First Experiments with the Simulation of Particulate Flows

M. Uhlmann

Dept. Combustibles Fósiles, CIEMAT
Avda. Complutense 22, 28040 Madrid (Spain)
markus.uhlmann@ciemat.es

February 2003

First Experiments with the Simulation of Particulate Flows

Uhlmann, M. (CIEMAT)
76 pp. 32 figs. 33 refs.

Abstract:

Several variants of a Eulerian-Lagrangian method for the simulation of particulate flows are implemented in a finite-difference framework. All methods have in common that they represent the presence of the solid fraction by means of artificial volume forces in the momentum equation of the fluid phase. Thereby, explicit gridding of the moving particles is avoided and a fixed grid can be used. The computations show that the direct forcing method (Kim *et al.*, 2001) is not adequate for our purposes due to large oscillations in the hydro-dynamical forces. The immersed boundary method of Peskin (2002) does provide accurate predictions of particle motion, however at the cost of a small time step.

Primer Experiencia con la Simulación Numérica de Flujos con Partículas

Uhlmann, M. (CIEMAT)
76 pp. 32 figs. 33 refs.

Resumen:

Algunas variantes de un método Euleriano-Lagrangiano para la simulación de flujos con partículas sólidas han sido implementados mediante diferencias finitas. Todos aquellos métodos representan la acción de cuerpos sólidos por fuerzas volumétricas artificiales añadidas a las ecuaciones de cantidad de movimiento de la fase fluida. De este modo la malla no tiene que ser adaptada a la posición de las partículas permitiendo la utilización de una malla fija. Nuestros cálculos muestran que el método de fuerzas directas (Kim *et al.*, 2001) no es adecuado para los fines de simulación de partículas móviles debido a la aparición de grandes oscilaciones de las fuerzas hidrodinámicas. El método de fronteras sumergidas (Peskin 2002) permite una predicción fiable del movimiento de partículas pero con el inconveniente de pasos de tiempo pequeños.

Contents

Chapter 1

Introduction

Particulate flows occurs frequently in nature—e.g. sedimentation of grains of sand near the bottom of rivers—and technological applications such as fluidized beds. Their accurate prediction is of large importance for industrial purposes. However, in the past feasible computation techniques have been limited to very idealized situations and lowest-order systems. Recently, some progress has been made through the use of various methods which could be collectively called "fictitious domain methods". In this approach, particles are not explicitly accounted for by the computational grid; instead, an artificial volume force term is added to the momentum equation of Navier-Stokes with the purpose of locally forcing the solution to obey the adequate constraint near the fluid-solid interface (no-slip condition) or within the solid domain (rigidity condition; zero-deformation-rate condition). Therefore, the mesh does not need to be adapted to the presence of particles and very fast numerical methods can be used to solve the fluid equations.

The heart of any such method is the way the volume forces are formulated. More specifically, one can classify the most promising variants from the literature by means of three criteria:

(i) which constraint is used?

(ii) how is the constraint applied?

(iii) how is the transfer between Eulerian and Lagrangian positions performed?

In the "direct forcing method", the constraint is immediately imposed upon the fluid equations: Fadlun *et al.* [1] use an implicit formulation of the no-slip condition (i.e. they directly modify the resulting linear system); Kim, Kim and Choi [2] resort to an explicit formulation of the no-slip condition (i.e. performing an explicit tentative step in order to obtain a value for the volume force); Kajishima *et al.* [3] use again an explicit formulation, but for the condition of rigid body movement in the particle sub-domain.

Pan *et al.* [4] and Patankar [5] use the method of distributed Lagrange multipliers in order to impose the condition of rigidity and zero-deformation-rate, respectively, within the particle sub-domain.

Höfler and Schwarzer [6] build upon the "immersed boundary method" of Peskin [7] where singular constraint forces at the fluid-solid interface are formulated by means of virtual springs and dampers (cf. [8]). Here, the boundary forces are transferred to the Eulerian grid by means of a regularized Dirac function with desirable continuity and smoothness properties. In the previously mentioned methods, linear or volume-fraction-averaged interpolation was used.

Finally, Zhang and Prosperetti [9] match a local analytic solution of Stokes flow around spheres (or cylinders in two dimensions) with the numerical solution of the full Navier-Stokes equations elsewhere. This method is iterative and the matching is performed via a "cage" of node points which surround each particle.

In terms of achieved realism of the simulations, it should be pointed out that Höfler and Schwarzer [6] have managed the largest number of particles, $\mathcal{O}(10000)$, albeit at relatively low

particle-diameter-based Reynolds numbers, $\mathcal{O}(10)$. Pan *et al.* [4] did simulate the fluidization of 1024 spheres at Reynolds numbers $\mathcal{O}(1000)$; however, the flow was only "pseudo-3d" since the domain spanned only 11 nodes in the third coordinate direction. Kajishima *et al.* [10] performed three-dimensional simulations of turbulent flow with $\mathcal{O}(1000)$ particles at a Reynolds number of 400. Their method, however, seems to be adequate only for relatively heavy particles (density ratio larger than 1.5).

In the present study we aim at implementing and validating a fictitious domain method in two-dimensions. For this purpose we have chosen to analyze in more detail the immersed boundary method and the direct forcing method.

The present document is arranged as follows. After presenting the basic fluid solver in § 2 we turn to the main concern, the fluid-particle coupling, in § 3. Chapter § 4 deals with the determination of the particle motion itself. It should be noted that inter-particle collision modeling has been excluded for the time being and will be considered in detail in the future.

Chapter 2

Basic Navier-Stokes solver

We consider the incompressible Navier-Stokes equations

$$\partial_t \mathbf{u} + \nabla p = -(\mathbf{u}\cdot\nabla)\mathbf{u} + \nu\nabla^2\mathbf{u} + \mathbf{f} \tag{2.1a}$$
$$\nabla\cdot\mathbf{u} = 0 \tag{2.1b}$$

subject to appropriate boundary conditions. Here $\mathbf{u}$ is the fluid velocity vector, p the pressure divided by the fluid density and ν the kinematic viscosity. The volume force $\mathbf{f}$ regroups the body forces arising from the solid-fluid coupling, to be introduced in § 3. In the present chapter we will set $\mathbf{f} = 0$ for simplicity. Please note that p is the pressure solely arising from the fluid motion, i.e. the pressure resulting from gravity has been canceled against the corresponding body force [11, p.176].

2.1 Projection method and time integration

We employ an *incremental-pressure projection method* (cf. [12]) for splitting the system (2.1) into two fractional steps. Using a semi-implicit scheme for the viscous terms and a three-step low-storage Runge-Kutta method with explicit non-linear terms, the semi-discrete system can be written as follows:

$$\frac{\mathbf{u}^* - \mathbf{u}^{k-1}}{\Delta t} = -\gamma_k\left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-1} - \zeta_k\left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-2} - 2\alpha_k\nabla p^{k-1} + \alpha_k\nu\nabla^2\left(\mathbf{u}^* + \mathbf{u}^{k-1}\right) \tag{2.2a}$$
$$\nabla^2\phi^k = \frac{\nabla\cdot\mathbf{u}^*}{2\alpha_k\Delta t} \tag{2.2b}$$
$$\mathbf{u}^k = \mathbf{u}^* - 2\alpha_k\Delta t\nabla\phi^k \tag{2.2c}$$
$$p^k = p^{k-1} + \phi^k - \alpha_k\Delta t\,\nu\nabla^2\phi^k \tag{2.2d}$$

where $k = 1, 2, 3$ is the Runge-kutta step count (with the level $k = 3$ being equivalent to $n + 1$), u^* the predicted, intermediate velocity and the intermediate variable ϕ is sometimes called "pseudo-pressure". The following set of coefficients, first published in reference [13], leads to overall second-order temporal accuracy for both velocity and pressure (cf. § A.1):

$$\alpha_k = \left\{\frac{4}{15}, \frac{1}{15}, \frac{1}{6}\right\}, \quad \gamma_k = \left\{\frac{8}{15}, \frac{5}{12}, \frac{3}{4}\right\}, \quad \zeta_k = \left\{0, -\frac{17}{60}, -\frac{5}{12}\right\}. \tag{2.3}$$

2.2 Spatial discretization

We use a second order finite-difference scheme on a uniform, staggered cartesian mesh. The computational domain $\Omega = \prod_{\alpha=1}^{nd}[0, L_\alpha]$ (with the number of spatial dimensions nd) is discretized

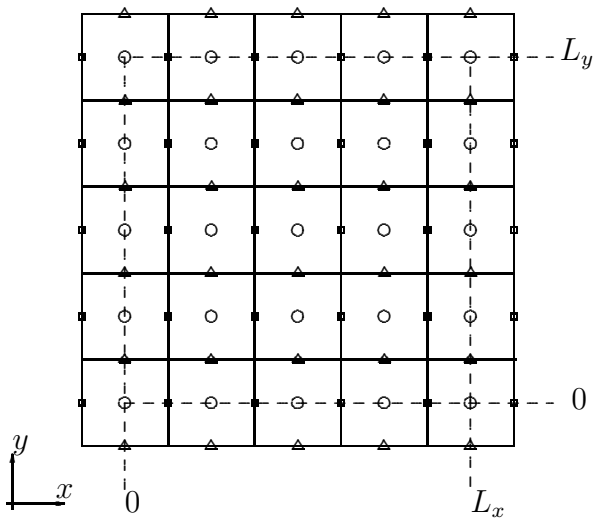

Figure 2.1: Discretization of a two-dimensional domain $[0, L_x] \times [0, L_y]$ with a uniform, staggered grid: $\circ$ - pressure nodes; $\Box$ – nodes of x-component of velocity; $\triangle$ – nodes of y-component of velocity .

as follows. The collocation points of the β-component of velocity in the α-direction are given by:

$$x_\alpha^\beta(i) = (i - 1 - \frac{1}{2}\delta_{\alpha\beta})\Delta x_\alpha \qquad i = 1, \ldots, n_\alpha + \delta_{\alpha\beta} \quad , \tag{2.4}$$

and the collocation points for pressure and pseudo-pressure are:

$$x_\alpha^p(i) = (i - 1)\Delta x_\alpha \qquad i = 1, \ldots, n_\alpha \quad , \tag{2.5}$$

$\Delta x_\alpha = L_\alpha/(n_\alpha - 1)$ being the mesh width in the α-direction and n_α the number of (pressure) grid points used in the α-direction (cf. figure 2.1).

Let us first restrict our attention to the case with two spatial dimensions ($nd = 2$) with $\mathbf{x} = (x, y)$ and $\mathbf{u} = (u, v)$ and the following short-hand notation for the discrete, nodal values of the variables:

$$u_{i,j} = u(x_1^1(i), x_2^1(j)) \,, \quad v_{i,j} = v(x_1^2(i), x_2^2(j)) \,, \quad p_{i,j} = p(x_1^p(i), x_2^p(j)) \,. \tag{2.6}$$

We now define the discrete averaging operators:

$$A_1^+ \, \phi = \frac{\phi_{i+1,j} + \phi_{i,j}}{2} \,, \quad A_1^- \, \phi = \frac{\phi_{i,j} + \phi_{i-1,j}}{2} \,, \tag{2.7}$$

noting that the operators with subscript "2" work analogously on the second index of the argument. Similarly, we define the discrete difference operators:

$$D_1^+ \, \phi = \frac{\phi_{i+1,j} - \phi_{i,j}}{\Delta x} \,, \quad D_1^- \, \phi = \frac{\phi_{i,j} - \phi_{i-1,j}}{\Delta x} \,, \quad D_1 \, \phi = \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x} \,. \tag{2.8}$$

From the above ingredients we construct the discrete divergence operator:

$$\mathbf{D}_h \cdot \mathbf{u} = D_1^+ u + D_2^+ v \quad , \tag{2.9}$$

the discrete Laplacian operator

$$L_h \, \phi = \sum_{\alpha=1}^{nd} D_\alpha^+ D_\alpha^- \phi \quad , \tag{2.10}$$

the discrete gradient operator

$$\mathbf{G}_h \, \phi = (D_1^- \phi, \, D_2^- \phi) \quad , \tag{2.11}$$

and finally the discrete advection operator

$$H_h(\mathbf{u})\mathbf{u} = \sum_{\beta=1}^{nd} D_\beta^+ \left((A_\beta^- u_\alpha)(A_\alpha^- u_\beta) \right) \quad . \tag{2.12}$$

The divergence operator $\mathbf{D}_h$ and the gradient operator $\mathbf{G}_h$ are both only one cell wide because their values are needed at an intermediate point. Please note that the above difference stencils are *a priori* conservative in the momentum equation (in particular, the divergence form for the advection term has been used); they are also conservative in the transport equation of kinetic energy if continuity is exactly satisfied [14].

We are now in the position to write the discrete form of equations (2.2), viz.

$$\begin{aligned}
\frac{\mathbf{u}^* - \mathbf{u}^{k-1}}{\Delta t} &= -\gamma_k H_h(\mathbf{u}^{k-1})\mathbf{u}^{k-1} - \zeta_k H_h(\mathbf{u}^{k-2})\mathbf{u}^{k-2} - 2\alpha_k \mathbf{G}_h p^{k-1} \\
&\quad + \alpha_k \nu L_h \left(\mathbf{u}^* + \mathbf{u}^{k-1} \right) && (2.13a) \\
L_h \phi^k &= \frac{\mathbf{D}_h \cdot \mathbf{u}^*}{2\alpha_k \Delta t} && (2.13b) \\
\mathbf{u}^k &= \mathbf{u}^* - 2\alpha_k \Delta t \mathbf{G}_h \phi^k && (2.13c) \\
p^k &= p^{k-1} + \phi^k - \alpha_k \Delta t \, \nu L_h \phi^k && (2.13d)
\end{aligned}$$

where equations (2.13a) and (2.13c) are evaluated at the velocity nodes x_α^β and equations (2.13b) and (2.13d) are evaluated at the pressure nodes x_α^p.

2.3 Domain boundary conditions

Velocity values at the boundaries are either determined by a periodicity condition or prescribed as Dirichlet conditions. The only case which deserves further mention is the one of an outflow. There we use a convective condition with the purpose of evacuating flow structures without perturbing the solution in the interior of the domain. The equation which is solved at the location of an outlet with normal vector $\mathbf{n}$ can be written as follows:

$$\partial_t \mathbf{u} + c \frac{\partial \mathbf{u}}{\partial n} = 0 \quad , \tag{2.14}$$

where c is the convective velocity obtained by averaging over the entire outflow plane, i.e. $u_c = \int_0^{L_\alpha} \mathbf{u}\cdot\mathbf{n} \mathrm{d}x_\alpha / L_\alpha$. However, in order to satisfy the global compatibility condition $\int_\Gamma \mathbf{u}\cdot\mathbf{n} \mathrm{d}s = 0$, the value of u_c needs to be chosen accordingly (e.g. if there is only a single outlet plane and a uniform inflow velocity u_∞ at the opposite plane, then $u_c = u_\infty$ for compatibility). Using a first-order upwind discretization of the spatial derivative and a third-order Runge-Kutta scheme in time, the equation reads for the β-component of velocity and e.g. a horizontal outflow at $x_1^\beta(n_1 + \delta_{1\beta})$:

$$\begin{aligned}
u^k_{n_1+\delta_{1\beta},j} = u^{k-1}_{n_1+\delta_{1\beta},j} \quad &- \quad \gamma_k \frac{\Delta t}{\Delta x} c^{k-1} \left(u^{k-1}_{n_1+\delta_{1\beta},j} - u^{k-1}_{n_1+\delta_{1\beta}-1,j} \right) \\
&- \quad \zeta_k \frac{\Delta t}{\Delta x} c^{k-2} \left(u^{k-2}_{n_1+\delta_{1\beta},j} - u^{k-2}_{n_1+\delta_{1\beta}-1,j} \right) \quad . && (2.15)
\end{aligned}$$

The solution of (2.13) does not require explicit boundary conditions for the pressure [15]. However, in practice one needs to supply boundary conditions to the solution procedure of the projection equation (2.13b). For simplicity we generally (i.e. except for a periodic case) use a homogeneous Neumann condition for pseudo-pressure, viz.

$$\frac{\partial p}{\partial n} = 0 \quad . \tag{2.16}$$

2.4 Solution of Helmholtz and Poisson problems

The discretized Helmholtz (2.13a) and Poisson equations (2.13b) can be recast into the form of the following linear problem:

$$a\,\phi_{i-1,j} + b\,\phi_{i,j} + c\,\phi_{i+1,j} + \phi_{i,j-1} - 2\phi_{i,j} + \phi_{i+1,j} = \Delta y^2\, f_{i,j} \quad , \tag{2.17}$$

to be solved for the unknowns $\phi_{i,j}$. The right-hand-side expression $f_{i,j}$ can be directly evaluated at all interior points, i.e those for which $2 \leq i,j \leq n_\alpha - 1 + \delta_{\alpha\beta}$ ($2 \leq i,j \leq n_\alpha - 1$, respectively for pseudo-pressure) due to the small widths of the stencils (2.9)-(2.12). At these points, the coefficients are simply:

$$a = c = \frac{\Delta y^2}{\Delta x^2}\,, \quad b = -\frac{2\Delta y^2}{\Delta x^2} + \lambda \Delta y^2 \quad , \tag{2.18}$$

with λ being the Helmholtz coefficient, i.e. $\lambda = -1/(\alpha_k \nu \Delta t)$ in equation (2.13a) and $\lambda = 0$ in equation (2.13b). The remaining (boundary) points are eliminated from the system (2.17) by modifying the coefficients a, b, c and right-hand-side f accordingly. For the actual solution of the resulting systems we have resorted to routines from the FISHPACK library [16, 17] especially adapted to our grid arrangement.

Alternatively, we have used the multi-grid solver D03EDF from the NAG library [18]. This second solution method was first used simply as a reference; since it allows for the values of the coefficients in equation (2.17) to vary freely in space, it was also used in situations where it was necessary to fix one nodal value of pseudo-pressure. In order for the method to be efficient, the number of grid points should be of the form $n_\alpha = m2^{l-1} + 1$ with $m, l \geq 2$. This in turn implies that we cannot (efficiently) use the multi-grid procedure simultaneously for the solution of the Helmholtz *and* Poisson equations.

Tests with FISHPACK on an Intel x86 machine as well as on an SGI MIPS R14000 show the convergence rate of figure 5.1. At linear dimensions of above $N_\alpha = 2048$, the convergence breaks down, presumably due to the accumulation of round-off errors.

2.5 Validation of the basic fluid solver

2.5.1 Taylor-Green vortices

We validate the Navier-Stokes solver for the case of decaying Taylor-Green vortices, for which an anlytical solution is available. The time-dependent velocity field is given by:

$$u(x,y,t) = \sin(k_x x)\cos(k_y y)e^{-(k_x^2+k_y^2)\nu t}, \quad v(x,y,t) = \frac{k_x}{k_y}\sin(k_y y)\cos(k_x x)e^{-(k_x^2+k_y^2)\nu t}, \tag{2.19}$$

and pressure by

$$p(x,y,t) = \frac{1}{2}\left(\cos^2(k_y y)\frac{k_x^2}{k_y^2} - \sin^2(k_x x)\right)e^{-2(k_x^2+k_y^2)\nu t}, \tag{2.20}$$

where k_x, k_y are the wavenumbers of the flow in the two coordinate directions. For this test we set $k_x = k_y = 2\pi$ and use as the domain size $L_x = L_y = 1$. Periodicity conditions are used at all boundaries and for all variables. The exact solution at $t = 0$ is imposed to initialise the computation.

Firstly, we verify the temporal convergence of the fractional step method (2.2) by a fully de-aliased Fourier pseudo-spectral discretization of all spatial operators. Using 36 modes in each direction, the remaining error is purely due to the temporal discretization. The equations have been advanced for a single time step with the viscosity being set to $\nu = 10^{-2}$. Figure 5.2 shows the convergence rate for velocity to be $\mathcal{O}(\Delta t^3)$ and for pressure to be $\mathcal{O}(\Delta t^2)$ such that the *accuracy of the scheme* is of second order for both variables (cf. remarks on the time accuracy in § A.1).

The convergence with grid size of the finite-difference scheme (2.13) is evaluated for a fixed time step of $\Delta t = 10^{-4}$ and integrating up to $t_{fin} = 10^{-3}$, while varying the number of grid

nodes in both spatial dimensions simultaneously. From figure 5.3 it is obvious that the expected second-order spatial convergence rate for both velocity and pressure is obtained.

Chapter 3

Fluid-particle coupling

The main difficulty in the numerical treatment of fluid-particle systems is to find an accurate and efficient method for the interaction between the two phases. Since we have focused upon fixed-grid methods, we have already ruled-out the costly option of "gridding" the fluid phase in between solid particles repeatedly at every instant; by choosing a finite-difference discretization we also leave aside finite-element-based Lagrange multiplier methods [4]. In the following we will therefore restrict our attention to three different methods of imposing the adequate constraints stemming from the motion of solids within the framework of the continuum flow of the fluid:

Direct forcing method. The no-slip condition is imposed at the interfaces between solid and fluid parts of the domain under consideration. This method has been developed for the simulation of fluid flow in complex geometries [1, 2], and only one application with moving boundaries has previously appeared in the literature [19]. We will see in § 3.1 that this method does not prove adequate for the simulation of fluid-particle systems.

Immersed boundary method. One can compute the constraint forces by attaching imaginary "springs" and "dampers" between tracer locations in the fluid phase and corresponding locations on the surface of the particles [6, 7, 20]. This setup serves to drive the fluid such that the difference between the two locations is reduced to a negligible value on time scales smaller than those of the phenomena under consideration. The disadvantage is that non-physical parameters (spring and damper constants) have to be chosen manually.

A variant of the direct forcing method based upon rigidity. The constraint of rigid-body motion is explicitly imposed within the solid portion of the domain in this method due to Kajishima [3, 10]. One specific issue is the verification of the divergence-free condition for the final field, which is not automatically guaranteed with the original method.

Aside from the formulation of the solid-body constraint, another key point which distinguishes the numerical methods is the way the respective singular forces (or velocitiies) are mapped to the Eulerian grid.

In order to keep the treatment of the two phases as modular as possible we choose an explicit coupling formulation, i.e. always using the most recent *known* values of the quantities describing the opposite phase. This so-called "loose coupling" corresponds to introducing a time lag of one time step and should have only a small effect when using sufficiently small time step sizes.

3.1 The direct forcing method: no-slip

The basic idea—as in the immersed boundary method of § 3.2—is to express the action of the solid portion upon the fluid by appropriately formulated volume forces $\mathbf{f}$ to be added to the

fluid equations locally (i.e. equations (2.1) with $\mathbf{f} \neq 0$). Here, the volume forces are not formally prescribed but obtained by requiring that the resulting velocity field verifies the non-slip conditions at the solid-fluid interface. This method is called "direct forcing" by the authors [1, 21].

Two variants of the method have been proposed, which we will call "implicit" and "explicit" according to how the no-slip condition is incorporated into the basic fluid dynamics equations at solid points in the interior of the computational domain.

3.1.1 Explicit method

This variant was proposed by Kim, Kim and Choi [2]. In order to obtain an explicit expression for the volume force $\mathbf{f}$ the predictor step for velocity (analogous to equation 2.2a) is first discretized by a fully explicit scheme (forward Euler for the viscous terms), which gives for the component in the β-direction:

$$f_\beta^k = \frac{U_\beta^k - u_\beta^{k-1}}{\Delta t} - 2\alpha_k \nu \nabla^2 u_\beta^{k-1} + 2\alpha_k \frac{\partial p^{k-1}}{\partial x_\beta} + \gamma_k [(\mathbf{u}\cdot\nabla)\mathbf{u}]_\beta^{k-1} + \zeta_k [(\mathbf{u}\cdot\nabla)\mathbf{u}]_\beta^{k-2} \quad (3.1)$$

at points $\mathbf{x}_h^\beta$ which coincide with the solid-fluid interface; U_β^k is the desired interface velocity, e.g. $U_\beta^k = 0$ for a particle at rest or a stationary wall. The calculated value for the volume force is then simply added to the right-hand-side (RHS) of the original implicit predictor equation (2.2a) and it is solved for $\mathbf{u}^*$ and updated via (2.2c), using the pseudo-pressure obtained from (2.2b). In practice, however, the interface does not coincide with a grid node location and one needs to resort to an interpolation scheme.

3.1.1.1 Interpolation procedure

First, a preliminary predicted vector field $\tilde{\mathbf{u}}^k$ is obtained by a fully explicit step with the volume force set to zero, viz.

$$\frac{\tilde{\mathbf{u}}^k - \mathbf{u}^{k-1}}{\Delta t} = -\gamma_k \left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-1} - \zeta_k \left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-2} - 2\alpha_k \nabla p^{k-1} + 2\alpha_k \nu \nabla^2 \mathbf{u}^{k-1} \quad . \quad (3.2)$$

This step is cheap and the result solely serves the purpose of interpolation in order to obtain the desired velocity at grid nodes adjacent to the solid-fluid interface.

The interpolation of the interface velocity u_β^Γ upon the adjacent grid node U_β^k using the field $\tilde{u}_\beta^k$ can be expressed formally as $\mathcal{I}(U_\beta^k, \tilde{u}_\beta^k, u_\beta^\Gamma) = 0$. A second-order interpolation scheme is used, i.e. involving only the direct internal and external neighbors of the interface. In general, the interpolation is bi-linear (fig. 3.1); however, in order to conserve an explicit nature of the interpolation formula, the scheme switches to linear interpolation locally if the bi-linear formula at a given point involves more than one unknown (fig. 3.2). Please note that due to the staggered grid arrangement, the interpolation stencil is not the same for all velocity components but needs to be evaluated separately.

Let $\mathcal{B}^\beta$ be the set of grid nodes $\mathbf{x}_h^\beta$ which are internal (i.e. located inside the solid) direct neighbors of the solid-fluid interface. For each $\mathbf{x}_h^\beta \in \mathcal{B}^\beta$ we identify a point $\mathbf{x}_p$ as the closest perpendicular foot from $\mathbf{x}_h^\beta$ on the interface (cf. figure 3.1). The set of the four neighbors of $\mathbf{x}_p$ is denoted as $\mathcal{C}^\beta = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$ and the number of its elements simultaneously contained in the set $\mathcal{B}^\beta$:

$$n_{\mathcal{B}}(\mathbf{x}_h^\beta) = |\mathcal{C}^\beta \cap \mathcal{B}^\beta| \quad . \quad (3.3)$$

We can then define the interpolation expression as follows:

$$\mathcal{I}(\tilde{u}_\beta^k, U_\beta^k) = \begin{cases} \left(\tilde{u}_\beta^k(\mathbf{x}_1)\cdot(x_2 - x_p) + \tilde{u}_\beta^k(\mathbf{x}_2)\cdot(x_p - x_1)\right)(y_3 - y_p) \\ + \left(\tilde{u}_\beta^k(\mathbf{x}_4)\cdot(x_2 - x_p) + \tilde{u}_\beta^k(\mathbf{x}_3)\cdot(x_p - x_1)\right)(y_p - y_1) \\ -u_\beta^\Gamma(\mathbf{x}_p)\Delta x \Delta y & \text{if } n_{\mathcal{B}}(\mathbf{x}_h^\beta) = 1 \\ \\ \mathcal{I}_{lin}(\tilde{u}_\beta^k, U_\beta^k) \quad \text{(linear interpolation)} & \text{else} \end{cases} \quad (3.4)$$

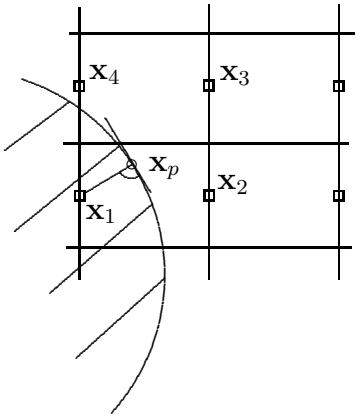

Figure 3.1: Case of bi-linear interpolation of the desired velocity at the point $\mathbf{x}_p$ to the grid node $\mathbf{x}_1$ with the aid of velocities at nodes $\mathbf{x}_2$, $\mathbf{x}_3$, $\mathbf{x}_4$.

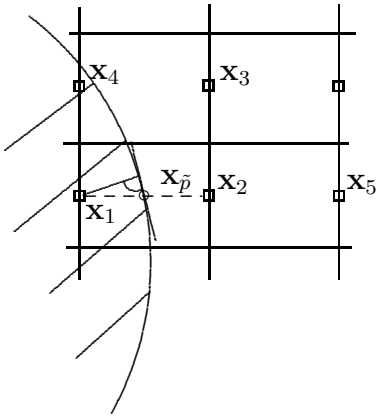

Figure 3.2: Case of linear interpolation of the desired velocity at the point $\mathbf{x}_{\tilde{p}}$ to the grid node $\mathbf{x}_1$ with the aid of velocities at nodes $\mathbf{x}_2$ and $\mathbf{x}_5$ only (since node $\mathbf{x}_4$ also needs interpolation).

where $u_\beta^\Gamma(\mathbf{x}_p)$ is the (time-dependent) desired velocity value of the interface at location $\mathbf{x}_p$. The interpolation relation $\mathcal{I}(U_\beta^k, \tilde{u}_\beta^k) = 0$ needs to be solved explicitly for $U_\beta^k = \tilde{u}_\beta^k(\mathbf{x}_i)$ with i such that $\mathbf{x}_i = \mathbf{x}_h^\beta$ (e.g. $i = 1$ in the situation of figure 3.1).

In the case of linear interpolation according to (3.4) a point $\mathbf{x}_{\tilde{p}}$ is defined as the intersection point on the nearest interface and on the connecting line between $\mathbf{x}_h^\beta$ and the closest neighbor in the fluid phase ($\mathcal{C}^\beta \cup \mathcal{B}^\beta$; cf. figure 3.2). According to the distance $|x_{\tilde{p}} - x_1|$, a two-node or a three-node linear formula is used. In the case of figure 3.2 this leads to:

$$\mathcal{I}_{lin}(\tilde{u}_\beta^k, U_\beta^k) = \begin{cases} \tilde{u}_\beta^k(\mathbf{x}_1) \cdot (x_2 - x_{\tilde{p}}) + \tilde{u}_\beta^k(\mathbf{x}_2) \cdot (x_2 - x_{\tilde{p}}) - u_\beta^\Gamma(\mathbf{x}_p)\Delta x & \text{if } |x_{\tilde{p}} - x_1| \leq |x_2 - x_{\tilde{p}}| \\ \\ \tilde{u}_\beta^k(\mathbf{x}_1)\Delta x - u_\beta^\Gamma(\mathbf{x}_p)2\Delta x & \\ +\tilde{u}_\beta^k(\mathbf{x}_2) \cdot (x_5 + x_1 - 2x_{\tilde{p}}) + \tilde{u}_\beta^k(\mathbf{x}_5) \cdot (x_2 + x_1 - 2x_{\tilde{p}}) & \text{else} \end{cases} \tag{3.5}$$

3.1.1.2 Recap of the algorithm

For convenience we summarize the complete algorithm for the explicit direct forcing method. Using the spatially discrete notation analogous to (2.13) and eliminating the explicit volume force term from the equations, the algorithm reads as follows.

(i) Compute the preliminary predicted velocity field ($\forall\, \mathbf{x}_h$) from:

$$\begin{aligned} \tilde{\mathbf{u}}^k &= \Delta t \alpha_k \nu L_h \mathbf{u}^{k-1} \\ &\quad + \Delta t \underbrace{\left(\frac{\mathbf{u}^{k-1}}{\Delta t} - \gamma_k H_h(\mathbf{u}^{k-1})\mathbf{u}^{k-1} - \zeta_k H_h(\mathbf{u}^{k-2})\mathbf{u}^{k-2} - 2\alpha_k \mathbf{G}_h p^{k-1} + \alpha_k \nu L_h \mathbf{u}^{k-1} \right)}_{\mathcal{H}^k} \end{aligned} \tag{3.6}$$

(ii) Interpolate the desired interface velocity $u_\beta^\Gamma(t)$ upon the grid to obtain U_β^k via:

$$\mathcal{I}(U_\beta^k, \tilde{u}_\beta^k, u_\beta^\Gamma(t)) = 0 \quad \forall \quad \mathbf{x}_h^\beta \in \mathcal{B} \quad . \tag{3.7}$$

(iii) Solve the Helmholtz equation for the predicted field $\mathbf{u}^*$, corrected at the nodes adjacent to

the solid-fluid interface:

$$(L_h - \frac{1}{\alpha_k \nu \Delta t})\mathbf{u}^* = \begin{cases} L_h \mathbf{u}^{k-1} - \dfrac{\mathbf{U}^k}{\nu \alpha_k \Delta t} & \forall \quad \mathbf{x}_h^\beta \in \mathcal{B} \\ -\dfrac{\mathcal{H}^k}{\nu \alpha_k} & \text{else} \end{cases} \quad . \tag{3.8}$$

(iv) Solve for pseudo-pressure ϕ^k $(\forall\, \mathbf{x}_h)$:

$$L_h \phi^k = \frac{\mathbf{D}_h \cdot \mathbf{u}^*}{2\alpha_k \Delta t} \quad . \tag{3.9}$$

(v) Update velocity and pressure $(\forall\, \mathbf{x}_h)$:

$$\mathbf{u}^k = \mathbf{u}^* - 2\alpha_k \Delta t \mathbf{G}_h \phi^k \tag{3.10a}$$

$$p^k = p^{k-1} + \phi^k - \alpha_k \Delta t\, \nu L_h \phi^k \quad . \tag{3.10b}$$

3.1.2 Implicit method

In the original proposal of Fadlun *et al.* [1], the no-slip constraint on the surface of the solid boundaries was imposed implicitly. This means w.r.t. the algorithm of § 3.1.1.2 that step (i) is obsolete and that the interpolation of step (ii) operates on the yet-to-be-determined field, i.e. $\mathcal{I}(u^*_\beta, u^\Gamma_\beta(t)) = 0$. Consequently, step (iii) here involves the solution of a linear algebraic system of equations with a structure which is different from the original simple form (2.17) by simply entering the implicit interpolation formula into the system matrix, viz.

$$\begin{cases} \mathcal{I}(u^*_\beta, u^\Gamma_\beta(t)) = 0 & \forall \quad \mathbf{x}_h^\beta \in \mathcal{B} \\ (L_h - \dfrac{1}{\alpha_k \nu \Delta t})\mathbf{u}^* = -\dfrac{\mathcal{H}^k}{\nu \alpha_k} & \text{else} \end{cases} \quad . \tag{3.11}$$

This results in a sparse system. Since we have only considered this variant as a reference method for comparison with the related explicit method, we have not optimized the solution of the system (3.11) but have used the relatively costly sparse matrix solver F11DEF from the NAG library with Bi-CGSTAB as the underlying algorithm.

The steps (iv) and (v) are carried out as in the explicit variant of the method.

3.1.3 Validation

3.1.3.1 Plane Poiseuille flow

In this case we simulate plane Poiseuille flow between two channel walls which are mimicked by embedded boundaries. The computational domain was chosen as $\Omega = [0,1] \times [0,1]$ with the no-slip walls (i.e. the embedded boundaries) located at $y_b = 0.1$, $y_t = 0.9$; the channel half-width is therefore $h = 0.4$. The viscosity is set to $\nu = 0.2$ which gives a value for the wall-shear-velocity-based Reynolds number of

$$Re_\tau \equiv \frac{u_\tau h}{\nu} = \frac{\sqrt{\nu u_{,y}(y = y_w)}h}{\nu} = \frac{\sqrt{2\nu}h}{\nu} \approx 1.265 \quad . \tag{3.12}$$

The initial velocity field is defined as the sum of the parabolic steady profile and a divergence-free perturbation, viz.

$$u(x, y, t_0) = (1 - y^2) + A\left(-6\tilde{y}^3 + 6\tilde{y}^5\right)\sin(2\pi x)\,, \tag{3.13}$$

$$v(x, y, t_0) = -2\pi A\left(\frac{1}{2} - \frac{3}{2}\tilde{y}^4 + \tilde{y}^6\right)\cos(2\pi x)\,, \tag{3.14}$$

with the normalized vertical coordinate $\tilde{y} \equiv (y - y_b - h))/h$ and the amplitude of the perturbation set to $A = 0.2$. Since the pressure corresponding to the above field is not known in closed form,

we initialize the pressure with a numerically obtained solution of its Poisson equation. The initial vorticity field is shown in figure 5.4.

The equations are advanced from $t_0 = 0$ until $t_{fin} = 2.1$ using a time step of $\Delta t = 0.0005$. Figure 5.5 shows the convergence history for three different spatial resolutions. After a common initial viscous decay of the perturbations, an asymptotic value of the error is reached, depending upon the resolution.

3.1.3.2 Taylor-Green vortices

Here we repeat the test proposed in reference [2], consisting of the simulation of decaying vortices inside a sub-domain embedded within the computational domain. The initial field is identical to the exact solution given in § 2.5 which also provides the time-dependent boundary conditions of the computational domain and the time-dependent velocity values $u_\beta^\Gamma(t)$ to be prescribed at the interface of the embedded sub-domain.

The computational domain has the dimension $\Omega = [-1.5, 1.5] \times [-1.5, 1.5]$. There are two cases: (i) the sub-domain boundary is located at $x, y = \pm 1$; and (ii) the sub-domain is rotated by 45° w.r.t. the grid lines, i.e. $|x| + |y| = 1$.

The viscosity is set to $\nu = 0.2$ which gives a value for the Reynolds number based on vortex size and maximum velocity of 10. The equations are advanced from $t_0 = 0$ until $t_{fin} = 0.3$ using a time step of $\Delta t = 0.001$.

Figure 5.6 shows the convergence of the numerical solution for the velocity as a function of the number of grid nodes used in each coordinate direction. Two features are noteworthy. First, the displayed non-uniform convergence is to be explained by the geometrical relation of the embedded boundary w.r.t. the computational grid, which is responsible for the amount of interpolation error introduced into the forcing values. When keeping this relation constant—i.e. for a constant relative interpolation error—the observed convergence rate is second order in space in both cases as in reference [2].

Secondly, it has to be noted that for arrangements where the embedded boundary coincided with one of the grid lines (or where one of its corners coincide with a grid node in the rotated case) no convergence could be obtained. This result is independent of the choice of the time step; it pertains to both, the implicit and explicit methods; it is equally obtained with both variants of the discrete Poisson solvers (FISHPACK and NAG multi-grid). Closer inspection shows that in those situations and at those locations very localized values for the divergence $\nabla \cdot \mathbf{u}^*$ are generated since there is no coupling of the volume forcing to the neighboring nodes through interpolation. Since this is essentially the RHS of the pseudo-pressure, the solution of the Poisson equation is dominated by these near-singularities and the scheme becomes unstable after some time of integration. Choi (personal communication) did not observe this phenomenon during the simulations reported in [2]. It should therefore be regarded as an unsolved issue.

3.1.3.3 Flow due to an oscillating plate

We consider the flow above a horizontal flat plate of infinite extension which is oscillating within its plane. Batchelor [11, p.191] gives the following analytical solution for this viscous-diffusion-dominated case:

$$u(y,t) = U \exp\left(-y\sqrt{\frac{f_{osc}}{2\nu}}\right) \cos\left(t\, f_{osc} - y\sqrt{\frac{f_{osc}}{2\nu}}\right) , \tag{3.15}$$

where U is the amplitude of the oscillating velocity, y the distance from the plate and f_{osc} the frequency of the oscillations.

Our simulation uses periodic conditions parallel to the plane (and only 5 grid points in this parallel direction) and Dirichlet conditions for velocity and pressure in the direction perpendicular to the plate, with 500 grid nodes uniformly distributed over a non-dimensional domain length of $L/\sqrt{\nu/f_{osc}} = 25$. The period of the oscillating motion is $T = 2\pi/f$. We have used a small time step with $CFL = 0.01$.

Plots of the plate-parallel velocity u at times $t/T = 0.1989$ and $t/T = 0.3979$ are shown in figure 5.7. The correspondence with the analytic solution (3.15) is very good throughout the "effective domain", i.e. for $y > 0$.

3.1.3.4 Flow around a stationary cylinder

An important first step in the direction of simulating particulate flows is the simulation of flow around a stationary cylinder which demonstrates the capability of the method to capture the hydrodynamic forces acting upon the solid body with reasonable precision. To this end we use a flow domain $\Omega = [-1.85, 6.15] \times [-4, 4]$ and a cylinder with diameter $D = 0.3$ located at the origin. The chosen domain size of approximatelty $27D \times 27D$ corresponds to the one of reference [22] and has to be considered as moderate. Therefore, some influence due to the boundaries, particularly the confinement of the flow at the lateral boundaries, is to be expected. However, much larger sizes would have been costly due to the present use of uniform grids.

We have performed computations with three values for viscosity corresponding to Reynolds numbers based upon cylinder diameter and free-stream velocity,

$$Re_D \equiv \frac{Du_\infty}{\nu} \tag{3.16}$$

of 1, 10 and 100. The grid consisted of $n_\alpha = 512$ uniformly distributed nodes, i.e. the coarser grid of reference [22]. The initial field was set to uniform, unidirectional flow, i.e. $u(t = 0) = u_\infty$, $v(t = 0) = 0$, $p(t = 0) = 0$. Since the non-slip velocity at the embedded solid body is not respected, strong initial perturbations develop which subsequently decay under the influence of viscosity. The boundary conditions at the outer domain were:

$$\left.\begin{array}{rcl} u(t) & = & u_\infty \\ v(t) & = & 0 \end{array}\right\} \quad \text{at} \quad \left\{\begin{array}{lcl} x = -1.85 & , & \forall\, y \\ y = \pm 4 & , & \forall\, x \end{array}\right. \quad , \tag{3.17}$$

and convective outflow at $(x = 4.2, y)$; homogeneous Neumann conditions for pseudo-pressure were applied throughout.

For $Re_D = 1, 10$ the flow reaches a steady state with a symmetric wake, while for $Re_D = 100$ alternating vortex shedding is obtained. Without the addition of external perturbations (as in the present case), the transition to the periodic state starts to be visible at around $55t_{ref}$ and the necessary integration time for reaching an asymptotic periodic solution is around $100t_{ref}$ where the reference time is defined as

$$t_{ref} \equiv \frac{D/2}{u_\infty} \quad . \tag{3.18}$$

The global quantities of interest are the drag and lift coefficients,

$$C_D \equiv \frac{F_D}{\frac{1}{2}\rho u_\infty^2 D}\,, \quad C_L \equiv \frac{F_L}{\frac{1}{2}\rho u_\infty^2 D}\,, \tag{3.19}$$

with F_D and F_L being the drag and lift forces, respectively; these forces are the negative of the integral of the volume forces. The Strouhal number is defined as

$$St \equiv \frac{f_n D}{u_\infty} \quad , \tag{3.20}$$

with the natural vortex shedding frequency f_n.

As explained in detail in § A.2, there are in general different methods for measuring the forces acting upon the embedded body. For steady flow, we can simply sum the (negative of the) volume forces directly; otherwise we can evaluate a momentum balance over a control volume enclosing the cylinder. It was verified that both methods give equivalent results (within 1% of each other) in steady flow. Hence, we will not further specify which of the methods was used in the subsequent results.

	present results			reference values		
Re_D	C_D	C'_L	St	C_D	C'_L	St
1	14.0643	–	–	12.56†	–	–
10	3.14318	–	–	3‡	–	–
100	1.39423	0.35573	0.174	1.33*	0.33*	0.165*

Table 3.1: Dimensionless coefficients obtained from the simulation of the flow around a stationary cylinder using the direct forcing method. The reference values correspond to: † Oseen's formula $C_D = 8\pi/(Re_D \log(7.4/Re_D))$ given e.g. in [11, p.246]; ‡ experimental data compiled by Schlichting [23, p.16]; * numerical computation of Park *et al.* (cf. [2]).

Table 3.1 shows the present computational results. It can be seen that the dimensionless coefficients are reasonably well captured, for the steady, low-Reynolds case as well as for the periodic case. As mentioned before, the correspondence with experimental results could be further improved by increasing the size of the comutational domain. Furthermore, the grid is not fully converged. Finally, figure 5.8 shows instantaneous fields of vorticity and pressure for the case at $Re_D = 100$. The development of the vortex street and especially the perturbation-free nature of the numerical outflow can be seen.

3.1.3.5 Flow around a translationally oscillating cylinder

The next step is to let the solid object perform a prescribed (forced) movement within the fluid environment. We add to the previous case a periodic vertical movement of the cylinder axis, viz.

$$y_c(t) = A_y \cos(2\pi f_f t)\,, \tag{3.21}$$
$$v_c(t) = \dot{y}_c = 2\pi f A_y \sin(2\pi f_f t)\,, \tag{3.22}$$

where the amplitude of the motion was chosen as a fraction of the cylinder's Diameter, $A_y = D/4$, and the frequency was equal to the natural shedding frequency at $Re_D = 100$, $f_f = f_n(Re_D = 100) \approx 0.166 u_\infty/D$. The horizontal position is fixed, $x_c(t) = 0$, and no angular movement is allowed. This case corresponds to the situation of a translationally oscillating cylinder in cross-flow which was investigated e.g. in reference [24], however at substantially higher Reynolds number. Presently, we keep all physical and numerical parameters of the stationary situation (§ 3.1.3.4).

Failure of the direct forcing method. Figure 5.9 shows the temporal evolution of the drag coefficient for the case $Re_D = 10$. A very noisy curve with noise levels of the order of the mean value is obtained. The comparison with the velocity of the cylinder's vertical motion shows that the noise is correlated: around the times when the cylinder is at rest, the noise is smallest and vice versa. The effect is due to the fact that node points which are part of the solid-fluid interface—and therefore receive a voume force contribution—suddenly drop out of the circumference of the circle during their motion relative to the Eulerian grid. Consequently, the pressure field is strongly perturbed and the sum of the volume forces applied to the embedded body oscillates. Further tests have shown that this phenomenon cannot be alleviated by lowering the time step. Interestingly, the amplitude of the oscillations of the volume force is even larger when the motion of the embedded body is slower (cf. figure 5.9). We conclude from this case that the no-slip variant of the direct forcing method is *not* adapted for the computation of interfaces traversing the grid in a general manner and therefore discard this method in view of our present objective of particle simulation. It becomes clear that some smoothing of the interface is necessary in order to prevent similar oscillatory behaviour; the immersed boundary method and the rigidity method both include some type of regularization.

3.2 The immersed boundary method: spring forces

The immersed boundary method has initially been developed by Peskin and co-workers for the simulation of the flow in the human heart and then extended to a variety of related problems [7]. The specific aim of the method was to allow for the numerical representation of singular forces at Lagrangian locations within a Eulerian solver. While in the original problem of immersed fibers or membranes (heart valves) the expression for the singular forces is known analytically (i.e. as a function of the local curvature of the fibers), the present case of immersed rigid bodies with finite volume requires the somewhat *ad hoc* introduction of adequate spring and damper forces [22]. In [8] the implications of the additional force field are discussed; a related method was implemented in reference [6].

3.2.1 Continuous formulation

First, we introduce the curvilinear coordinate s (with $0 \leq s \leq L_b$) which runs along the fluid-solid interface of the immersed body under consideration; $\mathbf{X}(s,t)$ defines the position of the interface in time. The body force of the Navier-Stokes equations (2.1) is then expressed as follows:

$$\mathbf{F}(s,t) = \kappa \left(\mathbf{X}^{(d)}(s,t) - \mathbf{X}(s,t) \right) + 2\gamma \, \partial_t \left(\mathbf{X}^{(d)}(s,t) - \mathbf{X}(s,t) \right) , \tag{3.23a}$$

$$\mathbf{f}(\mathbf{x},t) = \int_0^{L_b} \mathbf{F}(s,t) \, \delta \left(\mathbf{x} - \mathbf{X}(s,t) \right) \, \mathrm{d}s \, . \tag{3.23b}$$

The singular force $\mathbf{F}(s,t)$ which acts at the Lagrangian locations $\mathbf{X}(s,t)$ is given by the force obtained if springs with stiffness κ and dampers with damping constants γ would be attached between the present location $\mathbf{X}(s,t)$ of the interface and its *desired* location $\mathbf{X}^{(d)}(s,t)$. By "desired location" we mean the material locations given by the known location of the body's center and its angle of rotation together with the rigidity constraint; the center can be stationary, undergo a forced or a free movement. This formulation of the forces serves to maintain the solid close to its desired location against the action of the hydrodynamic forces. In practice, we have neglected the damping part of the forces altogether ($\gamma = 0$); the authors in [6] did retain it.

As indicated in equation (3.23b), the Lagrangian force is transferred to the Eulerian representation, $\mathbf{f}(\mathbf{x},t)$, by means of an integral which involves the Dirac function of the difference between the Eulerian and the Lagrangian position vectors. This operation is called "spreading". The adequate discretization of this step is one of the key points of the immersed boundary method (cf. § 3.2.2).

Finally, the Lagrangian locations need to be computed from the fluid velocity at the respective locations, $\mathbf{U}(s,t)$, which in turn is obtained by transferring the Eulerian velocities to the Langrangian points ("interpolation"), viz.

$$\mathbf{U}(s,t) = \int_\Omega \mathbf{u}(\mathbf{x},t) \, \delta \left(\mathbf{x} - \mathbf{X}(s,t) \right) \, \mathrm{d}\mathbf{x} \, , \tag{3.24a}$$

$$\partial_t \mathbf{X}(s,t) = \mathbf{U}(s,t) \, . \tag{3.24b}$$

3.2.2 Discrete formulation

In addition to the Eulerian grid introduced in § 2.2 we define a grid of equidistant intervals along the curvilinear coordinate s, i.e. the set

$$s_k = (k-1)\Delta s \qquad k = 1, \ldots, n_L \quad , \tag{3.25}$$

with $\Delta s \equiv L_b / n_L$. The spatial locations of the grid points $\mathbf{X}(s_k)$ are also called "Lagrangian marker points".

From here on, we suppose that the Eulerian grid is such that $\Delta x = \Delta y$ which is necessary for some of the identities between Eulerian and Lagrangian expressions to hold.

The Dirac function appearing in formulae (3.23) and (3.24) needs to be regularized in order for a discrete version to make sense. Peskin [7] constructs the following regularized delta function:

$$\delta_h(\mathbf{x}) = d_h\left(\frac{x}{\Delta x}\right)\cdot d_h\left(\frac{y}{\Delta y}\right)\frac{1}{\Delta x\Delta y} \tag{3.26a}$$

$$d_h(r) = \begin{cases} \frac{1}{8}\left(3-2|r|+\sqrt{1+4|r|-4r^2}\right) & |r|\le 1 \\ \frac{1}{8}\left(5-2|r|-\sqrt{-7+12|r|-4r^2}\right) & 1\le|r|\le 2 \\ 0 & \text{else} \end{cases} \tag{3.26b}$$

Important properties of this choice are:

(i) Compact support (with a radius of $2\Delta x$) which leads to computational efficiency in performing the "spreading" and "interpolation" operations between Eulerian and Lagrangian grids.

(ii) Continuity of the function $d_h(r)$ for real arguments r which allows for smooth transition of moving Lagrangian objects over the fixed grid.

(iii) Second-order accurate interpolation for *smooth* fields; at solid boundaries where derivatives of velocity and pressure have a jump, the method leads to first-order spatial accuracy.

In the following we will consider an explicit coupling between solid and fluid phases, meaning that the formulation of the volume forces is explicit (the viscous terms are still treated implicitly). Lai and Peskin [22] use a second-order (for smooth fields) time- and space-accurate predictor-corrector scheme. Here, we incorporate the immersed boundary method into the present second-order accurate three-step Runge-Kutta method. The resulting algorithm reads (for each Runge-Kutta substep):

$$\mathbf{F}^k(s_l) = \kappa\left(\mathbf{X}^{(d)}(s_l,t)-\mathbf{X}^{k-1}(s_l)\right)+2\gamma\left(\mathbf{U}^{(d)}(s_l,t)-\mathbf{U}^{k-1}(s_l)\right), \tag{3.27a}$$

$$\mathbf{f}^k(\mathbf{x}) = \sum_l \mathbf{F}^k(s_l)\,\delta_h\left(\mathbf{x}-\mathbf{X}^{k-1}(s_l)\right)\Delta s, \tag{3.27b}$$

$$\frac{\mathbf{u}^*-\mathbf{u}^{k-1}}{\Delta t} = \alpha_k\nu\nabla^2(\mathbf{u}^{k-1}+\mathbf{u}^*)-2\alpha_k\nabla p^{k-1}-\gamma_k\left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-1}-\zeta_k\left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-2} + \gamma_k\mathbf{f}^k+\zeta_k\mathbf{f}^{k-1}, \tag{3.27c}$$

$$\nabla^2\phi^k = \frac{\nabla\cdot\mathbf{u}^*}{2\alpha_k\Delta t}, \tag{3.27d}$$

$$\mathbf{u}^k = \mathbf{u}^*-2\alpha_k\Delta t\nabla\phi^k, \tag{3.27e}$$

$$p^k = p^{k-1}+\phi^k-\alpha_k\Delta t\,\nu\nabla^2\phi^k, \tag{3.27f}$$

$$\mathbf{U}^k(s_l) = \sum_{i,j}\mathbf{u}^k(\mathbf{x}_{i,j})\,\delta_h\left(\mathbf{x}_{i,j}-\mathbf{X}^{k-1}(s_l)\right)\Delta x\Delta y, \tag{3.27g}$$

$$\mathbf{X}^k(s_l) = \mathbf{X}^{k-1}(s_l)+\alpha_k\Delta t(\mathbf{U}^k(s_l)+\mathbf{U}^{k-1}(s_l)). \tag{3.27h}$$

Please note that the volume forces in (3.27c) are integrated with the same scheme as the nonlinear terms in the predictor step in order to guarantee second-order overall consistency (for smooth fields); for the same reason, the position of the marker points in (3.27h) is integrated with the same Crank-Nicholson-type scheme as the viscous terms. This formulation has implications for the determination of the total volume force added to the fluid during a full time step (cf. §A.2.3). Furthermore, it should be noted that the time-level of the desired locations in (3.27a) was not exactly specified for the time being. In this section we will use the prescribed locations at the end of the time step, i.e. $\mathbf{X}^{(d)}(s_l,t^{n+1})$, throughout the Runge-Kutta sub-steps; in the case of freely-moving particles, we will specify the two-way coupling in detail (cf. § 4).

	present results			reference values		
Re_D	C_D	C'_L	St	C_D	C'_L	St
1	14.4025	–	–	$12.56^{\dagger}$	–	–
10	3.23853	–	–	$3^{\ddagger}$	–	–
100	1.38292	0.316329	0.169	1.33^{*}	0.33^{*}	0.165^{*}

Table 3.2: Dimensionless coefficients obtained from the simulation of the flow around a stationary cylinder using the immersed boundary method. For more details cf. table 3.1.

The discretization of the spatial operators ∇, $\nabla \cdot ()$, ∇^2 is as before (§ 2.2). The bounds for the discrete summation in (3.27b) and (3.27g) are in practice considerably reduced due to the compactness of the regularized delta function; an adequate mask is instead applied.

The immersed boundary method has the disadvantage that an additional free parameter—the stiffness κ— is introduced. Its value needs to be selected manually such that (i) the solid body is not appreciably deformed during the computation, and (ii) the allowed time step restriction is not too severe. In reference [22], a rough estimate for the time step is given as $\Delta t \sim \sqrt{\Delta x/\kappa}$, which shows that considerations (i) and (ii) are opposed w.r.t. the value of κ. In practice, however, we were able to find a reasonable compromise.

Furthermore, the Lagrangian grid s_k needs to be selected. As a rule of thumb, Δs should be of the order of Δx for the grid transition to be smooth.

3.2.3 Validation

3.2.3.1 Flow around a stationary cylinder

The computations of the flow around a stationary cylinder in cross-flow were repeated with the same physical and numerical parameter values as described in § 3.1.3.4. Additionally, the Lagrangian grid and the stiffness parameter need to be specified in the present context. Our choice was guided by reference [22], where $\kappa = 48000$ was used. Furthermore, we chose $n_L = 50$. With these values, the maximum deviation of any marker point from its desired value was below $0.0002D$.

Table 3.2 shows the results which are of similar quality as the corresponding results obtained from the direct forcing method (cf. table 3.1). Also, comparison with the numerical results of reference [22] at the same grid resolution (for the case $Re_D = 100$ and considering their "formally second-order" predictor-corrector scheme) show that the present results are in closer agreement with experimental values.

3.2.3.2 Flow around a translationally oscillating cylinder

Again, we keep all parameters as in the previous section (except otherwise stated) and as in § 3.1.3.5. For all three values of the Reynolds number, a periodic flow is obtained. Figures 5.11-5.13 show the phase-space diagrams of drag and lift coefficient, respectively, vs. vertical position $y_c(t)$ of the cylinder. Smooth, non-physical perturbations are visible, particularly in the drag coefficient. In the latter case, their amplitude is of the order of 5% of the mean drag at $Re_D = 100$. Changing the value of the stiffness constant κ or introducing a non-zero damping term ($\gamma \neq 0$ in equation 3.23a) does not have any substantial influence. However, it was found that these spurious oscillations are correlated with the Eulerian grid. Figure 5.14 shows results from a computation at $Re_D = 10$ performed with double Eulerian resolution ($n_\alpha = 1024$); the Lagrangian grid was defined with $n_L = 75$, the stiffness set to $\kappa = 150000$ and the time step adjusted to $\Delta t = 0.0003$. The amplitude of the oscillations of the drag coefficient is visibly reduced and their frequency increased. As an alternative, the support of the regularized delta function (3.26b), being currently at its minimum $r = 2\Delta x$ for consistency with the integral identities, could be increased. Smoothing the delta function further would most probably reduce the observed oscillations, however at the

Re_D	E (immersed)	E (rigidity)
1	-3.946766	-3.620751
10	-0.583957	-0.542197
100	$+0.229322$	$+0.246812$

Table 3.3: The amount of energy transferred between fluid and cylinder $E \equiv \int_0^T \dot{y}_c C_L \mathrm{d}t/D$ over one period of the translational oscillation. Work is done on the cylinder when the quantity E is positive.

cost of reducing the accuracy of the scheme in the vicinity of solid bodies. We have not attempted this modification of the scheme.

Figure 5.15 shows an instantaneous vorticity and pressure field of the $Re_D = 100$ case. Comparison with the case of the stationary cylinder (figure 5.10) does not reveal a visible difference in vorticity, whereas the pressure field is strongly marked by the vertical motion of the body. A close-up of the velocity vectors around the solid-fluid interface is given in figure 5.16. The verification of the no-slip condition in an approximate sense can be checked visually. A more rigorous measure was not attempted.

Table 3.3 shows the amount of energy which is transferred between fluid and cylinder over one cycle of the periodic oscillation. It can be seen that the cylinder does work upon the fluid in both cases at $Re_D = 1, 10$, whereas work is done on the cylinder at $Re_D = 100$, for the chosen frequency.

We conclude from this test case that the immersed boundary method is a potential candidate for the simulation of particulate flows. However, its computational efficiency still needs to be established.

3.3 A variant of the direct forcing method: rigidity

The direct forcing method based upon the rigidity constraint [3, 10, 25] is specifically tailored for particle simulations, mainly with speed of execution in mind. The basic idea is to directly enforce rigid-body movement i.e.

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_c + \boldsymbol{\omega}_c \times \mathbf{r}(\mathbf{x}) \qquad \forall \quad \mathbf{x} \in \mathcal{S} \tag{3.28}$$

upon the fluid occupying the space of the solid, whose sub-volume is denoted as $\mathcal{S}$. Here, $\mathbf{u}_c$ is the velocity of the mass center of the solid (located at $\mathbf{x}_c$) and $\boldsymbol{\omega}_c$ its angular velocity; $\mathbf{r}(\mathbf{x})$ is a point's distance from the center, i.e. $\mathbf{r}(\mathbf{x}) = \mathbf{x} - \mathbf{x}_c$.

In this respect, the present method is similar to the direct forcing method based upon the no-slip condition, § 3.1, but applying volume forces throughout the solid part of the domain instead of just around the interface. Probably the most important difference between these two schemes is the fact that in the rigidity method a blending operation is applied to the additional force term which should in principle allow for smooth traversing of particles across the fixed grid.

The rigidity condition inside the solid is connected smoothly to the velocity field of the fluid away from the body by means of the solid volume fraction $0 \leq \xi_\beta \leq 1$, computed for each Eulerian grid cell, viz.

$$u_\beta = (1 - \xi_\beta)\, u_\beta^{fluid} + \xi_\beta\, u_\beta^{solid} \qquad \text{(no summation on } \beta) \quad . \tag{3.29}$$

Note that in the present case of a staggered grid arrangement, the volume fraction is a vector whose components are computed by considering a cell surrounding the location of the respective velocity component. The numerical evaluation of $\boldsymbol{\xi}(\mathbf{x}, t)$ is done by splitting the overlap between the circular particles and the cells into a polygon and a circular arc segment. The latter contribution is presently not neglected, contrary to the three-dimensional simulations in references [3, 10, 25].

3.3.1 Exactly rigid, approximately divergence-free

The original idea, incorporated into our present semi-implicit Runge-Kutta scheme, leads to the following algorithm:

$$\frac{\mathbf{u}^* - \mathbf{u}^{k-1}}{\Delta t} = \alpha_k \nu \nabla^2 (\mathbf{u}^{k-1} + \mathbf{u}^*) - 2\alpha_k \nabla p^{k-1} - \gamma_k \left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-1} - \zeta_k \left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-2} \tag{3.30a}$$

$$\nabla^2 \phi^k = \frac{\nabla\cdot\mathbf{u}^*}{2\alpha_k \Delta t}\,, \tag{3.30b}$$

$$\hat{\mathbf{u}} = \mathbf{u}^* - 2\alpha_k \Delta t \nabla \phi^k\,, \tag{3.30c}$$

$$p^k = p^{k-1} + \phi^k - \alpha_k \Delta t\, \nu \nabla^2 \phi^k\,, \tag{3.30d}$$

$$u^k_\beta = (1-\xi_\beta(t))\ \hat{u}_\beta + \xi_\beta(t)\, \tilde{u}^{k-1}_\beta\,, \qquad \text{(no summation)} \tag{3.30e}$$

$$\tilde{\mathbf{u}}^k = \mathbf{u}_c(t) + \boldsymbol{\omega}_c(t)\times\mathbf{r}(t)\,, \tag{3.30f}$$

We remark that in practice, not the full field of solid velocities $\tilde{\mathbf{u}}$ is evaluated, but expression (3.30f) is only computed at grid nodes where the volume fraction ξ_β of the respective component is non-zero.

The total volume force added over a full time step is computed as shown in § A.2.4.

As in the immersed boundary method, we have left unspecified the time-level at which the location and velocities of the solid body (and consequently the vector field of volume fractions) are evaluated. Again we use $\mathbf{x}_c(t^{n+1})$, etc. for the simulation of the flow around a body with forced movement in the present section.

From equations (3.30) it is obvious that the final velocity field $\mathbf{u}^k$ will not be divergence-free in the presence of solid bodies. Supposing for the time being that the volume fraction is a scalar (as it were if we used a collocated grid arrangement), we have for the divergence:

$$\nabla\cdot\mathbf{u}^k = \nabla\cdot\hat{\mathbf{u}} - \nabla\xi\cdot\hat{\mathbf{u}} - \xi\nabla\cdot\hat{\mathbf{u}} + \nabla\xi\cdot\tilde{\mathbf{u}}^{k-1} + \xi\nabla\cdot\tilde{\mathbf{u}}^{k-1} = \nabla\xi\cdot\left(\tilde{\mathbf{u}}^{k-1} - \hat{\mathbf{u}}\right) \quad , \tag{3.31}$$

which does not vanish in general. For the present staggered arrangement (i.e. vectorial volume fraction), there is no compact dyadic notation, but the conclusion is similar. One could imagine performing step (3.30f) before the projection steps (3.30b) and (3.30c) which would lead to a divergence-free field $\mathbf{u}^k$; this variant is presented in § 3.3.2. However, the rigidity condition would then not be satisfied exactly. A costly alternative (which has not been implemented in the present context) is to iterate until both conditions are satisfied to a reasonable degree (Kajishima, personal communication).

For the original scheme, it should be kept in mind that the results presented here are from flow fields with non-zero divergence, those values being very localized around the interface fluid-solid, i.e. where gradients of the volume fraction exist.

3.3.2 Approximately rigid, exactly divergence-free

Simply inverting the order by which the constraints of rigidity and zero divergence are imposed leads to the following modified algorithm:

$$\frac{\hat{\mathbf{u}} - \mathbf{u}^{k-1}}{\Delta t} = \alpha_k \nu \nabla^2 (\mathbf{u}^{k-1} + \hat{\mathbf{u}}) - 2\alpha_k \nabla p^{k-1} - \gamma_k \left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-1} - \zeta_k \left[(\mathbf{u}\cdot\nabla)\mathbf{u}\right]^{k-2} \tag{3.32a}$$

$$u^*_\beta = (1-\xi_\beta(t))\ \hat{u}_\beta + \xi_\beta(t)\, \tilde{u}^{k-1}_\beta\,, \qquad \text{(no summation)} \tag{3.32b}$$

$$\nabla^2 \phi^k = \frac{\nabla\cdot\mathbf{u}^*}{2\alpha_k \Delta t}\,, \tag{3.32c}$$

$$\mathbf{u}^k = \mathbf{u}^* - 2\alpha_k \Delta t \nabla \phi^k\,, \tag{3.32d}$$

$$p^k = p^{k-1} + \phi^k - \alpha_k \Delta t\, \nu \nabla^2 \phi^k\,, \tag{3.32e}$$

$$\tilde{\mathbf{u}}^k = \mathbf{u}_c(t) + \boldsymbol{\omega}_c(t)\times\mathbf{r}(t)\,, \tag{3.32f}$$

	present results			reference values		
Re_D	C_D	C'_L	St	C_D	C'_L	St
1	13.6683	–	–	$12.56^{\dagger}$	–	–
10	3.14770	–	–	$3^{\ddagger}$	–	–
100	1.32311	0.308481	0.169	1.33^{*}	0.33^{*}	0.165^{*}
$(\nabla \cdot \mathbf{u} = 0)$ 100	1.50019	0.342758	0.171			

Table 3.4: Dimensionless coefficients obtained from the simulation of the flow around a stationary cylinder using the rigidity method. The last line refers to the variant of § 3.3.2 which preserves zero divergence exactly. For more details cf. table 3.1.

3.3.3 Validation

3.3.3.1 Flow around a stationary cylinder

The computations of the flow around a stationary cylinder in cross-flow were repeated with the same physical and numerical parameter values as described in § 3.1.3.4 and results are shown in table 3.4; they are of similar quality as the corresponding results obtained from the direct forcing method and the immersed boundary method.

3.3.3.2 Flow around a translationally oscillating cylinder

Again, we keep all parameters as in § 3.1.3.5 and § 3.2.3.2.

Exactly rigid – approximately divergence-free. Let us first consider results obtained with the exactly rigid scheme of § 3.3.1. Figures 5.17-5.19 show the phase-space diagrams of drag and lift coefficients for the three different Reynolds numbers. Again, non-physical perturbations are visible, particularly in the drag coefficient. Their amplitude is substantially higher as in the case of the immersed boundary method and can be quantified as around 25% of the meand drag at $Re_D = 100$. Doubling the number of mesh points again reduces the amplitude of the oscillations visibly, as can be seen in figure 5.20 for $Re_D = 10$.

Figure 5.21 shows an instantaneous vorticity and pressure field of the $Re_D = 100$ case and a close-up of the velocity vectors around the solid-fluid interface is given in figure 5.22. No significant differences w.r.t. the results from the immersed boundary method are visible.

Exactly divergence-free – approximately rigid. Finally we turn to the exactly divergence-free variant of the rigidity method. For completeness, figures 5.23, 5.24 and 5.25 show, respectively, the phase-diagrams, an instantaneous field and a close-up of the velocity vectors around the cylinder. A very small non-rigid velocity can be detected inside the particle domain, introduced by the corrector step.

Chapter 4

Particle motion

4.1 Continuous formulation

Newton's second law of motion for a rigid, impermeable body submerged in a fluid states the following (e.g. [26]):

$$m_c \, \dot{\mathbf{u}}_c = \rho_f \oint_{\partial \mathcal{S}} \boldsymbol{\tau} \cdot \mathbf{n} \, \mathrm{d}\sigma + \mathbf{G}_c \,, \tag{4.1a}$$

$$I_c \, \dot{\boldsymbol{\omega}}_c = \rho_f \oint_{\partial \mathcal{S}} \left(\mathbf{r} \times (\boldsymbol{\tau} \cdot \mathbf{n}) \right) \, \mathrm{d}\sigma \,, \tag{4.1b}$$

where $\boldsymbol{\tau} \equiv -Ip + \nu \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right)$ is the hydrodynamic stress tensor (remember that presently p is the fluid pressure divided by fluid density), $\mathbf{n}$ the outward-pointing normal vector on the surface $\partial \mathcal{S}$, m_c the mass of a particle, I_c its moment of inertia and $\mathbf{G}_c$ the net gravitational force acting upon a particle.

The gravitational term can be expressed through the difference between the particle and the fluid densities, ρ_c and ρ_f, viz.

$$\mathbf{G}_c = V_c \left(\rho_c - \rho_f \right) \mathbf{g} \quad , \tag{4.2}$$

with $\mathbf{g}$ being the vector of gravitational acceleration and $V_c = \pi r_c^2$ is the volume (per unit depth) occupied by the two-dimensional particle.

For the present two-dimensional case, the mass and moment of inertia of circular particles, given per unit depth, are:

$$m_c = \rho_c \pi r_c^2 \,, \tag{4.3a}$$

$$I_c = \rho_c \frac{\pi}{2} r_c^4 \,. \tag{4.3b}$$

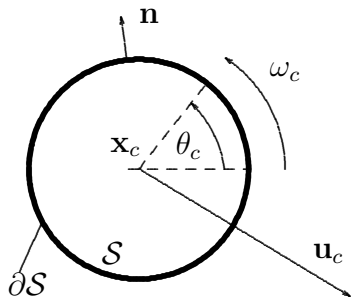

Figure 4.1: Some geometrical definitions pertaining to a cylindrical particle in motion.

Concerning the surface integrals of the stress tensor in (4.1), we refer the reader to appendix A.3 where it is shown how these terms can be evaluated from a momentum balance.

For later reference, let us resume the equations for the particle motion in the formulation which we have found most convenient for our purposes:

$$V_c\,(\rho_c-\rho_f)\,\dot{\mathbf{u}}_c = -\rho_f\int_{\mathcal{S}}\mathbf{f}\,\mathrm{d}V+\mathbf{G}_c\,, \tag{4.4a}$$

$$I_c\,\dot{\boldsymbol{\omega}}_c = -\rho_f\int_{\mathcal{S}}(\mathbf{r}\times\mathbf{f})\,\mathrm{d}V+\rho_f\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}}(\mathbf{r}\times\mathbf{u})\,\mathrm{d}V\,. \tag{4.4b}$$

These equations are exact for incompressible flow and supposing that the interface fluid-solid verifies the rigid-body-motion constraint.

In the framework of the rigidity method, the following simplification of the angular momentum equation applies:

$$\mathbf{u}(\mathbf{x})=\mathbf{u}_c+\boldsymbol{\omega}(r,t)\times\mathbf{r}(\mathbf{x})\quad \mathbf{x}\in\mathcal{S}: \qquad \frac{\pi}{2}r_c^4\,(\rho_c-\rho_f)\,\dot{\boldsymbol{\omega}}_c=-\rho_f\int_{\mathcal{S}}(\mathbf{r}\times\mathbf{f})\,\mathrm{d}V\,. \tag{4.5}$$

4.2 Discretization of the particle equations

With a three-step Runge-Kutta integration the linear momentum equation is solved as follows:

$$\frac{\mathbf{u}_c^k-\mathbf{u}_c^{k-1}}{\Delta t} = \frac{\rho_f}{V_c(\rho_c-\rho_f)}\left(-\gamma_k\int_{\mathcal{S}}\mathbf{f}^k\,\mathrm{d}V-\zeta_k\int_{\mathcal{S}}\mathbf{f}^{k-1}\,\mathrm{d}V\right)+2\alpha_k\mathbf{g}\,, \tag{4.6a}$$

$$\frac{\mathbf{x}_c^k-\mathbf{x}_c^{k-1}}{\Delta t} = \alpha_k\left(\mathbf{u}_c^k+\mathbf{u}_c^{k-1}\right)\,. \tag{4.6b}$$

Since the most recent forces $\mathbf{f}^k$ are a function of the particle velocities/positions at the previous sub-step ($\mathbf{u}_c^{k-1}$), the first part of the integration is done with the coefficients of an explicit right-hand-side (cf. § A.1). This also implies that the time level of all particle-related quantities in the fluid equations needs to be set to the most recent sub-step, i.e. $\mathbf{X}^{(d)}(s_l,t)\rightarrow\mathbf{X}^{(d)}(s_l)^{k-1}$ in equation (3.27) and $\xi_\beta(t)\rightarrow\xi_\beta^{k-1}$, $\mathbf{u}_c(t)\rightarrow\mathbf{u}_c^{k-1}$, $\boldsymbol{\omega}_c(t)\rightarrow\boldsymbol{\omega}_c^{k-1}$, $\mathbf{r}(t)\rightarrow(\mathbf{x}-\mathbf{x}_c^{k-1})$ in equations (3.30) and (3.32).

In the case of fully-rigid motion inside the particle domain (i.e. in variants of Kajishima's method), the discretization of the angular momentum equation is analogous, viz.

$$\mathbf{u}(\mathbf{x})=\mathbf{u}_c+\boldsymbol{\omega}(r,t)\times\mathbf{r}(\mathbf{x}) \qquad \mathbf{x}\in\mathcal{S}:$$

$$\frac{\boldsymbol{\omega}_c^k-\boldsymbol{\omega}_c^{k-1}}{\Delta t} = \frac{\rho_f}{\pi r_c^4(\rho_c-\rho_f)/2}\left(-\gamma_k\int_{\mathcal{S}}(\mathbf{r}\times\mathbf{f})^k\,\mathrm{d}V-\zeta_k\int_{\mathcal{S}}(\mathbf{r}\times\mathbf{f})^{k-1}\,\mathrm{d}V\right)\,, \tag{4.7a}$$

$$\frac{\boldsymbol{\theta}_c^k-\boldsymbol{\theta}_c^{k-1}}{\Delta t} = \alpha_k\left(\boldsymbol{\omega}_c^k+\boldsymbol{\omega}_c^{k-1}\right)\,, \tag{4.7b}$$

where $\boldsymbol{\theta}_c$ is the angular position of the particle. Note that the angular position of a circular particle is actually not needed for the rigidity method. In the case of the immersed boundary method, however, we need the angle in order to prescribe the desired Lagrangian marker locations as follows (valid in the two-dimensional case):

$$\mathbf{X}^{(d)}(s_l)=\mathbf{x}_c+r_c\begin{pmatrix}\cos\left(\frac{2\pi(l-1)}{n_L}+\theta_c\right)\\ \sin\left(\frac{2\pi(l-1)}{n_L}+\theta_c\right)\end{pmatrix}\quad, \tag{4.8}$$

and–in case we apply the damping term–the desired velocity at the Lagrangian marker locations:

$$\mathbf{U}^{(d)}(s_l)=\mathbf{u}_c+\boldsymbol{\omega}_c\times\left(\mathbf{X}^{(d)}(s_l)-\mathbf{x}_c\right)\quad. \tag{4.9}$$

In the case of the immersed boundary we also need to deal with the rate-of-change term in (4.4b). Since we know the velocity field only at the end of the three sub-intervals of the Runge-Kutta scheme, the best we can do is integrate the angular momentum of the particle as follows:

$$\begin{aligned}
\mathbf{u}(\mathbf{x}) = \mathbf{u}_c + \boldsymbol{\omega}(r,t) \times \mathbf{r}(\mathbf{x}) \qquad & \mathbf{x} \in \partial\mathcal{S}: \\
\frac{\boldsymbol{\omega}_c^k - \boldsymbol{\omega}_c^{k-1}}{\Delta t} &= \frac{\rho_f}{I_c} \left(-\gamma_k \int_{\mathcal{S}} (\mathbf{r} \times \mathbf{f})^k \, \mathrm{d}V - \zeta_k \int_{\mathcal{S}} (\mathbf{r} \times \mathbf{f})^{k-1} \, \mathrm{d}V \right. \\
&\quad \left. + \frac{\left(\int_{\mathcal{S}} (\mathbf{r} \times \mathbf{u}) \, \mathrm{d}V \right)^k - \left(\int_{\mathcal{S}} (\mathbf{r} \times \mathbf{u}) \, \mathrm{d}V \right)^{k-1}}{\Delta t} \right) . \qquad (4.10)
\end{aligned}$$

4.2.1 The direct forcing method based upon rigidity

We have implemented the particle motion into the direct forcing method based upon rigidity (cf. § 3.3.2). However, a lower bound for the allowed density ratio seems to be $\rho_p/\rho_f \approx 1.9$ for a stable computation in practice. Therefore, we did not further consider this method since the restriction is felt to be quite severe.

4.2.2 Full algorithm for the immersed boundary method

For reference purposes, the full equations solved at each Runge-Kutta sub-step k are repeated below.

$$\begin{aligned}
\mathbf{X}^{(d)}(s_l) &= \mathbf{x}_c^{k-1} + r_c \left(\cos\left(\frac{2\pi(l-1)}{n_L} + \theta_c^{k-1} \right), \sin\left(\frac{2\pi(l-1)}{n_L} + \theta_c^{k-1} \right) \right) && (4.11a) \\
\mathbf{U}^{(d)}(s_l) &= \mathbf{u}_c^{k-1} + \boldsymbol{\omega}_c \times \left(\mathbf{X}^{(d)}(s_l) - \mathbf{x}_c^{k-1} \right) && (4.11b) \\
\mathbf{F}^k(s_l) &= \kappa \left(\mathbf{X}^{(d)}(s_l) - \mathbf{X}^{k-1}(s_l) \right) + 2\gamma \left(\mathbf{U}^{(d)}(s_l) - \mathbf{U}^{k-1}(s_l) \right) , && (4.11c) \\
\mathbf{f}^k(\mathbf{x}) &= \sum_l \mathbf{F}^k(s_l) \, \delta_h \left(\mathbf{x} - \mathbf{X}^{k-1}(s_l) \right) \Delta s \, , && (4.11d) \\
\frac{\mathbf{u}^* - \mathbf{u}^{k-1}}{\Delta t} &= \alpha_k \nu \nabla^2 (\mathbf{u}^{k-1} + \mathbf{u}^*) - 2\alpha_k \nabla p^{k-1} \\
&\quad -\gamma_k \left[(\mathbf{u} \cdot \nabla)\mathbf{u} \right]^{k-1} - \zeta_k \left[(\mathbf{u} \cdot \nabla)\mathbf{u} \right]^{k-2} + \gamma_k \mathbf{f}^k + \zeta_k \mathbf{f}^{k-1} \, , && (4.11e) \\
\nabla^2 \phi^k &= \frac{\nabla \cdot \mathbf{u}^*}{2\alpha_k \Delta t} \, , && (4.11f) \\
\mathbf{u}^k &= \mathbf{u}^* - 2\alpha_k \Delta t \nabla \phi^k \, , && (4.11g) \\
p^k &= p^{k-1} + \phi^k - \alpha_k \Delta t \, \nu \nabla^2 \phi^k \, , && (4.11h) \\
\mathbf{U}^k(s_l) &= \sum_{i,j} \mathbf{u}^k(\mathbf{x}_{i,j}) \, \delta_h \left(\mathbf{x}_{i,j} - \mathbf{X}^{k-1}(s_l) \right) \Delta x \Delta y \, , && (4.11i) \\
\mathbf{X}^k(s_l) &= \mathbf{X}^{k-1}(s_l) + \alpha_k \Delta t (\mathbf{U}^k(s_l) + \mathbf{U}^{k-1}(s_l)) \, , && (4.11j) \\
\frac{\mathbf{u}_c^k - \mathbf{u}_c^{k-1}}{\Delta t} &= \frac{\rho_f}{V_c(\rho_c - \rho_f)} \left(-\gamma_k \int_{\mathcal{S}} \mathbf{f}^k \, \mathrm{d}V^{k-1} - \zeta_k \int_{\mathcal{S}} \mathbf{f}^{k-1} \, \mathrm{d}V^{k-2} \right) + 2\alpha_k \mathbf{g} \, , && (4.11k) \\
\frac{\mathbf{x}_c^k - \mathbf{x}_c^{k-1}}{\Delta t} &= \alpha_k \left(\mathbf{u}_c^k + \mathbf{u}_c^{k-1} \right) . && (4.11l) \\
\frac{\boldsymbol{\omega}_c^k - \boldsymbol{\omega}_c^{k-1}}{\Delta t} &= \frac{\rho_f}{I_c} \left(-\gamma_k \int_{\mathcal{S}} (\mathbf{r}^{k-1} \times \mathbf{f}^k) \, \mathrm{d}V^{k-1} - \zeta_k \int_{\mathcal{S}} (\mathbf{r}^{k-2} \times \mathbf{f}^{k-1}) \, \mathrm{d}V^{k-2} \right. \\
&\quad \left. + \frac{\left(\int_{\mathcal{S}} \left(\mathbf{r}^{k-1} \times \mathbf{u}^k \right) \, \mathrm{d}V^{k-1} \right) - \left(\int_{\mathcal{S}} \left(\mathbf{r}^{k-2} \times \mathbf{u}^{k-1} \right) \, \mathrm{d}V^{k-2} \right)}{\Delta t} \right) . && (4.11m) \\
\frac{\boldsymbol{\theta}_c^k - \boldsymbol{\theta}_c^{k-1}}{\Delta t} &= \alpha_k \left(\boldsymbol{\omega}_c^k + \boldsymbol{\omega}_c^{k-1} \right) . && (4.11n)
\end{aligned}$$

Ω	$\mathbf{x}_c(t=0)$	Δx	Δt	κ	γ	n_L
$[0,8]\times[-1,1]$	$(0.8,0)$	0.005	0.0002	$1.25\cdot 10^5$	3	125

Table 4.1: Numerical parameters used for the problem of the single free-falling particle: domain Ω, initial particle position $\mathbf{x}_c(t=0)$, grid size Δx, time step Δt, virtual spring stiffness κ, virtual damper constant γ and number of Lagrangian marker points n_L. Please note that the gravitation acts in the positive horizontal direction, i.e. $\mathbf{g}/|\mathbf{g}| = (0,1)$.

case	ν	$\lvert\mathbf{g}\rvert$	r_p	ρ_p	ρ_f	Re_D	Fr
(a)	0.001	9.81	0.1	1.01	1.00	22.98	0.082
(b)	0.005	9.81	0.1	1.01	1.00	11.42	0.041
(c)	0.01	9.81	0.1	1.01	1.00	6.60	0.024

Table 4.2: Physical parameters for the problem of the single free-falling particle. Re_D corresponds to the Reynolds number based upon terminal particle velocity and particle diameter $D = 2\,r_p$; analogously, the Froude number $Fr \equiv \frac{u}{\sqrt{|\mathbf{g}|D}}$ is based upon the terminal velocity and the particle diameter.

Please note that the temporal index attached to the differential volume $\mathrm{d}V$ in (4.11k) and (4.11m) refers to the time level at which the particle domain $\mathcal{S}$ is defined, i.e. the time level of the center coordinates $\mathbf{x}_c$ used in defining $\mathcal{S}$.

4.3 Validation

4.3.1 A single freely falling particle

First we consider the case of a single particle accelerating from rest due to gravity and finally reaching its terminal velocity when the gravitational and the hydro-dynamical forces are in equilibrium. Table 4.1 shows the numerical parameters which we have used for this case. Please note that the grid resolution corresponds to 40 nodes per particle diameter, which is quite well-resolved.

Table 4.2 shows the physical parameters we have chosen. The particles are only slightly heavier than the fluid with a density ratio of 1.01. Three values for viscosity have been chosen, leading to terminal Reynolds numbers from 6 to 23 (cf. table 4.2). Figure 5.26 shows the temporal evolution of the center velocity and the center position (in the direction of the gravitational vector) for all three viscosities. The hydro-dynamical forces (not shown) exhibit an equally smooth evolution.

4.3.2 The drafting-kissing-tumbling case

In this case, two particles are accelerating from rest due to gravity, one leading the other by 2 particle diameters. The trailing particle benefits from the wake effect and eventually catches up with the leading one, giving rise to an interaction between both. This problem has been considered by various authors in the past, amongst others [5, 9, 27].

The numerical parameters are unchanged from the previous case, i.e. those of table 4.1, except that presently the two initial particle positions are: $\mathbf{x}_c^{(1)} = (0.8, 0.001)$ and $\mathbf{x}_c^{(2)} = (1.2, 0)$. The slight out-of-center position of the trailing particle was deliberately set such that tumbling occurs, because it was noted in earlier computations that otherwise both particles maintained a perfectly aligned position for a long time after catching-up took place.

The physical parameters for the present computation correspond to those of case (a) of table 4.2. In order to facilitate a comparison with other authors, the present parameters shall be cast in non-dimensional form. Dimensional analysis tells us that this case bears three similarity parameters: the Reynolds number, the Froude number and the density ratio. We define the following reference quantities:

$$u_{ref} = \sqrt{|\mathbf{g}|\, D}\,, \quad l_{ref} = D\,, \quad t_{ref} = \sqrt{\frac{D}{|\mathbf{g}|}}\,. \tag{4.12}$$

Figure 5.27 shows the position of the two particles during 120 non-dimensional time units. This is the duration up to and including the initial particle interaction, i.e. drafting and kissing but no tumbling yet due to the limited container size. At larger times, the vicinity of the "bottom" of the container was interfering with the interaction. The results of reference [9] and data provided by T.W. Pan (private communication) for particle positions correspond closely and are omitted in the figure.

Also shown are the hydro-dynamical forces. Initial oscillations on the time scale of the virtual springs are visible. This is why the damping forces are necessary. At later stages, i.e. for $t/t_{ref} > 80$ the forces are oscillating with a moderate amplitude and on a short time scale. This effect is due to the two particles coming into contact, i.e. getting so close to each other that probably the flow field in between the particles cannot be adequately resolved any more. This brings us to the question of the necessity of inter-particle repulsion forces which will be discussed briefly in the conclusion, § 5.

Finally, figure 5.28 shows the velocity in the direction of the gravitational vector. We have plotted our present results against those of Zhang and Prosperetti [9] and T.W. Pan (private communication). The agreement between all three methods is excellent up to $t/t_{ref} \approx 80$. Afterwards, the results of reference [9] still match the present ones closely while in Pan's simulation the two particles "synchronize" their velocity nearly instantaneously when coming into contact. It should be noted that in Pan's computation, an explicit repulsion force is introduced while the other two methods are free of such treatment.

Chapter 5

Conclusions, open questions & outlook

5.1 Summary

In the present study we have analyzed various variants of the fictitious domain method for the simulation of particulate flow. In all cases, the presence of rigid particles is introduced into the fluid equations by means of additional volume forces in the momentum equations of Navier-Stokes. Therefore, no explicit "gridding" of the time-dependent particle positions is necessary and fast solution strategies can be employed.

In a first step we have presented the basic fluid solver upon which to build. The method is of the fractional-step type and semi-implicit with Crank-Nicholson for the viscous terms and a three-step Runge-Kutta scheme for the non-linear terms. The spatial discretization is achieved with second-order central finite-differences. The formal accuracy was validated for analytical test cases.

We have implemented and tested two direct forcing methods, one of them based upon the no-slip condition and very similar to the one of reference [2] and the other one based upon the rigidity condition as in reference [3]. Both seem not to be adequate for our purposes: the former leads to very strong oscillations of the hydro-dynamical force upon the particle due to a lack of Eulerian-to-Lagrangian smoothing; the latter does include some smoothing, but apparently not in a sufficient manner since it does not allow for computations of particles which are lighter than approximately twice the specific weight of the fluid. Furthermore, we have implemented a variant of the immersed boundary method similar to reference [6]. Here, our tests have been very encouraging, allowing for smooth transfer between the Lagrangian and Eulerian representation.

We have implemented a tracking procedure of particle paths via a Newton equation. For those "full" computations, we have only retained the most promising method, i.e. the immersed boundary method. Our computations of single freely-falling particles and of a two-particle interaction have been successful. In the latter case, comparisons with computations from other authors have been very positive.

5.2 Performance of the present method

Of course it is important to recapitulate the performance of the method if it shall be useful for large systems.

Operation count. Let us first consider the operation count of the immersed boundary method and restrict to the case of two space dimensions. The fluid step is dominated by the solution of the implicit systems, i.e. the Helmholtz and Poisson equations. When using a direct method

based upon cyclic reduction, the number of operations scales as $\mathcal{O}(n_1\, n_2\, \log(n_2))$ in both cases. Conversely, the principal work of the particle phase is due to the "interpolation" and "spreading" of Lagrangian/Eulerian quantities. In both cases, one has to evaluate a sum of the type

$$\sum_{i_p=1}^{n_p} \sum_{l=1}^{n_L} \sum_{i=i_1}^{i_2} \sum_{j=j_1}^{j_2} \delta_h(\mathbf{x}_{ij} - \mathbf{X}_l^{(i_p)}) \quad , \tag{5.1}$$

where i_1, i_2 refer to the indices which lead to the minimum operation count for a given support of the function δ_h (and analogously for j_1, j_2); n_p is the number of particles in the system. Therefore, the operation count scales as $\mathcal{O}(n_p\, n_L\, n_{st}(\delta_h)^2)$, where $n_{st}(\delta_h)$ is the stencil size of the discretized delta function, i.e. currently 4. Taking into account that for the number of marker points $\Delta s = \pi D/n_L \approx \Delta x = L_1/n_1$, we obtain: $\mathcal{O}(4^2\, n_p\, \pi n_1 D/L_1)$. This means that the work load scales linearly with the number of particles and also linearly with the number of grid points and the ratio between micro and macro length scales.

Time step. It has been mentioned in § 4.3 that the time step necessary for a stable solution of the fluid-particle system is very small when using the immersed boundary method. The restriction stems actually from the formulation of the singular boundary forces in terms of virtual springs. Lai and Peskin [22] mention that the maximum allowable time step scales as $\Delta t \approx \sqrt{\Delta x/\kappa}$, when disregarding the dampers. This means for the CFL number:

$$CFL = \frac{C\,||\mathbf{u}||}{\sqrt{\Delta x}\sqrt{\kappa}} \quad , \tag{5.2}$$

where the constant is of the order of π. Since the spring stiffness κ assures that the particles are really rigid, the question is: how low can we set the stiffness and still expect reasonable results? The answer to this question is presently not known in general, but needs to be determined from case to case. Unfortunately, it seems to lead to a restriction of the CFL number of the order of 0.01.

5.3 Future work

Parallelism. Although not essential for two-dimensional computations, we are currently working on a parallel implementation of the method. This tool will allow us to optimize the way the particle phase is treated in a multi-processor environment before taking the step to three space dimensions. The parallelization of the fluid phase should be straightforward: two-dimensional Cartesian processor mesh, one ghost cell at each (internal) boundary, exchange of the ghost cell data between neighbors by non-blocking communication after each intermediate solution step, parallel multi-grid solver for the Poisson equation of pseudo-pressure, ADI factorization with parallel tri-diagonal solver for the Helmholtz equations. On the other hand, distributing the work-load of the particle computations is more complicated. Especially, one has to deal with inherent load-balancing problems due to the fact that the particles are free to move across the domain. Still, it seems adequate to let each processor deal with all particles present in her sub-domain of the grid, implying to exchange particle data as they cross internal boundaries. Also, care needs to be taken as to share the work-load of the "spreading" and "interpolation" steps properly when a particle is overlapping an internal boundary.

Collision modeling. As we have noted in the drafting-kissing-tumbling case of § 4.3.2, the hydro-dynamic forces start to oscillate whenever particles get into very close contact, say of the order of the support of the discrete delta function (3.26). Therefore, in practice the interaction between particles needs to be treated in a special way. Presently it is not clear how to improve over the current practice of repulsive potentials (e.g. [6]). It is planned to undertake a study in which a highly-resolved computation of the controlled collision between two particles is compared

to results obtained by collision models. Reference to experimental data seems important for this step.

Three dimensional computations. Finally, the logical step is to extend the current method for three space dimensions, which should be straightforward. The computational cost being very large, the emphasis here is upon efficiency of the parallel implementation, and particularly upon the solution strategy for the Helmholtz and Poisson problems.

Acknowledgements

Useful discussions with A. Pinelli and F. Higuera are gratefully acknowledged.
I am grateful to Z. Zhang and T.W. Pan for providing their numerical results for comparison.

Bibliography

[1] E.A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *J. Comput. Phys.*, 161:35–60, 2000.

[2] J. Kim, D. Kim, and H. Choi. An immersed-boundary finite-volume method for simulations of flow in complex geometries. *J. Comput. Phys.*, 171:132–150, 2001.

[3] T. Kajishima, S. Takiguchi, and Y. Miyake. Modulation and subgrid scale modeling of gas-particle turbulent flow. In D. Knight and L. Sakell, editors, *Recent Advances in DNS & LES*, pages 235–244. Kluwer, 1999.

[4] T.W. Pan, D.D. Joseph, R. Bai, R. Glowinski, and V. Sarin. Fluidization of 1204 spheres: simulation and experiment. *J. Fluid Mech.*, 451:169–191, 2002.

[5] N.A. Patankar. A formulation for fast computations of rigid particulate flows. *CTR Res. Briefs*, pages 185–196, 2001.

[6] K. Höfler and S. Schwarzer. Navier-Stokes simulation with constraint forces: Finite-difference method for particle-laden flows and complex geometries. *Phys. Rev. E*, 2000.

[7] C.S. Peskin. The immersed boundary method. *Acta Numerica*, pages 1–39, 2002.

[8] D. Goldstein, R. Handler, and L. Sirovich. Modeling a no-slip boundary with an external force field. *J. Comput. Phys.*, 105:354–366, 1993.

[9] Z. Zhang and A. Prosperetti. A method for particle simulation. *J. Appl. Mech.*, 70:1–11, 2003.

[10] T. Kajishima and S. Takiguchi. Interaction between particle clusters and particle-induced turbulence. *Int. J. Heat Fluid Flow*, 23:639–646, 2002.

[11] G.K. Batchelor. *An introduction to fluid dynamics*. Cambridge U. Press, 1967.

[12] D.L. Brown, R. Cortez, and M.L. Minion. Accurate projection methods for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 168:464–499, 2001.

[13] M.M. Rai and P. Moin. Direct simulation of turbulent flow using finite-difference schemes. *J. Comput. Phys.*, 96:15–53, 1991.

[14] Y. Morinishi, T.S. Lund, O.V. Vasilyev, and P. Moin. Fully conservative higher order finite difference schemes for incompressible flow. *J. Comput. Phys.*, 143:90–124, 1998.

[15] A.E. Veldman. "Missing" boundary conditions? Discretize first, substitute next, and combine later. *SIAM J. Sci. Stat. Comput.*, 11(1):82–91, 1990.

[16] U. Schumann and R.A. Sweet. A direct method for the solution of Poisson's equation with Neumann boundary conditions on a staggered grid of arbitrary size. *J. Comput. Phys.*, 20:171–182, 1976.

[17] P. Swarztrauber and R. Sweet. Efficient FORTRAN subprograms for the solution of elliptic partial differential equations. Technical Report TN/IA-109, NCAR, 1975.

[18] Numerical Algorithms Group. *NAG Fortran Library Manual Mark 20*, 2002.

[19] R. Verzicco, J. Mohd-Yusof, P. Orlandi, and D. Haworth. Les in complex geometries using boundary body forces. *CTR Proc. Summer Prog.*, pages 171–186, 1998.

[20] Z. Li and M.-C. Lai. The immersed interface method for the Navier-Stokes equations with singular forces. *J. Comput. Phys.*, 171:822–842, 2001.

[21] J. Mohd-Yusof. Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries. *CTR Res. Briefs*, pages 317–327, 1997.

[22] M.-C. Lai and C.S. Peskin. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *J. Comput. Phys.*, 160:705–719, 200.

[23] H. Schlichting. *Boundary-Layer Theory.* McGraw-Hill, seventh edition, 1979.

[24] H.M. Blackburn and R.D. Henderson. A study of two-dimensional flow past an oscillating cylinder. *J. Fluid Mech.*, 385:255–286, 1999.

[25] T. Kajishima, S. Takiguchi, H. Hamasaki, and Y. Miyake. Turbulence structure of particle-laden flow in a vertical plane channel due to vortex shedding. *JSME Int. J., Series B*, 44(4):526–535, 2001.

[26] R. Jackson. *The dynamics of fluidized particles.* Cambridge U. Press, 2000.

[27] N.A. Patankar, P. Singh, D.D. Joseph, R. Glowinski, and T.-W. Pan. A new formulation of the distributed Lagrange multiplier/ficticious domain method for particulate flows. *Int. J. Multiphase Flow*, 26:1509–1524, 2000.

[28] R. Verzicco and P. Orlandi. A finite-difference scheme for three-dimensional incompressible flows in cylindrical coordinates. *J. Comput. Phys.*, 123:402–414, 1996.

[29] J. Kim and P. Moin. Application of a fractional-step method to incompressible Navier-Stokes equations. *J. Comput. Phys.*, 59:308–323, 1985.

[30] A.A. Wray. Minimal storage time-advancement schemes for spectral methods. unpublished manuscript.

[31] P. Spalart, R.D. Moser, and M.M. Rogers. Spectral methods for the Navier-Stokes equations with one infinite and two periodic directions. *J. Comput. Phys.*, 96:297–324, 1991.

[32] J. Stoer and R. Bulirsch. *Introduction to numerical analysis.* Springer, 1980.

[33] R. Aris. *Vectors, Tensors, and the Basic Equations of Fluid Mechanics.* Dover Science and Maths, 1962.

Figures

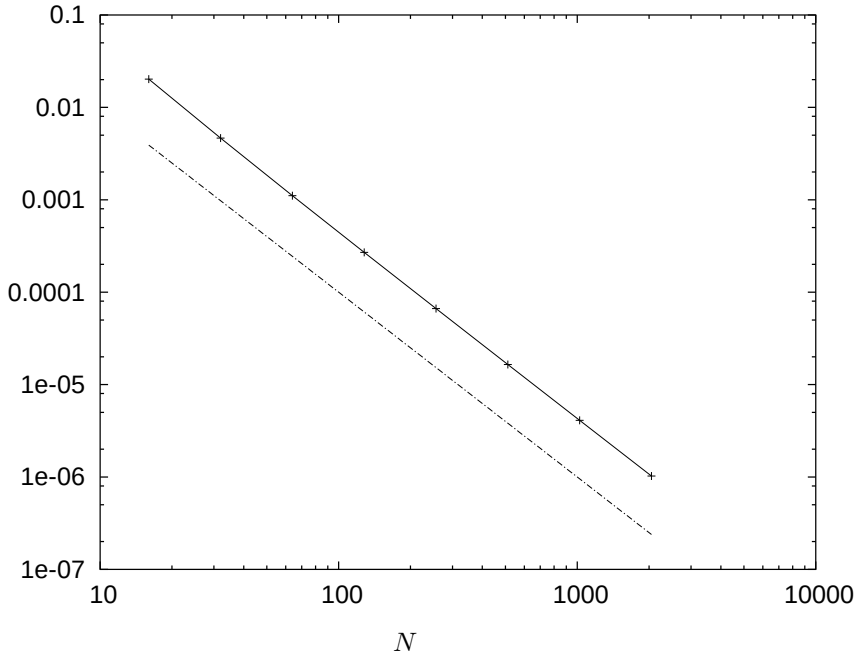

Figure 5.1: Maximum relative error of the solution of an analytic Poisson problem using modified routines from the FISHPACK library as a function of the number of nodes $N_x = N_y = N$. The dashed straight line has a slope of N^{-2}.

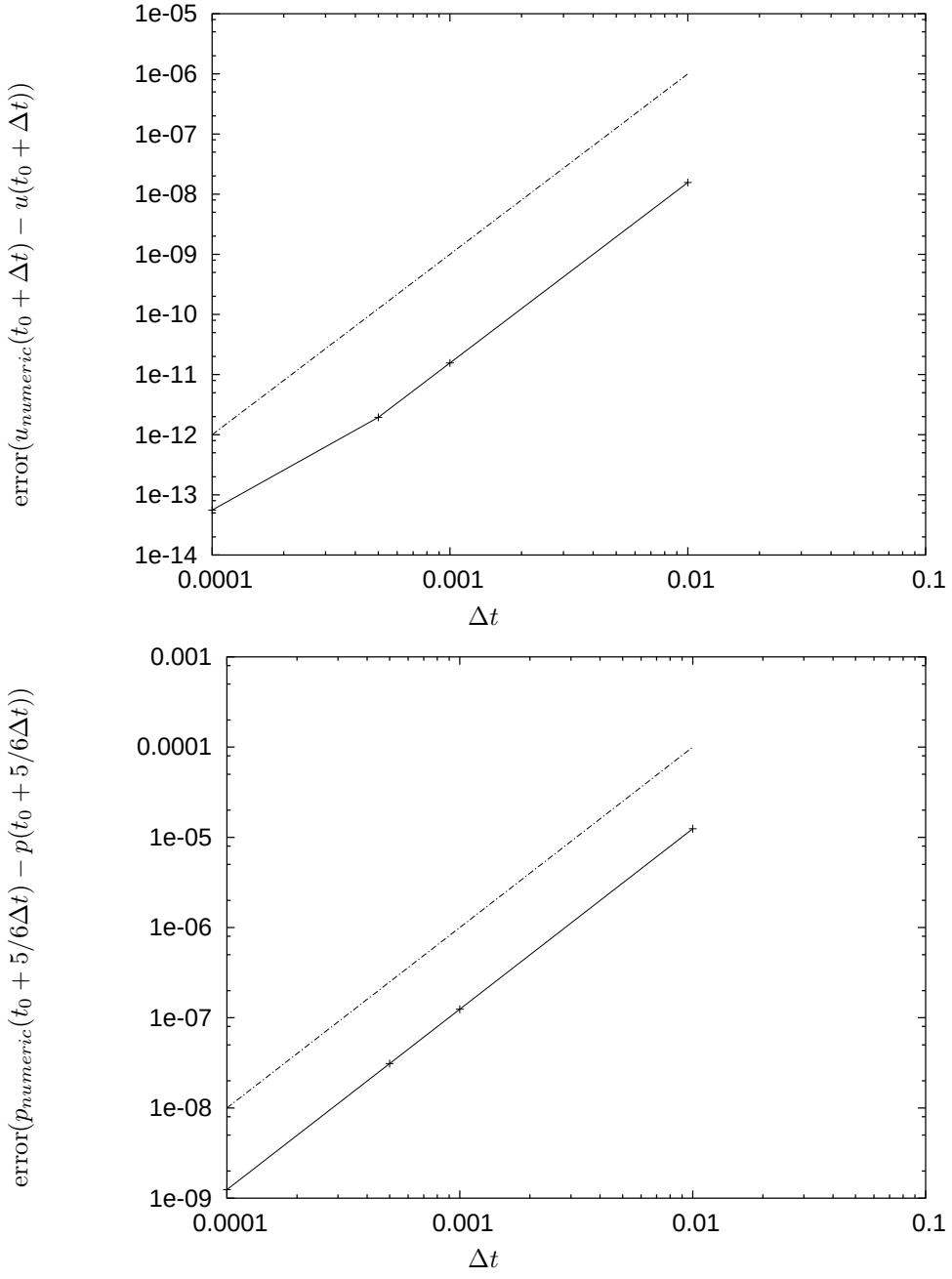

Figure 5.2: Maximum relative error of velocity (top) and pressure (bottom) as a function of the time step when solving the Taylor-Green problem on a periodic domain and using fully spectral spatial operators. The dashed straight reference lines are proportional to Δt^3 and Δt^2, respectively.

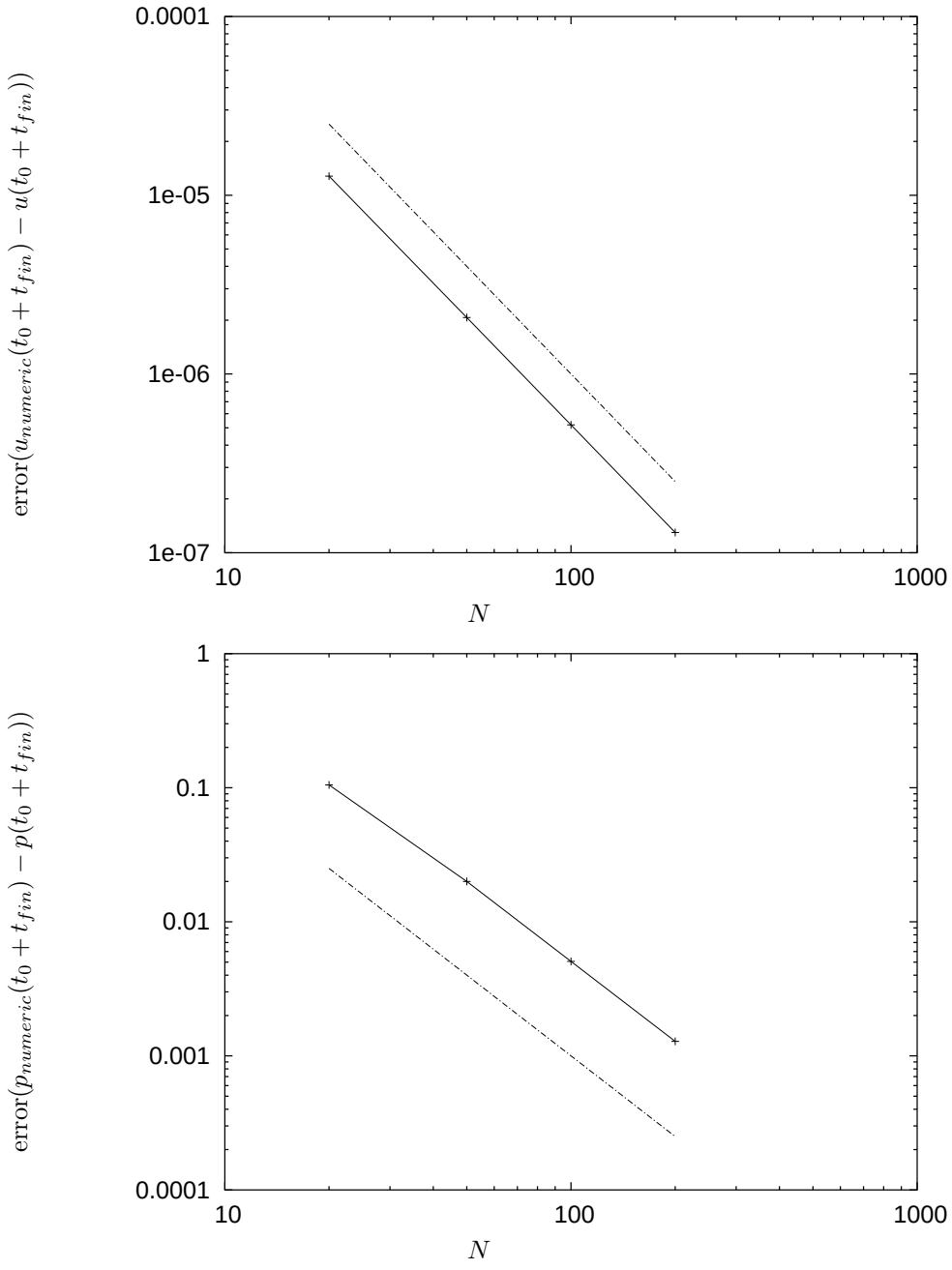

Figure 5.3: Maximum relative error of velocity (top) and pressure (bottom) as a function of the number of (pressure) grid nodes when solving the Taylor-Green problem on a periodic domain and using finite-difference operators. The dashed straight reference lines are proportional to N^{-2}.

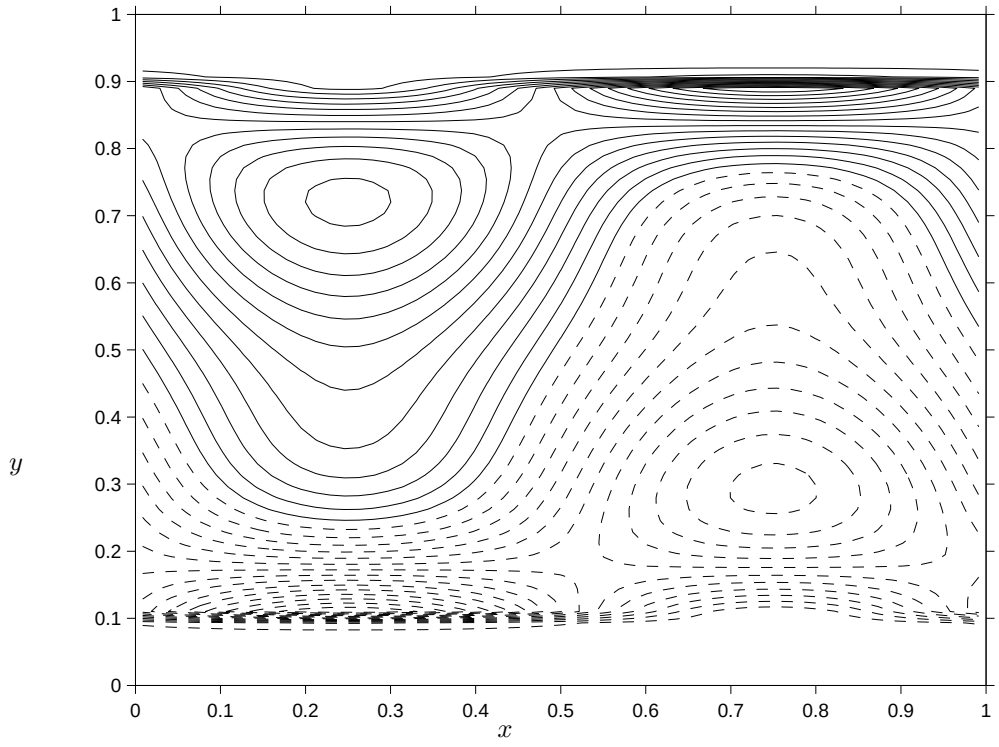

Figure 5.4: Initial vorticity field of the perturbed Poiseuille flow of § 3.1.3.1. Dashed lines correspond to negative vorticity.

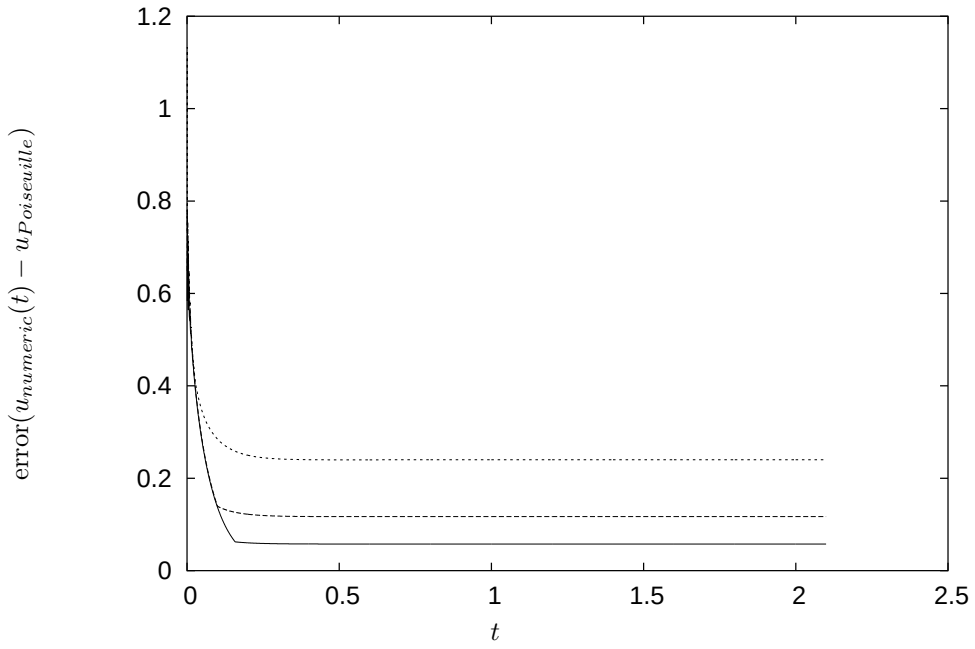

Figure 5.5: Time history of the convergence towards a parabolic unidirectional flow (Poiseuille) from an initially perturbed field. The different lines correspond to various spatial resolutions: $n_\alpha = 30$ (dotted), $n_\alpha = 60$ (dashed), $n_\alpha = 120$ (solid).

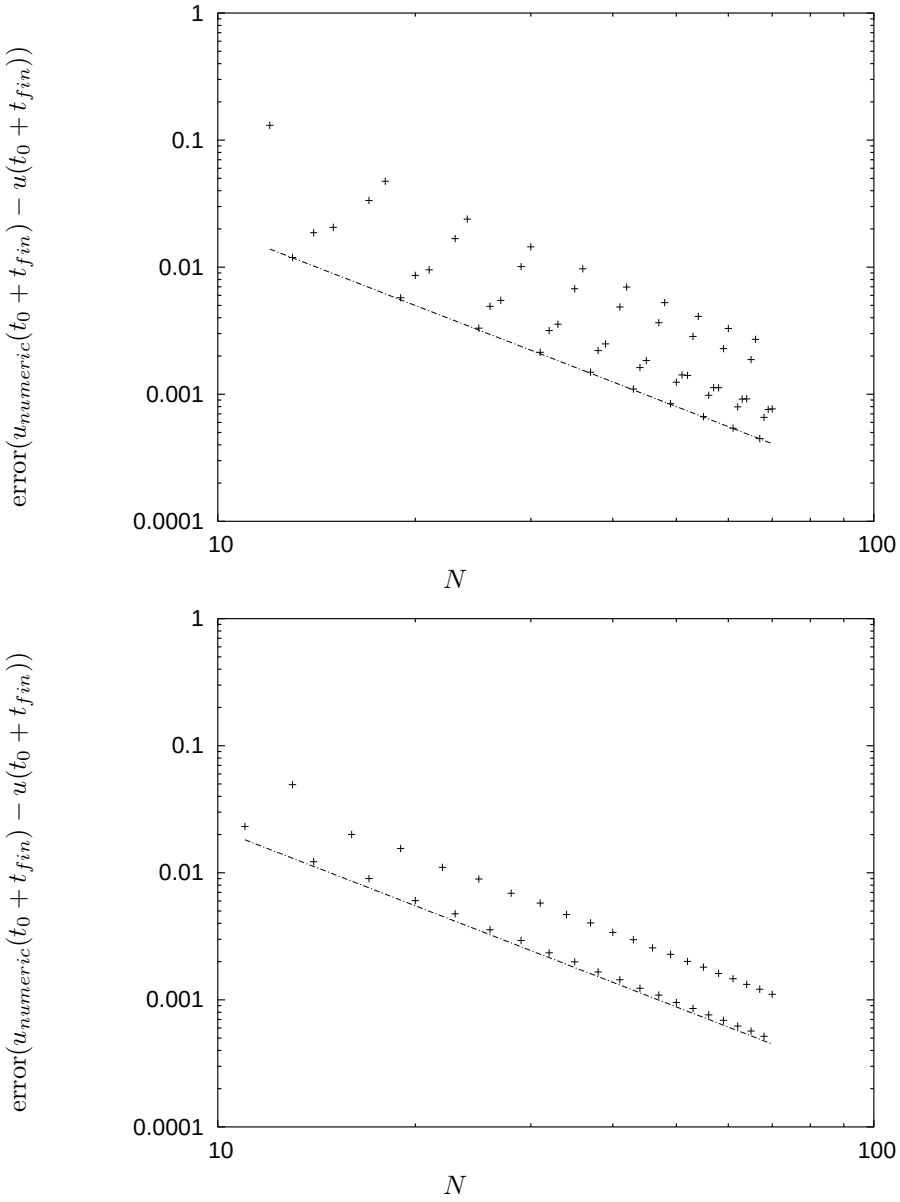

Figure 5.6: Maximum relative error of velocity as a function of the number of (pressure) grid nodes when solving the Taylor-Green problem in a sub-domain and using the direct explicit forcing method of reference [2] as described in § 3.1.1. The sub-domain is aligned with the cartesian grid (top) and rotated by 45° w.r.t. the grid lines (bottom). The dashed straight reference lines are proportional to N^{-2}. Please note that the numerical solution diverged for $N = 10, 11, 16, 22, 28, 34, 40, 46$ and $N = 10, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69$, respectively. Note that indistinguishable results were obtained with the implicit method of § 3.1.2 (not shown).

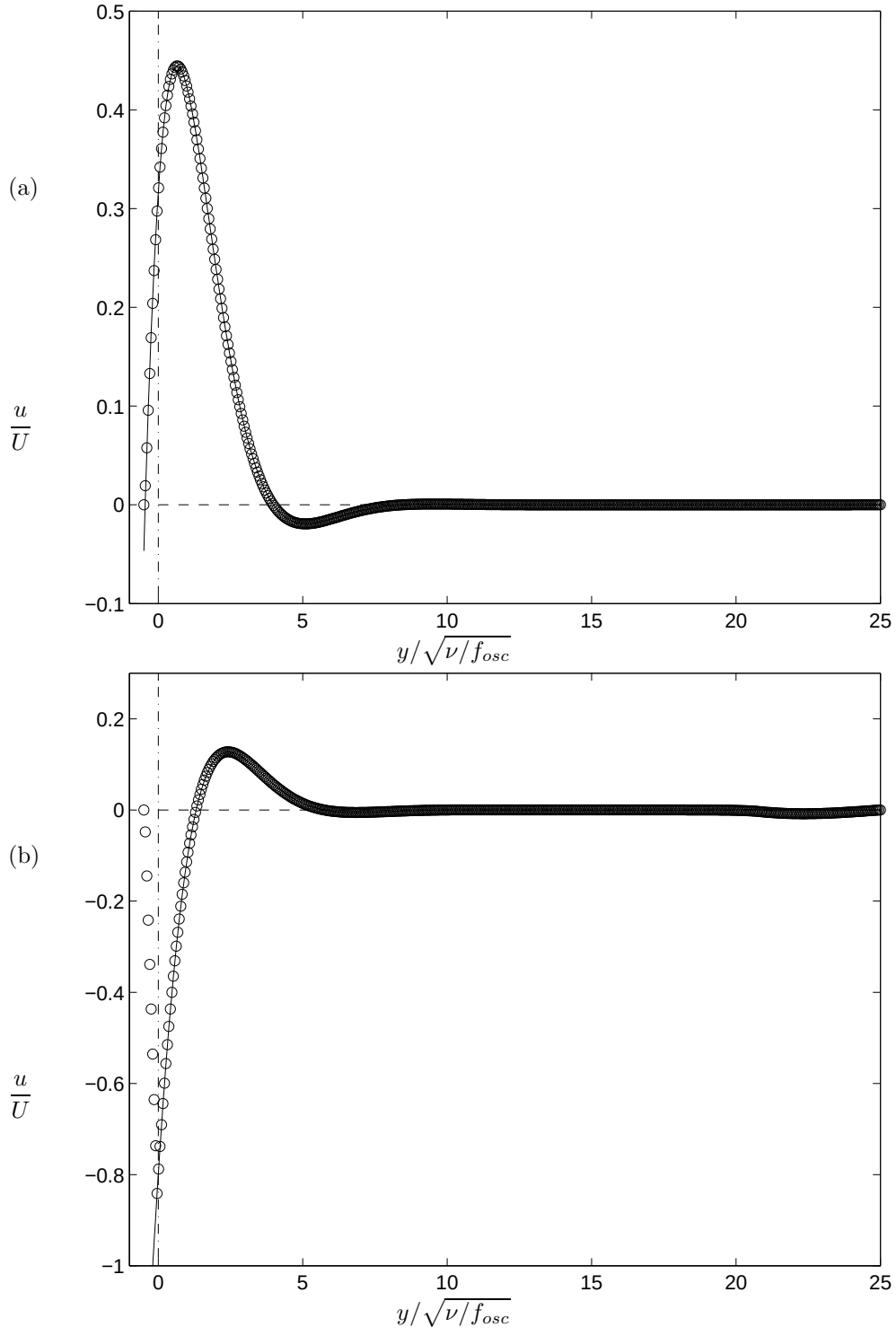

Figure 5.7: Results for the problem of flow induced by the oscillation of a flat plate (located at $y = 0$) and obtained by the direct forcing method of reference [2] as described in § 3.1.1. The plots show the velocity parallel to the plate at times: (a) $t/T = 0.1989$; (b) $t/T = 0.3979$. The solid line gives the analytic solution of Batchelor (3.15) and the symbols corrspond to the numerical results. Please note that the solution for $y < 0$ does not have any physical meaning and is in practice discarded.

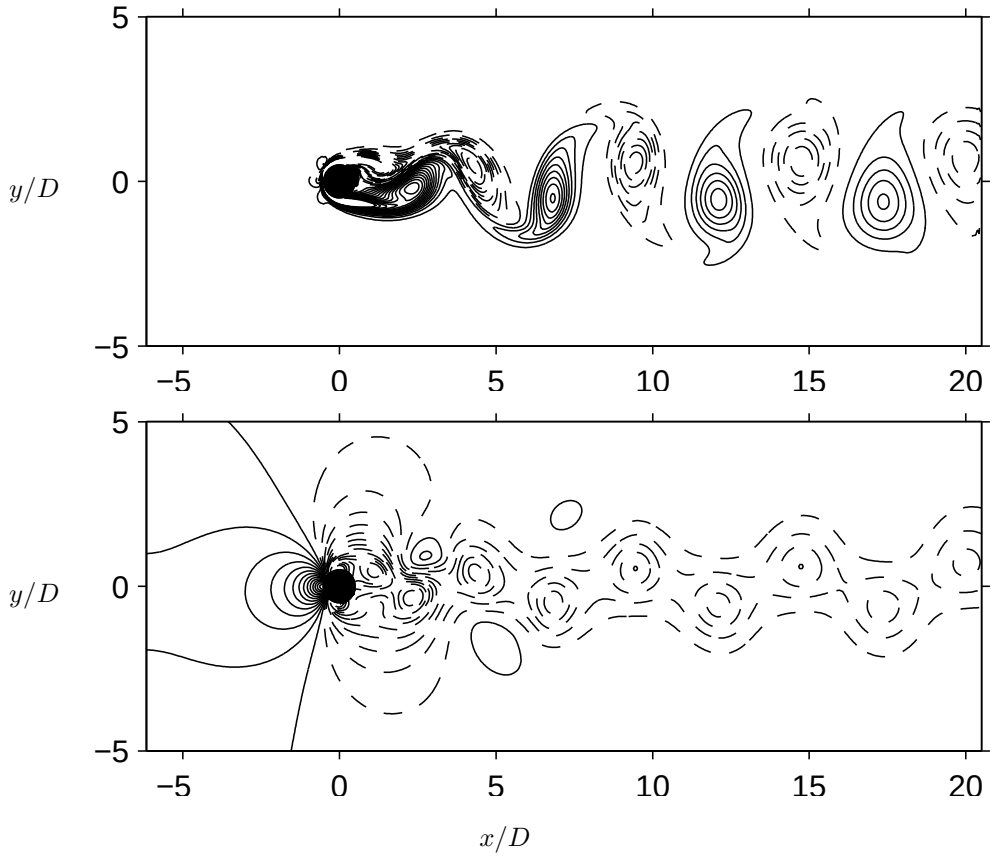

Figure 5.8: Instantaneous flow-fields obtained by the direct explicit forcing method in the case of a fixed cylinder in uniform cross-flow at $Re_D = 100$ and $t = 100$. Top: vorticity contours at values (-10:.69:10). Bottom: pressure fluctuation contours with respect to some arbitrary reference pressure at (-1:.069:1).

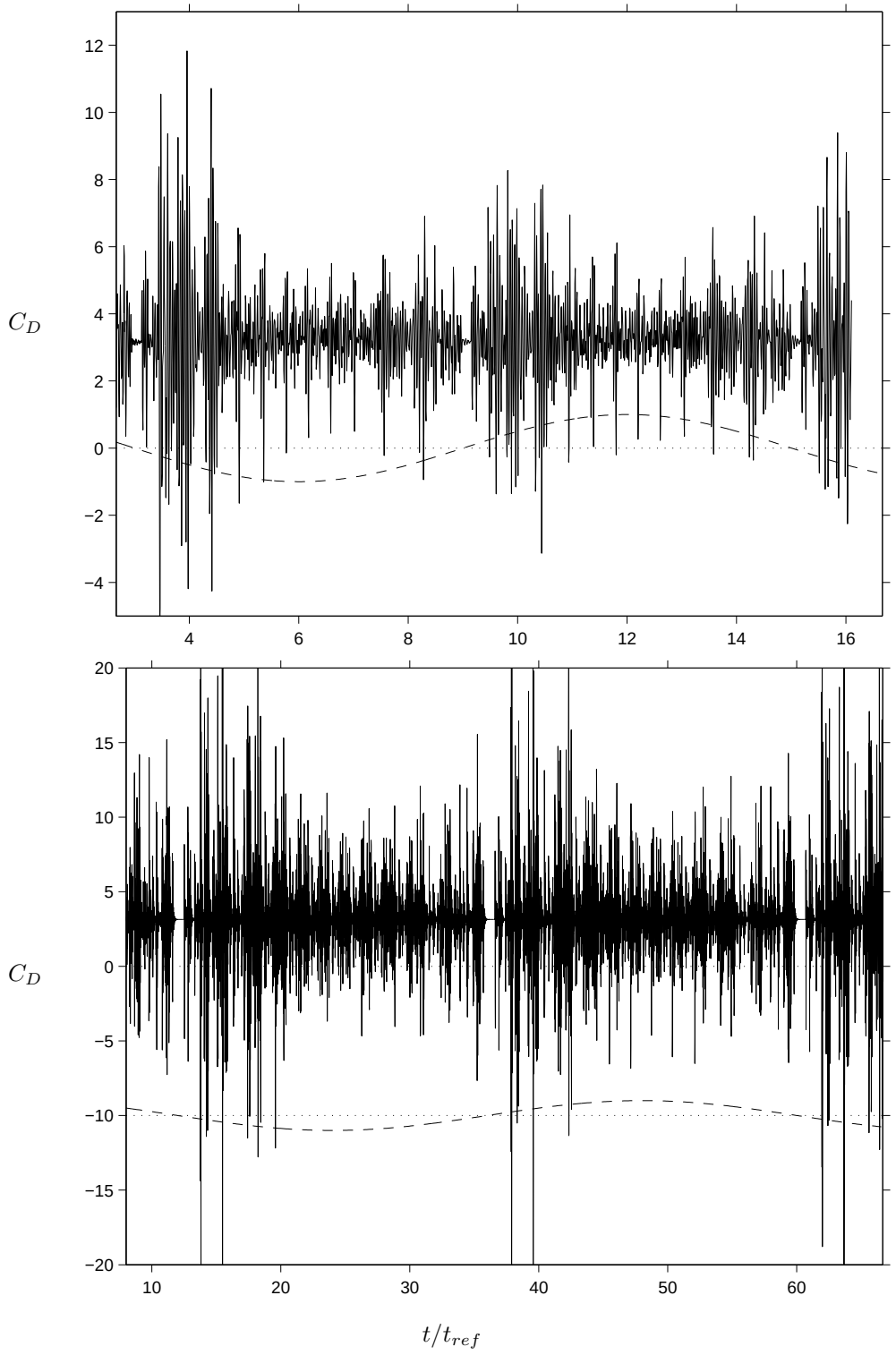

Figure 5.9: Drag coefficient obtained from the simulation of the flow around a translationally oscillating cylinder at $Re_D = 10$ using the explicit direct forcing method. The frequency of the oscillation is equal to the natural shedding frequency $f_f = f_n$ (top), and equal to one forth $f_f = f_n/4$ (bottom). For comparison, the dashed line shows the corresponding phase of the vertical velocity $v_c(t)$ of the solid body (vertically offset by -10 for clarity in the lower case). The correlation between low-noise regions of the drag and low translational velocity is obvious. Please note the different scales of both axes.

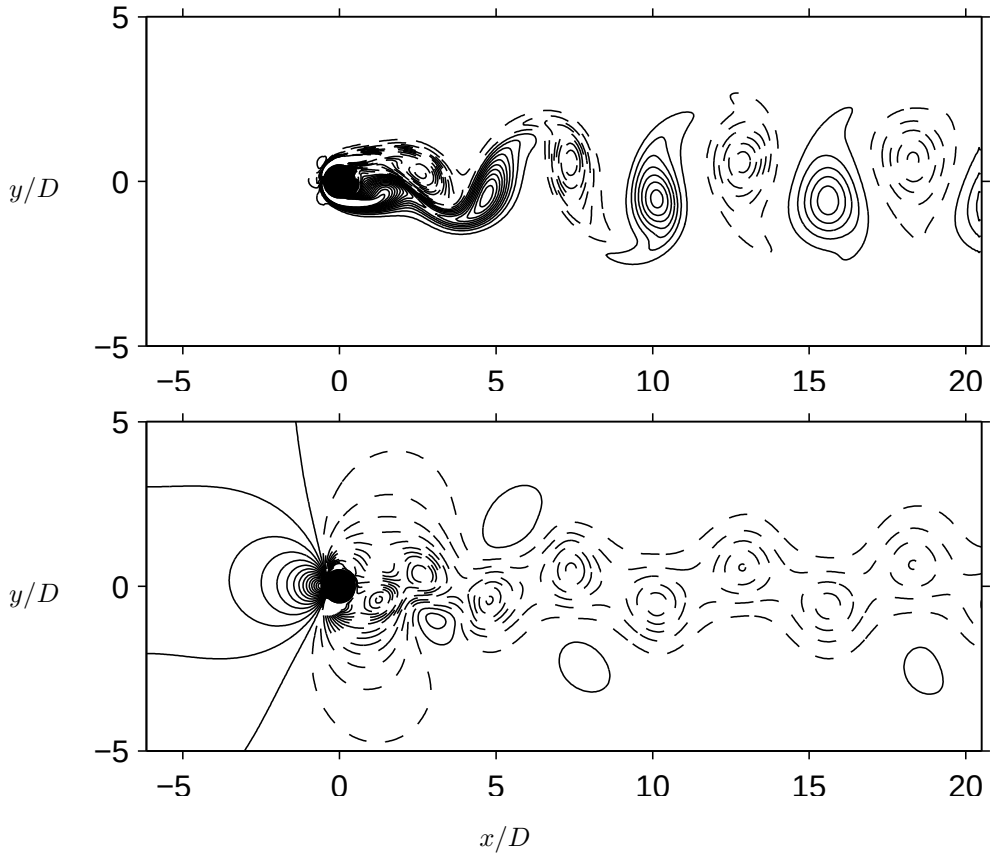

Figure 5.10: Instantaneous flow-fields obtained by the immersed boundary method in the case of a fixed cylinder in uniform cross-flow at $Re_D = 100$ and $t = 100$. Top: vorticity contours at values (-10:.69:10). Bottom: pressure fluctuation contours with respect to some arbitrary reference pressure at (-1:.069:1).

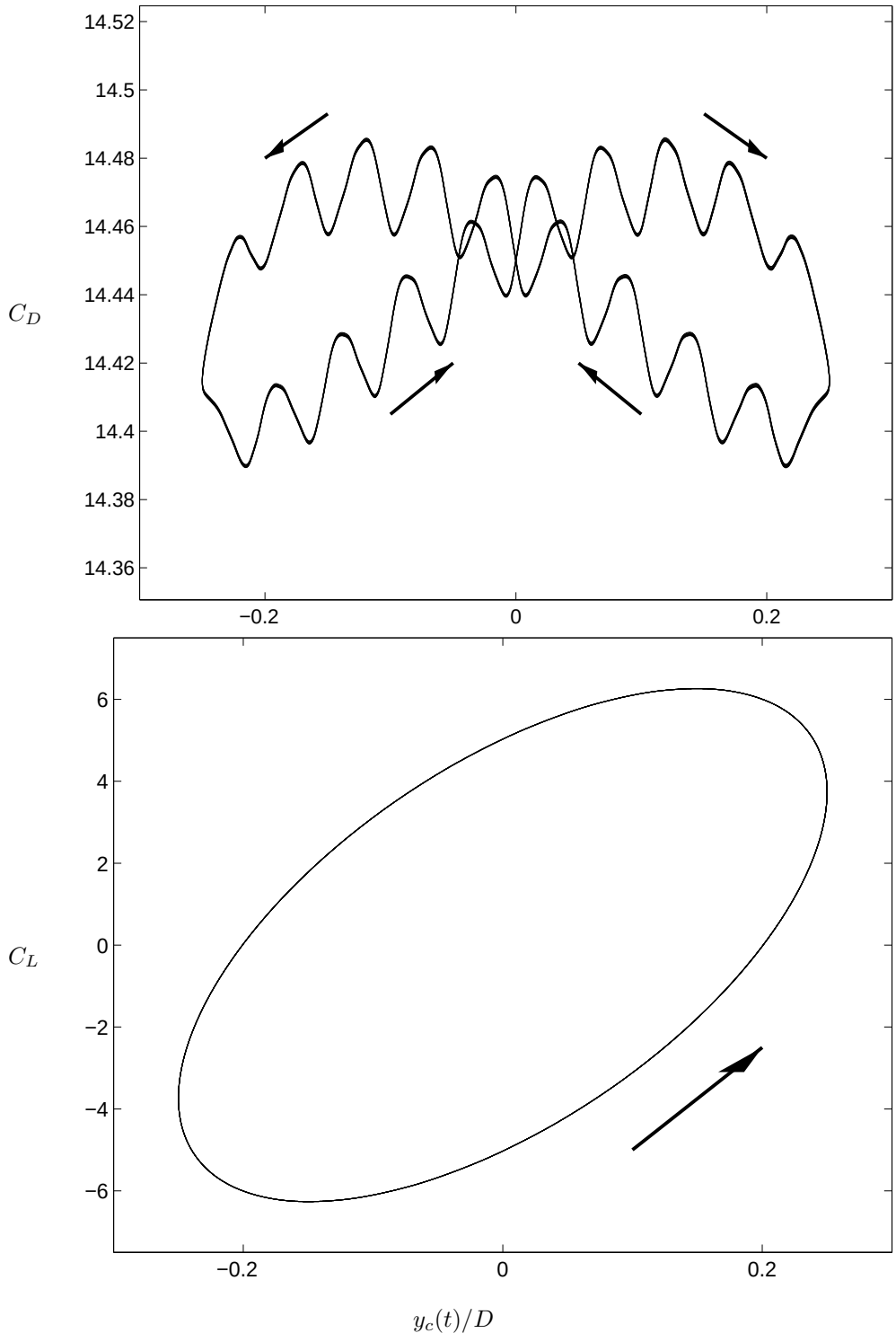

Figure 5.11: Dimensionless coefficients obtained by the immersed boundary method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 1$. The plots show phase-space diagrams of drag coefficient versus cylinder position (top) and lift coefficient vs. cylinder position (bottom). The arrows indicate the direction of the trajectories in time.

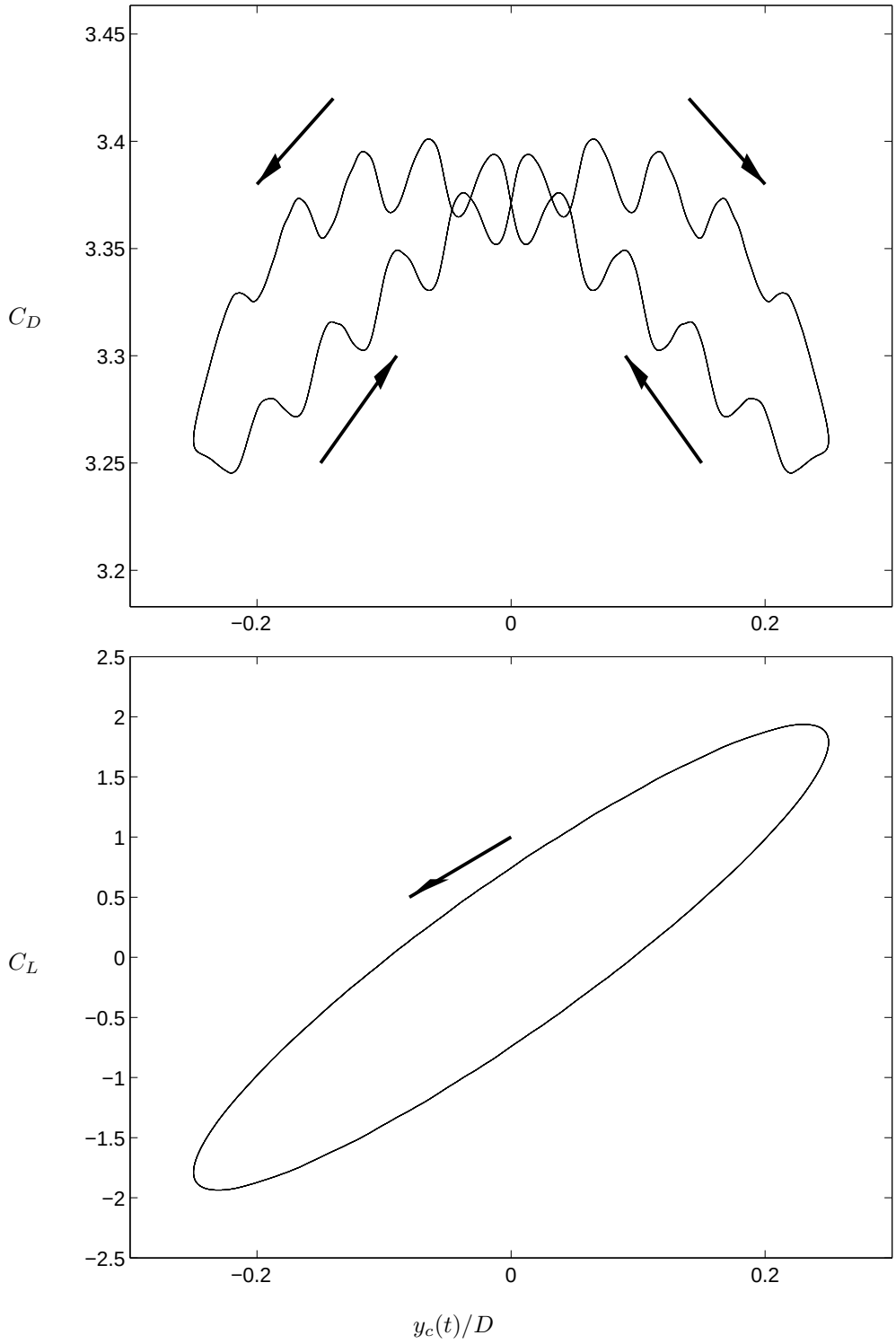

Figure 5.12: Dimensionless coefficients obtained by the immersed boundary method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 10$. The plots show phase-space diagrams of drag coefficient versus cylinder position (top) and lift coefficient vs. cylinder position (bottom). The arrows indicate the direction of the trajectories in time.

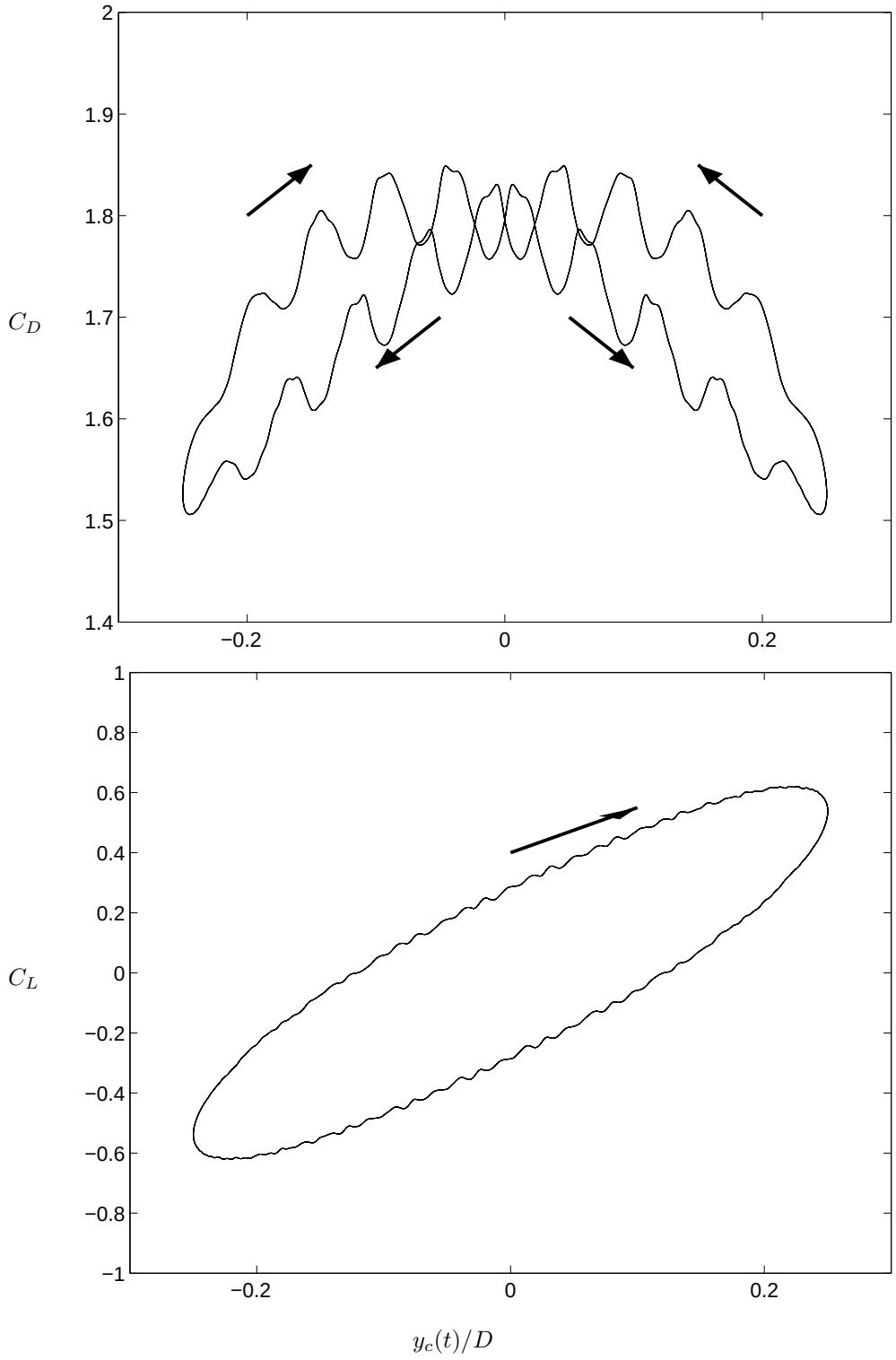

Figure 5.13: Dimensionless coefficients obtained by the immersed boundary method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 100$. The plots show phase-space diagrams of drag coefficient versus cylinder position (top) and lift coefficient vs. cylinder position (bottom). The arrows indicate the direction of the trajectories in time.

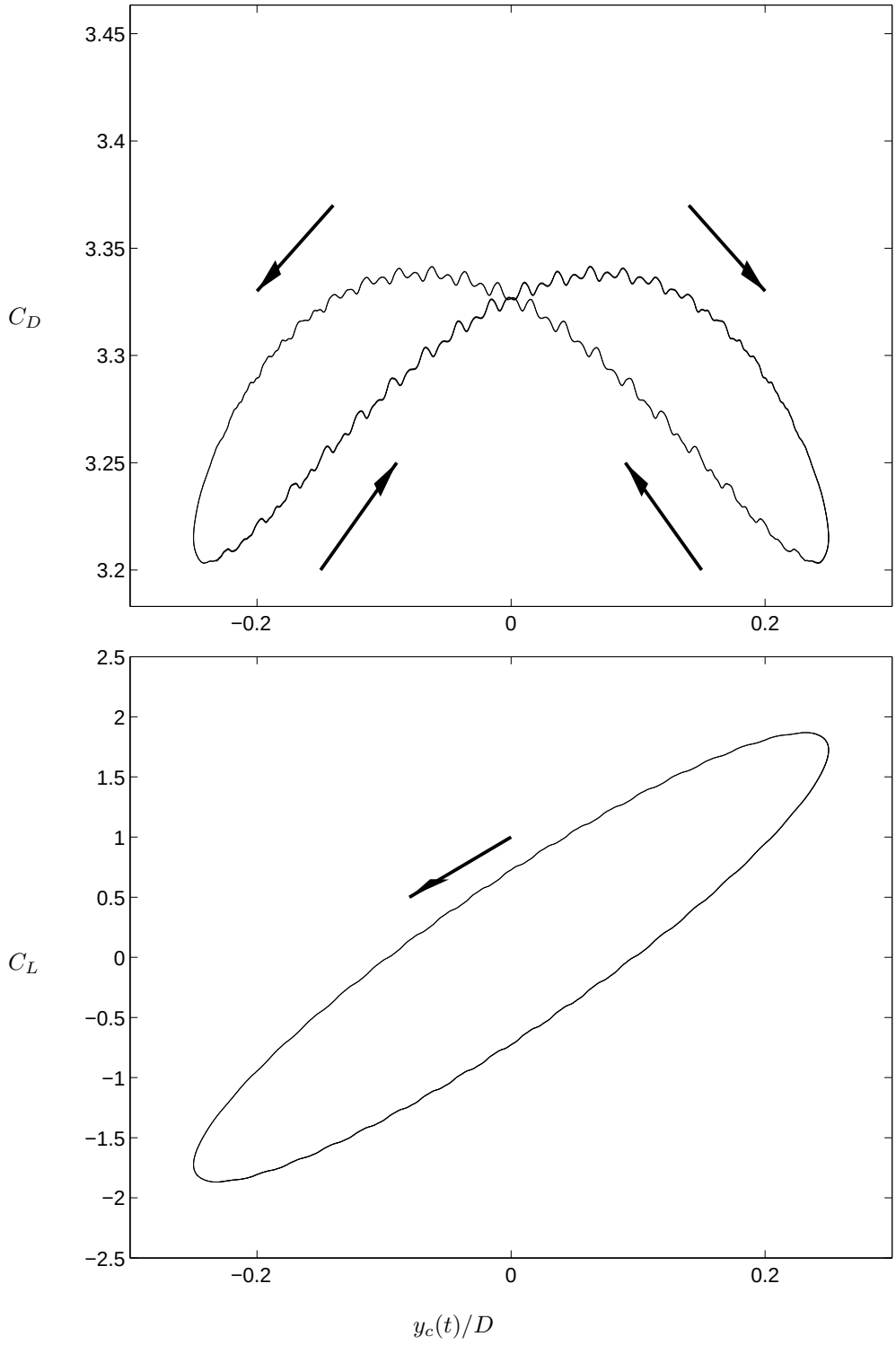

Figure 5.14: Dimensionless coefficients obtained by the immersed boundary method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 10$ at increased spatial resolution $n_\alpha = 1024$. The plots show phase-space diagrams of drag coefficient versus cylinder position (top) and lift coefficient vs. cylinder position (bottom). The arrows indicate the direction of the trajectories in time.

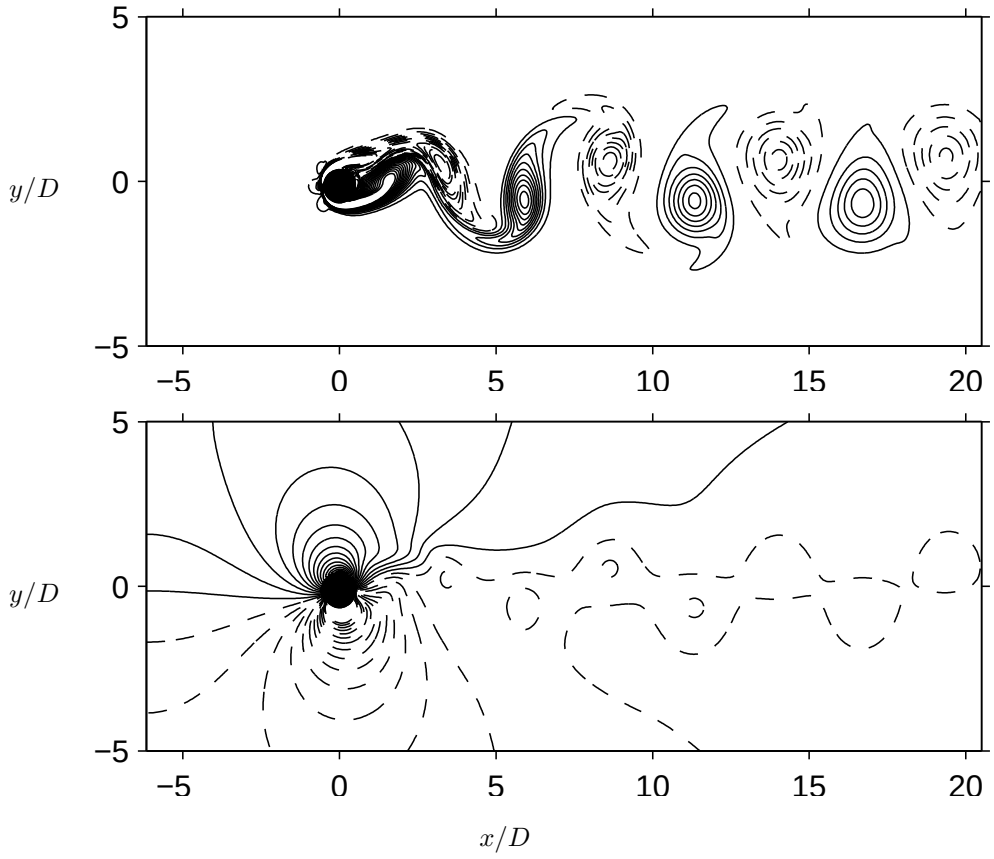

Figure 5.15: Instantaneous flow-fields obtained by the immersed boundary method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 100$ and at an instant when the cylinder's position is near the origin with nearly maximum (negative) vertical velocity ($v_c = -0.21$). Top: vorticity contours at values (-10:.69:10). Bottom: pressure fluctuation contours with respect to some arbitrary reference pressure at (-1:.069:1).

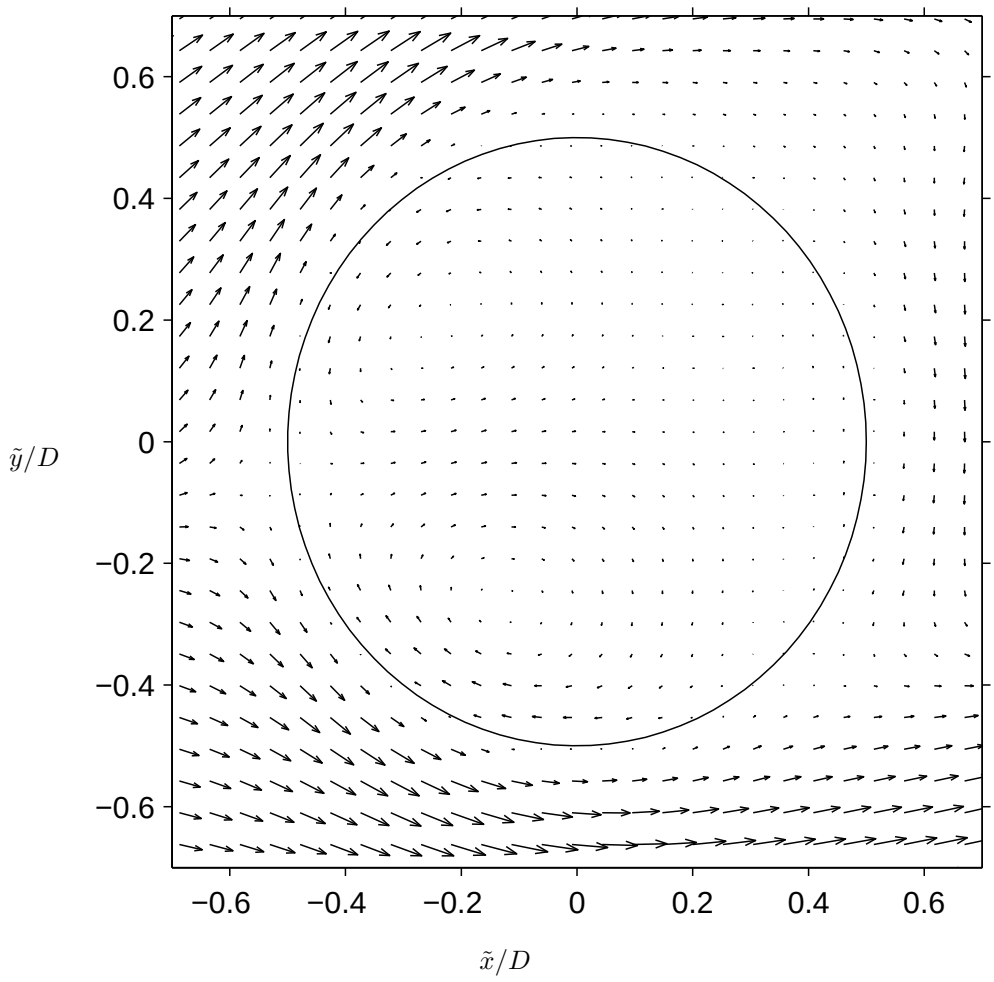

Figure 5.16: Close-up of the instantaneous flow-field obtained by the immersed boundary method in the case of figure 5.15. The velocity vectors (linearly interpolated to the pressure nodes) are shown in a reference frame $(\tilde{x}, \tilde{y})$ moving with the cylinder.

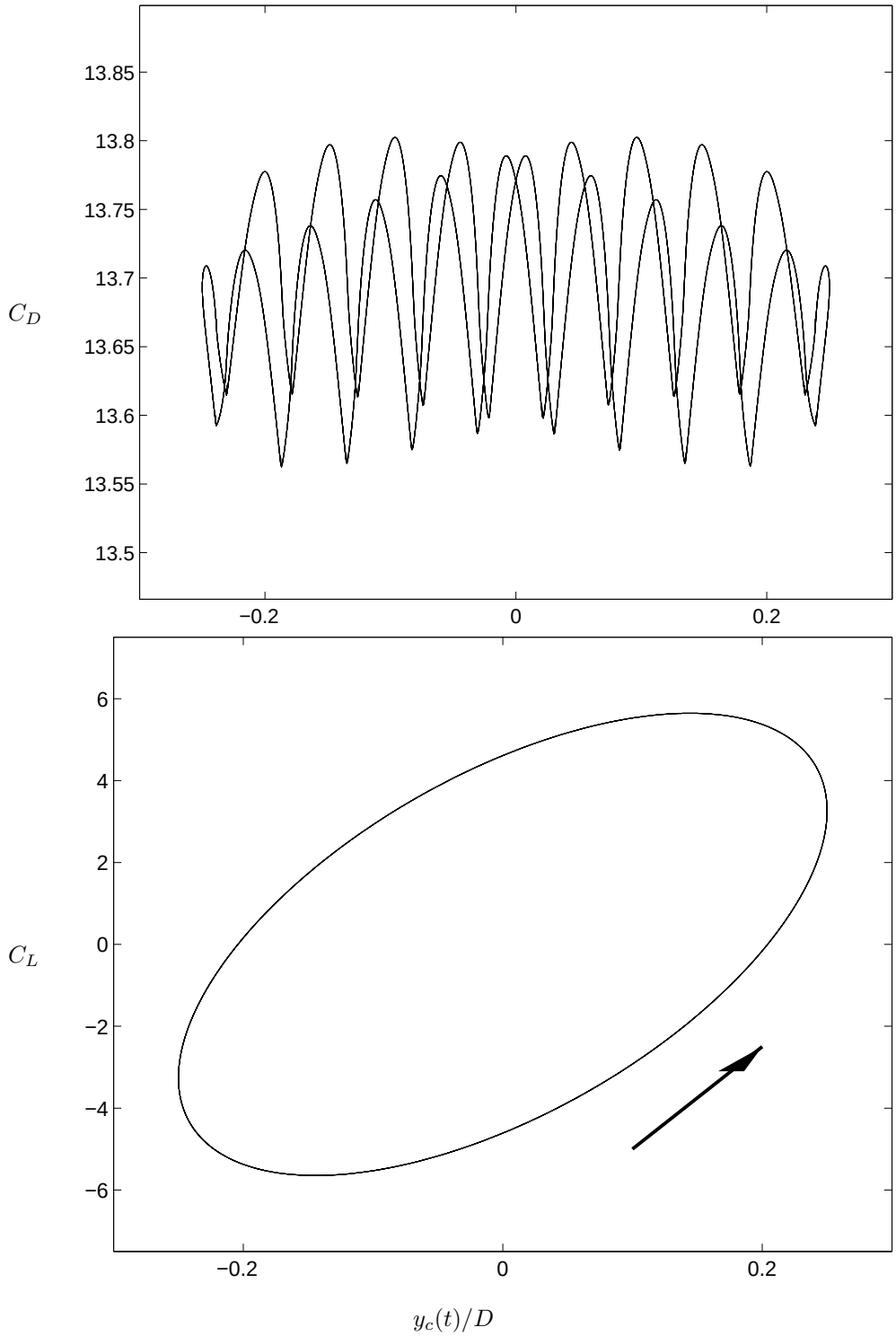

Figure 5.17: Dimensionless coefficients obtained by the rigidity method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 1$.

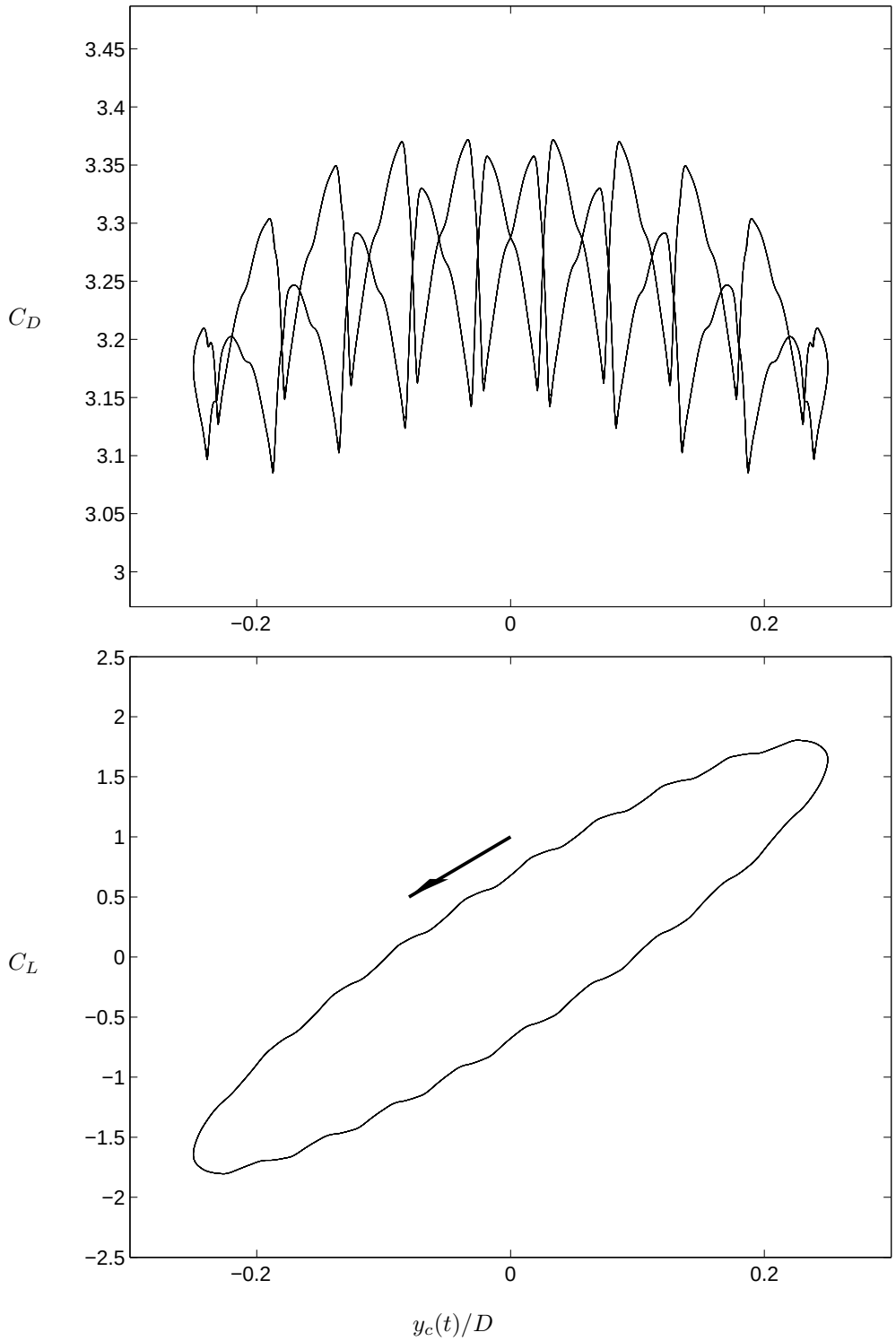

Figure 5.18: Dimensionless coefficients obtained by the rigidity method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 10$.

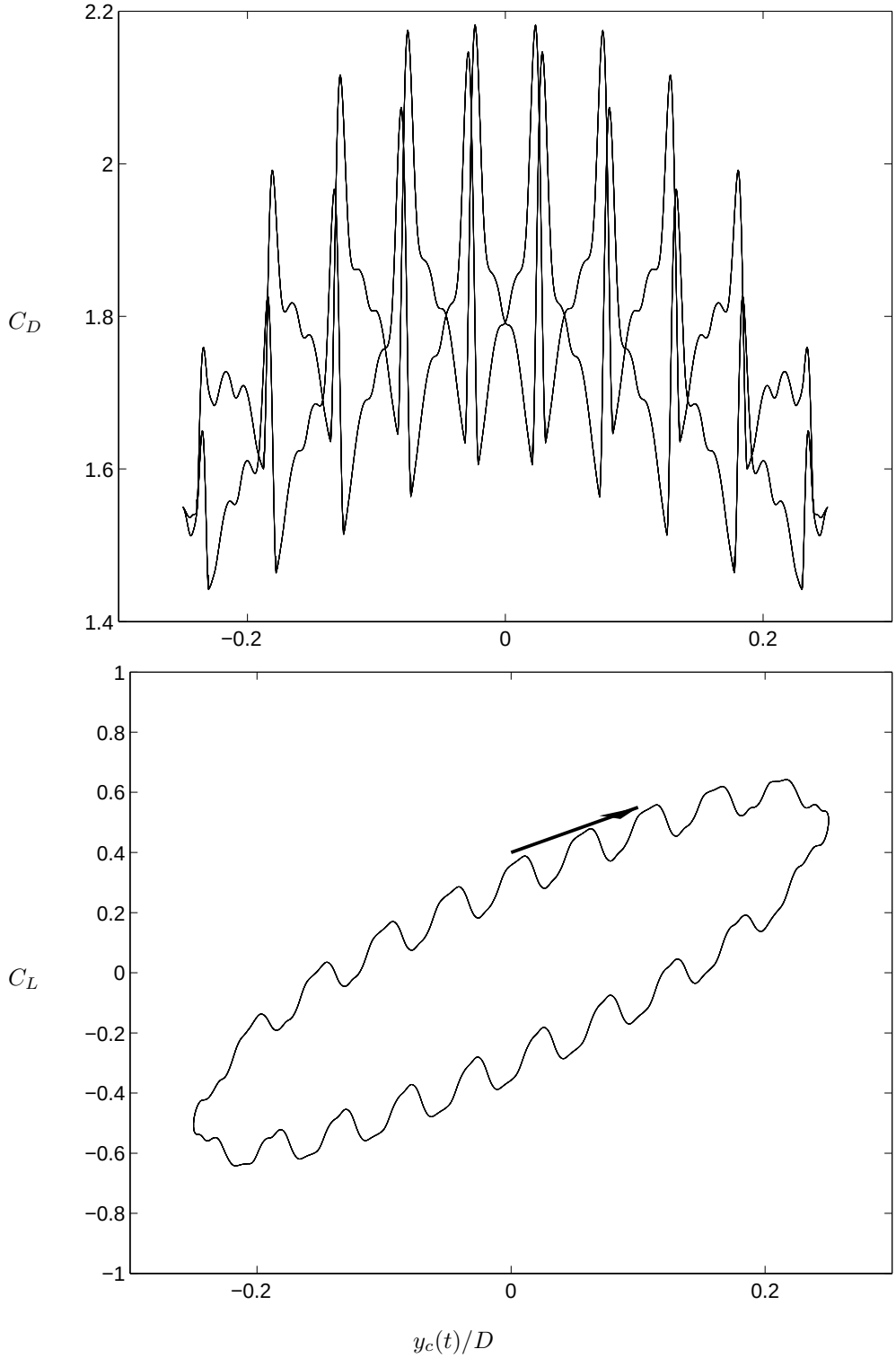

Figure 5.19: Dimensionless coefficients obtained by the rigidity method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 100$. Note that the vertical scale differs from that of figure 5.13.

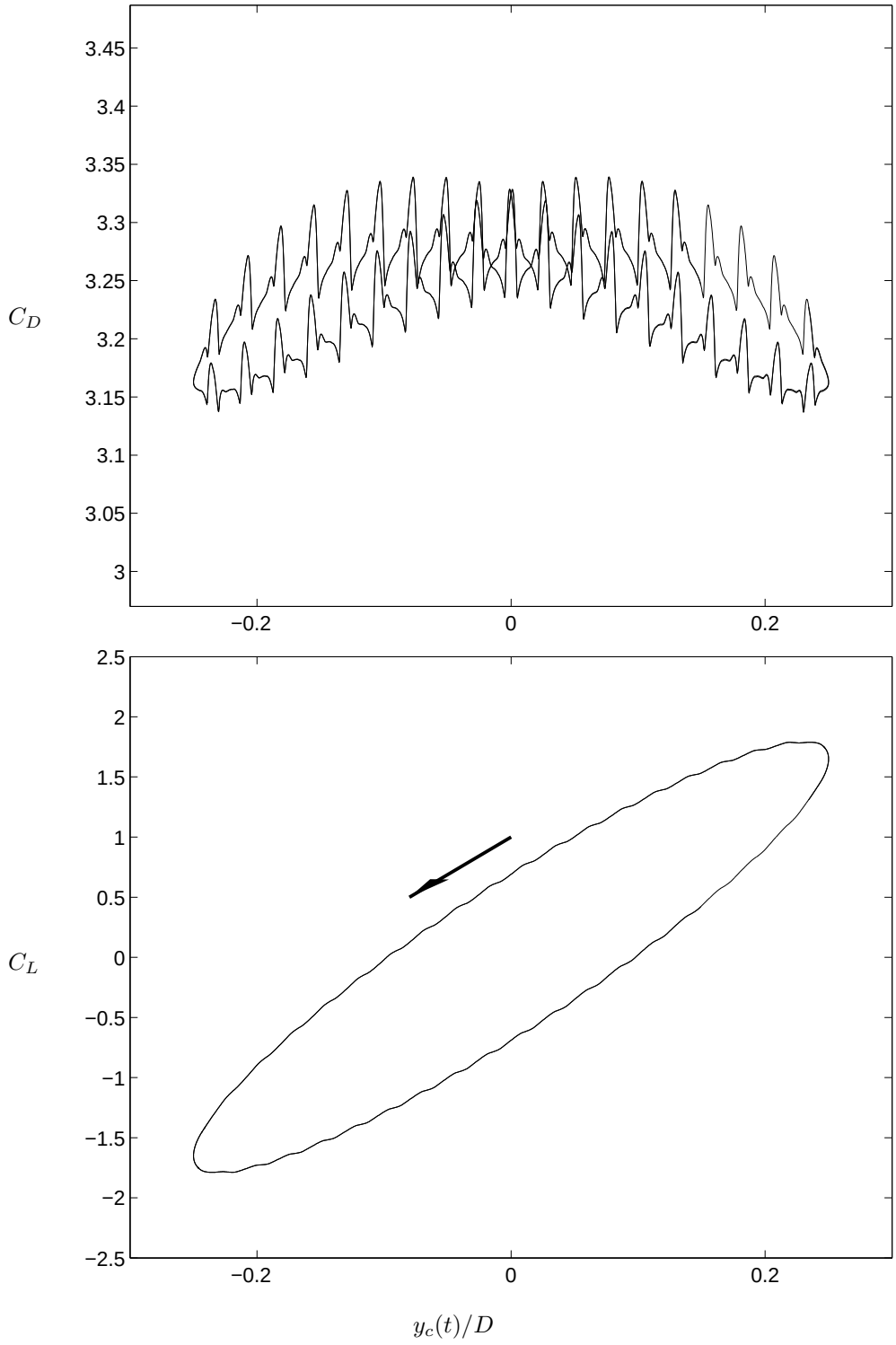

Figure 5.20: Dimensionless coefficients obtained by the rigidity method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 10$ at increased spatial resolution $n_\alpha = 1024$. The plots show phase-space diagrams of drag coefficient versus cylinder position (top) and lift coefficient vs. cylinder position (bottom). The arrows indicate the direction of the trajectories in time.

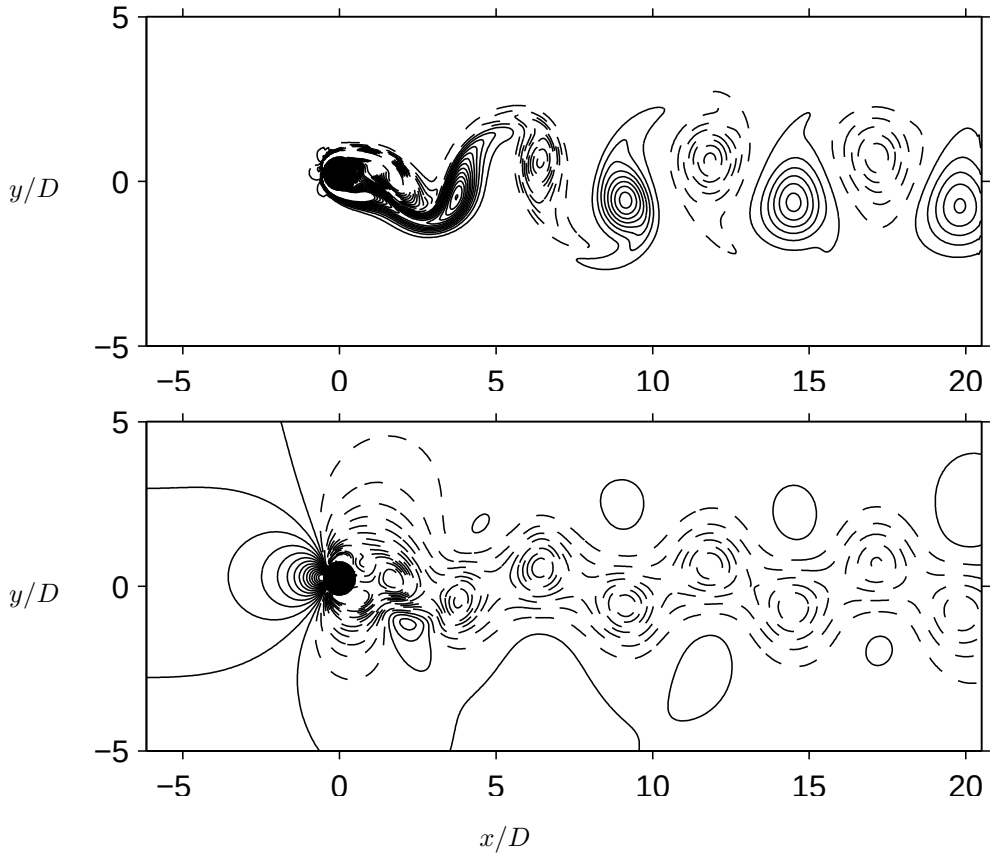

Figure 5.21: Instantaneous flow-fields obtained by the rigidity method in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 100$ and at an instant when the cylinder's position is near the upper extremum with approximately one third of the maximum (positive) vertical velocity ($v_c = 0.08$). Top: vorticity contours at values (-10:.69:10). Bottom: pressure fluctuation contours with respect to some arbitrary reference pressure at (-1:.069:1).

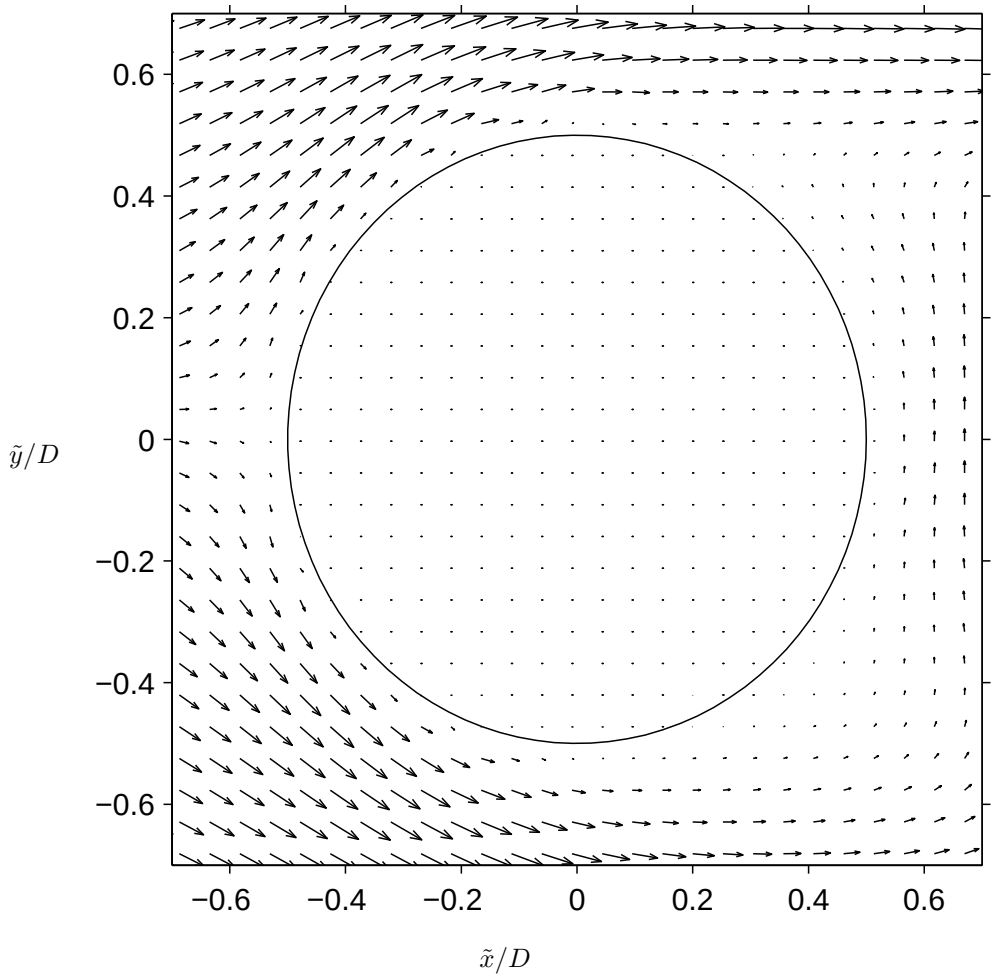

Figure 5.22: Close-up of the instantaneous flow-field obtained by the rigidity boundary method in the case of figure 5.21. The velocity vectors (linearly interpolated to the pressure nodes) are shown in a reference frame $(\tilde{x}, \tilde{y})$ moving with the cylinder. Note that the phase angle is different from the one of figure 5.16.

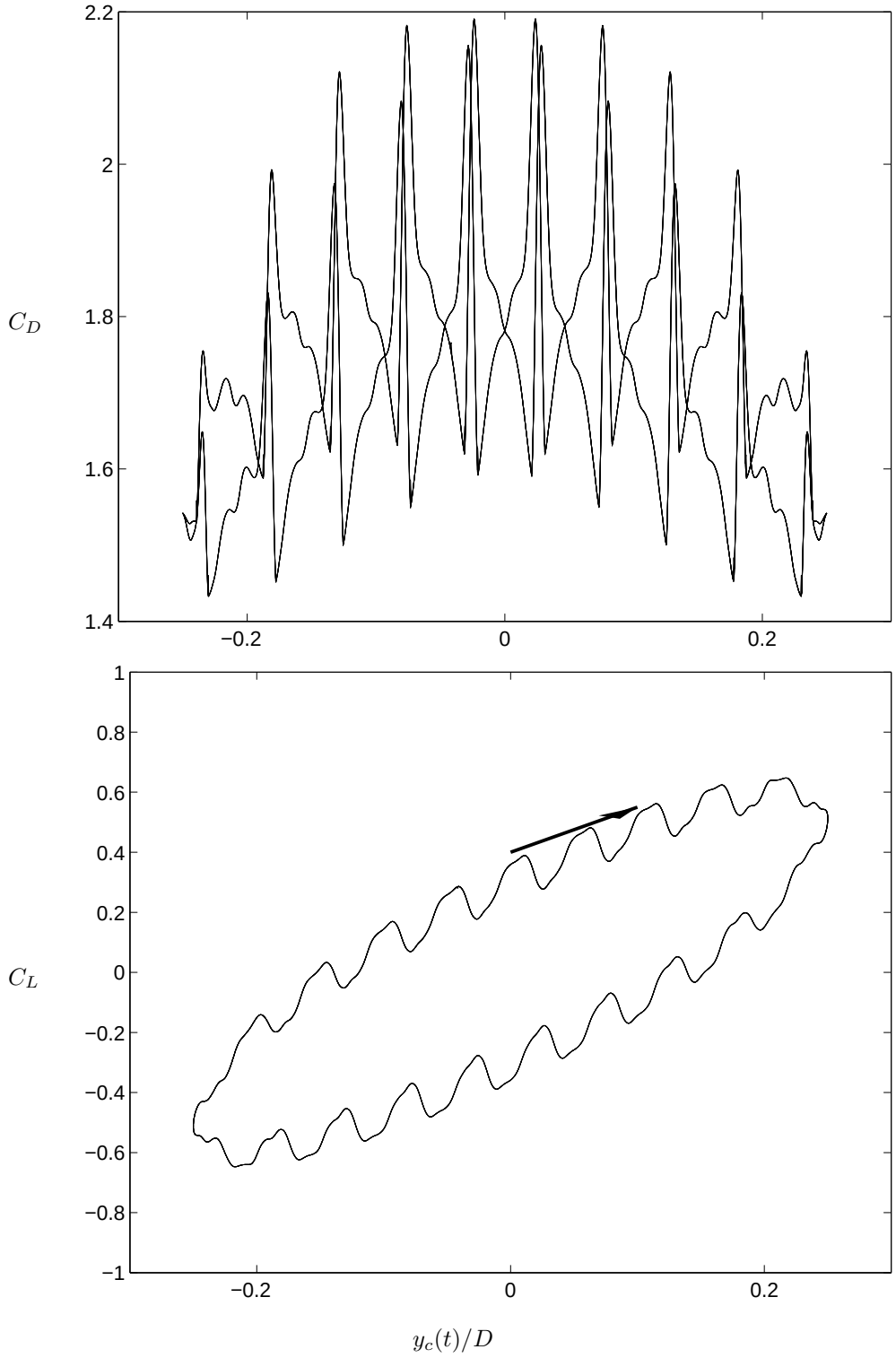

Figure 5.23: As figure 5.19, but using the divergence-free variant of the rigidity method.

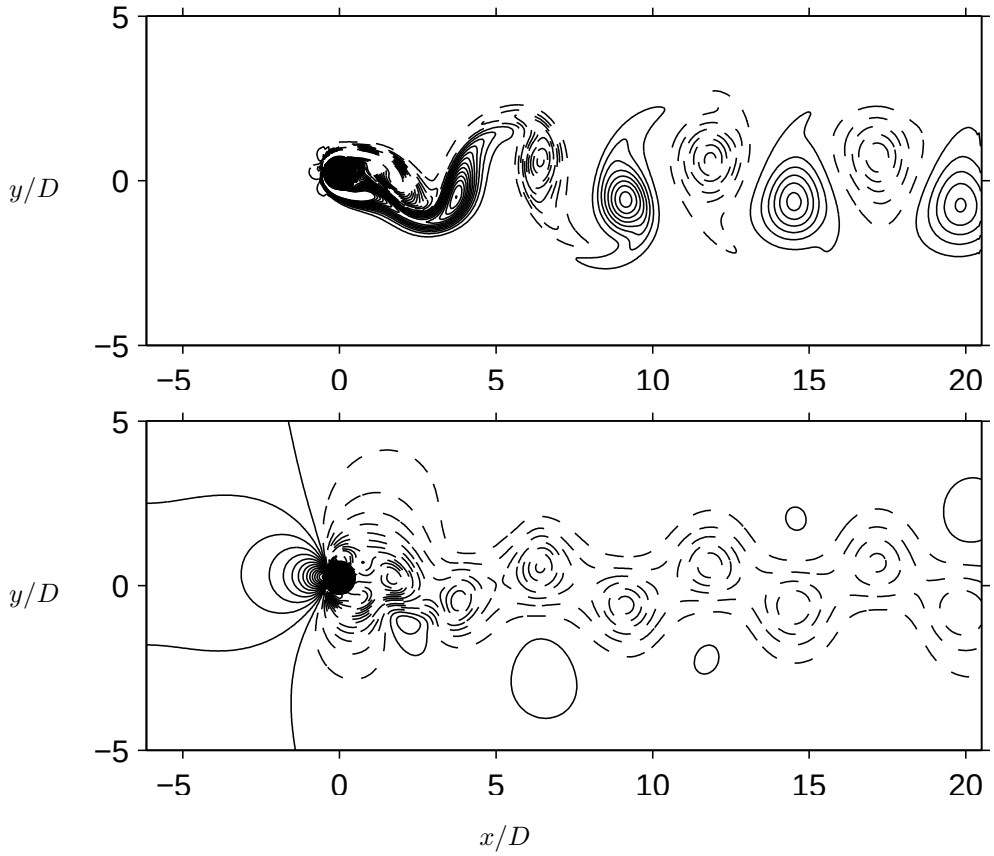

Figure 5.24: As figure 5.21, but using the divergence-free variant of the rigidity method.

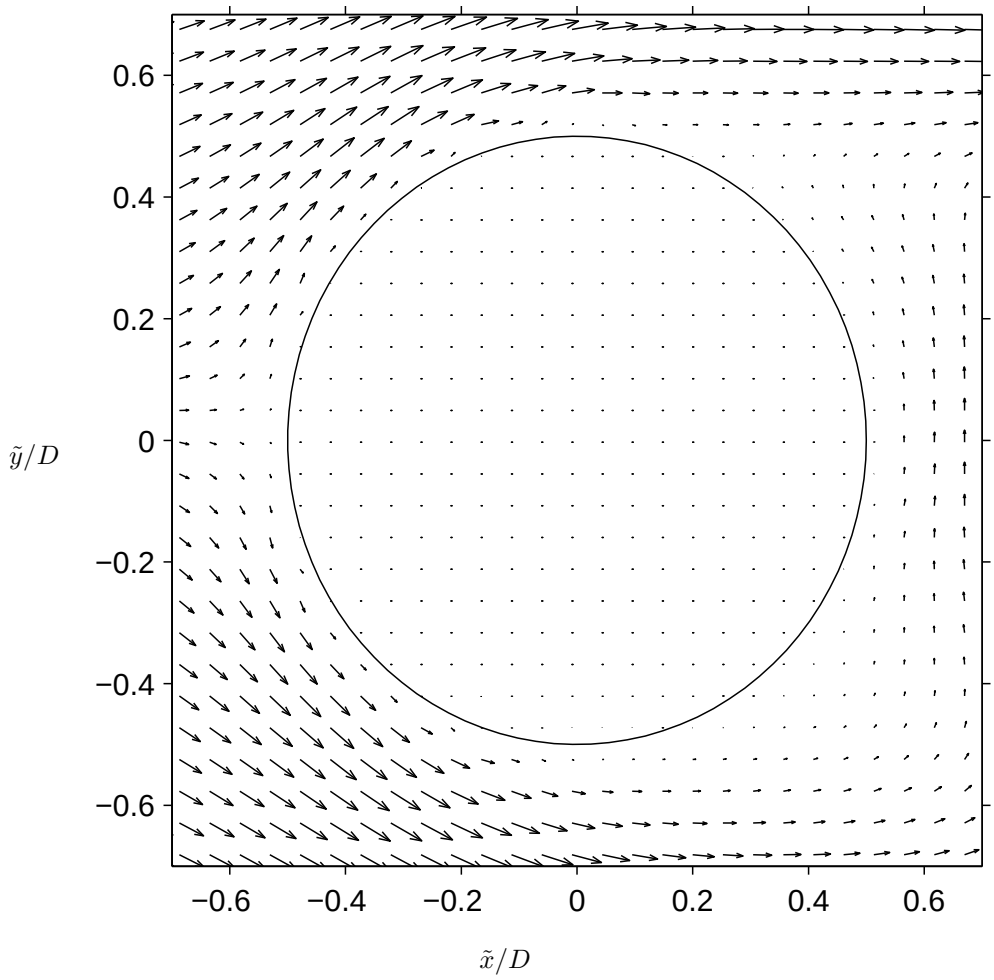

Figure 5.25: As figure 5.22, but using the divergence-free variant of the rigidity method.

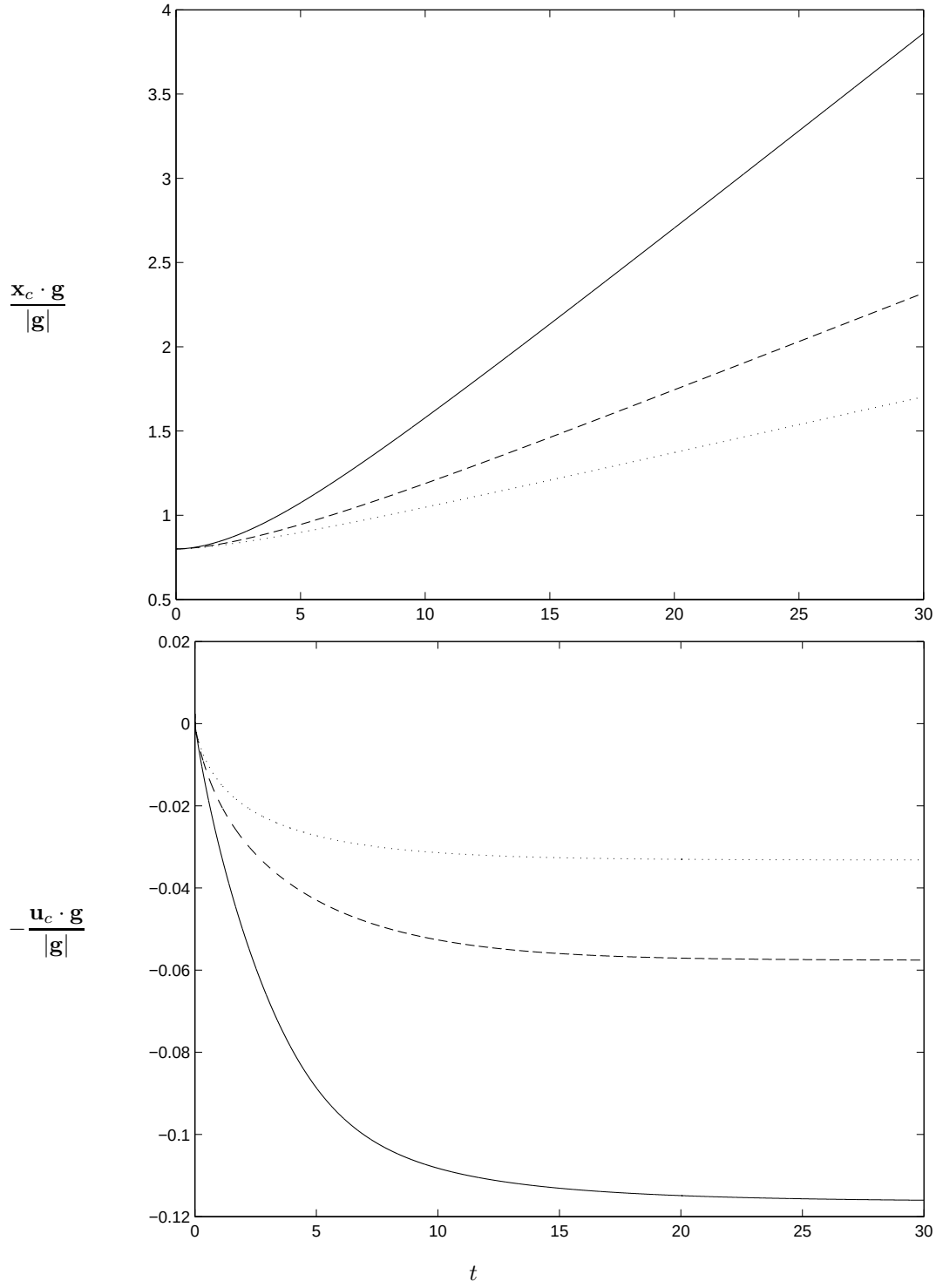

Figure 5.26: Temporal evolution of the center-position and the velocity for a single particle falling freely from rest at three different viscosities: —— $\nu = 10^{-3}$; - - - - $\nu = 5 \cdot 10^{-3}$ and ····· $\nu = 10^{-2}$.

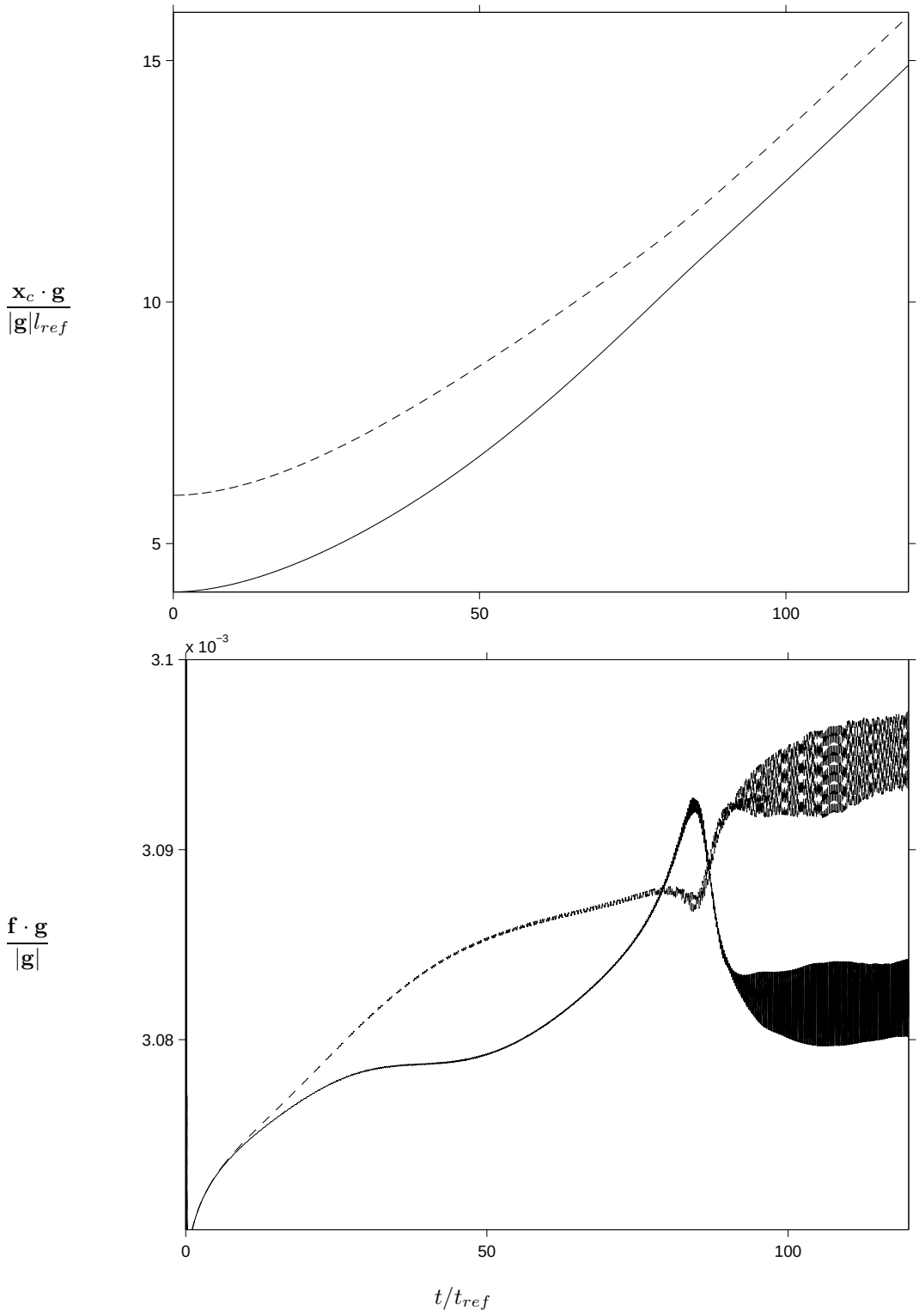

Figure 5.27: Temporal evolution of the center-position and the hydrodynamical force, both in the direction of the gravitation, for two particles falling freely from rest and initially trailing each other with a separation of D: —— leading; - - - - trailing.

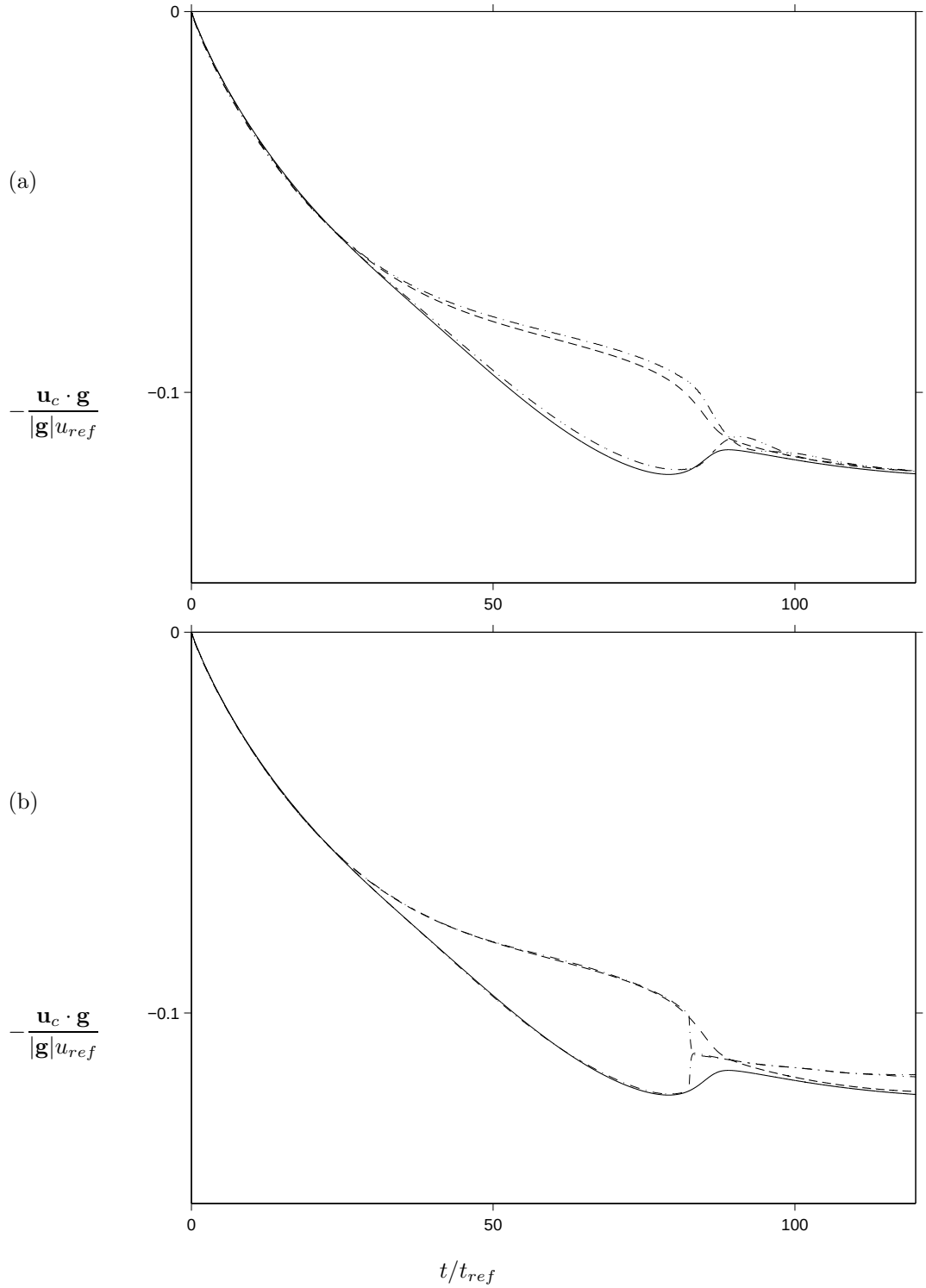

Figure 5.28: Temporal evolution of the center velocity in the direction of the gravitation, for two particles falling freely from rest and initially trailing each other with a separation of D: ——— leading; - - - - trailing. In plot (a), the chain-dotted lines correspond to numerical results from reference [9], in plot (b) to results from T.W. Pan (private communication).

Appendix A

Numerical details

A.1 Temporal accuracy

To our knowledge, the integration scheme (2.2) which is a combination of a fractional-step method, a semi-implicit integration and a Runge-Kutta scheme, first appeared in this form in reference [28]. Previously, Rai and Moin had given the present set of coefficients [13], however, in the framework of a pressure-less fractional step procedure similar to that of reference [29]. From Rai and Moin's work, it is therefore not clear which coefficient to use for the pressure term in a method which includes the latter in the predictor step. Reference [28] does not contain details regarding this point and, particularly, the consequences of choosing the value $2\alpha_k$. In the following we will analyze the truncation error of the present method in order to establish the overall time-accuracy of the integration scheme w.r.t. velocity and pressure. Related analyses of low-storage Runge-Kutta schemes have been performed by Wray [30] (for fully explicit discretizations) and Spalart, Moser and Rogers [31] (for semi-implicit and pressure-free methods).

Please note that for our numerical study we have chosen the present scheme rather than the method of [13] because the latter calls for an additional storage location due to the fact that the projection step involves the pseudo-pressure at two preceding time levels.

Turning to the analysis of the scheme (2.2), we first simplify the notation as follows:

$$\frac{\mathbf{u}^* - \mathbf{u}^{k-1}}{\Delta t} = +\gamma_k N(\mathbf{u}^{k-1}) + \zeta_k N(\mathbf{u}^{k-2}) - 2\alpha_k \nabla p^{k-1} + \alpha_k L\left(\mathbf{u}^* + \mathbf{u}^{k-1}\right) \tag{A.1a}$$

$$\mathbf{u}^k = \mathcal{P}\mathbf{u}^* \tag{A.1b}$$

$$p^k = p^{k-1} + \left(\frac{L_p}{2\alpha_k \Delta t} - \frac{\nu}{2}\nabla\cdot\right)\mathbf{u}^* \tag{A.1c}$$

where the spatial operators $N(\mathbf{u}) \equiv -(\mathbf{u}\cdot\nabla)\mathbf{u}$, $L_p \equiv (\nabla^2)^{-1}\nabla\cdot()$, $\mathcal{P} \equiv 1 - \nabla(\nabla^2)^{-1}\nabla\cdot()$ and $L \equiv \nu\nabla^2$ have been introduced. The three sub-steps in (A.1) are now successively substituted and the expressions for $u^{k=3}$ and $p^{k=3}$ are developed in Taylor series around time t^n. The resulting expansions can be simplified by reference to the following identities:

$$\begin{array}{llllll} L_p N(\mathbf{u}^n) = p^n, & L_p\mathbf{u}^n = 0, & \nabla\cdot\mathbf{u}^n = 0, \\ LL_p - \nu\nabla\cdot() = 0, & \nabla\cdot\nabla L_p = \nabla\cdot(), & L_p\nabla = 1, \\ \mathcal{P}\nabla\nabla\cdot() = 0, & \nabla\cdot\mathcal{P} = 0, & \mathcal{P}^l = \mathcal{P}, \\ (\nabla L_p)^l = \nabla L_p, & \mathcal{P}\nabla L_p = 0. \end{array} \tag{A.2}$$

On the other hand, the expansion of the exact velocity and pressure at time $t^{n+1} = t^n + \Delta t$ and

$t^{n+c} = t^n + c\Delta t$, respectively, read:

$$\begin{aligned}
\mathbf{u}(t^n + \Delta t) &= \mathbf{u}(t^n) + \Delta t \left\{ L\mathbf{u}(t^n) + \mathcal{P}N(\mathbf{u}(t^n)) \right\} \\
&\quad + \frac{\Delta t^2}{2} \left\{ \left(L + \mathcal{P}\frac{\partial N}{\partial \mathbf{u}} \right) \left(L\mathbf{u}(t^n) + \mathcal{P}N(\mathbf{u}(t^n)) \right) \right\} \\
&\quad + \frac{\Delta t^3}{6} \left\{ \left(L + \mathcal{P}\frac{\partial N}{\partial \mathbf{u}} \right)^2 \left(L\mathbf{u}(t^n) + \mathcal{P}N(\mathbf{u}(t^n)) \right) \right. \\
&\quad \left. + \mathcal{P}\frac{\partial^2 N}{\partial \mathbf{u}^2} \left(L\mathbf{u}(t^n) + \mathcal{P}N(\mathbf{u}(t^n)) \right)^2 \right\} \\
&\quad + \mathcal{O}(\Delta t^4)
\end{aligned} \tag{A.3}$$

$$\begin{aligned}
p(t^n + c\Delta t) &= p(t^n) + c\Delta t\, L_p \frac{\partial N}{\partial \mathbf{u}} \left\{ L\mathbf{u}(t^n) + \mathcal{P}N(\mathbf{u}(t^n)) \right\} \\
&\quad + (c\Delta t)^2\, L_p \left\{ \frac{\partial^2 N}{\partial \mathbf{u}^2} \left(L\mathbf{u}(t^n) + \mathcal{P}N(\mathbf{u}(t^n)) \right)^2 \right. \\
&\quad \left. + \frac{\partial N}{\partial \mathbf{u}} \left(L + \mathcal{P}\frac{\partial N}{\partial \mathbf{u}} \right) \left(L\mathbf{u}(t^n) + \mathcal{P}N(\mathbf{u}(t^n)) \right) \right\} \\
&\quad + \mathcal{O}(\Delta t^3)
\end{aligned} \tag{A.4}$$

Comparing the expansion of velocity (A.1b) with (A.3) leads to the following conditions in order for the $\mathcal{O}(\Delta t)$ term to match*:

$$\alpha_1 + \alpha_2 + \alpha_3 = \frac{1}{2}, \tag{A.5a}$$

$$\gamma_1 + \gamma_2 + \gamma_3 + \zeta_2 + \zeta_3 = 1\,, \tag{A.5b}$$

and for matching the $\mathcal{O}(\Delta t^2)$ term:

$$\gamma_1(\alpha_1 + 2\alpha_3 + 2\alpha_2) + \gamma_2(\alpha_2 + 2\alpha_3) + \gamma_3\alpha_3 + \zeta_2(\alpha_2 + 2\alpha_3) + \zeta_3\alpha_3 = \frac{1}{2} \tag{A.6a}$$

$$4\alpha_1(\alpha_2 + \alpha_3) + 4\alpha_2\alpha_3 + 2\alpha_1^2 + 2\alpha_2^2 + 2\alpha_3^2 = \frac{1}{2} \tag{A.6b}$$

$$2\alpha_1(\gamma_2 + \gamma_3 + \zeta_3) + 2\alpha_2\gamma_3 = \frac{1}{2} \tag{A.6c}$$

$$\gamma_1(\gamma_2 + \gamma_3 + \zeta_3) + \gamma_3(\gamma_2 + \zeta_2) = \frac{1}{2} \tag{A.6d}$$

Conditions which are consistent with (A.5) and (A.6) were presented in [31] and the values for γ_k and ζ_k† were indeed chosen identically. However, the scheme of [31] was not symmetric in the viscous term (i.e. different coefficients for the implicit and the explicit part). It can be easily verified that the present set of coefficients (2.3) verifies conditions (A.5) and (A.6) and therefore we have that $u^{k=3} - u^{n+1} = \mathcal{O}(\Delta t^3)$. Since one generally speaks of a method of order n for the approximation of a time derivative with the right-hand side being approximated with an error of $\mathcal{O}(\Delta t^n)$ [32, § 7.2.1], the present scheme is of second-order accuracy w.r.t. the velocity.

An additional feature of the present scheme is that it verifies $2\alpha_k = \gamma_k + \zeta_k$ which means that the partial time advancement of all sub-steps is the same for the viscous and the non-linear terms [31].

Comparing the expansion of pressure (A.1c) with (A.4) leads to the following condition in order for the $\mathcal{O}(1)$ term to match:

$$\gamma_3 + \zeta_3 = 2\alpha_3\,. \tag{A.7}$$

*Please note that it has been assumed *a priori* that $\zeta_1 = 0$ for the method to be self-starting.

†Also note that the index of ζ is shifted by one in reference [31].

For matching the $\mathcal{O}(\Delta t)$ term (with the time increment c as of yet undetermined), the following two conditions are found:

$$\gamma_3(\alpha_1+\alpha_2)+\zeta_3\alpha_1 = c\,\alpha_3 \tag{A.8a}$$
$$\gamma_3(\gamma_1+\gamma_2+\zeta_2)+\gamma_2\zeta_3 = c\,2\alpha_3 \tag{A.8b}$$

It is obvious that the above mentioned condition of equal time-increments $2\alpha_k = \gamma_k + \zeta_k$ automatically fulfills condition (A.7) and causes conditions (A.8) to reduce to one remaining condition. Inserting the set of present coefficients (2.3) leads to the value $c = 5/6$. Therefore, the pressure satisfies the consistency condition $p^{k=3} - p^{n+5/6} = \mathcal{O}(\Delta t^2)$. Since the equation for pressure $p = L_p N(u)$ does not involve a temporal derivative, this means that the scheme is of second order accuracy for the pressure.

A.2 Computation of the hydrodynamic forces acting upon an immersed body

Reference [22] discusses several ways to obtain the hydrodynamic forces acting upon a solid body immersed into a fluid. Clearly, one possiblity is to perform a momentum balance over a control volume fully enclosing the body (cf. § A.2.1). More desirably, however, is the evaluation of the forces by means of the specific volume forces already computed during the numerical integration. The (negative of the) result (cf. § A.2.2 and § A.2.3) is equivalent to the force upon the body in steady flow conditions; otherwise, the temporal rate of change term of the fluid within the solid domain needs to be evaluated as well (cf. § A.3).

The direct evaluation method, i.e. computing the integral over the stress tensor on the solid surface, was not considered here.

A.2.1 Momentum balance

Integrating the first component of the momentum equation (2.1) over a volume Ω_c gives (in tensor notation):

$$\partial_t \int_{\Omega_c} u_1 \,\mathrm{d}\mathbf{x} + \int_{\Omega_c} (u_1 u_j)_{,j} \,\mathrm{d}\mathbf{x} + \int_{\Omega_c} p_{,1} \,\mathrm{d}\mathbf{x} - \nu \int_{\Omega_c} (u_1)_{,jj} \,\mathrm{d}\mathbf{x} = \int_{\Omega_c} f_1 \,\mathrm{d}\mathbf{x} \quad , \tag{A.9}$$

and by using Green's theorem and its variants:

$$\partial_t \int_{\Omega_c} u_1 \mathrm{d}\,\mathbf{x} + \int_{\partial\Omega_c} u_1 \mathbf{u}\cdot\mathbf{n} \,\mathrm{d}\sigma + \int_{\partial\Omega_c} p n_1 \,\mathrm{d}\sigma - \nu \int_{\partial\Omega_c} u_{1,j} n_j \,\mathrm{d}\sigma = \int_{\Omega_c} f_1 \,\mathrm{d}\mathbf{x} \quad , \tag{A.10}$$

where $\mathbf{n} = (n_1, n_2)$ is the outward-pointing unit vector at the boundary $\partial\Omega_c$ of the control volume. The volume integral of the temporal term and the three line integrals need to be discretized in order to compute the integral of the body force term. The *a posteriori* temporal discretization is Euler implicit and the spatial operators are evaluated centrally, consistent with the present numerical scheme. Finally we have that $\rho_f \int_{\Omega_c} f_1 \,\mathrm{d}\mathbf{x} = F_D$ which means that the fluid density cancels in the expression for the drag coefficient (3.19). A similar procedure leads to the expression for the lift force.

A.2.2 Direct evaluation for direct forcing method

In this method the form of the body force term $\mathbf{f}$ is not known analytically but rather obtained at each time step numerically by imposing the no-slip/rigidity constraint at the interface between the two phases/inside the solid phase. This implies that the actual numerical value for this term used at each Runge-Kutta sub-step of the discrete scheme f_β^k corresponds to an unknown fraction of the full expressions. In other terms, instead of f_β^k one could have written $\xi_k f_\beta^k$ on the left-hand

side of equation (3.1) with ξ_k an undetermined coefficient. In any case, if we take up the notation of § A.1, add a term $\mathbf{f}^k$ to equation (A.1a) and successively substitute the three substeps, we obtain

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t} = \mathcal{P}\left(\alpha_1 L(\mathbf{u}^n) + \ldots + \gamma_1 N(\mathbf{u}^n) + \ldots + \zeta_1 N(\mathbf{u}^n) + \ldots + \mathbf{f}^1 + \mathbf{f}^2 + \mathbf{f}^3\right) \quad . \tag{A.11}$$

Therefore, the total body force added to the fluid over one full time step (between t^n and t^{n+1}) is equal to the sum of the volume force terms added at each Runge-Kutta substep, viz.

$$\mathbf{f}^{n+1} \approx \mathbf{f}^1 + \mathbf{f}^2 + \mathbf{f}^3 \quad , \tag{A.12}$$

except for the correction due to the projection step. Presently, we evaluate the volume integral of the body force by computing the sum of (A.12) at all forcing points of the solid body, viz.

$$\mathcal{F}^{n+1} = \int_{\Omega_{solid}} \mathbf{f}^{n+1}\,\mathrm{d}\mathbf{x} \approx \sum_{\in solid} \left(\mathbf{f}^1 + \mathbf{f}^2 + \mathbf{f}^3\right)\Delta x \Delta y \quad , \tag{A.13}$$

where the loose statement "$\in$ *solid*" means actually for each component β of the force that the node is part of the set of interface nodes, $\mathbf{x}_h^\beta \in \mathcal{B}$ (embedded boundary method), and that it is part of the set of solid nodes, $\mathbf{x}_h^\beta \in \mathcal{S}$ (rigidity method).

Finally, we have for the drag force that $F_D = -\rho_f \mathcal{F}_1^{n+1}$ in the case of steady flow.

A.2.3 Direct evaluation for immersed boundary method

Here, we are formally integrating an analytic expression and obtain for a full time step:

$$\begin{aligned}\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t} &= \mathcal{P}\left(\alpha_1 L(\mathbf{u}^n) + \ldots + \gamma_1 N(\mathbf{u}^n) + \ldots + \zeta_1 N(\mathbf{u}^n) + \ldots\right. \\ &\quad \left. + \gamma_1 \mathbf{f}^1 + \gamma_2 \mathbf{f}^2 + \gamma_3 \mathbf{f}^3 + \zeta_2 \mathbf{f}^1 + \zeta_3 \mathbf{f}^2\right) \quad .\end{aligned} \tag{A.14}$$

This means that the volume force added to the fluid during a full time step can be compute from

$$\mathbf{f}^{n+1} \approx \sum_{k=1}^{3} \left(\gamma_k \mathbf{f}^k + \zeta_k \mathbf{f}^{k-1}\right) \quad , \tag{A.15}$$

knowing that $\zeta_1 = 0$ and that $\sum_k \gamma_k + \zeta_k = 1$. Again, the correction due to the projection step (the symbolic operator $\mathcal{P}$) has been neglected. This leads to the following expression for the volume integral of the body force:

$$\mathcal{F}^{n+1} = \int_{\Omega_{solid}} \mathbf{f}^{n+1}\,\mathrm{d}\mathbf{x} = \int_0^{L_b} \mathbf{F}^{n+1}\,\mathrm{d}s \approx \sum_l \sum_{k=1}^{3} \left(\gamma_k \mathbf{F}^k(s_l) + \zeta_k \mathbf{F}^{k-1}(s_l)\right)\,\Delta s \quad . \tag{A.16}$$

A.2.4 Direct evaluation for rigidity method

Re-writing in (3.30) the step which imposes the rigidity constraint as a supplementary fractional step with a volume force

$$\frac{\mathbf{u}^k - \hat{\mathbf{u}}}{\Delta t} = 2\alpha_k \mathbf{f}^k \quad , \tag{A.17}$$

defined as

$$f_\beta^k \equiv \frac{\xi_\beta(t)\left(\tilde{u}_\beta^{k-1} - \hat{u}_\beta\right)}{2\alpha_k \Delta t} \quad , \tag{A.18}$$

the full scheme reads after successive substitution:

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t} = \mathcal{P}\left(\alpha_1 L(\mathbf{u}^n) + \ldots + \gamma_1 N(\mathbf{u}^n) + \ldots + \zeta_1 N(\mathbf{u}^n) + \ldots + 2\alpha_1 \mathbf{f}^1 + 2\alpha_2 \mathbf{f}^2 + 2\alpha_3 \mathbf{f}^3\right) \quad . \tag{A.19}$$

Therefore, we have for the volume integral over the body force:

$$\mathcal{F}^{n+1} = \int_{\Omega_{solid}} \mathbf{f}^{n+1}\,\mathrm{d}\mathbf{x} \approx \sum_{\in solid} \sum_{k=1}^{3} 2\alpha_k \mathbf{f}^k \Delta x \Delta y \quad . \tag{A.20}$$

A.3 Evaluation of the hydrodynamic stresses

Computation of the stress-related terms in (4.1) can be done either by directly evaluating their proper definition or by resorting to the equations of linear and angular momentum, respectively. Considering the control volume $\mathcal{S}$ which coincides with the particle, Cauchy's principle gives [33, p.100]:

$$\oint_{\partial\mathcal{S}} \boldsymbol{\tau}\cdot\mathbf{n}\,\mathrm{d}\sigma = -\int_{\mathcal{S}} \mathbf{f}\,\mathrm{d}V + \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}} \mathbf{u}\,\mathrm{d}V \tag{A.21a}$$

$$\oint_{\partial\mathcal{S}} (\mathbf{r}\times(\boldsymbol{\tau}\cdot\mathbf{n}))\,\mathrm{d}\sigma = -\int_{\mathcal{S}} (\mathbf{r}\times\mathbf{f})\,\mathrm{d}V + \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}} (\mathbf{r}\times\mathbf{u})\,\mathrm{d}V \tag{A.21b}$$

Since we can easily evaluate the first term on the right-hand-side of the equations of (A.21), we need to express the last term in order to obtain a value for the stress term on the left-and-side.

For convenience, let us split the velocity into a time-dependent translational component $\mathbf{u}_c(t)$ and a general remaining part, viz.

$$\mathbf{u}(\mathbf{x},t) = \mathbf{u}_c(t) + \mathbf{u}_a(\mathbf{x},t) \quad . \tag{A.22}$$

The no-slip condition at the interface of the two phases translates into the condition that $\mathbf{u} = \mathbf{u}_c + \boldsymbol{\omega}_c \times \mathbf{r}$. Therefore, we have that the velocity traversing this interface is equal to the translational velocity of the particle, i.e.

$$\mathbf{u}\cdot\mathbf{n} = \mathbf{u}_c\cdot\mathbf{n} \quad , \tag{A.23}$$

which implies that:

$$\mathbf{u}_a\cdot\mathbf{n} = (\mathbf{u}-\mathbf{u}_c)\cdot\mathbf{n} = 0 \quad , \tag{A.24}$$

or in other words: the velocity $\mathbf{u}_a$ in the moving system does not traverse the particle boundary.

A.3.1 Linear momentum

Noting that the control volume $\mathcal{S}$ constitutes a material element of fluid and making use of the incompressibility constraint, we can now write

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}} \mathbf{u}\,\mathrm{d}V &= \int_{\mathcal{S}} \frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t}\,\mathrm{d}V = \int_{\mathcal{S}} \left(\dot{\mathbf{u}}_c + \frac{\mathrm{D}\mathbf{u}_a}{\mathrm{D}t}\right)\mathrm{d}V \\
&= \pi r_c^2 \dot{\mathbf{u}}_c + \int_{\mathcal{S}} \partial_t \mathbf{u}_a\,\mathrm{d}V + \int_{\mathcal{S}} (\mathbf{u}\cdot\nabla)\,\mathbf{u}_a\,\mathrm{d}V \\
&= \pi r_c^2 \dot{\mathbf{u}}_c + \int_{\mathcal{S}} \partial_t \mathbf{u}_a\,\mathrm{d}V + \int_{\mathcal{S}} \nabla\cdot(\mathbf{u}\mathbf{u}_a)\,\mathrm{d}V \\
&= \pi r_c^2 \dot{\mathbf{u}}_c + \int_{\mathcal{S}} \partial_t \mathbf{u}_a\,\mathrm{d}V + \oint_{\partial\mathcal{S}} (\mathbf{u}\mathbf{u}_a)\cdot\mathbf{n}\,\mathrm{d}\sigma
\end{aligned} \tag{A.25}$$

Since $\mathbf{u}_a\cdot\mathbf{n} = 0$, the surface integral (last term in equation A.25) does not make any contribution to the rate of change of linear momentum.

Rigid body motion. In the case of rigid body motion *throughout* $\mathcal{S}$, the second term on the right-hand-side of (A.25) does not make any contribution and we obtain:

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_c + \boldsymbol{\omega}_c\times\mathbf{r}(\mathbf{x}) \quad \mathbf{x}\in\mathcal{S}: \qquad \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}} \mathbf{u}\,\mathrm{d}V = \pi r_c^2 \dot{\mathbf{u}}_c \quad . \tag{A.26}$$

Arbitrary motion. In the case of arbitrary fluid motion inside the domain $\mathcal{S}$, we can compute the volume integral in the following way. Let us introduce a streamfunction ψ_a which is related to the components of the velocity $\mathbf{u}_a$ by

$$u_a = \frac{\partial \psi_a}{\partial y}\,, \qquad v_a = -\frac{\partial \psi_a}{\partial x}\,. \tag{A.27}$$

Furthermore, let us work in polar coordinates (r, θ) where:

$$u_r = \frac{1}{r}\frac{\partial \psi_a}{\partial \theta}\,, \qquad u_\theta = -\frac{\partial \psi_a}{\partial r}\,, \tag{A.28}$$

and

$$u_a = u_r \cos(\theta) - u_\theta \sin(\theta)\,, \qquad v_a = u_r \sin(\theta) + u_\theta \cos(\theta)\,. \tag{A.29}$$

The (unknown) streamfunction can be developed in a Fourier series, viz.

$$\psi_a(r,\theta) = \psi_0(r) + \sum_{k=1}^{\infty} \left(\psi_k^s(r)\sin(k\theta) + \psi_k^c(r)\cos(k\theta)\right) \quad , \tag{A.30}$$

which leads to the following series for the radial and circumferential velocities:

$$u_r = \frac{1}{r}\sum_{k=1}^{\infty} k\left(\psi_k^s \cos(k\theta) - \psi_k^c \sin(k\theta)\right)\,, \tag{A.31}$$

$$u_\theta = -\psi_0' - \sum_{k=1}^{\infty} \left(\psi_k^{s\prime} \sin(k\theta) + \psi_k^{c\prime} \cos(k\theta)\right)\,. \tag{A.32}$$

Now, evaluating the volume integral for the first velocity component gives

$$\begin{aligned}
\int_{\mathcal{S}} u_a \,\mathrm{d}V &= \int_0^{2\pi}\int_0^{r_c} \left(u_r\cos(\theta) - u_\theta \sin(\theta)\right) r \,\mathrm{d}r\mathrm{d}\theta \\
&= \int_0^{2\pi}\int_0^{r_c} \cos(\theta)\left(\sum_{k=1}^{\infty} k\left(\psi_k^s(r)\cos(k\theta) - \psi_k^c(r)\sin(k\theta)\right)\right)\mathrm{d}r\mathrm{d}\theta \\
&\quad + \int_0^{2\pi}\int_0^{r_c} \sin(\theta)\left(\psi_0'(r) + \sum_{k=1}^{\infty}\left(\psi_k^{s\prime}(r)\sin(k\theta) + \psi_k^{c\prime}(r)\cos(k\theta)\right)\right) r\,\mathrm{d}r\mathrm{d}\theta
\end{aligned}$$

.

Using the following identities

$$\int_0^{2\pi} \sin(\theta)\sin(k\theta)\,\mathrm{d}\theta = \begin{cases} \pi & k = 1 \\ 0 & \text{else} \end{cases} \quad , \tag{A.33}$$

$$\int_0^{2\pi} \cos(\theta)\cos(k\theta)\,\mathrm{d}\theta = \begin{cases} \pi & k = 1 \\ 0 & \text{else} \end{cases} \quad , \tag{A.34}$$

$$\int_0^{2\pi} \sin(l\theta)\cos(k\theta)\,\mathrm{d}\theta = 0 \quad , \tag{A.35}$$

we obtain:

$$\int_{\mathcal{S}} u_a \,\mathrm{d}V = \pi \int_0^{r_c} \psi_1^s(r)\,\mathrm{d}r + \pi \int_0^{r_c} \psi_1^{s\prime}(r) r\,\mathrm{d}r \quad . \tag{A.36}$$

Integration by parts of the second integral on the right-hand-side leads to:

$$\int_{\mathcal{S}} u_a \,\mathrm{d}V = \pi r_c \psi_1^s(r_c) \quad . \tag{A.37}$$

In the case that the velocity is known at the boundary of the domain $\partial\mathcal{S}$, i.e. at $r = r_c$, we can evaluate the streamfunction and its expansion coefficients. In the case of a rigid body movement, i.e. superposition of a translation and a rigid rotation, the streamfunction reads:

$$\psi(r,\theta) = u_c r\sin(\theta) - v_c r\cos(\theta) - \frac{r^2}{2}\omega_c \quad , \tag{A.38}$$

which means that the contribution corresponding to the velocity $\mathbf{u}_a$ at the interface is simply:

$$\psi_a(r_c,\theta) = -\frac{r_c^2}{2}\omega_c \quad , \tag{A.39}$$

Comparison of (A.38) with the expansion (A.30) shows that:

$$\psi_0(r_c) = -\frac{r_c^2}{2}\omega_c\,, \quad \psi_k^s(r_c) = \psi_k^c(r_c) = 0 \quad \forall\, k \geq 1 \tag{A.40}$$

Therefore, the contribution of equation (A.37) vanishes. An analogous result is obtained for the second component of velocity such that we can write:

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_c + \boldsymbol{\omega}_c \times \mathbf{r}(\mathbf{x}) \quad \mathbf{x} \in \partial\mathcal{S}: \qquad \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}} \mathbf{u}\,\mathrm{d}V = \pi r_c^2 \dot{\mathbf{u}}_c \quad , \tag{A.41}$$

which is identical to equation (A.26) derived for the case of rigid body motion within *all* of the domain $\mathcal{S}$. This result means that the volume integral of a divergence-free velocity field over a circular domain, verifying rigid body movement on the boundary, does not depend on the specific form of the motion inside the domain.

A.3.2 Angular momentum

Rigid body motion. Concerning the angular momentum, we obtain analogously for the time derivative term—in the case of rigid body movement throughout $\mathcal{S}$:

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_c + \boldsymbol{\omega}_c \times \mathbf{r}(\mathbf{x}) \quad \mathbf{x} \in \mathcal{S}: \qquad \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}} \mathbf{r}\times\mathbf{u}\,\mathrm{d}V = \frac{1}{2}\pi r_c^4 \dot{\boldsymbol{\omega}}_c \quad . \tag{A.42}$$

Arbitrary motion. In the general case, we can proceed as in (A.27)–(A.41), expanding the streamfunction in a Fourier series along the circumferential direction and obtain:

$$\int_{\mathcal{S}} \mathbf{r}\times\mathbf{u}_a\,\mathrm{d}V = -2\pi\int_0^{r_c} r^2\psi_0'(r)\,\mathrm{d}r = -2\pi r_c^2\psi_0(r_c) + 4\pi\int_0^{r_c} r\psi_0(r)\,\mathrm{d}r \quad . \tag{A.43}$$

Substituting the coefficients of the streamfunction for the case of rigid-body motion imposed at the interface $\partial\mathcal{S}$, (A.40), we get:

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_c + \boldsymbol{\omega}_c \times \mathbf{r}(\mathbf{x}) \quad \mathbf{x} \in \partial\mathcal{S}: \qquad \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}} \mathbf{r}\times\mathbf{u}\,\mathrm{d}V = \pi r_c^4\dot{\omega}_c + 4\pi\partial_t\int_0^{r_c} r\psi_0(r)\,\mathrm{d}r \quad . \tag{A.44}$$

Thus, the rate of change of angular momentum of the fluid occupying the particle domain is not in general equal to its counterpart in pure rigid body motion.

General axismmetric motion. [‡] If the motion of fluid inside the particle domain $\mathcal{S}$ consists of a general axisymmetric motion $\mathbf{u}_a = \mathbf{u}_a(r,t)$ (superposed with the translational velocity $\mathbf{u}_c$), we can use the fact that the vorticity transport equation reduces to

$$\partial_t\omega = \frac{\nu}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\omega}{\partial r}\right) \quad , \tag{A.45}$$

[‡]This possibility was suggested by F. Higuera.

and evaluate the integral in (A.43) by substituting the vorticity for the streamfunction, using

$$\omega = \frac{1}{r}\frac{\partial(ru_\theta)}{\partial r} = -\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \psi_a}{\partial r}\right) \tag{A.46}$$

and performing successive partial integrations. This leads to:

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}_c + \boldsymbol{\omega}(r,t)\times\mathbf{r}(\mathbf{x}) \quad \mathbf{x}\in\mathcal{S}: \qquad \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{S}} \mathbf{r}\times\mathbf{u}\,\mathrm{d}V = \pi r_c^4\dot{\omega}_c - \pi\nu\left(r_c^3\omega'(r_c) + 2r_c^2\omega_c\right) \quad . \tag{A.47}$$

The radial derivative of the vorticity $\omega'(r_c)$, however, is not known analytically since we only enforce a Dirichlet value there.

A.3.3 Recap of the options

Therefore, we are left with the following alternatives for evaluating the hydrodynamic stress-term acting upon a submerged particle:

1. Evaluate the left-hand-side of (A.21) directly, i.e. by its definition and numerically. Drawback: this task involves interpolation on the surface of the particle and is sensitive to discretization errors.

2. Evaluate the additional force contribution from $\mathbf{f}$ numerically; evaluate the rate of change of the angular momentum numerically; use the analytic expression for the rate of change of the linear momentum (A.41). Remark: the force contribution is readily available during the numerical algorithm; the rate of change of the volume integral can be evaluated with less numerical difficulty than a surface integral.

3. As an approximation, neglect the deviation of the internal motion from an axisymmetric (plus translational) movement and use the analytical expression for the time-derivative of the angular momentum according to (A.47), evaluating, however, the radial derivative of vorticity numerically. Remark: it is difficult to justify the hypothesis of an axisymmetric movement inside the particle domain since the action of the volume force is *a priori* arbitrary and depends upon the flow outside the particle; in particular, dipole-type flow patterns are observed often in practivce.

4. As an approximation, neglect the deviation of the internal motion from a rigid-body movement and use the analytical expression for the time-derivative of the angular momentum according to (A.42). I.e. on the right-hand-side of (A.21) only evaluate the contribution from $\mathbf{f}$ numerically. Remark: this option is definitely the cheapest one from the computational point of view and has been used in reference [6] without further justification.[§] In the framework of the rigidity method of § 3.3 this option is valid.

In lack of a justification for the approximations 3. and 4. we currently use method 2. for evaluating the hydrodynamic forces.

[§]Note that in the above mentioned reference, forcing was only applied near the surface of the particle in practice for reasons of computational efficiency. Therefore, their case is comparable to the present one although the methods are different in theory.