

# Particle methods for direct numerical simulations of three-dimensional wakes†

Georges-Henri Cottet and Philippe Poncet‡

LMC-IMAG Université Joseph Fourier, Grenoble, BP 53, F-38041, France  
E-mail: Georges-Henri.Cottet@imag.fr and Philippe.Poncet@imag.fr

Received 20 September 2002

Published 17 October 2002

**Abstract.** In this paper we describe recent advances in the development of particle methods for the direct numerical simulations of three-dimensional wakes. Both body-fitted and immersed boundary techniques are considered. The accuracy and numerical cost of the proposed numerical methods are discussed on the benchmark cases of a flow past a cylinder and of a vortex ring impinging on a cylinder.

PACS numbers: 47.32.Cc, 02.70.Ns, 47.27.Vf

---

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Vortex-in-cell methods for 3D bluff-body flows</b>	<b>2</b>
2.1	Algorithm for cylinder wakes . . . . .	2
2.2	Numerical results . . . . .	4
<b>3</b>	<b>Immersed boundary techniques for particle methods</b>	<b>4</b>
3.1	No-through flow boundary condition for grid-free vortex methods . . . . .	5
3.2	No-through flow boundary condition for vortex in cell methods . . . . .	6
3.3	Diffusion, remeshing and no-slip boundary condition . . . . .	6
3.4	Numerical results . . . . .	7
<b>4</b>	<b>Summary and conclusions</b>	<b>9</b>

---

† This article was chosen from Selected Proceedings of the 4th International Workshop on Vortex Flows and Related Numerical Methods (UC Santa-Barbara, 17–20 March 2002) ed E Meiburg, G H Cottet, A Ghoniem and P Koumoutsakos.

‡ Present address: MIP, Complexe Scientifique de Rangueil, 31077 Toulouse Cedex 4, France.

## 1. Introduction

The direct numerical simulation of three-dimensional (3D) bluff-body flows remains a challenging problem in CFD. Even for rather simple geometries, such as a cylinder, the need for accuracy and robustness is very demanding for classical grid-based techniques. While accuracy often dictates the use of non-dissipative finite-difference or spectral element schemes, stability impose constraints on the compatibility of the grid and flow topologies and on the time-step values that can substantially slow down the methods.

In contrast, particle methods, when they use the appropriate tools, allow one to some extent by-pass the usual accuracy–stability dilemma. Their stability is indeed only constrained by the explicit diffusion solver ( $\nu\Delta t \leq Ch^{-2}$ ) and by the strain of the flow ( $\Delta t \leq C|\nabla u|^{-1}$ ). For moderate to high Reynolds numbers and the type of mesh sizes that are affordable in three dimensions, the first condition is rather weak and the second condition does not involve the mesh size. A number of validation studies [1]–[3] have been performed in recent years to confirm the capabilities of particle methods for the simulations of complex flows. The purpose of this paper is to review some more recent work devoted to the validation of vortex-in-cell methods and their application in the simulation of 3D wakes. We first recall the basic features of the method and describe its implementation in the context of cylinder wakes. We then discuss how to extend the method to more complex bodies treated as immersed boundaries. We in particular show that the concept of immersed boundaries is rather natural in vortex methods and does not really require much additional work compared to a body-fitted vortex method. We illustrate the methods by numerical examples and sketch some on-going applications.

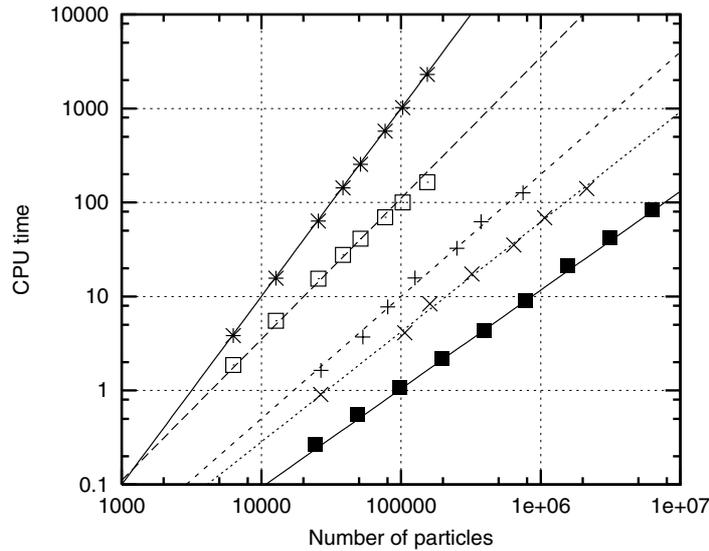
## 2. Vortex-in-cell methods for 3D bluff-body flows

Vortex methods, more generally particle methods, can be designed in a totally grid-free fashion or combined with Eulerian-grid solvers. For incompressible flows, grid-free methods consist in evaluating the velocity at each particle through a regularized Biot–Savart law. A considerable amount of work has been devoted (since the early 1980s) towards the development of fast  $N$ -body solvers that reduce the operation count of these evaluations from  $O(N^2)$  to  $O(N \log N)$ . However, the resulting algorithms are still orders of magnitude slower than FFT-based classical Poisson solvers. In contrast, vortex-in-cell (VIC) methods use an intermediate Eulerian grid on which vorticity is interpolated and velocity is computed. Non-dissipative interpolation functions reduce the discrepancy involved in the additional interpolation steps to a level that is compatible with the overall algorithm accuracy (see [1, 2]). The resulting vortex-in-cell codes can then compete, in terms of CPU, with classical Eulerian codes. Figure 1 shows a comparison between CPU reported by Lindsay and Krasny [4] for direct and fast summation algorithms, and VIC evaluations of velocities for different numbers of particles. For the sake of fairness, a scaling factor of 4, based on the CPU time needed for direct summation methods on the different platforms, has been applied to account for the difference in processor speeds between the SGI 75 MHz processor used by these authors and the alpha 500 MHz processor we were using. These comparisons clearly show the speed up offered by VIC methods whenever vorticity is not strongly localized, allowing high-resolution simulations of wakes in affordable CPU times.

### 2.1. Algorithm for cylinder wakes

Each time-step of our algorithm consists of the following sequence [1, 2]:

- interpolate particle vorticity onto the grid,
- compute velocity on the grid and interpolate velocity and stretching on particles,



**Figure 1.** Typical CPU times for velocity evaluations as a function of the number of particles, for VIC and grid-free calculations: (\*) direct summation [4], (□) fast multipole first-order calculation [4], (×) VIC method on a cylindrical grid filled with 65% particles, (+) VIC method on a cylindrical grid filled with 25% particles, (■) VIC method on a Cartesian grid filled with 100% particles.

- advance particle vorticity and location,
- remesh particles on regular locations,
- diffuse vorticity between particles by a particle strength exchange algorithm, and
- cancel slip at the boundary by the appropriate vorticity fluxes.

Concerning the computation of the velocity, the algorithm is based on the Helmholtz decomposition

$$\mathbf{u} = \nabla \times \boldsymbol{\psi} + \nabla \phi \quad (1)$$

where  $\boldsymbol{\psi}$  and  $\phi$  have to satisfy

$$-\Delta \boldsymbol{\psi} = \boldsymbol{\omega} \quad \text{in } \Omega \quad (2)$$

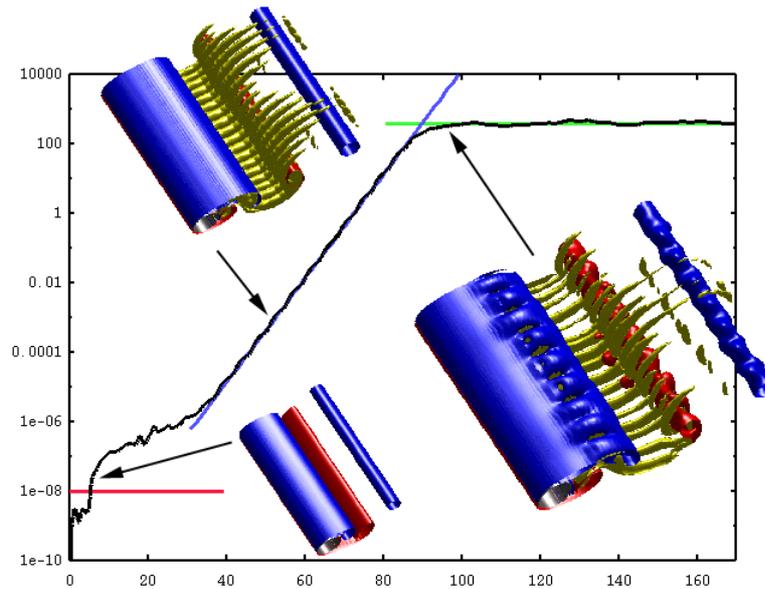
$$\nabla \cdot \boldsymbol{\psi} = 0 \quad \text{in } \Omega \quad (3)$$

$$\Delta \phi = 0 \quad \text{in } \Omega \quad (4)$$

$$\frac{\partial \phi}{\partial \mathbf{n}} = -(\nabla \times \boldsymbol{\psi}) \cdot \mathbf{n} \quad \text{on } \Gamma. \quad (5)$$

In the above equations  $\Omega$  and  $\Gamma$  respectively, denote the fluid domain and body surface. The use of a scalar potential allows one to decouple the boundary conditions and the vorticity, which greatly simplifies the calculations of  $\boldsymbol{\psi}$  in three dimensions. Details of the algorithm can be found in [5, 6]. It is worth noting that, to ensure that the vorticity flux boundary condition preserves divergence-free vorticity, the formula proposed in [7] for flat boundaries has to be changed for the azimuthal vorticity  $\omega_\theta$  into

$$\nu \left( \kappa \omega_\theta + \frac{\partial \omega_\theta}{\partial \mathbf{n}} \right) = \frac{u_z}{\Delta t}$$



**Figure 2.** The time history of the transverse enstrophy in a cylinder wake, and isosurfaces of vorticity at various stages of the development of 3D instabilities.

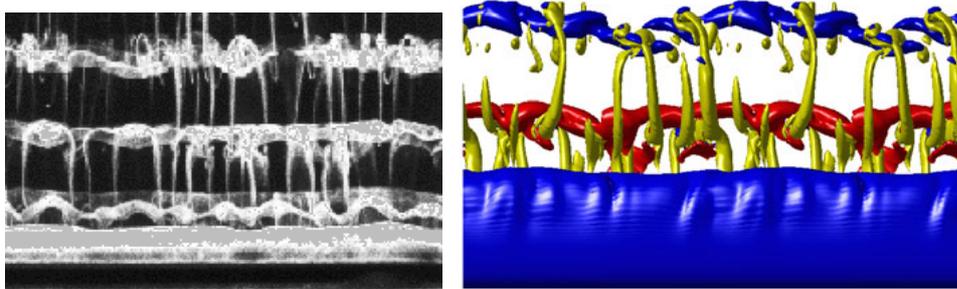
where  $\nu$  is the viscosity,  $\kappa$  denotes the curvature of the boundary and  $u_z$  the spanwise slip of the velocity (a complete derivation of this formula is given in [8]). However, for the Reynolds number considered in the applications, it can be shown [5, 8] that the curvature only induces a low-order correction over the two-dimensional (2D) vorticity flux formula.

## 2.2. Numerical results

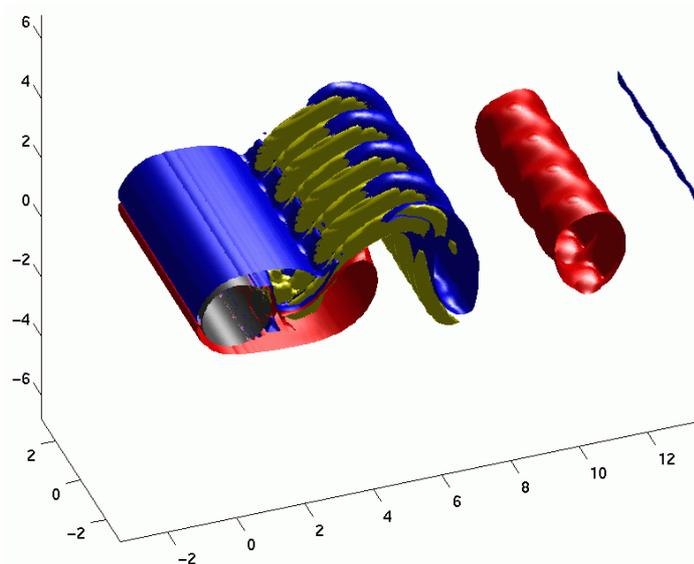
We present here some numerical results concerning the direct numerical simulation of a 3D cylinder wake at Reynolds number 400. The cylinder has a radius of 1. The computational domain extends to  $4\pi$  in the radial and spanwise directions. One highlight of the code is its robustness. For a grid of size of  $256 \times 128 \times 128$  the adimensionalized time-step is 0.1. The CPU time for one iteration is about 3 min on a single-processor alpha workstation, which gives about 3 h for a shedding cycle. Figure 2 summarizes the development of 3D instabilities in the wake, measured by the amount of vorticity in a plane perpendicular to the cylinder axis. Snapshots of vorticity iso-surfaces corresponding to the various stages of the flow development are also shown and allow one to visualize the A and B modes experimentally observed by Williamson [9]. More detailed results and discussions, as well as applications of the algorithm to the case of rotating cylinders, can be found in [6, 10]. Figure 3 shows a comparison between Williamson's experiments and vorticity isosurfaces when the 3D instabilities are fully developed. Figure 4 presents an animation illustrating how 3D instabilities spontaneously appear in the wake, and disappear under cylinder rotations. For clarity, only positive spanwise vorticity is shown on half of the cylinder span.

## 3. Immersed boundary techniques for particle methods

In immersed boundary approaches for simulations of bluff-body flows, flow equations are solved on both sides of the body boundary. No-slip boundary conditions are satisfied through the addition of a source term, confined on the boundary, in the right-hand side of the Navier–Stokes



**Figure 3.** 3D structures of vorticity. Left: the experimental work of [9] at  $Re = 270$ . Right: VIC method  $Re = 400$ , doubled spanwise by periodicity ( $t = 160$ ).



**Figure 4.** Animation showing 3D dynamics in the wake of a cylinder subject first to translation and then to translation and rotation.

equation. For finite-difference techniques, the construction of this force term and its interpolation on the grid is a delicate matter and seems to be critically linked to the particular discretization scheme [11]. Let us describe how boundary conditions on immersed boundaries are dealt with in vortex methods. We start with the no-through flow boundary condition and distinguish between grid-free and vortex-in-cell methods, then turn to the no-slip boundary condition.

### 3.1. No-through flow boundary condition for grid-free vortex methods

To start with, let us consider the Helmholtz decomposition (1)–(5). Let  $\bar{\omega}$  be an extension (for instance by 0 inside the body) of the vorticity in the entire space. Integral formulae allow us to write  $\psi$  and  $\phi$  in terms of extended vorticity and surface potentials. The vector potential  $\psi$  can then be expressed as

$$\psi = \int G(\mathbf{x} - \mathbf{y}) \bar{\omega}(\mathbf{y}) d\mathbf{y}$$

where  $G(\mathbf{x}) = (4\pi|\mathbf{x}|)^{-1}$ . One then computes  $\phi$  using a surface integral

$$\phi(\mathbf{x}) = \int_{\Gamma} G(\mathbf{x} - \mathbf{y})q(\mathbf{y}) \, d\mathbf{y}$$

where the single-layer potential  $q$  is obtained from the flux boundary condition (5) by solving the following Fredholm equation of the first kind on  $\Gamma$ :

$$-\frac{q(\mathbf{x})}{2} - \int_{\Gamma} \frac{\partial}{\partial \mathbf{n}_x} G(\mathbf{x} - \mathbf{y})q(\mathbf{y}) \, d\mathbf{y} = -(\nabla \times \boldsymbol{\psi}) \cdot \mathbf{n}(\mathbf{x}) \quad \text{for } \mathbf{x} \in \Gamma.$$

In a grid-free particle method, this equation is solved by mollifying the kernel  $G$ , then replacing  $\boldsymbol{\omega}$  by a set of particles and replacing surface integrals by summation over a finite number of source points on  $\Gamma$ . An important point, which makes particle methods well suited for immersed boundaries is that the solution to the above equation does not involve source terms in  $\Omega$ . The accuracy of the whole method, and in particular of the numerical integrations involved, is thus not conditioned by any compatibility of  $\Gamma$  and the underlying grid on which particles are initialized.

### 3.2. No-through flow boundary condition for vortex in cell methods

In this case, the Poisson equations (2) and (4) are solved by grid-based solvers in a finite size box containing the body. For wake calculations, the boundary conditions are, for instance, obtained from the potential flow corresponding to the far-field velocity profile. To solve (4), (5), our strategy consists in tagging grid-points near the obstacle—say at a distance less than a grid-size. If  $\tilde{\Gamma}$  denotes the resulting staircase-like boundary, one then needs to find  $g$ , a function vanishing outside  $\tilde{\Gamma}$  such that solving

$$\Delta\phi = g$$

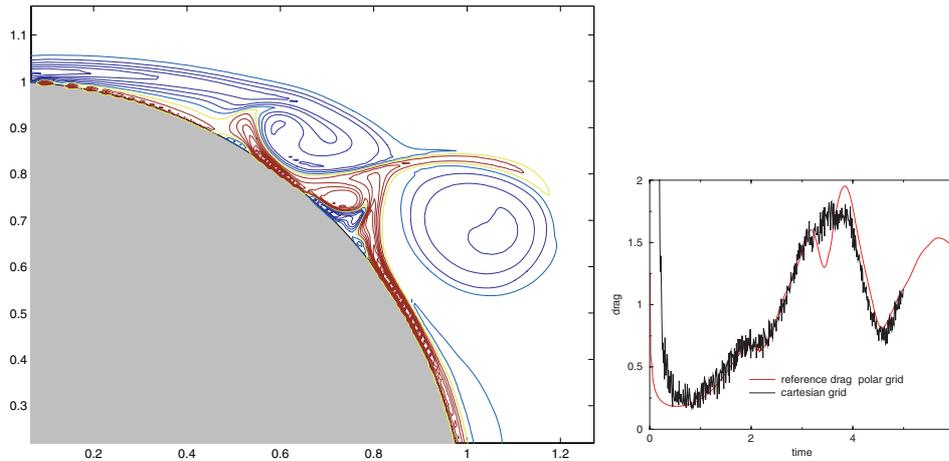
yields no through-flow on  $\tilde{\Gamma}$ :

$$\frac{\partial\phi}{\partial\mathbf{n}} = -(\nabla \times \boldsymbol{\psi}) \cdot \mathbf{n} \quad \text{on } \tilde{\Gamma}.$$

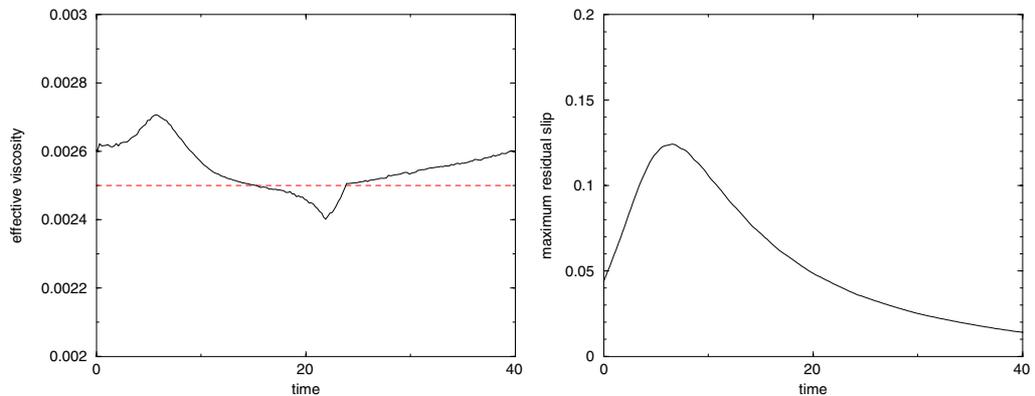
If  $\tilde{\Gamma}$  is discretized on  $N$  points, this constitutes a linear system, of size  $N$ , that we solve by a classical GMRES iterative solver. In this method, since the boundary condition is not exactly satisfied on  $\Gamma$  one might expect a loss of accuracy. However, for incompressible no-slip at the boundary  $\Gamma$  implies that  $\mathbf{u} \cdot \mathbf{n}(\mathbf{x}) = O(d(\mathbf{x}, \Gamma)^2)$ , and thus second-order accuracy is achieved.

### 3.3. Diffusion, remeshing and no-slip boundary condition

In a body-fitted vortex algorithm, the no-slip boundary condition is in general treated as the final stage in a viscous splitting algorithm. Diffusion is done first with no-flux boundary condition, then the resulting slip is cancelled by an appropriate flux of vorticity. Moreover, remeshing is, in general, performed by one-sided formulae near the boundary to enforce conservations of vorticity momentum within the fluid domain. If the underlying grid does not fit with the boundary, one may try to locally correct these formulae to achieve the same goals (see [12] for a method in this spirit). Our approach is more direct and is based on the observation that the vorticity flux introduced in the viscous splitting is designed to correct any wrong flux that might be introduced in the algorithm. This suggests that there is no need to use a particular boundary condition in the viscous scheme prior to the vorticity flux step, and that a simple-minded centred remeshing scheme extended through the boundary only introduces a spurious vorticity flux at the boundary that the vorticity flux scheme will fix in the final stage of the viscous splitting. In other words, remeshing and viscous algorithm designed for an unbounded flow can be used mostly unchanged in an immersed boundary code.



**Figure 5.** Flow past an impulsively started cylinder at  $Re = 9500$ . Left: close up of vorticity at  $t = 2.5$ . Right: comparison of drag with the results of [14].

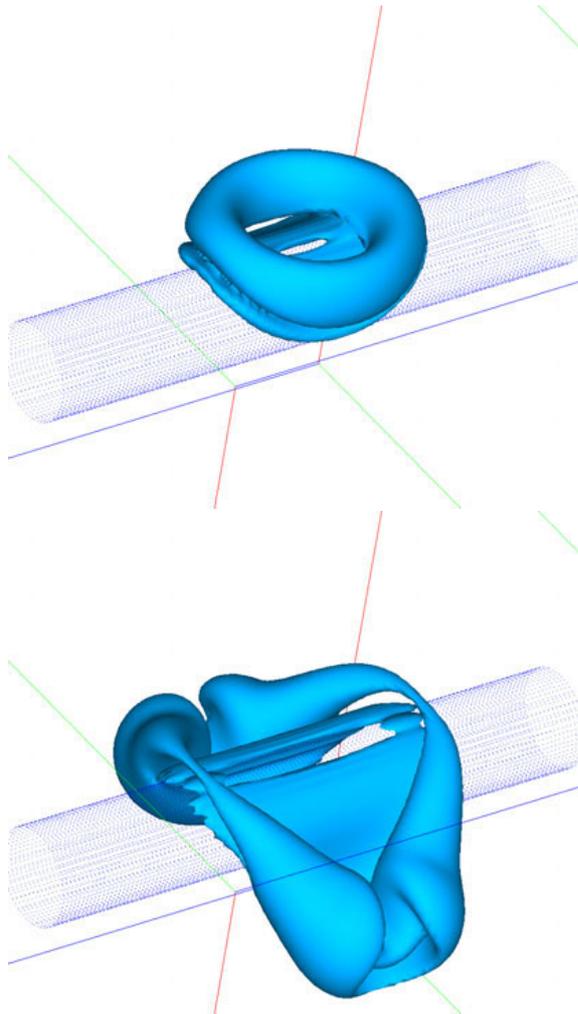


**Figure 6.** Collision ring-cylinder at  $Re = 400$ . Left: effective viscosity and target viscosity; right: residual maximum slip velocity.

### 3.4. Numerical results

We first show 2D results presented in [13] concerning an impulsively started cylinder at a Reynolds number of 9500. Poisson solvers, initialization and remeshing of particles were performed on Cartesian grids. The inter-particle and grid spacing was 0.01. This would correspond, in a polar-grid based code, to about 600 points on the surface of the cylinder, which is roughly the resolution used in reference calculations for this benchmark case. The computational box was the square  $[-3., 3.]^2$  for a cylinder of radius 1. Figure 5 shows a close up of vorticity contours around the separation point at time 2.5, and the drag history, in comparison with the reference results of [14]. The drag is computed using the formula

$$C_d = \frac{d}{dt} \int_{\Omega} \boldsymbol{\omega}(\mathbf{x}) \times \mathbf{x} \, d\mathbf{x}.$$



**Figure 7.** Collision ring-cylinder at  $Re = 400$  by the immersed-boundary vortex-in-cell method. Isosurfaces of vorticity magnitude at  $t = 10$  (top) and  $t = 60$  (bottom).

Despite the noise in the curve, which results from the fact that particle volumes are not correctly computed near the boundary, one can recognize the plateau at  $t \simeq 2$ , which is the signature of the secondary vorticity in the boundary cutting through the primary vorticity. The discrepancy at later times is probably due to the limited size of the computational box. Finally we consider the case of a ring hitting a 3D cylinder. Figure 6 gives the residual slip obtained throughout the simulation and the effective dissipation of the code. The grid spacing was in this case about one tenth of the cylinder radius. Despite this low resolution, the effective viscosity is rather satisfactory. The residual slip, relative to the maximal velocity peaks at a time which corresponds to the collision of the ring, then decreases.

Figure 7 shows isosurfaces of vorticity magnitude at times 10 and 60. The formation of a secondary vorticity ring agrees well with the results of [5] obtained with a cylindrical grid. More detailed validations of the method will be found in [8].

#### 4. Summary and conclusions

We have described two different particle methods for the DNS of 3D wakes: a body-fitted VIC method, and an immersed boundary method. The first method has been successfully used to recover the complex vorticity dynamics of cylinder wakes. Its robustness makes it a powerful tool that we are now using in optimization strategies for drag reduction. An immersed boundary method has also been designed to deal with more complex geometries. We have emphasized the fact that the concept of immersed boundaries is natural for particle methods and only requires minor alteration on plain VIC codes for unbounded flows on Cartesian grids. This brings some confidence that particle methods will complement more conventional grid-based methods for CFD in the future.

#### References

- [1] Ould-Salihi M L, Cottet G-H and El Hamraoui M 2000 Blending finite-differences and vortex methods for incompressible flow computations *SIAM J. Sci. Comput.* **22** 1655–74
- [2] Cottet G-H, Michaux B, Ossia S and Vanderlinden G 2002 A comparison of spectral and vortex methods in three-dimensional incompressible flows *J. Comput. Phys.* **175** 702–12
- [3] Ploumhans P, Winckelmans G S, Salmon J K, Leonard A and Warren M S 2002 Vortex methods for high-resolution simulation of three-dimensional bluff-body flows; application to the sphere at  $Re = 300, 500$  and 1000 *J. Comput. Phys.* **178** 427–63
- [4] Lindsay K and Krasny R 2001 A particle method and adaptive treecode for vortex sheet motion in 3D flow *J. Comput. Phys.* **172** 879–907
- [5] Poncet P 2001 Méthodes particulières pour la simulation des sillages tridimensionnels *PhD Thesis* University Joseph Fourier, Grenoble, France
- [6] Poncet P 2002 Topological aspects of the three-dimensional turbulent wake behind rotary oscillating circular cylinder, submitted
- [7] Cottet G-H and Koumoutsakos P D 2000 *Vortex Methods, Theory and Practice* (Cambridge: Cambridge University Press)
- [8] Cottet G-H and Poncet P 2002 Advances in direct numerical simulations of three-dimensional wall-bounded flows by particle in cell methods, in preparation
- [9] Williamson C H K 1996 Three-dimensional wake behind a cylinder *J. Fluid Mech.* **328** 328–45
- [10] Poncet P 2002 Vanishing of mode B in the wake behind a rotating circular cylinder *Phys. Fluids* **14** 2021–4
- [11] Fadlun E A, Verzicco R, Orlandi P and Mohd-Yusof J 2000 Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations *J. Comput. Phys.* **161** 35–60
- [12] Ploumhans P and Winckelmans G S 2000 Vortex methods for high-resolution simulations of viscous flow past bluff bodies of general geometry *J. Comput. Phys.* **165** 354–406
- [13] Cottet G-H 2000 Particles and immersed boundaries *Workshop on Penalization Methods and No-slip Flows (Bordeaux, France, June 2000)* unpublished
- [14] Koumoutsakos P D and Leonard A 1995 High-resolution simulations of the flow around an impulsively started cylinder using vortex methods *J. Fluid Mech.* **296** 1–38