JOURNAL OF COMPUTATIONAL PHYSICS 103, 231–242 (1992)

An Algorithm for Tracking Fluid Particles in a Spectral Simulation of Turbulent Channel Flow

K. KONTOMARIS AND T. J. HANRATTY

Department of Chemical Engineering, University of Illinois, Urbana, Illinois 61801

AND

J. B. MCLAUGHLIN

Department of Chemical Engineering, Clarkson University, Potsdam, New York 13676

Received October 30, 1990; revised November 1, 1991

The ability to follow individual fluid particles dispersing in a turbulent flow and to collect turbulence information along their trajectories is of key importance in many problems of practical and theoretical significance. With the availability of a direct numerical simulation of turbulence such information can be extracted directly from first principles without resorting to questionable assumptions. In this paper an algorithm for tracking fluid particles in a direct numerical simulation of turbulent channel flow is developed and tested. Fluid particle velocities are computed with an interpolation scheme that employs Lagrange polynomials of order 6 in the homogeneous directions of the channel and Chebyshev polynomials in the inhomogeneous normal direction. Errors in computed particle velocities and trajectories are assessed and it is shown that accurate single-particle Lagrangian statistics can be extracted both in the center and in the wall region of the channel.

1. INTRODUCTION

Analyses of turbulent transport are usually formulated with averaged conservation equations. These suffer from the well known closure problem in that unknown terms, introduced in the conservation equation by the averaging, have to be assumed or estimated in order to generate a closed system of equations.

An alternate approach, in which the physics emerges in a more natural way, has been available, since the pioneering work of Taylor [13] on "diffusion by continuous movements." Taylor described the dispersion process as the result of the random wandering of fluid particles in a turbulent field. Progress in using this formulation has been hindered because of the difficulties of studying Lagrangian characteristics of turbulence in the laboratory. However, with the availability of direct numerical simulations (DNS) of turbulent flows which provide instantaneous realizations of turbulent fields, tracing fluid particles, and extracting Lagrangian turbulence statistics becomes straightforward.

Deardorff and Peskin, as early as 1970, reported single- and two-particle Lagrangian statistics from a large eddy simulation (LES) of turbulent channel flow. Riley and Patterson [12] simulated particle diffusion in decaying isotropic turbulence. Yeung and Pope [17] (abbreviated to YP from now on), as well as Balachandar and Maxey [1] (BM), have also presented computed Lagrangian statistics extracted from a DNS of isotropic turbulence. Bernard *et al.* [2] used ensembles of particle paths to investigate the origin of Reynolds stress in a DNS of turbulent channel flow. McLaughlin [8] employed a numerical simulation of turbulent channel flow to investigate the dispersion and deposition of aerosol particles.

In numerical tracking experiments the instantaneous position of a particle does not, in general, coincide with a mesh point. A critical aspect of the calculation is, therefore, the evaluation of fluid particle velocities, by interpolating the known values of the Eulerian velocity fields at the mesh points. Research by (YP) and (BM) has thoroughly examined this problem for a homogeneous isotropic turbulence. The present paper explores the accuracy of various interpolation schemes for the case of a channel flow and makes recommendations for carrying out accurate and economical calculations.

Turbulent flow in a channel has a number of aspects which deserve special consideration: Considerable energy resides in smaller scales of motion, which are hard to interpolate accurately. Another difficulty stems from the presence of the wall which does not allow the application of periodicity of the velocity field in the direction normal to the channel walls. As a result, polynomial interpolation of high order, say nth, cannot be applied for particles which are

0021-9991/92 $5.00

within $n/2-1$ grid points away from the wall. This problem cannot be solved by reducing the order of interpolation close to the wall because accuracy would be sacrificed in a region where it is most needed. The high order of interpolation could be maintained by employing the necessary number of grid points, starting with the grid point at the wall. This approach would leave the fluid particle closer to one end of the interpolated interval. It is known, however, that high order polynomial interpolation may exhibit strong oscillations at the ends of the interval despite its accuracy in the middle.

In this paper the pseudospectral code of Lyons [5, 6] is used to simulate fully developed turbulent channel flow at a low Reynolds number. A large number of fluid particles are tagged simultaneously and their trajectories are traced by integrating the equation of particle motion. The most promising of the interpolation techniques proposed by (YP) and (BM) for isotropic turbulence are compared and evaluated for the case of turbulent channel flow. A new mixed polynomial-spectral interpolation, best suited to the spectral method used in the hydrodynamic simulation, is also implemented and tested. Estimates of interpolation errors at different distances from the channel walls are given. It is demonstrated that, with adequate spatial resolution in the flow simulation, accurate estimates of Lagrangian statistics in channel flow can be extracted at only a fraction of the cost expended on the hydrodynamic simulation of the flow. Appropriate simulation procedures are also discussed and some single particle statistics are reported. However, a major contribution of the paper is that it offers the opportunity of going beyond the calculation of particle trajectories from discrete data. Because a DNS is used, information about the fluid field seen by the particle is obtained which has not been available from laboratory experiments. This greatly expands theoretical opportunities in using Lagrangian methods to analyze turbulent fields.

2. DIRECT NUMERICAL SIMULATION OF FULLY DEVELOPED TURBULENT CHANNEL FLOW

In order to calculate individual realizations of fluid particle paths, detailed instantaneous information about the flow is required. With a DNS, the evolution of the fluctuating Eulerian velocity field at a large number of spatial locations is determined by solving the full three-dimensional time-dependent Navier–Stokes equations. The DNS of Lyons [6] for a turbulent channel flow used about 1×10^6 ($128\times65\times128$ in x, y, z) mesh points. His code is employed in this work to supply instantaneous Eulerian velocity values. The description of the algorithm offered below is brief and pays attention only to aspects relevant to this work; a more detailed exposition can be found in a thesis by Lyons [5] and in a recent paper by Lyons, McLaughlin, and Hanratty [6].

A statistically stationary and fully developed turbulent flow in a two-dimensional channel is considered. The flow geometry and the coordinate system used in the simulation are shown in Fig. 1. An incompressible, Newtonian fluid is confined between two smooth parallel plates separated by a distance $2H$. The plates are assumed infinitely long and wide and they are at rest with respect to the coordinate system. The flow is driven by a constant mean pressure gradient. The x-axis is parallel to the mean flow, the y-axis is perpendicular to the channel walls, and the z-axis points in the spanwise direction. All average hydrodynamic quantities are independent of the z coordinate because of the unboundedness of the flow in the z-direction. They are also independent of x, because there is no streamwise acceleration. The profile of the average shear stress is linear with distance from the wall and symmetric about the centerline. The computed variation of average hydrodynamic quantities with distance from the wall can be found in [5, 6].

The velocity field is subject to periodic boundary conditions in the streamwise and spanwise directions with periodicity lengths λ_x and λ_z, respectively, which determine also the size of the computational domain in these directions. No-slip conditions are applied at the rigid channel walls. The Navier–Stokes equations are integrated in time using the pseudospectral fractional step method originally developed by Orszag and Kells [11] and the added correction suggested by Marcus [7] to ensure that the proper boundary condition on the pressure field exists at the channel walls. The pseudospectral methods used to solve the Navier–Stokes equations are described in detail by McLaughlin [8]. All variables and most of the results are made dimensionless with wall parameters, the kinematic viscosity, ν, and the friction velocity, $u^*=\sqrt{|\tau_w|/\rho}$, where τ_w is the wall shear stress. Characteristic wall length, time, and pressure scales are then constructed as $L^*=\nu/u^*$, $T^*=\nu/u^{*2}$, $P^*=\rho u^{*2}$.

The velocity field is represented as a truncated series of the form

$$\mathbf{U}(x,y,z,t)=\sum_{l=-N_x/2}^{N_x/2-1}\sum_{m=-N_z/2}^{N_z/2-1}\sum_{n=0}^{N_y}\tilde{\mathbf{U}}(l,n,m,t)\times e^{2\pi i((lx/\lambda_x)+(mz/\lambda_z))}T_n\left(\frac{y}{H}\right), \tag{1}$$

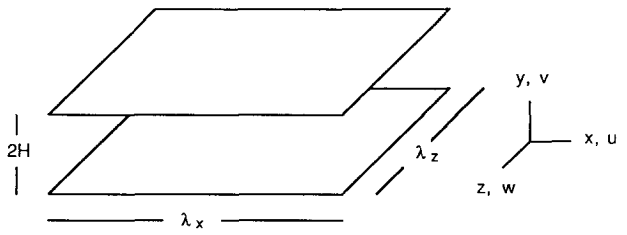


FIG. 1. Flow geometry and coordinate system used in the simulation.

where N_x, N_y+1, N_z are the number of grid points in the x, y, z directions, respectively. The nth order Chebyshev polynomial is defined by

$$T_n\left(\frac{y}{H}\right)=\cos(n\theta), \tag{2}$$

where

$$\theta=\cos^{-1}\left(\frac{y}{H}\right). \tag{3}$$

The use of Fourier series in the spanwise and streamwise directions satisfies the periodicity requirements in these directions automatically. The choice of a Chebyshev expansion in the direction normal to the channel walls avoids the Gibbs phenomenon and naturally increases the spatial resolution of the computation in the high shear region close to the walls where steep gradients are expected. The convective (or nonlinear) terms of the Navier–Stokes equations are evaluated in physical space. The aliasing errors thus incurred in the x- and z-directions are removed by truncation according to the "two-thirds" rule (Orszag [10]). Due to the explicit treatment of the nonlinear term, the Courant constraint on the largest time step size permissible, Δt, must be observed at all grid points in order to ensure numerical stability.

The main input parameters for the flow simulation are the computational box dimensions, the number of grid points in each direction, and the size of the time step. An initial velocity field is also required to start the computations. Results from three simulations, using boxes of different sizes and different grids, will be presented in this work. Some preliminary exploratory results were obtained in a small box of $\lambda_x=630$, $2H=250$, and $\lambda_z=630$, overlayed by a coarse grid consisting of $16\times33\times64$ or $16\times65\times64$ (in x, y, z directions, respectively) grid points. These grids (or simulations) are referred to as grid (or simulation) A and B, respectively. More accurate results were obtained by using a much finer grid (grid C) of $128\times65\times128$ mesh points and a larger computational box of $1900\times300\times950$. The size of the time step was 0.20 for the coarse grid runs and 0.25 for the high resolution runs. Lyons [5, 6] determined that these time step values are small enough for an accurate simulation of the Eulerian flow. The Reynolds number, based on the bulk velocity and twice the channel height, was 9048 for simulation C. The channel code has been validated by comparisons with the experimental data of Niederschulte [9] (Lyons [5, 6]). Statistical and structural information extracted from the computed flow fields are in good agreement with existing experimental results, even for the low resolution simulation A (Lyons [5, 6]).

3. FLUID PARTICLE TRACKING ALGORITHM

(a) *Equation of Motion*

The method for calculating the trajectory of a fluid particle is based on a numerical integration of the equation of particle motion. The position that the particle is initially assigned provides the initial condition for the integration. A record of the random particle velocities and positions at selected sampling times is stored for statistical post-processing. Let $\mathbf{X}(\mathbf{x}_0, t)$ and $\mathbf{V}(\mathbf{x}_0, t)$ denote the position and velocity at time t of the fluid particle originating at $\mathbf{x}_0$ at time $t=0$. The equation of motion of the particle is

$$\frac{\partial\mathbf{X}(\mathbf{x}_0, t)}{\partial t}=\mathbf{V}(\mathbf{x}_0, t), \tag{4}$$

subject to the initial condition at

$$t=0\colon \mathbf{X}(\mathbf{x}_0, t)=\mathbf{x}_0. \tag{5}$$

The Lagrangian particle velocity $\mathbf{V}(\mathbf{x}_0, t)$ is related to the Eulerian velocity $\mathbf{U}$ by

$$\mathbf{V}(\mathbf{x}_0, t)=\mathbf{U}[\mathbf{X}(\mathbf{x}_0, t), t]. \tag{6}$$

According to Eq. (6) the instantaneous particle velocity is the same as the fluid velocity at the instantaneous particle position. From the DNS, the Eulerian velocity is available at each time step on a three-dimensional grid. Since the instantaneous position of a particle does not, in general, coincide with a grid point, the particle velocity has to be evaluated by a three-dimensional interpolation of the Eulerian velocity grid point data. Accurate interpolation is required, since the Eulerian velocity fields vary sharply in space, and numerical errors in the calculated particle trajectory can grow rapidly with time. The continuous spectral series representation of the velocity field (1) does provide a convenient and accurate (up to the accuracy available from the flow simulation) way to calculate the velocity at any location within the flow domain. Unfortunately, the computational cost of a direct summation of the series in (1) is unacceptably high and makes this technique prohibitively expensive, except for the case of a small number of particles.

(b) *Evaluation of Fluid Particle Velocities*

Interpolation methods are usually based on the construction of an approximating function that reproduces the available discrete set of data exactly. Then approximate interpolated values can be obtained for arbitrary intermediate values of the independent variables. When the function underlying the data is of unknown form, polynomials

are usually chosen. The error of the approximation when a polynomial of degree $(n-1)$ is used decreases asymptotically as $O(h^n)$ as the spacing between the grid points, h, goes to zero. The approximation is therefore of order n. YP and BM have recently shown that the accuracy of three-dimensional linear interpolation, which has been the method of choice for some time (Deardorff and Peskin [4], Riley and Patterson [12], Bernard *et al.* [2]) because of its simplicity and its low computational cost, is rather poor when applied to turbulent velocity fields. They investigated various alternative interpolation methods for the case of homogeneous turbulence. It is shown by YP that cubic spline interpolation compares favorably with other interpolation schemes considered in their work. BM recommend the use of sixth-order Lagrange interpolating polynomials. Following the recommendations of YP and BM both cubic spline interpolation (CSI) and Lagrangian interpolation (LGI) were considered in the present study. Linear interpolation (LNI) was also tried for comparison purposes. The implementation of these interpolation techniques is discussed by YP and BM and will not be reproduced here.

In a channel flow, turbulence is affected by the presence of walls which generate high shear and small scales of motion in their vicinity. The spatial resolution of the computation of the Eulerian hydrodynamics fields is enhanced in the region close to the walls of the channel by the natural grid stretching associated with the use of Chebyshev polynomials. However, the resolution deteriorates with distance from the wall and becomes worst in the center of the channel. It is not obvious that the Lagrange or cubic spline interpolation scheme, recommended by BM and YP, would be as satisfactory in the nonhomogeneous normal direction of channel flow as in the homogeneous directions. Consequently, a more accurate interpolation method, based on the direct summation of the Chebyshev series used in the representation (1) of the Eulerian velocity fields, was also explored for interpolation in the normal direction.

If the summations over the indices l and m are carried out in Eq. (1), the velocity at any point (x, z) on the plane defined by a given value of y can be expressed as

$$\mathbf{U}(x, y, z, t) = \sum_{n=0}^{N_y} \tilde{\mathbf{U}}^*(x, n, z, t)\, T_n\left(\frac{y}{H}\right). \tag{7}$$

The Chebyshev coefficients in Eq. (7) become available during the course of the flow simulation. First the summation in (7) is carried out for a y value corresponding to the normal position of each particle ($y = X_2$) and (x, z) values corresponding to the 36 grid points surrounding the particle. Then a two-dimensional interpolation is performed over the plane $y = X_2$, using Lagrange polynomials of degree five. This mixed Lagrangian–Chebyshev interpolation scheme is denoted as LGCH.

Other hybrid spectral–polynomial schemes can be obtained by using other types of polynomials (in place of the Lagrange polynomials) in order to perform the two-dimensional interpolation in the plane $y = X_2$. A mixed Hermite–Chebyshev scheme, which is denoted as HMCH, was also tried.

(b) *Time-Stepping*

The work of YP has demonstrated that an explicit second-order Runge–Kutta time advancement scheme is sufficient for the integration of the equation of particle motion (2). In any case the time-stepping error can be easily controlled by keeping the tracking time step Δt_{tr} small. In this study an Adams–Bashforth scheme (also explicit and second-order accurate) is applied:

$$X_i(\mathbf{x}_0, t_{n+1}) = X_i(\mathbf{x}_0, t_n) + [\tfrac{3}{2} V_i(\mathbf{x}_0, t_n) - \tfrac{1}{2} V_i(\mathbf{x}_0, t_{n-1})] \cdot \Delta t_{\text{tr}}. \tag{8}$$

A well-known difficulty with multistep schemes like the Adams–Bashforth method is that they are not self-starting since, in order to extrapolate to the next point, they require information from previous points (e.g., from time step $(n-1)$ in the above formula (9)). A second-order accurate Runge–Kutta method (known as the improved Euler's method or Heun's method (Carnahan *et al.* [3]) is employed to start the integration. According to this algorithm a first-order Euler's method is employed twice in sequence:

$$X_i^*(\mathbf{x}_0, t_{n+1}) = X_i(\mathbf{x}_0, t_n) + V_i(\mathbf{x}_0, t_n) \cdot \Delta t_{\text{tr}} \tag{9}$$

$$X_i(\mathbf{x}_0, t_{n+1}) = X_i(\mathbf{x}_0, t_n) + \tfrac{1}{2}[V_i(\mathbf{x}_0, t_n) + V_i^*(\mathbf{x}_0, t_{n+1})] \cdot \Delta t_{\text{tr}}. \tag{10}$$

This scheme may be viewed as the simplest of the so-called predictor–corrector methods. Starting at the current time instant t_n, the predictor step (i.e., Eq. (9)) yields a preliminary estimate X_i^* of the ith component of the particle position vector at the next sampling time instant t_{n+1}. For this estimation, the current value of the particle velocity is used. This value is given by the Eulerian velocity at the current time and current particle position:

$$V_i(\mathbf{x}_0, t_n) = U_i[\mathbf{X}(\mathbf{x}_0, t_n), t_n]. \tag{11}$$

In the corrector step an improved estimate $X_i(\mathbf{x}_0, t_{n+1})$ is obtained from Eq. (10); the particle velocity used in this estimation is the weighted average of approximations at the ends of the time interval $[t_n, t_{n+1}]$, where

$$V_i^*(\mathbf{x}_0, t_{n+1}) = U_i[\mathbf{X}^*(\mathbf{x}_0, t_{n+1}), t_{n+1}]. \tag{12}$$

The Runge–Kutta time stepping scheme (9), (10) requires two interpolated velocity values for each particle and each velocity component at each time step. The Adams–Bashforth scheme requires only one, so the time-stepping is continued with (8).

4. TESTS IN PRESCRIBED VELOCITY FIELDS

(a) *Estimation of Velocity Interpolation Errors*

The interpolation error in obtaining particle velocities is, by far, the most serious of all the errors incurred in calculating particle paths and depends on the nature of the velocity field being interpolated, the spatial resolution, and the accuracy of the interpolation scheme. The performance of the interpolation schemes of interest is assessed by comparing the error incurred by each scheme in interpolating a prescribed velocity field which bears some qualitative resemblance to a typical turbulent field. With a prescribed field the exact value of the velocity everywhere in the computational domain is known analytically and the interpolation error averaged over a network of prescribed locations within the domain can be easily calculated. Following BM, a single sinusoidal mode of the following form is used as a sample velocity field:

$$U(x, y, z) = \sin(k_1 \cdot x + k_2 \cdot y + k_3 \cdot z). \tag{13}$$

As a simplification, only modes satisfying the condition $k_1 = k_2 = k_3 = k$ are examined. The computational domain for the test was taken to be a cubic box of side L overlayed with a grid of $N+1$ points in each direction that has a uniform spacing $h = L/N$.

The accuracy of the CSI and LGI schemes for individual Fourier modes of different wavenumbers is compared. The wavenumber $\mathbf{k} = (k_1, k_2, k_3) = k, k, k)$ takes values which are integer multiples of a fundamental wavenumber (a, a, a), with $a = 2\pi/L$:

$$k = n(2\pi/L), \tag{14}$$

where $n = -N/2, ..., 0, ..., (N/2-1)$. For $n = 0$ (i.e., a uniform velocity field, $U = 0$) all interpolation schemes (linear interpolation included) become exact. As BM observe, low wavenumber components (i.e., large scales) are interpolated more accurately than high wavenumbers. The accuracy with which a real turbulent field is interpolated depends on the energy distribution among the several wavenumber components of the velocity. It should be recalled, at this point, that because of the dealiasing procedure in the homogeneous directions of the flow, only modes up to $2/3(N/2+1)$, where N is the number of grid points in the corresponding direction, are retained in the flow simulation.

The velocity field at the grid points is computed analytically from the prescribed expression (13). A large number of fluid particles, N_{pr}, are allocated in the computational domain and exact values of the velocity at the particle locations, $V_{prx,i}$, are also computed analytically from (13). Approximate velocity values at the particle locations, $V_{pr,i}$, are computed with the spline or Lagrangian interpolation of the velocity at the grid points. These are compared to the exact values in order to evaluate absolute local interpolation errors:

$$e_i = |V_{pr,i} - V_{prx,i}|, \qquad i = 1, ..., N_{pr}. \tag{15}$$

As a measure of the overall interpolation error the root mean square value of e_i over all particles, e_{rms}, is computed. (This measure is similar to the one proposed by BM, except that only wavenumbers with equal components are taken into account. A numerical evaluation of this measure of error presents no difficulties even for three-dimensional fields. This was not the case with the analytical evaluation attempted by BM.)

In Fig. 2 the interpolation error, e_{rms}, is plotted versus the resolution, R, for several interpolation schemes, where

$$R = \frac{[\text{scale of motion}]}{[\text{grid spacing}]} = \frac{\lambda}{h} = \frac{2\pi}{|\mathbf{k}|}\frac{N}{L}, \tag{16}$$

and $|\mathbf{k}| = \sqrt{k_1^2 + k_2^2 + k_3^2}$ is the magnitude of the wavevector tested. Values of $N = 64$, $L = 2\pi$ and $N_{pr} = 500$ were used for the calculations. For high values of R the interpolation errors decrease at rates consistent with the orders of accuracy of each scheme. Linear interpolation is clearly the least accurate of the methods for all values of R. Cubic spline interpolation is superior to Lagrangian interpolation

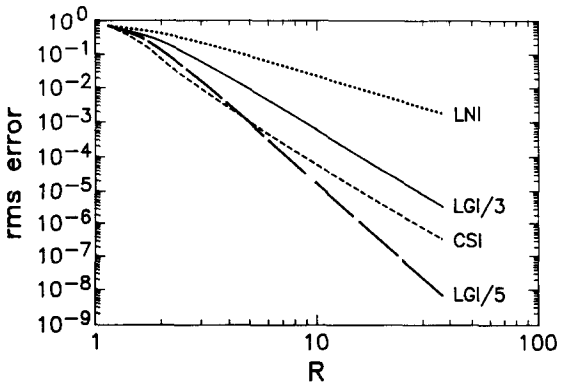


FIG. 2. Interpolation error versus resolution for interpolation of sinusoidal velocity fields using different interpolation schemes: linear (LNI), Lagrangian interpolation of degree 3 (LGI/3), Lagrangian interpolation of degree 5 (LGI/5) and cubic spline interpolation (CSI).

of the same order. For low values of R the CSI scheme is slightly more accurate than LGI of degree five [LGI/5). However, since LGI/5 is more accurate than CSI at large R, it is adopted in this work.

(b) *Time-Stepping Errors*

The time-stepping error due to the finite size of the time step Δt_{tr} is generally much less than the interpolation error, especially for coarser grids and less accurate interpolation schemes. The order of accuracy of the scheme by which the integration of the equation of particle motion is accomplished and the temporal resolution, Δt_{tr}, determine the magnitude of the error incurred at each time step. This error is accumulated over time and the cumulative time-stepping error depends also on the duration of tracking (T_{tr}).

A test of the particle tracking algorithm can be performed with a prescribed turbulence-like flow for which particle paths are known analytically. Such a simple test case is provided by a velocity field of the form:

$$U(x, y, z; t) = U_0 \tag{17a}$$

$$v(x, y, z; t) = 0 \tag{17b}$$

$$w(x, y, z; t) = A \cdot \sin(k_l \cdot x). \tag{17c}$$

This velocity field is periodic, frozen in time, and independent of y. Particles in this flow move sinusoidally in the horizontal plane x-z, and are uniformly translated in the x-direction. The trajectory of a particle can be derived by an analytic integration of the equation of particle motion ($dX/dt = U_0$, $dY/dt = 0$, $dZ/dt = A \cdot \sin(k_l \cdot x)$). The solution for the coordinates of a particle at time t, which was placed initially at point (X_0, Y_0, Z_0), is given as

$$X(t) = X_0 + U_0 \cdot t \tag{18a}$$

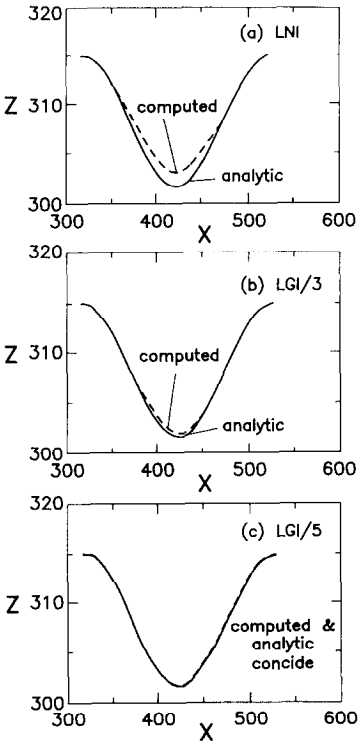
$$Y(t) = Y_0 \tag{18b}$$

$$Z(t) = Z_0 + \frac{A}{k_1 \cdot U_0}[\cos(k_1 \cdot X_0) - \cos(k_1 \cdot X_0 + k_1 \cdot U_0 \cdot t)]. \tag{18c}$$

Errors in calculated particle displacements can be quantified by comparing with (18).

The results of such comparisons for different wave numbers k_l, using Lagrange interpolating polynomials of different degrees and two different time differencing schemes and step sizes are shown in Figs. 3, 4, and 5. The parameters of the prescribed flow field are chosen as $U_0 = 5$ and $A = 1$ and the particle is initially placed at point (X_0, Y_0, Z_0) = (315, 0, 315). The computational box dimensions ($630 \times 250 \times 630$) and the grid ($16 \times 33 \times 64$) are the same as those for simulation A discussed in Section 2. Two values of the wavenumber $k_l = n \cdot a$ are examined: $n = 3$ and $n = 6$, where $a = 2\pi/L = 6.28/630 = 0.0099733$ is the fundamental wavenumber component in the x-direction. Had a real turbulent field existed in the same domain, no spectral modes (in the x-direction) higher than $n = 6$ would have survived the dealiasing procedure used in the flow simulation. The wavenumber corresponding to $n = 3$ is expected to carry a significant amount of energy and it must be interpolated accurately. Some inaccuracy can be tolerated in the interpolation of the smaller scales of motion (such as those represented by $n = 6$), since they normally possess insignificant amounts of energy and make a negligible contribution (confined to small diffusion times) to the turbulent diffusion process which is dominated by the large scales of motion.

In Fig. 3 the trajectory of a particle in a velocity field corresponding to $n = 3$ is computed numerically and compared to the exact trajectory predicted analytically. Different interpolation schemes (linear, Lagrangian of order four and six) are applied and a first-order accurate Euler scheme is used for time differencing. The tracking time ($T_{tr} = 42$ time units) is long enough for the particle to cover a distance equal to one wavelength ($\lambda = 2\pi/k = 210$ length units). A tracking time step ($\Delta t_{tr} = 0.5$) much smaller than this natural time scale is chosen in order to resolve the velocity variations. It is clear from Fig. 3(a) that the linear



the form (14), (15), (16)) for different interpolation schemes: (a) linear interpolation; (b) Lagrangian interpolation of degree 3; and (c) Lagrangian interpolation of degree 5 ($n = 3$, $\Delta t_{tr} = 0.5$, Euler time-stepping scheme).

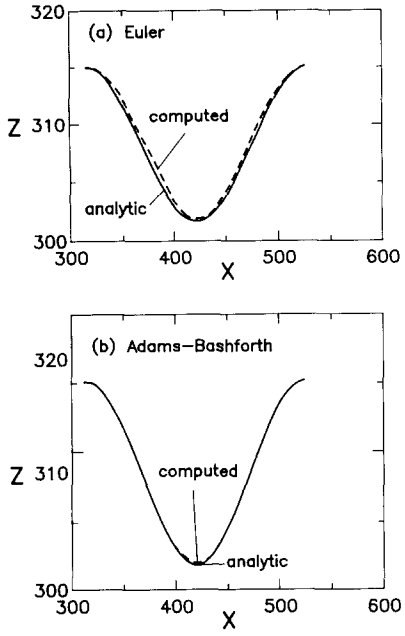


FIG. 4. Particle displacement errors in a prescribed velocity field (of the form (14), (15), (16)) using different time-stepping schemes: (a) Euler and (b) Adams–Bashforth ($n=3$, Lagrange interpolation of order 6, $\Delta t_{tr}=1$).

interpolation scheme fails to follow the sinusoidal variation of the particle trajectory while Lagrange polynomials of order six reproduce the analytic solution almost exactly (Fig. 3c.) Lagrange polynomials of order four achieve quite satisfactory levels of accuracy (Fig. 3b).

The effect of time-stepping errors is examined in Fig. 4. Doubling the size of the time step to $\Delta t_{tr}=1$ for the case where interpolation errors are negligible (i.e., sixth-order interpolation) has only an insignificant (although detectable) effect on the accuracy of the computed trajectory Fig. 4a as compared to Fig. 3(c). Increasing the accuracy of the time-differencing scheme to second-order eliminates any time-differencing errors even for the lower temporal resolution case ($\Delta t_{tr}=1$) as it is evident from Fig. 4b.

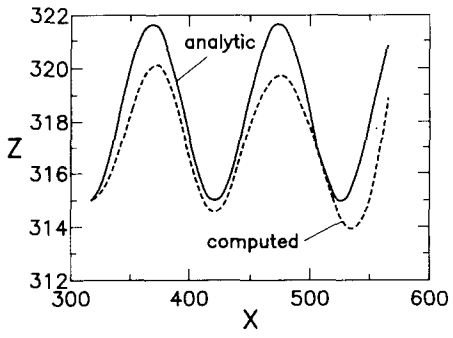


FIG. 5. Effect of low resolution on particle displacement errors in a prescribed velocity field (of the form (14), (15), (16), $n=6$, Lagrange interpolation of order 6, $\Delta t_{tr}=0.5$, Euler time-stepping scheme).

The effect of a deterioration in the spatial resolution is much more serious than time-stepping errors. Even a sixth-order interpolation scheme achieves only marginal accuracy in computing the trajectory of a particle in a rapidly varying small scale velocity field corresponding to $n=6$ (Fig. 5). Fortunately, such small scales are expected to play only a secondary role in the dispersion process.

5. TEST IN SIMULATED TURBULENT CHANNEL FLOW

(a) *Testing Procedure*

This section discusses results on the diffusion of fluid particles from a point source located in a turbulent channel flow. The emphasis is to test the sufficiency of the techniques developed, to produce Lagrangian flow information and to establish proper parameter values and simulation procedures. The acquisition of more accurate estimates of the Lagrangian characteristics of the flow (by increasing the spatial resolution of the simulation as well as the number of independent particle trajectories traced) will be the focus of a future paper.

Most of these exploratory results were obtained with grid A that has $16\times33\times64=33792$ grid points. The dimensions of the computational box ($\lambda_x=630$, $2H=250$, $\lambda_z=630$) are not adequate to accommodate the larger scales of motion as is apparent from Eulerian correlation and structure data collected by Lyons [5, 6]. Nonetheless, and despite the inadequate resolution, Eulerian statistics (at least of low order) showed satisfactory agreement with experimental data (Lyons [5, 6]). Therefore, the coarse grid simulation offers an economical way to test the tracking algorithm.

A number of fluid particles, N_{pr}, were tagged simultaneously at time $t=t_0$ and assigned initial positions that spread them over a horizontal xz-plane at a desired distance $y+$ from the bottom wall of the channel. The statistical homogeneity of the Eulerian fields in the streamwise and spanwise directions removes any statistical dependence on the x and z coordinates of the source location. Therefore, ensemble averaging over particles released at different x and z locations is permitted. Similarly, the stationarity of the Eulerian fields renders the time of release t_0 irrelevant, thus justifying ensemble averaging over particles released at different times, too. The number of particles allocated was restricted by the number of grid points employed over a horizontal plane to resolve the Eulerian fields. The particles are released at positions on the order of one grid spacing apart. Releasing more particles would not improve the statistical sample (except perhaps at very long diffusion times) because the motion of the particles would be initially strongly correlated. Several tracking experiments were

TABLE I

RMS Velocity Interpolation Errors on the Coarse Grid A for LGCH Interpolation

y^+	% error in V_x	% error in V_y	% error in V_z
1	1.07	12.06	5.75
2	1.04	9.26	5.68
5	1.11	8.19	5.45
15	0.99	8.27	6.17
125	0.31	8.14	7.49

performed in each turbulence simulation for computational economy.

The paths of the particles were traced (using the algorithm of the previous section) and time series of particle velocities and positions were generated from which Lagrangian statistics were estimated. These statistical quantities are expected to be functions of only the normal coordinate of the location of the source and the diffusion or elapsed time after release. The particle displacement is a nonstationary random process, since the spread of the particles in the homogeneous directions of the flow continues to grow, without limit. (The normal component of the dispersion, however, is limited eventually by the presence of the walls and has to approach a constant value corresponding to a uniform distribution across the channel cross section.) The particle velocity will also be nonstationary initially, due to the inhomogeneities of the flow in the normal direction. At long times it will become stationary as a result of the boundedness of the flow in the inhomogeneous direction.

(b) *Effect of the Interpolation Accuracy*

Velocity interpolation errors were assessed by comparing interpolated particle velocities (at the time of their release) calculated using the LGCH scheme of Section 3 to exact velocities calculated from (1). A large number of particles (1024) were randomly released over a horizontal plane at a given distance from the bottom wall and the root-mean-square difference between interpolated and exact velocities (normalized by the rms turbulence velocities) was computed as a measure of the interpolation errors incurred over one time step. The largest errors are expected with the low resolution simulation A (Table I). Table I shows that the largest errors appear in the normal velocity; these increase close to the wall, where more kinetic energy resides with the small scales of motion. Table II, which presents rms interpolation errors on the fine grid C, shows that the accuracy of the interpolation is improved significantly by improving the resolution of the simulation.

TABLE II

RMS Velocity Interpolation Errors on the Fine Grid C for LGCH Interpolation

y^+	% error in V_x	% error in V_y	% error in V_z
0.1	0.30	3.75	1.30
0.5	0.30	3.59	1.24
1.0	0.29	3.32	1.16
2.0	0.29	3.02	1.03
10.0	0.27	2.42	1.15
15.0	0.27	2.46	1.25
150.0	0.04	1.35	1.50

Figure 6 shows the trajectories of fluid particles originating in the center of the channel which were calculated by using the mixed Lagrangian–Chebyshev (LGCH) and fully Lagrangian interpolation of degree five (LGI/5). The two trajectories deviate only negligibly from each other even after a long integration time of 150 units made dimensionless with wall parameters. The difference in the time step size used in the two runs had no effect. (The effect of the time step size is examined shortly.) Such tests show that the use of the Chebyshev series in the normal direction (LGCH) offers only a small improvement over the fully Lagrangian interpolation (LGI/5) in the center of the channel. However, in the wall region, Lagrange polynomials of degree five cannot be used in the normal direction for particles closer than two grid points away from the wall.

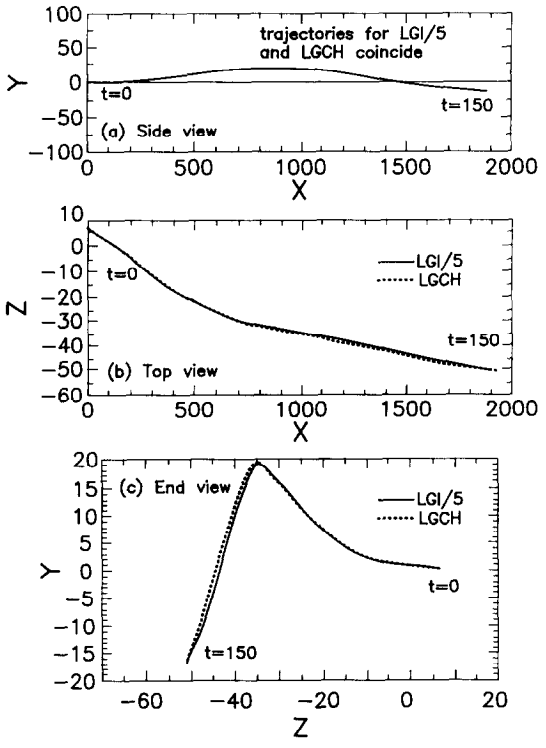


FIG. 6. Trajectories of the same particle in the center of turbulent channel flow traced for 150 wall time units using different interpolation schemes: LGI/5 ($\Delta t_{tr} = 0.4$) and LGCH ($\Delta t_{tr} = 0.6$).

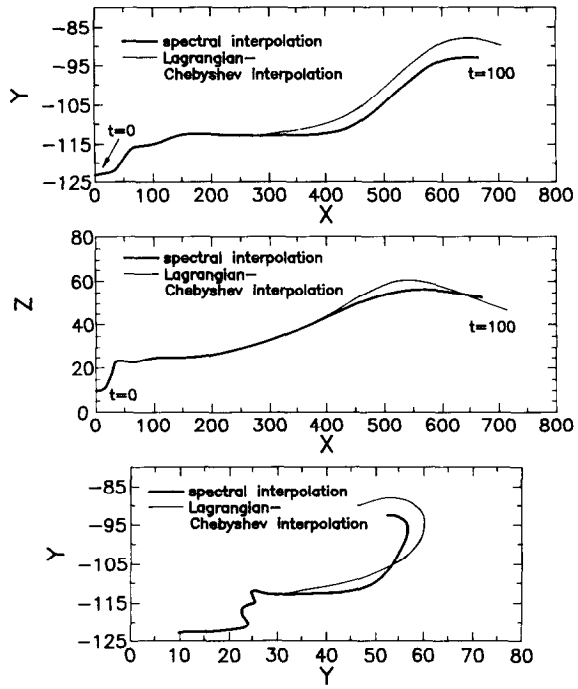


FIG. 7. Trajectories of the same particle released at $y^+ = 2$ in the coarse grid simulation A and traced using spectral and LGCH interpolation.

Errors in particle displacements in the wall region were quantified for the LGCH scheme by comparing computed trajectories of fluid particles with true trajectories traced using the exact spectral interpolation. A large number of particles (1024) were released at $y+ = 2$ in the coarse grid channel simulation A and their trajectories were traced for 100 wall time units. Figure 7 shows a comparison of the computed and exact trajectory of one of these particles. The trajectories are identical initially and they remain reasonably close even after a total time of 100 wall units. The rms displacement errors in the normal and spanwise directions averaged over all particles and normalized by the rms displacement at the same time are shown in Fig. 8. Again the largest errors are found in the normal direction.

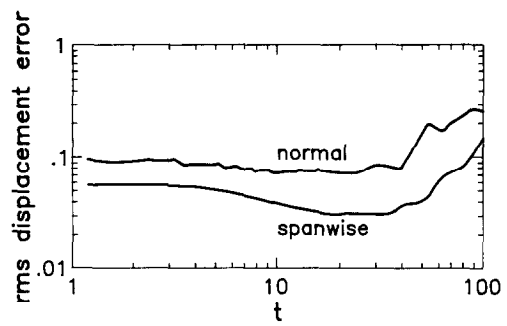


FIG. 8. Root-mean-square displacement errors normalized with the root-mean-square dispersion at the same time, averaged over 1024 particles released at $y^+ = 2$ in the coarse grid simulation A.

The errors do not show a significant accumulation over time up to about 50 wall time units. However, after the deviations between computed and true trajectories become comparable to the smallest turbulence scales of the simulation, displacement errors grow rapidly. It should be mentioned at this point that the statistical error from averaging over an ensemble of only 1024 particles also becomes significant after a time of about 80 wall units.

Lagrangian statistical quantities averaged over many particles are less sensitive to interpolation errors than individual particle trajectories. In Fig. 9 the computed mean square dispersions in the normal and spanwise directions are compared to the true mean square dispersions of particles traced with the exact spectral interpolation. The dispersion tensor is comprised of the second-order moments of the displacement fluctuation vector,

$$D_{ij}(t-t_0) = \langle Y_i'(\mathbf{x}_0, t) \cdot Y_j'(\mathbf{x}_0, t) \rangle, \tag{19}$$

where $\langle\ \rangle$ indicates averaging over all particles. The displacement vector, defined as

$$\mathbf{Y}(t-t_0) = \mathbf{X}(\mathbf{x}_0, t) - \mathbf{x}_0 = \int^t \mathbf{V}(\mathbf{x}_0, t)\, dt, \tag{20}$$

is used in place of the position vector $\mathbf{X}(\mathbf{x}_0, t)$ in order to describe the motion of a single fluid particle. The displacement fluctuation vector is obtained as $\mathbf{Y}'(\mathbf{x}_0, t) = \mathbf{Y}(\mathbf{x}_0, t) - \langle \mathbf{Y}(\mathbf{x}_0, t) \rangle$. It is noted that since the largest interpolation errors are incurred in the wall region of the coarse grid simulation, Fig. 7, 8, and 9 represent a worst case scenario and are indicative of an upper limit on the particle displacement errors.

Trilinear interpolation was found to be significantly less accurate than Lagrangian interpolation of order five. Lagrangian velocity autocorrelations, computed with either LGI/5 or LGCH and with a trilinear interpolation (LNI), are compared in Fig. 10. The Lagrangian autocorrelation is defined as

$$R_{ij}^L(t_0, s) = \frac{\langle V_i'(\mathbf{x}_0, t_0) \cdot V_j'(\mathbf{x}_0, t_0 + s) \rangle}{V_{i,\mathrm{rms}}(t_0) \cdot V_{j,\mathrm{rms}}(t_0 + s)}. \tag{21}$$

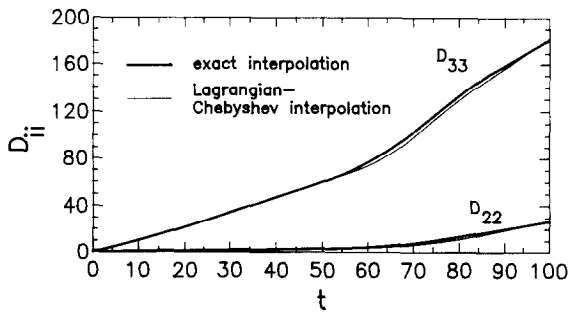


FIG. 9. Mean square dispersion of 1024 particles released at $y^+ = 2$ in the coarse grid simulation A and traced using spectral or LGCH interpolation.

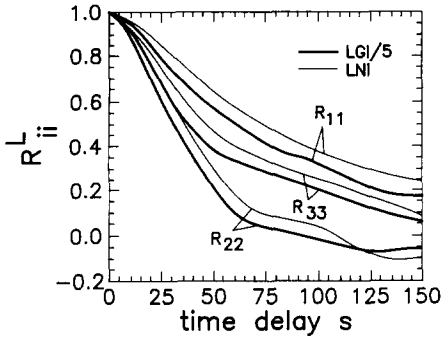


FIG. 10. Lagrangian velocity autocorrelations in the center of a channel using (a) Linear interpolation (LNI) and (b) Lagrangian interpolation of degree 5 (LGI/5).

The brackets $\langle\ \rangle$ indicate averaging over the ensemble of particles released in the center of the coarse grid channel flow simulation A. It should be noted that definition (21) accounts for the nonstationarity of the particle velocity. Figure 10 suggests that three-dimensional linear interpolation overestimates the velocity autocorrelation, because the LNI scheme misses the decorrelating effect of the small scales of motion.

The HMCH scheme makes use of cubic polynomials in the x and z directions. Asymptotically, the error of the approximation decreases as the fourth power of the grid spacings. The scheme involves performing the summation in (7) for the y-coordinate of each particle and (x, z) values corresponding to the four grid points surrounding the particle. Unlike the LGCH method, the HMCH method makes use of the values of the first and second spatial derivatives of the velocity at the above four points (see BM). Specifically, for each component of velocity, the values of the first derivatives with respect to x and z, as well as the second derivative with respect to x and z, are needed at the four points. Unfortunately, this means that the storage requirements for the HMCH scheme are significantly larger than for the other methods, since three-dimensional arrays are needed to store each of the spatial derivative fields. This large storage requirement is the major drawback of the HMCH scheme. On the other hand, BM have provided evidence that partial Hermite interpolation can be more accurate than sixth-order Lagrange interpolation for marginally resolved flows even though partial Hermite interpolation is a lower order scheme. For that reason, HMCH interpolation is compared with LGCH interpolation.

TABLE III

RMS Velocity Interpolation Errors on the Small Box, High Resolution Grid B for LGCH Interpolation

y^+	% error in V_x	% error in V_y	% error in V_z
0.1	0.80	8.39	4.04
0.5	0.73	8.38	3.85
1.0	0.70	7.65	3.75
2.0	0.65	8.39	3.54
10.0	0.75	5.65	4.03
15.0	0.69	6.16	5.14
50.0	0.48	8.06	6.42
150.0	0.23	7.66	9.28

TABLE IV

RMS Velocity Interpolation Errors on the Small Box, High Resolution Grid B for HMCH Interpolation

y^+	% error in V_x	% error in V_y	% error in V_z
0.1	0.17	1.75	0.86
0.5	0.16	1.76	0.83
1	0.15	1.62	0.81
2	0.15	1.73	0.76
10	0.16	1.18	0.82
15	0.15	1.27	1.04
50	0.10	1.59	1.25
125	0.05	1.53	1.80

The comparison between LGCH and HMCH interpolation is made on grid B. It should be recalled that grid A and grid B differ only in that grid B has 65 grid points in the y-direction while grid A has 33 grid points in the y-direction. Table III presents the rms errors for LGCH interpolation on grid B. By comparing the results in Tables I and III, it can be seen that, although the errors on grid B are consistently smaller than on grid A, the effect of increasing the number of grid points in the y-direction on the accuracy of the results is small.

Table IV presents the rms errors for HMCH interpolation on grid B. It can be seen that the errors are not only smaller than the errors with LGCH interpolation on grids A and B but even on grid C, which is not fine enough to resolve the smallest scales in the flow. These results appear to be consistent with BM's results for homogeneous turbulence. It is conceivable that, in applications requiring high interpolation accuracy on relatively coarse grids, HMCH interpolation may be a viable option. It is interesting to note that the error in the spanwise velocity increases near the middle of the channel with both methods (Tables I to IV) for reasons which are not understood.

(b) *Effect of the Time Step Size*

The effect of the size of the time step in computing Lagrangian statistics in a channel flow was also tested. The work of YP has shown that the Courant numerical stability constraint places a limit on the time step size used in the solution of the Navier–Stokes equation that is more stringent than is required for the integration of the equation

of particle motion. In order to examine the effect of time-stepping errors, the tracking time step was varied as an integer multiple of the time step used in the hydrodynamic simulation. Values of $\Delta t_{tr} = 0.2$, 0.4, and 0.6 were tried. Accurate interpolation (LGCH) was used in order to minimize interpolation errors which otherwise could obscure the smaller time-stepping errors. A sufficient number of particles ($N_{pr} = 2024$) was traced in order to also minimize statistical errors. Lagrangian autocorrelations computed according to (21) were identical for all three time step values up to a total time of 150 wall units. For these values of Δt_{tr}, the typical displacement of a fluid particle in the center of the channel in any direction is smaller than the grid spacing in the same direction. It is concluded that a value of $\Delta t_{tr} = 0.2$ guarantees negligible time-stepping errors.

(c) *Effect of the Ensemble Size*

A final source of error in the computed Lagrangian statistics results from averaging over an insufficient stochastic ensemble. The statistical error can be detected by monitoring statistical convergence and can be removed by increasing the sample size. The longer the particles are followed and allowed to spread, the larger is the number of particles that is required for good statistics. It was determined that averaging over at least about 4000 particles is necessary for meaningful first- and second-order Lagrangian statistics up to a diffusion time of 150 wall units.

It should be noted that for unbiased statistics averaging not only over an ensemble of independent particle trajectories, but also over an ensemble of independent turbulence realizations, is required in order for the effect of any peculiarities of the turbulence at the time of the release of the particles to be smeared out. Therefore, a tracking experiment should be repeated by releasing new sets of particles at later times. Averages should then be obtained for each time of release. The time between releases should be sufficiently long (on the order of one turbulence period $T^+ = 100$ or more) for the flow to evolve sufficiently to prevent any of the structural details of the turbulence at the time of the first release from being preserved. The statistical dependence on the time of release diminishes by extending the horizontal dimensions of the computational box since, according to the ergodic hypothesis, spatial and time averages of a stationary homogeneous turbulence are identical.

(d) *Comparisons with the Theory of Homogeneous Turbulence Diffusion*

Although channel flow is inhomogeneous in the [illegible] direction, Taylor's theory [13] is still approximately applicable for small to moderate diffusion times, when most of the diffusing particles remain in the nearly homogeneous part of the flow in the center of the channel. As the cloud of contaminated particles is convected downstream, it spreads around its center of gravity. The average linear cloud dimension at any time instant t can be estimated from the diagonal components of the dispersion as $[D_{11}(t) \cdot D_{22}(t) \cdot D_{33}(t)]^{(1/3)}$.

In the early stages of the diffusion process the dispersion varies parabolically with time:

$$D_{ii} \approx \overline{u_i^2} t^2, \tag{22}$$

where $\overline{u_i^2}$ is the turbulence intensity of the ith velocity component. In homogeneous turbulence and at sufficiently long times, the dispersion exhibits a linear growth with the time

$$D_{ii}(t - t_0) \approx 2\overline{u_i^2} \cdot \tau_{ii}^L \cdot (t - t_0), \tag{23}$$

where

$$\tau_{ii}^L = \int_0^\infty R_{ii}^L(s)\, ds \tag{24}$$

is a Lagrangian integral time scale. Figure 11 presents the computed mean square particle dispersion on a logarithmic graph. Lines of slopes 1 and 2 are drawn on the graph for easy comparison with Eqs. (22) and (23). It is clear that the behavior observed at small times (up to a value of about 20 wall units) agrees with Eq. (23). The curve for the streamwise dispersion, D_{11}, is located above those for dispersion in the transverse directions, D_{22} and D_{33}, in agreement (both qualitative and quantitative) with the corresponding Eulerian intensity values at the centerline. At longer times the dispersion in the flow direction does not approach a constant asymptote. The dispersion is larger than the predictions of Eq. (23) for a homogeneous field because particles at different y-locations are, on average, experiencing differences in streamwise velocity because of the changes in mean velocity. This enhanced longitudinal

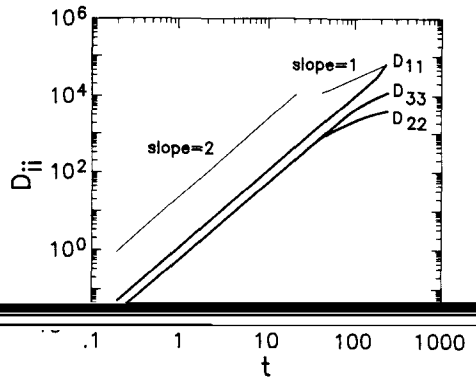


FIG. 11. Asymptotic behavior of the computed mean square particle dispersion in the center of the channel.

dispersion was first described by Taylor in a series of papers [14–16] which considered only the long-time behavior. In the normal direction the flow is bounded and the walls eventually limit the movement of the particles so that the dispersion in this direction has an upper limit determined by H. This is evident in Fig. 11 which shows D_{22} at long times to increase with time at a slower rate than the linear relation predicted by (23). On the other hand, dispersion in the unbounded z-direction, D_{33}, does show a linear increase with time at large times.

5. DISCUSSION

Of the interpolation schemes considered in this paper, HMCH interpolation is the most accurate for turbulent flow simulations with presently obtainable spatial resolution. Furthermore, this conclusion seems likely to remain true since, as computers increase in speed and memory, future researchers will exploit these changes by increasing the flow Reynolds number. For the purpose of tracking point particles, LNI is the only interpolation scheme that introduces serious errors for marginal resolution. However, when tracking solid particles, the drag force acting on the particle depends on the difference between the particle velocity and the undisturbed fluid velocity evaluated at the center of the particle and interpolation errors may be more serious in this case

One of the more important findings of this research is that the asymptotic error estimates are misleading when interpolation schemes are applied to currently feasible channel flow simulations. The probable explanation is that the flows are only marginally resolved so that significant energy resides in the smallest length scales included in the simulation.

When exact interpolation is used to compute the fluid velocity, the cpu time needed for the interpolation can be an order of magnitude more than the amount of cpu time needed for all of the rest of the calculations together. By comparison, when LGCH interpolation was used with 80,645 particles on grid C, the particle tracking calculations required only about 41 % of the total cpu time on a Cray-2 computer; such a calculation would be prohibitively expensive with exact interpolation. The same calculation with HMCH interpolation would require only slightly more CPU time although the memory requirements might prove unacceptable.

When both accuracy and storage are important considerations, LGCH interpolation appears to be a good choice. However, there are certainly other schemes that would perform about as well. For example, if the CSI scheme of YP were modified to include Chebyshev summation for the y-direction, it would probably perform about as well as LGCH for marginally resolved channel flows. Our primary goal has been simply to point out the accuracy that is obtainable with relatively simple, inexpensive interpolation schemes for current channel flow simulations.

ACKNOWLEDGMENTS

This work was supported by the Fluid Dynamics Program of the Office of Naval research (N00014-82K0324) and by the Transport Phenomena Program of the National Science Foundation (NSF CTS 89-19843). J. B. M. acknowledges support from the Department of Energy under Contract DE-FG02-88ER13919. We also acknowledge the support and facilities of the National Center for Supercomputing Applications at the University of Illinois, Urbana. Part of this research was conducted using the Cornell National Supercomputer Facility, a resource of the Cornell Theory Center, which is funded in part by the National Science Foundation, New York State, the IBM corporation, and members of the Center's Corporate Research Institute.

REFERENCES

1. S. Balachandar and M. R. Maxey, *J. Comput. Phys.* **83**, 96 (1989).
2. P. S. Bernard, M. F. Ashmawey, and R. A. Handler, *Phys. Fluids A* **1** (9) 1532 (1989).
3. B. Carnahan, H. A. Luther, and J. O. Wilkes, *Applied Numerical Methods* (Wiley, New York, 1969).
4. J. W. Deardorff and R. L. Peskin, *Phys. Fluids* **13**, 584 (1970).
5. S. L. Lyons, Ph.D. thesis (University of Illinois, Urbana, Illinois, 1989).
6. S. L. Lyons, T. J. Hanratty, and J. B. McLaughlin, *Int. J. Numer. Methods Fluids* **12** (1991).
7. P. S. Marcus, *J. Fluid Mech.* **146**, 45 (1984).
8. J. B. McLaughlin, *Phys. Fluids A* **1** (7), 1211 (1989).
9. M. A. Niederschulte, Ph.D. thesis (University of Illinois, Urbana, Ill., 1989).
10. S. A. Orszag, *J. Atmos. Sci.* **28**, 1074 (1971).
11. S. A. Orszag and L. C. Kells, *J. Fluid Mech.* **96**, 159 (1980).
12. J. J. Riley and G. S. Patterson, *Phys. Fluids* **17**, 292 (1974).
13. G. I. Taylor, *Proc. London Math. Soc.* **20**, 196 (1921).
14. G. I. Taylor, *Proc. R. Soc. London Ser. A* **219**, 186 (1953).
15. G. I. Taylor, *Proc. R. Soc. London Ser. A* **68**, 223 (1954).
16. G. I. Taylor, *Proc. R. Soc. London Ser. A* **77**, 225 (1954).
17. P. K. Yeung and S. B. Pope, *J. Comput. Phys.* **79**, 373 (1988).