

3-D Vortex Simulation of Flow Over A Circular Disk at An Angle of Attack

Adrin Gharakhani* and Mark J. Stock†
Applied Scientific Research, Santa Ana, CA 92705

A Lagrangian Vortex Method for grid free simulation of unsteady incompressible laminar flow over 3-D geometries is developed. To this end, the fluid vorticity is discretized using singular vortex particles and its inviscid transport predicted in a Lagrangian frame of reference using smooth velocity and velocity gradients. Diffusion is accounted for via the application of a Vorticity Redistribution Method, which facilitates automatic insertion of new vortex particles as vorticity spreads out due to diffusion. The object boundaries are discretized with triangular panels, and the wall velocity boundary conditions imposed by solving a Petrov-Galerkin discretization of the Fredholm boundary integral equation of the second kind for the vector-valued vortex sheets at the wall. A parallel adaptive oct-tree code is used to accelerate the velocity (and gradient) evaluations due to the vortex particles and panels. This paper describes the developed algorithm briefly and presents results from the simulation of impulsively started flow over a circular discoid, with thickness equal to a tenth of its diameter and positioned at a 30 degree angle of attack. The Reynolds number based on the freestream velocity and the discoid diameter is 1,000 for this problem.

I. Introduction

MESHLESS, particle-based algorithms are becoming increasingly popular in computational sciences because not only do they reduce preprocessing complexities substantially, by essentially removing the meshing stage, but also because they offer the potential to yield better accuracy and convergence properties due to the nature of the shape functions used. The Lagrangian Vortex-Boundary Element Method (LVBEM) described in this paper belongs to a special subclass of particle-based computational techniques, which are best suited for simulating vortex-dominated incompressible flow in/around complex geometries. The present LVBEM is completely *grid-free* in the fluid domain, and it provides solution based adaptivity as well as minimal numerical diffusion due to the Lagrangian nature of accounting for convection. Additionally, continuity and far-field boundary conditions (for external flows) are satisfied by construction.

An advanced LVBEM has been developed by Ploumhans *et al.*¹ and its robustness as a Direct Numerical Simulation (DNS) tool demonstrated using the example of flow over a sphere at three Reynolds numbers up to 1,000. The DNS algorithm presented in this paper differs from the latter most significantly in its accounting for diffusion of vorticity from the boundaries and in the fluid domain. The former use the so-called Particle Strength Exchange (PSE)² method, which suffers from the following disadvantages. PSE conserves *only* the zeroth moment of the diffusion equation. Further, the PSE formulation uses the particle "control volume" explicitly, which is not clearly defined and/or easily obtained; actually, control volumes are meaningless in grid-free computing. Moreover, it is necessary to redistribute the particles onto a uniform "grid" *once every few timesteps* to maintain the accuracy of PSE. This diminishes the appeal of vortex methods as a "grid-free" tool with minimal numerical diffusion, since regridding is associated with certain complications for complex geometries, and the application of traditional projection schemes introduces numerical errors back to the computations.^{1,3} A newly developed projection scheme based on radial basis functions holds strong promise for maintaining accuracy, but it is presently quite costly.³

The advanced algorithm in this paper uses the so-called Vorticity Redistribution Method (VRM),⁴⁻⁶ which offers the following advantages. VRM conserves the moments of the diffusion equation to arbitrarily high order.⁴⁻⁶ It does not require frequent redistribution onto uniform "grids" to maintain accuracy.⁵ Furthermore, it diffuses circulation, not vorticity; therefore, it does not use control volumes explicitly. By its virtue of diffusing circulation, VRM can be used with singular vortex particles, which is significant because singular particles represent the *exact* discrete

* President, adrin@Applied-Scientific.com, AIAA Senior Member.

† Research Engineer.

solution of the vorticity transport equations,⁷ and are thus ideal for DNS.⁵ In contrast, smoothed elements (of the type used in PSE) discretize the convolution of the transport equations with the smoothing function and are, therefore, of Large Eddy Simulation (LES) character.⁷ Another significant advantage of VRM over other grid-free methods is its ability to mathematically detect and fill "holes" that invariably develop when severe stretching of the vorticity field disperses the vortex elements, and/or when the vorticity field front expands due to diffusion.⁸ It should be mentioned here that VRM has been characterized incorrectly by some in the literature as being an expensive algorithm. However, although VRM may be computationally more expensive than PSE on a *per element* basis, its *actual* cost is equivalent to (and even somewhat less than) that of PSE, because VRM has a compact stencil and encompasses far fewer neighboring particles to diffuse vorticity to as compared to PSE, which uses a Gaussian core/shape function with an effective stencil half-size of more than twice the core radius. We have implemented and tested both PSE and VRM, and we have found the CPU times for the two to be roughly equal, and much smaller than the cost for evaluating the velocity of the particles, which contributes to more than 90 percent of the total cost per computational timestep, even with an efficient fast solver in place.

Ploumhans *et al.*¹ use a higher order version of the semi-analytic algorithm developed by Koumoutsakos *et al.*⁹ for diffusing the boundary (fluid) vorticity in the context of PSE (using a Neumann boundary condition for vorticity). We have developed an equivalent VRM formulation, whereby the boundary vortex sheets retain a fraction of their strength and give off the remainder to two "layers of particles" adjacent to the wall, such that the zeroth through second moments of the diffusion equation are preserved. Recall that VRM can diffuse the circulation of *singular* particles and sheets, thus obviating the "loss of vorticity/circulation" caused by the core overlap of *smooth* vortex elements with the wall. The presentation of this new formulation is deferred to the future. In this paper, a simpler strategy is presented, which conserves the circulation and the first moment of diffusion from the wall. Preliminary benchmark tests have shown that the latter is as accurate as the more complex implementation, at least for the cases tested.

The algorithm presented in this paper has been benchmarked successfully against more established methods and experimental data for the case of impulsively started flow over a sphere for Reynolds numbers of up to 100, based on the freestream velocity and the diameter.¹⁰ The Reynolds numbers were deliberately selected at this low range, because they correspond to a steady axisymmetric flow regime, which is quite difficult to capture accurately with vortex methods, because unless the diffusion solver is robust and can damp out the growth of instabilities, the simulations may predict unphysical flow unsteadiness and/or asymmetry. In these simulations, the flow displayed no unsteadiness and, more importantly, remained axisymmetric for the duration of the simulation (through the steady state solution), despite the fact that vortex particles were distributed non-uniformly and that *no particle remeshing* was used during the run.

In this paper, the formulation for the proposed hybrid LVBEM-VRM is presented briefly, followed by a demonstration of preliminary results from the simulation of flow over a circular discoid, with thickness equal to a tenth of its diameter. The discoid is positioned at a 30 degree angle of attack with respect to the freestream, and the Reynolds number based on the freestream velocity and the discoid diameter is set to 1,000.

II. Formulation

The governing equations for the transport of fluid vorticity are^{1,11}

$$\frac{\partial \bar{\omega}}{\partial t} + \bar{u} \cdot \bar{\nabla} \bar{\omega} = \bar{\omega} \cdot \bar{\nabla} \bar{u} + \frac{1}{\text{Re}} \nabla^2 \bar{\omega} \quad (1a)$$

$$\bar{\nabla} \cdot \bar{u} = 0 \quad (1b)$$

$$\bar{\omega} = \bar{\nabla} \times \bar{u} \quad (1c)$$

$$(\bar{u}_{\text{wall}}, \bar{u}_{\text{initial}}) = \text{prescribed} \quad (1d)$$

where \bar{u} and $\bar{\omega}$ are the velocity and vorticity vectors, respectively, $\bar{\nabla}$ is the gradient operator, t is time, Re is the Reynolds number based on the freestream velocity and some characteristic length scale, and $(\bar{u}_{\text{wall}}, \bar{u}_{\text{initial}})$ are the user specified boundary and initial conditions for velocity, respectively.

The discrete vorticity field is represented by N_v *singular* vortex particles having "vectorial circulation" $\bar{\Gamma}_p$:

$$\bar{\omega}(\bar{x}, t) = \sum_{p=1}^{N_V} \bar{\Gamma}_p(t) \delta(\bar{x} - \bar{x}_p) \quad (2)$$

where \bar{x} is the position vector and $\delta(\cdot)$ is the Dirac Delta function.

The particle velocities (and their gradients) are smooth in these computations and are evaluated by convolving the Biot-Savart integral for velocities with a smoothing (or core) function:¹¹

$$\bar{u}_\sigma(\bar{x}, t) = \sum_{p=1}^{N_V} K_\sigma(\bar{x} - \bar{x}_p) \times \bar{\Gamma}_p(t) + \bar{U}_\infty(\bar{x}, t) \quad (3a)$$

$$K_\sigma(\bar{x}) = K(\bar{x}) \int_0^{|\bar{x}|/\sigma} g(r) r^2 dr \quad , \quad K(\bar{x}) = -\frac{\bar{x}}{4\pi|\bar{x}|^3} \quad (3b)$$

where \bar{u}_σ is the smooth velocity vector, \bar{U}_∞ is the freestream velocity, and $g(r) = (3/4\pi) \exp(-r^3)$ is a core function with core radius σ . The smooth velocity gradient is obtained by differentiating Eq. (3) directly.¹¹

The time evolution of the discrete vorticity field is then obtained via the following set of equations for all vortex particles $p = 1, \dots, N_V$:

$$\frac{d\bar{x}_p}{dt} = \bar{u}_{\sigma,p} \quad (4a)$$

$$\frac{d\bar{\Gamma}_p}{dt} = \bar{\Gamma}_p \cdot \bar{\nabla} \bar{u}_{\sigma,p} \quad (4b)$$

$$\frac{\partial \bar{\omega}_p}{\partial t} = \frac{1}{\text{Re}} \nabla^2 \bar{\omega}_p \quad (4c)$$

where $\bar{u}_{\sigma,p}$ represents the smooth velocity of particle p , and $\frac{d}{dt}$ depicts the time derivative in the Lagrangian frame of reference. Equations (4a-c) represent a viscous splitting algorithm, whereby the convection and stretch of vorticity are evaluated along particle trajectories, and diffusion is obtained subsequently in an Eulerian frame.¹² A second-order predictor-corrector time integration is used for Eqs. (4a-b).

The solution of Eq. (4c) is obtained using the Vorticity Redistribution Method via the following equations⁴⁻⁶

$$\sum_{q=0}^{N_q} f_{pq} = 1 \quad (5a)$$

$$\sum_{q=1}^{N_q} f_{pq} \bar{x}_{pq} = 0 \quad (5b-d)$$

$$\sum_{q=1}^{N_q} f_{pq} x_{pq} y_{pq} = \sum_{q=1}^{N_q} f_{pq} y_{pq} z_{pq} = \sum_{q=1}^{N_q} f_{pq} z_{pq} x_{pq} = 0 \quad (5e-g)$$

$$\sum_{q=1}^{N_q} f_{pq} \bar{x}_{pq}^2 = 2\Delta t / \text{Re} \quad (5h-j)$$

where (x_{pq}, y_{pq}, z_{pq}) are the components of $\bar{x}_{pq} = \bar{x}_p - \bar{x}_q$, Δt is the computational timestep, and f_{pq} is the fraction of circulation $\bar{\Gamma}_p$ that each particle p gives off to particle q in its neighborhood of influence such that the zeroth

through second moment conditions of Eq. (4c) are preserved. Note that N_n is the number of particles within the neighborhood of influence and $q = 0$ represents p itself.

The system of equations (5) is generally under-determined and is solved in the L_∞ -norm using linear programming.⁴⁻⁶ If the system is not solvable for a given set of particles, new particles are inserted on a sphere of radius $R\sqrt{\Delta t/\text{Re}}$ until a solution is obtained. Stability analysis, constrained by the requirement that the solution be positive, $f_{pq} \geq 0$, yields $R \geq \sqrt{6}$. In this paper, $R = \sqrt{8}$ is used, which relaxes the stiffness of the system of equations while maintaining solution accuracy. Note that this particle insertion and solution strategy maintains the nominal inter-particle spacing at $O(R\sqrt{\Delta t/\text{Re}})$ for all times and throughout the computational domain. The radius of the sphere of influence containing the N_n neighboring particles; i.e., the stencil size, is set to $\sqrt{17\Delta t/\text{Re}}$. Also note that the core radius σ must be larger than the inter-particle spacing for stability reasons. In this paper, $\sigma = 1.5R\sqrt{\Delta t/\text{Re}}$ is used.

The no-slip/no-flux boundary conditions are imposed by placing vortex sheets at the wall, with their surface-tangent values $\vec{\gamma}$ obtained subsequent to the solution of the following Fredholm boundary integral equation of the second kind:

$$\frac{1}{2}\vec{\gamma}(\vec{x}) \times \vec{n}(\vec{x}) + \int_{\partial\Omega} \vec{\gamma}(\vec{x}') \times K(\vec{x} - \vec{x}') d\vec{x}' = \vec{U}_s(\vec{x}) \quad (6a)$$

$$\vec{U}_s(\vec{x}) = \vec{U}_\infty(\vec{x}) - \int_{\Omega} \vec{\omega}(\vec{x}') \times K(\vec{x} - \vec{x}') d\vec{x}' \quad (6b)$$

where \vec{n} is the unit normal pointing into the fluid domain Ω , with bounding surface $\partial\Omega$.

The solution of Eq. (6) is obtained in this paper in the Petrov-Galerkin formulation by discretizing the surface of the object with N_p contiguous triangular panels and assuming a singular particle distribution for $\vec{\gamma}$. Equation (6) is integrated with respect to \vec{x} , which yields the following area-integrated boundary element system:

$$\frac{1}{2}\vec{\Gamma}_i \times \vec{n}_i + \sum_{j \neq i}^{N_p} \vec{\Gamma}_j \times \int_{S_j} K(\vec{x} - \vec{x}_j) d\vec{x} = \vec{Q}_s \quad , \quad i = 1, \dots, N_p \quad (7a)$$

$$\vec{Q}_s = \int_{S_i} \vec{U}_\infty(\vec{x}) d\vec{x} - \sum_{p=1}^{N_v} \vec{\Gamma}_p \times \int_{S_i} K(\vec{x} - \vec{x}_p) d\vec{x} \quad (7b)$$

where $\vec{\Gamma}_i = \vec{\gamma}_i A_i$ on the surface is the "panel circulation" and A_i is the area of panel i . The surface integrals of the velocity kernel are evaluated analytically¹³ to get accurate solutions in situations where the source and target points are close to each other. Note that in Eq. (7b) the area-integrated influence of the vortex particles is evaluated using the singular, and not the smooth, velocity kernel to allow analytic integration.

The link between the vorticity generated at the wall and the interior domain is accounted for via diffusion. A simple strategy is used here, which conserves the circulation and the first moment of diffusion from the wall. To this end, at each timestep, all vortex particles within a numerical boundary layer of thickness $R\sqrt{\Delta t/\text{Re}}$ are removed prior to solving Eq. (7). Then, having solved Eq. (7), the circulation of each triangular panel is given off to a corresponding new particle, which is positioned in the fluid along the normal passing through the panel centroid at an elevation of $h = 2\sqrt{\Delta t/(\pi \text{Re})}$. This approach appears to stabilize the simulation and inhibits the unnecessary growth of the number of vortex elements that would otherwise arise from continual insertion of new particles into the flow at every timestep. The algorithm has been benchmarked successfully using the example of flow over a sphere at low Reynolds numbers.¹⁰

The computational complexity of VRM is $O(N_v)$. However, the particle velocity (and velocity gradient) evaluations by the classical Biot-Savart formulation are of $O(N_v^2)$ complexity. Similarly, the system of equations (7) is densely populated and $O(N_p^2)$ in terms of both storage and CPU requirements. To this end, an adaptive oct-tree code with complexity $O(N_v \log N_v)$ and $O(N_p \log N_p)$ has been implemented to speed up the particle and

panel velocity computations, respectively.¹⁴ Further speedup is achieved via efficient MPI-based parallelization of the latter as well as the VRM evaluations for a distributed computing environment using the standard Orthogonal Recursive Bisection (ORB) algorithm. The system of equations (7) is solved iteratively via GMRES.

The proposed algorithm for each timestep is as follows:

- 1) For a given distribution of vortex particles, evaluate the velocity field and its gradients via the parallel, adaptive fast oct-tree implementation of Eq. (3).
- 2) Convect and stretch the vortex particles according to (4a) and (4b), respectively.
- 3) Repeat steps (1) and (2) to complete a second-order time integration.
- 4) Diffuse the vorticity field (modify particle circulation) via VRM formulation, Eq. (6).
- 5) Remove the particles within the numerical boundary layer of thickness $R\sqrt{\Delta t/Re}$, as well as elements that might have jumped out of the fluid domain.
- 6) Impose the wall velocity boundary conditions by solving Eq. (7) for the corresponding wall vortex sheet vectors, using the parallel, adaptive fast oct-tree code.
- 7) Insert new particles in the numerical boundary layer, $h = 2\sqrt{\Delta t/(\pi Re)}$ away from the centroid of each vortex panel and with circulation equal to that of the corresponding vortex panel.
- 8) Repeat steps 1 through 7 for each timestep.

III. Results

Preliminary results from the vortex simulation of the early stages of impulsively started flow over a circular discoid are presented in this section. The discoid is tilted such that its equatorial plane is at a 30 degree angle with respect to the (incoming) freestream flow. The Reynolds number based on the (equatorial) diameter and the freestream velocity is 1,000 for this problem. The object was defined parametrically using the formula $x^2 + y^2 + (10z)^2 = (D/2)^2$, where D is the diameter of the discoid. (All variables are in nondimensional units.) The surface was discretized by splitting an icosahedron meshing of a sphere recursively for 6 iterations, yielding 14,580 triangles, and then "squeezing" the sphere linearly in the z direction to make the discoid. This creates a high quality (fairly uniform) mesh distribution on the top and bottom surfaces of the "lentic"; however, the mesh aspect ratios near the (relatively sharp) edges are not optimal due to the squeezing effect. An optimal meshing for the edges would require adaptive surface meshing, which is not implemented at this stage. The timestep was set to $\Delta t = 5 \times 10^{-3}$, and the simulation was performed for 330 steps up to time $t = 1.65$. This corresponds to the early stages of flow development, but it is sufficient to demonstrate the salient features of the proposed technology.

The simulation was performed in parallel on a cluster of 4 dual-CPU AMD Opteron 246 computers, which are interconnected with gigabit switches. Figure 1 depicts the (a) side and (b) top views of the distribution of the vortex particles over the 8 processors at $t = 1.65$, using the ORB-based domain decomposition. The discoid on the left and the trailing wake vortex are easily identifiable in Fig. 1b. Note that the domain decomposition is performed at each timestep; therefore, the per-processor particle topology changes continuously as the flow evolves. Also note that the flow and the particle distribution are fairly symmetric with respect to the xz plane (x denoting the streamwise

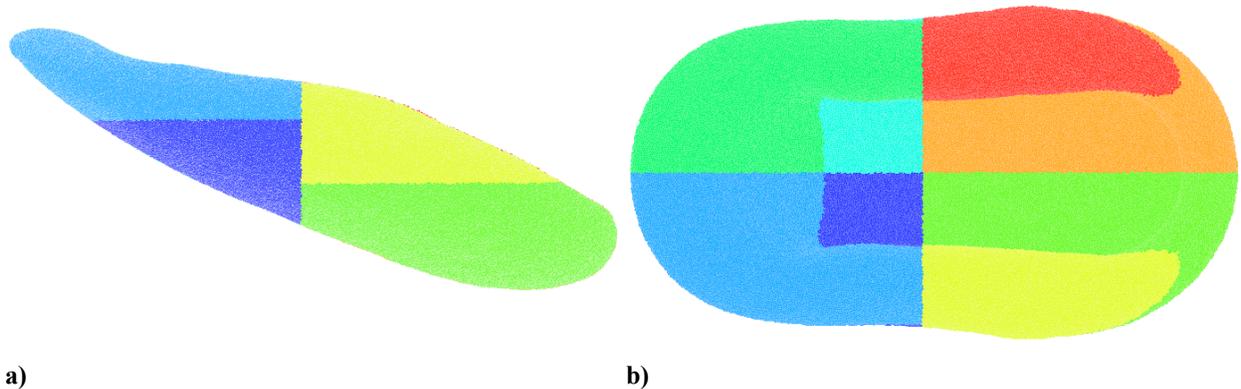


Figure 1. a) Side, and b) top views of the particle distribution on 8 processors

direction); as a result, the particles are distributed equally among four pairs of processors as seen in Fig. 1b. In contrast, there is no symmetry in the streamwise direction, which is reflected in the four different topologies observed in Figs. 1a-b. Recall that, by the nature of ORB, all processors have the same number of particles (to within ± 1). Therefore, a smaller volume implies a higher density of particles (e.g., yellow vs. green below it). This is because the wake consists of a thin planar volume in the central region (green and orange), but spiraling/swirling layers of planes in its "arms" (yellow and red), as can be verified in Figs. 2 and 4.

Figure 2a depicts the particle discretization of the vorticity field, which is color-coded by the magnitude of the strain rate. Figure 2b represents the side view of the particles within a thin prism of thickness 0.02 centered about the plane of symmetry. The colors represent the speed normalized by the maximum speed in the field, with red representing one. Figure 2c displays selected streaklines, with the colors representing the normalized speed. Note that a section of the horseshoe vortex is cut out in Fig. 2a to show the internal structure of the wake as it develops in time. Two sets of observations are in order here.

First, notice in Fig. 2b the existence of a separation bubble at the leading edge of the discoid. This bubble is at the plane of symmetry of a crescent shaped vortical structure, which spans the rim of the upper half of the discoid. The bubble is largest at the plane of symmetry and gradually tapers off toward the edges of the crescent, where it separates from the discoid to form the two "arms" of the horseshoe vortex. The path of the light blue streakline, in Fig. 2c, emanating from the leading edge along the centerline of the discoid "traces" this crescent at the upper half before it separates from the discoid. Also note the existence of two counter-rotating weak eddies at the trailing edge of the discoid, which is not visible clearly but occupies the dark blue region aft-end of the discoid in Fig. 2b. These two eddies are part of two small vortical structures that span along the rim of the trailing edge by roughly half a radius on either side of the plane of symmetry. It should be mentioned here that the flow is still developing, and that the aforementioned structures near the discoid do not necessarily represent the final flow configuration.

Second, notice in Figs. 2a and 2b the concentration of the vortex particles only in regions with significant vorticity, which exemplifies the implicit solution adaptivity of the hybrid LVBEM-VRM algorithm. The circular patch of vorticity in the far right of Fig. 2b is (a cross-section of) the initial vortex that is created due to the impulsive

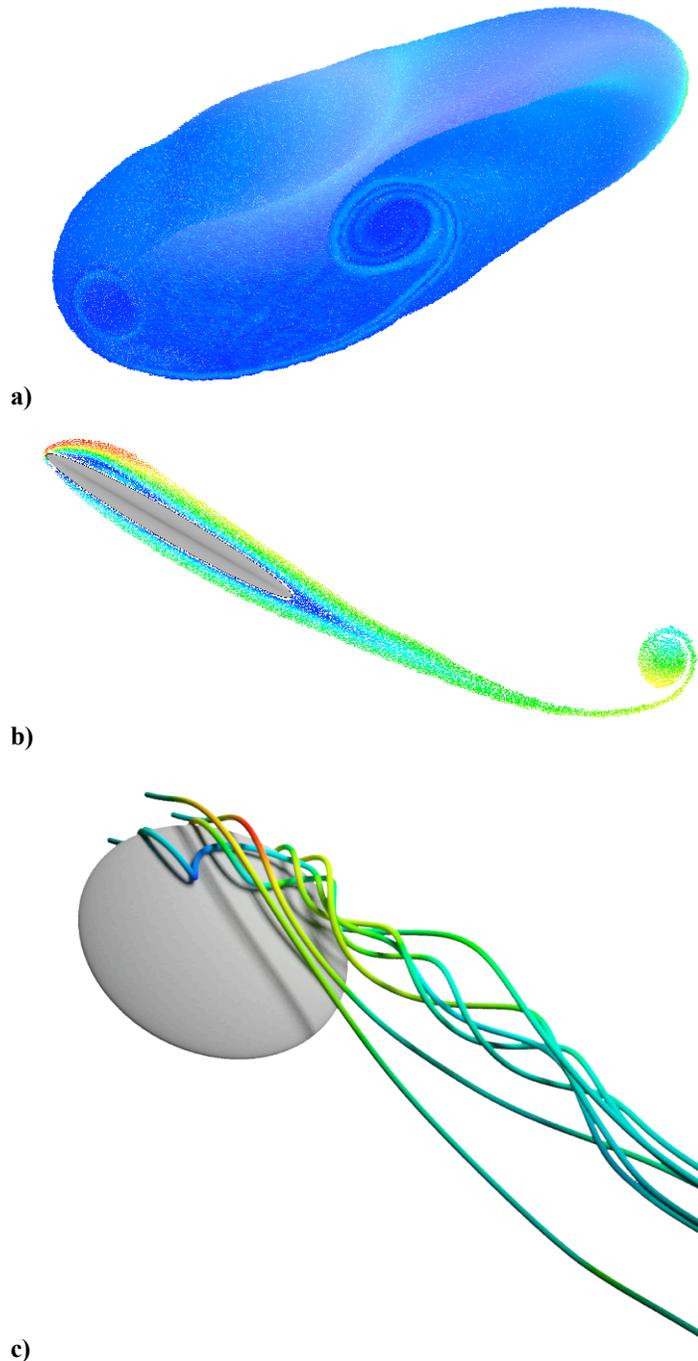


Figure 2. a) Particles, color-coded by the strain rate; b) the velocity field in the plane of symmetry; and c) selected streaklines color-coded by the speed.

start. As this vortex convects downstream its rolling motion entrains the vorticity that is subsequently shed from the object into the wake. The wake entrainment results in the continual thinning of the shear layer and stretching in the streamwise direction, and is seen in Fig. 2b as a spiraling braid that ends in the circular patch. A similar process takes place in the spanwise direction, whereby the two "arms" of the horseshoe vortex (emanating from the two sides of the discoid) slowly expand outwardly in the cross-stream direction as the wake convects downstream, while simultaneously entraining the flow from the central (thin layer) section of the wake into the spiraling flow of the "arms" (as seen in the cut-out of the right vortex core in Fig. 2a, as well as in Fig. 4). Note that this is similar to the familiar flow past the edges of airplane wings, which produces a swirling flow of the type seen in Fig. 2c. The material stretch due to these entrainment processes causes the particles to separate beyond the nominal interparticle spacing, and unless new particles are inserted appropriately, the core overlap condition of stability will be violated and the simulation will soon blow up. In the current simulation, particle insertion is accounted for automatically during the diffusion stage via VRM. It must be emphasized here that (relatively) long time stability has been achieved in this simulation without having to perform *any* remeshing, which would thicken the thin vortex layers and smear the fine scale internal structures of the horseshoe vortex cores observed in Figs. 2 and 4.

It should be mentioned here that while the surface mesh is symmetrically distributed, the particle distribution is certainly not symmetric or uniform. In fact, the VRM particle insertion strategy is one major contributor to this asymmetry. The question, then, is whether the flow remains reasonably symmetric. Figure 3 depicts the top view of selected streaklines about the discoid. There are seven distinct pairs of streaklines in the figure; each pair is seeded randomly at $(x, \pm y, z)$ on the two sides of the xz plane of symmetry and shares the same color for a quicker identification of the pair. Figure 3 shows the predicted flow to be symmetric, which is quite remarkable considering the irregularity of the particle distribution, the topological complexity of the flow, and the (relatively) late stage in the computation (without remesh). Note that the flow is not absolutely symmetric and that there are subtle differences between pairs of streaklines, which will undoubtedly grow as the simulation is extended to a much later stage. On the other hand, at Reynolds number 1,000, it is not realistic to expect a fully symmetric flow to persist for all times. Recall that the present algorithm has been shown to produce long-time simulation symmetry for flow over a sphere for up to Reynolds number 100, where the flow is known to be symmetric.¹⁰

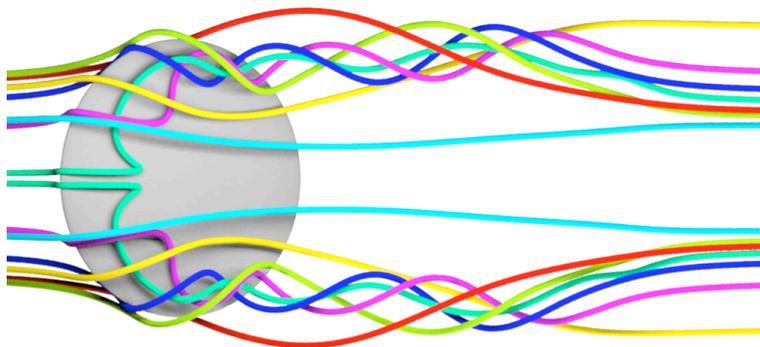


Figure 3. Top view of selected streaklines

Figure 4 depicts the vorticity field at various streamwise slices looking into the upstream, with the field rotated by 30 degrees so that the discoid is positioned horizontally. The $x = 0$ station slices through the center of the discoid. This figure shows, vividly, the dynamic adaptivity of the method and its tendency to focus the computational elements in regions with significant vorticity. Note that, as mentioned earlier, the formation of the horseshoe vortex starts at roughly half a radius downstream of the leading edge ($x \approx -0.25$). A discussion of the complex structure of the vortex core is beyond the scope of this paper, which is intended for demonstration of the salient features of the algorithm only. One such property is the preservation of the details of the vortex core with very little smearing, which is usually observed in (non-adaptive) grid-based simulations caused by excessive numerical diffusion. Notice that the wake experiences mild undulations in the vicinity of the plane of flow symmetry. This is clearly caused by numerical perturbations; however, although traced back to the surface of the discoid, the main source of this perturbation is not exactly obvious at this stage and requires further analysis.

IV. Conclusion

A hybrid Lagrangian Vortex-Boundary Element and Vorticity Redistribution Method is developed for the grid-free simulation of 3-D incompressible flow about complex objects. Computations are accelerated using a MPI-parallel adaptive fast oct-tree code. In this paper, the computational algorithm is described briefly, followed by a demonstration of the salient features of the approach using the example of the early stages of impulsively started

flow over a (lens shaped) circular discoid at a 30 degree angle of attack, and at Reynolds number 1,000 based on the freestream velocity and the discoid diameter. The simulation is shown to maintain stability and flow symmetry despite the fact that the vortex particles are distributed irregularly and that *no* remeshing is performed during the simulation. This provides partial confirmation of the claim that, by automatically inserting new particles as necessary, the Vorticity Redistribution Method for diffusion facilitates *truly* grid-free computations without having to resort to frequent remeshing. This is of course true for DNS-type simulations where diffusion is significant enough for the method to be effective.

Acknowledgments

This project was funded by NSF SBIR Phase II Grant No. 0060554.

References

- ¹Ploumhans, P., Winckelmans, G. S., Salmon, J. K., Leonard, A., and Warren, M. S., "Vortex Methods For Direct Numerical Simulation of Three-Dimensional Bluff Body Flows: Application To The Sphere At $Re = 300, 500, \text{ and } 1000$," *J. of Computational Physics*, Vol. 178, 2002, pp. 427-463.
- ²Degond, P., and Mas-Gallic, S., "The Weighted Particle Method For Convection-Diffusion Equations. Part I: The Case of An Isotropic Viscosity. Part II: The Anisotropic Case," *Mathematics of Computation*, Vol. 53, 1989, pp. 485-525.
- ³Barba, L. A., Leonard, A., and Allen, C. B., "Advances In Viscous Vortex Methods – Meshless Spatial Adaption Based On Radial Basis Function Interpolation," *International J. for Numerical Methods in Fluids*, Vol. 47, 2005, pp. 387-421.
- ⁴Shankar, S., and van Dommelen, L., "A New Diffusion Procedure For Vortex Methods," *J. of Computational Physics*, Vol. 127, 1996, pp. 88-109.
- ⁵Shankar, S., "A New Mesh-Free Vortex Method," *Ph.D. Thesis*, The Florida State University, 1996.
- ⁶Gharakhani, A., "Grid-Free Simulation of 3-D Vorticity Diffusion By A High-Order Vorticity Redistribution Method," in *15th AIAA Computational Fluid Dynamics Conference*, AIAA-2001-2640, Anaheim, CA, 2001, pp. 1-10.
- ⁷Cottet, G.- H., "Artificial Viscosity Models For Vortex Particle Methods," *J. of Computational Physics*, Vol. 127, 1996, pp. 299-308.
- ⁸Gharakhani, A., "A Grid-Free Method For LES of Incompressible Flow," in *The 2002 ASME Fluids Engineering Division Summer Meeting & Exhibition*, FEDSM2002-31370, Montreal, Canada, 2002.
- ⁹Koumoutsakos, P., Leonard, A., and Pepin, F., "Boundary Conditions For Viscous Vortex Methods," *J. of Computational Physics*, Vol. 113, 1994, pp. 52-61.
- ¹⁰Gharakhani, A., Sitaraman, J., and Stock, M. J., "A Lagrangian Vortex Method For Simulating Flow Over 3-D Objects," *Proceedings of 2005 ASME Fluids Engineering Division Summer Meeting and Exhibition*, FEDSM2005-77024, Houston, TX, June 2005.
- ¹¹Gharakhani, A., and Ghoniem, A. F., "Three Dimensional Vortex Simulation of Time Dependent Incompressible Internal Viscous Flows," *J. of Computational Physics*, Vol. 134, No. 1, 1997, pp. 75-95.
- ¹²Beale, J. T., and Majda, A., "Rates of Convergence For Viscous Splitting of the Navier-Stokes Equations," *Mathematics of Computation*, Vol. 37, No. 156, 1981, pp. 243-259.
- ¹³Gharakhani, A., and Ghoniem, A. F., "BEM Solution of the 3D Internal Neumann Problem and A Regularized Formulation For the Potential Velocity Gradients," *International J. for Numerical Methods in Fluids*, Vol. 24, No. 1, 1997, pp. 81-100.
- ¹⁴Wang, Q. X., "Variable Order Revised Binary Treecode," *J. of Computational Physics*, Vol. 200, 2004, pp. 192-210.

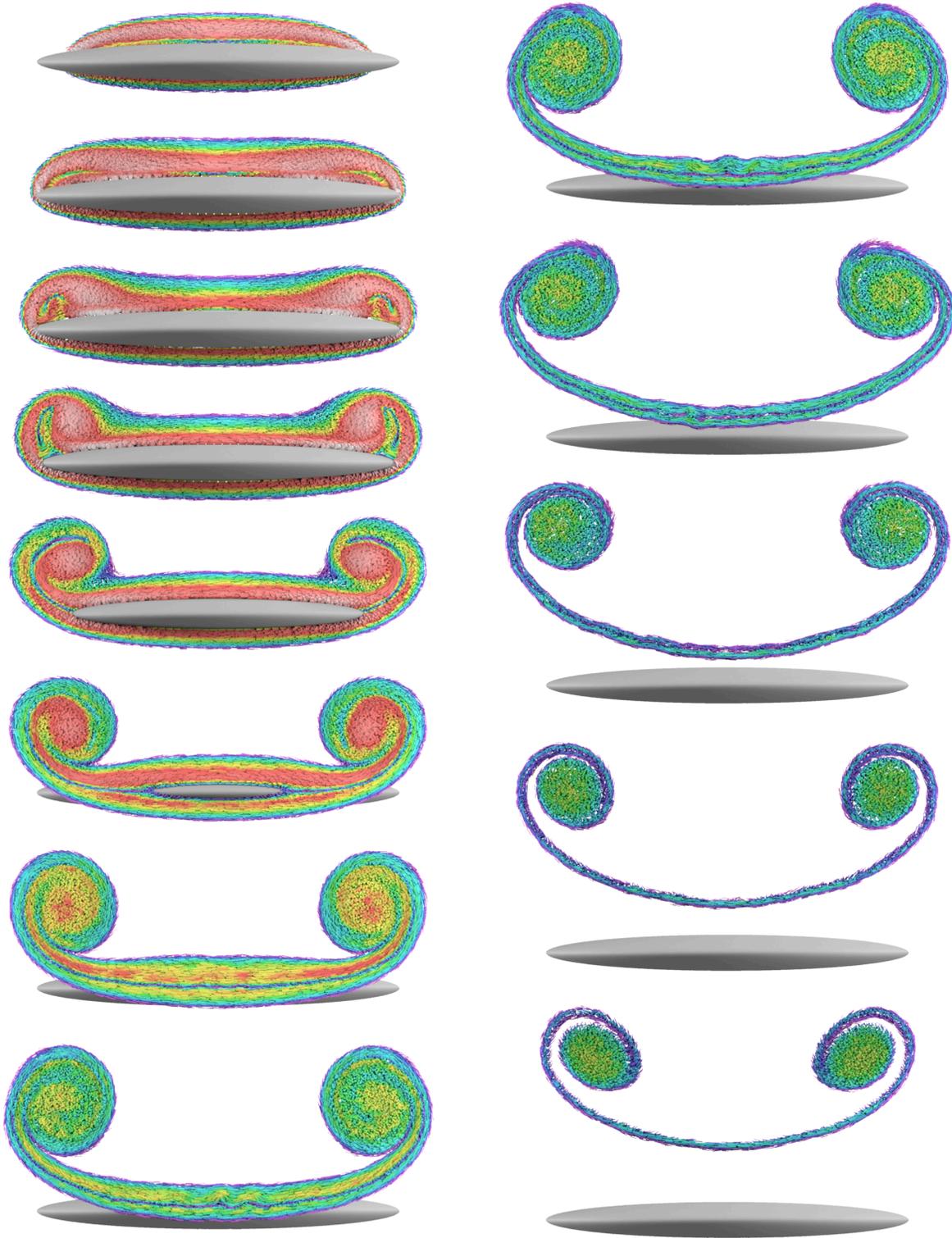


Figure 4. Selected slices of the vorticity field in the streamwise direction
 Top to bottom, left to right: $x = -0.5, -0.2, 0.0, 0.2, 0.5, 0.8, 1.1, 1.5, 1.8, 2.1, 2.5, 2.8, 3.1$