JOURNAL OF COMPUTATIONAL PHYSICS **136,** 546–558 (1997)
ARTICLE NO. CP975780

A Spectral Multidomain Method for the Numerical Simulation of Turbulent Flows

A. Pinelli,*[1] A. Vacca,† and A. Quarteroni‡

*School of Aeronautics, Polytechnic University of Madrid, Madrid, Spain; †Department of Civil Engineering, Second University of Naples, Naples, Italy; ‡Department of Mathematics, Polytechnic of Milan, Milan, Italy and CRS4, Cagliari, Italy

Received May 1, 1996; revised April 29, 1997

The primitive variable formulation of the unsteady incompressible Navier–Stokes equations in three space dimensions is discretized with a combined Fourier–Legendre spectral method. A semi-implicit pressure correction scheme is applied to decouple the velocity from the pressure. The arising elliptic scalar problems are first diagonalized in the periodic Fourier direction and then solved by a multidomain Legendre collocation method in the two remaining space coordinates. In particular, both an iterative and a direct version of the so-called projection decomposition method (PDM) are introduced to separate the equations for the internal nodes from the ones governing the interface unknowns. The PDM method, first introduced by V. Agoshkov and E. Ovtchinnikov and later applied to spectral methods by P. Gervasio, E. Ovtchinnikov, and A. Quarteroni is a domain decomposition technique for elliptic boundary value problems, which is based on a Galerkin approximation of the Steklov–Poincaré equation for the unknown variables associated to the grid points lying on the interface between subdomains. After having shown the exponential convergence of the proposed discretization technique, some issues on the efficient implementation of the method are given. Finally, as an illustration of the potentialities of the algorithm for the numerical simulation of turbulent flows, the results of a direct numerical simulation (DNS) of a fully turbulent plane channel flow are presented.

1. INTRODUCTION

Spectral methods have been and still remain the method of choice for numerical simulations of fluid phenomena where accuracy plays a fundamental role. The projection decomposition method, first introduced by [1], was later applied to spectral methods [2].

Their inherent high-order accuracy and low phase error motivated the development and implementation of several spectral-based direct numerical simulation (DNS) and large eddy simulation (LES) codes for the solution of turbulent flows [3, 4]. Nevertheless, almost all these methods have been designed to be extremely efficient for very simple geometries on supercomputers (highly vectorized shared memory machines). On the other hand, domain decomposition methods are becoming viable tools to overcome several intrinsic limitations of spectral methods, allowing for the use of the latter in a wider context [5–8]. The most obvious application of spectral multidomain methods is related to the solution of partial differential equations over complex geometries (i.e., geometries which cannot be trivially mapped in the standard $[-1, 1]$ square). Another important feature of this class of algorithms is the natural way in which they exploit the architectures of modern MIMD computers (including clusters of workstations) paving the way for large scale simulations otherwise accessible only to supercomputer users. In fact, the parallelization efficiency is extremely favourable to spectral multidomain methods: the ratio of computation time to communication time is larger for this family of methods than for others. This is mainly due to the high order accuracy provided by the spectral method combined with the fact that the discrete operators involve matrices not as sparse as other ''local'' methods (i.e., finite differences, finite element or finite volumes) [9].

The present Navier–Stokes solution algorithm differs from already well established spectral domain decomposition methods (i.e., the spectral element method). The main difference consists in the treatment of the elliptic kernels arising after the application of a continuous semi-implicit pressure correction scheme. Each scalar elliptic boundary value problem is transformed in a set of analogous problems over subdomains whose boundary values are provided by an abstract equation on the interface between subdomains. This procedure allows us to split the equations over two sets of geometrical elements: the subdomains themselves, where the solution is approximated by a local Legendre polynomial basis in the framework of a collocation method; and the subdomain interfaces, where the trace of the solution is approximated through a Galerkin method using a set of special basis functions. Such a sharp subdivision introduces a very flexible tool that is able to deal with more general discretizations such as a nonconforming multidomain partition of the original computational domain. Work is in progress in this direction and it will be the subject of a future publication.

[1] E-mail: pinelli@torroja.dmt.upm.es.

0021-9991/97 $25.00

The illustration of the present method will be divided into two main parts. First, both the continuous and the discrete versions of the algorithm are presented and validated in a two-dimensional framework; in the second part, some algorithm enhancements, including fast direct solvers for the elliptic kernels and a Fourier three-dimensional extension are presented. Finally, to prove that the final algorithm is an effective tool for the numerical simulation of turbulent flows, the results of a DNS of a fully turbulent Poiseuille flow are presented and validated.

2. TIME SPLITTING SCHEME

The incompressible Navier–Stokes equations can be reformulated in nondimensional form written as

$$\frac{\partial U}{\partial t}+\frac{1}{2}(U\cdot\nabla U+\nabla\cdot(UU))=-\nabla p+\frac{1}{\mathrm{Re}}\nabla^2 U \quad \text{in } \Omega \tag{1}$$

$$\nabla\cdot U=0 \quad \text{in } \Omega \tag{2}$$

$$U=G \quad \text{on } \partial\Omega \tag{3}$$

The advective terms have been considered in their skew-symmetric formulation [10] to avoid dealiasing procedures while semiconserving the energy. In the above equations, Ω is a reference domain in R^d, U is the velocity vector, p is the pressure, and G is a prescribed boundary vector function. Also, $\mathrm{Re}=u_d H_d/\nu$ is the Reynolds number based on a reference velocity u_d, a reference length H_d, and the kinematic viscosity ν.

In Eq. (1) and (2) velocity and pressure are coupled together by the incompressibility constraint, which makes them difficult to solve. Classical procedures to overcome this problem are provided by time splitting schemes [11, 12].

The basic idea consists in decoupling the computation of pressure and velocity at each time step. The terms associated with the spatial derivatives appearing in Eq. (1) and (2) might be computed at old, new, or intermediate time levels. Implicit treatment of the viscous terms allow us to overcome the most severe time step restriction when dealing with spectral methods [13] (i.e., $\Delta t\sim \mathrm{Re}/N^4$, N being the polynomial degree of the spectral discretization). The particular splitting method that has been selected to advance in time the equations is based on the semi-implicit pressure correction method developed by Van Kan [14]; the diffusive terms are treated with a Crank–Nicholson discretization while an Adams–Bashforth scheme is used for the advective terms. The following formulae summarise the time discretized version of the adopted fractional step scheme to advance the solution from $t^n=n\,\Delta t$ to $t^{n+1}=(n+1)\,\Delta t$ (the upper index refers to the time level),

$$\begin{aligned}\frac{1}{\Delta t}(\hat U-U^n)-\frac{1}{2\mathrm{Re}}\nabla^2(\hat U+U^n)&\\ =-\nabla p^n-\frac{3}{2}\mathscr{L}(U^n)+\frac{1}{2}\mathscr{L}(U^{n-1}) \quad &\text{in } \Omega\\ \hat U=G((n+1)\,\Delta t) \quad &\text{on } \partial\Omega\end{aligned} \tag{4}$$

$$\begin{aligned}\frac{1}{\Delta t}(U^{n+1}-\hat U)+\frac{1}{2}\nabla(p^{n+1}-p^n)=0 \quad &\text{in } \Omega\\ \nabla\cdot U^{n+1}=0 \quad &\text{in } \Omega\\ U^{n+1}\cdot\mathbf{n}=G((n+1)\,\Delta t) \quad &\text{on } \partial\Omega,\end{aligned} \tag{5}$$

where $\mathscr{L}(U)$ represents the advective term $\frac{1}{2}\,(U\cdot\nabla U+\nabla\cdot(UU))$, and $\mathbf{n}$ is the unity vector normal to the boundary $\partial\Omega$.

In the first step (4), a nonphysical intermediate velocity field $\hat U$ is computed: $\hat U$ does not satisfy the incompressibility condition. Then in the second step (5), $\hat U$ is projected onto the divergence free space to get a convenient velocity approximation of U^{n+1}.

The scheme with the given boundary conditions is nothing else than a second-order Crank–Nicholson Adams–Bashforth scheme on the whole problem, with an $\mathscr{O}(\Delta t^2)$ deviation in the tangential direction to the boundary. Indeed, the relation [11]

$$U^{n+1}\cdot\mathbf{t}=U((n+1)\Delta t)\cdot\mathbf{t}-\Delta\nabla(p^{n+1}-p^n)\cdot\mathbf{t} \quad \text{on } \partial\Omega, \tag{6}$$

holds, $\mathbf{t}$ being the unitary vectors tangent to the boundary. By applying the divergence operator to (5), it turns out that the projection step is equivalent to

$$\nabla^2(p^{n+1}+p^n)=\frac{2}{\Delta t}\nabla\cdot\hat U \quad \text{in } \Omega \text{ with } \frac{\partial p^{n+1}}{\partial\mathbf{n}}=0 \text{ on } \partial\Omega \tag{7}$$

$$U^{n+1}=\hat U-\frac{\Delta t}{2}\nabla\,(p^{n+1}-p^n) \quad \text{in } \overline{\Omega}. \tag{8}$$

Therefore, the selected time-stepping procedure leads to a cascade of scalar elliptic kernels (4), (7), to be solved at each time step. Namely, two (for the two-dimensional equations) Helmholtz problems for the determination of the predicted velocity field and one Poisson problem for the pressure need to be solved at each time step. It it then clear that, in order to achieve a globally efficient algorithm, it is of paramount importance to tackle effectively the mentioned scalar problems.

In the next sections the attention will be focused on the way each scalar elliptic problem can be efficiently solved in the framework of a spectral multidomain discretization.

3. MULTIDOMAIN HELMHOLTZ SOLVER

To illustrate the method in a simple manner, only two dimensional equations will be considered hereafter. The

following problem, representative of one of the elliptic scalar problems mentioned in the previous section, is considered,

$$\begin{aligned} -\nabla^2 u + \alpha u &= f \quad \text{in } \Omega \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{9}$$

where α is a real constant ≥ 0, $f \in L^2(\Omega)$, and Ω is an open connected set $\Omega \subset R^2$; in particular, $\overline{\Omega} = \cup_{i=1}^{M} \overline{\Omega}_i$ with $\overline{\Omega}_i$ is a closed rectangle having either common side or common vertex with each neighbour; α is either equal to zero (e.g., for the Poisson problem related with the pressure), or is equal to $2/\Delta t$ Re (e.g., for each one of the momentum equations).

Let us point out that the outlined conformal discretization is not mandatory for the present method. In fact, as it will be shown, the present algorithm allows for a complete decoupling of the solution between subdomains and interface. This feature makes the algorithm highly flexible allowing in principle both nonconformal partitioning of the original domain and use of hybrid discretizations (different space discretizations in different regions of the computational domain).

3.1. *Continuous Formulation*

The equivalent weak formulation of (9) is [8]

$$\begin{aligned} &\text{find } u \in H_0^1(\Omega) \text{ such that} \\ &\quad l(u, v) = (f, v)_{L^2(\Omega)} \quad \forall\, v \in H_0^1(\Omega), \end{aligned} \tag{10}$$

where $H_0^1(\Omega)$ is the real Hilbert space defined as

$$\begin{aligned} H_0^1(\Omega) \equiv \{u \in L^2(\Omega) &: \frac{\partial u}{\partial x_1} \in L^2(\Omega) \\ &\text{and } \frac{\partial u}{\partial x_2} \in L^2(\Omega),\ u|_{\partial\Omega} = 0\} \end{aligned} \tag{11}$$

equipped with the scalar product

$$l(u, v) = \int_\Omega (\nabla u \cdot \nabla v + \alpha u v)\, d\Omega \quad \forall u, v \in H_0^1(\Omega). \tag{12}$$

Following classical domain decomposition techniques, problem (10) is decoupled into a set of problems within each subdomain plus an additional problem at the interface Γ,

$$\Gamma = (\Omega \backslash \Omega_0) \backslash \partial\Omega \quad \text{with } \Omega_0 = \cup_{i=1}^{M} \Omega_i. \tag{13}$$

Let S be the space of the functions defined on Γ:

$$\begin{aligned} &S = \{z \,|\, \exists\, v \in H_0^1(\Omega) : z = v_{|\Gamma}\} \quad \text{equipped with the norm} \\ &\quad \|z\| = \text{Inf}\{\|v\|_{H_0^1(\Omega)} : z = v_{|\Gamma}\}. \end{aligned} \tag{14}$$

Let T denote the "trace operator" from $H_0^1(\Omega)$ onto S, defined as

$$T\phi = \phi_{|\Gamma} \quad \forall \phi \in H_0^1(\Omega). \tag{15}$$

The T operator allows us to identify two closed mutually orthogonal subspaces,

$$K \equiv \ker(T) = \{u_0 \in H_0^1(\Omega) : Tu_0 = 0\}, \tag{16}$$

where $\ker(T)$ is the kernel of operator T and its orthogonal complement $K^\perp$ is defined as

$$K^\perp \equiv \{\tilde{u} \in H_0^1(\Omega) : l(\tilde{u}, v_0) = 0 \quad \forall\, v_0 \in K\}. \tag{17}$$

Therefore, the solution $u \in H_0^1(\Omega)$ of problem (10) can be uniquely decomposed as

$$u = u_0 + \tilde{u}, \quad u_0 \in K, \quad \tilde{u} \in K^\perp. \tag{18}$$

Since the restriction T_0 of the operator T to $K^\perp$ is an isometric isomorphism between $K^\perp$ and S it follows that [15]

$$\forall\, \tilde{u} \in K^\perp \quad \exists!\, \psi \in S : \tilde{u} = T_0^{-1}\psi. \tag{19}$$

It is not difficult to see that $\tilde{u}$ satisfies the Helmholtz problem $-\nabla^2\tilde{u} + \alpha\tilde{u} = 0$ in Ω, with $\tilde{u}_{|\Gamma} = \psi$. For such a reason, the operator T_0^{-1} is usually termed as "harmonic extension" to $H_0^1(\Omega)$ of any function belonging to S [2]. Identity (18) can be reformulated as

$$u = u_0 + T_0^{-1}\psi \quad \text{with } u_0 \in K;\ \psi \in S. \tag{20}$$

Thus, problem (10) can be easily proven to be equivalent to the set of the two following ones.

Problem P1. Find $u_0 \in K$ such that

$$l(u_0, v_0) = (f, v_0)_{L^2(\Omega)} \quad \forall\, v_0 \in K. \tag{21}$$

Problem P2. Find $\psi \in S$ such that

$$l(T_0^{-1}\psi, T_0^{-1}z) = (f, T_0^{-1}z)_{L^2(\Omega)} \quad \forall z \in S. \tag{22}$$

Problem P1 is nothing else than the solution of M decoupled elliptic problems with homogeneous Dirichlet boundary conditions on both $\partial\Omega$ and Γ.

On the other hand, Problem P2 consists in finding the harmonic extension (T_0^{-1}) of a function ψ defined on the interface Γ which guarantees a jump of the conormal derivatives along Γ equal to the one induced by the solution

$u_0(x, y)$ of Problem P1. In fact, by applying the Green formula to (22), we obtain

$$\sum_{i=1}^{M} \int_{\partial\Omega_i \cap \Gamma} \frac{\partial}{\partial n_i} (u_{0_{|\Omega_i}} + \tilde{u}_{|\Omega_i})\, z\, d\Gamma = 0 \quad \forall z \in S, \tag{23}$$

where n_i is the normal unit vector on $\partial\Omega_i$ (directed outward) and $v_{|\Omega_i}$ denotes the restriction of the function v to Ω_i. Equation (23) highlights that for the implementation of Problem P2 it is sufficient to provide basis functions defined only along Γ (i.e., basis functions for S).

The introduction and definition of Problems P1 and P2 represent the key point to completely decouple the solution of problem (10) into two pieces, u_0 and $\tilde{u}$, leading to two different geometrical entities: u_0 to the nodes internal to the subdomains, and $\tilde{u}$ to the nodes lying on the interfaces between the subdomains.

3.2. *Discrete Formulation*

Let $\mathbf{P}_N$ denote the space of algebraic polynomials of degree $\leq N_x$ with respect to the x variable and N_y with respect to the y one. Let, moreover, (x_k, y_l) $(0 \leq k \leq N_x;\ 0 \leq l \leq N_y)$ denote the $(N_x + 1) \otimes (N_y + 1)$ nodes of the Gauss–Lobatto–Legendre integration formula (e.g., Davis and Rabinowitz [16, 13]). We recall that $x_0 = y_0 = -1$, $x_{N_x} = y_{N_y} = +1$ and $L'_{N_x}(x_k) = 0$, $L'_{N_y}(y_l) = 0$ for all k and l $(1 \leq k \leq N_x - 1, 1 \leq l \leq N_y - 1)$, where L_N represents the N$^{\text{th}}$ Legendre polynomial. Correspondingly, let ω_{kl} denote the weights associated to the nodes (x_k, y_l) of the above-mentioned integration formula. Then we look for a discrete function (i.e., Problem P1), $u_0 \in K$, such that $u_0^i \equiv u_{0_{|\Omega_i}} \in \mathbf{P}_N^{0,i}$, for each $i = 1 \cdots N$ with $\mathbf{P}_N^{0,i} = \{v \in \mathbf{P}_N(\Omega_i) |\ v_{|\partial\Omega i} = 0\}$ satisfying the generalised Galerkin problem,

$$l_N(u_0^i, v_0^i) = (f, v_0^i)_N^i \quad \forall\, v_0^i \in \mathbf{P}_N^{0,i}, \tag{24}$$

where we have set

$$(f, v_0^i)_N^i = \sum_{k=0}^{N_x} \sum_{l=0}^{N_y} f(x_k^i, y_l^i)\, v_0^i(x_k^i, y_l^i)\, \omega_{kl}^i \tag{25}$$

and

$$l_N(u_0^i, v_0^i) = (\nabla u_0^i, \nabla v_0^i)_N^i + \alpha\, (u_0^i, v_0^i)_N^i. \tag{26}$$

Since, (x_k^i, y_l^i) are the nodes in Ω_i corresponding to the images of (x_k, y_l) through the mapping: $\Omega \to \Omega_i$, and $\omega_{kl}^i = \omega_{kl} \cdot \text{meas}(\Omega_i)/\text{meas}(\Omega)$, we have that $(u, v)_N^i$ represents the Gauss–Lobatto–Legendre approximation to the integral $\int_{\Omega_i} uv\, ds$. In particular, choosing as test functions v_0^i, the Lagrange polynomials of degree $N_x \otimes N_y$ associated with the internal Gauss–Lobatto nodes (x_k^i, y_l^i) $(1 \leq k \leq N_x - 1,\ 1 \leq l \leq N_y - 1)$, and using discrete integration by parts, we obtain from (24) the following equivalent collocation statement (see, e.g., Quarteroni and Valli [8, Chap. 6]):

$$\begin{aligned} -\nabla^2 u_0^i + \alpha u_0^i &= f \text{ at } (x_k^i, y_l^i), \\ 1 \leq k \leq N_x - 1;\ 1 \leq l \leq N_y - 1, \\ u_0^i &= 0 \quad \text{on } \partial\Omega_i. \end{aligned} \tag{27}$$

As concerns Problem P2, namely the solution of Equation (23), let $\{\xi_i\}$, $i = 1, \ldots, \infty$, be a set of linearly independent polynomials which constitute a dense basis for S. Due to the character of the trace operator T_0 it is possible to approximate any harmonic function in $K^\perp$ and particularly $\tilde{u}$ as

$$\tilde{u} = \sum_{k=1}^{K} a_k (T_0^{-1} \xi_k). \tag{28}$$

Consequently, applying a standard Galerkin technique, problem (23) can be approximated with the following set of K algebraic equations,

$$\begin{aligned} \sum_{i=1}^{M} \int_{\partial\Omega_i \cap \Gamma} \left(\frac{\partial}{\partial n} \left(\sum_{k=1}^{K} a_k (T_0^{-1} \xi_k) \right) \right) \xi_h d\Gamma \\ = -\sum_{i=1}^{M} \int_{\partial\Omega_i \cap \Gamma} \frac{\partial u_0}{\partial n}\, \xi_h d\Gamma \quad \forall\, h = 1, \ldots, K. \end{aligned} \tag{29}$$

The previous set of algebraic equations (29) can be recast in matrix notation as

$$\mathbf{A}a = b, \tag{30}$$

where $a = a_k$, $k = 1, \ldots, K$, is the vector containing the unknown coefficients of the solution, and $b = b_k$, $k = 1, \ldots, K$, contains the coefficient of the data in Eq. (29).

The matrix $\mathbf{A}$ is dense with entries a priori unknowns. Nevertheless, Eq. (29) provides the tool to compute the action of $\mathbf{A}$ over a vector a containing the Galerkin coefficients of the jump of the normal derivatives along the interface Γ. Besides, the condition number of $\mathbf{A}$ can grow like $O(K)$, unless the basis $\{\xi_k\}$ is not properly chosen. To solve system (30), an iterative technique is therefore recommended. To simplify the illustration of the present algorithm, let us consider one generic iteration $a^{j+1} = a^j - \alpha\, (\mathbf{A}a^j - b)$ (e.g., the j^{th} one) of the Richardson iterative method:

$$\begin{aligned} a_k^{j+1} = a_k^j - \alpha \sum_{i=1}^{M} \int_{\partial\Omega_i \cap \Gamma} \\ \left(\frac{\partial}{\partial n} \left(\sum_{h=1}^{K} a_h^j T_0^{-1} \xi_h \right) - \frac{\partial u_0}{\partial n} \right) \xi_k d\Gamma, \quad k = 1 \cdots K. \end{aligned} \tag{31}$$

From Eq. (31), it appears evident that at each iteration one evaluation of the normal derivative of the harmonic function $\tilde{u}^j = \sum_{h=1}^{K} a_h^j T_0^{-1} \xi_h$ is required to compute the actual value of the right-hand side of Eq. (31). If we set $g^j = \sum_{h=1}^{K} a_h^j \xi_h$, we can first solve M independent discrete Helhmoltz problems of the form: find $\tilde{u}^{j,i} \in \mathbf{P}_N$ such that

$$\begin{aligned} l_N(\tilde{u}^{j,i}, v_0^i) &= 0 \quad \forall v_0^i \in \mathbf{P}_N^{0,i} \\ \tilde{u}^{j,i} &= g^j \quad \text{at the GL nodes of } \partial\Omega_i \cap \Gamma \\ \tilde{u}^{j,i} &= 0 \quad \text{at the GL nodes of } \partial\Omega_i \cap \Omega. \end{aligned} \tag{32}$$

Then we compute the normal derivatives of $\tilde{u}^{j,i}$ on $\partial\Omega_i \cap \Gamma$ to generate the integrands in (31). Finally, we replace each integral with its Gauss–Lobatto approximation on each side of Γ.

In summary, the iterative scheme is advanced until convergence as follows:

A. Given:

1. the coefficients b_k of the jump of the normal derivative $\partial u_0/\partial n$ along Γ,

2. the coefficients a_k^0 of a guess solution of $\tilde{u}$ along Γ: $j = 0$ (j being the iterations counter).

B. Compute:
$j = j + 1$;

1. the harmonic extension of the function defined along Γ by its a_k^j by solving (32);

2. the coefficient of the jump of the normal derivative $\partial\tilde{u}^j/\partial n$ along Γ.

C. Update the coefficient of the solution through a step of (31).

D. Goto **B** until convergence of the a_k's.

Remark 1. Matrix **A** of Eq. (30) is symmetric because it is obtained from discretization of a self-adjoint problem using same test and trial functions. This allows for the use of a conjugate gradient scheme rather than a Richardson one (31).

Remark 2. The convergence rate of the conjugate gradient procedure strongly depends on the choice of the basis $\{\xi_i\}$. For the present work the basis functions proposed by Ovtchinnikov [17] have been used. These consistute a nearly optimal basis, in the sense that the condition number of system (30) is bounded by a constant independent of K, where K is the dimension of the subspace of S generated by span$\{\xi_i\}$, $i = 1, \ldots, K$. Details about the construction of such a set of optimal basis can be found in [2].

Remark 3. When multiple solutions of Eq. (30) are required (for example, in the case of nonstationary elliptic problems like the heat equation or the incompressible Navier–Stokes equation) the efficiency of the procedure adopted to solve the interface problem becomes a key point for the efficiency of the whole algorithm. In this case one can either rely upon construction of the inverse of $\boldsymbol{A}$ in a preprocessing stage using the given iterative procedure (see Section 5), or one can build an efficient preconditioner to accelerate the conjugate gradient algorithm. Here we did not investigate the selection of an efficient one, but the family of preconditioners obtainable through the probing technique of Chan and Keyes [18] seems to suit the actual formulation of the interface problem. In fact, the probing method allows for generating small band preconditioners (typically tridiagonal ones) containing a lumping of the Shur complement **A** action onto the band of the preconditioner (i.e., the tridiagonal part). The combined use of optimal basis and probing techniques should provide an efficient preconditioner. In fact, the former makes the conditioning number of **A** independent of the number of unknowns on the interface, while the latter would reduce its spectral radius. The use of this class of methods would become compulsory when dealing with real 3D problems where storage of the whole exact inverse of **A** soon becomes prohibitive.

4. EXTENSION TO THE NAVIER–STOKES EQUATIONS

As already mentioned, when the incompressible Navier–Stokes equations are solved in the framework of a pressure correction scheme (4), (7) with the diffusive part treated implicitly, the time-advancing technique consists in the recursive solutions of Helmholtz-like problems.

One of the most attractive features of the continuous pressure correction scheme is the possibility of choosing the same degree of approximation for both pressure and velocity. Indeed, from our numerical experience it turned out that with this technique there is no need to satisfy any form of compatibility condition (i.e., Brezzi–Babũska condition). Such a feature simplifies the multidomain implementation since pressure and velocity share the same grid nodes on the interface, thus avoiding complicated interpolation procedures. The two aforementioned features make the global scheme very efficient and easy to implement.

When the projection decomposition algorithm is applied to achieve the solution of each scalar elliptic problem, at the end of each time step the velocity field fails to be globally continuous in Ω, we simply have $u \in L^2(\Omega)$. Indeed, the projection step (8) updates the solenoidal velocity field trough a gradient of a scalar field, $p^{n+1} - p^n$, which is only $C^0(\Omega)$, but not $C^1(\Omega)$, as the normal derivatives can have jumps at the interfaces. Even though the spectral collocation method guarantees an exponential decay of the mismatching of the solenoidal velocity field at the interfaces, a coarse discretization might introduce numerical

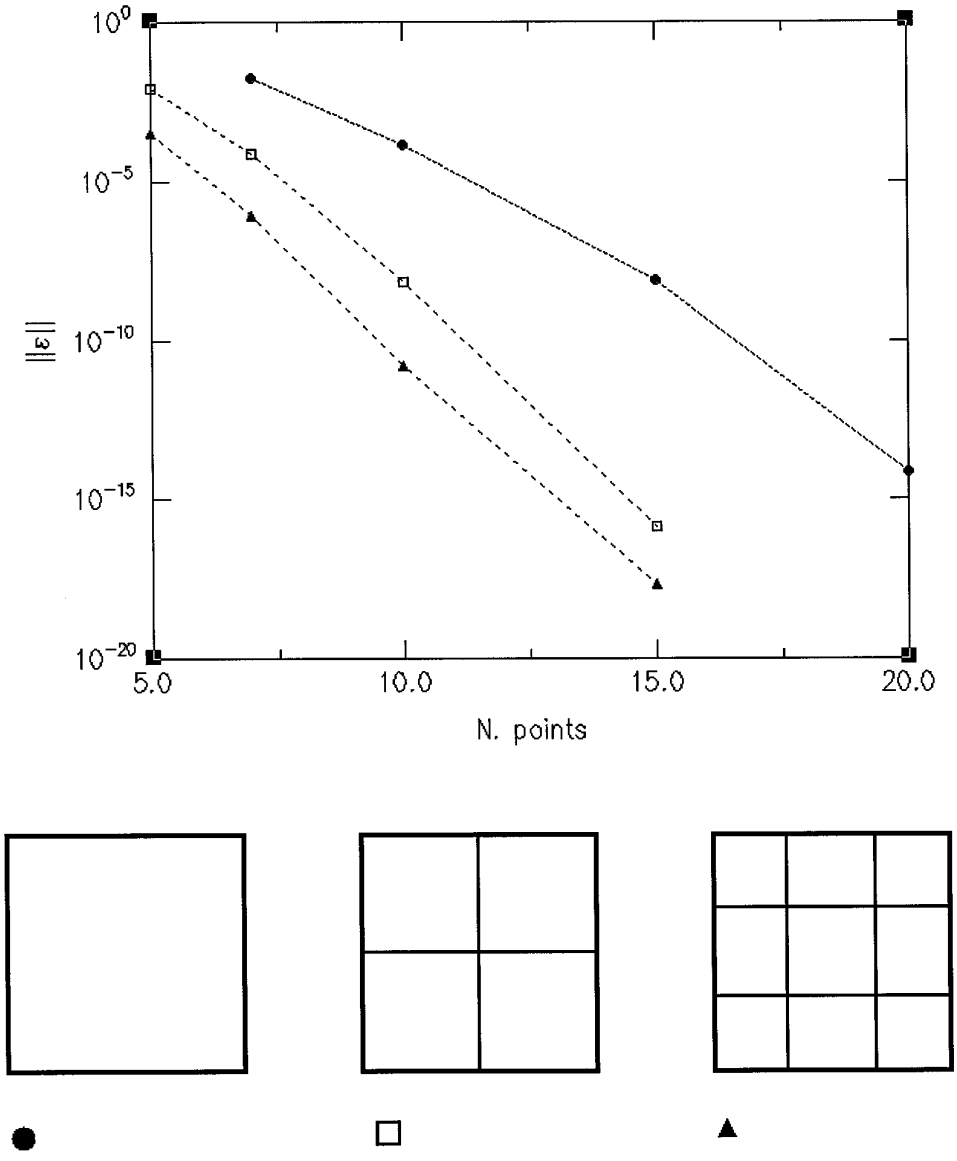

FIG. 1. $H^1(\Omega)$ error for different domain partitions.

boundary layers at the interfaces, leading to catastrophic instabilities.

To avoid such a problem it is sufficient to rely on a *weak* collocation updating rather than on a *strong* one (more details can be found in [8]). To make the point clear the one-dimensional equation $u^{n+1} = \hat{u} + dp/dx$ is considered applied on an interval $[0, L]$, split in two subintervals $[0, L] = [0, \Gamma] \cup [\Gamma, L]$. The weak projection step will then read as

$$\int_0^L u^{n+1} La_k dx = \int_0^L \tilde{u} La_k dx + \int_0^L \frac{dp}{dx} La_k dx, \qquad (33)$$

where the test functions are the already mentioned Lagrange polynomials $La_k(x)$ constructed on the Gauss–Lobatto (GL) nodes. Exploiting the Gauss quadrature formula and the definition of the Lagrange polynomials the final form of the updating reads as

- for the internal nodes in $[0, \Gamma[$

$$u^{n+1}\omega_j^1 = \hat{u}\,\omega_j^1 + \frac{dp}{dx}\,\omega_j^1;\ \omega_j^1 \text{ being the } j^{\text{th}} \text{ GL quadrature weight of interval } [0, \Gamma[;$$

- for the internal nodes in $]\Gamma, L]$

$$u^{n+1}\omega_j^2 = \hat{u}\,\omega_j^2 + \frac{dp}{dx}\,\omega_j^2;\ \omega_j^2 \text{ being the } j^{\text{th}} \text{ GL quadrature weight of interval }]\Gamma, L];$$

• on the interface node Γ

$$u_\Gamma^{n+1}(\omega_N^1 + \omega_1^2) = \hat{u}_\Gamma(\omega_N^1 + \omega_1^2) + \left(\frac{dp^1}{dx}\omega_N^1 + \frac{dp^2}{dx}\omega_1^2\right).$$

Such a procedure is of straightforward extension to a two-dimensional domain, and in this case it can also be extended to include the treatment of internal corners.

One more comment is deserved by the treatment of the convective terms of the Navier–Stokes equations. As pointed out by Browning *et al.* [19], finite difference discretization in a simple patching context might lead to inaccurate results or even numerical instabilities. This problem might be of paramount importance when dealing with direct numerical simulation (DNS) of turbulent flows in regions where the viscosity plays a modest role. In the finite difference case the problem is encountered because different grids display different dispersion properties that might induce partial reflection or cancellations of waves crossing the interfaces. In the spectral case the extremely small errors associated with the end points guarantees the clean passage of waves across the interface if enough resolution is provided. To show evidence of the last statement, the solution of a linear transport equation applied to a two-dimensional Gaussian pulse has been considered:

$$\frac{\partial\theta}{\partial t} + u\frac{\partial\theta}{\partial x} + v\frac{\partial\theta}{\partial y} - k\nabla^2\theta = f(x,y) \quad \text{in } \Omega,\ \theta = \theta_{\text{exact}} \text{ on } \partial\Omega. \tag{34}$$

The diffusive coefficient k has been fixed to a very low value ($k = 10^{-8}$) and the forcing $f(x, y)$ is selected to make $\theta_{\text{exact}} = e^{-2[(x-u\,t)^2+(y-v\,t)^2]}$ the exact solution of the given differential problem (i.e., to balance the diffusive term). The convective speed $U = (u, v)$ has been set to $(1/\sqrt{2}, 1/\sqrt{2})$ to keep into account eventual problems due to non-grid aligned advection. The exact solution is imposed as the time-dependent boundary condition at the domain ($\Omega = (1, 0) \times (0, 1)$) edges. The initial condition had a quarter of the pulse inside the domain and the solution was advanced in time until reaching the condition with three-quarters of the pulse outside the domain. Keeping the time step very small to make the time errors negligible, the $H^1(\Omega)$ norm of the error is measured against the number of points per direction per subdomain. Figure 1 displays the results, together with the domain partitioning configurations. Clearly, the error decreases exponentially fast as the polynomial degree N is increased within each subdomain, testifying to a spectral behaviour. Of course, since the method can be considered as a spectral counterpart of the finite element *p version*, increased accuracy is obtained by increasing the order of the internal polynomials rather than by resorting to a further partitioning [20]. In fact, the last strategy would lead only to an algebraic rate of convergence [21]. Table I summarises the same test for the infinity norm of the error measured both for the internal points and for the interface, showing how the error is uniformly distributed.

Figure 2 highlights the clean passage of the pulse through an internal corner; neither reflections nor spurious modes are detectable. Finally, as a test of accuracy of the whole Navier–Stokes algorithm, the classical Taylor–Green analytical solution of the two-dimensional equations has been considered. The exact solution reads

$$u(x,y) = -\cos(\pi x)\sin(\pi y)e^{-t/2\pi^2} \tag{35}$$

$$v(x,y) = \sin(\pi x)\cos(\pi y)e^{-t/2\pi^2} \tag{36}$$

$$p(x,y) = -1/4\,(\cos(2\pi x) + \cos(2\pi y))e^{-t\pi^2} \tag{37}$$

on the domain $\Omega = (0, 2) \times (0, 2)$. The following set of boundary conditions are applied:

• on the edges $x = 0$ and $x = 2$, homogeneous Dirichlet conditions for u and homogeneous Neumann conditions for v.

• on the edges $y = 0$ and $y = 2$, homogeneous Dirichlet conditions for v and homogeneous Neumann conditions for u.

The tests have concerned both time and space accuracy. Again, the latter has been measured, imposing an extremely small value for the time step. A four-subdomain of equal surface configuration has been considered and

TABLE I

Number of points per subdomain	L_∞ error internal nodes **4 subdomains**	L_∞ error interface nodes **4 subdomains**	L_∞ error internal nodes **9 subdomains**	L_∞ error interface nodes **9 subdomains**
5×5	8.1×10^{-2}	3.1×10^{-2}	1.6×10^{-2}	1.1×10^{-2}
7×7	7.9×10^{-3}	4.5×10^{-3}	4.1×10^{-4}	3.3×10^{-4}
10×10	3.5×10^{-5}	1.0×10^{-5}	7.4×10^{-7}	5.3×10^{-7}
15×15	4.1×10^{-7}	1.1×10^{-7}	5.9×10^{-8}	4.4×10^{-8}

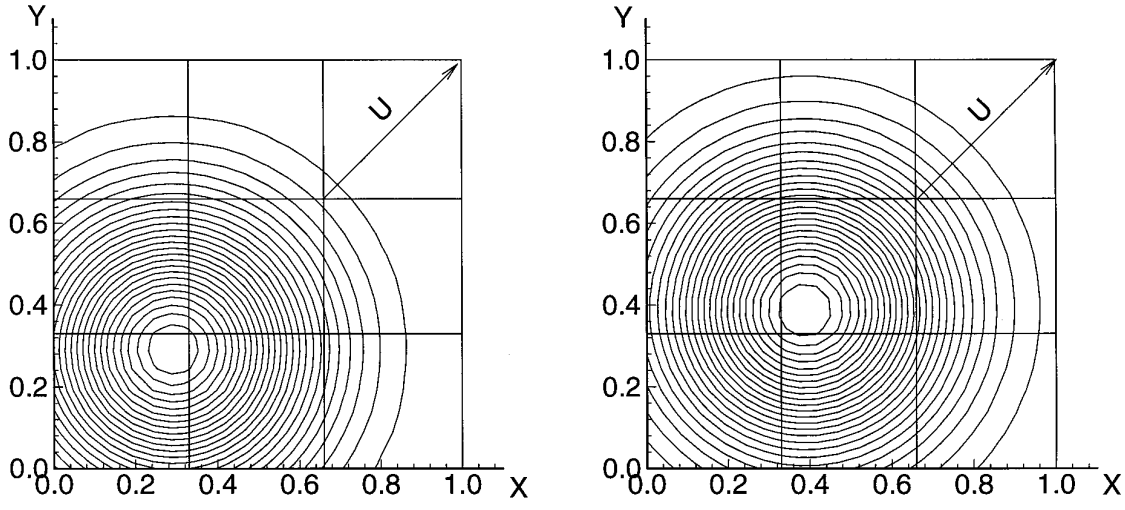

FIG. 2. Passage of the Gaussian pulse through the interface (left, time = T_0; right, time = $T_0 + \Delta T$).

the error has always been measured according with the $H^1(\Omega)$ norm. Table II, showing the results of different polynomial degrees per subdomain, summarises the accuracy measurements for one of the velocity components. From the given results, spectral convergence of the solution is quite evident.

In order to measure the time accuracy of the present scheme, we considered the same test case with a prescribed discretization in space (4 subdomains 14×14 nodes each) sufficient to deliver full spatial accuracy. In Table III we present the relative $L^2(\Omega)$ norm of the velocity error achieved after 1 time unit. From the results it turns out that the adopted scheme is indeed second order in time for the velocity.

5. EFFICIENCY ENHANCEMENTS OF THE FRACTIONAL STEP ALGORITHM

The key to efficiency of any fractional step type of algorithm is the solution procedure for the elliptic kernels arising from the time discretization. In the present case both the solution of Helmholtz boundary value problems on the whole domain Ω, and solutions of the same type of differential subproblems over each subdomain Ω_i need to be solved repetitively.

As concerns the latter, meaning the decoupled Dirichlet problems involved in the PDM procedure, it is possible to apply a diagonalization procedure [22]. The Legendre collocation approximation to one of the mentioned subproblems can be recast as

$$\mathbf{UD} + \mathbf{D}^{\mathrm{T}}\mathbf{U} - \alpha\mathbf{IU} = \mathbf{F}. \tag{38}$$

where $\mathbf{D}$ is the collocated Lagrange second derivative matrix acting on the subdomain internal nodes, $\mathbf{U}$ is the unknown matrix ordered by rows, and $\mathbf{F}$ is a modified right-hand side matrix keeping into account the effects of boundary values. In a first stage, the eigenvalues of $\mathbf{D}$, its left and right eigenvector system (ordered by columns) and the respective inverses are determined:

$$\mathbf{E}_{\mathrm{r}}^{-1}\,\mathbf{D}\,\mathbf{E}_{\mathrm{r}} = \mathbf{\Lambda} \tag{39}$$

$$\mathbf{E}_{\mathrm{l}}^{-1}\,\mathbf{D}^{\mathrm{T}}\,\mathbf{E}_{\mathrm{l}} = \mathbf{\Lambda}. \tag{40}$$

Matrices $\mathbf{E}_{\mathrm{r}}$, $\mathbf{E}_{\mathrm{l}}$, $\mathbf{E}_{\mathrm{r}}^{-1}$, $\mathbf{E}_{\mathrm{l}}^{-1}$ and the diagonal eigenvalue matrix $\mathbf{\Lambda}$ are computed and stored in a preprocessing phase. Indicating with $\hat{\mathbf{U}} = \mathbf{E}_{\mathrm{r}}^{-1}\,\mathbf{U}\,\mathbf{E}_{\mathrm{l}}$ and with $\hat{\mathbf{F}} = \mathbf{E}_{\mathrm{r}}^{-1}\,\mathbf{F}\,\mathbf{E}_{\mathrm{l}}$ the diagonalized problem,

$$\mathbf{\Lambda}\hat{\mathbf{U}} + \hat{\mathbf{U}}\mathbf{\Lambda} - \boldsymbol{\alpha}\hat{\mathbf{U}} = \hat{\mathbf{F}}, \tag{41}$$

is inverted and the final solution is recovered as

$$\mathbf{U} = \mathbf{E}_{\mathrm{r}}\hat{\mathbf{U}}\mathbf{E}_{\mathrm{l}}^{-1}. \tag{42}$$

Having solved the eigenvalue problem in a preprocessing stage, the recursive solution cost turns out to be of order N^3 operations, N being the number of nodes used to discretized each direction within a single subdomain.

TABLE II

Pol. degree ($N_x = N_y$)	Total nodes	H^1 error for velocity
6	144	$4. \times 10^{-4}$
9	324	$1. \times 10^{-7}$
13	676	$9. \times 10^{-13}$

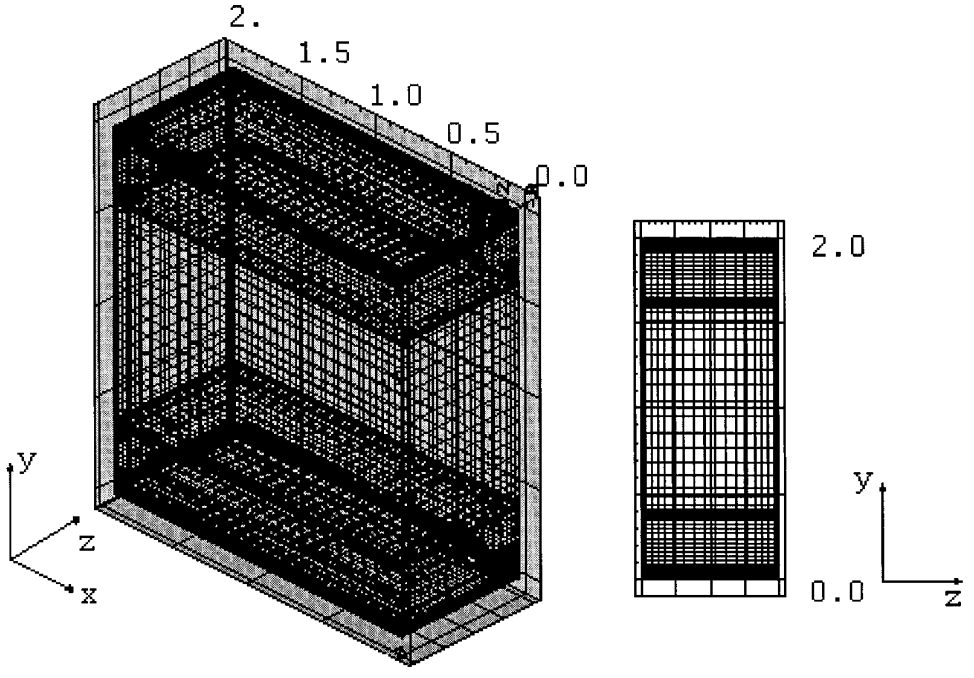

FIG. 3. Pouiselle flow grid configuration. On the right a view of a plane normal to the mean flow.

As concerns the solution of the whole domain Ω it is possible to determine (always as a preprocessing procedure) the inverse of the operator $\mathbf{A}$ (see (30)), handling the interface problem. With reference to Section 3, Problem P1 and the differential problem leading to the solution on the interface P2 are reconsidered hereafter as

$$\mathbf{A}a_k = b_k. \tag{43}$$

Here the a_k's refer to the Galerkin coefficients of the solution on the interface, and the b_k's are the Galerkin coefficients of the jump of the normal derivatives produced by the solution of the M Problems P1.

Next the K problems

$$\mathbf{A}a_k = \delta_{kj}, \quad k = 1 \cdots K \tag{44}$$

are considered, meaning problems with a jump of the normal derivatives leading to a unitary Galerkin coefficient k and zero values for all the other coefficients j ($j \neq k$). Successive inversions, through the iterative procedure outlined in Section 3, allow for constructing by columns the operator $\mathbf{A}^{-1}$. The latter might, then, be considered as the inverse of the Schur–Galerkin complement that, applied to the Galerkin coefficients of the computed normal derivatives jumps (Problem P1), release the coefficients of the solution on the interface to guarantee zero weak normal derivative jumps between subdomains. It is also remarked that matrix $\mathbf{A}^{-1}$ is symmetric because it is obtained from the discretization of a self-adjoint problem (22). This property leads to an evident storage and operation count reduction.

TABLE III

Time step size	L^2 error for velocity
0.1	$4. \times 10^{-2}$
0.01	$5. \times 10^{-4}$
0.001	$3. \times 10^{-6}$
0.0001	$5. \times 10^{-8}$

6. THREE-DIMENSIONAL EXTENSION

In this section, a generalisation of the method which allows for the simulation of three-dimensional flows with one periodic direction is presented. For this class of flows it is possible to take advantage of the classical Fourier decomposition of the flow variables in the periodic direction. This choice leads to reduce all the three-dimensional scalar differential problems in the physical space (momentum equations and pressure correction equation) into a sequence of two-dimensional scalar differential problems in terms of the transformed variables. Once the two-dimensional problems are set up, it is possible to take advantage of the projection decomposition method to solve them efficiently.

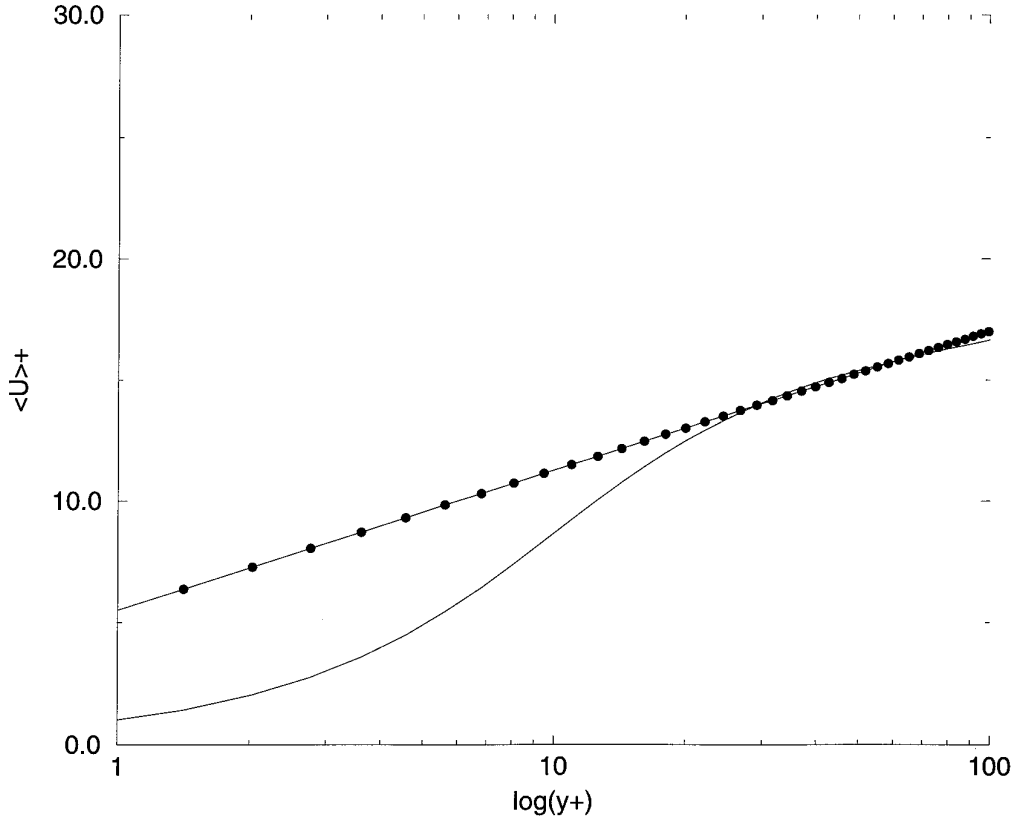

FIG. 4. Mean streamwise velocity near the wall. Actual computations (solid lines) plotted against logarithmic wall law ($u/u_\tau = 2.5 \log(y^+) + 5$) (●).

In particular, let

$$u_i^n(x,y,z) = \sum_{k=-N/2}^{N/2-1} \tilde{u}_{i,k}^n(x,y) e^{Ikz}, \quad i = 1,2,3, \tag{45}$$

$$p^{n+}(x,y,z) = \sum_{k=-N/2}^{N/2-1} \tilde{p}_k^{n+1}(x,y) e^{Ikz}, \tag{46}$$

$$\hat{u}_i(x,y,z) = \sum_{k=-N/2}^{N/2-1} \tilde{\hat{u}}_{i,k}(x,y) e^{Ikz}, \quad i = 1,2,3, \tag{47}$$

and

$$\delta\hat{u}_i(x,y,z) = \sum_{k=-N/2}^{N/2-1} \delta\tilde{\hat{u}}_{i,k}(x,y) e^{Ikz}, \quad i = 1,2,3, \tag{48}$$

with $I = \sqrt{-1}$. Applying the same methodology as for the two-dimensional case, the three-dimensional algorithm can be reformulated as

For every $n = 0, 1, \ldots$ (n being the time counter):

1. For $i = 1, 2, 3$, solve for $\tilde{\hat{u}}_{i,k}$ (the predicted velocity field) the momentum equations (4), for $k = -N/2, \ldots, N/2 - 1$,

$$\left(-\tilde{\nabla}^2 + k^2 + \frac{2\mathrm{Re}}{\Delta t}\right)\delta\tilde{\hat{u}}_{i,k} = \tilde{rhs}_{i,k}, \quad l = 1,2; \tag{49}$$

2. Solve for $\tilde{p}_k^{n+1}$ the pressure correction (7) for $k = -N/2, \ldots, N/2 - 1$,

$$\left(-\tilde{\nabla}^2 + k^2\right)\tilde{p}_k^{n+1} = -\frac{2}{\Delta t}\frac{\partial\tilde{\hat{u}}_{l,k}}{\partial x_l} + \frac{2Ik}{\Delta t}\tilde{\hat{u}}_{3,k} + \left(-\frac{\partial^2}{\partial x_l^2} + k^2\right)\tilde{p}_k^n; \tag{50}$$

3. For $i = 1, 2, 3$, update the velocity field, as in (7), for $k = -N/2, \ldots, N/2 - 1$,

$$u_{i,k}^{n+1} = \tilde{\hat{u}}_{i,k} - \frac{\Delta t}{2}\begin{cases} \dfrac{\partial(\tilde{p}_k^{n+1} - \tilde{p}_k^n)}{\partial x_i}, & \text{if } i = 1,2, \\ Ik.\tilde{p}_k^{n+1}, & \text{otherwise.} \end{cases} \tag{51}$$

The subscript l has been introduced to stress the fact that the collocated derivatives are computed in the two nonperiodic directions only. The term $\tilde{rhs}_{i,k}$ represents the kth mode of the transform of the right-hand side of the ith

momentum equation, and the operator $\tilde{\nabla}^2$ represents the two-dimensional laplacian operator in the two nonperiodic directions. The treatment of the boundary conditions is straightforward and does not introduce any supplementary difficulty.

6.1. DNS OF A TURBULENT CHANNEL FLOW

The DNS of low Reynolds numbers fully turbulent Poiseuille flows can nowadays be considered a classical test case since the literature is very rich in both experimental and numerical results. This flow [23] might be considered periodic both in stream and spanwise directions if the dimensions of the computational box are made large enough. In the present case the streamwise direction (x-direction) is taken as the Fourier one, while, to impose periodicity in the spanwise direction (z-direction) the edges of the subdomains normal to z are virtually joined to generate an interface. This last procedure guarantees spanwise weak periodicity of the solution in the sense that the derivatives normal to the planes delimiting the computational box in z will only match in a weak sense. This procedure allows us to test the effectiveness of the treatment of the internal corners as well. All the lengths are made nondimensional with the channel half-height h, and the velocity is nondimensionalized with the centerline velocity U_c. With this selection the Reynolds number $\mathrm{Re} = U_c\, h/\nu$ has been fixed to the value of 3850 to be able to compare quantitatively with the benchmark simulation of Kim *et al.* [24]. Following the minimal flow unit approach proposed by Jiménez and Moin [23] the dimensions of the computational box are fixed to (2, 2, 0.8) in streamwise, normal to the wall (y-direction) and spanwise directions, respectively. The grid configuration, together with a section normal to the mean flow, is displayed in Fig. 3). It consists of five subdomains; the dimensions of the first and of the last are selected to guarantee full capture of the wall sublayer (height of about 20 wall units). Each subdomain contains 20×20 nodes, while in the Fourier direction 36 modes are employed. The present case has been run on an IBM RS6000 360H workstation with about 100 Mflops peak performance. The cpu required for each full-time iteration is about 4.5 s when the inverse of the Schur–Galerkin complement is computed and stored in a preprocessing stage.

After having reached a statistical steady state some typi-

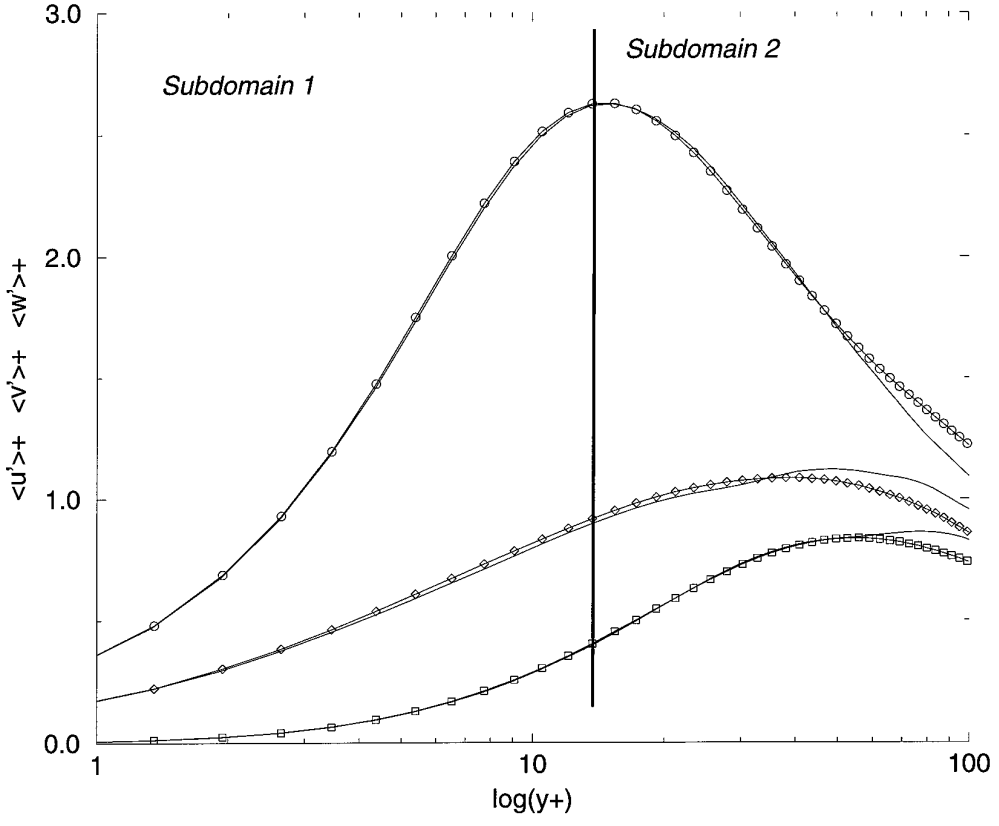

FIG. 5. Root mean square velocity fluctuations normalized with the wall shear velocity. Solid lines (present computation); symbols from Kim *et al.* [24], respectively (○) streamwise velocity component fluctuations, (□) vertical velocity component fluctuations ($\overline{v^2}^{1/2}$), and (◇) spanwise velocity component fluctuations.

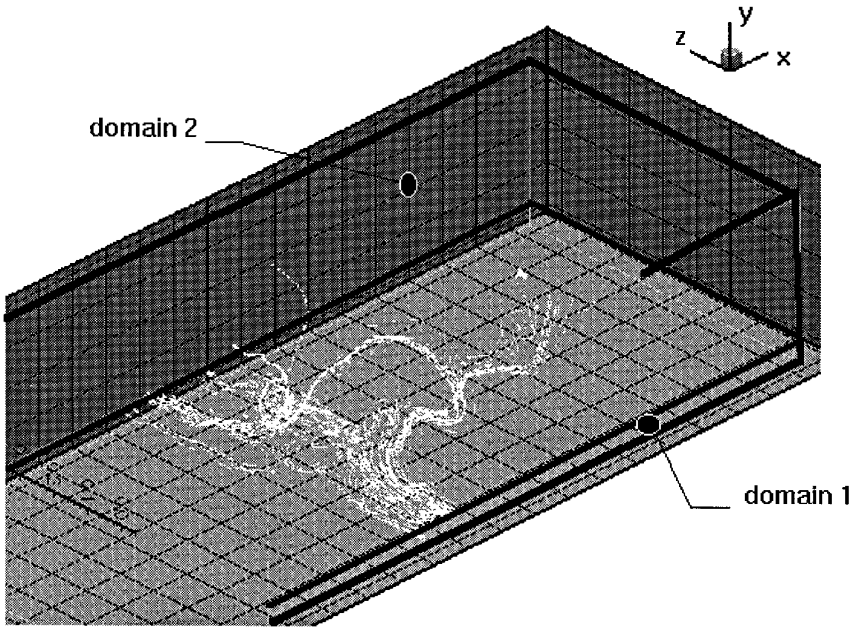

FIG. 6. Selected instantaneous vortex lines in a turbulent Pouiselle flow.

cal turbulent quantities have been measured to assess the quality of the obtained results. In Fig. 4 the obtained mean velocity profile is plotted against the logarithmic wall ($u/u_\tau = 2.5 \log (y^+) + 5$). In Fig. 5 the computed turbulence intensities are compared with the data of Kim *et al.* [24] at the same Reynolds number ($\mathrm{Re}_\tau = 180$). As expected, the results match perfectly the ones of [24] in the wall region, while, as an effect of the minimal flow unit approach, the velocity fluctuations are lower in the core region [23]. Finally, in Fig. 6 the instantaneous distribution of selected vortex lines highlights the presence of a typical hairpin vortex within the computational box.

Apparently both the statistical quantities and the qualitative behaviour of the flow field are unaffected by the underlying domain partitioning. This indicates that the present method might allow for simulations of turbulent flows over complex geometries.

7. CONCLUSIONS

The present work has been concerned with the solution of the unsteady incompressible Navier–Stokes equations, using a high-order collocated spectral multidomain method. The rationale behind the choice and development of the method is given both by the natural way in which a parallel implementation of the present algorithm can be achieved and by the possibility of coupling the potentially high accuracy of spectral methods with the flexible framework offered by multidomain methods. In particular, the present multidomain method is of great generality since its foundation is completely independent of both the numerical discretization selected for each subdomain and of the way the domain partition is made. This property should then allow for relatively cheap DNS and LES of internal flows with complex boundary shapes, such as turbulent flows in rough channels. Indeed, the data from the channel DNS simulation seems to confirm the viability of the present algorithm to deal with complex turbulent flow configurations. At the same time it should be stressed that the capability of selecting the accuracy in determined flow regions might reveal that it is a powerful tool for resolved large eddy simulations in complex configurations (i.e., when approximate wall conditions are not available).

ACKNOWLEDGMENTS

The first two authors are grateful for the support and the computer time provided by IRSIP (Istituto Richerche e Sistemi Informatici Parallel CNR, Napoli, Italy) and, in particular, to Professor Vaccaro. They are also indebted to Professor Jiménez of the Polytechnic University of Madrid for constructive discussions and support. The first author acknowledges the European Union financial support through the HCM programme. The third author acknowledges the Sardinia Regional Government, the C.N.R. Funds (Special Project, ''Fluid and Molecular Dynamics'') and the Project ''Computational Mathematics'' of the Italian M.U.R.S.T. (funded 40%).

REFERENCES

1. V. Agoshkov and E. Ovtchinnikov, *Projection Decomposition Method*, Technical Report CRS4, Cagliari, Italy, 1993.

2. P. Gervasio, E. Ovtchinnikov, and A. Quarteroni, The spectral projection decomposition method for elliptic equations, *SIAM J. Numer. Anal*. **34,** 1616 (1997).
3. R. D. Moser, P. Moin, and A. Leonard, *J. Comput. Phys.* **175,** 524 (1983).
4. M. R. Malik, T. A. Zang, and M. Y. Hussaini, *J. Comput. Phys.* **61,** 64 (1985).
5. Y. Maday and A. T. Patera, Spectral element methods for the incompressible Navier–Stokes equations, *in State of the Art Surveys in Computational Mechanics,* edited by A. Noor and J. T. Oden (ASME, New York, 1989).
6. A. T. Patera, *J. Comput. Phys.* **54,** 468–488 (1984).
7. R. Henderson and G. Karniadakis, *J. Comput. Phys.* **122,** 191–217 (1995).
8. A. Quarteroni and A. Valli, *Approximation Methods for Partial Differential Equations* (Springer-Verlag, Heidelberg, 1994).
9. G. De Pietro, A. Pinelli, and A. Vacca, A Parallel Implementation of a Spectral Multi-Domain Solver for the Incompressible Navier-Stokes Equations, in *Parallel CFD Conference 1985* (Elsevier, Amsterdam, 1995).
10. T. Zang. *Appl. Num. Math.* **7,** 27–40 (1991).
11. P. Gresho. *Int. J. for Num. Methods in Fluids,* **11,** 587–602 (1990).
12. A. Chorin, *J. Comput. Phys.* **2,** 12 (1967).
13. C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics* (Springer-Verlag, New York, 1988).
14. J. Van Kan, *J. Sci. Stat. Comput.* **7,** 870 (1986).
15. J. L. Lions and E. Magenes, *Nonhomogeneous Boundary Value Problems and Applications, Vol 1.* (Springer-Verlag Heidelberg 1972).
16. P. Davis and P. Rabinowitz, *Methods of Numerical Integration. (Academic Press New York 1984).*
17. E. Ovtchinnikov, On the Construction of a Well Conditioned Basis for the Projection Decomposition Method. CRS4 Technical Report Cagliari, Italy., 1993.
18. T. F. Chan and D. E. Keyes, Interface Preconditioning for Domain-Decomposed Convection-Diffusion Operators. In T. F. Chan, R. Glowinski, J. Periaux, and O. B. Widlund, editors, *'Third International Symposium on Domain Decomposition Methods for PDEs'*. SIAM Philadelphia, 1990.
19. F. Browning, H. Kreiss, and J. Oliger, *Math. Comp.,* **27,** 29–38 (1973).
20. P. G. Ciarlet, *The Finite Element Method for Elliptic Problems.* (North-Holland Amsterdam 1978).
21. E. Ronquist, ''Spectral Elements Methods for Fluid Dynamics'' (Von Karman Institute, Rhode. St., Genese, Belgium, 1991). [Lecture Series 1991–01].
22. G. Golub and C. Van Loan, *Matrix Computations.* (The Johns Hopkins Univ. Press, 1989). [Series in Math. Sci.]
23. J. Jimenéz and P. Moin, *J. Fluid Mech.* **225,** 213 (1991).
24. J. Kim, P. Moin, and R. Moser. *J. Fluid Mech.* **117,** 133–171 (1987).