INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
*Int. J. Numer. Meth. Fluids* 2004; **45**:775–795 (DOI: 10.1002/fld.710)

# A virtual boundary method with improved computational efficiency using a multi-grid method

Johan Revstedt*,†

*Div. of Fluid Mechanics, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden*

SUMMARY

The flow around spherical, solid objects is considered. The boundary conditions on the solid boundaries have been applied by replacing the boundary with a surface force distribution on the surface, such that the required boundary conditions are satisfied. The velocity on the boundary is determined by extrapolation from the flow field. The source terms are determined iteratively, as part of the solution. They are then averaged and are smoothed out to nearby computational grid points. A multi-grid scheme has been used to enhance the computational efficiency of the solution of the force equations. The method has been evaluated for flow around both moving and stationary spherical objects at very low and intermediate Reynolds numbers. The results shows a second order accuracy of the method both at creeping flow and at $Re = 100$. The multi-grid scheme is shown to enhance the convergence rate up to a factor 10 as compared to single grid approach. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: virtual boundary method; Cartesian grid; multi-grid; sphere; moving boundaries

## 1. INTRODUCTION

In almost all practical CFD applications one needs to account for solid boundaries with complex geometries. Considering stationary boundaries there are several approaches for dealing with this. For example, one may use structured body-fitted grids or unstructured grids to be able to describe the required boundary geometry. The main advantage of such approaches is that the boundary will be well defined and specifying boundary conditions is relatively straight forward. However, generalized co-ordinates, in a solver using finite differences, require that the co-ordinate transform matrices (Jacobians) have to be stored or be recalculated. In addition to this requirement, by co-ordinate transformation additional terms appear in the governing equations. These terms imply a considerable increase in the number of computational operations. Furthermore, non-uniformities in the grid cause a slow-down in the convergence rate of

*Correspondence to: Johan Revstedt, Div. of Fluid Mechanics, Lund Institute of Technology, P.O. Box 118, S-221 00 Lund, Sweden.
†E-mail: jorev@ms.vok.lth.se

Contract/grant sponsor: Swedish National Energy Administration (STEM); contract/grant number: 12126-1c

Published online 19 April 2004

*Received 29 March 2003*
*Revised 22 January 2004*

many iterative solvers. Also, the grid generation has to be carried out with great care since degenerate computational cells can, at least locally, dramatically increase the numerical errors. Furthermore, considering moving boundaries one will either have to regenerate the grid as the boundary is moved, which will lead to increased computational effort or one will have to make use of overlapping, moving grids resulting in loss of accuracy in the required interpolations. Using a finite volume formulation one will avoid the problems associated with co-ordinate transformation. However, in such solvers the order of accuracy of the discretization cannot without a considerable amount of work be raised beyond two. Using uniform Cartesian grids, a major advantage is that implementing higher order discretization schemes is quite straight forward. Also, the solver can be made very efficient and additional errors due to poor grid quality are avoided. However, treatment of the boundaries will be more complex, i.e. one will have to employ special methods for specifying boundary conditions so that a high order of accuracy is retained. As a low order approximation one may use what is often referred to as blocking, i.e. computational cells that totally or partially filled with solid will have pre-set values of all variables and are excluded from the solution process. Although, computationally efficient this method will lead to a first order (piece-wise linear) approximation of the surface. Improving this approach one might consider higher order extrapolation of boundary conditions from the flow region to the blocked cells [1]. However, this method has some potential drawbacks related to mass conservation and insufficient grid resolution. Alternatively, one may use a method based on the Volume of Fluid concept, denoted Volume of Solid (VOS) [2]. Hence, one of the fluids is replaced by a solid by using an infinite viscosity for that phase. Close to the object surface the fluid viscosity is modified based on the local volume fraction of solid in the cells cut by the object surface.

Methods for describing complex moving non-deformable boundaries can be divided into three major groups depending on the grid structure used. One may consider using a combination of moving and stationary grids which are overlapping, i.e. the surrounding is described on a stationary grid and a moving grid is attached to the object. The main advantage of these methods is that boundary conditions on the object can be set easily. A major disadvantage is associated with numerical problems in the information transport between the grids, i.e. the interpolation procedure implies reduced computational efficiency and potentially also accuracy of the numerical scheme as compared to a single (stationary) grid approach. Also, this approach would be unsuited when considering deformable objects.

Using the arbitrary Lagrangian Eulerian (ALE) method, e.g. References [3, 4] one needs only one grid which deforms with the object movement. Again, a major advantage of this kind of method is the possibility to easily set the surface boundary conditions. However, as the object is moving the grid will deform, which might lead to large errors (at least locally) if care is not taken in the grid generation.

Replacing the boundaries by momentum sources, also referred to as ‘fictitious domain’ methods or ‘virtual (or immersed) boundary’ methods, can be advantageous when describing both stationary and moving boundaries on Cartesian grids. Since the ‘boundary’ only exists as a part of the numerical solution to the equations one needs not to regenerate the grid as the objects is moved. Also, the forces acting on the object will be directly available, hence studies of fluid–structure interaction is more straight forward since no extra computations are required to calculate the displacement of the object. However, there the computational time is increased due to the solution of the extra boundary equations. The method considered here, i.e. modelling of the boundaries by adding momentum sources, originates from the early works

 

by Peskin [5] who applied the basic approach for studying the flow of blood past an elastic (initially 2D) heart-valve leaflet. Recently, a second order accurate immersed boundary method was presented by Lai and Peskin [6]. Other major contributions to the field have been made by Zaleski [7, 8], Trygvasson [9], Glowinski [10–13] and Shyy [14, 15]. The main differences among the different approaches lies in the implementation of the technique rather than on the basic principles. Thus, the basic approach is generally the same but methods for determining the magnitude of the sources differ. The idea is to determine the effect an object on the fluid by adding source terms to the momentum equations. In the method by Glowinski *et al.* [10] the discontinuity is considered directly and related to the pressure gradient. Goldstein *et al.* [16] developed a method based on the same principles as are used in control system theory for a two mode controller, i.e. the calculation of the source terms are based on an integrated part and a part directly proportional to the velocity:

$$F_i(x_i, t) = \alpha \int_0^t u_i(x_i, \tau)\, \mathrm{d}\tau + \beta u_i(x_i, t) \tag{1}$$

where $\alpha$ and $\beta$ are constants. This approach was used by Saiki and Biringen [17] for studying flows around cylinders. Their results correlate well with experimental data both for stationary and oscillating cylinders. Fadlun *et al.* [18] compared this method to the more direct method by Modh-Yusof [19] which has the advantage that no constants has to be set *a priori*. Instead the magnitude of the force is based directly on the momentum balance in the flow, i.e.

$$\frac{u_i^{l+1} - u_i^l}{\Delta t} = \mathrm{RHS}^{l+1/2} + F_i^{l+1/2} \tag{2}$$

$$F_i^{l+1/2} = -\mathrm{RHS}^{l+1/2} + \frac{V_i^{l+1} - u_i^l}{\Delta t} \tag{3}$$

where RHS contains the convective, pressure and viscous terms and $V_i$ is the velocity of the boundary. The approach used in this work is similar to the one by Goldstein *et al.* [16] concerning how the source term is determined (Equation (1)) however there are differences in how the forces are distributed from the surface to the surrounding fluid. The application of virtual boundary methods is mainly in two areas, detailed studies of two-phase flows, e.g. particle suspensions [13, 20], and for modelling of moving solid boundaries in practical applications such as internal combustion engines [18] and stirred tanks [21].

The goal of the present work is to improve the virtual boundary method (VBM) of Revstedt and Fuchs [22] both in terms of accuracy and computational efficiency. Alternative methods have been compared for all the components of the model. The improved method has been applied to flow around solid spherical objects Here, we consider both stationary and moving three-dimensional objects at low as well as higher Reynolds number. First, creeping flow is considered, both for a stationary sphere in a free stream and a sphere affected by gravity in a initially still fluid. Thereafter higher Reynolds numbers are considered both for stationary and oscillating spheres. Together these simulations should give a good overview of the performance and capabilities of the method under different flow situations. The results have been compared to competing approaches as well as data (both analytical and experimental) from literature.

 

## 2. MATHEMATICAL AND NUMERICAL FORMULATION

### 2.1. Governing equations

The equations governing isothermal, incompressible flow of a Newtonian fluid can be written as

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{4}$$

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu \frac{\partial}{\partial x_j}\frac{\partial u_i}{\partial x_j} + \Phi_i \tag{5}$$

where $\Phi_i$ is a source term.

The system of Equations (4)–(5) is well posed if the $d$-conditions are imposed on all the boundaries, where $d$ is the dimension of the problem. In the case of solid walls, one applies usually the no-slip condition which implies that the local fluid velocity equals to the velocity of the boundary at the same point. For non-solid object boundaries, the stresses has to be in balance. This type of boundary conditions can be given implicitly and hence has to be determined as part of the solution. At inflow and outflow, the velocity vector and its gradient, respectively, are often assumed. The same number of conditions that has to be imposed even when other expressions are used. Other types of conditions, such as for a boundary with surface tension, could be applied equally.

The body forces $\Phi_i$, in our case vanish normally. However, we shall replace some of the boundaries by a force distribution on the surface of the boundary. In this case the body $\Phi_i$ are computed on the boundaries so as to satisfy the local boundary conditions. Thus, $\Phi_i$ do not vanish only on certain surfaces. Consider a boundary in the form of a closed surface $\Gamma$ with the parameterization of the surface given by $X_i(s_i, t)$, then the force in the flow field can be written as

$$\Phi_i(x_j, t) = \iint_\Gamma F_i(s_j, t)\delta(x_j - X_j)\,\mathrm{d}s_1\,\mathrm{d}s_2 \tag{6}$$

where $F_i$ is the force on the surface and $\delta$ is the three-dimensional Dirac delta function. This singularity in the flow field has of course impact on the numerics involved in solving the equations as will be discussed below.

### 2.2. Flow solver

The incompressible Navier–Stokes are discretized on a system of locally refined Cartesian grids (e.g. Reference [23]). The dependent variables are defined on a staggered grid. This arrangement has the advantage that the system requires three boundary conditions (e.g. the velocity vector) on all boundary points. The different terms of the momentum and continuity equations are approximated by finite-differences. Basically, one may use finite-differences of any order. However, for higher order (i.e. more than two) additional boundary conditions has to be specified. Here, we use upwind finite-differences of first- or third-order accuracy. The lower order scheme (first order for the convective terms and second order for the others) implies that the low order terms dominate, leading to a high level of numerical dissipation.

 

Using directly higher order (third and fourth order approximations, respectively), leads to a less robust solver with considerably slower convergence rate. To combine numerical efficiency with higher order accuracy, we introduce the higher order terms as a 'single-step' defect correction [24]. One can show that for smooth problems this procedure is adequate to maintain the theoretical accuracy of the high order scheme.

The time integration is done by a three-level implicit scheme. In each time step, the system of equations is solved iteratively using a multi-grid solver. The relaxation scheme within the multi-grid solver comprises of point wise relaxation of the momentum equations coupled with a point wise smoothing of the continuity equation. At the latter step, both the velocity vector and the pressure are corrected so that the residuals of the momentum equations shall not be changed as the continuity equation is satisfied. This approach is equivalent to an approximate diagonalization of the system of equations [23, 24].

### 2.3. *Boundary conditions*

The basic idea behind this boundary method is that the forces exerted by the flow on the object should equal the forces exerted by the object on the flow. The method is based on the work by Revstedt and Fuchs [22] and includes the following steps:

(1) *Discretization of the surface.* A 2D grid is generated on the object surface such that the mesh size is somewhat smaller than that of the computational grid.
(2) *Calculation of the boundary velocity.* The velocity in each node on the surface mesh is determined from the velocities of the surrounding nodes of the computational grid.
(3) *Determining the boundary forces.* The forces on the boundary are calculated based on the local velocity defect, i.e. the difference between the velocity obtained in step 2 and the boundary condition.
(4) *Distribution of the force field.* The resulting forces are distributed back to the computational grid and will appear as source terms in the momentum equations.

The velocity on the boundary can be determined by averaging or interpolation. Here, three-dimensional Lagrangian polynomials based on the interpolation formula by Ericsson and Fuchs [25] are used.

$$u_i^{\mathrm{f}} = \sum_{r=1}^{n+1}\sum_{s=1}^{n+1}\sum_{t=1}^{n+1}\left(\prod_{k=1_{k\neq r}}^{n+1}\frac{\xi-\xi_k}{\xi_r-\xi_k}\right)\left(\prod_{k=1_{m\neq s}}^{n+1}\frac{\eta-\eta_m}{\eta_s-\eta_m}\right)\left(\prod_{k=1_{p\neq t}}^{n+1}\frac{\zeta-\zeta_p}{\zeta_t-\zeta_p}\right)u_{i_{rst}} \tag{7}$$

These polynomials can be either centred around the boundary (interpolation) or end at the boundary (extrapolation), where the latter approach should be the more reasonable. The reason for this is twofold. It is not certain that a higher order interpolation gives higher order accuracy of the result. This is due to the fact that when the solution is converged one should, because of the no-slip condition, have a discontinuous velocity profile over the surface. However, since the finite differences are taken over the boundary this is in reality not the case. Also, from a physical point of view it is more reasonable to use the velocities in the nodes outside the surface of the object. Although the equations are solved in the whole domain the flow inside the solid object has no physical meaning and hence should not affect the surface velocity directly. Likewise, if the boundary is a fluid interface one would extrapolate to the boundary from both sides. Both interpolation and extrapolation using third and fifth order polynomials have been used here.

 

The surface force contributions are calculated based on the difference between the calculated boundary velocity, $u_i^{\mathrm{f}}$, and the boundary condition, $u_i^{\mathrm{b}}$, according to

$$\Delta F_i = \frac{\alpha}{h^2}(u_i^{\mathrm{f}} - u_i^{\mathrm{b}}) = \alpha \frac{\delta u_i}{h^2} \tag{8}$$

$$F_i^{n+1} = F_i^n + \Delta F_i^{n+1} \tag{9}$$

where $\alpha$ is a relaxation parameter and $n$ denotes iteration number. Hence, close to the object surface it is assumed that viscous effects dominate the flow, which is true provided the grid resolution is sufficient.

As was stated above, the boundary represents a discontinuity in the flow and the force exerted can accordingly be described by a Dirac function (Equation (6)). Of course, this cannot be represented directly on the computational grid and instead a distribution function has to be used to transfer the forces back to the computational grid. For this the approach from Revstedt and Fuchs [22] is kept, i.e. a Gaussian distribution function is used. However, one may also consider other distribution functions, such as the discrete delta function by Lai and Peskin [6]. Using the Gaussian approximation, the source terms in the momentum equations (5) can then be written as

$$\Phi_i = \frac{1}{N} \sum_{k=1}^{N} G_F F_i \tag{10}$$

where $N$ is the number of contributions from the surface grid to a certain grid point and $G_F$ is a Gaussian distribution function

$$G_F = \frac{1}{(\sigma\sqrt{2\pi})^3} \mathrm{e}^{-(\xi^2+\eta^2+\zeta^2)/2\sigma^2} \tag{11}$$

where the standard deviation, $\sigma$, is of the order of the computational grid size. As for the boundary velocity, the distribution of source terms may be performed on either one or both sides of the boundary. Again, both methods have been applied. However, using the same reasoning as for the velocity, the force should only distributed to the interior of the body. Thereby not directly affecting the flow field of interest.

As was noted by Revstedt and Fuchs [22], the convergence rate using this method is rather poor. The obvious reason is of course that instead of having only one iterative process (that of solving the flow field) we also have to iteratively solve for the shape of the boundary. Hence, the flow situation changes as the object is formed, i.e. we 'waste' some iterations in the flow field initially to solve a problem that is quite far from the final one. It can also be explained as, that this force method, as it is converging, introduces high frequency error components with large amplitude to the flow field. The smoothing process requires that for elliptic problems as the ones considered in this work the information from the boundary is transported to the whole domain. This data transfer rate is rather low if only the finest grid is used. However, a multi-grid scheme can easily rectify this.

The multi-grid implementation used here has the following steps. After a given grid is relaxed and the corresponding source terms are determined according to steps 1–4 above, a coarse grid problem for the boundary force is defined. The boundary 'equation' on the given

 

grid (superscript $k$) is defined as

$$\Delta F_i^k = R_i^k \tag{12}$$

$R_i^k$ will vanish on all grids when the boundary conditions are satisfied. The 'error' on the finest grid ($M$) is

$$R_i^M = \alpha \frac{u_i^{\mathrm{f}} - u_i^{\mathrm{b}}}{h^2} \tag{13}$$

The 'boundary equation' on the coarser grid ($k$) is defined through the 'error' in the finer grid $R_i^{k+1}$:

$$R_i^k = I_k^{k+1} R_i^{k+1} + \alpha \frac{u_i^{\mathrm{f}} - u_i^{\mathrm{b}}}{h^2} \tag{14}$$

where the velocities are taken on the coarse grid. An initial value for the coarse grid force is given by

$$\tilde{F}_i^k = I_k^{k+1} \tilde{F}_i^{k+1} \tag{15}$$

where the prolongation operator $I_k^{k+1}$ is a linear interpolation.

In the full multi-grid calculations, one has to satisfy Equation (12) during the relaxation process. Once, the coarsest grid has been fully solved, the correction to the body force (i.e. the change in $\tilde{F}_i^k$ between before and after relaxations) is interpolated and added to the existing force on the surface node points. The main advantage of this scheme is that the corrections of the boundary force are distributed to the whole field at high rate on the coarse grids and hence leading to an overall faster convergence. It should also be noted that the force MG-method need not be applied to all the grid levels of the flow solver as will be demonstrated below. It might even be undesirable from an accuracy point of view to calculate and distribute source terms on computational grids where the grid spacing is larger than the object size. For grid levels excluded from the force MG and ordinary restriction operator is used to transfer the distributed source terms to coarser grid, i.e.

$$\Phi_i^{k-1} = I_k^{k-1} \Phi_i^k \tag{16}$$

here the restriction operator is an ordinary top-hat filtering. This approach is also used in the single grid simulations.

## 3. COMPUTATIONAL SET-UP

The method has been applied to the test case of flow around a solid spherical object. First, creeping flow is considered, both for a stationary sphere in a free stream and a sphere affected by gravity in a initially still fluid. Thereafter higher Reynolds numbers, $Re = 100, 300$ and 1000, are considered both for stationary and oscillating spheres. Since the wake of a sphere becomes unsteady at about $Re = 280$ (e.g. Reference [26]) temporally resolved simulations are required for the $Re = 300$ and 1000 cases. Together these simulations should give a good overview of the performance and capabilities of the method under different flow situations.

 

Table I. Computed cases for Stokes flow.

| Case | Boundary velocity | Force distribution | Multi-grid | Grid resolutions (cells per unit length) |
|---|---|---|---|---|
| 3EXT | 3rd order extrapolation | 1-sided | No | 8, 16, 32, 64 |
| 3EXTMG | 3rd order extrapolation | 1-sided | Yes | 8, 16, 32, 64 |
| 5INT1s | 5th order interpolation | 1-sided | No | 8, 16, 32 |
| 5INT2s | 5th order interpolation | 2-sided | No | 8, 16, 32 |
| 3INT | 3rd order interpolation | 1-sided | No | 8, 16, 32 |
| BLK | Blocking | — | — | 8, 16, 32, 64 |
| EXB | Exact boundary conditions | — | — | 8, 16, 32, 64 |

Table II. Computed cases for moderate Reynolds numbers.

| Case | Reynolds number | Boundary velocity | Force distribution | Multi-grid | Grid resolutions | Note |
|---|---|---|---|---|---|---|
| 3EXT | 100 | 3rd order extrapolation | 1-sided | No | 8, 16, 32 | |
| 3EXTMG | 100 | 3rd order extrapolation | 1-sided | Yes | 8, 16, 32 | |
| 5INT1s | 100 | 5th order interpolation | 1-sided | No | 8, 16, 32 | |
| 5INT2s | 100 | 5th order interpolation | 2-sided | No | 8, 16, 32 | |
| 5INT1s | 300 | 5th order interpolation | 1-sided | No | 32 | |
| 5INT2sMG | 300 | 5th order interpolation | 2-sided | Yes | 32 | $St = 0$, 0.3, 3.0 |
| 5INT2s | 300 | 5th order interpolation | 2-sided | No | 32 | $St = 0$, 0.3, 3.0 |
| 5INT2s | 1000 | 5th order interpolation | 2-sided | No | 32 | |

### 3.1. *Creeping flow*

The computational domain used for the stationary sphere is a cubic box with the side length $16D_{sp}$, centred in this box is a spherical object of diameter $D_{sp}$. The exact solution, derived by Stokes [27] is set as boundary condition on all outer boundaries. In the reference cases, where blocking of cells is used, either zero velocity or the exact solution is used on the surface of the sphere, depending on the accuracy required. A succession of local grid refinements are used around the object. The grid resolution has been varied from 16 to 64 computational cells per unit length resulting in about 3 000 000 cells in the best resolved cases. The simulated cases are summarised in Table I. For the falling sphere the size of the computational domain was $8D \times 8D \times 16D$. Again with local refinements around the sphere, giving a resolution of 32 cells per unit length. The time step was set using a CFL number of 0.1 based on a reference velocity of unity.

### 3.2. *Moderate Reynolds numbers*

For $Re = 100$, and 300 the computational domain is a rectangular box of size $12D \times 12D \times 24D$ with the sphere placed in the centre. For $Re = 1000$ the domain is elongated to $36D$ with the sphere still in the same position. In these cases local refinements are also used. The finest with 32 cells per unit length. The computed cases and grid configurations are presented in Table II. On the inflow boundary a constant velocity is set, on the outflow a ordinary Neumann

 

condition adjusted to ensure global mass conservation. On all other boundaries slip conditions are used. The temporal resolution was also in this case set using $N_{CFL} = 0.2$.

## 4. RESULTS

### 4.1. Creeping flow

Consider creeping fluid motion ($Re \ll 1$) around a solid spherical object. For this particular flow case an exact solution was first derived by Stokes [27]. Using the spherical polar co-ordinate system the velocity components can be written as

$$u_r = U \cos\theta \left(1 + \frac{a^3}{2r^3} - \frac{3a}{2r}\right) \tag{17}$$

$$u_\theta = U \sin\theta \left(-1 + \frac{a^3}{4r^3} + \frac{3a}{4r}\right) \tag{18}$$

where $a$ is the sphere radius and $U$ is the free stream velocity.

Figure 1 shows the error relative to the Stokes' solution for creeping flow as a function of grid resolution. The results from using blocking and from applying the exact solution on the boundary have been added as references. The former is a first order method and the latter should represent the best solution one can obtain with this computational code. From Figure 1 together with Table III, showing the overall order of accuracy calculated over the three coarsest grids, three main results can be extracted. Using interpolation to determine the surface velocity gives smaller errors than using extrapolation from the external flow field, however the order of accuracy is somewhat higher in the extrapolation case. Also, the order of the interpolation polynomial seems to influence the result only marginally. Using the multi-grid algorithm also seems to have very limited effect the order of accuracy, as would be expected. Concerning the force distribution, there are distinct differences in the levels of the errors between using single- and double-sided distribution from the surface. The double-sided distribution yields a significantly larger error and also lower order of accuracy (see Table III). This can be explained by studying Figure 2(a) showing profiles of velocity in the main flow direction at the stagnation points and the shoulders of the sphere. The velocity in the single-sided distribution case follows the exact solution closely, whereas in the double-sided distribution case there is a fairly large deviation. The reason for this behaviour is that the source terms in the single-sided case only affects the flow field inside the body leaving the exterior or actual flow field to be affected only through the boundary condition. As can be seen in the close up in Figure 2(b) the effects of the double-sided distribution are visible far beyond the distribution range. However, the distance of influence will probably be substantially decreased in more advection dominated flows.

In the previous model by Revstedt and Fuchs [22] the convergence rate was quite poor. However, using the multi-grid method proposed above one can substantially increase the convergence rate. Figure 3 shows the error in boundary velocity as a function of work units. A work unit is here defined as the amount of computational work needed for one iteration on the finest grid of the flow solver, i.e. on the next coarser grid one iteration corresponds to $\frac{1}{8}$

 

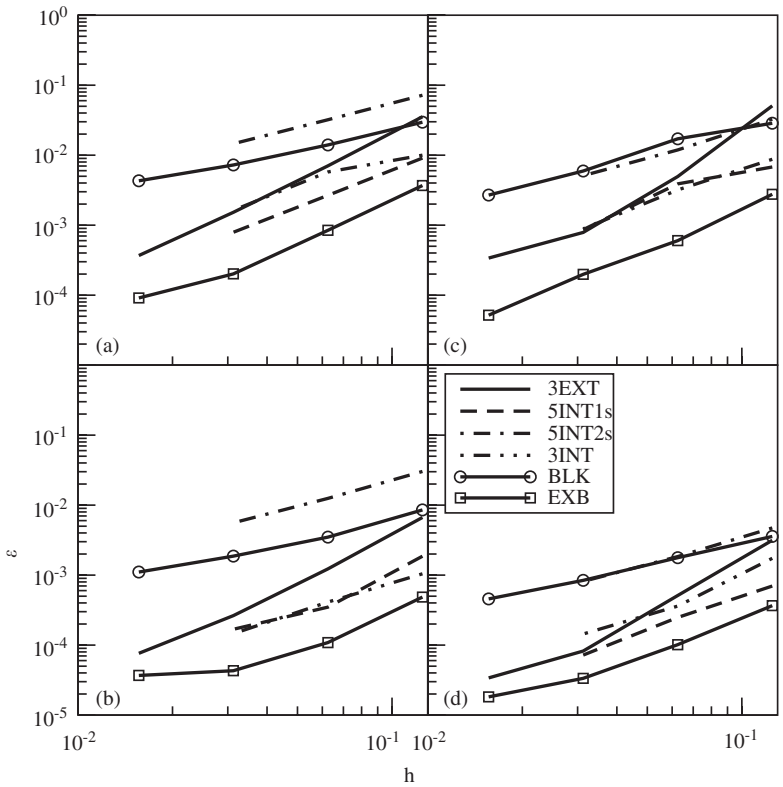


Figure 1. Maximum and $L_2$ norms of velocity errors. (a) $L_2$ norm $x$ direction, (b) max norm $x$ direction, (c) $L_2$ norm $y$ direction, (d) max norm $y$ direction.

Table III. Order of accuracy of the Stokes flow solution in the stream wise ($W$) and cross stream ($U$) directions considering the maximum and $L_2$ norms.

| Case | $W_{\max}$ | $W_{L_2}$ | $U_{\max}$ | $U_{L_2}$ |
|---|---|---|---|---|
| 3rd order extrapolation | 2.3 | 2.3 | 3.0 | 2.6 |
| 3rd order extrapolation with multi-grid | 1.9 | 2.0 | 2.0 | 2.1 |
| 5th order interpolation | 1.8 | 1.7 | 1.5 | 1.6 |
| 5th order interpolation with 2-sided distribution | 1.1 | 1.2 | 1.3 | 1.3 |
| 3rd order interpolation | 1.8 | 1.7 | 1.4 | 1.3 |
| Blocking | 1.0 | 1.1 | 1.0 | 1.0 |
| Exact boundary conditions | 2.1 | 1.7 | 1.9 | 1.7 |

work unit since the grid resolution is changed a factor 2. The grid structure of the flow solver (5 multi-grid levels) is the same in all cases but the number of grid levels on the surface has been varied from 1 to 5. As can be seen there is a continuous increase in convergence rate with increasing number of multi-grid levels. As a comparison we also used a scheme

 

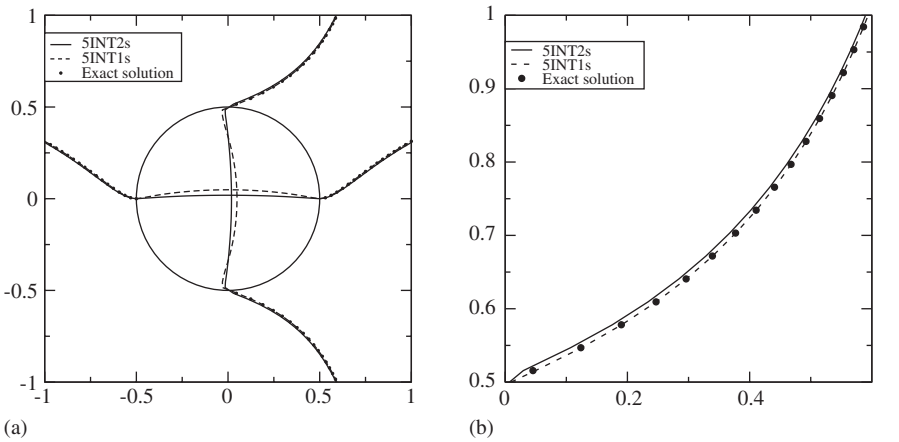


Figure 2. Velocity in the main flow direction at four locations (a) and close up of the upper part (b).

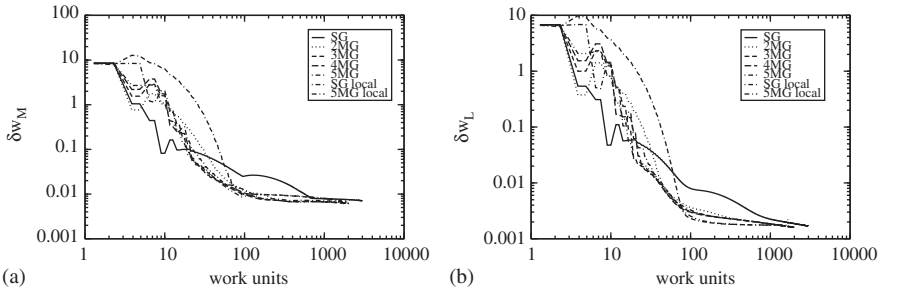


Figure 3. Error in boundary velocity as a function of work units, maximum norm (a) and $L_2$ norm (b).

Table IV. CPU-time relative to the single grid case for the same number of work units.

| Case | Relative CPU-time |
|---|---|
| Single grid | 1 |
| 2 MG-levels | 1.16 |
| 3 MG-levels | 1.19 |
| 4 MG-levels | 1.20 |
| 5 MG-levels | 1.21 |
| Single grid + local | 2.85 |
| 5 MG-levels + local | 3.10 |

where additional relaxations are performed in the cells affected by the source terms. This leads also to an increase in convergence rate comparable to using 5 MG-levels. However, 'local' relaxations will, as they are implemented here, also increase the computational time substantially. Considering the CPU-time required relative to the single grid case (Table IV), the multi-grid case will require about 20% more computational effort for the same number of work units. Whereas the local relaxations increase the required CPU-time by almost 300%.

 

Although, it should be noted that the local relaxations method has not been optimised here, only used as is.

Using this method one is able to obtain the force exerted by the flow on the object by integrating the distribute source terms over the volume of interest as was shown by Revstedt and Fuchs [22]. This can be used for determining the boundary conditions for a freely moving object. It is here demonstrated for a spherical object affected by gravity in a still fluid at low Reynolds number (approx. 0.0001) and high density ratio ($\delta\rho = 7000$). Considering Newton's second law of motion one can write the force balance for such an object as

$$m\frac{\mathrm{d}U_i}{\mathrm{d}t} = \mathscr{F}_i \tag{19}$$

where $m$ is the mass of the object, $U_i$ is the object velocity and $\mathscr{F}_i$ is the sum of all the forces acting on the body. The right-hand side of Equation (19) is determined by integration of the source terms over the volume $\Omega$

$$\mathscr{F}_i = \int_\Omega \Phi_i \,\mathrm{d}\Omega \tag{20}$$

which should equal the acceleration times the mass of the body. One is then able to set the boundary condition for the next time step as

$$U_i = \frac{1}{m}(\mathscr{F}_i \Delta t) = \frac{\mathrm{d}U_i}{\mathrm{d}t}\Delta t \tag{21}$$

Since, in this case, the Reynolds number is very low (Stokes flow) and the density ratio is very high one can neglect all contributions to the right hand side of Equation (19) except drag and buoyancy. Hence,

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} = \frac{18U_i^2}{Re_p\,\delta\rho} - (1-\delta\rho)g_i \tag{22}$$

which of course has the solution

$$U_i(t) = \frac{Re\, g_i}{18}\left(\frac{1}{\delta\rho} - 1\right) e^{-18\delta\rho/Re\, t} \tag{23}$$

$$U_i(0) = 0 \tag{24}$$

Figure 4(a) shows the object velocity as a function of time for the 5INT2s and 5INT1s approaches. After a relatively short acceleration phase, the object reaches the terminal velocity. The discrepancy in terminal velocity between the simulations and the analytical solution can be seen in Figure 4(b). The difference in error between 5INT2s and 5INT1s is quite large and the larger error in terminal velocity for 5INT2s is probably due to the same effect as was shown for a stationary sphere in Figure 2.

### *4.2. Moderate Reynolds numbers*

To see the influence of higher Reynolds number and to investigate the behaviour of the method in time dependent flows, simulations have been made for stationary spheres at $Re = 100, 300,$ and 1000. The three test cases considered here have been chosen to cover the flow regimes

 

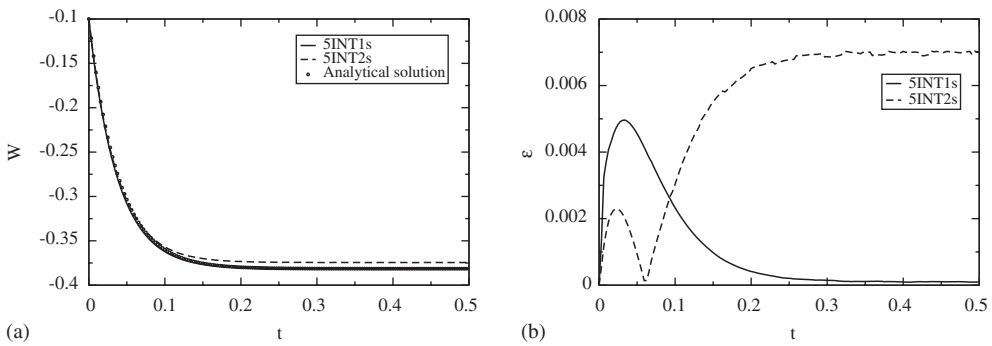


Figure 4. Velocity (a) and error compared to the analytical solution (b) of a spherical object in free fall.

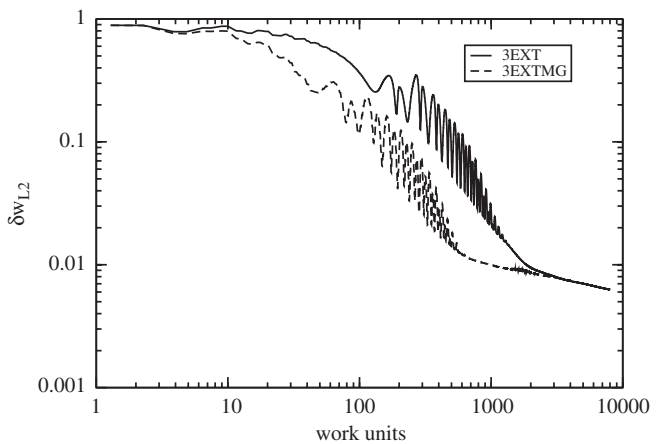


Figure 5. $L_2$-norm of velocity error at $Re = 100$ for single and multi-grid cases.

from stationary symmetric flows to time dependent vortex shedding flows. At $Re = 100$ the flow is still stationary and axisymmetric. As the Reynolds number is increased a number of instabilities appear leading to an unsteady flow, this process is described by Tomboulides and Orszag [28]. Johnson and Patel [26] reports that the flow becomes unsteady at a Reynolds number of 280, so that at $Re = 300$ one has a time dependent flow field. At $Re = 1000$ the flow is more chaotic and several distinct frequencies are present [28].

First consider the flow at $Re = 100$. Due to the lack of an exact solution the order of accuracy has been estimated by considering the $L_1$-norm of the velocity,

$$\|u\|_1 = \frac{1}{N}\sum_{n=1}^{N}(\sqrt{u_i u_i})_n \tag{25}$$

 

Table V. Order of accuracy ($p$) as defined by Equation (26) at $Re = 100$.

| Case | $p$ |
|---|---|
| 3rd order extrapolation | 2.0 |
| 3rd order extrapolation with multi-grid | 1.7 |
| 5th order interpolation | 1.9 |
| 5th order interpolation with 2-sided distribution | 1.1 |

Hence, evaluating the flow using three simulations with the mesh spacing halved between each simulation one may calculate the order of accuracy from

$$p = \frac{\ln(\|u_h - u_{h/2}\|_1 / \|u_{h/2} - u_{h/4}\|)_1}{\ln 2} \tag{26}$$

For the case $Re = 100$ the same methods for determining boundary velocity and distributing forces as for Stokes' flow have been used. The results from the accuracy study using Equation (26) are shown in Table V. As can be seen, the second order behaviour of the extrapolation method is retained at this Reynolds number. Again a slight decrease in the order of accuracy is seen when using the multi-grid approach. Also, comparing the fifth order interpolation cases one can see the influence of the force distribution. Again, we observe the same behaviour as for Stokes' flow, i.e. two-sided force distributions lower the order of accuracy from 2 to 1. Concerning computational efficiency one can see that the multi-grid method gives a substantially increased convergence rate although the speed-up is not as large as for Stokes' flow (Figure 5).

As mentioned above, at $Re = 300$ the flow is time dependent, which will somewhat alter the requirements of the VBM since the boundary velocity will have to be converged in each time step, i.e. in relatively few work units. Since, for $Re = 100$ there was no difference between the extrapolation and the interpolation cases only the 5INT1s and 5INT2s approaches are used for $Re = 300$. The time averaged flow fields are depicted in Figure 6, the flow is at this Reynolds number not axisymmetric (not even in average) although there is a plane of symmetry, as was noted by Johnson and Patel [26]. Hence, there will be a lift force with a well defined direction acting on the sphere. This can clearly be seen in Figure 6, showing the averaged velocity fields in planes parallel and perpendicular to the lift force. As can be seen from Figure 7, the 5INT1s fail to predict the drag coefficient accurately, and if run longer the computation will diverge. This might be caused by the higher order accuracy in combination with insufficient resolution of the boundary layer. Further increasing the grid resolution was unfortunately not possible due to too long simulation times. However, the 5INT2s gives a stable solution. The 5INT2s case has been compared to data available in the literature and in Figure 8(a) are shown the drag and lift coefficients compared to the numerical results obtained by Johnson and Patel [26]. The agreement is good although the present results over-predict the drag coefficient somewhat. Figure 8(b) shows the spectral density of the forces, the main frequency peak is observed at approximately $St = 0.13$ and a smaller peak at the first harmonic ($St \approx 0.26$). This has been compared with the data from simulations by Johnson and Patel [26], Tomboulides and Orszag [28] and Ploumhans *et al.* [29]. As can be seen in Table VI, the frequency is a bit under-predicted in the present study, however comparing the time averaged values of $C_D$ and $C_L$ as well as the amplitudes thereof, the agreement is very good.

 

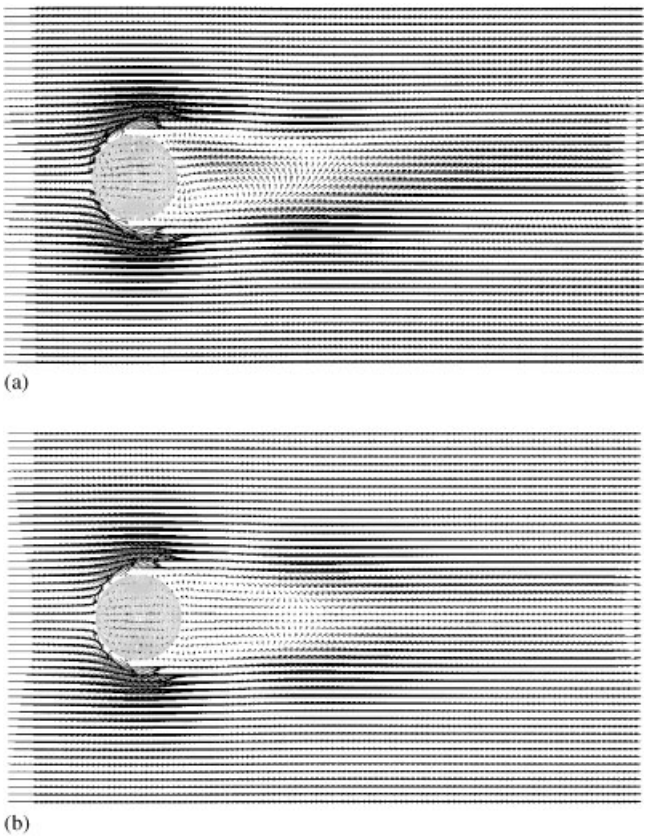


Figure 6. Average velocity fields at $Re = 300$ in the planes parallel (a) and normal to the lift force (b).

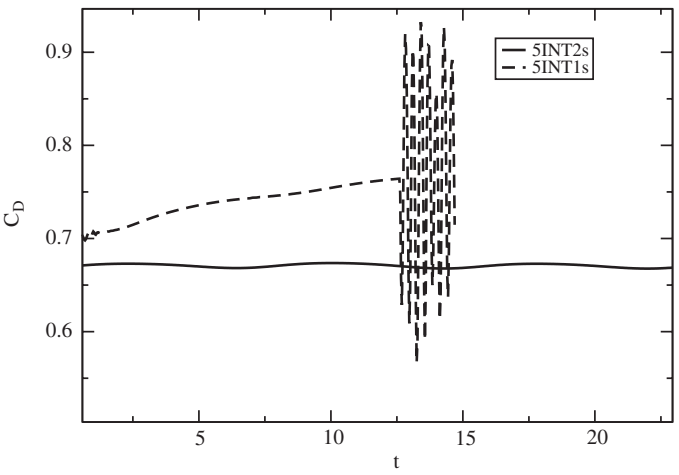


Figure 7. Time history of drag coefficient at $Re = 300$ for first and second order methods.

 

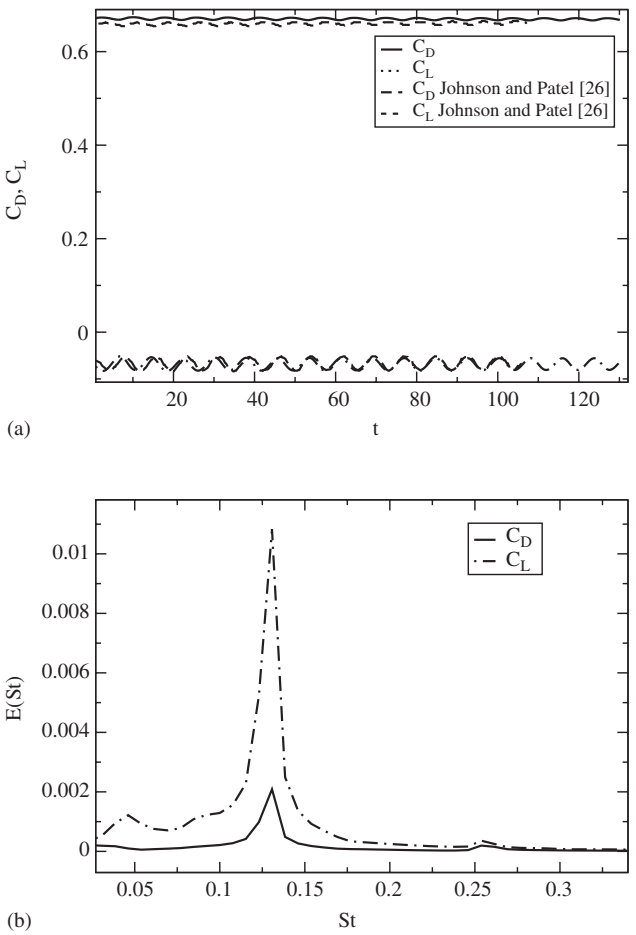


Figure 8. Time history (a) and spectral content (b) of the lift and drag coefficients at $Re = 300$.

Considering the convergence rate one might expect the multi-grid to be even more useful since low errors in the boundary conditions are required in each time step. However, as is seen in Figure 9(a), the multi-grid method is only effective in the start up sequence of the simulation, i.e. as the spherical object is 'created'. Thereafter the variations in conditions between to time steps is so small that the errors in boundary velocity can be kept at a low level even in the single grid simulation. In order to study the method under conditions with higher temporal variations, two cases with a sphere oscillating in the cross-stream directions were considered. The amplitude was in both cases $0.2D$ and the Strouhal numbers were 0.3 and 3.0, respectively. The results of this can be seen in Figure 9(b) and 9(c), showing the convergence history in one representative time step. For the lower Strouhal number the effects of the multi-grid is only marginal but as the Strouhal number is increased the importance of it becomes much greater.

At $Re = 1000$ again only the double sided distribution is used. As can be seen in Figure 10(a) the is more chaotic, with more frequencies present, than for $Re = 300$. However,

 

Table VI. Force coefficients and shedding frequency for $Re = 300$.

| | $St$ | $C_D$ | $\Delta C_D$ | $C_L$ | $\Delta C_L$ |
|---|---|---|---|---|---|
| Present study | 0.131 | 0.67 | $2.4 \times 10^{-3}$ | $-0.068$ | $1.2 \times 10^{-2}$ |
| Johnson and Patel [26] | 0.137 | 0.656 | $3.5 \times 10^{-3}$ | $-0.069$ | $1.6 \times 10^{-2}$ |
| Tomboulides and Orszag [28] | 0.136 | 0.671 | $2.8 \times 10^{-3}$ | — | — |
| Ploumhans *et al.* [29] | 0.135 | 0.683 | $2.5 \times 10^{-3}$ | $-0.061$ | $1.4 \times 10^{-2}$ |

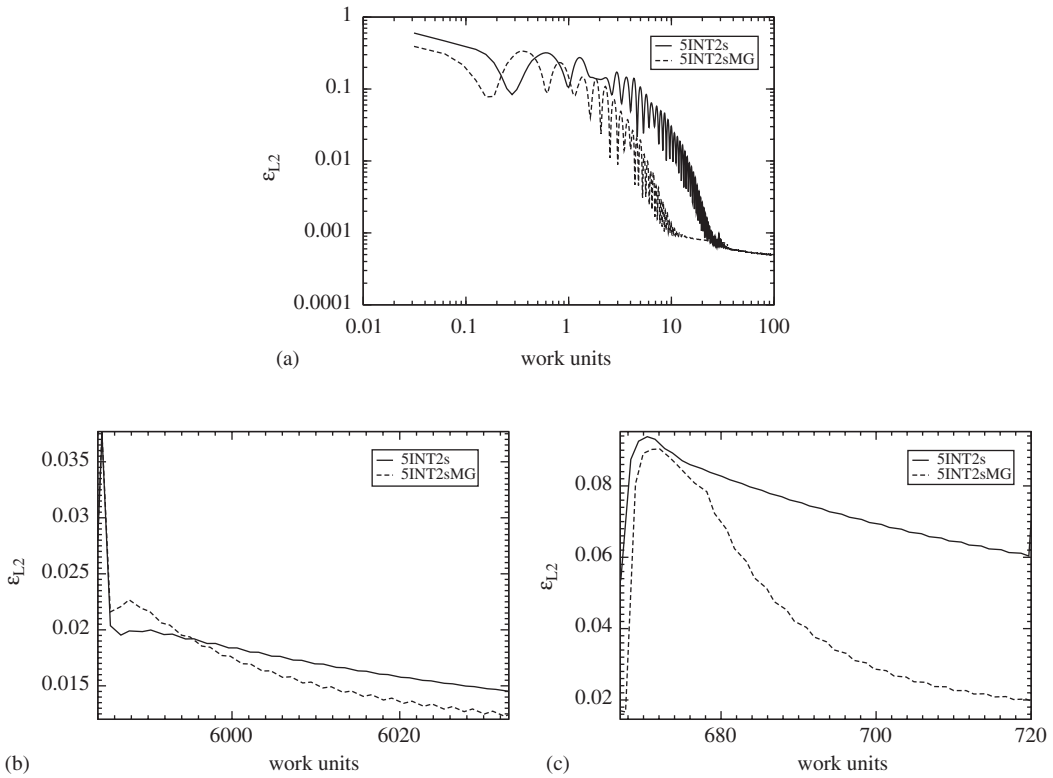


Figure 9. $L_2$-norm of velocity error at $Re = 300$ and $St = 0$(a), 0.3(b), 3.0(c).

the time averaged flow field is axisymmetric (Figure 10(b)). In Figure 11 the time average and fluctuation of velocity in the main flow direction is plotted and compared to the experimental data of Wu and Faeth [30] and data from the simulations by Tomboulides and Orszag [28]. As can be seen, the present method captures well the strength and length of the recirculation zone and also the fluctuations are in good agreement with previous data. The discrepancies found further down stream in the wake might be due to insufficient grid resolution in that region of the computational domain.

 

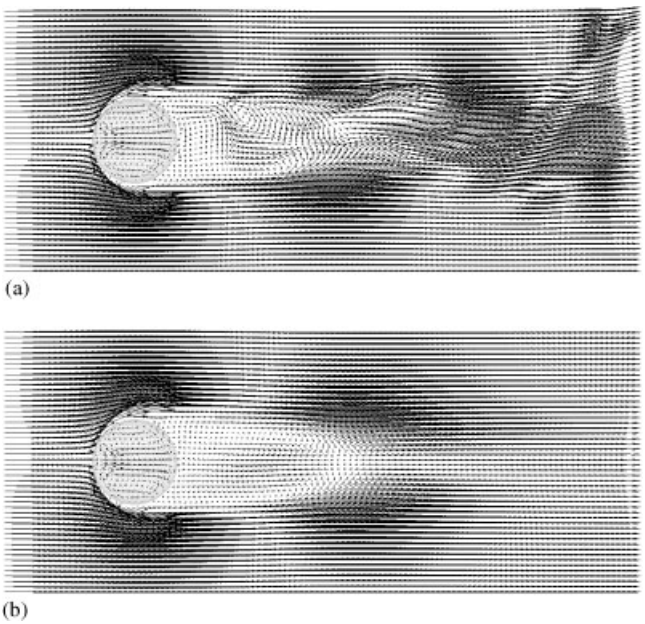


Figure 10. Instantaneous and averaged velocity fields at $Re = 1000$.

## 5. CONCLUDING REMARKS

A second order accurate method for describing complex boundaries have been developed and applied to flow around solid spherical objects. Good agreement with both experimental data and data from other simulation approaches was achieved. A multi-grid scheme for the surface forces was also introduced. Major advantages with this type of method is that grid generation and implementation of higher order discretization is straight forward since the grid is Cartesian. However, resolving the flow close to the boundary is essential to achieve good overall accuracy. Since Cartesian grids are used and the equations are solved also in the interior of the object one will, in order to resolve the interface, also resolve the interior. Hence, one will 'waste' computational cells, although this might be remedied by using refinement only close to the interface (so called 'clustering'). Anyway, it was concluded by Fadlun *et al.* [18] that the computational power required for their approach was substantially lower than for a code using unstructured, body-fitted and deforming grid.

The following main conclusions can be drawn from the results of this work:

- The order of accuracy is largely dependent on how the source terms are distributed to the computational domain. The order of the interpolation/extrapolation scheme to determine the boundary velocity seems to have less influence, at least if the order is above 2. It was indicated by Revstedt and Fuchs [22] that using a second order Lagrangian polynomial decreases the overall order of accuracy. In the present state the method is, at best, second order accurate. By avoiding to take finite differences across the interface one

 

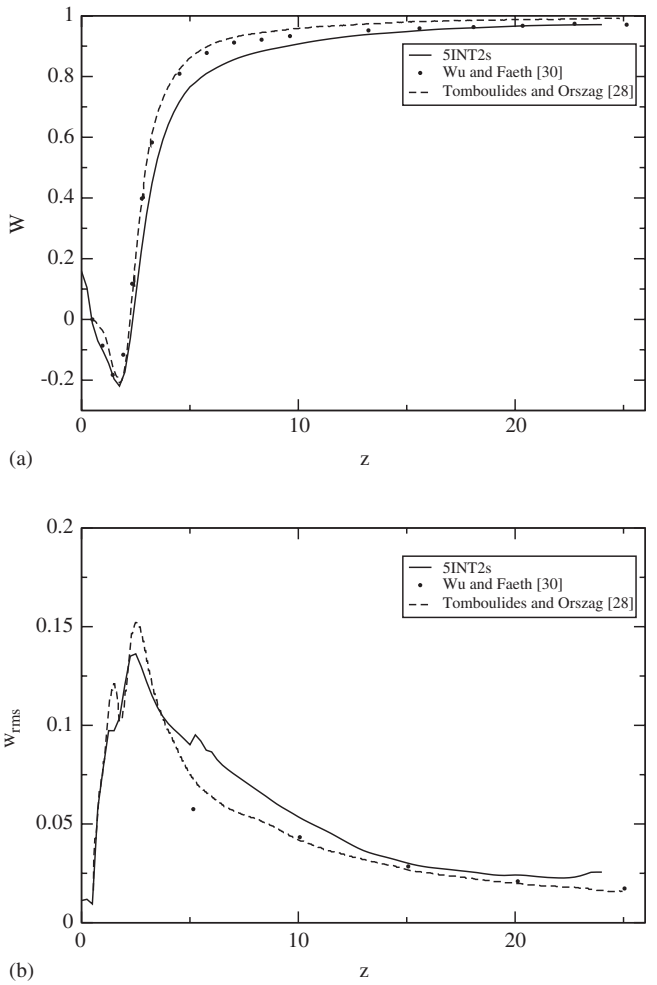


Figure 11. Mean (a) and rms. of fluctuation (b) of the main flow velocity along the wake centre line for $Re = 1000$.

might increase the order of accuracy further. This can for example be achieved by using a essentially non-oscillatory (ENO) scheme in the discretization.

- The multi-grid scheme substantially increases the rate of convergence, up to a factor 10. This is shown to be most useful for stationary flows and time dependent situations with rapid changes in boundary conditions. For time dependent flows with stationary boundaries the effect of the multi-grid scheme was only seen in the start-up of the simulation, i.e. as the boundary is formed. Hence, the full potential of the multi-grid is utilized in stationary flows, e.g low Reynolds number flows but might also be considered for statistically stationary high Reynolds number flows when using a Reynolds averaged Navier–Stokes (RANS) approach, and for flows with time dependent boundaries, e.g. high frequency oscillating objects.

 

- For higher Reynolds numbers (above 300) the second order method fails to converge. This might be due to insufficient resolution of the boundary layer, causing the higher order interpolation to introduce instabilities in the solution. However, the first order approach still gives very good results as compared to literature data.
- As the forces acting on the boundary are directly available, one may, after verification of the method at higher Reynolds numbers and by coupling it to a model for structural analysis, obtain an efficient method for studying vibrating/deformable structures, for example tube bundles.

## ACKNOWLEDGEMENTS

This work was financially supported by the Swedish National Energy Administration (STEM) through the Fluid Mechanics program. The support of the centre for scientific computing at Lund University, LUNARC, is also greatfully acknowledged.

## REFERENCES

1. Gullbrand J, Bai X-S, Fuchs L. Large eddy simulation of turbulent reacting flows using Cartesian grid and boundary corrections. *AIAA Paper* 1998; 98–3317.
2. Lörstad D, Fuchs L. A volume of fluid (VOF) method for handling solid objects using fixed Cartesian grids. In *Moving Boundaries VI—Computational Modelling of Free and Moving Boundary Problems*, Sarler B, Brebbia CA (eds). WIT Press: Southampton, 2001; 143–152.
3. Hu HH, Patankar NA, Zhu MY. Direct numerical simulations of fluid-solid systems using Arbitrary Lagrangian–Eulerian technique. *Journal of Computational Physics* 2001; **169**:427–462.
4. Farhat C, Geuzaine P, Grandmont C. The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. *Journal of Computational Physics* 2001; **174**:669–694.
5. Peskin CS. Numerical analysis of blood flow in the heart. *Journal of Computational Physics* 1977; **25**: 220–252.
6. Lai M-C, Peskin CS. An immersed boundary method with formal second order accuracy and reduced numerical viscosity. *Journal of Computational Physics* 2000; **160**:705–719.
7. Lafaurie B, Mantel T, Zaleski S. Direct Navier–Stokes simulation of the near-nozzle region. In *ILASS-Europe-98*, 1998; 54–59.
8. Lafaurie BN, Scardovelli, Zaleski S, Zanetti G. Modelling merging and fragmentation in multiphase flows with SURFER. *Journal of Computational Physics* 1994; **113**:134–147.
9. Esmaeeli A, Tryggvason G. Direct numerical simulations of bubbly flows. Part 1. Low Reynolds number arrays. *Journal of Fluid Mechanics* 1998; **377**:313–345.
10. Glowinski R, Pan T-W, Periaux J. A fictitious domain method for external incompressible viscous flow modeled by Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1994; **112**:133–148.
11. Glowinski R, Pan T-W, Periaux J. A fictitious domain method for Dirichlet problem and applications. *Computer Methods in Applied Mechanics and Engineering* 1994; **111**:283–303.
12. Glowinski R, Pan T-W, Periaux J. Fictitious domain method for the simulation of Stokes flow past a moving disk. In *Computational Fluid Dynamics '96*, Desideri JA *et al.* (eds). Wiley: New York, 1996; 64–70.
13. Glowinski R, Pan T-W, Hesla TI, Joseph DD. A distributed Lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flows* 1999; **25**:755–794.
14. Shyy W, Udaykumar HS, Choi D. Structured moving grid and geometric conservation laws for fluid flow computation. *Numerical Heat Transfer*, *Part A* 1998; **34**:369–397.
15. Shyy W, Udaykumar HS, Rao MM, Smith RW. *Computational Fluid Dynamics with Moving Boundaries*. Taylor & Francis: Washington DC, 1995.
16. Goldstein D, Handler R, Sirovich L. Modeling a no-slip flow boundary with an external force field. *Journal of Computational Physics* 1993; **105**:354–366.
17. Saiki EM, Biringen S. Numerical simulation of a cylinder in uniform flow: application of a virtual boundary method. *Journal of Computational Physics* 1996; **123**:450–465.
18. Fadlun EA, Verzicco R, Orlandi P, Modh-Yusof J. Combined immersed-boundary finite-difference methods for three-dimensional complex flow situations. *Journal of Computational Physics* 2000; **161**:35–60.

 

19. Mohd-Yusof J. Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries. In *Annual Research Briefs*, Moin P, Reynolds WC (eds), Center For Turbulence Research: Stanford University, 1997; 317–327.
20. Patankar NA, Singh P, Joseph DD, Glowinski R, Pan T-W. A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flows* 2000; **26**:1509–1524.
21. Revstedt J, Kovács T, Fuchs L, Trägårdh C. Influence of impeller type on the flow structure in a stirred reactor. *AIChE Journal* 2000; **46**:2373–2382.
22. Revstedt J, Fuchs L. Handling complex boundaries on a Cartesian grid using surface singularities. *International Journal for Numerical Methods in Fluids* 2001; **35**:125–150.
23. Fuchs L, Zhao H-S. Solution of three-dimensional viscous incompressible flows by a multi-grid method. *International Journal for Numerical Methods in Fluids* 1984; **4**:539–555.
24. Fuchs L. Defect-corrections and higher numerical accuracy. In *Proceedings of the GAMM-workshop on* '*Efficient Solvers for Elliptic Systems*', *Notes on Numerical Methods in Fluid Mechanics*, Hackbusch W (ed.). Vieweg: Braunschweig, 1984; 52–66.
25. Ericsson M, Fuchs L. Modelling of turbulent flow in an IC-engine configuration. In Atluri SN, Yagawa G, Cruse TA (eds). *Computational Mechanics*, Springer Verlag: Berlin, Germany, 1995; 983–988.
26. Johnson TA, Patel VC. Flow past a sphere up to a Reynolds number of 300. *Journal of Fluid Mechanics* 1999; **378**:19–70.
27. Stokes GG. On the effect of the internal friction of fluids on the motion of pendulums. *Transactions of the Cambridge Philosophical Society* 1851; **9**(II):8–106.
28. Tomboulides AG, Orszag SA. Numerical investigation of transitional and weak turbulent flow past a sphere. *Journal of Fluid Mechanics* 2000; **416**:45–73.
29. Ploumhans P, Winckelmans GS, Salmon JK, Leonard A, Warren MS. Vortex methods for direct numerical simulation of three-dimensional bluff body flows: application to the sphere at $Re = 300, 500$ and 1000. *Journal of Computational Physics* 2002; **178**:427–463.
30. Wu JS, Faeth GM. Sphere wakes in still surroundings at intermediate Reynolds numbers. *AIAA Journal* 1993; **31**(8):1448–1455.

 