

A Fourier–Wavelet Monte Carlo Method for Fractal Random Fields

Frank W. Elliott Jr., David J. Horntrop, and Andrew J. Majda

Courant Institute of Mathematical Sciences, 251 Mercer Street, New York, New York 10012

Received August 2, 1996; revised December 23, 1996

A new hierarchical method for the Monte Carlo simulation of random fields called the Fourier–wavelet method is developed and applied to isotropic Gaussian random fields with power law spectral density functions. This technique is based upon the orthogonal decomposition of the Fourier stochastic integral representation of the field using wavelets. The Meyer wavelet is used here because its rapid decay properties allow for a very compact representation of the field. The Fourier–wavelet method is shown to be straightforward to implement, given the nature of the necessary precomputations and the run-time calculations, and yields comparable results with scaling behavior over as many decades as the physical space multiwavelet methods developed recently by two of the authors. However, the Fourier–wavelet method developed here is more flexible and, in particular, applies to anisotropic spectra generated through solutions of differential equations. Simulation results using this new technique and the well-known nonhierarchical simulation technique, the randomization method, are given and compared for both a simple shear layer model problem as well as a two-dimensional isotropic Gaussian random field. The Fourier–wavelet method results are more accurate for statistical quantities depending on moments higher than order 2, in addition to showing a quite smooth decay to zero on the scales smaller than the scaling regime when compared with the randomization method results. The only situation in which the nonhierarchical randomization method is more computationally efficient occurs when no more than four decades of scaling behavior are needed and the statistical quantities of interest depend only on second moments. © 1997 Academic Press

1. INTRODUCTION

The generation of random velocity fields which are fractal and self-similar over many scales by accurate and efficient Monte Carlo methods is an important and challenging problem in a wide variety of applications including turbulent diffusion in the environment [1], solute transport in groundwater [2], and random topography in statistical physics [3, 4]. In the simplest context of a zero mean stationary scalar Gaussian random field $v(x)$ in one dimension, these fractal fields are completely characterized by the mean

$$\langle v(x) \rangle = 0$$

and the structure function

$$\langle [v(x) - v(y)]^2 \rangle = C_H |x - y|^{2H}, \quad (1.1)$$

where $0 < H < 1$ is the Hurst exponent and $\langle \cdot \rangle$ denotes the expected value.

Here we develop a new Monte Carlo method based upon a wavelet expansion of the Fourier space representation of the fractal random fields in (1.1). This method is capable of generating a velocity field with the Kolmogoroff spectrum ($H = \frac{1}{3}$ in (1.1)) over many (10 to 15) decades of scaling behavior comparable to the physical space multiwavelet algorithm developed by two of the authors in recent work [5–7]. However, since the Fourier–wavelet method developed below is a spectral method, it is much more flexible for generating random fields with anisotropic spectra and, in particular, spectra generated through the solutions of partial differential equations.

A wide variety of computational approaches have been used to generate fractal random fields. Some of these algorithms involve hierarchical methods [4–7] while others utilize nonhierarchical methods [8–12] with varying degrees of success. Hierarchical methods involve the decomposition of a random field into distinct characteristic scales. Because of the relatively compact nature of this type of representation, hierarchical methods are particularly attractive for simulating random fields with large regimes of scaling behavior. The nonhierarchical methods, such as the randomization method [8] and its variants [9], as well as the moving average method [10, 11], typically require strong bandwidth limitations in the spectrum and only produce a small number of decades of scaling behavior. One commonly used hierarchical simulation technique, successive random addition [4, 13], was recently shown in [14] to be inconsistent with a stationary random field. Recently, two of the authors have introduced, validated, and used a new hierarchical method based upon a multiwavelet expansion (MWE) of the moving average stochastic integral representation to generate fractal random fields as well as passive scalar advection with such fields [5–7, 10]. Even though the MWE technique is limited to random fields with power law spectra, it provides for the accurate generation of fields with an excess of 12 decades of scaling behavior. Below we will introduce the Fourier–wavelet method which is

based upon a wavelet expansion of the Fourier stochastic integral representation of a homogeneous Gaussian random field. The Fourier-wavelet method has a simpler form than the MWE method and can be applied to a wider variety of physical problems since the Fourier-wavelet method can be applied to general spectra. Further discussion of the differences between the MWE approach and the Fourier-wavelet method is given at the end of Section 2.

We now give an outline of the remaining portion of this paper. Section 2 begins with a brief description of the stochastic integral representation of a homogeneous Gaussian random field and a brief review of wavelet theory prior to actually giving the theoretical Fourier-wavelet representation of the field as a double summation over the product of Gaussian weights and deterministic physical space “kernels.” The Meyer wavelet and its Fourier transform are then rigorously shown to have the appropriate properties in order to ensure that the terms in the Fourier-wavelet representation of the field decay to a high polynomial order. Given this rather rapid convergence, the effect of the truncation of the doubly infinite sum in the Fourier-wavelet representation to give the simulation method is shown to be rather small. Section 2 closes with a discussion of some theoretical implementation issues including the efficient generation of the needed Gaussian weights and the interpolations required to evaluate the deterministic “kernels” at values other than those precomputed using a fast Fourier transform.

The purpose of Section 3 is to give some simulation results using the Fourier-wavelet method and make comparisons with both analytic formulas and simulation results using a popular nonhierarchical method, the randomization method [8]. Section 3 begins with a discussion of the simple shear layer model problem and the exactly solvable statistical quantities that will be used as benchmarks for the Monte Carlo simulations. Simulation results from the Fourier-wavelet method and randomization method are then given and compared. It is seen that the Fourier-wavelet method gives more accurate results for statistics depending on higher than second-order moments, as well as showing a smoother decay to zero on the scales below the scaling regime than the randomization method. However, it is shown that the randomization method is more computationally efficient if only four or five decades of scaling behavior are desired and if the statistical quantities of interest depend only on second moments. Section 3 concludes with a discussion of the simulation of a two-dimensional isotropic Gaussian random field. These results point out the necessity of the use of an “on demand” random number generator in order to make practical use of the Fourier-wavelet method, as well as the fact that trends observed in the simple shear layer simulations also hold in the simulations with two-dimensional random fields.

2. DEVELOPMENT OF THE FOURIER-WAVELET METHOD

In this section we develop the Fourier-wavelet method. We begin by recalling the general theory of random fields in one dimension and using this theory to develop our method in full generality. Then we discuss the choice of basis for an accurate, sparse approximation of the field. Finally, we detail a simple technique for implementing this method and analyze the errors involved. In this section we restrict ourselves to one-dimensional fields, taking up the subject of two-dimensional fields in Section 3. For a rigorous generalization of such one-dimensional results to higher dimensional fields, we refer the reader to earlier work of two of the authors in [6, 15].

The description of the general structure begins with a brief review of the theory of Gaussian random functions and their representation and continues by introducing the necessary wavelet theory used to change the spectral representation to the Fourier-wavelet representation. We then turn to truncation of the Fourier-wavelet representation to produce an efficient, hierarchical method for generating one-dimensional random fields. Finally, we discuss our selection of the Meyer wavelet as an efficient choice for the Fourier-wavelet method.

In our discussion of the implementation of the method, we consider several key issues relating to the efficiency and accuracy of the method. Simulation of the random field occurs in two phases, a precomputation phase and a run-time phase. In the precomputation phase, a collection of kernels is computed using the discrete Fourier transform. For this procedure, we analyze the effects of aliasing and interpolation on the accuracy of these kernels. In the run-time phase, the kernels are evaluated at several points and summed with random weights. For the computation of the random weights, we describe a procedure which is adapted to sparse summations of the kind that occur in the Fourier-wavelet method.

2.1. General Structure of the Fourier-Wavelet Method

In preparation to discuss our method, we now review the basic theory of homogeneous Gaussian random functions on the real line. Our method extends to two and three dimensions by using the plane-wave construction which was devised in [15] and implemented in conjunction with the multiwavelet method in [6].

2.1.1. Introduction to Gaussian Random Fields

Recall that a homogeneous, Gaussian, random function $v(x)$ with mean 0 is completely characterized by its moment equations

$$\begin{aligned} \langle v(x) \rangle &= 0 \\ \langle v(x + x')v(x') \rangle &= R(x) \\ &= \int_{-\infty}^{+\infty} e^{-i2\pi kx} E(k) dk. \end{aligned} \tag{2.1}$$

Here $R(x)$ and $E(k)$ are the covariance and spectrum of the random field $v(x)$.

We now describe the spectral representation and the moving average representation of a one-dimensional, homogeneous Gaussian random field $v(x)$. Both of these representations are integrals using the standard Wiener process $W(\cdot)$ as an integrator. The spectral representation and the moving average representation are given by the formulas

$$v(x) = \int_{-\infty}^{+\infty} e^{-i2\pi kx} E^{1/2}(k) dW(k) \tag{2.2a}$$

$$v(x) = \int_{-\infty}^{+\infty} (\mathcal{F}^{-1}E^{1/2})(y) dW(y), \tag{2.2b}$$

where $dW(\cdot)$ is the differential of the standard Wiener process, as discussed below, and the symbol \mathcal{F}^{-1} in formula (2.2b) is an inverse Fourier transform. We define the Fourier transform and its inverse by

$$(\mathcal{F}f)(k) = \int_{-\infty}^{+\infty} e^{-i2\pi xk} f(x) dx \tag{2.3}$$

$$(\mathcal{F}^{-1}f)(x) = \int_{-\infty}^{+\infty} e^{i2\pi xk} f(k) dk.$$

Now, we define and represent $dW(k)$, which is sometimes called *white noise*. Integration of a function $f(k)$ with respect to $dW(k)$ linearly maps the function f to a Gaussian random variable described by the moment equations

$$\begin{aligned} \left\langle \int f(k) dW(k) \right\rangle &= 0 \\ \left\langle \int f(k) dW(k) \int \overline{g(k) dW(k)} \right\rangle &= \int f(k) \overline{g(k)} dk \\ &= (f, g), \end{aligned} \tag{2.4}$$

where (\cdot, \cdot) is the inner product for square integrable functions. The equations in (2.4) provide a means of expanding white noise in terms of an orthonormal basis; we use such an expansion in deriving our Fourier-wavelet method. An *orthonormal basis* is a sequence of functions $\{\phi_n | n = 1, 2, \dots\}$ which for any square integrable function f satisfies

$$(\phi_m, \phi_n) = \delta_{m,n} \tag{2.5a}$$

$$f = \sum_{m=1}^{\infty} (f, \phi_m) \phi_m, \tag{2.5b}$$

where $\delta_{m,n}$ is the Kronecker delta. Given any orthonormal basis $\{\phi_m | m = 1, 2, \dots\}$ and any independent, identically distributed Gaussian sequence $\{\gamma_m | m = 1, 2, \dots\}$ with mean 0 and variance 1, it is easily verified that

$$dW(k) = \sum_{m=1}^{\infty} \gamma_m \phi_m(k) dk \tag{2.6}$$

satisfies (2.4) so that it is a representation of white noise (see [5], for example).

2.1.2. Elementary Wavelet Theory

We now briefly review some wavelet theory and utilize it to construct series for each of the representations of $v(x)$ in (2.2); we use these series as the basis of our method which we develop in the next subsection. We begin by considering the definition of a mother wavelet ϕ and show how its Fourier transform $(\mathcal{F}\phi)$ also generates a basis. We then use both ϕ and $(\mathcal{F}\phi)$ to generate white noise, dW . Finally, this formula for dW is used to construct series for the representations of $v(x)$ in (2.2).

Discrete wavelet analysis is based on using the dilates and translates of a single function ϕ as a basis for a space of functions. Let ϕ be a square integrable function on the real line, and let

$$\phi_{m,n}(x) = 2^{-m/2} \phi(2^{-m}x - n), \tag{2.7}$$

for $m, n = 0, \pm 1, \pm 2, \dots$. If the set of all the dilates and translates of ϕ

$$\{\phi_{m,n} | m, n = 0, \pm 1, \pm 2, \dots\}$$

is an orthonormal basis (as defined in (2.5)) for the square integrable functions, then ϕ is a *mother wavelet*. Each dilate and translate has the Fourier transform

$$\begin{aligned} (\mathcal{F}\phi_{m,n})(k) &= \int_{-\infty}^{\infty} e^{-i2\pi kx} \phi_{m,n}(x) dx \\ &= e^{+i2\pi 2^m n k} 2^{m/2} (\mathcal{F}\phi)(2^m k). \end{aligned} \tag{2.8}$$

According to Plancherel's theorem, the Fourier transform preserves inner products of square integrable functions. Therefore the Fourier transform maps the orthonormal basis $\{\phi_{m,n} | m, n = 0, \pm 1, \pm 2, \dots\}$ into the orthonormal basis $\{(\mathcal{F}\phi_{m,n}) | m, n = 0, \pm 1, \pm 2, \dots\}$.

2.1.3. The Fourier Wavelet Representation

We will now show that we can use ϕ and $(\mathcal{F}\phi)$ to write the representation of $dW(k)$ in (2.2), in terms of infinite series with random weights. We can use ϕ or $(\mathcal{F}\phi)$ to represent white noise, dW , according to the formulas

$$dW(k) = \left\{ \sum_{m,n=-\infty}^{\infty} (\mathcal{F}\phi_{m,n})(k)\gamma(m,n) \right\} dk, \quad (2.9a)$$

$$dW(y) = \left\{ \sum_{m,n=-\infty}^{\infty} \phi_{m,n}(y)\gamma(m,n) \right\} dy, \quad (2.9b)$$

where $\{\gamma(m,n)\}$ is a doubly indexed sequence of independent, identically distributed, standard Gaussian random variables. Two of the authors used an expansion of dW like formula (2.9b) to synthesize the random field $v(x)$ in x -space; the result was the multiwavelet expansion (see [5]). We now use the other formula (2.9a) to synthesize the random field $v(x)$ in k -space; as the discussion at the end of this section indicates, the result is a hierarchical method which is easier to implement, faster, and more flexible than the multiwavelet expansion for problems where a general spectrum is specified.

Proceeding with the development of this method, we substitute (2.9a) into the representation in (2.2a) to obtain

$$\begin{aligned} v(x) &= \int_{-\infty}^{+\infty} e^{+i2\pi kx} E^{1/2}(k) \\ &\times \left\{ \sum_{m,n=-\infty}^{\infty} 2^{m/2} (\mathcal{F}\phi)(2^m k) e^{-i2\pi 2^m n k} \gamma(m,n) \right\} dk \\ &= \sum_{m,n=-\infty}^{\infty} \gamma(m,n) \int_{-\infty}^{+\infty} e^{+i2\pi k(2^{-m}x-n)} E_{m,0}^{1/2}(k) (\mathcal{F}\phi)(k) dk, \end{aligned} \quad (2.10)$$

where subscripts of $E_{m,0}^{1/2}$ reflect the dilation and translation of $E^{1/2}$, respectively; i.e. $E_{m,n}^{1/2} = 2^{-m/2} E(2^{-m}x - n)$. Simplifying Eq. (2.10), we obtain the *Fourier–wavelet representation* for the field $v(x)$,

$$v(x) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_m(2^{-m}x - n)\gamma(m,n) \quad (2.11a)$$

$$f_m(x) = (\mathcal{F}^{-1}((\mathcal{F}\phi)E_{m,0}^{1/2}))(x). \quad (2.11b)$$

We refer to the function f_m as the *kernel* for scale m . For the purpose of implementing (2.11), we truncate the summation in m , and, for each scale m , we truncate the summation in n . To ensure that this truncation has little effect on the accuracy of the method, we must choose a mother wavelet ϕ so that $|f_m(y)|$ decays rapidly in $|y|$. In the next section, we describe the rationale for our choice, the Meyer wavelet.

2.2. Choice of the Basis for the Fourier–Wavelet Method

In this section we begin the job of turning the representation in (2.11) into an efficient simulation method. Two considerations guide us in this task. First, the summation

in (2.11) must be truncated in both m and n with little loss of accuracy. Second, the functions $f_m(x)$ should be easily and accurately computable. We address the first of these concerns immediately and defer the second to the section on theoretical implementation issues. Nevertheless, both of these concerns can be satisfied if we can assure that $(\mathcal{F}\phi)$ is compactly supported and has several bounded derivatives. Furthermore, for the spectra $E(k)$ that we consider, $(\mathcal{F}\phi)$ will have the desired properties if we choose ϕ to be the Meyer wavelet (see [16]).

We formulate a truncation formula based on the decay estimates of the function $f_m(x)$. Assuming that $(\mathcal{F}\phi)$ has p classical derivatives satisfying

$$|2\pi|^{-p} \int_{-\infty}^{\infty} \left| \frac{\partial^p (\mathcal{F}\phi)}{\partial k^p} \right| = A_{mp} < \infty, \quad (2.12)$$

it can be shown that

$$|f_m(x)| \leq A_{mp} |x|^{-p}. \quad (2.13)$$

Using the decay specified in (2.13), we can change the Fourier–wavelet representation in (2.11) by centering the summation in n about $2^{-m}x$ and by truncating the summations in both n and m . Thus, we obtain an approximation of $v(x)$

$$v_a(x) = \sum_{m=m_{\min}}^{m_{\max}} \tilde{v}_m(x) \quad (2.14a)$$

$$\tilde{v}_m(x) = \sum_{n=n_{\min}}^{n_{\max}} \gamma(m,n + \lfloor 2^{-m}x \rfloor) f_m(2^{-m}x - \lfloor 2^{-m}x \rfloor - n). \quad (2.14b)$$

Here, the notation $\lfloor x \rfloor$ refers to the greatest integer less than the real number x . The algebraic decay described in (2.13) assures that the choice a finite bandwidth $n_{\max} - n_{\min} + 1$ in (2.14b) causes a rapidly decaying error in this summation. We estimate this error according to the formula

$$\langle |\tilde{v}_m^\infty(x) - \tilde{v}_m(x)|^2 \rangle \leq 2A_{mp}^2 \frac{(n_{\max} - n_{\min})^{-2p+1}}{2p - 1}, \quad (2.15)$$

where

$$\tilde{v}_m^\infty(x) = \lim_{\substack{n_{\min} \rightarrow -\infty \\ n_{\max} \rightarrow \infty}} \tilde{v}_m(x).$$

The truncation process in (2.14) leads us to the formula for the *Fourier–wavelet method*

$$v_a(x) = \sum_{m=m_{\min}}^{m_{\max}} \sum_{n=n_{\min}}^{n_{\max}} \gamma(m, n + \lfloor 2^{-m}x \rfloor) f_m(2^{-m}x - \lfloor 2^{-m}x \rfloor - n). \tag{2.16}$$

Of course,

$$v(x) = \lim_{\substack{m_{\min} \rightarrow -\infty \\ m_{\max} \rightarrow \infty}} \lim_{\substack{n_{\min} \rightarrow -\infty \\ n_{\max} \rightarrow \infty}} v_a(x),$$

which maintains consistency with the Fourier–wavelet representation (2.11).

Our choice of a wavelet ϕ for the Fourier–wavelet method is dictated by the class of spectra $E(k)$ we wish to consider. Specifically, we wish to allow $E(k)$ to be any function which is smooth except for possible strong infra-red and ultraviolet singularities, nonintegrable algebraic singularities at $k = 0$ and at $k = \infty$, respectively. Since

$$(\mathcal{F}f_m)(k) = 2^{-m/2} E^{1/2}(2^{-m}k) (\mathcal{F}\phi)(k), \tag{2.17}$$

$(\mathcal{F}f_m)$ has p bounded derivatives, provided that $(\mathcal{F}\phi)$ is compactly supported away from the origin and has p bounded derivatives. This will give f_m a sufficiently rapid decay so that we can perform the truncation of the doubly indexed sum shown in (2.14) without a significant loss in the accuracy of the field $v(\cdot)$. In the paragraphs that follow we justify our choice for ϕ , the Meyer wavelet, and explore the relevant properties of its Fourier transform $(\mathcal{F}\phi)$.

We now describe the basic structure of the Fourier transform, $(\mathcal{F}\phi)$, of the Meyer wavelet (see [16, 21]). The spectrum $(\mathcal{F}\phi)$ of the Meyer wavelet ϕ is given by the formulas

$$(\mathcal{F}\phi)(k) = (2\pi)^{-1/2} \operatorname{sgn}(k) e^{ink} b(|k|), \tag{2.18}$$

where

$$\operatorname{sgn}(k) = \begin{cases} -1, & k < 0, \\ 0, & k = 0, \\ +1, & k > 0, \end{cases}$$

and $b(k)$ is a p -times differentiable function with values in $[0, 1]$ and support in $[\frac{1}{3}, \frac{2}{3}]$. Because the function $b(k)$ limits the portion of the spectrum $E_{m,0}^{1/2}$ appearing in $(\mathcal{F}f_m)$, we refer to $b(k)$ as a *window function*. This window function ensures that $(\mathcal{F}f_m)$ will have no singularities and will have bounded, compactly supported derivatives to order p . Thus f_m meets the decay condition (2.13). We compute $b(k)$ using the p th order *perfect B-spline* to ensure that $b(k)$ has p bounded derivatives (see [17]). For a discussion of appropriate choices for the window function $b(k)$, we refer the reader to Appendix A. The phase factor $\operatorname{sgn}(k)e^{ink}$

ensures that $(\mathcal{F}\phi)$ defines the Fourier transform of a wavelet.

We now examine the functions $f_m(x)$ computed using the Meyer wavelet ϕ and one important spectrum of practical interest involving a fractal random field, the Kolmogoroff spectrum

$$E(k) = |k|^{-5/3} \tag{2.19}$$

which corresponds to the exponent $H = \frac{1}{3}$ in (1.1). We need only concern ourselves with the behavior of $f_0(x)$ because

$$\begin{aligned} (\mathcal{F}f_m) &= (\mathcal{F}\phi)(k) E_{m,0}^{1/2}(k) \\ &= (\mathcal{F}\phi)(k) 2^{-m/2} |2^{-m}k|^{-5/6} \\ &= 2^{m/3} (\mathcal{F}f_0)(k) \end{aligned} \tag{2.20}$$

so that $f_m(x) = 2^{m/3} f_0(x)$. The top plot in Fig. 1 shows the graph of $f_0(x)$ using the Meyer wavelet with $b(k)$ computed with the second-order perfect B-spline; the bottom plot in Fig. 1 shows the graph of f_0 using the Meyer wavelet with $b(k)$ computed with the fourth-order perfect B-spline (see Appendix A). Even though (2.13) predicts that the graph in the lower part of Fig. 1 would decay at a greater asymptotic rate, there is little observable difference in the decay rate of these two functions. Nevertheless, it is apparent that as p increases, so does the coefficient A_{0p} . Therefore, there is little practical *numerical* difference in using the second-order perfect B-spline as compared to the fourth-order perfect B-spline.

2.3. Theoretical Implementation Issues

The implementation of the Fourier–wavelet method for simulating a random field $v(x)$ is based on two kinds of computations, random weight evaluations and kernel evaluations. The method evaluates the product of the value of a kernel at a point

$$f_m(2^{-m}x - \lfloor 2^{-m}x \rfloor - n) \tag{2.21}$$

with a random weight

$$\gamma(m, \lfloor 2^{-m}x \rfloor + n) \tag{2.22}$$

and sums the result over $m = m_{\min}, \dots, m_{\max}$ and $n = n_{\min}, \dots, n_{\max}$, as in (2.16). Given that all we know about the functions f_m are their Fourier transforms, we solve the problem of accurate and efficient computation of the inverse Fourier transform of the functions f_m . Given that most random number generators produce random variables in a fixed sequence, we also recall our technique for generating the elements of a random sequence in an

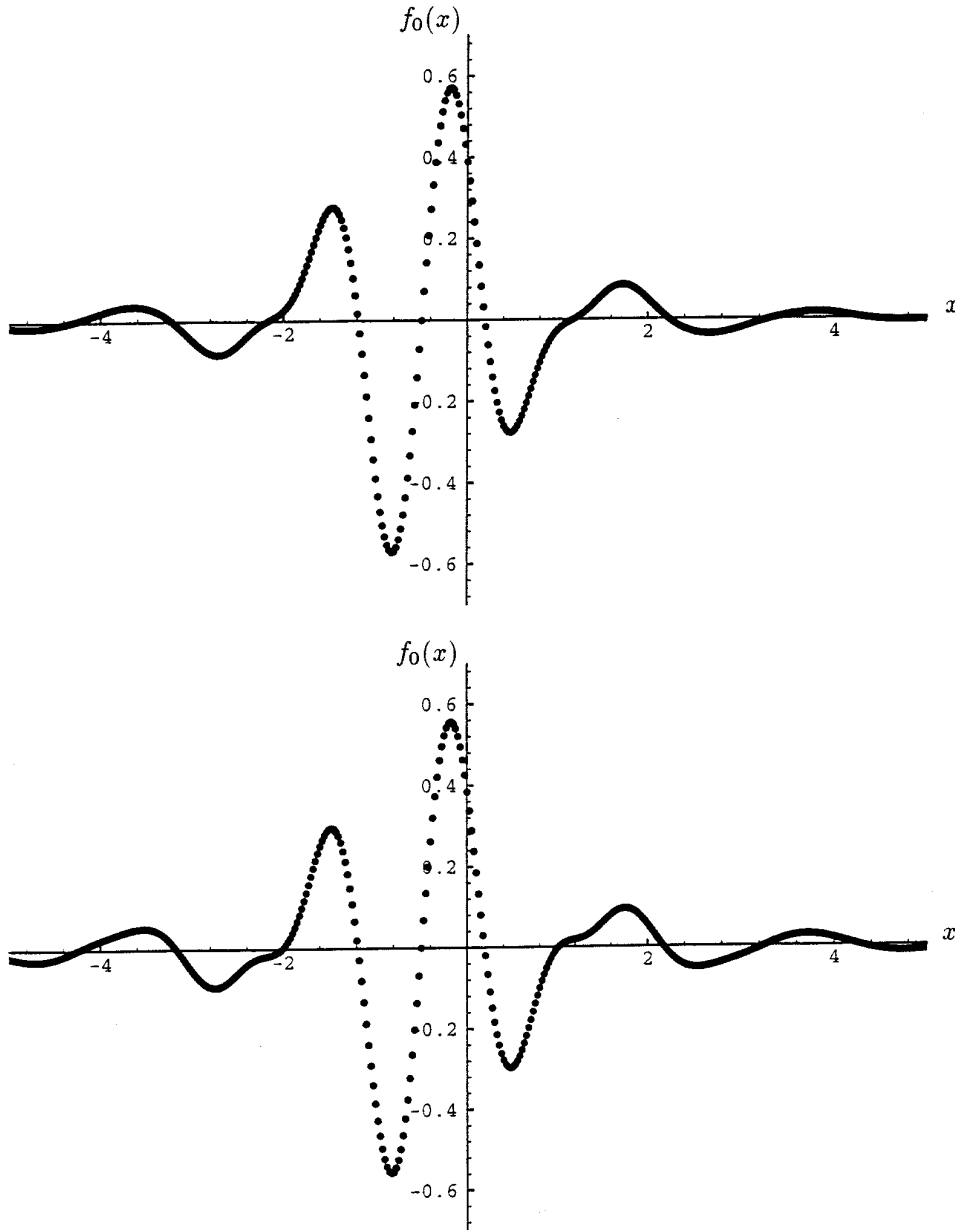


FIG. 1. Demonstration of the effect calculating f_0 using the Meyer wavelet based upon the second order perfect B-spline (upper plot) versus the fourth order perfect B-spline (lower plot). Note that f_0 decays slightly more rapidly in the lower plot.

arbitrary order, without storing previously generated elements [5, 6, 19].

Before considering the distinct issues in computing the random and deterministic terms, we caution the reader that careful computation of the integer and fractional parts of $2^{-m}x$ is necessary for the accurate computation of both the random and deterministic terms. Therefore, we suggest that (2.14) be computed in decreasing order of scale 2^m , rescaling the formula so that $m \leq 0$. We also suggest that the integer and fractional parts of $2^{-m}x$ be computed recursively from $2^{-m-1}x$.

2.3.1. Accurate Evaluation of the Kernels

We now proceed to the computation of the kernels f_m . The computation of f_m is done in two phases. In the pre-computation phase, $f_m(y)$ is computed on a regular grid using an inverse fast Fourier transform (FFT). In our case of interest, f_m has a power-law spectrum so that only f_0 needs to be computed. In general, the inverse FFT produces an *aliasing error*, which is the error in approximating f_m by a trigonometric polynomial. Furthermore, since the inverse FFT only computes f_m on a grid, interpolation is

required between grid points. Therefore, in the evaluation phase, $f_m(y)$ is interpolated, which induces an *interpolation error*. Below, we bound the aliasing and interpolation errors in terms of the resolution of the FFT grid and the properties of the Fourier transform of f_m . We expect these error estimates to be poor in comparison to error estimates obtainable through successive refinements of the inverse FFT. However, we give these estimates to provide a qualitative analysis of the error.

Aliasing Error in the Inverse FFT. Our analysis of the effect of aliasing begins with discretizing the inverse Fourier transform integral. To compute $f_m(y)$ from $(\mathcal{F}f_m)(k)$, we discretize the variables as $y = y'\alpha$ and $k = k'\beta$, where y' and k' are integer indices and α^{-1} and β^{-1} are large natural numbers. Using a trapezoidal rule for integration, we obtain

$$f_m(\alpha y') = \int_{-\infty}^{+\infty} e^{i2\pi y'k} (\mathcal{F}f_m)(k) dk \quad (2.23a)$$

$$\doteq \sum_{k' \in \mathbb{Z}} \beta (\mathcal{F}f_m)(\beta k') e^{i2\pi y' k' \alpha \beta} \quad (2.23b)$$

$$= \sum_{| \beta k' | \leq K} \beta (\mathcal{F}f_m)(\beta k') e^{i2\pi y' k' \alpha \beta}. \quad (2.23c)$$

Here, the truncation in the sum in (2.23c) arises because the support of $(\mathcal{F}f_m)(k)$ is limited to $|k| \leq K$. Therefore, the approximation of $f_m(\alpha y')$ is a trigonometric polynomial. The sum in (2.23c) can be computed as an inverse FFT on a vector of $\alpha^{-1}\beta^{-1}$ elements.

Since (2.23c) approximates $f_m(\alpha y')$ by a function with period β^{-1} , we examine the accuracy of this approximation over an interval with length smaller than β^{-1} . The truncation in (2.14) implies that we only need to evaluate $f_m(y)$ on the interval (n_{\min}, n_{\max}) . Therefore, we assume that

$$(n_{\max} - n_{\min} + 1) \leq \frac{1}{2}\beta^{-1}. \quad (2.24)$$

We use the Poisson summation formula (see [18]) and (2.17) to estimate the *aliasing error*,

$$\begin{aligned} & \left| f_m(\alpha y') - \sum_{| \beta k' | \leq K} \beta (\mathcal{F}f_m)(\beta k') e^{i2\pi y' k' \alpha \beta} \right| \\ &= \left| f_m(\alpha y') - \sum_{k' \in \mathbb{Z}} f_m(\alpha y' - \beta^{-1} k') \right| \leq 2pA_{mp}(2\beta)^p. \end{aligned} \quad (2.25)$$

Therefore, the error due to aliasing is $O(\beta^p)$ so that the smoothness of $(\mathcal{F}f_m)(k)$ reduces the aliasing error as well as the truncation error (2.15).

Interpolation Error. Since $(\mathcal{F}f_m)(k)$ has compact support, we know that $f_m(y)$ is a real analytic function which converges everywhere. Therefore, $f_m(y)$ can be accurately

approximated by interpolation to all orders. This allows us to evaluate $f_m(y)$ on grid points $y = \alpha y'$, where y' is an integer and to interpolate between grid points. Assuming that α^{-1} and β^{-1} are large natural numbers, we can write (2.23c) as an inverse discrete Fourier transform

$$f(\alpha y') = \sum_{| \beta k' | \leq K} \beta (\mathcal{F}f)(\beta k') e^{i2\pi y' \alpha k' \beta}. \quad (2.26)$$

Thus, $f_m(y)$ can be precomputed on a grid before the simulation of the field $v(x)$ begins. When $f_m(y)$ is evaluated at a point y which is off the grid, we interpolate linearly according to the usual scheme:

$$\begin{aligned} y_0 &= \alpha \lfloor \alpha^{-1} y \rfloor \\ s &= (y - y_0) \alpha \end{aligned} \quad (2.27)$$

$$f_m(y) \doteq f_m(x_0) + s(f_m(y_0 + \alpha) - f_m(y_0)).$$

This linear interpolation has accuracy $O(\alpha^2)$ according to

$$\begin{aligned} & | \{ f_m(y_0) + s(f_m(y_0 + \alpha) - f_m(y_0)) \} - f_m(y) | \\ & \leq \alpha^2 \sup_{\xi} | f_m''(\xi) | \\ & = \alpha^2 (2\pi)^2 \left(\frac{4}{3}\right)^2 A_{m0}. \end{aligned} \quad (2.28)$$

2.3.2. Efficient Generation of Random Variables

The Fourier–wavelet method (2.14) gives a sparse, accurate expansion of the random field $v(x)$; however, great care must be taken in computing the terms $\gamma(m, n)$ if the value of this sparsity is to be realized. Equation (2.14) demonstrates that the number of random terms needed to represent $v(x)$ at all points in the interval $[0, 2^{m_{\max}}]$ is $O(2^{m_{\max} - m_{\min}})$. However, the number of terms necessary to compute $v(x)$ at a single point is $(n_{\max} - n_{\min} + 1)(m_{\max} - m_{\min} + 1)$. Therefore, it would be desirable to compute the terms $\gamma(m, n)$ as needed rather than to precompute them and to store them. Previously, two of the authors have described a means of computing the necessary random variables on demand using a reversible random number generator so that very little memory is required for the generation of random variables [5, 6, 19].

2.4. Brief Comparison of the Fourier–Wavelet and Multiwavelet Methods

We now assess the new method that we have developed and implemented. As demonstrated in (2.14) and (2.15), the Fourier–wavelet method was designed so that it maintains the localization in space of the multiwavelet method (see [5]). This property gives the multiwavelet method its ability to simulate random fields efficiently over an unprecedented range of scales. As we show in the next

section, the Fourier–wavelet method inherits this same ability. Furthermore, the Fourier–wavelet method attains improvements in simplicity, speed, and flexibility. We now enumerate these improvements.

The Fourier–wavelet method is much simpler to implement and faster than multiwavelet method. In the precomputation phase, the Fourier–wavelet method uses a standard FFT to compute the kernels f_m , while the multiwavelet method exactly computes the corresponding functions using customized code. Furthermore, at run time, the Fourier–wavelet method evaluates the kernel through simple interpolation while the multiwavelet method performs an expensive nonlinear function evaluation.

The Fourier–wavelet method is also much more flexible than the multiwavelet method. For example, the Fourier–wavelet method is applicable to fields with general spectra, whereas the multiwavelet method specifically relies on the presumed power law structure of the spectrum. Furthermore, because the Fourier–wavelet method is a spectral method, it is compatible with spectral techniques for solving partial differential equations. The multiwavelet method has no such compatibility with anisotropic spectra generated by solutions of partial differential equations.

3. SIMULATION RESULTS USING THE FOURIER–WAVELET METHOD

In this section, the Fourier–wavelet method from Section 2 is computationally validated using several different analytically known statistical quantities for various velocity fields including a simple shear layer and an isotropic two-dimensional random field. A comparison is also made with the simulation results obtained through the use of another well-known, but nonhierarchical, simulation method, the randomization method. It will be shown that the Fourier–wavelet approach gives very accurate results and can be readily used to simulate random fields containing many decades of scaling behavior. In fact, the Fourier–wavelet method will be shown to be much more readily expandable than the randomization method through the clever use of random number generation techniques such as the “on demand” random number generator described in the previous section and in [5, 19]. However, the Fourier–wavelet method is computationally faster than the randomization method only for fields containing more than four or five decades of scaling. On the other hand, the Fourier–wavelet technique simulation results show much less deviation from Gaussian behavior than do the results from the randomization method.

This section begins with a brief introduction to the instructive model problem of turbulent transport with a simple shear layer velocity field [10, 12, 20]. The statistical quantities of interest such as the structure function and mean square dispersion are introduced and analytic formu-

las are given for the simple shear layer model. This is followed by a comparison of simulation results using the Fourier–wavelet method with the analytic behavior. The randomization method is then briefly reviewed and simulation results are compared with those results previously described from the Fourier–wavelet method. Finally, a two-dimensional isotropic random field is simulated using the random shearing direction approach in conjunction with both the Fourier–wavelet method and the randomization method and the results from both of these methods are compared.

3.1. The Simple Shear Layer Model Problem

We begin by describing the problem of turbulent transport of a passive scalar by a simple shear layer velocity field,

$$\mathbf{v} = \begin{pmatrix} \bar{w} \\ v(x) \end{pmatrix}, \quad (3.1)$$

where \bar{w} is a constant mean flow and $v(x)$ is a stationary, mean 0, Gaussian random field with a Kolmogoroff spectrum

$$E(k) = |k|^{-5/3}. \quad (3.2)$$

This problem is of particular interest because many statistical quantities with nontrivial behavior can be computed exactly for this problem, making it ideal for use as a benchmark for computer simulations [10, 12, 20]. The first statistical quantity that is used as a benchmark depends only on the velocity field and is known as the structure function,

$$D(x) = \langle (v(x) - v(0))^2 \rangle. \quad (3.3)$$

For the velocity field with the spectrum as in (3.2), it is possible to analytically compute the structure function as seen in [12, 20] to be

$$D(x) = \frac{2^{8/3} \pi^{5/3}}{\sqrt{3} \Gamma(\frac{8}{3})} x^{2/3}. \quad (3.4)$$

The structure function is useful as a measure of the range of validity of the simulated velocity field.

Another useful statistical quantity that depends only on the velocity field is the kurtosis. The kurtosis of a random field is the ratio of the fourth moment to the square of the second moment and must take the value 3 for a Gaussian field such as that considered here,

$$\frac{\langle (v(x) - v(0))^4 \rangle}{\langle (v(x) - v(0))^2 \rangle^2} = 3. \quad (3.5)$$

Simulation of the kurtosis is important as a means of observing deviations from Gaussian behavior as well as determining the range of validity of the simulation.

The final statistical quantity considered in this section depends on the particle trajectory in the y -direction. The components of the particle trajectory can be obtained by integrating (3.1) to get

$$\begin{aligned} X(t) &= \bar{w}t + X(0) \\ Y(t) &= \int_0^t v(X(s)) ds + Y(0). \end{aligned} \quad (3.6)$$

By following the trajectory of various particles, it is possible to gain a greater understanding of the spread of the passive scalar. The physically interesting quantity studied in the simulations here is known as the mean square dispersion and is defined by

$$\langle l_y^2 \rangle(t) = \langle (Y_1 - Y_2)^2 \rangle, \quad (3.7)$$

where Y_1 and Y_2 are the trajectories associated with two different particles which are chosen here to start a distance x_0 apart in the x -component and at the same value of the y -component. As was shown in [12, 20], it is possible to analytically compute the mean square dispersion for the simple shear layer model problem in (3.1) and (3.2) and obtain the exact formula

$$\begin{aligned} \langle l_y^2 \rangle(t) &= \frac{2^{14/3} \pi^{35/6} \Gamma(-\frac{4}{3})}{\bar{w}^2 \Gamma(\frac{14}{6})} \left[-|x_0|^{8/3} - |\bar{w}t|^{8/3} \right. \\ &\quad \left. + \frac{1}{2} |x_0 - \bar{w}t|^{8/3} + \frac{1}{2} |x_0 + \bar{w}t|^{8/3} \right]. \end{aligned} \quad (3.8)$$

3.2. Simulation Results for the Simple Shear Layer Model Problem

In this section we give some simulation results for the simple shear layer problem just described in Section 3.1 using the Fourier–wavelet method as introduced and described in Section 2. Section 3.2.1 features some structure function and kurtosis results. Section 3.2.2 contains a comparison of the results in Section 3.2.1 with simulation results from the well-known randomization method [8] and contains the conclusions that can be made based upon the structure function and kurtosis results. A discussion of mean square dispersion simulation results is given in Section 3.2.3.

3.2.1. Structure Function and Kurtosis Simulation Results

In this section, some structure function and kurtosis simulation results using the Fourier–wavelet method as introduced in Section 2 are given. All of these results are given

in terms of variables rescaled so that the largest scale in (2.16) is 1; i.e., the outer sum in (2.16) is over the octaves $M = m_{\max} - m_{\min}$ to 0. We begin by describing some parameter choices that will be in common to all the structure function and kurtosis simulations for the simple shear layer problem.

From looking at the formulation of the Fourier–wavelet method in (2.16), it is clear that there are several parameter choices that need to be made. The first issue relates to the order of the spline used in generating the f_m as defined in (2.11b). From the results in Fig. 1 which show f_m generated using the second-order perfect B-spline and the fourth-order perfect B-spline, one would expect that the impact of the choice between these two splines would be negligible on the simulation results. In fact, this behavior was observed in results not shown here; thus, all results given here will be based upon f_m being generated by the second-order perfect B-spline B_2 . Another parameter that must be chosen is the bandwidth of the inner sum in (2.16). Certainly from looking at the rapid decay of f_m in Fig. 1 it is clear that a relatively modest bandwidth is required. In results not shown, a bandwidth as small as 10 gave accurate results; to be conservative, the results given here use a bandwidth of 20 ($n_{\min} = -10$ and $n_{\max} = 9$). Another issue that always arises in simulation is the number of realizations over which averaging should be done; here, it was found that 2000 realizations would give accurate results.

The first structure function simulation results using the Fourier–wavelet method are given in Fig. 2. The top part of this figure contains the simulated values as dots and the analytic value from (3.4) as a solid line. The bottom part of this figure is a plot of the ratio of the simulated value to the analytic value and is useful as a measure of the error in the simulation. This simulation contains $M = 18$ octaves which is the maximum number of octaves for which our workstation is able to straightforwardly precompute and store all of the Gaussians which could be needed in the summation in (2.16). It is interesting to note that roughly three decades of accurate scaling are obtained in the middle scales with marked underestimation at both extremes of this almost 5.5 decade simulation region. Thus, if one were required to precompute and store the necessary random variables, it would not be possible to achieve more than around three decades of correct scaling on a typical workstation.

The plots in Fig. 3 depict the structure function simulation results using the Fourier–wavelet method when $M = 40$ octaves are included in the field. Because of the special nature of this simulation in that all of the points of field evaluation are known in advance thereby making known which of the Gaussians in (2.16) will be used, it is possible to compute the random numbers as they are needed for each succeeding point without using a reversible random number generator. This technique was quite useful here

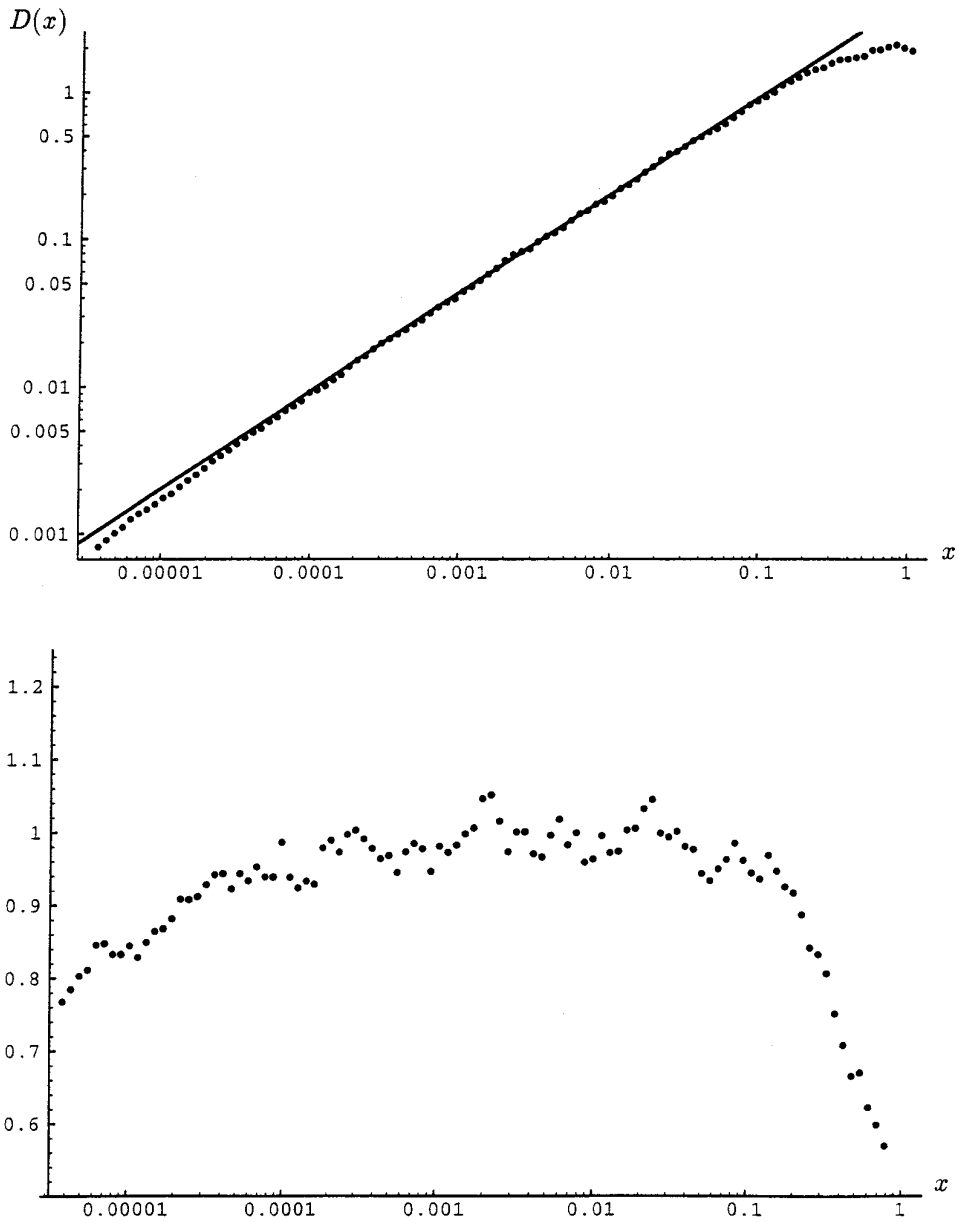


FIG. 2. Structure function simulation results for the simple shear layer model problem using the Fourier-wavelet method with $M = 18$ octaves, a bandwidth of 20, and 2000 realizations. The upper plot shows the simulated structure function as dots and the analytic value as a solid line; the lower plot is the ratio of the simulated value to the analytic value. Around three decades of accurate scaling are observed here.

as a means of quickly studying the technical capabilities of the Fourier-wavelet method without introducing the programming complications associated with the reversible random generator. The validity of this approach was verified computationally by comparing 18 octave simulations with complete storage of random numbers and with random number generation on this “as needed” basis. The results in Fig. 3 for the $M = 40$ octave simulation show quite good agreement with the analytic value over the

middle nine decades of this over 12-decade simulation. Further evidence of the quality of the simulation over these nine decades can be seen by considering the log-log least squares fit of the exponent which is 0.668 versus a true value of $\frac{2}{3}$ while the fit of the coefficient is 0.635 versus a true value of 0.639. It is also interesting to compare these results with those shown in Fig. 2; an increase in the number of decades in the overall field yielded a corresponding increase in the number of decades of accurate results; i.e.,

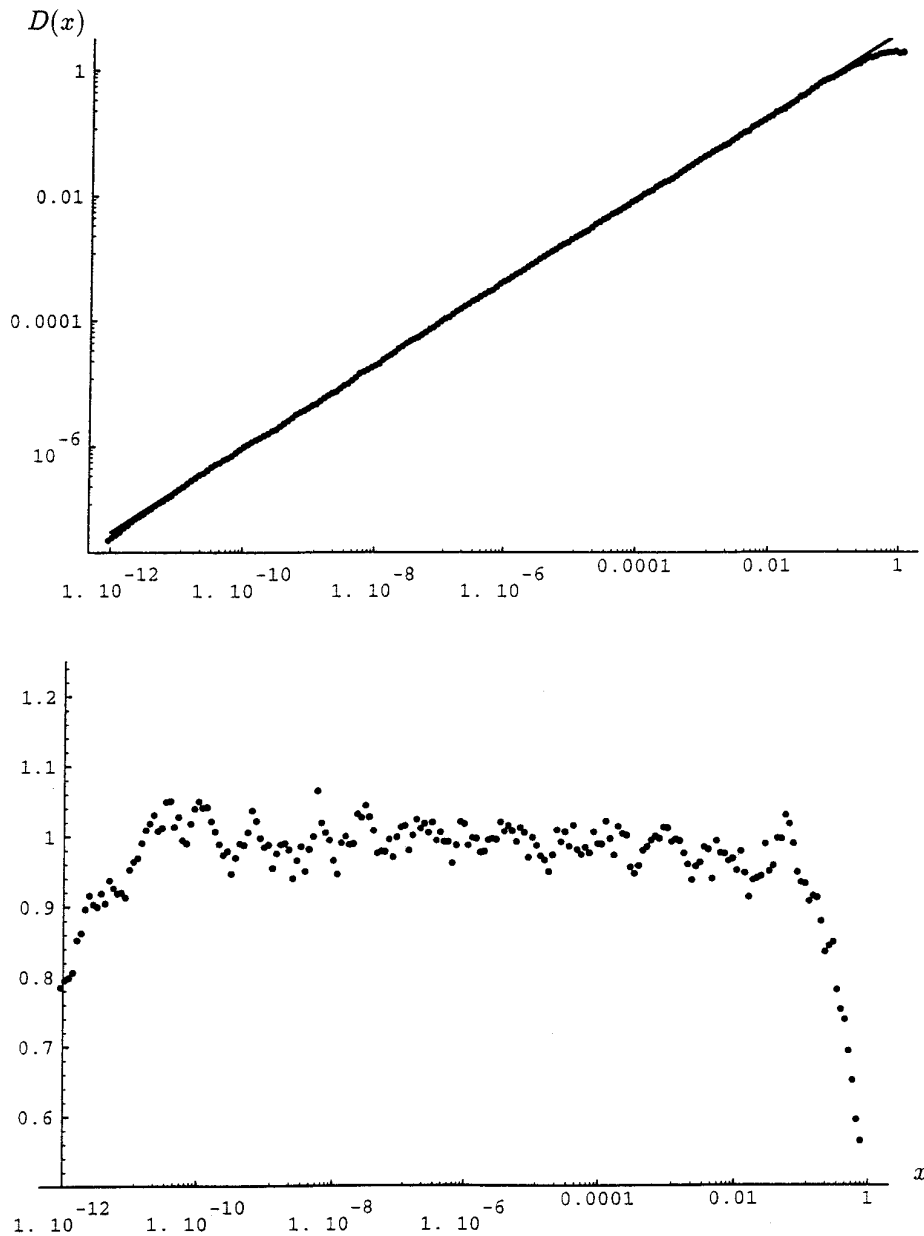


FIG. 3. Structure function simulation results for the simple shear layer model problem using the Fourier–wavelet method with $M = 40$ octaves and the other parameters unchanged from Fig. 2. Around nine decades of accurate scaling behavior are observed here.

there is a linear relationship between the number of octaves included in (2.16) and the number of decades of scaling behavior.

Figure 4 contains the Fourier–wavelet method simulation results for the kurtosis as in (3.5) using a field with $M = 40$ octaves. The simulated results tend to oscillate within 0.5 of the true value of 3 over around nine decades. It is important to note that the nine decades of most accurate results correspond with the same decades of accurate results seen in Fig. 3 for the structure function. Thus, the

Fourier–wavelet method gives highly Gaussian results over its region of validity.

3.2.2. Comparison with Simulation Results for the Randomization Method

The purpose of this section is to provide a computational basis of comparison for the Fourier–wavelet method in order to help elucidate the strengths and weaknesses of this new method. While the above results are definitely

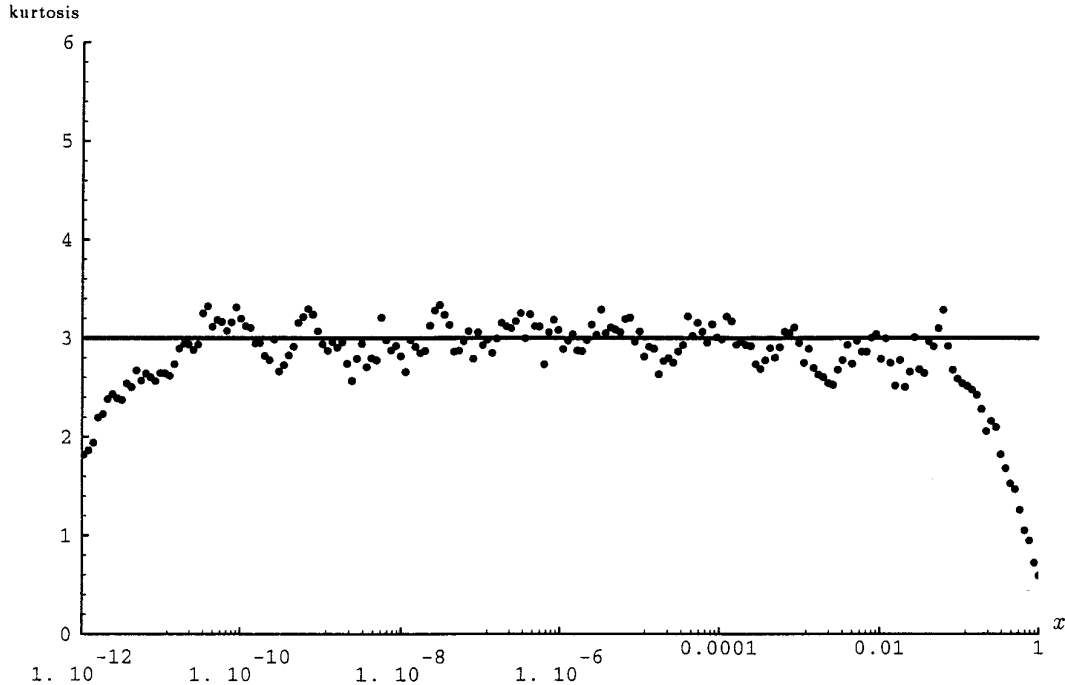


FIG. 4. Kurtosis simulation results for the simple shear layer model problem using the Fourier–wavelet method with the same parameters as in Fig. 3. The nine decades of accurate scaling behavior seen here are the same decades as observed in Fig. 3.

quite encouraging, much additional insight can be gained by making a comparison with the commonly used randomization method [8]. In this section, we begin by briefly describing the randomization method for random field simulation. This is followed by a description of the simulation results for the structure function and kurtosis. Comparisons regarding such issues as expandability and computational difficulty are made with the appropriate results using the Fourier–wavelet method.

3.2.2.A. Introduction to the Randomization Method. We recall that (2.2a) gives the theoretical basis upon which the Fourier–wavelet method is derived. This formula also provides the basis for the randomization method [8, 12]. The randomization method results from a discretization of the integral with randomly spaced wave number nodes. It is typical to select the normalized spectral density function as the probability density function for the wave numbers. In other words, $p(k) = (1/v_0^2)E(k)$ when $v_0^2 = \int_{-\infty}^{\infty} E(k) dk$. As was shown in [8, 12], dividing the spectrum into equal energy partitions is a useful means of ensuring that every realization of the velocity field contains a typical distribution of scales. For the case with \tilde{M} partitions, let $\lambda_0, \lambda_1, \dots, \lambda_{\tilde{M}-1}, \lambda_{\tilde{M}}$ be the partition points. For component i , the probability density function of the wave numbers is

$$p_i(k) = \begin{cases} p(k), & \lambda_{i-1} \leq k < \lambda_i \\ 0, & \text{otherwise.} \end{cases} \quad (3.9)$$

Using these definitions, each realization of the random field can be written as

$$v_a(x) = \frac{1}{\sqrt{\tilde{M}}} \sum_{i=1}^{\tilde{M}} v_0 [a_i \cos(2\pi k_i x) + b_i \sin(2\pi k_i x)], \quad (3.10)$$

where $a_i, b_i \sim \text{Gsn}(0, 1)$ and $k_i \sim p_i$ with all of these random variates independent of each other. It is important to point out that in (3.10) each k_i must come from a distinct range of wave numbers, thereby ensuring that every realization is always influenced by the most energetic part of the spectrum.

We now give specific details regarding how to use the randomization method for a problem with a fractal velocity field. Certainly for our spectrum in (3.2), an infrared cutoff is necessary to give the velocity field finite energy ($v_0^2 < \infty$). Here a step function cutoff is used, $E(k) = 0$ for $k \leq k_{\min}$. For the Kolmogoroff spectrum in (3.2) with this cutoff, the partition points become

$$\lambda_i = k_{\min} \left(\frac{\tilde{M} - i}{\tilde{M}} \right)^{-3/2}. \quad (3.11)$$

The largest scale included in this simulation is determined by the smallest wavenumber to be $1/k_{\min}$. A conservative, yet quite reasonable, estimate of the smallest scale in the simulated field, especially for larger values of \tilde{M} , can be

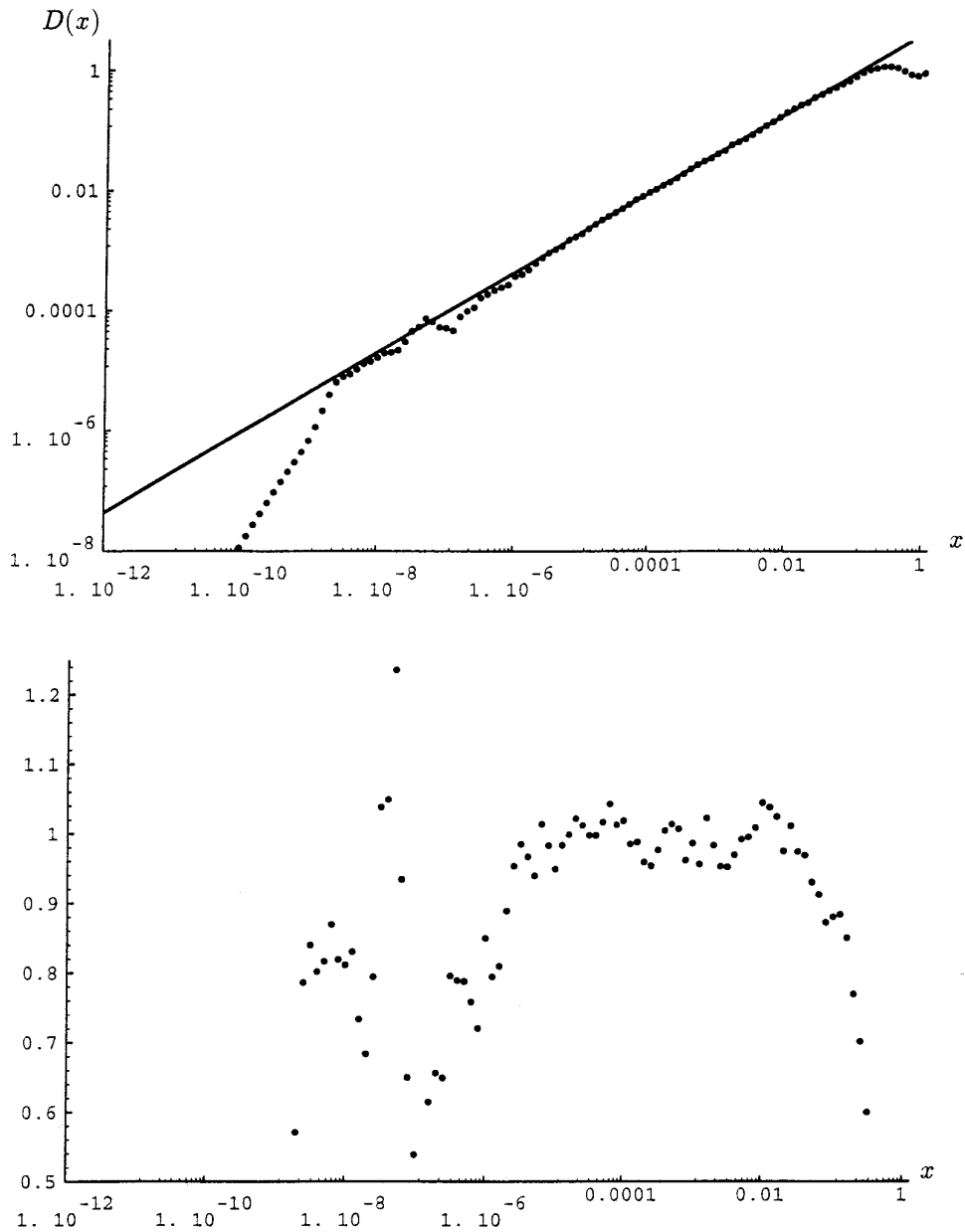


FIG. 5. Structure function simulation results for the simple shear layer model problem using the randomization method with $\tilde{M} = 256$ compartments and 2000 realizations. In the upper plot, the simulated values are dots while the analytic value is a solid line; the lower plot depicts the ratio of the simulated value to the analytic value. Around four decades of scaling behavior are observed here.

obtained by considering the largest noninfinite partition point, thus giving $(1/k_{\min})\tilde{M}^{-3/2}$ as the smallest scale in the problem. Thus, we estimate that the simulated field contains $\tilde{M}^{3/2}$ scales or $\frac{3}{2}\log(\tilde{M})$ decades. In order to make the largest scale in our problem consistent with the choice for the Fourier-wavelet results, we select $k_{\min} = 1$.

3.2.2.B. Structure Function and Kurtosis Randomization Method Results. Figure 5 contains the structure function simulation results using the randomization method with

$\tilde{M} = 256$ compartments and 2000 realizations. We observe roughly four decades of accurate scaling behavior which is roughly what was predicted above, based upon the partition points. In results not shown, the simulation with $\tilde{M} = 16$ compartments had about 3.5 decades of accurate scaling which is much larger than the two decades predicted by the partition points; the $\tilde{M} = 4096$ compartment simulation had accurate scaling over around 5.5 decades which was in line with the prediction. The enhanced agreement between the observed region of scaling and that predicted

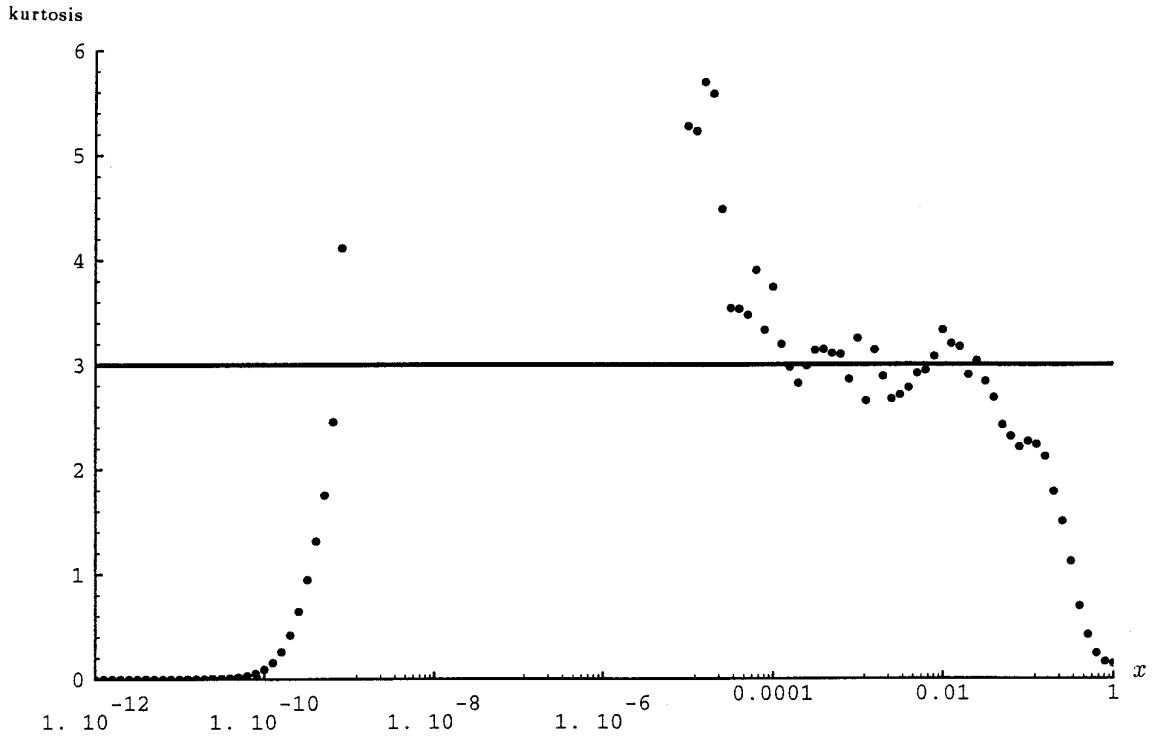


FIG. 6. Kurtosis simulation results for the simple shear layer model problem using the randomization method with the same parameters as in Fig. 5. Only three decades of scaling behavior are seen here versus the four decades indicated by the structure function simulation in Fig. 5.

by the partition points for larger \tilde{M} is reasonable when one considers the fact that the partition point estimate neglects the factor of \tilde{M}^{-1} of the total energy on the smallest scales. It is important to point out here that the two to three decades of rather noisy behavior on the scales below the scaling regime before the final decay to 0 on the smallest scales as seen in Fig. 5 is quite typical behavior, regardless of the number of compartments used.

The kurtosis results in Fig. 6 are from the randomization method with $\tilde{M} = 256$ compartments and 2000 realizations. The simulated kurtosis is within 0.5 of the true value of 3 for roughly three decades and takes explosively large values on the small scales before finally decaying to 0. It is important to point out that the kurtosis is accurate over fewer decades than the corresponding structure function; while both quantities begin to agree with the theoretical value at the same large scale, the kurtosis dramatically deviates from the true value about one decade earlier than does the structure function in Fig. 5. This sort of behavior was also observed for other values of the number of compartments which are not included here. Thus, we see that the randomization method is much more accurate for second-order statistics than for higher order statistics.

3.2.2.C. Comparison of Fourier-Wavelet Results with Randomization Method Results. We now compare the

results above with those described in Section 3.2.1 for the Fourier-wavelet method. One of the first noteworthy differences in results has to do with the fact that the statistical quantities decay much more smoothly to 0 on the scales below the scaling regime when using the Fourier-wavelet method for both the structure function and especially the kurtosis. The kurtosis from the randomization method was several times too large at several locations, whereas the kurtosis was never as large as twice its true value in the Fourier-wavelet simulations. Unlike the randomization method, the Fourier-wavelet method results also have accurate kurtosis results over the entire scaling regime observed in the structure function simulations. Thus, the Fourier-wavelet method clearly produces a more Gaussian-like field over the scaling regime and has much less noise in small scales below the scaling regime.

Another interesting comparison has to do with the ability to expand the region of validity of each simulation method. Here we assume that one is interested in statistical quantities that depend only on second-order moments; if accurate higher order moments are important, then the Fourier-wavelet method is more well-suited, as was discussed in the above paragraph. On a most basic level, we saw that the region of validity of the simulation results tended to grow linearly as the number of octaves M included by the Fourier-wavelet method was increased; on

the other hand, the scaling regime only grew logarithmically with the number of compartments \bar{M} for the randomization method. Of course, it is necessary to consider both computer memory limitations and computational difficulty to get a complete picture of this comparison.

Memory limitations play a much greater role in the Fourier–wavelet method than in the randomization method. If one precomputed all the Gaussians that could possibly be needed in (2.16) for the Fourier–wavelet method, only about three decades of scaling in the simulated random field can be achieved due to memory limitations on a typical workstation. However, through the clever use of a random number generator such as the reversible random number generator described in [5, 6, 19] one is able to eliminate the storage difficulties associated with the Fourier–wavelet method by only generating those random numbers that are actually used in the sum in (2.16). The randomization method is unable to produce fields with more than about 10 decades due to memory limitations that cannot be eliminated through the use of the reversible random number generator because of the nonlocal nature of the randomization method (3.10); i.e., every velocity field evaluation (3.10) requires every Gaussian weight, as well as every wave number. However, the memory problem is never an issue in randomization method simulations in practice, since it is not computationally feasible to evaluate (3.10) for the huge number of compartments required.

We now assume that the Gaussian weights used in the Fourier–wavelet representation in (2.16) are generated on an “on demand” basis, using the reversible random number generator. As we will discuss below, there is a very clear trade-off in computational difficulty between the Fourier–wavelet method and the randomization method. This is determined by looking at the number of transcendental operations required for each evaluation of the random field since each transcendental operation is much more computationally difficult than a single floating point operation. For example, the Fourier–wavelet method requires one Gaussian (about two transcendental operations on average) for each term in the double sum in (2.16) while the randomization method requires two trigonometric evaluations per compartment as in (3.10). If one ignores the necessary calculations to obtain the Gaussian weights and the wave numbers for each realization of the randomization method, it is clear that the crossover in computational efficiency occurs roughly when the number of compartments in the randomization method exceeds the product of the bandwidth and number of scales in the Fourier–wavelet method. This crossover occurs for simulations where roughly four to five decades of scaling are desired in the results. Thus, for simulations with fewer than four to five decades of scaling behavior, the randomization method is more computationally efficient; for more than four to five decades of scaling behavior, the Fourier–

wavelet method, in conjunction with the reversible random number generator, is more computationally efficient.

3.2.3. Mean Square Dispersion Simulation Results

We now turn our attention to a study of the mean square dispersion related to the simple shear layer model problem in (3.1). We recall from (3.7) that the mean square dispersion depends on the expected value of the square of the difference between two particle trajectories which we select to begin at the origin and at $(x_0, 0)$. Since the mean square dispersion depends only on second-order statistics, the randomization method and the Fourier–wavelet method should give very similar results for the mean square dispersion. This is actually the case. Here, we focus on the results using the Fourier–wavelet method. All of the time integrations required here are computed through the use of the trapezoidal rule.

The remainder of this section is organized as follows. We begin with a discussion of the rescaling of this problem. Section 3.2.3B contains a discussion of various parameter choices, all of which were validated computationally. Some simulation results using the Fourier–wavelet method follow in Section 3.2.3C, and conclusions are given in Section 3.2.3D.

3.2.3.A. Rescaling the Problem. The mean square dispersion results are given in this section in terms of variables rescaled according to physical scales of the problem. A natural length scale associated with the physical problem is the initial separation distance, $L = x_0$, which is used to rescale the particle trajectory Y . There are also two time scales associated with the physical problem, the eddy turnover time τ_v and the sweep time $\tau_{\bar{w}}$. The sweep time is related to the strength of the mean field and is defined by

$$\tau_{\bar{w}} = L/\bar{w}. \quad (3.12)$$

The eddy turnover time is a measure of the strength of the random velocity field v . We recall from (3.4) that the structure function is $D(x) = C|x|^{2/3}$ which has units $\text{length}^2/\text{time}^2$. Thus, the eddy turnover time can be written as

$$\tau_v = L/(\sqrt{C}L^{1/3}). \quad (3.13)$$

Time in these results is rescaled by the eddy turnover time τ_v making the relevant time scales in the results $\tau_v = 1$ and $\tau_{\bar{w}}/\tau_v$.

3.2.3.B. Parameter Choices for Simulations. There are many simulation parameters that must be chosen for these mean square dispersion simulations. Many of these choices are inspired by the structure function and kurtosis results

in Section 3.2.1. For instance, the bandwidth of the inner sum in (2.16) is chosen to be the amply large value 20, i.e., $n_{\min} = -10$ and $n_{\max} = 9$. The values of f_m are calculated here based on the second-order perfect B-spline B_2 since higher order splines did not give improvements earlier. Using 1000 realizations is adequate to achieve accurate results. All of these choices were also confirmed for the mean square dispersion in simulation results not shown.

The choice of the time step was determined by a systematic and careful simulation study. First the time step Δt was determined relative to τ_v by studying a situation in which τ_w/τ_v was quite large. This study showed that Δt was not particularly sensitive to the size of τ_v but should nonetheless be chosen less than 0.5 of τ_v . Then the situation in which τ_w/τ_v was small relative to τ_v was used to determine that $\Delta t \leq 0.1$ (τ_w/τ_v) is appropriate. Thus, in all the simulations presented here,

$$\Delta t < 0.5, \quad \Delta t \leq 0.1 (\tau_w/\tau_v) \quad (3.14)$$

are used to determine the time step. For the specific runs given in the next subsection, $\tau_w/\tau_v \approx 1.3$ and $\Delta t = 0.1$.

3.2.3.C. Mean Square Dispersion Simulation Results Using the Fourier–Wavelet Method. The goal of this section is to discuss the effect of varying the number of scales included in the simulated field in (2.16). We recall from the structure function and kurtosis results that the simulated field tends to be an underestimate at the smallest scales and at the largest scales. In order to avoid the problems at the smallest scales, we consider a simulated field with $m_{\min} = -16$ and use an initial particle separation distance $x_0 = 2^{-12}$. These simulations are run to time 500 at which time the value of the x -trajectory is of order 2^{-4} . The simulation results in Fig. 7 are for $m_{\max} = -2$ and show a very clear underestimation of the mean square dispersion by the simulation. In fact, the underestimate has grown to in excess of 20% by time 500, showing quite clearly that more than two octaves must be included above the largest scale. The results in Fig. 8 with $m_{\max} = 0$ (four octaves above the largest scale) show agreement within 6% of the true value from (3.8) throughout the entire time range. In fact, further evidence of the accuracy of this simulation can be seen by considering the log–log least squares fit over times 100 to 500. The fit for the exponent is 0.651 versus a true value of 0.676, while the simulated coefficient is 0.185 versus a true value 0.125. Thus four octaves need to be included above the largest scale in the problem to avoid underestimation problems.

In results not shown here, the effect of additional adjustments in m_{\max} was studied. A further increase in the size of m_{\max} did not result in a further enhancement in the accuracy of the results; in fact, the results tended to be a

bit less accurate. This result can be explained by considering the form of the Fourier–wavelet method in (2.16) and the definition of the mean square dispersion in (3.7); having an excessive number of octaves above the largest scale in the problem will require the subtraction of large numbers in order to get differences several orders of magnitude smaller thereby leading to potential dynamic range issues. Another reason for not including an excessive number of octaves in the field is the savings in computational time that results from not doing these unnecessary calculations.

3.2.3.D. Conclusions from Mean Square Dispersion Simulation Results. Many interesting things regarding the use of the Fourier–wavelet simulation method have been learned from these mean square dispersion simulations. First, we have noted that all of the parameter choices that were made for the structure function and the kurtosis carry through to this problem with integrated velocity fields. This fact can be quite important for simulations in which we are able to compute statistics such as a structure function but have no analytic formulas for the statistical quantities in which we are interested. We have also shown that it is important to include around four octaves above the largest scale in which one is interested in order to avoid the underestimation effect at the upper extreme of the simulated field; such extraneous effects could easily go unnoticed in a simulation in which the analytic behavior of the desired quantity is unknown.

3.3. The Two-Dimensional Random Velocity Field

In this section, the two-dimensional problem is introduced and a summary of results is given for some exactly analytically computable statistical quantities such as the structure tensor. At the end of this section, the random shearing direction simulation technique is introduced. This technique allows one to straightforwardly generalize the one-dimensional techniques used in the previous section to higher dimensional problems.

3.3.1. Introduction to the Two-Dimensional Field

We now turn our attention to the problem of simulating a two-dimensional, isotropic, Gaussian random field \mathbf{v} with mean 0 and radial spectral density function

$$E(|k|) = |k|^{-5/3}. \quad (3.15)$$

One of the statistical quantities in which we are interested is the structure tensor which is a direct generalization of the structure function as was defined in (3.4). For the two-dimensional \mathbf{v} now under consideration, the structure tensor can be defined as

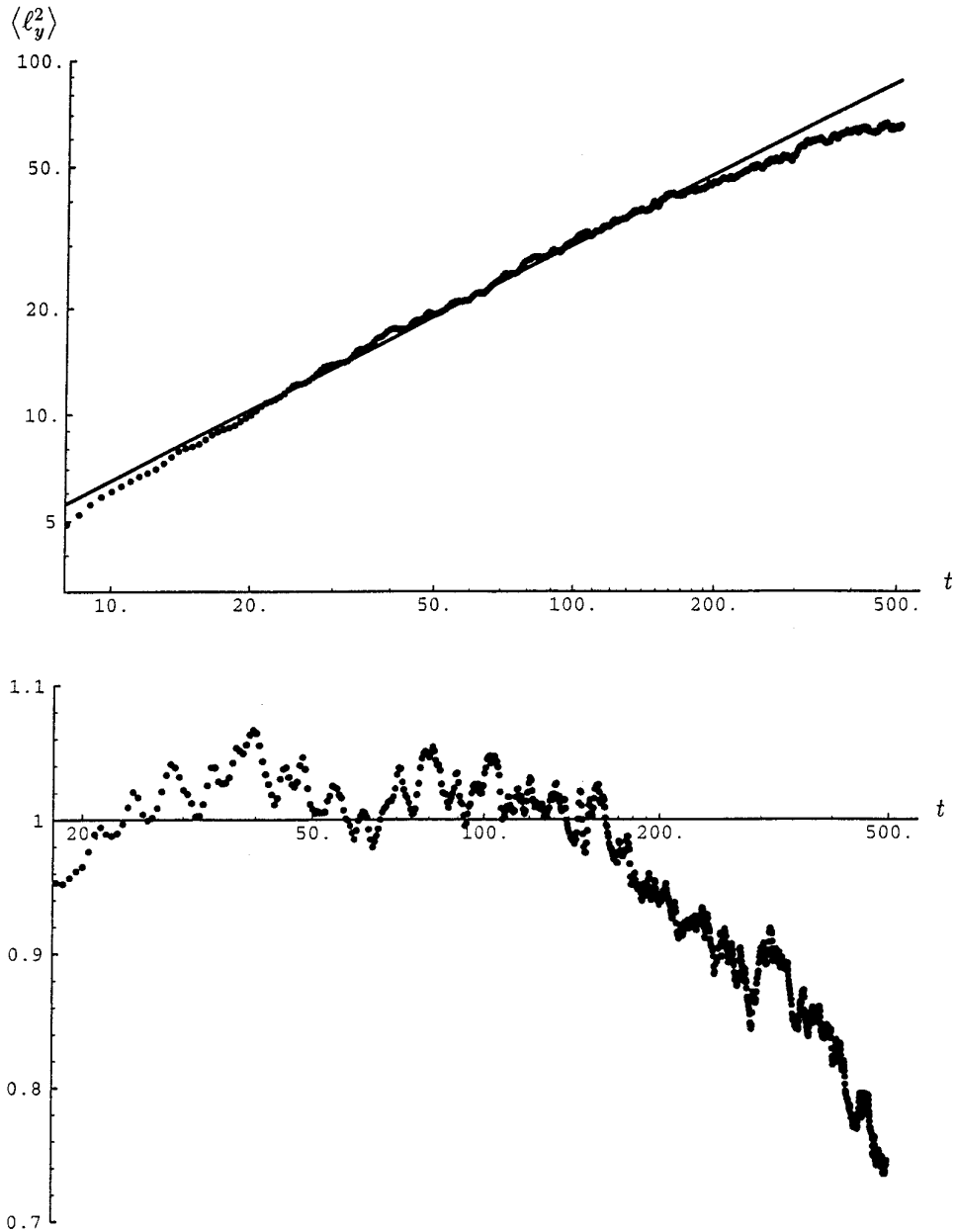


FIG. 7. Mean square dispersion simulation results for the simple shear layer model problem using the Fourier-wavelet method with $m_{\min} = -16$, $m_{\max} = -2$, and an initial particle separation $x_0 = 2^{-12}$. The simulated values are dots and the analytic value is a thin solid line in the upper plot; the ratio of the simulated value to the analytic value is given in the lower plot. At longer times, the simulated value greatly underestimates the true value demonstrating the difficulties that arise near the largest scales in the simulated field.

$$D(\mathbf{x}) = \langle (\mathbf{v}(\mathbf{x} + \mathbf{x}') - \mathbf{v}(\mathbf{x}')) \otimes (\mathbf{v}(\mathbf{x} + \mathbf{x}') - \mathbf{v}(\mathbf{x}')) \rangle. \quad (3.16)$$

$$D(\mathbf{x}) = \begin{pmatrix} \lambda_{\parallel} & 0 \\ 0 & \lambda_{\perp} \end{pmatrix}, \quad (3.17)$$

Through the use of the covariance tensor, it is possible to analytically compute the structure tensor for the choice $\mathbf{x} = (\delta)$. Details of this calculation are given in [6]. The results are

where λ_{\parallel} and λ_{\perp} are the parallel and perpendicular eigenvalues with

$$\lambda_{\parallel} = \frac{27\pi^{1/6}}{2^{7/3}} \Gamma\left(\frac{5}{6}\right) x^{2/3} \quad (3.18)$$

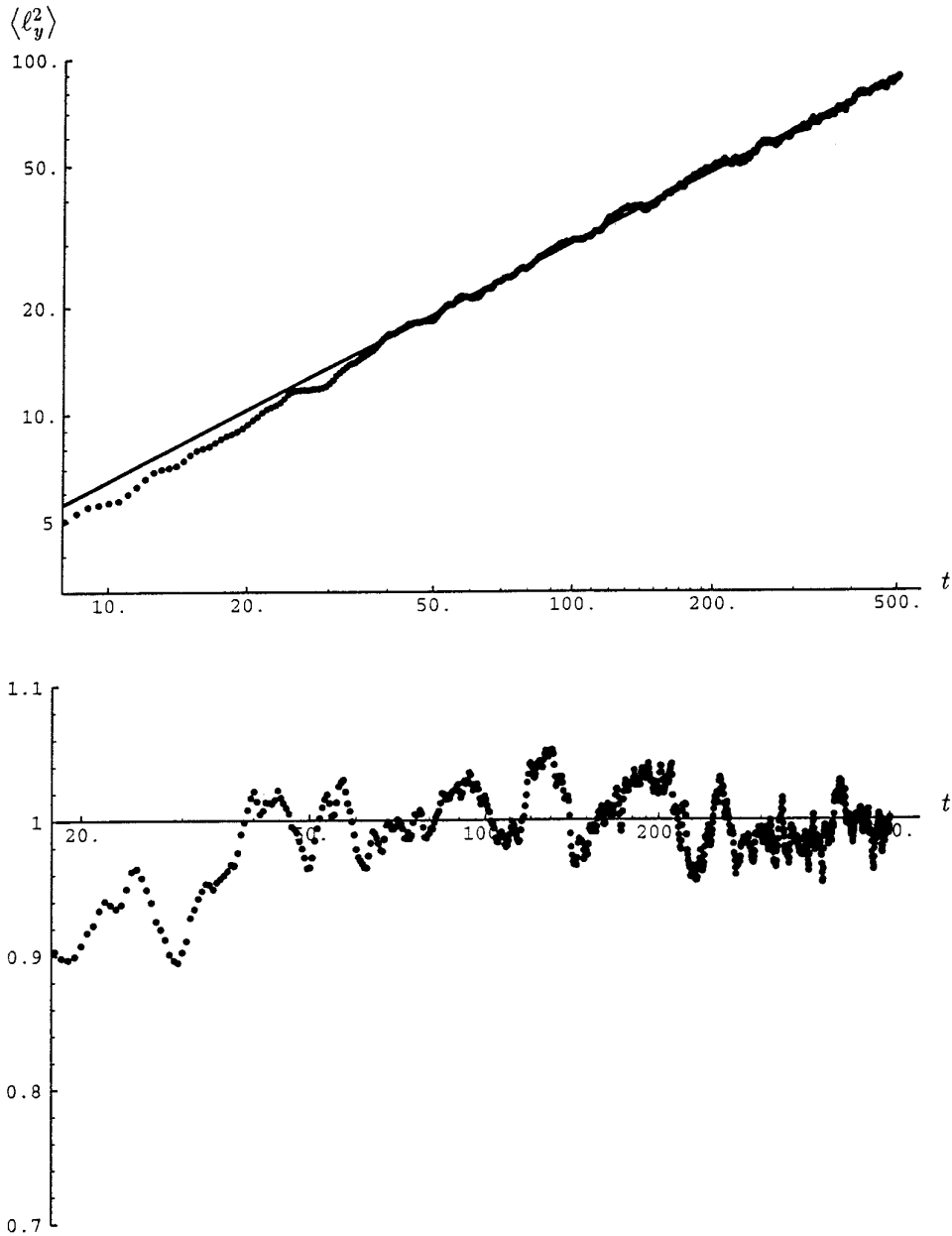


FIG. 8. Mean square dispersion simulation results for the simple shear layer model problem using the Fourier-wavelet method with m_{\max} increased to $m_{\max} = 0$ and all other parameters unchanged from Fig. 7. Good agreement between simulation and theory is seen throughout the times plotted here.

and

$$\lambda_{\perp} = \frac{27\pi^{1/6}}{2^{4/3}} \Gamma\left(\frac{11}{6}\right) x^{2/3}. \quad (3.19)$$

In this work, we will focus on λ_{\parallel} to assess the quality of our simulation results. Since the field is isotropic, it is not a limitation that the above formulas assume that

\mathbf{x} lies along the x -axis; one merely must rotate so that the point of interest appears to lie along the positive x -axis.

The other statistical quantity of interest here is the kurtosis. By analogy with the kurtosis in (3.5) for the one-dimensional case, we know that each component of the velocity field should also have kurtosis 3. Just as for the structure tensor, we will focus on the velocity field component parallel to the x -axis.

3.3.2. Description of Random Shearing Direction Method

The random shearing direction technique has been described in detail in [6, 15] and will only be briefly summarized here for the two-dimensional case, although, of course, it can be written in a more general form for higher dimensions. Intuitively, this method can be thought of as taking several one-dimensional random fields and placing them in various directions on a circle. Symbolically this method may be written as

$$\mathbf{v}_a(\mathbf{x}) = \frac{\sqrt{\pi}}{\sqrt{M_d}} \sum_{j=1}^{M_d} \boldsymbol{\omega}^\perp v_{a_j}(\mathbf{x} \cdot \boldsymbol{\omega}_j), \quad (3.20)$$

where M_d is the number of directions used, $\boldsymbol{\omega}_j$ is the unit vector for each one of the M_d directions, and v_{a_j} is one realization of a one-dimensional random field which can be computed using *any* simulation technique. In the following section, both the Fourier–wavelet and the randomization methods will be used to generate each v_{a_j} . In [15] it is shown that the representation in (3.20) is theoretically quite accurate for stationary, isotropic, incompressible Gaussian random fields such as those considered here. It is proven in [6] that the best strategy (lowest variance) means for selecting the unit isotropic vectors $\boldsymbol{\omega}_j$ in two dimensions is to choose them equally spaced on the unit circle. It is also shown in [6] through the use of an energy criterion that it is desirable to consider the number of directions M_d to be at least 16, regardless of the method employed to generate each v_{a_j} .

3.4. Simulation Results for the Two-Dimensional Random Velocity Field

In this section we summarize the results obtained through the use of the random shearing direction method (3.20) in conjunction with the Fourier–wavelet method (2.16) and the randomization method (3.10). We remark here that our version of the randomization algorithm in two dimensions is new. The standard randomization method [8] simply makes random directional assignments in each energy shell. The version of the randomization method utilized here has much lower variance than the standard version. All of the results given below are calculated using 2000 realizations and $M_d = 32$ directions. The directions were offset from the positive x -axis by an angle of $\pi/9$ in order to eliminate the possible effects of calculating statistics along one of the shearing directions. The results are also given in rescaled coordinates so that the longest length scale in the simulated velocity field is 1.

3.4.1. Structure Tensor and Kurtosis Results Using the Fourier–Wavelet Method

In this subsection, results are given from simulations using the Fourier–wavelet method in conjunction with the

random shearing direction technique. In addition to the simulation parameters already mentioned above regarding number of realizations and shearing directions, all results in this section use a bandwidth of 20 in the inner sum in (2.16) and f_m is calculated based upon the fourth-order perfect B-spline. The number of octaves ($M = 13$) included in each shearing direction is the maximum that can be straightforwardly stored on a typical workstation when all the Gaussians are precomputed.

Figure 9 contains the simulation results for λ_{\parallel} of the structure tensor. Just as for the previous figures, in the top plot the dots represent the simulation results while the solid line is the analytic value from (3.18). The bottom plot shows the ratio of the simulated value to the analytic value and is useful as a measure of the error. Because of the very limited nature of the field, only about 1.5 decades of scaling behavior are observed. While this range of scaling behavior is in line with what one would expect, based upon the one-dimensional results from Section 3.2, it certainly points out that the only way in which one can practically use the Fourier–wavelet method in two dimensions is through the clever use of random number generation techniques.

Simulation results for the kurtosis of the component of the velocity field parallel to the x -axis using the Fourier–wavelet method in conjunction with the random shearing direction approach are contained in Fig. 10. Again, almost 1.5 decades of reasonable agreement with the true value of 3 is seen. (The kurtosis was within 0.4 of the true value over this region.) Just as in one-dimension, the kurtosis gives valid results over the same region of validity as the structure tensor and decays rather smoothly to 0 on the largest and smallest scales.

3.4.2. Structure Tensor and Kurtosis Results Using the Randomization Method

Some simulation results using the randomization method in conjunction with the random shearing direction technique are now given. Each of the $M_d = 32$ shear direction in these results has $\tilde{M} = 256$ compartments. Just as before, averaging is done over 2000 realizations.

The plots in Fig. 11 contain simulation results for the λ_{\parallel} component of the structure tensor. Around 5.5 decades of accurate scaling behavior can be seen in these plots. When compared with the one-dimensional results in Fig. 5, we see a little more than one additional decade of correct scaling behavior. The small scales below the scaling regime are also a bit less noisy than those observed in the one-dimensional simulations. The results for other numbers of compartments \tilde{M} also showed similarly enlarged regions of scaling; this behavior is indeed quite reasonable when one considers the fact that using M_d directions is effectively like taking M_d samples from each compartment, thereby

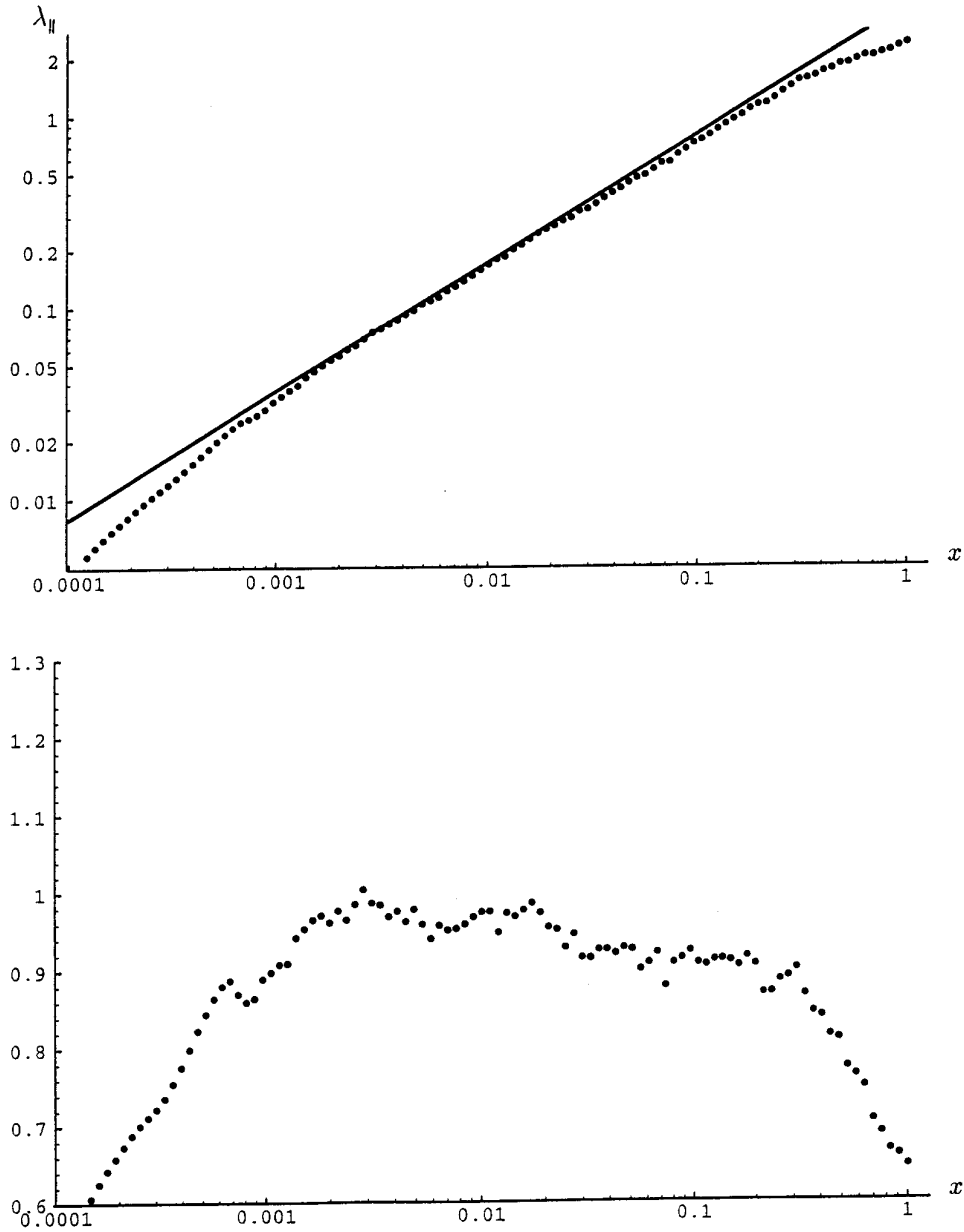


FIG. 9. Simulation results for λ_{\parallel} of the structure tensor for the two-dimensional random field using the Fourier-wavelet method in conjunction with the random shearing direction technique with $M_d = 32$ directions and $M = 13$ octaves. The simulation results are dots while the analytic value is the solid line in the upper plot. The lower plot is the ratio of the simulated value to the analytic value. Around 1.5 decades of scaling behavior are observed.

giving just a bit less than $\log M_d$ additional decades. Thus, while the same basic trends that held in one dimension also hold here, the simulation results in two-dimensions tend to be less noisy and contain a scaling regime enhanced in size by the number of directions used when compared with the one-dimensional results.

The kurtosis of the component of the velocity parallel to the x -axis is shown in Fig. 12. The simulated kurtosis stays within 0.5 of the true value of 3 for almost four

decades before becoming explosively large on the smaller scales. When compared with the one-dimensional case, there is roughly one additional decade of accurate scaling behavior for the same reasons mentioned above for the structure tensor. Just as seen in the one-dimensional case, the kurtosis is accurate only over the largest scales in the scaling regime in the structure tensor and has very noisy behavior on the scales below this abbreviated scaling regime. Thus, the same trends observed in Section 3.2 for

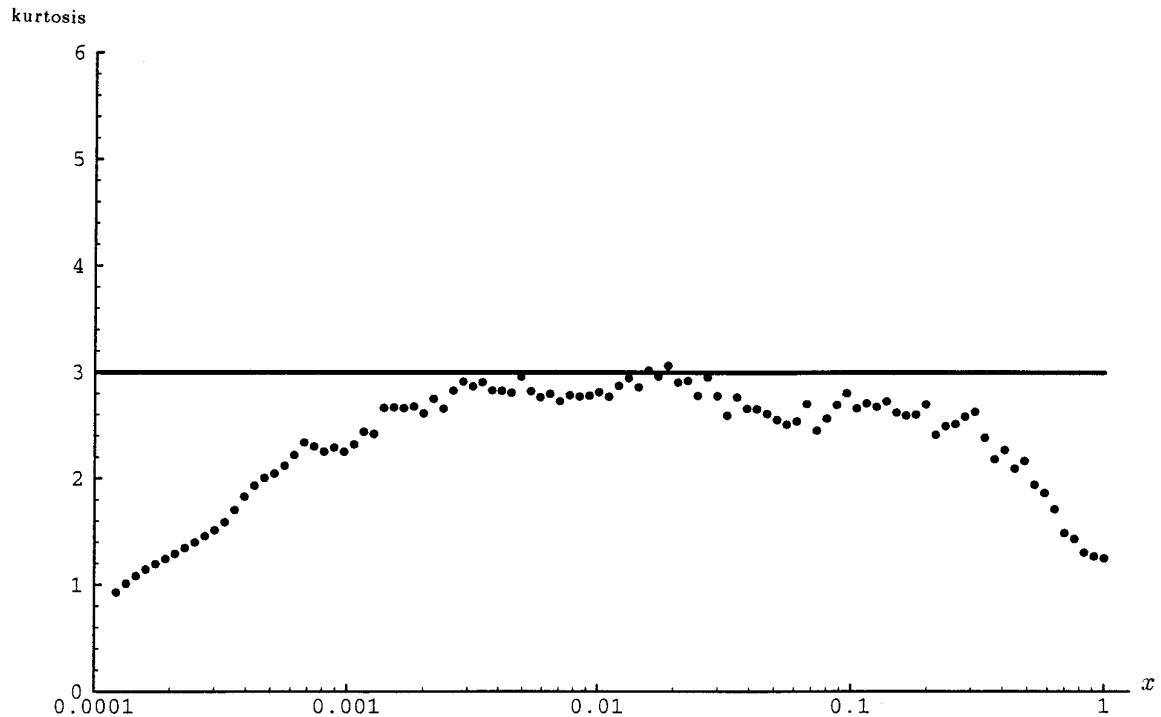


FIG. 10. Simulation results for the kurtosis of the component of the velocity field parallel to the x -axis using the same simulation technique and parameters as in Fig. 9. Scaling behavior is observed for the same 1.5 decades as in Fig. 9.

the simple shear layer model also hold here but with an enhanced region of validity.

4. SUMMARY AND CONCLUSIONS

In this paper, a new simulation method has been derived, analyzed, and implemented. In Section 2, the Fourier-wavelet method (2.16) was derived, based upon a mathematically rigorous wavelet decomposition of the stochastic representation formula (2.2a). This method as derived is applicable to a wide variety of processes, even those with infrared and ultraviolet singularities in their spectra. The rationale for the choice of wavelet basis (the Meyer wavelets) and the method of computing that basis using perfect B-splines are explained in terms of creating the most efficient simulation technique possible. Rigorous error bounds for the method are also derived in Section 2.

In Section 3, the Fourier-wavelet method was implemented for both a simple shear layer model problem (in which the randomness is confined to one component of the velocity field) and a two-dimensional random field. A comparison of the results from the Fourier-wavelet method and a version of the randomization method was made. Our version of the randomization algorithm in two dimensions is new and combines the one-dimensional randomization method with a random plane wave decomposition [6, 15]. The standard randomization method [8] simply

makes random directional assignments in each energy shell. The following are the major conclusions and observations made from these simulations:

- The Fourier-wavelet method gives results which have the same scaling regimes for both second- and fourth-order statistical quantities. The Fourier-wavelet results also showed a smooth decay to zero in those scales outside the scaling regime. On the other hand, the randomization method yielded smaller scaling regimes for higher order statistics compared with the scaling regime for the second-order statistics; the results outside the scaling regime were rather noisy before decaying to zero.

- It is worthwhile to use simulations of simpler quantities such as the structure function which depends only on the velocity field as guidance in the choice of simulation parameters for more complicated statistical quantities such as the mean square dispersion which depends on particle trajectories.

- Trends and comparisons made in the simple shear layer problem also applied in the two-dimensional problem with the exception that the plane wave randomization method simulation results had longer ranges of validity in two-dimensions as a result of Monte Carlo averaging through the number of shearing directions.

- In order to make practical use of the Fourier-wavelet method, an efficient means of “on demand” generation of

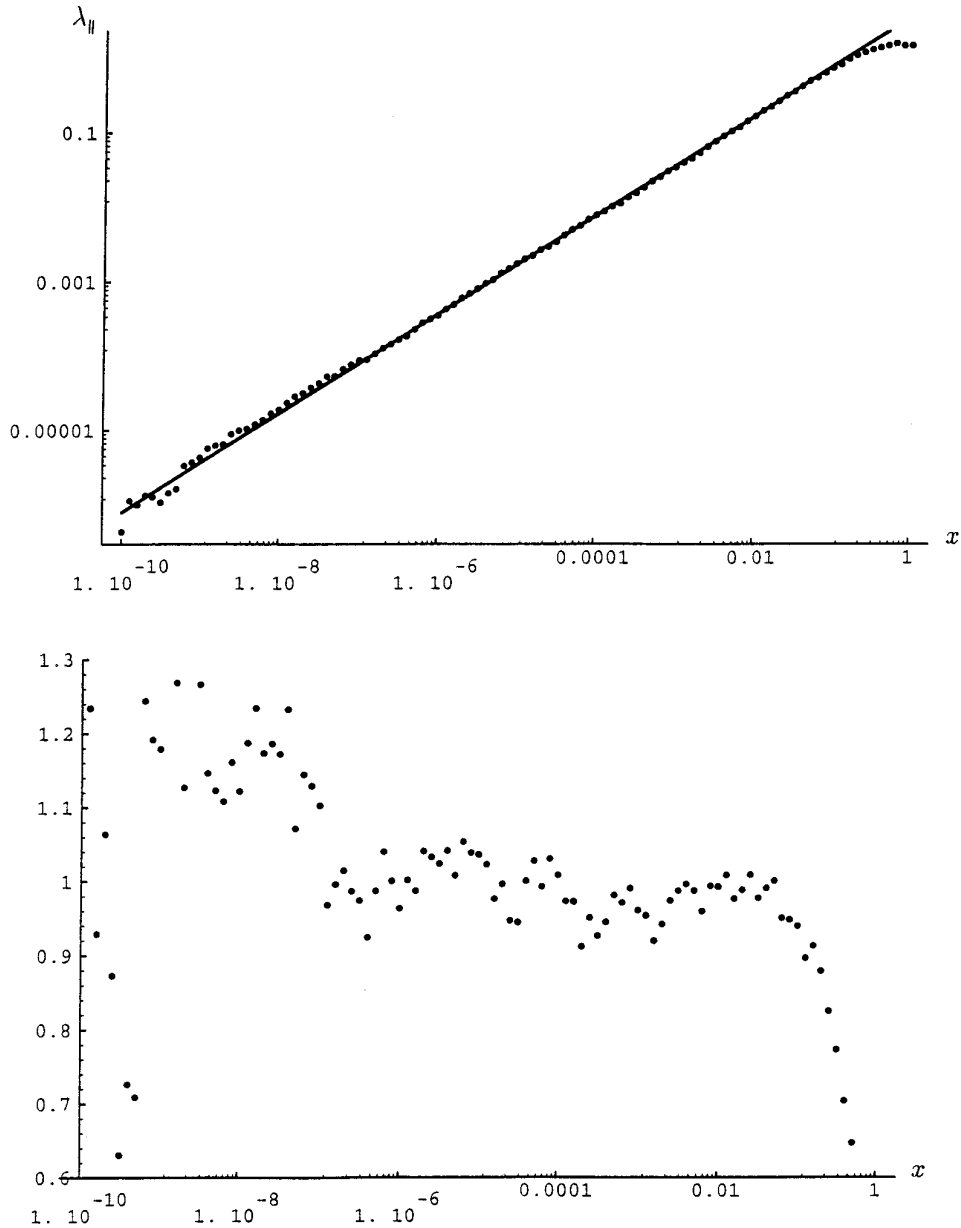


FIG. 11. Simulation results for $\lambda_{||}$ of the structure tensor for the two-dimensional random field using the randomization method in conjunction with the random shearing direction technique with $M_d = 32$ directions and $\tilde{M} = 256$ compartments. Nearly 5.5 decades of accurate scaling behavior are seen here; note that the longer scaling regime seen here when compared with Fig. 5 is a result of averaging over the shearing directions.

random numbers must be used. An example of such a generator is the reversible random number generator described in Section 2.

- If statistical quantities only depending on second moments are desired and if no more than four decades of scaling behavior are needed, then the randomization method is more computationally efficient. Otherwise, the Fourier-wavelet method in conjunction with the reversible random number generator should be used.

There are many other interesting problems that are related to turbulent transport simulations that are very much worth exploring. An accurate pair dispersion simulation to validate Richardson's law for the Kolmogoroff spectrum is one of these important problems. Another interesting extension of the methods described here is the simulation of time decorrelated random fields as well as fields with anisotropic spectra generated through solutions of partial differential equations. The authors are currently pursuing these issues.

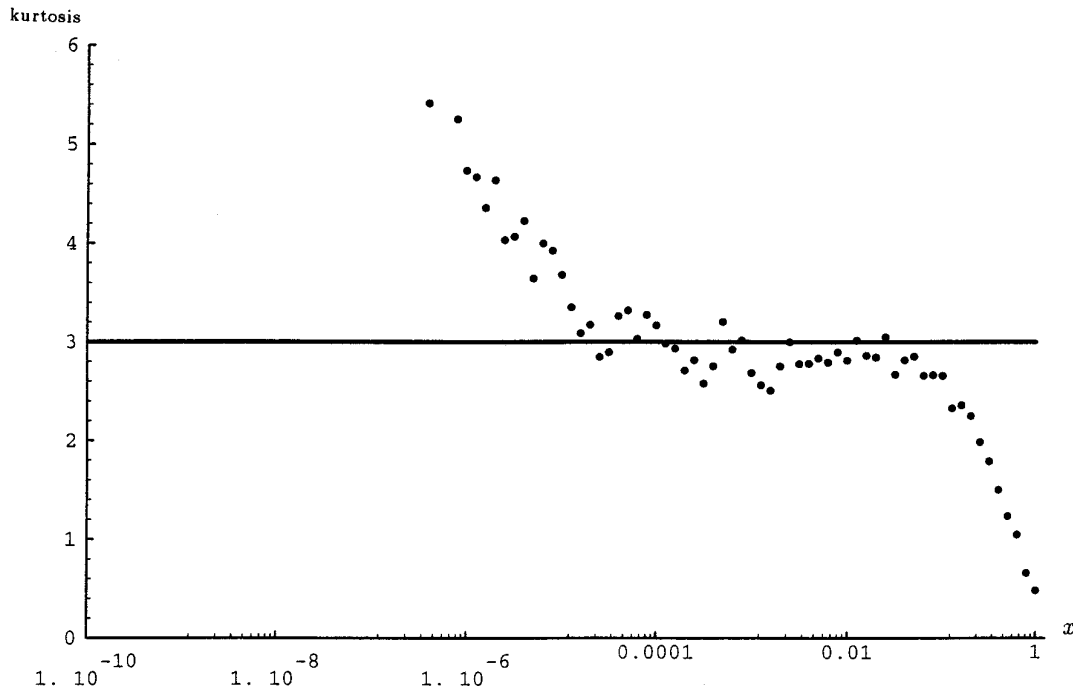


FIG. 12. Simulation results for the kurtosis of the component of the velocity field parallel to the x -axis using the same simulation technique and parameters as in Fig. 11. Only four decades of accurate scaling behavior are seen here versus the 5.5 decades indicated by the structure tensor simulation in Fig. 11.

APPENDIX A: DEFINITION AND RELEVANT PROPERTIES OF THE MEYER WAVELET

This appendix describes the structure and properties of the wavelet ϕ as related to the implementation of the Fourier–wavelet method. As discussed in Section 2.2, the structure and properties are best described in terms of the Fourier transform of the wavelet $(\mathcal{F}\phi)$. Accordingly, we give the required properties of $(\mathcal{F}\phi)$, define the Fourier transform of the Meyer wavelet, and discuss a heuristic for optimizing this wavelet for our purposes.

Our desire in choosing ϕ is to make the kernels $f_m(x)$ as easy to compute as possible. More specifically we would like the spectrum of the kernels,

$$(\mathcal{F}f_m)(k) = 2^{-m/2} E^{1/2} (2^{-m}k) (\mathcal{F}\phi)(k), \quad (\text{A.1})$$

to have p derivatives and to be compactly supported. Together, these two properties assure that $f_m(x)$ is an analytic function that decays like $A|x|^{-p}$ for some $A > 0$ as described in (2.13). The analyticity of the function $f_m(x)$ ensures that the interpolation error (2.28) is small, and the rapid decay ensures fast convergence of the truncated representation (2.14) and a small error due to aliasing (2.25). Generalizing from the case of a power law spectrum, we assume that $E(k)$ is smooth except for possible algebraic divergences at 0 and at ∞ . Therefore, $(\mathcal{F}f_m)(k)$ will have p derivatives

and be compactly supported provided that $(\mathcal{F}\phi)(k)$ has these properties and is 0 in the neighborhood of 0.

As we demonstrate, the spectrum of the Meyer wavelet $(\mathcal{F}\phi)$ has the desired properties. The spectrum $(\mathcal{F}\phi)$ of the Meyer wavelet ϕ is given by

$$(\mathcal{F}\phi)(k) = (2\pi)^{-1/2} \text{sgn}(k) e^{ink} b(|k|), \quad (\text{A.2})$$

where

$$\text{sgn}(k) = \begin{cases} -1, & k < 0, \\ 0, & k = 0, \\ +1, & k > 0, \end{cases}$$

and $b(k)$ is a p -times differentiable function with values in $[0, 1]$ and support in $[\frac{1}{5}, \frac{4}{5}]$. As noted earlier, because $b(k)$ limits the portion of $E^{1/2}$ that appears in $(\mathcal{F}f_m)(k)$, $b(k)$ is called a window function. The window function $b(k)$ has the formula

$$b(k) = s(\nu(\mu(k))) \quad (\text{A.3})$$

where

$$s(k) = \sin\left(\frac{\pi}{2}k\right) \quad (\text{A.4})$$

and

$$\mu(k) = \begin{cases} 0, & |k| < \frac{1}{3}, \\ 3(|k| - \frac{1}{3}), & \frac{1}{3} \leq |k| < \frac{2}{3}, \\ 1 + \frac{3}{2}(\frac{2}{3} - |k|), & \frac{2}{3} \leq |k| < \frac{4}{3}, \\ 0, & |k| \geq \frac{4}{3}. \end{cases} \quad (\text{A.5})$$

The function ν , which is described below, is a p -times differentiable transition function that assures that $b(k)$ is p -times differentiable while the function μ assures that $b(k)$ has the correct support.

Formulas (A.3), (A.4), and (A.5) do not determine an exact form for $(\mathcal{F}\phi)$ because the choice of the function $\nu(\cdot)$ is somewhat arbitrary (see [16]). The only requirement is that $\nu(\cdot)$ have p classical derivatives and satisfy the conditions

$$\begin{aligned} \nu(x) &= \begin{cases} 0, & x \leq 0, \\ 1, & x \geq 1; \end{cases} \\ \nu'(x) &\geq 0, \\ 1 &= \nu(x) + \nu(1-x). \end{aligned} \quad (\text{A.6})$$

One possible choice for ν which satisfies the conditions in (A.6) is the example given by Daubechies [16]

$$\nu_D(x) = \begin{cases} 0, & x \leq 0, \\ x^4(35 - 84x + 70x^2 - 20x^3), & 0 < x < 1, \\ 1, & x \geq 1, \end{cases} \quad (\text{A.7})$$

which has four classical derivatives. Given that $E^{1/2}(k)$ is smooth away from the origin, the four derivatives of $\nu_D(x)$ guarantee that $(\mathcal{F}f)$ has four bounded derivatives so that

$$\lim_{|x| \rightarrow \infty} |f(x)| |x|^4$$

is bounded. While the function in (A.7) would certainly be a reasonable choice for ν , in our calculations we make a different selection for the reasons discussed below.

We now describe a heuristic for optimizing the choice of ν in implementing the Meyer wavelet. Our objective is to control the numerical decay of $f_m(x)$ in (2.13). To do this we need to limit the *magnitude* of the derivatives of $(\mathcal{F}f_m)$, assuring that the coefficients A_{mp} in (2.13) are small. The magnitude of the derivatives of $(\mathcal{F}f_m)$ depends

strongly on the derivatives of $b(k)$; therefore, we try to choose $\nu(k)$ to limit the magnitude of the derivatives of $b(k)$. In the belief that the largest derivative of $\nu(k)$ will make the greatest contribution to the overall derivative of $b(k)$, we seek a $\nu(k)$ which will minimize the value of the highest order derivative.

We now pursue the function $\nu(k)$ with the smallest possible p th derivative so that we can use it to construct $(\mathcal{F}\phi)$. As stated in [17], every p -differentiable function $\nu_p(k)$ satisfying the conditions in (A.6) also satisfies the inequality

$$|D_+^p \nu_p(k)| \geq 4^{p-1}(p-1)! \quad (\text{A.8})$$

for all $0 < k < 1$, where D_+ is the derivative from the right. One function, the integral of the p th order perfect B-spline,

$$\begin{aligned} \nu_p(k) &= (-1)^p \frac{4^{p-1}}{p} \left((k-y_0)_+^p + (k-y_p)_+^p \right. \\ &\quad \left. + 2 \sum_{r=1}^{p-1} (-1)^r (k-y_r)_+^p \right) \end{aligned} \quad (\text{A.9})$$

with

$$y_r = \frac{1}{2} \left(\cos\left(\frac{p-r}{p}\pi\right) + 1 \right)$$

and

$$k_+ = \begin{cases} 0, & k \leq 0, \\ k, & k > 0, \end{cases}$$

actually achieves this lower bound (see [17, p. 137]). This family of functions satisfies our strategy for minimizing the derivatives of $(\mathcal{F}f_m)$ for all p . Therefore, we use it in implementing the spectrum of the Meyer wavelet $(\mathcal{F}\phi)$.

ACKNOWLEDGMENTS

Andrew J. Majda's research is partially supported by the following grants: National Science Foundation DMS-9596102, Army Research Office DAAH04-95-1-0345, and Office of Naval Research N00014-96-1-0043. David J. Horntrop and Frank W. Elliott Jr. were supported as postdoctoral researchers by grants from the Army Research Office (DAAH04-95-1-0345) and the National Science Foundation (DMS-9596102), respectively.

REFERENCES

1. G. Csanady, *Turbulent Diffusion in the Environment* (Reidel, Dordrecht, 1973).
2. G. Dagan, Theory of solute transport by groundwater, *Ann. Rev. Fluid Mech.* **19**, 183 (1987).

3. J. Feder, *Fractals* (Plenum, New York, 1988).
4. R. Voss, Random fractal forgeries, in *Fundamental Algorithms for Computer Graphics*, edited by R. Earnshaw (Springer-Verlag, Berlin, 1985), p. 805.
5. F. Elliott and A. Majda, A wavelet Monte Carlo method for turbulent diffusion with many spatial scales, *J. Comput. Phys.* **113**, 82 (1994).
6. F. Elliott and A. Majda, A new algorithm with plane waves and wavelets for random turbulent velocity fields with many spatial scales, *J. Comput. Phys.* **117**, 146 (1995).
7. F. Elliott and A. Majda, Pair dispersion over an inertial range spanning many decades, *Phys. Fluids* **8**, 1052 (1996).
8. K. Sabelfeld, *Monte Carlo Methods* (Springer-Verlag, New York, 1991).
9. J. Fung, J. Hunt, N. Malick, and R. Perkins, Kinematic simulation of homogeneous turbulence by unsteady random Fourier modes, *J. Fluid Mech.* **236**, 281 (1992).
10. F. Elliott, D. Hornthrop, and A. Majda, Monte Carlo methods for turbulent tracers with long range and fractal random velocity fields, *Chaos*, submitted.
11. M. McCoy, A numerical study of turbulent diffusion, Ph.D. thesis, Department of Mathematics, University of California at Berkeley, 1975.
12. D. Hornthrop, Monte Carlo simulation for turbulent transport, Ph.D. thesis, Program in Applied and Computational Mathematics, Princeton University, 1995.
13. J. Viccelli and E. Canfield, Functional representation of power-law random fields and time series, *J. Comput. Phys.* **95**, 29 (1991).
14. F. Elliott, A. Majda, D. Hornthrop, and R. McLaughlin, Hierarchical Monte Carlo methods for fractal random fields, *J. Stat. Phys.* **81**, 717 (1995).
15. A. Majda, Random shearing direction models for isotropic turbulent diffusion, *J. Stat. Phys.* **75**, 1153 (1994).
16. I. Daubechies, *Ten Lectures on Wavelets* (SIAM, Philadelphia, 1992).
17. L. Schumaker, *Spline Functions: Basic Theory* (Wiley, New York, 1981).
18. G. Folland, *Real Analysis*, (Wiley, New York, 1984).
19. F. Elliott, Efficient functional generation of Gaussian random variables, in preparation.
20. D. Hornthrop, and A. Majda, Subtle statistical behavior in simple models for random advection diffusion, *J. Math. Sci. Univ. Tokyo* **1**, 23 (1994).
21. P. Auscher, G. Weiss, and M. Wickerhauser, Local sine and cosine bases of Coifman and Meyer and the construction of smooth wavelets, in *Wavelets: A Tutorial in Theory and Applications*, edited by C. Chui (Academic Press, Boston, MA, 1992), p. 237.