PERGAMON

Computers & Fluids 29 (2000) 525–546

computers & fluids

www.elsevier.com/locate/compfluid

A domain decomposition method for simulating advection dominated, external incompressible viscous flows

J.-L. Guermond [a,*], H.Z. Lu [b]

[a] *LIMSI, UPR-CNRS 3251, BP 133, 91403, Orsay, France*

[b] *Laboratoire de Chimie théorique, Département de Chimie, Universite de Sherbrooke, 2500 Boulevard de l'Université Sherbrooke, Quebec, Canada J1K 2R1*

Received 23 October 1997; received in revised form 8 January 1999; accepted 29 March 1999

Abstract

We introduce a domain decomposition method for simulating 2D external, incompressible viscous flows. In each subdomain that is close to a connected physical boundary, the velocity and the pressure are approximated on a body fitted, finite difference grid. In the subdomain that is far from the solid boundary (i.e., the neighborhood of infinity), we develop a characteristics method to approximate the velocity and the vorticity on a Cartesian grid. The two methods are coupled by means of a Schwarz type strategy. This method is tested by simulating the flow past one or two cylinders. Some tests are performed with moving cylinders. Comparisons with numerical and experimental data illustrate the efficiency of the method.

1. Introduction

One feature of external flows is that a boundary condition at infinity must be enforced. When simulating such flows on a fixed computational grid, this condition is often replaced by an artificial boundary condition which is imposed at the external limit of the

* Corresponding author. Tel.: +33-1-6985-8069; fax: +33-1-6985-8088.

E-mail addresses: guermond@limsi.fr (J.L. Guermond), hlu@gauss.chimie.usherb.ca (H.Z. Lu).

0045-7930/00/$ - see front matter
PII: S0045-7930(99)00017-1

simulation domain. Many artificial boundary conditions that minimize reflections at the outer boundary have been proposed in the literature. However, all these techniques assume that the artificial boundary is asymptotically far enough from the physical boundary to guarantee accuracy. Hence, in practice it is difficult to determine a priori where to place the outer boundary (see, e.g., Deuring [8], Quarteroni [17], Charton–Nataf–Rogier [6]).

Mesh adaptation is another issue related to the numerical simulation of advection dominated external flows. Since for such flows the vorticity is generated in thin boundary layers located near the solid boundary and concentrates further downstream in a wake, the outer flow is almost curl free. Hence, an adaptive method should naturally concentrate grid points in the boundary layer and the wake. Moreover, if there are several obstacles immersed in the fluid and if these obstacles are in relative motion, the grid generation of the shape changing flow domain may be a non-trivial task.

To cope with the numerical issues identified above, a domain decomposition strategy has been proposed by Cottet [5] and Guermond et al. [11]. This method consists in adopting a domain decomposition strategy. The fluid domain is decomposed into a set of subdomains. Small subdomains are composed of the immediate vicinity of the solid boundaries, where the viscous effects ensure that the no-slip condition is satisfied. The complement of these small subdomains compose a large subdomain. In the small subdomains that surround the solid boundaries, the Navier–Stokes equations are formulated in the Eulerian coordinates and are solved by using standard finite difference technique. In the large subdomain, the Lagrangian coordinates are adopted and the numerical simulation is carried out by means of a vortex method (cf., e.g., Chorin [3] or Rehbach [18] for details on this technique).

The goal of the present paper is to present a numerical method that follows these ideas. However, the method presented hereafter differs from the one introduced in the references above on three points. First, instead of using the stream-function and the vorticity in the inner subdomains, we propose to solve the Navier–Stokes equations formulated in the primitive variables. The motivation for this choice is that it yields the pressure at the boundary without resorting to an auxiliary computation and the method can be extended in three dimensions quite easily. Second, to avoid the hypothesis of hyperbolic degeneracy of the flow near the subdomains' interface that is made in [11], we adopt an overlapping strategy together with an alternating Schwarz algorithm (cf. Lions [13]); as a result, the transmission condition that we use does not require the viscous diffusion to be dominated by the advection. Third, to guarantee the accuracy of the method, we approximate $\boldsymbol{u}$–ω in the external domain by using a compound strategy combining a characteristics method and the Biot–Savart technique. This approach differs from that of the references above where a classical, low order, vortex method is used.

This paper is organized into five parts. In Section 2, we set some notations and we introduce the two formulations of the Navier–Stokes equations that we use, namely the $\boldsymbol{u}$–p and the $\boldsymbol{u}$–ω formulations. The time discretization is presented in Section 3. The spatial approximations used in each subdomain is reviewed in detail in Section 4. We illustrate the present method and we compare it to other techniques and experimental data in Section 5. Some conclusions are drawn in the last section.

2. Preliminaries

2.1. Two formulations of the Navier–Stokes equations

Let us denote by Ω the fluid domain and B the boundary of a connected solid obstacle that is immersed in the fluid (see Fig. 1). For the sake of simplicity of the presentation, we assume that the flow domain is one-connected; multiply connected domains can be handled similarly. The normal on B and the tangent are denoted by $\boldsymbol{n}$ and $\boldsymbol{\tau}$, respectively. The normal is oriented so that it points inside the obstacle. We define $(\boldsymbol{i},\boldsymbol{j})$ an orthogonal normed basis of $\mathbb{R}^2$ and we embed $\mathbb{R}^2$ into $\mathbb{R}^3$ so that $(\boldsymbol{i},\boldsymbol{j},\boldsymbol{k})$ is an orthogonal normed basis of $\mathbb{R}^3$. The Cartesian coordinates are hereafter denoted by (x, y, z).

In two dimensions, the Navier–Stokes equations modeling the external incompressible viscous flow past one obstacle can be written in terms of velocity, $\boldsymbol{u}$, and pressure, p, as follows:

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla p + \boldsymbol{u}\cdot\nabla\boldsymbol{u} - \nu\Delta\boldsymbol{u} = 0 \quad \text{in } \Omega$$

$$\nabla\cdot\boldsymbol{u} = 0 \quad \text{in } \Omega$$

$$\boldsymbol{u} = \boldsymbol{a} \quad \text{on } B$$

$$\boldsymbol{u}(\boldsymbol{x}) \to U_\infty(t)\boldsymbol{i} \quad \text{for } |\boldsymbol{x}| \to \infty$$

$$\boldsymbol{u}(t=0) = \boldsymbol{u}^0 \tag{1}$$

where ν is the kinematic viscosity of the fluid, $U_\infty\boldsymbol{i}$,$\boldsymbol{a}$ and $\boldsymbol{u}^0$ are the boundary and the initial data. These data are assumed to satisfy the following compatibility conditions:

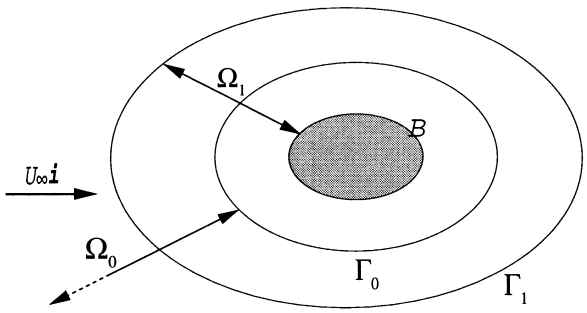

Fig. 1. Sketch of the domain decomposition: the shaded region is a rigid body.

$$\begin{cases} \boldsymbol{u}^0 \cdot \vec{n} = \boldsymbol{a}(t=0) \cdot \vec{n} & \text{on } B \\ \boldsymbol{u}^0(\boldsymbol{x}) \to U_\infty(t=0)\boldsymbol{i} & \text{when } |\boldsymbol{x}| \to \infty \\ \oint_B \boldsymbol{a} \cdot \vec{n} = 0. \end{cases} \tag{2}$$

An alternative formulation of the Navier–Stokes equations consists in adopting the velocity and the vorticity (ω) as dependent variables. This formulation is deduced from the primitive one by applying the curl operator to the momentum equation:

$$\frac{\partial \omega}{\partial t} + \boldsymbol{u} \cdot \nabla \omega = \nu \Delta \omega \quad \text{in } \Omega$$

$$\omega(t=0)\boldsymbol{k} = \nabla \wedge \boldsymbol{u}^0 \tag{3}$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in } \Omega$$

$$\nabla \wedge \boldsymbol{u} = \omega \boldsymbol{k} \quad \text{in } \Omega$$

$$\boldsymbol{u}(\boldsymbol{x}) \to U_\infty(t)\boldsymbol{i} \quad \text{when } |\mathbf{x}| \to \infty$$

$$\boldsymbol{u} = \boldsymbol{a} \quad \textit{on } B \tag{4}$$

$$\oint_B \boldsymbol{\tau} \cdot \left[\frac{\partial \boldsymbol{a}}{\partial t} + \boldsymbol{a} \cdot \nabla \boldsymbol{u} \right] \mathrm{d}l - \nu \oint_B \frac{\partial \omega}{\partial \boldsymbol{n}} \, \mathrm{d}l = 0 \tag{5}$$

The equivalence of the above two formulations is well known; see, e.g., Daube et al. [7] for other details. One important feature of the vorticity is that its support is almost compact, whereas that of the velocity is unbounded. This argument supports the idea that the vorticity can be used to simulate flows in unbounded domains.

2.2. The domain decomposition method

In this work, we adopt a domain decomposition method with overlapping subdomains. The geometrical decomposition of the flow domain is illustrated in Fig. 1: the flow domain Ω is decomposed into two one-connected overlapping subdomains $\Omega_1 \cup \Omega_0$; Ω_1 is a neighborhood of the physical boundary, whereas Ω_0 is a neighborhood of infinity.

3. The time discretization

Let T be a real positive number; we shall seek a semi-discrete approximation of the Navier–Stokes equations in the time interval $[0,T]$ by means of a time-marching algorithm. We introduce a time discretization as follows. Let N be a positive integer; we denote by $\delta t = T/N$

the time step and we set $t_n = n\delta t$, $t_{n-1/2} = (n-1/2)\delta t$. Now assuming that $(\boldsymbol{u}_1^k, p_1^{k-1/2})_{k=0}^n$ and $(\boldsymbol{u}_0^k, \omega_0^k)_{k=0}^n$ are known approximations of $(\boldsymbol{u}(t_k), p(t_{k-1/2}))_{k=0}^n$ and $(u(t_k), \omega(t_k))_{k=0}^n$ in the subdomains Ω_1 and Ω_0, we shall build approximations $(\boldsymbol{u}_1^{n+1}, p_1^{n+1/2})$ and $(\boldsymbol{u}_0^{n+1}, \omega_0^{n+1})$ of $(\boldsymbol{u}(t_{n+1}), p(t_{n+1/2}))$ and $(\boldsymbol{u}(t_{n+1}), \omega(t_{n+1}))$ in Ω_1 and Ω_0, respectively.

3.1. The approximation of $(\boldsymbol{u}(t_{n+1}), \omega(t_{n+1}))$ in Ω_0

We use the $\boldsymbol{u}$–ω formulation (Eqs. (3)–(5)) to approximate $(\boldsymbol{u}(t_{n+1}), \omega(t_{n+1}))$ in Ω_0. The approximation ω_0^n being known only on Ω_0, we extend ω_0^n to the whole fluid domain as follows:

$$\tilde{\omega}_0^n = \begin{cases} \omega_0^n & \text{in } \Omega_0 \\ \omega_1^n & \text{in } \Omega_1 \backslash \Omega_0 \end{cases}. \tag{6}$$

With $\tilde{\omega}_0^n$ as an approximation of $\omega(t_n)$ in the whole fluid domain Ω, we can exploit Eq. (4) to deduce an approximation of $u(t_n)$. By introducing the Green function ($G(\boldsymbol{x}) = \log(|\boldsymbol{x}|)/2\pi$) of the Laplacian in two dimensions, we obtain:

$$\begin{aligned} \tilde{\boldsymbol{u}}_0^n(\boldsymbol{x}) &= U_\infty^n \boldsymbol{i} + \int_\Omega \tilde{\omega}_0^n(\boldsymbol{y}) \boldsymbol{k} \wedge \nabla_y G(\boldsymbol{x}-\boldsymbol{y})\, \mathrm{d}\boldsymbol{y} \\ &+ \oint_B \nabla_y G(\boldsymbol{x}-\boldsymbol{y}) \big[\boldsymbol{n} \cdot \boldsymbol{a}(\boldsymbol{y}, t_n)\big]\, \mathrm{d}l_y; \quad \text{for } \boldsymbol{x} \in \Omega \\ &- \oint_B \nabla_y G(\boldsymbol{x}-\boldsymbol{y}) \wedge \big[\boldsymbol{n} \wedge \boldsymbol{a}(\boldsymbol{y}, t_n)\big]\, \mathrm{d}l_y \end{aligned} \tag{7}$$

This formula is commonly referred to in the literature as the Biot–Savart law. An alternative method of approximation of $\boldsymbol{u}(t_n)$ consists in using the Biot–Savart integral only in Ω_0 as follows:

$$\tilde{\boldsymbol{u}}_0^n(\boldsymbol{x}) = \begin{cases} \boldsymbol{u}_1^n(\boldsymbol{x}) & \text{for } \boldsymbol{x} \in \Omega_1 \\ \left[\begin{array}{l} U_\infty^n \boldsymbol{i} + \displaystyle\int_{\Omega_0} \tilde{\omega}_0^n(\boldsymbol{y}) \boldsymbol{k} \wedge \nabla_y G(\boldsymbol{x}-\boldsymbol{y})\, \mathrm{d}\boldsymbol{y} \\ + \displaystyle\oint_{\Gamma_0} \nabla_y G(\boldsymbol{x}-\boldsymbol{y}) \big[\boldsymbol{n} \cdot \boldsymbol{u}_1^n(\boldsymbol{y})\big]\, \mathrm{d}l_y \\ - \displaystyle\oint_{\Gamma_0} \nabla_y G(\boldsymbol{x}-\boldsymbol{y}) \wedge \big[\boldsymbol{n} \wedge \boldsymbol{u}_1^n(\boldsymbol{y})\big]\, \mathrm{d}l_y \end{array} \right] & \text{for } \boldsymbol{x} \in \Omega_0 \backslash \Omega_1 \end{cases} \tag{8}$$

Since Eq. (8) is cheaper to compute, it has been retained in the computer code that we have developed to test the present method.

With initial data on the vorticity and the velocity field defined everywhere in the flow field, we can use a characteristic method to obtain an approximation of $\omega(t_{n+1})$ in Ω_0 from Eq. (3). For instance, a first-order approximation is given by:

$$\omega_0^{n+1}(\boldsymbol{x}) = \tilde{\omega}_0^n\big(\xi(\boldsymbol{x},t_{n+1};t_n)\big) + \nu\delta t\Delta\tilde{\omega}_0^n(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \Omega_0.$$

Here, $\xi(\boldsymbol{x},t;s)$ is the solution of the ordinary differential equation:

$$\begin{aligned} &\frac{\mathrm{d}\xi}{\mathrm{d}s} = \tilde{\boldsymbol{u}}_0^n(\xi) \\ &\xi(\boldsymbol{x},t;t) = \boldsymbol{x} \end{aligned}.$$

One simple approximation of $\xi(\boldsymbol{x},t_{n+1};t_n)$ is, for instance:

$$\xi(\boldsymbol{x},t_{n+1};t_n) \simeq \boldsymbol{x} - \delta t\tilde{\boldsymbol{u}}_0^n(\boldsymbol{x}).$$

The method of characteristics is introduced in detail in Refs. [9,16]. To obtain second-order accuracy in time we use the following approximation for all $\boldsymbol{x}$ in Ω_0:

$$\omega_0^{n+1}(\boldsymbol{x}) = \frac{4}{3}\tilde{\omega}^n\big(\xi(\boldsymbol{x},t_{n+1};t_n)\big) - \frac{1}{3}\tilde{\omega}^{n-1}\big(\xi(\boldsymbol{x},t_{n+1};t_{n-1})\big) + \frac{2\nu}{3}\delta t\Delta\Big(2\tilde{\omega}_0^n - \tilde{\omega}_0^{n-1}\Big)(\boldsymbol{x}), \tag{9}$$

where $\xi(\boldsymbol{x},t;s)$ is the solution of the ordinary differential equation:

$$\begin{cases} \dfrac{\mathrm{d}\xi}{\mathrm{d}s} = 2\tilde{\boldsymbol{u}}_0^n(\xi) - \tilde{\boldsymbol{u}}_0^{n-1}(\xi) \\ \xi(\boldsymbol{x},t;t) = \boldsymbol{x} \end{cases}. \tag{10}$$

As a result, one obtains the approximation ω_0^{n+1} of $\omega(t_{n+1})$ in Ω_0. Note that we have been obliged to extend the definition of ω_0^n to the whole fluid domain (or at least in a domain sufficiently larger than Ω_0), so that the value of the vorticity can be defined at the foot of every characteristics $\xi(\boldsymbol{x},t_{n+1};t_n)$. Furthermore, the extension procedure allows us to treat the boundary value problem for ω_0 in Ω_0 as an initial value problem in a larger domain. It can be shown that the error induced by the explicit treatment of the viscous diffusion is large only in a numerical boundary layer located at the boundary of the extended computational domain; hence, if $\Omega_1 \setminus \Omega_0$ is large enough, the numerical boundary layer is located outside of Ω_0. In the case of a moving obstacle, we have to set the time step δt small enough, so that the domain where the method of characteristics is used is larger than Ω_0 at t_{n+1}. Clearly, there is no difficulty to satisfy this condition if the domain $\Omega_1 \setminus \Omega_0$ is not too narrow.

3.2. The approximation of $(\boldsymbol{u}(t_{n+1}),p(t_{n+1/2}))$ in Ω_1

Benefiting from the possibility offered by domain decomposition methods to use different formulations and/or different methods of approximation in each subdomain, we use the $\boldsymbol{u}$–p formulation (Eq. (1)) in the subdomain Ω_1.

To restrict the system (1) to Ω_1, we need to replace the condition at infinity (1) by a transmission condition on the interface Γ_1. Assuming, for the time being, that such a boundary condition $(\boldsymbol{u}_{\Gamma_1}^{n+1})$ is given on the interface Γ_1, we build the approximation $(\boldsymbol{u}_1^{n+1},p_1^{n+1/2})$ as follows:

$$\frac{\boldsymbol{u}_1^{n+1}-\boldsymbol{u}_1^n}{\delta t}-\nu\Delta\frac{\boldsymbol{u}_1^n+\boldsymbol{u}_1^{n+1}}{2}+\nabla p_1^{n+1/2}=[\boldsymbol{u}\cdot\nabla\boldsymbol{u}]^{n+1/2}\quad\text{in }\Omega_1$$

$$\nabla\cdot\boldsymbol{u}_1^{n+1}=0\quad\text{in }\Omega_1$$

$$\boldsymbol{u}_1^{n+1}=\boldsymbol{a}(t_{n+1})\quad\text{on } B$$

$$\boldsymbol{u}_1^{n+1}=\boldsymbol{u}_{\Gamma_1}^{n+1}\quad\text{on }\Gamma_1 \tag{11}$$

where $[\phi]^{1/2}$ denotes the second-order extrapolation $\frac{3}{2}\phi^n-\frac{1}{2}\phi^{n-1}$.

Now, the problem reduces to obtaining an approximation of the transmission condition $\boldsymbol{u}_{\Gamma_1}^{n+1}$ on the interface Γ_1. Two alternatives are possible: either we apply the Biot–Savart integral (7) in the whole flow domain Ω, or we restrict it to the external subdomain Ω_0 by using Eq. (8) at t_{n+1}. Restricting the Biot–Savart integral to Ω_0 may seem to be the cheapest alternative. In this case we have

$$\boldsymbol{u}_{\Gamma_1}^{n+1}(\boldsymbol{x})=U_\infty^{n+1}\boldsymbol{i}+\int_{\Omega_0}\omega_0^{n+1}(\boldsymbol{y})\boldsymbol{k}\wedge\nabla_y G(\boldsymbol{x}-\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}$$

$$+\oint_{\Gamma_0}\nabla_y G(\boldsymbol{x}-\boldsymbol{y})\big[\boldsymbol{n}\cdot\boldsymbol{u}_1^n(\boldsymbol{y})\big]\,\mathrm{d}l_y\quad\forall\boldsymbol{x}\in\Gamma_1$$

$$-\oint_{\Gamma_0}\nabla_y G(\boldsymbol{x}-\boldsymbol{y})\wedge\big[\boldsymbol{n}\wedge\boldsymbol{u}_1^n(\boldsymbol{y})\big]\,\mathrm{d}l_y+\mathcal{O}(\delta t), \tag{12}$$

where $\boldsymbol{u}_1^n$ has been chosen as a $\mathcal{O}(\delta t)$ approximation of $\boldsymbol{u}_1(t_{n+1})$ on Γ_0. Clearly, this approximation is first-order accurate in time. Second-order accuracy might be obtained by using $2\boldsymbol{u}_1^n-\boldsymbol{u}_1^{n-1}$ as an approximation of $\boldsymbol{u}_1(t_{n+1})$ on Γ_0. We have not tested this possibility to avoid extrapolating the velocity in time; hence, to have second-order accuracy in time, we have chosen the first alternative.

To apply the Biot–Savart integral (7) to the whole fluid domain, we need an extended approximate vorticity field $\tilde{\omega}_1^{n+1}$. Since, ω_0^{n+1} is a good approximation on Ω_0, we only need an approximation of the vorticity at time t_{n+1} on the domain $\Omega_1\setminus\Omega_0$. For this purpose, we compute an approximate velocity field as follows:

$$\hat{\boldsymbol{u}}_1^{n+1}(\boldsymbol{x})=\begin{cases}\boldsymbol{a}(\boldsymbol{x},t_n+1) & \boldsymbol{x}\in B\\ \boldsymbol{u}_1^n(\boldsymbol{x})+\delta t\big[\boldsymbol{u}_1(\boldsymbol{x})\cdot\nabla\boldsymbol{u}_1(\boldsymbol{x})-\nu\Delta\boldsymbol{u}_1(\boldsymbol{x})\big]^n & \boldsymbol{x}\in\Omega_1.\end{cases} \tag{13}$$

Clearly, $\hat{\boldsymbol{u}}_1^{n+1}=\boldsymbol{u}_1(t_{n+1})-\delta t\nabla p_1(t_{n-1})+\mathcal{O}(\delta t^2)$; as a result, $\nabla\wedge\hat{\boldsymbol{u}}_1^{n+1}(\boldsymbol{x})$ is a $\mathcal{O}(\delta t^2)$ approximation of $\omega_1(t_{n+1})$. Hence, by defining

$$\tilde{\omega}_1^{n+1}(\boldsymbol{x})=\begin{cases}\omega_0^{n+1}(\boldsymbol{x}) & \text{for } \boldsymbol{x}\in\Omega_0,\\ \nabla\wedge\hat{\boldsymbol{u}}_1^{n+1}(\boldsymbol{x}) & \text{for } \boldsymbol{x}\in\Omega_1\backslash\Omega_0,\end{cases} \tag{14}$$

we obtain a $\mathcal{O}(\delta t^2)$ approximation of $\omega(t_{n+1})$. As a result, the desired approximation of the transmission condition ($\boldsymbol{u}_{\Gamma_1}^{n+1}$) is given by:

$$\boldsymbol{u}_{\Gamma_1}^{n+1}(\boldsymbol{x}) = U_\infty^{n+1}\boldsymbol{i} + \int_\Omega \tilde{\omega}_1^{n+1}(\boldsymbol{y})\boldsymbol{k} \wedge \nabla_y G(\boldsymbol{x}-\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}$$

$$+ \oint_B \nabla_y G(\boldsymbol{x}-\boldsymbol{y})\big[\boldsymbol{n}\cdot\boldsymbol{a}(\boldsymbol{y},t_n+1)\big]\,\mathrm{d}l_y \quad \text{for } \boldsymbol{x}\in\Gamma_1$$

$$- \oint_B \nabla_y G(\boldsymbol{x}-\boldsymbol{y}) \wedge \big[\boldsymbol{n}\wedge\boldsymbol{a}(\boldsymbol{y},t_n+1)\big]\,\mathrm{d}l_y \tag{15}$$

3.3. Flow chart of the algorithm

In this section we summarize the algorithm developed above.

1. Computation of ω_0^{n+1} in Ω_0
   (a) Evaluate $\tilde{\omega}_0^n$ (extension of ω_0^n to Ω) by means of Eq. (6), i.e.,
   – Use ω_0^n in Ω_0,
   – Use $\nabla\wedge\boldsymbol{u}_1^n$ in $\Omega_1\backslash\Omega_0$.

   (b) Computation of $\tilde{\boldsymbol{u}}_0^n$ in Ω by means of Eq. (8), i.e.,
   – Use $\boldsymbol{u}_1^n$ in Ω_1,
   – Use Biot–Savart integral in $\Omega_0\backslash\Omega_1$.

   (c) Evaluate $\tilde{\omega}_0^{n+1}$ by using (Eq. (9)), i.e., apply the characteristics method + explicit treatment of viscosity + truncate solution to Ω_0.

2. Computation of $(\boldsymbol{u}_1^{n+1}, p_1^{n+1/2})$ in Ω_1
   (a) Evaluate $\boldsymbol{u}_{\Gamma_1}^{n+1}$
   – Evaluate $\tilde{\omega}_1^{n+1}$ (extension of ω_0^{n+1} to Ω) by using Eqs. (13) and (14).
   – Compute $\boldsymbol{u}_{\Gamma_1}^{n+1}$ by means of the Biot–Savart integral (15).

   (b) Compute the new solution $(\boldsymbol{u}_1^{n+1}, p^{n+1/2})$, using the boundary value $\boldsymbol{u}_{\Gamma_1}^{n+1}$ and Eq. (11).

4. The spatial approximation

4.1. The hybrid approximation in Ω_0

In this section, we describe the spatial approximation used in domain Ω_0.

First, we build the hybrid approximation $\tilde{\omega}_0^n$ of $\omega(t_n)$ by means of Eq. (6). For this purpose, we define an arbitrarily large Cartesian grid covering the whole fluid domain (see Fig. 2) with mesh-size h. The approximation ω_1^n, defined on the finite difference grid in $\Omega_1\diagdown\Omega_0$, is

transferred to the Cartesian grid by means of a high order $\mathbb{Q}_2$ interpolation. Since the computer memory is finite, we only keep a list of the grid points where the vorticity is significantly large enough, say $|\tilde{\omega}_0^n| \geq \epsilon \simeq h^2\delta t$. Let $\{\boldsymbol{P}_i\}$ be the union of grid points where the reconstructed initial approximation $\tilde{\omega}_0^n$ satisfies $|\tilde{\omega}_0^n(\boldsymbol{P}_i)| \geq \epsilon$. At the next time step t_{n+1}, if δt is small enough, the possible grid points where $|\omega(\boldsymbol{x},t_{n+1})| \geq \epsilon$ must be in $\{\boldsymbol{P}_i\}$ or be neighboring points of $\{\boldsymbol{P}_i\}$ denoted by $\{\boldsymbol{Q}_i\}$ the union of these grid points. Hereafter, the cell to which $\boldsymbol{Q}_i$ belongs is denoted by Q_i. For the sake of simplicity of the presentation, we assume that $Q_i = \{\boldsymbol{x} \in \mathbb{R}^2; \|\boldsymbol{x} - \boldsymbol{Q}_i\|_\infty \leq h/2\}$.

Second, we calculate a discrete approximation of $\tilde{\boldsymbol{u}}_0^n$. For the grid points $\boldsymbol{Q}_j$ in Ω_1, we use the $\mathbb{Q}_1$ interpolation of $\boldsymbol{u}_1^n$ in Ω_1. For the points $\boldsymbol{Q}_j \in \Omega_0 \setminus \Omega_1$, we build quadrature formulas for the integrals in Eq. (8). To minimize the interpolation error induced by the transfer of information between Ω_1 and Γ_0 in Eq. (8), we assume that the interface Γ_0 is a grid line of the finite difference grid in Ω_1. We denote by $\{(\boldsymbol{y}_i, l_i)\}$ the partition of Γ_0 induced by the trace of the finite difference grid in Ω_1 on Γ_0. The length and the center of the ith segment are denoted by l_i and $\boldsymbol{y}_i$, respectively. Finally, for all $\boldsymbol{Q}_j$ in $\Omega_0 \setminus \Omega_1$ we approximate $\tilde{\boldsymbol{u}}_0^n(\boldsymbol{Q}_j)$ as follows:

$$\tilde{\boldsymbol{u}}_0^n(\boldsymbol{Q}_j) = U_\infty^n \boldsymbol{i} + \sum_{\boldsymbol{y}_i} \nabla G(\boldsymbol{Q}_j - \boldsymbol{y}_i)\left[\boldsymbol{n} \cdot \boldsymbol{u}_1^n\right](\boldsymbol{y}_i) l_i - \sum_{\boldsymbol{y}_i} \nabla G(\boldsymbol{Q}_j - \boldsymbol{y}_i) \wedge \left[\boldsymbol{n} \wedge \boldsymbol{u}_1^n\right](\boldsymbol{y}_i) l_i$$

$$\sum_{Q_i \neq Q_j, Q_i \subset \Omega_0} \mathrm{meas}(Q_i)\tilde{\omega}_0^n(\boldsymbol{Q}_i)\boldsymbol{k} \wedge \nabla G(\boldsymbol{Q}_j - \boldsymbol{Q}_i)$$

$$+ \sum_{\boldsymbol{Q}_i \neq \boldsymbol{Q}_j, Q_i \cap \Gamma_0 \neq \varnothing} \mathrm{meas}(Q_i \cap \Omega_0)\tilde{\omega}_0^n(\boldsymbol{Q}_i)\boldsymbol{k} \wedge \nabla G\left(\boldsymbol{Q}_j - \tilde{\boldsymbol{Q}}_i\right) \tag{16}$$

Here $\mathrm{meas}(Q_i \cap \Omega_0)$ is the surface of $Q_i \cap \Omega_0$. If $Q_i \subset \Omega_0$, then $\mathrm{meas}(Q_i)$ is equal to h^2. If $Q_i \cap \Gamma_0$ is not empty, we denote by $\tilde{\boldsymbol{Q}}_i$ the geometrical center of $Q_i \cap \Omega_0$ (see Fig. 3). In principle we should use $\tilde{\omega}_0(\tilde{\boldsymbol{Q}}_i)$ in Eq. (16); however, since we use a constant approximation of $\tilde{\omega}_0^n$ in each cell, we are led to use $\tilde{\omega}_0^n(\boldsymbol{Q}_i)$ in the place of $\tilde{\omega}_0^n(\tilde{\boldsymbol{Q}}_i)$.

If we calculate directly the velocity on $\{\boldsymbol{Q}_j \in \Omega_0 \setminus \Omega_1\}$ by using Eq. (16), the operation count is of order $\mathcal{O}(N^2)$ with N being the number of points in $\{\boldsymbol{Q}_j \in \Omega_0 \setminus \Omega_1\}$. This computational

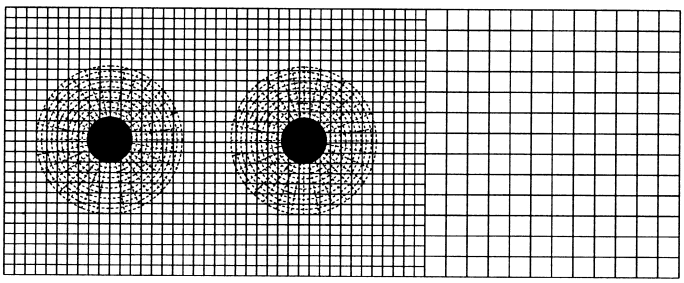

Fig. 2. The grid used in the case of two cylinders.

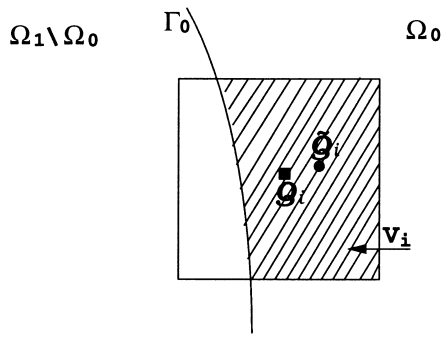

Fig. 3. Intersection of a cell Q_i with the interface Γ_0.

cost is not acceptable for large values of N; in the calculations presented therein, we have adopted a fast multipole version of Eq. (16). This procedure brings the computational cost to $\mathcal{O}(N)$ (for more detail, see Refs. [2,19]).

Finally, an approximation of $\omega(t_{n+1})$ is built on the set of points $\{\boldsymbol{Q}_j \in \Omega_0\}$ by means of Eqs. (9) and (10). First, we calculate an approximation of the foot of each characteristic by integrating Eq. (10) by means of a second-order Runge-Kutta approximation and the $\mathbb{Q}_1$ interpolation of $\tilde{u}_0^n$ and $\tilde{u}_0^{n-1}$ on the grid points $\{\boldsymbol{Q}_i\}$. Once the foot of each characteristics is evaluated, we calculate $\tilde{\omega}_0(\boldsymbol{\xi}(\boldsymbol{Q}_i,t_{n+1};t_n))$ by using the Lagrange interpolation of order 2 in the space $\mathbb{P}_2 = \{1,x,y,xy,x^2,y^2\}$ on the closest neighboring points as depicted in Fig. 4. Whenever we need the value of $\tilde{\omega}_0^n$ (*or* $\tilde{\omega}_0^{n-1}$) at a grid point, we search in the list that keeps track of the points where the vorticity is significantly large; if the point is not in the list, we assume the vorticity to be zero. The Laplace operator in Eq. (10) is approximated by the usual second order centered finite difference scheme.

Without any difficulty, different Cartesian grids in different regions can be used as it has been done in the numerical cases presented hereafter (see Fig. 2).

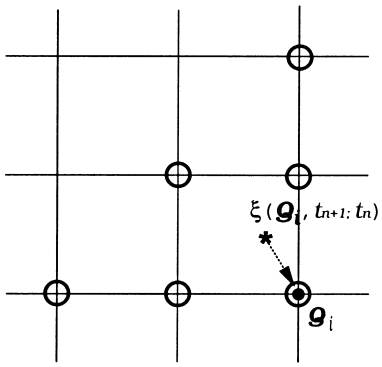

Fig. 4. The Lagrange interpolation for the method of characteristics. $\boldsymbol{Q}_i$ is a grid point, $\boldsymbol{\xi}(\boldsymbol{Q}_i,t_{n+1};t_n)$ is the foot of the characteristics, the circles denote the nodes on which the $\mathbb{P}_2$ Lagrange interpolation is performed.

4.2. The finite difference approximation in Ω_1

The finite difference approximation of $\boldsymbol{u}(t_{n+1})$ and $p(t_{n+1})$ in Ω_1 is carried out in two steps. First, we evaluate the transmission condition (15), then we discretize the system (11).

The flow subdomain Ω_1 is discretized by means of a semi-staggered grid. The set of the grid cells is denoted by $\{C_i\}$; with each cell center we associate its center $\boldsymbol{C}_i$. The partition of the solid boundary B induced by the trace of the grid in Ω_1 is denoted by $\{(\boldsymbol{y}_i,l_i)\}$. The interface Γ_0 is defined as being a grid line of the grid in Ω_1. The velocity degrees of freedom are located at the cell vertices, whereas those of the pressure and the vorticity are located at the cell centers.

First, we evaluate $\tilde{\omega}_1^{n+1}$ by using the discrete counterparts of Eqs. (13) and (14). The Laplacian $\Delta\boldsymbol{u}_1^n$ is approximated by means of centered second-order finite difference, whereas the advection term $(\boldsymbol{u}_1^n \cdot \nabla\boldsymbol{u}_1^n)$ is approximated by upwind third-order finite difference. The curl of the velocity, $\nabla \wedge \hat{\boldsymbol{u}}_1^{n+1}$, is evaluated at the cell center by second-order finite difference. Second, we approximate the transmission velocity $\boldsymbol{u}_{\Gamma_1}^{n+1}$ given by Eq. (15) as follows:

$$
\begin{aligned}
\boldsymbol{u}_{\gamma_1}^{n+1}(\boldsymbol{x}) &= U_\infty^{n+1}\boldsymbol{i} + \sum_{\boldsymbol{y}_i}\nabla G(\boldsymbol{x}-\boldsymbol{y}_i)\left[\boldsymbol{n}\cdot\boldsymbol{a}^{n+1}\right](\boldsymbol{y}_i)l_i \\
&- \sum_{\boldsymbol{y}_i}\nabla G(\boldsymbol{x}-\boldsymbol{y}_i)\wedge\left[\boldsymbol{n}\wedge\boldsymbol{a}^{n+1}\right](\boldsymbol{y}_i)l_i \\
&+ \sum_{C_j\in\Omega_1\Omega_0}\mathrm{meas}(\boldsymbol{C}_j)\tilde{\omega}_1^{n+1}(\boldsymbol{C}_j)\boldsymbol{k}\wedge\nabla G(\boldsymbol{x}-\boldsymbol{C}_j) \\
&+ \omega_0^{n+1}(\boldsymbol{Q}_i)\boldsymbol{k}\wedge\int\!\!\int_{Q_i}\nabla G(\boldsymbol{x}-\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \\
&+ \sum_{x\notin Q_j\subset\Omega_0}\mathrm{meas}(\boldsymbol{Q}_j)\omega_0^{n+1}(\boldsymbol{Q}_j)\boldsymbol{k}\wedge\nabla G(\boldsymbol{x}-\boldsymbol{Q}_j) \\
&+ \sum_{x\notin Q_j,\ Q_j\cap\Gamma_0\neq\varnothing}\mathrm{meas}(\boldsymbol{Q}_j\cap\Omega_0)\omega_0^{n+1}(\boldsymbol{Q}_j)\boldsymbol{k}\wedge\nabla G\left(\boldsymbol{x}-\tilde{\boldsymbol{Q}}_j\right)
\end{aligned}
\tag{17}
$$

where $\boldsymbol{x}$ belongs to Γ_1 and the cell Q_i. The cell Q_i is singled out since, to guarantee the accuracy, we use the analytical calculation of the integral $\int\!\!\int_{Q_i}\nabla G(\boldsymbol{x}-\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}$ (which is not difficult to evaluate because of the simplicity of the integrated function (∇G) and of the integration domain (the square Q_i)).

Finally, the velocity and the pressure are approximated in Ω_1 by means of a discrete counterpart of Eq. (11) that we present now. To avoid the complexity induced by the coupling of the pressure and the velocity, we adopt a fractional step projection method (see Refs. [3,20]).

$$\begin{cases} \dfrac{\tilde{\boldsymbol{u}}_1^{n+1}-\boldsymbol{u}_1^n}{\delta t}-\nu\Delta\dfrac{\boldsymbol{u}_1^n+\tilde{\boldsymbol{u}}_1^{n+1}}{2}+\nabla_{p_1}^{n+1/2}=[\boldsymbol{u}\cdot\nabla\boldsymbol{u}]^{n+1/2} \quad \text{in } \Omega_1 \\ \tilde{\boldsymbol{u}}_1^{n+1}=\boldsymbol{a}(t_{n+1}) \quad \text{on } B \\ \tilde{\boldsymbol{u}}_1^{n+1}=\boldsymbol{u}_{\Gamma_1}^{n+1} \quad \text{on } \Gamma_1 \end{cases} \tag{18}$$

$$\begin{cases} \dfrac{\boldsymbol{u}_1^{n+1}-\tilde{\boldsymbol{u}}_1^{n+1}}{\delta t}+\nabla\left(p_1^{n+1/2}-p_1^{n-1/2}\right) \quad \text{in } \Omega_1 \\ \nabla\cdot\boldsymbol{u}_1^{n+1}=0 \quad \text{in } \Omega_1 \\ \boldsymbol{n}\cdot\boldsymbol{u}_1^{n+1}=\boldsymbol{n}\cdot\boldsymbol{a}(t_{n+1}) \quad \text{on } B \\ \boldsymbol{n}\cdot\boldsymbol{u}_1^{n+1}=\boldsymbol{n}\cdot\boldsymbol{u}_{\Gamma_1}^{n+1} \quad \text{on } \Gamma_1 \end{cases} \tag{19}$$

This scheme has been proposed by Van Kan [21]. An analysis of convergence of this type of scheme can be found in Guermond [10]. The discrete counterpart of Eq. (18) is obtained by replacing the differential operators by finite differences. The velocity is approximated at the cell vertices, the pressure is approximated at the cell centers. The gradient (∇p) and the Laplacian ($\Delta\boldsymbol{u}$) are approximated by means of centered second-order finite differences, whereas the advection term ($\boldsymbol{u}\cdot\nabla\boldsymbol{u}$) is approximated by upwind third order finite difference. The linear system associated with Eq. (19) is solved by means of a multigrid technique.

There are many ways to solve the projection step (19). The most common one consists in applying the divergence operator to Eq. (19) to obtain a Poisson equation that controls $(p_1^{n+1/2}-p_1^{n-1/2})$. See Ref. [14] for other details.

5. Numerical results and comparison

In this section, we report on the numerical performance of the present domain decomposition method. We compare the results obtained by the present technique with experimental data and other numerical results.

5.1. Results for one circular cylinder

To assess the accuracy of the present domain decomposition method we have tested it on the impulsively-started circular cylinder problem. The velocity scale is the velocity at infinity U_∞ and the length scale is the cylinder radius r. The Reynolds number is defined as follows: $Re = U_\infty 2r/\nu$

The objective of the first set of tests is to show that the results of the present domain decomposition method are consistent with those of a more standard single-domain technique. The benchmark, single-domain, method is the finite difference technique that is used in Ω_1 assuming the radius of Ω_1 to be equal to 20 cylinder radius and enforcing the velocity at the outer boundary to be equal to the irrotational one. For the domain decomposition method, we define Ω_1 as being a ring whose inner radius is r and whose outer radius is $2r$ (i.e., the interface Γ_1 is a circle of radius $2r$). The interface Γ_0 is a circle of radius $1.5r$. We have made tests for

two values of the Reynolds number: $Re = 3000$ and 10,000. We used $\delta t = 0.01$ in the first case and $\delta t = 0.005$ in the second case. We have reported in Fig. 5 the streamline patterns of the flow at times $t = 1$, 2, 3, 4, and 5. The results for $Re = 3000$ are on the left and those for $Re =$ 10,000 are on the right. On each figure we compare the results from the present domain decomposition method with those from the single-domain computation. The results of the domain decomposition method are at the bottom of the figures whereas the single-domain results are at the top. The agreement between the two series of calculations seems to show that the domain decomposition technique has the same accuracy as that of the single-domain method.

To further assess this statement we have compared the vorticity and the pressure distributions on the solid cylinder given by the two methods for $Re = 3000$. The results are reported in Fig. 6. The results from the present domain decomposition method (DEC-lines) are compared with those of the single-domain, finite-difference method (FD-symbols). The very close agreement between the two series of results confirms that, at least in Ω_1, the domain decomposition method has the same accuracy properties as those of the single-domain technique.

To go beyond the self-consistency tests, we have compared the results of the present technique with other published results. In Ref. [12], the vorticity distribution on the cylinder for $1 \leq t \leq 6$ and $Re = 3000$ is reported. From a global point of view, the vorticity distribution plotted in Fig. 6 is very much alike that reported in Ref. [12]. A precise comparisons for $t = 1$, $t = 2$ and $t = 4$ is made in Table 1. In this table, we compared the location and strength of the maximal value of the vorticity on the cylinder in the aft recirculation region, the secondary eddy and the fore boundary layer (the upper part of the cylinder corresponds to $0 \leq \theta \leq \pi$). This comparison shows a pretty good agreement between our results and those of Ref. [12].

Table 1
Comparison of location and strength of the maximal value of vorticity on the cylinder in the aft recirculation region, the secondary eddy and the fore boundary layer (on the upper part of the cylinder), $Re = 3000$

	Aft recirculation	Secondary eddy	Fore boundary layer
$t = 1$			
θ (present)	2.04		0.54
θ [12]	2.09		0.56
ω (present)	−85.0		26.0
ω [12]	−84.0		29.0
$t = 2$			
θ (present)	2.08	0.77	0.50
θ [12]	2.11	0.80	0.53
ω (present)	−81.0	−4.7	92.0
ω [12]	−86.0	−11.0	118.0
$t = 4$			
θ (present)	2.20	0.83	0.31
θ [12]	2.21	0.84	0.32
ω (present)	−73.0	−37.0	50.0
ω [12]	−78.0	−43.0	57.0

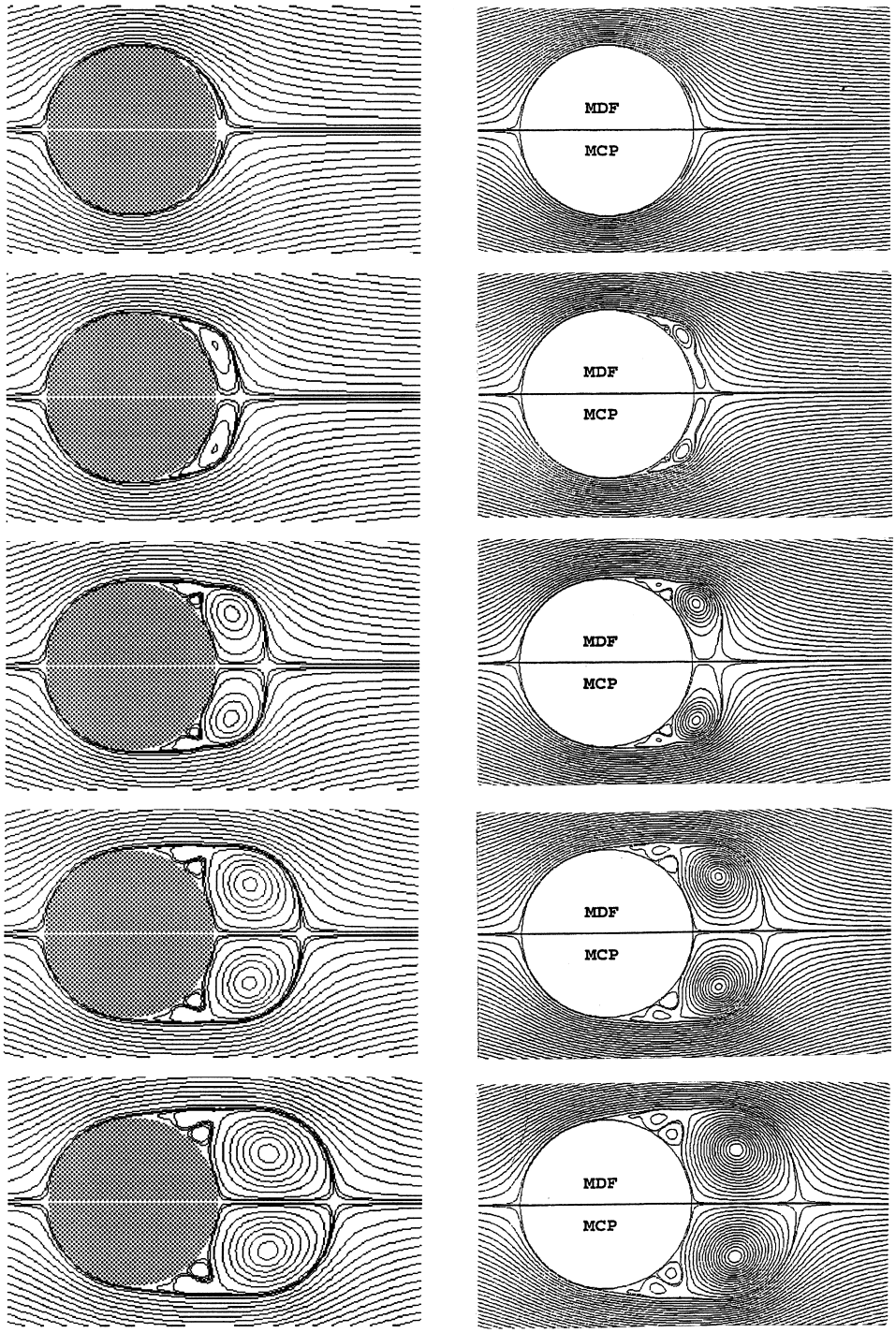

Fig. 5. Streamline patterns about an impulsively-started cylinder at times $t = 1$, 2, 3, 4 and 5 (from top to bottom) for $Re = 3000$, $\delta t = 0.01$ (left) and $Re = 10{,}000$, $\delta t = 0.005$ (right). On each figure we compare the results from the present domain decomposition method (bottom of figure) with those from the single-domain computation (top of figure).

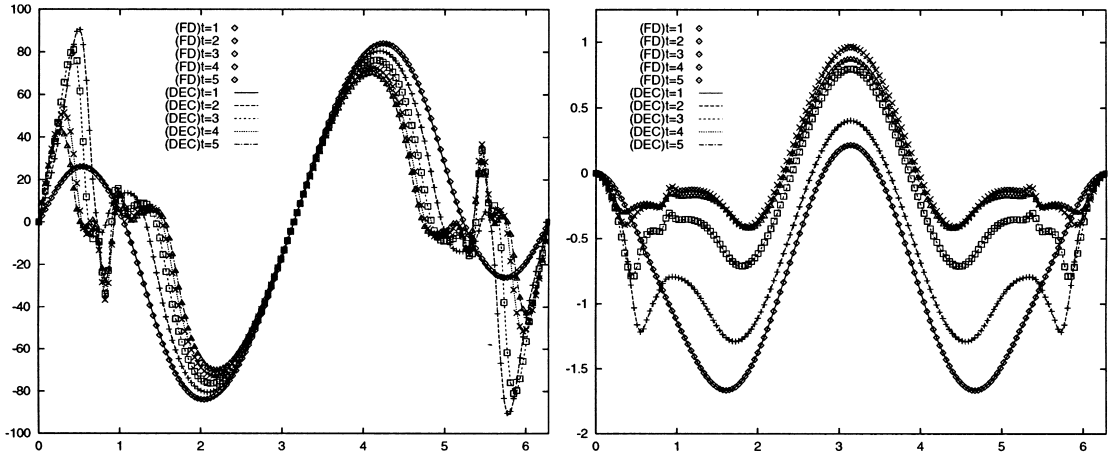

Fig. 6. Comparison between domain decomposition method (lines) and the single-domain method (symbols) for the vorticity (left) and pressure (right) distribution on the cylinder, $Re = 3000$.

To further evaluate the method, we have made comparisons on the radial velocity on the symmetry axis behind the cylinder for times $t = 1$, 2, 3, 4, and 5 for $Re = 3000$. The results are reported in Fig. 7, There are three sets of results: the experimental results from Loc–Bouard [1] (EXP-symbols), the results obtained by the present method (DEC-dashed lines), and the results of the single-domain technique (FD-solid lines). We remark that the velocity obtained by our domain decomposition technique is continuous at the interface Γ_1 (circle of radius 2) and the domain decomposition technique is consistent with the single-domain method. Note also that

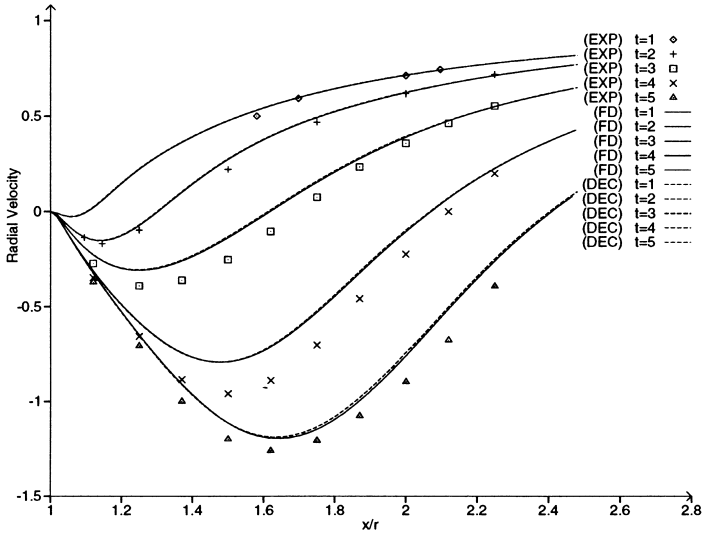

Fig. 7. Comparison between experimental results (symbols), present results (dashed lines) and the finite difference results (solid lines) for radial velocity on the symmetry axis behind the cylinder, $Re = 3000$.

there is a noticeable difference between the numerical results and the experimental ones. To settle this matter we have made comparison with results published in Ref. [4]. We have reported in Table 2 the radius where the radial velocity is minimal together with the value of the velocity for $t = 2$, 3, 4, and 5. There is a pretty good agreement between the three sets of results on the location of the minimum velocity, but the three references disagree on the minimal value of the velocity. Note though, that our results are between those of Refs. [1] and [4]. Finally, this test indicates that for this problem a precise numerical benchmark is needed.

In conclusion, the present results are in excellent coincidence with that of the single-domain finite difference method. Furthermore, a good agreement between the present results and other numerical or experimental results is observed.

5.2. Flow past two cylinders in tandem

In practice there may be several obstacles in the flow. To illustrate the flexibility of the present method, we simulate the flow past two circular cylinders in tandem. For each cylinder, Ω_1 and the interface Γ_0 are the same as in the previous example: Ω_1 is a ring of external radius $2r$ and Γ_0 is a circle of radius $1.5r$.

The flow configuration is complex due to the interaction between the two wakes. It is well known that the spacing between the two cylinders greatly influences the flow structure. In the following we restrict ourselves to a Reynolds number equal to 200. The time step is set to 0.03. We compare the results obtained by the domain decomposition technique with experimental data for different cylinder spacings. We denote by L the distance between the centers of the two cylinders, and $D = 2r$ is the diameter of the cylinders.

As usual, to accelerate the onset of instabilities and reduce the time of simulation, we have perturbed the symmetric flow. Here, we have chosen to rotate the downstream cylinder on itself for a short period of time after $t = 0$.

In Fig. 8, we show the instantaneous iso-vorticity lines for four different cylinder spacings: $L/D = 2$, $L/D = 3.5$, $L/D = 3.75$, and $L/D = 4$. We clearly observe that there are mainly two flow patterns:

- For small relative spacings ($L/D < 3.625$), there is no vortex street behind the upstream cylinder, whereas one is created behind the downstream one.
- For large relative spacings ($L/D > 3.625$), vortex streets appear behind both cylinders.

Table 2
Comparison of the minimal radial velocity on the symmetry axis behind the cylinder, $Re = 3000$. First column: numerical results from [4]; second column: present method; third column: experimental results from [1]

	Ref. [4]		Present		Ref. [1]	
t	r	$u_{\min}$	r	$u_{\min}$	r	$u_{\min}$
2	1.131	−0.121	1.147	−0.155	1.151	−0.164
3	1.249	−0.259	1.248	−0.328	1.249	−0.379
4	1.497	−0.724	1.477	−0.810	1.507	−0.948
5	1.654	−1.112	1.642	−1.207	1.628	−1.250

To further illustrate the change of the flow pattern as the relative spacing between the cylinders increases, we have plotted in Fig. 9 the time variation of the drag and lift coefficients on the two cylinders for $L/D = 2$, 3.5, 3.75, and 4. We observe that the upstream cylinder has a higher drag coefficient than the downstream one. Note also that the lift amplitude on the downstream cylinder is larger than that on the upstream one; the reason for this being the interaction of the downstream cylinder with the wake of the upstream one. It is clear on this figure that there is a dramatic change of the flow pattern for $3.5 \leq L/D \leq 3.75$.

The change of the flow pattern with respect to the cylinders' relative spacing can also be studied by looking at the vortex shedding frequency. We have calculated this frequency for $2 \leq L/D \leq 10$. Each time the upstream cylinder has been observed shedding vortices, the shedding frequencies of the upstream and the downstream cylinders have been observed to be equal. In Fig. 10, we have plotted the ratio of the Strouhal number St of the downstream cylinder to that of a single cylinder S_0 as a function of the relative spacing L/D. This figure clearly shows a discontinuous change of the Strouhal number which occurs for $3.62 \leq L/D \leq 3.67$. In this figure we also compare our numerical results with numerical results

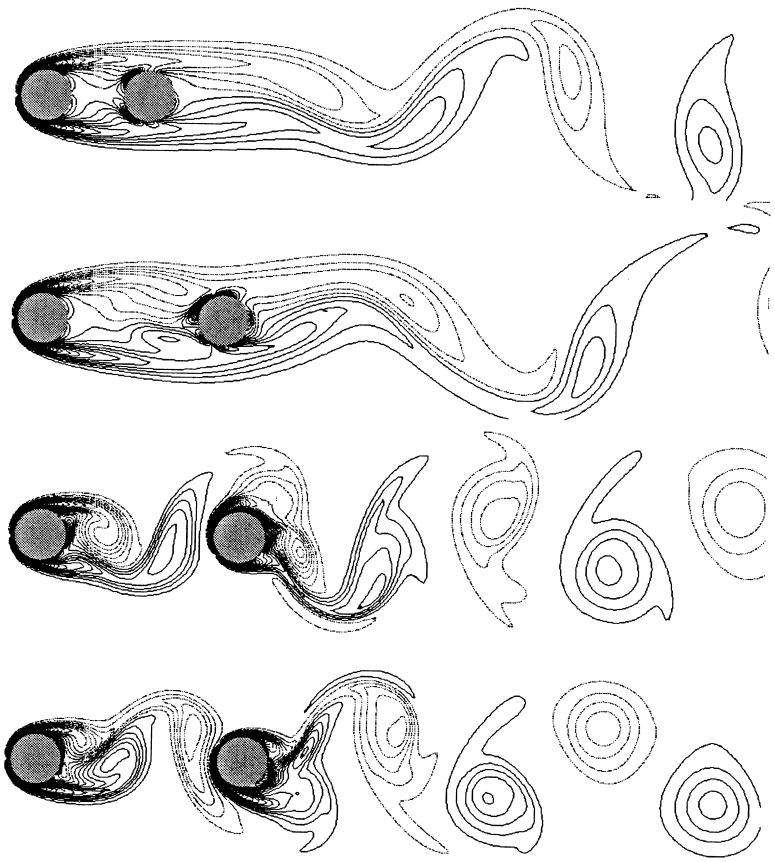

Fig. 8. Two types of flow patterns depending on the value of relative cylinder spacing L/D (iso-vorticity lines). From top to bottom $L/D = 2$ (t = 148), 3.5 (t = 138), 3.75 (t = 140), and 4 (t = 138). $Re = 200$ and $\delta t = 0.03$.

from Wang [22] that have been obtained by means of an integral-characteristics method. A similar discontinuous behavior is observed in Ref. [22]. We have also reported in Fig. 10 experimental data from Ohmi et al. [15] that have been obtained at a slightly lower Reynolds number: $Re = 120$. Even though the numerical and the experimental Reynolds numbers are different, we observe a good qualitative agreement between the experimental data and the present numerical results.

5.3. Two cylinders in relative motion

To illustrate the flexibility of the present domain decomposition technique, we study the flow around two obstacles in relative motion. The domain decomposition that is adopted in the reference frame of each cylinder is the same as in the case of two fixed cylinders: Ω_1 is a ring of external radius $2r$ and Γ_0 is a circle of radius $1.5r$. In general, this problem is difficult to be handled by the numerical methods that are based on a single domain approximation, since for

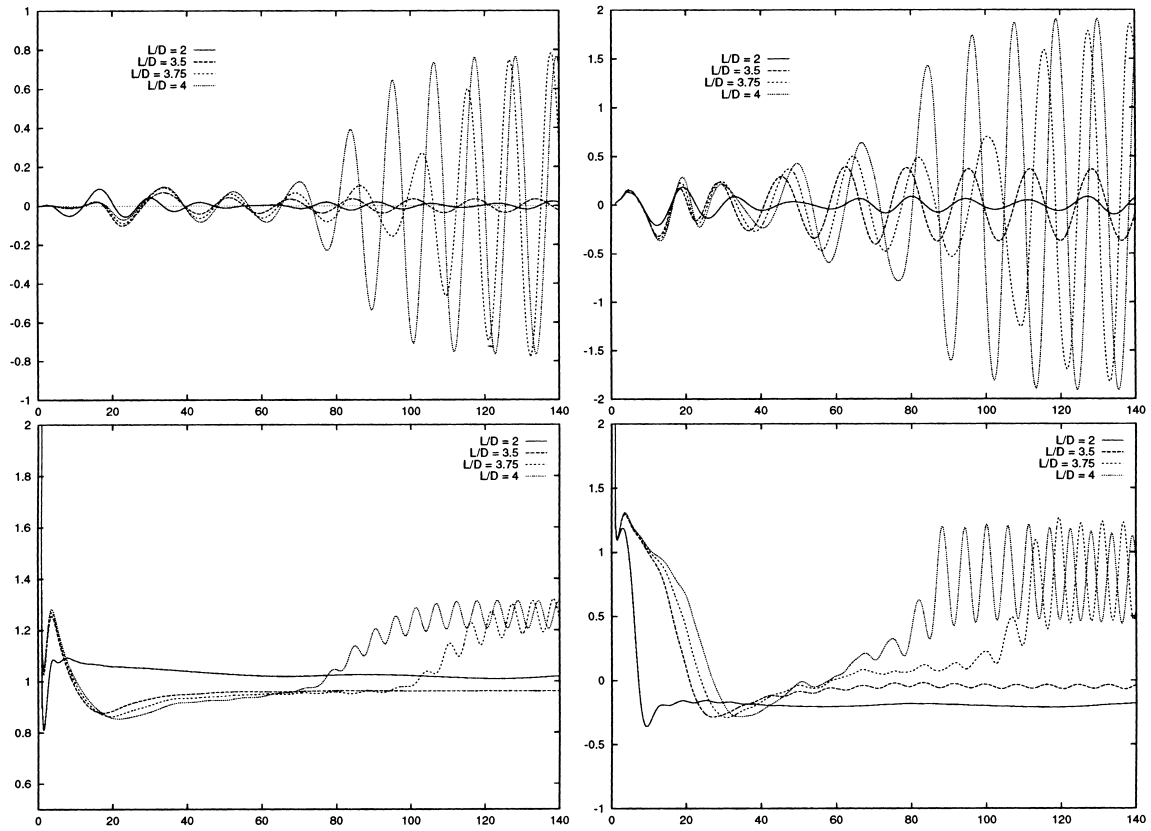

Fig. 9. Time variation of drag and lift coefficients on the two cylinders for different cylinder spacings L/D: (top left) lift coefficients on upstream moving cylinder; (top right) lift coefficients on downstream fixed cylinder; (bottom left) drag coefficients on upstream cylinder; (bottom right) drag coefficients on downstream cylinder.

this type of methods, the flow domain needs to be remeshed at each time step. For the present method no remeshing is needed. With the present domain decomposition method, we have simulated the flow past two tandem cylinders in relative motion for the Reynolds number equal to 200. The time step is set to 0.02. A vertical oscillation is enforced on the upstream cylinder, whereas the downstream one is fixed. The amplitude of the oscillations is equal to $r/2$ and the frequency f is set to be equal to the vortex shedding frequency that is observed when the two cylinders are fixed.

If Fig. 11, we have plotted instantaneous iso-vorticity lines for the cases $L/D = 2$ ($f = 0.0623$), $L/D = 3$ ($f = 0.0672$) and $L/D = 4$ ($f = 0.0905$). We observe a fully developed vortex shedding behind the upstream cylinder for the three different cylinder spacings considered. This is in contrast with the case of two fixed cylinders studied above where almost no shedding occurred for $L/D = 2$. Furthermore, the intensity of the vorticity behind the downstream cylinder is more important than that observed in the case of two fixed cylinders. Note also that the flow pattern is more complex.

In Fig. 12, the time variation of the drag and lift coefficients on the two cylinders is shown. As in the case of two fixed cylinders, the results reveal a sharp variation of the coefficients with respect to the cylinders' relative spacing. Furthermore, this figure shows more clearly than Fig. 11 that a vortex shedding occurs behind the upstream cylinder for all the cylinders' spacings, though the shedded vorticity decreases with the cylinder spacing. Because of the important vortex–obstacle interaction induced by the enforced movement of the upstream cylinder, the time evolution of the drag and lift is more irregular than in Fig. 9. This effect is amplified when the cylinders are close.

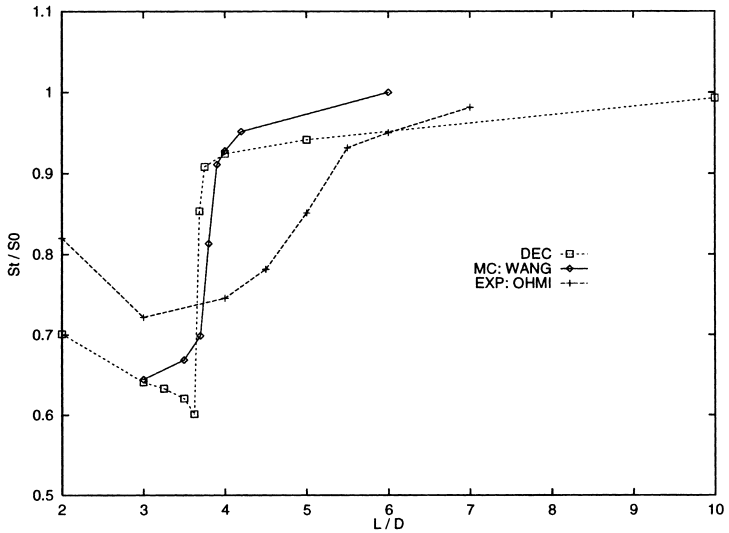

Fig. 10. Strouhal number vs. relative spacing of cylinders. DEC: present results, $Re = 200$; MC: results of an integral-characteristics method by Z.T. Wang [22], $Re = 200$; EXP: experimental results of K. Ohmi et al. [15] $Re = 120$.

6. Conclusions

We have presented in this paper a domain decomposition method for simulating two-dimensional external incompressible viscous flows. The method consists in using formulations and numerical techniques that are adapted to the flow structure in each subdomain. The subdomains overlap and are coupled by means of a Schwarz type strategy. One feature of the method consists in treating the initial-boundary-value problem in the external subdomain as an initial-value problem by benefiting from the overlapping of the subdomains.

The next step consists in extending the present method to three dimensions. This work is currently being developed.

We finish this paper by addressing the complexity issue. It is clear that the proposed domain decomposition technique is more complex than the finite difference, single-domain method when the shape of the computational domain is fixed in time. On the other hand, the present method may be useful for computational domain that vary in time; for instance, the present method accounts quite easily for moving obstacles as shown in Section 5.3. Furthermore, the DDM has proved to be faster, in terms of CPU, than the single domain method, provided the

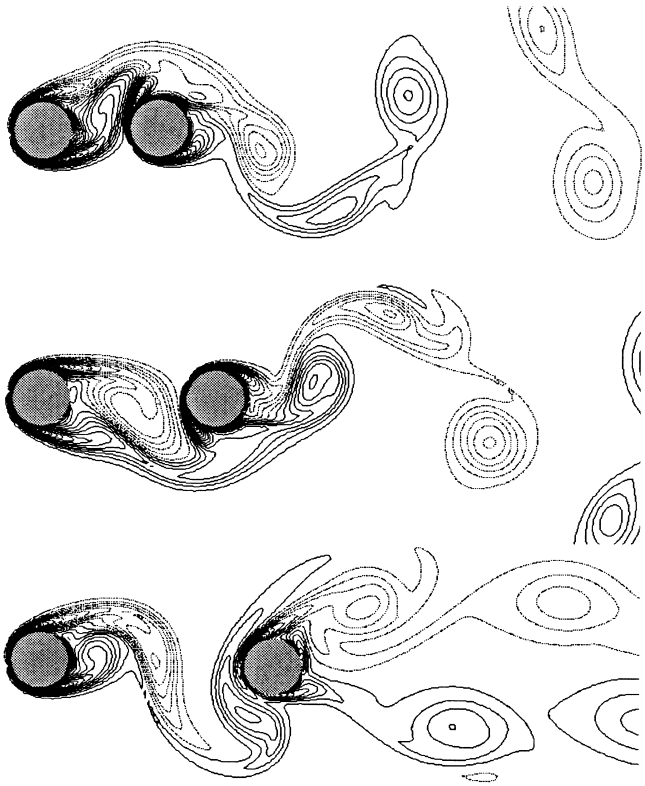

Fig. 11. Iso-vorticity lines for two cylinders in relative motion for different cylinder spacings L/D. From top to bottom $L/D = 2$ ($t = 128$), 3 ($t = 128$), and 4 ($t = 138$). $Re = 200$, $\delta t = 0.02$

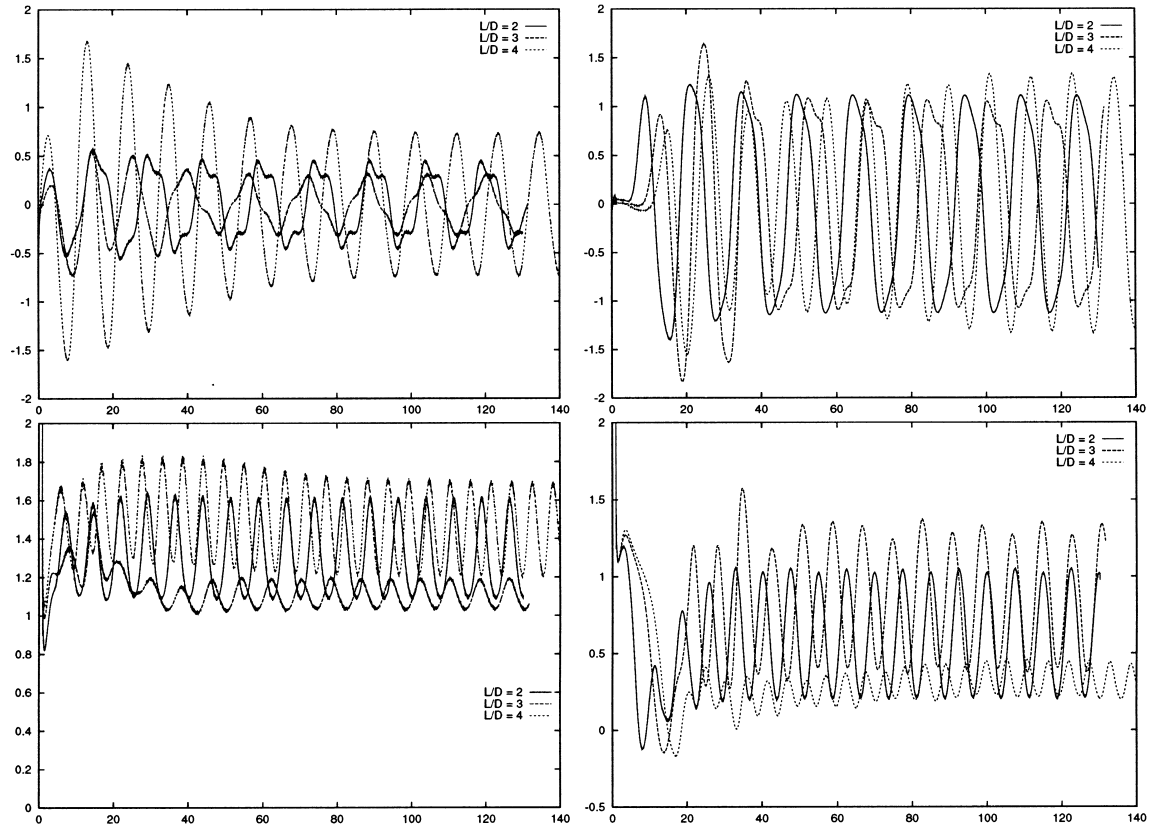

Fig. 12. Time variation of drag and lift coefficients on two cylinders in relative motion for different cylinder spacings L/D: (top left) lift coefficients on upstream moving cylinder; (top right) lift coefficients on downstream fixed cylinder; (bottom left) drag coefficients on upstream moving cylinder; (bottom right) drag coefficients on downstream fixed cylinder.

computation of the velocity at the Cartesian grid points in ω_0 is performed by means of a fast multipole technique.

References

[1] Loc TP, Bouard R. Numerical solution of the early stage of the unsteady viscous flow around a circular cylinder: a comparison with experimental visualization and measurements. J Fluid Mech 1985;160:93–117.
[2] Carrier J, Greengard L, Rokhlin V. A fast adaptive multipole algorithm for particle simulation. SIAM J Sci Stat 1988;9:669–86.
[3] Chorin AJ. Numerical solution of the Navier–Stokes equations. Math Comp 1968;22:745–62.
[4] Chou MH, Huang W. Numerical study of high-Reynolds-number flow past a bluff object. Int J Numer Methods Fluids 1996;23:711–32.
[5] Cottet G-H. Particle-grid domain decomposition methods for the Navier–Stokes equations in exterior domains, in A.M.S. Providence, RI,. Lectures in Applied Mathematics 1991;28:103–17.

[6] Charton P, Nataf F, Rogier F. Méthode de décomposition de domaine pour l'équation d'advection–diffusion. CR Acad Sci Paris Série I 1991;313:623–6.
[7] Daube O, Guermond J-L, Sellier A. Sur la formulation vitesse-tourbillon des équations de Navier–Stokes en écoulement incompressible. CR Acad Sci Paris Série II 1991;313:377–82.
[8] Deuring P. Finite element methods for the Stokes system in three-dimensional exterior domains. Math Meth Appl Sci 1997;20:245–69.
[9] Douglas J, Russell TF. Numerical methods for convection dominated diffusion problems based on combining the method of characteristics with finite element methods or finite difference method. SIAM J Numer Anal 1982;19:871–85.
[10] Guermond J-L. Some practical implementations of projection methods for Navier–Stokes equations. Modél Math Anal Numér 1996;30:637–67.
[11] Guermond J-L, Huberson S, Shen WZ. Simulation of 2D external viscous flows by means of a domain decomposition method. J Comput Physics 1993;108:343–52.
[12] Koumoutsakos P, Leonard A. High-resolution simulations of the flow around an impulsively started cylinder using vortex methods. J Fluid Mech 1995;296:1–38.
[13] Lions P-L. On the Schwarz alternating Method I. In: Glowinski R, Golub GH, Meurant GA, Périaux J, editors. First International Symposium on Domain Decomposition Methods for Partial Differential Equation. Philadelphia: SIAM, 1988. p. 1–42.
[14] Lu HZ. Simulation of external incompressible viscous flows by coupling finite differences and vortex method. Doctoral thesis, Université Paris XI, 1996.
[15] Ohmi K, Imaichi K. Flow visualization VI. In: Proceedings of the Sixth International Symposium on Flow Visualization. Yokohama, Japan: Springer-Verlag, 1992.
[16] Pironneau O. On the transport-diffusion algorithm and its applications to the Navier–Stokes equations. Numer Math 1982;38:309–32.
[17] Quarteroni A. Domain decomposition and parallel processing for the numerical solution of partial differential equations. Surv Math Ind 1991;1:75–118.
[18] Rehbach C. Calcul numérique d'écoulements tridimensionnels instationnaires avec nappes tourbillonaires. La Recherche Aérospatiale 1977;5:289–98.
[19] Rokhlin V. Rapid solution of integral equations of classical potentiel theory. J Comput Phys 1993;60:187–207.
[20] Temam R. Sur l'approximation de la solution des équations de Navier–Stokes par la méthode de pas fractionnaires. Arch Rat Mech Anal 1969;33:377–85.
[21] Van Kan J. A second order accurate pressure-correction scheme for viscous incompressible flow. SIAM J Sci Stat Comput 1986;7:870–91.
[22] Wang ZT. Résolution numérique des équations de Navier–Stokes en formulation vitesse-tourbillon par une méthode d'équations intégrale. Doctoral thesis, École Polytechnique de Paris, 1996.