Journal of Computational Physics 228 (2009) 491–515

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A high resolution spatially adaptive vortex method for separating flows. Part I: Two-dimensional domains ☆

Issam Lakkis [a,*], Ahmed Ghoniem [b]

[a] Department of Mechanical Engineering, American University of Beirut, Beirut, Lebanon
[b] Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA

ARTICLE INFO

Article history:
Received 25 February 2008
Received in revised form 25 August 2008
Accepted 29 September 2008
Available online 17 October 2008

Keywords:
Incompressible flow
Numerical methods
Grid free
Vortex methods
Redistribution method
High resolution
Spatial adaptivity

ABSTRACT

A grid-free high-resolution spatially-adaptive vortex method for two-dimensional incompressible flow in bounded domains is presented. The computational algorithm is based on operator splitting in which convection and diffusion are handled separately every time step. In the convection step, computational elements are convected with velocities obtained by fast approximations of the Biot–Savart superposition with second-order Runge–Kutta time integration scheme. Diffusion is performed using the smooth redistribution method that employs a Gaussian basis function for vorticity in the interior. Near solid walls, the core functions are modified to conserve circulation. The no-slip boundary condition is enforced by creating of a vortex sheet that is redistributed to neighboring elements using the redistribution method. The proposed method enables accurate and smooth recovery of the vorticity and does not require explicit use of vortex images or occasional re-meshing. Algorithms for reduction in computational cost by accurately removing elements in overcrowded regions and for spatial adaptivity that allows for variable core sizes and variable element spacing are presented. Computations of flow around an impulsively started cylinder for Reynolds number values of 1000, 3000, and 9500 are preformed to investigate various aspects of the proposed method.

1. Introduction

Vortex methods [10,27,11] are grid-free Lagrangian computational methods originally devised for simulating incompressible fluid flow at large Reynolds number [8]. Presently, vortex methods are capable of handling complex geometries and the associated boundary conditions as well viscous diffusion over a large range of Reynolds number. In these methods, the vorticity field is discretized using vortex elements or "blobs". Operator splitting of the vorticity equation enables convection and diffusion of vorticity to be numerically carried out as separate steps. Convection is simulated by transporting conserved quantities such as circulation along particles' trajectories, where the particles velocities are obtained using a Biot–Savart summation over all the computational elements. Several methods have been developed for solving the diffusion equation including random walk [8], core expansion [24,27], particle strength exchange (PSE) method [16], and redistribution methods [35,26].

Vortex methods have been successfully used to investigate the evolution of vortex sheets [7,36,18], high Reynolds number wakes [38,6], three-dimensional problems [34,21,31,17,1], non-reacting buoyant plumes [19], reacting flows in shear layers [37], co-axial jets [30], and fires [25,19]. High resolution spatially adaptive vortex methods [22,33,3,14] have been

☆ Supported by the American University of Beirut, URB.
* Corresponding author.
E-mail address: issam.lakkis@aub.edu.lb (I. Lakkis).

0021-9991/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2008.09.025

developed and used for simulation of separating flows and accurate evaluation of lift and drag. Cottet et al. [14] employed variable vortex blobs and associated spatial adaptivity by introducing a mapping between the spatially varying physical domain and a uniform mapped domain. The PSE scheme for diffusion is carried out in the mapped domain. Ploumhans et al. [33] reported results of high resolution simulations of flow over bluff bodies including a cylinder, a square and the 2-D "Apollo" capsule. The variable resolution method of [33] is based on high-order redistribution schemes in the presence of a solid boundary in the context of the PSE scheme. The no-slip boundary condition is enforced by creating a vortex sheet that cancels the slip at the boundary which is then accurately diffused into the flow domain in a conservative manner. Spatial adaptivity is made possible by mapping of the redistribution onto a non-uniform grid which is coarser away from the solid boundary. Barba et al. [3] employed radial basis function (RBF) interpolation techniques to spatial adaption of Lagrangian vortex particles. Core spreading was employed for diffusion with core size control enforced during the adaption process. The adaption process is essentially a mapping to a new set of particles that does not necessarily have to be uniformly spaced. To accurately capture flow separation from a boundary as well as the associated small flow features, near-boundary diffusion should be modeled accurately and the no-slip boundary condition should be properly enforced. Spatial adaptivity is another feature that reduces the computational cost and yet maintains high resolution near the boundary.

Aspects of high resolution vortex methods include (i) fast (multipole) solvers [20,4], (ii) accurate diffusion using the Particle Strength Exchange (PSE) scheme [15,16] and the redistribution method [35,26], (iii) accurate enforcement of the no-slip boundary condition for viscous flow [23,32,28,5], and (iv) spatial adaptivity of elements positions [14,12,11]. Vortex methods primarily differ in the manner they handle the viscous sub-step. The PSE scheme approximates the diffusion operator by an integral which is discretized among neighboring elements. PSE offers spatial adaptivity by remapping elements with variable core size onto uniform blobs [13,14]. Error control in the PSE requires re-meshing, i.e. interpolation from a scattered set of elements to a pre-described mesh. Barba et al. [2,3] discuss the errors incurred by re-meshing and propose a "completely meshless" and spatially adaptive method based on radial basis function (RBF) interpolation.

This paper is organized as follows. First, the vortex method in two dimensions and the no-slip boundary condition implementation are reviewed in Section 2. The smooth redistribution method is then discussed in Section 3 for both flow elements of variable cores as well as boundary sheet elements. Section 4 presents novel algorithms for reduction in the number of elements and for spatial adaptivity using elements with variable cores and variable spacings. The computational algorithm is presented in Section 5. Computing the lift and drag coefficients is discussed in Section 6. In Section 7, various parameters and aspects of method are investigated in terms of the canonical problem of the flow over an impulsively started cylinder and uniform flow over an oscillating cylinder. In this respect, the impact of (i) core-size elements'-spacing overlap, (ii) time step size, and (iii) redistribution length and spatial adaptivity are discussed. Conclusion and future work are finally presented.

2. Two-dimensional vortex method

Vortex methods employ the velocity-vorticity formulation by numerically solving the vorticity transport equation according to the viscous splitting algorithm. In its basic form, this algorithm consists of successive handling of convection and diffusion of vorticity in each time step as follows:

$$\text{convection (inviscid) sub-step} \quad \frac{d\boldsymbol{\omega}}{dt} = 0 \tag{1}$$

$$\text{diffusion (viscous) sub-step} \quad \frac{\partial \boldsymbol{\omega}}{\partial t} = \nu \nabla^2 \boldsymbol{\omega} \tag{2}$$

The vorticity field is approximated by the superposition

$$\omega(\mathbf{x}, t) = \sum_{i=1}^{N} \Gamma_i(t) f_{\sigma_i}(\mathbf{x}, \mathbf{x}_i) \tag{3}$$

where N is the number of vortex elements and f_σ is the basis function of core radius σ. In the convection sub-step, each element is convected according to its velocity which, in the reference frame of a solid body moving with translational velocity $\mathbf{u}_b$, is computed according to the Helmholtz decomposition

$$\mathbf{u} = \mathbf{u}_\omega + \mathbf{u}_\Omega + \mathbf{u}_\infty - \mathbf{u}_b + \mathbf{u}_{ext} \tag{4}$$

where $\mathbf{u}_\omega$ is the vortical velocity component in free space, $\mathbf{u}_\Omega$ is the velocity due to the solid body rotation at an angular speed of Ω, $\mathbf{u}_\infty$ is the free stream velocity, and $\mathbf{u}_{ext}$ is selected such that the no-through flow boundary condition at the solid boundary is satisfied. Depending on the method of enforcing the no-through flow boundary condition, $\mathbf{u}_{ext}$ is the velocity field due to either (i) a vorticity sheet (γ_+) [23,32,28,5] at the solid boundary inside the flow field as depicted in Fig. 1a, or (ii) a potential sheet (q_-) at the solid boundary inside the solid as depicted in Fig. 1b, or (iii) images of vortex elements ($\omega^\star$) along with a potential sheet $q_{\infty-}$ that cancels the normal component of $\mathbf{u}_\infty - \mathbf{u}_b$ at the solid boundary, as depicted in Fig. 1c [35].

In vortex methods, satisfying the no-slip boundary condition poses a challenge since it is not explicitly a boundary condition for the vorticity. In the context of the viscous splitting algorithm, the no-through flow boundary condition at solid

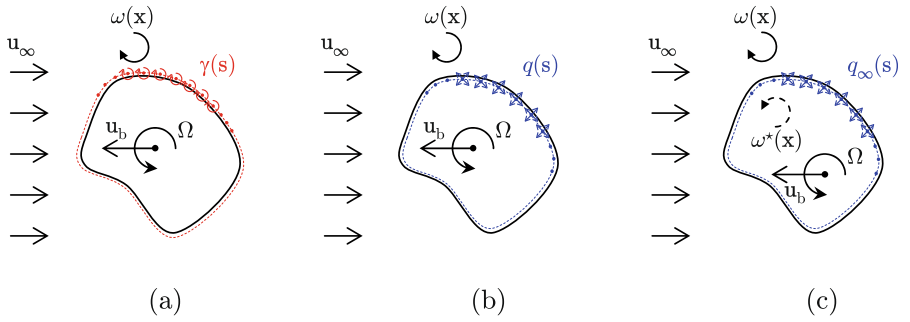

Fig. 1. Methods for enforcing the no-through flow boundary condition.

boundary is enforced in the inviscid step. Next, diffusion is carried out using the smooth redistribution method. The no-slip velocity boundary condition on solid wall is then imposed. Various approaches rely in principle on Lighthill's model [29] in which the no-slip boundary condition is physically manifested by creation of vorticity at the boundary such that it cancels the slip velocity induced by the previous convection step. Rigorous treatment of this requirement focuses on replacing the no-slip boundary condition with an equivalent vorticity boundary condition that relates vorticity production at the solid boundary to the slip velocity. In this regard, Chorin [9] created vortex blobs at the boundary to cancel the slip velocity. In order to overcome vorticity leakage outside the flow domain and artificial diffusion near the boundary, he later proposed a vortex sheet method coupled with Prandtl boundary layer equations near boundary. This latter method suffers in accuracy in regions of flow separation from the boundary. To diffuse the vortex sheets/blobs, Chorin employed random walk and reflected back into the flow those sheets that cross the boundary. Cottet and Koumoutsakos [11] proposed a formulation in which the no-slip boundary condition is satisfied in the context of fractional step algorithms by replacing the no-slip boundary condition with an equivalent vorticity flux boundary condition $\nu\partial\omega/\partial n = -\partial(\mathbf{u}\cdot\hat{\mathbf{s}})/\partial t$. This formulation combines viscous diffusion and the no-slip boundary condition in the sense that the generated vortex sheet $\mu(\mathbf{x}, t+\Delta t)$ satisfies the no-slip condition of the diffused interior vorticity with zero vorticity flux at the boundary S

$$-\frac{1}{2}\mu(\mathbf{x}, t+\Delta t) + \nu\int_t^{t+\Delta t}\int_S \frac{\partial}{\partial n}G(\mathbf{x}-\mathbf{x}', t-\tau)\mu(\mathbf{x}', t)d\mathbf{x}'d\tau = -\frac{\partial}{\partial t}(\mathbf{u}\cdot\hat{\mathbf{s}}) \tag{5}$$

Time integration of Eq. (5) was implemented in [11] using the midpoint rule and was later improved [28,33] by accounting for more accurate time integration using Gauss quadrature and for exact contribution over nearby particle volumes.

In this paper, it is proposed to redistribute the double layer potential obtained by solving Eq. (5) to existing neighboring elements in the interior by conserving moments of vorticity in a manner compatible with the redistribution method for diffusion. Depending on the spacing of the neighboring elements, new elements may be injected near the boundary. Vorticity leakage outside the fluid domain is avoided by a proper selection of the basis function for vorticity. The basis function chosen for this purpose is

$$f_\sigma(\mathbf{x}-\mathbf{x}_0) = \frac{1}{\pi\sigma^2}\left[e^{-\frac{(x-x_0)^2+(y-y_0)^2}{\sigma^2}} + e^{-\frac{(x-x_0)^2+(y+y_0)^2}{\sigma^2}}\right]H(y) \tag{6}$$

where $H(\,)$ is the Heaviside function and the coordinates (x, y) are chosen to be centered at the closest point on the boundary to the element with y normal to the boundary pointing into the fluid. Such a selection guarantees that the vorticity is always contained in the fluid domain if the local radius of curvature R of the solid boundary is larger than twice the core radius of the vortex element. This requirement is in accordance with the requirement of using smaller boundary elements to capture larger variations of the boundary as well as the requirement to have a smaller diffusion length scale and element spacing for accurate diffusion and convection in these regions. The accuracy of Eq. (6) in terms of conserving vorticity is assessed by considering the error in the vorticity gradient normal to a boundary element of size Δs at distance y_0 from a vortex element of core radius σ. The local radius of curvature at the center of the boundary element is R. The error, given by

$$\pi\sigma^2\int_{\Delta s}\left(\frac{\partial f}{\partial r}\right)_{r=R} R d\theta = -2\frac{R}{\sigma}\frac{\Delta s}{\sigma}e^{-\frac{R^2}{\sigma^2}}\left(e^{-\frac{(R+y_0)^2}{\sigma^2}} + e^{-\frac{(R-y_0)^2}{\sigma^2}}\right) + O(\Delta s^2), \tag{7}$$

essentially indicates that error is negligible for $R \geqslant 4\sigma$ or $y_0 \geqslant 4\sigma$. Therefore an element near the boundary must have a core radius less that one quarter of the local radius of curvature of the boundary. As the distance y_0 from the boundary increases beyond R, the element may have a larger core radius, which in view of the overlap requirement, allows for coarser spacing of elements away from the boundary.

The following algorithm is employed to satisfy the no-slip boundary condition for the viscous sub-step from t to $t+\Delta t$ starting with $\omega_0(\mathbf{x}, t)$:

(1) Solve the diffusion problem (from t to $t+\Delta t$) by satisfying the no-flux boundary condition. This is performed using the redistribution scheme among elements of core function given by Eq. (6), as outlined in Section 3.1.

(2) Calculate the slip velocity at the boundary.
(3) Satisfy the no-slip boundary condition by generating a diffused vortex sheet described by Eq. (5).
(4) Redistribute the vortex sheet into interior vortex elements. Diffusion of vortex sheet elements by redistribution to interior vortex elements requires special reformulation of the redistribution method, as described in Section 3.2.

3. The smooth redistribution method

In the smooth redistribution method, diffusion of the element vorticity is represented by transferring fractions of its circulation to neighboring elements in such a way that various moments of the diffusion equation are conserved. The smooth redistribution method [26] differs from the original redistribution method [35] in assuming a smooth core function rather than the singular Dirac-delta function for the vorticity distribution associated with an element. With sufficient overlap between neighboring elements, the smooth redistribution method allows for accurate and smooth recovery of the vorticity field. In unbounded domains, the core function of element located at $\mathbf{x}_0$ is selected to be the Green function of the diffusion equation

$$f_\sigma(\mathbf{x}-\mathbf{x}_0)=\frac{1}{\pi\sigma^2}e^{-\frac{|\mathbf{x}-\mathbf{x}_0|^2}{\sigma^2}} \tag{8}$$

where σ is the core radius of the element. Such a choice for the core function enables diffusion of an element by expanding its core according to $\sigma(t+\Delta t)=\sqrt{\sigma(t)^2+4\nu\Delta t}$. Diffusion by core expansion endows the smooth redistribution method with extra flexibility in terms of controlling the core sizes of the elements. Note that the core size has to be large enough to guarantee overlap but not too large in which case the convection sub-step loses accuracy.

3.1. Diffusion of vortex elements

In this section, redistribution equations governing diffusion of a vortex element inside the flow of a bounded domain are presented. The core function is selected to be the Green's function of the diffusion equation in a semi-infinite domain. Essentially, it is assumed that each vortex element diffuses in a semi-infinite domain whose boundary coincides with the nearest boundary element. In the local coordinates of the boundary element (panel) nearest to the vortex element, the core function is given by Eq. (6). The vorticity distribution for element of circulation Γ_0, position $\mathbf{x}_0$, core radius σ_0 is then given by

$$\omega(\mathbf{x},\mathbf{x}_0)=\Gamma_0 f_{\sigma_0}(\mathbf{x}-\mathbf{x}_0) \tag{9}$$

The objective is to diffuse the element by transferring fractions of its circulation to neighboring elements in such a manner that the various moments of vorticity are conserved

$$\int_{-\infty}^{\infty}\int_{0}^{\infty}x^m y^n\sum_{i=1}^{M}\omega(\mathbf{x},\mathbf{x}_i)\,dx\,dy=\int_{-\infty}^{\infty}\int_{0}^{\infty}x^m y^n\tilde{\omega}(\mathbf{x},\mathbf{x}_0)\,dx\,dy \tag{10}$$

where M is the number of neighbors involved in the redistribution process, m and n are integers that assume values of 0, 1, 2, etc. and $\tilde{\omega}$ is the vorticity distribution of the element after diffusion over a time step of Δt, which is given by replacing σ_0^2 with $\tilde{\sigma}_0^2\equiv\sigma_0^2+4\nu\Delta t$ in Eq. (6). For $m+n\leqslant 2$, the linear system governing fractions f_i of circulation Γ_0 that must transferred to neighbors i is

$$\sum_{i=1}^{M}f_i=1 \tag{11}$$

$$\sum_{i=1}^{M}x_i f_i=x_0 \tag{12}$$

$$\sum_{i=1}^{M}\left(\frac{\sigma_i}{\sqrt{\pi}}e^{-y_i^2/\sigma_i^2}+y_i\operatorname{erf}\left(\frac{y_i}{\sigma_i}\right)\right)f_i=\frac{\tilde{\sigma}_0}{\sqrt{\pi}}e^{-y_0^2/\tilde{\sigma}_0{}^2}+y_0\operatorname{erf}\left(\frac{y_0}{\tilde{\sigma}_0}\right) \tag{13}$$

$$\sum_{i=1}^{M}\left(\frac{\sigma_i^2}{2}+x_i^2\right)f_i=\frac{\tilde{\sigma}_0^2}{2}+x_0^2 \tag{14}$$

$$\sum_{i=1}^{M}\left(\frac{\sigma_i^2}{2}+y_i^2\right)f_i=\frac{\tilde{\sigma}_0^2}{2}+y_0^2 \tag{15}$$

$$\sum_{i=1}^{M}x_i\left(\frac{\sigma_i}{\sqrt{\pi}}e^{-y_i^2/\sigma_i^2}+y_i\operatorname{erf}\left(\frac{y_i}{\sigma_i}\right)\right)f_i=x_0\left(\frac{\tilde{\sigma}_0}{\sqrt{\pi}}e^{-y_0^2/\tilde{\sigma}_0{}^2}+y_0\operatorname{erf}\left(\frac{y_0}{\tilde{\sigma}_0}\right)\right) \tag{16}$$

For elements away from the boundary, i.e. for $y_0\gg\sigma_0$, the above system corresponds to the smooth redistribution scheme in unbounded domains [26]. Diffusion of a vortex element using the redistribution scheme guarantees uniformity in its neighbors spacing by injecting new elements from a set of injection candidates uniformly distributed on a circle centered at the element.

The radius of this circle (injection radius) is equal to the desired elements spacing. This conforms with the fact that the moments equations will yield positive fractions for the neighbors within the search radius only when this spacing criterion is met.

As an example, the following two cases are considered: diffusion of point vortex located at $x_0 = 0,\ y_0 = 1.5$ and diffusion of point vortex located at $x_0 = 0,\ y_0 = 1.01$ in the exterior of a solid cylinder centered at $(0,0)$ of radius $R = 1$ subject to zero vorticity flux $\partial\omega/\partial n = 0$ at the cylinder boundary. The parameters of the redistribution scheme are: injection radius of $\sqrt{6\nu\Delta t}$, search radius of $\sqrt{12\nu\Delta t}$, core radius of $\sigma = \sqrt{24\nu\Delta t}$. The number of candidates for injection is 40. The diffusion time step is $\nu\Delta t = 0.0001$. Solving the linear system comprising the redistribution equations (11)–(16) is performed using a non-negative least square solver with L_2 accuracy of 10^{-6}. Up to the fourth moment are conserved $(m + n \leqslant 4)$ leading to 15 equations. The number of unknowns is equal to the number of neighbors within the specified search radius. Fig. 2 shows vorticity contours for the two cases at two times $\nu t = 0.01$ and $\nu t = 0.05$. Solid lines correspond to (A) redistribution according to the scheme described above whereas dashed line correspond to (B) using the smooth redistribution scheme in unbounded domain followed by reflecting into the flow those elements that are injected inside the solid cylinder during redistribution. As can be seen from Fig. 2, the scheme proposed in this paper provides slightly improved accuracy near the boundary. Note that scheme (B) is derived from Shankar [35], with the difference that Shankar used point vortices instead of vortex blobs in addition to maintaining a ring of uniformly spaced vortices near the solid boundary to improve accuracy near the solid boundary.

3.2. Diffusion of vortex sheet

In this section, redistribution equations governing diffusion of a vortex sheet element of circulation $\Gamma = \gamma\Delta s$ are presented, where γ is the vortex sheet strength and Δs is the element length. The objective in this case is to replace the sheet element with vortex blobs. In the local coordinates (x, y) of the sheet element (with y normal to boundary), the vorticity distribution of the sheet element is

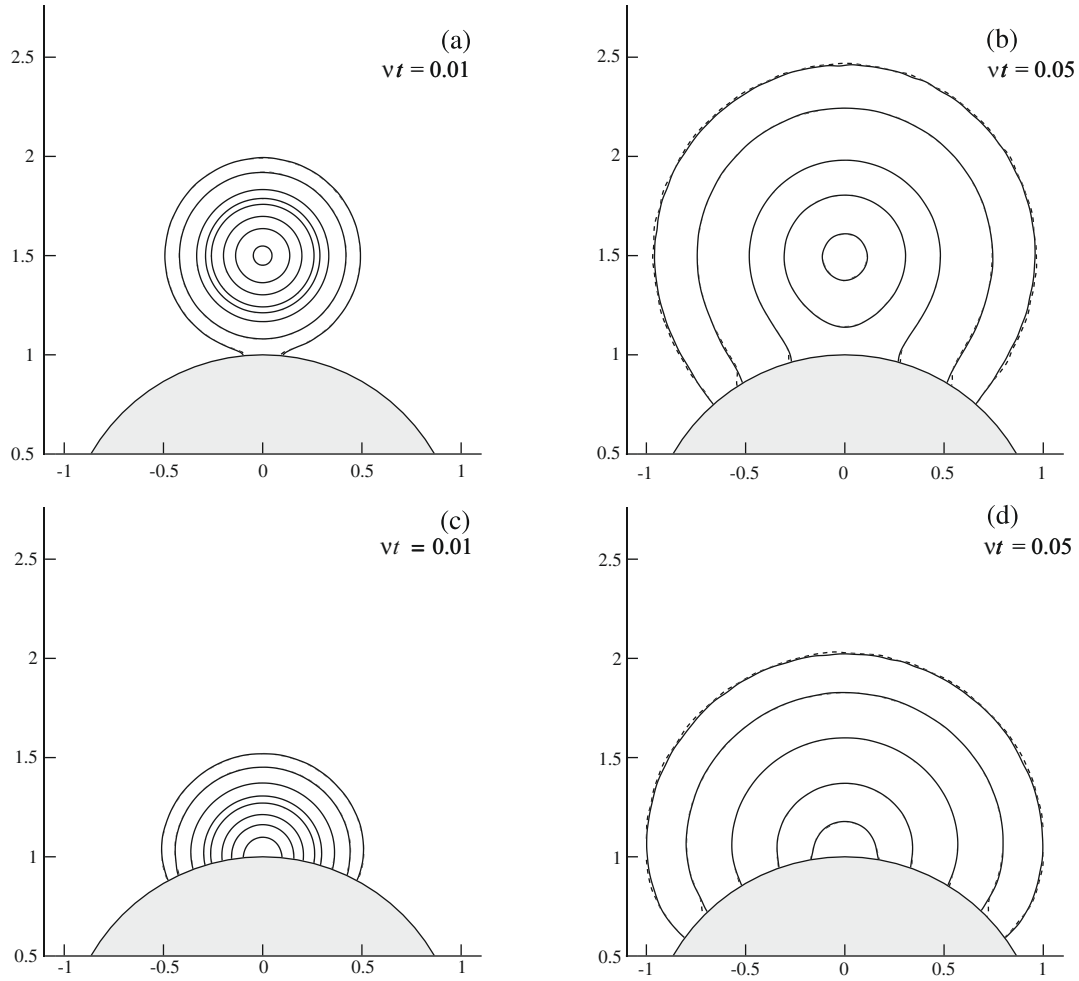

Fig. 2. Diffusion of a point vortex at $(0, 1.5)$: iso-vorticity contours at (a) $\nu t = 0.01$ and (b) $\nu t = 0.05$. Diffusion of a point vortex at $(0, 1.01)$: iso-vorticity contours at (c) $\nu t = 0.01$ and (d) $\nu t = 0.05$. Solid lines correspond to current work. Dashed lines correspond to smooth redistribution with reflection.

$$\omega(\mathbf{x},t)=\frac{\Gamma}{\Delta s}\frac{e^{-\frac{y^2}{4\nu\Delta t}}}{\sqrt{4\pi\nu\Delta t}}\left[\operatorname{erf}\left(\frac{x+\Delta s/2}{\sqrt{4\nu\Delta t}}\right)-\operatorname{erf}\left(\frac{x-\Delta s/2}{\sqrt{4\nu\Delta t}}\right)\right] \tag{17}$$

is to be replaced with vortex blobs $(\Gamma_i,\sigma_i,(x_i,y_i))$ of vorticity distribution

$$\omega(\mathbf{x},\mathbf{x}_i,t)=\frac{\Gamma_i}{\pi\sigma_i^2}\left[e^{-\frac{(x-x_i)^2+(y-y_i)^2}{\sigma_i^2}}+e^{-\frac{(x-x_i)^2+(y+y_i)^2}{\sigma_i^2}}\right]H(y) \tag{18}$$

where (x,y) are the local coordinates centered at the boundary panel nearest to (x_i,y_i) with y normal to panel. Conserving zeroth, first and second moments of vorticity results in the following linear system governing fraction f_i of circulation Γ that must be transferred to neighboring vortex blobs $i=1,\ldots,M$

$$\sum_{i=1}^{M}f_i=1 \tag{19}$$

$$\sum_{i=1}^{M}x_if_i=0 \tag{20}$$

$$\sum_{i=1}^{M}\left(\frac{\sigma_i}{\sqrt{\pi}}e^{-y_i^2/\sigma_i^2}+y_i\operatorname{erf}\left(\frac{y_i}{\sigma_i}\right)\right)f_i=\sqrt{\frac{4\nu\Delta t}{\pi}} \tag{21}$$

$$\sum_{i=1}^{M}\left(\frac{\sigma_i^2}{2}+x_i^2\right)f_i=\frac{\Delta s^2}{12}+\frac{4\nu\Delta t}{2} \tag{22}$$

$$\sum_{i=1}^{M}\left(\frac{\sigma_i^2}{2}+y_i^2\right)f_i=\frac{4\nu\Delta t}{2} \tag{23}$$

$$\sum_{i=1}^{M}x_i\left(\frac{\sigma_i}{\sqrt{\pi}}e^{-y_i^2/\sigma_i^2}+y_i\operatorname{erf}\left(\frac{y_i}{\sigma_i}\right)\right)f_i=0 \tag{24}$$

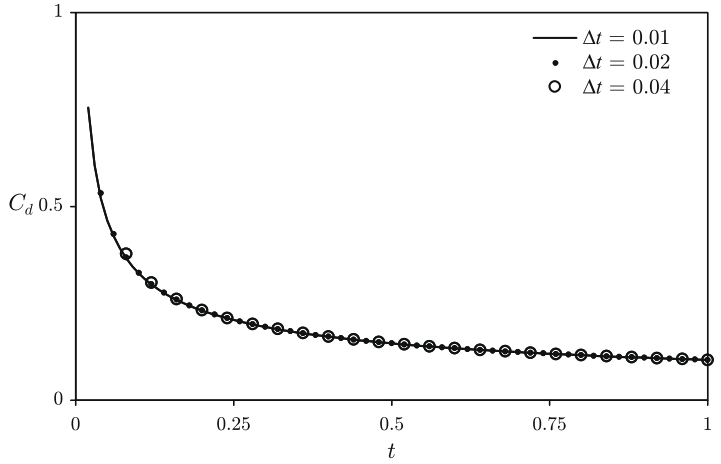

Fig. 3. Drag coefficient versus time showing impact of time step for the case of vortex sheet generation and diffusion with no convection for $\mathrm{Re}=9500$.

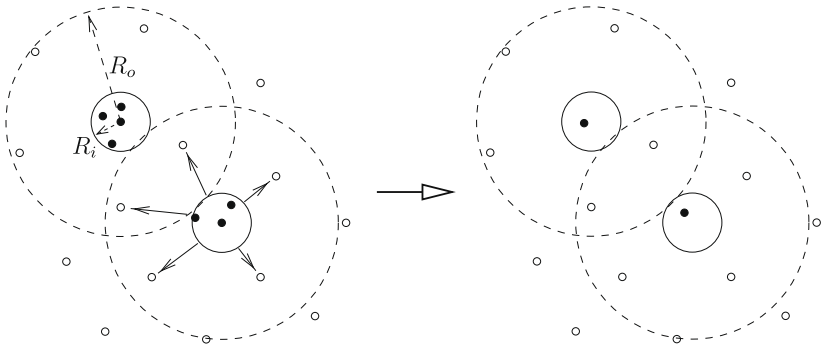

Fig. 4. Removal of elements in overcrowded regions.

In redistributing a sheet element, fractions of its circulation are transferred to existing neighboring (interior) vortex elements. Similar to diffusion of an interior vortex blob, new blobs are injected in a manner that guarantees uniformity in elements' spacing. The sheet element does not belong to the set of neighbors so that it carries zero circulation after redistribution. If blobs are selected to be on the sheet element centered at $(0,0)$ with y as normal, then three blobs with $\sigma = \sqrt{4\nu\Delta t}$ are sufficient to satisfy the above system with $f_1 = 2/3,\ x_1 = 0,\ y_1 = 0,\ f_2 = 1/6,\ x_2 = -\Delta s/2,\ y_2 = 0$ and $f_3 = 1/6,\ x_3 = \Delta s/2,\ y_3 = 0$.

As an example, the problem of vorticity generation and diffusion for an impulsively started cylinder is considered. In this experiment, vortex elements are diffused but not convected. Every time step, elements are diffused using the redistribution scheme described in the previous section and the no-slip boundary condition is enforced by vorticity generation at the boundary. The injection radius, search radius and core radius are same as in the previous example. Fig. 3 shows the drag coefficient versus time for the case of $Re = 9500$ for different values of the time step Δt. For small times, diffusion is dominant and the elements have not yet diffused enough to experience significant convection. As a result, even though convection is not included in these computations, the drag coefficients presented in Fig. 3 are accurate at small times. Comparisons with the drag coefficient for the flow over an impulsively started cylinder (including convection) are presented in Section 7.

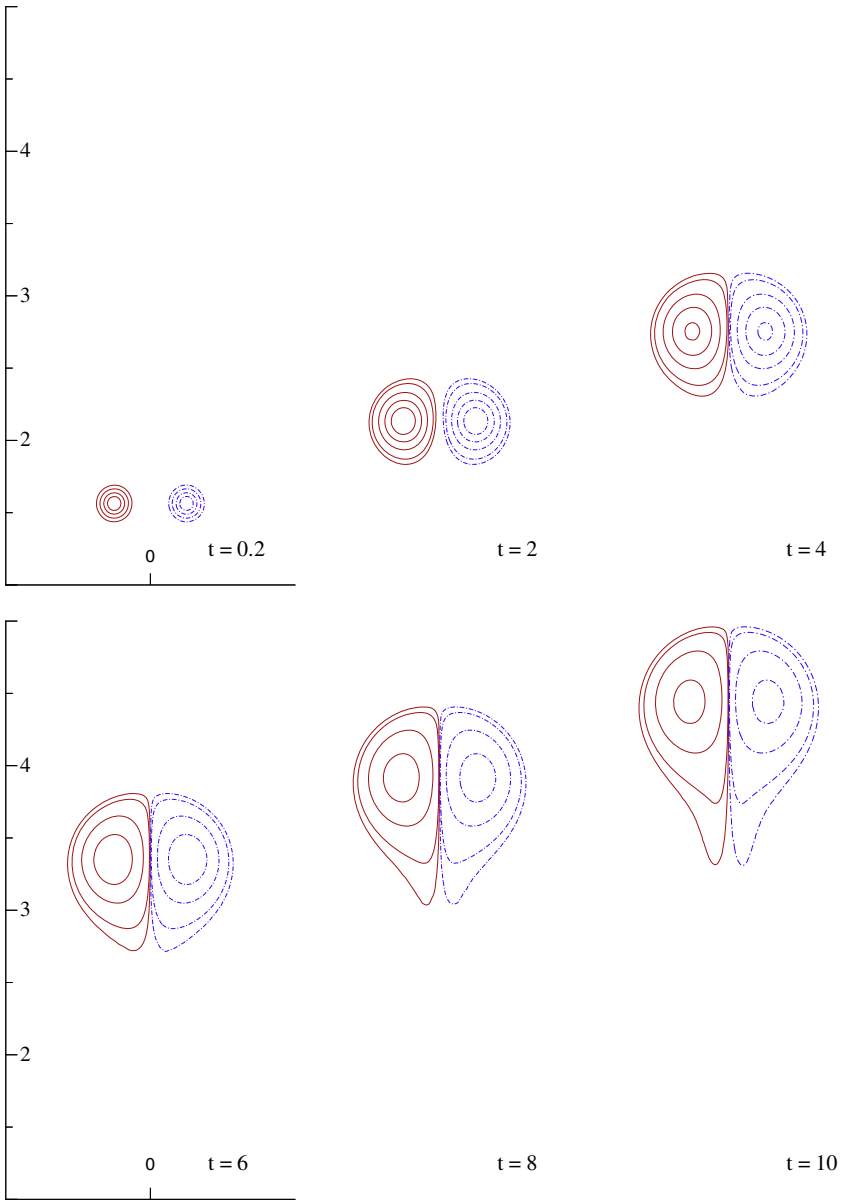

Fig. 5. Vorticity contours for two counter-rotating vortices at $t = 0.2, 2, 4, 6, 8$ and 10.

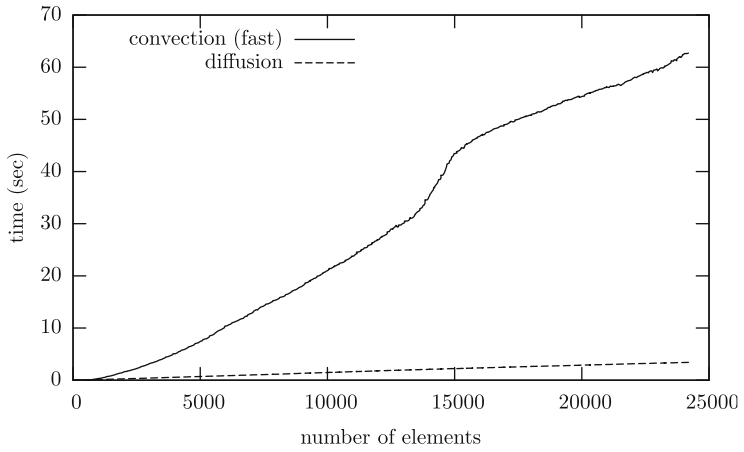

Fig. 6. CPU times consumed by the smooth redistribution scheme versus the fast multipole ($N \log N$) convection scheme versus the number of elements.

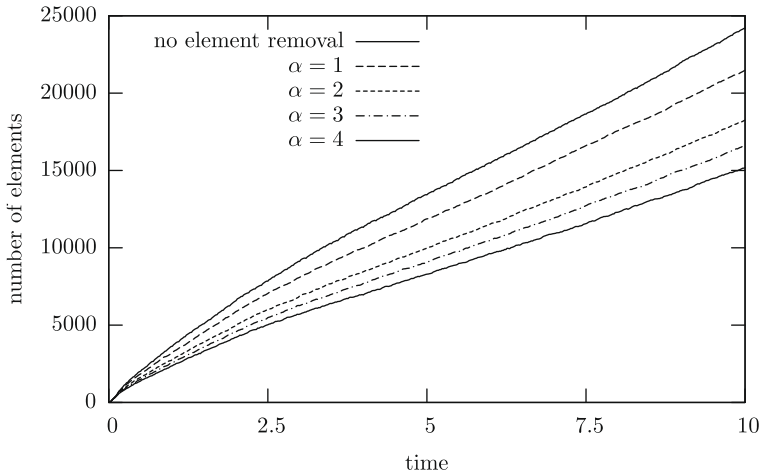

Fig. 7. Number of elements versus time for different redistribution radii.

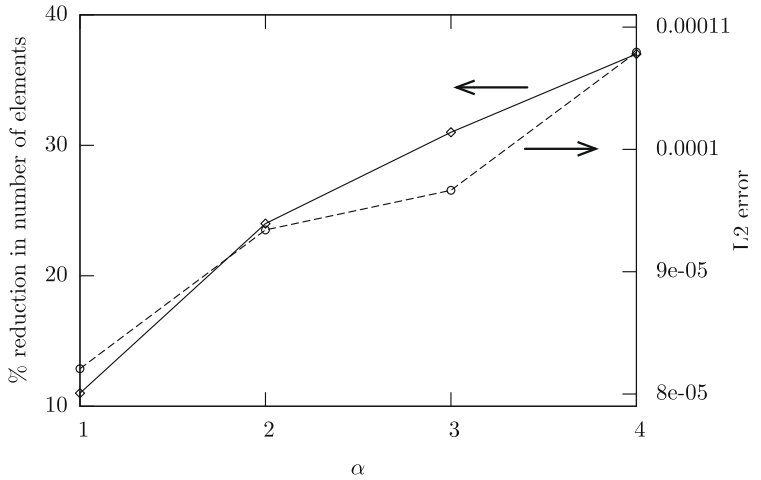

Fig. 8. Reduction in the number of elements and the corresponding error versus redistribution radius at $t = 10$.

4. Algorithms for reducing the number of elements and variable-core variable-spacing spatial adaptivity

The redistribution scheme for diffusion, if implemented properly, creates the minimum possible number of elements for the desired accuracy. This is because injection of new elements on the injection circle maximizes the distance to the closest existing neighbor. However, convection at high Reynolds number and particularly in regions of large strain rates, the uniformity in elements' spacing, which is favored but not enforced by the redistribution scheme, deteriorates. This is essentially one of the main reasons behind the occasional re-meshing in the PSE scheme. Instead of re-meshing, it is proposed in this paper to solve this problem by redistributing the circulations of elements within an inner search radius R_i to the neighbors within a search radius R_o, as depicted in Fig. 4. The elements removal and strengths redistribution are performed using the same scheme and computational algorithm employed for diffusion, which is discussed in Section 5. If N_i is the number of elements to be removed (within circle of radius R_i) and N_o is the number of elements in the annulus between the circles of radii R_o and R_i, then removal of the N_i elements must satisfy the conservation of the various moments of vorticity according to

$$\int\int x^m y^n \sum_{i=1}^{N_i+N_o} \omega(\mathbf{x},\mathbf{x}_i)dx\,dy = \int\int x^m y^n \sum_{i=1}^{N_o} \omega(\mathbf{x},\mathbf{x}_i)dx\,dy + \int\int x^m y^n \omega(\mathbf{x},\mathbf{x}_0)dx\,dy$$

In the process shown in Fig. 4 all the elements in the inner circle are removed and a new element of circulation Γ_0 is created at the center of vorticity $\mathbf{x}_0$ of the removed elements, i.e. $\Gamma_0 = \sum_{i=1}^{N_i}\Gamma_i$ and $\int\int x^m y^n \sum_{i=1}^{N_i}\omega(\mathbf{x},\mathbf{x}_i)dx\,dy = \int\int x^m y^n \omega(\mathbf{x},\mathbf{x_0})dx\,dy$ for $m=1,\ n=0$ and $m=0,\ n=1$. This guarantees satisfaction of zeroth- and first-order moments resulting in minimal modification of strength of elements in the outer circle as well as minimal injection of new elements during the process. To illustrate the impact of the proposed algorithm, we consider the convection–diffusion problem of two counter-rotating vortices of circulations $\Gamma_1 = 1$ and $\Gamma_2 = -1$ with initial positions of $(-0.25, 0)$ and $(0.25, 0)$, respectively. The initial vorticity distributions are given by Eq. (8) with $\sigma_0 = \sqrt{16\nu\Delta t}$, with $\nu = 0.001$ and $\Delta t = 0.02$. Simulations were performed with $R_i = \sqrt{\alpha\nu\Delta t}$ for $\alpha = 0, 1, 2, 3$ and 4. For the case with $\alpha = 0$, merging of elements within a neighborhood radius of $\sqrt{0.5\nu\Delta t}$ was performed every 5 time steps. The merging was conducted in a manner that the zeroth and first moments of vorticity are conserved, as proposed in [35].

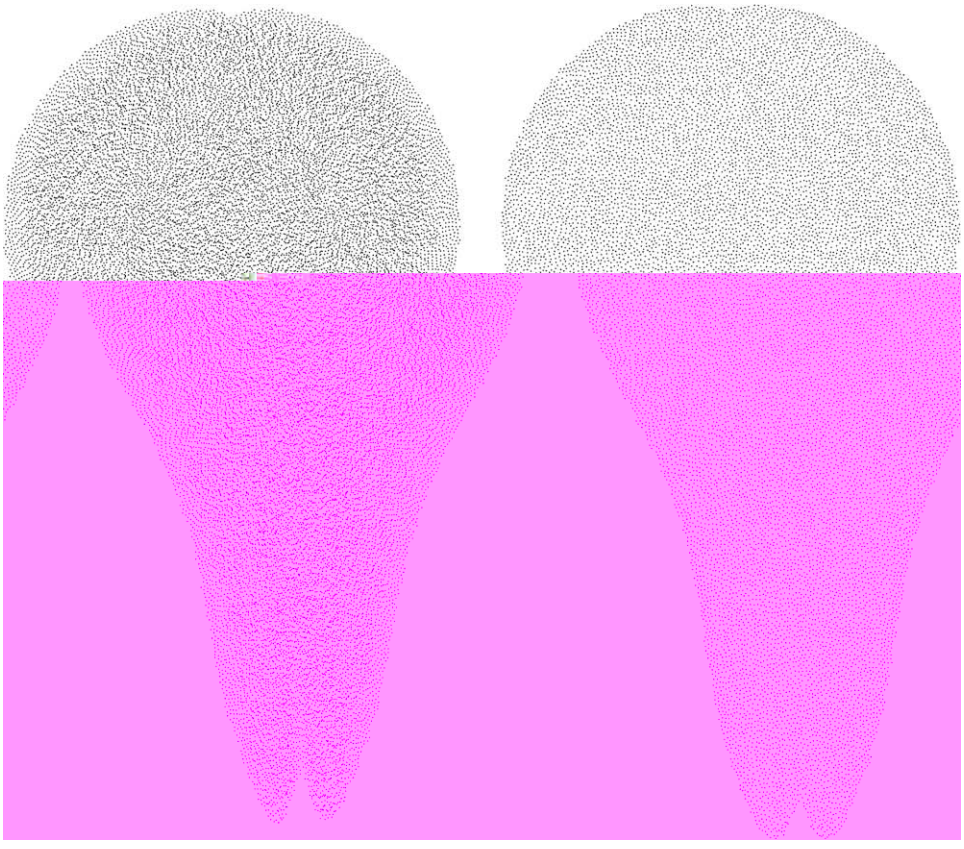

Fig. 9. The computational elements position at $t = 10$ for (left) $\alpha = 0$ and (right) $\alpha = 4$.

Vorticity contours are shown in Fig. 5 at times $t = 0.2, 2, 4, 6, 8$ and 10, respectively. Solid line patterns correspond to positive vorticity and dashed line patterns correspond to negative vorticity. The mutual interaction between the two vortices is such that each vortex pushes the other upwards. The CPU time consumed by the diffusion step and that consumed by the convection step are presented in Fig. 6 versus the number of elements. Note that except for small number of elements, the fraction of time consumed by the smooth redistribution scheme is very small compared to the time consumed by an $N \log N$ fast multipole scheme for convection. For example, for $N = 24,189$ elements, diffusion by redistribution consumed 2.4 CPU seconds whereas the velocity computation consumed 63 CPU seconds. So if convection is performed using a second-order Runge–Kutta integration scheme, then the diffusion step consumes less than 2% of the time consumed by the convection–diffusion time step.

Reduction in the number of elements for different values of α is shown in Fig. 7. In the figure, the number of element versus time is plotted for $\alpha = 0, 1, 2, 3$ and 4. Note that the growth in the number of elements with time is linear, a characteristic of the smooth redistribution scheme. The percent reduction in the number of elements versus the redistribution radius is presented in Fig. 8 at $t = 10$. On the same plot is shown the L_2 norm of the error in the vorticity integrated over the uniform grid employed for the vorticity distributions of Fig. 5 (for $t = 10$). The errors were obtained in reference to the case with $\alpha = 0$. For example, for $\alpha = 4$, we get a 40% reduction in the number of elements with an error of $\Delta^2 \sqrt{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} [\omega_{ij}(\alpha = 0) - \omega_{ij}(\alpha = 4)]^2} \simeq 0.00011$ where Δ is the cell size of an $N_x \times N_y$ uniform grid. The reduction in the number of elements within the scheme described above is essentially further promoting the uniformity in element spacing since it is requiring the neighborhood radius of a certain element (and thus the elements' spacing) to be larger than $R_i = \sqrt{\alpha \nu \Delta t}$. Fig. 9 shows the elements locations at $t = 10$ for $\alpha = 0$ compared with those for the case $\alpha = 4$. The reduction in the number of elements as well as the enhanced uniformity in elements' spacing is clearly observed.

Spatial adaptivity is achieved through variable element spacing, assumed to be a given function of space $h(x, y)$. To ensure a smooth vorticity distribution, the cores of the elements must overlap according to $\sigma_i / h_i \simeq 2$, where h_i is the element spacing in the neighborhood of element i. The element core size is then a function of space $\sigma(x, y)$. In the context of the redistribution scheme, the following observations regarding the constraints governing the variable-spacing variable-core scheme are pointed out:

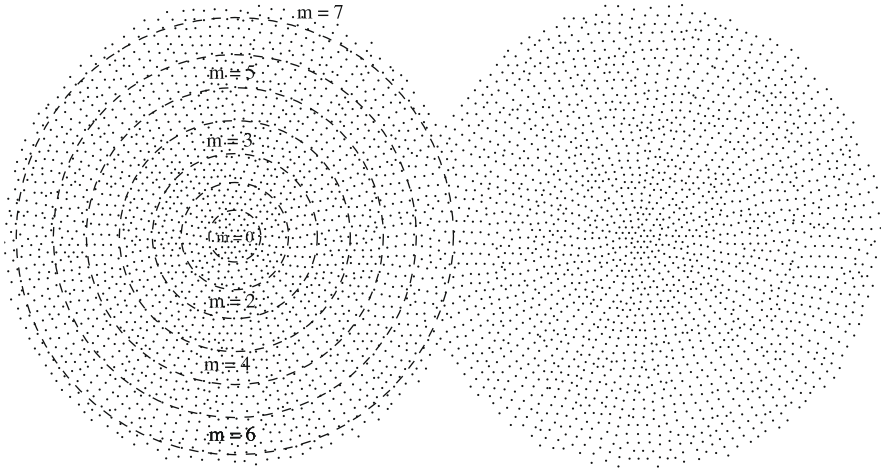

Fig. 10. Core radii vary in increments; $\sigma^2 = \sigma_0^2 + m(4\nu\Delta t)$.

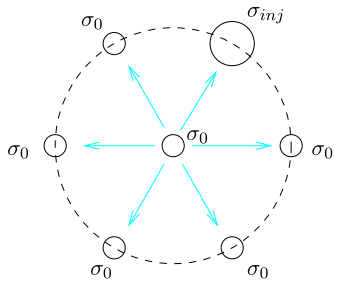

Fig. 11. Schematic: redistributing to elements of different core sizes.

- Convection may move an element from a region of low σ to a region of large σ or vice versa, so the spatial adaptivity scheme should incorporate the ability to expand or shrink the core of an element accordingly. This poses a challenge since core expansion entails diffusion; expanding an element core from σ_1 to σ_2 is essentially diffusing the element over a time period of $(\sigma_2^2 - \sigma_1^2)/4\nu$. If this time period is an integer multiple of the time step, $(\sigma_2^2 - \sigma_1^2)/4\nu = m\Delta t$, then expanding the core from σ_1 to σ_2 could be achieved by core expansion over m time steps. But if m is too large, then the element may have moved during these m time steps to a position with different core size requirement. Noting that having m large implies that the desired core size variation in space is more than can be physically manifested by diffusion, the solution to this problem is to have the variation of $\sigma(\mathbf{x})$ such that the length l_σ over which σ^2 varies by $4\nu\Delta t$, to be larger than the distance traveled by an element at $\mathbf{x}$ over Δt, i.e. $l_\sigma > |\mathbf{u}|\Delta t$. The element spacing h, however, is chosen to also satisfy the same requirement, i.e. $h > |\mathbf{u}|\Delta t$. Therefore, the condition $l_\sigma \sim h$ must be satisfied. As for core reduction, reducing a core from σ_1 to σ_2 essentially entails backward diffusion over a time period of $(\sigma_2^2 - \sigma_1^2)/4\nu$. If this period is an integer multiple of the convection time step, then reducing the core from σ_1 to σ_2 can be achieved by setting the element core to σ_2

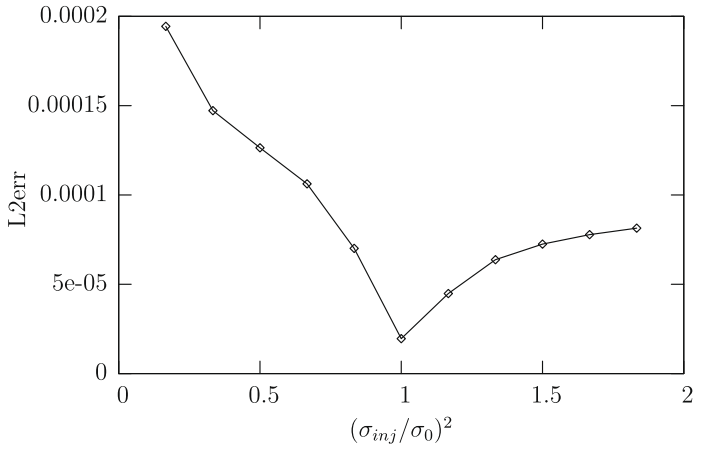

Fig. 12. L_2 norm of the integrated point-wise error in vorticity versus σ_{inj}/σ_0.

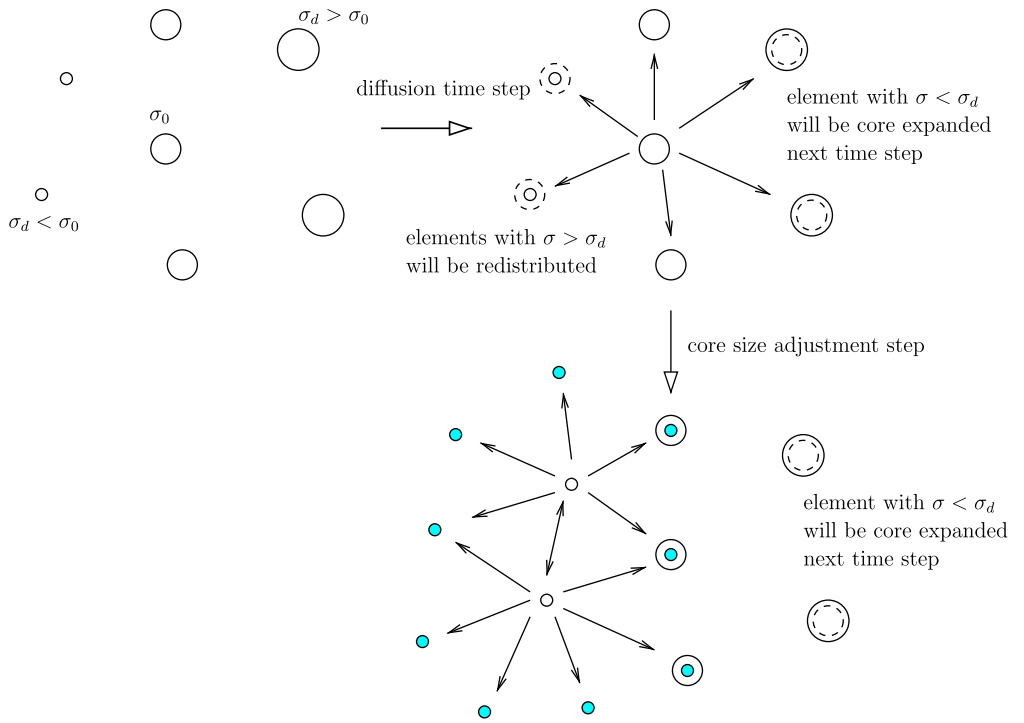

Fig. 13. Flowchart of the algorithm proposed for spatial adaptivity.

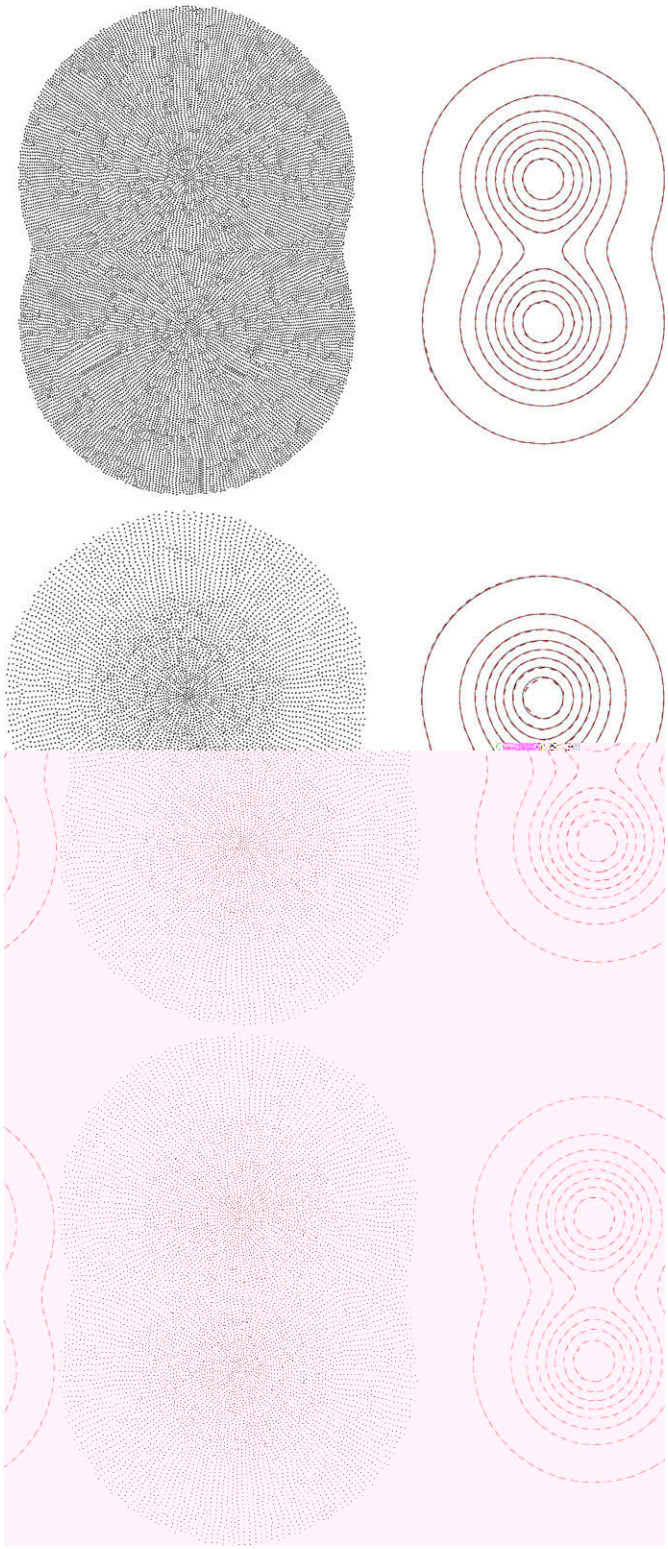

Fig. 14. Elements and vorticity contours obtained by redistribution (dashed) overlaid on exact solution. Top: diffusion using single core $\sigma = \sqrt{24\nu\Delta t}$ single spacing $h \sim \sqrt{6\nu\Delta t}$. Center: diffusion using variable-core variable-spacing algorithm. Redistribution with fixed cores. Bottom: diffusion using variable-core variable-spacing algorithm. Redistribution with variables cores.

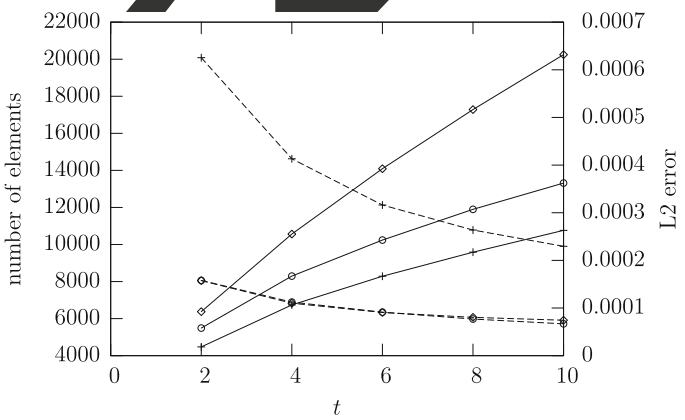

Fig. 15. Number of elements (solid lines) and accuracy (dashed lines) of the variable-spacing variable-core algorithms. The plots with diamonds correspond to redistribution with fixed cores. Plots with circles correspond to variable-core variable-spacing with redistribution with fixed cores. Plot with the + symbol correspond to variable-core variable-spacing with redistribution with variable cores.

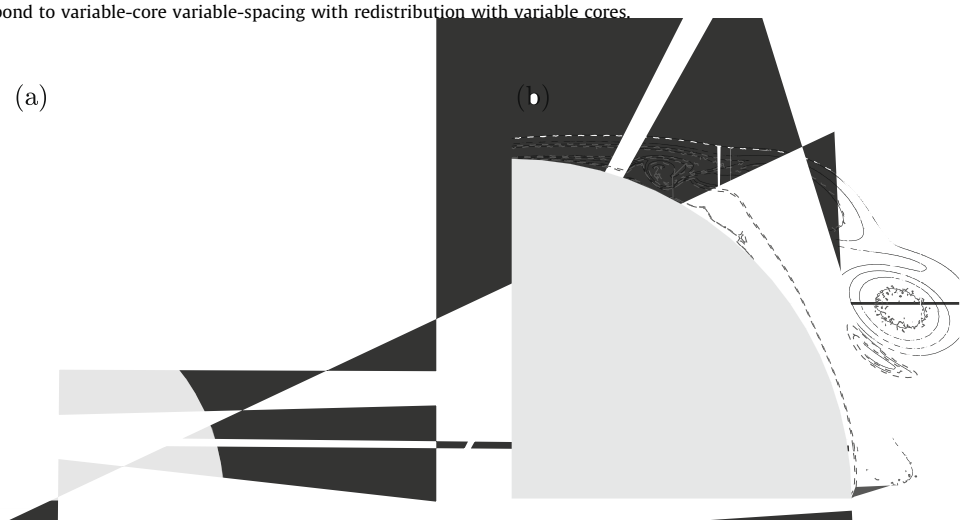

Fig. 16. Vorticity contours for (i) $\sigma/h = \sqrt{2/3}$ (dashed lines) and (ii) $\sqrt{2/3} \leqslant \sigma \leqslant 3$ (solid lines). (a) $\mathrm{Re} = 3000, t = 5$ and (b) $\mathrm{Re} = 9500, t = 3$.

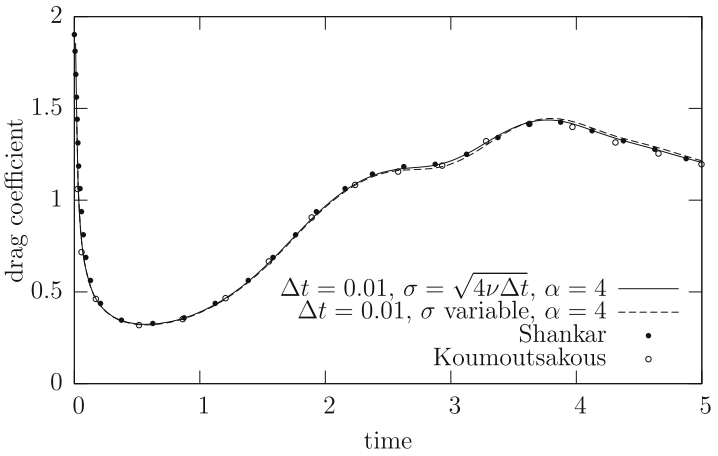

Fig. 17. Comparison of time evolution of the drag coefficient with [35] and [22] for $\mathrm{Re} = 3000$.

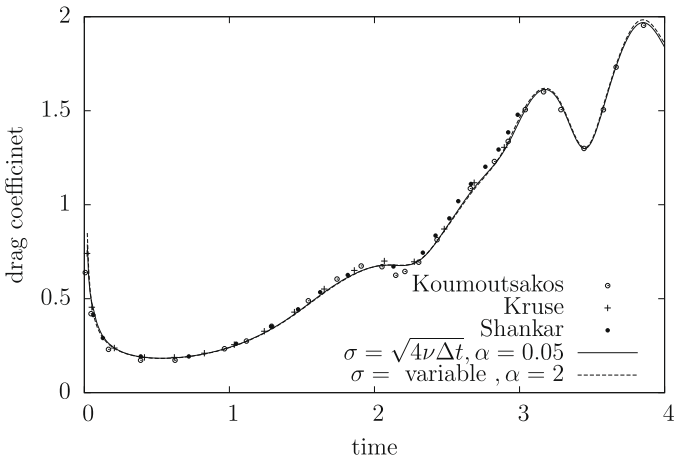

Fig. 18. Comparison of time evolution of the drag coefficient with [35,39] and [22] for $\mathrm{Re} = 9500$.

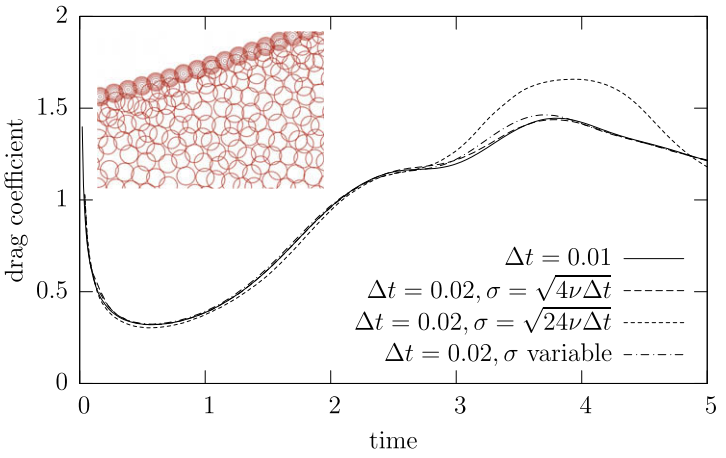

Fig. 19. Impact of core size on time evolution of drag coefficient.

are integer multiples of $4\nu\Delta t$ and these squares vary by $\pm 4\nu\Delta t$ as proposed above, then even redistributing to larger cores can be avoided and replaced by redistributing to new elements of core size σ_0 positioned on top of the existing neighbors with core sizes $\sqrt{\sigma_0^2 + 4\nu\Delta t}$. In the next diffusion step, the elements with core size σ_0 located in regions where the desired core size is $\sqrt{\sigma_0^2 + 4\nu\Delta t}$ are simply diffused by core expansion.

Based on the previous discussion, spatial adaptivity is rendered possible by adopting a variable-spacing variable-core algorithm that follows the recommendations made above. A flowchart of the algorithm is presented in Fig. 13. The square of the elements core sizes (σ^2) must vary by $\pm 4\nu\Delta t$ over a distance larger than the local average element spacing $h \sim \sigma/2$. Diffusion by redistribution is done among those neighbors that only share the same core size. Injection of new elements is such that priority is given to positions coinciding with neighboring elements with different cores. After diffusion, all

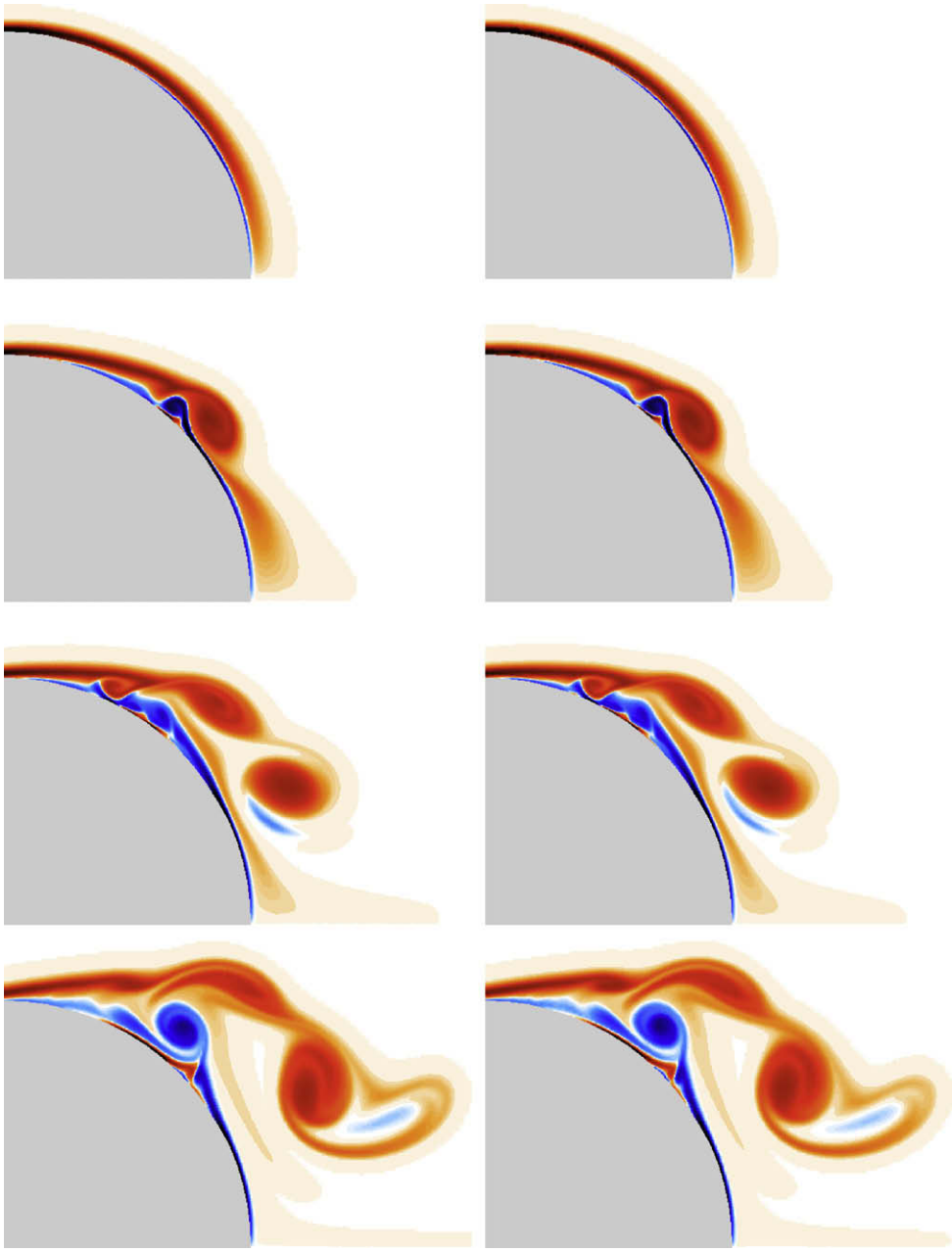

Fig. 20. Vorticity fields for $\mathrm{Re} = 9500$ at $t = 1, 2, 3$ and 4. Left: $\Delta t = 0.01, \sigma = \sqrt{4\nu\Delta t}$. Right: $\Delta t = 0.01$, σ variable.

the elements are scanned to check whether their core radii are according to the desired distribution. If not, then increasing the core size is done by core expansion $\sigma_2^2 = \sigma_1^2 + 4\nu\Delta t$ over the next time step, whereas reducing the core size is done by setting $\sigma_2^2 = \sigma_1^2 - 4\nu\Delta t$ and then redistribute over Δt.

As an example, diffusion of two vortices is considered. The circulations of the vortices are $\Gamma_1 = 1$ and $\Gamma_2 = 1$ and their initial positions are $(-0.25, 0)$ and $(0.25, 0)$, respectively. The initial vorticity distributions are given by Eq. (8) with $\sigma_0 = \sqrt{24\nu\Delta t}$, with $\nu = 0.001$ and $\Delta t = 0.01$. The following three scenarios are compared: (i) diffusion with fixed cores, $\sigma = \sigma_0$, (ii) variable-core variable-spacing with diffusion among fixed cores followed by core size correction according to the algorithm of Fig. 13, and (iii) variable-core variable-spacing with diffusion among different cores. Comparisons between these scenario are presented in Fig. 14 in terms of the computational elements and the vorticity contours overlaid on top of the exact solution. Spatial adaptivity is apparent for scenarios (ii) and (iii) with both cases showing significantly less elements at no cost (case iii) or little cost (case ii) in accuracy of point-wise vorticity as observed from vorticity contours. Reduction in the number of elements as well as the corresponding L_2 norm of the integrated point-wise vorticity error for the three cases are plotted in Fig. 15 versus time. Once again, the peculiar behavior of the error decreasing in time is observed with scenario (iii) having similar accuracy to scenario (i) with 40 less elements. As follows from the discussion above, redistribution with variable core is not as accurate, though it yields the minimum number of elements among the scenarios considered.

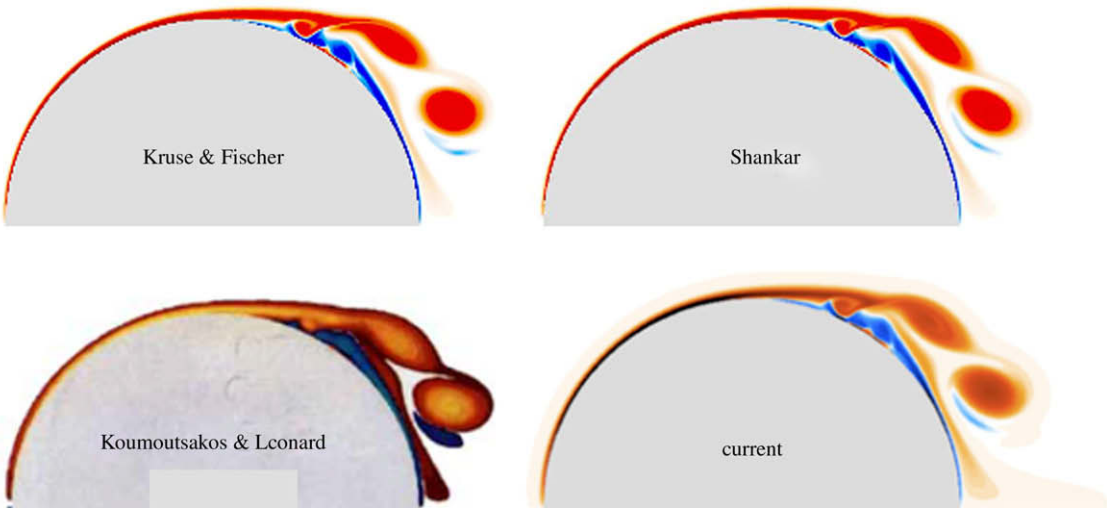

Fig. 21. Vorticity fields at $t = 3$ for $\mathrm{Re} = 9500$.

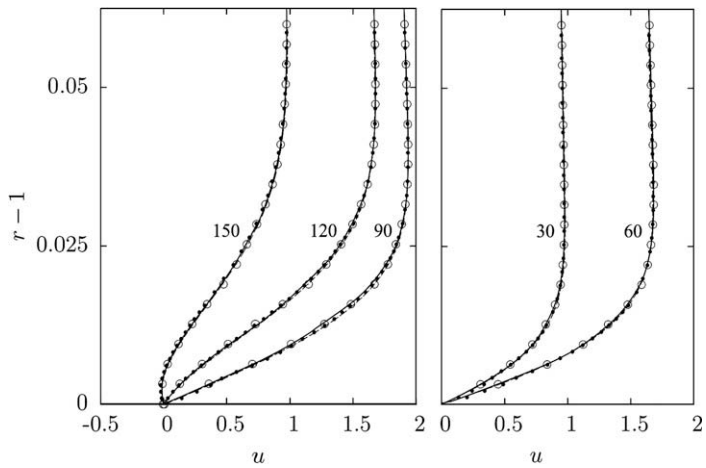

Fig. 22. Tangential velocity versus radial distance at $t = 0.5$ at different angles. Dashed lines are second-order boundary layer theory. Filled circles are results from Shankar [35]. Solid lines are present solutions for $\sigma/h = \sqrt{2/3}$ and empty circles are present solutions for variable cores $\sqrt{2/3} \leqslant \sigma/h \leqslant 3$.

5. Computational algorithm

The computational algorithm is based on Strang operator splitting in which, during a time step Δt, vortex elements are first diffused over $\Delta t/2$, then convected over Δt, and then diffused again over $\Delta t/2$. Each diffusion sub-step is performed using the smooth redistribution method. Convection is performed using a second-order Runge–Kutta integration scheme. The no-slip boundary condition is enforced in conjunction with each diffusion step as discussed in Section 2. The algorithm is outlined as follows:

(1) Viscous sub-step (diffusion + no-slip) from t to $t + \Delta t/2$ as outlined at the end of Section 2.
(2) Convection – Runge–Kutta step 1: convect elements over the half time step according to

$$\mathbf{x}_i\left(t + \frac{\Delta t}{2}\right) = \mathbf{x}_i(t) + \frac{\Delta t}{2}\mathbf{u}_i(t) \tag{25}$$

(3) Convection – Runge–Kutta step 2: convect elements according to

$$\mathbf{x}_i(t + \Delta t) = \mathbf{x}_i(t) + \frac{\Delta t}{2}\left(\mathbf{u}_i(t) + \mathbf{u}_i\left(t + \frac{\Delta t}{2}\right)\right) \tag{26}$$

(4) Viscous sub-step (diffusion + no-slip) from $t + \Delta t/2$ to $t + \Delta t$ as outlined at the end of Section 2.

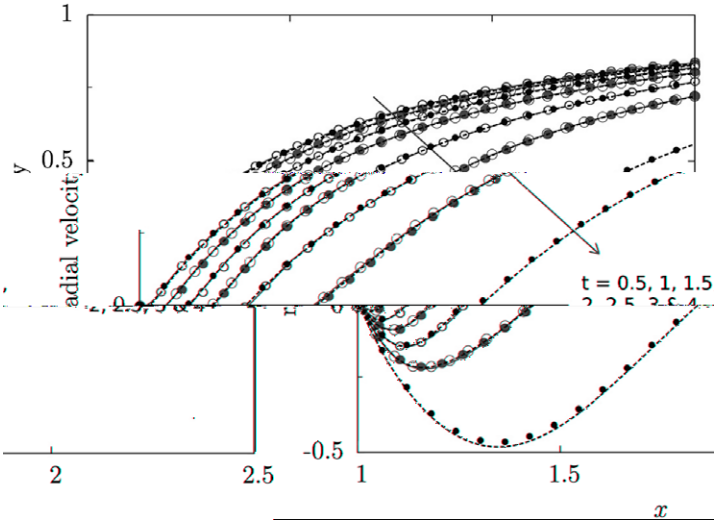

Fig. 23. Radial velocity profiles along the rear symmetry axis at $t = 0.5, 1, 1.5, 2, 2.5, 3$ and 4. Dashed lines: $\Delta t = 0.01, \sigma = \sqrt{4\nu\Delta t}$. Black filled and empty circles: $\Delta t = 0.01, \sigma$ variable. Grey filled circles: Shankar [35]. Solid line: Kruse and Fischer [39].

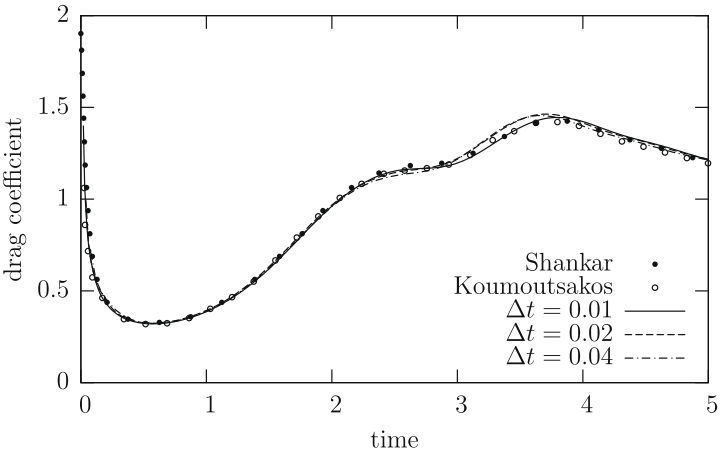

Fig. 24. Impact of time step on time evolution of the drag coefficient for the case $\mathrm{Re} = 3000$. All cases have variable cores $\sigma = \sqrt{4m\nu\Delta t}$ for m positive integer. All cases also have the same redistribution fraction $\alpha = 4$.

Tables were constructed to provide the velocity due to elements that are a distance less than or equal to 4σ from the solid boundary. For elements that are a distance larger than 4σ from the boundary, the vorticity field is approximated by the unbounded core function (Eq. (8)). A fast multipole algorithm of order $N \log N$ was employed to compute the velocity field with direct summations included for neighboring elements.

6. Lift and drag coefficients

The force acting on the solid body may be obtained by taking a control volume that contains both the fluid and the solid body and that extends to infinity. The control volume is selected to move with the solid body so that the integral form of the conservation of momentum is

$$-\mathbf{F}_b + M_b \frac{d\mathbf{U}_b}{dt} = \frac{d}{dt}\int_{C.V.} \rho \mathbf{u} d\mathbf{x} + \int_{C.S.} \rho \mathbf{u}\mathbf{u}\cdot\hat{\mathbf{n}} dS \tag{27}$$

Noting that the momentum flux term is zero since at infinity the velocity is $\mathbf{U}_\infty - \mathbf{U}_b$ where $\mathbf{U}_b$ is the velocity of the body, so that

$$-\mathbf{F}_b + M_b \frac{d\mathbf{U}_b}{dt} = \frac{d}{dt}\int_{C.V.} \rho \mathbf{u}_\omega d\mathbf{x} \tag{28}$$

The force on the body may be expressed in terms of the vorticity field as

$$\mathbf{F}_b = -\frac{d}{dt}\int_{C.V.} \rho(\boldsymbol{\omega}\times\mathbf{x}) d\mathbf{x} + M_b \frac{d\mathbf{U}_b}{dt} \tag{29}$$

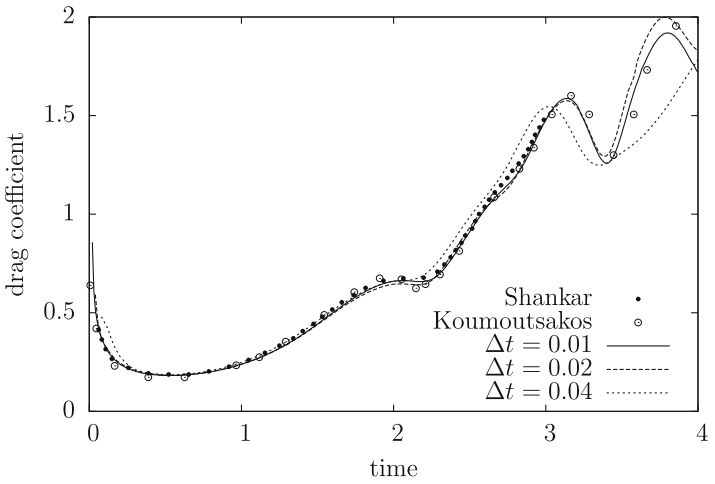

Fig. 25. Impact of time step on time evolution of the drag coefficient for $\mathrm{Re} = 9500$. $\sigma = \sqrt{m\nu\Delta t}$ where m varies from 4 to 24.

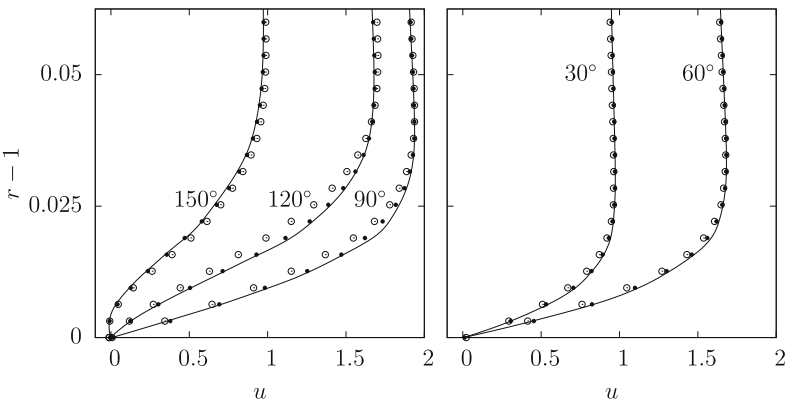

Fig. 26. Impact of time step on the tangential velocity profile (at $t = 0.5$) across the boundary layer at different angles. Solid lines: $\Delta t = 0.01$, filled circles $\Delta t = 0.02$, empty circles $\Delta t = 0.05$.

Noting that $\omega(\mathbf{x}) = \sum_{i=1}^{N} \Gamma_i f_{\sigma_i}(\mathbf{x}_i, \mathbf{x})$, then for two-dimensional flows around bodies moving at constant velocity the lift and drag are

$$\mathbf{F}_d = \frac{d}{dt} \sum_{i=1}^{N} \Gamma_i \int_{C.V.} y f_{\sigma_i}(\mathbf{x}_i, \mathbf{x}) d\mathbf{x} \tag{30}$$

and

$$\mathbf{F}_l = -\frac{d}{dt} \sum_{i=1}^{N} \Gamma_i \int_{C.V.} x f_{\sigma_i}(\mathbf{x}_i, \mathbf{x}) d\mathbf{x} \tag{31}$$

where the density is set to unity. The integrals in the lift and drag coefficients are recognized as the first y and x moments of vorticity, respectively, which for a single element, are provided by the right-hand sides of Eqs. (13) and (12), respectively. The lift and drag coefficients for the impulsively started flow over a cylinder discussed in the next section are $C_l = F_l/U_\infty^2 R$ and $C_d = F_d/U_\infty^2 R$, respectively, where R is the cylinder radius.

7. Computations of flow over an impulsively started cylinder

In this section, the impact of various parameters on the accuracy and computational cost of the method are investigated. For this purpose, the canonical problem of the flow over a cylinder is considered. Accuracy of the methodology presented in this paper is investigated by comparing the drag coefficient, vorticity distribution, and velocity profiles in the boundary layer with computational results reported by the redistribution method [35], the PSE method [22], and the Spectral Element Method [39]. Cost of the algorithm is reported in terms of the time evolution of the number of computational elements. The parameters investigated are (i) core-size elements spacing overlap, (ii) time step, and (iii) redistribution fraction α and spatial adaptivity. Tests are conducted primarily for the cases of $\mathrm{Re} = 3000$ and $\mathrm{Re} = 9500$. The performance of the spatial adaptivity algorithm is investigated for the case of $\mathrm{Re} = 1000$ with the cylinder undergoing forced angular oscillations.

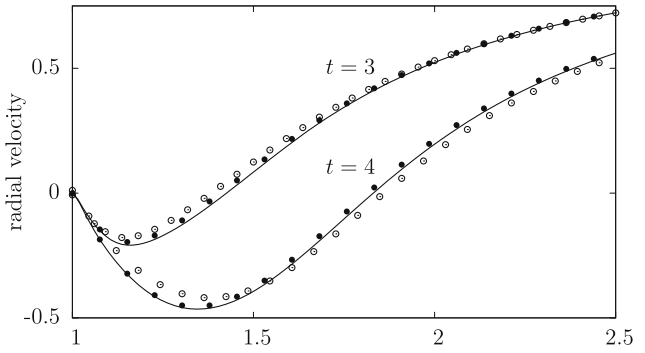

Fig. 27. Impact of time step on the radial velocity profile along the rear symmetry axis. Solid lines: $\Delta t = 0.01$, filled circles $\Delta t = 0.02$, empty circles $\Delta t = 0.04$.

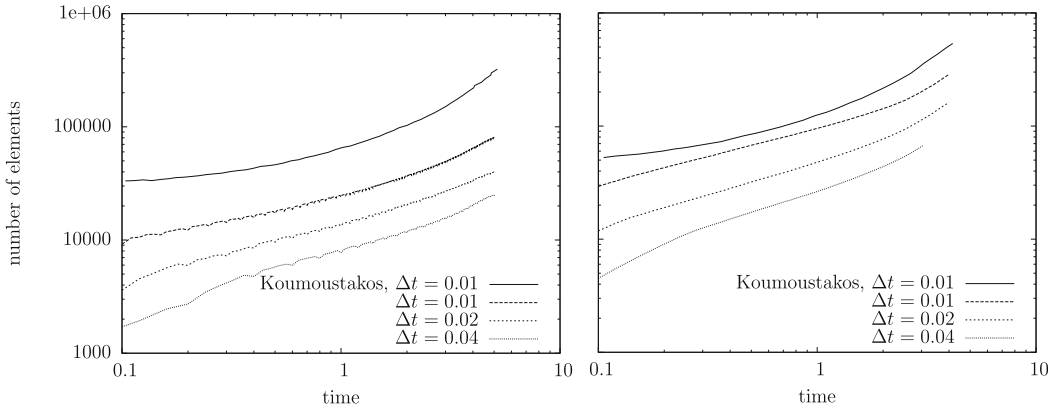

Fig. 28. Comparison of the number of elements vs time for different values of the time step with [22]. Left: $\mathrm{Re} = 3000$, Right: $\mathrm{Re} = 9500$.

7.1. The impact of core size – elements spacing overlap

The overlap ratio (σ/h), the ratio of the core size σ to average elements spacing h, plays an important role in the behavior of the smooth redistribution method [26]. Overlap ratios satisfying $\sigma/h < 1$ introduce smaller errors due to convection of elements and larger errors in point-wise vorticity, while larger values, $\sigma/h > 1$, introduce smooth point-wise vorticity distribution at the expense of larger errors introduced by elements convection. The redistribution method, in its original form [35], dealt with point vortices $\omega = \Gamma_i \delta(x - x_i)\delta(y - y_i)$, for which case the overlap ratio $\sigma/h = 0$. In their simulations of flow over a cylinder, Shankar et al. [35] enforced the no-slip boundary condition by creating a vortex sheet of strength $-2\mathbf{u}\cdot\hat{\mathbf{s}}$ and replacing the vortex sheet by a set of point vortices equally spaced on the boundary so that each element is created with a circulation of $-2\mathbf{u}\cdot\hat{\mathbf{s}}\Delta s$, where Δs is the elements spacing. The simulations presented in [35] yielded accurate results for the drag coefficient, vorticity field, and boundary layer velocity profiles. In these simulations, the vorticity field is represented as a set of delta functions (desingularized for convection), and evaluation of the point-wise vorticity field is carried out using an infinite-order smoothing function. Redistribution using Gaussian cores was later presented in [26] and values of $\sigma/h \sim 2$ proved to be an optimal compromise. The scheme proposed for satisfying the no-slip boundary condition creates elements at the boundary with cores $\sigma = \sqrt{4\nu\Delta t}$. For an average element spacing $h = \sqrt{6\nu\Delta t}$, the corresponding overlap ratio, $\sim \sqrt{2/3}$, is not large enough for the point-wise vorticity to be smooth. Fig. 16 shows the vorticity contours for the flow over an impulsively started cylinder at $t = 5$ for the case $\mathrm{Re} = 3000$ and $t = 3$ for the case $\mathrm{Re} = 9500$. For an overlap ratio of $\sim \sqrt{2/3}$, the vorticity contours are noisy. On the same figure are shown vorticity contours with core sizes that increase from $\sigma = \sqrt{4\nu\Delta t}$ at the boundary to $\sigma = \sqrt{54\nu\Delta t}$ at the farthest point from the cylinder. For this overlap range of $[\sqrt{2/3}, 3]$ the vorticity contours are much smoother than those for the case of $\sigma/h = \sqrt{2/3}$.

The drag coefficient versus time for both cases, compared with those presented by the redistribution method [35], spectral element method [39] and the particle strength exchange method [22]), are shown in Figs. 17 and 18. Both cases are in good agreement with [35,39] and [22], with the case of $\sigma/h = \sqrt{2/3}$ slightly more accurate at later times due to the smaller error introduced by convecting elements with finite cores by moving their centers.

If on the other hand, a constant overlap ratio of $\sigma/h = 2$ is desired everywhere, then the elements injected at the boundary are diffused by core expansion until their core size reaches $\sqrt{24\nu\Delta t}$ after which they are diffused by redistribution, as

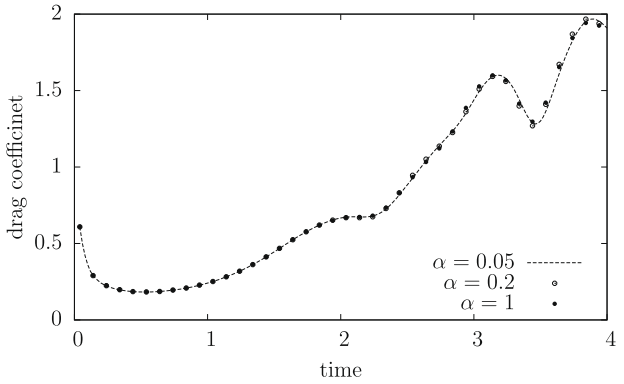

Fig. 29. Impact of redistribution length $\sqrt{\alpha\nu\Delta t}$ on time evolution of drag coefficient for $\mathrm{Re} = 9500$. $\Delta t = 0.02, \sigma = \sqrt{4\nu\Delta t}$.

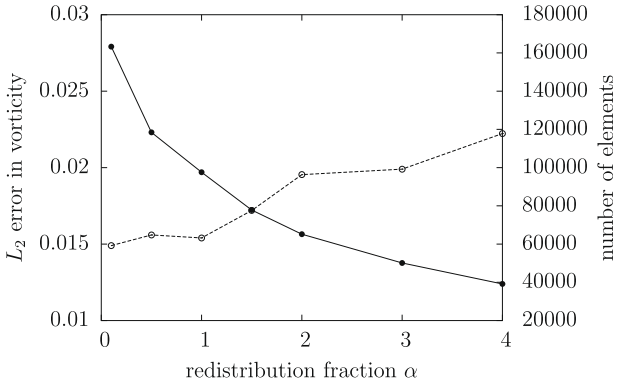

Fig. 30. Impact of redistribution fraction α on the number of elements and error in vorticity for the case $\mathrm{Re} = 9500$ at $t = 3$.

shown in inset of Fig. 19 (ratio σ/h is not to scale in the figure). In this case, the drag coefficient deviates significantly from the correct value at and beyond separation (for $t \geqslant 3$) as shown in Fig. 19 for the case $\mathrm{Re} = 3000$. This is expected because larger cores introduce larger errors in convection that propagate in time and more importantly because elements with large cores near the boundary do not see the boundary as locally semi-infinite, which is required by the redistribution scheme for diffusion near a solid boundary as discussed in Section 3.1. For the case of $\mathrm{Re} = 9500$, vorticity fields at $t = 1, 2, 3$ and 4 are presented in Fig. 20. Comparison with [35,39] and [22] at $t = 3$ is shown in Fig. 21.

Tangential velocity profiles across the boundary layer (at $t = 0.5$) are presented in Fig. 22 at angles 30°, 60°, 90°, 120° and 150° from the front symmetry line. On the same plot are shown results from Shankar [35] (filled circles) and predictions of second-order boundary layer theory (dashed lines). Comparison of radial velocity profiles along the rear symmetry axis at different times is shown in Fig. 23.

7.2. Impact of time step

The impact of the time step on the time evolution of the drag coefficient is presented in Figs. 24 and 25 for the cases of $\mathrm{Re} = 3000$ and 9500, respectively. The three time steps considered ($\Delta t = 0.01, 0.02,$ and 0.04) are again compared with results from [35] and [22]. The variable-cores variable-spacing algorithm has been employed in all the cases so that elements near the boundary have cores of size $\sqrt{4\nu\Delta t}$ with the core size and elements spacing increasing for elements farther from the boundary. For larger time steps, errors in the drag coefficient are observed and in particular after the point of separation ($t > 3$ for $\mathrm{Re} = 3000$ and $t > 2$ for $\mathrm{Re} = 9500$). This is attributed primarily to the fact that elements near the boundary with core sizes $\sqrt{4\nu\Delta t}$ see the boundary as locally semi-infinite becomes less accurate as Δt increases, as discussed in the previous section. Impact of time step on the tangential velocity profile across the boundary layer at various angular positions for $\mathrm{Re} = 9500$ at $t = 0.5$ is shown in Fig. 26. The radial velocity profiles along the rear symmetry axis at $t = 3$ and 4 are presented in Fig. 27 for the different times steps. The significant savings in computational cost the present method offers are expressed in Fig. 28, where it is shown that the number of elements is considerably less than that of the PSE scheme

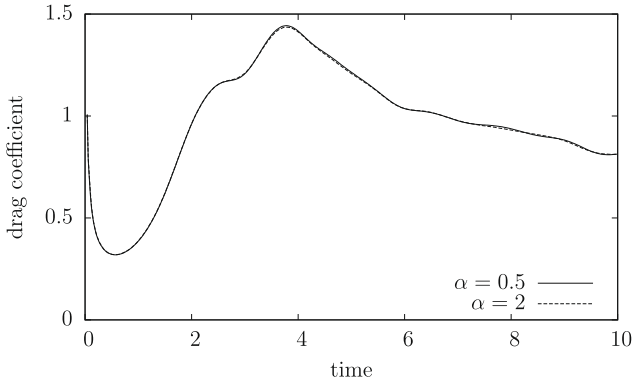

Fig. 31. Impact of redistribution length $\sqrt{\alpha\nu\Delta t}$ on time evolution of drag coefficient for $\mathrm{Re} = 3000$. $\Delta t = 0.02$, $\sigma = \sqrt{4\nu\Delta t}$.

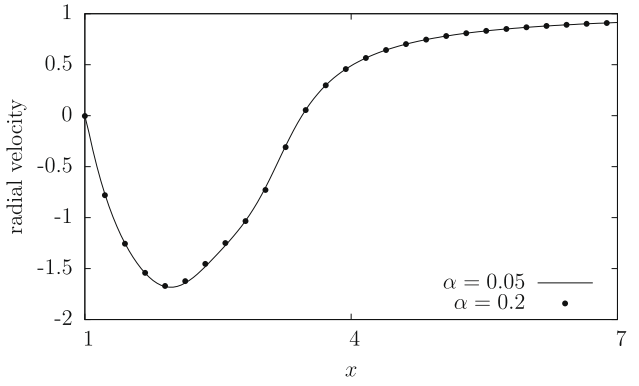

Fig. 32. Impact of redistribution length $\sqrt{\alpha\nu\Delta t}$ on radial velocity profile along the rear symmetry line at $t = 10$ for $\mathrm{Re} = 3000$. $\Delta t = 0.02$, $\sigma = \sqrt{4\nu\Delta t}$.

[22]. For example, for $\mathrm{Re} = 3000$ at $t = 5$, the number of elements is 80,000 whereas the number of elements is more than 300,000 using the PSE scheme for the same time step.

7.3. Impact of redistribution length and spatial adaptivity

The number of elements can be significantly reduced by invoking the variable-core variable-spacing algorithm for spatial adaptivity. As discussed in Section 4, the algorithm removes elements in overcrowded regions by replacing all elements within a redistribution length $\sqrt{\alpha\nu\Delta t}$ by a single element while conserving the various moments of vorticity by involving neighboring elements in the redistribution process. This results in a significant reduction in the number of elements with

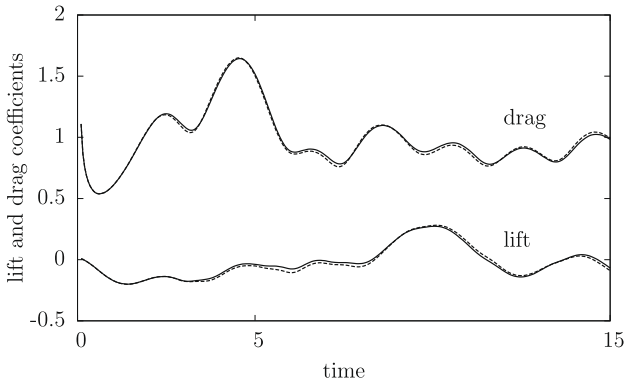

Fig. 33. Drag and lift coefficients. Solid lines: fixed core-fixed spacing. Dashed lines: variable core-variable spacing.

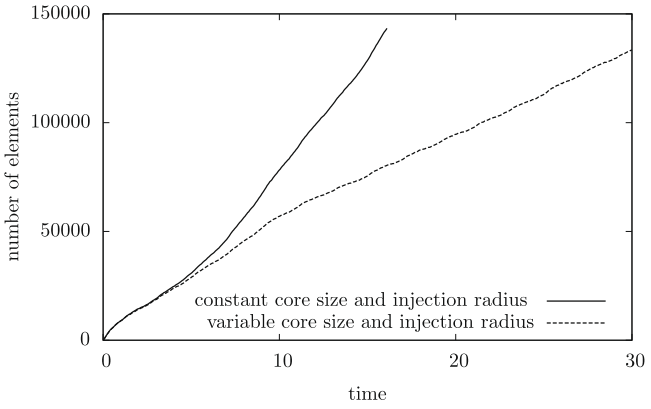

Fig. 34. Impact of spatial adaptivity on the time evolution of the number of elements.

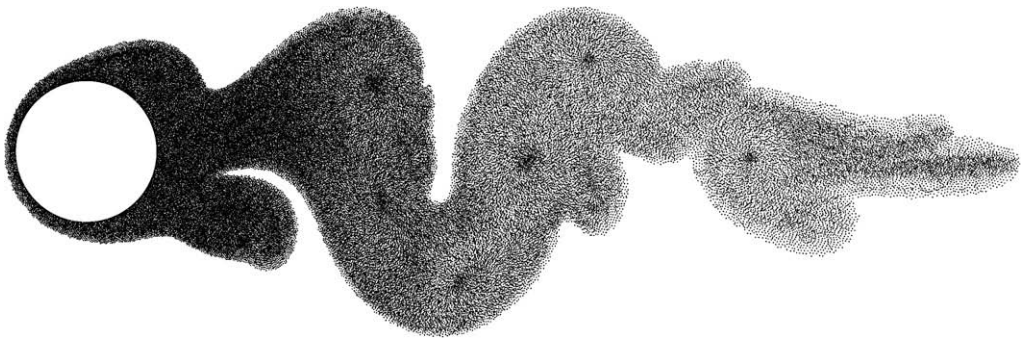

Fig. 35. Elements positions at $t = 15$ for the variable core-variable spacing case.

little loss in accuracy. Fig. 29 shows that increasing the redistribution fraction α from 0.05) to 1 results in virtually no change in the drag coefficient for the case $\mathrm{Re} = 9500$. The corresponding impact of increasing the redistribution fraction on the number of elements and accuracy are presented in Fig. 30. The accuracy is measured in terms of the L_2 norms of error in vorticity integrated over the computational domain $\left(\iint(\omega-\omega^*)^2 dx\,dy\right)^{1/2}$, where ω^* is a reference vorticity field corresponding to $\Delta t = 0.01$. The figure shows that for $\alpha \leqslant 1$, increasing α results in dramatic reduction in number of elements at minimal loss in accuracy. For example, increasing $\alpha = 0.05$ to $\alpha = 1$, the number of elements is reduced from 165,000 elements to less than 100,000 elements with very little loss in accuracy (from 0.0149 to 0.0154).

Fig. 31 shows the impact of increasing α to 2 on the drag coefficient over a longer period time for the case $\mathrm{Re} = 3000$. While the drag coefficient history is virtually identical, the number of elements dropped from 268,793 to 140,173. The radial velocity profiles along the rear symmetry line at $t = 10$, shown in Fig. 32 for the two cases, further proves that this significant reduction in number of elements comes at virtually no cost.

To investigate the impact of spatial adaptivity over longer times, the case of uniform flow over an oscillating cylinder is considered. The Reynolds number based on the free stream velocity is $\mathrm{Re} = 1000$ and the cylinder is undergoing angular oscillations with angular velocity $\Omega = \Omega_{\max}\sin 2\pi f t$, where $\Omega(t)$ is the angular velocity at time t, and $\Omega_{\max} = 1$ and $f = 0.25$ are, respectively, the amplitude and frequency of oscillations. Two simulations were conducted for a duration of a few oscillation periods, one with fixed core size $\sigma = \sqrt{4\nu\Delta t}$ and injection radius (average elements spacing) of $\sqrt{6\nu\Delta t}$, and the second with variable core size $\sigma = \sqrt{m\nu\Delta t}$, where m increases from 4 at the solid boundary to a maximum of 384 away from the solid boundary. The corresponding injection radius is $\sigma/2$ for $m > 24$ and $\sqrt{6\nu\Delta t}$ for $m < 24$. The lift and drag coefficients time evolution for the two cases presented in Fig. 33 shows that the chosen spatial adaptivity is capable of predicting these coefficients over the time period to an acceptable accuracy. The reduction in number of elements offered by spatial adaptivity is presented in Fig. 34 in terms of time evolution of the number of elements for the two cases considered. The elements positions at $t = 15$ are depicted in Fig. 35 for the case with spatial adaptivity. The parameters were chosen such that the average elements spacing increases from $\bar{h}_{\min} = h_0/\sqrt{2}$ near the solid wall to a maximum of $\bar{h}_{\max} = 4h_0$

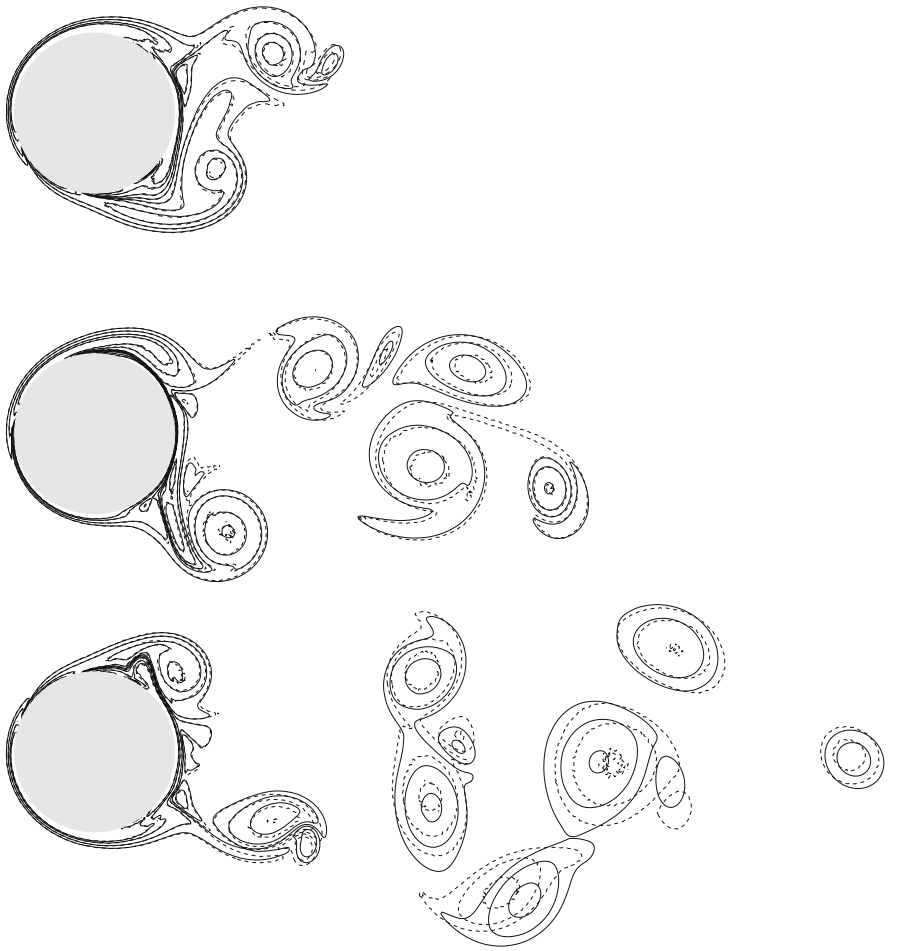

Fig. 36. Vorticity contours at $t = 5, 10$ and 15. Solid lines: fixed core-fixed spacing. Dashed lines: variable core-variable spacing.

away from the boundary ($h_0 = \sqrt{6\nu\Delta t}$). The cost of such significant reduction in the number of elements, and in particular, increasing the elements spacing by a factor of $4\sqrt{2}$, is some loss in accuracy in the vorticity field away from the boundary, as seen in Fig. 36, where the vorticity field is shown in terms of vorticity contours at $t = 5, 10$ and 15 for the case with spatial adaptivity and that without spatial adaptivity. Tradeoff between accuracy and cost is made possible by choice of the maximum to minimum average elements spacing ratio $\bar{h}_{\max}/\bar{h}_{\min}$, and by the distance over which this variation takes place.

8. Conclusion

Novel contributions to a high-resolution spatially adaptive 2D vortex method in bounded domains are presented. Redistribution equations governing diffusion of vortex elements near the solid boundary are presented in Section 3.1 so that the no-flux boundary condition of the vorticity at solid boundary is approximately satisfied by the choice of the core function of Eq. (6). The smooth redistribution scheme is extended to accurately satisfy the no-slip boundary condition. In this respect the no-slip boundary condition is enforced by creation of a vortex sheet that is subsequently removed by redistributing its strength to neighboring element using the redistribution scheme of Section 3.2. Computational cost is considerably reduced by removal of elements within inner search radius $\sqrt{\alpha\nu\Delta t}$ which virtually no loss in accuracy for $\alpha \leqslant 1$. The variable-core variable-spacing algorithm (Section 4) endows the method with spatial adaptivity taking advantage of core expansion and the smooth redistribution methods for diffusion. Spatial adaptivity offers a tradeoff between accuracy and cost by controlling rates of increase of the elements core size and spacing as a function of distance from the solid boundary.

Acknowledgment

This research was partially supported by the University Research Board of the American University of Beirut and the Reacting Gas Dynamics Laboratory at the Massachusetts Institute of Technology.

References

[1] C. Anderson, C. Greengard, Vortex Methods, Lecture Notes in Mathematics, vol. 1360, Springer Verlag, 1988.
[2] L.A. Barba, A. Leonard, C.B. Allen, Advances in viscous vortex methods—meshless spatial adaption based on radial basis function interpolation, Int. J. Numer. Meth. Fluids 47 (2005) 387–421.
[3] L.A. Barba, A. Leonard, C.B. Allen, Vortex method with meshless spatial adaption for accurate simulation of viscous, unsteady vortical ows, Int. J. Numer. Meth. Fluids 47 (2005) 841–848.
[4] J.E. Barnes, P. Hut, A hierarchical $o(n \log n)$ force calculations algorithm, Nature 324 (1986) 446.
[5] R. Benhaddouch, Treatment of a Neumann boundary condition by a particle strength exchange method, in: Proceedings of the Third International Workshop on Vortex Flows and Related Numerical Methods, ESAIM (European Series in Applied and Industrial Mathematics), August 24–27, 1999.
[6] A. Cheer, Unsteady separated wake behind an impulsively started cylinder in slightly viscous flow, J. Fluid. Mech. 201 (1989) 485–505.
[7] A.J. Chorin, Vortex methods for rapid flows, in: Proceedings of the 2nd International Congress on Numerical Methods of Fluid Mechanics, 1972.
[8] A.J. Chorin, Numerical study of slightly viscous flow, J. Fluid Mech. 57 (1973) 785–796.
[9] A.J. Chorin, Vortex sheet approximations of boundary layers, J. Comput. Phys. 27 (1978) 428–442.
[10] A.J. Chorin, Vortex methods, Pam report 593, Department of Mathematics, University of Berkeley, 1993.
[11] G.-H. Cottet, P. Koumoutsakos, Vortex Methods: Theory and Applications, Cambridge University Press, Cambridge, UK, 2000.
[12] G.-H. Cottet, M. Lemine, M. El Hamraoui, Recent developments in vortex methods for the simulation of unsteady incompressible flows, in: Electronic Proceedings of the Third International Workshop on Vortex Flows and Related Numerical Methods, ESAIM (European Series in Applied and Industrial Mathematics), August 1989.
[13] G.-H. Cottet, M.-L. Ould Salihi, M. El Hamraoui, Multi-purpose regridding in vortex methods, in: ESAIM: Proceedings of the Third International Workshop on Vortex Flows and Related Numerical Methods, vol. 7, 1999, pp. 94–103.
[14] Georges-Henri Cottet, Petros Koumoutsakos, Mohamed Lemine Ould Salihi, Vortex methods with spatially varying cores, J. Comput. Phys. 162 (2000) 164–185.
[15] P. Degond, S. Mas-Gallic, The weighted particle method for convection–diffusion equations. I. The case of an isotropic viscosity, Math. Comput. 53 (188) (1989) 485–507.
[16] P. Degond, S. Mas-Gallic, The weighted particle method for convection–diffusion equations. Part 2: The anisotropic case, Math. Comput. 53 (188) (1989) 509–525.
[17] A. Gharakani, A 3-D vortex-boundary element method for the simulation of unsteady, high Reynolds number flows, PhD Thesis, Massachusetts Institute of Technology, 1995.
[18] G. Ghoniem, A.F. Heidarinejad, A. Krishnan, Numerical simulation of a thermally stratified shear layer using the vortex element method, J. Comput. Phys. 79 (1988) 135–166.
[19] I. Ghoniem, A.F. Lakkis, M. Soteriou, Numerical simulation of the dynamics of large fire plumes and the phenomenon of puffing, in: Proceedings of the 26th International Symposium on Combustion, The Combustion Institute, 1996, pp. 1531–1539.
[20] L. Greengard, V. Rohklin, A fast algorithm for particle simulations, J. Comput. Phys. 73 (1987) 325.
[21] O. Knio, A.F. Ghoniem, 3-dimensional vortex simulation of rollup and entrainment in a shear-layer, J. Comput. Phys. 97 (1991) 172–223.
[22] P. Koumoutsakos, A. Leonard, High-resolution simulations of the flow around an impulsively started cylinder using vortex methods, J. Fluid Mech. 296 (1995) 1–38.
[23] P. Koumoutsakos, A. Leonard, F. Pepin, Boundary conditions for viscous vortex methods, J. Comput. Phys. 113 (1994) 52–61.
[24] K. Kuwahara, H. Takami, Numerical studies of two-dimensional vortex motion by a system of point vortices, J. Phys. Soc. Jpn. 34 (1) (1973) 247–253.
[25] I. Lakkis, A.F. Ghoniem, Lagrangian simulation of fireplumes, in: Proceedings of the 7th AIAA/ASME Joint Thermo, and Heat Transfer Conference, vol. 1, 1998, pp. 215–226.
[26] Issam Lakkis, Ahmed Ghoniem, Axisymmetric vortex method for low-Mach number, diffusion-controlled combustion, J. Comput. Phys. 182 (2) (2002) 435–475.
[27] A. Leonard, Vortex methods for flow simulation, J. Comput. Phys. 37 (1980) 289–335.
[28] A. Leonard, D. Shiels, J.K. Salmon, G.S. Winckelmans, P. Ploumhans, Recent advances in high resolution vortex methods for incompressible flows, in: Proceedings of the 13th AIAA Computational Fluid Dynamics Conference (97-2108), June 29–July 2 1997.
[29] M.J. Lighthill, Introduction, Boundary Layer Theory, Numbers 54–61, Oxford I. Press, New York, 1963.

[30] V.C. Luu, Numerical study of the reactive flow in a two-stream, coaxial-jet, axisymmetric bluff-body combustor, PhD Thesis, Massachusetts Institute of technology, 1996.
[31] L.-F. Martins, A.F. Ghoniem, Vortex simulation of the intake flow in a planar piston-chamber device, Int. J. Numer. Meth. Fluids 12 (1991) 237–260.
[32] S. Mas-Gallic, Particle approximation of a linear convection–diffusion problem with Neumann boundary conditions, SIAM J. Numer. Anal. 32 (4) (1995) 1098.
[33] P. Ploumhans, G.S. Winckelmans, Vortex methods for high-resolution simulations of viscous flow past bluff bodies of general geometry, J. Comput. Phys. 165 (2000) 354–406.
[34] P. Ploumhans, G.S. Winckelmans, J.K. Salmon, A. Leonard, M.S. Warren, Vortex methods for direct numerical simulation of three-dimensional bluff body flows: application to the sphere at Re = 300, 500, and 1000, J. Comput. Phys. 178 (2002) 427–463.
[35] S. Shankar, L. Dommelen, A new diffusion procedure for vortex methods, J. Comput. Phys. 127 (1) (1996) 88–109.
[36] M. Soteriou, A.F. Ghoniem, Effects of the free-stream density ratio on free and forced spatially-developing shear layers, Phys. Fluids (1995) 2036. 2050.
[37] M. Soteriou, A.F. Ghoniem, On the application of the infinite reaction rate model in the simulation of the dynamics of exothermic mixing layers, Combust. Sci. Technol. 105 (1995) 377–397.
[38] S. Subramaniam, A new mesh-free vortex method, PhD Thesis, The Florida State University, 1996.
[39] G.W. Kruse, P. Fischer, Private Communication, Center for Fluid Mechanics, Brown University, Providence, RI, 1006.