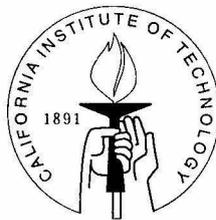


# The Filtered Advection-Diffusion Equation: Lagrangian Methods and Modeling

Thesis by  
Piet Moeleker

In Partial Fulfillment of the Requirements  
for the Degree of  
Doctor of Philosophy



California Institute of Technology  
Pasadena, California

2000  
(Submitted May 2, 2000)

© 2000

Piet Moeleker

All Rights Reserved

## Acknowledgments

I would like to take this opportunity to thank my advisor A. Leonard for guiding me throughout my Ph.D. studies. His open-door policy, his grasp of the material, and his interest in my work and general well-being made my study fruitful and enjoyable.

D.I. Pullin stood always ready to give me his helpful comments and insights. He provided not only the numerical code for the 3D forced isotropic turbulence calculations, but also the necessary instructions to use it to the fullest.

Thanks are also appropriate for my friends in the Iris Lab for helping me with all my computer related problems.

This research was in part sponsored by grant DE-AC03-98EE50506 from the Department of Energy.

## Abstract

This research focuses on the incompressible scalar advection-diffusion equation. After applying a Gaussian filter, an infinite series expansion is found for the advection term to obtain a closed equation. Only the first two terms in this expansion are retained yielding the tensor-diffusivity subgrid model. This model can be interpreted as a tensor-diffusivity term which is proportional to the rate-of-strain tensor of the large-scale filtered velocity field. Due to the negative diffusion in the stretching directions, care needs to be taken in the choice of a numerical method.

The scalar field is decomposed in a collection of anisotropic or axisymmetric Gaussian particles. Equations of motion for the location and the shape/size of the particles are derived using an expansion in Hermite polynomials. A novel, accurate remeshing scheme was found resulting in explicit expressions for the amplitudes of the new set of particles.

A stagnation flow was used for illustrative purposes and validation. Using a 2D time-dependent velocity field yielding chaotic advection, both axisymmetric and anisotropic particles yield good agreement with filtered direct numerical simulations and compare favorably with the Smagorinsky subgrid model. Computational efficiency makes axisymmetric particles the preferred choice.

A literature study using a 3D stationary one-parameter chaotic velocity field was used to validate model and particle-method in 3D. For highly chaotic fields good agreement was obtained with this study. Computations have been performed for 3D forced isotropic periodic turbulence to study scalar mixing. Comparisons with literature are made.

It was shown that when the unfiltered velocity field is known, the most accurate results are obtained by moving particles using this field. It was concluded that a good subgrid model modifies the equation of motion to get a good approximation to the unfiltered velocity field.

## Samenvatting

Om hele complexe stromingen, zoals bijvoorbeeld het weer, nauwkeurig te berekenen met een computer is teveel computertijd nodig. Dit komt omdat men vele roosterpunten nodig heeft om uit te rekenen wat op een kleine schaal gebeurt (bijvoorbeeld een paar kilometer), aangezien dat invloed heeft op wat zich op grote schaal afspeelt (bijvoorbeeld een paar honderd kilometer). Om toch voorspellingen te doen voor deze stromingen, berekent men alleen wat er gaande is op de grote schaal en modelleert men het effect van de kleine schaal op de grote schaal.

Dit onderzoek gaat over zo een model. Wij bestuderen de scalaire transportvergelijking. Deze vergelijking beschrijft de verspreiding van een grootte die geen effect heeft op het snelheidsveld. Zo valt te denken aan een rooksignaal over een auto om de stroming zichtbaar te maken of aan de verspreiding van giftige gassen in de atmosfeer.

Numeriek geven wij de scalaire grootte weer als een verzameling deeltjes. Elk deeltje heeft zijn erg sterkte, plaats en (elliptische) vorm. De som van alle deeltjes levert dan bijvoorbeeld het rooksignaal op. Wij leiden voor elk deeltje een vergelijking af om de nieuwe plaats en vorm te berekenen op een later tijdstip. Omdat de deeltjes in tijd groeien en erg elliptisch worden, wat numerieke fouten tot gevolg heeft, vervangen wij de oude verzameling deeltjes zo en nu dan door een nieuwe verzameling deeltjes, die allemaal weer klein en rond zijn.

Het model en de numerieke methode zijn getest in twee en drie dimensies. Wij zijn begonnen met eenvoudige problemen, waarvoor een exacte oplossing bekend is. Onze berekeningen zijn ook vergeleken met andere berekeningen in de literatuur of met zeer nauwkeurige berekeningen, die zowel de kleine als de grote schalen berekend hebben. Ons model levert goede resultaten op in vergelijking met een ander model. Tenslotte zijn nog diverse berekeningen gemaakt aan de verspreiding van een scalaire grootte in een turbulent snelheidsveld om het menggedrag te bestuderen.

# Contents

<b>Acknowledgments</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>Samenvatting</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Large Eddy Simulation and Subgrid Modeling . . . . .	2
1.2 Numerical Implementation . . . . .	4
1.3 Overview . . . . .	5
<b>2 Tensor-Diffusivity Subgrid Model</b>	<b>6</b>
2.1 Filtering . . . . .	6
2.2 Tensor-Diffusivity Subgrid Model . . . . .	7
2.3 Material Frame Indifference . . . . .	10
2.4 Smagorinsky Subgrid Model . . . . .	11
<b>3 Lagrangian Particle Method</b>	<b>13</b>
3.1 Finite Difference Method . . . . .	13
3.2 Particle Method . . . . .	15
3.3 Anisotropic Gaussian Particles . . . . .	17
3.4 Remeshing . . . . .	21
<b>4 2D Flow Examples</b>	<b>25</b>
4.1 Rotating Flow . . . . .	25
4.2 Stagnation Flow . . . . .	27
4.2.1 Theory . . . . .	27
4.2.2 Numerical Tests . . . . .	28

4.3	2D Time-Dependent Flow Field . . . . .	30
4.3.1	Numerical Implementation . . . . .	31
4.3.2	DNS Solution and Smagorinsky Model . . . . .	32
4.3.3	Lagrangian Particle Method . . . . .	33
4.3.4	Random Initial Field . . . . .	42
4.3.5	Use of the Unfiltered Velocity Field . . . . .	43
<b>5</b>	<b>3D Flow Example</b>	<b>48</b>
5.1	3D Velocity Field . . . . .	48
5.2	Numerical Results . . . . .	51
5.2.1	Gaussian Initial Field . . . . .	51
5.2.2	Random Initial Field . . . . .	55
<b>6</b>	<b>Forced Turbulence</b>	<b>59</b>
6.1	Forced Isotropic Turbulence . . . . .	59
6.2	Numerics . . . . .	60
6.3	Scalar Mixing . . . . .	61
<b>7</b>	<b>Conclusions/Summary</b>	<b>69</b>
	<b>Bibliography</b>	<b>72</b>
<b>A</b>	<b>Derivations</b>	<b>78</b>
A.1	Subgrid Model . . . . .	78
A.2	Root Mean Square . . . . .	79
A.3	Anisotropic Gaussian Particles . . . . .	81
A.4	Remeshing . . . . .	84
<b>B</b>	<b>Hermite Polynomials</b>	<b>87</b>

# Chapter 1 Introduction

The scalar advection-diffusion or transport equation describes the motion of a passive scalar (one that has no influence on the velocity field) under the forces of advection and diffusion. Examples include temperature in an incompressible flow field, a dye in a fluid flow for visualisation purposes, and the spreading of trace elements in the atmosphere.

For large values of the Péclet number, defined as  $Pe = \frac{UL}{\kappa}$ , where  $U$  and  $L$  are characteristic velocity and length scales and  $\kappa$  is the diffusivity, one needs a fine computational grid to resolve both the large and small length scales. Our research is interested in computations for large Péclet numbers or large Schmidt numbers, defined as  $Sc = \frac{\nu}{\kappa}$ , for which the smallest length scale in the scalar field  $\eta_\psi$  scales as  $\eta_\psi = \eta \left(\frac{\kappa}{\nu}\right)^{1/2}$ , where  $\eta$  is the Kolmogorov length scale, the smallest scale in the velocity field. See for example Tennekes and Lumley [46].

It can be shown that the Kolmogorov length scales as  $\frac{\eta}{L} = Re^{-3/4}$  [25] and thus we find for the smallest scale in the scalar field  $\frac{\eta_\psi}{L} = Sc^{-1/2} Re^{-3/4}$ . In order to resolve this scale, we need at least  $Sc^{1/2} Re^{3/4}$  grid points in each spatial direction, which in 3D leads to  $Sc^{3/2} Re^{9/4}$  for the total number of points. For a modest Reynolds number of 1000 and a Schmidt number of 100, this leads to 6 billion grid points. With the currently available computational power, these computations are just not feasible for direct numerical simulation (DNS).

In the absence of viscosity, we can give a similar argument. Consider an interface, where the scalar quantity is initially zero on one side and unity on the other side. We will assume the interface is being stretched by a strain rate  $S$  and denote the thickness of the interface by  $\delta$ . This thickness  $\delta$  will be proportional to  $\sqrt{\frac{\kappa}{S}}$ . We also assume that the strain rate  $S$  scales as  $\frac{U}{L}$ , where  $U$  and  $L$  are characteristic velocity and length scales, respectively. If we introduce the Péclet number  $Pe = \frac{UL}{\kappa}$ , we

find that  $\frac{\delta}{L}$  scales as  $\frac{1}{\sqrt{Pe}}$  and so the number of gridpoints needed in each spatial direction scales as  $\sqrt{Pe}$ .

## 1.1 Large Eddy Simulation and Subgrid Modeling

There are several options available to reduce the necessary computational time by finding an approximation to the DNS solution. Two of the most common methods are the Reynolds averaged equations [14] and large-eddy simulation (LES) [28, 31].

Reynolds averaging splits all variables in a time or ensemble average value and a perturbation, followed by averaging the equations of motion. Since the average of a perturbed quantity is zero, all perturbed quantities drop out except in the non-linear terms. One needs to add an equation relating the remaining perturbed quantities in terms of averaged variables. The application of this method to the Navier-Stokes equations is known as Reynolds Averaged Navier-Stokes equations (RANS) and the remaining term in perturbed quantities is the Reynolds stress. The eddy-viscosity model is a widely used choice to close the system.

LES applies a filtering operator to the equation of motions thereby removing or strongly reducing the high wave number components resulting in an equation for the large scale structures. Similar to the Reynolds averaged equations, the filtering operator introduces an extra variable. A model needs to be added relating the extra variable to the other variables for closure. This extra equation is called the subgrid model, as it describes the effect of the small scales on the large scale structure. An example is the widely used Smagorinsky subgrid model [44]. This research has used the LES technique.

In LES, it is common to calculate statistical quantities for the flow field and compare these with the results obtained using DNS or experiments. It is expected that the actual flow fields computed with different LES methods or by DNS will deviate in time due to turbulent and chaotic effects, but that statistical quantities will be well predicted by the LES computation. This research will not only look at the variance or root mean square of the scalar field for comparison reasons, but as

well to the differences in the scalar fields itself. It is expected that when the actual scalar fields are close, statistical quantities derived from these fields are close as well. In this view, comparing scalar fields using LES with filtered DNS results is a more stringent test and is a reasonable approach at least for short to intermediate times.

This research will test the relatively new tensor-diffusivity subgrid model. Bedford and Yeo [5, 51] and Leonard [29] showed that for a Gaussian filter the filtered product of two variables can be expressed in terms of an infinite sum of the filtered variables. By applying the Gaussian filter to the scalar advection-diffusion equation and retaining only the first two terms in the infinite sum, the tensor-diffusivity subgrid model is obtained.

The tensor-diffusivity model can be classified as a so-called deconvolution-type model, which finds an approximation for the filtered or convoluted products by unfiltering or deconvolution. Leonard [27] was the first to use a deconvolution model by using an expansion in Taylor polynomials for the filtered non-linear term. Bardina et al. [3] used the assumption of scale-similarity and Shah and Ferziger [43] and Geurts [16] used a polynomial expansion of the filter kernel. The deconvolution of the resolved scales directly was used by Domaradzki and Saiki [13]. Finally, Stolz and Adams [45] called their method the approximate deconvolution technique, which is based on a truncated series expansion of the inverse-filter kernel.

The tensor-diffusivity subgrid model is parameter free and depends only on the size of the Gaussian filter. It can be interpreted as a tensor-diffusive term proportional to the rate-of-strain tensor of the filtered velocity field. Carati [9] showed that other filters than a Gaussian, such as the top-hat filter and all discrete filters, lead as well to the tensor-diffusivity model up to a multiplicative constant.

The tensor-diffusivity subgrid model is material frame independent and allows for backscatter. In the stretching directions, the model operates as a negative diffusion, which makes the equation ill-conditioned. For this reason, using a straightforward finite difference method or spectral method will lead to instabilities. Mathematical models of physical processes can lead to ill-posedness; see for example the discussion in Barenblatt et al. [4] or Krasny [24]. To obtain a well-posed problem, some form of

regularization is required taking into account that the results can depend strongly on the method chosen. Leonard and Winckelmans [30] were able to obtain good results for the momentum equation by adding an extra dynamic eddy-viscosity term to the tensor-diffusivity model.

## 1.2 Numerical Implementation

Our work will regularize the problem by decomposing the scalar field in a collection of Lagrangian particles, each of them well behaved for large wave numbers. Next to using (axisymmetric) Gaussian particles, which are used widely in vortex methods [28], anisotropic Gaussian particles are introduced. The location and the size of the axisymmetric particles are functions of time. Anisotropic particles are characterized by a positive definite symmetric matrix giving the shape (ellipticity and orientation) of a particle. This shape matrix and the location are assumed to be functions of time.

To numerically account for the diffusivity term, different methods have been used in the past. For example, in the random-walk method, a random displacement is added to the motion of each particle [10]. The core spreading method increases the size of the Gaussian particles over time and solves the diffusive part exactly [28]. However, Greengard [18] showed that the core spreading method approximates the wrong equation in the limit of an infinite number of particles.

Rossi [41, 42] revamped the core spreading method recently by introducing a splitting and merging scheme for axisymmetric Gaussian particles. He showed that by using a splitting scheme, the size of the particles could be controlled and he proved convergence to the vorticity equation. The merging scheme was used to keep the total number of particles reasonably small. This work will also use the core expansion method.

Instead of splitting and merging particles, it was chosen to control the core size of the particles by replacing the old set with a complete new set with each particle again with a small core size and axisymmetric shape. A novel remeshing procedure will be discussed resulting in explicit expressions for the amplitudes of the particles

of the new set.

### **1.3 Overview**

This thesis is composed as follows. Chapter 2 discusses the tensor-diffusivity subgrid model in detail. The model is derived and it is shown that it allows for backscatter and is material frame independent. Chapter 3 discusses the numerical particle method. It will be demonstrated that a finite-difference code yields instabilities. Anisotropic and axisymmetric Gaussian particles are introduced and the equations of motion are derived. The remeshing scheme is presented. Chapter 4 discusses the 2D results. It starts with a simple rotating flow and a stagnation flow followed by an indepth discussion of a 2D time-dependent periodic velocity field that yields chaotic advection. Comparisons with DNS data and the Smagorinsky subgrid model are made. Chapter 5 discusses the results for a 3D stationary velocity field and makes a comparison with literature. Chapter 6 deals with forced isotropic 3D turbulence in a periodic domain. Finally, chapter 7 gives a short summary and lists the conclusions of this research.

## Chapter 2 Tensor-Diffusivity Subgrid Model

This chapter derives the tensor-diffusivity subgrid model for the scalar advection-diffusion equation using a Gaussian filtering operator. An infinite series expansion is found for the extra unknown variable for closure. The tensor-diffusivity subgrid model is obtained by retaining only the first two terms in this expansion. Several characteristics of the model will be discussed.

### 2.1 Filtering

The transport of a passive scalar quantity  $\psi(\mathbf{x}, t)$  in an incompressible velocity field  $\mathbf{u}(\mathbf{x}, t)$  is governed by the scalar advection-diffusion or transport equation,

$$\frac{\partial \psi}{\partial t} + \mathbf{u} \cdot \nabla \psi = \kappa \nabla^2 \psi, \quad (2.1)$$

where  $\kappa$  is the scalar diffusivity constant. Examples of scalar transport include the visualization of fluid flows with a dye or the spreading of small concentrations of toxic elements in the atmosphere. For large Péclet numbers,  $Pe = \frac{UL}{\kappa}$ , a large computational effort is required to compute both the large and small scale structure. To reduce the computational effort, one can resolve only the large scale structures and model the effect of the small scale structures on these large scale structures. This procedure is known as large eddy simulation (LES). LES applies a filtering operation to the equations of motion thereby removing (or smoothing out) the smaller scales (high wave number components) at the expense of introducing an extra unknown vector (variable). In order to close the resulting equation, an additional relation between the extra unknown vector and the other variables/vectors needs to be established. These relations or models are called subgrid models.

Define a filtering operator  $\mathcal{L}$  operating on an arbitrary function  $g(\mathbf{x})$  as the con-

volution integral between a smooth filtering function  $F(\mathbf{x})$  and  $g(\mathbf{x})$ , i.e.,

$$\widehat{g}(\mathbf{x}) \equiv \mathcal{L}\{g(\mathbf{x})\} = \int_{\Omega} F(\mathbf{x} - \mathbf{x}')g(\mathbf{x}')d\mathbf{x}', \quad (2.2)$$

where a hat denotes a filtered quantity and the integral is taken over the entire spatial domain  $\Omega$ .

Apply the filtering operator  $\mathcal{L}$  to (2.1). Since the filtering operator and differentiation with respect to time and space commute, we find using the incompressibility of the flow,

$$\frac{\partial \widehat{\psi}}{\partial t} + \nabla \cdot \widehat{\mathbf{u}}\widehat{\psi} = \kappa \nabla^2 \widehat{\psi}. \quad (2.3)$$

In order to solve this equation, a relation between the extra variable  $\widehat{\mathbf{u}}\widehat{\psi}$  and the other variables needs to be added, the so-called closure relation or subgrid model.

In this study, we will restrict our attention to a Gaussian filter with a characteristic length scale  $\sigma$ ,

$$F(\mathbf{x}) = \frac{1}{(\sigma\sqrt{\pi})^d} \exp\left(-\frac{|\mathbf{x}|^2}{\sigma^2}\right), \quad (2.4)$$

where  $d$  indicates the spatial dimension of the filter (2 or 3). The effect of filtering on the flow field is often best represented in Fourier space. As the Fourier transform of a Gaussian is a Gaussian, this filter effectively reduces the amplitude of the high wave number components of the field.

## 2.2 Tensor-Diffusivity Subgrid Model

Assume that both the unknown scalar function  $\psi(\mathbf{x}, t)$  and the velocity field  $\mathbf{u}(\mathbf{x}, t)$  are continuous differentiable. The unknown vector  $\widehat{\mathbf{u}}\widehat{\psi}$  can then be expressed for a Gaussian filter as

$$\widehat{\mathbf{u}}\widehat{\psi}(\mathbf{x}, t) = \sum_{n=0}^{\infty} \frac{1}{n!} \left(\frac{\sigma^2}{2}\right)^n \frac{\partial^n \widehat{\mathbf{u}}}{\partial x_{i_1} \partial x_{i_2} \dots \partial x_{i_n}} \frac{\partial^n \widehat{\psi}}{\partial x_{i_1} \partial x_{i_2} \dots \partial x_{i_n}}, \quad (2.5)$$

where a summation over repeated indices is implied. This equation was first established by Bedford and Yeo [5, 51] using Fourier analysis, and they refer to it as the YB-III series. They also showed how to extend (2.5) to filtered functions of more than two quantities. Later Leonard [29] obtained independently the same result. His derivation is reproduced in appendix A.1.

Next take the divergence of (2.5) and use the incompressibility of the flow field to get

$$\begin{aligned} \nabla \cdot \widehat{\mathbf{u}}\widehat{\psi}(\mathbf{x}, t) &= \sum_{n=0}^{\infty} \frac{1}{n!} \left( \frac{\sigma^2}{2} \right)^n \frac{\partial^n \widehat{\mathbf{u}}}{\partial x_{i_1} \partial x_{i_2} \dots \partial x_{i_n}} \cdot \frac{\partial^n \nabla \widehat{\psi}}{\partial x_{i_1} \partial x_{i_2} \dots \partial x_{i_n}} \\ &= \widehat{\mathbf{u}} \cdot \nabla \widehat{\psi} + \frac{\sigma^2}{2} \frac{\partial \widehat{u}_i}{\partial x_j} \frac{\partial^2 \widehat{\psi}}{\partial x_i \partial x_j} + \frac{\sigma^4}{8} \frac{\partial^2 \widehat{u}_i}{\partial x_j \partial x_k} \frac{\partial^3 \widehat{\psi}}{\partial x_i \partial x_j \partial x_k} + \dots \end{aligned} \quad (2.6)$$

This equation closes (2.3) and is still exact. From now on, we will retain only the first two terms in (2.6) assuming that the higher order terms are sufficiently small to be neglected. Using the symmetry of  $\frac{\partial^2 \widehat{\psi}}{\partial x_i \partial x_j}$  in  $x_i$ - $x_j$  and introducing the strain rate tensor  $S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$  will transform (2.3) in

$$\frac{\partial \widehat{\psi}}{\partial t} + \widehat{\mathbf{u}} \cdot \nabla \widehat{\psi} = \kappa \nabla^2 \widehat{\psi} - \frac{\sigma^2}{2} \widehat{S}_{ij} \frac{\partial^2 \widehat{\psi}}{\partial x_i \partial x_j}. \quad (2.7)$$

The second term on the right-hand side is the subgrid model. Note that if we set  $\sigma = 0$  (no filtering), we will obtain (2.1) again.

A second order derivative is indicative of a diffusive process. We can therefore interpret the extra term as an added effective diffusivity  $-\frac{\sigma^2}{2} \widehat{S}_{ij}$ , which is a function of the spatial direction. Introduce locally the principal coordinate system of  $\widehat{\mathbf{S}}$ . Call these directions  $x_1$ ,  $x_2$ , and  $x_3$ , corresponding to the (real) eigenvalues  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$ , respectively, where we choose  $\lambda_1 \geq \lambda_2 \geq \lambda_3$ . Since  $\text{trace}(\widehat{\mathbf{S}}) = \lambda_1 + \lambda_2 + \lambda_3 = \nabla \cdot \widehat{\mathbf{u}} = 0$  due to incompressibility, at least one of the eigenvalues ( $\lambda_1$ ) of  $\widehat{\mathbf{S}}$  is greater than zero. In this direction, the total diffusivity can be said to be  $-\frac{\sigma^2}{2} \lambda_1 + \kappa$ . Since  $\lambda_1 > 0$ , the added diffusivity due to the subgrid model is negative and the total diffusivity

in the  $x_1$  direction becomes effectively negative for  $\lambda_1 > \frac{2\kappa}{\sigma^2}$ . This negative diffusion constitutes a numerical problem and possibly also a theoretical problem and calls for regularization as will be discussed later. Since we can regard the subgrid model as an anisotropic diffusivity term, the model will be referred to as the tensor-diffusivity model.

It was shown by Carati et al. [9] that the use of spatial filters other than Gaussian results in doubly infinite expansions like (2.5). They showed that the leading two terms in such an expansion are the same up to 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.

To give some insight into the effect of the extra term, 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,  $\widehat{\psi}(\mathbf{x}, 0) = e^{ikx_1}$ , where  $k$  is the wave number. The filtered advection-diffusion equation (2.7) simplifies to

$$\frac{\partial \widehat{\psi}}{\partial t} + \lambda_1 x_1 \frac{\partial \widehat{\psi}}{\partial x_1} = -\frac{\sigma^2}{2} \lambda_1 \frac{\partial^2 \widehat{\psi}}{\partial x_1^2}. \quad (2.8)$$

It is straightforward to verify that  $\widehat{\psi} = e^{ik \exp(-\lambda_1 t) x_1} \exp[\sigma^2 k^2 (1 - \exp(-2\lambda_1 t))/4]$  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  $k \rightarrow \infty$ . The addition of molecular diffusion counteracts this behavior and avoids it only if  $\kappa > \frac{\sigma^2}{2} \lambda_1$ .

Define the root mean square value (or variance) of the scalar field  $\widehat{\psi}(t)$  as

$$\widetilde{\psi}(t) \equiv \sqrt{\int_{\Omega} (\widehat{\psi}(\mathbf{x}, t) - \widehat{\psi}_c)^2 d\mathbf{x}}, \quad (2.9)$$

where the integration is taken over the entire spatial domain  $\Omega$  and  $\widehat{\psi}_c$  is the average value of the scalar  $\widehat{\psi}$  over the spatial domain. The root mean square value is a measure

for how well the scalar is mixed. Appendix A.2 shows that we can write for the time derivative of  $\tilde{\psi}^2$

$$\frac{d\tilde{\psi}^2}{dt} = -2\kappa \int_{\Omega} \left( \nabla \hat{\psi} \right)^2 d\mathbf{x} + \sigma^2 \int_{\Omega} \hat{S}_{ij} \frac{\partial \hat{\psi}}{\partial x_i} \frac{\partial \hat{\psi}}{\partial x_j} d\mathbf{x}. \quad (2.10)$$

In absence of the subgrid model ( $\sigma = 0$ ),  $\tilde{\psi}$  is a decaying function of time. The term due to the subgrid model can be positive or negative, which can be shown by locally using the principal coordinate system of  $\mathbf{S}$ . This implies that the subgrid model will allow for backscatter.

### 2.3 Material Frame Indifference

Nature is unaware of the frame of reference an observer uses. Different observers in different frames of reference will record the same physical phenomena. In mathematical terms this results in certain transformation properties for the governing equations between different frames of reference. If the equations have these transformation properties, the equations are said to be material frame indifferent. The Navier-Stokes equation and the scalar advection-diffusion equation are examples of this category; see for example Fureby [15] or Gurtin [19]. Subgrid scale models also need to satisfy these transformations. It was shown by Fureby [15] that rotational symmetry for a filtering operator, among these the Gaussian filter, is a necessary condition to preserve the material frame indifference.

In this section, we will show that the tensor-diffusivity subgrid model applied to the scalar advection-diffusion equation is material frame indifferent. Start out with two observers, one using the  $\mathbf{x}$  coordinate system and the other one, the  $\mathbf{x}^*$  system. The motion in both frames of reference are related by the transformation

$$x_i^* = Q_{ij}(t)x_j + c_i(t), \quad (2.11a)$$

$$t^* = t, \quad (2.11b)$$

where  $\mathbf{Q}$  is a rotation matrix with its inverse equal to its transpose and its determinant equal to unity. The vector  $\mathbf{c}$  indicates the distance between the origin of both systems. Both the rotation matrix  $\mathbf{Q}$  and the displacement vector  $\mathbf{c}$  are assumed to be functions of time. The velocity in the starred system can be written after differentiating (2.11a)

$$u_i^* = Q_{ij}(t)u_j + \dot{Q}_{ij}(t)x_j + \dot{c}_i(t). \quad (2.12)$$

Using the chain rule, one can show for any function  $f$ ,

$$\frac{\partial f}{\partial x_i^*} = Q_{ji}^{-1} \frac{\partial f}{\partial x_j} = Q_{ji}^T \frac{\partial f}{\partial x_j}. \quad (2.13)$$

The tensor-diffusivity term in the starred coordinate system is related to the unstarred system by

$$-\frac{\sigma^2}{2} \frac{\partial u_i^*}{\partial x_j^*} \frac{\partial^2 \psi}{\partial x_i^* \partial x_j^*} = -\frac{\sigma^2}{2} Q_{kj}^{-1} \frac{\partial}{\partial x_k} \left( Q_{il} u_l + \dot{Q}_{il} x_l \dot{c}_i \right) Q_{mi}^{-1} \frac{\partial}{\partial x_m} Q_{nj}^{-1} \frac{\partial \psi}{\partial x_n}, \quad (2.14)$$

which can be reduced to

$$-\frac{\sigma^2}{2} \frac{\partial u_i^*}{\partial x_j^*} \frac{\partial^2 \psi}{\partial x_i^* \partial x_j^*} = -\frac{\sigma^2}{2} \frac{\partial u_i}{\partial x_j} \frac{\partial^2 \psi}{\partial x_i \partial x_j} - \frac{\sigma^2}{2} Q_{mi}^{-1} \dot{Q}_{in} \frac{\partial^2 \psi}{\partial x_m \partial x_n}. \quad (2.15)$$

We can show that the last term on the right-hand side is zero, since

$$Q_{mi}^{-1} \dot{Q}_{in} \frac{\partial^2 \psi}{\partial x_m \partial x_n} = \frac{Q_{mi}^T \dot{Q}_{in} + \dot{Q}_{mi}^T Q_{in}}{2} \frac{\partial^2 \psi}{\partial x_m \partial x_n} = \frac{d}{dt} \left( \frac{Q_{mi}^T Q_{in}}{2} \right) \frac{\partial^2 \psi}{\partial x_m \partial x_n} = 0. \quad (2.16)$$

Hence we conclude that the tensor-diffusivity term is material frame indifferent.

## 2.4 Smagorinsky Subgrid Model

In order to assess the applicability of the tensor-diffusivity subgrid model, we will compare it numerically with the well-known and widely used Smagorinsky subgrid model, first proposed by Smagorinsky in 1963 to model the unresolved scales in large

scale atmospheric computations [31, 44]. He introduced an extra eddy-viscosity term for the momentum equation.

For the filtered advection-diffusion equation, a similar approach can be used by introducing an eddy diffusivity  $\kappa_\tau$ ,

$$\frac{\partial \widehat{\psi}}{\partial t} + \widehat{\mathbf{u}} \cdot \nabla \widehat{\psi} = \kappa \nabla^2 \widehat{\psi} + \nabla \cdot \kappa_\tau \nabla \widehat{\psi}, \quad (2.17)$$

where  $\kappa_\tau = (C\Delta x)^2 \sqrt{2\widehat{S}_{ij}\widehat{S}_{ij}}$ .  $\Delta x$  is a measure for the grid size and  $C$  is a constant, which is normally chosen between 0.1 and 0.2.

## Chapter 3 Lagrangian Particle Method

Results using a finite difference scheme are presented as an illustration of the problems associated with the negative diffusion of the tensor-diffusivity subgrid model. This is followed by a discussion of the characteristics of the particle method which we propose to regularize and avoid this problem. Anisotropic and axisymmetric Gaussian particles are introduced and their equations of motion are derived. A novel remeshing scheme is discussed.

### 3.1 Finite Difference Method

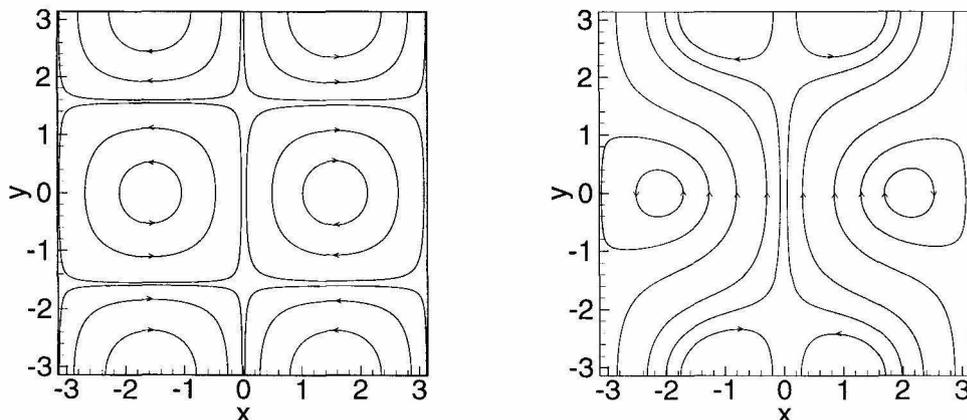


FIGURE 3.1: Streamline pattern of (3.1) for  $\epsilon_x = 0$ ,  $\epsilon_y = 0.5$  at  $t = 0$  (left) and  $\omega_y t = \pi/2$  (right).

A second-order central difference scheme has been implemented to solve (2.7) numerically. The incompressible velocity field,

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \sin(x) \sin(y) + \epsilon_x \cos(\omega_x t) \\ \cos(x) \cos(y) + \epsilon_y \sin(\omega_y t) \end{pmatrix}, \quad (3.1)$$

has been used, where  $\epsilon_x$  and  $\epsilon_y$  indicate amplitudes of a sinusoidal perturbation and  $\omega_x$  and  $\omega_y$  their respective frequencies. We will use  $\epsilon_x = 0.0$ ,  $\epsilon_y = 0.5$ , and  $\omega_y = 1.0$  in this section. Figure 3.1 shows the streamline pattern at  $t = 0$  and  $\omega_y t = \frac{\pi}{2}$ .

The diffusivity constant has been set equal to  $\kappa = 0.001$  and the filtering constant is chosen to be  $\sigma = 0.15$ . As an initial condition for the unfiltered scalar field, the Gaussian,  $\exp\left(-\frac{(x - 0.3)^2 + (y + 0.4)^2}{\tau^2}\right)$ , has been used, where  $\tau = 0.5$ . The computational initial condition is thus the given Gaussian filtered using (2.4) with  $\sigma = 0.15$ . An equidistant grid between  $-2\pi$  and  $2\pi$  has been set up with 401 grid points in each spatial direction (gridspacing 0.03). The fourth order Runge-Kutta method as discussed in Ferziger and Perić [14] was used to integrate in time with timestep  $dt = 0.005$ .

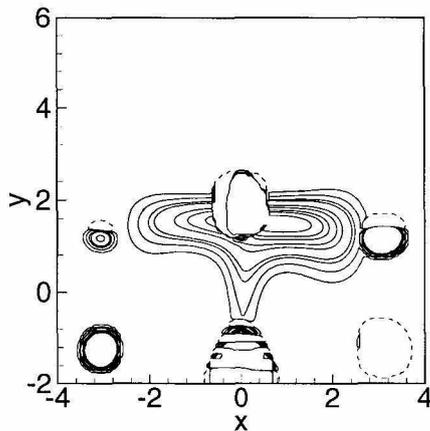


FIGURE 3.2: Contour plot using the tensor-diffusivity model in a finite difference calculation at  $t = 2$ . Dashed contour is 0 and the others are 0.0001, 0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.3, and 0.5.

Contourlines of the solution at  $t = 2$  are shown in figure 3.2. Instabilities started to appear at the locations where the effects of negative diffusion are largest. Outside these areas, the solution is still good as will be shown in the next chapter. Using different grid sizes and/or time steps might shift the time of onset of the instabilities, but will not prevent them from occurring if one uses the above numerical scheme.

There are different ways to regularize the problem using a finite difference approach. For example, Leonard and Winckelmans [30] and Winckelmans et al. [50]

were able to obtain good results using a mixed model for the momentum equation by using the tensor-diffusivity model supplemented by an eddy-viscosity model. We have also performed computations using a mixed model, using the tensor-diffusivity model supplemented by a Smagorinsky term. This model was able to smooth out the instabilities slightly, but could not prevent them from occurring. By using larger values of the diffusivity constant  $\kappa$ , e.g., 0.01, the finite difference code gave good results. The negative diffusion was offset completely by the large diffusivity in these cases. However, this study wants to focus on small values of  $\kappa$ . Another way to regularize a finite difference method is by choosing a different spatial or time discretization scheme which is stable even for the negative eigenvalues. This option has not been investigated.

Instead of using a finite difference method, Leonard [26] tried to use a spectral method and his results lead to similar instabilities.

## 3.2 Particle Method

The instabilities in a finite difference calculation are due to the negative diffusion of the subgrid model. The model equation (2.7) we want to solve is ill-posed. On occasion mathematical models of physical processes can lead to these type of equations; see for example Barenblatt et al. [4] or Krasny [24]. We need some form of regularization 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 (2.1), information about the high wave number components is lost. In order to have the solution with the regularization approximate the solution of (2.7) closely, we need a regularization that maintains control over the entire wave number spectrum. We will regularize the problem by decomposing the scalar field in a collection of Lagrangian particles, each of them well-behaved for large wave numbers.

Assume that we can approximate the scalar field  $\widehat{\psi}(\mathbf{x}, t)$  as a sum of  $N$  anisotropic

Gaussian particles given by

$$\widehat{\psi}(\mathbf{x}, t) = \sum_{k=1}^N \frac{a_k \sqrt{\det(\mathbf{M}_k)}}{(\sqrt{\pi} \delta_k)^d} \exp\left(-\frac{(\mathbf{x} - \mathbf{x}_k)^T \mathbf{M}_k (\mathbf{x} - \mathbf{x}_k)}{\delta_k^2}\right) = \sum_{k=1}^N \widehat{\psi}_k(\mathbf{x}, t), \quad (3.2)$$

where  $d$  is the spatial dimension of the problem and  $\mathbf{M}_k$  is a  $d \times d$  symmetric time-dependent positive definite matrix with elements  $m_{ij}$ . A measure for the size of each particle is given by the constant  $\delta_k$ . Each particle  $k$  is centered at  $\mathbf{x}_k = \mathbf{x}_k(t)$ . Integration of each particle in space results in  $a_k$ , the amplitude of a particle. A schematic sketch of three different anisotropic particles is given in figure 3.3. By setting  $\mathbf{M}_k$  equal to the identity matrix, we obtain the well-known and widely used axisymmetric Gaussian particles, which will be discussed later.

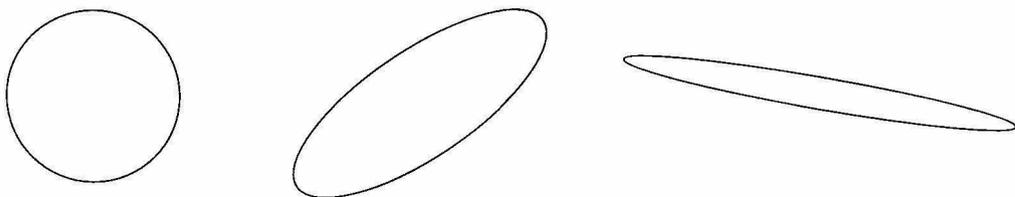


FIGURE 3.3: Schematic sketch of anisotropic particles.

The Fourier transform  $\mathcal{F}$  of (3.2) is given by

$$\mathcal{F}\{\widehat{\psi}\}(\mathbf{k}, t) = \sum_{k=1}^N a_k \exp\left(-\frac{\delta_k^2 \mathbf{k}^T \mathbf{M}_k^{-1} \mathbf{k}}{4}\right) e^{i\mathbf{k} \cdot \mathbf{x}_k}, \quad (3.3)$$

where  $\mathbf{k}$  is the wave number. This function appears to be well-behaved for large wave numbers as long as  $\mathbf{M}_k$  remains positive definite, giving us the desired control over all wave numbers. There is another argument for choosing Gaussian basic functions. Consider (2.1) for  $\mathbf{u} = 0$ , which reduces the equation to the ordinary heat equation  $\frac{\partial \psi}{\partial t} = \kappa \nabla^2 \psi$  having as solution (in two dimensions) the Gaussian  $\psi(\mathbf{x}, t) = \frac{1}{4\pi\kappa t} \exp\left(-\frac{\mathbf{x}^2}{4\kappa t}\right)$ .

To numerically account for the molecular diffusion of a scalar or for the diffusion of vorticity due to viscous effects, different methods have been used in the past. There is, for example, the random-walk method, which adds a random variable to

the motion of each particle. The core spreading method increases the size of the Gaussian particles over time to simulate the effect of diffusion. More than a decade ago, the core spreading method was used extensively, as it solved the diffusive part of the equation exactly [29]. However, Greengard [18] showed that the core spreading method approximates the wrong equation in the limit of an infinite number of particles. Greengard considered the two-dimensional vorticity equation and approximated the vorticity field by (axisymmetric) Gaussian particles. He showed that the time-derivative of the modified vorticity equation does not equal the time derivative of the vorticity equation and hence concluded that the core spreading method does not approximate the vorticity equation. The error is due to the convective part in the vorticity equation, since the vorticity is effectively not convected by the local velocity field, but by an averaged velocity depending on the size of a particle. Greengard's conclusion does not only apply to the vorticity equation, but as well to the scalar advection-diffusion equation.

Recently, Rossi [41, 42] revamped the use of the core spreading method for the viscous vorticity equation by introducing a splitting and merging scheme for axisymmetric Gaussian particles. He showed that by splitting the particles and thus controlling the core size of the particles, convergence to the vorticity equation is obtained. The merging scheme he employed was only used to keep the total number of particles reasonable. Our method will make use of the core expansion method.

### 3.3 Anisotropic Gaussian Particles

Assume that we can decompose the scalar field as in (3.2). We substitute this decomposition in the filtered advection-diffusion equation (2.7) and expand each term in a series of Hermite polynomials. Hermite polynomials are used since they are in a natural way connected to Gaussian functions (see appendix B). Setting the coefficients of the lowest order Hermite polynomials equal to zero leads to the equations of motion for the location ( $\mathbf{x}_k$ ) and the shape ( $\mathbf{M}_k$ ) of each particle. An extensive derivation in two dimensions can be found in appendix A.3.

For the time derivative of the location of the particles, we obtain

$$\frac{d\mathbf{x}_k}{dt} = \overline{\hat{\mathbf{u}}}^k - \frac{1}{2}\sigma^2\overline{\nabla^2\hat{\mathbf{u}}}^k, \quad (3.4)$$

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

$$\overline{f}^k(\mathbf{x}_k, t) = \frac{\sqrt{\det(\mathbf{M}_k)}}{(\sqrt{\pi}\delta_k)^d} \int_{\Omega} f(\mathbf{x}, t) \exp\left(-\frac{(\mathbf{x} - \mathbf{x}_k)^T \mathbf{M}_k (\mathbf{x} - \mathbf{x}_k)}{\delta_k^2}\right) d\mathbf{x}, \quad (3.5)$$

which corresponds to the lowest order coefficient of the expansion of  $f(\mathbf{x}, t)$  in Hermite polynomials. The integration is taken over the entire spatial domain  $\Omega$ . The overline can be interpreted as a weighted average over the anisotropic Gaussian particle  $k$ . Since this averaged value depends on  $k$ , two particles that have the same location  $\mathbf{x}_k$ , but different shapes  $\mathbf{M}_k$ , can move in different directions or at different velocities.

If we consider the next order in the expansion, we find the following expression for the time derivative of the matrix  $\mathbf{M}_k$ ,

$$\begin{aligned} \frac{d\mathbf{M}_k}{dt} = & -\overline{\nabla\hat{\mathbf{u}}}^k \mathbf{M}_k - \mathbf{M}_k \overline{\nabla\hat{\mathbf{u}}}^{kT} - \frac{4\kappa}{\delta_k^2} \mathbf{M}_k \mathbf{M}_k + \frac{\sigma^2}{\delta_k^2} \mathbf{M}_k \left( \overline{\nabla\hat{\mathbf{u}}}^k + \overline{\nabla\hat{\mathbf{u}}}^{kT} \right) \mathbf{M}_k \\ & + \frac{\sigma^2}{2} \left( \overline{\nabla\nabla^2\hat{\mathbf{u}}}^k \mathbf{M}_k + \mathbf{M}_k \overline{\nabla\nabla^2\hat{\mathbf{u}}}^{kT} \right), \end{aligned} \quad (3.6)$$

where use has been made of the symmetry of  $\mathbf{M}_k$ . A capital  $T$  superscript stands for transpose.

For the following analysis, we will neglect the last term on the right-hand side of (3.6) assuming it is small. We will also assume that there is no diffusivity acting ( $\kappa = 0$ ) and that at  $t = 0$   $\mathbf{M}_k$  is equal to the identity matrix. It is then straightforward to show by substitution in (3.6) and making use of the symmetry of  $\mathbf{M}_k$  that the solution is given by

$$\mathbf{M}_k = \frac{\delta_k^2}{\sigma^2} \left[ \mathbf{I} + \left( \frac{\delta_k^2}{\sigma^2} - 1 \right) \mathbf{G}_k \right]^{-1}, \quad (3.7)$$

where the superscript -1 denotes a matrix inverse and the left Cauchy-Green strain tensor  $\mathbf{G}_k$  satisfies  $\frac{d\mathbf{G}_k}{dt} = \mathbf{G}_k \overline{\nabla\hat{\mathbf{u}}}^k + \overline{\nabla\hat{\mathbf{u}}}^{kT} \mathbf{G}_k$ , with initial condition  $\mathbf{G}_k(0) = \mathbf{I}$ . The left Cauchy-Green strain tensor can be written as  $\mathbf{G}_k = \mathbf{F}_k \mathbf{F}_k^T$ , where  $\mathbf{F}_k$  is the

deformation tensor satisfying  $\frac{d\mathbf{F}_k}{dt} = \widehat{\nabla \mathbf{u}}^{kT} \mathbf{F}_k$  (which is consistent with the equation for the time-derivative of  $\mathbf{G}_k$ ).

Since  $\mathbf{G}_k$  is a non-singular positive definite matrix, we see from (3.7) that  $\mathbf{M}_k$  is as well a non-singular positive definite matrix as long as  $\delta_k > \sigma$ . The finest scale in the unfiltered advection-diffusion equation can be said to be the Dirac delta function. After applying the filtering operator (2.4), this delta function is transformed into a Gaussian with characteristic size  $\sigma$ . The constraint that  $\delta_k$  has to be larger than  $\sigma$  to keep  $\mathbf{M}_k$  positive definite is thus a reasonable one. The effect of the viscous term  $\kappa \mathbf{M}_k \mathbf{M}_k$  is to increase the size of the particles uniformly and will not effect the positive definiteness of  $\mathbf{M}_k$ . As long as  $\mathbf{M}_k$  remains positive definite, particle  $k$  will remain finite valued, giving the desired regularization and control over the high wave numbers. In the presence of the last term of (3.6), it can not be guaranteed that the shape matrix  $\mathbf{M}_k$  remains positive definite, but numerical computations showed that  $\mathbf{M}_k$  remained positive definite in all cases considered, provided that  $\delta_k > \sigma$ .

Instead of representing anisotropic Gaussian particles using a general quadratic form as in (3.2), Rossi [39] used geometrical variables in two dimensions to describe anisotropic particles and derived equations of motion for the vortex equation, which can be shown to be consistent with the equations given here.

In order to assess the benefits of using anisotropic Gaussian particles, we have also used axisymmetric Gaussian particles. By setting the shape matrix  $\mathbf{M}_k$  equal to the identity matrix in (3.2) and assuming that  $\delta_k = \delta_k(t)$ , we get

$$\widehat{\psi}(\mathbf{x}, t) = \sum_{k=1}^N \frac{a_k}{(\sqrt{\pi} \delta_k)^d} \exp \left( -\frac{(\mathbf{x} - \mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k)}{\delta_k^2} \right). \quad (3.8)$$

We can derive equations for the time evolution of  $\mathbf{x}_k$  and  $\delta_k$  by performing a similar analysis as for the anisotropic particles or we could simplify the equations obtained for anisotropic particles requiring that the shape matrix  $\mathbf{M}_k$  remains a diagonal matrix with identical elements on its diagonal. Both methods will lead to the same result. The equation for the location of the particles is again given by (3.4). If we consider the

next order coefficients, we end up with two incompatible equations for  $\frac{d\delta_k^2}{dt}$ . Assuming that both the core size  $\delta_k$  and the filtering constant  $\sigma$  are small, we can combine both equations to give

$$\frac{d\delta_k^2}{dt} = 4\kappa + \mathcal{O}(\delta_k^2, \sigma^2). \quad (3.9)$$

We see that for axisymmetric particles, even in the absence of the subgrid model, the core size  $\delta_k$  has to be small to obtain accurate numerical solutions. This problem of the core expansion method was already pointed out by Greengard [18]. However, Rossi [41, 42] showed that by employing a splitting operator on the particles and thus keeping the core size of the particles within arbitrary small bounds, the core expansion method can still be used. Note that the anisotropic particles do not need this requirement. However, for different reasons to be discussed below, they need to be split or remeshed as well.

In vortex methods, it is common to decompose the vorticity field in terms of axisymmetric Gaussian particles. If one wants to use the core expansion method, this decomposition is substituted in the vorticity equation and a similar expansion as given above will lead to the equations of motion. It is, however, common to use Taylor polynomials instead of Hermite polynomials. An expansion in Taylor polynomials will lead as well to (3.9) for the size of the particles, but the location of the particles is determined by the velocity of their respective center, i.e.,  $\frac{d\mathbf{x}_k}{dt} = \widehat{\mathbf{u}}(\mathbf{x}_k, t)$ . The expansion in Hermite polynomials led to an averaged velocity over the particle. Numerical experiments showed that using an averaged velocity leads to better results when compared to filtered DNS data.

It should be noted that the author also tried to derive equations of motion using a Taylor expansion instead of a Hermite expansion for both the axisymmetric and anisotropic particles. In the absence of the subgrid model ( $\sigma = 0$ ), both types of particles lead to their respective equations using a Hermite expansion, except that all variables are evaluated locally at the center of each particle instead of an averaged value over the particle. The tensor-diffusivity subgrid turns out to be a higher order effect for axisymmetric particles and thus the equations with and without the model

are identical. For the anisotropic particles, we were not able to find a consistent set of equations. It can be concluded that an expansion in Taylor polynomials is not appropriate in the presence of the subgrid model.

### 3.4 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 as shown by Greengard [18] by keeping the effective core size of the particles within limits, where the effective core size is defined by  $\frac{\delta_k}{\sqrt{\lambda_{k,1}}}$ , where  $\lambda_{k,1}$  is the largest eigenvalue of  $\mathbf{M}_k$ . Second, in order to get a smooth scalar field, the overlap parameter, defined as the ratio of the effective core size over the corresponding interparticle distance, has to be of order unity. Due to straining, particles tend to move apart from each other in some directions, thereby decreasing the overlap parameter making it necessary to remesh to maintain a smooth scalar field. Finally, anisotropic particles can become very elliptical due to straining, which increases the numerical errors.

Rossi [41, 42] used a splitting and merging scheme. This research has tested his scheme, but found that the results were not satisfactory. The pointwise difference before and after the splitting and merging scheme turned out to be quite large if one tried to keep the total number of particles reasonable. However, we found a better approach by replacing all the particles with a set of new particles.

Assume that at time  $t$ , we want to replace all the  $N$  old particles with a set of  $M$  new identical axisymmetric particles each with initial core size  $\tau$ . The new set of particles is spread out on a regular equidistant mesh with grid spacing  $h$  between neighboring particles in each direction. The location of the  $l$ -th new particle is denoted by  $\boldsymbol{\xi}_l$ . 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 field, i.e., we want to minimize

the following error function  $\epsilon(\mathbf{x})$ ,

$$\begin{aligned} \epsilon(\mathbf{x}) = & \sum_{k=1}^N \frac{a_k \sqrt{\det(\mathbf{M}_k)}}{(\sqrt{\pi} \delta_k)^d} \exp\left(-\frac{(\mathbf{x} - \mathbf{x}_k)^T \mathbf{M}_k (\mathbf{x} - \mathbf{x}_k)}{\delta_k^2}\right) \\ & - \sum_{l=1}^M \frac{b_l}{(\sqrt{\pi} \tau)^d} \exp\left(-\frac{(\mathbf{x} - \boldsymbol{\xi}_l)^2}{\tau^2}\right). \end{aligned} \quad (3.10)$$

By using an approximation based on a least square error method, which is given in detail in appendix A.4 for one spatial dimension, we were able to find the following explicit expressions for the unknown coefficients  $b_l$ 's (in one dimension),

$$b_l = \sum_{k=1}^N \frac{a_k \sqrt{\lambda_k}}{\sqrt{\pi} \sqrt{\delta_k^2 - \lambda_k \tau^2}} \exp\left(-\frac{\lambda_k (\xi_l - x_k)^2}{\delta_k^2 - \lambda_k \tau^2}\right) h, \quad (3.11)$$

where  $\lambda_k$  is the eigenvalue (and only element) of the shape matrix  $\mathbf{M}_k$  of particle  $k$ . In case of  $d$  spatial dimensions, the expressions for the new amplitudes are obtained by multiplying (3.11)  $d$  times in the principle coordinate system of the shape matrix  $\mathbf{M}_k$  for each particle  $k$  and transforming back to the original coordinate system. In two dimensions, we end up with

$$\begin{aligned} b_l = & \sum_{k=1}^N \frac{a_k \sqrt{\det(\mathbf{M}_k)}}{\pi \sqrt{(\delta_k^2 - \lambda_{k,1} \tau^2)(\delta_k^2 - \lambda_{k,2} \tau^2)}} \times \\ & \exp\left(-\frac{(\boldsymbol{\xi}_l - \mathbf{x}_k)^T (\delta_k^2 \mathbf{M}_k + \tau^2 \det(\mathbf{M}_k) \mathbf{I}) (\boldsymbol{\xi}_l - \mathbf{x}_k)}{(\delta_k^2 - \lambda_{k,1} \tau^2)(\delta_k^2 - \lambda_{k,2} \tau^2)}\right) h^2, \end{aligned} \quad (3.12)$$

and in three dimensions,

$$\begin{aligned} b_m = & \sum_{k=1}^N \frac{a_k \sqrt{\det(\mathbf{M}_k)}}{\pi^{3/2} \tilde{\delta}_k} h^3 \times \\ & \exp\left(-\frac{(\boldsymbol{\xi}_l - \mathbf{x}_k)^T (\sigma^2 \delta_k^2 \mathbf{M}_k^2 - (\delta_k^4 + \sigma^2 \delta_k^2 \text{Tr}(\mathbf{M}_k)) \mathbf{M}_k - \sigma^2 \det(\mathbf{M}_k) \mathbf{I}) (\boldsymbol{\xi}_l - \mathbf{x}_k)}{\tilde{\delta}_k^2}\right), \end{aligned} \quad (3.13)$$

where  $\tilde{\delta}_k^2 = (\delta_k^2 - \lambda_{k,1} \tau^2)(\delta_k^2 - \lambda_{k,2} \tau^2)(\delta_k^2 - \lambda_{k,3} \tau^2)$ . The eigenvalues of the shape

matrix  $\mathbf{M}_k$  are denoted by  $\lambda_k$  with a subscript 1, 2, or 3. Equations for axisymmetric Gaussian particles are obtained after setting  $\mathbf{M}_k$  equal to the identity matrix for all  $k$ .

As can be seen from (3.11), the remeshing procedure requires  $\tau^2 < \frac{\delta_k^2}{\lambda_k}$  for all core sizes  $\delta_k$  and eigenvalues  $\lambda_k$ , which puts an upper bound on the core size of the new particles. For axisymmetric particles, the core size is initially given by, say,  $\delta_0$ . Assume that the new set of particles will have again a core size equal to  $\delta_0$ . The equation for the core size of circular particles (3.9) shows that this size grows linearly with the square root of time and this remeshing method is therefore always possible for axisymmetric particles in the present of diffusivity.

Consider next the case of anisotropic Gaussian particles. The core size  $\delta_k$  is fixed and we assume that the old and new set of particles have identical core sizes. For this remeshing scheme to be applicable, all the eigenvalues of the shape matrix  $\mathbf{M}_k$  have to be less than unity. In the absence of diffusivity ( $\kappa = 0$ ) and the last term on the right-hand side of (3.6), it can be shown theoretically that this will be the case. The addition of diffusivity will decrease all eigenvalues over time. If the last term of (3.6) is present, eigenvalues can become larger than unity, and thus the remeshing procedure can not be used in this case. Different approximations can be used to circumvent this problem at the expense of accuracy. However, most numerical experiments showed that this scheme was still applicable in a wide variety of cases.

After an initialization procedure in which the initial scalar field is set up, the numerical code can be split in three subroutines. First, there is the integration routine, second there is the remeshing scheme and finally, there is the routine that computes the scalar field. The remeshing step and the computation of the scalar field step do not have to be executed each cycle. The integration routine scales linearly with the total number of particles and is by far the least expensive routine of the three. Both the remeshing routine and the one that computes the scalar field calculate the contribution of each particle to a regular grid. Assume that this grid has the same number of total points as the total number of particles  $N$ . Then, if there are  $d$  spatial dimensions, we will have  $\sqrt[d]{N}$  gridpoints in each spatial direction and we need

$N \left( \sqrt[d]{N} \right)^d = N^2$  operations. Or, if we have  $M$  particles in each spatial direction, we end up with  $M^4$  operations in 2D and  $M^6$  operations in 3D.

## Chapter 4 2D Flow Examples

This chapter discusses results for three different 2D test problems. The first two, a rotating flow and a stagnation flow, are simple examples for which analytic solutions are known. They are used to validate the numerical code and to illustrate several aspects of the model and method. It is followed by an indepth discussion of a time-dependent incompressible velocity field yielding chaotic advection. Comparisons are made with filtered direct numerical simulation (DNS) data and the Smagorinsky subgrid model.

### 4.1 Rotating Flow

We will start with a simple example for illustrative purposes. Assume no subgrid model ( $\sigma = 0$ ) and set the diffusivity constant  $\kappa$  equal to zero. Consider a rotating flow in two dimensions with cylindrical velocity components  $u_r = 0$  and  $u_\theta = r$ , where  $u_r$  is the radial velocity and  $u_\theta$  the azimuthal velocity component. The distance to the origin is  $r$ . In terms of Cartesian coordinates the velocity components are  $u = -y$  and  $v = x$ . The velocity gradient is a constant matrix given by

$$\nabla \mathbf{u} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \quad (4.1)$$

The location of the particles is governed by (3.4). Evaluation of this equation shows that each particle moves with the velocity of its respective center, i.e.,  $\frac{d\mathbf{x}_k}{dt} = \hat{\mathbf{u}}(\mathbf{x}_k) = \mathbf{u}(\mathbf{x}_k)$ . We consider next the time development of the shape of the anisotropic particles, given by (3.6), which can be reduced to

$$\frac{d\mathbf{M}_k}{dt} = \begin{bmatrix} -2m_{12} & m_{11} - m_{22} \\ m_{11} - m_{22} & 2m_{12} \end{bmatrix} \quad (4.2)$$

with the general solution

$$\begin{aligned}
m_{11}(t) &= \frac{C_1}{2}(1 - \cos(2t)) - C_2 \sin(2t) + \frac{C_3}{2}(1 + \cos(2t)), \\
m_{12}(t) &= -\frac{C_1}{2} \sin(2t) + C_2 \cos(2t) + \frac{C_3}{2} \sin(2t), \\
m_{22}(t) &= \frac{C_1}{2}(1 + \cos(2t)) + C_2 \sin(2t) + \frac{C_3}{2}(1 - \cos(2t)),
\end{aligned} \tag{4.3}$$

where  $C_1$ ,  $C_2$ , and  $C_3$  are constants. It is easy to show that  $\det(\mathbf{M}_k) = m_{11}m_{22} - m_{12}^2 = C_1C_3 - C_2^2 = \text{constant}$ . For the simple case of  $C_1 = C_3 = 1$  and  $C_2 = 0$ , we have  $m_{11}(t) = m_{22}(t) = 1$  and  $m_{12}(t) = 0$ , i.e., a particle that is initially circular remains circular. The equations show that particles only rotate in time as a solid body. It is easy to verify that above equations of motion for the particles are exact solutions to the original advection-diffusion equation (2.1).

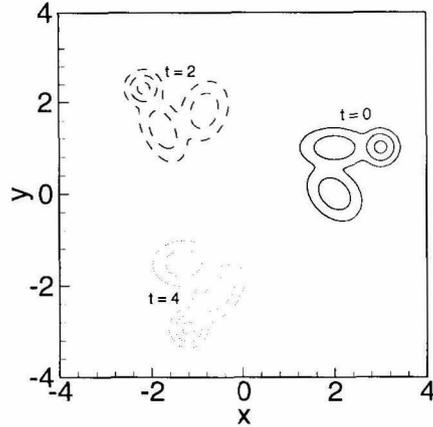


FIGURE 4.1: Contour lines 0.1, 1, and 5 for a rotating flow at  $t = 0, 2$ , and  $4$ . No subgrid model was used.

$x_k$	$y_k$	$a_k$	$\delta_k$	$m_{11}$	$m_{22}$	$m_{12}$
2.0	0.0	1.0	0.3	1.0	1.0	0.3
3.0	1.0	1.0	0.2	1.0	1.0	0.0
2.0	1.0	1.0	0.4	1.0	3.0	0.0

TABLE 4.1: Test data for the rotating flow.

For the numerical computation we used three particles for which the data is given

in table 4.1. Figure 4.1 gives contour plots of  $\psi$  for times  $t = 0, 2,$  and  $4$ . One sees that all three particles rotate as a solid body around the origin, where a complete rotation is made after  $t = 2\pi$ . The numerical results are in agreement with the analytical solution.

## 4.2 Stagnation Flow

### 4.2.1 Theory

Consider a two-dimensional stagnation flow given by the incompressible velocity components  $\hat{u} = cx$  and  $\hat{v} = -cy$ , where  $c$  is an arbitrary constant. Assume no molecular diffusivity ( $\kappa = 0$ ). Consider furthermore only one anisotropic Gaussian particle  $k$  in the flow initially located at  $\mathbf{x}_k$  with the identity matrix as shape function ( $\mathbf{M}_k = \mathbf{I}_2$ ). For this specific velocity field the tensor-diffusivity subgrid model is exact and thus the filtered advection-diffusion equation (2.7) is also. The (numerical) particle-method using anisotropic Gaussian particles is also an exact solution to (2.7).

Since the Laplacian of the velocity field is zero, the equation of motion for the location of the particle (3.4) reduces to

$$\frac{d\mathbf{x}_k}{dt} = \overline{\mathbf{u}}^k = \hat{\mathbf{u}}(\mathbf{x}_k) = \begin{pmatrix} cx_k \\ -cy_k \end{pmatrix}, \quad (4.4)$$

where the result can be obtained using a simple integration. We conclude that the particle will move with the velocity of its center. If we look at the equation for the shape of the particle (3.6), it can be reduced to

$$\frac{dm_{11}}{dt} = -2cm_{11} + \frac{2\sigma^2}{\delta^2}c(m_{11}^2 - m_{12}^2), \quad (4.5a)$$

$$\frac{dm_{22}}{dt} = 2cm_{22} + \frac{2\sigma^2}{\delta^2}c(m_{12}^2 - m_{22}^2), \quad (4.5b)$$

$$\frac{dm_{12}}{dt} = 2\frac{\sigma^2}{\delta^2}cm_{12}(m_{11} - m_{22}). \quad (4.5c)$$

Since at  $t = 0$ ,  $m_{12}$  equals zero, we can derive from (4.5c) that  $m_{12}$  equals zero for

all time. The equations for  $m_{11}$  and  $m_{22}$  become then decoupled and their solution is given by

$$m_{11}(t) = \frac{\delta^2/\sigma^2}{1 + \left(\frac{\delta^2}{\sigma^2} - 1\right) e^{-2ct}}, \quad (4.6a)$$

$$m_{22}(t) = \frac{\delta^2/\sigma^2}{1 + \left(\frac{\delta^2}{\sigma^2} - 1\right) e^{2ct}}, \quad (4.6b)$$

where  $m_{11}(0) = m_{22}(0) = 1$  has been used as an initial condition.

For  $\delta < \sigma$ , the solution blows up at  $t = -\frac{1}{2c} \ln\left(1 - \frac{\delta^2}{\sigma^2}\right)$ . For  $\delta > \sigma$ , the solution stays 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 blow-up of the solution for  $\delta < \sigma$  is therefore not relevant to our applications, since such Gaussians do not correspond to meaningful basis elements for the unfiltered field.

### 4.2.2 Numerical Tests

Four different test runs have been performed, numbered run 1 through 4, all using  $c = 1$ . Each of these runs used a Runge-Kutta method with adaptive step size control to march forward in time as discussed in section 16.2 of Press et al. [36]. Run 1 starts with one big particle with an amplitude equal to unity, a core size equal to 0.5, and is centered at (0,2) on the  $y$ -axis. We will use  $\sigma = 0.2$  for all runs.

The other three test runs have used 148 (smaller) particles on a regular mesh to approximate the one large particle of run 1. Table 4.2 gives more detailed information about each of the four runs. The first column gives the run number, the second one indicates the type of particles used and the next column gives the core size  $\delta_k$  of each of the individual particles. The last column gives the number of particles in the simulation ( $N$ ). For all runs starting with 148 particles, the particles were separated by  $h = 0.2$  in both spatial directions. For runs 2 and 3 this amounts to an overlap parameter of  $\delta_k/h = 1.25$  and for run 4 to  $\delta_k/h = 0.75$ . The initial scalar fields of runs 2 through 4 are identical to the one of run 1, except for small spatial discretization

errors.

Run	Particles	$\delta_k$	$N$
1	Anisotropic	0.50	1
2	Anisotropic	0.25	148
3	Axisymmetric	0.25	148
4	Anisotropic	0.15	148

TABLE 4.2: Test data for the stagnation flow.

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. Define the aspect ratio for each particle as the largest eigenvalue over the smallest eigenvalue of the shape function  $\mathbf{M}_k$ . 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  $\frac{m_{22}}{m_{11}}$ . At  $t = 1$ , the aspect ratio for run 1 equals 23.26 and for run 2 equals 4.79, but for this problem the solution remains exact.

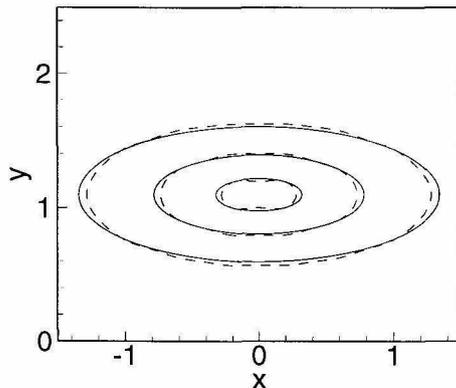


FIGURE 4.2: Contour lines 0.1, 0.5, and 1.0 for runs 1 and 2 (solid - results are virtually identical) and 3 (dashed) at  $t = 0.6$ . Filtering constant  $\sigma = 0.20$ .

Figure 4.2 gives the contour lines of  $\psi$  for runs 1, 2, and 3 at  $t = 0.6$ . 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 not correspond with a realistic unfiltered field. Figure 4.3 illustrates this blow-up for run 4 at  $t = 0.3$ . It is interesting to note that the particles take on an elliptical shape oriented  $90^\circ$  from the direction as one would expect. For the parameters given in run 4, the theoretical results predict that the solution blows up at  $t = 0.41$ , which also happens in the numerical calculation. The blow-up is probably due to the amplification of the high spatial frequencies in the initial data.

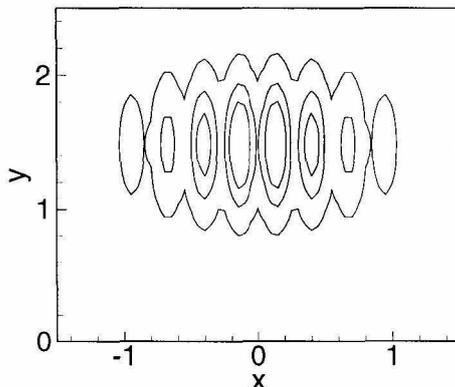


FIGURE 4.3: Contour lines 0.1, 0.5, and 1.0 for run 4 at  $t = 0.3$  with particle sizes of  $\delta_k = 0.15$  and filtering constant  $\sigma = 0.2$ .

### 4.3 2D Time-Dependent Flow Field

This section features an indepth discussion of the two-dimensional incompressible velocity field given by (3.1) in order to show and test different aspects of the particle method and the subgrid model. The computations in this section will use  $\epsilon_x = 0$ ,  $\epsilon_y = 0.5$ , and  $\omega_y = 0.5$  unless otherwise stated. Figure 3.1 gives streamlines of this velocity field for two different times. Note that we should not forget to filter this velocity field using (2.4) before using it as an initial condition for (2.7).

This section starts with a discussion of the numerical details of the particle method. A comparison between filtered DNS data, the Smagorinsky subgrid model

and the tensor-diffusivity model will be made. The effect of remeshing and the benefits of using anisotropic versus axisymmetric particles are investigated. This section ends with a novel approach for marching particles forward in time.

### 4.3.1 Numerical Implementation

All computations use a diffusivity of  $\kappa = 0.001$  and a filtering constant of  $\sigma = 0.15$  unless otherwise stated. The same initial condition for the unfiltered field as in the finite difference scheme has been used,  $\exp\left(-\frac{(x - 0.3)^2 + (y + 0.4)^2}{\tau^2}\right)$ , where  $\tau = 0.5$ . After filtering this field with (2.4), the initial condition is plotted in figure 4.4. Contour lines are 0.0001, 0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.3, and 0.5, just as in most other color plots in this chapter.

The numerical computation starts with finding a good approximation to the initial scalar field. We use a Gaussian initial function, as it is easily approximated by a sum of smaller Gaussians. Every particle is assumed to have an axisymmetric shape ( $\mathbf{M}_k = \mathbf{I}_2$ ) and is placed on a regular mesh with a fixed core size  $\delta_k$ . Only the amplitude needs to be determined based on the initial scalar field we want to approximate.

For a Gaussian initial field, we can use (3.12), which holds for 2D, to find the amplitudes of the particles. Particles with an amplitude smaller than a certain cut-off value ( $10^{-4}$ ) are disregarded. If we use 360 grid points between -6 and 6 in both spatial directions and set the core size equal to  $\delta_k = 0.15$ , we end up with 6504 particles and an overlap parameter of 4.5 (grid spacing  $h = 0.033$ ). Numerical experiments show that if the core size is equal to or slightly larger than the filtering constant  $\sigma$ , the remeshing scheme yields the most accurate results.

A fifth order Runge-Kutta method with adaptive step size control as discussed in section 16.2 of Press et al. [36] was used to march the particles forward in time. After each time step, one could compute the scalar field and/or remesh it. Computing or remeshing the field both require a large computational effort. In the case of calculating the scalar field, a regular equidistant grid is set up. In order to find the scalar

field, we have to find in principle the contribution of each particle to each individual grid point. Because the exponential tail of the Gaussian approaches zero fast, each particle contributes effectively only to the neighboring grid points. For the remeshing procedure, a similar grid is set up on the location of all the new particles. The contribution of each of the old particles to all the neighboring grid points is determined using (3.12).

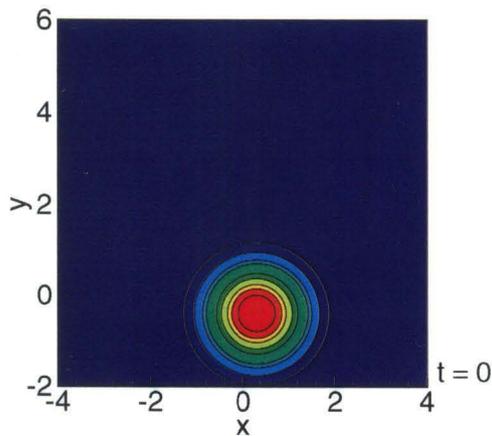


FIGURE 4.4: Initial Gaussian distribution centered at  $(0.3, -0.4)$  with  $\tau = 0.5$ .

### 4.3.2 DNS Solution and Smagorinsky Model

In order to test the accuracy of the particle method, a central second-order difference method has been implemented to solve (2.1) 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 Perić [14] was used to integrate in time with timestep  $dt = 0.005$ . In order to compare this converged DNS solution with the results of the tensor-diffusivity subgrid model, the solution has been filtered using  $\sigma = 0.15$  using a quadrature function over all grid points.

Figure 4.5 compares the DNS solution with the filtered DNS solution at  $t = 9$ . As expected, the filter smooths out the large gradients in the solution. The left column of figure 4.6 shows contour plots for the filtered DNS solution at times  $t = 3, 6,$  and  $9$ .

In order to compare the results of the tensor-diffusivity model using Gaussian particles, a finite difference computation using the Smagorinsky subgrid model has been made (gridspacing 0.016, step size 0.005). Results are given in the right column of figure 4.6 for a Smagorinsky constant of  $C_s = 0.2$ . The use of other constants was investigated, but  $C_s = 0.2$  gave the 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 some areas.

### 4.3.3 Lagrangian Particle Method

We will next compare the filtered DNS solution with the results obtained using the particle method. As stated before, 6504 particles were used initially. The left column in figure 4.8 gives the solution without the subgrid model using anisotropic particles. The solution has not been remeshed. Soon after the start of the computation, the solution starts to show differences with the filtered DNS solution. After turning on the tensor-diffusivity model, the results are depicted in the right column of 4.8. 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

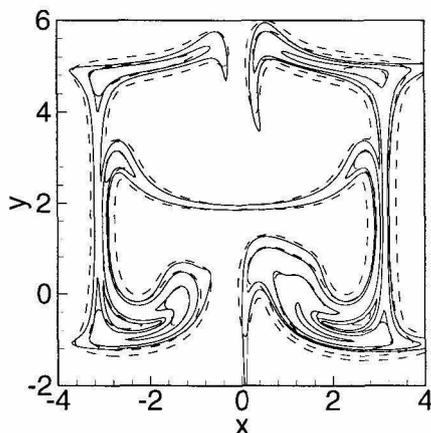


FIGURE 4.5: Comparison between DNS solution (solid) and filtered DNS solution using  $\sigma = 0.15$  at  $t = 9$ . Contour lines are 0.001, 0.01, 0.1, and 0.5. The DNS solution was obtained using a finite difference method.

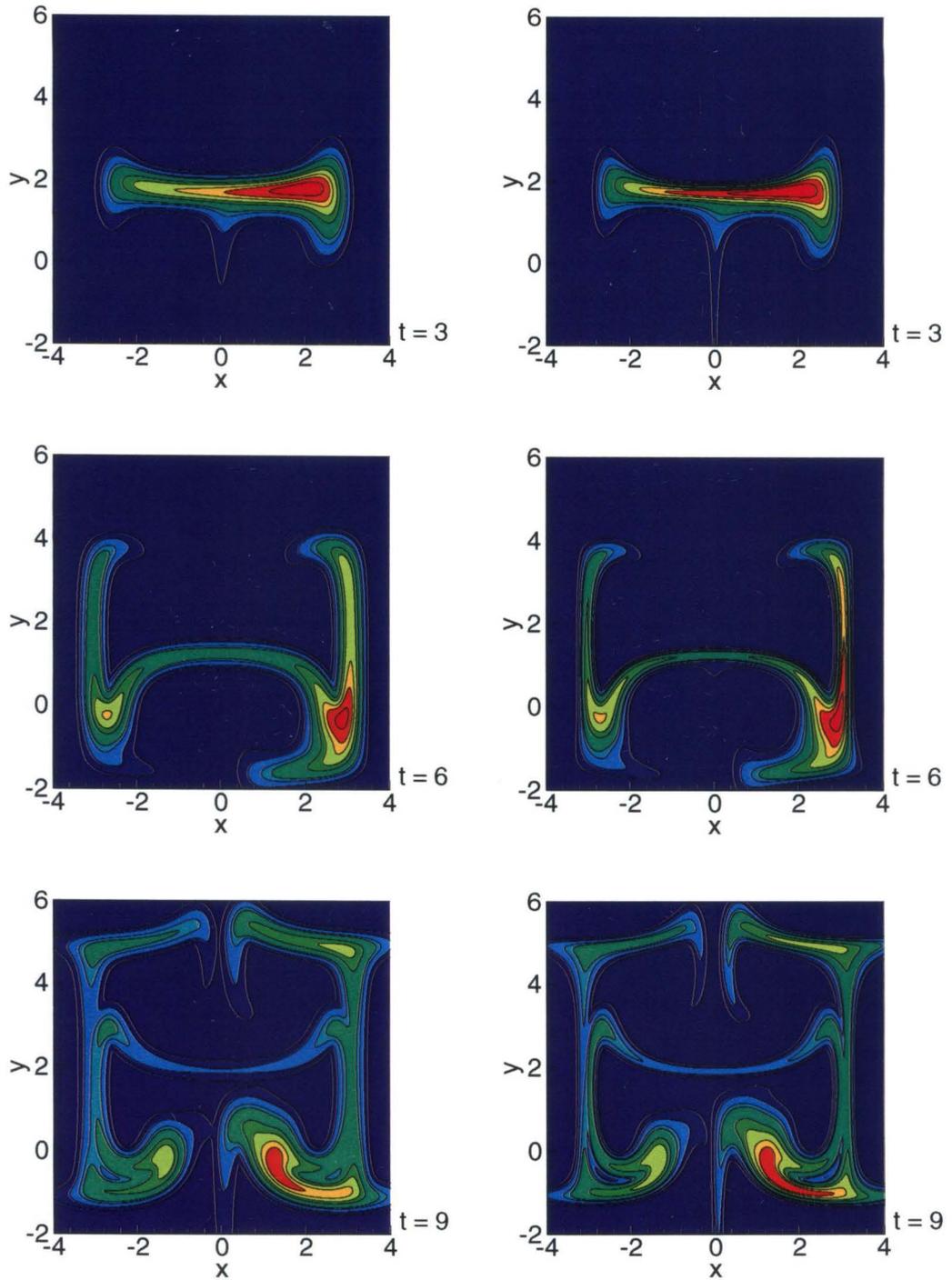


FIGURE 4.6: Contour plots for filtered DNS solution (left column) and Smagorinsky subgrid model (right column). Both computations used a finite difference method.

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% larger than 25. At  $t = 9$ , the maximum aspect ratio has increased to 365 and more than half of the particles have an aspect ratio greater than 10.

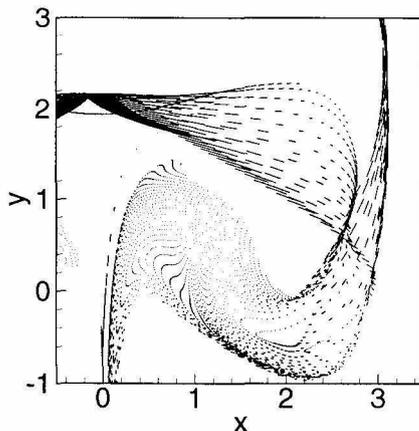


FIGURE 4.7: Direction and aspect ratio of anisotropic Gaussian particles at  $t = 9$ . The length is a measure for the aspect ratio. No remeshing was used and  $\sigma = 0.15$ .

The location of the anisotropic particles is given in figure 4.7 at  $t = 9$ . Each line segment represents a particle. The length of the segment is an indication for the aspect ratio and the 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.

In order to improve the accuracy of the solution, the scalar field has been remeshed every unit of time. The same grid used for computing the initial condition has been used in the remeshing procedure. Since the field spreads out over time, the total number of particles increases throughout the computation. Starting with 6504 parti-

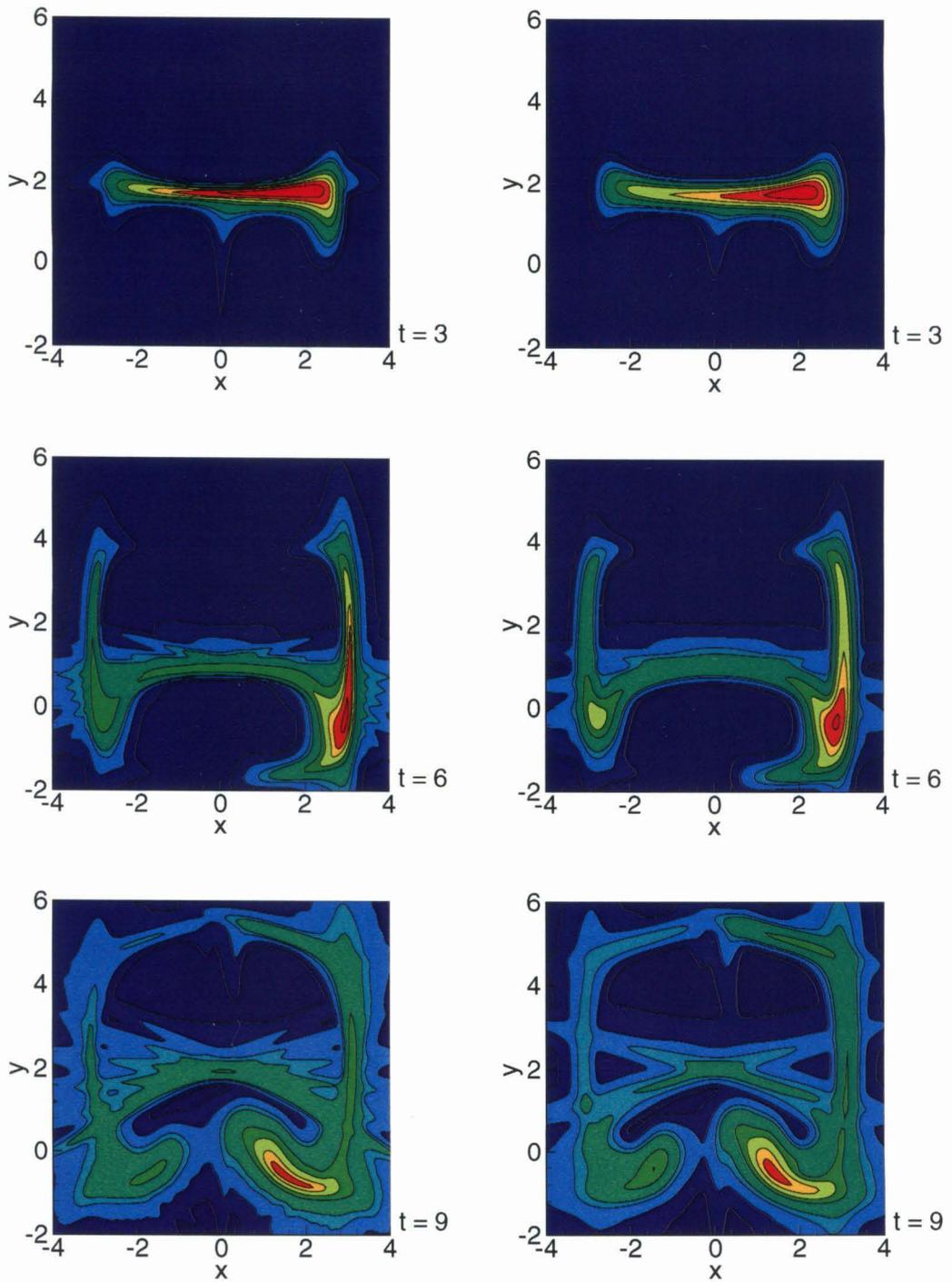


FIGURE 4.8: Contour plots for the Lagrangian particle method using anisotropic particles and no subgrid model (left column) and the tensor-diffusivity model (right column). No remeshing was used and  $\sigma = 0.15$ .

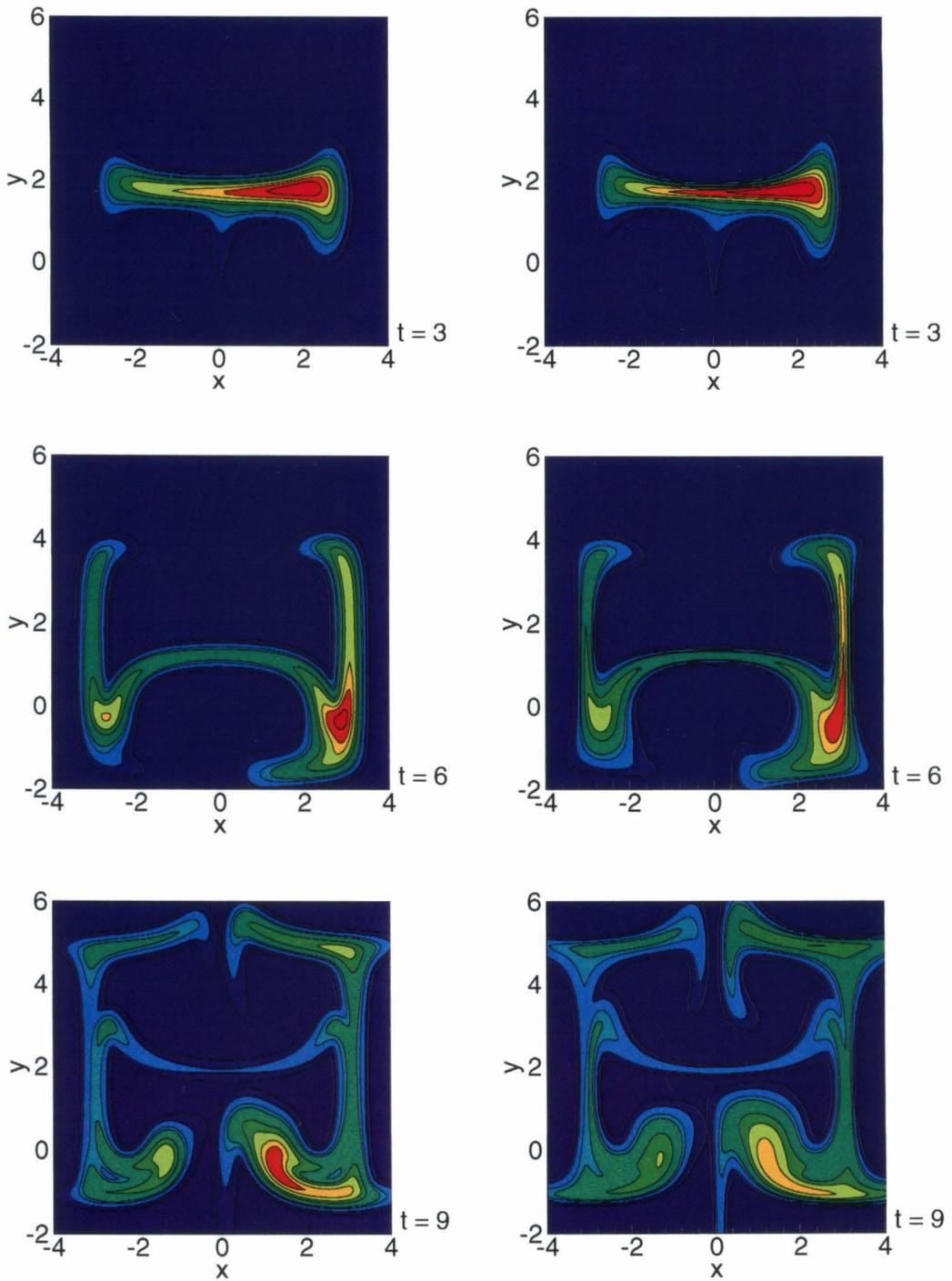


FIGURE 4.9: Contour plots for anisotropic particles with the tensor-diffusivity model and remeshing (left column) and with only remeshing (right column). Both computations were remeshed every time unit and  $\sigma = 0.15$ .

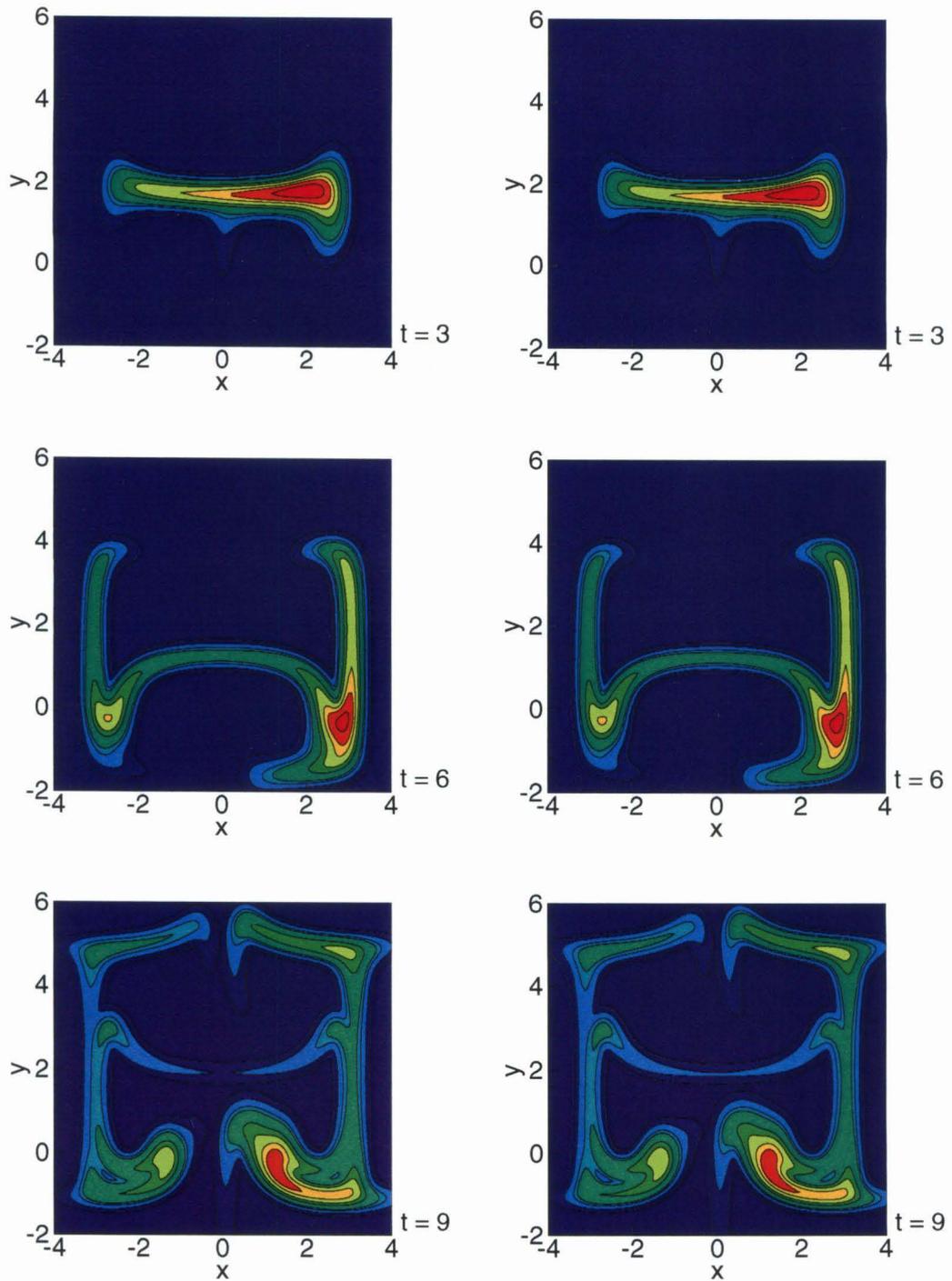


FIGURE 4.10: Contour plots for axisymmetric particles using the tensor-diffusivity subgrid model (left column) and the unfiltered velocity field subgrid model (right column). Remeshed every time unit and  $\sigma = 0.15$ .

cles, the number reaches 18745 particles at  $t = 9$ . The contour plots are given in the left column of figure 4.9. Very good agreement between this solution and the filtered DNS solution is obtained.

In order to verify that the good agreement is partially due to the tensor-diffusivity subgrid model and not only to the remeshing scheme, a computation has been carried out where the subgrid model was turned off and the solution was remeshed every time unit. The results are given in the right column of figure 4.9. Some of the large structures are still recognizable, but the solution without the subgrid model is clearly not as good as with.

To see the effect of using a higher order particle method, the solution using axisymmetric particles is plotted in the left column of 4.10. The solution was remeshed every time unit resulting in 20520 particles at  $t = 9$ . In absence of remeshing, the solution for anisotropic particles is still accurate at  $t = 3$ . This is not the case for axisymmetric particles anymore. However, with remeshing, 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.

The solution that is presented has converged using about 6500 (anisotropic) particles initially and remeshing every time unit. Good solutions were obtained using at least 1900 particles (overlap parameter 2.4). Remeshing every 1 to 3 time units was needed for good solutions. The remeshing scheme is more accurate if particles have had some growth from diffusion. However, waiting too long before the remeshing procedure is applied will lead to inaccuracies in the integration scheme. Remeshing is necessary because of the growing core size of the particles due to diffusivity and the changing shape due to the straining in the velocity field. Computations carried out for larger and smaller values of the diffusivity showed that remeshing was still needed every time unit indicating that the straining in the velocity field is the dominant factor for remeshing in this example and not diffusion. In the absence of diffusion, the remeshing method presented here will not work and one has to resort to another

method.

To obtain a converged DNS solution using a finite difference scheme, we needed about  $800 \times 800 = 64,000$  grid points. If we would have used a particle method to compute a DNS solution, we expect that we would need about the same number of particles as grid points in the finite difference calculation for a converged solution. Using a filtering constant of  $\sigma = 0.15$  and the tensor-diffusivity model, we showed that about 2000 particles result in a good solution, thereby reducing the number of datapoints needed considerably just as the computational time needed while still calculating the large scale flow structures accurately.

We know that the smallest possible scale in the unfiltered field (a delta function) corresponds with a Gaussian of width  $\sigma$  in the filtered field. On the other hand, any Gaussian with a core size larger than  $\sigma$  in the filtered field corresponds with a Gaussian of a smaller core size in the unfiltered field. Since all particles have a core size larger than  $\sigma$ , we can find an approximation to the unfiltered solution. Figure 4.11 shows a comparison at  $t = 9$  between the DNS solution and the unfiltered solution obtained using anisotropic and axisymmetric particles.

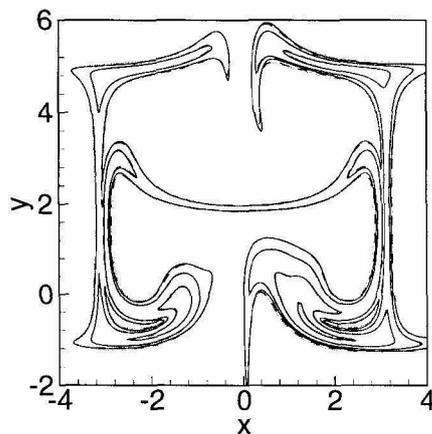


FIGURE 4.11: Comparison between DNS solution (solid) and an unfiltered approximation using anisotropic particles (dashed) and axisymmetric particles (dotted) at  $t = 9$ . Contour lines are 0.001, 0.01, 0.1, and 0.5.

The agreement is remarkable for both anisotropic and axisymmetric particles. It is hard to see, but there is a small wave pattern on the lowest (0.001) and highest (0.5)

contour lines. This is most likely caused by particles that do not overlap very well anymore, since the core size of each particle has effectively been reduced, decreasing the overlap ratio.

All computations so far have used  $\sigma = 0.15$ . For smaller values of  $\sigma$  less detail is filtered out and to obtain good numerical results, we need more particles. For larger values of  $\sigma$ , we expect to use less particles. However, we retained only the first two terms in the infinite expansion (2.6) assuming that the remaining terms are small. For large values of  $\sigma$ , this is not the case anymore and we expect the model to break down. There is another complication for anisotropic particles. While the integration procedure is more accurate than for axisymmetric particles, the remeshing procedure starts to break down for larger values of  $\sigma$ . Due to the last term in (3.6), the eigenvalues of the shape matrix  $\mathbf{M}_k$  can become larger than unity. In order to apply the remeshing procedure we choose  $\tau$  in (3.12) smaller than the new core size of the particles in order to satisfy  $\tau^2 < \frac{\delta_k^2}{\lambda_k}$ . This will introduce errors. Since the computation using axisymmetric particles only let the core size of the particles grow in time, the remeshing scheme remains accurately.

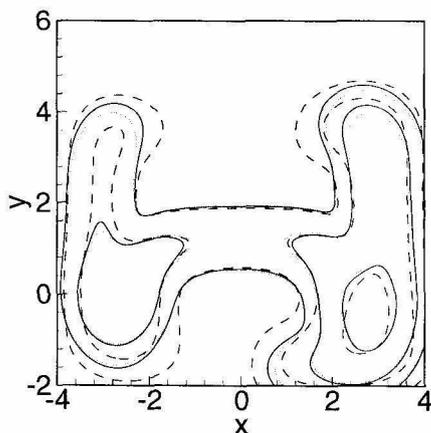


FIGURE 4.12: Comparison between filtered DNS (solid), anisotropic particles (dashed) and axisymmetric particles (dotted) for  $\sigma = 0.50$  at  $t = 6$ . Contour lines are 0.001, 0.01, 0.1, and 0.5. The computations using particles were remeshed every time unit.

Figure 4.12 compares the results for filtered DNS, anisotropic, and axisymmetric

particles for a computation using  $\sigma = 0.5$ . Both methods are about to break down for these large values of  $\sigma$ . The solution using axisymmetric particles is better due to more accurate results for remeshing.

#### 4.3.4 Random Initial Field

Several tests have been performed using a random initial scalar field with the velocity field given by (3.1). For all these computations, periodic boundary conditions have been used at  $-\pi$  and  $\pi$  in both spatial directions. This corresponds with one period of the velocity field. A regular rectangular grid of 150 particles in each direction is laid out resulting in a total of 22,500 particles. The constant of diffusivity  $\kappa$  is set equal to 0.001 and the filtering constant  $\sigma$  to 0.10. The initial core size of the anisotropic Gaussian particles is equal to  $\delta_k = \sigma$ . The set of particles is remeshed after every time unit. Particles moving across a boundary of the spatial domain will appear again on the other side of the domain.

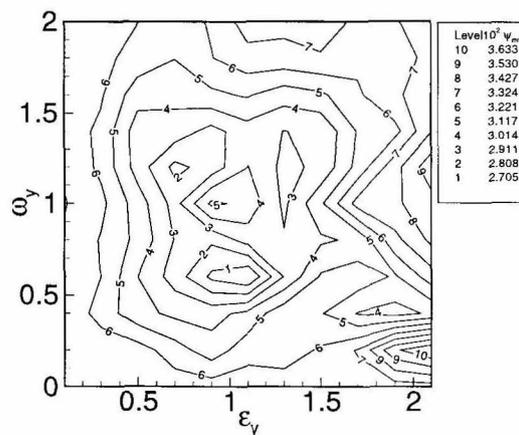


FIGURE 4.13: Contour plot of the root mean square value as a function of  $\epsilon_y$  and  $\omega_y$  of the 2D velocity-field given by (3.1) ( $\epsilon_x = 0$ ).

The amplitude of each of the particles was set initially equal to a random number between -1 and 1 times the gridspacing squared. It is thus expected that the integral of the scalar field over the spatial domain is close to zero. In the long time limit, it is expected that the scalar field is completely mixed and that the scalar function

is constant everywhere and close to zero. A good measure for how well the scalar quantity is mixed is given by the root mean square of the scalar field; see (2.9). Initially, the rms value is equal to 0.101626 for each of the computations.

Test runs have been performed to establish the effect of the parameters  $\epsilon_y$  and  $\omega_y$  ( $\epsilon_x = 0$ ) on mixing. Runs have been performed for a range of frequencies between 0 and 2 and a range of amplitudes between 0 and 2.1. To compare the effect of mixing, a contour plot of the root mean square of the scalar field is given in figure 4.13 at time  $t = 10$ . For these range of values, mixing occurs most efficiently at an amplitude near  $\epsilon_y = 1.1$  and a frequency near  $\omega_y = 0.6$ . The frequencies considered here are quite low. For  $\omega_y = 0.6$ , it takes more than 10 time units to complete one period, which is longer than the time scale of the entire computation. Note that for high values of the amplitude  $\epsilon_y$  mixing is not very efficient any more. In these cases, the perturbation itself dominates the other terms in the velocity profile.

### 4.3.5 Use of the Unfiltered Velocity Field

Recall that our goal is to solve (2.3) for the filtered scalar field. We approximate this equation by (2.7), which we solve numerically. For large  $\sigma$ , this approximation is not valid in general and we should include more terms of (2.6) to improve the solution, which will lead to corrections to (3.4) to move the particles in time.

We hypothesize that (3.4) with such additional corrections ultimately leads to the unfiltered velocity field given by (3.1). On the other hand, in the absence of diffusivity, a delta function moving with the local velocity field is an exact solution of (2.1). In the filtered space, this delta function corresponds to a Gaussian with core size  $\sigma$ . By choosing the (initial) core size of a particle equal to  $\sigma$ , which corresponds to a delta function in the unfiltered field, we therefore obtain an exact solution by moving the particle with the unfiltered velocity field.

In the presence of diffusion, the initial delta function in the unfiltered field will grow in time as per (3.9). Moving these particles, which become Gaussians with small cores, with the unfiltered velocity field will introduce errors. The same thing will

happen in the filtered space. However, we expect that these errors can be controlled by remeshing the scalar field every so often and replacing the old set of particles which have grown, by a new set of particles each with a core size equal to  $\sigma$  again. These new particles correspond again with delta functions in the unfiltered space which can be solved exactly by moving them with the unfiltered velocity field.

In our case, the unfiltered velocity field happens to be available. Thus in this section, computations have been carried out by moving particles using the value at the center of the particle of the unfiltered velocity field (3.1). Note that these computations are different from the ones carried out before in which the filtered version of (3.1) was used with the exception of the DNS calculation.

Computations for  $\sigma = 0.15$  give good agreement with the filtered DNS data as can be seen in the right column of figure 4.10. Figure 4.14 gives the results for  $\sigma = 0.5$ . Axisymmetric particles were used. Whereas the subgrid model starts to break down at  $\sigma = 0.5$ , using the unfiltered velocity field still yields good results.

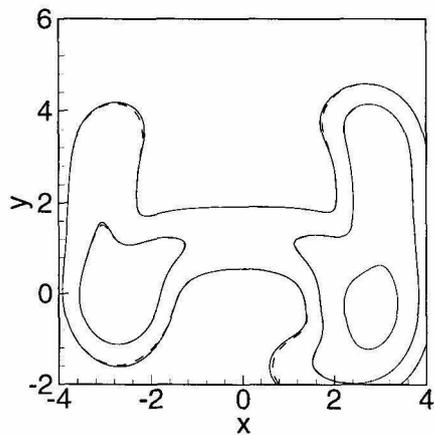


FIGURE 4.14: Comparison between filtered DNS (solid) and using the original velocity field to move particles at  $t = 6$ . Contour lines are 0.001, 0.01, and 0.1 and  $\sigma = 0.50$ . The particle solution was remeshed every time unit.

Note that we can express the unfiltered velocity field  $\mathbf{u}$  in terms of the filtered

velocity field  $\hat{\mathbf{u}}$  to end up with (in one dimension)

$$u(x, t) = \sum_{n=0}^{\infty} \frac{(-1)^n}{2^n} (2n-1)(2n-3)\dots 1 \frac{\partial^n \hat{u}}{\partial x^n} = \hat{u} - \frac{\sigma^2}{2} \frac{\partial^2 \hat{u}}{\partial x^2} + \frac{3\sigma^4}{4} \frac{\partial^4 \hat{u}}{\partial x^4} + \dots \quad (4.7)$$

This result indicates that the right-hand side of (3.4) is an approximation to the unfiltered velocity field.

Using the unfiltered velocity field to march particles forward in time can be interpreted as a subgrid model. When the unfiltered field is available, very good results are obtained. If this field is not known, we expect that a good subgrid model gives an approximation to this unfiltered field.

Note that when one uses the unfiltered velocity field, the filtering constant  $\sigma$  drops out of the equations of motion. It is only used to determine the initial size of the particles. By choosing a smaller value for  $\sigma$ , less details of the solution are filtered out. However, this comes at a computational cost. In order to obtain a smooth scalar field, neighboring particles need to overlap. By using a smaller value for  $\sigma$  (and thus for the size of the particles), more particles are needed for a smooth scalar field. Thus the available computational time puts a limit on the minimum size of  $\sigma$ . This method is not suited for solving the advection-diffusion equation (2.1) directly, which corresponds to the case  $\sigma = 0$ , which would require an infinite number of particles to get a smooth scalar field.

As a nice finishing touch to this chapter, a computation for longer time is presented on the following two pages to visualize the process of mixing in figures 4.15a-b. The unfiltered velocity field was used to move the axisymmetric particles around. Diffusivity was set to  $\kappa = 0.001$ , filtering constant  $\sigma = 0.15$ . For the velocity field, the parameters  $\epsilon_x = 0.5$ ,  $\omega_x = 2.0$ ,  $\epsilon_y = 0.5$ , and  $\epsilon_y = 1.0$  have been used to maximize mixing of the scalar. An initial Gaussian with  $\tau = 1.0$  located at  $(x, y) = (0.3, 1.2)$  was used. An infinite domain was used. If particles go outside the grid used for remeshing, which is larger than the plots, they are lost. The solution was remeshed every time unit. Times from  $t = 0$  up to  $t = 33$  are shown.

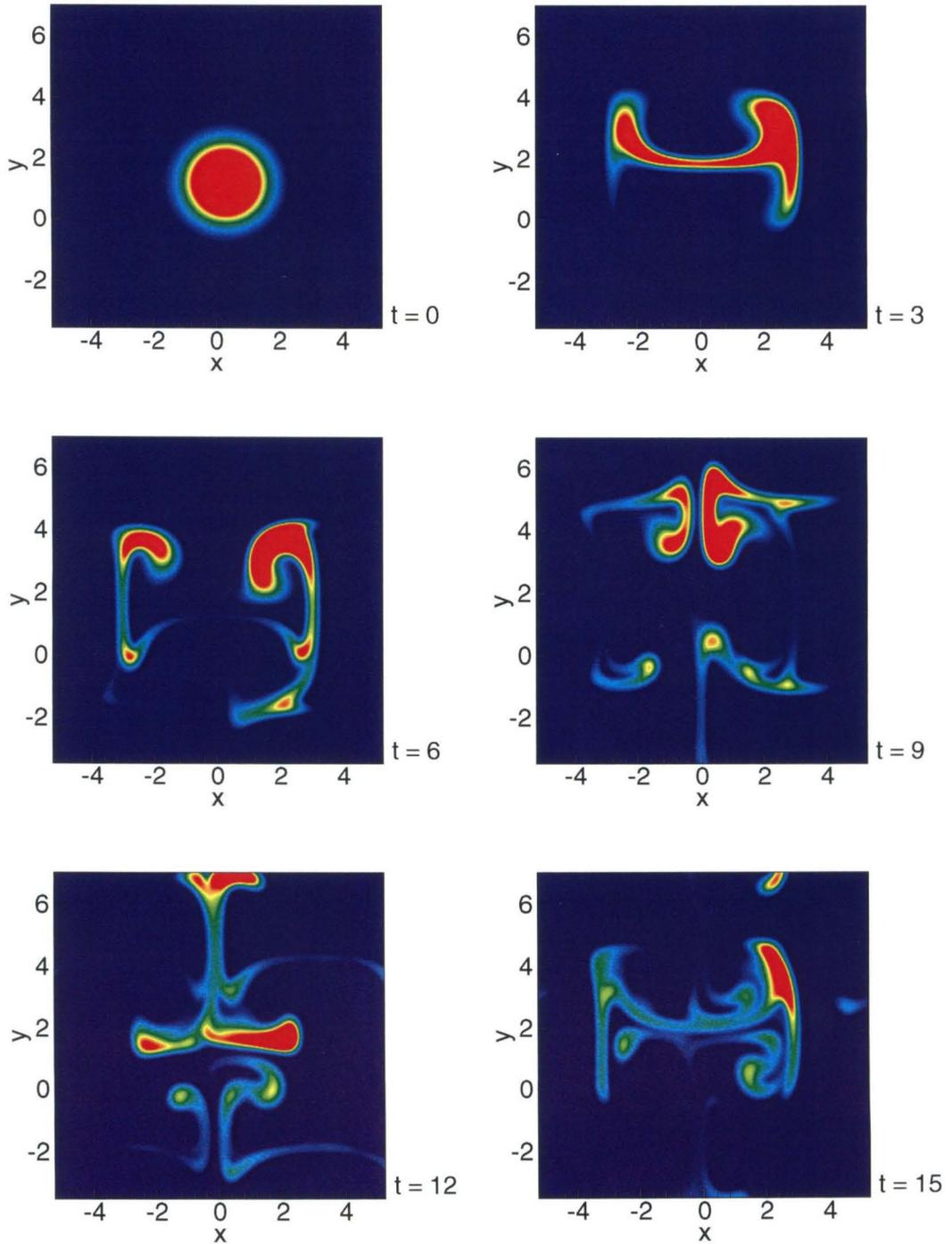


FIGURE 4.15A: Example calculation of tensor-diffusivity subgrid model using the unfiltered velocity subgrid model,  $\sigma = 0.15$  and axisymmetric particles. Solution was remeshed every time unit.

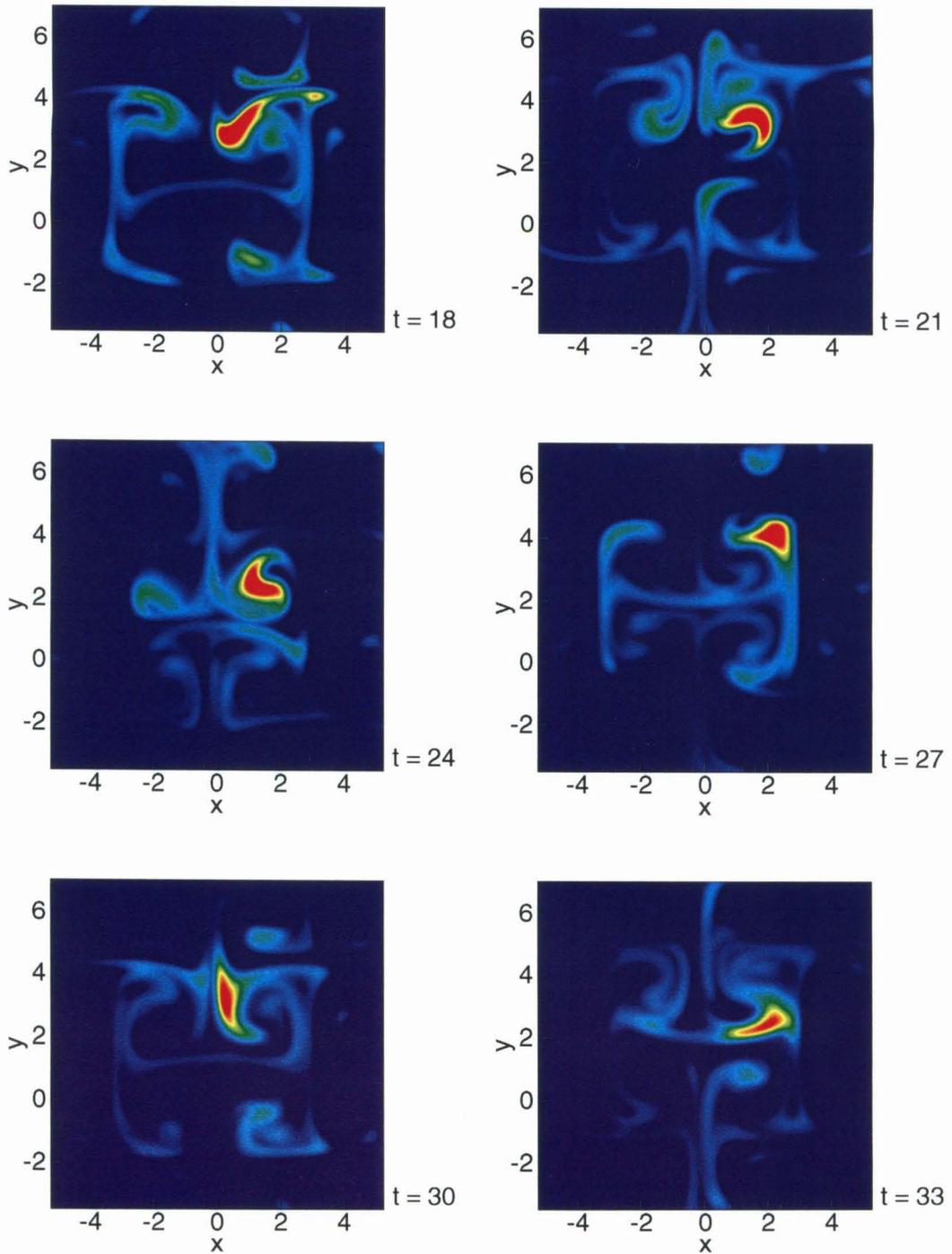


FIGURE 4.15B: Example calculation of tensor-diffusivity subgrid model using the unfiltered velocity subgrid model,  $\sigma = 0.15$  and axisymmetric particles. Solution was remeshed every time unit.

## Chapter 5 3D Flow Example

This chapter discusses the 3D flow example used by Toussaint et al. [47] and compares their results with our 3D particle method. They used a one-parameter 3D steady velocity field that yields chaotic advection to study mixing for small values of the diffusivity by employing a spectral method. Their computations showed that there is an optimal parameter value for which mixing is most efficient. The extension of the 2D particle method to 3D is straightforward.

### 5.1 3D Velocity Field

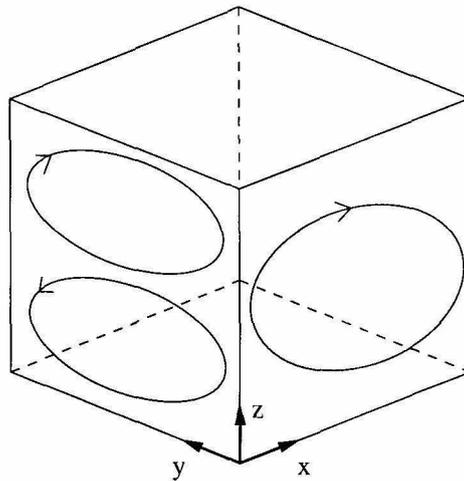


FIGURE 5.1: Sketch of the 3D velocity field given by (5.1) consisting of a summation of two 2D velocity fields.

The 3D velocity field we consider is a superposition of two 2D velocity fields  $\mathbf{U}_1$  and  $\mathbf{U}_2$  each consisting of sinusoidal functions. The total velocity field  $\mathbf{U}$  can be

written as

$$\mathbf{U} = \mathbf{U}_1 + \mathbf{U}_2 = U_1 \begin{pmatrix} -\sin(\pi x) \cos(\pi z) \\ 0 \\ \cos(\pi x) \sin(\pi z) \end{pmatrix} + U_2 \begin{pmatrix} 0 \\ -2 \sin(\pi y) \cos(2\pi z) \\ \cos(\pi y) \sin(2\pi z) \end{pmatrix}. \quad (5.1)$$

A schematic sketch of the velocity field is given in figure 5.1.

The properties of this velocity field depend on the parameters  $U_1$  and  $U_2$ . Since we are interested in the efficiency of mixing, we will choose  $U_1$  and  $U_2$  such that all velocity fields we consider have the same power input. Following Toussaint et al. [47] we require that the global viscous dissipation rate is constant, which leads to the following relation between  $U_1$  and  $U_2$ ,

$$U_1^2 + \frac{25}{4}U_2^2 = 1. \quad (5.2)$$

We will choose  $U_1$  as the independent variable. Note that if  $U_1$  is 0 or 1, the velocity field is purely 2D.

Figure 5.2 shows Poincaré sections at  $x = 0.5$  for four different values of  $U_1$ . A Poincaré section is obtained by following a streamline and plotting a dot every time the streamline crosses a certain section,  $x = 0.5$  in our case, going either way. A random distribution of dots is indicative of a velocity field that yields chaotic motion and regular structures on the cross section are indicative of regular particle motion. For  $U_1$  equal to zero and unity, the flow field is two-dimensional and the Poincaré sections are regular. For values of  $U_1$  between 0 and about 0.3, the particle paths are predominantly chaotic as can be seen in figure 5.2. For higher values of  $U_1$ , regular particle paths appear and at  $U_1 = 0.75$  the particle paths are predominantly regular. It is thus expected that scalar mixing is more efficient for fully chaotic particle motions.

Another approach to determine if a velocity field yields chaotic motion is to consider pathlines of particles that are initially closely spaced. Figure 5.3 shows the pathlines of three anisotropic particles close together at  $t = 0$  (separated by 0.01)

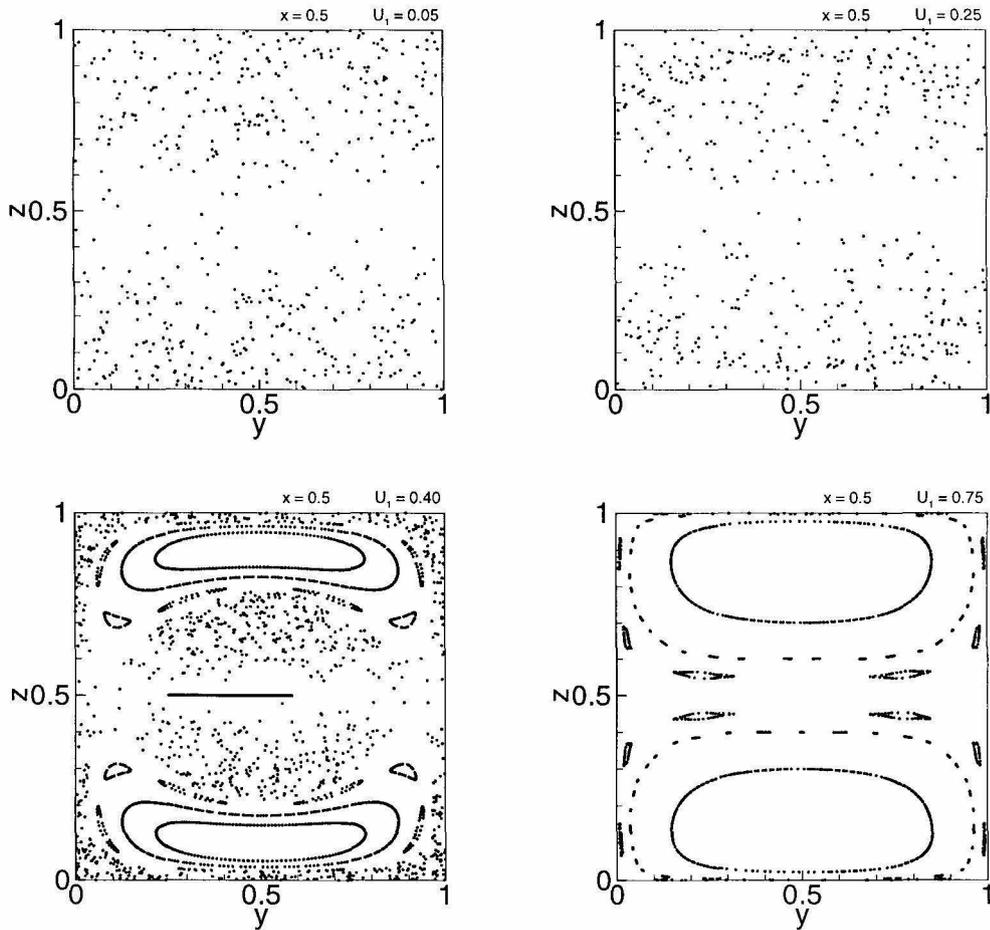


FIGURE 5.2: Poincaré sections at  $x = 0.5$  for  $U_1 = 0.05, 0.25, 0.4,$  and  $0.75$  of the velocity field given by (5.1).

up to  $t = 40$ . We used (3.4) to move particles in time using the velocity field (5.1). The filtering constant  $\sigma$  and the core size of the particles were set equal to 0.15. The diffusivity  $\kappa$  was chosen to be  $10^{-3}$ . Note that the diffusivity has influence on the shape of the particles and thus the location of the particles, since they are determined by a weighted averaged value over the shape function.

It is clear that the initial condition has a strong influence on the location of a particle at subsequent times. This also implies that convergence of the scalar field is difficult to obtain. Increasing the number of particles or changing the initial location of particles can lead to totally different fields at long times. However, in order to assess mixing, we will be interested in statistical quantities, for which it is easier to

establish convergence. In particular, the root mean square of the scalar field  $\tilde{\psi}(t)$  as defined by (2.9) will be used.

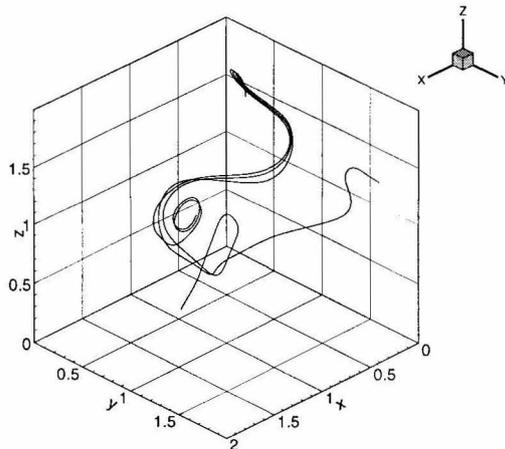


FIGURE 5.3: Pathlines for three different initial conditions using anisotropic particles with  $\sigma = \delta_k = 0.15$ ,  $\kappa = 10^{-3}$ , and  $U_1 = 0.25$  for  $t \in [0..40]$ .

## 5.2 Numerical Results

### 5.2.1 Gaussian Initial Field

Since we have an analytical expression for the unfiltered velocity field, we can use this field to march particles forward in time as explained at the end of the previous chapter. As an example calculation, we used a periodic domain between 0 and 1 in all three spatial directions, each direction containing 25 axisymmetric particles giving a total of 15625. This results in an initial grid spacing of 0.04. The filtering constant was set equal to  $\sigma = 0.15$ , as was the core size  $\delta_k$ , resulting in an overlap parameter of  $\delta_k/h = 3.75$ . Diffusivity was chosen to be  $\kappa = 10^{-4}$ . A Gaussian centered at  $(x, y, z) = (0.8, 0.7, 1.1)$  with a core size (in the unfiltered space)  $\tau = 0.3$  was used as the initial condition.

Results are given for times 0 through 8 with increments of 1 in figures 5.4a-c. The plots in the left column show the evolution of the contour surfaces 0.03 (green) and 0.06 (blue). The right column shows contour plots of the  $x = 1$  surface (which is

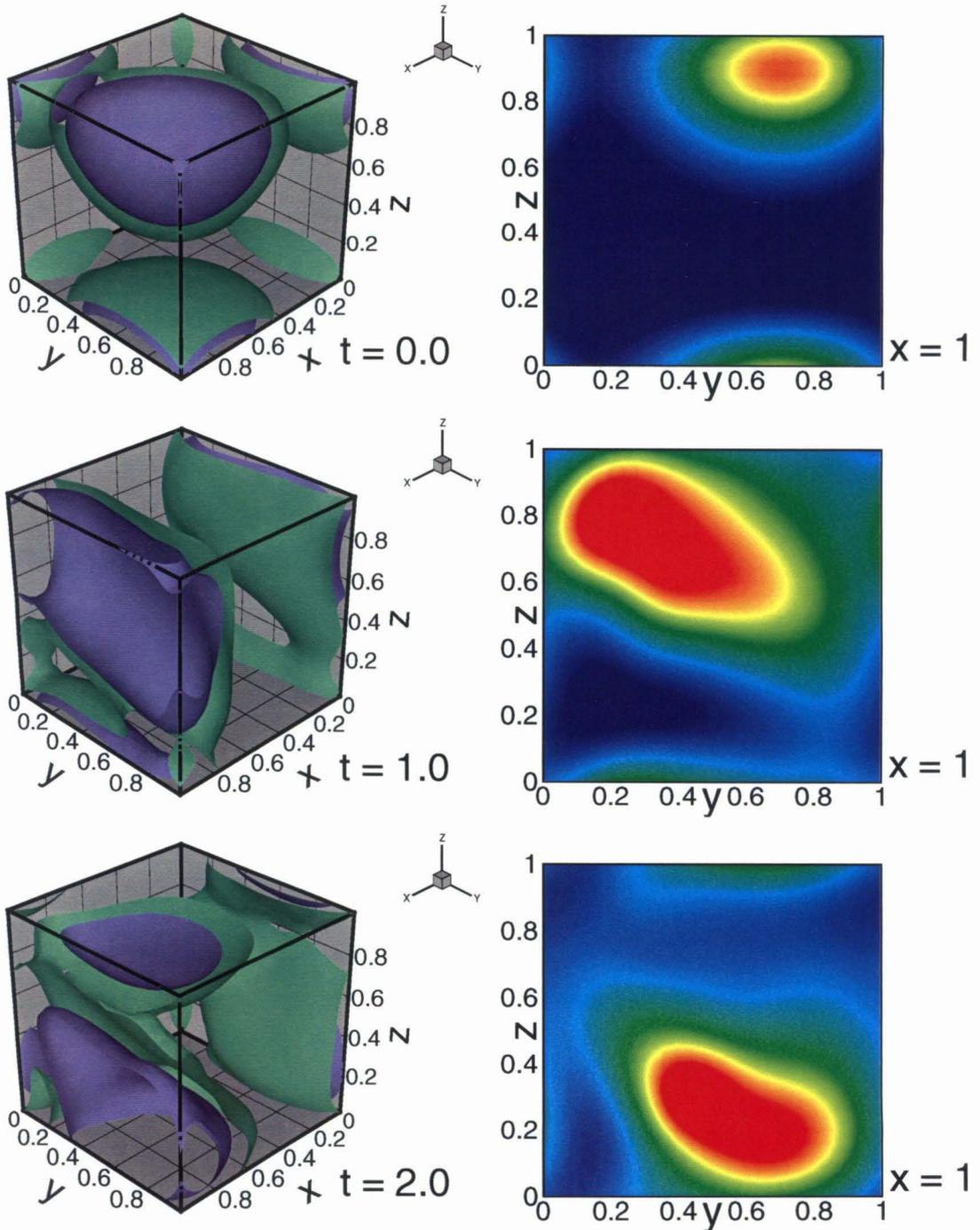


FIGURE 5.4A: Example calculation with Gaussian initial field showing a surface plot (green - 0.03; blue - 0.06) and a contour plot at subsequent times. 15625 axisymmetric particles were used,  $\sigma = \delta_k = 0.15$  and  $\kappa = 10^{-4}$ .

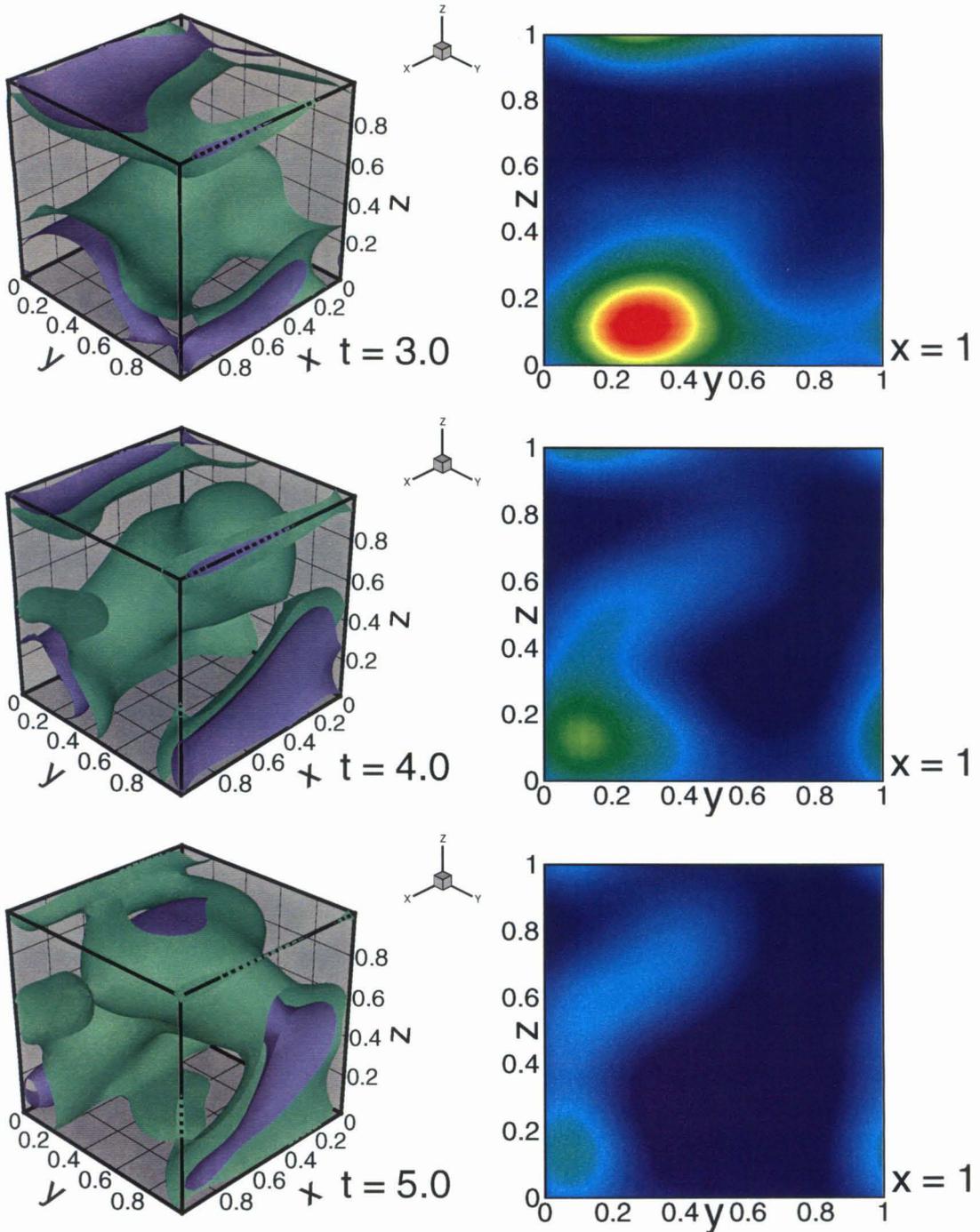


FIGURE 5.4B: Example calculation with Gaussian initial field showing a surface plot (green - 0.03; blue - 0.06) and a contour plot at subsequent times. 15625 axisymmetric particles were used,  $\sigma = \delta_k = 0.15$  and  $\kappa = 10^{-4}$ .

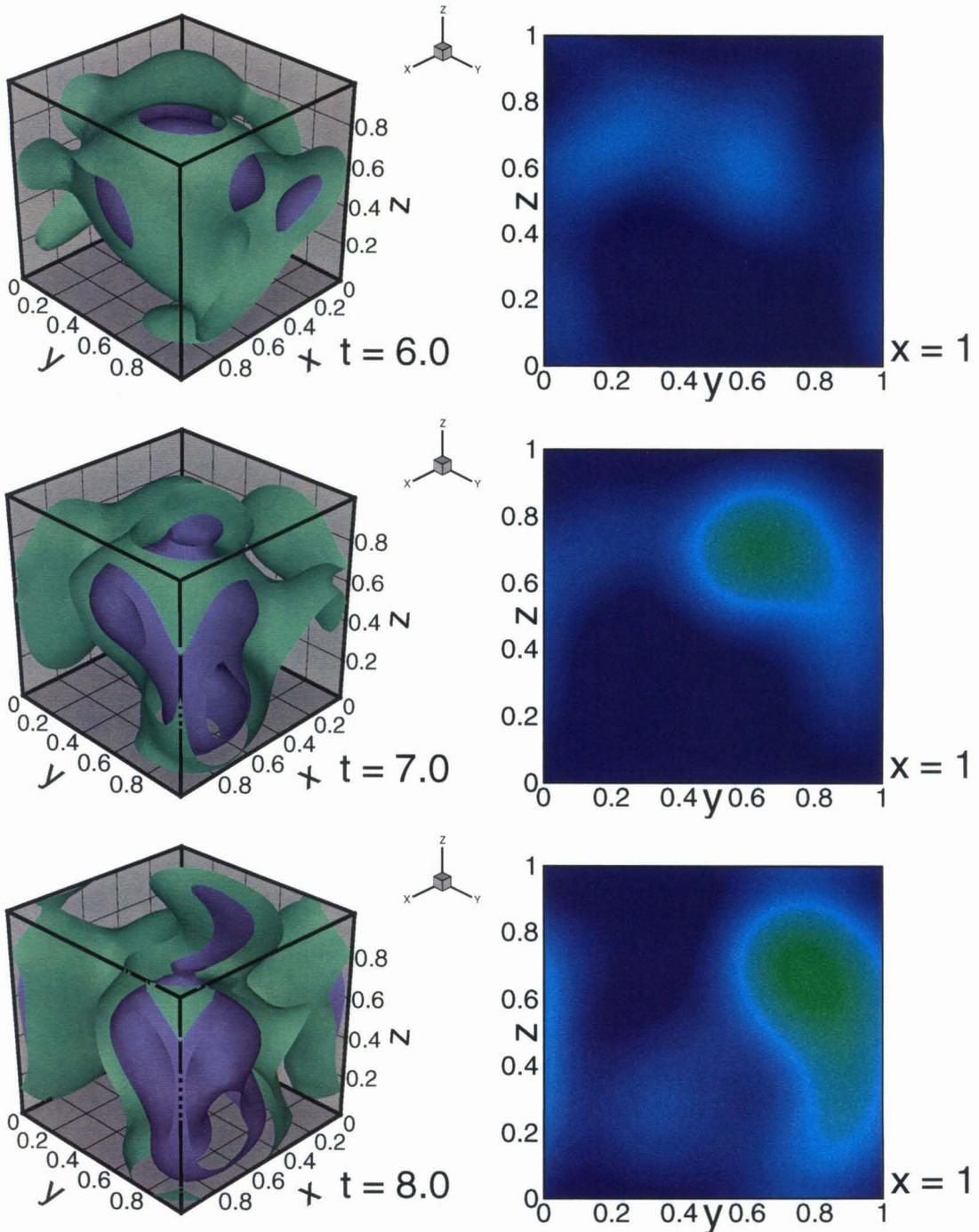


FIGURE 5.4C: Example calculation with Gaussian initial field showing a surface plot (green - 0.03; blue - 0.06) and a contour plot at subsequent times. 15625 axisymmetric particles were used,  $\sigma = \delta_k = 0.15$  and  $\kappa = 10^{-4}$ .

identical to the  $x = 0$  surface). One can clearly see the mixing process taking place. Note that the contour surfaces remain fairly smooth. This is due to the filtering. If we want to see more detail, we need to lower the value of  $\sigma$  and increase the number of particles, thereby also increasing the computational costs.

### 5.2.2 Random Initial Field

Toussaint et al. were interested in the mixing of a random initial scalar field for the velocity field given in (5.1) for different values of  $U_1$ , where  $U_2$  follows from (5.2). They used a spectral method to solve the scalar advection-diffusion equation and chose  $\kappa = 10^{-4}$  in most of their computations, which had resolutions up to  $48^3$  spectral modes in total. As the velocity and length scales are of order unity, the Péclet number is  $10^4$ . It is expected that for this high Péclet number, the scalar field is not fully resolved by using only  $48^3$  modes. However, Toussaint et al. do not use a subgrid model and only show that the root mean square value of the scalar field  $\tilde{\psi}$ , as defined in (2.9), appears to be close to convergence for  $48^3$  modes, a less stringent requirement than convergence of the scalar field.

The initial field of Toussaint et al. consisted of a random combination of sinusoidal functions. Unfortunately, they did not give sufficient detail for this research to reproduce their field exactly. They track the root mean square value  $\tilde{\psi}$  as a function of time. They show in their figure 7 that after the transient effects have died out,  $\tilde{\psi}$  decays on a logarithmic scale linearly in time. For different values of the velocity parameter  $U_1$ , they have plotted the slope of this decay (the mean rate of decay) in their figure 8. This figure shows that there is an optimal velocity parameter for which mixing is most efficient close to  $U_1 = 0.25$ .

This section will try to reproduce these results using a subgrid model. As the unfiltered velocity field is known, we will choose to move axisymmetric particles with this unfiltered field. The filtering constant  $\sigma$  and (initial) core size  $\delta_k$  of the particles was set equal to 0.15. We used 25 particles in each spatial direction between 0 and 1 giving a total of 15,625 particles. This leads to a grid spacing of 0.04 and an overlap

parameter of 3.75. Just as in the computations by Toussaint et al. periodic boundary conditions were used. Particles that move through a boundary surface reappear on the other side of the cube.

The initial condition of Toussaint et al. is not fully known in detail as mentioned above. We used (in the filtered space) the sum of the sinusoidal functions  $a_{n,i} \sin(n\pi x_i)$  for  $n = 2, 4, 6,$  and  $8,$  and  $x_i$  any of the three spatial dimensions. A random amplitude  $a_{n,i}$  between 0 and 1 was chosen for each mode.

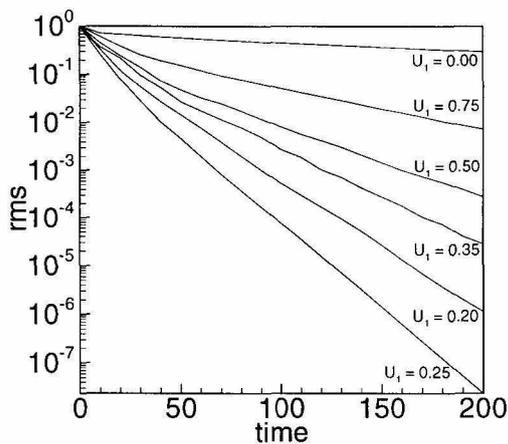


FIGURE 5.5: Root mean square of the scalar field  $\tilde{\psi}_u$  as a function of time for different values of the parameter  $U_1$ . The solution was remeshed after 10 time units and  $\delta_k = \sigma = 0.15$ . The unfiltered velocity field subgrid model was used.

The root mean square of the scalar field we compute will be the one of the filtered field using  $\sigma = 0.15$ , which is not equal to the one of the unfiltered field. Denote the root mean square value of the unfiltered field by  $\tilde{\psi}_u$ , which is given by

$$\tilde{\psi}_u^2 = \int_{\Omega} \psi^2 d^3 \mathbf{x} = \int_{\Omega} \widehat{\psi}^2 d^3 \mathbf{x}, \quad (5.3)$$

The root mean square value we compute is the integral over  $\widehat{\psi}^2$ , which is not equal to the one over  $\tilde{\psi}^2$ . By using (2.5), we can find the following approximation for  $\tilde{\psi}_u$

$$\tilde{\psi}_u^2 \approx \int_{\Omega} \widehat{\psi}^2 + \frac{\sigma^2}{2} \frac{\partial \widehat{\psi}}{\partial x_i} \frac{\partial \widehat{\psi}}{\partial x_i} d^3 \mathbf{x}. \quad (5.4)$$

Computations have been performed for a large range of values of  $U_1$ . Every 10 time units the remeshing procedure was applied. As stated before, we used axisymmetric particles that move in time with the unfiltered velocity field (5.1). Figure 5.5 shows  $\tilde{\psi}_u$  (as calculated by (5.4)) as a function of time for different values of  $U_1$ . The results have been normalized using the value  $\tilde{\psi}_u$  at  $t = 0$ .

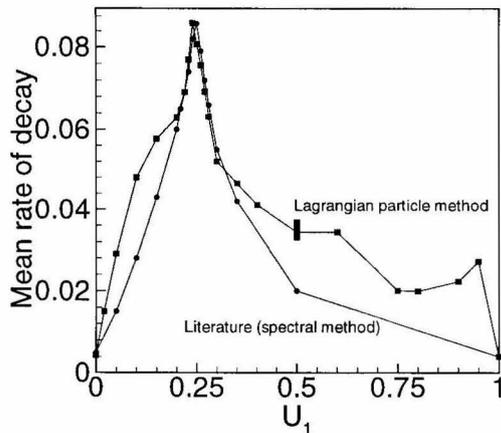


FIGURE 5.6: Mean rate of decay of  $\tilde{\psi}_u$  as a function of  $U_1$ . Spectral method results are from Toussaint et al. [47]. The particle method used the unfiltered velocity field subgrid model, axisymmetric particles and was remeshed every 10 time units.  $\kappa = 0.0001$  and  $\delta_k = \sigma = 0.15$ .

After an initial step decay,  $\tilde{\psi}_u$  appears to decay linearly in time on a logarithmic scale. We have plotted the slope of this line (using  $t > 50$ ) in figure 5.6 for different values of  $U_1$ . For  $U_1 = 0.5$ , we have performed several runs using different initial values for  $a_{n,i}$  to get a feel for the sensitivity to initial conditions. The vertical bar gives the range of decay rates we obtained. The data points obtained by Toussaint et al. have been plotted as well. Very good agreement is obtained for values of  $U_1$  between 0.2 and 0.3. For these values, the velocity field yields chaotic particle paths for almost all particles. The fully 2D limit cases,  $U_1$  is 0 and 1, are also in good agreement.

There is substantial disagreement for values of  $U_1$  above 0.4. Unfortunately, Toussaint et al. do not provide a lot of data points in that regime. There are several possibilities for this discrepancy. First of all, the velocity field has regions where the

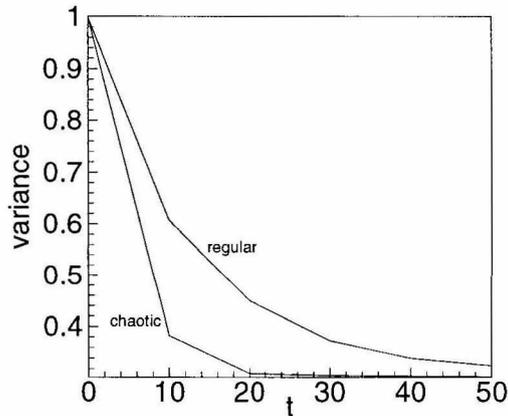


FIGURE 5.7: Variance as a function for time an initial Gaussian blob in an area that yields regular advection and one in an area that yields chaotic advection for  $U_1 = 0.4$  using the unfiltered velocity field subgrid model and  $\sigma = 0.15$ .

particle paths are regular and other regions where the motion is chaotic. By performing two tests using a Gaussian initial blob, one initially in the region where the particle paths are regular and one in the region where they are chaotic, it is shown in figure 5.7 that mixing is considerably faster for the blob in the chaotic region. Hence, the mean rate of decay depends on the specific initial condition used. Higher rates of convergence are obtained if more scalar is present initially in the chaotic regions. Second, Toussaint et al. have only one data point between  $U_1 = 0.35$  and  $U_1 = 1.0$ , the area where the results differ most. Third, our research solves the filtered advection-diffusion equation, whereas Toussaint and et al.'s solve the unfiltered equation directly. It is unknown what effect the filtering has on a velocity field with areas with regular and chaotic path lines. Finally, the effect of large Péclet numbers using a coarse grid without subgrid model (as Toussaint et al. did) is also not known.

## Chapter 6 Forced Turbulence

This chapter discusses results for scalar mixing in a forced turbulent velocity field in a periodic 3D domain. The turbulence is computed using a de-aliased spectral method capable of computing the scalar field as well.

The first section in this chapter discusses the forced turbulence code followed by a short section about the numerical implementation of the particle method. Comparisons between both methods for different Schmidt numbers are made.

### 6.1 Forced Isotropic Turbulence

Incompressible forced isotropic turbulence in a 3D periodic box of sides  $2\pi$  is considered. This flow is solved using a spectral code provided by D.I. Pullin and his research group. For more information about this code the reader is referred to [33, 34, 37]. The Fourier-Galerkin pseudo-spectral code uses a 3/2 de-aliasing rule. A second-order explicit Runge-Kutta scheme is used to march forward in time.

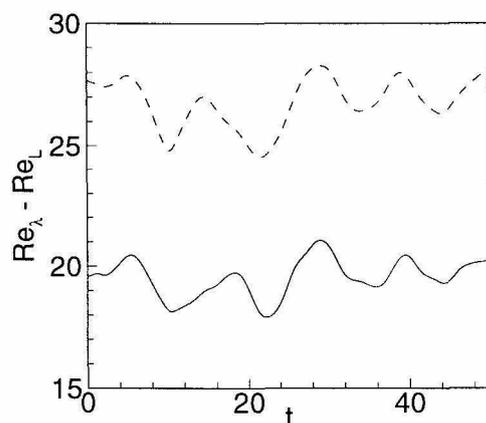


FIGURE 6.1:  $Re_\lambda$  (solid) and  $Re_L$  (dashed) as function of time for a  $32^3$  calculation. A de-aliased spectral method was used.

Computations have been performed using 32 cubed data points. An initial velocity field was run for about 20 eddy turn-over times to remove any transient results from the initial velocity field. Figure 6.1 shows the Taylor's Reynolds number  $Re_\lambda$  and the Reynolds number based on the integral length scale  $L Re_L$  as a function of time. The average  $Re_\lambda$  is 19.3 and the average  $Re_L$  is 26.4. The average Kolmogorov scale  $\eta$  is  $\eta = 0.0846L$ . The maximum wave number  $k_{\max}$  times the Kolmogorov scale  $\eta$  is 1.83 indicating that the velocity field was fully resolved in this computation. The average eddy turn-over time is 2.96.

## 6.2 Numerics

Up till now all numerical examples have used an analytic velocity field. The velocity at every particle location was known. In this example, the velocity is only known at a set of discrete grid points. A quadratic interpolation scheme using 3 grid points in each spatial direction was used to approximate the velocity between neighboring grid points.

This interpolation scheme was tested for the 3D stationary velocity field as discussed in the previous chapter. Good agreement between the interpolation scheme and the analytical expressions was found using  $64^3$  data points.

We chose the time step to be 0.025. First, the velocity field is updated using a second order explicit Runge Kutta method. Simultaneously, the scalar field is computed using a spectral method (without a subgrid model). After each time step, the new velocity field is used to update the location and the size of the particles.

A Gaussian with core size 1.5 is used as initial condition for the scalar field for the computation using the spectral method. For the particle method computation, the initial field is first filtered with  $\sigma = 0.30$ . The core size of the particles is set equal 0.30 and in total  $32^3$  particles are used. Since the original velocity field is known, particles will be marched forward in time using this unfiltered velocity field. Thus we do not use the tensor-diffusivity subgrid model in this chapter.

### 6.3 Scalar Mixing

For a Schmidt number (the ratio of viscosity over diffusivity) equal to one,  $Sc = 1$ , the spectral method fully resolves the scalar field. One can not compare the scalar spectra of the spectral method and particle method directly. The particle method solved the scalar field after applying a Gaussian filter with  $\sigma = 0.3$ . In Fourier space, this means that each wave number was multiplied by  $\exp\left(-\frac{k^2\sigma^2}{4}\right)$ . We have multiplied the scalar spectrum in wave number  $k$  with  $\exp\left(\frac{k^2\sigma^2}{t}\right)$  to better compare the results of the particle method with the spectral method. The results for times 0, 3, and 6 are given in figure 6.2.

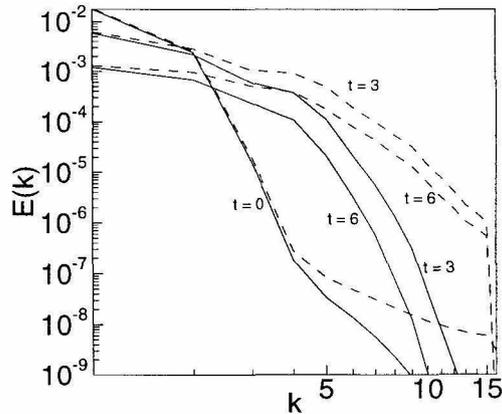


FIGURE 6.2: Scalar spectra at three different times for the particle method (solid) and spectral method (dashed) at  $Re_\lambda = 19.3$  and  $Sc = 1$ . The results for the particle method have been multiplied with  $\exp\left(\frac{k^2\sigma^2}{t}\right)$  for comparison reasons.

Due to mixing, scalar variances are transformed from the lower wave numbers to the higher wave numbers where they dissipate as can be seen in the figure. At  $t = 0$ , there is good agreement between the scalar spectra, except for the high wave numbers. At  $t = 3$  and 6, the spectra do not compare very well except for the very low wave numbers, which determines the large scale structure of the scalar field. The particle method predicts a spectrum that drops off faster than the spectral method for high wave numbers.

Another way of comparing the results is to look at slices cut through the 3D periodic box. Figures 6.3 and 6.4 consider a cut at  $x = \pi$  at times 2 and 10. Contourlines for three different calculations are visible. Solid lines indicate the spectral method. The results of the particle method are dotted. As expected, the dotted lines have less small scale structure than the solid lines. Since the scalar advection-diffusion equation and the filtering operator commute, we can filter the results of the spectral method and compare with the particle method. These filtered results are dashed.

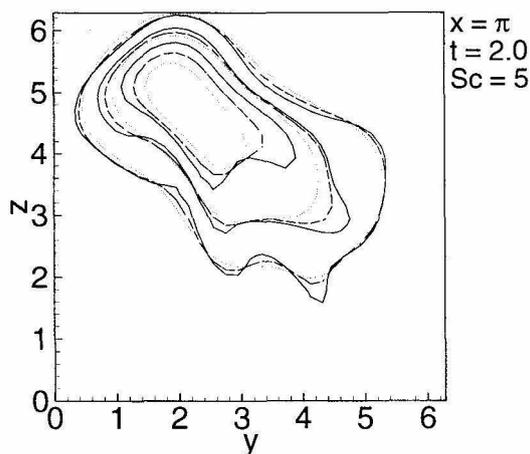


FIGURE 6.3: Scalar contour lines for the spectral method (solid), filtered spectral method (dashed) and particle method (dotted). Contour levels are 0.2, 0.4, and 0.6 and  $\sigma = 0.30$ . The unfiltered velocity field subgrid model was used for the particle method results.

The solution obtained using a good subgrid model is expected to reproduce these dashed lines accurately. One can see that there is fairly good agreement between the filtered spectral method data and the particle method at  $t = 2$ . Also at larger times, there is good agreement between the large scale flow structures.

For long time, it is expected that the scalar field will reach a constant value everywhere, say  $\psi_c$ . We integrate the square of the difference between the local value of the scalar field and  $\psi_c$  over the box. The square root of this integral is the variance or root mean square value of the scalar field,  $\tilde{\psi}$ . It is expected that this value tends to zero for long times. The variance as a function of time is plotted in figure 6.5. The initial value of the variance for the spectral and particle method are not identical as

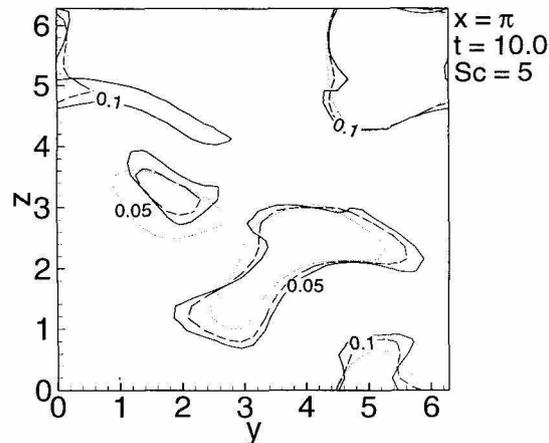


FIGURE 6.4: Scalar contour lines for the spectral method (solid), filtered spectral method (dashed) and particle method (dotted) for  $\sigma = 0.15$ . The unfiltered velocity field subgrid model was used for the particle method results.

the initial fields are not (the one for the particle method is a filtered field). But, they both predict the same rate of mixing (same slope).

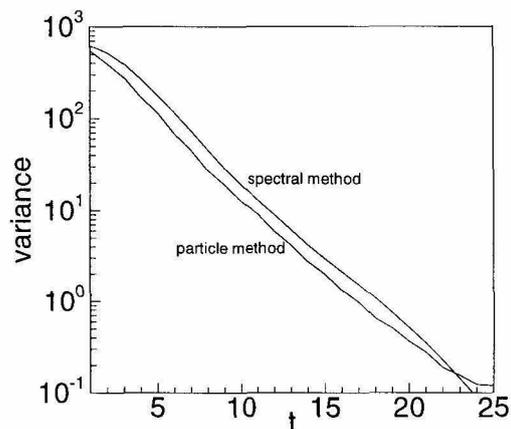


FIGURE 6.5: Scalar variance as a function of time for the spectral and particle method at  $Re_\lambda = 19.3$  and  $Sc = 1$ . The results for the particle method are filtered results using  $\sigma = 0.30$ , whereas the ones for the spectral method are not filtered, hence the difference in initial condition. The slopes indicates similar rates of mixing.

The slope for the particle method flattens out after  $t = 22$ . At this time, the scalar field is almost fully mixed and the particle method is about to break down. At these long times, the scalar field can be viewed as a small perturbation on top of an

averaged value. Represented this scalar field in terms of Gaussian particles is hard. The particle method can not capture the small difference on top of the averaged field very well. If we had used an initial scalar field with zero mean, the particle method will be able to reach longer times.

A comparison between the particle method and the filtered results of the spectral method are given in figures 6.7a-c. Each plot shows, at a different time, the contour surface 0.15 and contour plots between 0 and 0.30 on three sides of the cube. In the limit of long time, the scalar field will reach 0.075 everywhere. As can be seen, good agreement is obtained between the filtered data and the subgrid model data.

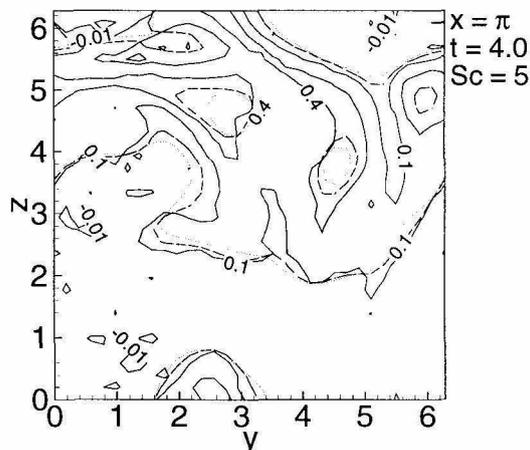


FIGURE 6.6: Scalar contour lines for the spectral method (solid), filtered spectral method (dashed) and particle method (dotted). Filtering constant was  $\sigma = 0.15$ . The unfiltered velocity field subgrid model was used for the particle method results, which was remeshed every time unit.

A comparison for  $Sc = 5$  at  $t = 4$  is made in figure 6.6. At this Schmidt number, the spectral method is about to break down. The smallest scale in the scalar field is the Batchelor scale  $\eta_B$ . For large Schmidt numbers, we can write [46]

$$\eta_B k_{\max} = \eta k_{\max} Sc^{-1/2}, \quad (6.1)$$

where  $\eta_B$  has been non-dimensionalized with the maximum wave number  $k_{\max}$ . For  $Sc = 5$ , this computation leads to  $\eta_B k_{\max} = 0.82$ , which is smaller than one indicating

that the scalar field is not fully resolved. Figure 6.6 shows that there is still good agreement for the large scale structures in the flow; however, there are areas where the scalar is negative. For larger values of the Schmidt number, the areas where the scalar function is negative become larger.

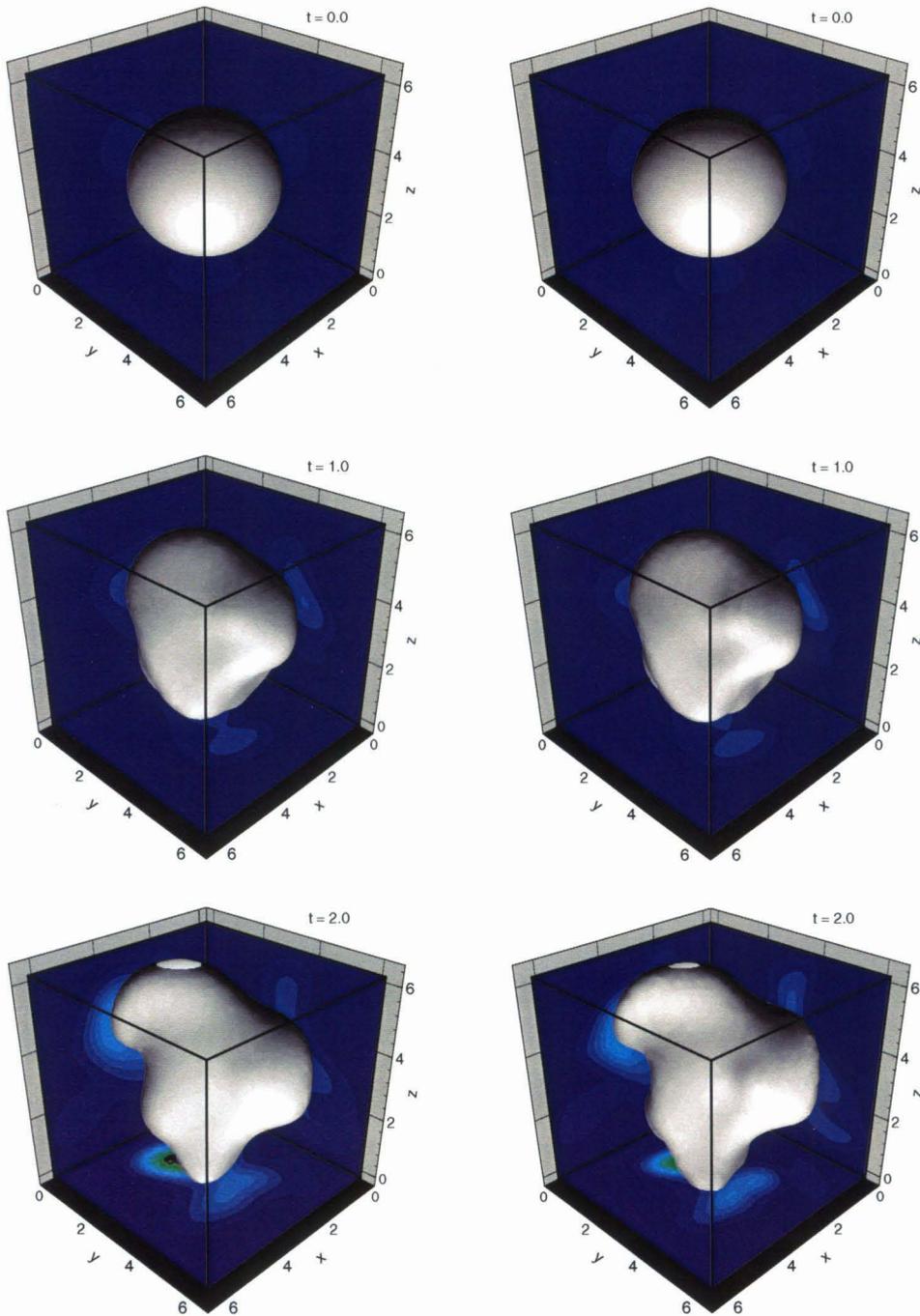


FIGURE 6.7A: Scalar mixing in a turbulent field for  $Re_\lambda = 19.3$ ,  $Sc = 1$ , and  $\sigma = 0.3$  at subsequent times for the particle method (left column) and filtered spectral method (right column). Contour surface is 0.15 and contour levels vary from 0 to 0.30. The particle method uses the unfiltered velocity field subgrid model and was remeshed every time unit.

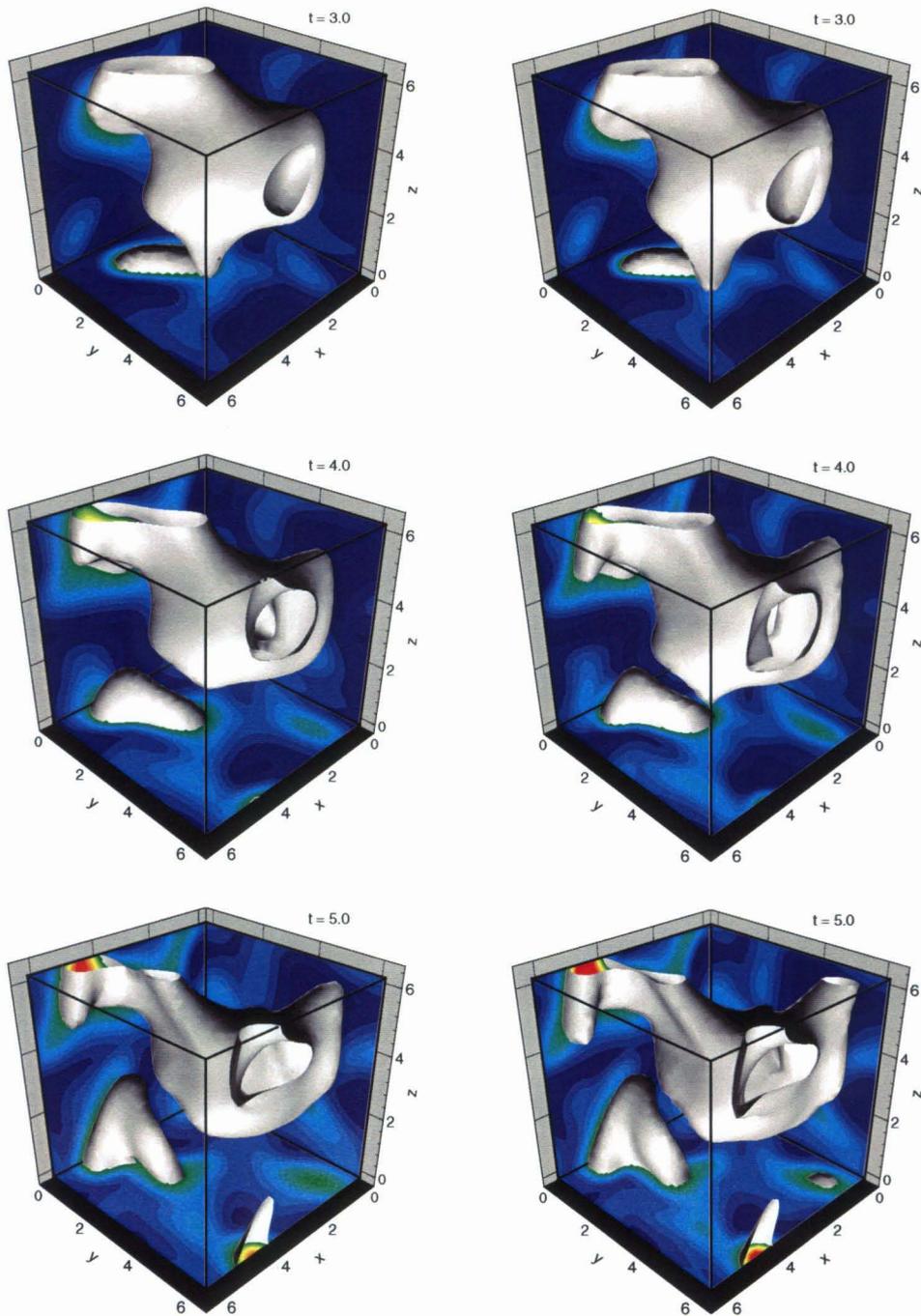


FIGURE 6.7B: Scalar mixing in a turbulent field for  $Re_\lambda = 19.3$ ,  $Sc = 1$ , and  $\sigma = 0.3$  at subsequent times for the particle method (left column) and filtered spectral method (right column). Contour surface is 0.15 and contour levels vary from 0 to 0.30. The particle method uses the unfiltered velocity field subgrid model and was remeshed every time unit.

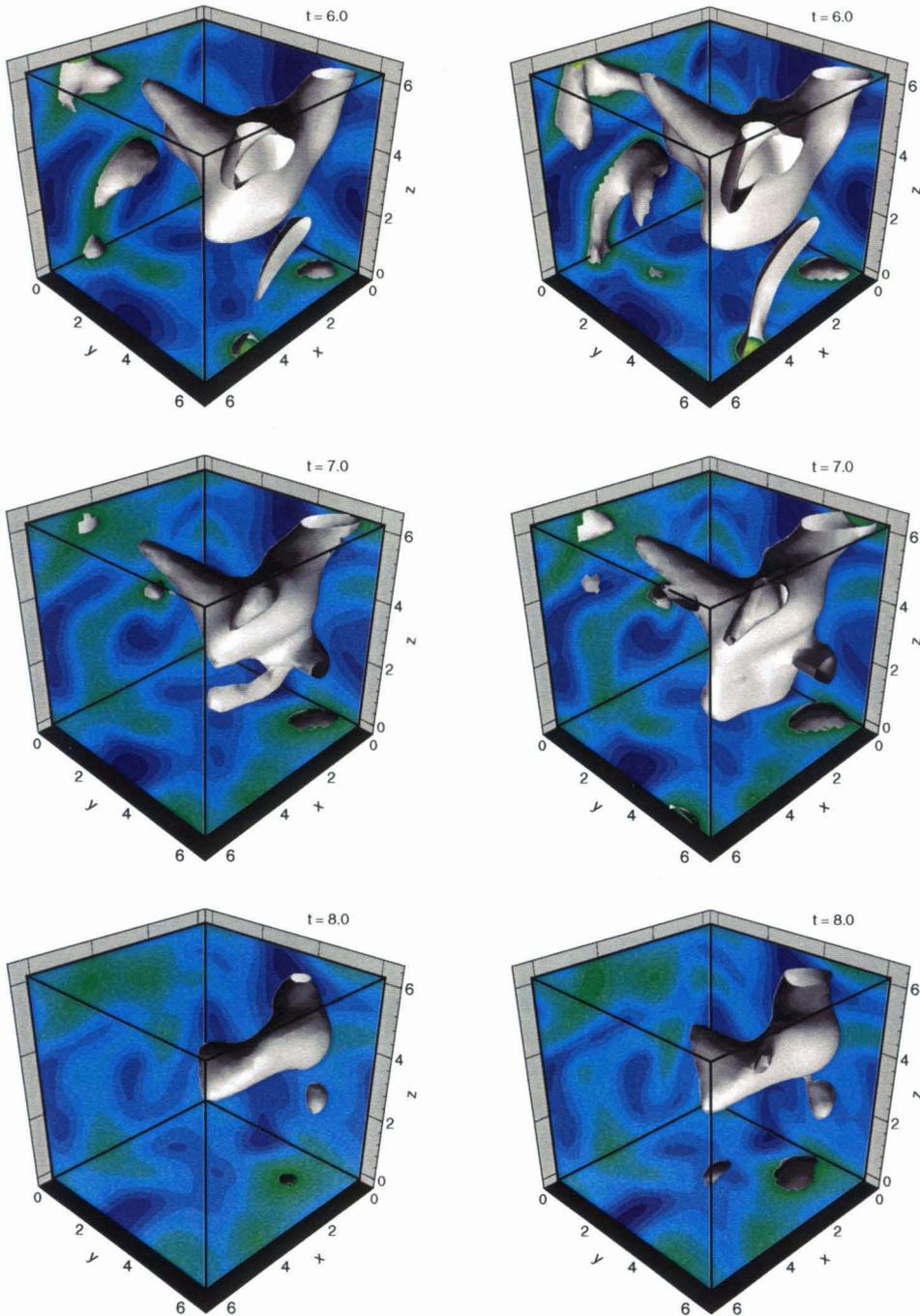


FIGURE 6.7C: Scalar mixing in a turbulent field for  $Re_\lambda = 19.3$ ,  $Sc = 1$ , and  $\sigma = 0.3$  at subsequent times for the particle method (left column) and filtered spectral method (right column). Contour surface is 0.15 and contour levels vary from 0 to 0.30. The particle method uses the unfiltered velocity field subgrid model and was remeshed every time unit.

## Chapter 7 Conclusions/Summary

Summarizing this research,

- By using a Gaussian filter, an (exact) infinite series expansion was found to close the filtered incompressible scalar advection-diffusion equation. This procedure is also possible for similar filters (like the top-hat filter) and different equations (like Navier-Stokes).
- The tensor-diffusivity subgrid model is obtained by retaining only the first to terms in this expansion. The model can be interpreted as a tensor-diffusivity term proportional to the rate-of-strain tensor of the filtered velocity field.
- The tensor-diffusivity subgrid model is material frame indifferent and allows for backscatter. In the stretching directions, it exhibits negative diffusion. This calls for regularization of the numerical method. A straightforward finite-difference or spectral code will lead to growing instabilities.
- The scalar field was decomposed in a set of anisotropic or axisymmetric Gaussian particles, each well behaved for large wave numbers, thereby providing the necessary regularization. Equations of motion for the location and shape/size of the particles were derived using an expansion in Hermite polynomials. Diffusivity was solved using the core expansion method.
- To prevent large core sizes or very elliptical particles, at certain times the old set of particles is replaced by a new set of particles, each particle again small and axisymmetric. An explicit expression was found for the amplitudes of the new set using a least square error approximation.
- Both the tensor-diffusivity subgrid model and the anisotropic particles are exact solutions for a stagnation flow. This flow was used to validate and illustrate several aspects of the particle model and method.

- A 2D time-dependent periodic velocity field yielding chaotic advection was used extensively. Comparison with filtered DNS data yields good agreement for both the axisymmetric and anisotropic particles. The tensor-diffusivity model results for this flow in more accurate results than a Smagorinsky subgrid model.
- For large values of the filtering constant, the subgrid model breaks down. More terms in the infinite expansion have to be retained to maintain good accuracy.
- Anisotropic particles achieve higher accuracy than axisymmetric particles for the time integration. This is offset by better results for the remeshing procedure for axisymmetric particles. Due to computational efficiency, axisymmetric particles are the preferred choice.
- A 3D stationary chaotic one-parameter velocity-field was used to study the effects of mixing and compare our results with a literature study using a spectral method and small values of the diffusivity. Similar rates of mixing were obtained for fully chaotic velocity fields.
- A 3D forced isotropic periodic turbulence code was used to compare our results with a spectral method and literature.
- When the unfiltered velocity field is known, the best results are obtained by moving particles using this field. It is therefore concluded that a good subgrid model modifies the equation of motion to get a good approximation to the unfiltered velocity field.

	td model		unfilt. vel. field	
	$\sigma \neq 0$	$\sigma = 0$	$\sigma \neq 0$	$\sigma = 0$
axisymmetric	×		×	n/a
anisotropic	×	×		n/a

TABLE 7.1: Overview of the different test runs performed using axisymmetric and anisotropic particles and the tensor-diffusivity or unfiltered velocity field model.

Table 7.1 gives an overview of the different calculations that were carried out using the Lagrangian particle method and the different subgrid models. There was a

choice between using anisotropic and axisymmetric particles and two different subgrid models have been tested: tensor-diffusivity and the unfiltered velocity field model. For the tensor-diffusivity model calculations were performed using both axisymmetric and anisotropic particles in chapter 4. For the anisotropic particles, computations with the subgrid model turned off have been performed as well. See for example the left column of figure 4.8 and the right column of figure 4.9. The unfiltered velocity subgrid model can not be turned off. Only axisymmetric particles were used for this subgrid model. For this model to work, the unfiltered velocity field has to be known. All the computations in chapters 5 and 6 used the unfiltered velocity subgrid model and axisymmetric Gaussian particles.

## Bibliography

- [1] M. Abramowitz and I.A. Stegun. *Handbook of Mathematical Functions*. Dover Publications, Inc., New York, 1972.
- [2] N.A. Adams and A. Leonard. Deconvolution of subgrid-scales for the simulation of shock-turbulence interaction. In P.R. Voke, N.D. Sandham, and L. Kleiser, editors, *Direct and Large-Eddy Simulation III*, pages 201–212. Kluwer Academic Publishers, 1999.
- [3] J.J. Bardina, J.H. Ferziger, and W.C. Reynolds. Improved turbulence models based on large eddy simulation of homogeneous, incompressible, turbulent flows. Technical Report TF-19, Stanford University, California, 1983.
- [4] 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(6):1414–1439, 1993.
- [5] K.W. Bedford and W.K. Yeo. Conjunctive filtering procedures in surface water flow and transport. In Boris Galperin and Steven A. Orszag, editors, *Large Eddy Simulation of Complex Engineering and Geophysical Flows*, pages 513–537. Cambridge University Press, 1993.
- [6] D. Bogucki, J.A. Domaradzki, and P.K. Yeung. Direct numerical simulations of passive scalars with  $Pr > 1$  advected by turbulent flow. *J. Fluid Mech.*, 343:111–130, 1997.
- [7] T. Bohr, M.H. Jensen, G. Paladin, and A. Vulpiani. *Dynamical Systems Approach to Turbulence*. Cambridge University Press, 1998.

- [8] G. Brethouwer and F.T.M. Nieuwstadt. Mixing of weakly and strongly diffusive passive scalars in isotropic turbulence. In P.R. Voke, N.D. Sandham, and L. Kleiser, editors, *Direct and Large-Eddy Simulation III*, pages 311–322. Kluwer Academic Publishers, 1999.
- [9] D. Carati, G.S. Winckelmans, and H. Jeanmart. Exact expansions for filtered-scales modelling with a wide class of les filters. In P.R. Voke, N.D. Sandham, and L. Kleiser, editors, *Direct and Large-Eddy Simulation III*, pages 213–224, Kluwer Academic Publishers, 1999.
- [10] A.J. Chorin. Numerical study of slightly viscous flow. *J. Fluid Mech.*, 57(4):785–796, 1973.
- [11] G. Comte-Bellot and S. Corrsin. Simple Eulerian time correlation of full and narrow-band velocity signals in grid-generated, 'isotropic' turbulence. *J. Fluid Mech.*, 48(2):273–337, 1971.
- [12] G.-H. Cottet and P. Koumoutsakos. *Vortex Methods: Theory and Applications*. Cambridge University Press, 1999.
- [13] J.A. Domaradzki and E.M. Saiki. A subgrid-scale model based on the estimation of unresolved scales of turbulence. *Phys. Fluids*, 9:2148–2164, 1997.
- [14] J.H. Ferziger and M. Perić. *Computational Methods for Fluid Dynamics*. Springer, 1996.
- [15] C. Fureby. On subgrid scale modeling in large eddy simulations of compressible flow. *Phys. Fluids*, 8(5):1301–1311, May 1996.
- [16] B. Geurts. Inverse modeling for large-eddy simulation. *Phys. Fluids*, 9:3585–3587, 1997.
- [17] C.H. Gibson and W.H. Schwarz. The universal equilibrium spectra of turbulent velocity and scalar fields. *J. Fluid Mech.*, 16(3):365–384, 1963.

- [18] C. Greengard. The core spreading vortex method approximates the wrong equation. *J. Comput. Phys.*, 61:345–348, 1985.
- [19] M.E. Gurtin. *An Introduction to Continuum Mechanics*. Academic Press, 1981.
- [20] R.W. Hockney and J.W. Eastwood. *Computer Simulation using Particles*. Adam Hilger, 1988.
- [21] M.-J. Huang. *Theoretical and Computational Studies of Isotropic Homogeneous Turbulence*. Ph.D. thesis, California Institute of Technology, 1994.
- [22] Robert M. Kerr. Higher-order derivative correlations and the alignment of small-scale structures in isotropic numerical turbulence. *J. Fluid Mech.*, 153:31–58, 1985.
- [23] Robert M. Kerr. Velocity, scalar and transfer spectra in numerical turbulence. *J. Fluid Mech.*, 211:309–332, 1990.
- [24] R. Krasny. A study of singularity formation in a vortex sheet by the point vortex approximation. *J. Fluid Mech.*, 167:65–93, 1986.
- [25] L. Landau and E. Lifshitz. *Fluid Mechanics*. Pergamon Press, 1959.
- [26] A. Leonard. Private discussion, 1997.
- [27] A. Leonard. Energy cascade in large eddy simulations of turbulent fluid flows. *Adv. Geophys.*, 18A:237–248, 1974.
- [28] A. Leonard. Vortex methods for flow simulation. *J. Comput. Phys.*, 37(3):289–335, October 1980.
- [29] A. Leonard. Large-eddy simulation of chaotic convection and beyond. *AIAA 97-0204*, 1997.
- [30] A. Leonard and G.S. Winckelmans. A tensor-diffusivity subgrid model for large-eddy simulation. In P.R. Voke, N.D. Sandham, and L. Kleiser, editors, *Direct*

- and Large-Eddy Simulation III*, pages 147–162, Kluwer Academic Publishers, 1999.
- [31] M. Lesieur and O. Métais. New trends in large-eddy simulations of turbulence. *Annu. Rev. Fluid. Mech.*, 28:45–82, 1996.
- [32] S. Liu, C. Meneveau, and J. Katz. On the properties of similarity subgrid-scale models as deduced from measurements in a turbulent jet. *J. Fluid Mech.*, 275:83–119, 1994.
- [33] A. Misra. *Large-Eddy Simulation using a Vortex-Based Subgrid Stress Model*. Ph.D. thesis, California Institute of Technology, 1998.
- [34] A. Misra and D.I. Pullin. A vortex-based subgrid stress model for large-eddy simulation. *Phys. Fluids*, 9(8):2443–2454, August 1997.
- [35] P. Moeleker and A. Leonard. Lagrangian methods for the tensor-diffusivity subgrid model. Submitted to *J. Comput. Phys.*
- [36] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C, the Art of Scientific Computing*. Cambridge University Press, 1995.
- [37] D.I. Pullin. A vortex-based model for the subgrid flux of a passive scalar. To appear in *Phys Fluids*.
- [38] L. Råde and B. Westergren. *Mathematics Handbook for Science and Engineering*. Birkhäuser, 1995.
- [39] L.F. Rossi. High order vortex methods with deforming elliptical Gaussian blobs 1: derivation and validation. Submitted to *SIAM Journal on Scientific Computing*.
- [40] L.F. Rossi. High order vortex methods with deforming elliptical Gaussian blobs 2: Merging. Submitted to *SIAM Journal on Scientific Computing*.

- [41] L.F. Rossi. Resurrecting core spreading vortex methods: a new scheme that is both deterministic and convergent. *SIAM J. Sci. Comput.*, 17(2):370–397, March 1996.
- [42] L.F. Rossi. Merging computational elements in vortex simulations. *SIAM J. Sci. Comput.*, 17(4):1014–1027, July 1997.
- [43] K.B. Shah and J. Ferziger. A new non-eddy viscosity subgrid-scale model and its application to channel flow. In *CTR Annual Research Briefs 1995*. Stanford University, California, 1995.
- [44] J. Smagorinsky. General circulation experiments with the primitive equations. *Mon. Weather Rev.*, 91(3):99–164, 1963.
- [45] S Stolz and N.A. Adams. An approximate deconvolution procedure for large-eddy simulation. To appear in *Phys. Fluids*.
- [46] H. Tennekes and J.L. Lumley. *A First Course in Turbulence*. The MIT press, 1972.
- [47] V. Toussaint, P. Carrière, and F. Raynal. A numerical Eulerian approach to mixing by chaotic advection. *Phys. Fluids*, 7(11):2587–2600, November 1995.
- [48] V. Toussaint and Ph. Carrière. Diffusive cut-off scale of fractal surfaces in chaotic mixing. *International Journal of Bifurcation and Chaos*, 9(3):443–454, 1999.
- [49] B. Vreman, B. Geurts, and H. Kuerten. Large-eddy simulation of the temporal mixing layer using the Clark model. *Theoretical and Computational Fluid Dynamics*, 8:309–324, 1996.
- [50] G.S. Winckelmans, A.A. Wray, O.V. Vasilyev, and H. Jeanmart. Explicit-filtering large-eddy simulation using the tensor-diffusivity model supplemented by a dynamic smagorinsky term. Submitted for publication in *Physics of Fluids*.

- [51] 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.
  
- [52] P.K. Yeung and S.B. Pope. An algorithm for tracking fluid particles in numerical simulations of homogeneous turbulence. *J. Comput. Phys.*, 79(2):373–416, December 1988.

## Appendix A Derivations

### A.1 Subgrid Model

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

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

where a sum over repeated indices is implied and  $f(\mathbf{x})$  and  $g(\mathbf{x})$  are arbitrary functions in  $C^\infty$ . By definition,

$$\widehat{fg}(\mathbf{x}) = \frac{1}{(\sigma\sqrt{\pi})^d} \int_{\Omega} f(\mathbf{x}')g(\mathbf{x}') \exp\left(-\frac{|\mathbf{x}' - \mathbf{x}|^2}{\sigma^2}\right) d\mathbf{x}', \quad (\text{A.2})$$

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

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

After partial integrating the right-hand side  $n$  times, we find  $\overline{f}_n(x) = \frac{\sigma^n}{2^n n!} \frac{\partial^n \widehat{f}(x)}{\partial x^n}$ . Plugging above results in (A.2) gives

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

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

$$\widehat{fg}(x) = \sum_{n=0}^{\infty} \frac{\sigma^n}{2^n n!} \frac{\partial^n \widehat{f}(x)}{\partial x^n} 2^n n! \overline{g_n}(x) = \sum_{n=0}^{\infty} \frac{\sigma^n}{2^n n!} \frac{\partial^n \widehat{f}(x)}{\partial x^n} \sigma^n \frac{\partial^n \widehat{g}(x)}{\partial x^n}, \quad (\text{A.5})$$

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

## A.2 Root Mean Square

We will show that the time derivative of  $\tilde{\psi}^2$  can be written as (where it has been assumed that the average value  $\widehat{\psi}_c$  of the scalar field is zero),

$$\frac{d\tilde{\psi}^2}{dt} = -2\kappa \int_{\Omega} (\nabla \widehat{\psi})^2 d\mathbf{x} + \sigma^2 \int_{\Omega} \widehat{S}_{ij} \frac{\partial \widehat{\psi}}{\partial x_i} \frac{\partial \widehat{\psi}}{\partial x_j} d\mathbf{x}, \quad (\text{A.6})$$

where the root mean square value  $\tilde{\psi}(t)$  is defined by (2.9). Denote the boundary of the domain  $\Omega$  by  $\Gamma$  and assume that there is no flow through the boundary, i.e.,  $\mathbf{u} \cdot \widehat{\mathbf{n}} = 0$  on  $\Gamma$  where  $\widehat{\mathbf{n}}$  is the unit normal on the boundary. Also, assume that there is no gradient of  $\widehat{\psi}$  across the boundary,  $\nabla \widehat{\psi} \cdot \widehat{\mathbf{n}} = 0$ .

Multiply the filtered advection-diffusion equation (2.7) by  $\widehat{\psi}$  and integration  $\Omega$  leads to

$$\int_{\Omega} \widehat{\psi} \frac{\partial \widehat{\psi}}{\partial t} d\mathbf{x} + \int_{\Omega} \widehat{\psi} \widehat{\mathbf{u}} \cdot \nabla \widehat{\psi} d\mathbf{x} = \int_{\Omega} \widehat{\psi} \kappa \nabla^2 \widehat{\psi} d\mathbf{x} - \int_{\Omega} \frac{\sigma^2}{2} \widehat{\psi} \widehat{S}_{ij} \frac{\partial^2 \widehat{\psi}}{\partial x_i \partial x_j} d\mathbf{x} \quad (\text{A.7})$$

We will now manipulate each term of this equation. Starting with the first term on the left-hand side, we get

$$\int_{\Omega} \widehat{\psi} \frac{\partial \widehat{\psi}}{\partial t} d\mathbf{x} = \int_{\Omega} \frac{1}{2} \frac{\partial \widehat{\psi}^2}{\partial t} d\mathbf{x} = \frac{1}{2} \frac{d}{dt} \int_{\Omega} \widehat{\psi}^2 d\mathbf{x} = \frac{1}{2} \frac{d\tilde{\psi}^2}{dt}. \quad (\text{A.8})$$

If one uses the incompressibility of the flow and the divergence theorem, it is straightforward to show that the second term on the left-hand side vanishes by making use

of the boundary conditions,

$$\int_{\Omega} \widehat{\psi} \widehat{\mathbf{u}} \cdot \nabla \widehat{\psi} d\mathbf{x} = \int_{\Omega} \nabla \cdot \left( \frac{\widehat{\psi}^2}{2} \widehat{\mathbf{u}} \right) d\mathbf{x} = \int_{\Gamma} \frac{\widehat{\psi}^2}{2} \widehat{\mathbf{u}} \cdot \widehat{\mathbf{n}} d\Gamma = 0. \quad (\text{A.9})$$

The first term on the right-hand side can be manipulated in a similar way,

$$\begin{aligned} \int_{\Omega} \widehat{\psi} \kappa \nabla^2 \widehat{\psi} d\mathbf{x} &= \kappa \int_{\Omega} \left( \frac{1}{2} \nabla \cdot (\nabla \widehat{\psi}^2) - \nabla \widehat{\psi} \cdot \nabla \widehat{\psi} \right) d\mathbf{x} \\ &= \kappa \int_{\Gamma} \widehat{\psi} \nabla \widehat{\psi} \cdot \widehat{\mathbf{n}} d\Gamma - \kappa \int_{\Omega} (\nabla \widehat{\psi})^2 d\mathbf{x} = -\kappa \int_{\Omega} (\nabla \widehat{\psi})^2 d\mathbf{x}. \end{aligned} \quad (\text{A.10})$$

Finally, we look at the second term on the right-hand side. We can write

$$-\int_{\Omega} \frac{\sigma^2}{2} \widehat{\psi} \widehat{S}_{ij} \frac{\partial^2 \widehat{\psi}}{\partial x_i \partial x_j} d\mathbf{x} = -\frac{\sigma^2}{4} \int_{\Omega} \widehat{S}_{ij} \frac{\partial^2 \widehat{\psi}^2}{\partial x_i \partial x_j} d\mathbf{x} + \frac{\sigma^2}{2} \int_{\Omega} \widehat{S}_{ij} \frac{\partial \widehat{\psi}}{\partial x_i} \frac{\partial \widehat{\psi}}{\partial x_j} d\mathbf{x}. \quad (\text{A.11})$$

The first term on the right-hand side of this equation is equal to zero, which can be shown by using the symmetry in the  $i$  and  $j$  indices and replacing  $\widehat{S}_{ij}$  by  $\frac{\partial \widehat{u}_i}{\partial x_j}$ . We get

$$\begin{aligned} -\frac{\sigma^2}{4} \int_{\Omega} \widehat{S}_{ij} \frac{\partial^2 \widehat{\psi}^2}{\partial x_i \partial x_j} d\mathbf{x} &= -\frac{\sigma^2}{4} \int_{\Omega} \frac{\partial}{\partial x_j} \left( \widehat{u}_i \frac{\partial^2 \widehat{\psi}^2}{\partial x_i \partial x_j} \right) d\mathbf{x} + \frac{\sigma^2}{4} \int_{\Omega} \widehat{u}_i \frac{\partial}{\partial x_i} \nabla^2 \widehat{\psi}^2 d\mathbf{x} \\ &= -\frac{\sigma^2}{2} \int_{\Gamma} \left( \widehat{u}_i \frac{\partial \widehat{\psi}}{\partial x_i} \frac{\partial \widehat{\psi}}{\partial x_j} \widehat{n}_j + \widehat{u}_i \widehat{\psi} \frac{\partial^2 \widehat{\psi}}{\partial x_i \partial x_j} \widehat{n}_j \right) d\Gamma + \frac{\sigma^2}{4} \int_{\Omega} \frac{\partial}{\partial x_i} \left( \widehat{u}_i \nabla^2 \widehat{\psi}^2 \right) d\mathbf{x} \\ &= \frac{\sigma^2}{4} \int_{\Gamma} \widehat{u}_i \nabla^2 \widehat{\psi}^2 \widehat{n}_i d\Gamma = 0, \end{aligned} \quad (\text{A.12})$$

where it has been used that on the boundary  $\frac{\partial^2 \widehat{\psi}}{\partial x_i \partial x_j} \widehat{n}_j = \frac{\partial}{\partial x_i} \left( \frac{\partial \widehat{\psi}}{\partial x_j} \widehat{n}_j \right) = 0$  just as  $\widehat{u}_i \widehat{n}_i = 0$ .

Combining (A.7)-(A.12) leads to the desired expression. In case of periodic boundary conditions or an infinite domain, a similar derivation will give the same end result.

### A.3 Anisotropic Gaussian Particles

We want to solve (2.7) using the decomposition given in (3.2). This section will derive the equations for the time derivative of the location ( $\mathbf{x}_k$ ) and shape ( $\mathbf{M}_k$ ) of the particles by substituting (3.2) in (2.7) and expanding each term in a sum of Hermite polynomials. Setting the coefficients of the lowest order terms equal to zero gives the desired result.

For ease of argument and without loss of generality, the derivation will be given in two dimensions for one particle  $k$ . Results for the unfiltered advection-diffusion equation follow simply by setting  $\sigma$  equal to zero. For this particle  $k$ , go over to the principle coordinate system  $(\xi, \eta)$  of matrix  $\mathbf{M}_k$  where  $(1, 0)$  and  $(0, 1)$  are the orthogonal unit eigenvectors corresponding to the eigenvalues  $\lambda_\xi$  and  $\lambda_\eta$ , respectively. In the principle coordinate system, the particle  $k$  is given by

$$\widehat{\psi}_k(\boldsymbol{\xi}, t) = \frac{a_k \sqrt{\lambda_\xi \lambda_\eta}}{\pi \delta_k^2} \exp\left(-\frac{\lambda_\xi (\xi - \xi_k)^2 + \lambda_\eta (\eta - \eta_k)^2}{\delta_k^2}\right). \quad (\text{A.13})$$

Let  $H_m$  denote the  $m$ -th Hermite polynomial, which are treated in appendix B, and set  $H_{m,n} = H_m^\xi H_n^\eta = H_m^\xi \left(\frac{\lambda_\xi (\xi - \xi_k)}{\delta_k}\right) H_n^\eta \left(\frac{\lambda_\eta (\eta - \eta_k)}{\delta_k}\right)$ . Finally set the matrix  $\boldsymbol{\Lambda}_k$  equal to a diagonal matrix with the eigenvalues  $\lambda_\xi$  and  $\lambda_\eta$  on its diagonal.

The filtered advection-diffusion equation does not change due to the transformation except that all the derivatives are now taken with respect to  $\boldsymbol{\xi}$  and not  $\mathbf{x}$  anymore. We will now express each of the terms in this filtered advection-diffusion equation in a sum of Hermite polynomials  $H_{m,n}$ . Starting with the time derivative of  $\widehat{\psi}$ , we find

$$\begin{aligned} \frac{\partial \widehat{\psi}}{\partial t} &= \left[ \frac{\frac{d}{dt}[\det(\boldsymbol{\Lambda}_k)]}{\det(\boldsymbol{\Lambda}_k)} + \frac{2}{\delta_k^2} \frac{d\boldsymbol{\xi}_k^T}{dt} \boldsymbol{\Lambda}_k (\boldsymbol{\xi} - \boldsymbol{\xi}_k) - \frac{(\boldsymbol{\xi} - \boldsymbol{\xi}_k)^T}{\delta_k^2} \frac{d\boldsymbol{\Lambda}_k}{dt} (\boldsymbol{\xi} - \boldsymbol{\xi}_k) \right] \widehat{\psi}_k \\ &= \left[ -\frac{1}{4\lambda_\xi} \frac{d\lambda_\xi}{dt} H_{2,0} - \frac{1}{4\lambda_\eta} \frac{d\lambda_\eta}{dt} H_{0,2} + \frac{\sqrt{\lambda_\xi}}{\delta_k} \frac{d\xi_k}{dt} H_{1,0} + \frac{\sqrt{\lambda_\eta}}{\delta_k} \frac{d\eta_k}{dt} H_{0,1} \right] \widehat{\psi}_k, \end{aligned} \quad (\text{A.14})$$

where use has been made of  $\frac{\frac{d}{dt}[\det(\mathbf{A})]}{\det(\mathbf{A})} = \frac{dA_{ij}}{dt} (A^{-1})_{ji}$  for any matrix  $\mathbf{A}$ . The advection term ( $\widehat{\mathbf{u}} \cdot \nabla \widehat{\psi} = \nabla \cdot \widehat{\mathbf{u}} \widehat{\psi}$ ) can be expressed in a series of Hermite polynomials by

first expanding the velocity field in an infinite series of Hermite polynomials. Next, carry out the gradient operator using (B.6). We find

$$\begin{aligned}\hat{\mathbf{u}} \cdot \nabla \hat{\psi} &= \sum_{m,n=0}^{\infty} \nabla \cdot \left( \bar{\mathbf{u}}_{m,n} H_{m,n} \hat{\psi}_k \right) = \sum_{m,n=0}^{\infty} \bar{\mathbf{u}}_{m,n} \cdot \nabla \left( H_{m,n} \hat{\psi}_k \right) \\ &= \sum_{m,n=0}^{\infty} \left[ -\frac{\sqrt{\lambda_\xi} \bar{u}_{m,n}}{\delta_k} H_{m+1,n} - \frac{\sqrt{\lambda_\eta} \bar{v}_{m,n}}{\delta_k} H_{m,n+1} \right] \hat{\psi}_k,\end{aligned}\tag{A.15}$$

where the functions  $\bar{\mathbf{u}}_{m,n}(\boldsymbol{\xi}_k, t)$  are defined by (see (B.9))

$$\bar{\mathbf{u}}_{m,n}(\boldsymbol{\xi}_k, t) = \frac{\sqrt{\lambda_\xi \lambda_\eta}}{h_m h_n \delta_k^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left(-\frac{(\boldsymbol{\xi} - \boldsymbol{\xi}_k)^T \boldsymbol{\Lambda}_k (\boldsymbol{\xi} - \boldsymbol{\xi}_k)}{\delta_k^2}\right) \hat{\mathbf{u}}(\boldsymbol{\xi}, t) H_{m,n} d\xi d\eta.\tag{A.16}$$

The diffusion term can be written in terms of Hermite polynomials as

$$\begin{aligned}\kappa \nabla^2 \hat{\psi} &= \left[ -\frac{2\kappa}{\delta_k^2} (\boldsymbol{\Lambda}_k)_{ii} + \frac{4\kappa}{\delta_k^4} (\boldsymbol{\xi} - \boldsymbol{\xi}_k)^T \boldsymbol{\Lambda}_k \boldsymbol{\Lambda}_k (\boldsymbol{\xi} - \boldsymbol{\xi}_k) \right] \hat{\psi}_k \\ &= \left[ \frac{\kappa \lambda_\xi}{\delta_k^2} H_{2,0} + \frac{\kappa \lambda_\eta}{\delta_k^2} H_{0,2} \right] \hat{\psi}_k.\end{aligned}\tag{A.17}$$

For the tensor-diffusivity term, we can write

$$\begin{aligned}-\frac{\sigma^2}{2} \hat{S}_{ij} \frac{\partial^2 \hat{\psi}}{\partial \xi_i \partial \xi_j} &= \left[ \frac{\sigma^2}{\delta_k^2} \hat{S}_{ij} \Lambda_{ij} - \frac{2\sigma^2}{\delta_k^4} (\boldsymbol{\xi} - \boldsymbol{\xi}_k)^T \boldsymbol{\Lambda}_k \hat{\mathbf{S}} \boldsymbol{\Lambda}_k (\boldsymbol{\xi} - \boldsymbol{\xi}_k) \right] \hat{\psi}_k \\ &= \left[ -\frac{\sigma^2 \lambda_\xi}{2\delta_k^2} \frac{\partial \hat{u}}{\partial \xi} H_{2,0} - \frac{\sigma^2 \lambda_\eta}{2\delta_k^2} \frac{\partial \hat{v}}{\partial \eta} H_{0,2} - \frac{\sigma^2 \sqrt{\lambda_\xi \lambda_\eta}}{2\delta_k^2} \left( \frac{\partial \hat{u}}{\partial \eta} + \frac{\partial \hat{v}}{\partial \xi} \right) H_{1,1} \right] \hat{\psi}_k.\end{aligned}\tag{A.18}$$

Next, we express the components of the strain rate tensor in an infinite series of

Hermite polynomials and after using (B.7) and (B.8), we end up with

$$\begin{aligned}
& -\frac{\sigma^2}{2} \widehat{S}_{ij} \frac{\partial^2 \widehat{\psi}}{\partial \xi_i \partial \xi_j} = \\
& \sum_{m,n=0}^{\infty} \left\{ -\frac{\sigma^2 \sqrt{\lambda_\xi \lambda_\eta}}{2\delta_k^2} \left( \frac{\overline{\partial \widehat{u}}}{\partial \eta} + \frac{\overline{\partial \widehat{v}}}{\partial \xi} \right)_{m,n} (H_{m+1}^\xi + 2mH_{m-1}^\xi)(H_{n+1}^\eta + 2nH_{n-1}^\eta) \right. \\
& \quad - \frac{\sigma^2 \lambda_\xi}{2\delta_k^2} \frac{\overline{\partial \widehat{u}}}{\partial \xi}_{m,n} (H_{m+2,n} + 4mH_{m,n} + 4m(m-1)H_{m-2,n}) \\
& \quad \left. - \frac{\sigma^2 \lambda_\eta}{2\delta_k^2} \frac{\overline{\partial \widehat{v}}}{\partial \eta}_{m,n} (H_{m,n+2} + 4nH_{m,n} + 4n(n-1)H_{m,n-2}) \right\} \widehat{\psi}_k.
\end{aligned} \tag{A.19}$$

From now on, use the notation  $\bar{f}_{0,0} = \bar{f}$ . We can relate  $\bar{f}_{m,n}$  to  $\bar{f}$  by (B.10). If we consider the coefficient of  $H_{0,0}$ , it is easy to show that this one is always zero, using the incompressibility of the flow. By setting the  $H_{1,0}$  and  $H_{0,1}$  coefficients equal to zero, we end up with the following two equations for the time derivative of  $\xi_k$  and  $\eta_k$ , respectively,

$$\frac{d\xi_k}{dt} = \bar{u} - \frac{\sigma^2}{2} \left( \frac{\overline{\partial^2 \widehat{u}}}{\partial \xi^2} + \frac{\overline{\partial^2 \widehat{u}}}{\partial \eta^2} \right), \tag{A.20a}$$

$$\frac{d\eta_k}{dt} = \bar{v} - \frac{\sigma^2}{2} \left( \frac{\overline{\partial^2 \widehat{v}}}{\partial \xi^2} + \frac{\overline{\partial^2 \widehat{v}}}{\partial \eta^2} \right). \tag{A.20b}$$

If we transform now back to the original coordinate system and combine both equations, we end up with (3.4).

Next consider the coefficients of  $H_{2,0}$  and  $H_{0,2}$ . We can find, respectively,

$$\frac{d\lambda_\xi}{dt} = -2\lambda_\xi \frac{\overline{\partial \widehat{u}}}{\partial \xi} - 4\kappa \frac{\lambda_\xi \lambda_\xi}{\delta_k^2} + \frac{2\sigma^2}{\delta_k^2} \lambda_\xi \lambda_\xi \frac{\overline{\partial \widehat{u}}}{\partial \xi} + \sigma^2 \lambda_\xi \left( \frac{\overline{\partial^3 \widehat{u}}}{\partial \xi^3} + \frac{\overline{\partial^3 \widehat{u}}}{\partial \xi \partial \eta^2} \right), \tag{A.21a}$$

$$\frac{d\lambda_\eta}{dt} = -2\lambda_\eta \frac{\overline{\partial \widehat{v}}}{\partial \eta} - 4\kappa \frac{\lambda_\eta \lambda_\eta}{\delta_k^2} + \frac{2\sigma^2}{\delta_k^2} \lambda_\eta \lambda_\eta \frac{\overline{\partial \widehat{v}}}{\partial \eta} + \sigma^2 \lambda_\eta \left( \frac{\overline{\partial^3 \widehat{v}}}{\partial \xi^2 \partial \eta} + \frac{\overline{\partial^3 \widehat{v}}}{\partial \eta^3} \right). \tag{A.21b}$$

For  $H_{1,1}$ , we get

$$\begin{aligned}
0 = & -\lambda_\xi \frac{\overline{\partial \hat{u}}}{\partial \eta} - \lambda_\eta \frac{\overline{\partial \hat{v}}}{\partial \xi} + \frac{\sigma^2 \lambda_\xi \lambda_\eta}{\delta_k^2} \left( \frac{\overline{\partial \hat{u}}}{\partial \eta} + \frac{\overline{\partial \hat{v}}}{\partial \xi} \right) \\
& + \frac{\sigma^2 \lambda_\xi}{2} \left( \frac{\overline{\partial^3 \hat{u}}}{\partial \xi^2 \partial \eta} + \frac{\overline{\partial^3 \hat{u}}}{\partial \eta^3} \right) + \frac{\sigma^2 \lambda_\eta}{2} \left( \frac{\overline{\partial^3 \hat{v}}}{\partial \xi \partial \eta^2} + \frac{\overline{\partial^3 \hat{v}}}{\partial \xi^3} \right).
\end{aligned} \tag{A.22}$$

It is straightforward to check that if we combine (A.21)-(A.22) in matrix form, we end up with

$$\begin{aligned}
\frac{d\mathbf{\Lambda}_k}{dt} = & -\overline{\nabla \hat{\mathbf{u}}} \mathbf{\Lambda}_k - \mathbf{\Lambda}_k \overline{\nabla \hat{\mathbf{u}}}^T - \frac{4\kappa}{\delta_k^2} \mathbf{\Lambda}_k \mathbf{\Lambda}_k + 2 \frac{\sigma^2}{\delta_k^2} \mathbf{\Lambda}_k \widehat{\mathbf{S}} \mathbf{\Lambda}_k \\
& + \frac{\sigma^2}{2} \overline{\nabla \nabla^2 \hat{\mathbf{u}}} \mathbf{\Lambda}_k + \frac{\sigma^2}{2} \mathbf{\Lambda}_k \overline{\nabla \nabla^2 \hat{\mathbf{u}}}^T.
\end{aligned} \tag{A.23}$$

If we transform back to the original coordinate system, we get (3.6).

## A.4 Remeshing

In order to find a good remeshing procedure, we want to minimize (3.10). Without loss of generality, assume only one spatial direction which reduces (3.10) to

$$\epsilon(x) = \sum_{k=1}^N \frac{a_k \sqrt{\det(\lambda_k)}}{\sqrt{\pi} \delta_k} \exp\left(-\frac{\lambda_k (x - x_k)^2}{\delta_k^2}\right) - \sum_{l=1}^M \frac{b_l}{\sqrt{\pi} \tau} \exp\left(-\frac{(x - \xi_l)^2}{\tau^2}\right), \tag{A.24}$$

where the shape matrix  $\mathbf{M}_k$  has been replaced by  $\lambda_k$ , the eigenvalue (and only element) of  $\mathbf{M}_k$ . We square (A.24), integrate over  $x$ , and set the derivative with respect to the unknown  $b_l$ 's equal to zero (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

$$\sum_{l=1}^M \frac{b_l}{\sqrt{2}} \exp\left(-\frac{(\xi_l - \xi_i)^2}{2\tau^2}\right) = \sum_{k=1}^N \int_{-\infty}^{\infty} \frac{a_k \sqrt{\lambda_k}}{\sqrt{\pi} \delta_k} \exp\left(-\frac{\lambda_k (x' - x_k)^2}{\delta_k^2}\right) \exp\left(-\frac{(x' - \xi_i)^2}{\tau^2}\right) dx'. \quad (\text{A.25})$$

Express the first exponential on the right-hand side as

$$\exp\left(-\frac{\lambda_k (x' - x_k)^2}{\delta_k^2}\right) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{\pi} \sqrt{\delta_k^2 - \lambda_k \tau^2}} \exp\left(-\frac{(x' - x_k - \hat{x})^2}{\tau^2}\right) \exp\left(-\frac{\lambda_k \hat{x}^2}{\delta_k^2 - \lambda_k \tau^2}\right) d\hat{x}, \quad (\text{A.26})$$

which is only possible for  $\tau^2 < \frac{\delta_k^2}{\lambda_k}$ . This puts an upper bound on the core size of the new particles. Upon substitution in (A.25), carrying out the integration over  $x'$  and replacing  $\hat{x}$  by  $x' - x_k$ , we have

$$\begin{aligned} \sum_{l=1}^M \frac{b_l}{\sqrt{2}} \exp\left(-\frac{(\xi_l - \xi_i)^2}{2\tau^2}\right) &= \sum_{k=1}^N \int_{-\infty}^{\infty} \frac{a_k \sqrt{\lambda_k}}{\sqrt{2\pi} \sqrt{\delta_k^2 - \lambda_k \tau^2}} \exp\left(-\frac{\lambda_k (x' - x_k)^2}{\delta_k^2 - \lambda_k \tau^2}\right) \exp\left(-\frac{(x' - \xi_i)^2}{2\tau^2}\right) dx'. \end{aligned} \quad (\text{A.27})$$

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

$$\exp\left(-\frac{(x' - \xi_i)^2}{2\tau^2}\right) \approx \sum_{l=1}^M m(x' - \xi_l) \exp\left(-\frac{(\xi_l - \xi_i)^2}{2\tau^2}\right), \quad (\text{A.28})$$

where the function  $m(x)$  has to be chosen. See for example Cottet and Koumoutsakos [12] for a discussion of possible choices. After substitution of this equation in (A.27), 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,

$$b_l = \sum_{k=1}^N \int_{-\infty}^{\infty} \frac{a_k \sqrt{\lambda_k}}{\sqrt{\pi} \sqrt{\delta_k^2 - \lambda_k \tau^2}} \exp\left(-\frac{\lambda_k (x' - x_k)^2}{\delta_k^2 - \lambda_k \tau^2}\right) m(x' - \xi_l) dx'. \quad (\text{A.29})$$

We have found that the simplest possible choice,  $m(x) = \delta(x)h$ , where  $\delta(x)$  is the Dirac delta function, gives good results. Equation (A.29) then reduces to (3.11). Different choices for  $m(x)$  have been considered and tested, such as a polynomial times a Gaussian, but the Dirac delta function had the best overall performance.

## Appendix B Hermite Polynomials

The orthogonal Hermite polynomials<sup>1</sup>  $H_n(x)$  of degree  $n$  are defined by

$$\begin{aligned} H_n(x) &= (-1)^n e^{x^2} \frac{d^n e^{-x^2}}{dx^n} \\ &= n! \sum_{m=0}^{\lfloor \frac{n}{2} \rfloor} \frac{(-1)^m}{m!(n-2m)!} (2x)^{n-2m}, \end{aligned} \quad (\text{B.1})$$

where  $\lfloor k \rfloor$  is the largest integer less or equal to  $k$ . The first six Hermite polynomials are

$$\begin{aligned} H_0(x) &= 1, & H_2(x) &= 4x^2 - 2, & H_4(x) &= 16x^4 - 48x^2 + 12, \\ H_1(x) &= 2x, & H_3(x) &= 8x^3 - 12x, & H_5(x) &= 32x^5 - 160x^3 + 120x. \end{aligned} \quad (\text{B.2})$$

Hermite polynomials are orthogonal with respect to the weight function  $e^{-x^2}$ , i.e.,

$$\int_{-\infty}^{\infty} e^{-x^2} H_n(x) H_m(x) dx = \begin{cases} 0 & \text{if } m \neq n, \\ 2^n n! \sqrt{\pi} = h_n & \text{if } m = n. \end{cases} \quad (\text{B.3})$$

The following relations between Hermite polynomials of different order are straightforward to derive:

$$H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x), \quad (\text{B.4})$$

$$\frac{dH_n(x)}{dx} = 2nH_{n-1}(x), \quad (\text{B.5})$$

$$H_n(x)e^{-x^2} = -\frac{d}{dx} \left( H_{n-1}(x)e^{-x^2} \right), \quad (\text{B.6})$$

$$H_1(x)H_n(x) = H_{n+1}(x) + 2nH_{n-1}(x), \quad (\text{B.7})$$

$$H_2(x)H_n(x) = H_{n+2}(x) + 4nH_n(x) + 4n(n-1)H_{n-2}(x), \quad (\text{B.8})$$

---

<sup>1</sup>For more detailed information on Hermite polynomials see, e.g., [1].

where  $H_n(x) = 0$  for  $n < 0$ . Assuming that  $f(x)$  is an arbitrary smooth function, it can be expanded in an infinite sum of Hermite polynomials as  $f(x) = \sum_{n=0}^{\infty} \bar{f}_n H_n(x)$ , where the coefficients  $\bar{f}_n$  follow from

$$\bar{f}_n = \frac{1}{h_n} \int_{-\infty}^{\infty} e^{-x^2} f(x) H_n(x) dx. \quad (\text{B.9})$$

Using partial integration, one can relate the coefficients  $\bar{f}_n$  to the first coefficient of derivatives of  $f(x)$  as follows:

$$\bar{f}_n = \frac{1}{2^n n!} \overline{f_0^{(n)}}, \quad (\text{B.10})$$

where  $f^{(n)} = \frac{d^n f}{dx^n}$ .