Journal of Computational Physics **167,** 1–21 (2001)
doi:10.1006/jcph.2000.6669, available online at http://www.idealibrary.com on IDEAL®

# Lagrangian Methods for the Tensor-Diffusivity Subgrid Model

Piet Moeleker and Anthony Leonard

*Graduate Aeronautical Laboratories, California Institute of Technology Pasadena, California 91125*
E-mail: piet@galcit.caltech.edu

Received February 16, 2000; revised October 20, 2000

A subgrid-scale model based on a truncated exact series expansion for Gaussian filtered products is considered for the incompressible scalar advection–diffusion equation. This model can be interpreted as a tensor diffusivity term proportional to the rate-of-strain tensor of the large-scale filtered velocity field. To control negative diffusion in the stretching directions, a Lagrangian method is used. The scalar field is represented in terms of a collection of anisotropic or axisymmetric Gaussian particles. An expansion in Hermite polynomials leads to equations of motion for particle velocity and shape based on a weighted average. A new accurate remeshing method, taking advantage of the properties of the subgrid model, is proposed and tested. A stagnation flow is used to demonstrate several theoretical and numerical aspects of the model. Better agreement with filtered DNS data is obtained than with the Smagorinsky subgrid model for a 2D time-dependent sinusoidal flow, which yields chaotic advection. The use of anisotropic particles leads to slightly more accurate results than the use of axisymmetric particles. Computational efficiency, however, makes the latter therefore the preferred choice. 

*Key Words:* computational fluid dynamics; scalar transport; large eddy simulation; Lagrangian particle method; tensor-diffusivity subgrid model.

## 1. INTRODUCTION

The scalar advection–diffusion equation describes the motion of a scalar quantity under the advection of a velocity field and diffusion. For small diffusivity values, a large computational effort is required to account for both the large- and small-scale structures. To reduce this effort a filtering operation may be used to obtain an equation for the large-scale structures in which the effect of the small scales has to be modeled. These simulations are known as large eddy simulations (LES) and the models as subgrid models.

The tensor-diffusivity subgrid model will be the focus of this article. This model was derived independently by Bedford and Yeo [1, 2] and Leonard [3] after using a Gaussian



0021-9991/01 $35.00

filter and expressing the filtered products in an infinite sum of known filtered quantities. Retaining the first two terms in this expansion gives the subgrid model. The method is ill-conditioned so that some form of regularization is required.

A particle method that provides a suitable regularization is used to numerically solve the model equation. So-called anisotropic Gaussian particles will be used resulting in equations that have higher order accuracy compared to the widely used axisymmetric Gaussian particles. The anisotropic particles will have nine degrees of freedom, three for the location in space and the six remaining ones for the size and orientation. To assess the benefits of a higher order scheme, axisymmetric particles have been used as well.

To numerically account for the molecular diffusion of a scalar or the diffusion of vorticity due to viscous effects, a variety of methods have been used by different researchers. For example, in the random-walk method a random displacement is added to the motion of each particle [4]. The core spreading method increases the size of the Gaussian particles over time to simulate the effect of diffusion [5]; one particular advantage is that it solves the diffusive part of the equation exactly. However, Greengard [6] showed that the core spreading method approximates the wrong equation in the limit of an infinite number of particles.

Recently, Rossi [7, 8] revamped the use of the core spreading method for the viscous vorticity equation by introducing a splitting and merging scheme for the axisymmetric Gaussian particles. He showed that by splitting the particles and thus controlling the core size of the particles, convergence of the vorticity equation is obtained in the limit of an infinite number of particles and splitting the particles continuously. The merging scheme he employed was only used to keep the total number of particles reasonable. In the present work the core expansion method has been used as well and a remeshing scheme has been implemented to keep the core size of the particles within certain bounds.

This article is composed as follows. Section 2 introduces the tensor-diffusivity model. We elaborate on the particle method in Section 3.1 and on the remeshing procedure in Section 3.2. Section 4 deals with several test cases to illustrate the method and demonstrate several ideas. Three appendices are attached to give several mathematical derivations in more detail.

## 2. TENSOR-DIFFUSIVITY SUBGRID MODEL

The transport of a passive scalar quantity $\psi(\boldsymbol{x}, t)$ in an incompressible velocity field $u(\boldsymbol{x}, t)$ is governed by

$$\frac{\partial \psi}{\partial t} + \boldsymbol{u} \cdot \boldsymbol{\nabla} \psi = \kappa \nabla^2 \psi, \tag{1}$$

where $\kappa$ is the diffusivity coefficient. Consider a spatial Gaussian filter with characteristic length scale $\sigma$ and spatial dimension $d$, given by

$$F(\boldsymbol{x}) = \frac{1}{(\sigma\sqrt{\pi})^d} \exp\left(-\frac{|\boldsymbol{x}|^2}{\sigma^2}\right). \tag{2}$$

Convolving this filter with (1) gives the filtered advection–diffusion equation

$$\frac{\partial \hat{\psi}}{\partial t} + \boldsymbol{\nabla} \cdot \widehat{\boldsymbol{u}\psi} = \kappa \nabla^2 \hat{\psi}, \tag{3}$$

where convolved or filtered quantities are indicated by a hat. Under the assumption that both $\psi(\boldsymbol{x}, t)$ and $\boldsymbol{u}(\boldsymbol{x}, t)$ are continuous and differentiable, the unknown filtered function $\widehat{\boldsymbol{u}\psi}$ can be expressed in known quantities as

$$\widehat{\boldsymbol{u}\psi}(\boldsymbol{x}, t) = \sum_{n=0}^{\infty} \frac{1}{n!} \left(\frac{\sigma^2}{2}\right)^n \frac{\partial^n \hat{\boldsymbol{u}}}{\partial x_{i_1} \partial x_{i_2} \cdots \partial x_{i_n}} \frac{\partial^n \hat{\psi}}{\partial x_{i_1} \partial x_{i_2} \cdots \partial x_{i_n}}, \tag{4}$$

where Einstein's summation convention is used. This result was independently established by Bedford and Yeo [1, 2] and Leonard [3]. Leonard's derivation uses an expansion in Hermite polynomials and makes use of their properties. See Appendix A for more details. Upon substitution of (4) in (3) and retaining only the first two terms in the infinite expansion, we can approximate (3) as

$$\frac{\partial \hat{\psi}}{\partial t} + \hat{\boldsymbol{u}} \cdot \boldsymbol{\nabla} \hat{\psi} = \kappa \nabla^2 \hat{\psi} - \frac{\sigma^2}{2} \hat{S}_{ij} \frac{\partial^2 \hat{\psi}}{\partial x_i \partial x_j}, \tag{5}$$

where the strain rate tensor $S_{ij} = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i})$ has been introduced. The extra term can be interpreted as an added diffusivity with an effective diffusivity $-\frac{\sigma^2}{2}\hat{S}_{ij}$, which depends on the spatial direction (hence, the name tensor-diffusivity subgrid model). Since trace $(\hat{\boldsymbol{S}}) = \boldsymbol{\nabla} \cdot \hat{\boldsymbol{u}} = 0$, at least one of the eigenvalues of the strain rate tensor has to be greater than zero, indicative of a direction where the subgrid model acts as negative diffusion. Say $\lambda_1$ is the largest eigenvalue of $\hat{S}$ in the direction $x_1$; then the total effective diffusivity in this direction is given by $-\frac{\sigma^2}{2}\lambda_1 + \kappa$, which results in negative diffusion for $\lambda_1 > \frac{2\kappa}{\sigma^2}$.

It was shown by Carati *et al*. [9] that the use of spatial filters other than Gaussian results in doubly infinite expansions. However, the leading two terms in such an expansion are the same, except for a multiplicative constant, for a large class of different filters including the top-hat filter and all discrete filters. This implies that all these filters result in the tensor-diffusivity subgrid model given above.

It is desirable that a subgrid model does not depend on the frame of reference an observer chooses, as nature is unaware of our choices. In mathematical terms, this is reflected by proper transformation properties between different frames of reference [10]. An equation is called frame indifferent if it has these properties. Examples are the Navier–Stokes equation and the scalar advection–diffusion equation. It is desirable for subgrid-scale models to satisfy these transformations as well. It was shown by Fureby [11] that only filtering operators with rotational symmetry, among these the Gaussian filter, will preserve material frame indifference. Consider the transformation between a starred and unstarred coordinate system given by a time-dependent rotation $\boldsymbol{Q}$ (with $\boldsymbol{Q}^T = \boldsymbol{Q}^{-1}$) and relative velocity $c$,

$$x_i^* = Q_{ij}(t) x_j + c_i(t). \tag{6}$$

After applying this transformation to the tensor-diffusivity subgrid model, one can show that the models in the starred and unstarred system are related by

$$-\frac{\sigma^2}{2} \hat{S}_{ij}^* \frac{\partial^2 \hat{\psi}^*}{\partial x_i^* \partial x_j^*} = -\frac{\sigma^2}{2} \hat{S}_{ij} \frac{\partial^2 \hat{\psi}}{\partial x_i \partial x_j}, \tag{7}$$

which establishes material frame indifference.

To give some insight into the effect of the extra term in (5), consider the following example. Assume there is (approximately) a uniform velocity field in the principal $x_1$ direction given by $u_1 = \lambda_1 x_1$ and set the diffusivity $\kappa$ equal to zero. A simple wave in the $x_1$ direction is used as an initial condition, $\hat{\psi}(\boldsymbol{x}, 0) = e^{ikx_1}$, where $k$ is the wave number. The filtered advection–diffusion equation (5) simplifies to

$$\frac{\partial \hat{\psi}}{\partial t} + \lambda_1 x_1 \frac{\partial \hat{\psi}}{\partial x_1} = -\frac{\sigma^2}{2}\lambda_1 \frac{\partial^2 \hat{\psi}}{\partial x_1^2}. \tag{8}$$

It is straightforward to verify that $\hat{\psi} = e^{ik\exp(-\lambda_1 t)x_1}\exp[\sigma^2 k^2(1-\exp(-2\lambda_1 t))/4]$ is a solution. The wave number of this solution is $k\exp(-\lambda_1 t)$, which decreases in time due to stretching. The amplitude of the wave is given by $\exp[\sigma^2 k^2(1-\exp(-2\lambda_1 t))/4]$ and increases in time. The initial rate of increase becomes arbitrarily large as $k \to \infty$. The addition of molecular diffusion avoids this undesirable behavior only if $\kappa > \frac{\sigma^2}{2}\lambda_1$.

Additional understanding comes from the evolution equation for the root mean square of the scalar field $\tilde{\psi}(t)$,

$$\frac{d\tilde{\psi}^2}{dt} = -2\kappa \int_\Omega (\boldsymbol{\nabla}\hat{\psi})^2 \, d\boldsymbol{x} + \sigma^2 \int_\Omega \hat{S}_{ij} \frac{\partial \hat{\psi}}{\partial x_i}\frac{\partial \hat{\psi}}{\partial x_j} \, d\boldsymbol{x}. \tag{9}$$

By using the principal coordinate system of $\boldsymbol{S}$, one can ascertain that the second term on the right-hand side can be both positive and negative, implying that the subgrid model allows for backscatter.

On occasion mathematical models of physical processes lead to ill-posedness; see for example Barenblatt *et al*. [12] or Krasny [13]. Some form of regularization is required to obtain a well-posed problem. Care needs to be taken in the choice of regularization, since the results can depend strongly on the approach used. By filtering the advection–diffusion equation (1), information about the high wave number components is lost. In order to have the solution with the regularization approximating the solution of (5) closely, we need a regularization that maintains control over the entire wave number spectrum. Computations solving (5) directly using a spectral or finite difference method show that growing instabilities are introduced. For the finite difference approach, an example will be given in Section 4.2. Good results to regularize a finite difference method using the tensor-diffusivity model for the momentum equation have been obtained by Leonard and Winckelmans [14] by adding an extra dynamic eddy viscosity term. Our work will regularize the problem by decomposing the scalar field into a collection of Lagrangian particles, each of which are well behaved for large wave numbers.

## 3. NUMERICAL METHOD

### 3.1. *Particle Method*

As discussed in the previous section we use a particle method to numerically solve (5) by approximating the scalar field $\psi(\boldsymbol{x}, t)$ (dropping the hats) by a sum of $N$ anisotropic Gaussian particles

$$\psi(\boldsymbol{x}, t) = \sum_{k=1}^{N} \frac{a_k\sqrt{\det(\boldsymbol{M}_k)}}{(\sqrt{\pi}\delta_k)^d} \exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{x}_k)^T \boldsymbol{M}_k (\boldsymbol{x}-\boldsymbol{x}_k)}{\delta_k^2}\right), \tag{10}$$

where each particle $k$ is centered at $\boldsymbol{x}_k$ and $d$ is the spatial dimension of the problem. The core size of each particle is given by $\delta_k$ and the amplitude by $a_k$. The positive definite matrix $\boldsymbol{M}_k$ is called the shape function. Only the location $\boldsymbol{x}_k$ and the shape function $\boldsymbol{M}_k$ are assumed to be functions of time. These particles were recently used by Rossi [15] in a viscous vortex method. The Fourier transform of (10) is given by

$$\mathcal{F}\{\psi\}(\boldsymbol{k},t)=\sum_{k=1}^{N}a_k\exp\left(-\frac{\delta_k^2\boldsymbol{k}^T\boldsymbol{M}_k^{-1}\boldsymbol{k}}{4}\right)e^{i\boldsymbol{k}\cdot\boldsymbol{x}_k}, \tag{11}$$

where $\boldsymbol{k}$ is the wave number. This function appears to be well behaved for large wave numbers as long as $\boldsymbol{M}_k$ remains positive definite, giving us the desired control over all wave numbers.

Equations for the time evolution of the shape function $\boldsymbol{M}_k$ and the location $\boldsymbol{x}_k$ are found by substituting (10) in (5) and expanding each term in a series of Hermite polynomials. An expansion in Hermite polynomials is used instead of the more commonly used Taylor polynomials, as the former are in a more natural way connected to Gaussians. It is expected that the contribution of the lower order terms in the expansion is most significant. By setting the coefficients of the lower order terms in the expansion equal to zero, we obtain the desired equations of motion. A detailed derivation of this procedure can be found in Appendix B. The end result for the location of the particles is

$$\frac{d\boldsymbol{x}_k}{dt}=\overline{\boldsymbol{u}}^k-\frac{\sigma^2}{2}\overline{\nabla^2\boldsymbol{u}}^k, \tag{12}$$

where an overline over an arbitrary function $f(\boldsymbol{x},t)$ is defined by

$$\overline{f}^k(\boldsymbol{x}_k,t)=\frac{\sqrt{\det(\boldsymbol{M}_k)}}{(\sqrt{\pi}\delta_k)^d}\int_\Omega f(\boldsymbol{x},t)\exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{x}_k)^T\boldsymbol{M}_k(\boldsymbol{x}-\boldsymbol{x}_k)}{\delta_k^2}\right)d\boldsymbol{x}. \tag{13}$$

The overline can be interpreted as a weighted average over the anisotropic Gaussian particle $k$. Since this average depends on the particle $k$, two particles that occupy the same location $\boldsymbol{x}_k$, but different shapes $\boldsymbol{M}_k$, can move in different spatial directions. For the time evolution of the matrix $\boldsymbol{M}_k$, we find

$$\begin{aligned}\frac{d\boldsymbol{M}_k}{dt}=&-\overline{\boldsymbol{\nabla}\boldsymbol{u}}^k\boldsymbol{M}_k-\boldsymbol{M}_k\overline{\boldsymbol{\nabla}\boldsymbol{u}}^{kT}-\frac{4\kappa}{\delta_k^2}\boldsymbol{M}_k\boldsymbol{M}_k+\frac{\sigma^2}{\delta_k^2}\boldsymbol{M}_k(\overline{\boldsymbol{\nabla}\boldsymbol{u}}^k+\overline{\boldsymbol{\nabla}\boldsymbol{u}}^{kT})\boldsymbol{M}_k\\&+\frac{\sigma^2}{2}\left(\overline{\boldsymbol{\nabla}\nabla^2\boldsymbol{u}}^k\boldsymbol{M}_k+\boldsymbol{M}_k\overline{\boldsymbol{\nabla}\nabla^2\boldsymbol{u}}^{kT}\right).\end{aligned} \tag{14}$$

If we consider the next order coefficients in the expansion, we see that these are $O(\delta_k^3)$ and $O(\sigma^2\delta_k)$. Combined with the truncation error of the subgrid model (4), $O(\sigma^4)$, the total error is $O(\delta_k^3,\sigma^2\delta_k,\sigma^4)$.

To assess the benefits of using a higher order particle method and for comparison reasons, we have also considered the widely used axisymmetric Gaussian particles, which are obtained by setting $\boldsymbol{M}_k$ equal to the identity matrix in (10). The core size $\delta_k$ is now assumed to be a function of time, as is the location $\boldsymbol{x}_k$. The governing equations can be derived in a similar way using an expansion in Hermite polynomials or the equations for anisotropic

Gaussian particles can be simplified using the constraint that the particles remain axisymmetric for all time. Both derivations yield the same result. For the time evolution of the location of the particles, we obtain (12) again and for the core size $\delta_k$,

$$\frac{d\delta_k^2}{dt} = 4\kappa + O\left(\delta_k^2, \sigma^2\right). \tag{15}$$

Note that we can use (15) only for small values of $\sigma$ and $\delta_k$. Inaccuracies are introduced when $\delta_k$ becomes larger, a problem of the core expansion method as shown by Greengard [6].

### 3.2. *Remeshing*

There are three main reasons which necessitate the remeshing of the scalar field or splitting and merging of the particles every so often. First, we need to circumvent the inherent problem of the core expansion method by keeping the effective core size of the particles within limits, where the effective core size is defined as $\frac{\delta_k}{\lambda_{k,1}}$, where $\lambda_{k,1}$ is the largest eigenvalue of $\boldsymbol{M}_k$. Second, to get a smooth scalar field, the overlap parameter, defined as the ratio of the effective core size $\frac{\delta_k}{\lambda_{k,1}}$ over the corresponding interparticle distance $d$, has to be of order unity. Owing to straining, particles tend to move apart from each other in some directions, thereby decreasing the overlap parameter and making a remesh necessary to maintain a smooth scalar field. Finally, anisotropic particles can become very elliptical also as the result of straining, which increases the numerical errors.

Assume that at time $t$, we want to replace all the $N$ old particles with a set of $M$ new identical axisymmetric particles with initial core size $\tau$. The new set of particles is spread out on a regular rectangular mesh with grid spacing $h$ between neighboring particles in each direction. The location of the $l$-th new particle is denoted by $\xi_l$. The amplitudes $b_l$'s of the new particles have to be chosen appropriately to minimize the error between the old and the new scalar fields. Using an approximation based on a least-square-error method, we were able to find the following explicit expressions for the unknown coefficients $b_l$'s in two dimensions:

$$\begin{aligned} b_l = \sum_{k=1}^{N} & \frac{a_k h^2 \sqrt{\det(\boldsymbol{M}_k)}}{\pi \sqrt{\left(\delta_k^2 - \lambda_{k,1}\tau^2\right)\left(\delta_k^2 - \lambda_{k,2}\tau^2\right)}} \\ & \times \exp\left( - \frac{(\boldsymbol{\xi}_l - \boldsymbol{x}_k)^T \left(\delta_k^2 \boldsymbol{M}_k + \tau^2 \det(\boldsymbol{M}_k) I\right)(\boldsymbol{\xi}_l - \boldsymbol{x}_k)}{\left(\delta_k^2 - \lambda_{k,1}\tau^2\right)\left(\delta_k^2 - \lambda_{k,2}\tau^2\right)} \right). \end{aligned} \tag{16}$$

Here the eigenvalues of the shape matrix $\boldsymbol{M}_k$ are $\lambda_{k,1}$ and $\lambda_{k,2}$. A derivation of this equation (in one dimension) can be found in Appendix C. To get sensible results we need to satisfy $\tau^2 < \frac{\delta_k^2}{\lambda_k}$ for all $k$ and both eigenvalues $\lambda_k$. This puts an upper bound on the new core size of the particles.

In all the computations presented in this paper, we set the new core size $\tau$ equal to the initial core size ($\delta_k$) at $t = 0$ to have the same core size of the particles initially and after each remesh procedure. It then follows that all eigenvalues of the shape matrix $\boldsymbol{M}_k$ for all particles $k$ have to be less than unity ($\lambda_k < 1$) for this remeshing procedure to be applicable. It can be shown theoretically that in the absence of the last term of (14), this requirement is always met. In the presence of this term, numerical experiments show that if the influence of the subgrid model is small ($\sigma$ small) compared to the advection and diffusion terms,

this requirement is met as well. The derivation of (16) in case of axisymmetric particles is straightforward.

## 4. TEST CASES

### 4.1. *Stagnation Flow*

We will start with a simple problem for illustrative purposes. Consider a two-dimensional stagnation flow given by the incompressible velocity components $u = cx$ and $v = -cy$, where $c$ is an arbitrary constant. Assume no molecular diffusivity ($\kappa = 0$) and consider only one anisotropic particle $k$ initially located at $\boldsymbol{x}_k$ with initial shape matrix $\boldsymbol{M}_k = \boldsymbol{I}_2$.

For this specific velocity field, the tensor-diffusivity subgrid model is exact and thus the filtered advection–diffusion equation (5) is also. The (numerical) particle method using anisotropic Gaussian particles is an exact solution to (5) as well and will be used to test some basic features of the model and the numerical implementation.

Let us first solve the equation of motion for the anisotropic particles analytically. The equation of motion for the location of the particle (12) reduces to

$$\frac{d\boldsymbol{x}_k}{dt} = \vec{\boldsymbol{u}}^k = \boldsymbol{u}(\boldsymbol{x}_k) = \begin{pmatrix} cx_k \\ -cy_k \end{pmatrix}. \tag{17}$$

Equation (14) for the shape matrix $\boldsymbol{M}_k$ with elements $m_{ij}$ can be reduced to

$$\frac{dm_{11}}{dt} = -2cm_{11} + \frac{2\sigma^2}{\delta_k^2} c\left(m_{11}^2 - m_{12}^2\right), \tag{18a}$$

$$\frac{dm_{22}}{dt} = 2cm_{22} + \frac{2\sigma^2}{\delta_k^2} c\left(m_{12}^2 - m_{22}^2\right), \tag{18b}$$

$$\frac{dm_{12}}{dt} = \frac{2\sigma^2}{\delta_k^2} cm_{12}(m_{11} - m_{22}), \tag{18c}$$

with the solution (using the initial condition $\boldsymbol{M}_k = \boldsymbol{I}_2$)

$$m_{11}(t) = \frac{\delta_k^2/\sigma^2}{1 + \left(\delta_k^2/\sigma^2 - 1\right)e^{-2ct}}, \tag{19a}$$

$$m_{22}(t) = \frac{\delta_k^2/\sigma^2}{1 + \left(\delta_k^2/\sigma^2 - 1\right)e^{2ct}}, \tag{19b}$$

$$m_{12}(t) = 0, \tag{19c}$$

which corresponds to the exact solution of the problem. For $\delta_k < \sigma$, the solution blows up at $t = -\frac{1}{2c}\ln(1 - \delta_k^2/\sigma^2)$, whereas for $\delta_k > \sigma$, the solution will stay finite for all time. In the unfiltered scalar field, the most singular structures are delta functions. Applying a Gaussian filter with width $\sigma$ transforms these delta functions to Gaussians with width $\sigma$. The blowup of the solution for $\delta_k < \sigma$ is therefore not relevant to our applications, since such Gaussians do not correspond to meaningful basis elements for the unfiltered field.

Four different test runs have been performed for the stagnation flow, numbered 1 through 4, all using $c = 1$ and $\sigma = 0.2$. Each of these runs uses the fifth-order Cash–Karp Runge–Kutta method with adaptive step size for error control as discussed in Section 16.2 of Press

**TABLE I**
**Test Data for the Stagnation Flow**

| Run | Type of particles | $N$ | $\delta_k$ | $\delta_k/d$ |
|---|---|---|---|---|
| 1 | Anisotropic | 1 | 0.50 | n/a |
| 2 | Anisotropic | 148 | 0.25 | 1.25 |
| 3 | Axisymmetric | 148 | 0.25 | 1.25 |
| 4 | Anisotropic | 148 | 0.15 | 0.75 |

*et al.* [16] to march forward in time. The initial condition is a Gaussian with core size $\delta_k = 0.5$ and amplitude $a = 1$ centered at $(x, y) = (0, 2)$.

Table I gives information about each of the different runs. The first column gives the run number and the second one the type of Gaussian particles that were used. Next, the total number of particles ($N$) is listed and the core size of the particles at $t = 0$. The last column gives the overlap parameter at the start of the simulation. The first run represents the exact initial condition with just one particle, whereas the other three runs use 148 particles on a regular two-dimensional grid with gridspacing $h = 0.2$ to approximate the initial condition of run 1. For runs 2 and 3, the overlap parameter is 1.25, and for run 4, it is 0.75, all leading to smooth scalar fields. No remeshing scheme was used in any of these runs.

Run 1 will exhibit only a small time-stepping error, whereas run 2 has both a small time-stepping and spatial discretization error. The errors for these two runs are small enough to get very good agreement with the theoretical solution, as long as anisotropic particles are used with core sizes larger than $\sigma$. Define the aspect ratio for each particle as the largest eigenvalue over the smallest eigenvalue of $\boldsymbol{M}_k$. A large aspect ratio is indicative of particles that are stretched a lot, which is undesirable from a numerical point of view. Since $m_{12} = 0$ for all time, the aspect ratio is here equal to $m_{22}/m_{11}$. At $t = 1$, the aspect ratio for run 1 equals 23.26 and for run 2 it equals 4.79.

Figure 1 gives the comparison between runs 1, 2, and 3 at $t = 0.6$. As stated before, the anisotropic particles yield the exact theoretical result (except for small time-stepping and spatial discretization errors). The results for runs 1 and 2 are virtually indistinguishable while the solution using axisymmetric particles starts to show inaccuracies.

It was shown theoretically that if the core size of the particles is chosen smaller than the filtering size $\sigma$, the solution will blow up in a finite time, as these small-core particles do

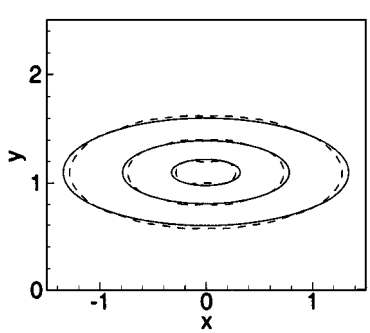


**FIG. 1.** Contourlines 0.1, 0.5, and 1.0 for runs 1 and 2 (solid–results are virtually identical) and 3 (dashed) at $t = 0.6$.

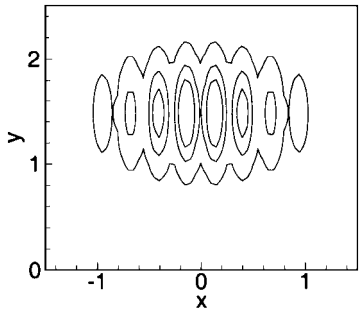


**FIG. 2.** Contourlines 0.1, 0.5, and 1.0 for run 4 at $t = 0.3$.

not correspond to a realistic unfiltered field. Figure 2 illustrates this blowup for run 4 at $t = 0.3$. It is interesting to note that the particles take on an elliptical shape oriented 90° from the direction one would expect. For the parameters given for run 4, the theoretical results predict that the solution blows up at $t = 0.41$, which also happens in the numerical calculation.

### 4.2. *2D Flow*

To show and test different aspects of the particle method and the tensor-diffusivity subgrid model, we use the incompressible velocity field given by

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \sin(x)\sin(y) \\ \cos(x)\cos(y) + \epsilon \sin(\omega t) \end{pmatrix}, \tag{20}$$

where $\epsilon$ is the amplitude and $\omega$ is the frequency of a sinusoidal perturbation. We will use $\epsilon = 0.5$ and $\omega = 1.0$ in this section. Figure 3 shows the streamline pattern at $t = 0$. As an initial condition for the unfiltered scalar field, the Gaussian, $\exp\{[(x - 0.3)^2 + (y + 0.4)^2]/\rho^2\}$, has been used, where $\rho = 0.5$. The filtered initial condition is plotted in Fig. 4. The contourlines are 0.0001, 0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.3, and 0.5, just as in all the contour plots in this section. The diffusivity constant has been set equal to $\kappa = 0.001$ for all computations in this section.

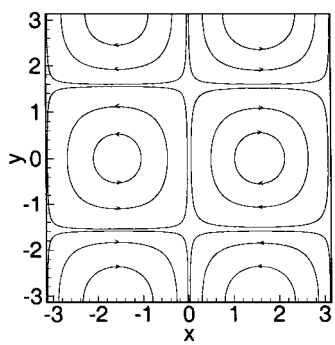


**FIG. 3.** Streamline pattern of (20) at $t = 0$.

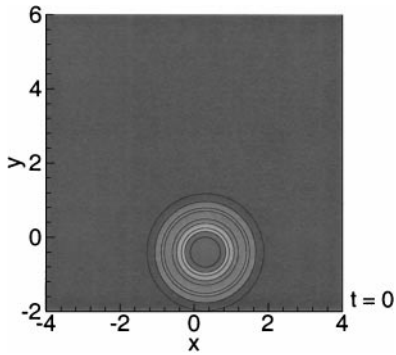


**FIG. 4.** Initial Gaussian distribution centered at (0.3, −0.4).

To test the accuracy of the various schemes, we have computed a DNS solution using a central second-order difference method to solve (1) directly. A regular equidistant grid with a grid spacing of 0.016 was set up (800 by 800 grid points between $-2\pi$ and $2\pi$). The fourth-order Runge–Kutta method as discussed in Ferziger and Peric [17] was used to integrate in time with time step $dt = 0.005$. To compare this converged DNS solution with the results of the LES, this solution has been filtered using $\sigma = 0.15$. The left column of Fig. 5 shows contour plots for the filtered DNS solution at times $t = 3$, 6, and 9.

For purposes of comparison, a finite difference computation using the Smagorinsky subgrid model (with a gridspacing 0.016 and a step size 0.005) has been made. This simulation used the same Gaussian filter with $\sigma = 0.15$ as the Lagrangian computations, thus leading to identical initial scalar fields. Results are given in the right column of Fig. 5 for a Smagorinsky constant of $C_s = 0.2$ and a turbulent Prandtl number $\mathrm{Pr}_t = 1$. The use of other constants was investigated, but $C_s = 0.2$ gave best results. By comparison with the filtered DNS solution, one concludes that the Smagorinsky model recovers the general features of the flow, but it is not diffusive enough in several areas and too diffusive in others.

The addition of the tensor-diffusivity subgrid model to the finite difference code will lead to inherent instabilities in the solution. Negative values of the scalar function appear directly after the start of the computation and lead to a blowup that is exponential in time. This is undoubtedly due to the negative diffusion in the subgrid model. Figure 6 shows contourlines at $t = 2$ for a finite difference calculation (with a grid spacing of 0.03 and a step size of 0.005) with the tensor-diffusivity subgrid model at $t = 2$. The dashed contour-line is 0. Instabilities appear first at the locations where the effects of negative diffusion are highest. Outside these areas the solution is still good. A spectral method will lead to similar instabilities.

As noted earlier, we expect that the Lagrangian particle method will provide the required regularization of the tensor diffusivity model. To implement the particle method, the filtered initial condition is approximated by a sum of Gaussian particles that are initially axisymmetric. In total, 6504 particles were used, each with a core size of 0.15 resulting in an overlap parameter of 4.5. Numerical experiments show that if the core size is equal to or slightly larger than the filtering constant $\sigma$, the remeshing scheme yields the best results. Also, as discussed above, the core size must be larger than the filtering constant $\sigma$. Thus initial condition has been used for all the Lagrangian calculations in this section.

The left column in Fig. 7 gives the solution using the particle method without the subgrid model using anisotropic particles. The solution has not been remeshed. Time integration was performed by the algorithm mentioned in the previous section. Fairly quickly after

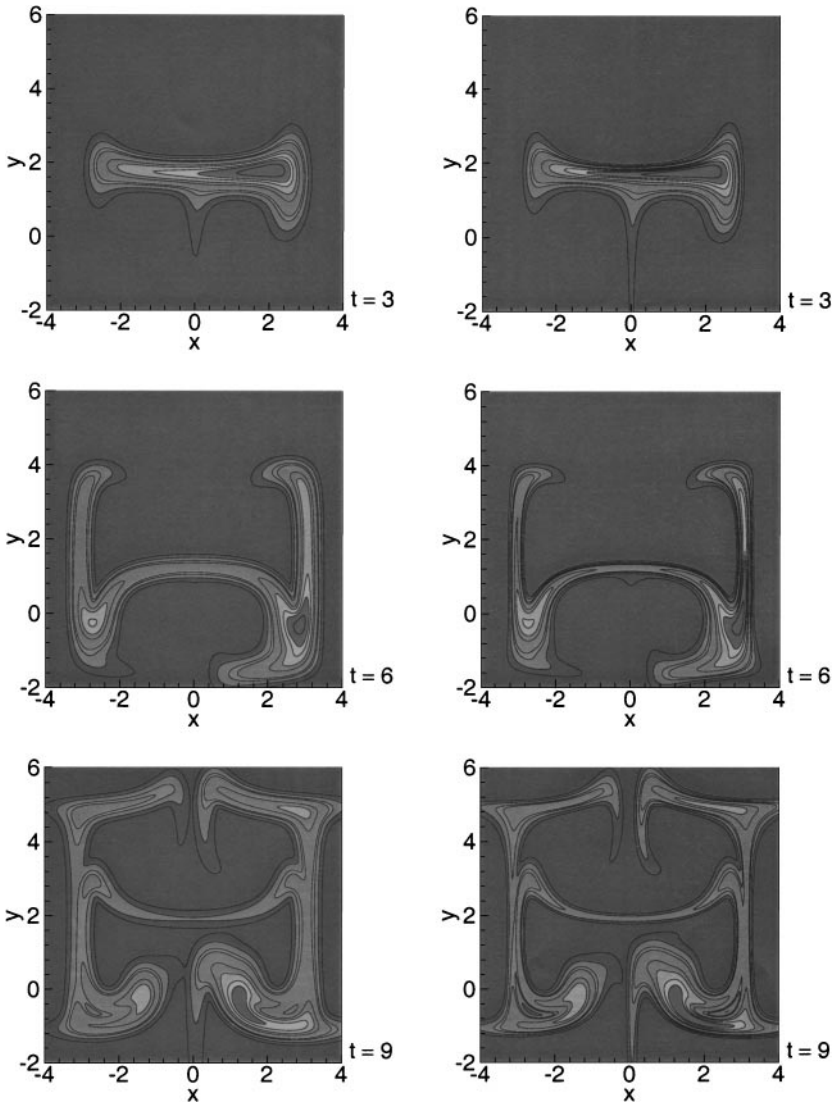


**FIG. 5.** Contour plots for filtered DNS solution (left column) and Smagorinsky subgrid model (right column).

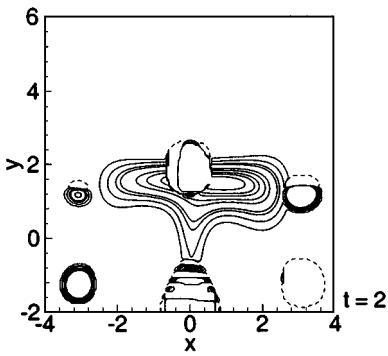


**FIG. 6.** Contour plot using the tensor-diffusivity model in a finite difference calculation.

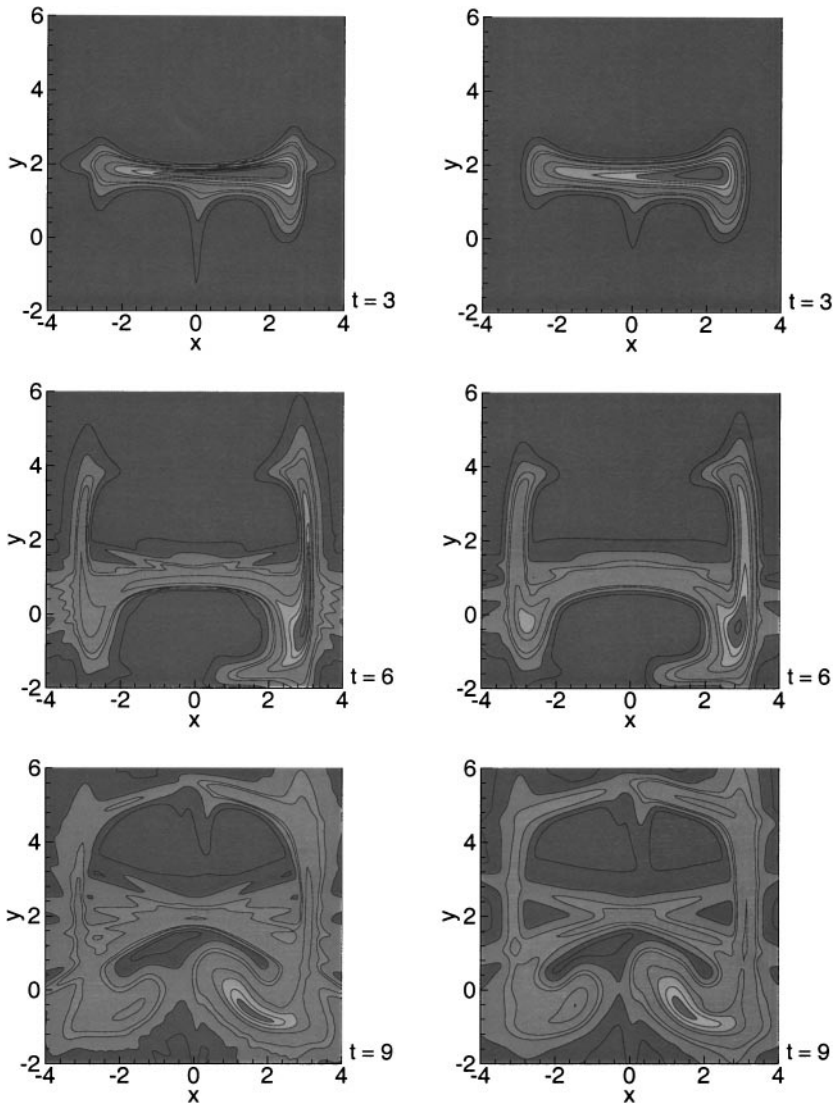


**FIG. 7.** Contour plots for Lagrangian particle method using anisotropic particles and no subgrid model (left column) and the tensor-diffusivity model (right column). No remeshing has been used.

the start of the computation, the solution starts to show differences from the filtered DNS solution. The results after turning on the tensor-diffusivity model are depicted in the right column of 7. Again no remeshing scheme was used. Up to about $t = 3$, this solution is in good agreement with the filtered DNS one. The errors are due to the ever growing core size and changing shape, the increased aspect ratio, and the separation of neighboring particles. At $t = 1$, the maximum aspect ratio has increased to 1.6 and at $t = 4$ to 107. At this time, about 81% of the particles have an aspect ratio larger than 2, and 8% have one larger than 25. At $t = 9$, the maximum aspect ratio has increased until 365 and more than half of the particles have an aspect ratio greater than 10.

The location of the anisotropic particles is given in Fig. 8 at $t = 9$. Each line segment represents a particle. The length of the segment is an indication of the aspect ratio and the

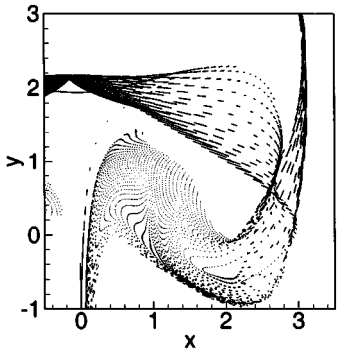


**FIG. 8.** Direction and aspect ratio of anisotropic Gaussian particles at $t = 9$, where the length is a measure of the aspect ratio.

direction of the segment indicates the long axis (direction of the smallest eigenvalue) of the particle. There are regions where the particles are stretched considerably, especially where one of the velocity components is small. It can be inferred that particle paths can cross each other. Even in a computation using a stationary velocity field, this is possible, since particles that are located at the same position will not necessarily move in the same direction. This depends on the shape of the particles.

To improve the accuracy of the solution, we next remesh the scalar field every unit of time. Since the field spreads out over time, the total number of particles increases throughout the computation. Starting with 6504 particles, the number reaches 18,745 particles at $t = 9$. The results are given in the left column of Fig. 9. Very good agreement between this solution and the filtered DNS solution is obtained.

To see the effect of using a higher order particle method, the solution using axisymmetric particles is plotted in the right column of Fig. 9. The solution was remeshed every time unit, resulting in 20,520 particles at $t = 9$. The use of axisymmetric particles leads to results that are almost as good as when one uses anisotropic particles. Since the computation using axisymmetric particles is significantly faster than using anisotropic particles (8 minutes versus 36 minutes), they are the preferred choice based on computational efficiency.

Table II summarizes the seven different computational runs performed in this section. The first column gives the run number, followed by the method used, where FD stands for finite difference and Lagr for Lagrangian method. The model indicates if the computation was

**TABLE II**
**Runs for the 2D-Flow Test Problem**

| Run | Method | Model | No. Particles/Grid Points | Remeshed | CPU Time |
|---|---|---|---|---|---|
| 1 | FD | DNS | $800 \times 800$ points | — | 24 hr |
| 2 | FD | Smagorinsky | $800 \times 800$ points | — | 24 hr |
| 3 | FD | TD | $800 \times 800$ points | — | — |
| 4 | Lagr | No model | 6504 particles | no | 12 min |
| 5 | Lagr | TD—Anisotropic | 6504 particles | no | 9 min |
| 6 | Lagr | TD—Anisotropic | 6504–21,593 particles | yes | 36 min |
| 7 | Lagr | TD—Isotropic | 6504–23,960 particles | yes | 8 min |

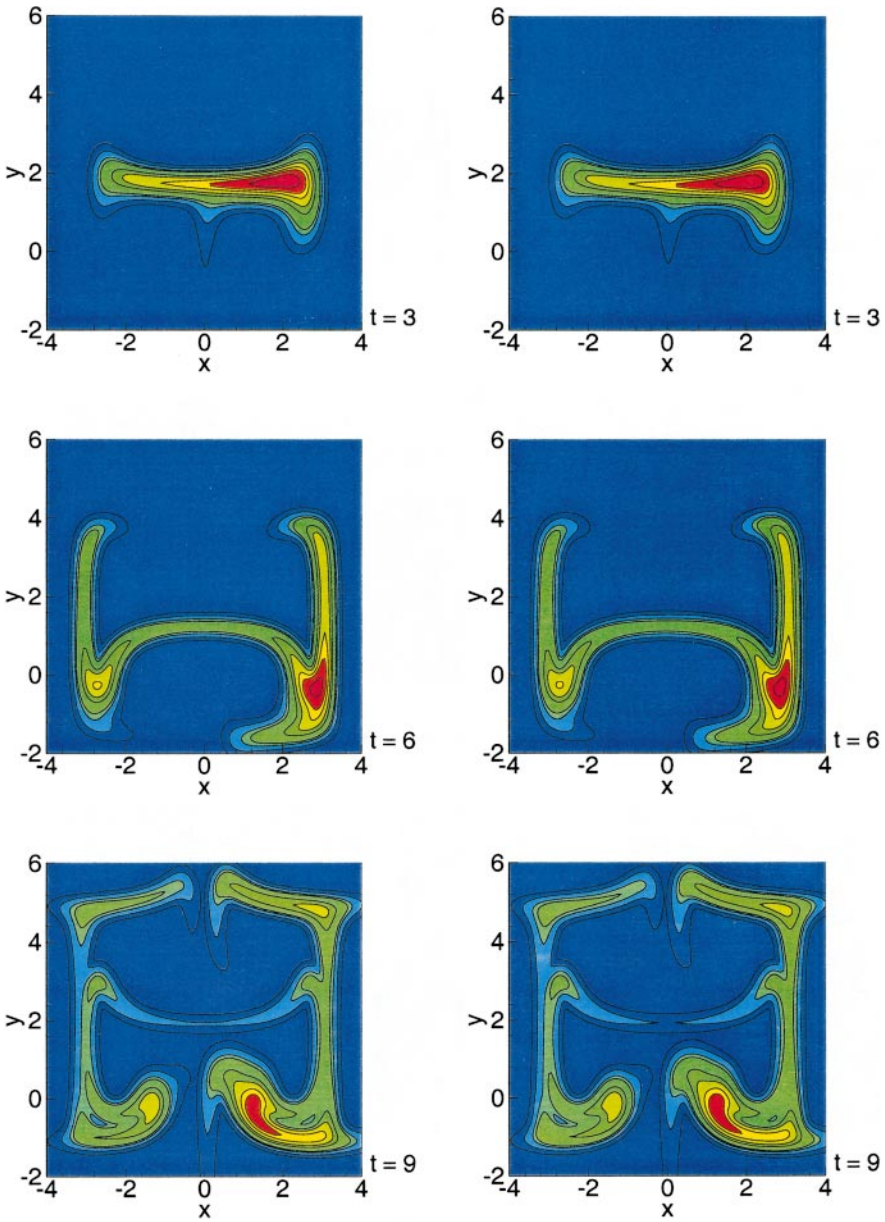


**FIG. 9.** Contour plots for anisotropic particles (left column) and axisymmetric particles (right column). The solution has been remeshed every time unit.

DNS or used the Smagorinsky or tensor-diffusivity (TD) model. In the case of a Lagrangian method, it also indicates if anisotropic or isotropic Gaussian particles were used. For the finite difference method, the grid size is given and for the Lagrangian method the total number of particles at the beginning of the run. The number of particles at the end of a run is printed if it differs from the initial number. This is only the case if remeshing was used as indicated in the remeshed column.

The approximate CPU time on a Pentium 650 MHz processor is given. At first glance, the Lagrangian appears to be about two orders of magnitude faster than the finite difference calculations. However, note that the grid sizes also differ by two orders of magnitude. The clear advantage of the particle method is that particles are only needed in areas where a significant scalar quantity is present. No valuable computational time is lost in areas where no scalar quantity is present.

## 5. CONCLUSIONS AND FUTURE RESEARCH

A new subgrid model was obtained by expanding the unknown variable in terms of an infinite sum of known variables and truncating this series. The model was named the tensor-diffusivity model and does not have any degrees of freedom. It was shown that the subgrid model introduces negative diffusion in at least one spatial direction and is frame indifferent.

Anisotropic Gaussian particles were introduced and equations of motion were derived using an expansion in Hermite polynomials. Equations for the widely used axisymmetric Gaussian particles were obtained as well. To prevent particles from becoming too large, too elliptical, or too widely separated in a particular direction, a remeshing scheme tailored to the characteristics of the model has been implemented.

A stagnation flow can be solved exactly and was used to illustrate different aspects of the subgrid model and the particle method. A simple two-dimensional incompressible flow has been used to show that the particle method yields accurate results. Even for stationary flows, path lines can cross. The particle method resulted in more accurate solutions than those of the Smagorinsky subgrid model.

Both anisotropic and axisymmetric Gaussian particles can be used to obtain good results. Anisotropic particles do not have to be remeshed as often as axisymmetric particles, but because of computational efficiency and better accuracy for the remeshing procedure, axisymmetric particles are the preferred choice to use in this type of simulation.

Research is currently underway to apply the tensor-diffusivity model in three dimensions on simple model flows as well as to more realistic flows such as decaying and forced homogeneous turbulence. For these flows, an exact velocity field is not known and interpolation schemes are needed, adding to the cost of the model. Alternate schemes for regularization and obtaining better solutions are being considered. In cases where the velocity field is fully resolved, we found a novel method for the time evolution of the location of the particles, which is currently being tested.

## APPENDIX A

It will be shown that the filtered product $\widehat{fg}$ is related to the filtered functions $\hat{f}$ and $\hat{g}$ for a Gaussian filter as

$$\widehat{fg}(\boldsymbol{x}) = \sum_{n=0}^{\infty} \frac{1}{n!} \left(\frac{\sigma^2}{2}\right)^n \frac{\partial^n \hat{f}}{\partial x_{i_1} \partial x_{i_2} \cdots \partial x_{i_n}} \frac{\partial^n \hat{g}}{\partial x_{i_1} \partial x_{i_2} \cdots \partial x_{i_n}}, \tag{A.1}$$

where a sum over repeated indices is implied and $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ are arbitrary functions in

$C^\infty$. By definition,

$$\widehat{fg}(\boldsymbol{x}) = \frac{1}{(\sigma\sqrt{\pi})^d}\int_V f(\boldsymbol{x}')g(\boldsymbol{x}')\exp\left(-\frac{|\boldsymbol{x}'-\boldsymbol{x}|^2}{\sigma^2}\right)d\boldsymbol{x}', \tag{A.2}$$

where $d$ is the spatial dimension and the integration is taken over the infinite spatial domain $V$. For simplicity and without loss of generality, assume one spatial dimension. Expand $f(x')$ in a series of Hermite polynomials as $f(x') = \sum_{n=0}^{\infty} \bar{f}_n(x) H_n(\frac{x'-x}{\sigma})$, where $H_n$ is the $n$-th Hermite polynomial and the coefficients $\bar{f}_n(x)$ are given by

$$\bar{f}_n(\boldsymbol{x}) = \frac{1}{2^n n!}\frac{1}{\sigma\sqrt{\pi}}\int_{-\infty}^{\infty} f(x')\exp\left(-\frac{(x'-x)^2}{\sigma^2}\right)H_n\left(\frac{x'-x}{\sigma}\right)dx'. \tag{A.3}$$

After partial integration of the right-hand side $n$ times, we find $\bar{f}_n(x) = \frac{\sigma^n}{2^n n!}\frac{\partial^n \hat{f}(x)}{\partial x^n}$. Plugging these results in (A.2) gives

$$\widehat{fg}(x) = \sum_{n=0}^{\infty}\frac{\sigma^n}{2^n n!}\frac{\partial^n \hat{f}(x)}{\partial x^n}\int_{-\infty}^{\infty}\frac{1}{\sigma\sqrt{\pi}}g(x')\exp\left(-\frac{(x'-x)^2}{\sigma^2}\right)H_n\left(\frac{x'-x}{\sigma}\right)dx'. \tag{A.4}$$

The integral on the right-hand side can be written, following (A.3), as $2^n n!\bar{g}_n(x)$ to end up with

$$\widehat{fg}(x) = \sum_{n=0}^{\infty}\frac{\sigma^n}{2^n n!}\frac{\partial^n \hat{f}(x)}{\partial x^n}2^n n!\bar{g}_n(x) = \sum_{n=0}^{\infty}\frac{\sigma^n}{2^n n!}\frac{\partial^n \hat{f}(x)}{\partial x^n}\sigma^n\frac{\partial^n \hat{g}(x)}{\partial x^n}, \tag{A.5}$$

which is the one-dimensional version of (A.1).

## APPENDIX B

We want to solve the filtered advection–diffusion equation (5) by approximating the unknown scalar field $\psi(\boldsymbol{x}, t)$ by a sum of $N$ anisotropic Gaussian particles given by (10). Time evolution equations for the shape matrix $\boldsymbol{M}_k$ and the location $\boldsymbol{x}_k$ are derived by expanding each term of (5) in a series of Hermite polynomials and setting the coefficients of lower order powers to zero. Without loss of generality, only one particle $k$ and two dimensions are considered. In the principle coordinate system $(\xi, \eta)$ of matrix $\boldsymbol{M}_k$, this particle $k$ is given by

$$\psi_k(\boldsymbol{\xi}, t) = \frac{a_k\sqrt{\lambda_\xi\lambda_\eta}}{\pi\delta_k^2}\exp\left(-\frac{\lambda_\xi(\xi-\xi_k)^2 + \lambda_\eta(\eta-\eta_k)^2}{\delta_k^2}\right), \tag{B.1}$$

where $(1, 0)$ and $(0, 1)$ are the orthogonal unit eigenvectors corresponding to the eigenvalues of $\boldsymbol{M}_k$, $\lambda_\xi$ and $\lambda_\eta$, respectively. Set $H_{m,n} = H_m^\xi H_n^\eta = H_m^\xi\{[\lambda_\xi(\xi-\xi_k)]/\delta_k\}H_n^\eta\{[\lambda_\eta(\eta-\eta_k)]/\delta_k\}$ where $H_m$ is the $m$-th Hermite polynomial. Introduce the diagonal matrix $\boldsymbol{\Lambda}_k$ with the eigenvalues $\lambda_\xi$ and $\lambda_\eta$ as its elements. The filtered advection–diffusion equation (5) does not

change due to the transformation except that all the derivatives are now taken with respect to the $\boldsymbol{\xi}$ coordinate. Hats on top of variables will be dropped.

The time derivative of $\psi_k$ can be expressed in a sum of Hermite polynomials $H_{m,n}$ as

$$\begin{aligned}\frac{\partial \psi_k}{\partial t} &= \left[\frac{\frac{d}{dt}[\det(\boldsymbol{\Lambda}_K)]}{\det(\boldsymbol{\Lambda}_k)} + \frac{2}{\delta_k^2}\frac{d\boldsymbol{\xi}_k^T}{dt}\boldsymbol{\Lambda}_k(\boldsymbol{\xi}-\boldsymbol{\xi}_k) - \frac{(\boldsymbol{\xi}-\boldsymbol{\xi}_k)^T}{\delta_k^2}\frac{d\boldsymbol{\Lambda}_k}{dt}(\boldsymbol{\xi}-\boldsymbol{\xi}_k)\right]\psi_k \\ &= \left[-\frac{1}{4\lambda_\xi}\frac{d\lambda_\xi}{dt}H_{2,0} - \frac{1}{4\lambda_\eta}\frac{d\lambda_\eta}{dt}H_{0,2} + \frac{\sqrt{\lambda_\xi}}{\delta_k}\frac{d\xi_k}{dt}H_{1,0} + \frac{\sqrt{\lambda_\eta}}{\delta_k}\frac{d\eta_k}{dt}H_{0,1}\right]\psi_k, \qquad \text{(B.2)}\end{aligned}$$

where the equality $\{\frac{d}{dt}[\det(\boldsymbol{A})]/\det(\boldsymbol{A})\} = \frac{dA_{ij}}{dt}(A^{-1})_{ji}$, valid for any nonsingular matrix $\boldsymbol{A}$, has been used. The advection term ($\boldsymbol{u}\cdot\boldsymbol{\nabla}\psi = \boldsymbol{\nabla}\cdot\boldsymbol{u}\psi$) can be expressed in a series of Hermite polynomials by first expanding the velocity field in an infinite series of Hermite polynomials, followed by carrying out the gradient operator using $H_n(x)e^{-x^2} = -\frac{d}{dx}[H_{n-1}(x)\exp(-x^2)]$,

$$\begin{aligned}\boldsymbol{u}\cdot\boldsymbol{\nabla}\psi_k &= \sum_{m,n=0}^{\infty}\boldsymbol{\nabla}\cdot(\bar{\boldsymbol{u}}_{m,n}H_{m,n}\psi_k) = \sum_{m,n=0}^{\infty}\bar{\boldsymbol{u}}_{m,n}\boldsymbol{\nabla}\cdot(H_{m,n}\psi_k) \\ &= \sum_{m,n=0}^{\infty}\left[-\frac{\sqrt{\lambda_\xi}}{\delta_k}\bar{u}_{m,n}H_{m+1,n} - \frac{\sqrt{\lambda_\eta}}{\delta_k}\bar{v}_{m,n}H_{m,n+1}\right]\psi_k,\end{aligned} \qquad \text{(B.3)}$$

where the functions $\bar{\boldsymbol{u}}_{m,n}$ are defined by

$$\bar{\boldsymbol{u}}_{m,n}(\boldsymbol{\xi}_k,t) = \frac{\sqrt{\lambda_\xi}\lambda_\eta}{h_m h_n \delta_k^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\exp\left(-\frac{(\boldsymbol{\xi}-\boldsymbol{\xi}_k)^T\boldsymbol{\Lambda}_k(\boldsymbol{\xi}-\boldsymbol{\xi}_k)}{\delta_k^2}\right)\boldsymbol{u}(\boldsymbol{\xi},t)H_{m,n}\,d\xi\,d\eta, \qquad \text{(B.4)}$$

and $h_n = 2^n n!\sqrt{\pi}$. Note that $\bar{\boldsymbol{u}}_{m,n} = O(\delta_k^{m+n})$. The diffusion term can be written in terms of Hermite polynomials as

$$\begin{aligned}\kappa\nabla^2\psi_k &= \left[-\frac{2\kappa}{\delta_k^2}(\Lambda_k)_{ii} + \frac{4\kappa}{\delta_k^4}(\boldsymbol{\xi}-\boldsymbol{\xi}_k)^T\boldsymbol{\Lambda}_k\boldsymbol{\Lambda}_k(\boldsymbol{\xi}-\boldsymbol{\xi}_k)\right]\psi_k \\ &= \left[\frac{\kappa\lambda_\xi}{\delta_k^2}H_{2,0} + \frac{\kappa\lambda_\eta}{\delta_k^2}H_{0,2}\right]\psi_k. \qquad \text{(B.5)}\end{aligned}$$

Finally, consider the tensor-diffusivity term

$$\begin{aligned}-\frac{\sigma^2}{2}S_{ij}\frac{\partial^2\psi_k}{\partial\xi_i\partial\xi_j} &= -\frac{\sigma^2}{2}\left[-\frac{2}{\delta_k^2}S_{ij}\Lambda_{ij} + \frac{4}{\delta_k^4}(\boldsymbol{\xi}-\boldsymbol{\xi}_k)^T\boldsymbol{\Lambda}_k\boldsymbol{S}\boldsymbol{\Lambda}_k(\boldsymbol{\xi}-\boldsymbol{\xi}_k)\right]\psi_k \\ &= -\frac{\sigma^2}{2}\left[\frac{\lambda_\xi}{\delta_k^2}\frac{\partial u}{\partial\xi}H_{2,0} + \frac{\lambda_\eta}{\delta_k^2}\frac{\partial v}{\partial\eta}H_{0,2} + \frac{\sqrt{\lambda_\xi\lambda_\eta}}{\delta_k^2}\left(\frac{\partial u}{\partial\eta}+\frac{\partial v}{\partial\xi}\right)H_{1,1}\right]\psi_k.\end{aligned} \qquad \text{(B.6)}$$

After expressing the components of the strain rate tensor ($\frac{\partial u_i}{\partial \xi_j}$) in a series of Hermite polynomials we end up with

$$
\begin{aligned}
&-\frac{\sigma^2}{2} S_{ij} \frac{\partial^2 \psi_k}{\partial \xi_i \partial \xi_j} \\
&\quad = \sum_{m,n=0}^{\infty} -\frac{\sigma^2}{2} \Bigg\{ \frac{\sqrt{\lambda_\xi \lambda_\eta}}{\delta_k^2} \left( \frac{\overline{\partial u}}{\partial \eta} + \frac{\overline{\partial v}}{\partial \xi} \right)_{m,n} \left( H^{\xi}_{m+1} + 2m H^{\xi}_{m-1} \right) \left( H^{\eta}_{n+1} + 2n H^{\eta}_{n-1} \right) \\
&\qquad + \frac{\lambda_\xi}{\delta_k^2} \frac{\overline{\partial u}}{\partial \xi_{m,n}} \left( H_{m+2,n} + 4m H_{m,n} + 4m(m-1) H_{m-2,n} \right) \\
&\qquad + \frac{\lambda_\eta}{\delta_k^2} \frac{\overline{\partial v}}{\partial \eta_{m,n}} \left( H_{m,n+2} + 4n H_{m,n} + 4n(n-1) H_{m,n-2} \right) \Bigg\} \psi_k .
\end{aligned} \tag{B.7}
$$

Next we will set the coefficients in front of the lower order Hermite polynomials equal to zero. Owing to incompressibility, the coefficient of $H_{0,0}$ is automatically zero. Setting the coefficients of $H_{1,0}$ and $H_{0,1}$ equal to zero, we get

$$
\frac{d\xi_k}{dt} = \bar{u} - \frac{\sigma^2}{2} \left( \frac{\overline{\partial^2 u}}{\partial \xi^2} + \frac{\overline{\partial^2 u}}{\partial \eta^2} \right), \tag{B.8}
$$

$$
\frac{d\eta_k}{dt} = \bar{v} - \frac{\sigma^2}{2} \left( \frac{\overline{\partial^2 v}}{\partial \xi^2} + \frac{\overline{\partial^2 v}}{\partial \eta^2} \right), \tag{B.9}
$$

where the shortcut $\bar{f} = \bar{f}_{0,0}$ has been introduced. Using partial integration, we can express $\bar{f}_{m,n}$ in terms of derivatives of $\bar{f}$. If we transform back to the original coordinate system and combine both equations, we end up with (12). Setting the coefficients of $H_{2,0}$ and $H_{0,2}$ equal to zero, we find

$$
\frac{d\lambda_\xi}{dt} = -2\lambda_\xi \frac{\overline{\partial u}}{\partial \xi} - 4\kappa \frac{\lambda_\xi \lambda_\xi}{\delta_k^2} + \frac{2\sigma^2}{\delta_k^2} \lambda_\xi \lambda_\xi \frac{\overline{\partial u}}{\partial \xi} + \sigma^2 \lambda_\xi \left( \frac{\overline{\partial^3 u}}{\partial \xi^3} + \frac{\overline{\partial^3 u}}{\partial \xi \partial \eta^2} \right), \tag{B.10a}
$$

$$
\frac{d\lambda_\eta}{dt} = -2\lambda_\eta \frac{\overline{\partial v}}{\partial \eta} - 4\kappa \frac{\lambda_\eta \lambda_\eta}{\delta_k^2} + \frac{2\sigma^2}{\delta_k^2} \lambda_\eta \lambda_\eta \frac{\overline{\partial v}}{\partial \eta} + \sigma^2 \lambda_\eta \left( \frac{\overline{\partial^3 u}}{\partial \xi^2 \partial \eta} + \frac{\overline{\partial^3 v}}{\partial \eta^3} \right), \tag{B.10b}
$$

and for $H_{1,1}$

$$
\begin{aligned}
0 = &-\lambda_\xi \frac{\overline{\partial u}}{\partial \eta} - \lambda_\eta \frac{\overline{\partial v}}{\partial \xi} + \frac{\sigma^2 \lambda_\xi \lambda_\eta}{\delta_k^2} \left( \frac{\overline{\partial u}}{\partial \eta} + \frac{\overline{\partial v}}{\partial \xi} \right) \\
&+ \frac{\sigma^2 \lambda_\xi}{2} \left( \frac{\overline{\partial^3 u}}{\partial \xi^2 \partial \eta} + \frac{\overline{\partial^3 u}}{\partial \eta^3} \right) + \frac{\sigma^2 \lambda_\eta}{2} \left( \frac{\overline{\partial^3 v}}{\partial \xi \partial \eta^2} + \frac{\overline{\partial^3 v}}{\partial \xi^3} \right).
\end{aligned} \tag{B.11}
$$

It is straightforward to check that if we combine the three equations above in matrix form, we obtain

$$
\begin{aligned}
\frac{d\boldsymbol{\Lambda}_k}{dt} = &-\overline{\boldsymbol{\nabla u}} \boldsymbol{\Lambda}_k - \boldsymbol{\Lambda}_k \overline{\boldsymbol{\nabla u}}^T - \frac{4\kappa}{\delta_k^2} \boldsymbol{\Lambda}_k \boldsymbol{\Lambda}_k + 2\frac{\sigma^2}{\sigma_k^2} \boldsymbol{\Lambda}_k \boldsymbol{S} \boldsymbol{\Lambda}_k \\
&+ \frac{\sigma^2}{2} \overline{\boldsymbol{\nabla} \nabla^2 \boldsymbol{u}} \boldsymbol{\Lambda}_k + \frac{\sigma^2}{2} \boldsymbol{\Lambda}_k \overline{\boldsymbol{\nabla} \nabla^2 \boldsymbol{u}}^T .
\end{aligned} \tag{B.12}
$$

If we transform back to the original coordinate system, we get (14).

## APPENDIX C

Without loss of generality, Eq. (16) will be derived in one dimension, using the same notation as in Section 3.2. Define the error $\epsilon(x)$ of the approximation between the old and the new scalar fields as the difference between both fields,

$$\epsilon(x) = \sum_{k=1}^{N} \frac{a_k \sqrt{\det(\lambda_k)}}{\sqrt{\pi}\delta_k} \exp\left(-\frac{\lambda_k (x - x_k)^2}{\delta_k^2}\right) - \sum_{l=1}^{M} \frac{b_l}{\sqrt{\pi}\tau} \exp\left(-\frac{(x - \xi_l)^2}{\tau^2}\right), \quad \text{(C.1)}$$

where the shape matrix $\boldsymbol{M}_k$ has been replaced by $\lambda_k$, the eigenvalue (and only element) of $\boldsymbol{M}_k$. We will use the least-square-error approximation, which results in a linear system of $M$ equations for the unknown $b_l$'s, where the $i$-th equation is given by

$$\sum_{l=1}^{M} \frac{b_l}{\sqrt{2}} \exp\left(-\frac{(\xi_l - \xi_i)^2}{2\tau^2}\right) = \sum_{k=1}^{N} \int_{-\infty}^{\infty} \frac{a_k \sqrt{\lambda_k}}{\sqrt{\pi}\delta_k} \exp\left(-\frac{\lambda_k (x' - x_k)^2}{\delta_k^2}\right) \exp\left(-\frac{(x' - \xi_i)^2}{\tau^2}\right) dx'. \quad \text{(C.2)}$$

Express the first exponential on the right-hand side as

$$\exp\left(-\frac{\lambda_k (x' - x_k)^2}{\delta_k^2}\right) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{\pi}\sqrt{\delta_k^2 - \lambda_k \tau^2}} \exp\left(-\frac{(x' - x_k - \hat{x})^2}{\tau^2}\right) \exp\left(-\frac{\lambda_k \hat{x}^2}{\delta_k^2 - \lambda_k \tau^2}\right) d\hat{x}, \quad \text{(C.3)}$$

which is only possible for $\tau^2 < \frac{\delta_k^2}{\lambda_k}$ for all $k$. This puts an upper bound on the new core size of the particles. Substituting in (C.2), carrying out the integration over $x'$, and replacing $\hat{x}$ by $x' - x_k$, we have

$$\sum_{l=1}^{M} \frac{b_l}{\sqrt{2}} \exp\left(-\frac{(\xi_l - \xi_i)^2}{2\tau^2}\right) = \sum_{k=1}^{N} \int_{-\infty}^{\infty} \frac{a_k \sqrt{\lambda_k}}{\sqrt{2\pi}\sqrt{\delta_k^2 - \lambda_k \tau^2}} \exp\left(-\frac{\lambda_k (x' - x_k)^2}{\delta_k^2 - \lambda_k \tau^2}\right) \exp\left(-\frac{(x' - \xi_i)^2}{2\tau^2}\right) dx'. \quad \text{(C.4)}$$

The last exponential on the right-hand side is now approximated as

$$\exp\left(-\frac{(x' - \xi_i)^2}{2\tau^2}\right) \approx \sum_{l=1}^{M} m(x' - \xi_l) \exp\left(-\frac{(\xi_l - \xi_i)^2}{2\tau^2}\right), \quad \text{(C.5)}$$

where the function $m(x)$ has to be chosen. See for example Cottet and Koumoutsakos [18] for a discussion of possible choices. After substitution of this equation in (C.4), we have on both sides of the equation a sum over the set of new particles, and we find the following

explicit expressions for the unknown coefficients $b_l$'s:

$$b_l = \sum_{k=1}^{N} \int_{-\infty}^{\infty} \frac{a_k \sqrt{\lambda_k}}{\sqrt{\pi}\sqrt{\delta_k^2 - \lambda_k \tau^2}} \exp\left( -\frac{\lambda_k (x' - x_k)^2}{\delta_k^2 - \lambda_k \tau^2} \right) m(x' - \xi_l)\, dx'. \tag{C.6}$$

We have found that the simplest possible choice, $m(x) = \delta(x)h$, where $\delta(x)$ is the Dirac delta function, gives very good results. Equation (C.6) then reduces to

$$b_l = \sum_{k=1}^{N} \frac{a_k \sqrt{\lambda_k}}{\sqrt{\pi}\sqrt{\delta_k^2 - \lambda_k \tau^2}} \exp\left( -\frac{\lambda_k (x_l - x_k)^2}{\delta_k^2 - \lambda_k \tau^2} \right) h. \tag{C.7}$$

In case of $d$ spatial dimensions, the expressions for the new amplitudes are obtained by multiplying (C.7) $d$ times in the principle coordinate system of the shape matrix $\boldsymbol{M}_k$ for each particle $k$, before transforming back to the orginal coordinate system. In two dimensions, we obtain (16).

## ACKNOWLEDGMENT

This research has been made possible by a grant from the Department of Energy.

## REFERENCES

1. K. W. Bedford and W. K. Yeo, Conjunctive filtering procedures in surface water flow and transport, in *Large Eddy Simulation of Complex Engineering and Geophysical Flows*, edited by B. Galperin and S. A. Orszag (Cambridge University Press, Cambridge, UK, 1993), pp. 513–537.
2. W. K. Yeo, *A Generalized High Pass/Low Pass Averaging Procedure for Deriving and Solving Turbulent Flow Equations*, Ph.D. thesis (The Ohio State University, 1987).
3. A. Leonard, Large-eddy simulation of chaotic convection and beyond, AIAA Paper 97–0204 (1997).
4. A. J. Chorin, Numerical study of slightly viscous flow, *J. Fluid Mech*. **57**, 785 (1973).
5. A. Leonard, Vortex methods for flow simulation, *J. Comput. Phys*. **37**, 289 (1980).
6. C. Greengard, The core spreading vortex method approximates the wrong equation, *J. Comput. Phys*. **61**, 345 (1985).
7. L. F. Rossi, Resurrecting core spreading vortex methods: A new scheme that is both deterministic and convergent, *SIAM J. Sci. Comput*. **17**, 370 (1996).
8. L. F. Rossi, Merging computational elements in vortex simulations, *SIAM J. Sci. Comput*. **17**, 1014 (1997).
9. D. Carati, G. S. Winckelmans, and H. Jeanmart, Exact expansions for filtered-scales modelling with a wide class of les filters, in *Direct and Large-Eddy Simultation III*, edited by P. R. Voke, N. D. Sandham, and L. Kleiser (Kluwer Academic, Dordrecht/Norwell, MA, 1999), pp. 213–224.
10. M. E. Gurtin, *An Introduction to Continuum Mechanics* (Academic Press, San Diego, 1981).
11. C. Fureby, On subgrid scale modeling in large eddy simulations of compressible flow, *Phys. Fluids* **8**, 1301 (1996).
12. G. I. Barenblatt, M. Bertsch, R. Dal Passo, and M. Ughi, A degenerate pseudoparabolic regularization of a nonlinear forward–backward heat equation arising in the theory of heat and mass exchange in stably stratified turbulent shear flow, *SIAM J. Math. Anal*. **24**, 1414 (1993).
13. R. Krasny, A study of singularity formation in a vortex sheet by the point vortex approximation, *J. Fluid Mech*. **167**, 65 (1986).
14. A. Leonard and G. S. Winckelmans, A tensor-diffusivity subgrid model for large-eddy simulation, in *Direct and Large-Eddy Simultation III*, edited by P. R. Voke, N. D. Sandham, and L. Kleiser (Kluwer Academic Publishers, Dordrecht/Norwell, MA, 1999), pp. 147–162.

15. L. F. Rossi, High order vortex methods with deforming elliptical Gaussian blobs 1: Derivation and validation, submitted for publication.
16. W. H. Press, S. A. Teukolsky, W. T. Vettering, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing* (Cambridge University Press, Cambridge, UK, 1995).
17. J. H. Ferziger and M. Perić, *Computational Methods for Fluid Dynamics* (Springer-Verlag, Berlin/New York, 1996).
18. G.-H. Cottet and P. Koumoutsakos, *Vortex Methods: Theory and Applications* Cambridge University Press, Cambridge, UK, 1999).