

An accurate time advancement algorithm for particle tracking

Pavel P. Popov^{a,*}, Randall McDermott^b, Stephen B. Pope^a

^a Sibley School of Mechanical and Aerospace Engineering, Cornell University, 245 Upson Hall, Ithaca, NY 14853, USA

^b Building and Fire Research Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899-8663, USA

ARTICLE INFO

Article history:

Received 25 January 2008

Received in revised form 12 June 2008

Accepted 19 June 2008

Available online 28 June 2008

Keywords:

Particle tracking

Time advancement

Turbulent reactive flow

Large eddy simulation

Particle method

Filtered density function

Second-order Runge–Kutta

ABSTRACT

We describe a particle position time advancement algorithm that is designed for use with several subgrid velocity reconstruction schemes used in LES/FDF methods, and potentially in other applications. These reconstruction schemes yield a subgrid velocity field with desirable divergence properties, but also with discontinuities across cell faces. Therefore, a conventional time advancement algorithm, such as second-order Runge–Kutta (RK2), does not perform as well as it does with a smooth velocity field. The algorithm that we describe, called Multi-Step RK2 (MRK2), builds upon RK2 by breaking up the time step into two or more substeps whenever a particle crosses one or more velocity discontinuities. When used in conjunction with the parabolic edge reconstruction method, MRK2 performs considerably better than RK2: both the final position of an advected particle, and the final area of a 2D infinitesimal area element are second-order accurate in time (as opposed to first-order accurate for RK2). Furthermore, MRK2 has the theoretical advantage that it better preserves the continuity of the mapping between initial and final particle positions.

© 2008 Elsevier Inc. All rights reserved.

1. Introduction

In this paper, we consider some aspects of particle tracking in hybrid finite-volume/particle PDF methods for turbulent reactive flows. In these methods, and potentially in other CFD applications, we have a large number of particles (on the order of 10^7) whose positions are initially random and evolve by the ODE:

$$\frac{d\mathbf{X}^*}{dt} = \mathbf{U}^* = \mathbf{U}(\mathbf{X}^*(t), t), \quad (1)$$

where $\mathbf{X}^*(t)$ is a particle's position as a function in time, and $\mathbf{U}(\mathbf{X}^*(t), t)$ is the velocity experienced by the particle. Usually, this velocity consists of a deterministic component and a random term which is part of the turbulence model [5]; in the present paper, we consider the deterministic part only. In general, velocity information is available at discrete locations on a finite-volume (FV) grid, and therefore a particle tracking algorithm consists of two parts: a velocity interpolation scheme which interpolates the FV velocity onto the particle locations, and a time advancement scheme which updates the particle locations, using the interpolated velocity.

In recent research on FV/particle PDF methods for reacting flows, accurate particle tracking has been recognized as an important condition for maintaining numerical consistency between the finite-volume and the particle aspects of the solution. Muradoglu et al. [2] define mean particle mass density, $q(\mathbf{x}, t)$, as the expectation of the total mass of particles in an infinitesimal region, normalized by that region's volume:

$$q \equiv \langle m^* \delta(\mathbf{X}^* - \mathbf{x}) \rangle \quad (2)$$

* Corresponding author.

E-mail address: ppp7@cornell.edu (P.P. Popov).

(where m^* denotes a particle's mass), and have shown that an important consistency condition is that the mean particle mass density should remain equal to the FV mean density:

$$q = \langle \rho \rangle, \tag{3}$$

provided that the initial conditions are consistent, $q(\mathbf{x}, 0) = \langle \rho(\mathbf{x}, 0) \rangle$. The meaning of Eq. (3) is that the expected mass of all particles inside a certain region S must be equal to the expected mass of fluid in S as given by the FV density field. In the incompressible case, for example, this means that if particles are initially uniformly distributed in the computational domain, then they should remain so for all time.

In order to satisfy the above consistency condition, Muradoglu et al. [2] employ a position-correction algorithm which introduces a small displacement in the particle positions after each time step, in order to enforce Eq. (3). A similar position-correction algorithm has been employed by Zhang and Haworth [4].

In a different approach, Jenny et al. [3] note that if particles move with a velocity:

$$\mathbf{U}^* = \tilde{\mathbf{U}} + \mathbf{u}^*, \tag{4}$$

where $\tilde{\mathbf{U}}$ is a deterministic component interpolated from the FV velocity, and \mathbf{u}^* is a random component with zero mean, then the following equation holds:

$$\left(\frac{\partial}{\partial t} + \tilde{\mathbf{U}} \cdot \nabla \right) \ln q = -\nabla \cdot \tilde{\mathbf{U}}, \tag{5}$$

which has the same form as the mean continuity equation:

$$\left(\frac{\partial}{\partial t} + \tilde{\mathbf{U}} \cdot \nabla \right) \ln \langle \rho \rangle = -\nabla \cdot \tilde{\mathbf{U}}, \tag{6}$$

where $\tilde{\mathbf{U}}$ is the mean velocity. Therefore, if Eq. (3) is satisfied at $t = 0$, it will also be satisfied implicitly at future time, provided that the velocity interpolation scheme yields accurate values for the reconstructed velocity and its divergence, and provided that an accurate time advancement scheme is used. Addressing the velocity interpolation issue, Jenny et al. [3] introduce a 2D velocity interpolation scheme with desirable divergence properties. McDermott and Pope [1] improve upon this scheme, and extend it to 3D, calling the new scheme the parabolic edge reconstruction method (PERM). It has been shown [1] that PERM performs better than standard multilinear interpolation in satisfying the above-mentioned consistency condition in the particle tracking limit (i.e., when there are no velocity fluctuations).

In the present work, we make a further improvement in particle tracking by using a time-stepping algorithm which has been specifically designed for use in conjunction with PERM. The new algorithm, called Multi-Step Runge–Kutta 2 (MRK2) is quite similar to the standard second-order Runge–Kutta (RK2), but it provides a more accurate treatment of particles which cross a velocity discontinuity. Although MRK2 is motivated by the PERM reconstruction, it can also be used as an alternative to RK2 in all applications in which particles need to be advected through a velocity field with discontinuities at known locations.

To illustrate the benefits of PERM and MRK2, Fig. 1 shows final particle distributions for a simple 2D incompressible test flow with a uniform initial particle distribution. In this case, the final particle distribution should be uniform as well. It can be seen in Fig. 1 that, when RK2 is used as the time advancement scheme, the PERM velocity interpolation yields a more uniform final particle distribution than standard bilinear interpolation. We can also see that the final particle distribution becomes even more uniform when the new time-stepping scheme, MRK2, is used in conjunction with the PERM velocity interpolation.

1.1. Properties of the PERM velocity reconstruction

The reader is referred to [1] for a thorough description of PERM and its properties – here we focus only on its aspects which are relevant to the MRK2 time-stepping scheme.

The PERM velocity reconstruction scheme uses discretized velocity information in the form of face-average velocity components in the direction normal to a given grid cell face (we shall refer to this as the ‘FV velocity’, for the sake of brevity). The PERM reconstructed velocity within a given grid cell depends only on the FV velocity on the faces of that cell and its nearest neighboring cells. Its functional form, in two dimensions (the extension to 3D is rather intuitive) is

$$\begin{aligned} u(q, r) &= (1-r) \left[\bar{u}_S + \left(q - \frac{1}{2} \right) A_{u,S}^{(1)} + \frac{1}{2} \left\{ \left(q - \frac{1}{2} \right)^2 - \frac{1}{4} \right\} A_{u,S}^{(2)} \right] + r \left[\bar{u}_N + \left(q - \frac{1}{2} \right) A_{u,N}^{(1)} + \frac{1}{2} \left\{ \left(q - \frac{1}{2} \right)^2 - \frac{1}{4} \right\} A_{u,N}^{(2)} \right] \\ v(q, r) &= (1-q) \left[\bar{v}_W + \left(r - \frac{1}{2} \right) A_{v,W}^{(1)} + \frac{1}{2} \left\{ \left(r - \frac{1}{2} \right)^2 - \frac{1}{4} \right\} A_{v,W}^{(2)} \right] + q \left[\bar{v}_E + \left(r - \frac{1}{2} \right) A_{v,E}^{(1)} + \frac{1}{2} \left\{ \left(r - \frac{1}{2} \right)^2 - \frac{1}{4} \right\} A_{v,E}^{(2)} \right] \end{aligned}$$

where u, v respectively the horizontal and vertical velocity components, N, S, E, W is the standard compass convention for 2D grids, q, r are respectively the horizontal and vertical local coordinates (e.g., $q = 0$ on the West face and $q = 1$ on the East

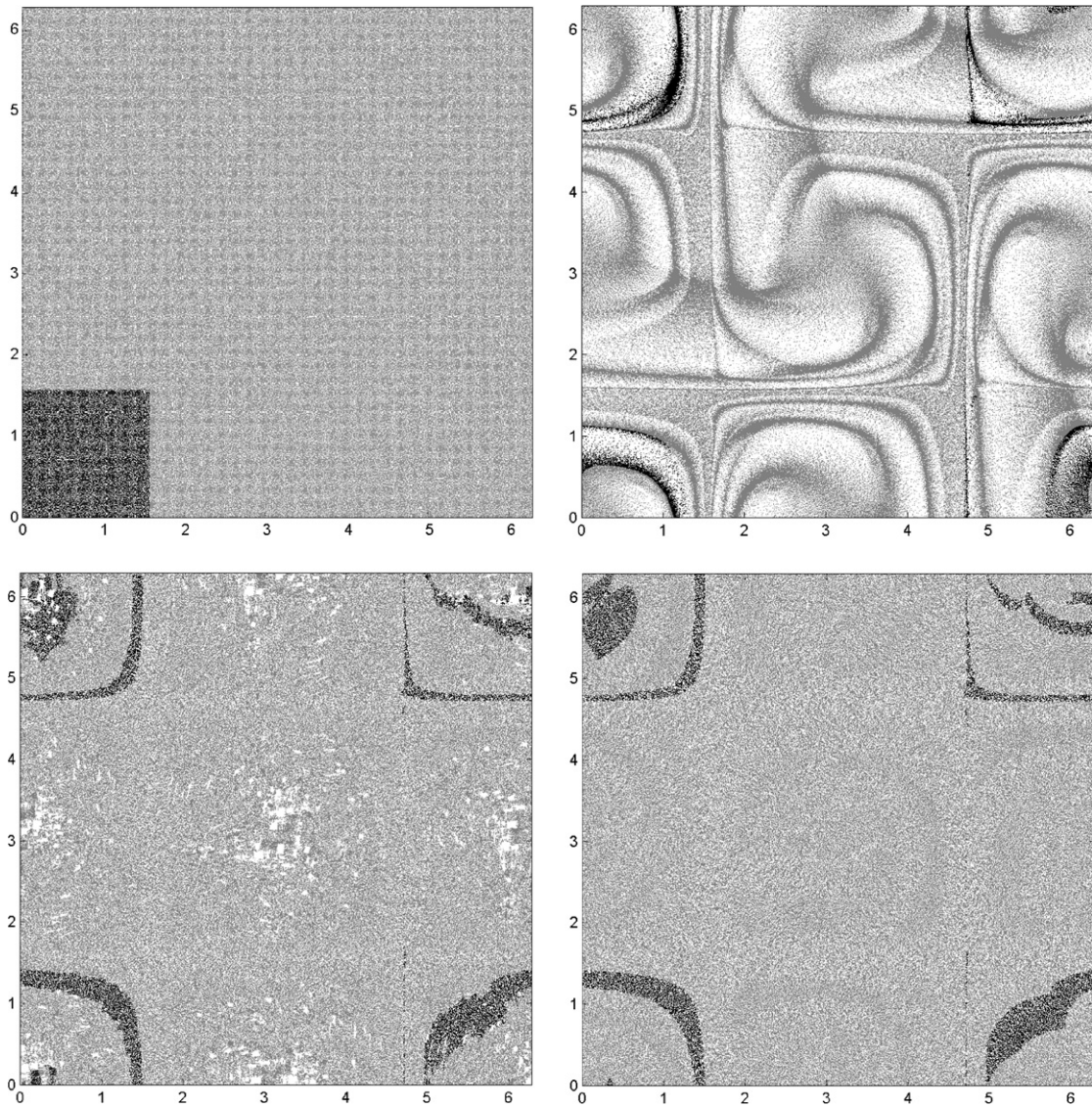


Fig. 1. Comparison between the performance of the bilinear and PERM velocity interpolation schemes, and between the RK2 and MRK2 time-stepping schemes. We consider the 2D, periodic incompressible flow field given by Eqs. (16) and (17) in Section 3. 409,600 particles are initially uniformly spaced inside the domain (top left). Each particle is represented by a light grey dot, except for those initially in the bottom left corner, which are dark grey, for visualization purposes. The particles are advected using different velocity interpolation and time integration schemes. Because the test flow is incompressible and the particles are initially uniformly distributed, they should remain so. Top right: final particle distribution using bilinear interpolation on a 4×4 grid and RK2 time integration. Bottom left: final particle distribution using PERM interpolation on a 4×4 grid and RK2 time integration. Bottom right: final particle distribution using PERM interpolation on a 4×4 grid and MRK2 time integration. It can be seen that the combination of PERM interpolation and MRK2 time stepping performs best at preserving the uniformity of the particle distribution.

face. Note that q as used here is different from the mean particle mass density defined in Eq. (2)), and $\bar{u}_s, \Delta_{u,s}^{(1)}, \Delta_{u,s}^{(2)}$, etc. are parameters which are determined by an algorithm described in [1].

Some of the important properties of PERM are enumerated below. They indicate why a standard time-stepping scheme, such as RK2, does not perform well with this particular velocity reconstruction. For simplicity, we consider here a grid composed of 2D or 3D cubes, of side Δx .

1. For a given velocity component, in the component direction, the reconstructed field is continuous, piecewise parabolic, and second-order accurate with respect to Δx .
2. The flux of the reconstructed velocity field through any given grid cell face is identically equal to the flux implied by the FV face-normal velocity.
3. For a given velocity component, in the component-normal directions, the reconstructed field is piecewise continuous, piecewise linear, and second-order accurate with respect to Δx .

4. The magnitudes of the velocity discontinuities implied by Property 3 are proportional to Δx^2 .
5. The divergence of the reconstructed velocity field is bi- or tri-linear, piecewise continuous, and second-order accurate with respect to Δx .
6. The magnitudes of the divergence discontinuities implied by Property 5 are proportional to Δx^2 .

It should be noted that, as described in [1], the PERM reconstruction scheme requires a Cartesian grid (either uniform or non-uniform in each coordinate). It is also possible to extend PERM to cylindrical coordinate grids.

Furthermore, it should be noted that for a velocity field with the functional form of PERM it is not possible to integrate Eq. (1) analytically – to see why this is so, note that any bi- or tri-linear velocity field is a particular case of a PERM field, and Eq. (1) cannot be integrated exactly for a general bi- or tri-linear velocity field, even if it is time-independent (the differential equation of the Lorentz attractor [6], for example, is a particular case of a tri-linear velocity field). Therefore, we have to utilize a numerical time-integration scheme for the integration of Eq. (1) – a natural candidate is second-order Runge–Kutta (RK2).

However, second-order time accuracy for the RK2 scheme requires that the velocity field is everywhere continuous, and differentiable (in both space and time) everywhere except on a set of measure zero. Therefore, due to the discontinuities in the PERM reconstructed velocity, an RK2 solution is only first-order accurate in time, for a fixed grid size. Also, applying RK2 to a discontinuous velocity field leads to the violation of an important continuity principle, namely: if two particles are initially infinitesimally close, they remain so. These two statements are demonstrated in the next section.

On the other hand, Items 4 and 6 above imply that for a fine mesh the discontinuities in velocity and divergence, and their detrimental effect on the performance of RK2, are negligible. More specifically, if we decrease both the grid size and time step, keeping the Courant number fixed, a solution which combines PERM and RK2 is second-order accurate. Nevertheless, it is often the case in large eddy simulation/filtered density function (LES/FDF) methods that the grid used provides only marginal resolution of the filtered velocity field, and hence the discontinuities in the PERM reconstructed velocity field should not be neglected.

2. Examination of RK2 for discontinuous velocity fields

Let us begin with a simple example of the problems that we encounter when we use RK2 on a discontinuous velocity field. Instead of thinking about a grid, consider an infinite, time-independent velocity field in 2D, which has the following form:

$$U(x,y) = 1, \tag{7}$$

$$V(x,y) = \begin{cases} 0, & \text{for } x < 0, \\ 1, & \text{for } x \geq 0. \end{cases} \tag{8}$$

Here, we can think of the regions $x < 0$ and $x > 0$ as two grid cells, with the region $x = 0$ as the face between them. Note that this velocity field has the kind of discontinuity which can be caused by PERM: U is continuous in the x -direction, and V is continuous in the y -direction, but V is not continuous in the x -direction, its component-normal direction. Now, consider two particles at time $t = 0$: one at $(X_{1,0}, Y_{1,0}) = (-\Delta t - \epsilon, 0)$ and the other at $(X_{2,0}, Y_{2,0}) = (-\Delta t + \epsilon, 0)$, where $0 < \epsilon \ll \Delta t$. An RK2 step of length Δt has the form:

$$\mathbf{X}^{(1)} = \mathbf{X}_0 + \Delta t \mathbf{U}(\mathbf{X}_0, 0), \tag{9}$$

$$\mathbf{X}_{\Delta t}^{\text{RK2}} = \mathbf{X}_0 + \frac{1}{2} \Delta t (\mathbf{U}(\mathbf{X}_0, 0) + \mathbf{U}(\mathbf{X}^{(1)}, \Delta t)). \tag{10}$$

Substituting their initial positions for \mathbf{X}_0 in Eqs. (9) and (10), we determine that the final positions of the two particles, after one RK2 step of length Δt , are $(X_{1,\Delta t}^{\text{RK2}}, Y_{1,\Delta t}^{\text{RK2}}) = (-\epsilon, 0)$ and $(X_{2,\Delta t}^{\text{RK2}}, Y_{2,\Delta t}^{\text{RK2}}) = (\epsilon, \Delta t/2)$. On the other hand, with perfect time advancement, the final particle positions at time Δt are $(X_{1,\Delta t}, Y_{1,\Delta t}) = (-\epsilon, 0)$ and $(X_{2,\Delta t}, Y_{2,\Delta t}) = (\epsilon, \epsilon/2)$. Therefore, we can see that the RK2 position of the second particle differs from the correct position by a distance of $(\Delta t - \epsilon)/2$. Keeping in mind that we set $\epsilon \ll \Delta t$, we have that this single time step introduces an $O(\Delta t)$ error in the final position of the second particle. Therefore, RK2 is first-order accurate in time overall, and for a time step that crosses a discontinuity it is zeroth-order accurate (in the sense that if all time steps introduced an error of the same magnitude as that introduced by a discontinuity-crossing step, the overall error in the final position of a particle would be $O(\Delta t^0)$).

Furthermore, if we take the limit as $\epsilon \rightarrow 0$, we can see that the distance between the initial positions of the two particles goes to zero, but the distance between the final positions after the RK2 step goes to $\Delta t/2$. As previously mentioned, this is a violation of a fundamental continuity principle, namely that if $\mathbf{X}(\mathbf{Y}, t)$ is the Lagrangian mapping between a particle's initial and final positions (i.e., if a particle is at \mathbf{Y} at time 0, it will be at $\mathbf{X}(\mathbf{Y}, t)$ at time t), then $\mathbf{X}(\mathbf{Y}, t)$ is continuous in both \mathbf{Y} and t , except at regions where there is a velocity discontinuity and the face-normal velocity is zero.

The considerable RK2 position error for the second of the two particles is introduced by the following fact: even though under perfect time advancement the second particle is in the region $x > 0$ only for a time $\epsilon \ll \Delta t/2$, the corrected velocity of the RK2 step $(\mathbf{U}(\mathbf{X}_{2,0}, 0) + \mathbf{U}(\mathbf{X}_2^{(1)}, \Delta t))/2$, is the average velocity that the particle would experience if it were in that region for a time $\Delta t/2$. In other words, RK2 does not account for how much time a particle spends on each side of a discontinuity. This is the motivation for the Multi-Step RK2 scheme, which we now describe in detail.

2.1. Description of MRK2

In this section, we present a description of a Multi-Step RK2 time step. We consider a particle that is initially at the position \mathbf{X}_0 at time 0, and which is to be advected for a time step of length Δt . For the moment, let us consider the case in which there is only one velocity discontinuity, as in the example above.

First, we take an RK2 predictor step, according to Eq. (9):

$$\mathbf{X}^{(1)} = \mathbf{X}_0 + \Delta t \mathbf{U}(\mathbf{X}_0, 0).$$

If \mathbf{X}_0 and $\mathbf{X}^{(1)}$ are in the same cell, then we take the remainder of a standard RK2 time step, according to Eq. (10), which gives the final result:

$$\mathbf{X}_{\Delta t}^{\text{MRK2}} = \mathbf{X}_{\Delta t}^{\text{RK2}} = \mathbf{X}_0 + \Delta t (\mathbf{U}(\mathbf{X}_0, 0) + \mathbf{U}(\mathbf{X}^{(1)}, \Delta t)) / 2.$$

Otherwise, let \mathbf{X}' be the point where the ray from \mathbf{X}_0 to $\mathbf{X}^{(1)}$ first intersects a cell face. For Cartesian, cylindrical and unstructured tetrahedral grids, in order to determine \mathbf{X}' we have to identify the faces of the current grid cell, each of which is a piece of a plane or a cylinder in 2D or 3D, and identify the points where the ray from \mathbf{X}_0 to $\mathbf{X}^{(1)}$ crosses each of them, if it does (a simple problem in analytic geometry). Out of these points, \mathbf{X}' is the one closest to \mathbf{X}_0 .

We make the following second-order accurate estimate for the time, Δt_1 , that it takes for the particle to reach this discontinuity:

$$\Delta t_1 = \Delta t \|\mathbf{X}' - \mathbf{X}_0\| / \|\mathbf{X}^{(1)} - \mathbf{X}_0\|, \quad (11)$$

where $\|\cdot\|$ denotes the 2-norm. Next, we break up the RK2 predictor step into two substeps. The first predictor substep, from time 0 to Δt_1 , and from \mathbf{X}_0 to \mathbf{X}' , accounts for the advection of the particle before it reaches the discontinuity. The second predictor substep, which accounts for advection after the particle crosses the discontinuity, takes the particle from time Δt_1 to Δt and from \mathbf{X}' to $\mathbf{X}^{(2)}$, where $\mathbf{X}^{(2)}$ is given by

$$\mathbf{X}^{(2)} = \mathbf{X}' + (\Delta t - \Delta t_1) \mathbf{U}^+(\mathbf{X}', 0). \quad (12)$$

There are two possible definitions of velocity at \mathbf{X}' , where the velocity field is discontinuous. The value $\mathbf{U}^+(\mathbf{X}', 0)$ denotes the velocity experienced by the particle after it has crossed the discontinuity. Similarly, below we use $\mathbf{U}^-(\mathbf{X}', 0)$ to denote the velocity experienced by the particle before it crosses the discontinuity. (We have assumed, for the moment, that the particle does not cross another discontinuity between \mathbf{X}' and $\mathbf{X}^{(2)}$.)

Next, we make second-order linear approximations in time to estimate the velocities at \mathbf{X}' at the intermediate time Δt_1

$$\mathbf{U}^-(\mathbf{X}', \Delta t_1) \approx \widehat{\mathbf{U}}^-(\mathbf{X}', \Delta t_1) = [(\Delta t - \Delta t_1) / \Delta t] \mathbf{U}^-(\mathbf{X}', 0) + [\Delta t_1 / \Delta t] \mathbf{U}^-(\mathbf{X}', \Delta t), \quad (13)$$

$$\mathbf{U}^+(\mathbf{X}', \Delta t_1) \approx \widehat{\mathbf{U}}^+(\mathbf{X}', \Delta t_1) = [(\Delta t - \Delta t_1) / \Delta t] \mathbf{U}^+(\mathbf{X}', 0) + [\Delta t_1 / \Delta t] \mathbf{U}^+(\mathbf{X}', \Delta t). \quad (14)$$

Finally, we calculate the corrected velocities for each of the two substeps, and obtain the final particle position

$$\mathbf{X}_{\Delta t}^{\text{MRK2}} = \mathbf{X}_0 + (\Delta t_1 / 2) (\mathbf{U}(\mathbf{X}_0, 0) + \widehat{\mathbf{U}}^-(\mathbf{X}', \Delta t_1)) + ((\Delta t - \Delta t_1) / 2) (\widehat{\mathbf{U}}^+(\mathbf{X}', \Delta t_1) + \mathbf{U}(\mathbf{X}^{(2)}, \Delta t)). \quad (15)$$

As already mentioned, for a particle which crosses the discontinuity, we break up the time step into two substeps. The two terms added to \mathbf{X}_0 in Eq. (15) are respectively the contributions of each of those two substeps to the advancement of the particle. For simplicity's sake, the algorithm presented above allows for two substeps at most, and can therefore deal with at most one velocity discontinuity crossed by a particle during a time step. By allowing for a greater number of substeps (e.g., breaking up the substep from Δt_1 to Δt into two substeps if the ray from \mathbf{X}' to $\mathbf{X}^{(2)}$ crosses another discontinuity), the algorithm is extended to deal with an arbitrary number of discontinuities crossed by a particle in one time step.

Below, we make a few important observations about MRK2:

1. The MRK2 procedure is almost the same as taking two RK2 steps – one from 0 to Δt_1 and one from Δt_1 to Δt – with the one difference being that the MRK2 procedure does not require knowledge of the velocity field at the intermediate time Δt_1 , and uses instead a linear interpolation in time between the velocity fields at time 0 and Δt . This difference is essential in an LES/FDF context, because during each time step a fraction, approximately equal to the Courant number, of the total number of particles in the domain cross a cell face. For a typical LES/FDF calculation with 10^7 particles and a Courant number of 0.1, this means that for each time step there would be approximately 10^6 particles crossing a cell face, each at a different intermediate time Δt_1 . It would be extremely inefficient, if not infeasible, to calculate the whole reconstructed velocity field at each of the (of order 10^6) intermediate times. Instead, MRK2 uses just two fields to obtain the needed local information with the required accuracy.
2. For a fixed grid size, and subject to a condition which is described in the next item, it can be shown analytically that an MRK2 step which crosses a discontinuity introduces an $O(\Delta t^2)$ error in the final particle position, and preserves the continuity of the Lagrangian mapping $\mathbf{X}(\mathbf{Y}, \Delta t)$. Therefore, MRK2 preserves continuity (in a restricted sense) and is second-order accurate in time, since the number of discontinuities crossed by a particle does not increase with decreasing time step.

3. The condition necessary for Item 2 is that the velocity normal to a face (discontinuity) is not identically zero. MRK2 does not preserve the continuity of $\mathbf{X}(\mathbf{Y}, \Delta t)$ where the face-normal velocity is zero. To see this, consider a velocity field similar to the example at the beginning of Section 2, this time with $U(x, y) = 0$, and consider two particles: one at $(X_{1,0}, Y_{1,0}) = (-\epsilon, 0)$ and the other at $(X_{2,0}, Y_{2,0}) = (\epsilon, 0)$. The result of an MRK2 time step is the same as that of an RK2 time step, and of perfect time advancement: $(X_{1,\Delta t}, Y_{1,\Delta t}) = (-\epsilon, 0)$ and $(X_{2,\Delta t}, Y_{2,\Delta t}) = (\epsilon, \Delta t)$. Taking limits as $\epsilon \rightarrow 0$, we can see why continuity is violated. Note that this is a property of the particular velocity field considered, not a shortcoming of MRK2 (i.e., even with perfect time advancement there is a continuity violation).
4. The MRK2 procedure is applicable to a wide variety of grids – it has only two grid-dependent aspects, namely the evaluation of the interpolated velocity at an arbitrary point in space, and the determination of the point \mathbf{X}' where the ray from \mathbf{X}_0 to $\mathbf{X}^{(1)}$ first intersects a cell face. The first of these aspects, the ability to evaluate interpolated velocity at an arbitrary point, is a general requirement for any reasonable velocity interpolation scheme. The second aspect, the ability to determine efficiently \mathbf{X}' , depends on the grid geometry and, as mentioned in the description of MRK2, reduces to a set of simple analytic geometry problems, such as finding the intersection of a ray with a plane (or cylinder). Therefore, in addition to Cartesian and cylindrical grids, MRK2 in itself is also applicable all grids with planar or cylindrical cell faces (e.g., unstructured tetrahedral grids). However, we should also keep in mind that the PERM velocity interpolation scheme, which is the primary motivation for the modification from RK2 to MRK2, has currently been developed only for Cartesian and cylindrical grids.

3. Comparison between the performance of RK2 and MRK2 when applied to a PERM reconstructed velocity field

In this section, we apply both RK2 and MRK2 to a 2D test problem, and compare the performance of the two time-stepping schemes based on three criteria: preservation of continuity, accuracy relative to perfect time advancement, and the maintaining of a consistent subgrid particle distribution.

We use the following problem framework: we have a 2D domain, $[0, 2\pi] \times [0, 2\pi]$, with periodic boundary conditions. We consider two 2D periodic flows. Flow 1 is incompressible, given by the formula

$$U(x, y, t) = 1 - 2 \cos(x - t) \sin(y - t), \quad (16)$$

$$V(x, y, t) = 1 + 2 \sin(x - t) \cos(y - t). \quad (17)$$

Flow 2 is compressible, given by

$$U(x, y, t) = 1 - 2 \cos(x - t) \sin(y - t) + \frac{1}{2} \sin(x) \cos(2t), \quad (18)$$

$$V(x, y, t) = 1 + 2 \sin(x - t) \cos(y - t) + \frac{1}{2} \sin(y) \cos(2t). \quad (19)$$

Streamline plots of these two test flows, for $t = 0$, are shown in Fig. 2.

In order to compute the reconstructed field, the PERM reconstruction scheme uses values of U (according to Eq. (16) or (18)) at the centers of the vertical cell faces, and values of V (according to Eq. (17) or (19)) at the centers of the horizontal cell faces – this simulates FV velocity information. Once the subgrid velocity is reconstructed by PERM, the respective time advancement scheme is used to advect the particles. The time step Δt is directly proportional to the Courant number (C), which is defined here as

$$C = \max(\max(U(x, y, t)\Delta t/\Delta x), \max(V(x, y, t)\Delta t/\Delta x)), \quad (20)$$

where Δx is the side of a grid cell, and the latter two maxima are over x, y, t .

3.1. Preservation of continuity, and a qualitative analysis of particle distributions

Here, we present results for the incompressible Flow 1, Eqs. (16) and (17). One of the beneficial properties of PERM is that if the velocity field is divergence free at the FV level, then the subgrid reconstructed velocity is also divergence free. Therefore, applying PERM to Flow 1, we obtain a subgrid velocity field that is divergence free, and so (with perfect time advancement) a particle distribution which is initially uniform remains uniform.

Here, we advect 409,600 particles which are initialized in the following manner: the domain is broken up into 640×640 squares of equal size, and a single particle is placed randomly, with uniform probability density, in each square. This allows for an initial particle distribution which is both very uniform and does not degenerate when a large value of strain is applied to it. Note that the number of particles that we are using is extremely large – 25,600 particles per cell, for a 4×4 grid. Naturally, in practical LES/FDF applications this number is much smaller. The reason why we are using so many here is for the reader to be able to better visualize the mappings $\mathbf{X}(\mathbf{Y}, t)$ between initial and final particle positions. In the following figures, each of the particles is shown as a single light grey dot, except for those which are initially in the region $[0, \pi/2] \times [0, \pi/2]$, at the lower-left corner. These particles are dark grey, in order to help visualize the strain of the particle distribution.

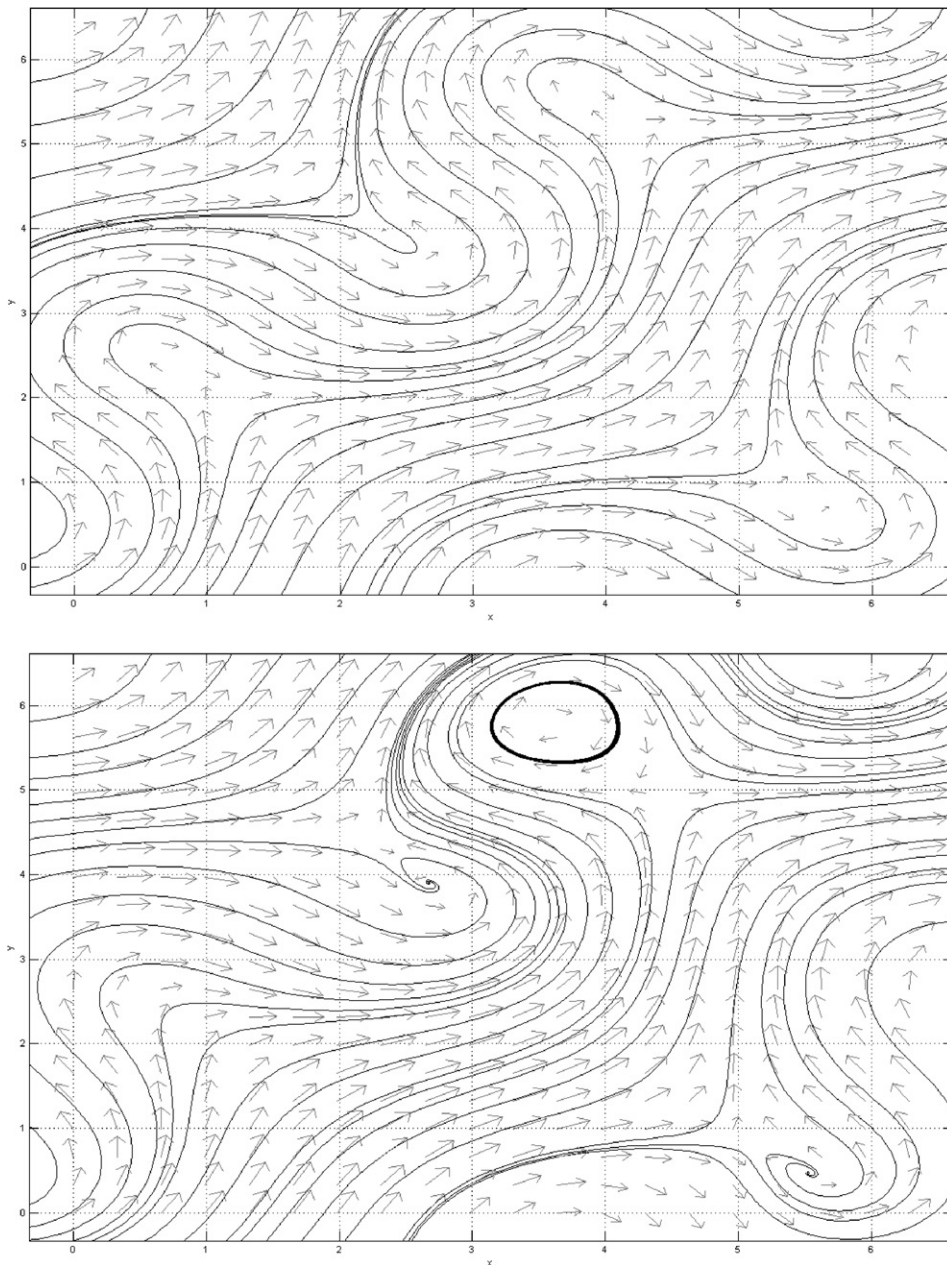


Fig. 2. Top: streamlines, at $t = 0$, of the incompressible Flow 1 (Eqs. (16) and (17)). Bottom: streamlines, at $t = 0$ of the compressible Flow 2 (Eqs. (18) and (19)).

Fig. 3 shows results for a 4×4 grid, with $C = 1$. We note that this is a rather large Courant number - it is used mainly to emphasize the effects of both time advancement schemes. After a single time step, we can see that the RK2 distribution (top left of Fig. 3) has considerable voids where there are no particles at all - this indicates that the mapping $\mathbf{X}(\mathbf{Y}, \Delta t)$ is not surjective (onto), since there are values of \mathbf{X} which cannot be attained for any value of \mathbf{Y} . We can also see regions in the RK2 distributions where the particles are twice as dense - this indicates that the mapping $\mathbf{X}(\mathbf{Y}, \Delta t)$ is not injective (one-to-one), since regions which were initially separate are mapped on top of each other. This is a serious problem - it introduces error in the subgrid particle distribution (as we will see later on) and violates the fundamental principle that for a periodic flow $\mathbf{X}(\mathbf{Y}, \Delta t)$ is both injective and surjective.

A single MRK2 time step, too, has these problems, but to a much lesser extent - we can see (from the top right of Fig. 3) four elongated gaps where there are no particles, but the size of these gaps is much smaller than the size of the holes in the particle distribution created by RK2. These gaps in the MRK2 distribution are to be expected - they correspond to regions where the face-normal velocity is zero, as explained in Section 2. Comparing the particle distributions after one flow-through

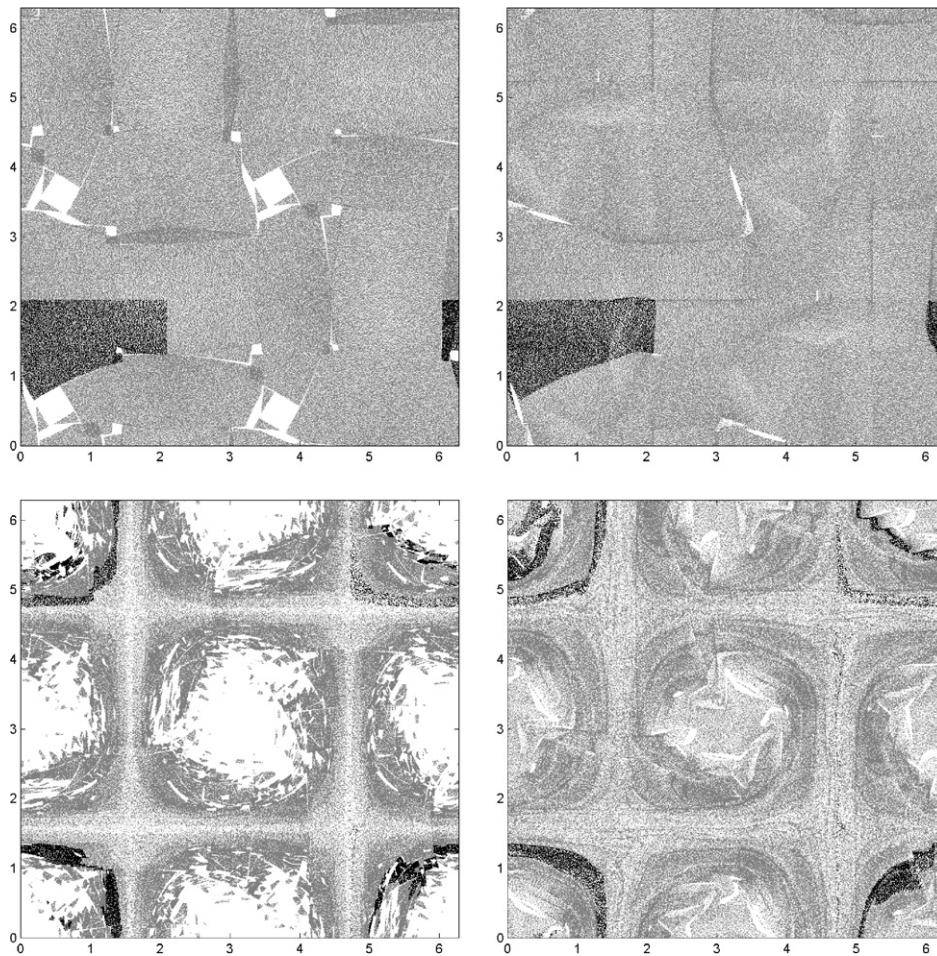


Fig. 3. Particle distributions in Flow 1 for a 4×4 grid, $C = 1$. Top left: RK2, one time step ($\Delta t = 0.448$); top right: MRK2, one time step; bottom left: RK2, one flow-through time (12 time steps); bottom right: MRK2, one flow-through time. The dark grey particles are those that are in the lower-left corner of the domain at $t = 0$.

time (12 time steps), we can see the same differences but with greater magnitude. Whereas, MRK2 does not preserve the uniformity of the particle distribution very well, it does much better than RK2, where there are considerably larger void regions.

Fig. 4 shows results for the same value of the Courant number, $C = 1$, but for a finer 8×8 grid. Because the magnitudes of the discontinuities in PERM are $O(\Delta x^2)$ [1], the velocity discontinuities in this case are theoretically smaller by a factor of four, and RK2 does almost as well as MRK2.

Fig. 5 shows results for a 4×4 grid, with a smaller Courant number, $C = 0.25$. We can again see that RK2 does not preserve continuity very well – there are noticeable voids in the RK2 distribution, both after one time step and after one flow-through time. Although we know theoretically that there are also voids in the MRK2 distributions, these cannot be perceived on the plots. We also note that MRK2 performs better at preserving the sharp interface between the light and dark grey regions.

Fig. 6 shows results for an 8×8 grid, $C = 0.25$. We cannot perceive a difference between the two particle distributions, which suggests that, due to the smaller value of velocity discontinuities, the performance of RK2 is similar to that of MRK2.

From these comparisons of particle distributions produced by RK2 and MRK2, we conclude that when the velocity discontinuities yielded by the reconstruction are considerable (as for the 4×4 grid), MRK2 performs better than RK2 at preserving the continuity of the particle distribution and the consistency between the subgrid particle distribution and the LES filtered density.

3.2. Accuracy relative to perfect time advancement

As we are focused on developing and demonstrating an effective time advancement scheme for a given velocity reconstruction, we are interested in the error of a given solution relative to perfect time advancement. Since we do not have

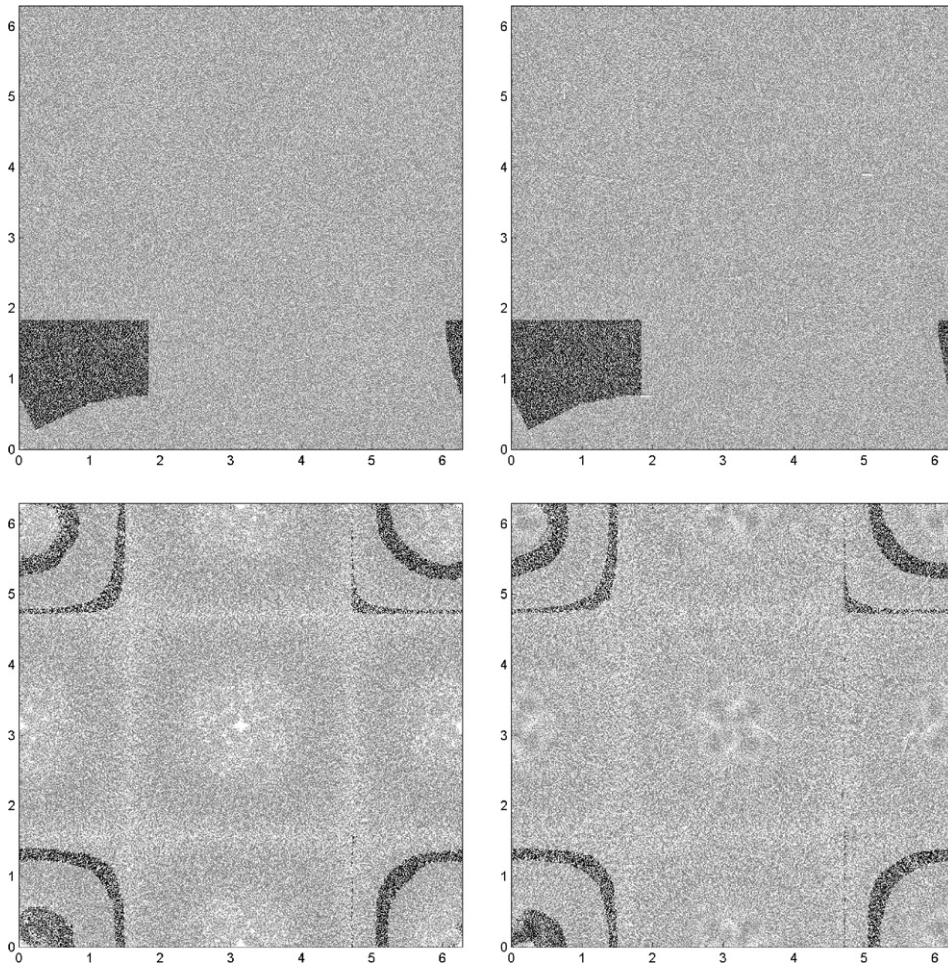


Fig. 4. Particle distributions in Flow 1 for an 8×8 grid, $C = 1$. Top left: RK2, one time step ($\Delta t = 0.224$); top right: MRK2, one time step; bottom left: RK2, one flow-through time (24 time steps); bottom right: MRK2, one flow-through time. The dark grey particles are those that are in the lower-left corner of the domain at $t = 0$.

an exact perfect time advancement solution, we use instead an MRK2 solution with a very low Courant number ($C = 1/512$) as an approximation.

First of all, we consider position error. We use $N = 256$ particles, uniformly distributed across the domain at time 0. We denote by $\mathbf{Y}^{(i)}$ the initial position of the i th particle, and we denote by $\mathbf{X}^{\text{RK2}}(\mathbf{Y}^{(i)}, t)$, $\mathbf{X}^{\text{MRK2}}(\mathbf{Y}^{(i)}, t)$, and $\mathbf{X}(\mathbf{Y}^{(i)}, t)$ the corresponding Lagrangian position mappings yielded by RK2, MRK2 and perfect time advancement, respectively. The position error of a given RK2 solution is then defined as

$$\epsilon_x^{\text{RK2}} = \frac{1}{N} \left(\sum_{i=1}^N \|\mathbf{X}^{\text{RK2}}(\mathbf{Y}^{(i)}, t) - \mathbf{X}(\mathbf{Y}^{(i)}, t)\| \right). \quad (21)$$

The MRK2 position error is defined analogously. As we already mentioned, it can be shown analytically that, with Δt being the length of the time step, the RK2 position error is of order $O(\Delta t)$, and the MRK2 position error is of order $O(\Delta t^2)$.

We also consider error of infinitesimal volume expansion (which we will refer to as dV error, for the sake of brevity). Note that we refer here to two-dimensional volume, i.e., area. For an infinitesimal material element whose initial volume is dV_0 , and whose initial position is \mathbf{Y} , the final volume, dV_t , is given by

$$dV_t = dV_0 \det \left(\frac{\partial X_j(\mathbf{Y}, t)}{\partial Y_k} \right), \quad (22)$$

where $\mathbf{X}(\mathbf{Y}, t)$ is the Lagrangian position mapping previously mentioned. This is the basis of the change of variables formula, which in this context tells us that the volume at time t of a set S is

$$V_t = \int_S \det \left(\frac{\partial X_j(\mathbf{Y}, t)}{\partial Y_k} \right) |d\mathbf{Y}|, \quad (23)$$

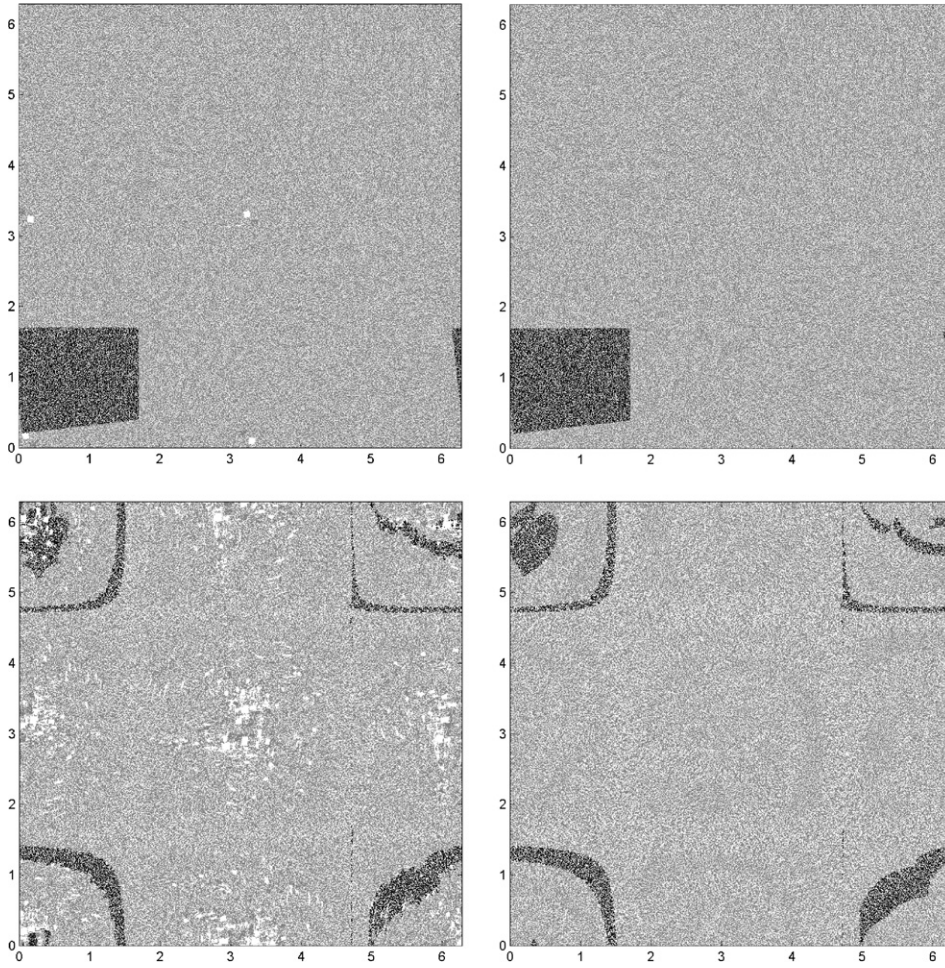


Fig. 5. Particle distributions in Flow 1 for a 4×4 grid, $C = 0.25$. Top left: RK2, one time step ($\Delta t = 0.112$); top right: MRK2, one time step; bottom left: RK2, one flow-through time (48 time steps); bottom right: MRK2, one flow-through time. The dark grey particles are those that are in the lower-left corner of the domain at $t = 0$.

provided that the mapping $\mathbf{X}(\mathbf{Y}, t)$ is one-to-one. Therefore, accurate values of $\det(\partial X_j / \partial Y_k)$ are essential for achieving an accurate subgrid particle distribution. We also note that $\mathbf{X}(\mathbf{Y}, t)$ may not be one-to-one if its continuity is violated (such as in Fig. 3, where particles are twice as dense in some places, because certain regions are mapped on top of each other), and therefore continuity of the position mapping, which is considered in the previous subsection, is just as essential for getting the right value of V_t as are accurate values of $\det(\partial X_j / \partial Y_k)$.

With this in mind, we define dV error of a given RK2 solution as

$$\epsilon_{dV}^{\text{RK2}} = \frac{1}{N} \sum_{i=1}^N \left| \det \left(\frac{\partial X_j^{\text{RK2}}(\mathbf{Y}^{(i)}, t)}{\partial Y_k} \right) - \det \left(\frac{\partial X_j(\mathbf{Y}^{(i)}, t)}{\partial Y_k} \right) \right|. \tag{24}$$

The MRK2 dV error is similarly defined. For RK2, it can be proven analytically that the dV error is $O(\Delta t^2)$ when the divergence of the PERM reconstructed field is continuous everywhere, and $O(\Delta t)$ when the divergence is discontinuous from cell to cell, which is the case for all compressible flows.

Here, we cannot calculate the necessary Jacobians exactly, but a sufficiently accurate numerical estimate which we use is

$$\det \left(\frac{\partial X_j}{\partial Y_k} \right) \approx A_t / A_0, \tag{25}$$

where A_0 is the initial area of a triangle initially of sides 10^{-8} formed by three particles at time 0, and A_t is the area of the triangle formed by those three particles at time t .

Here, we present results for the compressible Flow 2, Eqs. (18) and (19). Fig. 7 shows plots of position error and dV error for 4×4 and 8×8 grid solutions, with the Courant number ranging from 1 down to $1/32$. For both position error and dV error, RK2 exhibits first-order behavior, whereas MRK2 exhibits second-order behavior. This is consistent with previously

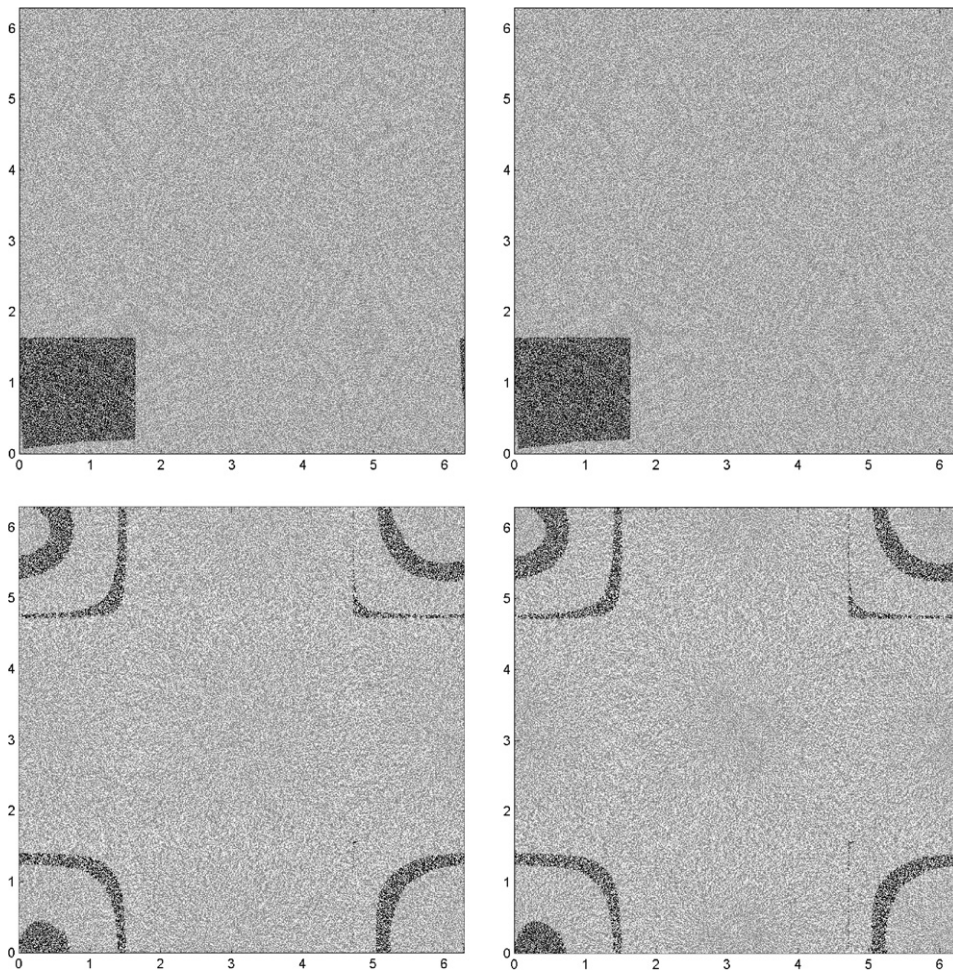


Fig. 6. Particle distributions in Flow 1 for an 8×8 grid, $C = 0.25$. Top left: RK2, one time step ($\Delta t = 0.056$); top right: MRK2, one time step; bottom left: RK2, one flow-through time (96 time steps); bottom right: MRK2, one flow-through time. The dark grey particles are those that are in the lower-left corner of the domain at $t = 0$.

stated analytic results, and it also indicates that for MRK2, dV error is $O(\Delta t^2)$ – a result which we have not been able to prove rigorously.

It should also be noted that, for larger values of C , MRK2 is computationally slower than RK2. In order to compare the efficiency of MRK2 with that of RK2, Fig. 8 shows plots of the position and dV errors from Fig. 7 versus simulation time, for a simulation with 1600 particles per cell, instead of just the 256 particles (for the entire domain) that we used to determine the errors. We use a larger number of particles here to ensure that the cost of all the operations other than particle advection (such as the determination of the PERM coefficients) is negligible.

From Fig. 8 we see that MRK2 becomes more efficient as the error (and hence the Courant number) decreases – for the 4×4 grid, MRK2 is more efficient for $C \leq 1/2$, and for the 8×8 grid, MRK2 is more efficient for $C \leq 1/4$. On the other hand, Fig. 8 takes into account only the computational cost for particle advection – if we also consider the cost of chemistry calculations, which is usually much higher than the cost of particle advection, we conclude that MRK2 may be more efficient than RK2 even for larger values of C .

Table 1 shows values of the error, as well as simulation time for a 1600-particles-per-cell simulation, for $C = 1$ and for $C = 0.25$. For both values of C , and for both meshes, MRK2 has a smaller position error: $\epsilon_x^{\text{MRK2}}/\epsilon_x^{\text{RK2}} < 0.69$. For both meshes, the MRK2 dV error is less than the RK2 dV error for $C = 0.25$, but for $C = 1$ and the 8×8 mesh, the MRK2 dV error is greater than the RK2 dV error. This, however, is not a major concern, because as we already noted, $C = 1$ is a very large value for the Courant number – in most applications, we expect a smaller value, such as 0.25, to be used.

3.3. Accuracy of the subgrid particle distribution

Previously, McDermott and Pope [1] have shown that a PERM reconstructed velocity field, with RK2 as the time-stepping scheme, performs much better than linear interpolation in maintaining a subgrid particle distribution which is consistent

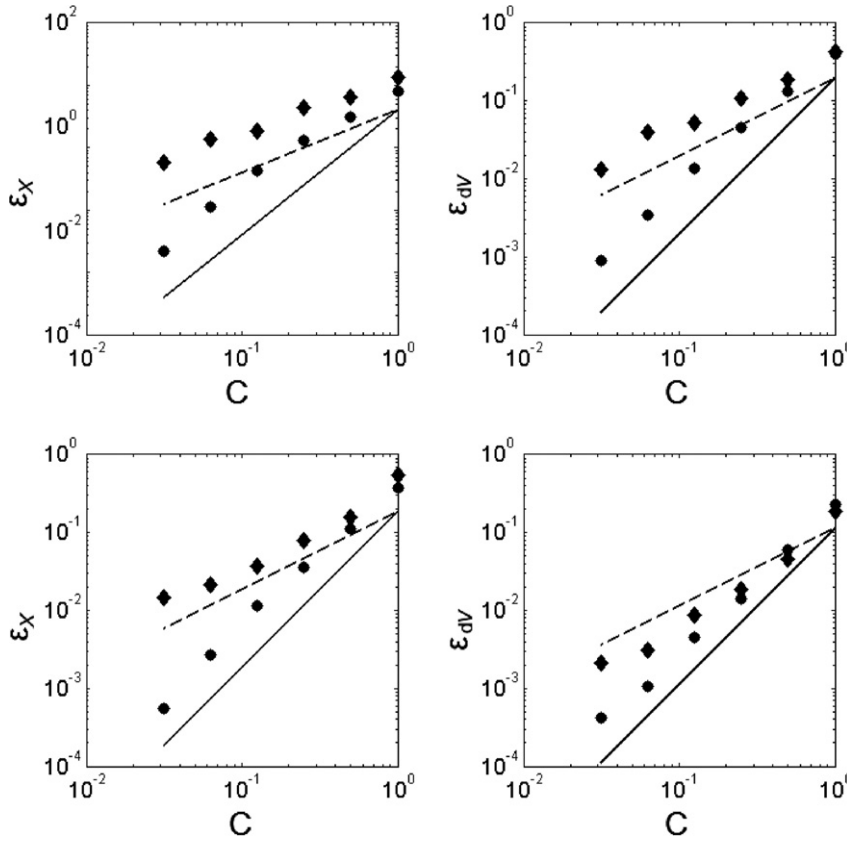


Fig. 7. Results for position and dV error in Flow 2 at $t = 8.97$. Top left: position error for a 4×4 grid; top right: dV error for a 4×4 grid; bottom left: position error for a 8×8 grid; bottom right: dV error for a 8×8 grid. The diamonds denote RK2 error, the circles denote MRK2 error. The dashed line indicates first-order convergence, and the solid line indicates second-order convergence.

with the LES filtered density. Here, we demonstrate that this consistency becomes even better when MRK2 is used as the time-stepping scheme.

For the present tests, we use the incompressible Flow 1, Eqs. (16) and (17). The particles are initialized in a manner similar to that in Section 3.1 (by breaking up the domain into as many squares as there are particles, and initializing a single particle randomly into each square). Then the mean particle mass density is initially the same everywhere in the flow, and it remains so under perfect time advancement, for a divergence-free velocity field such as the one we consider. Hence, the expected number of particles in a set S is directly proportional to the area of S .

We quantify the accuracy of the subgrid particle distribution in the following way: we break up the domain into equal sized squares, which we call sampling squares. We use sampling squares of three different sizes, with sides $\pi/4$, $\pi/8$, and $\pi/16$ (respectively $1/2$, $1/4$ and $1/8$ of the sidelength of a grid cell, for a 4×4 grid). At any given timestep, let N_i be the number of particles in the i th sampling square, let N_{mean} be the expected number of particles for a square of this size, and let N_{sq} be the number of sampling squares in the domain. The subgrid particle distribution error is then defined as

$$\epsilon_{\text{SG}} = \left(\sum_{i=1}^{N_{\text{sq}}} | (N_i - N_{\text{mean}}) / N_{\text{mean}} | \right) / N_{\text{sq}}. \tag{26}$$

This subgrid distribution error has two contributions: first, a probabilistic error, due to the random particle initialization; and, second, a bias error, due to the particle advection scheme (an example of this are the considerable voids in the particle distribution left by RK2 at $C = 1$). For this test flow, the PERM reconstructed velocity field has zero divergence, and therefore a perfect time advancement solution contains only the probabilistic component of the error. Therefore, we compare the subgrid distribution errors given by the MRK2, RK2 and perfect time advancement solutions, making the assumption that whatever is in excess of the perfect time advancement error is mostly bias error.

Fig. 9 shows results for 65,536 particles, $C = 0.25$ and a 4×4 grid. As may be seen, the MRK2 subgrid distribution error is much closer to the perfect time advancement error than is the RK2 error. From this we deduce that, for the coarse 4×4 grid, the bias error caused by MRK2 is smaller than the bias error caused by RK2.

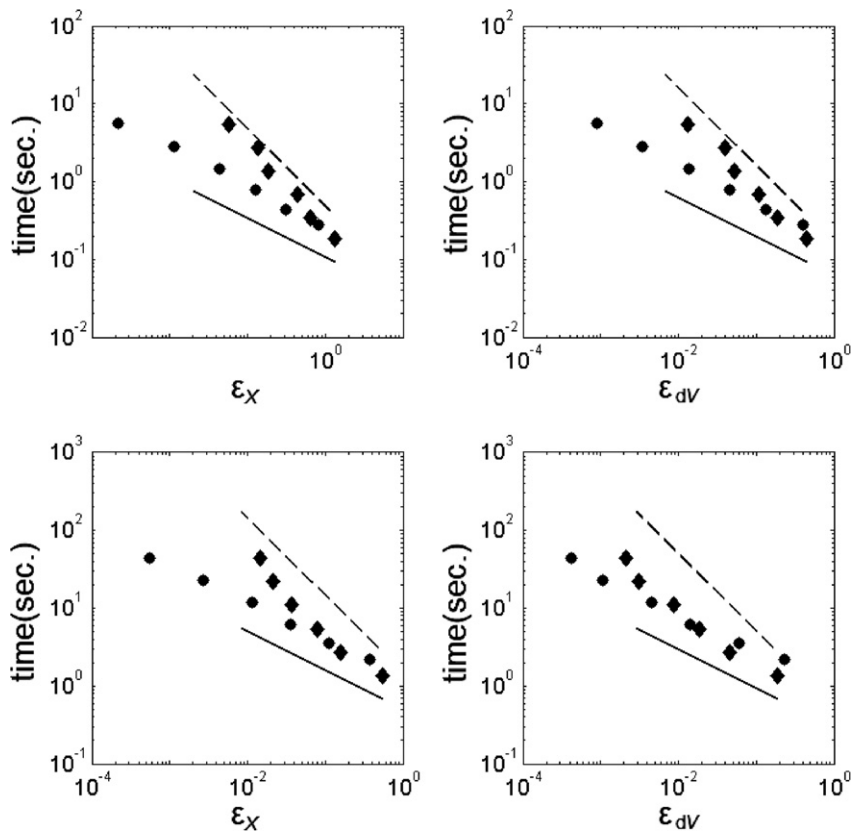


Fig. 8. Comparison between the efficiency of MRK2 and RK2, based on position and dV error, and the computational cost of particle advection for a 1600-particles-per-cell simulation of Flow 2 at $t = 8.97$. Top left: position error for a 4×4 grid; top right: dV error for a 4×4 grid; bottom left: position error for a 8×8 grid; bottom right: dV error for a 8×8 grid. The diamonds denote RK2 error, the circles denote MRK2 error. The dashed line indicates first-order convergence, and the solid line indicates second-order convergence.

Table 1

Summary of position and dV errors for Flow 2 at $t = 8.97$

Case	ϵ_x		ϵ_{dV}	
	RK2	MRK2	RK2	MRK2
$4 \times 4, C = 1$	1.32×10^0	8.17×10^{-1}	4.35×10^{-1}	3.97×10^{-1}
$4 \times 4, C = 0.25$	4.35×10^{-1}	1.29×10^{-1}	1.07×10^{-1}	4.56×10^{-2}
$8 \times 8, C = 1$	5.47×10^{-1}	3.80×10^{-1}	1.88×10^{-1}	2.33×10^{-1}
$8 \times 8, C = 0.25$	7.83×10^{-2}	3.55×10^{-2}	1.84×10^{-2}	1.41×10^{-2}

Fig. 10 shows results for 65,536 particles, $C = 0.5$ and a 8×8 grid. The time step is the same as for the results on Fig. 6, but the velocity field is better resolved by PERM. Here, the MRK2, RK2 and perfect time advancement errors are much closer together, and MRK2 does not provide a big advantage over RK2. Again, we observe that for a finer grid RK2 does not perform considerably worse than MRK2, because the discontinuities in velocity are smaller.

3.4. Computational cost

We have seen that MRK2 has superior accuracy than RK2, and has better performance in preserving the consistency between the subgrid particle distribution and the filtered LES density. One disadvantage that MRK2 has over RK2 is that it is slower – this was mentioned in Section 3.2, where we compared the efficiency of MRK2 with that of RK2.

For the well-optimized Fortran 90 implementation of MRK2 which we used to obtain the results in the previous sections, each additional substep after the first one costs approximately 1.5 times the cost of an RK2 step. In other words, if the total time for one RK2 time step for one particle is t_{step} , the total time for an MRK2 time step, for a particle which crosses n discontinuities, is approximately $t_{\text{step}}(1 + 1.5n)$. Therefore, for a flow in which each particle crosses one discontinuity per time

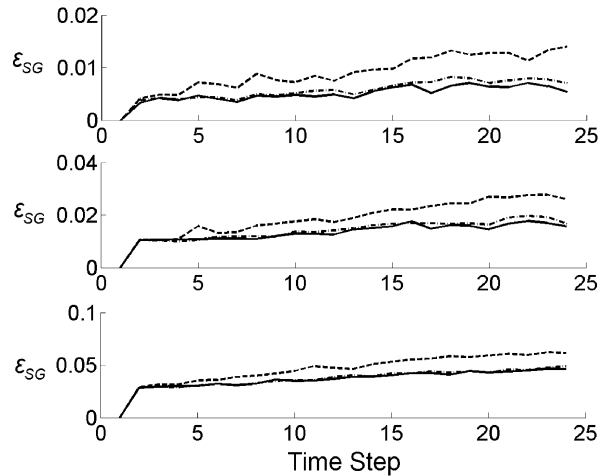


Fig. 9. Results for subgrid particle distribution error for Flow 1. $C = 0.25$, ($\Delta t = 0.112$), 4×4 grid top: error for sampling squares of side $\pi/4$; middle: error for sampling squares of side $\pi/8$; bottom: error for sampling squares of side $\pi/16$. The errors for the RK2, perfect time advancement, and MRK2 solutions are shown respectively by the dashed, solid, and dash-dot curves.

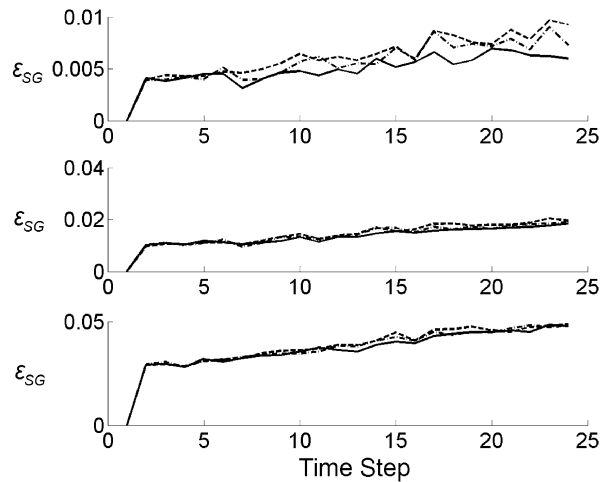


Fig. 10. Results for subgrid particle distribution error for Flow 1. $C = 0.5$, ($\Delta t = 0.112$), 8×8 grid top: error for sampling squares of side $\pi/4$; middle: error for sampling squares of side $\pi/8$; bottom: error for sampling squares of side $\pi/16$. The errors for the RK2, perfect time advancement, and MRK2 solutions are shown respectively by the dashed, solid, and dash-dot curves.

step (hence requiring two substeps per time step) an MRK2 calculation would require 2.5 times more time than an RK2 calculation would.

Fortunately, in most flows of practical interest the number of particles which require two or more substeps is generally small. For the test cases from the previous sections, the MRK2 calculations have a 70% computational overhead at $C = 1$, and a 20% computational overhead at $C = 0.25$, relative to the RK2 calculations. For smaller values of C the computational overhead is even smaller, since the ratio between the number of particles which cross a discontinuity and the number of particles which do not cross one decreases with decreasing Courant number.

4. Conclusions

We have outlined a new time-stepping scheme, based on second-order Runge–Kutta, which is particularly suited for advection of a large number of particles through a discontinuous velocity field, such as the one yielded by the parabolic edge reconstruction method (PERM). We have demonstrated that the new scheme, MRK2, preserves better than RK2 the continuity of the Lagrangian mapping between initial and final positions. We have also shown that MRK2 is second-order accurate in time, as opposed to RK2, which is first-order accurate for a discontinuous velocity field. Additionally, we have shown that MRK2 preserves better than RK2 the consistency between the subgrid particle distribution and the LES filtered density.

On the other hand, we have seen that the advantages of MRK2 over RK2 diminish when a more refined grid is used for velocity reconstruction, and that MRK2 has a computational overhead relative to RK2 (20% at $C = 0.25$) which is not negligible. Therefore, whereas RK2 is sufficiently accurate (and faster) for problems with a fine mesh, it is preferable to use MRK2 in problems where the mesh provides only marginal resolution of the velocity, which is often the case in LES/FDF methods.

Finally, it should be noted that the main principle of MRK2 – breaking up a step into two or more substeps every time a velocity discontinuity is encountered – can be applied in a straightforward manner to time advancement schemes other than RK2. The Midpoint Euler scheme, for example, can be modified to deal with velocity discontinuities similarly to the way MRK2 does. This would be preferable in a situation where velocity information is known at the middle of the time step.

Acknowledgments

This work is supported by the Air Force Office of Scientific Research, Grant FA9550-06-1-0048. This work was performed while the second author held a National Research Council Postdoctoral Research Associateship at the National Institute of Standards and Technology.

References

- [1] R. McDermott, S.B. Pope, The parabolic edge reconstruction method (PERM) for Lagrangian particle advection, *Journal of Computational Physics* 227 (2008) 5447–5491.
- [2] M. Muradoglu, S.B. Pope, D.A. Caughey, The hybrid method for the PDF equations of turbulent reactive flows: consistency conditions and correction algorithms, *Journal of Computational Physics* 172 (2001) 841–878.
- [3] P. Jenny, S.B. Pope, M. Muradoglu, D.A. Caughey, A hybrid algorithm for the joint PDF equation of turbulent reactive flows, *Journal of Computational Physics* 166 (2001) 218–252.
- [4] Y.Z. Zhang, D.C. Haworth, A general mass consistency algorithm for hybrid particle/finite-volume PDF methods, *Journal of Computational Physics* 194 (2004) 156–193.
- [5] S.B. Pope, *Turbulent Flows*, Cambridge University Press, Cambridge, UK, 2000.
- [6] E.N. Lorenz, Deterministic nonperiodic flow, *Journal of the Atmospheric Sciences* 20 (1963) 130–141.