SIAM J. SCI. COMPUT.
Vol. 26, No. 3, pp. 885–906

ACHIEVING HIGH-ORDER CONVERGENCE RATES WITH DEFORMING BASIS FUNCTIONS*

LOUIS F. ROSSI†

Abstract. This article studies the use of moving, deforming elliptical Gaussian basis functions to compute the evolution of passive scalar quantities in a two-dimensional, incompressible flow field. We compute an evolution equation for the velocity, rotation, extension, and deformation of the computational elements as a function of flow quantities. We find that if one uses the physical flow velocity data calculated from the basis function centroid, the method has only second-order spatial accuracy. However, by computing the residual of the numerical method, we can determine adjustments to the centroid data so that the scheme will achieve fourth-order spatial accuracy. Simulations with nontrivial flow parameters demonstrate that the methods exhibit the properties predicted by theory.

Key words. convection-diffusion, particle methods, computational fluid dynamics, deforming blobs

AMS subject classifications. 35Q30, 41A25, 65M12, 65M60, 76D05

DOI. 10.1137/S1064827503425286

1. Introduction.

This paper examines a category of high spatial order particle methods using moving, deforming elements to capture passive scalar quantities in a two-dimensional, incompressible fluid flow. Grid-free numerical methods using moving basis functions are appealing to many engineers, scientists, and mathematicians because of the advantages they offer for certain flow problems. In particular, they are considered naturally adaptive because they dedicate computational effort where field values are nontrivial and nowhere else. For example, many investigators use Lagrangian methods to approximate passive scalar fields governed by the convection-diffusion equations. Also, vortex methods which use moving basis functions to capture the vorticity field in the nonlinear Navier–Stokes equations are in common use by scientists and engineers. Finally, astrophysicists have found smooth particle hydrodynamics (SPH), where moving basis functions are used to represent density, momentum, and other quantities, to be effective tools for studying large-scale astrophysical flows. In the latter two cases, the flow velocity is calculated from the representation of the computed fields. While the latter two families of methods are necessarily nonlinear, any computational issues with the linear convection-diffusion equation are issues of concern for the nonlinear schemes as well. This paper examines issues related to using deforming basis functions for the linear convection-diffusion problem, but all results pertain to nonlinear applications as well. In fact, the work discussed in this paper is presented with an eye toward developing a full vortex method using deforming basis functions, but results related to the determination of the velocity field from the computed field are beyond the scope of this paper. This paper examines deforming elements when the velocity field is a given feature of the problem and challenges the conventional wisdom that particle trajectories should follow physical streamlines. When we analyze the consistency of deforming elliptical Gaussians, a

*Received by the editors March 28, 2003; accepted for publication (in revised form) January 15, 2004; published electronically January 12, 2005. This work was supported by National Science Foundation grant DMS-9971800.

http://www.siam.org/journals/sisc/26-3/42528.html

†Department of Mathematical Sciences, University of Delaware, Newark, DE 19716 (rossi@math.udel.edu).

particularly suitable category of deforming basis function for the convection-diffusion equations, we find that one can improve the accuracy of the method by adding a correction to the particle velocity. The adjustment is related to the core size of the computational element and the curvature of the velocity field, and arises as a compensation for the fact that the convection of the scalar quantity is a nonlocal property shared among overlapping elements. These issues apply to Lagrangian methods regardless of how the velocity field is determined and so are relevant to both the linear convection-diffusion equations and nonlinear methods.

While deforming blobs are the focus of this article, it should be noted that many investigators have studied the accuracy of rigid moving basis functions. When using rigid basis functions, the convection-diffusion equation is split into convective and diffusive steps. Accuracy for each operation has been explored with substantial progress in both areas. Different-shaped basis functions achieve different orders of accuracy in the convective step in vortex methods, and surveys on this topic can be found in sources like [1, 2, 3, 4, 35]. We shall see in section 2 that the famous *moment condition,* used to construct high-order rigid basis functions, appears through a different route to consistency that is applied in this work. Nonetheless, high-order Lagrangian methods for the convection-diffusion or Navier–Stokes equations have remained elusive. In another approach, Lowengrub and Shelley [15] have boosted the order of Lagrangian schemes by coupling point vortex dynamics to an underlying curvilinear coordinates system that does not necessarily move with the fluid velocity as Lagrangian methods do. Later, with Merriman [16], they expanded this work by incorporating high-order corrections on a rectangular grid to achieve fourth-order spatial accuracy. To capture diffusive processes, there have been significant activities for both passive scalar quantities and vorticity, for example, see [5, 6, 24, 34]. The use of deforming elements is a different approach which seeks to capture both convective and diffusive terms in a single semidiscrete system of equations.

Many investigators have explored both nonaxisymmetric and deforming basis functions in the hopes that they would adaptively resolve passive scalar fields or other nonlinear fields. Meiburg [19] developed a scheme in which radially symmetric blobs representing sections of a vortex sheet would expand or contract based on normal flow deviations, but this scheme violates incompressibility and is specialized for vortex sheets. Marshall and Grant [18] have used highly anisotropic elements to satisfy the no-slip, no normal flow boundary conditions efficiently. Teng [31, 32] and Teng, Ying, and Zhang [33] originally used rigid elliptical patches to resolve boundary layers more efficiently and later used deforming elliptical patches to simulate the evolution of vortex sheets. Teng establishes a theoretical $O(l^2 \log l)$ rate of convergence for his method, where l is the core width of the computational elements. Kida and Nakajima [10] and Kida, Nakajima, and Suemitsu [11] have used simple core spreading and a second-order core spreading method. The second-order method is similar to that of Lu and Ross [17] based on an integral correction and continual regridding of the computational elements. This group has shown that core spreading without refinement or corrections is reasonable for small times, while the corrected method is valid for all finite times [9, 10, 11]. Earlier, Rossi [25] developed a corrected core spreading vortex method (CCSVM) which refines elements that have grown beyond a specified core size. This maintains spatial resolution for all finite time at a cost of introducing more computational elements, although the problem size growth can be mitigated through a merging algorithm [26]. Ojima and Kamemoto [23] propose a scheme using deforming vortex elements that stretch with local flow deviations, but

the resulting element is replaced with an isotropic element of equal volume at the end of each time step, thus avoiding having to calculate the Biot–Savart integral for anisotropic elements. To simulate the convection *and diffusion* of passive scalar quantities, Leonard [13] used deforming elliptical Gaussian basis functions, noting that they remain self-similar under the linearized convection-diffusion equations. Also, Moeleker and Leonard [20] propose using anisotropic elements with velocity field corrections based on subgrid scale when computing Gaussian filtered scalar quantities, although they report modest gains in accuracy at a high computational cost. In their calculations, the velocity of each blob's centroid is a basis function weighted average plus subgrid scale contributions. We shall see in section 4 that the method has several connections to the method derived in this paper.

Along these lines, there have been a small number of investigations aimed at modifications to the velocity field in Lagrangian schemes to improve the quality of the computation. Leonard [12] noted many years ago that a vorticity weighted velocity computation for vortex methods would conserve total energy, while using the velocity field measured at the centroid would not. Monaghan [21, 22] has performed investigations with a method called XSPH, a variant of SPH computations, where velocity fields are averaged over nearby particles. He finds that the averaging process increases dispersion and "keeps the particles orderly in the absence of viscosity."

There are two important issues related to spatial accuracy which are beyond the scope of this paper. The Biot–Savart integral over an elliptical Gaussian basis function is essential if one were to develop a vortex method using these deforming basis functions. These issues have been resolved for both direct computations [27] and fast multipole summation but will not be treated here. Also, this paper will not discuss the remeshing, refinement, and merging of elements due to particle distortion effects. These methods are important if one wants to maintain spatial accuracy over extended periods of time. However, remeshing, refinement, and merging all involve additional sources of spatial error which must be controlled and add nothing to the discussion of the basic spatial accuracy associated with using deforming elements.

This paper is structured in the following manner. In section 2, we describe spatial accuracy and develop an approach that is particularly helpful when working with heterogeneous distributions of anisotropic particles. In section 3, we examine the spatial accuracy of deforming elliptical Gaussian basis functions if one uses centroid velocity data. In section 4, we find that the spatial accuracy can be improved if one corrects the velocity field. In section 5, we attempt to continue this process and will come to understand why one cannot indefinitely bootstrap the spatial accuracy of Lagrangian methods with elliptical Gaussians. In section 6, we derive the specific dynamics of elliptical Gaussian basis functions necessary to implement the scheme. Finally, in section 7, nontrivial experiments will verify the convergence properties determined in sections 3–5. In addition to summary remarks, section 8 includes some discussion of issues relevant to this work but beyond the scope of this paper.

2. Interpretations of particles and measuring spatial accuracy. This paper examines a class of particle schemes for approximating the incompressible convection-diffusion equations

$$\partial_t \rho + (\vec{u} \cdot \nabla)\rho = \frac{1}{\mathrm{Pe}} \nabla^2 \rho, \tag{2.1a}$$

$$\nabla \cdot \vec{u} = 0, \tag{2.1b}$$

where ρ is a passive scalar quantity that moves and diffuses with the flow. The $\vec{u}$ is the known fluid velocity field, and Pe is the Peclét number which is a dimensionless quantity expressing the ratio of convective to diffusive effects in the flow.

To simulate solutions to this PDE, we approximate the field as a linear combination of N moving basis functions or blobs,

$$\widehat{\rho} = \sum_{i=1}^{N} \gamma_i \phi(\vec{x} - \vec{x}_i, \dots), \tag{2.2}$$

where N is the total number of basis functions, γ_i is the amplitude of the ith basis function, and $\vec{x}_i$ is the centroid of the ith basis function. Each basis function may have other parameters as well, indicated by the ...'s. In a systematic manner, evolution equations for the $\vec{x}_i$'s and perhaps other parameters (indicated by ...) are determined, reducing the PDE (2.1) to a finite system of ODEs.

The interpretation of the basis function ϕ and its role is critical to this discussion. In the particle tracking literature, the common interpretation is that the computational element represents a finite amount of the passive scalar substance concentrated at a single point. A large number of elements are used to generate a statistical description of the passive scalar field, and the structure of ϕ plays no role at all. Thus, $\widehat{\rho}$ has no pointwise meaning at all, but $\langle \gamma_i \rangle$ represents the computation field where $\langle \cdot \rangle$ represents an average over an area with a sufficiently large number of particles. These techniques have low spatial accuracy, but the computations are embarrassingly parallel, so there is no motivation to improve the spatial accuracy. For vortex methods, the structure of ϕ plays a more important role because one must compute the velocity field from the vorticity field, and there has been a substantial amount of activity investigating different structures for ϕ. For most in the vortex methods community, the interpretation is still that the computational element represents a small amount of circulation concentrated at a single point in the fluid flow (for example see pages 19–21 in [4]). The justification for using a function ϕ that is something other than a delta distribution $\delta(\vec{x})$ is that it yields greater spatial accuracy in the velocity computation, but the vorticity field is still interpreted as $\langle \gamma_i \rangle$.

To analyze any numerical method, we must quantify its ability to approximate the PDEs which are satisfied by the exact solution. In this paper, we shall use $\widehat{\rho}$ as an induced prolongation for the numerical approximation of ρ. Therefore, we shall be comparing $\widehat{\rho}$ to ρ directly and studying the distance between these two functions as a measure of the convergence of the particle method. To measure this distance, we consider the residual

$$Rf = \partial_t f + \vec{u} \cdot \nabla f - \frac{1}{\mathrm{Pe}} \nabla^2 f, \tag{2.3}$$

where $\vec{u}$ is the fluid velocity field. Necessarily, the residual of the exact solution of (2.1) is zero: $R\rho = 0$. It can be shown that $R\widehat{\rho}$ is a measurement of the spatial error of a particle method if one treats $\widehat{\rho}$ as the induced prolongation of the numerical representation [25]. This residual is analogous to the notion of consistency for a finite difference scheme where one measures the extent to which the exact solution satisfies the difference equation for the approximate method [30]. Also, it can be contrasted with finite element methods where one satisfies a weak formulation over a class of functions over a finite element [14]. For particle methods, one measures the extent to which the prolongation satisfies the exact PDE via the residual. There is an analogue for stability that is related to the well posedness of the exact and approximate systems, but this will not be explored here. Some discussion can be found in [14, 25].

If we consider a simple scheme where basis functions move but do not deform, we would have (2.2) together with the autonomous system

$$\frac{d\vec{x}_i}{dt} = \widehat{u}(\vec{x}_i). \tag{2.4}$$

For most particle methods, computational elements move with the velocity of the fluid $\widehat{u} = \vec{u}$, but we shall see later that there are distinct advantages to using an altered velocity field.

If $\frac{1}{\mathrm{Pe}} = 0$, we see that

$$R\widehat{\rho} = \sum_{i=1}^{N} \gamma_i \nabla_{\vec{x}} \phi(\vec{x} - \vec{x}_i, \dots) \cdot \left[\widehat{u}(\vec{x}) - \vec{u}(\vec{x}_i)\right]. \tag{2.5}$$

If h represents the typical distance between elements and σ is the radius of support of ϕ, then when $\frac{h}{\sigma} \ll 1$, the residual can be expressed as a Riemann sum. In this limit, $\vec{x}_i$ becomes a continuous spatial variable $\vec{t}$ and $\gamma(\vec{t})$ can be interpreted as a continuous extension of γ_i:

$$R\widehat{\rho} \approx \iint_{-\infty}^{\infty} \frac{\gamma(\vec{t})}{h^2} \nabla_{\vec{x}} \phi(\vec{x} - \vec{t}, \dots) \cdot \left[\widehat{u}(\vec{x}) - \vec{u}(\vec{t})\right] d\vec{t}. \tag{2.6}$$

We observe that $\frac{\gamma}{h^2}$ is an appropriate macroscopic quantity analogous to a passive scalar density, and one expects this function to have a finite regular limit as $h \to 0$. Since the flow field is incompressible and ϕ is a function of the difference $\vec{x} - \vec{t}$ and there are no other spatial dependencies,

$$\begin{aligned}
R\widehat{\rho} &\approx \iint_{-\infty}^{\infty} \frac{\gamma(\vec{t})}{h^2} \nabla_{\vec{x}} \phi(\vec{x} - \vec{t}, \dots) \cdot \left[\widehat{u}(\vec{x}) - \vec{u}(\vec{t})\right] d\vec{t}, \\
&= \iint_{-\infty}^{\infty} \frac{\gamma(\vec{t})}{h^2} \nabla_{\vec{t}} \phi(\vec{x} - \vec{t}, \dots) \cdot \left[\widehat{u}(\vec{t}) - \vec{u}(\vec{x})\right] d\vec{t}, \\
&= -\iint_{-\infty}^{\infty} \phi(\vec{x} - \vec{t}, \dots) \frac{\nabla_{\vec{t}} \gamma(\vec{t})}{h^2} \cdot \left[\widehat{u}(\vec{t}) - \vec{u}(\vec{x})\right] d\vec{t}, \\
&= \iint_{-\infty}^{\infty} \phi(\vec{t}, \dots) \frac{\nabla_{\vec{t}} \gamma(\vec{x} - \vec{t})}{h^2} \cdot \left[\widehat{u}(\vec{x} - \vec{t}) - \vec{u}(\vec{x})\right] d\vec{t}.
\end{aligned} \tag{2.7}$$

We impose the weak requirement that the product of the velocity field and ϕ decay at infinity. Therefore, we see that the integral in (2.7) governs the order of accuracy of the method.

From this expression, one can see that the shape of the basis function can affect the accuracy of the method. If one considers that the term $\nabla_{\vec{t}} \gamma(\vec{x} - \vec{t}) \cdot \left[\widehat{u}(\vec{x} - \vec{t}) - \vec{u}(\vec{x})\right]$ can be written as a Taylor series in $\vec{t}$, the order of accuracy of the method depends upon the moments of ϕ. To determine the spatial accuracy, we define p to be the smallest integer such that $|\vec{k}| = p$ and

$$\iint_{-\infty}^{\infty} \phi(\vec{t}) \vec{t}^{\,\vec{k}} d\vec{t} \neq 0, \tag{2.8}$$

where $\vec{k} = [k_1, k_2]^T$, $k_1, k_2 > 0$ is a multi-index, $|\vec{k}| = k_1 + k_2$, and $\vec{t}^{\,\vec{k}} = s_1^{k_1} s_2^{k_2}$. Since the first $p - 1$ moments arising from the Taylor series will vanish, we see that

the method converges like σ^p, where σ is the width of the basis function ϕ. This is precisely the moment condition rigorously determined by Beale and Majda [1, 2, 3]. For instance, it is well known that Gaussian basis functions,

$$\phi(\vec{x}) = \frac{1}{4\pi\sigma^2} \exp\left(-\frac{|\vec{x}|^2}{4\sigma^2}\right), \tag{2.9}$$

will yield a second-order method if $\widehat{u} = \vec{u}$. In this case, the core width is σ. A Gaussian has a nontrivial zeroth moment, but if $\widehat{u} = \vec{u}$, then

$$\lim_{\vec{t}\to\vec{0}} \nabla_{\vec{t}}\gamma(\vec{x}-\vec{t}\,) \cdot \left[\widehat{u}(\vec{x}-\vec{t}\,) - \vec{u}(\vec{x})\right] = 0,$$

so the zeroth moment will not contribute to the spatial error. In other words, the zeroth-order coefficient of $\nabla_{\vec{t}}\gamma(\vec{x}-\vec{t}\,) \cdot \left[\widehat{u}(\vec{x}-\vec{t}\,) - \vec{u}(\vec{x})\right]$ is zero. This is true for any basis function shape. Since the Gaussian is radially symmetric, the first moments are zero. The second moments are not zero and make the leading order contribution to the error, so the method has a spatial accuracy of $O(\sigma^2)$. Of course, a high-order method is not necessarily highly accurate, and an effective scheme is one that realizes a high-order convergence rate and is accurate in typical parameter regimes.

3. Deforming basis functions. If we wish to include diffusion ($\frac{1}{\mathrm{Pe}} \neq 0$), particle methods become much more complex, and there are two basic approaches. The first and most common among practitioners of Lagrangian methods is operator splitting, where one divides each time integration step into a convective step and a diffusive step. The second approach is the focus of this work, where we capture diffusion and convection together in one system of ODEs by using deforming basis functions.

The best studied of these methods are the core spreading methods and corrected core spreading methods [9, 10, 11, 25, 26]. The approach is based on the observation that there are known exact solutions to the PDE

$$\partial_t\rho + \widehat{u}\cdot\nabla\rho = \frac{1}{\mathrm{Pe}}\nabla^2\rho, \tag{3.1}$$

where $\widehat{u} \equiv \widehat{u}(t)$ is a spatially constant vector. The solutions have the form of a moving spreading blob:

$$\phi(\vec{x},t) = \frac{1}{4\pi\sigma_i^2} \exp\left[-\frac{|\vec{x}-\vec{x}_i|^2}{4\sigma_i^2}\right], \tag{3.2a}$$

$$\frac{d\sigma_i^2}{dt} = \frac{1}{\mathrm{Pe}}, \tag{3.2b}$$

$$\frac{d\vec{x}_i}{dt} = \widehat{u}. \tag{3.2c}$$

Such elements have an additional degree of freedom corresponding to the core size, and having this additional degree of freedom allows one to capture the diffusion term in (3.1). This deformation has an adverse effect on schemes using cores that satisfy the moment condition. For instance, Beale and Majda [3] constructed a fourth-order inviscid method by superposing two Gaussian basis functions. However, if one attempts to capture diffusion with core spreading, the growth of the cores will destroy the cancellations in the moment conditions, so that the viscous method is second-order and the advantage over having single Gaussian elements is lost. To simulate a flow, one would arrange a number of such elements to approximate the initial scalar field by specifying initial conditions for the position and width of every basis function

described by (3.2). Since the basis functions for a particle method are localized, we presume that the centroid velocity field is close to uniform over the support of the basis function. Thus, if we are trying to solve (2.1), the conventional wisdom is to set $\widehat{u}$ for basis function i equal to centroid velocity field $\vec{u}(\vec{x}_i)$. Of course, $\vec{u}$ does vary in general over the support of each of the basis functions, and this is the fundamental source of spatial error in the method. After a little analysis, one finds that the accuracy of the method (3.2) is $O(\sigma^2)$.

Since the core spreads at a constant rate, the accuracy has a finite lower bound independent of the initial conditions on the basis functions. This property leads to the conclusion that the method, if uncorrected, is fundamentally inconsistent [7]. For vortex calculations, the method was corrected by introducing a spatial refinement procedure which replaces elements that grow beyond any specified tolerance with a configuration of thinner elements [25]. Also, a technique for replacing nearly overlapping elements with a single element as long as the operation induces an error below a specified tolerance was developed to control the growth of N through refinement [26]. An alternative to spatial refinement is to use remeshing techniques discussed in some generality in [4]. The fundamental goal with all of these methods is to replace a configuration of particles with undesirable properties with a configuration of particles with desirable properties while disturbing the field $\widehat{\rho}$ as little as possible. Kida and Nakajima [10] and Kida, Nakajima, and Suemitsu [11] remind us that the lower bound on the error may be quite small and that meaningful computations can be performed for short times without correcting the method at all. With this in mind, we restrict our discussion to spatial accuracy with the understanding that the Peclét number limits the accuracy of the method and that remeshing or refinement could be introduced to augment the method.

If core spreading achieves second-order accuracy for (2.1) by taking a zeroth-order approximation to the velocity field (3.1), one can seek improvements by using basis functions that solve (2.1) with a linear approximation to the velocity field. Suppose we have a basis function ϕ that is an exact solution to the convection-diffusion equations with a linear flow field

$$\partial_t \rho + \{\widehat{u} + \mathbf{F}[\vec{x}]\} \cdot \nabla \rho - \frac{1}{\mathrm{Pe}} \nabla^2 \rho = 0, \tag{3.3}$$

where $\widehat{u} \equiv \widehat{u}(t)$ is a vector-valued function as before and $\mathbf{F}$ is a 2×2 linear operator. Assuming that a deforming basis function ϕ can be found that will satisfy (3.3), conventional wisdom suggests that we should let $\widehat{u} = \vec{u}(\vec{x}_i)$ and $\mathbf{F} = D\vec{u}(\vec{x}_i)$, where $D\vec{u}$ is the matrix of partial derivatives of the flow velocity,

$$\mathbf{F} = \begin{bmatrix} u_x(\vec{x}_i) & u_y(\vec{x}_i) \\ v_x(\vec{x}_i) & v_y(\vec{x}_i) \end{bmatrix}, \text{ and } \vec{u} = \begin{bmatrix} u \\ v \end{bmatrix}. \tag{3.4}$$

The linearized convection-diffusion equation (3.3) permits solutions that take the form of elliptical Gaussians:

$$\begin{aligned} &\phi(\vec{x}; \vec{x}_i, \sigma_i, a_i, \theta_i) \\ &\quad = \frac{1}{4\pi\sigma_i^2} \exp\left\{ - \frac{[c_i(x - x_i) + s_i(y - y_i)]^2 / a_i^2 + [-s_i(x - x_i) + c_i(y - y_i)]^2 a_i^2}{4\sigma_i^2} \right\}, \end{aligned} \tag{3.5}$$

where $\vec{x} = \left[\begin{smallmatrix} x \\ y \end{smallmatrix}\right]$, $c_i = \cos(\theta_i)$, and $s_i = \sin(\theta_i)$. In addition to the usual parameters of amplitude (γ_i), position ($\vec{x}_i$), and width (σ_i), these basis functions have an orientation (θ_i) and an aspect ratio (a_i^2) as shown in Figure 1. The evolution of all of these parameters is discussed in detail in section 6. While the evolution equations are

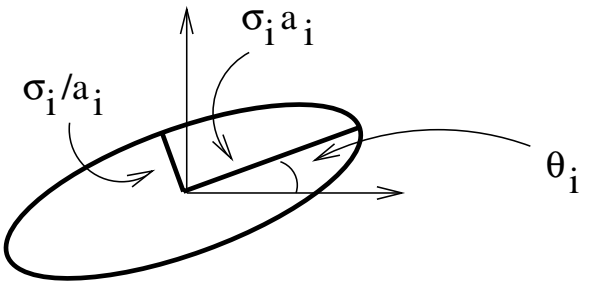

FIG. 1. *A schematic diagram of an elliptical Gaussian basis function as defined in* (3.5). *The ellipse is representative of a single level set.*

central to any implementation of the method, one does not need to know them to study the spatial accuracy of the system; one only needs to know that the basis function is an exact solution to the approximate PDE (3.3).

If we compute the residual using deforming elliptical Gaussians following the same procedures as described above, we find

$$R\rho \approx \iint_{-\infty}^{\infty} \phi(\vec{t},\dots)\frac{\nabla_{\vec{t}}\gamma(\vec{x}-\vec{t})}{h^2}\cdot[\widehat{u}(\vec{x}-\vec{t})+\mathbf{F}(\vec{x}-\vec{t})[\vec{t}]-\vec{u}(\vec{x})]d\vec{t}. \tag{3.6}$$

In addition to using a continuation of the amplitudes γ_i as before, this form also employs continuations of θ_i, a_i^2, and σ_i because one must permit inhomogeneous collections of basis functions. If we were to use the centroid velocity and velocity derivatives for $\widehat{u}$ and $\mathbf{F}$, respectively, and seek the leading order term from the residual, we find

$$\begin{aligned} R\rho &\approx \iint_{-\infty}^{\infty} \phi(\vec{t})\frac{\nabla_{\vec{t}}\gamma(\vec{x}-\vec{t})}{h^2}\cdot\left\{\frac{1}{2}D^2\vec{u}(\vec{x})[\vec{t},\vec{t}]+\dots\right\}d\vec{t} \\ &\approx \iint_{-\infty}^{\infty} \phi(\vec{t})\frac{\nabla\gamma(\vec{x})}{h^2}\cdot\left\{\frac{1}{2}D^2\vec{u}(\vec{x})[\vec{t},\vec{t}]+\dots\right\}d\vec{t}, \end{aligned} \tag{3.7}$$

where $D^2\vec{u}$ is the bilinear operator of second partial derivatives of the velocity field. For instance, if $f(x)$ is a scalar function, Df is its gradient. Thus, if one were to write out all the components, one would have

$$D^2\vec{u}(\vec{x})[\vec{t},\vec{t}] = \begin{bmatrix} u_{xx}t_x^2+2u_{xy}t_xt_y+u_{yy}t_y^2 \\ v_{xx}t_x^2+2v_{xy}t_xt_y+v_{yy}t_y^2 \end{bmatrix}, \text{ where } \vec{t}=\begin{bmatrix} t_x \\ t_y \end{bmatrix}. \tag{3.8}$$

To compute these moments, we will treat this integral as if the basis function parameters do not vary over the support of the computational element. This is not an additional source of error because inhomogeneities in a^2, σ, or θ could be included in the $\nabla\gamma$ function which has no requirements placed upon it. When examining the residual, it is redundant to calculate terms involving both $\vec{u}$ components (u and v) because $\nabla\gamma$ can be anything. Therefore, we will replace the vector field $\vec{u}$ with the scalar function u for the remainder of this paper with the understanding that results for u can be applied directly to v by substituting one function for the other. For the leading order moment in (3.7), we find that

$$\iint_{-\infty}^{\infty} \phi(\vec{t})D^2u(\vec{x})[\vec{t},\vec{t}]d\vec{t} = 2\sigma^2\left(u_{xx}M_{xx}+2u_{xy}M_{xy}+u_{yy}M_{yy}\right), \tag{3.9}$$

where

$$M_{xx} = c^2a^2 + s^2/a^2, \tag{3.10a}$$

$$M_{xy} = cs(a^2 - a^{-2}), \tag{3.10b}$$

$$M_{yy} = c^2/a^2 + s^2a^2. \tag{3.10c}$$

Thus, we see that using deforming basis functions that move and deform based solely on the linearization of the centroid velocity field offers no improvement in the order of spatial accuracy, and one might as well use the simpler core spreading method in (3.2) and save the trouble of implementing a more complex system. However, we will not admit defeat when we have a good grip on the leading order source of error. Instead, we will determine a $\widehat{u}$ and $\mathbf{F}$ such that this leading order term will disappear.

4. Improvements using corrected velocity fields. We shall define

$$\widehat{u} = u + w, \tag{4.1}$$

$$\mathbf{F} = Du + \mathbf{Dw}, \tag{4.2}$$

as altered velocity data to improve the spatial accuracy of the deformed basis functions. The functions w and $\mathbf{Dw}$ are considered corrections to the linearized velocity field at the basis function centroid. Note that $\mathbf{Dw}$ is so designated for consistency of notation, but it is not necessarily equivalent to Dw. Substituting these into (3.6) and calculating the Taylor series for the u component,

(4.3)

$$\begin{aligned}
R\rho \approx \frac{1}{h^2}\iint_{-\infty}^{\infty} \phi(\vec{t}) &\left[\underbrace{\frac{\partial\gamma}{\partial x}(\vec{x})}_{(c)} - D\frac{\partial\gamma}{\partial x}(\vec{x})[\vec{t}] + \dots\right] \\
&\times\Bigg\{\underbrace{u(\vec{x})}_{(a)} - \underbrace{Du(\vec{x})[\vec{t}]}_{(b)} + \underbrace{\frac{1}{2}D^2u(\vec{x})[\vec{t},\vec{t}]}_{(c)} - \frac{1}{3!}D^3u[\vec{t},\vec{t},\vec{t}] + \frac{1}{4!}D^4u(\vec{x})[\vec{t},\vec{t},\vec{t},\vec{t}] + \dots \\
&+ \underbrace{w(\vec{x})}_{(c)} - Dw(\vec{x})[\vec{t}] + \frac{1}{2}D^2w(\vec{x})[\vec{t},\vec{t}] - \frac{1}{3!}D^3w(\vec{x})[\vec{t},\vec{t},\vec{t}] + \dots \\
&+ \underbrace{Du(\vec{x})[\vec{t}]}_{(b)} - \underbrace{D^2u(\vec{x})[\vec{t},\vec{t}]}_{(c)} + \frac{1}{2}D^3u(\vec{x})[\vec{t},\vec{t},\vec{t}] - \frac{1}{3!}D^4u(\vec{x})[\vec{t},\vec{t},\vec{t},\vec{t}] + \dots \\
&+ \mathbf{Dw}(\vec{x})[\vec{t}] - D\mathbf{Dw}(\vec{x})[\vec{t},\vec{t}] + \frac{1}{2}D^2\mathbf{Dw}(\vec{x})[\vec{t},\vec{t},\vec{t}] + \dots - \underbrace{u(\vec{x})}_{(a)}\Bigg\}\, d\vec{t}.
\end{aligned}$$

One can see that terms (a) and (b) cancel exactly, but term (c) does not unless we choose $w(\vec{x})$ such that it cancels the other terms. To eliminate term (c), we observe that $\iint \phi(\vec{t})d\vec{t} = 1$ and define

$$\begin{aligned}
w(\vec{x}) &= \iint \phi(\vec{t})\frac{1}{2}D^2u(\vec{x})[\vec{t},\vec{t}]d\vec{t} \\
&= \sigma^2\left[u_{xx}(\vec{x})M_{xx} + 2u_{xy}(\vec{x})M_{xy} + u_{yy}(\vec{x})M_{yy}\right].
\end{aligned} \tag{4.4}$$

Thus, we see that using velocity data at the centroid hobbles a deforming blob method to second order, and the correction w from (4.4) in the velocity field allows the method to achieve fourth-order spatial accuracy. The adjustment $\mathbf{D}w$ plays no role at this point.

The velocity field correction has connections to the work of Moeleker and Leonard [20], who use elliptical Gaussian elements to compute filtered passive-scalar quantities. If one neglects their subgrid scale corrections, one is left with a method where basis functions move with a basis function weighted average of the velocity field which agrees to $O(\sigma^4)$ with the scheme derived in this section. As noted earlier, $\widehat{u} = \vec{u}$ arises through a truncated Taylor series. Moeleker and Leonard [20] determine the averaged velocity term by expanding and truncating the convection-diffusion equations in a Hermite polynomial series. Different truncations and approaches will lead to different evolution equations for the blobs. In this paper, we determine the evolution equations by minimizing the residual of the total computed field in powers of σ as a direct means of controlling the spatial accuracy.

5. "Please, Sir, may I have some more [accuracy]?" There is no a priori reason why the corrections should stop at this point. In fact, one could anticipate a series of corrections of the form

$$w(\vec{x}) = \sigma^2 w_1(\vec{x}) + \sigma^4 w_2(\vec{x}) + \dots, \tag{5.1a}$$

$$\mathbf{Dw}(\vec{x}) = \sigma^2 \mathbf{Dw_1}(\vec{x}) + \sigma^4 \mathbf{Dw_2}(\vec{x}) + \dots. \tag{5.1b}$$

We know from (4.4) that

$$w_1(\vec{x}) = u_{xx}(\vec{x}) M_{xx} + 2u_{xy}(\vec{x}) M_{xy} + u_{yy}(\vec{x}) M_{yy}. \tag{5.2}$$

With this correction, we can cancel terms from (4.3) and write down the leading order errors if the velocity field is corrected with w_1. Things become slightly more complicated because errors arise through interactions with derivatives of $\frac{\partial \gamma}{\partial x}$:

$$\begin{aligned} R\rho \approx \frac{1}{h^2} \iint_{-\infty}^{\infty} \phi(\vec{t}) \Bigg[& \underbrace{\frac{\partial \gamma}{\partial x}(\vec{x})}_{(c,d)} - \underbrace{D\frac{\partial \gamma}{\partial x}(\vec{x})[\vec{t}]}_{(e)} + \underbrace{\frac{1}{2} D^2 \frac{\partial \gamma}{\partial x}(\vec{x})[\vec{t},\vec{t}]}_{(X)} + \dots \Bigg] \\ & \times \Bigg\{ \underbrace{\frac{1}{2} D^2 u(\vec{x})[\vec{t},\vec{t}]}_{(c),(X)} - \underbrace{\frac{1}{3!} D^3 u[\vec{t},\vec{t},\vec{t}]}_{(e)} + \underbrace{\frac{1}{4!} D^4 u(\vec{x})[\vec{t},\vec{t},\vec{t},\vec{t}]}_{(d)} + \dots \\ & + \underbrace{\sigma^4 w_2(\vec{x})}_{(d)} - \underbrace{\sigma^2 D w_1(\vec{x})[\vec{t}]}_{(e)} + \underbrace{\sigma^2 \frac{1}{2} D^2 w_1(\vec{x})[\vec{t},\vec{t}]}_{(d)} + \dots \\ & + -\underbrace{D^2 u(\vec{x})[\vec{t},\vec{t}]}_{(c),(X)} + \underbrace{\frac{1}{2} D^3 u(\vec{x})[\vec{t},\vec{t},\vec{t}]}_{(e)} - \underbrace{\frac{1}{3!} D^4 u(\vec{x})[\vec{t},\vec{t},\vec{t},\vec{t}]}_{(d)} + \dots \\ & + \underbrace{\sigma^2 \mathbf{Dw_1}(\vec{x})[\vec{t}]}_{(e)} - \underbrace{\sigma^2 D \mathbf{Dw_1}(\vec{x})[\vec{t},\vec{t}]}_{(d)} + \dots \Bigg\} d\vec{t}. \end{aligned} \tag{5.3}$$

Here we see that there are three groups that contribute to fourth-order spatial accuracy, but there are only two functions w_2 and $\mathbf{Dw_1}$ available. A proper choice for w_2 and $\mathbf{Dw_1}$ will cancel terms (d) and (e), respectively.

(5.4a)
$$\sigma^2 \iint \phi(\vec{t})\mathbf{Dw_1}(\vec{x})[\vec{t}]t_x d\vec{t} = \iint \phi(\vec{t}) \left\{ -\frac{1}{3} D^3 u[\vec{t},\vec{t},\vec{t}] + \sigma^2 D w_1(\vec{x})[\vec{t}] \right\} t_x d\vec{t},$$

(5.4b)
$$\sigma^2 \iint \phi(\vec{t})\mathbf{Dw_1}(\vec{x})[\vec{t}]t_y d\vec{t} = \iint \phi(\vec{t}) \left\{ -\frac{1}{3} D^3 u[\vec{t},\vec{t},\vec{t}] + \sigma^2 D w_1(\vec{x})[\vec{t}] \right\} t_y d\vec{t},$$

(5.4c)
$$\sigma^4 w_2(\vec{x}) = \iint \phi(\vec{t}) \left\{ \frac{1}{8} D^4 u(\vec{x})[\vec{t},\vec{t},\vec{t},\vec{t}] - \sigma^2 D\mathbf{Dw_1}(\vec{x})[\vec{t},\vec{t}] - \sigma^2 \frac{1}{2} D^2 w_1(\vec{x})[\vec{t},\vec{t}] \right\} d\vec{t}.$$

The solution to this system is

(5.5a)
$$\mathbf{Dw_1}(\vec{x}) = -\begin{bmatrix} u_{xxx}(\vec{x})M_{xx} + 2u_{xxy}(\vec{x})M_{xy} + u_{xyy}(\vec{x})M_{yy} \\ u_{xxy}(\vec{x})M_{xx} + 2u_{xyy}(\vec{x})M_{xy} + u_{yyy}(\vec{x})M_{yy} \end{bmatrix},$$

(5.5b)
$$\begin{aligned} w_2(\vec{x}) = \frac{5}{2} \big[& u_{xxxx}(\vec{x})M_{xx}^2 + 4u_{xxxy}(\vec{x})M_{xx}M_{xy} \\ & + 2u_{xxyy}(\vec{x})(M_{xx}M_{yy} + 2M_{xy}^2) + 4u_{xyyy}(\vec{x})M_{yy}M_{xy} \\ & + u_{yyyy}(\vec{x})M_{yy}^2 \big]. \end{aligned}$$

However, term (X) lends a nontrivial contribution, and there are no undetermined functions remaining to eliminate it. Thus, we see that exact cancellations like (a) and (b) in (4.3) are critical because the velocity correction method has strict limitations. Of course, higher order spatial accuracy would be possible if one used a class of basis functions that more closely solves (2.1). For instance, one could use basis functions that solve (2.1) for quadratic flows, in which case exact cancellations would eliminate all terms at second order, and we anticipate that corrections to the velocity field evaluations could be applied at fourth order.

6. The dynamics of deforming elliptical Gaussian basis functions. We begin by choosing the origin to be the center of the basis function and orienting (3.5) along the principal axes ($\theta = \theta_i$) which we shall refer to as X and Y. We force this coordinate system to satisfy the local equation (3.3). We denote the X and Y components of $\widehat{u}$ as u and v, respectively. Expanding in powers of X and Y, we can equate each coefficient with zero to derive the appropriate dynamical system, and set $\theta = 0$. In the interest of brevity, we drop all indices so that we can focus on the dynamics of an individual basis function, except where $\vec{x}$ and $\vec{x}_i$ are used to

distinguish between the independent variable and the position of the vortex element.

$$O(1): \quad -\frac{1}{\sigma^2}\frac{d}{dt}(\sigma^2) + \frac{1}{\mathrm{Pe}}\frac{\left(a^{-2}+a^2\right)}{2\sigma^2} = 0, \tag{6.1a}$$

$$O(X): \quad a^{-2}\left(\frac{d}{dt}x_i - u\right) = 0, \tag{6.1b}$$

$$O(Y): \quad a^{2}\left(\frac{d}{dt}y_i - v\right) = 0, \tag{6.1c}$$

$$O(X^2): \quad \frac{d}{dt}(\sigma^2)a^{-2} + \sigma^2 a^{-4}\frac{d}{dt}(a^2) - 2\sigma^2 F_{11}a^{-2} - \frac{1}{\mathrm{Pe}}a^{-4} = 0, \tag{6.1d}$$

$$O(Y^2): \quad \frac{d}{dt}(\sigma^2)a^{2} - \sigma^2\frac{d}{dt}(a^2) + 2\sigma^2 F_{11}a^{2} - \frac{1}{\mathrm{Pe}}a^{4} = 0, \tag{6.1e}$$

$$O(XY): \quad -\sigma^2\frac{d\theta}{dt}(a^{-2}-a^2) - \sigma^2[F_{12}a^{-2} + F_{21}a^2] = 0, \tag{6.1f}$$

where

$$\mathbf{F} = \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & -F_{11} \end{bmatrix}. \tag{6.2}$$

From the previous expressions, we can write down the following evolution equations for the basis function parameters:

$$\frac{d\vec{x}_i}{dt} = \widehat{u}, \tag{6.3a}$$

$$\frac{d\sigma^2}{dt} = \frac{1}{2\mathrm{Pe}}(a^2 + a^{-2}), \tag{6.3b}$$

$$\frac{da^2}{dt} = 2a^2F_{11} + \frac{1}{2\sigma^2\mathrm{Pe}}(1-a^4), \tag{6.3c}$$

$$\frac{d\theta}{dt} = \frac{F_{12}a^{-2} + F_{21}a^2}{a^2 - a^{-2}}. \tag{6.3d}$$

For practical applications, we must transform $\vec{x}_i$, $\widehat{u}$, $\mathbf{F}$ into a laboratory reference frame so that

$$\mathbf{F} \leftarrow R_\theta \mathbf{F} R_{(-\theta)}, \tag{6.4}$$

where R_θ is a standard rotation matrix. Then, we can rewrite (6.3) as

$$\frac{d}{dt}\vec{x}_i = \widehat{u}, \tag{6.5a}$$

$$\frac{d}{dt}(\sigma^2) = \frac{1}{2\mathrm{Pe}}(a^2 + a^{-2}), \tag{6.5b}$$

$$\frac{d}{dt}(a^2) = 2\left[F_{11}(c^2-s^2) + (F_{12}+F_{21})sc\right]a^2 + \frac{1}{2\sigma^2\mathrm{Pe}}(1-a^4), \tag{6.5c}$$

$$\frac{d}{dt}\theta = \frac{F_{21}-F_{12}}{2} + \left[\frac{F_{21}+F_{12}}{2}(s^2-c^2) + 2F_{11}sc\right]\frac{(a^{-2}+a^2)}{(a^{-2}-a^2)}. \tag{6.5d}$$

The decomposition of the evolution equation for orientation into rotational and strain-induced motions in equation (6.5d) is accomplished through further algebraic manipulation.

From system (6.5), we observe certain properties in these elliptical Gaussians. The first equation is no surprise. From the second equation, we see that elongation of the basis functions augments spreading as one might expect. As with all core spreading methods, the width, σ, must be numerically controlled by adaptive refinement or remeshing, but this is beyond the scope of this paper.

In (6.5c), one can see that in the absence of viscosity, basis functions can elongate exponentially, and the growth of the aspect ratio is bounded above by

$$a^2 \leq e^{2\lambda_M t}, \tag{6.6}$$

where λ_M is the largest positive value of λ_i over a relevant spatial or temporal domain and

$$\lambda_i = F_{11,i}(c_i^2 - s_i^2) + (F_{12,i} + F_{21,i})s_i c_i. \tag{6.7}$$

(Indices are included because λ_M represents a bound over all basis functions.) We assume $\widehat{u}$ and its derivatives are known and bounded, so one can be certain that λ_M exists. Deforming basis functions can be used for infinite Pe calculations, but it is important to note that unlimited growth in the aspect ratio as $t \to \infty$ is catastrophic for spatial accuracy. In section 7, we shall present examples with finite and infinite Peclét number.

To analyze this method, the typical domain is the trajectory of a computational element through the flow, or perhaps the entire flow over all space and time. The bound, λ_M, is valid for all indices, i, and all time and assumes unit initial aspect ratios. If $\frac{1}{\mathrm{Pe}} \neq 0$, viscosity naturally regulates the elongation, and there is an upper bound, $a_i^2 \leq a_M^2$:

$$a_M^2 = 2\mathrm{Pe}\lambda_M l^2 + \sqrt{4\mathrm{Pe}^2\lambda_M^2 l^4 + 1} = \mathrm{Pe}_{\mathcal{E}} + \sqrt{\mathrm{Pe}_{\mathcal{E}}^2 + 1}. \tag{6.8}$$

Thus, the maximum elongation of each basis function is controlled by the dimensionless quantity

$$\mathrm{Pe}_{\mathcal{E}} = 2\mathrm{Pe}\lambda_M l^2, \tag{6.9}$$

which from this point on shall be referred to as the *basis function Peclét number* somewhat related to the grid Peclét number used to analyze finite difference schemes.

We know that the spatial accuracy of the method depends upon l, where $\sigma \leq l$ and $a^2 - a^{-2}$. Here, we find that if the Peclét number is finite, diffusion can check the growth of a^2. Thus, an investigator might be tempted to use two numerical parameters to control l and $a^2 - a^{-2}$, respectively. In fact, we have just seen that this is not necessary. For small l (and, therefore, small $\mathrm{Pe}_{\mathcal{E}}$), we can expand (6.8) as

$$a_M^2 \approx 1 + \mathrm{Pe}_{\mathcal{E}} + \frac{1}{2}\mathrm{Pe}_{\mathcal{E}}^2 + \ldots = 1 + O(l^2). \tag{6.10}$$

Thus, $a^2 - a^{-2} = O(l^2)$, and l is the only relevant numerical parameter for this scheme, and no additional remeshing based on aspect ratio is required to allow the method to converge.

In (6.5d), we see that rotation is induced through two distinct mechanisms. The first term on the right-hand side corresponds to revolution due to pure rotation in $\mathbf{F}$. The second term corresponds to rotation induced by strain that is not aligned with

the major or minor axis. This latter term can be a source of stiffness in the system if a_i is close to 1. If a_i is not close to unity, the evolution equations can be solved directly.

Since one expects some aspect ratios to remain close to 1 in some flow regimes, it is crucial that one deal with this stiffness when solving the evolution equations in system (6.5) for the vorticity field. When a_i is close to unity, the orientation of the element rapidly moves toward an equilibrium where the two terms in the evolution equation balance one another. Dropping indices from this point onward, we can calculate these equilibria by setting $\frac{d}{dt}\theta = 0$ in (6.5):

$$\frac{F_{21} - F_{12}}{2}(a^{-2} - a^2)(c^2 + s^2) + \left[\frac{F_{21} + F_{12}}{2}(s^2 - c^2) + 2d_{11}sc\right](a^{-2} + a^2) = 0. \tag{6.11}$$

Collecting like terms in c and s and dividing by c^2, one arrives at a quadratic equation for $\tan\theta$. Solving this equation, one finds that the four equilibria (two from the quadratic plus two from the twofold symmetry of the basis function) are described by

$$\begin{aligned} \tan(\theta^{\pm}) &= -k \pm \sqrt{k^2 + \frac{a^{-2}F_{12} + a^2F_{21}}{a^2F_{12} + a^{-2}F_{21}}}, \\ k &= \frac{F_{11}(a^2 + a^{-2})}{a^2F_{12} + a^{-2}F_{21}}. \end{aligned} \tag{6.12}$$

Of course, one root corresponds to a stable equilibrium and the other corresponds to an unstable equilibrium. If a is close to unity and the problem is very stiff, one can effectively and accurately integrate (6.5) by assuming that the orientation of the computational element is evolving through stable, local equilibrium so that $\theta \equiv \theta(\mathbf{F}, a)$, found by solving (6.12). Therefore, one integrates only a and σ in time. After treating the potential stiffness in θ, we can use standard methods, such as "Adams family" integrators, to solve (6.5).

7. Demonstration of convergence properties. The most useful examples of exact solutions to mixing problems involve steady flows with closed streamlines. If $\frac{1}{\mathrm{Pe}} = 0$ and ρ is constant on streamlines initially, the initial scalar field will remain unchanged for all time. Even though computational elements will move and deform substantially, the exact solution will not. As an aside, it is tempting to perform computations with an arbitrary reversible flow field so that the exact scalar field returns to its initial configuration after some time T. The weakness of diagnostics such as these is that the evolution equations for the particles are also reversible, so the particle method will appear suspiciously accurate. However, this test is really a measurement of the temporal accuracy of the time integration routines being used and not the spatial accuracy of the particle method.

When $\frac{1}{\mathrm{Pe}} \neq 0$, one can follow a similar approach and use flows with closed *circular* streamlines. Since convective mixing has been eliminated for the exact problem, one can determine an exact solution easily by convolving the initial distribution with a Green's function for the diffusion equation. For sample calculations, we shall study the flow field induced by the stream function

$$\psi = \frac{\pi}{4}r^2 + \delta r^2(r^4 + c_1r^2 + c_2), \tag{7.1}$$

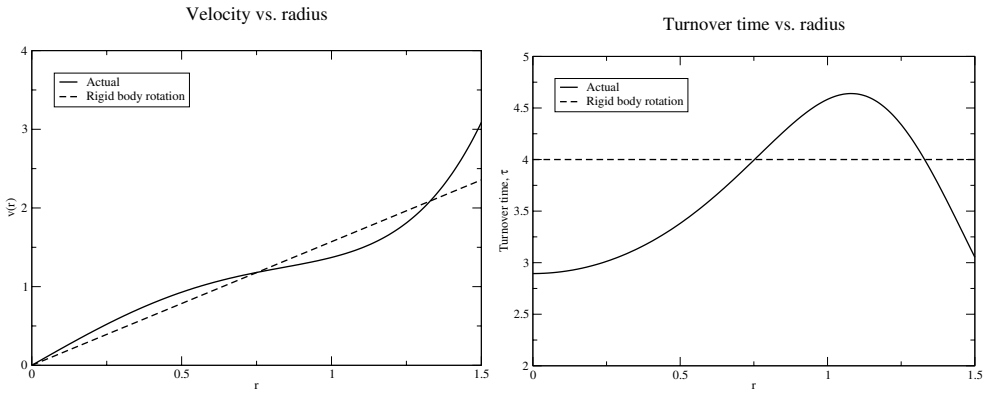

FIG. 2. *Velocity and turnover time for the flow field described by* ψ *in* (7.1) *with* $\delta = \frac{1}{10}$, $c_1 = -\frac{7}{2}$, *and* $c_2 = 3$. *Rigid body rotation is shown for comparison.*

where $r = |\vec{x}|$. If $\delta = 0$, the stream function corresponds to rigid body rotation with a turnover time of $T = 4$. In this case, the basis functions will not deform because local velocity deviations will correspond to rotation only, and the method is extremely close to exact. When we perturb the rigid body rotation, we introduce differential rotation and shearing while retaining the circular streamlines. For initial conditions, we select

$$\rho(r, 0) = 4 \exp\left(-4r^2\right). \tag{7.2}$$

Since $\rho \equiv \rho(r)$, we can write down the exact solution for all time by evaluating a convolution with the Green's function for the diffusion equation with the initial data (7.2),

$$\rho_{\text{ex}}(r, t) = \left(\frac{1}{16} + \frac{t}{\text{Pe}}\right)^{-1} \exp\left(-\frac{r^2}{\frac{1}{16} + \frac{t}{\text{Pe}}}\right). \tag{7.3}$$

For the perturbation, we choose $\delta = \frac{1}{10}$, $c_1 = -\frac{7}{2}$, and $c_2 = 3$. The properties of this flow field are shown in Figure 2. For the disturbed velocity profile, turnover times vary considerably over the support of the passive scalar field and there are large induced shears.

The initial conditions are generated by laying down particles on a regular grid inside a disk of radius 5/4. The amplitude is chosen using the exact-deregularization procedure described in [28], so that the initial L_2 error is less than 10^{-6} for all experiments. Initial core sizes (σ^2) were chosen to be 2.56×10^{-2}, 1.28×10^{-2}, 6.4×10^{-3}, 3.2×10^{-3}, 1.6×10^{-3}, and 8.0×10^{-4}. The initial distance between each basis function is $\frac{\sigma}{h} = 2$ to ensure reasonable overlap, and we find that increasing this ratio has no significant impact on spatial error or convergence rates.

The evolution of the scalar quantity was simulated for one unperturbed turnover time ($T = 4$) and compared with the exact solution (7.3). The results are shown in Figure 3. The simulations included Peclét numbers $\text{Pe} = \infty$, $\text{Pe} = 10^4$, and $\text{Pe} = 10^3$ with and without the velocity field corrections. We solve the system of ODEs (6.5) with third-order Adams–Bashforth time-stepping. The time step is chosen to be very small ($\Delta t = 10^{-3}$) so that method can be considered semidiscrete for all intents and purposes. On a 2.53 gigahertz Pentium 4 computer, a single time step required

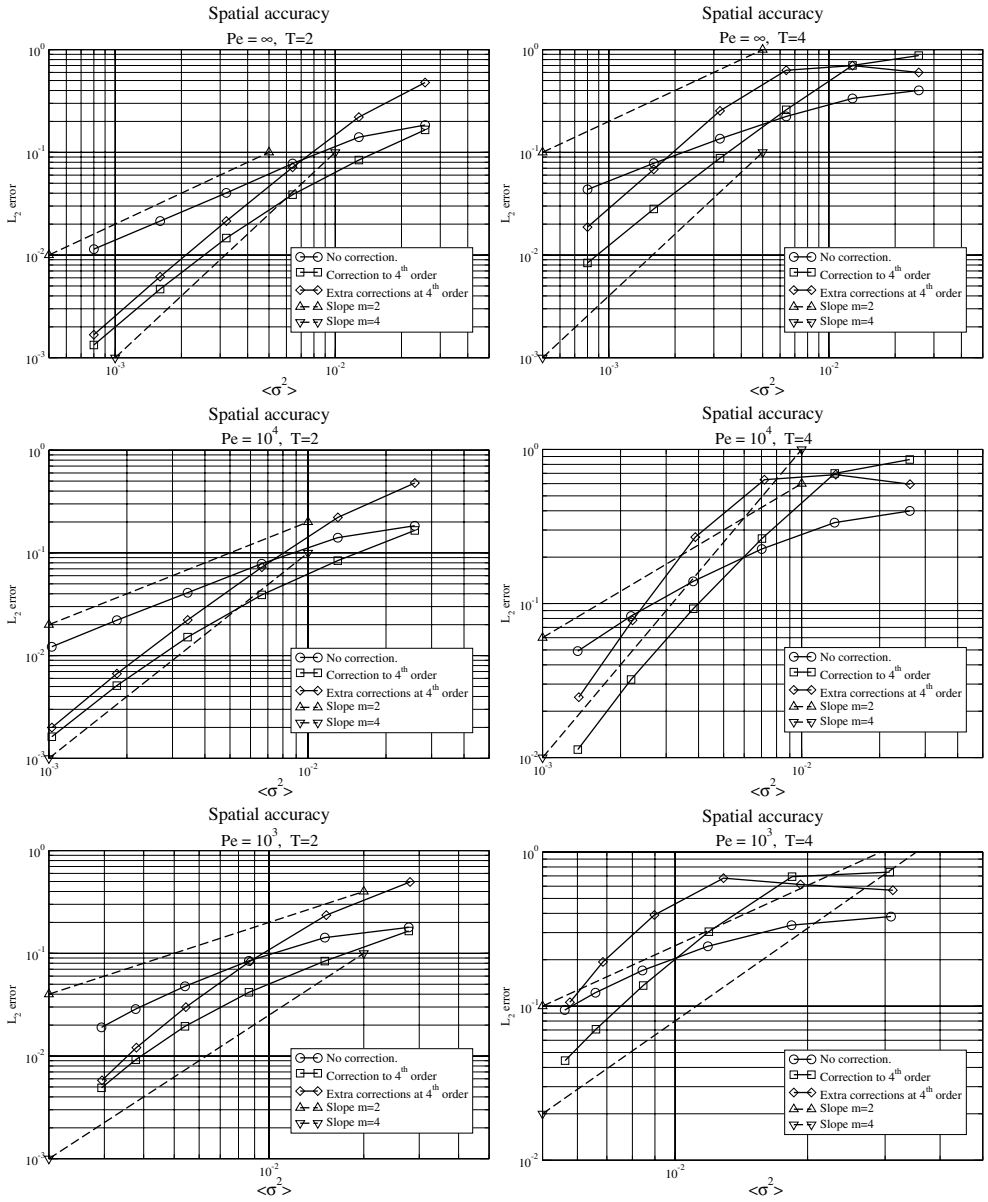

FIG. 3. *Improved spatial convergence properties for the convection-diffusion equations at three different Peclét numbers. The characteristic core width* $\langle \sigma^2 \rangle$ *is the scalar-weighted average core width at* $T = 2$ *and* $T = 4$*, respectively. The* L_2 *field error is calculated after one half and one full turnover time. These graphs demonstrate the improved spatial accuracy when using corrected velocity fields (*$\Box$*'s) compared to using centroid velocity data (*$\bigcirc$*'s). The third curve (*$\Diamond$*'s) is data when some of the fourth-order terms are eliminated using* (5.4)*. It is impossible to eliminate all of them using elliptical Gaussian basis functions with the velocity correction technique proposed in this paper.*

between $\frac{1}{50}$ of a second and $\frac{1}{3}$ of a second of CPU time, depending on the core size and number of particles required. The amount of time required scaled linearly with the number of blobs, as one would expect. While there is some advantage to canceling

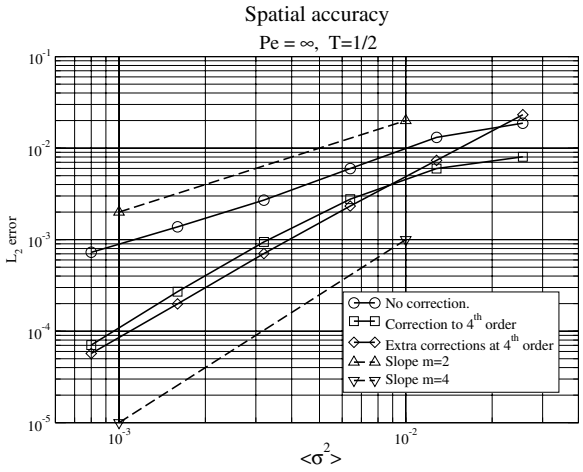

FIG. 4. *Slight advantages in partial error cancellation for short times. We see that canceling two of the fourth-order terms offers some improvement. However, the advantages are slight and disappear for larger T.*

the two fourth-order terms described in section 5 for short times (see Figure 4), this advantage disappears quickly, as seen in Figure 3. This is a generic property with many different concentric flow fields. There are two obvious explanations. The first is that some cancellation is occurring between the fourth-order terms, and removing two might remove beneficial cancellations. The second is that the dynamics are different for the two fourth-order methods because the corrected velocity fields are different. For instance, where there is no diffusion, the scalar-weighted average aspect ratio over all of the particles is more than 6.4 at $T = 4$ when the two fourth-order terms are eliminated, while it is slightly more than 5.5 when the fourth-order terms are left in place. Since the size of the error terms depends upon quantities like $a^2 - a^{-2}$, these effects can muddy the waters slightly. As noted in section 6, diffusion damps growth in the aspect ratio so the effects are less noticeable.

When applying corrections to the velocity field, the centroid velocity data can be very different from velocity data that is adjusted to achieve fourth-order accuracy. Both are incompressible flow fields and, therefore, preserve volumes. However, particles need not follow the physical flow field, as can be seen in Figure 5. Similar behavior is observed at all Peclét numbers and initial core sizes, although less so for smaller core sizes.

While it can be difficult to quantify the core size of heterogeneous, nonisotropic elements, we propose using

$$\langle \sigma^2 \rangle = \frac{\sum_{n=1}^{N} \gamma_i \sigma_i^2}{\sum_{n=1}^{N} \gamma_i} \tag{7.4}$$

as a reasonable assessment. In Figure 3, we see that the fourth-order scheme proposed in this paper does, in fact, achieve fourth-order convergence and represents a significant improvement over second-order computations with centroid velocity data. For some quantitative perspective on the size of the errors, one should consider that the maximum height of the initial data (7.2) is 4 and that the $\iint \rho d\vec{x} = \pi$. In Figure 6, one can see the differences with contrasted contour plots. Similar behavior is observed

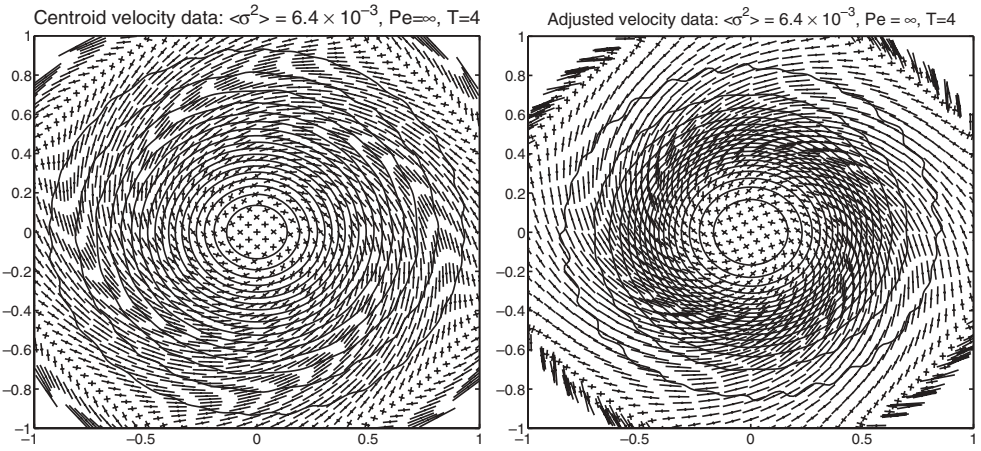

FIG. 5. *Typical basis function positions at* $T = 4$. *When using centroid data, basis functions follow the streamline. However, the higher spatial order method need not do so. Similar behavior is seen for all experiments, although to a lesser extent when* $\langle\sigma^2\rangle$ *is smaller. Blobs are shown as* $\times$*'s, where the arm lengths are* $\frac{1}{5}\sigma_i a_i$ *and* $\frac{1}{5}\frac{\sigma_i}{a_i}$, *and we see that aspect ratios vary considerably over the domain due to variations in the local shear. Contour lines of* $\widehat{\rho}$ *are superposed at intervals of* 0.4, *and the* L_2 *error for this choice of core size is about the same for both methods. On the left, the simulation uses centroid velocity data and the particle densities remain uniform. On the right, the simulation uses adjusted velocity data to achieve fourth-order accuracy. In this case, particles no longer follow the concentric streamlines, and density variations occur. In both plots, skinnier basis functions can have aspect ratios of* 15 *or more.*

for all Peclét numbers that were computed.

8. Conclusions, future work, and related projects. In this paper, the following ideas have been presented and verified:

- Deforming elliptical Gaussian basis functions can be used to achieve fourth-order spatial accuracy for the convection-diffusion equations.
- There is a systematic procedure for determining the spatial accuracy of methods using deforming basis functions based on the residual of the computed field, and this residual can be used to determine useful velocity field corrections.
- We have presented the evolution equations for a tumbling, diffusing elliptical basis function in a linear flow field.
- The predicted spatial accuracy for the method using deforming elliptical Gaussian blobs is realized in nontrivial flows with and without diffusion.
- There may be some advantages to canceling some of the fourth-order terms, but generally it is better to leave them in place.

Still, more remains to be done, particularly if one wishes to use methods like these for nonlinear problems for extended simulation times.

Remeshing and conditioning issues. There are several reasons why one would be interested in remeshing. First, we have seen that the accuracy of the method depends upon the aspect ratio of the elements in addition to the core width. Second, if the Peclét number is infinite, we will require a means of replacing a configuration of deformed elements with a configuration of axisymmetric elements from time to time because aspect ratios can grow without bound. If the Peclét number is finite, the core width will grow with time and create a situation where there is a lower bound on the

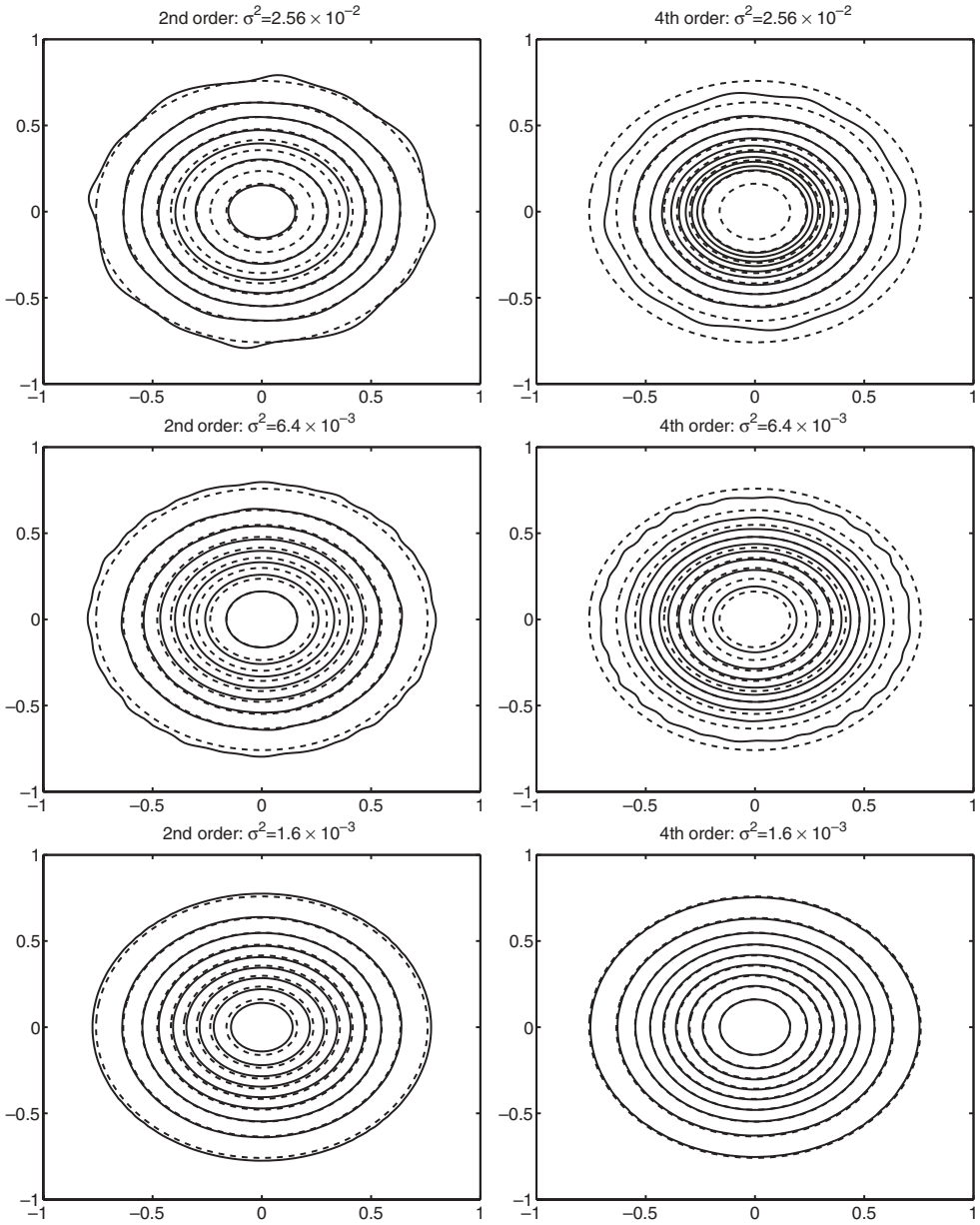

FIG. 6. *Comparison of scalar fields with second- and fourth-order schemes at* $\mathrm{Pe}=\infty$. *The exact solution (dashed) is shown together with the computed (solid) contours. The second-order results are in the left-hand column, and the fourth-order results are in the right-hand column. Each row represents different core sizes, diminishing from top to bottom. The contour increments are* 0.4 *scalar units.*

spatial error, rendering it inconsistent for any initial core width.

Since this method has a high spatial accuracy, a priority should be placed on using high-accuracy remeshing as well, or the benefit of the Lagrangian method will be lost to effort devoted to remeshing with large numbers of elements. There are two main methods for treating this problem. The first requires global remeshing where

all elements are replaced with a new regular configuration of elements. A survey of such methods can be found in [4]. Perhaps the most suitable technique is a least-squares remeshing method for elliptical Gaussians in two dimensions developed by Moeleker and Leonard [20]. The second is a local refinement method where individual elements that exceed a particular tolerance are replaced with a configuration of elements that approximate the original single element. The growth in problem size is mitigated by a local merging algorithm for nearly overlapping elements. These ideas were first proposed as a correction to axisymmetric core spreading by Rossi [25, 26] and have been explored by Shiels [29] as well. Recently, the author has determined a class of refinement methods that achieve successively higher order spatial accuracy and hopes to find practical methods to employ them. The local merging ideas originally used for axisymmetric core spreading have been substantially improved using weak estimates to achieve good problem size reduction for any user-specified tolerance. This merging method is unpublished at present but is fully implemented in the current version of BlobFlow, an open source vortex code with deforming elements. This implementation, along with a variety of related tools, is available at no cost at http://www.math.udel.edu/∼rossi/BlobFlow. The fourth-order velocity field corrections have not been added to the current version at this time.

Direct Biot–Savart interactions for elliptical Gaussian elements for vortex computations. To use these ideas for vortex computations, one must be able to compute the velocity field (and its derivatives) induced by an elliptical Gaussian basis function. While there is no known representation for the Biot–Savart integral of a anisotropic elliptical Gaussian in terms of elementary functions, there is a very accurate fourth-order, asymptotic approximation in ϵ, where

$$\epsilon = \frac{a-1}{a+1}. \tag{8.1}$$

This approximation for the velocity field and its derivatives compares favorably with exhaustive numerical integrations even for large aspect ratios [27].

Fast multipole summation for elliptical Gaussian elements for vortex computations. To effectively implement vortex methods, one must reduce the element-element velocity interactions that require a time complexity of $O(N^2)$. While the original Greengard–Rohklin [8] algorithm is inapplicable to nonsingular kernels like the anisotropic elements discussed in this paper, their ideas can be adapted for anisotropic elements to achieve $O(N)$ complexity. Again, this is implemented in the latest version of BlobFlow, but the velocity corrections have not be incorporated at the time of this writing.

To summarize, Lagrangian methods using deforming elliptical Gaussian basis functions can achieve fourth-order spatial accuracy for the full convection-diffusion equations. More work lies ahead in fundamental areas such as refinement and remeshing, and areas related to nonlinear studies such as velocity computations.

Acknowledgments. The author would like to acknowledge many useful conversations with Tony Leonard of Caltech, and the keen eyes and sharp minds of the three anonymous referees.

REFERENCES

[1] J. T. Beale and A. Majda, *Rates of convergence for viscous splitting of the Navier-Stokes equations*, Math. Comp., 37 (1981), pp. 243–259.

[2] J. T. Beale and A. Majda, *Vortex methods* II: *Higher order accuracy in two and three dimensions*, Math. Comp., 39 (1982), pp. 29–52.
[3] J. T. Beale and A. Majda, *High order accurate vortex methods with explicit kernels*, J. Comput. Phys., 58 (1985), pp. 188–205.
[4] G.-H. Cottet and P. D. Koumoutsakos, *Vortex Methods: Theory and Practice*, Cambridge University Press, Cambridge, UK, 2000.
[5] A. Gharakhani, *A Higher Order Vorticity Redistribution Method for* 3-D *Diffusion in Free Space*, Technical report SAND2000-2505, Sandia National Laboratories, Albuquerque, NM, 2000.
[6] A. Gharakhani, *Grid-Free Simulation of* 3-D *Vorticity Diffusion by a High-Order Vorticity Redistribution Method*, Technical report AIAA-2001-2640, American Institute of Aeronautics and Astronautics, 2001.
[7] C. Greengard, *The core spreading vortex method approximates the wrong equation*, J. Comput. Phys., 61 (1985), pp. 345–348.
[8] L. Greengard and V. Rokhlin, *A fast algorithm for particle simulations*, J. Comput. Phys., 73 (1987), pp. 325–348.
[9] T. Kida, *Theoretical and numerical results of a deterministic two-dimensional vortex method*, Sādhanā, 23 (1998), pp. 419–441.
[10] T. Kida and T. Nakajima, *Core spreading vortex methods in two-dimensional viscous flows*, Comput. Methods Appl. Mech. Engrg., 160 (1998), pp. 273–298.
[11] T. Kida, T. Nakajima, and H. Suemitsu, *Second order core spreading vortex method in two-dimensional viscous flows*, JSME Internat. J. Fluids & Therm. Engrg. B, 41 (1998), pp. 441–446.
[12] A. Leonard, *Vortex methods for flow simulation*, J. Comput. Phys., 37 (1980), pp. 289–335.
[13] A. Leonard, *AIAA* 97-0204: *Large-eddy simulation of chaotic convection and beyond*, in Proceedings of the 35th Aerospace Sciences Meeting and Exhibit, American Institute of Aeronautics and Astronautics, 1997, pp. 1–8.
[14] P. Linz, *Theoretical Numerical Analysis*, John Wiley, New York, 1979.
[15] J. S. Lowengrub and M. J. Shelley, *Smooth grid methods for the vorticity formulation of the Euler equations*, in Vortex Dynamics and Vortex Methods, Lectures Appl. Math. 28, AMS, Providence, RI, 1991, pp. 423–432.
[16] J. S. Lowengrub, M. J. Shelley, and B. Merriman, *High-order and efficient methods for the vorticity formulation of the Euler equations*, SIAM J. Sci. Comput., 14 (1993), pp. 1107–1142.
[17] Z. Y. Lu and T. J. Ross, *Diffusing-vortex numerical scheme for solving incompressible Navier-Stokes equations*, J. Comput. Phys., 95 (1991), pp. 400–435.
[18] J. S. Marshall and J. R. Grant, *A method for determining the velocity induced by highly anisotropic vorticity blobs*, J. Comput. Phys., 126 (1996), pp. 286–298.
[19] E. Meiburg, *Incorporation and test of diffusion and strain effects in the two-dimensional vortex blob technique*, J. Comput. Phys., 82 (1989), pp. 85–93.
[20] P. Moeleker and A. Leonard, *Lagrangian methods for the tensor-diffusivity subgrid model*, J. Comput. Phys., 167 (2001), pp. 1–21.
[21] J. J. Monaghan, *Why particle methods work*, SIAM J. Sci. Statist. Comput., 3 (1982), pp. 422–433.
[22] J. J. Monaghan, *Smoothed particle hydrodynamics*, Ann. Rev. Astron. Astrophys., 30 (1992), pp. 543–574.
[23] A. Ojima and K. Kamemoto, *Numerical simulation of unsteady flow around three dimensional bluff bodies by an advanced vortex method*, JSME Int. J. Fluids & Therm. Mixing B, 43 (2000), pp. 127–135.
[24] T. A. Prickett, T. G. Naymik, and C. G. Lonnquist, *A "random walk" solute transport model for selected groundwater quality evaluations*, Illinois Water Survey Bull., 65 (1981), p. 103.
[25] L. F. Rossi, *Resurrecting core spreading vortex methods: A new scheme that is both deterministic and convergent*, SIAM J. Sci. Comput., 17 (1996), pp. 370–397.
[26] L. F. Rossi, *Merging computational elements in vortex simulations*, SIAM J. Sci. Comput., 18 (1997), pp. 1014–1027.
[27] L. F. Rossi, *High Order Vortex Methods with Deforming Elliptical Gaussian Blobs* 1: *Derivation and Validation*, Technical Report 2001-11, University of Delaware, Newark, DE, 2002.
[28] L. F. Rossi, J. F. Lingevitch, and A. J. Bernoff, *Quasi-steady monopole and tripole attractors for relaxing vortices*, Phys. Fluids, 9 (1997), pp. 2329–2339.
[29] D. Shiels, *Simulation of Controlled Bluff Body Flow with a Viscous Vortex Method*, Ph.D. thesis, California Institute of Technology, Pasadena, CA, 1998.

[30] J. C. Strikwerda, *Finite Difference Schemes and Partial Difference Equations*, Wadsworth and Brooks/Cole Advanced Books and Software, New York, 1989.
[31] Z.-H. Teng, *Elliptic-vortex method for incompressible flow at high Reynolds number*, J. Comput. Phys., 46 (1982), pp. 54–68.
[32] Z.-H. Teng, *Variable-elliptical-vortex method for incompressible flow simulation*, J. Comput. Math., 4 (1986), pp. 255–262.
[33] Z.-H. Teng, L.-A. Ying, and P. Zhang, *Convergence of the variable-elliptic-vortex method for Euler equations*, SIAM J. Numer. Anal., 32 (1995), pp. 754–774.
[34] X.-H. Wen and C.-S. Kung, *Implementation of the constant displacement scheme in random walk*, Computers & Geosciences, 22 (1996), pp. 369–377.
[35] G. S. Winckelmans and A. Leonard, *Contributions to vortex particle methods for the computation of three-dimensional incompressible unsteady flows*, J. Comput. Phys., 109 (1993), pp. 247–273.