

Vortex particles and tree codes: I. flows with arbitrary crossing between solid boundaries and particle redistribution lattice; II. vortex ring encountering a plane at an angle.

P. Ploumhans¹, G. S. Winckelmans & J. K. Salmon

ploumhan@term.ucl.ac.be, gsw@term.ucl.ac.be, Center for Systems Engineering and Applied Mechanics, Unité TERM, Place du Levant 2, Université catholique de Louvain, Louvain-la-Neuve 1348, Belgium;
johns@cacr.caltech.edu, Center for Advanced Computing Research, California Institute of Technology, Pasadena, CA 91125, USA.

The first part of the paper deals with 2-D vortex methods. We briefly review the numerical method and we extensively describe two points in more details: the redistribution around an arbitrary body and a new scheme to diffuse a vortex sheet onto particles in a way that is most accurate. These two improvements allow to perform DNS of 2-D flows past bodies of arbitrary shape. We test and validate them on the flow past an impulsively started circular cylinder. The second part of the paper is about 3-D flows with solid boundaries. We test the viscous method on the interaction between a vortex ring and a plane. We also detail a modified version of the Particle Strength Exchange (PSE) scheme close to the solid boundary so as to enhance the accuracy of the method. Fast vortex codes are used for both 2-D and 3-D computations. The 3-D code is also parallel.

1 2-D flows past bodies of arbitrary shape

We wish to solve the vorticity equation for incompressible 2-D flows,

$$\frac{D\omega}{Dt} = \nu \nabla^2 \omega, \quad (1)$$

with no-slip at the wall and using vortex particles (position \mathbf{x}_p , circulation Γ_p). The no-slip boundary condition is modelled by a flux of vorticity at the wall (Koumoutsakos, Leonard & Pépin, 1994; Koumoutsakos & Leonard, 1995; Koumoutsakos & Shiels, 1996; Leonard *et al.*, 1997). One time step of the method may be summarized as:

1. compute $\frac{d\mathbf{x}_p}{dt}$ using \mathbf{u} (computed from ω using the fast tree code)
 $\frac{d\Gamma_p}{dt}$ from diffusion using the Particle Strength Exchange scheme (PSE)
2. update $\mathbf{x}_p(t)$ and $\Gamma_p(t)$ for the time step Δt .
3. compute the vortex sheet, $\Delta\gamma$, so as to cancel the slip velocity at the wall.
4. diffuse $\Delta\gamma$ onto the nearby particles (i. e. , modify their strength) for the time step Δt .
5. redistribute if necessary.

In the next two sections, we detail points 3, 4 and 5 of the method. Points 1 and 2 are extensively described in the literature.

¹funded by *Fonds pour la Formation à la Recherche dans l'Industrie et dans l'Agriculture (FRIA)*, parallel computer time provided by *Action de Recherche Concertée (ARC)* of profs. R. Keunings & V. Legat.

1.1 Vortex sheet computation

After having moved the particles with the local velocity and updated their strength according to the PSE, there is a slip velocity, $\Delta \mathbf{U}_{\text{slip}}$, at the wall. We put a vortex sheet, $\Delta \gamma$, on the surface of the body, whose strength is computed so as to cancel the panel-averaged tangential slip velocity (Leonard *et al.*, 1997). The vortex sheet is discretized using M boundary elements (i.e., “vortex sheet panels”) each of size $O(h)$ and each with uniform strength. Computing the panel strengths amounts to solving a linear system. This is so because, on top of inducing a uniform tangential velocity underneath themselves equal to $\mathbf{n} \times \Delta \gamma / 2$, the panels also induce a velocity on one another. However, for the simple 2-D geometries that correspond to the **infinite plane** and the **circle**, one obtains an analytical solution for the required panel strength without having to actually solve the linear system: $\Delta \gamma = 2 \mathbf{n} \times \Delta \mathbf{U}_{\text{slip}}$.

1.2 Vortex sheet diffusion

The total flux to be emitted into the flow for the other substep of the diffusion process is given by

$$\nu \frac{\partial \omega}{\partial \mathbf{n}} = \frac{\Delta \gamma}{\Delta t} . \tag{2}$$

This flux must be emitted during a time Δt . In effect, the vortex sheet $\Delta \gamma$ must be distributed to neighbor particles by discretizing the Greens’ integral for the inhomogeneous Neumann problem corresponding to the diffusion equation. See Koumoutsakos *et al.* (1994) for a simple scheme in 2-D. Winckelmans (1993, private communication) and Leonard *et al.* (1997), have proposed an improved scheme. Consider a uniform panel of strength $\Delta \gamma$ located along the y -axis, from $-b/2$ to $b/2$, and diffusing to the right (the left being the body into which the vortex sheet does not diffuse), as shown in Fig. 1. Then, in this scheme, the amount of circulation, $\Delta \Gamma_i$, that must be received by the particle located at $x_i > 0, y_i$ (any sign) is given by

$$\Delta \Gamma_i = \int_{x_i-h/2}^{x_i+h/2} \int_{y_i-h/2}^{y_i+h/2} \Delta \omega \, dx \, dy \tag{3}$$

with h^2 the fluid area associated to each particle. $\Delta \omega$ stands for the change in vorticity due to the flux from the panel acting over a time Δt , and is itself the result of a time integral:

$$\Delta \omega = \int_0^{\Delta t} \frac{d\omega}{dt} dt . \tag{4}$$

Thus, the amount of circulation received by the particle i is

$$\Delta \Gamma_i = \int_0^{\Delta t} \frac{d\Gamma_i}{dt} dt \tag{5}$$

where $\frac{d\Gamma_i}{dt}$ can be integrated exactly:

$$\begin{aligned} \frac{d\Gamma_i}{dt} = \int \int \frac{d\omega}{dt} \, dx \, dy &= \frac{\Delta \gamma}{\Delta t} \left(\left[\text{erfc}(s) \right]_{\frac{x_i-h/2}{\sqrt{4\nu t}}}^{\frac{x_i+h/2}{\sqrt{4\nu t}}} \right) \\ &\left(\sqrt{4\nu t} \frac{1}{2} \left(\left[\text{ierfc}(s) \right]_{\frac{(y_i-b/2)+h/2}{\sqrt{4\nu t}}}^{\frac{(y_i-b/2)-h/2}{\sqrt{4\nu t}}} - \left[\text{ierfc}(s) \right]_{\frac{(y_i+b/2)+h/2}{\sqrt{4\nu t}}}^{\frac{(y_i+b/2)-h/2}{\sqrt{4\nu t}}} \right) \right) \end{aligned} \tag{6}$$

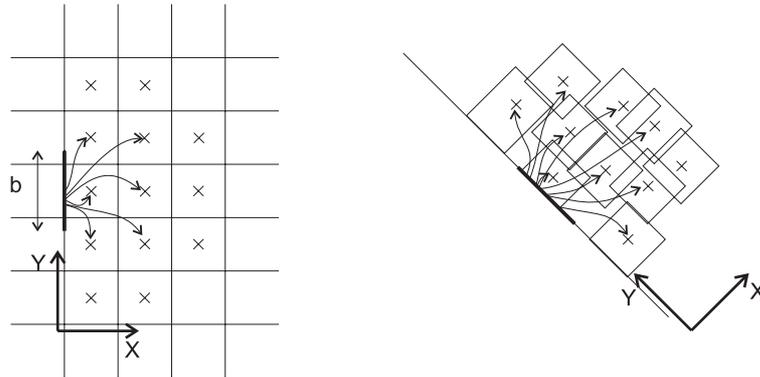


Figure 1: panel to particles diffusion: particles on a regular lattice aligned with panel (left); arbitrary position of particles near the wall (right).

where $\operatorname{erfc}(s) = \int_s^\infty \frac{2}{\sqrt{\pi}} \exp(-v^2) dv$ and $\operatorname{ierfc}(s) = \int_0^\infty \operatorname{erfc}(v) dv = \frac{1}{\sqrt{\pi}} \exp(-s^2) - s \operatorname{erfc}(s)$. Notice that $h_l/2 = x_i$ if $0 \leq x_i < h$ and $h_l/2 = h/2$ otherwise. This allows for particles in the “first layer” to be closer or further away to the y -axis than $h/2$ and it ensures that the scheme remains conservative in such cases, i.e., that $\Delta\gamma$ is exactly distributed to the particles: $b \Delta\gamma = \sum_i \Delta\Gamma_i$.

The time integral in Eq. (5) is computed using Gauss quadrature (3 or 4 points). If particles are on a regular lattice aligned with the panel, this scheme is conservative. In practice, however, the spatial distribution of the particles is not uniform: the proposed scheme is not exactly conservative. We are thus led to make the following correction in order to enforce conservation: instead of using $\Delta\Gamma_i$ given by Eq. (5), we use

$$\Delta\Gamma_{i,\text{conserv}} = \Delta\Gamma_i + \frac{(\Delta\Gamma_i)^2}{\sum_j (\Delta\Gamma_j)^2} \left(b \Delta\gamma - \sum_j \Delta\Gamma_j \right) \tag{7}$$

where j runs over all particles concerned by the panel $\Delta\gamma$. This scheme minimizes $\sum_i \frac{(\Delta\Gamma_i - \Delta\Gamma_{i,\text{conserv}})^2}{(\Delta\Gamma_i)^2}$ with the constraint that $(b \Delta\gamma) - (\sum_i \Delta\Gamma_{i,\text{conserv}}) = 0$.

In order for diffusion with the above schemes to work properly, the spatial distribution of the particles must remain fairly uniform. This is another reason why particle redistribution is needed every 5 to 10 time steps. It is also needed to minimize the PSE and convection errors.

1.3 Particle redistribution

In particle methods DNS, one needs to maintain the condition that particle cores overlap. This calls for a particle redistribution scheme. It consists in replacing the distorted set of vortex particles by a new set where the particles are, again, located on a $h \times h$ lattice. Consider first the normalized 1-D problem with unit spacing. In the $\Lambda_3(x)$ scheme, an old particle located at $-1/2 \leq x \leq 1/2$

gives

$$\left\{ \begin{array}{l} \frac{1}{48}(3-2x)(4x^2-1) \\ \frac{1}{16}(1-2x)(9-4x^2) \\ \frac{1}{16}(1+2x)(9-4x^2) \\ \frac{1}{48}(3+2x)(4x^2-1) \end{array} \right. \text{ to the particle located at } \begin{array}{l} -\frac{3}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \\ \frac{3}{2} \end{array}, \quad (8)$$

If a wall is present, the redistribution of particles close to the wall must be modified, because we can't allow particles inside the body. We do this by using decentered Λ'_p schemes. We here detail two such schemes, considering that an old particle is located at $-\frac{1}{2} \leq x \leq \frac{1}{2}$ and that the wall is at $x = -\frac{1}{2}$ for the Λ'_2 scheme and at $x = -1$ for the Λ'_3 scheme:

$$\Lambda'_2(x) \text{ gives } \left\{ \begin{array}{l} \frac{1}{2}(x-2)(x-1) \\ x(2-x) \\ -\frac{1}{2}x(1-x) \end{array} \right. \text{ to the particle located at } \begin{array}{l} 0 \\ 1 \\ 2 \end{array}, \quad (9)$$

$$\Lambda'_3(x) \text{ gives } \left\{ \begin{array}{l} -\frac{4}{3}(1+2x)(2x-1)(2x-3) \\ 4(3-2x)(1-2x)(3+2x) \\ 4(1+2x)(2x+3)(3-2x) \\ \frac{4}{3}(2x+3)(2x-1)(2x+1) \end{array} \right. \text{ to the particle located at } \begin{array}{l} -\frac{1}{2} \\ \frac{1}{2} \\ \frac{3}{2} \\ \frac{5}{2} \end{array}, \quad (10)$$

Our approach consists in using a Λ_3 scheme for particles located more than 3/2 from the wall, a Λ'_3 scheme for particles with distance between 1/2 and 3/2, and a Λ'_2 scheme for particles less than 1/2 from the wall.

In 2-D, the problem of deciding which scheme to use near the boundaries is more involved because, in general, the boundary crosses the redistribution lattice in an arbitrary way. We proceed in two steps. First, we redistribute an old particle in the x -direction and create temporary particles. We use a Λ_3 scheme if it is possible (that is, if it doesn't introduce particles inside the body); if not, we use a Λ'_3 scheme and if it still leads to the creation of particles inside the body, we use a Λ'_2 scheme. This redistribution in the x -direction has thus created 4 (Λ_3 or Λ'_3) or 3 (Λ'_2) temporary particles. Each temporary particle is then redistributed in the y -direction using, in order of preference, a Λ_3 , Λ'_3 or Λ'_2 scheme. Note that the scheme used in the y -direction is not necessarily the same for each temporary particle. We give a penalty p_i to each new particle, with

$$p_i = \left\{ \begin{array}{l} 0 \text{ if only centered schemes were used} \\ 1 \text{ if a } \Lambda'_p \text{ scheme was used for the } x \text{ or the } y\text{-direction} \\ 2 \text{ if } \Lambda'_p \text{ schemes were used for both the } x \text{ and } y\text{-directions} \end{array} \right.$$

This leads to the creation of N_{np} new particles. We then compute a global penalty,

$$P_{XY} = \frac{\sum_{i=1}^{N_{np}} p_i}{N_{np}}. \quad (11)$$

In a second step, we compute the global penalty P_{YX} obtained when doing the redistribution first in the y -direction and then in the x -direction. In the end, we retain the order of redistribution that leads to the lowest global penalty. If $P_{XY} = P_{YX}$, we use an average of the two schemes. This duplication of the redistribution in both directions has no significant computational cost, because P_{XY} is zero for almost all particles (that is, except those very close to the wall). If P_{XY} vanishes, the second redistribution step does not need to be completed.

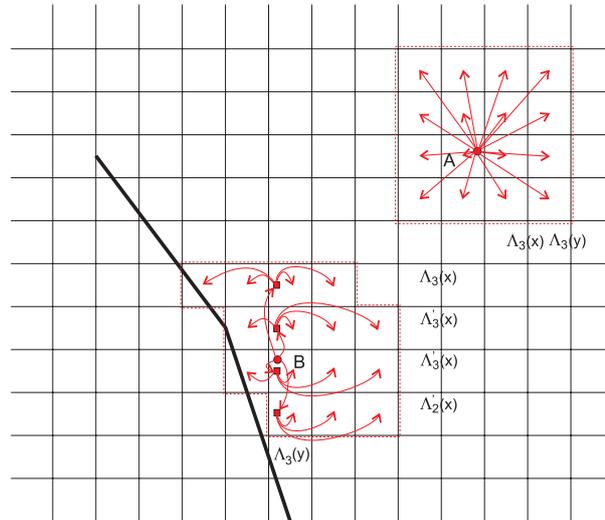


Figure 2: 2-D redistribution schemes; Λ_3 for point A, $\Lambda_3\Lambda_3'\Lambda_2'$ for point B.

Fig. 2 gives an example of the $\Lambda_3\Lambda_3'\Lambda_2'$ redistribution scheme. Particle A is far from the walls, so we use a Λ_3 scheme in both direction. Particle B is close enough to the wall that we must use decentered schemes: we have $P_{XY} = \frac{5}{4}$ and $P_{YX} = \frac{11}{15}$, so we redistribute first in the y -direction and then in the x -direction.

1.4 Test case: the flow past an impulsively started cylinder

We will now test the “general redistribution” scheme and the new “panel to particles diffusion” scheme on the flow past an impulsively started cylinder. This flow is interesting because: (1) it is a benchmark for bluff body flows; (2) there exists an analytical solution valid for short times (Bar-Lev & Yang, 1975); (3) this flow has been extensively studied by Koumoutsakos and Leonard (1995) (referred hereafter as KL) using a vortex method, but with a redistribution technique specifically designed for the circle.

Fig. 3 shows the difference between the KL approach and our general method. In the KL method, the particles are initially (and also after each redistribution) located in a way that “follows” the boundary, but each particle still has an area of h^2 associated to it. This is clearly the best one can do in order to get accurate results for the flow past the cylinder. However, it cannot be extended to arbitrary geometry.

We have implemented a redistribution technique similar to what was used in KL, in order to have a “KL reference” solution for long times. Redistribution is done with a Λ_3 scheme in the θ direction and with, in order of preference, a Λ_3 , Λ_3' or Λ_2' scheme in the r direction. We compare our results with the analytical solution and with the results obtained with the KL approach for short times. Long time comparisons are also made. We compare, as a function of $T = \frac{tU_\infty}{D}$, and for $Re = \frac{U_\infty D}{\nu} = 550$, the isocontours of vorticity, the x component of momentum, $I_x = \int_\Omega y \omega \, d\Omega = \sum_p y_p \Gamma_p$, and the drag coefficient: $C_D = \frac{F_x}{\frac{1}{2} \rho U_\infty^2 D}$ with $F_x = -\rho \frac{dI_x}{dt}$ where $\frac{dI_x}{dt} \approx \frac{I_x(t+\Delta t) - I_x(t-\Delta t)}{2 \Delta t}$.

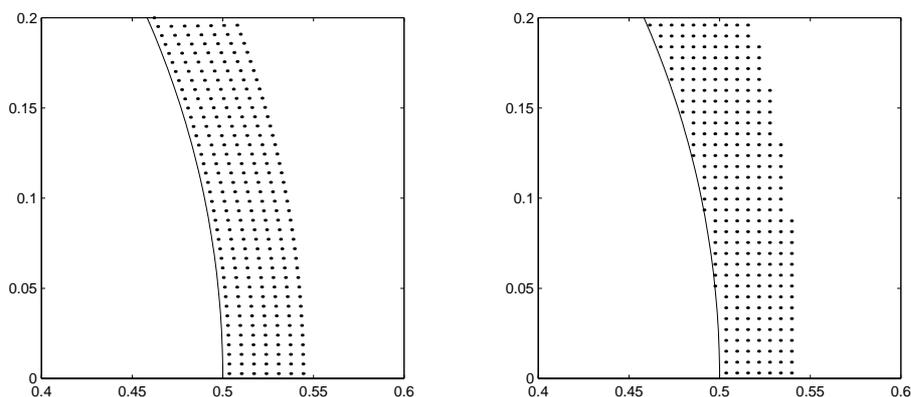


Figure 3: The KL redistribution technique (left), and the general one (right).

The parameters of the simulation are: $\Delta T = 0.01$, $h/D = 6.0288 \cdot 10^{-3}$. We use gaussian particles with core function: $\zeta_\sigma = 1/(2\pi\sigma^2) \exp(-r^2/(2\sigma^2))$, with $\sigma/h = 1$. Redistribution is done every 5 time steps. If a new particle has $\Gamma < 0.001 \Gamma_{\max}$ it is deleted (in order to avoid too high a growth rate for the number of particles). We also enforce the presence of 7 layers of particles around the body after each redistribution. There are 592 panels. Eq. (5) is integrated with 4 Gauss points. The integration scheme for convection is a second order Adams-Bashforth (second order Runge-Kutta for the first step and after each redistribution); for PSE diffusion, we use a first order Euler explicit scheme. We use a fast tree code, with multipole expansions of third order. The mean error estimate (on the norm of the velocity) is $\sim 3 \cdot 10^{-3}$. The number of particles goes from ~ 5000 to ~ 70000 during the simulation. The total run time is ~ 3 hours on a DEC alpha, 433 Mhz.

Fig. 4 shows isocontours of vorticity for the KL approach and for our general approach. The agreement between these techniques is very good, although the minimum and maximum value of ω may be slightly different (because they correspond to particles very close to the wall). Small oscillations in the low-value isocontour of the general method are present, because it is located next to the region without particles.

Fig. 5 shows a comparison between the analytical expression of I_x for short times, I_x computed with the KL, and with the general approach. It also shows a comparison for longer times. The drag coefficient, C_D , is also shown: short times and longer times. The C_D curves exhibit a high frequency noise. This is due to the redistribution: Λ_p schemes indeed conserve the first p moments of the vorticity, but not the time derivative of those moments. As might be expected, this noise has a greater amplitude for the general scheme than for the KL approach. The reason is that, with the KL method, particles are never very close to the body.

A convergence analysis of our general method is shown in Fig. 6. We have made two simulations of the same flow: one at $h/D = 6.0288 \cdot 10^{-3}$, $\Delta T = 0.01$ and one at $h/D = 3.0144 \cdot 10^{-3}$, $\Delta T = 0.0025$. Thus, both have the same $\frac{\nu \Delta t}{h^2} = \frac{1}{2}$. This figure shows that the general method indeed converges well: the values of I_x and C_D obtained with the smaller h/D are remarkably close to the analytical

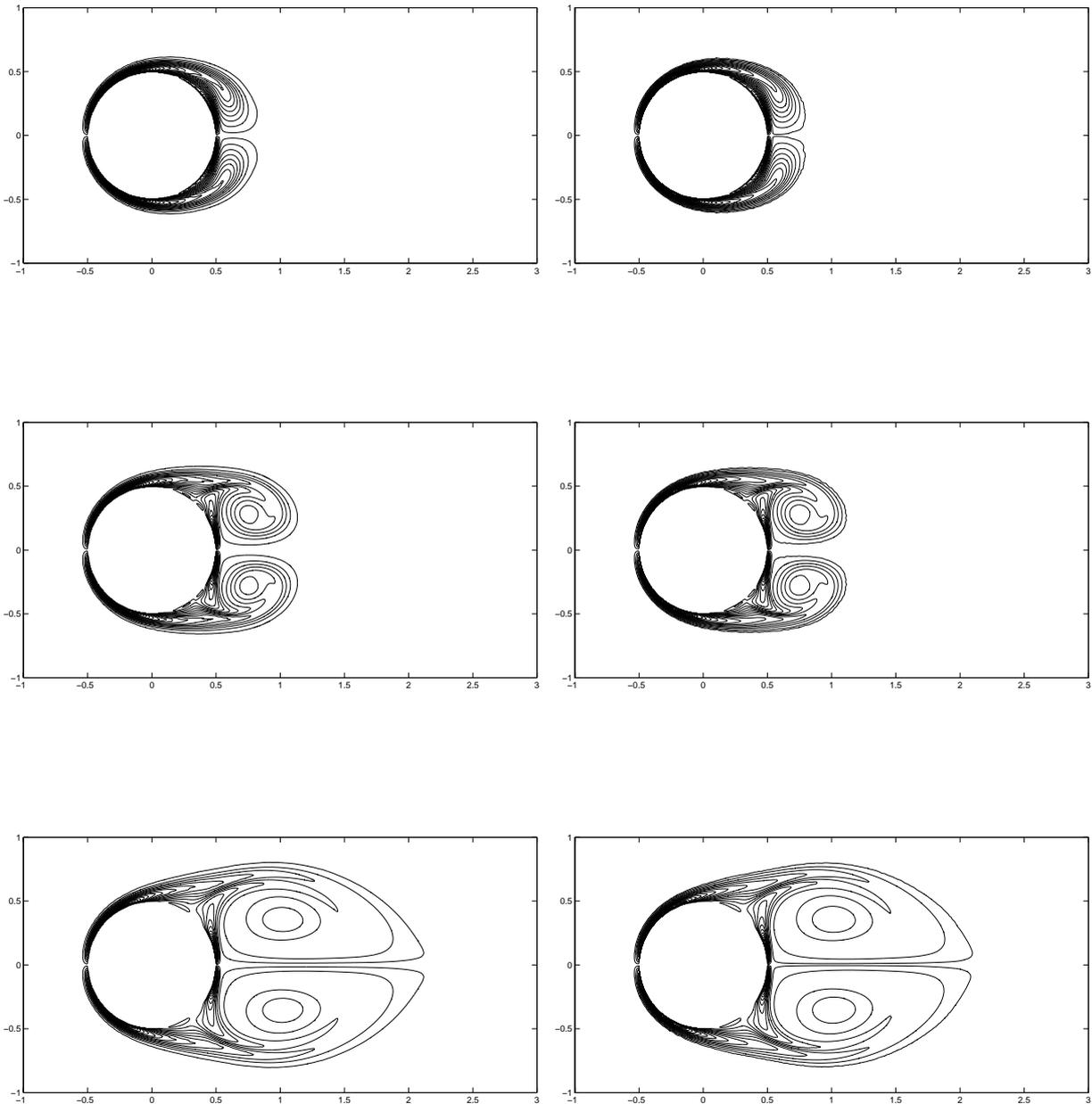


Figure 4: Iso ω comparison for $T = 1, 2$ and 5 ; KL (left), general (right). Iso levels are (top to bottom, left to right): -47 to 47 , -43 to 43 , -37 to 37 , -47 to 47 , -43 to 43 , -41 to 41 , by step of 2 .

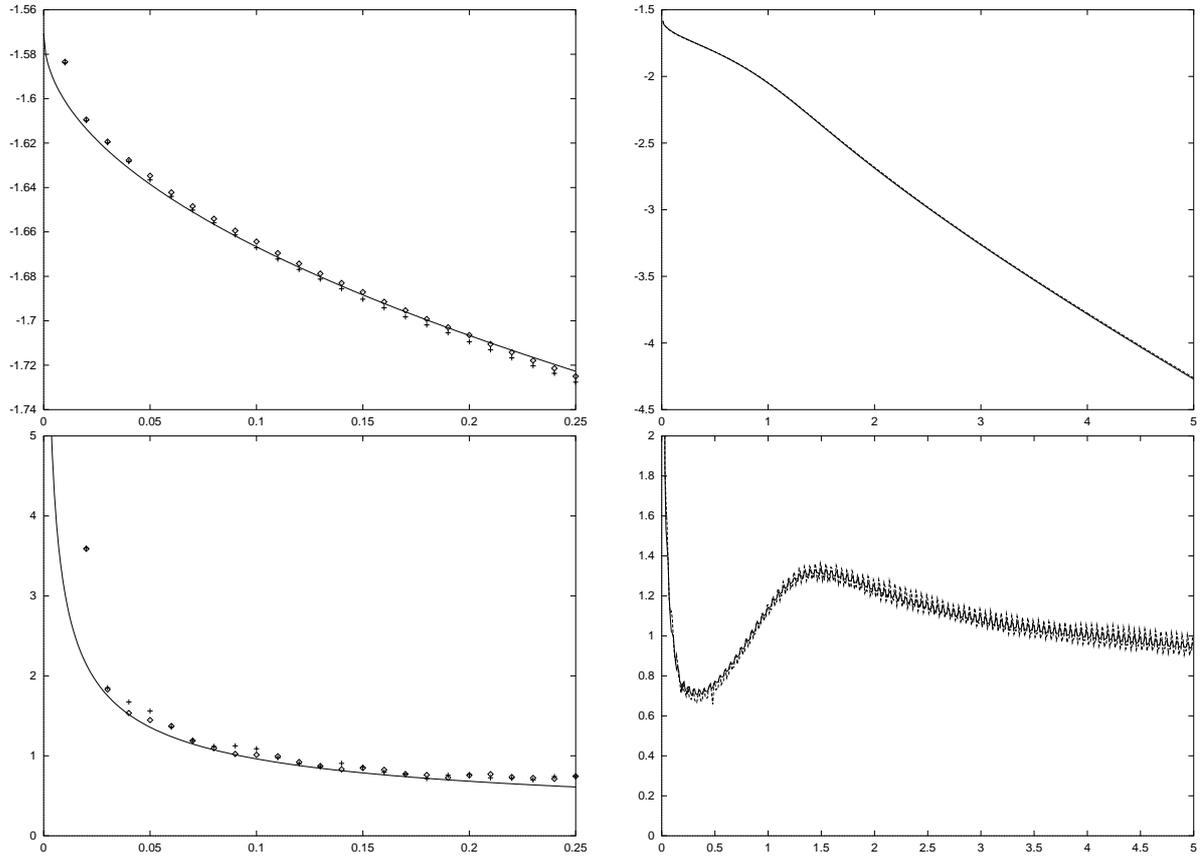


Figure 5: Short time I_x comparison (top left): analytical (solid), KL (\diamond), general (+). Long time I_x comparison (top right): KL (solid), general (dashed). Short time C_D comparison (bottom left): analytical (solid), KL (\diamond), general (+). Long time C_D comparison (bottom right): KL (solid), general (dashed).

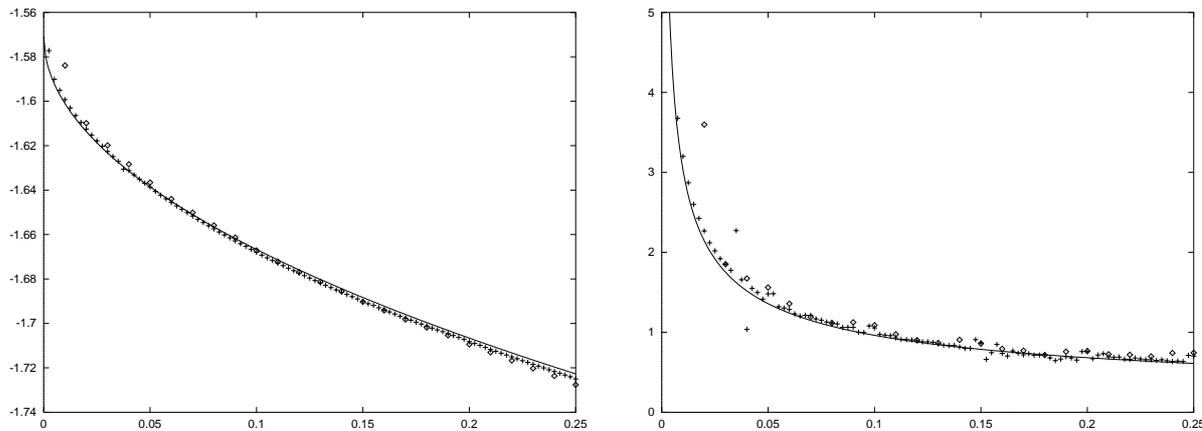


Figure 6: Convergence of I_x for the general approach (left) and of C_D (right): analytical (solid), $h/D = 6.0288 \cdot 10^{-3}$, $\Delta T = 0.01$ (\diamond); $h/D = 3.0144 \cdot 10^{-3}$, $\Delta T = 0.0025$ (+).

values.

2 3-D flows with no-slip boundary condition

We wish to solve the 3-D vorticity equation,

$$\frac{D\boldsymbol{\omega}}{Dt} = (\nabla\mathbf{u}) \cdot \boldsymbol{\omega} + \nu \nabla^2 \boldsymbol{\omega}, \quad (12)$$

with no-slip at the wall and using vortex particles (position \mathbf{x}_p , strength $\boldsymbol{\alpha}_p$). We consider here a flat surface at $y = 0$ and a flow above that plate ($y > 0$). The no-slip boundary condition corresponds to a flux of vorticity at the wall for ω_x and ω_z , and to a Dirichlet boundary condition: $\omega_y = 0$. One time step of the 3-D method can be summarized as:

1. compute $\frac{d\mathbf{x}_p}{dt}$ using \mathbf{u} (computed from $\boldsymbol{\omega}$ using the fast tree code)
 $\frac{d\boldsymbol{\alpha}_p}{dt}$ from diffusion using the modified PSE (with $\frac{\partial\omega_x}{\partial y} = \frac{\partial\omega_z}{\partial y} = 0$ and $\omega_y = 0$),
 and from stretching using $\nabla\mathbf{u}$ (computed from $\boldsymbol{\omega}$ using the fast tree code)
2. update $\mathbf{x}_p(t)$ and $\boldsymbol{\alpha}_p(t)$ for the time step Δt .
3. compute the vortex sheet, $\Delta\boldsymbol{\gamma}$, so as to cancel the slip velocity at the wall. Here

$$\Delta\gamma_x = 2 \Delta U_{\text{slip},z} \text{ and } \Delta\gamma_z = -2 \Delta U_{\text{slip},x}. \quad (13)$$

4. diffuse $\Delta\boldsymbol{\gamma}$ onto the nearby particles (i. e. , modify their strength) for the time step Δt ,

$$\nu \frac{\partial\omega_x}{\partial y} = \frac{\Delta\gamma_x}{\Delta t} \text{ and } \nu \frac{\partial\omega_z}{\partial y} = \frac{\Delta\gamma_z}{\Delta t}. \quad (14)$$

5. redistribute if necessary (Λ_3 scheme in x and z -direction, $\Lambda_3\Lambda'_3\Lambda'_2$ scheme in y -direction).

Steps 1 and 2 are described in the literature (e. g. , Winckelmans & Leonard, 1993; Winckelmans *et al.*, 1995, 1996), except for the modified PSE which will be explained in the next section. Steps 3 and 5 are trivial because of the simple plane geometry.

2.1 Enforcement of the no-slip boundary condition in 3-D

Steps 1, 3 and 4 of the method contribute to the boundary condition enforcement. During Step 1, we use a modified PSE for particles close to the wall (less than 5σ): we use image particles when computing the PSE. Image particles are such that, if a particle at position $x\mathbf{e}_x + y\mathbf{e}_y + z\mathbf{e}_z$ has weight $\alpha_x\mathbf{e}_x + \alpha_y\mathbf{e}_y + \alpha_z\mathbf{e}_z$, the corresponding image particle is at $x\mathbf{e}_x - y\mathbf{e}_y + z\mathbf{e}_z$ and has weight $\alpha_x\mathbf{e}_x - \alpha_y\mathbf{e}_y + \alpha_z\mathbf{e}_z$. This guarantees that, at the wall: (1) $\omega_y = 0$, (2) $\frac{\partial\omega_x}{\partial y} = \frac{\partial\omega_z}{\partial y} = 0$ and (3) $\frac{\partial^2\omega_y}{\partial y^2} = 0$ during the PSE substep.

After having moved the particles with the local velocity, and updated their strength according to the modified PSE and to the stretching, there is a slip velocity, $\Delta\mathbf{U}_{\text{slip}}$, at the wall. We compute the vortex sheet, $\Delta\boldsymbol{\gamma}$, on the surface so as to cancel the slip velocity underneath the sheet (Step 3).

We then transfer the strength of the vortex sheet onto the neighbor particles. We do this by solving the diffusion equation for a time Δt . The 3-D equivalent of Eq. (6) is

$$\begin{aligned} \frac{d\alpha_i}{dt} = & \frac{\Delta\gamma}{\Delta t} \cdot \left([\text{erfc}(u)] \frac{\frac{y_i-h_l/2}{\sqrt{4\nu t}}}{\frac{y_i+h/2}{\sqrt{4\nu t}}} \right) \\ & \left(\sqrt{4\nu t} \frac{1}{2} \left([\text{ierfc}(u)] \frac{\frac{x_i+h/2-b/2}{\sqrt{4\nu t}}}{\frac{x_i-h/2-b/2}{\sqrt{4\nu t}}} - [\text{ierfc}(u)] \frac{\frac{x_i+h/2+b/2}{\sqrt{4\nu t}}}{\frac{x_i-h/2+b/2}{\sqrt{4\nu t}}} \right) \right) \\ & \left(\sqrt{4\nu t} \frac{1}{2} \left([\text{ierfc}(u)] \frac{\frac{z_i-h/2-f/2}{\sqrt{4\nu t}}}{\frac{z_i+h/2-f/2}{\sqrt{4\nu t}}} - [\text{ierfc}(u)] \frac{\frac{z_i-h/2+f/2}{\sqrt{4\nu t}}}{\frac{z_i+h/2+f/2}{\sqrt{4\nu t}}} \right) \right) \end{aligned} \tag{15}$$

with $h_l/2 = y_i$ if $0 \leq y_i < h$ and $h_l/2 = h/2$ otherwise. This is valid for a rectangular panel of size $b \times f$. The panels we use for the discretization of the vortex sheet are sometimes triangular. In order to still use Eq. (15), we replace each triangular panel by a square of size $b \times b$, with a surface, b^2 , equal to the surface of the triangle. Once again, the amount of circulation received by particle i is finally obtained from

$$\Delta\alpha_i = \int_0^{\Delta t} \frac{d\alpha_i}{dt} dt, \tag{16}$$

the integral being computed, typically, with 4 Gauss points. As in 2-D, this scheme is conservative if particles are on a regular lattice and are aligned with the panel. To enforce conservation in all situations, we use the same type of correction as the one described in Section 1.2.

Another point of interest is related to the divergence of ω . Maintaining a nearly divergence-free vorticity field is crucial for accurate 3-D simulations. The problem already arises for unbounded flows. One can certainly expect further complications when dealing with wall bounded flows. Casciola, Piva and Bassanini (1996) have extensively studied the problem of vorticity generation on a flat surface. They have proved that if ω is initially divergence free, and if the vortex sheet, $\Delta\gamma$, has zero 2-D divergence,

$$\frac{\partial\Delta\gamma_x}{\partial x} + \frac{\partial\Delta\gamma_z}{\partial z} = 0, \tag{17}$$

then the diffusion of $\Delta\gamma$ conserves the zero divergence. The problem is thus to produce a divergence free $\Delta\gamma$. Casciola, Piva and Bassanini (1996) have also proved that, if $\Delta\gamma$ is calculated as in Eq. (13) and if $\Delta\mathbf{U}_{\text{slip}}$ is 2-D irrotational,

$$\frac{\partial\Delta\mathbf{U}_{\text{slip},x}}{\partial z} - \frac{\partial\Delta\mathbf{U}_{\text{slip},z}}{\partial x} = 0, \tag{18}$$

then Eq. (17) is automatically satisfied. One can show that, if $\omega_y = 0$ at $y = 0$ and if ω is divergence free, then $\Delta\mathbf{U}_{\text{slip}}$ will indeed verify Eq. (18). The use of the modified PSE, coupled with a divergence free vorticity field, guarantees that the enforcement of the boundary condition on ω_x and ω_z conserves the zero divergence of the vorticity. The conclusion is that, with the proposed algorithm, the presence of the wall does not change the nature of the divergence problem: one needs to maintain ω nearly divergence free in the bulk of the flow, as is also required for unbounded flow computations.

2.2 3-D application: vortex ring at an angle of attack

As an application/validation of the method, we consider a vortex ring of radius R , of Gaussian vorticity distribution, $\omega(r) = \frac{\Gamma}{2\pi a^2} \exp(-\frac{1}{2}\frac{r^2}{a^2})$ with $a/R = 0.1$, with a Reynolds number, $Re =$

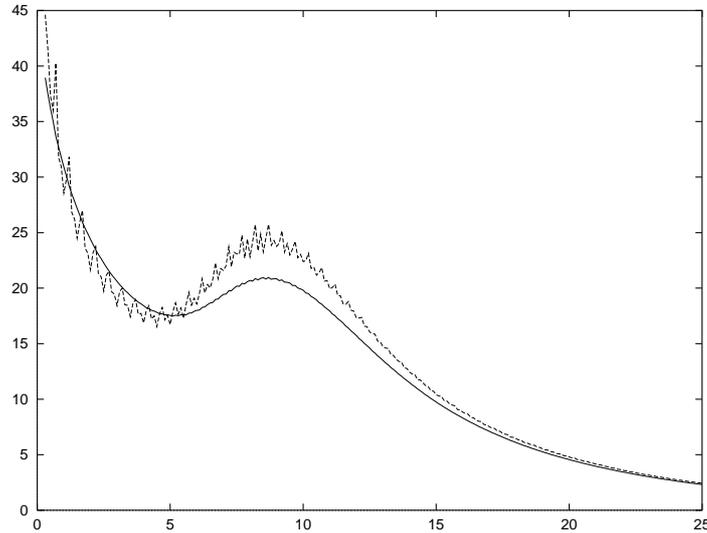


Figure 7: $2\mathcal{E}$ (solid) and $-\frac{1}{\nu} \frac{dE}{dt}$ (dashed) as a function of $T = \frac{t\Gamma}{R^2}$: $Re = \Gamma/\nu = 400$, $\alpha = 15$ deg, $\frac{\Delta t \Gamma}{R^2} = 0.1$, $\frac{h}{R} = 0.05$ and $\frac{\sigma}{h} = 1$.

$\Gamma/\nu = 400$, inter-particle spacing, $h/R = 0.05$, initial height $H/R = 2$, angle of attack of $\alpha = 15$ deg. The time step is $\Delta T = \Delta t \Gamma/R^2 = 0.1$, and the particle core size is $\sigma/h = 1$ (Gaussian core function). The time integration scheme is the $O(\Delta t^2)$ Adams-Bashforth. Since it is not self-starting, a second order Runge-Kutta scheme is used for the first time step and after each redistribution. The infinite flat surface is modelled by a circle of radius $6R$, made of roughly equilateral triangles (~ 86000 panels), with $h_{\text{panel}} \approx h$. Redistribution takes place every 5 time steps, and we guarantee a layer of 4 particles up to a radius of $5R$ after each redistribution. We use a Λ_3 scheme in the x and z -directions, and a $\Lambda_3 \Lambda'_3 \Lambda'_2$ approach in the y -direction. New particles with $\|\alpha\|$ smaller than $0.001 \|\alpha\|_{\text{max}}$ are deleted (in order to avoid too high a growth rate for the number of particles). The average slip velocity over a panel is computed with a 4 points Hammer rule, and Eq. (16) is integrated with a 4 points Gauss quadrature. This simulation used a fast and parallel tree code (Barnes & Hut, 1986; Greengard & Rohklin, 1987; Warren & Salmon, 1993; Salmon, Warren & Winckelmans, 1994; Salmon & Warren, 1994; Warren & Salmon, 1995(a), 1995(b); Winckelmans *et al.*, 1995, 1996) using multipole expansions of order 2. The code ran for 30 hours on 4 processors (DEC alpha, 533 Mhz). The number of particles goes from ~ 148000 to ~ 346000 particles. For the fast code, the mean error estimate (on the velocity norm) is $\sim 2 \cdot 10^{-3}$. Fig. 8 shows iso-contours of vorticity.

One way to make a global validation of this computation is to compare the energy decay rate and the enstrophy. Indeed, if one defines the kinetic energy, $E = \frac{1}{2} \int_{\Omega} \mathbf{u} \cdot \mathbf{u} \, d\Omega$, and the enstrophy, $\mathcal{E} = \frac{1}{2} \int_{\Omega} \boldsymbol{\omega} \cdot \boldsymbol{\omega} \, d\Omega$, then the relation

$$\frac{dE}{dt} = -2\nu\mathcal{E} \quad (19)$$

should be satisfied. Fig. 7 shows that, although Eq. (19) is not exactly satisfied, there is a good agreement. Note that there is an increase in enstrophy, \mathcal{E} , between $T \approx 5$ and $T \approx 8$. This is due to the stretching and to the vorticity flux at the wall. Further validations of the method will be conducted in the future: higher Reynolds number (Orlandi & Verzicco, 1993) and higher angle of attack (Lim, 1989; Verzicco & Orlandi, 1994, 1996).

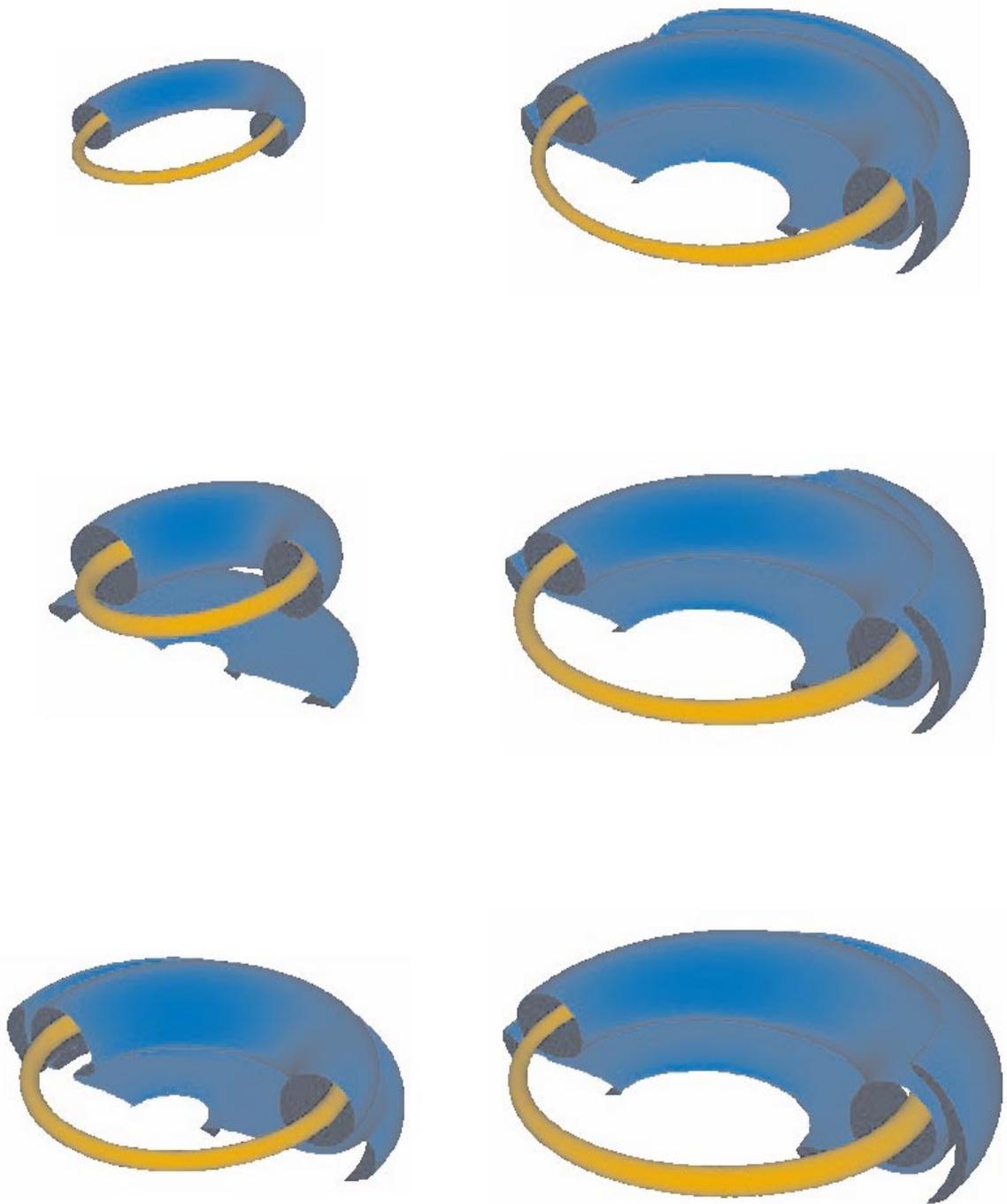


Figure 8: isocontours of vorticity (15% and 85% of $\|\boldsymbol{\omega}\|_{max}$), $T = t\Gamma/R^2 = 0, 5, 10, 15, 20, 25$ (top to bottom, left to right), $Re = \Gamma/\nu = 400$, $\alpha = 15$ deg, $\|\boldsymbol{\omega}\|_{max} = 12.77, 4.54, 3.46, 2.39, 1.45, 0.93$.

3 Conclusion

We have presented and validated two improvements of the 2-D vortex method that allow to perform DNS of flows past bodies of arbitrary shape. We have also developed the viscous boundary condition for 3-D vortex methods, including a modified PSE scheme near the wall. It has been validated on the simulation of a vortex ring encountering a plane at an angle of attack.

References

- 1 BAR-LEV, M. AND YANG, H. T.: Initial flow field over an impulsively started circular cylinder, *J. Fluid Mech.* **72** (1975), 625–647.
- 2 BARNES, J. E., AND HUT, P.: A hierarchical O (N log N) force calculations algorithm, *Nature*, **324** (1986), 446–449.
- 3 CASCIOLA, C. M., PIVA, R. AND BASSANINI, P.: Vorticity Generation on a Flat Surface in 3D Flows, *J. Comput. Phys.* **129** (1996), 345–356.
- 4 GREENGARD, L. AND ROHKLIN, V.: A fast algorithm for particle simulations, *J. Comput. Phys.* **73** (1987), 325–348.
- 5 KOUMOUTSAKOS, P., LEONARD, A., AND PEPIN, F.: Boundary conditions for viscous vortex methods, *J. Comput. Phys.*, **113** (1994), 52–61.
- 6 KOUMOUTSAKOS, P. AND LEONARD, A.: High resolution simulation of the flow around an impulsively started cylinder using vortex methods, *J. Fluid Mech.* **296** (1995), 1–38.
- 7 KOUMOUTSAKOS, P. AND SHIELS, D.: Simulation of the viscous flow normal to an impulsively started and uniformly accelerated flat plate, *J. Fluid Mech.* **328** (1996), 177–228.
- 8 LEONARD, A., SHIELS, D., SALMON, J.K., WINCKELMANS, G.S., PLOUMHANS, P.: Recent Advances in High Resolution Vortex Methods for Incompressible Flows, *Proc. 13th Computational Fluid Dynamics Conf.*, Snowmass Village, CO, June 29 - July 2, 1997, AIAA #97-2108.
- 9 LIM, T. T.: An experimental study of a vortex ring interacting with an inclined wall, *Exps. Fluids*, **7** (1989), 453–463.
- 10 ORLANDI, P. AND VERZICCO, R.: Vortex rings impinging on walls: axisymmetric and three-dimensional simulations, *J. Fluid Mech.* **256** (1993), 615–646.
- 11 SALMON, J. K., WARREN, M. S., AND WINCKELMANS, G. S.: Fast parallel tree codes for gravitational and fluid dynamical N-body problems, *Int. J. Supercomputer Applications*, **8** (1994), 129–142.
- 12 SALMON, J. K. AND WARREN, M. S.: Skeletons from the Treecode Closet, *J. Comput. Phys.*, **111**(1) (1994), 136–155.
- 13 VERZICCO, R. AND ORLANDI, P.: Normal and Oblique Collisions of a Vortex Ring with a Wall, *Mechanica*, **29** (1994), 383–391.
- 14 VERZICCO, R. AND ORLANDI, P.: Wall/vortex-ring interactions, *Appl. Mech. Rev.* **49** (1996), 447–461.
- 15 WARREN, M.S. AND SALMON, J.K.: A parallel hashed oct-tree N-body algorithm, In *Supercomputing '93* (1993), 12–21, Los Alamitos, IEEE Comp. Soc.
- 16 WARREN, M.S. AND SALMON, J.K.: A portable parallel particle program, *Computer Physics Communications*, **87** (1995), 266–290.
- 17 WARREN, M.S. AND SALMON, J.K.: A parallel, portable and versatile treecode, In *Seventh SIAM Conference on Parallel Processing for Scientific Computing* (1995), 319–324, Philadelphia. SIAM.
- 18 WINCKELMANS, G. S. AND LEONARD, A.: Contributions to vortex particle methods for the computation of three-dimensional incompressible unsteady flows, *J. Comput. Phys.*, **109** (2) (1993), 247–273.
- 19 WINCKELMANS, G.S., SALMON, J.K., LEONARD, A. AND WARREN, M.S.: Three-dimensional vortex particle and panels methods: Fast tree-code solvers with active error control for arbitrary distributions/geometries, In *Proc. Forum on Vortex Methods for Engineering Applications* (1995), Albuquerque, NM, 22–24 Feb. 1995, 25–43.
- 20 WINCKELMANS, G. S., SALMON, J. K., WARREN, M. S., LEONARD, A. AND JODOIN B.: Application of fast parallel and sequential tree codes to computing three-dimensional flows with the vortex element and boundary element methods, *Second International Workshop on Vortex Flows and Related Numerical*

Methods, Montréal, Canada, Aug. 1995, electronic Proc. 1996: <http://www.emath.fr/proc/Vol.1/> (17 pages).