Technique for backward particle tracking in a flow field

A. Nahum and A. Seifert
School of Mechanical Engineering, Faculty of Engineering, Tel Aviv University, P.O. Box 39040, Tel Aviv 69978, Israel
(Received 19 December 2005; revised manuscript received 3 February 2006; published 7 July 2006)

A numerical method to determine the history of particle paths is presented and its application for mixing quantification is illustrated. When more than one source exists in a flow field, the current technique can reveal the particle's identity found in any time and place in the field, by backward tracking its origin. Since the particle position at a preceding time is not known, the velocity vector is implicit. To resolve this uncertainty, a three-stage iterative procedure is developed and implemented. The current particle velocity is multiplied by the time increment and the product is used to estimate the previous time increment particle position. Two velocity matrices are generated on a mesh around the estimated position. The first matrix is the Eulerian velocity field interpolated on the mesh. The second matrix contains velocity vectors that point to the current particle position. A correlation matrix is calculated from the two velocity matrices in order to resolve the actual particle position in the previous time increment. Determination of the time increments' size is performed by checking the maximum of the correlation matrix. The new algorithm was validated using a numerical solution of the confined twin-jet flow at low Reynolds number. This flow performs a Hopf bifurcation at a Reynolds number of about 30, therefore chaotic trajectories might exist in the flow. First, the convergence of the backward particle path to the streamlines in steady flow was demonstrated. Convergence of the particle paths for various time increments was achieved also at the unsteady two-dimensional confined twin-jet flow. When the flow has more than one source, the proposed tracking method can be applied to generate complete and ordered particle source images in any desired position and time in the flow field. Maps of particle sources are used to visualize the flow patterns and the stretched interface between the two fluid sources. Those maps are demonstrated as a powerful tool for mixing quantification and can be implemented also for pollution source detection and many other fluid dynamics applications.

DOI: 10.1103/PhysRevE.74.016701 PACS number(s): 47.11.−j, 47.52.+j, 47.54.−r

I. INTRODUCTION

The dynamics of a particle in a flow field is of fundamental importance to gain better understanding of flow and transport phenomenon. Since the Eulerian frame of reference cannot provide direct answers for the evolution of a passive scalar in the flow field, particle tracking techniques are implemented in many applications of fluid dynamics. For instance, the investigation of particle path evolution in turbulent flow [1] is vital in the understanding of the physical mechanism that generates stretching and folding of material lines.

In recent years, particle tracking techniques have been vastly utilized for geophysical flows. The applications are widespread, from following the advection of pollutants in the atmosphere [2] to the investigation of the interaction between boundary currents and adjacent circulation regions in the oceans [3]. Another application that will be reviewed in this paper is related to the mixing between two or more fluids at low Reynolds number flows. This application is related to the emerging field of Lab-on-a-chip, where fast and efficient mixing in the microscale is essential for improved device performance [4].

Since mixing mechanisms at the microscale are complicated by the absence of turbulence and fabrication issues, while being of practical importance, the field of microfluidics is of great current interest (see, e.g., the recent review published by Squires and Quake [5] and the former publications [6,7] to name a few). In a typical micromixer, the flow must form chaotic patterns in order that two or more fluids would be mixed efficiently [8,9]. Therefore, the flow in those microchannels could be complex and disordered, although the Reynolds numbers could be extremely low.

Lagrangian particles' dynamics are essential since the Eulerian flow field cannot provide details of the fluid interface; Indeed, most of the published studies in the field of micromixing utilize particle tracking algorithms. Okkels and Tabeling [10] proposed to follow the evolution of the interfacial line between two fluids and to calculate from it a folding quantity. This quantity, which is a measure of the fluid interface length, is used as a basis for mapping regions of enhanced mixing in the parameter space. In order to follow the interfacial line, high-density particle tracking should have been implemented in their algorithm. Although this approach marks the boundaries of the two fluids, it has an intrinsic difficulty due to the high sensitivity to numerical errors. In addition, when several flow sources are concerned, this approach can be time consuming. In another study [11], the authors calculated Poincaré maps and Lyapunov exponent maps in the entire flow field and related it to stirring. In both studies, mixing efficiency was indirectly related to various chaotic flow characteristics. Therefore, those methods cannot be directly compared with microflow visualization experiments [12].

A direct measure of mixing can be obtained by solving the mass transport equation with the flow equations [13]. However, this approach is time consuming, simply because the mesh should be extremely dense to capture the irregular evolving fluid interface. To overcome this problem, following tagged particles from both fluid sources can provide sufficient indication for mixing. This approach was used by

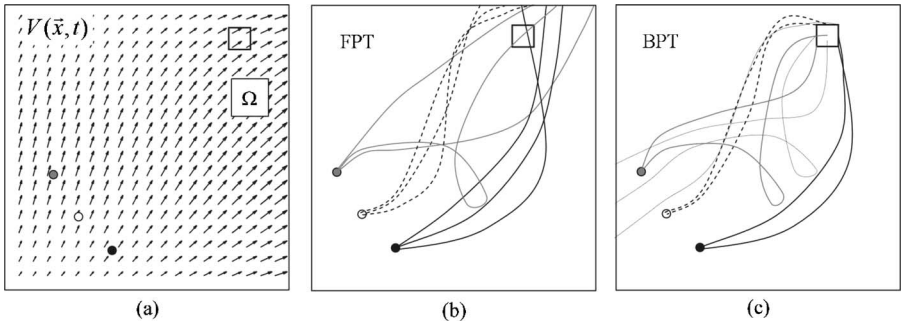

FIG. 1. An example case for the advantage of the BPT vs the FPT. (a) A general flow with three sources of particles and a region of interest Ω. (b) A conceptual experiment, describing a realistic result of using (b) FPT and (c) BPT to determine the particle distribution in Ω.

Kang *et al.* [14], who obtained the map of particle locations, where every inlet fluid was marked by a different color. The method was implemented on the Herringbone grooved mixer, which was presented earlier by Stroock *et al.* [15]. Although particle location images provide a direct indication for mixing, the method has drawbacks. In order to generate high-resolution images at far downstream locations, a huge number of particles should be tracked, and thus the computational effort is challenging. This is because not all the particles from the inlet are tracked at the desired section, even for relatively long simulation times. Another drawback originates from the fact that there is no guarantee that all the space of interest will be occupied by particles. The reason is the tendency of the ordered particles at the inlet to become disordered downstream. This fact makes the calculation of particle statistics difficult to compare between different cases.

It is therefore concluded that no currently published method is sufficiently efficient to calculate ordered particle distribution images, which are required for direct mixing quantification. The objective of this paper, therefore, is to develop a backward particle tracking algorithm to achieve this goal.

To illustrate the benefit of backward particle tracking (BPT) as compared to forward particle tracking (FPT), consider the problem of tracking particles from multiple sources (say, smoke from three chimneys) in a known two-dimensional, time-dependent flow field $\vec{V}(\vec{x},t)$ [as conceptually shown in Fig. 1(a)]. The flow field could be laminar or turbulent and could be obtained numerically, experimentally, or analytically. Our goal is to find the particle's distribution in a selected region of interest Ω (say, a city) at a selected time, t_{final}. An example of implementing the FPT technique for nine particles emanating from three sources is presented in Fig. 1(b). As the particle's paths might be chaotic, it is apparent that most of those particles will not provide information on the particle's distribution in Ω at time t_{final}. Different selections of particle starting times from the *known* sources will not change the result significantly. This is due to the inability to predict *a priori* which particle would be found in Ω at t_{final}. In addition, particles that incidentally find their way to Ω will not be distributed evenly and might cross Ω at different times. The only way to overcome the random distribution problem of the FPT is to track a vast number of particles and anticipate that some of them will end up at Ω in time t_{final}. It is concluded that the FPT technique would obtain inferior results in determining the particle's source distribution at Ω. These deficiencies could be solved by using a BPT technique.

By using a BPT technique, we can select ordered points in Ω at time t_{final} and find their source, as presented in Fig. 1(c). Those points could be selected to be ordered in any desirable manner as they can represent points of interest in a city near a contamination source. After the source of the particle is found, it can be tagged (or colored) and the particle distribution in Ω can be determined. The benefit of using BPT over FPT is apparent. First, it allows to track only those particles that would be located in the points of interest in Ω at time t_{final}. In this case, the required calculation effort to be made would be significantly reduced with respect to FPT. In addition, the resolution of the resultant particle's distribution can be selected by the user, therefore a complete and ordered particle image could be obtained in any required resolution. This ability is important for mixing application, since the resulting particle distribution images can be directly compared with images that were acquired by a camera in a colored mixing experiment.

In the following sections, the currently known FPT method will be reviewed and utilized as a reference to compare and test the characteristics of the BPT method.

II. FORWARD PARTICLE TRACKING ALGORITHM

Algorithms for FPT in a flow field are frequently used by researchers in fluid dynamics, as reviewed above. The tracking procedure can be implemented only if the Eulerian velocity field is known *a priori* in both time and space. The Eulerian velocity field can be obtained by various techniques, for instance flow field measurements such as time-resolved PIV (particle image velocimetry), numerical simulations, or from an analytic solution of the flow equations. In any case, the basic approach to track particles in the flow is to select the initial time and position in the flow and follow the particle path by solving the state equation (2.1a) using the initial condition (2.1b),

$$\dot{\vec{x}}(t) = \vec{V}(\vec{x}(t),t), \tag{2.1a}$$

$$\vec{x}(t_0) = \vec{x}_0, \tag{2.1b}$$

where $\vec{x}(t)$ is the particle position vector as a function of time, t, in three dimensions and $\vec{V}(\vec{x}(t),t)$ is the Eulerian velocity field. The solution of Eq. (2.1a) can be obtained by integration in time to determine the position of the particle at time t_e, in the form:

$$\vec{x}(t_e) = \int_{t_0}^{t_e} \vec{V}(\vec{x}(t),t)dt + \vec{x}(t_0), \tag{2.2}$$

where t_0, t_e are the initial (previous) and current times, respectively. Note that by using this state equation, it is implied that the particle is a passive scalar in the flow, i.e., this equation follows the evolution of mathematical points in the flow and neglects the particle inertia and diffusion effects.

Equation (2.2) can be solved numerically using various numerical schemes. The explicit Euler method of integration is the simplest integration technique, and will lead to relatively large integration errors; however, it is currently selected in order to simplify the comparison between the forward and backward particle tracking techniques. (Other methods, such as the fourth-order Runge-Kutta scheme, should be used in order to reduce the integration error.) Implementing the explicit Euler method to Eq. (2.2) yields the position of the particle at the next time step, t_{k+1}, in the form:

$$\vec{x}_{k+1} = \vec{x}_k + \vec{V}(\vec{x}_k,t_k)dt_k, \tag{2.3}$$

where the index k represents the current known time step, t_k. After discretization, the tracking algorithm is performed in three stages. The first stage is to select the time increment, dt_k, of the integration.

Since both forward and backward particle tracking techniques should work at any flow fields, a comment should be made regarding the selection of the time increment in light of the flow nature—laminar or turbulent. For any flow, the time increment (dt_k) should be small enough, such that the Eulerian flow field does not significantly change between the current and next time. In addition, it should be monitored that with the dt_k, a particle does not progress in space through significant velocity gradients. For laminar flow fields, the characteristic time and spatial scales are physically determined by the solution of the Eulerian flow field. For example, in a laminar flow behind bluff body, the velocity field has a dominant frequency and mode shape that characterizes the flow. This is not the case when turbulent flow is considered.

Turbulent flow fields are characterized by multiple scales in time and space. Since turbulent flow fields could not be solved analytically, the numerical or experimental grid, which is used to resolve the flow, acts as a low-pass filter to the high frequencies (small scale structures). Therefore, the time increment of the tracking algorithm for turbulent flows should be selected to capture the finest resolved scales. When using an adequate time increment, the particle dynamics will not be directly affected by the large-scale structures, and from a local point of view the flow can be regarded as laminar. Therefore, for both laminar and turbulent flows, the selection of the integration time step is crucial to obtain realistic particle paths and there is no mathematical difference between laminar and turbulent flows. This is true for both forward particle tracking and for backward particle tracking techniques.

In the second stage, the velocity of the particle is interpolated in space and time from the Eulerian flow field. The interpolation method can be, for instance, linear, cubic, or any other method. The third and final step is to calculate the displacement of the particle, $d\vec{x}$, from the interpolated velocity and the selected dt_k by substitution into Eq. (2.3). The algorithm then returns with the new particle location, $\vec{x}_{k+1}$, to the first stage of the algorithm for the next time step calculation.

It should be noted that many publications and current studies are focused on finding solutions to reduce the numerical errors, which are associated with particle tracking [16]. The first source of error is the solution of the Eulerian flow field. If solved numerically, it transfers the solution errors to the particle paths. The second source of error is the interpolation method in time and space, which is used to extract the particle velocity from the discrete Eulerian velocity field. This error is eliminated when the flow field is known analytically. The third source of error is related to the numerical integration scheme, which is used to integrate Eq. (2.2). Different numerical schemes will lead to different particle paths, even in flows that have analytical steady solutions [17]. The last error results from the selection of the size of the time increment, as discussed previously. The weak point of all tracking algorithms is the accumulation of all the above-mentioned errors as the particles progress in the flow field.

The currently developed BPT algorithm is based on the same numerical procedures as the current application of the FPT methods. However, it requires an additional stage with respect to the FPT, therefore one cannot expect to retrieve more reliable results in terms of minimizing integration error with the BPT algorithm. The following section presents the BPT technique, which is implemented using the simplest numerical scheme, the explicit Euler integration method. As stressed before, this integration method is not optimally suited for this application, but it will serve as a basis to simply explain the BPT technique and compare it to the FPT approach. Implementation of the advanced numerical integration scheme, the fourth-order Runge-Kutta, is demonstrated in the Appendix.

III. BACKWARD PARTICLE TRACKING (BPT) ALGORITHM

Consider a particle that is located at a given position, $\vec{x}_i$, and time, t_i, in a known Eulerian velocity field, $\vec{V}$ (where i represents the current time step). It is required to follow the trajectory history of this particle throughout the flow field. The equation to be solved is again Eq. (2.2). After discretization using the explicit Euler integration method, the result is identical to that of Eq. (2.3). However, in this case we seek the value of the first term on the left-hand side of Eq. (2.3). Rearranging Eq. (2.3), we obtain the following equation from which we can calculate the current particle location:

$$\vec{x}_{i+1} = \vec{x}_i - \vec{V}(\vec{x}_{i+1},t_{i+1})dt_i, \tag{3.1}$$

where the index i is substituted instead of k and it represents the currently known time step. Index i is progressing forward as time is decreased, i.e., at the starting point of the integra-

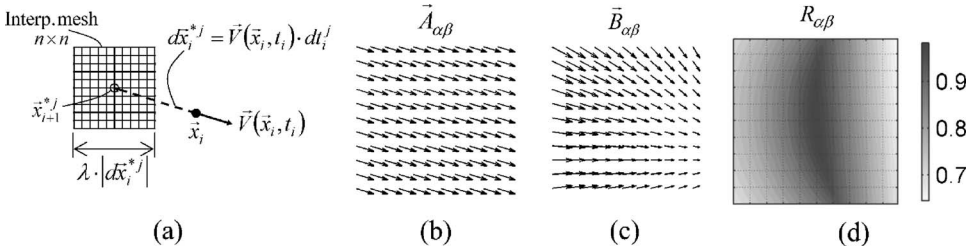

FIG. 2. An example of (a) the interpolation mesh and nomenclature, (b), (c) the resulting matrices $\vec{A}_{\alpha\beta}$ and $\vec{B}_{\alpha\beta}$ with the circle indicating the location of the max correlation of (d) the correlation matrix $R_{\alpha\beta}$.

tion, $i=1$ is the final time. Equation (3.1) cannot be solved explicitly, since knowledge of the particle velocity at the previous step, $i+1$, is required. To solve Eq. (3.1), we calculate the particle position $\vec{x}_{i+1}$ and the velocity vector $\vec{V}(\vec{x}_{i+1}, t_{i+1})$ using an iterative method that contains three stages: estimation, correlation, and validation.

At the estimation stage, a time increment dt_i^j is selected to be small enough according to the criteria described in Sec. II. The superscript j represents the current iteration step in the process. Then, by using the current velocity, $\vec{V}(\vec{x}_i, t_i)$, an estimation of the particle location is obtained in the form

$$\vec{x}_{i+1}^{*j} = \vec{x}_i - \vec{V}(\vec{x}_i, t_i) dt_i^j, \tag{3.2}$$

where the asterisk represents the estimated previous particle position. The estimated location, $\vec{x}_{i+1}^{*j}$, narrows the space for the search of the actual particle position to a specific region in the flow.

At the correlation stage, an interpolation mesh Ψ is constructed around the estimated location, $\vec{x}_{i+1}^{*j}$, as illustrated in Fig. 2(a) (for clarity, only two dimensions are presented). The meshed volume is a cube centered on $\vec{x}_{i+1}^{*j}$, with a total edge size that is proportional to the distance between $\vec{x}_{i+1}^{*j} = \vec{V}(\vec{x}_i, t_i) dt_i^j$ and $\vec{x}_i$, with a proportionality factor λ. The matrix Ψ, which is selected with initial dimensions $n \times n \times n$, contains n^3 grid points with the position coordinates $\vec{\psi}_{\alpha\beta\gamma}^j$ (α, β, γ are indexes of the array in three dimensions). On Ψ, two matrices containing $n \times n \times n$ velocity vectors are generated. The first matrix, $\vec{A}_{\alpha\beta\gamma}^j$, is the interpolated Eulerian velocity vector matrix on the mesh. Each one of the vectors in the matrix $\vec{A}_{\alpha\beta\gamma}^j$ would lead a particle from its location on the interpolation mesh to the vicinity of $\vec{x}_i$. An example of the matrix $\vec{A}_{\alpha\beta\gamma}^j$ is presented in Fig. 2(b). The second matrix, $\vec{B}_{\alpha\beta\gamma}^j$, is defined by the following equation:

$$\vec{B}_{\alpha\beta\gamma}^j = \frac{\vec{\psi}_{\alpha\beta\gamma}^j - \vec{x}_i}{dt_i^j}. \tag{3.3}$$

$\vec{B}_{\alpha\beta\gamma}^j$ contains artificial velocity vectors, all of which are calculated such that it is directed exactly towards $\vec{x}_i$. Moreover, according to Eq. (3.3), the magnitude of the vectors in $\vec{B}_{\alpha\beta\gamma}^j$ are such that they will lead a particle from $\vec{\psi}_{\alpha\beta\gamma}^j$ to $\vec{x}_i$ with a convection time of dt_i^j exactly [shown in Fig. 2(c)].

The idea behind the generation of the two above-mentioned matrices is to compare the two vector matrices in order to find a real velocity vector in $\vec{A}_{\alpha\beta\gamma}^j$ that has the highest correlation to an artificially calculated velocity vector in $\vec{B}_{\alpha\beta\gamma}^j$. This is done by correlating the two matrices using the following equation (the summation convention is not valid in this case):

$$R_{\alpha\beta\gamma}^j = \frac{\vec{A}_{\alpha\beta\gamma}^j \cdot \vec{B}_{\alpha\beta\gamma}^j}{\max\{|\vec{A}_{\alpha\beta\gamma}^j|, |\vec{B}_{\alpha\beta\gamma}^j|\}^2}. \tag{3.4}$$

The values contained in matrix $R_{\alpha\beta\gamma}^j$ are between -1 and $+1$, where -1 represents opposite vectors with the same magnitude between corresponding cells in $\vec{A}_{\alpha\beta\gamma}^j$ and $\vec{B}_{\alpha\beta\gamma}^j$, and $+1$ represents identical vectors. The indices α, β, γ, which yield the highest correlation, $\max(R_{\alpha\beta\gamma}^j)$, define the requested position vector $\vec{x}_{i+1}^{*j}$ for iteration j. An example for the correlation matrix, $R_{\alpha\beta\gamma}^j$, is presented in Fig. 2(d). When the $\max(R_{\alpha\beta\gamma}^j)$ is exactly $+1$, the BPT is converged to the FPT, and both methods should track the same paths, when using the same time increment. Otherwise the BPT will contain an additional error, which can be minimized by selecting a minimal required value, R_C, for $\max(R_{\alpha\beta\gamma}^j)$. If the value $\max(R_{\alpha\beta\gamma}^j)$ is higher than R_C, the procedure is completed and the actual position at the previous time step, $i+1$, is determined as $\vec{x}_{i+1} = \vec{x}_{i+1}^j$. Otherwise, the procedure continues to the next iteration. At the next iteration, the value of the time increment is halved with respect to the previous time increment according to the formula $dt_i^{j+1} = 0.5 dt_i^j$ and the mesh dimensions are doubled in each direction in the form $2n \times 2n \times 2n$. Thus, for the next iteration, the search region is twice as close to $\vec{x}_i$, has eightfold smaller search volume, and has twice as high mesh density, hopefully to obtain higher maximum correlation. This iterative procedure is terminated when the requested correlation value R_c is achieved.

After the position of the particle, $\vec{x}_{i+1}$, is determined, the next time step is calculated with the same procedure and parameters until the requested time or position of the particle is attained. When the goal of the particle tracking is mixing analysis, the particle source can be detected and an attribute can be assigned to this particle (for example, a specific color).

To validate the BPT algorithm, it was implemented in MATLAB (Mathworks, Inc.). Only the BPT iterative stage was tested for convergence, since it was the only additional stage with respect to the FPT algorithm. This was performed by running three different tests. The goal of the first test (test case 1) is to find the minimal required time increment such

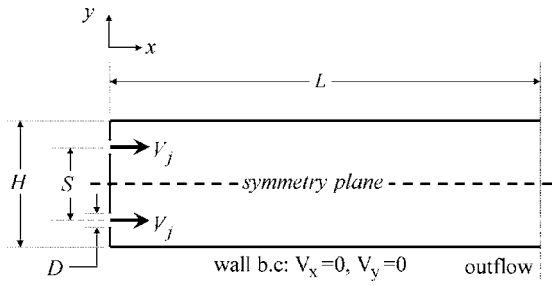

FIG. 3. The plain confined twin jet geometry and its boundary conditions.

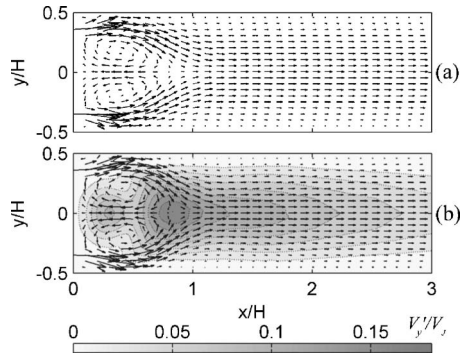

FIG. 4. Normalized mean velocity vectors and contour map (10 levels) of the vertical velocity standard deviation (V_y') for (a) $R_e=20$, steady flow and for (b) $R_e=30$, periodic flow (CFD generated flow field).

that the particle paths are converged to the flow streamlines in a steady flow. This test was performed without the validation stage of the interpolation algorithm. Therefore, the time increment was held fixed. The purpose of the second test (test case 2) was to confirm that the ordered particle maps obtained in an unsteady flow will converge as the time increment is decreased. The aim of the third test (test case 3) is to compare directly the BPT method with the FPT method. This is done by tracking particles backward from an ordered array at downstream location to their source and then tracking them forward from the same place where the BPT tracking ended. For any particle, the error between the BPT starting point and the FPT end point will be calculated. If the iterative stage of the algorithm is efficient, those two points should converge.

The errors caused by the numerical scheme, the interpolation method, and the Eulerian flow field are not considered in this validation. Ignoring those errors could be justified by the existence of the same errors also in the FPT and by the purpose of the current study, i.e., the comparison of the forward and backward particle tracking algorithms.

IV. ALGORITHM VALIDATION USING THE PLAIN CONFINED TWIN JET FLOW

To validate the BPT algorithm and to demonstrate its advantages, it was implemented on the plain confined twin-jet flow. This flow was selected since it has a potential for micro-mixing applications and it is sufficiently complex in order to develop chaotic particle paths. The selected Eulerian flow field was obtained by numerical solution of the plain confined twin-jet flow [18]. The geometry and the boundary conditions are shown in Fig. 3. The time-dependent, two-dimensional, incompressible Navier-Stokes equations were solved numerically using a commercial code (FEMLAB™, Comsol Inc.) with approximately 20 000 grid points. The code resolved the Eulerian velocity field $\vec{V}(\vec{x},t)$ and the pressure $P(\vec{x},t)$, which is not utilized in this work. The software code is based on the finite-element method, and the current results were validated numerically versus Soong *et al.* [18] and experimentally in Nahum *et al.* [19]. The selected geometrical parameters were $S/H=0.7$ (the ratio between jet separation S and channel width H) and $D/H=0.07$ (the ratio between the jet width D and H). With this geometry, the flow undergoes a Hopf bifurcation for the above geometry at a Reynolds number, R_e, of 30, where $R_e=V_jD/\upsilon$, V_j is the mean jet velocity, and υ is the kinematic viscosity. From nonlinear dynamics theory, particles in this flow can perform chaotic paths, since the flow is two-dimensional and time-dependent. From a Lagrangian viewpoint, the particle paths in this flow can be as complex as in a turbulent flow. This makes the confined twin-jet flow suitable as a benchmark problem to validate the BPT algorithm.

Two cases of this flow are currently considered. The first is for steady flow at the subcritical $R_e=20$, and the second is for periodic flow at the critical $R_e=30$. Examples of both flow fields are presented in Fig. 4. For $R_e=20$ [Fig. 4(a)], the flow is steady and symmetric, with a central recirculation zone, showing two counter-rotating vortices, and two symmetric recirculation zones near the upper and lower entrance corners. As the Reynolds number was increased to $R_e=30$, the flow became periodic, but remained symmetric in the average sense.

The velocity vectors [arrows in Fig. 4(b)] and the vertical velocity standard deviation (STD) are presented in Fig. 4(b) (the STD is shown as a gray level contour plot). It is shown that most of the fluctuation energy is concentrated near the jet interaction region. Those fluctuations are responsible to the transfer of momentum between the jets, and therefore to enhanced mixing. The Strouhal number, $S_t=fS/V_j$, associated with this flow is $S_t=0.07$, where f is the dominant frequency of the vertical velocity fluctuation, V_y'.

It is advantageous to demonstrate the BPT and mixing analysis with the confined twin-jet flow because no other method known to the authors could be used to construct particle images for this flow. For example, following the interface between the two fluids could not be performed since the interface between both jets is not defined. The recirculation region contains particles from both inlets, and their identities are not known. Another method of solving the flow and mass equation simultaneously requires that the recirculation region material will be defined. If not, the flow and the mass equations should be solved from an initial empty channel state, until the mean (in the time domain) mass distribution at the recirculation region(s) will be constant. This approach could require a significant effort and time to obtain the desired results. Therefore, only with the BPT could ordered particle maps of this flow be attained for the confined twin-jet flow.

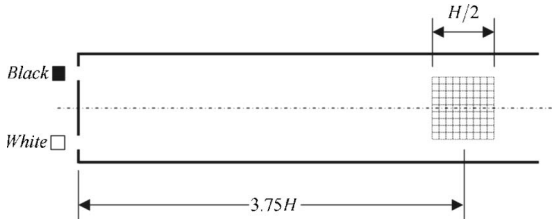

FIG. 5. The selected location of the mesh used for code validation shown with respect to the channel. Particles' sources found at upper or lower inlets are marked with black or white, respectively.

V. ALGORITHM VALIDATION

For the first test (test case 1), 100 ordered particles were tracked back to find their source in the steady flow field at $R_e=20$. The particles' initial location was on a square mesh centered on $x/H=3.75, y/H=0$, with an edge length of $0.5H$ (see Fig. 5). Particles originating at the upper or lower inlets were tagged with 0 (black) and 1 (white), respectively. The value 0.5 (gray) was assigned to particles for which the algorithm did not find their source for the entire duration of the analysis. This case occurred when a particle was trapped in the entrance vortices region [as shown in Fig. 3(a)] or crossed the entrance wall. From the tracking results, the square mesh was mapped with the three colors representing the particle sources.

Different particle maps were obtained for five different time increments. The BPT parameters were selected to be $\lambda=0.33$ and $n=100$. The time increment scaling was $t^+=tV_j/S$. The scaling factor S/V_j designated the time scale of the twin jet period at the critical R_e. Figure 6 presents the

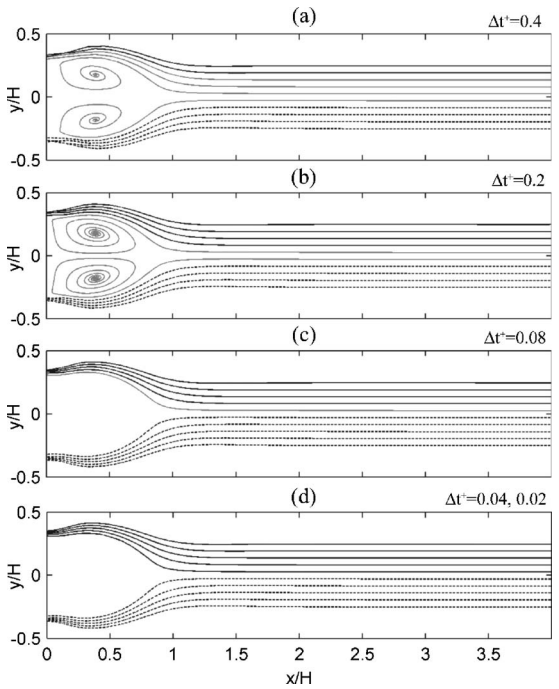

FIG. 6. Ten backward tracked particle paths starting at $x/H=4$ for different time increments, as indicated above each chart ($R_e=20$, steady flow field).

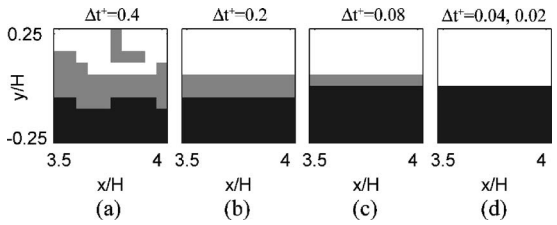

FIG. 7. A comparison between four particle maps (10 by 10 particles) as the time increment decreases from $\Delta t^+=0.4$ and 0.2 as indicated above the charts, $R_e=20$ (steady flow).

particle paths in the steady flow field, $R_e=20$, tracked backwards from a selected section at $x/H=4$. Since the flow field was steady, the particles' paths should converge to the streamlines of the flow. It is shown in Figs. 6(a) and 6(b) that for the two largest time increments, $\Delta t^+=0.4, 0.2$, two particles were trapped in the entrance vortices between the two jets, therefore they do not follow the streamlines that originate at the jet entrances. As the time increment was decreased to $\Delta t^+=0.08$ [Fig. 6(c)], none of the selected particles were trapped in the entrance vortices. However, one of the particles missed the exact location of the jet entrance. When the time increment was decreased further to $\Delta t^+=0.04$ or less, all particle origins were successfully determined.

The particle maps for $R_e=20$ flow field are presented in Fig. 7. As Δt^+ was decreased, the number of particles whose sources were found was increased. To obtain a quantitative measure of the accuracy of the particle paths calculated by the BTP algorithm, a comparison between the particle paths and the streamlines was performed.

The mean absolute distance, $\langle d\rangle$, between the particle path and the associated streamline was calculated along the path of the particle for different time increments, Δt^+. This was performed for all the particles in the map and the result was normalized to the streamline length, l_p. Figure 8 presents the average of the normalized mean distance between the particle path and the associated streamline, $\langle\langle d\rangle/l_p\rangle$, as a function of the time increment, Δt^+, with the maximal and minimal values shown as error bars. It was shown that as Δt^+ was

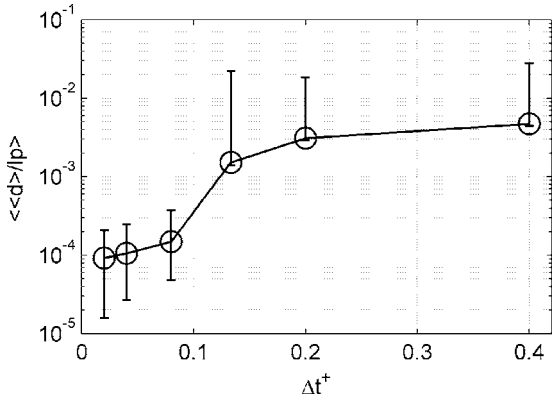

FIG. 8. The average of the normalized mean absolute distance, $\langle d\rangle$, between the particle path and the streamline vs step size (averaged over 100 particles). Error bar indicates the minimal and maximal values of $\langle\langle d\rangle/l_p\rangle$, with l_p being the streamline length.

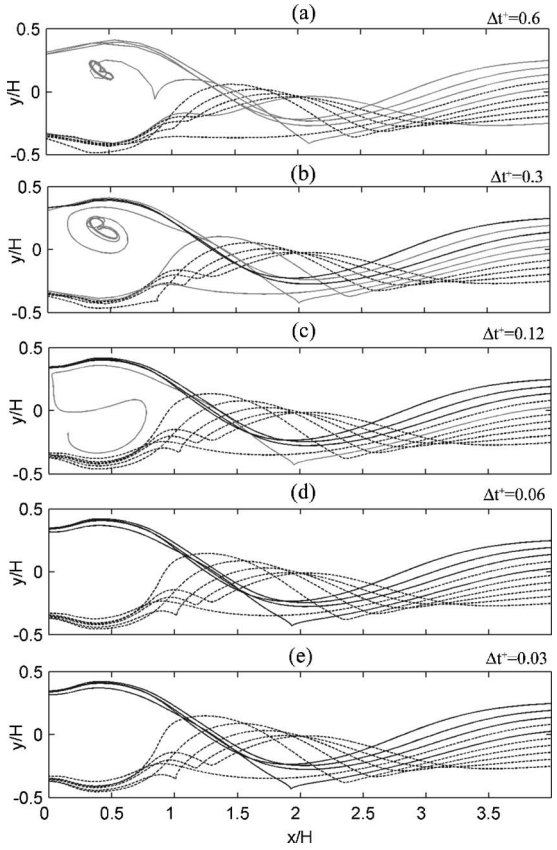

FIG. 9. Ten backward tracked particle paths starting from $x/H=4$ for different time increments as indicated above charts ($R_e=30$, periodic flow).

decreased, the particle path distance from the streamline was decreased on the average, i.e., the particle paths converged to the streamlines. It was also observed that the variation of the mean distance was significantly reduced with the increase in Δt^+, which is a result of the detection of all particle sources as Δt^+ was decreased. It is clearly shown that $\Delta t^+=0.08$ is the optimal time increment for the current test conditions.

For test case 2, the particle paths in the flow field at $R_e=30$ were tracked using the same geometrical parameters as for $R_e=20$. Since at $R_e=30$ the flow field was periodic, the particle paths were not expected to follow the streamlines of the flow. This implies that no-source particles could have been found in this flow, since the selected total analysis time could have been shorter than the time it takes a certain particle to be convected through the entire channel. In Fig. 9, the particle paths for 10 evenly spaced locations across the channel $(x/H=4)$ are presented. It was shown that as Δt^+ was decreased, the number of no-source particles was decreased, and the particle paths for $\Delta t^+=0.06$ closely resemble the case of $\Delta t^+=0.03$. Although in this cross section and for the Δt^+ used all particle sources were found, it might not always be possible to detect all particle sources since the trajectories might be chaotic.

In Fig. 10, six particle maps are shown for the $R_e=30$ flow field. It was observed that as the time increment was decreased, the particle maps preserve the same structure. To quantify the convergence, the particle maps were compared

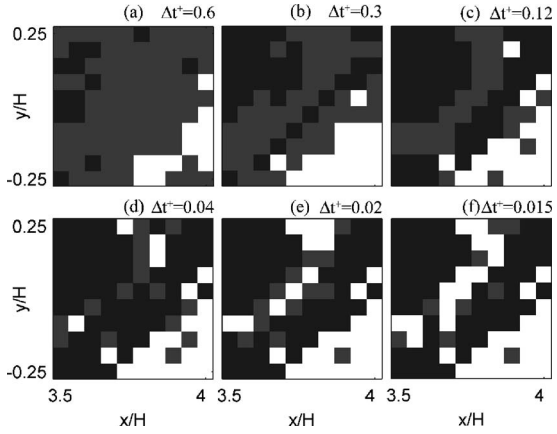

FIG. 10. A comparison between six particle maps (10 by 10 particles) as time increment decreases, as indicated above charts, $R_e=30$ (periodic flow).

to the map with the smallest time increment, i.e., $\Delta t^+=0.015$. Figure 11 presents the percentage of particle mismatch as a function of the time increment. It was observed that more than 90% of the particles were obtained correctly for $\Delta t^+=0.02$ with respect to the $\Delta t^+=0.015$ map. It should be noted that one could not expect to achieve perfect matching between the two maps where chaotic trajectories exist. From the above results it was decided to select $\Delta t^+=0.02$ as the maximal time increment allowed for the mixing analysis.

In test case 3, 100 particles are tracked backward and forward in the flow field to find the position error. The expected error originates from the additional iterative stage in the BPT algorithm. In order to compare the two algorithms without bias, the same time increments that were used in the BPT stage are used for the FPT stage. If the particles will return to the same point, the BPT generated particle path will follow exactly the FPT results. It should be noted that although we expect the BPT and FPT particle paths to converge, there is no way to utilize the FPT method to retrieve ordered particle maps. The FPT algorithm can be used in this case only after the BPT was performed. This is because before the BPT is implemented, one cannot select 100 points for the FPT, which their paths will end up in an ordered manner at the required locations.

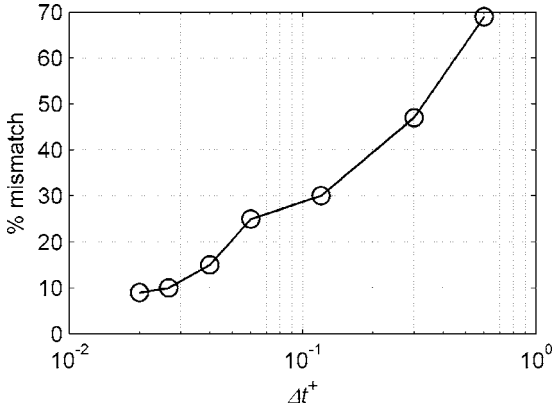

FIG. 11. Percentage of mismatch particles found in various particle maps with respect to the $\Delta t^+=0.015$ map ($R_e=30$, periodic flow).

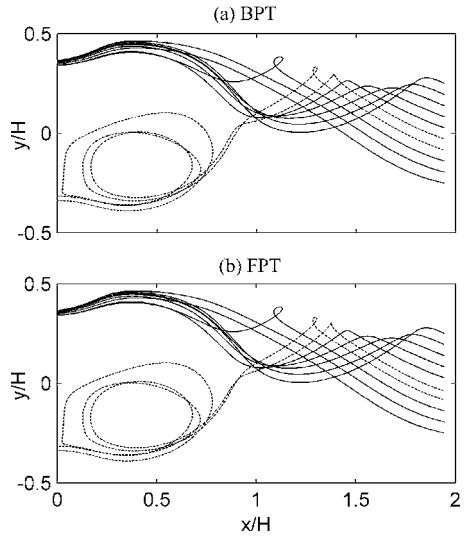

FIG. 12. Validation of the BPT algorithm with respect to the FPT algorithm by tracking 10 evenly spaced particles from $x/H=1.9$ to their source and back ($R_e=30$, periodic flow).

The threshold for the maximum correlation, R_c, was selected to be 0.99, and the interpolation mesh size, n, was selected to be 100. The simulations were performed for different sizes of Ψ, with $\lambda=0.05$ to 0.33, to determine the effect of λ on the accuracy of the BPT. The confined twin-jets flow at $R_e=30$ was selected for test case 3. An array of 10 by 10 particles with edge size of $0.5H$, centered at $x/H=1.75$, was used for this test.

A selected example of 10 particle paths is shown in Fig. 12 for $\lambda=0.33$. It can be qualitatively seen that the 10 selected particle paths from the BPT simulations are identical to the FPT simulation paths. A quantitative analysis was performed by calculating the distance ε of the final end point of the FPT with respect to the starting point of the BPT. This distance should be zero if the error of the iterative stage of the BPT is zero. The solid line in Fig. 13 shows the normalized mean distance $\langle\varepsilon\rangle/H$ of the particles versus the selected λ, and the error bars indicate the maximal and minimal obtained values. The number above the error bars indicates the number of particles with $\langle\varepsilon\rangle/H$ of more than 0.05. Those particles are found to be also the same particles for which the

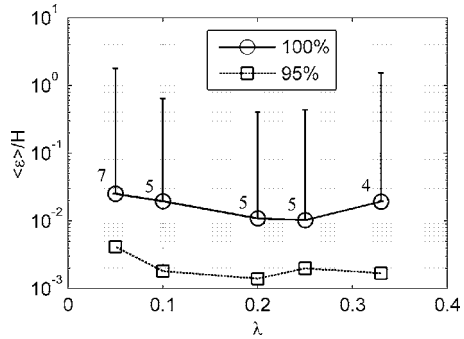

FIG. 13. The normalized mean distance $\langle\varepsilon\rangle/H$ between the starting position of the BPT to the final position of the FPT of the same particle ($R_e=30$, periodic flow).

BPT algorithm did not find their sources. It is shown that for $\lambda=0.2-0.25$, the error is minimized. Although five particles were diverged significantly from the BPT path, the value of the normalized mean error is less than 0.01. The dashed line represents the error where the 5% of the most divergent particles are excluded from the calculation. It can be seen that in this case the mean error is reduced to 0.001. Therefore, 95% of the particles were found, with good accuracy, to be on the same path for the BPT and the FPT algorithms.

VI. MIXING ANALYSIS

As indicated before, mixing quantification is vital for the design, fabrication, and characterization of efficient micromixers. The utilization of the BPT algorithm for the confined twin-jet micromixer was performed for the flow case at $R_e=30$. Maps of 50 by 50 particles were obtained across the flow with edge sizes of $0.8H$ in each direction. The selected edge size ($0.8H$) was sufficient for mixing analysis since 94.4% of the mean flow passes through this region. In the current tracking analysis, the BPT validation stage (the third stage in the algorithm) was implemented, thus the time increment was not kept constant. The maximum value of the correlation matrix was checked, and if it was found to be smaller than $R_C=0.99$, the next iteration was performed with half the time increment.

From the analysis of test case 2, the average number of steps required for a particle to reach its origin was calculated. In the mixing analysis, the algorithm was terminated when the number of steps reached ten times this value. The average simulation time for all particles was $t^+=25$. This was performed in order to increase the probability to find the particle source even if it was trapped in the vortices region. In the case in which the algorithm did not find the particle source, the particle was tagged with the value 0.5 (gray). Therefore, this tag was considered as a perfectly mixed fluid cell. The rationale behind this assumption was that such a no-source particle resided in the channel, on average ten times longer than other particles, therefore there is a higher probability of mixing.

Figure 14 shows the particle maps of size $0.8H$ by $0.8H$ at various downstream locations. It can be seen that the interface between the two colors was elongated with respect to the expected straight line when the flow is steady at the lower Re. Mixing is also observed to be qualitatively improved as the fluids move further downstream.

The particle maps were utilized to quantify mixing as a function of scale by the mixing variance coefficient (MVC) method [20]. The particle maps were divided to s by s cells, where s is smaller than the number of particles on the map edge. For every cell, the mean particle value, P_i, was calculated, where i represents the ith cell. Perfect mixing in a cell was achieved when $P_i=0.5$, and no mixing was defined when $P_i=0$ or $P_i=1$. The MVC for scale $1/s$ was defined as:

$$\eta_{\mathrm{mix}}(s)=\frac{1}{s^2}\sum_{i=s^2}(P_i-0.5)^2. \tag{6.1}$$

From this definition, the MVC was bounded between zero, indicating perfect mixing, and 0.25, indicating no mixing.

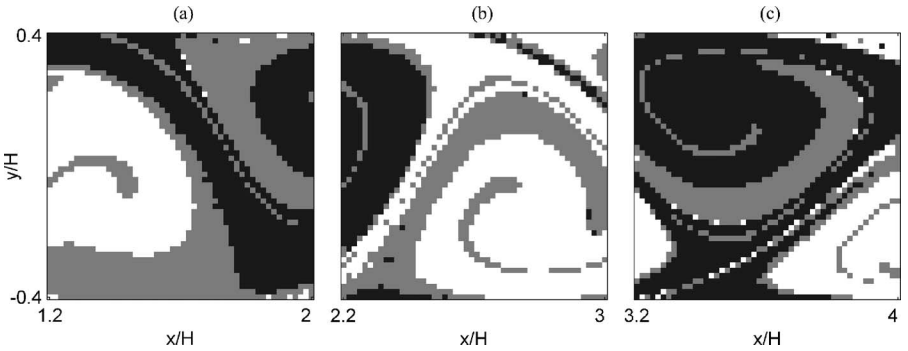

FIG. 14. Particle maps containing 2500 particles which were obtained at different streamwise positions in the flow (periodic flow, $R_e=30$), as indicated by the ordinate. Black or white particles are presenting particles that originated from the upper or lower jets, respectively. Gray indicates particles whose source was not identified. In most cases, they are trapped at the vortices region between the entrance jets.

Figure 15 presents the MVC as a function of the scale, $1/s$, for the maps presented in Fig. 14. As expected, it is shown that for all locations, as the scale was decreased, the MVC was increased. This is understood when considering the size of the structures in the particle maps. In addition, mixing was improved for all scales with respect to the reference map results. It is shown that the MVC was not significantly changed as a function of x/H, therefore the mixer length can be reduced while keeping the same mixing performance. In summary, the combination of the particle maps with the MVC method illustrates the benefits of using the backward particle tracking for mixing analysis.

VII. CONCLUSIONS

A technique for backward particle tracking (BPT) in a flow field was developed and validated. The ability of the BPT technique to determine the complete particle distribution in any place in the flow was demonstrated. Furthermore, the BPT technique can identify the location of a contamination found anywhere in the flow field, using the flow field evolution. An iterative numerical method to find the location and velocity of a particle at previous time steps was implemented. Using the confined twin-jet flow field at $R_e=20$ (steady flow), it was found that the backward tracking algorithm has converged to the streamlines when a sufficiently small time increment was used. The particle paths were also converged with respect to the smallest time increment, when the flow was periodic at $R_e=30$. Mixing analysis was performed using the resultant particles' maps of the BPT algorithm. Particle maps at selected locations in the flow field were successfully obtained. Those maps illustrated that, when $R_e=30$, the patterns of the particles become irregular, with elongated interface with respect to the steady flow. The above findings were quantified by a simple mixing efficiency criterion which could be used to design efficient mixing devices. It is concluded that the BPT can be implemented to observe the evolution of fluid interfaces, and therefore can be compared to experimental results in a more direct manner than forward particle tracking algorithms.

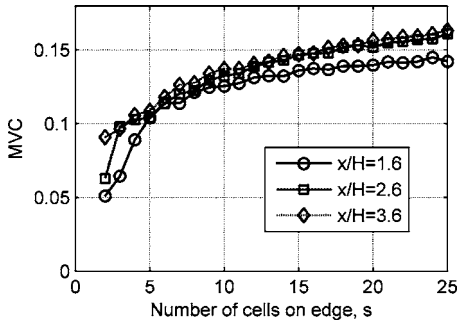

FIG. 15. Mixing variance coefficient (MVC) as a function of the number of cells for three different x/H locations in the channel. Perfect mixing is achieved when the MVC is 0.

APPENDIX: ADVANCED NUMERICAL SCHEME

A fourth-order Runge-Kutta (RK4) numerical scheme can replace the direct Euler integration method. The advantage of using RK4 with respect to the direct Euler integration method was investigated thoroughly in many studies and is not within the scope of the current paper. The BPT technique does not alter the characteristics of those methods.

Consider the discretization of Eq. (2.2) using the RK4 scheme (the index k is increasing in time),

$$\vec{x}_{k+1} = \vec{x}_k + \frac{1}{6}(\vec{r}_1 + 2\vec{r}_2 + 2\vec{r}_3 + \vec{r}_4), \tag{A1}$$

where

$$\vec{r}_1 = \vec{V}(\bar{x}_k, t_k)dt_{k+1},$$

$$\vec{r}_2 = \vec{V}(\bar{x}_k + 0.5\vec{r}_1, t_k + 0.5dt_{k+1})dt_{k+1},$$

$$\vec{r}_3 = \vec{V}(\bar{x}_k + 0.5\vec{r}_2, t_k + 0.5dt_{k+1})dt_{k+1},$$

$$\vec{r}_4 = \vec{V}(\bar{x}_k + \vec{r}_3, t_k + dt_{k+1})dt_{k+1}. \tag{A2}$$

In our case, we are interested in finding the position at the previous time increment, $\vec{x}_k$, from the information from the next time increment. Rearranging the equation and substituting an index, which is decreasing in time, i, instead of k, we yield

$$\vec{x}_{i+1} = \vec{x}_i - \frac{1}{6}(\vec{r}_1 + 2\vec{r}_2 + 2\vec{r}_3 + \vec{r}_4), \tag{A3}$$

$$\vec{r}_1 = \vec{V}(\bar{x}_{i+1}, t_{i+1})dt_i,$$

$$\vec{r}_2 = \vec{V}(\bar{x}_{i+1} + 0.5\vec{r}_1, t_{i+1} + 0.5dt_i)dt_i,$$

$$\vec{r}_3 = \vec{V}(\bar{x}_{i+1} + 0.5\vec{r}_2, t_{i+1} + 0.5dt_i)dt_i,$$

$$\vec{r}_4 = \vec{V}(\bar{x}_{i+1} + \vec{r}_3, t_{i+1} + dt_i)dt_i. \tag{A4}$$

The calculation of the vectors $\vec{r}_1$, $\vec{r}_2$, $\vec{r}_3$, and $\vec{r}_4$ cannot be performed explicitly. As described in Sec. III, an implicit iterative stage should be utilized to find $\vec{x}_{i+1}$. $\vec{r}_1$ is calculated on the interpolation mesh, around the estimated particle position. Then a series of substitutions in Eq. (A4) is performed to retrieve $\vec{r}_2, \vec{r}_3, \vec{r}_4$. The obtained values are substituted in Eq. (A3) to find $\vec{x}_{i+1}$.

[1] P. K. Yeung and S. B. Pope, J. Comput. Phys. **79**, 373 (1988).
[2] M. Gomez-Gesteira, P. Montero, R. Prego, J. J. Taboada, P. Leitao, M. Ruiz-Villarreal, R. Neves, and V. Perez-Villar, Oceanologica Acta **22**, 167 (1999).
[3] S. Wiggins, Annu. Rev. Fluid Mech. **37**, 295 (2005).
[4] B. H. Weigl, R. L. Bardell, and C. R. Cabrera, Adv. Drug Delivery Rev. **55**, 349 (2003).
[5] T. M. Squires and S. R. Quake, Rev. Mod. Phys. **77**, 977 (2005).
[6] D. J. Beebe, G. A. Mensing, and G. M. Walker, Annu. Rev. Biomed. Eng. **4**, 261 (2002).
[7] H. A. Stone, A. D. Stroock, and A. Ajdari, Annu. Rev. Fluid Mech. **36**, 381 (2004).
[8] H. Aref, Phys. Fluids **14**, 1315 (2002).
[9] J. M. Ottino, Annu. Rev. Fluid Mech. **22**, 207 (1990).
[10] F. Okkels and P. Tabeling, Phys. Rev. Lett. **92**, 038301 (2004).
[11] X. Z. Niu and Y. K. Lee, J. MEMS **13**, 454 (2003).
[12] R. A. Truesdell, P. V. Vorobieff, L. A. Sklar, and A. A. Mammoli, Phys. Rev. E **67**, 066304 (2003).
[13] C. R. Kaplan, D. R. Mott, and E. S. Oran, in AIAA Aerospace Sciences Meeting & Exhibit (AIAA, Reno, NE, 2004).
[14] T. G. Kang and T. H. Kwon, J. MEMS **14**, 891 (2004).
[15] A. D. Stroock, S. K. W. Dertinger, A. Ajdari, I. Mezic, H. A. Stone, and G. M. Whitesides, Science **295**, 647 (2002).
[16] E. Hairer, BIT **40**, 726 (2000).
[17] E. Hairer, BIT **41**, 996 (2001).
[18] C. Y. Soong, P. Y. Tzeng, and C. D. Hsieh, Phys. Fluids **10**, 2910 (1998).
[19] A. Nahum and A. Seifert, Phys. Fluids **18**, 064107 (2006).
[20] F. Bottausci, I. Mezic, C. D. Meinhart, and C. Cardonne, Proc. R. Soc. London, Ser. A **362**, 1001 (2004).