2D fluid under the action of gravitational forces of the form $\mathbf{f} = (0, -1)$. The considered domain Ω and the pressure isolines are displayed in Fig. 5.

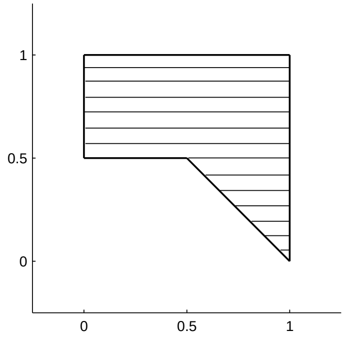


Figure 5 – The isobars for the 2D no flow test.

The spatial approximation has been determined by a regular mesh of step

$h = 0.05$. The time step has been $k = 0.01$. The following parameters were chosen: $\sigma = 0.51$, $\theta = 0.5$ and $a = 0.5$. We have started the computations from zero velocity field and pressure at time $t = 0$.

The computed isobars, in accordance with the theoretical prediction, are horizontal straight lines. The exact error for the pressure (in norm L^∞) is less than 3.4×10^{-7}. The L^∞ norm of the computed velocity is less than 3.28×10^{-11}.

We consider now a squared cavity of unit side filled by a fluid. We assume that the upper wall slips with constant velocity and we try to determine which is the effect of this on the fluid (see [13]).

The results we obtain for various different parameters θ at time $T = 10$ have been presented in Fig. 6 and 7. This test has been performed for a Reynolds number $Re = 1000$, with spatial mesh size $h = 0.01$ and time approximation step $k = 0.01$. We have taken $a = 0.5$ and $\sigma = 0.51$.

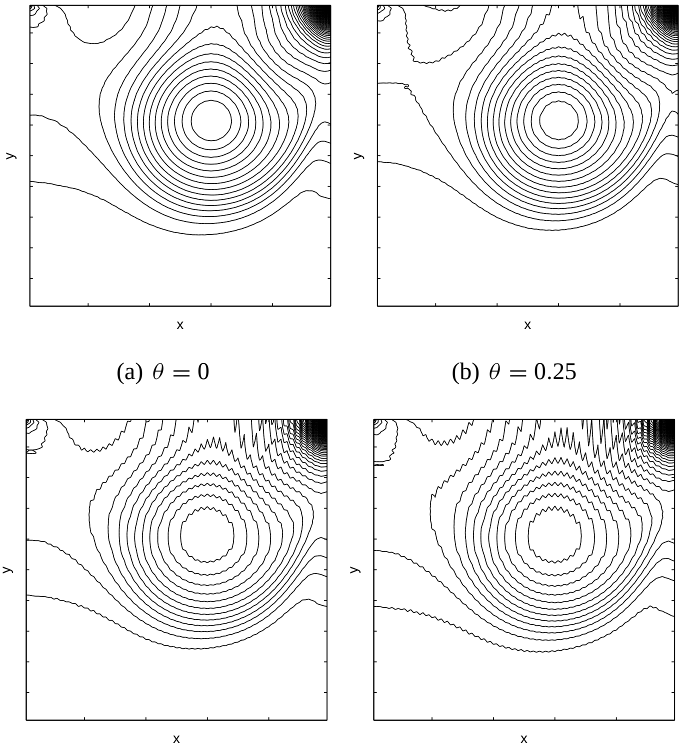
It can be observed that, as θ increases, the occurrence of spurious pressures becomes more important. A detailed analysis of the reasons that lead to this phenomenon has been presented in [8] and [10]. There, a discrete filtering operator that eliminates these undesirable fluctuations has been introduced.

The numerical results corresponding to the particular case $\theta = 1$ without and with filtering have been displayed in Fig. 7(a) and Fig. 7(b), respectively.

In order to make a comparison with the results furnished by Ghia [13], Botella and Peyret [4] and Griebel [20], we have fixed a regular spatial approximation with 129 points in each direction. We have taken again $k = 0.01$, $Re = 1000$ and a final time $T = 30$. The computed streamlines and isobars are presented in Fig. 8.

Some computed values of the velocity field and the pressure, together with the deviations of these results from those in [13], [4] and [20] are also given in [9]. We can observe that the proposed algorithm leads to numerical solutions whose behavior is correct.

In a similar way, we have displayed in Fig. 9 the streamlines and isobars corresponding computed for this test for $Re = 4000$ at time $T = 50$, using a spatial mesh size $h = 0.01$ and a time approximation step $k = 0.005$. Of course, in this case we have had to take a smaller k in order to ensure numerical stability of the Burgers problems.



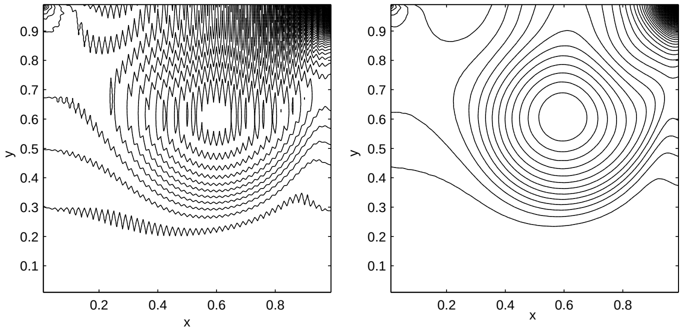
(a) $\theta = 0$

(b) $\theta = 0.25$

(c) $\theta = 0.5$

(d) $\theta = 0.75$

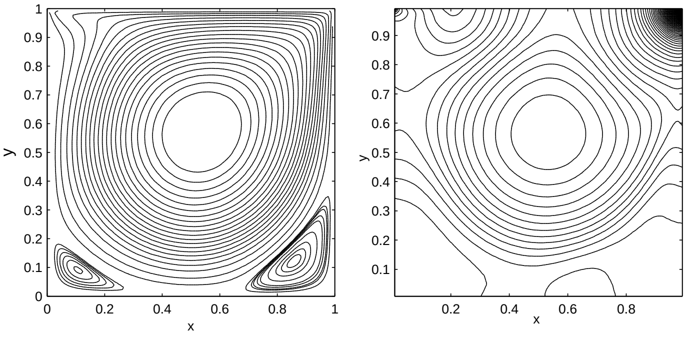
Figure 6 – The isobars for the 2D cavity test with $Re = 1000$, $T = 10$.



(a) Without pressure filtering

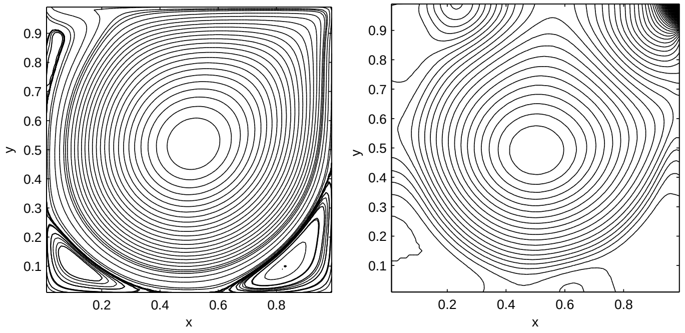
(b) With pressure filtering

Figure 7 – The 2D cavity test with $Re = 1000$, $T = 10$ and $\theta = 1$.



(a) The streamlines. (b) The isobars.

Figure 8 – The 2D cavity test with $Re = 1000$.



(a) The streamlines. (b) The isobars.

Figure 9 – The 2D cavity test with $Re = 4000$, $T = 50$, $h = 0.01$ and $k = 0.005$.

7 Three-dimensional numerical experiments

We have also considered the no-flow and the cavity tests for a 3D fluid.

For the no-flow test, the results obtained at time $T = 3$ for $h = 0.05$ and $k = 0.01$ are presented in Fig. 10. We have used here the following parameter values: $\theta = 0.5$, $\sigma = 0.51$ and $a = 0.5$. Again, the algorithm has started from a vanishing velocity field and a vanishing pressure. We have displayed the surfaces $P =$ const. noticing that the exact error, in norm L^{∞}, is less than 10^{-5}.

For the 3D cavity test, we have compared our results to those furnished by [16]. For instance, the results obtained for Re $= 1000$ have been presented in Fig. 11–12, respectively. Good agreement is found.

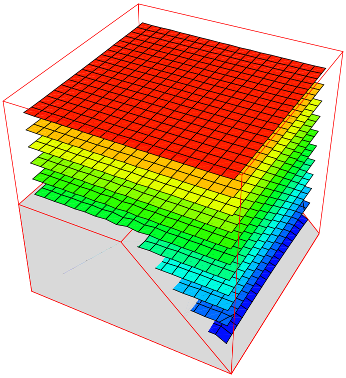
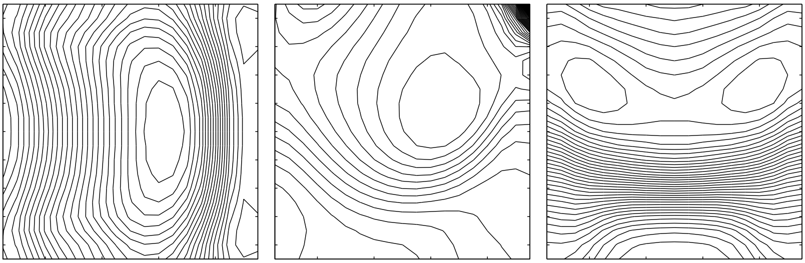


Figure 10 – The 3D no-flow test. Numerical results with $k = 0.01$ and $h = 0.05$.



(a) $z = 0.5$ (b) $y = 0.5$ (c) $x = 0.5$

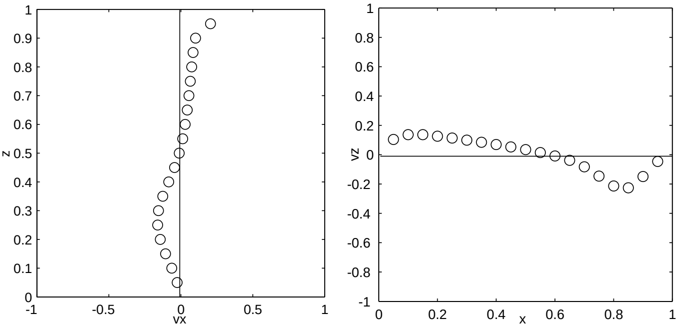
Figure 11 – The isobars for the 3D cavity test with $\mathrm{Re} = 1000$ and $T = 10$.

8 Final comments

The proposed parallel algorithm leads to satisfactory numerical results for both 2D and 3D Navier-Stokes fluids with small or moderate Reynolds numbers. The theoretical results have been confirmed by a set of numerical experiments.

Although we have only reported here Reynolds numbers up to 4000 in 2D tests and up to 1000 in 3D tests, other experiments reveal that, with an appropriate choice of the parameters, we can obtain realistic results for larger Re.

The algorithm we have used is very flexible can be easily adapted to many particular flows. Another positive fact is the incorporation of SDI methods for the solution of Poisson problems, which can be combined with multigrid



(a) V_x at $x = 0.5$.

(b) V_z at $z = 0.5$.

Figure 12 – The velocity profiles for the 3D cavity test with $\mathrm{Re} = 1000$ and $T = 10$ at $y = 0.5$.

techniques in view of their smoother properties (see [12]).

Let us finally mention that other boundary conditions can be considered. Indeed, SDI methods can be adapted to the numerical solution of Poisson equations with Neumann and Robin boundary conditions (see [8] and [3]).

REFERENCES

[1] I.I. Albarreal. *PhD Thesis.* University of Sevilla, Spain (2004).

[2] I.I. Albarreal, M.C. Calzada, J.L. Cruz, E. Fernández-Cara, J.R. Galo and M. Marín. *Convergence analysis and error estimates for a parallel algorithm for solving the Navier–Stokes equations*, Numer. Math., **93** (2002), 201–221.

[3] I.I. Albarreal, M.C. Calzada, J.L. Cruz, E. Fernández-Cara, J.R. Galo and M. Marín. *A new method for the numerical solution of elliptic and parabolic problems with Neumann boundary conditions*, to appear.

[4] O. Botella and R. Peyret. *Benchmark spectral results on the lid-driven cavity flow*, Computers & Fluids, **27** (1998), 421–433.

[5] J.L. Cruz, M.C. Calzada, E. Fernández-Cara and M. Marín. *A parallel algorithm for solving the incompressible Navier-Stokes problem* Comp. Math. with Appl., **25**(9): (1993), 51–58.

[6] J.L. Cruz, M.C. Calzada, E. Fernández-Cara and M. Marín. *A convergence result for a parallel algorithm for solving the Navier-Stokes equations*, Comp. Math. with Appl., **35**(4): (1998), 71–88.

[7] E. Dean and R. Glowinski. *On some finite element methods for the numerical simulation of incompressible viscous flow*, in "Incompressible computational fluid dynamics", M.D. Gun-

zburger and Nicolaides RA eds., Cambridge U.P. (USA), (1993), 17–65.

[8] J.R. Galo. *PhD Thesis*. University of Sevilla, Spain (2002).

[9] J.R. Galo, I.I. Albarreal, M.C. Calzada, J.L. Cruz, E. Fernández-Cara and M. Marín. *A parallel algorithm in time and space for the Navier–Stokes equations*, to appear.

[10] J.R. Galo, I.I. Albarreal, M.C. Calzada, J.L. Cruz, E. Fernández-Cara and M. Marín. *Analysis of a finite difference regularizing scheme with the same grid for the velocity and the pressure*, to appear.

[11] J.R. Galo, I.I. Albarreal, M.C. Calzada, J.L. Cruz, E. Fernández-Cara and M. Marín. *The convergence of the parallel SDI method for elliptic problems*, to appear.

[12] J.R. Galo, M.C. Calzada, J.L. Cruz, M. Marín, I.I. Albarreal and E. Fernández-Cara. *The smoothing effect of a simultaneous directions parallel method as applied to Poisson problems*, to appear in Numer. Methods for Differ. Eqn.

[13] U. Ghia, K.N. Ghia and C.T. Shin. *High-Resolutions for incompressible flow using the Navier-Stokes equations and a multigrid method*, J. Comp. Physics, **48** (1982), 387–411.

[14] R. Glowinski. *Numerical Methods for Nonlinear Variational Problems* (2nd edition), Springer-Verlag, New York (1984).

[15] R. Glowinski. *Finite element methods for incompressible viscous flow*. Handbook of numerical analysis, Vol. IX, 3–1176, North-Holland, Amsterdam (2003).

[16] K. Kakuda and H. Kalo. *Petrov–Galerkin finite element approach using exponential functions for three–dimensional incompressible viscous flow problems*, Finite Element in Fluids (1993), 204–213.

[17] O.A. Ladyzhenskaya. *The Mathematical Theory of Viscous Incompressible Flow*. Gordon and Breach, New York (1969).

[18] J.L. Lions. *Quelques Méthodes de Résolution des Problèmes aux Limites non Linéaires*. Dunod, Gauthiers-Villars, Paris (1969).

[19] T. Lu, P. Neittaanmäki and X.C. Tai. *A parallel splitting-up method for partial differential equations and its applications to Navier–Stokes equations*, RAIRO Modél. Math. Anal. Numér. **26**(6): (1992), 673–708.

[20] M. Griebel and F. Koster. *NaSt3DGP*.
http://wwwwissrech.iam.uni-bonn.de/research/projects/koster/NaSt3DGP/index.htm
(20 January 2002).

[21] R. Témam. *Navier-Stokes Equations* (2nd edition), North-Holland, Amsterdam (1984).

[22] S. Turek. *Efficient solvers for incompressible flow problems*. Springer-Verlag, Berlin (1999).