Electronic Transactions on Numerical Analysis.
Volume 34, pp. 136-151, 2009.

ISSN 1068-9613.

ETNA
Kent State University
http://etna.math.kent.edu

IMPLICITLY PRECONDITIONED AND GLOBALIZED RESIDUAL METHOD FOR SOLVING STEADY FLUID FLOWS*

JEAN-PAUL CHEHAB† AND MARCOS RAYDAN‡

Dedicated to Víctor Pereyra on the occasion of his 70th birthday

Abstract. We develop a derivative-free preconditioned residual method for solving nonlinear steady fluid flows. The new scheme is based on a variable implicit preconditioning technique associated with the globalized spectral residual method. The new scheme is robust and allows numerical computation of the steady state of the two-dimensional incompressible Navier-Stokes equations (NSE), which we consider here in both primary variables and streamfunction-vorticity formulations. The results are encouraging and agree with those reported in the literature.

Key words. nonlinear systems of equations, residual methods, globalization strategies, preconditioning, Navier-Stokes equations.

AMS subject classifications. 76D05, 65H10, 76M20, 90C30

1. Introduction.

Preconditioning is a widely-used approach to accelerating numerical methods for solving linear as well as non-linear problems. For linear systems, it is widely developed and very well understood. However, the art of preconditioning iterative methods for nonlinear problems remains a challenge, and it is not so well understood.

The emergence of non-monotone residual methods, such as the one introduced by Barzilai and Borwein in optimization [2, 13, 26] and globalized versions which enhance its robustness and effectiveness [12, 22, 23, 27], gives the possibility of efficiently solving large-scale nonlinear problems, incorporating in a natural way a preconditioning strategy. Non-monotone globalization strategies for nonlinear problems have become popular in the last few years. These strategies make it possible to define globally convergent algorithms without monotone decrease requirements. The main idea behind non-monotone strategies is that, frequently, the first choice of a trial point along the search direction hides significant information about the problem structure, and that such knowledge can be destroyed by imposing a decrease condition.

In this work we adapt and extend the ideas introduced in [12, 22] for large-scale nonlinear problems like those that appear in the solution of the steady fluid flow problem. In particular, we add a preconditioning strategy fully described in [10]. The *lid-driven cavity* problem is a numerically challenging standard benchmark problem for nonlinear solvers that corresponds to computing either evolving or steady flows of the two-dimensional incompressible Navier-Stokes equation on a rectangular cavity. To compute steady states, two approaches are commonly considered: time-dependent methods that compute the steady state by applying a time marching scheme to the evolutive NSE (for Reynolds numbers that are lower than the bifurcation value), and methods that solve the steady-state NSE by fixed point or Newton-like schemes. It is well known that direct solution of the steady-state NSE is more difficult than time marching, and that it requires very robust schemes, especially as the Reynolds number

* Received April 4, 2008. Accepted April 6, 2009. Published online on September 17, 2009. Recommended by José Castillo.

† Laboratoire Amienois de Mathématiques Fondamentales et Appliquées, UMR 6140, UFR de mathématiques et informatique, Université de Picardie Jules Verne, 33 rue Saint Leu, 80037 Amiens, France; INRIA Nord-Europe, project SIMPAF, 59655 Villeneuve d'Ascq, France; and Laboratoire de Mathématiques d'Orsay bat. 425, CNRS 6148, ANEDP, Univ. Paris-Sud Orsay, F-91405 (jean-paul.chehab@u-picardie.fr).

‡ Departamento de Cómputo Científico y Estadísticas, Universidad Simón Bolívar, Ap. 89000, Caracas, 1080-A, Venezuela, and Departamento de Computación, Facultad de Ciencias, Universidad Central de Venezuela, Ap. 47002, Caracas 1041-A, Venezuela (marcos.raydan@ciens.ucv.ve).

Re increases. The literature on this topic is very rich, from, e.g., the relaxation schemes proposed by Crouzeix [11] to the more recent defect-correction methods; see [31] and the references therein. However, these methods are very closely related to the structure of the NSE and use a linearization of the equation at each step.

Our aim in this article is to compute the solution of the steady-state NSE by an implicit preconditioned version of the spectral residual method, with globalization. The method we introduce here uses only the solution of the linear part of the equation, which can be obtained efficiently with a fast solver (e.g., FFT and multigrid).

The article is organized as follows. First, in Section 2, after describing the inverse preconditioning strategy and the globalization strategy for the residual scheme, we derive our new algorithm combining the dynamical and the optimization approach. Then, in Section 3, we adapt the discretization of the steady-state two-dimensional incompressible Navier-Stokes equations to the framework of the nonlinear scheme. Finally, in Section 4, as a numerical illustration, we present the solution of the steady-state NSE for different Reynolds numbers (up to $\mathrm{Re} = 5000$). We solve the problem in the primary variables as well as in stream function-vorticity formulation. Our results agree with the ones in the literature and show the robustness of the proposed method.

2. The basic algorithm and its ingredients. In a general framework, let us consider the nonlinear system of equations

$$F(x) = 0, \tag{2.1}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ is a continuously differentiable mapping. This framework generalizes the nonlinear systems that appear, for example, after discretizing the steady-state models for fluid flows, to be discussed later in this work.

To solve (2.1), some iterative schemes have been recently presented that systematically use the residual vectors as search directions [12, 22]; i.e., the iterations are defined as

$$x_{k+1} = x_k \pm \lambda_k\, F(x_k), \tag{2.2}$$

where $\lambda_k > 0$ is the step-length and the search direction is either $F(x_k)$ or $-F(x_k)$ depending on which one is a descent direction for the merit function

$$f(x) = \|F(x)\|_2^2 = F(x)^T F(x). \tag{2.3}$$

These schemes are effective and competitive with Newton-Krylov schemes for large-scale nonlinear systems [5, 6, 21] when the step lengths are suitably chosen. The convergence of (2.2) is associated with a derivative-free non-monotone line search, fully described in [12], which will be discussed in the forthcoming subsections.

There are many choices of the step length λ_k for which convergence is guaranteed. The well-known non-monotone spectral choice has interesting properties; it is defined as the absolute value of

$$\frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}, \tag{2.4}$$

where $s_{k-1} = x_k - x_{k-1}$, and $y_{k-1} = F(x_k) - F(x_{k-1})$. Obtaining the step length using (2.4) requires a reduced amount of computational work, accelerates the convergence of the process, and involves the last two iterations in a way that incorporates first-order information into the search direction [2, 13, 17, 26].

In the forthcoming subsections, we discuss the preconditioned version of the basic scheme (2.2) and the specific inverse preconditioner to be used in this work, the globalization strategy to guarantee global convergence, and how to obtain the step lengths that accelerate the convergence while improving the stability of the process.

2.1. The preconditioned scheme. In order to present the preconditioned version of (2.2) we extend the ideas discussed in [23] for unconstrained minimization to the solution of (2.1). The well-known and somehow ideal Newton's method for solving (2.1) from an initial guess x_0 can be written as

$$x_{k+1} = x_k - J_k^{-1} F(x_k), \tag{2.5}$$

where $J_k = J(x_k)$ and $J(x)$ is the Jacobian of F evaluated at x.

Recently, a preconditioned scheme, associated to the gradient direction, was proposed to solve unconstrained minimization problems [23]. When solving (2.1), the preconditioned version of (2.2) produces the iterates

$$x_{k+1} = x_k + \lambda_k z_k, \tag{2.6}$$

where $z_k = \pm C_k\ F(x_k)$, C_k is a nonsingular approximation to J_k^{-1}, and the scalar λ_k is given by

$$\lambda_k = (\lambda_{k-1}) \frac{z_{k-1}^T z_{k-1}}{z_{k-1}^T y_{k-1}}. \tag{2.7}$$

In (2.6), if $C_k = I$ (the identity matrix) for all k, then $z_k = \pm F(x_k)$, and since $s_{k-1} = \lambda_{k-1} z_{k-1}$ then λ_k coincides with (2.4). Therefore, if $C_k = I$ the method reduces to (2.2). On the other hand, if the sequence of iterates converges to x^*, and we improve the quality of the preconditioner such that $C(x_k)$ converges to $J^{-1}(x^*)$ then, as discussed in [10], λ_k tends to one and we recover Newton's method, which possesses fast local convergence under standard assumptions [14]. In that sense, the iterative scheme (2.6) is flexible, and allows intermediate options by choosing suitable approximations C_k between the identity matrix and the inverse of the Jacobian matrix. For building suitable approximations to $J^{-1}(x_k)F(x_k)$, we will test implicit preconditioning schemes that do not require the explicit computation of C_k, as described in Section 2.2.

2.2. Inverse preconditioning schemes. By adapting our recent work on approximating the Newton direction using an ordinary differential equation (ODE) model, we will obtain the inverse preconditioner for (2.1) in the framework of the iterative global preconditioned residual algorithm presented in the next subsection. To that end, we develop an automatic and implicit scheme to directly approximate the preconditioned direction z_k at every step in (2.6) without *a priori* knowledge of the Jacobian of F and with only a reduced and controlled amount of storage and computation. As we will discuss later, this new scheme avoids as much as possible the cost of any calculations involving matrices, and also allows us to obtain asymptotically the Newton direction by improving the accuracy in the ODE solver.

The method we introduce here starts from the numerical integration of the Newton flow aimed at computing the root of F as the stable steady state of

$$\frac{dx}{dt} = -(J(x))^{-1} F(x). \tag{2.8}$$

Notice that the preconditioned gradient method can be obtained by applying a forward Euler method to (2.8); see, e.g., [1]. The value $\|F(x)\|$ is decreasing along the integral curves and converges at an exponential rate to zero; for details, see [1, 20] and references therein. Introducing the decoupling

$$\frac{dx}{dt} = -z \tag{2.9}$$

$$J(x) z = F(x), \tag{2.10}$$

we see that the algebraic condition that links z to x is, in fact, a preconditioning equation. In order to relax its resolution, a time derivative in z is added as

$$\frac{dx}{dt} = -z, \tag{2.11}$$

$$\frac{dz}{dt} = F(x) - J(x)z. \tag{2.12}$$

This last system allows us to compute numerically the root of F by an explicit time marching scheme since the steady state is asymptotically stable; see [10] for more details. Let t_k be discrete times, and denote by x_k and z_k the numerical approximations to $x(t_k)$ and $z(t_k)$. The application of the simple forward Euler method to (2.11)–(2.12) reads

$$x_{k+1} = x_k - (t_{k+1} - t_k)z_k, \tag{2.13}$$

$$z_{k+1} = z_k + (t_{k+1} - t_k)\left(F(x_k) - J(x_k)z_k\right). \tag{2.14}$$

REMARK 2.1. As stated above, we want to avoid the computation of the Jacobian matrix, so $J(x)z$ is classically approached by a finite difference scheme

$$J(x)z \simeq \frac{F(x + \tau z) - F(x)}{\tau},$$

for a small given positive real number τ.

Notice that the dynamics of the differential system (2.11)–(2.12) can be very slow. As proposed in [10], a way to speed up the convergence to the steady state is to artifically introduce two time scales by computing for every discrete time t_k an approximation of the steady state of the equation in z. More precisely, we write

Step 1. With optimization method 1, compute z_k as an approximation of the steady state of

$$\frac{dz}{dt} = F(x_k) - \frac{F(x_k + \tau z) - F(x_k)}{\tau},$$
$$z(0) = z_{k-1}.$$

Step 2. With optimization method 2, compute x_{k+1} from x_k by

$$x_{k+1} = x_k + \lambda_k z_k.$$

The preconditioning relies on the accuracy for solving step 1. As optimization method 1, we proposed in [10] to apply some iterations of Cauchy-like schemes that we describe in Section 2.3. As optimization scheme 2, which defines the step λ_k, we use the spectral gradient method. Promising results were obtained in [10] on some classical optimization problems. However, the solution of steady-state NSE needs a more robust scheme for the time marching of x^k. The scheme IPR described below becomes crucial in practical cases. We now present the general form of the scheme.

ALGORITHM 2.2 (Implicit Preconditioned Residual Method (IPR)).
Step 1. With Cauchy-like minimization, compute z_k as an approximation of the steady state of

$$\frac{dz}{dt} = F(x_k) - \frac{F(x_k + \tau z) - F(x_k)}{\tau},$$
$$z(0) = z_{k-1}.$$

Step 2. With GPR, compute x_{k+1} from x_k by

$$x_{k+1} = x_k + \lambda_k z_k.$$

2.3. Generalized Cauchy-type minimization for z_k. We now describe several ways of computing Cauchy-type step lengths for building z_k in Step 1 of the IPR method. The computation of a steady state by an explicit scheme can be accelerated by enhancing the stability domain of the scheme, since this allows the use of larger time steps. In that sense, the accuracy of a time marching scheme is not a priority. A simple way to derive more stable methods is to use parametrized one-step schemes and to fit the parameters not to increase the accuracy as in classical Runge-Kutta methods, but to improve the stability. For example, in [4, 9], a method was proposed to iteratively compute fixed points with larger descent parameter starting from a specific numerical time scheme. More precisely, defining $G(z) = F(x^k) - (F(x^k + \tau z) - F(x^k))/\tau$, this method consists of integrating the differential equation

$$\begin{cases} \dfrac{dy}{dt} = G(y), \\ y(0) = z_{k-1}, \end{cases} \tag{2.15}$$

by the p-step scheme

Given $y_0 = z_{k-1}$
For $j = 0, \ldots$ until convergence
  Set $K_1 = G(y_j)$
  For $m = 2, \ldots, p$
    Set $K_m = G(x_k + \Delta t\, K_{m-1})$
  Set $y_{k+1} = y_j + \Delta t\, \sum_{i=0}^{p} \beta_i K_i$

Here $\sum_{i=1}^{p} \beta_i = 1$. At convergence, we obtain $\bar{y}$, and we set $z_k = \bar{y}$, which will be used as our preconditioned direction at iteration k.

Classically, the convergence can be accelerated by computing at each iteration the step length in order to minimize the Euclidean norm of the current residual. This gives rise to variants of the Cauchy scheme [8]. Of course, the minimizing parameter becomes harder to compute as p increases. We list now the optimal values of the parameters for $p = 1, 2, 3$. The case $p = 3$ represents, as far as we know, a new choice of step length for residual methods. In what follows, for any two vectors v and w, $\langle v, w\rangle = v^T w$, $r^k = F(x_k)$, and

$$Av = \frac{F(x_k + \tau v) - F(x_k)}{\tau} \approx J(x_k)v.$$

- $p = 1$ (Cauchy method)

$$\beta_i^k = 1, \quad \Delta t_k = \frac{\langle Ar^k, r^k\rangle}{\|Ar^k\|^2}.$$

- $p = 2$ (Enhanced Cauchy 1 (EC1) see [9, 10])

We set

$$a = \|r^k\|_2^2, \qquad b = \langle Ar^k, r^k\rangle, \qquad c = \|Ar^k\|_2^2,$$
$$d = \langle A^2r^k, r^k\rangle, \qquad e = \langle A^2r^k, Ar^k\rangle, \qquad f = \langle A^2r^k, A^2r^k\rangle,$$

$$\Delta t_k = \frac{fb - ed}{fc - e^2}, \qquad \beta_1 = 1 - \frac{\Delta t_k e - d}{\Delta t_k^2 f}, \qquad \beta_2 = 1 - \beta_1.$$

- $p = 3$ (Enhanced Cauchy 2 (EC2))
  We set

$$a = \|Ar^k\|_2^2, \qquad b = \|A^2r^k\|_2^2, \qquad c = \|A^3r^k\|_2^2,$$
$$d = \langle Ar^k, r^k\rangle, \qquad e = \langle A^2r^k, r^k\rangle, \qquad f = \langle A^3r^k, r^k\rangle,$$
$$g = \langle A^2r^k, Ar^k\rangle, \qquad \alpha = \langle A^3r^k, Ar^k\rangle, \qquad \gamma = \langle A^3r^k, A^2r^k\rangle,$$

$$\Delta t_k = \frac{-\alpha\gamma e - g\gamma f + \alpha f b + d\gamma^2 - dcb + gce}{g^2c + \alpha^2 b - acb + a\gamma^2 - 2\alpha\gamma g},$$

$$\beta_1 = 1 + \frac{(\Delta t_k)(\alpha\gamma + gc) + \gamma f - ec}{(\Delta t_k)^2(\gamma^2 - bc)},$$

$$\beta_2 = -\frac{(\Delta t_k)^2(-\alpha\gamma + gc) + (\Delta t_k)(\gamma f - ec + \alpha b - \gamma g) - fb + \gamma e}{(\Delta t_k)^3(-bc + \gamma^2)},$$

$$\beta_3 = 1 - \beta_1 - \beta_2.$$

2.4. Globalization strategy. In order to guaranee that the implicit preconditioned residual (IPR) method converges from any initial guess, we need to add a globalization strategy. This is an important feature, especially when dealing with highly nonlinear flow problems and high Reynolds numbers. Indeed, for highly nonlinear problems, pure methods (i.e., without a globalization strategy) will diverge unless the initial guess is very close to the solution, which is not a practical assumption. To avoid the derivatives of the merit function, which are not available, we will adapt the recently developed globalization strategy of La Cruz et al. [12] to our preconditioned version.

Assume that $\{\eta_k\}$ is a sequence such that $\eta_k > 0$ for all $k \in \mathbb{N}$ and

$$\sum_{k=0}^{\infty} \eta_k < \infty. \tag{2.16}$$

Assume also that $0 < \gamma < 1$ and $0 < \sigma_{\min} < \sigma_{\max} < \infty$. Let M be a positive integer. Let $\tau_{\min}, \tau_{\max}$ be such that $0 < \tau_{\min} < \tau_{\max} < 1$. Given an arbitrary initial point $x_0 \in \mathbb{R}^n$, an algorithm that allows us to obtain x_{k+1} starting from x_k is given below.

ALGORITHM 2.3 (Global Implicit Preconditioned Residual Method (GIPR)).
Step 1.
Choose λ_k such that $|\lambda_k| \in [\sigma_{\min}, \sigma_{\max}]$ (described in the next subsection).
Build the vector $C_k\, F(x_k)$ (inverse preconditioned direction).

Compute $\bar{f}_k = \max\{f(x_k), \ldots, f(x_{\max\{0,k-M+1\}})\}$.
Set $z \leftarrow -\lambda_k C_k\, F(x_k)$.
Set $\alpha_+ \leftarrow 1$, $\alpha_- \leftarrow 1$.

Step 2.
If $f(x_k + \alpha_+ z) \leq \bar{f}_k + \eta_k - \gamma\alpha_+^2 \|z\|_2^2$ then
Define $z_k = z, \alpha_k = \alpha_+, x_{k+1} = x_k + \alpha_k z_k$.
Else if $f(x_k - \alpha_- z) \leq \bar{f}_k + \eta_k - \gamma\alpha_-^2 \|z\|_2^2$ then
Define $z_k = -z, \alpha_k = \alpha_-, x_{k+1} = x_k + \alpha_k z_k$.
Else
Choose $\alpha_{+\mathrm{new}} \in [\tau_{\min}\alpha_+, \tau_{\max}\alpha_+]$, $\alpha_{-\mathrm{new}} \in [\tau_{\min}\alpha_-, \tau_{\max}\alpha_-]$.
Replace $\alpha_+ \leftarrow \alpha_{+\mathrm{new}}, \quad \alpha_- \leftarrow \alpha_{-\mathrm{new}}$.
Go to Step 2.

REMARK 2.4. As discussed in [12], the algorithm is well defined, i.e., the backtracking process (choosing α_{+new} and α_{-new}) is guaranteed to terminate successfully in a finite number of trials. Moreover, global convergence is also established in [12]. Indeed, if the symmetric part of the Jacobian of F at any x_k is positive (or negative) definite for all k, then the sequence $\{f(x_k)\}$ tends to zero.

3. Application to the solution of the steady 2D lid-driven cavity.

3.1. The problem. The equilibrium state of a driven square cavity is described by the steady-state Navier-Stokes equation, which, in primary variables, can be written as

$$\begin{aligned} -\frac{1}{\mathrm{Re}}\Delta U + \nabla P + (U\cdot\nabla U) &= f \text{ in } \Omega =]0,1[^2, \\ \nabla\cdot U &= 0 \text{ in } \Omega =]0,1[^2, \\ U &= \mathrm{BC} \text{ on } \partial\Omega. \end{aligned} \tag{3.1}$$

Here $U = (u,v)$ is the velocity field, P is the pressure, and f is the external force. For our applications, we will consider the so-called driven cavity case, so $f = 0$ and the fluid is driven by a proper boundary condition BC. We denote the sides of the unit square Ω by $\Gamma_i, i = 1,\ldots,4$: Γ_1 is the lower horizontal side, Γ_3 is the upper horizontal side, Γ_2 is the left vertical side, and Γ_4 is the right vertical side. In Figure 3.1, we show the location of the main vortices of the steady state of the lid driven cavity: Primary Vortex, Top Left Vortex (TL1), Bottom Right Vortices (BR1 and BR2), and Bottom Left Vortex (BL1). For additional details on this benchmark application, see [25].

We distinguish two different driven flows, according to the choice of the boundary conditions on the velocity. More precisely $U = V = 0$ at the boundary except for U on Γ_3, where we have two options:

- $U(x,1) = 1$: Cavity A (lid driven cavity)
- $U(x,1) = (1-(1-2x)^2)^2$: Cavity B (regularized lid driven cavity)

3.2. Discretization and implementation in primary variables.

3.2.1. Discretization. The discretization is performed on staggered grids of MAC type in order to verify a discrete inf-sup (or Babushka-Brezzi) condition that guarantees the stability; see [25]. Taking N discretization points in each direction on the pressure grid, we obtain

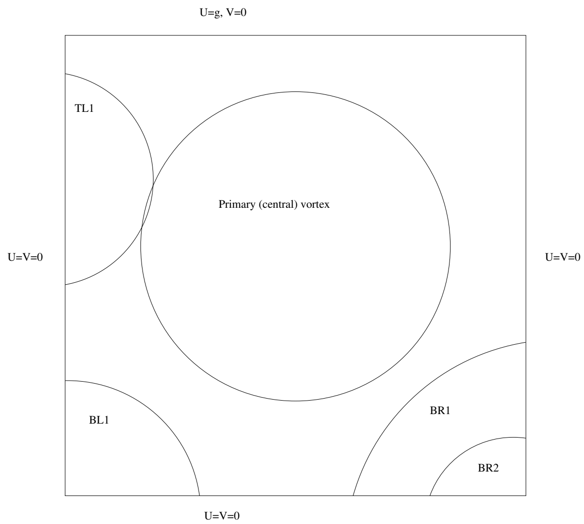

FIGURE 3.1. *The lid driven cavity - Schematic localization of the mean vortex regions*

the linear system

$$\begin{cases} \nu A_u U + B_x P + NL_u(U,V) - F_1 = 0, \\ \nu A_v V + B_y P + NL_v(U,V) - F_2 = 0, \\ \hfill B_x^t U + B_y^t V = 0, \end{cases} \tag{3.2}$$

where $U, V \in \mathbb{R}^{N(N-1)}$, $P \in \mathbb{R}^{N \times N}$, and A_u and A_v are the discretization matrices of minus the Laplacian operator on the U and V grids, respectively. Similarly, B_x and B_y are the discretization matrices of the first derivative in x and y for the pressure in the velocity grids. The operators L_u and L_v are the nonlinear terms on U and V respectively; and, finally, F_1 and F_2 are the discrete external forces for the horizontal and the vertical velocities. Equation (3.2) is then a square linear system in $2 \times N(N-1) + N^2$ unknowns.

3.2.2. Implementation. The discrete problem (3.2) can be written as a nonlinear system

$$\mathcal{F}(U,V,P) = 0,$$

with the obvious notation.

Now, let $\mathcal{S}$ be the Stokes solution operator defined by

$$\mathcal{S}(F,G,0) \mapsto (U,V,P),$$

where (U,V,P) solves the linear part of (3.2), i.e., the Stokes problem

$$\begin{cases} \nu A_u U + B_x P = F, \\ \nu A_v V + B_y P = G, \\ B_x^t U + B_v^t V = 0. \end{cases} \tag{3.3}$$

Finally we introduce the functional $\mathcal{G}$

$$\mathcal{G}((U,V,P) = \mathcal{S}(\mathcal{F}(U,V,P)).$$

The scheme consists of applying the implicit globalized preconditioned gradient method to the differential system

$$\begin{cases} \dfrac{dX}{dt} = -Z, \\ \dfrac{dZ}{dt} = \mathcal{G}(X) - \mathcal{H}Z, \end{cases} \tag{3.4}$$

where $X = (U, V, P)$ and where $\mathcal{H}$ is an approximation to the gradient of $\mathcal{G}(X)$, as described in Remark 2.1 of Section 2.3.

3.3. The $\omega - \psi$ formulation. One of the advantage of the $\omega - \psi$ formulation is that the NSE are decoupled into two problems: a convection diffusion equation and a Poisson problem. In particular, we can use the FFT to solve the linear problems, as described in this section.

3.3.1. The formulation. The $\omega - \psi$ formulation is obtained by taking the curl of the NSE [16, 25]. Let $\omega = \partial u/\partial y - \partial v/\partial x$, and write $u = \partial\psi/\partial y$, $v = -\partial\psi/\partial x$, where $\nabla\psi = \omega$. Then we have the equations

$$-\frac{1}{\mathrm{Re}}\Delta\omega + \frac{\partial\psi}{\partial y}\frac{\partial\omega}{\partial x} - \frac{\partial\psi}{\partial x}, \frac{\partial\omega}{\partial y} = 0, \tag{3.5}$$

$$\Delta\psi = \omega, \tag{3.6}$$

$$\omega(x, 0) = \omega_0(x). \tag{3.7}$$

The boundary conditions on ω are derived using the standard approach based on the discretization of $\Delta\psi$ on the boundaries; see [16, 25]. With the conditions on u and v [16, 25], we have

$$\omega(x, 0, t) = \frac{\partial^2\psi}{\partial y^2}(x, 0, t) \text{ on } \Gamma_1,$$

$$\omega(x, 1, t) = \frac{\partial^2\psi}{\partial y^2}(x, 1, t) \text{ on } \Gamma_3,$$

$$\omega(0, y, t) = \frac{\partial^2\psi}{\partial x^2}(0, y, t) \text{ on } \Gamma_2,$$

$$\omega(1, y, t) = \frac{\partial^2\psi}{\partial x^2}(1, y, t) \text{ on } \Gamma_4.$$

Therefore, since $\psi_{\partial\Omega} = 0$ and $u = \partial\psi/\partial y$, $v = -\partial\psi/\partial x$, we obtain by using Taylor expansions

$$\omega_{i,0} = \frac{1}{2h^2}\left(\psi_{i,1} - 8\psi_{i,2}\right), \tag{3.8}$$

$$\omega_{i,N+1} = \frac{1}{2h^2}\left(-\psi_{i,N-1} + 8\psi_{i,N} - 6hg(ih)\right), \tag{3.9}$$

$$\omega_{0,j} = \frac{1}{2h^2}\left(\psi_{1,j} - 8\psi_{2,j}\right), \tag{3.10}$$

$$\omega_{N+1,j} = \frac{1}{2h^2}\left(-\psi_{N-1,j} + 8\psi_{N,j}\right). \tag{3.11}$$

Here $g(x)$ denotes the boundary condition function for the horizontal velocity at the boundary Γ_3. Homogeneous Dirichlet boundary conditions apply to ψ, and the operators are discretized by second order centered schemes on a uniform mesh composed by N points in each direction of the domain of step-size $h = 1/(N+1)$. The total number of unknowns is then $2N^2$.

The boundary conditions on ω are implemented iteratively according to the relations (3.8)–(3.11), making the finite difference scheme second order accurate.

3.3.2. Implementation. With (3.8)–(3.11), we can compute the boundary conditions for ω. We denote by $\partial_x^h(\psi)$, $\partial_y^h(\psi)$, and $\partial_\Delta^h(\psi)$ the contributions of the boundary conditions to the discretization operators of ∂_x, ∂_y, and $-\Delta$. The problem to solve is

$$F_1(\omega,\psi) = \frac{1}{\mathrm{Re}}\left(A\omega + \partial_\Delta^h(\psi)\right) + D_y\psi\left(D_x\omega + \partial_x^h(\psi)\right) - D_x\psi\left(D_y\omega + \partial_y^h(\psi)\right) = 0,$$
$$F_2(\omega,\psi) = A\psi + \omega = 0.$$

Here A is the discretization matrix of $-\Delta$, and D_x and D_y are the discretization matrices of ∂_x and ∂_y respectively. The problem to solve is then

$$F(\omega,\psi) = \begin{bmatrix} F_1(\omega,\psi) \\ F_2(\omega,\psi) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

For convenience, we set $X = (\omega,\psi)$. Now, as we did for the primary variables formulation, we return to the dynamical system framework of the method: we set

$$\mathcal{G}(\omega,\psi) = \begin{bmatrix} A^{-1}F_1(\omega,\psi) \\ A^{-1}F_2(\omega,\psi) \end{bmatrix},$$

and we consider the evolutionary system

$$\begin{cases} \dfrac{dX}{dt} = -Z, \\ \dfrac{dZ}{dt} = \mathcal{G}(X) - \mathcal{H}Z, \end{cases} \tag{3.12}$$

where $\mathcal{H}Z$ is an approximation of the gradient of $\mathcal{G}(X)$ at Z, as described in Remark 2.1. Here, A is the classical pentadiagonal finite difference matrix for the Laplace operator on a square, and linear systems with A can be cheaply solved using fast solvers such as FFT or multigrid. In this paper, we will use the FFT.

4. Numerical results.

4.1. General implementation of the algorithm. We now list the information required by the GIPR method:

- The positive integer M.
- The parameters γ and η_k.
- The initial value of the descent parameter λ_0.
- The merit function. We use the Euclidian norm of the residual $\|F(X)\|_2$.
- The accuracy of the global method: the solution is considered accurate when $\|F(X)\|_2 < 10^{-6}$.
- The accuracy imposed for solving the preconditioning equation

$$\frac{F(x^k + \tau z) - F(x^k)}{\tau} - F(x^k) = 0,$$

which is characterized by

- The choice of the optimization method 1. In our implementation we have used the Enhanced Cauchy 2 as discussed above.
- The number τ. We set $\tau = 10^{-8}$.
- The number of iteration nprec, which can vary at each step. We choose to increase nprec as the norm of the residual $r^k = F(x_k)$ decreases in order to improve the preconditioning near the solution:

  Adaptive computation of nprec:

  nprec_0 given
  for $k = 0, \ldots$ until convergence
  if $\|r^k\|_2 < 0.1$ then
  $\mathrm{nprec} = \text{ceil} \times \big(-\log_{10}(\|r^k\|_2) + 1\big) \times \mathrm{nprec}_0$.

The reasoning behind the decision to improve the quality of the preconditioner near the solution is this: near an isolated solution, the merit function f is strictly convex, and hence it makes sense to reduce the condition number of the approximated positive-definite Hessian of f in the neighborhood of the solution. This is achieved by increasing nprec as the norm of the residual, i.e., the value of the merit function, is reduced.

4.2. Computation of steady states of NSE. We now present the numerical solution of the steady state of the two-dimensional driven cavity for different Reynolds numbers. Our results agree with those in the literature [3, 7, 15, 18, 19, 24, 28–30] (see figures and tables below); and to prove the robustness of the solution method, we take as the initial guess the solution of the Stokes problem, which becomes farther from the steady state as the Reynolds number increases. We will pay special attention to the solution of NSE in the $\omega - \Psi$ formulation. However, let us mention that the scheme applies also to NSE in primary variables $(U - P)$, the linear solver being a Stokes solver. The crucial practical point is to have at our disposal a fast solver for the linear problems: FFT or multigrid for the $\omega - \Psi$ formulation and multigrid Uzawa [7] for the $U - P$ formulation.

As we see in our results, the globalization strategy is important while the residual is not small enough. Furthermore, the preconditioning makes sense "close to the solution." For that reason, we choose to activate the preconditioning progressively as the residual decreases by increasing the number of inner iterations in the solution of the preconditioning step (step 1 of the scheme). This allows us to obtain fast convergence at the end while saving computational time at the beginning.

We observe that the number of outer iterations increases with the Reynolds number but not so much with the dimension of the problem. In all cases, the first part of the convergence process is devoted to "maintaining" the iterates in a neighborhood of the solution. All the computations have been done with MATLAB 7 on a 2Ghz dual core PC with two gigabytes of RAM.

We now present the parameters of the scheme that we used for solving the flow in cavity B and also in cavity A for the stream function-vorticity formulation of NSE. N is the number of discretization points in each direction of the domain. We give the parameter values in Tables 4.1–4.2.

TABLE 4.1
Cavity B

Re	N	γ	M	nprec_0	Prec. Method	Adapted nprec	α_0
1000	127	9×10^2	2	4	Enhanced Cauchy 2	yes	10
2000	127	9×10^{11}	2	4	Enhanced Cauchy 2	yes	10^4
5000	255	9×10^2	2	4	Enhanced Cauchy 2	yes	10^4

TABLE 4.2
Cavity A

Re	N	γ	M	nprec_0	Prec. Method	Adapted nprec	α_0
1000	127	9×10^2	2	4	Enhanced Cauchy 2	yes	10
3200	255	9×10^6	2	4	Enhanced Cauchy 2	yes	10^4

The results are reported in Figures 4.1–4.3. Most of the work is done at the beginning of the iterations, while the globalization is acting to stabilize the iterates. This phenomenon is amplified as the Reynolds number Re becomes large. An acceleration of the convergence is obtained when the residual is small enough, since nprec increases. The shapes of the solutions are identical with that of the literature, such as in [3, 7, 15, 16, 18, 19, 24, 28–30]; particularly, the special values agree.

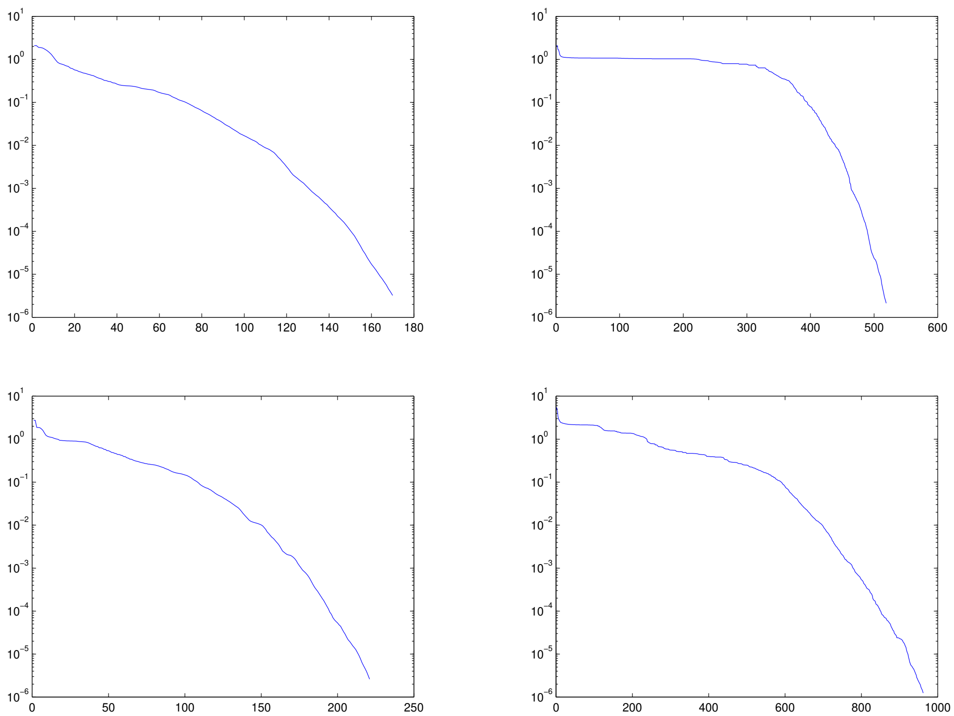

FIGURE 4.1. *Convergence history (residual norm versus iterations): First row, Cavity B,* $\mathrm{Re} = 1000$ *(left), and* $\mathrm{Re} = 2000$ *(right); second row, cavity A,* $\mathrm{Re} = 1000$ *(left), and* $\mathrm{Re} = 3200$ *(right).*

We also report some special values of the solution for Cavity B and $\mathrm{Re} = 5000$ in Table 4.3, where we also compare them with results in the literature.

4.3. Solution of NSE in primary variables. We now present the numerical results for the solution of the steady-state NSE in primary variables. We change the value of γ during the iterations in order to increase the non-monotonicity of the GIPR as follows

$$\text{if } \|r^k\| < 10^{-3} \text{ then } \gamma = 0.9.$$

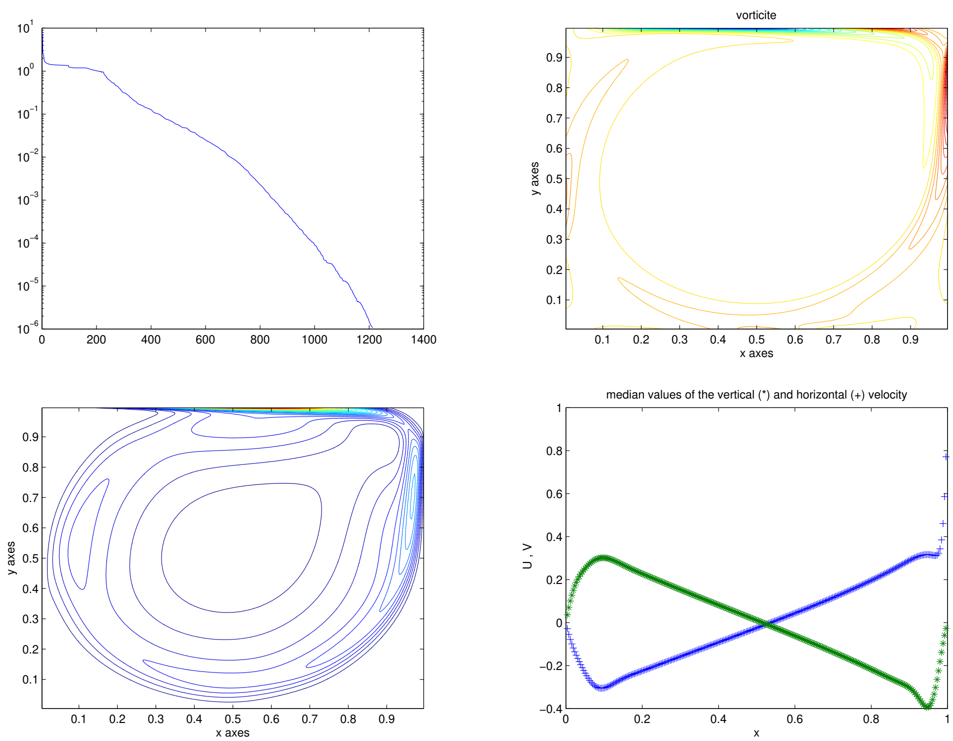

FIGURE 4.2. *Steady-state NSE,* $\mathrm{Re} = 5000$*, N=255. Residual norm versus iterations (top left), isolines of the vorticity (top right), isolines of the kinetic energy (bottom left), and median values of the horizontal and of the vertical velocity (bottom right).*

This practical choice reduces the number of line searches (backtrackings) near the solution, and so it reduces the overall computational effort for convergence. In Table 4.4, we present the numerical solution of the cavity B problem for $\mathrm{Re} = 400$ and $\mathrm{Re} = 1000$. The level curves of the pressure, the vorticity, the kinetic energy and the stream function agree with those in the literature. Notice that fewer iterations are required for convergence than are needed for the same example using the stream-vorticity formulation. This is due mainly to the fact that the boundary conditions are implemented exactly for the primary variables formulation, whereas they are approximated iteratively for the stream-vorticity formulation. However, the computational effort required at each iteration of the primary variables formulation is greater than that needed for the stream-vorticity formulation. Indeed, in the first case a coupled system of $3N^2$ variables needs to be solved, whereas two systems of N^2 variables need to be solved in the second case; see Section 3.

5. Concluding remarks. We have presented a robust scheme to solve steady fluid flows that involves an implicit preconditioned search direction to approximate the Newton direction directly. Our approach is in sharp contrast with the classical Newton type schemes (e.g., Newton-Krylov [21], defect corrections [31]) in which the nonlinear problem is linearized at every Newton iteration and a preconditioner is developed for the inner linear solver in the standard way. Here, we propose to avoid the linearization process by preconditioning (implicitly) the nonlinear problem. The efficiency of the scheme is increased when a fast solver is used for the preconditioned linear system. The results we obtain on the numerical

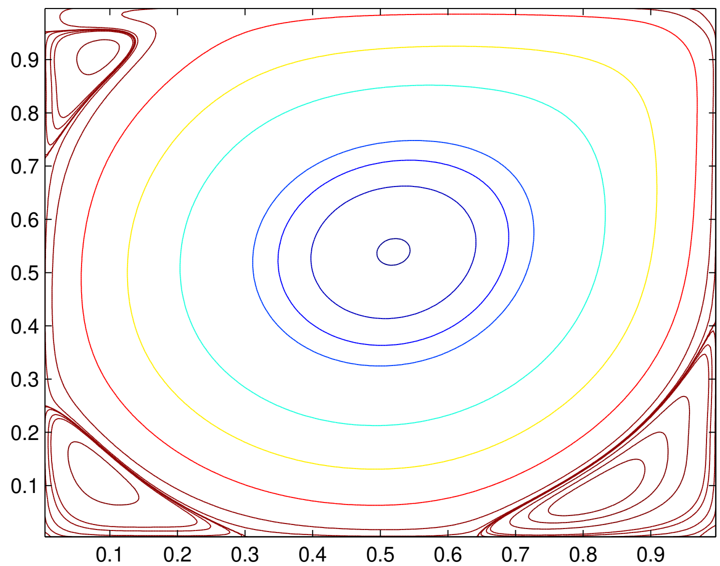

FIGURE 4.3. *Steady-state NSE, Cavity B,* $\mathrm{Re} = 5000$*, N=255 isolines of the stream function.*

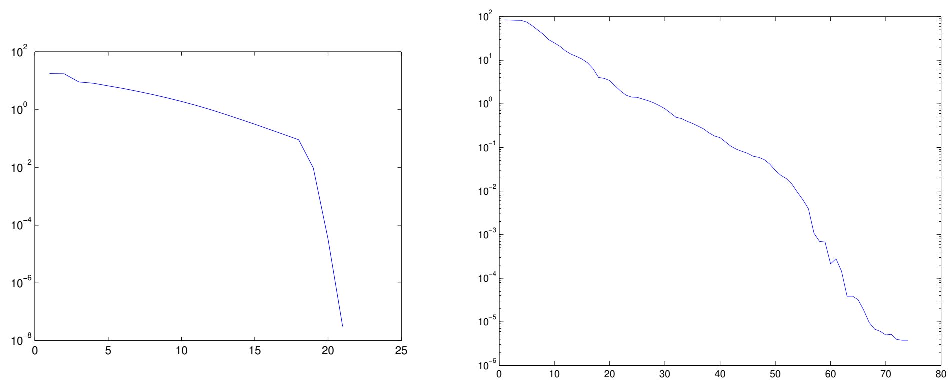

FIGURE 4.4. *Convergence history (residual norm versus iterations), Cavity B, N=63,* $\mathrm{Re} = 400$ *(left) and N=127,* $\mathrm{Re} = 1000$ *(right).*

solution of NSE, by adding a globalization strategy, show that the proposed method is robust. As it has been already established, it is harder to solve directly the steady-state NSE than to compute the steady state by time marching schemes applied to the evolutionary equation.

The new method is also flexible since the choice of the preconditioning step is completely free. We would like to stress that the preconditioned globalized spectral residual method can be applied to a large number of scientific computing problems, especially when no (simple) preconditioner can be built, such as in Computational Fluid Dynamics (CFD), and also in numerical linear algebra when solving Riccati matrix equations or some other nonlinear matrix problems. These are topics that deserve further investigation.

Acknowledgments. This work was supported by SIMPAF project from INRIA futurs.

TABLE 4.3
Cavity B. Re $= 5000$

	IGPR	IGPR	Shen [29]	Pascal [24]
Formulation	$\omega - \psi$	$\omega - \psi$	U, P	U, P
Discretization x	FD	FD	Spectral	FEM
Grid / Mesh	127×127	255×255	33×33	129×129
Δt			0.03	0.05
Vortex				
x	0.5234	0.51953	0.516	0.5390
y	0.539	0.539	0.531	0.5313
ψ	-0.07761	-0.085211	-0.08776	-0.0975
ω	-1.2687	-1.3866		-2.169
Vortex (B L)				
x	0.078125	0.078125	0.094	0.0859
y	0.125	0.125	0.094	0.1172
ψ	6.8393×10^{-4}	7.95×10^{-4}	7.5268×10^{-4}	6.723×10^{-4}
ω	0.7468	0.844		0.7310
Vortex (B R)				
x	0.8203	0.8164	0.922	0.8047
y	0.08593	0.082	0.094	0.0781
ψ	1.8528×10^{-3}	2.041×10^{-3}	0.77475×10^{-3}	2.42×10^{-3}
ω	1.42177	1.58687		2.009
Vortex (T L)				
x	0.07812	0.0859	0.078	0.0781
y	0.9062	0.9101	0.92	0.906
ψ	5.6645×10^{-4}	7.149×10^{-4}	6.778×10^{-4}	7.86×10^{-4}
ω	0.88813	1.098		1.159

TABLE 4.4
Cavity B.

Re	N	γ_0	M	nprec_0	Prec. Method	Adapted nprec	α_0
400	63	10^4	2	5	Enhanced Cauchy 2	yes	10^2
1000	127	10^4	2	4	Enhanced Cauchy 2	yes	10^4

REFERENCES

[1] F. ALVAREZ, H. ATTOUCH, J. BOLTE, AND P. REDONT, *A second-order gradient-like dissipative dynamical system with hessian-driven damping. application to optimization and mechanics*, J. Math. Pures Appl., 81 (2002), pp. 747–779.

[2] J. BARZILAI AND J. M. BORWEIN, *Two-point step size gradient methods*, IMA J. Numer. Anal., 8 (1988), pp. 141–148.

[3] M. BEN-ARTZI, J.-P. CROISILLE, D. FISHELOV, AND S. TRACHTENBERG, *A pure-compact scheme for the streamfunction formulation of Navier-Stokes equations*, J. Comput. Phys., 205 (2005), pp. 640–664.

[4] C. BREZINSKI AND J.-P. CHEHAB, *Nonlinear hybrid procedures and fixed point iterations*, Numer. Funct. Anal. Optim., 19 (1998), pp. 465–487.

[5] P. BROWN AND Y. SAAD, *Hybrid Krylov methods for nonlinear systems of equations*, SIAM J. Sci. Comput., 11 (1990), pp. 450–481.

[6] ———, *Convergence theory of nonlinear Newton-Krylov algorithms*, SIAM J. Optim., 4 (1994), pp. 297–330.

[7] C.-H. BRUNEAU AND C. JOURON, *An efficient scheme for solving steady incompressible Navier-Stokes equations*, J. Comput. Phys., 89 (1990), pp. 389–413.

[8] A. CAUCHY, *Méthodes générales pour la résolution des systèmes d'équations simultanées*, C. R. Acad. Sci.

Par., 25 (1847), pp. 536–538.
[9] J.-P. CHEHAB AND J. LAMINIE, *Differential equations and solution of linear systems*, Numer. Algorithms, 40 (2005), pp. 103–124.
[10] J.-P. CHEHAB AND M. RAYDAN, *Inverse and adaptive preconditioned gradient methods for nonlinear problems*, Appl. Numer. Math., 55 (2005), pp. 32–47.
[11] M. CROUZEIX, *Approximation et méthodes itératives de résolution d'inéquations variationelles et de problèmes non linéaires*, IRIA, 12 (1974), pp. 139–244.
[12] W. L. CRUZ, J. M. MARTÍNEZ, AND M. RAYDAN, *Spectral residual method without gradient information for solving large-scale nonlinear systems*, Math. Comp., 75 (2006), pp. 1449–1466.
[13] Y. H. DAI AND L. Z. LIAO, *r-linear convergence of the Barzilai and Borwein gradient method*, IMA J. Numer. Anal., 22 (2002), pp. 1–10.
[14] J. E. DENNIS, JR. AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.
[15] E. ERTURK, T. C. CORKE, AND C. GÖKÇÖL, *Numerical solutions of 2-d steady incompressible driven cavity flow at high Reynolds numbers*, Internat. J. Numer. Methods Fluids, 48 (2005), pp. 747–774.
[16] D. EUVRARD, *Résolution numérique des équations aux dérivées partielles*, third ed., Dunod, Paris, 1994.
[17] R. FLETCHER, *On the Barzilai-Borwein method*, in Optimization and Control with Applications, L. Qi, K. L. Teo, and X. Q. Yang, eds., Springer, New York, 2005, pp. 235–256.
[18] U. GHIA, K. N. GHIA, AND C. T. SHIN, *High-Re solutions of incompressible flow using the Navier-Stokes equations and the multigrid method*, J. Comput. Phys., 48 (1982), pp. 387–411.
[19] O. GOYON, *High-Reynolds number solutions of Navier-Stokes equations using incremental unknowns*, Comput. Methods Appl. Mech. Engrg., 130 (1996), pp. 319–335.
[20] U. HELMKE AND J. B. MOORE, *Optimization and Dynamical Systems.*, Springer, Berlin, 1994.
[21] C. T. KELLEY, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia, 1995.
[22] W. LA CRUZ AND M. RAYDAN, *Nonmonotone spectral methods for large-scale nonlinear systems*, Optim. Methods and Softw., 18 (2003), pp. 583–599.
[23] F. LUENGO, M. RAYDAN, W. GLUNT, AND T. HAYDEN, *Preconditioned spectral gradient method*, Numer. Algorithms, 30 (2002), pp. 241–258.
[24] F. PASCAL, *Méthodes de Galerkin non linéaires en discrétisation par éléments finis et pseudo-spectrale. Application à la mécanique des fluides*, Ph.D. thesis, Département de Mathématiques, Université Paris XI Orsay, Jan. 1992.
[25] R. PEYRET AND R. TAYLOR, *Computational Methods for Fluid Flows*, Springer, New York, 1983.
[26] M. RAYDAN, *On the Barzilai and Borwein choice of the steplength for the gradient method*, IMA J. Numer. Anal., 13 (1993), pp. 321–326.
[27] ———, *The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem*, SIAM J. Optim., 7 (1997), pp. 26–33.
[28] J. SHEN, *Numerical simulation of the regularized driven cavity flows at high Reynolds numbers*, Comput. Methods Appl. Mech. Engrg., 80 (1990), pp. 273–280.
[29] ———, *Hopf bifurcation of the unsteady regularized driven cavity flow*, J. Comput. Phys., 95 (1991), pp. 228–245.
[30] L. SONKE TABUGUIA, *Etude numérique des équations des Navier-Stokes en milieux multiplement connexes, en formulation vitesse-tourbillon, par une approche multidomaine*, Ph.D. thesis, Département de Mathématiques, Université Paris XI Orsay, 1989.
[31] S. TUREK, *Efficient Solvers for Incompressible Flow Problems*, Springer, Berlin, 1999.