Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 194 (2004) 156–193
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A general mass consistency algorithm for hybrid particle/finite-volume PDF methods

Y.Z. Zhang, D.C. Haworth *

Department of Mechanical and Nuclear Engineering, The Pennsylvania State University, 232 Research Building East, University Park, PA 16802, USA

Received 7 April 2003; received in revised form 25 August 2003; accepted 25 August 2003

Abstract

An algorithm is devised to maintain a correct spatial distribution of computational particles in hybrid particle/finite-volume (FV) probability density function (PDF) methods for chemically reacting turbulent flows. The approach is, by design, compatible with existing FV computational fluid dynamics (CFD) codes that are used to model practical engineering flows in complex geometric configurations. The algorithm is suitable for general three-dimensional incompressible or compressible, steady or time-dependent flows using structured or unstructured, stationary or deforming computational meshes. It is compatible with a variety of element shapes commonly used in research and engineering CFD codes including hexahedra, prisms and tetrahedra. Robustness, accuracy and efficiency of the approach are demonstrated via computations for several two- and three-dimensional steady and unsteady flow configurations using computational meshes that vary in element type and in mesh quality. Both composition PDF and velocity PDF methods are employed. This work broadens the accessibility of PDF methods for practical turbulent combustion systems.

AMS: 65C20; 76F55; 76M25; 80A32

Keywords: PDF methods; Particle methods; Finite-volume methods; Turbulent combustion modeling

1. Introduction

Turbulence closure based on the solution of a modeled probability density function (PDF) transport equation originated with the work of Lundgren [1]. Subsequently, PDF methods for chemically reacting and non-reacting turbulent flows have been developed by several researchers (see [2] for references).

And in their modern form, PDF methods for chemically reacting turbulent flows, including their associated Lagrangian particle-based solution methods, now largely follow the approach developed by Pope and coworkers.

* Corresponding author. Tel.: +1-814-863-6269; fax: +1-814-865-3389.
E-mail address: dch12@psu.edu (D.C. Haworth).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2003.08.032

A benefit of PDF methods for turbulent reacting flows is that chemical reaction and other one-point processes (e.g., radiative emission) appear in closed form. Advantages of PDF-based modeling for dealing with finite-rate chemistry and turbulence/chemistry interactions have been amply demonstrated over the preceding 15–20 years. Most PDF modeling studies reported to date have been limited to canonical flow configurations (e.g., statistically stationary axisymmetric jet flames [3]), although some applications to more complex configurations (three-dimensional time-dependent flows [4] and/or realistic combustion-device geometry [5]) have appeared.

Application of PDF methods to complex engineering flows of practical interest has been hindered by the unconventional Lagrangian particle-based methods that have been developed to solve PDF transport equations. To bring the benefits of PDF methods to bear in such flows, it is desirable to integrate the particle-based solution schemes into existing grid-based research and engineering computational fluid dynamics (CFD) codes. Hybrid particle/finite-volume (FV) algorithms have been developed for that purpose. While hybrid PDF algorithms have been available for many years [6–8], early approaches suffered from poor efficiency, robustness and accuracy. In part, this may be because they failed to respect fundamental requirements for consistency between the particle and FV representations that are implicit in the underlying mathematical/physical formulation. More recent work has addressed key numerical method and physical modeling issues in hybrid PDF methods. This includes particle/FV consistency and assessment of errors and convergence rates in up to two-dimensional statistically stationary flows [9–12], mean estimation and particle tracking for three-dimensional time-dependent flows using unstructured deforming meshes [4], and extension to thermal radiation and turbulence/radiation interactions [13]. However, significant work remains to realize efficient, robust and accurate hybrid schemes that are suitable for general three-dimensional time-dependent flows. In particular, an algorithm has been lacking to enforce a key consistency requirement in hybrid particle/FV PDF methods: namely, that the local mass or volume distribution must remain consistent between the Lagrangian (particle) and Eulerian (FV) representations.

The purpose of this research has been to broaden the accessibility of PDF methods so that they can be brought to bear in complex engineering flows. Towards that end, an efficient, robust and accurate particle/FV mass consistency algorithm is presented that is suitable for three-dimensional time-dependent flows on unstructured deforming computational meshes and that is compatible with multiple FV element types (hexahedra and degenerate hexahedra, including tetrahedra). Other issues in hybrid PDF methods also are addressed. The algorithm has been developed primarily with segregated pressure-based FV CFD codes in mind, although a similar approach can be followed for other solver types. It is intended primarily for "tightly coupled" hybrid particle/FV methods [10] where the particle code is run on each FV time step.

The remainder of the article is organized as follows. Salient aspects of PDF methods are reviewed in Section 2, and the nature of the mass consistency constraint is discussed. The CFD code that has been used in the present study is introduced in Section 3. The new mass consistency algorithm is developed in Section 4, and other issues in using particle methods with unstructured CFD codes are addressed in Section 5. The benefits and significance of the new algorithm are illustrated in Section 6 via computations for canonical configurations (a lid-driven cavity and an axisymmetric free jet) and for a simplified reciprocating piston–cylinder assembly (an idealized IC engine). In the final section, results are summarized and conclusions are drawn.

2. PDF methods for chemically reacting turbulent flows

For concreteness, we consider a single-phase (gaseous) multicomponent chemically reacting mixture. A low-Mach-number approximation is invoked so that spatial gradients in pressure do not affect the thermochemical equations. Then the mixture mass density ρ, specific heats c_p and c_v, and species chemical production rates $\underline{S}$ are functions only of composition (e.g., species mass fractions), enthalpy and a reference

pressure p_0 that is, at most, a function of time: $p_0 = p_0(t)$. Species mass fractions and enthalpy comprise the *s composition variables* for the system. We limit our attention to high-Reynolds-number flows; turbulent transport of mass, momentum, chemical species and enthalpy dominate molecular transport processes. Thermal radiation is neglected.

The PDF considered is the Eulerian one-point joint velocity-composition PDF of three velocity components $\underline{U} = \underline{U}(\underline{x}, t)$ and s scalar composition variables $\underline{\phi} = \underline{\phi}(\underline{x}, t)$: $f_{\underline{U},\underline{\phi}}(\underline{V}, \underline{\psi}; \underline{x}, t)$. This PDF is a density in the $3 + s$-dimensional $\underline{V} - \underline{\psi}$ phase space; it is, in addition, a function of up to three spatial coordinates $\underline{x}$ and of time t. The composition space can be large in the case of hydrocarbon-air systems. For example, 550 chemical species are considered in a recent detailed n-heptane/air oxidation mechanism [14]. Composition PDF's and velocity-composition-frequency PDF's [15] also can be treated using the algorithm that is developed here; thermal radiation and turbulence/radiation interactions can be considered [13,16]; and a similar approach can be used for particle-based filtered-density-function methods that are being developed as subgrid-scale models for large-eddy simulation [17–21].

Starting from the Eulerian partial differential equations (PDEs) expressing conservation of mass, momentum, chemical species and enthalpy, a transport equation for $f \equiv f_{\underline{U},\underline{\phi}}(\underline{V}, \underline{\psi}; \underline{x}, t)$ can be derived [2]:

$$\begin{aligned}
&\rho(\underline{\psi}, p_0)\frac{\partial f}{\partial t} + \rho(\underline{\psi}, p_0)V_j\frac{\partial f}{\partial x_j} + \left(\rho(\underline{\psi}, p_0)g_j - \frac{\partial\langle p\rangle}{\partial x_j}\right)\frac{\partial f}{\partial V_j} + \frac{\partial[\rho(\underline{\psi}, p_0)S_\alpha(\underline{\psi}, p_0)f]}{\partial\psi_\alpha} \\
&\quad = \frac{\partial}{\partial V_j}\left[\left\langle\frac{\partial p'}{\partial x_j} - \frac{\partial\tau_{ij}}{\partial x_i}\middle|\underline{V}, \underline{\psi}\right\rangle f\right] + \frac{\partial}{\partial\psi_\alpha}\left[\left\langle\frac{\partial J_i^\alpha}{\partial x_i}\middle|\underline{V}, \underline{\psi}\right\rangle f\right].
\end{aligned} \tag{1}$$

Here summation is implied over repeated indices i, j or α within a term ($i, j = 1, 2, 3$; $\alpha = 1, \ldots, s$). Angled brackets $\langle\rangle$ denote conventional mean quantities, and a tilde $\sim$ is used to denote a density- (Favre-) averaged mean quantity. Mean quantities are related to the PDF by,

$$\begin{aligned}
\langle Q\rangle &= \langle Q(\underline{x}, t)\rangle = \int_{\underline{\psi}}\int_{\underline{V}} Q(\underline{V}, \underline{\psi}) f_{\underline{U},\underline{\phi}}(\underline{V}, \underline{\psi}; \underline{x}, t)\,\mathrm{d}\underline{V}\,\mathrm{d}\underline{\psi}, \\
\tilde{Q} &= \tilde{Q}(\underline{x}, t) = \langle\rho\rangle^{-1}\int_{\underline{\psi}}\int_{\underline{V}} \rho(\underline{\psi}, p_0) Q(\underline{V}, \underline{\psi}) f_{\underline{U},\underline{\phi}}(\underline{V}, \underline{\psi}; \underline{x}, t)\,\mathrm{d}\underline{V}\,\mathrm{d}\underline{\psi},
\end{aligned} \tag{2}$$

where integration is over the entire $3 + s$-dimensional phase space. Fluctuations with respect to the conventional mean are denoted using a single prime $'$ and double primes $''$ are used for fluctuations about a Favre-mean quantity,

$$\begin{aligned}
Q' &= Q'(\underline{x}, t) = Q(\underline{x}, t) - \langle Q(\underline{x}, t)\rangle, \\
Q'' &= Q''(\underline{x}, t) = Q(\underline{x}, t) - \tilde{Q}(\underline{x}, t).
\end{aligned} \tag{3}$$

The term $\left\langle\frac{\partial p'}{\partial x_j}\middle|\underline{V}, \underline{\psi}\right\rangle$ represents the mean of the fluctuating pressure gradient, conditioned on the velocity being equal to $\underline{V}$ and the composition being equal to $\underline{\psi}$. Similarly, $\left\langle\frac{\partial\tau_{ij}}{\partial x_i}\middle|\underline{V}, \underline{\psi}\right\rangle$ and $\left\langle\frac{\partial J_i^\alpha}{\partial x_i}\middle|\underline{V}, \underline{\psi}\right\rangle$ are the conditional means of the divergence of the viscous stress tensor τ_{ij} and of composition variable α's diffusive flux $\underline{J}^\alpha$, respectively. Transport in physical space by the velocity $\underline{V}$, transport in velocity space by the body force ($\underline{g}$) and the mean pressure gradient ($\partial\langle p\rangle/\partial x_j$) and transport in composition space by chemical reaction (chemical source terms $\underline{S}$) appear in closed form; these are the terms on the left-hand side of Eq. (1). On the right-hand side are the terms to be modeled. These represent transport in velocity space due to the fluctuating pressure gradient and molecular viscosity, and transport in composition space by molecular diffusion.

Because of its large dimensionality (up to $7 + s$ independent variables), the PDF transport equation is not amenable to numerical solution using conventional grid-based methods. Lagragian Monte Carlo particle methods have been developed as an alternative.

2.1. Lagrangian particle representation

The turbulent reacting flow is represented by a large number N_p of computational particles. The ith particle is assigned a mass $m^{(i)}$; in the simplest case, $m^{(i)} = m/N_p$ where m is the total system mass. In addition, each particle is characterized by three position coordinates $\underline{x}^{(i)}(t)$, three velocity components $\underline{U}^{(i)}(t)$ and s scalar compositions $\underline{\phi}^{(i)}(t)$ (again, species mass fractions and enthalpy).

Each computational particle can be interpreted as a delta function discretization of the PDF. Formally, a discrete mass density function $\mathscr{F}^*$ is introduced

$$\mathscr{F}^*(\underline{V},\underline{\psi},\underline{x};t) \equiv \sum_{i=1}^{Np} m^{(i)}\delta\big(\underline{V}-\underline{U}^{(i)}(t)\big)\delta\Big(\underline{\psi}-\underline{\phi}^{(i)}(t)\Big)\delta\big(\underline{x}-\underline{x}^{(i)}(t)\big), \tag{4}$$

where $\delta(\underline{x}-\underline{x}^{(i)}(t))$ is a three-dimensional delta function at the particle location, and similarly for $\delta(\underline{V}-\underline{U}^{(i)}(t))$ and $\delta(\underline{\psi}-\underline{\phi}^{(i)}(t))$. For a consistent discrete representation, we require that,

$$\langle \mathscr{F}^*(\underline{V},\underline{\psi},\underline{x};t)\rangle = \rho(\underline{\psi},p_0) f_{\underline{U},\underline{\phi}}(\underline{V},\underline{\psi};\underline{x},t). \tag{5}$$

In an infinitesimal time increment $\mathrm{d}t$, the position, velocity and composition of each particle evolve according to,

$$\begin{aligned}
\mathrm{d}\underline{x}^* &= \underline{U}^*\mathrm{d}t,\\
\mathrm{d}\underline{U}^* &= (\underline{g}-\nabla\langle p\rangle^*/\rho(\underline{\phi}^*,p_0))\,\mathrm{d}t + \underline{A}^*_{p',\mathrm{mix}}\mathrm{d}t,\\
\mathrm{d}\underline{\phi}^* &= \underline{S}(\underline{\phi}^*,p_0)\,\mathrm{d}t + \mathrm{d}\underline{\phi}^*_{\mathrm{mix}}.
\end{aligned} \tag{6}$$

Here the superscript * refers to any particle, $1 \leqslant i \leqslant N_p$, and a mean quantity with superscript * refers to the mean value evaluated at the particle location: e.g., $\langle p\rangle^* = \langle p(\underline{x}^*(t),t)\rangle$. The term $\underline{A}^*_{p',\mathrm{mix}}$ denotes a particle acceleration due to the fluctuating pressure gradient and molecular viscosity (velocity mixing); and $\mathrm{d}\underline{\phi}^*_{\mathrm{mix}}$ is the increment in composition due to molecular diffusion (scalar mixing). $\underline{A}^*_{p',\mathrm{mix}}\mathrm{d}t$ and $\mathrm{d}\underline{\phi}^*_{\mathrm{mix}}$ are modeled using either stochastic or deterministic processes. And these models for particle behavior provide effective closure models for the terms that appear on the right-hand side of Eq. (1).

2.2. Hybrid particle/FV methods

Hybrid methods seek to take advantage of the respective strengths of particle-based and grid-based solution procedures. For a particle method, strengths include exact treatments of advection in physical space and of one-point source terms in the PDF transport equation. For grid-based methods, robust procedures have been developed to deal with the (linear) coupling between the mean continuity and mean momentum equations through the mean pressure. Therefore, it is expedient to decompose the particle velocity into its mean and fluctuating components,

$$\underline{U}^*(t) = \tilde{\underline{U}}^* + \underline{u}''^*(t), \tag{7}$$

and to employ a conventional grid-based CFD solver to compute the mean velocity field $\tilde{\underline{U}}(\underline{x},t)$ while using a particle method for the fluctuating velocity $\underline{u}''^*(t)$. Scalar composition variables can be decomposed similarly. Formally, the PDF transport equation (Eq. (1)) can be recast as an equation for the PDF of the fluctuating velocity and composition, and the particle equations (Eq. (6)) can be recast as equations for particle fluctuating velocity and composition (see [10], for example).

Here the focus is on the advection of particles in physical space by the FV mean velocity field.

2.3. The spatial distribution of particle mass

In the representation outlined above, each computational particle represents a specified mass of fluid; this is a *mass density function* method. Therefore, the distribution of particle mass (or volume) in physical space is not arbitrary; it must remain consistent with the fluid mass (or volume) distribution. Specifically, the condition,

$$\int_{\underline{\psi}} \int_{\underline{V}} \rho(\underline{\psi}, p_0) f_{\underline{U},\underline{\phi}}(\underline{V}, \underline{\psi}; \underline{x}, t)\, \mathrm{d}\underline{V}\, \mathrm{d}\underline{\psi} = \langle \rho(\underline{x}, t) \rangle \tag{8}$$

must be satisfied for all time for $f_{\underline{U},\underline{\phi}}(\underline{V}, \underline{\psi}; \underline{x}, t)$ to remain a valid PDF. In the discrete representation, the consistency requirement can be written as,

$$\int_{\underline{\psi}} \int_{\underline{V}} \langle \mathscr{F}^* \rangle\, \mathrm{d}\underline{V}\, \mathrm{d}\underline{\psi} = \langle \rho(\underline{x}, t) \rangle. \tag{9}$$

While the mass consistency condition can be expressed in several alternative forms (e.g., [2,12,22]), Eqs. (8) and (9) are appropriate for present purposes. The initial distributions of particle masses and of particle locations in physical space are prescribed in a manner that is consistent with the initial spatial distribution of mean fluid mass (Section 5). Then in order for the discrete particle representation to remain a valid discretization of the modeled PDF transport equation, it is necessary that the spatial distribution of mean particle mass remain consistent with the distribution of mean fluid mass for all time.

The conditions under which an initially valid particle distribution will remain valid as the system evolves in time have been enumerated by Pope [2]. An evolution equation for the particle mass density distribution in physical space has been derived by Haworth and Pope [22]. An analysis for constant-density flow is presented in Chapter 12 of [23]. And this and other consistency issues have been developed further by Muradoglu et al. [12]. For present purposes, the salient result of these analyses can be summarized as follows: For an initially valid particle distribution to remain valid as the system evolves in time, it is necessary and sufficient that the particle system evolves in a manner that is consistent with the mean continuity equation. The mean continuity equation, in turn, is satisfied if and only if the mean pressure field satisfies the Poisson equation that results from taking the divergence of the mean momentum equation.

The mean continuity, mean momentum, and mean pressure equations are,

$$\frac{\partial \langle \rho \rangle}{\partial t} + \frac{\partial \langle \rho \rangle \tilde{u}_i}{\partial x_i} = 0, \tag{10}$$

$$\frac{\partial \langle \rho \rangle \tilde{u}_j}{\partial t} + \frac{\partial \langle \rho \rangle \tilde{u}_j \tilde{u}_i}{\partial x_i} = \frac{\partial (\langle \tau_{ij} \rangle + \tau_{T,ij})}{\partial x_i} - \frac{\partial \langle p \rangle}{\partial x_j} + \langle \rho \rangle g_j \quad (j = 1, 2, 3), \tag{11}$$

and

$$\frac{\partial^2 \langle p \rangle}{\partial x_j x_j} = \frac{\partial^2 \langle \rho \rangle}{\partial t^2} - \frac{\partial^2 \langle \rho \rangle \tilde{u}_i \tilde{u}_j}{\partial x_i \partial x_j} + \frac{\partial^2 (\langle \tau_{ij} \rangle + \tau_{T,ij})}{\partial x_i \partial x_j} + g_j \frac{\partial \langle \rho \rangle}{\partial x_j}. \tag{12}$$

Here $\tau_{T,ij} = -\langle \rho \rangle \widetilde{u_i'' u_j''}$ is the effective turbulent stress.

2.4. Implications for hybrid particle/FV methods

In the present approach, consistency among the mean pressure, mean velocity and mean density fields is ensured (at a control volume, or element, level) by a FV algorithm. From a practical point of view, then, the

mass consistency condition implies that the mean velocity field that is used to advect particles in physical space must be consistent with the mean continuity equation. The mean velocity field for particle advection is obtained via interpolation from the FV-computed mean velocity field. Thus this interpolated mean velocity field should satisfy Eq. (10). While this may appear straightforward, there are several subtleties.

First, it is not straightforward to reconstruct a continuous mean velocity field that satisfies Eq. (10) locally from discrete values available on a mesh in a multidimensional flow. To illustrate this, consider the two-dimensional steady incompressible example shown in Fig. 1. There discrete values of the mean velocity are available at the four vertices of the unit square, and bilinear interpolation is used to reconstruct a continuous mean velocity field in the interior of the square. Edges are aligned with the $x = x_1$ and $y = x_2$ coordinate directions, as shown. For simplicity, mean velocity components in the x_1 and x_2 directions are denoted by u and v, respectively. The mean continuity equation reduces to $\partial u/\partial x + \partial v/\partial y = 0$. Here $\partial u/\partial x = (y-1)u_1 + (1-y)u_2 + yu_3 - yu_4$ and $\partial v/\partial y = (x-1)v_1 - xv_2 + xv_3 + (1-x)v_4$. Since $\partial u/\partial x$ is a function only of y and $\partial v/\partial y$ is a function only of x, the mean continuity cannot be satisfied locally at every point in the unit square using this representation. (An alternative velocity interpolation scheme that resolves this issue for two-dimensional rectangular elements has been presented by Jenny et al. [11].)

Second, in FV methods the governing PDEs are satisfied only in an integral sense over control volumes that correspond to the computational cells or elements. In a consistent hybrid particle/FV method, a correct particle distribution should be enforced at the same level. The mean continuity equation can be integrated over an n-faced FV computational element e to yield,

$$\frac{\mathrm{d}m^e}{\mathrm{d}t} = -\sum_{f=1}^{n} Q^{e,f,FV}. \tag{13}$$

Here $Q^{e,f,\mathrm{FV}}$ is the mass flowrate (mass per unit time) leaving FV element e across face f. Element-face mass flowrates or fluxes are central to FV CFD formulations. These flowrates are central to the present particle/FV mass consistency algorithm as well.

And third, even if the mean velocity field used for particle advection did satisfy the mean continuity equation locally, or if the implied face fluxes satisfied Eq. (13) at the element level, that would not guarantee

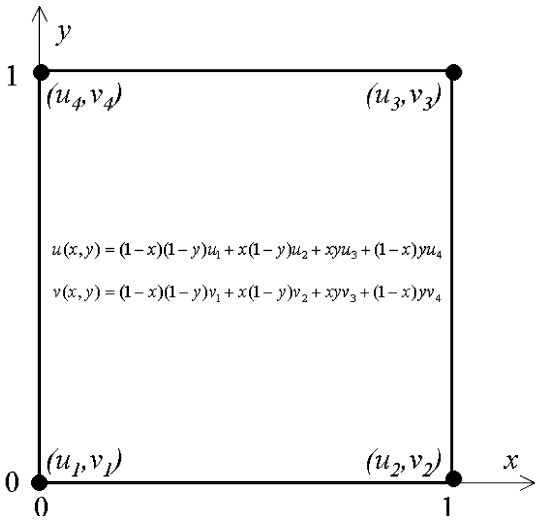

Fig. 1. A two-dimensional interpolation example. Finite-volume mean velocity components u and v are available at the four vertices of the unit square. A continuous mean velocity field field for particle advection in the interior of the square is constructed using bilinear interpolation.

consistency between the particle mass/volume distribution and the fluid mass/volume distribution for all time. The two may diverge as a result of discretization errors and other numerical inaccuracies (e.g., finite number of particles; noise from stochastic models used to simulate particle acceleration, etc.). This will be demonstrated and discussed further in the examples presented below (Section 6). An essential ingredient of the present algorithm is a particle position correction that drives any local deviation between the mean particle mass and the fluid mass distributions to zero at the element level, regardless of how the deviation arises.

In the absence of an algorithm to control the spatial distribution of particles, large deviations between the mean particle mass distribution and the fluid mass distribution (e.g., the mean density field) arise in hybrid particle/FV computations. The problem is particular acute for multidimensional flows, and on computational meshes where there are large variations in element volumes and/or large departures from orthogonality. The present algorithm is designed to maintain consistency between the spatial distribution of mean particle mass and the FV mass at the FV element level.

3. The FV code

The algorithm developed here is intended to be as generic as is practicable. Where specific choices had to be made, these have been based on the CFD code that is the immediate target of this research. Thus a brief overview of the underlying FV code is provided here. Details can be found in [24–26].

The CFD code solves the coupled PDEs for a chemically reacting multicomponent gas mixture using a FV method. The equations are formulated in terms of density-weighted (Favre-averaged) mean quantities. The fundamental equations are: a pressure correction equation (Eq. (12), reformulated as an equation for a pressure difference); the mean momentum equations (Eq. (11)); a mean internal energy or enthalpy equation; and mean chemical species equations. Mean density is obtained using an equation of state.

An iteratively implicit pressure-based sequential (segregated) solution procedure is used. The procedure accommodates incompressible and/or compressible flows and steady and/or transient flows. It is applicable for essentially arbitrary Mach number. Pressure/momentum/continuity coupling is accomplished using a pressure-corrector scheme patterned after PISO [28,29]. The equations are solved on an unstructured non-orthogonal mesh of convex principally six-faced hexahedral elements; edge and/or face degeneracies are permissible (prism, pyramid and tetrahedron; Fig. 2). Cell-centered co-located variables are employed. The

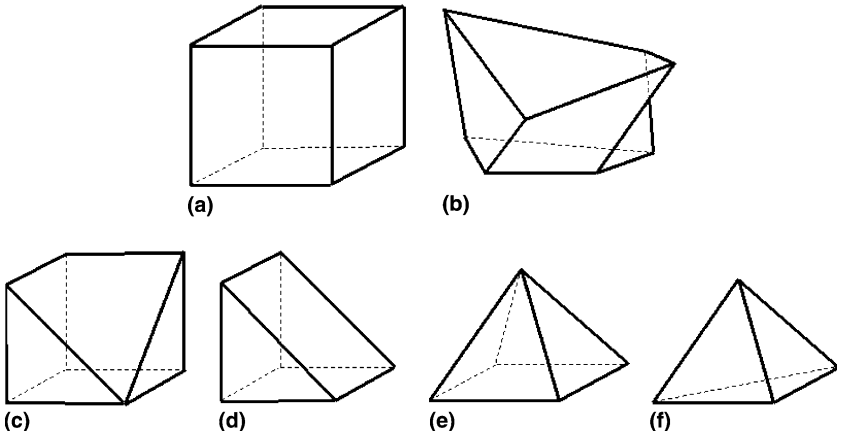

Fig. 2. Finite-volume element shapes. Logically six-faced convex volume elements (hexahedra) are employed that can be mapped to the unit cube (a). Elements can be non-orthogonal (b). Edge and/or face degeneracies are allowed (lower row) including prisms (d), pyramids (e) and tetrahedra (f).

discretization is first-order in time and up to second-order in space. Additional features that have been implemented to accommodate complex geometry and flow include non-aligned interfaces [30,31] and solution-adaptive local mesh refinement [32]. Available physical models include two-equation turbulence models and a variety of turbulent combustion models.

4. A mass consistency algorithm

A three-stage procedure is used to establish a mean velocity field for particle advection.

- Stage 1: Establish a continuous approximation to the FV mean velocity field using discrete values available at mesh vertices and appropriate interpolation functions.
- Stage 2: Compute a correction velocity field to enforce consistency between FV element-face mass flowrates and the element-face mass flowrates implied by the Stage 1 velocity field.
- Stage 3: Monitor the deviation between FV element-level mass and the mean particle mass in each element, and compute a second correction velocity field to drive the deviation towards zero.

This can be thought of as a predictor–corrector scheme, with Stages 1 and 2 providing a "predicted" mean velocity field for particle advection, and Stage 3 providing a "corrected" field that enforces the element-level particle/FV mass consistency requirement.

4.1. Required FV quantities

Element-face mass fluxes are central to FV CFD schemes, and it is reasonable to expect that this quantity is available from any underlying FV code. The FV mass flowrate out of element e through face f is denoted by $Q^{e,f,\mathrm{FV}}$ (Eq. (13); dimensions of mass per unit time).

It is presumed that FV dependent variables are available at mesh vertices. In the case of an element-centered CFD code (such as the present one), vertex values are established by interpolation. For example, the value of a FV mean quantity Φ at vertex l, Φ^l, can be determined from element-centered values Φ^e by,

$$\Phi^l = \sum_{e\in l} \Lambda^{e,v}\Phi^e \Bigg/ \sum_{e\in l} \Lambda^{e,v}. \tag{14}$$

Here the notation $e \in l$ denotes the set of FV elements that share vertex l, v is the local element vertex id ($1 \leqslant v \leqslant 8$) corresponding to global vertex id l, and $\Lambda^{e,v}$ is the volume centroid weight factor for vertex v of element e (Eq. (A.11)). For vertices that lie on boundary faces, the actual boundary values are assigned to Φ^l.

Trilinear basis functions are used to establish continuous fields of mean mass density and mean velocity within each element from FV vertex values; see Appendix A and Fig. 3. This approach accommodates convex hexahedral elements, and any legitimate (non-zero volume) edge and/or face degeneracies of hexahedral elements including tetrahedra, in a single consistent formulation. As pointed out earlier, trilinear basis functions *locally* are incompatible with the mean continuity equation. That is not an issue in the present algorithm.

4.2. Stage 1

The Stage 1 mean velocity field is determined by trilinear interpolation from FV vertex values,

$$\underline{U}^e(\underline{x},t) = \sum_{l\in e} b^{e,v}(\underline{\xi})\underline{U}^l(t), \tag{15}$$

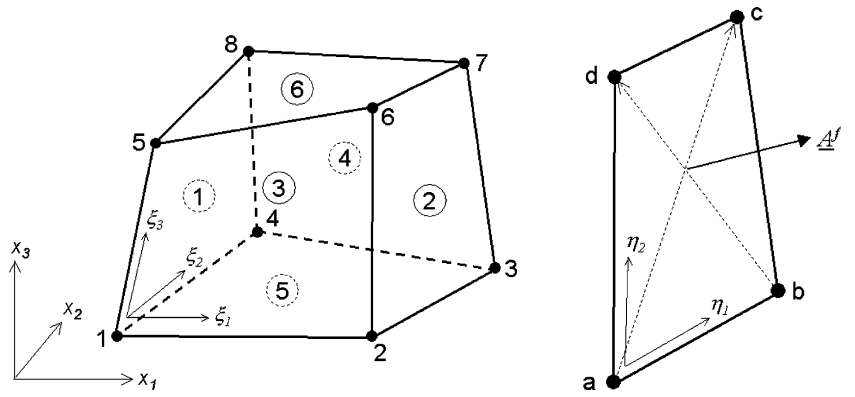

Fig. 3. Element and face conventions. Left: volume element conventions. Vertex numbering (1–8), face numbering (circled, 1–6) and local logical coordinates (ξ_1, ξ_2, ξ_3) for a six-faced finite-volume element. Right: face conventions. For face 1: $abcd = 8415$ and $(\eta_1, \eta_2) = (1 - \xi_3, 1 - \xi_2)$; for face 2: $abcd = 6237$ and $(\eta_1, \eta_2) = (1 - \xi_3, \xi_2)$; for face 3: $abcd = 6512$ and $(\eta_1, \eta_2) = (1 - \xi_1, 1 - \xi_3)$; for face 4: $abcd = 3487$ and $(\eta_1, \eta_2) = (1 - \xi_1, \xi_3)$; for face 5: $abcd = 3214$ and $(\eta_1, \eta_2) = (1 - \xi_2, 1 - \xi_1)$; and for face 6: $abcd = 8567$ and $(\eta_1, \eta_2) = (1 - \xi_2, \xi_1)$.

where the $b^{e,v}(\underline{\xi})$ are the trilinear basis functions given by Eq. (A.2). Throughout this section, angled brackets and tildes denoting conventional and density-weighted means, respectively, are dropped for clarity. Thus ρ represents the mean fluid density $\langle\rho\rangle$ and $\underline{U}$ represents the density-weighted mean fluid velocity $\tilde{\underline{U}}$.

The Stage 1 mean velocity field is continuous across adjacent elements that share a common face, and hence through the entire computational domain.

4.3. Stage 2

The mass flowrate out of element e across element face f that is implied by the trilinear mapping is denoted by $Q^{e,f,\mathrm{TL}}$. This flowrate is obtained by integration over the element face: $Q^{e,f,\mathrm{TL}} = \int_f \rho\underline{U} \cdot \mathrm{d}\underline{A}$, with the $\rho\underline{U}$ product being evaluated from FV mean vertex values using the trilinear basis functions. The result can be expressed in terms of the mean mass densities and velocities at the four vertices defining face f, and of the outward-pointing unit normal vector to the face $\underline{n}^{e,f} = (n_1^{e,f}, n_2^{e,f}, n_3^{e,f})$ (Eq. (A.5)):

$$Q^{e,f,\mathrm{TL}} = A^f \sum_{l \in f} \lambda^{f,v} \rho^l (U_1^l n_1^{e,f} + U_2^l n_2^{e,f} + U_3^l n_3^{e,f}). \tag{16}$$

Here A_f is the face projection area (Eq. (A.4)) and $\lambda^{f,v}$ is the area weight factor for local vertex v (corresponding to global vertex l) of face f (Eq. (A.8)). In general, the face mass flowrate $Q^{e,f,\mathrm{TL}}$ given by Eq. (16) is not equal to the FV element-face mass flowrate $Q^{e,f,\mathrm{FV}}$.

An element-level correction velocity field is introduced to enforce consistency between the FV element-face mass flowrates and the element-face mass fiowrates implied by the Stage 1 velocity field. Stage 2 correction quantities are denoted using the hat $\hat{}$ notation. An element-level mass flowrate correction for face f is defined in terms of element-level correction velocities for the four vertices incident on face f by,

$$\hat{Q}^{e,f} \equiv Q^{e,f,\mathrm{FV}} - Q^{e,f,\mathrm{TL}} = A^f \sum_{l \in f} \lambda^{f,v} \rho^l (\hat{U}_1^{e,v} n_1^{e,f} + \hat{U}_2^{e,v} n_2^{e,f} + \hat{U}_3^{e,v} n_3^{e,f}). \tag{17}$$

For each computational element, there are 24 correction velocity components $\hat{U}_i^{e,v}$ to be determined (three velocity components i for each of eight vertices v) subject to six constraints $\hat{Q}^{e,f}$ (six element faces f).

Eighteen additional constraints are provided by requiring that the normal component of the correction mass flux be uniform across each of the six faces of element e. With v_1, v_2, v_3 and v_4 denoting the four vertices of face f (corresponding to global vertex id's l_1, l_2, l_3 and l_4),

$$\begin{aligned}\rho^{l_1}\hat{U}_1^{e,v_1}n_1^{e,f} + \rho^{l_1}\hat{U}_2^{e,v_1}n_2^{e,f} + \rho^{l_1}\hat{U}_3^{e,v_1}n_3^{e,f} &= \rho^{l_2}\hat{U}_1^{e,v_2}n_1^{e,f} + \rho^{l_2}\hat{U}_2^{e,v_2}n_2^{e,f} + \rho^{l_2}\hat{U}_3^{e,v_2}n_3^{e,f} \\ &= \rho^{l_3}\hat{U}_1^{e,v_3}n_1^{e,f} + \rho^{l_3}\hat{U}_2^{e,v_3}n_2^{e,f} + \rho^{l_3}\hat{U}_3^{e,v_3}n_3^{e,f} \\ &= \rho^{l_4}\hat{U}_1^{e,v_4}n_1^{e,f} + \rho^{l_4}\hat{U}_2^{e,v_4}n_2^{e,f} + \rho^{l_4}\hat{U}_3^{e,v_4}n_3^{e,f} \end{aligned} \tag{18}$$

provides three additional equations for each of the six faces f. While the constraint of uniform correction mass flux across each face is somewhat arbitrary, it has the advantage of simplicity and it leads to a set of algebraic equations that can be solved in closed form for the 24 Stage 2 correction velocity components. Details of the algebraic solution are provided in Appendix B. As noted there, the correction velocity field is continuous within each computational element but is only piecewise continuous through the computational domain, in general. That is, the value of the correction velocity is not necessarily the same for each element that shares a given vertex.

The mean velocity field used to advect particles is then the sum of the continuous Stage 1 velocity field and the piecewise continuous Stage 2 correction velocity field. Advection of particles using this mean velocity field still does not guarantee consistency between the FV element mass and the mean particle mass in each element as the system evolves in time. For example, in the case of a first-order particle advection scheme with zero particle fluctuating velocity, each particle's position $\underline{x}^*$ advances in time simply according to,

$$\underline{x}^*(t+\Delta t) = \underline{x}^*(t) + \underline{U}^*\Delta t, \tag{19}$$

where $\underline{U}^*$ is the mean advection velocity (sum of Stage 1 and Stage 2 velocities) evaluated at the particle's beginning-of-timestep position. Clearly, deviations in the particle mass distribution can build up over time, particularly in the case of steep spatial and/or temporal gradients in mean velocity, low particle number densities (small number of particles per FV element) and large particle Courant numbers. Here a particle Courant number can be defined as,

$$C^* \equiv |\underline{U}^*|\Delta t/\mathscr{L}^e, \tag{20}$$

where Δt is the computational timestep and $\mathscr{L}^e$ is a characteristic length of the computational element (e.g., $\mathscr{L}^e = (V^e)^{1/3}$, where V^e is the volume of computational element e). Even in the dense particle limit (where deterministic errors become independent of the number of particles [33]) and with higher-order particle advection schemes [34–36], deviations between the mean particle mass distribution and the fluid mass distribution will develop and grow as a result of non-uniformities in particle mass distribution within FV elements [S.B. Pope, Cornell University, personal communication, 2002]; this will be demonstrated in Section 6. The deviations are addressed in Stage 3.

4.4. Stage 3

Stages 1 and 2 seek to establish *a priori* a "good" mean velocity field for particle advection based on the FV mean velocity field and element-face mass fluxes. By contrast, Stage 3 is an *a posteriori* particle position correction. State 3 addresses directly the deviation between element mass and mean particle mass within each element, and redistributes the particles in physical space in a manner that drives the deviation towards zero, without regard to how the deviation arose.

The Stage 3 velocity correction is denoted using a double-hat $\hat{}$ notation, and is formulated in the same way as Stage 2:

$$\hat{\hat{Q}}^{e,f} = A^f \sum_{l \in f} \lambda^{f,v} \rho^l \left(\hat{\hat{U}}_1^{e,v} n_1^{e,f} + \hat{\hat{U}}_2^{e,v} n_2^{e,f} + \hat{\hat{U}}_3^{e,v} n_3^{e,f} \right), \tag{21}$$

where $\hat{\hat{Q}}^{e,f}$ is an element-face mass flowrate. The difference between Stages 2 and 3 lies in the specification of the face flowrates.

The fluid mass associated with FV element e is denoted by $m^{e,\mathrm{FV}}$, and $m^{e,P}$ denotes a mass of particles associated with FV element e; a more precise specification will be given shortly. The difference between these two masses, divided by a timescale τ_{S3} (S3 for "Stage 3"), is a quantity that can be interpreted as an element-level mass residual R^e,

$$R^e \equiv (m^{e,\mathrm{FV}} - m^{e,P})/\tau_{\mathrm{S3}}. \tag{22}$$

The idea is to find a set of face flowrates $\hat{\hat{Q}}^{e,f}$ that will drive these residuals toward zero; then from the face flowrates, a set of element-level correction velocities $\underline{\hat{\hat{U}}}^{e,v}$ is constructed following the same procedure as in Stage 2.

The computation of element-face mass flowrates to counter non-zero element-level mass residuals is central to many FV CFD algorithms. In that case, the mass residual for each element is defined as the difference between the net rate of accumulation of fluid mass in the element and the net efflux of fluid mass across the element's bounding faces. Segregated pressure-based FV schemes based on SIMPLE [37] and/or PISO [28,29], for example, include one or more pressure/velocity corrector steps that compute corrections to the pressure and velocity fields along with the necessary element-face mass fluxes to drive the mass residuals to zero. This is accomplished by solving a symmetric linear system for a pressure correction field. The result is a set of element-face mass flowrates $\hat{\hat{Q}}^{e,f}$ that satisfy,

$$R^e = -\sum_{f \in e} \hat{\hat{Q}}^{e,f}. \tag{23}$$

In the event that this capability is not provided in the underlying FV code, a separate elliptic solver module can be provided to solve for $\hat{\hat{Q}}^{e,f}$, given R^e. An advantage of using the functionality of the underlying FV solver directly is that the Stage 3 velocities then are, by construction, consistent with the underlying governing equations (the compressible Navier–Stokes equations).

The form of the equations used in the present study to compute $\hat{\hat{Q}}^{e,f}$ is given in Appendix C. A single pressure corrector step is taken with the mass residuals set to R^e of Eq. (22). It is emphasized that the pressure correction field and the implied velocity field are *not* used to update the mean pressure and mean velocity fields in the underlying FV code.

It remains to specify $m^{e,\mathrm{FV}}$, $m^{e,P}$ and τ_{S3}. In the most aggressive approach, $m^{e,\mathrm{FV}} = m^{e,\mathrm{FV}}(t) = \rho^{e,\mathrm{FV}}(t) V^e(t)$ (the instantaneous fluid mass in the FV element), $m^{e,P} = \sum_{i \in e} m^{(i)}(t)$ (the instantaneous particle mass in the element) and $\tau_{\mathrm{S3}} = \Delta t$ (the computational time step). This corresponds to an attempt to maintain consistency between element mass and particle mass on each computational timestep, and goes beyond the formal consistency requirement that applies only to the *mean* particle mass distribution. With this approach, one might anticipate relatively large Stage 3 correction velocity magnitudes and large timestep-to-timestep variations in Stage 3 velocities as the algorithm attempts to correct for "shot noise" as individual particles pass from one element to another.

A less aggressive approach is to relax the mean particle mass towards the mean FV element mass. This can be accomplished, for example, by averaging the element and/or particle masses over specified time windows,

$$m^{e,\mathrm{FV}} = \int_{t-\tau_{\mathrm{avg,FV}}}^{t} m^{e,\mathrm{FV}}(t')\,\mathrm{d}t'/\tau_{\mathrm{avg,FV}},$$
$$m^{e,P} = \int_{t-\tau_{\mathrm{avg,P}}}^{t} m^{e,P}(t')\,\mathrm{d}t'/\tau_{\mathrm{avg,P}} \tag{24}$$

and/or specifying a timescale τ_{S3} that corresponds to several computational time steps. In addition, the Stage 3 correction can be applied on every computational time step or less often – every N_3 time steps, say (timescale $\tau_{S3'} = N_3\Delta t$). The timescales $\tau_{\mathrm{avg,FV}}$, $\tau_{\mathrm{avg,P}}$, τ_{S3} and $\tau_{S3'}$ can be specified based on physical reasoning (e.g., estimates of the flow timescales) or heuristically (e.g., a specified number of timesteps). Results for several combinations are discussed in Section 6.

4.5. Steady flows

The algorithm has been developed primarily with time-dependent flows in mind. It applies immediately to steady flows in the case where a stationary solution is obtained by time marching. Examples are provided in Section 6. For iterative steady-flow algorithms, a similar approach can be followed with averaging over multiple iterations replacing time averaging.

4.6. Non-hexahedral meshes

The algorithm has been designed to be compatible with edge and/or face degeneracies of convex six-faced hexahedral FV elements. In particular, it reduces to the appropriate form in the case of tetrahedral elements (see Appendices A and B). However, a more efficient algorithm for strictly tetrahedral meshes can be derived directly following the same steps as outlined above and in Appendices A and B. In that case, the natural choice for element logical coordinates and basis functions is the normalized height above the triangular element faces, following standard finite-element practice (e.g., [38]).

4.7. Deforming meshes

In the case where the computational mesh deforms in time, there is an additional contribution to the FV element-face mass flowrate from grid motion. The present mass consistency algorithm applies directly, provided that the grid-motion element-face fluxes are included in $Q^{e,f,\mathrm{FV}}$. That is the approach that has been taken here.

4.8. Inflow/outflow boundaries

Special provisions are required to accommodate mass flow through the boundaries of the computational domain. At each boundary face where there is net inflow, new computational particles are introduced such that the total particle mass introduced over the computational timestep is exactly equal to the FV mass that enters over the timestep. And at each boundary face where there is net outflow, the total particle mass extracted is exactly equal to the FV mass that leaves over the timestep. Thus the total particle mass and total fluid mass in the computational domain remain equal at all times, and the Stage 3 particle position correction (particle redistribution *within* the computational domain) is applied directly with zero correction flowrates $\hat{\hat{Q}}^{e,f}$ imposed at faces that correspond to inflow or outflow boundaries.

4.9. Non-aligned interfaces and local mesh refinement

Data structures are more complicated in cases where a face is shared by more than two FV elements. This situation arises with non-aligned interfaces [30,31] and solution-adaptive local mesh refinement via element splitting [32], for example. This is accommodated naturally in the present vertex- and face-based mass consistency algorithm, as long as the correct face mass flowrates are provided by the underlying FV solver. The details are not presented here.

4.10. Comparison to Muradoglu, Pope and Caughey [12]

An alternative mass consistency correction algorithm has been published by Muradoglu et al. [12]. A *complete* set of consistency conditions for hybrid particle/FV methods (beyond particle/FV mass consistency) was considered in that work. That algorithm also is based on the solution of an elliptic equation with a source term that corresponds to the difference between the particle and FV mass distributions. By contrast to the present approach, the algorithm was aimed at "loosely coupled" hybrid particle/FV schemes for statistically stationary flows; results were presented only for two-dimensional flows on structured meshes; and the elliptic equation to be solved contained several empirical parameters.

5. Other numerical aspects

Other aspects of the hybrid particle/FV method are summarized briefly in this section. For the most part, these follow the approaches that have been developed and implemented by Subramaniam and Haworth [4].

5.1. Particle initialization

Particle initial masses and positions are prescribed in a manner that is consistent with the element-level FV mass distribution. The number of particles in each element is proportional to the element volume and the three logical coordinates for each particle are sampled from a random distribution uniform on [0., 1.]. A maximum total number of particles $N_{P,\max}$ is specified, and a minimum and maximum number of particles per element N_{pe} are prescribed ($N_{pe,\min}$ and $N_{pe,\max}$, respectively). The latter two values accommodate meshes having large variations in element volume. Initially, then, the mass of particles within each element is exactly equal to the FV element mass.

The specification of initial particle properties is not directly relevant to the present discussion. These also are prescribed in a manner that is consistent with FV mean values.

5.2. Particle tracking

A particle tracking algorithm for three-dimensional unstructured stationary or deforming meshes, large particle Courant numbers (>10, say), and large variations in FV element size (up to a factor of 10^6 variation in element volume) has been described in [4]. Briefly, particles are tracked from element to element using the same trilinear basis functions that are used for the present mass consistency algorithm. A particle leaves its current host element when the value of one of the particle's logical coordinates reaches zero or unity; and its new host element and initial logical coordinates in the new element are known from the FV connectivity pointers. For expediency, the approximation described in Appendix A has been used for all cases presented in Section 6 (Fig. 4); this has been found to result in negligible degradation in accuracy.

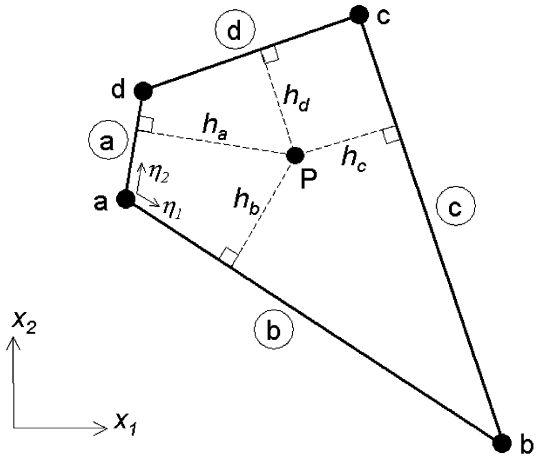

Fig. 4. The logical coordinates corresponding to point P are given approximately by $\eta_1 \approx h_a/(h_a + h_c), \eta_2 \approx h_b/(h_b + h_d)$.

5.3. Particle number density control

To maintain an acceptable distribution of particles as they move in physical space ($N_P \leqslant N_{P,\max}; N_{pe,\min} \leqslant N_{pe} \leqslant N_{pe,\max}$ for $1 \leqslant e < N_E$, where N_E is the number of FV elements), particle *cloning* and *annihilation* algorithms are employed. Particles are cloned in elements having $N_{pe} < N_{pe,\min}$ by splitting a particle of mass m^* into two particles each having mass $m^*/2$ and each having the same properties as the original particle: the highest-mass particles in each element are cloned. Particles are annihilated in elements having $N_{pe} > N_{pe,\max}$ by collapsing three particles selected at random (with preference given to low-mass particles) into two particles in a manner that preserves particle mass and mean particle properties exactly and that minimizes the reduction in second moments of particle properties (artificial "mixing"). This annihilation scheme does not exactly preserve the particle property distribution. However, it has the advantage of conserving species mass and other properties at the "microscopic" (discrete event) level.

5.4. Mean estimation

A fundamental issue in hybrid particle/FV PDF methods (although not of direct relevance here) is the estimation of mean quantities from noisy particle data. Several algorithms based on the same trilinear basis functions that are used in the present mass consistency algorithm have been presented in [4].

5.5. Sorting

A significant reduction in computational time required for particle operations is realized by sorting and reindexing particles by their host FV element on each computational timestep. In the current mass consistency algorithm, for example, one then can form the required correction velocities element-by-element (versus particle-by-particle). This also facilitates parallelization using an element-based domain decomposition.

5.6. Summary

The sequence of operations for a single FV computational timestep is as follows: Mesh deformation and updates to FV geometric information are performed first; the particle solution then is advanced over the

timestep: and finally, the underlying FV hydrodynamic solution is performed. Here the FV algorithm is first order in time. The mean velocity field (Stages 1 and 2) for particle advection uses the most recently available FV information (i.e., from the end of the previous timestep). And the Stage 3 correction velocity field similarly is based on the most recently available particle/FV mass residuals (post particle advection). In practice, Stages 2 and 3 are applied simultaneously. Higher order in time would require multiple particle steps for each FV timestep.

The computational overhead for the mass consistency algorithm essentially is that of performing one additional pressure correction in the underlying FV solver. The additional cost in particle operations is negligible. This overhead can be reduced further through judicious specification of convergence tolerances in the iterative implicit linear equation solvers that are used for the mass-consistency pressure correction; in general, it is not necessary to converge the pressure correction equation to the same accuracy as is required for the FV hydrodynamic solution. Moreover, with the mass consistency algorithm in place it has been found that the time spent in particle-number-density-control operations (cloning/annihilation) is reduced significantly, and solutions can be obtained using lower particle number densities and larger computational timesteps than otherwise would be required. Therefore, the net effect generally is a *reduction* in CPU time compared to performing a hybrid particle/FV solution without the mass consistency algorithm. Timing information is discussed for one test case in the following section.

6. Test cases

To illustrate the importance of mass consistency in hybrid particle/FV PDF methods and the robustness and accuracy of the new algorithm, results for several flow configurations are presented. In all cases, second-order accurate spatial discretizations are used in the underlying FV solver and first-order particle advection schemes are used. The latter serves to emphasize the robustness of the approach. The focus is on differences between the FV mass distribution and the particle mass distribution at the element level.

6.1. Lid-driven cavity

This constant-property two-dimensional steady-flow configuration has long served as a benchmark validation case for CFD. The computational domain is an $L \times L$ square in x and y with zero velocity at the bottom $(y/L = 0)$ and side $(x/L = 0, x/L = 1)$ walls and with the top wall $(y/L = 1)$ driven tangentially to the right at a constant speed of U_W (Fig. 5). The key parameter is the Reynolds number $Re_W \equiv \rho U_W L/\mu$, where μ is the constant fluid viscosity. Here a laminar case with $Re_W = 1000$ is considered (no turbulence model); benchmark numerical data for this Reynolds number can be found in [39] and CFD results obtained using the present FV algorithm have been reported in [32]. Appropriate velocity, length and time scales are $\mathscr{V} = U_W$, $\mathscr{L} = L$ and $\mathscr{T} = L/U_W$.

The lid-driven cavity may appear to be remote from the three-dimensional time-dependent arbitrary-geometry configurations that are the ultimate target for the algorithm. In fact, this is a deceptively challenging and revealing configuration for a Lagrangian particle method. The flow is characterized by steep velocity gradients, no-slip walls, recirculation and ambiguity in the specification of the velocity at the intersections of the fixed and driven walls. Moreover, this configuration serves to validate the non-trivial requirement that the algorithm should yield a stationary particle mass distribution in steady flow.

Tortuous computational meshes make the problem even more challenging (Fig. 5). Four mesh topologies have been employed: uniform Cartesian meshes; "Z"-meshes, random meshes and mixed-element meshes. For the first three mesh types, all elements are non-degenerate six-faced hexahedra and there is a single layer of elements in the z direction. For the mixed-element meshes, the interior FV elements are five-faced prisms that have been generated from a random hexahedral mesh by splitting each interior

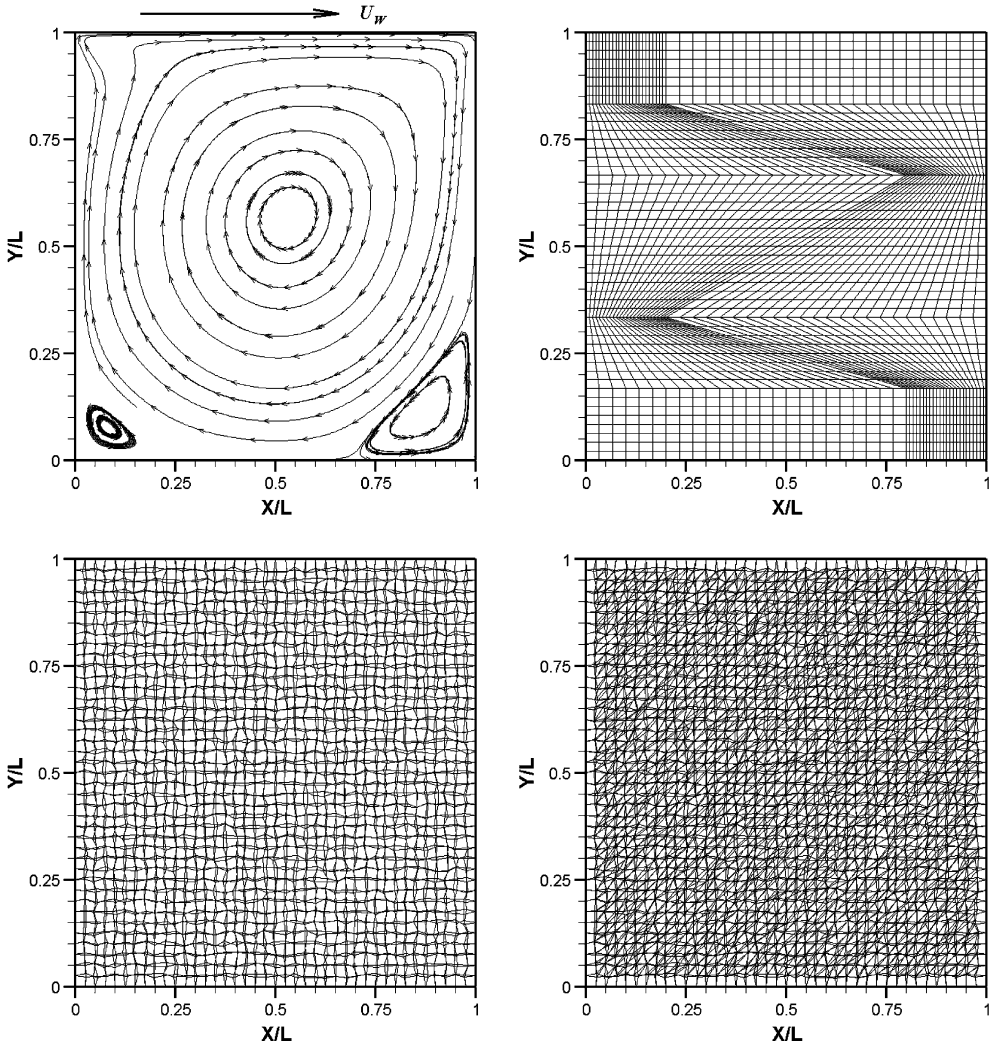

Fig. 5. Computational configuration and finite-volume meshes for the lid-driven cavity. Upper left: Computed steady state streamlines on a 6400-element uniform Cartesian mesh. Upper right: A 2304-element Z-mesh. Lower left: A 1600-element random mesh with $\alpha = 0.5$. Lower right: A 3044-element mixed-element mesh based on a 40×40, $\alpha = 0.5$ random mesh. For the two lower meshes, element faces on both $z =$ constant planes are shown.

hexahedral element into two prisms. While structured meshes would suffice for the first three mesh types, here vertex and element id's have been randomized (in some cases) to ensure that the unstructured-mesh pointer structures are fully exercised. Examples of the computational meshes employed (mesh type, number of elements N_E and ratio of maximum to minimum element volume $V_{\max}/V_{\min}$) are provided in Table 1. While element volumes do not vary greatly, departures from orthogonality are significant for the non-uniform meshes. Meshes having considerably larger element volume ratios (exceeding a factor of 10^5) are considered in later test cases.

The Z-mesh (e.g., [40]) and random mesh (e.g., [41,42]) both have been employed extensively as archetypical non-orthogonal meshes in the development and validation of CFD algorithms. For the random

Table 1
Lid-driven cavity examples

Case	$N_E(\alpha)$[a]	V_{max}/V_{min}	C^+_{glob}	N_{pe} min/nom/max	σ^+	ρ'_{rms}@ $t^+ = 10$		
						dp^k_{max}@ $t^+ = 10$		
						Stage 1	Stage 2	Stage 3[b]
U1	400	1.0	1.0	30/40/80	0.5	2.90 47.2	0.364 1.92	0.0997 0.296
U2	1600	1.0	1.0	30/40/80	0.5	3.96 153.0	0.430 4.66	0.0991 0.394
U3	6400	1.0	1.0	30/40/80	0.5	4.48 354.	0.624 14.1	0.103 0.420
Z1	2304	4.0	0.2	30/40/80	0.5	2.23 98.5	0.297 3.39	0.0744 0.389
Z2	2304	4.0	1.0	30/40/80	0.5	2.51 83.7	0.677 17.1	0.120 0.545
Z3	2304	4.0	5.0	30/40/80	0.5	2.97 117.	1.57 48.0	0.151 0.742
R1	1600 (0.5)	2.6	0.1	30/40/80	0.5	3.00 117.	0.199 0.998	0.0486 0.199
R2	1600 (0.5)	2.6	1.0	30/40/80	0.5	4.40 173.	0.454 4.16	0.104 0.527
R3	1600 (0.5)	2.6	5.0	30/40/80	0.5	4.59 179.	0.828 11.9	0.139 0.975
M1	3044 (0.5)	6.5	1.0	20/40/80	0.5	2.39 125.	0.313 5.61	0.141 0.729
M2	3044 (0.5)	6.5	1.0	40/80/160	0.5	1.69 89.8	0.182 2.73	0.068 0.479
M3	3044 (0.5)	6.5	1.0	80/160/320	0.5	1.96 99.2	0.178 2.18	0.0524 0.345

All timescales are equal to the computational timestep Δt. Case names beginning with "U" are uniform meshes; with "Z" are Z-meshes; with "R" are random meshes; and with "M" are mixed-element (prism/hex) meshes.

[a] α is relevant only for random meshes and for mixed-element meshes (Eq. (25)).

[b] A stationary distribution is achieved only with Stage 3.

mesh, the (x, y) location of the ijth interior vertex ($2 \leqslant i \leqslant N_x - 1, 2 \leqslant j \leqslant N_y - 1$, with $N_x = N_y = N_E^{1/2} + 1$) is given by,

$$x_{ij} = \frac{i - 1 + \alpha(\eta_{ij,x} - 0.5)}{N_x - 1}, \qquad y_{ij} = \frac{j - 1 + \alpha(\eta_{ij,y} - 0.5)}{N_y - 1}, \tag{25}$$

where $\eta_{ij,x}$ and $\eta_{ij,y}$ each are sampled from a random distribution uniform on [0., 1.]. The parameter α controls the mesh quality; a uniform orthogonal mesh corresponds to $\alpha = 0$, and mesh quality deteriorates with increasing α. Moreover, different random numbers have been used to generate the meshes on the two $z =$ constant planes so that the mesh distortion is three-dimensional, even though the flow is two-dimensional. A mixed-element mesh is included to demonstrate that the algorithm as derived applies immediately to degenerate hexahedra, as discussed in Appendices A and B.

A converged steady solution first is obtained using the FV solver and a time-marching algorithm. An ensemble of computational particles having approximately uniform particle number density in physical space is initialized such that the mass of particles in each FV element is equal to the FV element mass; this is taken as time $t = 0$. Particle positions and velocities then are advanced in time according to,

$$x_i^*(t + \Delta t) = x_i^*(t) + U_i^*(t)\Delta t, \qquad U_i^*(t) = U_{i,\mathrm{FV}}^* + \frac{1}{2}\frac{\partial |\underline{U}_{\mathrm{FV}}^*|^2}{\partial x_i}\sigma_U^2 + |\underline{U}_{\mathrm{FV}}^*|\sigma_U \eta_i / \Delta t^{1/2}. \tag{26}$$

Here $\underline{U}_{\mathrm{FV}}^* \equiv \underline{U}_{\mathrm{FV}}(\underline{x}^*(t))$ where $\underline{U}_{\mathrm{FV}}(\underline{x})$ is obtained via interpolation from the FV-computed velocity field, $\underline{\eta}$ is a vector of three independent identically distributed random numbers sampled from a standardized Gaussian distribution (zero mean, unit variance), and σ_U is a "noise" parameter for the particle velocity distribution; $(|\underline{U}_{\mathrm{FV}}^*|\sigma_U)^2$ is an effective diffusivity. For $\sigma_U = 0$, particles move as fluid particles in the limit $\Delta t \to 0$ (the particle tracking limit). There is no feedback of particle-based quantities into the FV computation.

The particle mass distribution is monitored as a function of normalized time, $t^+ \equiv t/\mathscr{T} = tU_W/L$. Ideally, the particle mass distribution should remain equal to the initial uniform mass distribution. The principal quantity of interest then is the deviation between the element-level particle mass density ρ_P^k (particle mass in element k, divided by the element volume) and the FV mass density ρ_{FV}^k (here $\rho_{\mathrm{FV}}^k \equiv 1$, $k = 1, \ldots, N_E$):

$$\mathrm{d}\rho^k \equiv \frac{\rho_P^k - \rho_{\mathrm{FV}}^k}{\rho_{\mathrm{FV}}^k}. \tag{27}$$

A global indicator of the deviation between the particle and FV mass distributions is the root-mean-square value of $\mathrm{d}\rho^k$ over the computational domain:

$$\rho'_{\mathrm{rms}} \equiv \left[\frac{\sum_{k=1}^{N_E} (\mathrm{d}\rho^k)^2}{N_E}\right]^{1/2}. \tag{28}$$

A systematic study has been carried out to quantify the effects of variations in mesh density (number of FV elements N_E), mesh quality (non-uniformity of element volumes and departures from orthogonality), number of computational particles N_P, computational time step Δt and velocity noise level σ_U. Key parameters are: the nominal, minimum and maximum number of particles per element $N_{pe,\mathrm{nom}} \equiv N_P/N_E$,

$N_{pe,\min}$ and $N_{pe,\max}$; the normalized particle velocity noise level $\sigma^+ \equiv \sigma_U/\Delta t^{1/2}$ and a global "particle Courant number",

$$C^+_{\text{glob}} \equiv U_W \Delta t N_E^{1/2}/L, \tag{29}$$

normalized mesh size,

$$h^+ \equiv L/N_E^{1/2}, \tag{30}$$

time-averaging parameters τ_{S3} (Eq. (22)), $\tau_{\text{avg},P}$ (Eq. (24)) and $\tau_{S3'}$ ($\tau_{\text{avg,FV}}$ is not relevant for this steady flow); and mesh type (Table 1). Unless otherwise specified, $\tau_{S3} = \tau_{\text{avg},P} = \tau_{S3'} = \Delta t$ (the computational time step).

Example results for a highly distorted Z-mesh (Fig. 5) are given in Figs. 6 and 7. There $N_E = 2304$ ($h^+ = 0.02083$), $N_{pe,\text{nom}} = 40$, $N_{pe,\min} = 30$, $N_{pe,\max} = 80$, $C^+_{\text{glob}} = 1.0$ and $\sigma^+ = 0.5$.

Time evolution of the instantaneous rms mass density deviation (Eq. (28)) is plotted in Fig. 6; instantaneous fields of the element-level mass density deviation (Eq. (27)) at time $t^+ = 10$ are shown in Fig. 7. Results are shown for different specifications of the particle mean advection velocity: no correction (Stage 1 only), first-level velocity correction (Stages 1 and 2) and second-level position correction (Stages 1–3). Several observations can be made. First, in the absence of any correction the deviation between particle and FV mass distributions increases continuously in time; particles accumulate in the upper right-hand corner of the domain near the intersection of the driven and fixed walls, where velocity gradients are steep and vertex velocity specification is ambiguous. Second, activation of Stage 2 reduces the deviation by more than one order of magnitude; however, the deviation continues to increase in time. And third, activation of the Stage 3 particle position correction reduces the deviation even further; equally significant, the deviation becomes statistically stationary, as it should for this steady flow.

Qualitatively similar results have been found for all cases investigated (Table 1). Figs. 8 and 9, for example, show the time evolution of the instantaneous rms mass density deviation with variations in global particle Courant number C^+_{glob} and particle number density $N_{pe,\text{nom}}$ for a random mesh ($N_E = 1600$, $\alpha = 0.5$,

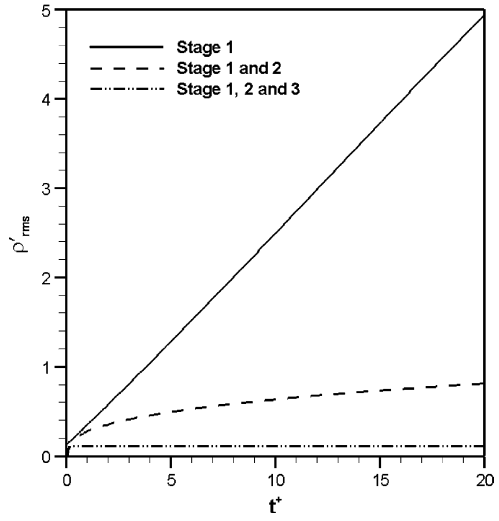

Fig. 6. Time evolution of instantaneous rms mass density deviation over the computational domain ρ'_{rms} (Eq. (28)) for the lid-driven cavity using the 2304-element Z-mesh of Fig. 5. Parameter values correspond to Case Z2 in Table 1 ($N_E = 2304$, $C^+_{\text{glob}} = 1.0$, $N_{pe,\text{nom}} = 40$, $\sigma^+ = 0.5$).

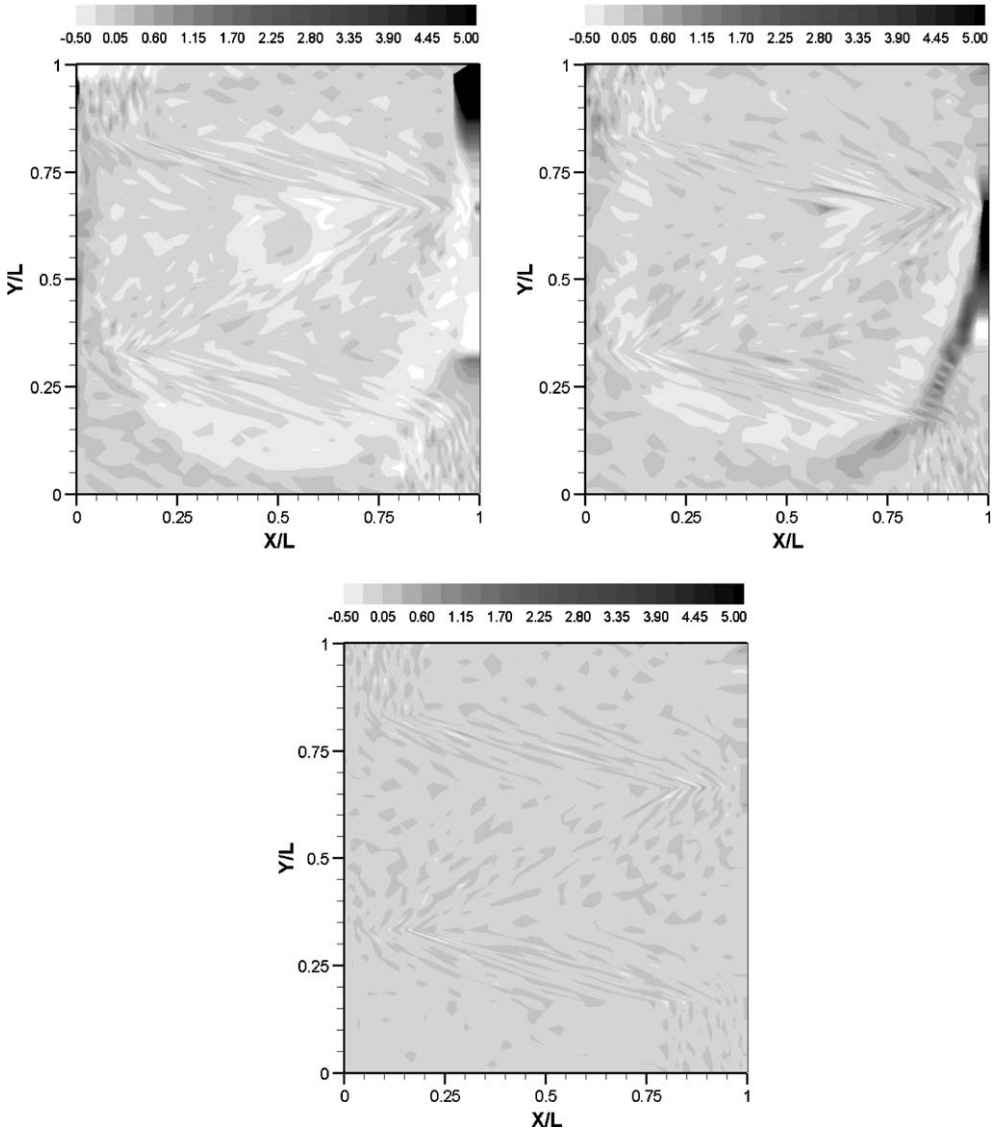

Fig. 7. Instantaneous element-level mass density deviation $\mathrm{d}\rho^k$ (Eq. (27)) at time $t^+ = 10$ for the lid-driven cavity using the 2304-element Z-mesh of Fig. 5. Parameter values correspond to Case Z2 in Table 1. Upper left: Stage 1. Upper right: Stages 1 and 2. Lower: Stages 1, 2 and 3.

$N_{pe,\mathrm{nom}} = 40$, $\sigma^+ = 0.5$) and for a mixed-element mesh ($N_E = 3044$, $\alpha = 0.5$, $C^+_{\mathrm{glob}} = 1.0$, $\sigma^+ = 0.5$), respectively. Again, stationary distributions are realized only with the Stage 3 correction, and the magnitude of the deviation decreases with decreasing C^+_{glob} and with increasing $N_{pe,\mathrm{nom}}$.

Further examples illustrating the behavior of the rms and maximum instantaneons mass density deviation with variations in global particle Courant number C^+_{glob} (Eq. (29)), particle velocity noise level σ^+ (Eq. (26)), particle number density $N_{pe,\mathrm{nom}}$ and relative mesh size h^+ (Eq. (30)) are provided in Fig. 10. These results have been obtained for uniform meshes; similar results are found for all mesh types (Table 1). Key findings are as follows. First, in the absence of the Stage 3 particle position correction, mass deviations

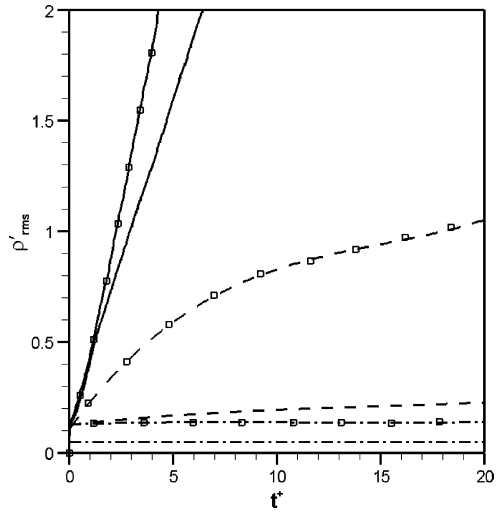

Fig. 8. Effect of variations in global particle Courant number C^{+}_{glob} on evolution of instantaneous rms mass density deviation over the computational domain ρ'_{rms} (Eq. (28)) for the lid-driven cavity using a random mesh. Parameter values correspond to Cases R1 and R3 in Table 1 ($N_E = 1600,\ \alpha = 0.5$, $N_{pe,\mathrm{nom}} = 40$, $\sigma^{+} = 0.5$). Solid lines: Stage 1. Dashed lines: Stages 1 and 2. Dash-dot lines: Stages 1, 2 and 3. Lines without symbols: $C^{+}_{\mathrm{glob}} = 0.1$. Lines with symbols: $C^{+}_{\mathrm{glob}} = 5.0$.

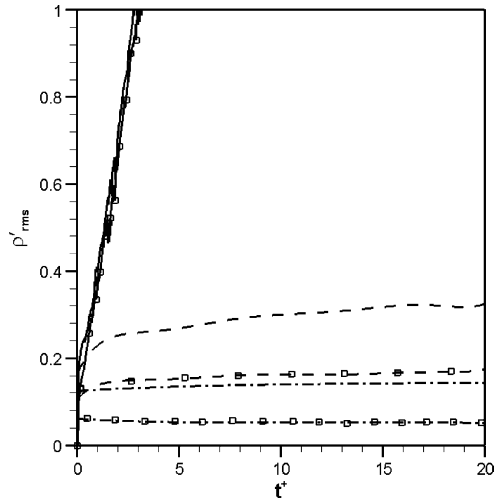

Fig. 9. Effect of variations in particle number density $N_{pe,\mathrm{nom}}$ on evolution of instantaneous rms mass density deviation over the computational domain ρ'_{rms} (Eq. (28)) for the lid-driven cavity using a mixed-element mesh. Parameter values correspond to Cases M1 and M3 in Table 1 ($N_E = 3044$, $\alpha = 0.5$, $C^{+}_{\mathrm{glob}} = 1.0$, $\sigma^{+} = 0.5$). Solid lines: Stage 1. Dashed lines: Stages and 2. Dash-dot lines: Stages 1, 2 and 3. Lines without symbols: $N_{pe,\mathrm{nom}} = 40$. Lines with symbols: $N_{pe,\mathrm{nom}} = 160$.

develop and grow in time even in the limit $\Delta t \to 0$ and $N_{pe,\mathrm{nom}} \to \infty$. This results from non-uniformities in the particle mass density distribution within FV elements (sub-grid-scale non-uniformity). Second, stationary distributions are realized only with the Stage 3 particle position correction, and the mass density deviation varies most strongly with variations in C^{+}_{glob} and $N_{pe,\mathrm{nom}}$. And third, with the Stage 3 particle

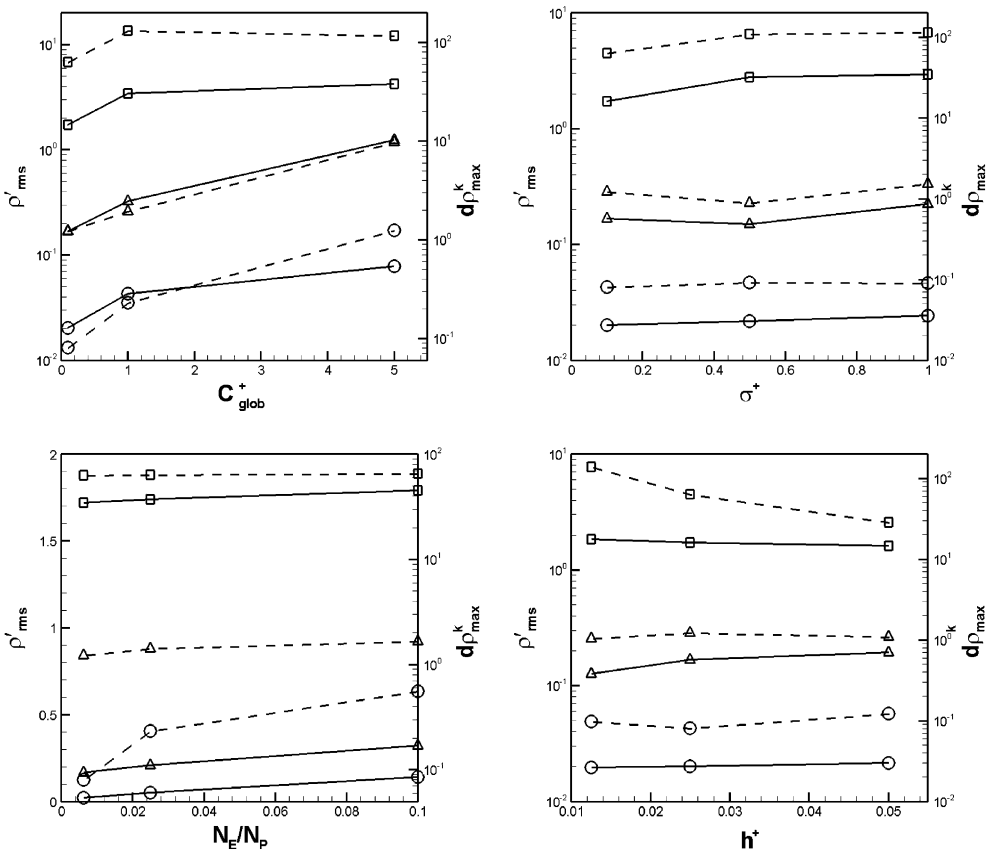

Fig. 10. Instantaneous global rms mass density deviation ρ'_{rms} (Eq. (28)) and maximum element-level mass density deviation $\mathrm{d}\rho^k_{max}$ (Eq. (27)) at time $t^+ = 10$ for the lid-driven cavity using a uniform mesh. Upper left: Variations in global particle Courant number C^+_{glob} (Eq. (29)) with $N_E = 1600$, $N_{pe,\mathrm{nom}} = 160$, $\sigma^+ = 0.1$. Upper right: Variations in particle velocity noise level σ^+ Eq. (26) with $N_E = 1600$, $C^+_{\mathrm{glob}} = 0.1$, $N_{pe,\mathrm{nom}} = 160$. Lower left: Variations in particle number density $N_{pe,\mathrm{nom}}$ with $N_E = 1600$, $C^+_{\mathrm{glob}} = 0.1$, $\sigma^+ = 0.1$. Lower right: Variations in relative mesh size h^+ with $C^+_{\mathrm{glob}} = 0.1$, $N_{pe,\mathrm{nom}} = 160$, $\sigma^+ = 0.1$. Squares: Stage 1. Triangles: Stages 1 and 2. Circles: Stages 1, 2 and 3. Solid lines: rms (left scale). Dashed lines: maximum (right scale).

position correction activated, the element-level mass density deviation can be driven to arbitrarily small values for any h^+ and σ^+ in the limit $C^+_{\mathrm{glob}} \to 0$ and $N_{pe,\mathrm{nom}} \to \infty$.

In the cases examined so far, the timescales τ_{S3}, $\tau_{\mathrm{avg},P}$ and $\tau_{\mathrm{S3}'}$ have been equal to the computational time step Δt. As discussed earlier, this corresponds to forcing the *instantaneous* particle mass distribution to remain consistent with the FV mass distribution at the element level, and goes beyond the formal consistency requirement that applies only to the *mean* particle mass distribution. The effects on mass density deviation and on the magnitudes of the correction velocities of variations in each of these several timescales have been explored; one example is provided in Fig. 11. There the time evolution of the instantaneous rms mass deviation is plotted for a coarse uniform mesh ($N_E = 400$, $\sigma^+ = 0.0$, $C^+_{\mathrm{glob}} = 1.0$, $N_{pe,\mathrm{nom}} = 160$) with variations in $\tau_{\mathrm{avg},P}$ and $\tau_{\mathrm{S3}'}$ ($\tau_{\mathrm{S3}} = \Delta t$). It can be seen that variations in $\tau_{\mathrm{avg},P}$ have little effect, while the mass density deviation increases with increasing $\tau_{\mathrm{S3}'}$.

In general, the benefits in accuracy and computational efficiency resulting from variations in the several algorithm time scales have been found to be small. Henceforth, all timescales are taken to be equal to the computational time step Δt.

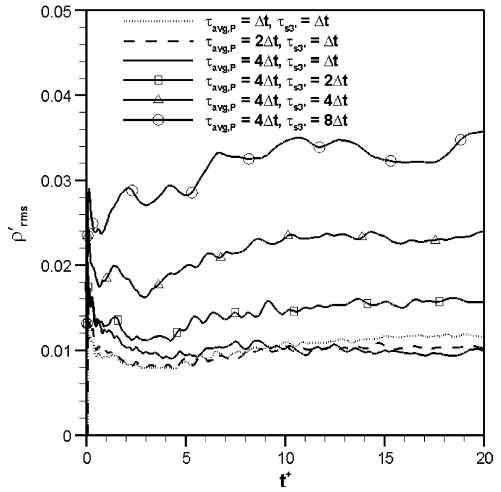

Fig. 11. Time evolution of instantaneous rms mass density deviation over the computational domain (Eq. (28)) for the lid-driven cavity using a 400-element uniform mesh ($N_E = 400$, $C^+_{glob} = 1.0$, $N_{pe,nom} = 160$, $\sigma^+ = 0.0$) with variations in $\tau_{avg,P}$ $\tau_{S3'}$.

Velocity correction magnitudes are plotted in Figs. 12 and 13. Normalized particle velocity correction magnitudes for Stages 2 and 3 are defined as,

$$v^*_{corr,2} \equiv |\underline{\hat{U}}|^*/U_W, \quad v^*_{corr,3} \equiv |\underline{\hat{\hat{U}}}|^*/U_W. \tag{31}$$

Maximum and rms values over the computational domain then are computed:

$$v^*_{3,max} \equiv \max_{i=1,\ldots,N_P} \{v^{(i)}_{corr,3}\}, \quad v^*_{3,rms} \equiv \left[\frac{\sum_{i=1}^{N_P}\left(v^{(i)}_{corr,3}\right)^2}{N_P}\right]^{1/2} \tag{32}$$

and similarly for Stage 2.

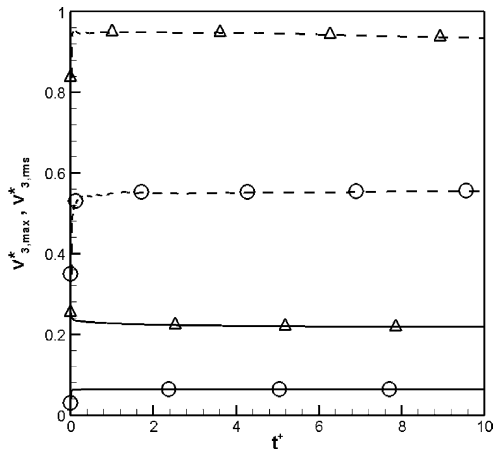

Fig. 12. Time evolution of maximum and rms Stage 3 normalized velocity corrections over the computational domain (Eq. (32)) for the lid-driven cavity using a 1600-element uniform mesh ($N_E = 1600$, $C^+_{glob} = 0.1$, $N_{pe,nom} = 160$, $\sigma^+ = 0.1$). Circles: Stage 2 activated. Triangles: Stage 2 deactivated. Solid lines: rms values. Dashed lines: maximum values.

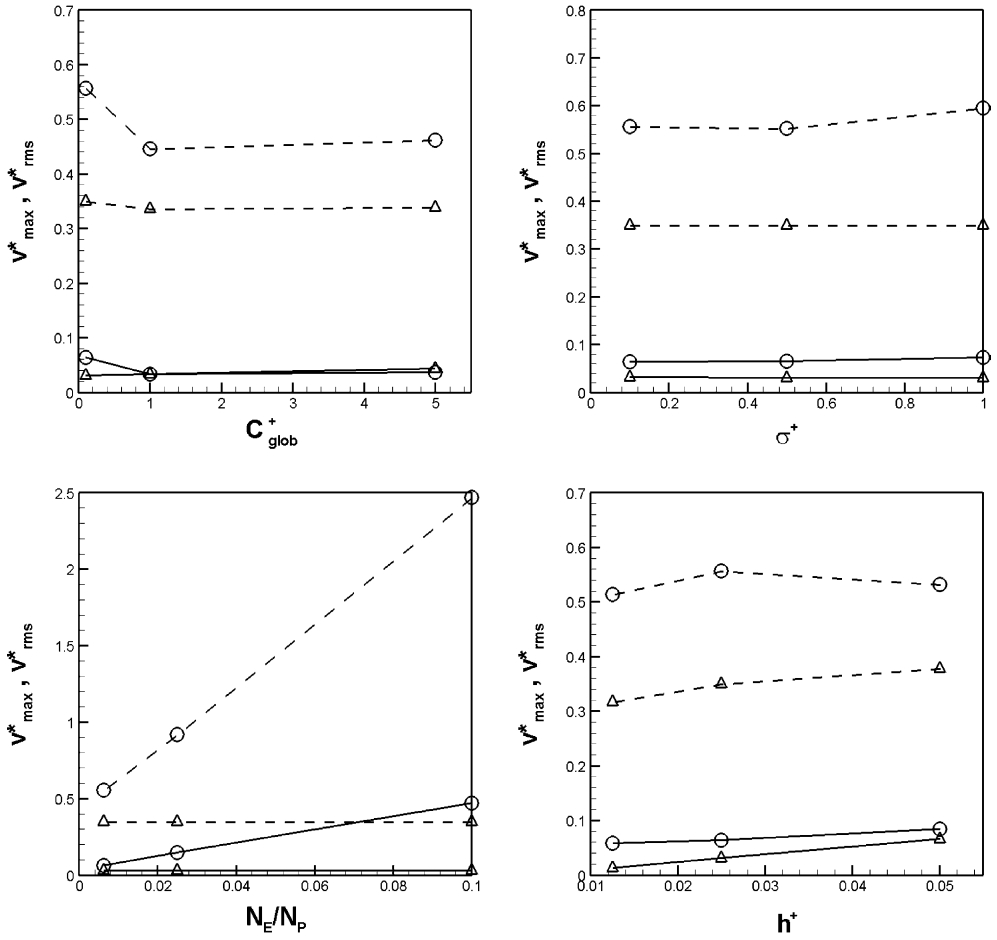

Fig. 13. Instantaneous global maximum and rms normalized velocity correction magnitudes (Eq. (32)) at time $t^+ = 10$ for the lid-driven cavity using a uniform mesh. Upper left: Variations in global particle Courant number C^+_{glob} Eq. (29) with $N_E = 1600$, $N_{pe,\text{nom}} = 160$, $\sigma^+ = 0.1$. Upper right: Variations in particle velocity noise level σ^+ (Eq. (26)) with $N_E = 1600$, $C^+_{\text{glob}} = 0.1$, $N_{pe,\text{nom}} = 160$. Lower left: Variations in particle number density $N_{pe,\text{nom}}$ with $N_E = 1600$, $C^+_{\text{glob}} = 0.1$, $\sigma^+ = 0.1$. Lower right: Variations in relative mesh size h^+ with $C^+_{\text{glob}} = 0.1$, $N_{pe,\text{nom}} = 160$, $\sigma^+ = 0.1$. Circles: Stage 3 correction magnitude. Triangles: Stage 2 correction magnitude. Solid lines: rms values. Dashed lines: maximum values.

Fig. 12 shows that the required Stage 3 velocity correction magnitude is considerably lower with Stage 2 in place. Moreover, the steady-state rms mass density deviation is approximately 10% lower with Stage 2 active, compared to applying the Stage 3 correction directly to the Stage 1 velocity field without Stage 2 (not shown). The variations in the maximum and rms Stage 2 and Stage 3 normalized correction velocity magnitudes with changes in global particle Courant number, particle velocity noise level, particle number density and relative mesh size are shown in Fig. 13. Particle number density has the largest effect; the Stage 3 correction velocity decreases monotonically as the number of particles per element increases. For fixed particle number density, the required correction velocities increase for very small particle Courant numbers.

6.2. A variable-density free jet

Jets of freon-12 (dichlorodifluoromethane) into air were studied by Arcoumanis et al. [43] to investigate fundamental aspects of turbulent mixing with large density variations, for application to in-cylinder fuel injection in reciprocating-piston internal-combustion engines. Rayleigh scattering was used to extract ensemble-averaged (in a simplified engine configuration) or time-averaged (in a free jet configuration) profiles of freon mole fraction mean and variance. This configuration has been the subject of one earlier modeling study [4]. Here the freon/air jet serves to exercise the mass consistency algorithm in a configuration that has inflow/outflow boundaries and large spatial variations in mass density that are comparable to those encountered in a practical turbulent combustion system.

Freon vapor is injected at 3.9 m/s into stagnant air through a 10 mm diameter orifice. The computational configuration is a 10° wedge (axisymmetric) mesh of 3968 non-uniform elements, with a, single element in the azimuthal direction. The ratio of maximum to minimum element volume is 1.9×10^4. A time-marching algorithm is used to obtain a stationary solution to the governing equations. Results have been obtained using two different physical models: a moment closure (modeled equation for freon mass fraction variance), and a composition PDF method; in both cases, a standard $k-\varepsilon$ model has been used for turbulent transport. The two models are almost (but not exactly) equivalent at the second-moment level; details are provided in Subramaniam and Haworth [4]. Here approximately 200 K particles have been used for the composition PDF calculations ($N_{pe,\mathrm{nom}} \approx 50$): a simple pair-exchange stochastic mixing model has been used to simulate mixing [2] ($\mathrm{d}\underline{\phi}^*_{\mathrm{mix}}$ term in Eq. (6)); and there is no chemical reaction.

The mixture mass density and specific heats are prescribed assuming an ideal-gas mixture of freon and air. With Y_α, χ_α and W_α denoting the mass fraction, mole fraction and molecular weight, respectively, for species α, the following relations pertain:

$$\begin{aligned} &\chi_\alpha = Y_\alpha W/W_\alpha, \quad Y_\alpha = \chi_\alpha W_\alpha/W, \\ &W = \sum_{\alpha=1}^{N_S} W_\alpha \chi_\alpha = \left[\sum_{\alpha=1}^{N_S} Y_\alpha/W_\alpha\right]^{-1}, \quad \rho = pW(RT)^{-1}. \end{aligned} \tag{33}$$

Here W is the mixture molecular weight, R is the universal gas constant and $W_{\mathrm{freon}}/W_{\mathrm{air}} = 4.19$. Neglecting the influence of pressure fluctuations on mixture mass density (appropriate for low Mach number), the mean mixture mass density $\langle\rho\rangle$ is given by,

$$\langle\rho\rangle = pR^{-1}\langle W/T\rangle. \tag{34}$$

Since the present focus is on particle/FV mass consistency and not on the treatment of the energy equation, we further neglect the influence of temperature fluctuations on mixture mean mass density. The equation of state used to prescribe the mean mass density is then,

$$\langle\rho\rangle = p(RT)^{-1}\langle W\rangle, \tag{35}$$

where $p(=\langle p\rangle)$ and $T(=\tilde{T})$ are computed using the FV algorithm and $\langle W\rangle$ is evaluated based on particle compositions. The mixture specific heat is computed as,

$$c_p = Y_{\mathrm{freon}} c_{p,\mathrm{freon}} + Y_{\mathrm{air}} c_{p,\mathrm{air}} \tag{36}$$

with $c_{p,\mathrm{freon}} = 601$ J/kg K and $c_{p,\mathrm{air}} = 1009$ J/kg K. In the moment closure, fluctuations in molecular weight effectively are neglected.

Fig. 14 shows computed steady-state freon mole fraction variance contours for the moment closure and the PDF models. The composition PDF yields higher peak variance, and steeper radial gradients. The

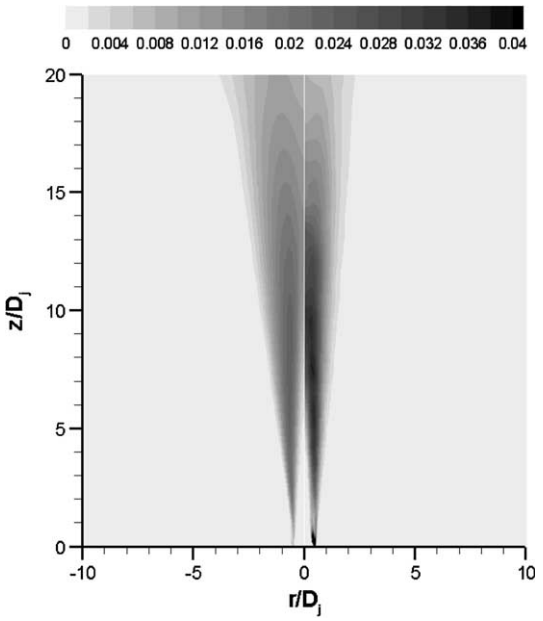

Fig. 14. Computed contours of neon mole fraction variance using a moment closure (left: modeled variance equation) and a composition PDF method (right: Stages 1, 2 and 3). The PDF results have been time averaged to reduce statistical noise.

variance is a key quantity in non-premixed turbulent combustion. In the case of a chemically reacting turbulent jet flame, for example, the mean reaction rate would be (approximately) proportional to the fuel mass fraction variance.

A quantitative comparison between computations and measurements is provided in Fig. 15. There the quantity plotted is the rms freon mole fraction, normalized by the local mean value: $\widetilde{x''^2_{\text{freon}}}^{1/2}(z)/\widetilde{\chi_{\text{freon}}}(z)$.

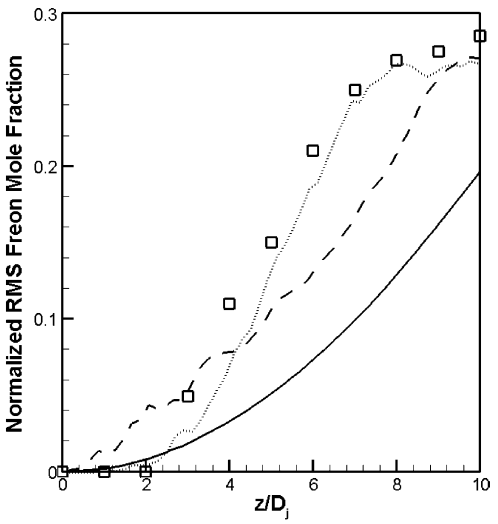

Fig. 15. Computed (lines) and measured, (symbols; [43]) centerline profiles of normalized rms freon mole fraction. Solid line: moment closure. Dashed line: composition PDF method with Stage 1 only. Dotted line: composition PDF method with Stages 1, 2 and 3. Here no time averaging has been performed for the PDF results.

The PDF method yields a higher variance that is closer to the experimentally measured value, compared to the moment closure. The effect of the mass consistency algorithm also is evident in Fig. 15. In the absence of any corrections (Stage 1 only), the PDF model fails to capture the "potential core" of the freon jet close to the nozzle exit.

The moment closure solution can be brought closer to the experimental measurements with further grid refinement, while the PDF results shown are nearly grid independent. This serves to illustrate an important advantage of the Lagrangian particle approach: numerical dissipation is inherently low and accurate solutions can be obtained using coarser computational meshes than those required for an equivalent moment closure.

Finally, the global rms mass density deviation over the computational domain (Eq. (28)) is plotted as a function of time in Fig. 16. There $t^+ = tU_j/D_j$, where U_j and D_j are the freon nozzle velocity and diameter, respectively. As was seen earlier for the lid-driven cavity, the error is much lower with the corrections active; and again, a stationary error distribution is realized only with Stage 3 active.

6.3. A simplified piston–cylinder assembly

As a final example, we consider turbulent flow in a motored (no combustion) piston–cylinder assembly that represents an idealized reciprocating-piston IC engine. The configuration studied experimentally by Morse et al. [44] (Fig. 17) has been the subject of several CFD modeling studies. Here a velocity PDF method is used to provide the effective turbulent stress (the Reynolds stress) terms that appear in the mean momentum equations (Eq. (11)). The present emphasis is particle/FV mass consistency; relevant aspects of the physical modeling and particle/FV coupling are discussed first for completeness.

A simplified Langevin model [45] has been used for particle velocities. This model corresponds to isotropic dissipation and Rotta's linear return-to-isotropy for the Reynolds stresses at the second-moment level. Formally, it is the PDF of the fluctuating velocity (the difference between the local instantaneous velocity and the local mean velocity) that is computed; see Muradoglu et al. [10], for example. In that case, the particle fluctuating velocity is advanced over a computational time step according to,

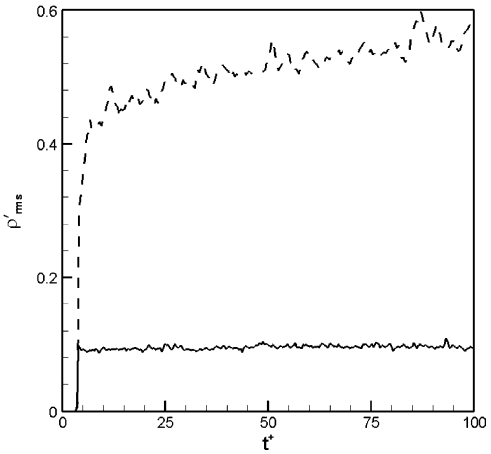

Fig. 16. Time evolution of instantaneous mis mass density deviation over the computational domain (Eq. (28)) for the freon jet. Dashed line: Stage 1 only. Solid line: Stages 1, 2 and 3.

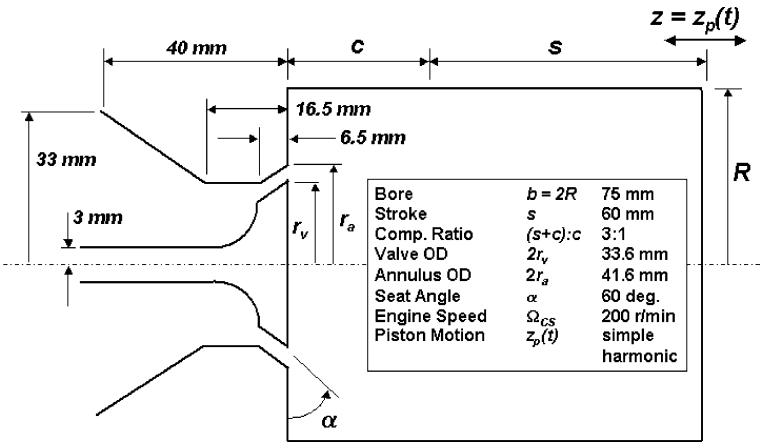

Fig. 17. The piston–cylinder assembly of Morse et al. [44].

$$u_i''^*(t+\Delta t) = u_i''^*(t) - \left(\frac{1}{\rho^*} - \frac{1}{\langle\rho\rangle^*}\right)\frac{\partial\langle p\rangle^*}{\partial x_i}\Delta t - \frac{\partial\tilde{U}_i^*}{\partial t}\Delta t - U_j^*\frac{\partial\tilde{U}_i^*}{\partial x_j}\Delta t - \left(\frac{1}{2}+\frac{3}{4}C_0\right)u_i''^*\Delta t/\tau + (C_0\varepsilon\Delta t)^{1/2}\eta_i. \tag{37}$$

Here $C_0 = 2.1$ is a model constant, $\underline{\eta}$ is a vector of three independent identically distributed random numbers sampled from a standardized Gaussian distribution (zero mean, unit variance) and $\tau = k/\varepsilon$ is a turbulence timescale. Again, this scheme is first-order in time. A standard k–ε model with wall functions is retained on the FV side to provide ε and the turbulence timescale τ [46]. More sophisticated velocity PDF models (e.g., models that account explicitly for ''rapid'' pressure terms [45]) and/or alternative forms of the ε equation are required to realize good quantitative agreement with experimental measurements in this configuration. No comparisons with experimental measurements are offered here.

The Reynolds stresses are estimated from particle values using the mean-estimation algorithm described by Subramaniam and Haworth [4]. A robust coupling strategy has been devised where the standard k–ε model implicit diffusion term is retained in the FV mean momentum equations; an explicit source term then is added to account for the difference between the particle-derived Reynolds-stress divergence and the k–ε diffusion term. This formally cancels the k–ε term, and ensures a well-conditioned matrix in the implicit momentum predictor (a ''deferred correction'' approach). This allows stable solutions to be obtained using fewer computational particles and larger computational timesteps than would be required for a direct implementation of the particle-derived Reynolds-stress divergence as a momentum-equation source term. Still, the number of particles required remains larger (by at least a factor of two) than that required for a composition PDF method.

Hybrid particle-mesh velocity PDF calculations have been performed for a variety of mesh densities and particle number densities using two-dimensional (planar) meshes, axisymmetric (wedge) meshes and coarse three-dimensional meshes. Example results at 90° after top-dead center on the intake stroke are shown in Fig. 18; these are instantaneous contours (no averaging). Here results have been obtained using a 5890-element 10° wedge mesh with a single element in the azimuthal direction. The ratio of maximum to minimum element volume is 2.5×10^5. The nominal particle number density is $N_{pe,\mathrm{nom}} \approx 120 (N_P \approx 700,000)$; particle number density control maintains the number of particles per element between 100 and 400. The computational timestep is 5.2083×10^{-5} s (1/16 crankangle degree at 200 r/min). Two engine cycles are computed and results are shown for the second engine cycle.

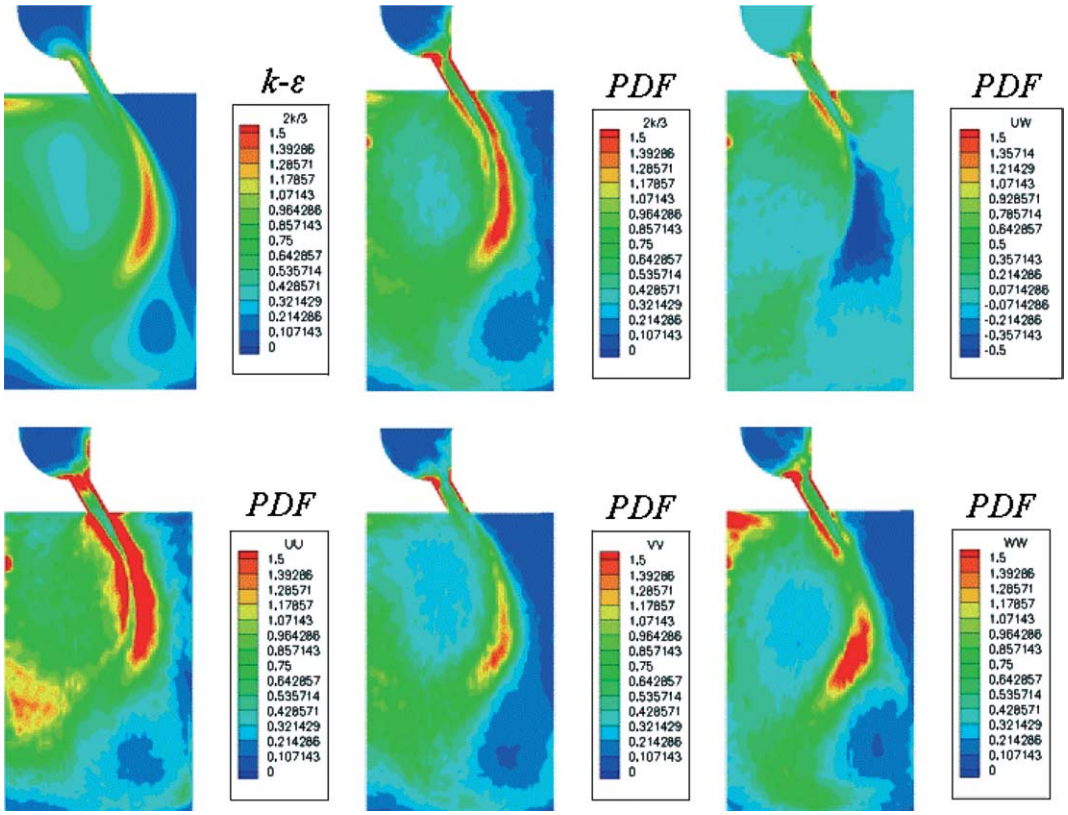

Fig. 18. Computed Reynolds stress components at 90° after piston top-dead center on the intake stroke for the simplified piston–cylinder assembly [44]. Upper left: $2k/3$, k–ε model. Upper center: 2k/3. velocity PDF model. Upper right: $\widetilde{u_r''u_z''}$, velocity PDF model. Lower left:$\widetilde{u_r''u_r''}$, velocity PDF model. Lower center: $\widetilde{u_\theta''u_\theta''}$ velocity PDF model. Lower right: $\widetilde{u_z''u_z''}$ velocity PDF model.

The PDF calculation shows strong peaks in k in the shear layers at both the inside and the outside of the annular intake jet, while a standard k–ε model yields a strong peak only at the outside. In general, turbulence levels from the PDF calculation tend to be higher than from k–ε. There is significant anisotropy in the PDF-computed normal stresses; mean-velocity-gradient production feeds the radial and axial components ($\widetilde{u_r''u_r''}$ and $\widetilde{u_z''u_z''}$) while the azimuthal component ($\widetilde{u_\theta''u_\theta''}$) relies solely on redistribution for its energy. And by contrast to k–ε, the PDF-based Reynolds stresses are realizable, by construction. Noise is evident in the PDF solution; in spite of this, the numerical solution holds together reliably, even with the present second-order centered advection scheme in the mean momentum equations.

Particle/FV mass consistency is examined by plotting the normalized deviation between the instantaneous in-cylinder particle mass $M_P(t)$ and the instantaneous in-cylinder FV fluid mass $M_{\mathrm{FV}}(t)$:

$$\epsilon_M = \epsilon_M(t) \equiv \frac{M_P(t) - M_{\mathrm{FV}}(t)}{M_{\mathrm{FV}}(t)}. \tag{38}$$

For the parameter set used to generate Fig. 18, a stable solution through a full engine cycle could not be obtained in the absence of the mass consistency algorithm. Results are shown instead for a coarse two-dimensional planar mesh in Fig. 19 (1593 elements, $N_{pe,\mathrm{nom}} < 100$, 1/8-crankangle-degree computational time step). The effect of the mass consistency algorithm is dramatic. In the absence of any correction (Stage

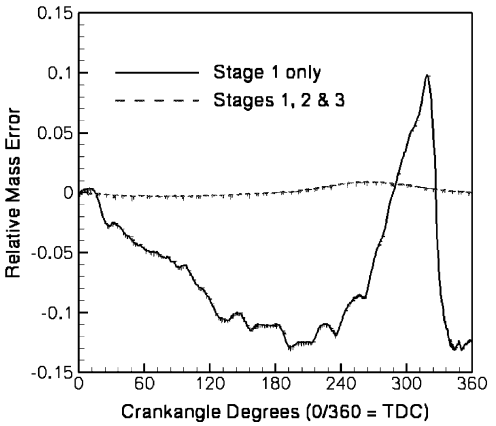

Fig. 19. Evolution of normalized difference between computed instantaneous in-cylinder particle mass and instantaneous in-cylinder finite-volume fluid mass (Eq. (38)) for the simplified piston–cylinder assembly [44].

1 only), the in-cylinder particle mass deviates from the fluid mass by more than 10% at piston bottom-dead-center (180°). This implies that if the intake valve were to be closed at this instant, the trapped particle mass representing fresh fuel–air mixture (in the case of a premixed-charge spark-ignition engine) would be in error by more than 10%; this would have a significant impact on the subsequent mixing and combustion event. With the Stage 3 correction in place, the mass deviation is reduced by approximately two orders of magnitude over most of the engine cycle. Stage 1 results would improve with higher-order particle time integration [34–36] and/or a smaller computational time step; here the maximum particle Courant number (Eq. (20)) exceeds 10. But this example using a low-order particle scheme and large computational time step serves well to illustrate the robustness of the mass consistency algorithm.

CPU times with and without the mass consistency algorithm are compared in Table 2; the total number of computational particles is the same in both cases (to within less than 1%). The computational overhead essentially is that of performing one additional pressure correction per timestep; that cost is more than compensated by reductions in computational effort in other parts of the code. There is a significant reduction in particle number density control, in particular. Fewer cloning and annihilation operations are required because the spatial distribution of particle mass/volume remains consistent with the local FV element mass/volume. For this velocity PDF example, there is a small decrease in the computational effort required to solve the FV pressure correction and mean momentum equations; that is a result of the smoother Reynolds stress

Table 2
Relative computational effort for Stages 1, 2 and 3 compared to Stage 1 alone

Description	Stages 1, 2 and 3/Stage 1 alone
CPU time for particle number density control	0.50
Total CPU time for particle side	0.76
CPU time for FV pressure correction equation[a]	0.99
CPU time for FV mean momentum equations	0.98
CPU time in FV linear equation solvers[b]	1.04
Total CPU time	0.87

Results are for a two-dimensional simplified piston–cylinder assembly (Fig. 17) using a velocity PDF method. There are 1593 FV elements, $N_{pe,\mathrm{nom}} \approx 100$, the computational time step corresponds to 1/16-crankangle-degree of rotation at an engine speed of 200 rpm and computations are from 360 to 540 crank-angle degrees.

[a] Excludes CPU time for the Stage 3 particle position correction.

[b] Includes CPU time for the Stage 3 particle position correction.

fields that feed back into the mean momentum equations with the mass consistency algorithm activated. The total time spent in the FV linear equation solvers (including the Stage 3 pressure correction) is slightly higher with the mass consistency algorithm activated. Overall, the algorithm yields significant improvement in accuracy (Fig. 19) with a net reduction in CPU time. Similar results have been found for other test cases.

7. Conclusion

The goal of this research has been to advance the accessibility of Lagrangian particle PDF methods for modeling chemically reacting turbulent flows in complex engineering configurations. Towards that end a robust, computationally efficient, accurate and general mass consistency algorithm has been developed that is compatible with existing FV CFD codes used in research and in engineering applications.

The approach is to maintain consistency between mean particle mass and FV fluid mass at the FV element level. Key features of the algorithm include the central role played by FV element-face mass flowrates, a multi-stage approach to minimize the particle position correction (spatial redistribution of particles) required in Stage 3, and direct use of the underlying FV solver's elliptic pressure correction scheme to compute a set of element-face mass flowrates that drives the difference between particle mass and fluid mass towards zero at the element level.

The algorithm has been designed primarily with six-faced (hexahedral) FV elements and segregated pressure-based CFD solvers in mind. However, it applies immediately to other element types (edge and face degeneracies of hexahedral elements, including tetrahedra). And alternative interpolation strategies (other than vertex-based trilinear basis functions) could be employed. In the case where a pressure correction scheme is not available in the underlying CFD code to compute the Stage 3 element-face mass flowrates, a separate elliptic equation could be implemented for that purpose (see [12], for example). The algorithm is intended primarily for time-accurate computations, but can be used for steady flows as well.

Low-order methods have been used here to integrate the Lagrangian particle equations. This serves to emphasize the robustness of the new algorithm. Higher-order particle schemes [34–36] can be used immediately.

The direct computational overhead for the present mass consistency algorithm is essentially that of performing one additional pressure correction step in the underlying FV CFD code. However, the pressure correction equation for Stage 3 generally need not be solved to the same accuracy as that required for the CFD flow solution itself. Moreover, with the mass consistency algorithm in place it has been found that the time spent in particle-number-density-control operations (cloning/annihilation) is reduced significantly, and solutions can be obtained using lower particle number densities and larger computational timestep than otherwise would be required. Therefore, the net effect is generally a *reduction* in CPU time compared to performing a hybrid particle/FV solution without the mass consistency algorithm.

Several parameters have been built into the algorithm to provide additional control and flexibility, in particular for steady flows: for example, the timescales τ_{S3} (Eq. (22)) $\tau_{avg,P}$ and $\tau_{avg,FV}$ (Eq. (24)), and $\tau_{S3'}$. In practice, little benefit in accuracy or efficiency has been found by setting any of these to be different from the computational time step Δt. Thus the current recommended practice is to enforce particle/FV mass consistency at the element level on each computational time step. Again, it is emphasized that this goes beyond the formal requirement that applies only to the *mean* particle mass distribution.

Acknowledgements

This research has been supported by National Science Foundation Grant No. 0121573, by the General Motors Research and Development Center Powertrain Systems Lab and by the CD adapco Group. The

first author also has been supported under a grant from GM to Penn State's College of Engineering, "GM Partners for Future Innovation".

Appendix A. Trilinear mapping

Element and face local numbering conventions are indicated in Fig. 3. Local (logical) element coordinates ξ_1, ξ_2 and ξ_3 $(0 \leqslant \xi_i \leqslant 1)$ are related to global coordinates x_1, x_2 and x_3 in element e through a trilinear mapping,

$$x_i = \sum_{v=1}^{8} b^{e,v}(\underline{\xi}) x_i^l \quad (i = 1, 2, 3), \tag{A.1}$$

where

$$\begin{aligned} &b^{e,1}(\underline{\xi}) = (1-\xi_1)(1-\xi_2)(1-\xi_3), \quad b^{e,2}(\underline{\xi}) = \xi_1(1-\xi_2)(1-\xi_3), \\ &b^{e,3}(\underline{\xi}) = \xi_1\xi_2(1-\xi_3), \quad b^{e,4}(\underline{\xi}) = (1-\xi_1)\xi_2(1-\xi_3), \\ &b^{e,5}(\underline{\xi}) = (1-\xi_1)(1-\xi_2)\xi_3, \quad b^{e,6}(\underline{\xi}) = \xi_1(1-\xi_2)\xi_3, \\ &b^{e,7}(\underline{\xi}) = \xi_1\xi_2\xi_3, \quad b^{e,8}(\underline{\xi}) = (1-\xi_1)\xi_2\xi_3. \end{aligned} \tag{A.2}$$

Here l is the global vertex index corresponding to local element vertex v. The same trilinear basis functions are used to establish continuous fields of computed mean dependent variables from vertex values (an *isoparametric* representation).

On each element face, one of the three logical coordinates has the value zero or one, and the trilinear mapping collapses to a bilinear mapping. Faces are not planar, in general. The outward-pointing face area projection vector is defined as,

$$\underline{A}^f \equiv \int_0^1 \int_0^1 \left(\frac{\partial \underline{x}}{\partial \eta_1} \times \frac{\partial \underline{x}}{\partial \eta_2} \right) \mathrm{d}\eta_1 \, \mathrm{d}\eta_2. \tag{A.3}$$

This is equivalent to one-half of the cross-product of two face diagonal vectors,

$$\underline{A}^f = \tfrac{1}{2} \overrightarrow{ac} \times \overrightarrow{bd}, \tag{A.4}$$

where $\overrightarrow{ac} = \underline{x}^c - \underline{x}^a$, for example. The face projection area is $A^f = (\underline{A}^f \cdot A^f)^{1/2}$ and the face outward-pointing unit normal vector $\underline{n}^f$ is,

$$\underline{n}^f = \underline{A}^f / A^f. \tag{A.5}$$

The face-averaged value of any quantity Φ is given by,

$$\overline{\Phi}^f \equiv \int_0^1 \int_0^1 \Phi(\eta_1, \eta_2) \left(\frac{\partial \underline{x}}{\partial \eta_1} \times \frac{\partial \underline{x}}{\partial \eta_2} \right) \mathrm{d}\eta_1 \, \mathrm{d}\eta_2 \cdot \underline{A}^f / (\underline{A}^f \cdot \underline{A}^f). \tag{A.6}$$

In particular, the face centroid is found by taking Φ to be η_1 and η_2 in turn:

$$\begin{aligned} \overline{\eta}_1 &= \frac{1}{2} + \frac{1}{12} (\overrightarrow{ab} \times \overrightarrow{dc}) \cdot \underline{A}^f / (\underline{A}^f \cdot \underline{A}^f), \\ \overline{\eta}_2 &= \frac{1}{2} + \frac{1}{12} (\overrightarrow{bc} \times \overrightarrow{ad}) \cdot \underline{A}^f / (\underline{A}^f \cdot \underline{A}^f). \end{aligned} \tag{A.7}$$

It is noteworthy that these forms yield the correct results for the area and centroid of a triangular face, when two face vertices coincide [P.J. O'Rourke and M.S. Sahota, Los Alamos National Laboratory, personal communication, 2002].

The face average of a quantity can be expressed in terms of values at the four face vertices,

$$\begin{aligned} &\Phi^f = \sum_{v=a}^{d} \lambda^{f,v}\Phi^l, \text{ where} \\ &\lambda^{f,a} = (1-\bar{\eta}_1)(1-\bar{\eta}_2), \quad \lambda^{f,b} = \bar{\eta}_1(1-\bar{\eta}_2), \\ &\lambda^{f,c} = \bar{\eta}_1\bar{\eta}_2, \quad \lambda^{f,d} = (1-\bar{\eta}_1)\bar{\eta}_2. \end{aligned} \tag{A.8}$$

The element volume is defined by,

$$V^e \equiv \int_0^1 \int_0^1 \int_0^1 \left(\frac{\partial \underline{x}}{\partial \xi_1} \times \frac{\partial \underline{x}}{\partial \xi_2} \right) \cdot \frac{\partial \underline{x}}{\partial \xi_3} \, \mathrm{d}\xi_1 \, \mathrm{d}\xi_2 \, \mathrm{d}\xi_3 \tag{A.9}$$

and volume-averaged quantities are defined by,

$$\overline{\Phi}^e \equiv \int_0^1 \int_0^1 \int_0^1 \Phi(\underline{\xi}) \left(\frac{\partial \underline{x}}{\partial \xi_1} \times \frac{\partial \underline{x}}{\partial \xi_2} \right) \cdot \frac{\partial \underline{x}}{\partial \xi_3} \, \mathrm{d}\xi_1 \, \mathrm{d}\xi_2 \, \mathrm{d}\xi_3 / V^e. \tag{A.10}$$

The resulting expressions for the element volume V^e and for the volume centroid $\overline{\underline{\xi}}$ can be written in terms the eight element vertex positions: those expressions are not central to the present development. However, again it is noteworthy that they reduce to the correct forms for the volume and centroid of a tetrahedron. The element-averaged value is given in terms of values at the eight element vertices by,

$$\begin{aligned} &\overline{\Phi}^e = \sum_{v=1}^{8} \Lambda^{e,v}\Phi^l, \text{ where} \\ &\Lambda_{e,v} = b^{e,v}(\overline{\underline{\xi}}). \end{aligned} \tag{A.11}$$

Advantages of the trilinear mapping include relative simplicity, continuity of mean fields from element to element across shared faces and natural accommodation of edge and face degeneracies (to zero length and zero area, respectively). Other advantages in the context of PDF methods for chemically reacting turbulent flows have been pointed out by Subramaniam and Haworth [4]. Moreover, in the present case trilinear basis functions are compatible with the FV geometric computations that are employed in the underlying CFD code.

A disadvantage is that it is relatively expensive to compute the values of the logical coordinates $\underline{\xi}$ that correspond to a given physical position $\underline{x}$ within an element. That computation requires inverting the nonlinear mapping of Eqs. (A.1) and (A.2). This type of computation arises in particle tracking, for example, where a particle's position is advanced in physical space and one needs to know the corresponding logical coordinates for particle/mesh interpolation. This difficulty can be dealt with by making the following approximation.

Approximate values of the logical coordinates $\underline{\xi}$ that correspond to a given physical position $\underline{x}$ can be computed quickly in the case of planar element faces. Bilinear (in two dimensions) or trilinear (in three dimensions) coordinates for an interior point can be estimated using the heights above the faces. In the two-dimensional example of Fig. 4, for example,

$$\eta_1 \approx \frac{h_a}{h_a + h_c}, \quad \eta_2 \approx \frac{h_b}{h_b + h_d}. \tag{A.12}$$

In three dimensions, an element face can be approximated as a plane that passes through the face centroid (Eq. (A.7)) and whose normal is given by Eq. (A.5). This definition is consistent for the two elements that share a given face. However, in general it allows physical positions $\underline{x}$ that lie in multiple FV elements (more than one element for which $0 \leqslant \xi_i \leqslant 1$, $i = 1, 2, 3$) or in no FV element (no element for which $0 \leqslant \xi_i \leqslant 1$, $i = 1, 2, 3$). In practice, these ambiguities can be resolved using simple rules. With this approximation, one does not automatically recover a consistent treatment for a tetrahedron. However, a direct treatment for tetrahedra proceeds similarly, using heights above triangular faces as the basis functions/logical coordinates [38].

Appendix B. Correction velocities

In this appendix the algebraic solution for element-level vertex correction velocities is outlined, and characteristics of the resulting correction velocity field are discussed. The local element face and vertex numbering convention of Fig. 3 is adopted (element faces indexed $f = 1$–6, vertices indexed $v = 1$–8); the superscript e element index is dropped, for clarity. The element-face mass flux correction is abbreviated as $\mathscr{Q}^f : \mathscr{Q}^f = \hat{Q}^{e,f}/A^f$ (Eqs. (17) and (A.4)); and the vertex mean density–velocity product is abbreviated as $q_i^v : q_i^v = \rho^v \hat{U}_i^{e,v}$. The same procedure applies to the Stage 3 correction, with $\hat{\hat{Q}}^{e,f}$ replacing $\hat{Q}^{e,f}$ and $\hat{\hat{U}}_i^{e,v}$ replacing $\hat{U}_i^{e,v}$.

B.1. Derivation

Six equations for the 24 q_i^v's enforce the desired face mass fluxes (Eq. (16)),

$$\begin{aligned}\mathscr{Q}^1 &= (\lambda^{1,8}q_1^8 + \lambda^{1,4}q_1^4 + \lambda^{1,1}q_1^1 + \lambda^{1,5}q_1^5)n_1^1 + (\lambda^{1,8}q_2^8 + \lambda^{1,4}q_2^4 + \lambda^{1,1}q_2^1 + \lambda^{1,5}q_2^5)n_2^1 \\ &\quad + (\lambda^{1,8}q_3^8 + \lambda^{1,4}q_3^4 + \lambda^{1,1}q_3^1 + \lambda^{1,5}q_3^5)n_3^1, \\ \mathscr{Q}^2 &= (\lambda^{2,6}q_1^6 + \lambda^{2,2}q_1^2 + \lambda^{2,3}q_1^3 + \lambda^{2,7}q_1^7)n_1^2 + (\lambda^{2,6}q_2^6 + \lambda^{2,2}q_2^2 + \lambda^{2,3}q_2^3 + \lambda^{2,7}q_2^7)n_2^2 \\ &\quad + (\lambda^{2,6}q_3^6 + \lambda^{2,2}q_3^2 + \lambda^{2,3}q_3^3 + \lambda^{2,7}q_3^7)n_3^2,\end{aligned} \tag{B.1}$$

and similarly for each of the remaining four faces. The 24 weights $\lambda^{f,v}$ and the outward-pointing unit normal vectors n_i^f are known from the element geometry and the trilinear mapping (Eqs. (A.8) and (A.5), respectively); and the six face mass fluxes $\mathscr{Q}^f$ are prescribed according to Eq. (17) (in the case of the Stage 2 correction) or Eq. (21) (in the case of the Stage 3 correction). The weights satisfy,

$$\sum_{v \in f} \lambda^{f,v} = 1. \tag{B.2}$$

Eighteen equations impose the constraint of uniform density-normal velocity product across each face (Eq. (18)):

$$\begin{aligned}
&q_1^8 n_1^1 + q_2^8 n_2^1 + q_3^8 n_3^1 = q_1^4 n_1^1 + q_2^4 n_2^1 + q_3^4 n_3^1 \quad \text{(face 1)} \\
&q_1^8 n_1^1 + q_2^8 n_2^1 + q_3^8 n_3^1 = q_1^1 n_1^1 + q_2^1 n_2^1 + q_3^1 n_3^1 \quad \text{(face 1)} \\
&q_1^8 n_1^1 + q_2^8 n_2^1 + q_3^8 n_3^1 = q_1^5 n_1^1 + q_2^5 n_2^1 + q_3^5 n_3^1 \quad \text{(face 1)}, \\
&q_1^6 n_1^2 + q_2^6 n_2^2 + q_3^6 n_3^2 = q_1^2 n_1^2 + q_2^2 n_2^2 + q_3^2 n_3^2 \quad \text{(face 2)} \\
&q_1^6 n_1^2 + q_2^6 n_2^2 + q_3^6 n_3^2 = q_1^3 n_1^2 + q_2^3 n_2^2 + q_3^3 n_3^2 \quad \text{(face 2)} \\
&q_1^6 n_1^2 + q_2^6 n_2^2 + q_3^6 n_3^2 = q_1^7 n_1^2 + q_2^7 n_2^2 + q_3^7 n_3^2 \quad \text{(face 2)}
\end{aligned} \tag{B.3}$$

and similarly for each of the remaining four faces (three additional equations per face).

This set of 24 simultaneous linear equations in 24 unknowns has been solved using Matlab. The solution can be manipulated into a fairly compact form. With

$$\underline{\mathscr{D}}^v = \frac{\mathscr{D}^f(\underline{n}^g \times \underline{n}^h) + \mathscr{D}^g(\underline{n}^h \times \underline{n}^f) + \mathscr{D}^h(\underline{n}^f \times \underline{n}^g)}{\underline{n}^f \cdot (\underline{n}^g \times \underline{n}^h)}, \tag{B.4}$$

where face indices f, g and h refer to the three element faces that share vertex v, and $\underline{j}_1, \underline{j}_2$ and $\underline{j}_3$ denoting Cartesian unit vectors in the global x_1, x_2 and x_3 directions, respectively, the solution can be written as,

$$q_i^v = \underline{j}_i \cdot \underline{\mathscr{D}}^v. \tag{B.5}$$

The vertex correction velocities are recovered by dividing by the vertex (mean) density: $\hat{U}_i^{e,v} = q_i^v/\rho^v$.

B.2. Degenerate hexahedra

Equivalent algorithms for alternative element shapes can be developed following the same procedure as that outlined above for hexahedral elements. For example, a tetrahedral element has four faces ($f = 1, 2, 3, 4$), four vertices ($v = 1, 2, 3, 4$), and all weights are equal to 1/3 ($\lambda^{f,v} = 1/3$). Four equations for the 12 correction velocity components correspond to matching the specified face correction fluxes; and eight others (two for each of the four faces) correspond to the constraint of uniform density-normal velocity product across each face. The result for a tetrahedron is identical to Eq. (B.4). In fact, Eq. (B.4) applies directly to *any* element where exactly three non-degenerate faces are shared by each vertex (e.g., prism, tetrahedron) with f, g and h referring to the three non-degenerate faces. Slight modifications are required for other element shapes (e.g., a pyramid).

B.3. Continuity of the correction field

The correction mass flowrates are continuous across element faces, by construction; the flowrates for the two elements that share a face differ only in sign. However, the correction velocity field is not necessarily continuous across element faces. In the case of orthogonal elements, the correction velocity field is continuous across element faces, and the value of the correction velocity at a vertex on a element face is the same for the two elements that share the face. However, even for orthogonal elements the value of the correction velocity at a vertex generally is different for those elements that share the vertex but do not share a common face.

Appendix C. Pressure correction algorithm

The essential steps in the mean pressure/momentum/continuity coupling to advance the finite–volume solution over one computational timestep are: (1) Momentum predictor – compute a new mean velocity field using the current mean pressure field; this velocity field does not satisfy mean continuity at the element level. (2) Pressure/velocity correctors – compute corrections to the mean pressure and mean velocity fields to enforce element-level mean mass conservation. The momentum predictor and pressure corrector each require the solution of a sparse implicit linear system that corresponds to a linearized discretized form of the corresponding governing pde; the velocity corrector is explicit. The discretized momentum predictor and pressure/velocity corrector equations are derived from Eqs. (11) and (12) using Eq. (10) as a constraint.

For present purposes, the feature of interest in the pressure algorithm is the computation of a set of element-face mass flowrates that enforce element-level mass conservation. The full set of equations is

lengthy; the essentials can be found in [27], for example. Only a skeletal outline is given here. Similar capabilities are available in other pressure-based FV solvers.

The element-centered pressure correction for FV element e, $\hat{p}^e$, is obtained from the solution of a sparse implicit symmetric linear system of the form,

$$A_{e,p}\hat{p}^e - \sum_{f=1}^{n}(A_{f,p}\hat{p}^f) = R^e_{\dot{m}} + S^e_p. \tag{C.1}$$

Here superscript f refers to the neighboring FV element through face f of element e, $R^e_{\dot{m}}$ is the continuity residual for element e and S^e_p contains all other source-term contributions. The continuity residual is the difference between the net rate of accumulation of mass in element e and the net rate of efflux of mass out of element e through its n bounding faces (see Eq. (13)),

$$R^e_{\dot{m}} = \frac{m^e(t+\Delta t) - m^e(t)}{\Delta t} + \sum_{f=1}^{n} Q^{e,f,\mathrm{FV}}. \tag{C.2}$$

Here $m^e(t) = \langle\rho\rangle^e(t)V^e(t)$ is the mean fluid mass in element e at time t, and a first-order forward time difference has been used to approximate the time derivative. The residual $R^e_{\dot{m}}$ is non-zero following the momentum predictor step; it is equal to zero following the first pressure/velocity corrector step. The diagonal coefficients $A_{e,p}$, off-diagonal coefficients $A_{f,p}$ and source terms S^e_p are functions of the mesh geometry, the current mean velocity field and fluid properties.

The explicit velocity corrector equation has the form,

$$\hat{U}^e_i = \sum_{f=1}^{n}(A_{f,U_i}\hat{p}^f) + S^e_{U_i}, \tag{C.3}$$

where a similar notation has been used.

The element-face mass flowrates $\hat{Q}^{e,f,\mathrm{FV}}$ then are explicit functions of the current velocity and pressure fields,

$$\hat{Q}^{e,f,\mathrm{FV}} = \sum_{f=1}^{n}(B_{f,p}p^f) + \sum_{f=1}^{n}(B_{f,U_i}U^f_i) + S^e_Q. \tag{C.4}$$

Immediately following the first pressure/velocity corrector step, the element-face mass flowrates satisfy,

$$-\sum_{f=1}^{n}\hat{Q}^{e,f,\mathrm{FV}} = R^e_{\dot{m}}, \tag{C.5}$$

where $R^e_{\dot{m}}$ is the continuity residual from the end of the momentum predictor step.

The FV pressure algorithm thus provides a convenient and consistent way to compute element-face mass flowrates that drive non-zero element-level continuity residuals to zero. In the present particle/FV mass consistency scheme, a single FV pressure/velocity corrector step is taken with the element-level mass residuals given by the difference between FV element mass and the mean particle mass in the element (Eq. (22)). The resulting pressure and velocity correction fields are discarded. And the computed element-face mass flowrates are used to construct the Stage 3 particle mean velocity correction field.

References

[1] T.S. Lundgren, Model equation for nonhomogeneous turbulence, Phys. Fluids 12 (1969) 485–497.
[2] S.B. Pope, PDF methods for turbulent reactive flows, Prog. Energy Combust. Sci. 85 (1985) 119–192.

[3] Q. Tang, J. Xu, S.B. Pope, Probability density function calculations of local extinction and NO production in piloted-jet turbulent methane/air flames, Proc. Combust. Inst. 28 (2000) 133–192.
[4] S. Subramaniam, D.C. Haworth, A PDF method for turbulent mixing and combustion on three-dimensional unstructured deforming meshes, Int. J. Engine Res. 1 (2000) 171–190.
[5] S. James, M.S. Anand, S.B. Pope, The Lagrangian PDF transport method for simulations of gas turbine combustor flows, AIAA Paper No. AIAA 2002-4017, 2002.
[6] M.S. Anand, S.B. Pope, H.C. Mongia, A PDF method for turbulent recirculating flows, in: Turbulent Reactive Flows, Lecture Notes in Engineering, vol. 40, Springer-Verlag, Berlin, 1989, pp. 672–693.
[7] D.C. Haworth, S.H. El Tahry, Application of a PDF method to in-cylinder flows in reciprocating engines, in: Seventh Symposium on Turbulent Shear Flows, Stanford University, 21–23 August 1989, pp. 13.1.1–13.1.6, 2.
[8] D.C. Haworth, S.H. El Tahry, Probability density function approach for multidimensional turbulent flow calculations with application to in-cylinder flows in reciprocating engines, AIAA J. 29 (1991) 208–218.
[9] J. Xu, S.B. Pope, Numerical studies of pdf/Monte Carlo methods for turbulent reactive flows, J. Comput. Phys. 152 (1999) 192–230.
[10] M. Muradoglu, P. Jenny, S.B. Pope, D.A. Caughey, A consistent hybrid finite-volume/particle method for the PDF equations of turbulent reactive flows, J. Comput. Phys. 154 (1999) 342–371.
[11] P. Jenny, S.B. Pope, M. Muradoglu, D.A. Caughey, A hybrid algorithm for the joint PDF equation of turbulent reactive flows, J. Comput. Phys. 166 (2001) 218–252.
[12] M. Muradoglu, S.B. Pope, D.A. Caughey, The hybrid method for the PDF equations of turbulent reactive flows: consistency conditions and correction algorithms, J. Comput. Phys. 172 (2001) 841–878.
[13] G. Li, M.F. Modest, Application of composition PDF methods in the investigation of turbulence–radiation interactions, J. Quant. Spectrosc. Radiat. Transfer 73 (2002) 461–472.
[14] H.J. Curran, P. Gaffuri, W.J. Pitz, C.K. Westbrook, A comprehensive modeling study of *n*-heptane oxidation, Combust. Flame 114 (1998) 149–177.
[15] S.B. Pope, Y.L. Chen, The velocity-dissipation probability density function model for turbulent flows, Phys. Fluids 2 (1990) 1437–1449.
[16] S. Mazumder, M.F. Modest, A PDF approach to modeling turbulence-radiation interactions in nonluminous flames, Int. J. Heat Mass Transfer 42 (1998) 971–991.
[17] S.B. Pope, Computations of turbulent combustion: progress and challenges, Proc. Combust. Inst. 23 (1990) 591–612.
[18] F. Gao, E.E. O'Brien, A large-eddy simulation scheme for turbulent reacting flows, Phys. Fluids A 5 (1993) 1282–1284.
[19] P.J. Colucci, F.A. Jaberi, P. Givi, S.B. Pope, Filtered density function for large eddy simulation of turbulent reacting flows, Phys. Fluids 10 (1998) 499–515.
[20] F.A. Jaberi, P.J. Colucci, S. James, P. Givi, S.B. Pope, Filtered mass density function for large-eddy simulation of turbulent reacting flows, J. Fluid Mech. 401 (1999) 85–122.
[21] L.Y.M. Gicquel, P. Givi, F.A. Jaberi, S.B. Pope, Velocity filtered density function for large eddy simulation of turbulent flows, Phys. Fluids 14 (2002) 1196–1213.
[22] D.C. Haworth, S.B. Pope, Monte Carlo solutions of a joint PDF equation for turbulent flows in general orthogonal coordinates, J. Comput. Phys. 72 (1987) 311–346.
[23] S.B. Pope, Turbulent Flows, Cambridge University Press, Cambridge, MA, 2000.
[24] D.C. Haworth, S.H. El Tahry, M.S. Huebler, S. Chang, Multidimensional port-and-cylinder flow calculations for two- and four-valve-per-cylinder engines: influence of intake configuration on flow structure, SAE Paper No. 900257, 1990.
[25] D.C. Haworth, S.H. El Tahry, M.S. Huebler, A global approach to error estimation and physical diagnostics in multidimensional computational fluid dynamics, Int. J. Numer. Meth. Fluids 17 (1993) 75–97.
[26] B. Khalighi, S.H. El Tahry, D.C. Haworth, M.S. Huebler, Computation and measurement of flow and combustion in a four-valve engine with intake variations, SAE Paper No. 950287, 1995.
[27] J.H. Ferziger, M. Peric, Computational Methods for Fluid Dynamics, third ed., Springer-Verlag, Berlin, 2001.
[28] R.I. Issa, Solution of the implicitly discretised fluid flow equations by operator splitting, J. Comput. Phys. 62 (1986) 40–65.
[29] R.I. Issa, A.D. Gosman, A.P. Watkins, The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme, J. Comput. Phys. 62 (1986) 66–82.
[30] D.C. Haworth, M.S. Huebler, S.H. El Tahry, W.R. Matthes, Multidimensional calculations for a two-stroke-cycle engine: a detailed scavenging model validation, SAE Paper No. 932712, 1993.
[31] D.C. Haworth, J.M. Maguire, W.R. Matthes, R. Rhein, S.H. El Tahry, Dynamic fluid flow analysis of oil pumps, SAE Paper No. 960422, 1996.
[32] S. Chang, D.C. Haworth, Adaptive grid refinement using cell-level and global imbalances, Int J. Numer. Meth. Fluids 24 (1997) 375–392.
[33] T.D. Dreeben, S.B. Pope, Nonparametric estimation of mean fields with application to particle methods for turbulent flows, Sibley School of Mechanical and Aerospace Engineering Report FDA 92-13, Cornell University, Ithaca, NY, 1992.

[34] D.C. Haworth, S.B. Pope, A second-order Monte Carlo method for the solution of the Ito stochastic differential equation, Stoch. Anal. Appl. 4 (1986) 151–186.
[35] W.C. Welton, S.B. Pope, Model calculations of compressible turbulent flows using smooth particle hydrodynamics, J. Comput. Phys. 134 (1997) 150–168.
[36] R. Cao, S.B. Pope, Numerical integration of stochastic differential equations: weak second-order mid-point scheme for application in the composition PDF method, J. Comput. Phys. 185 (2003) 194–212.
[37] S.V. Patankar, Numerical Heat Transfer and Fluid Flow, McGraw-Hill, New York, 1980.
[38] K.H. Huebner, D.L. Dewhirst, D.E. Smith, D.G. Byrom, The Finite Element Method for Engineers, fourth ed., John Wiley and Sons, New York, 2001.
[39] U. Ghia, K.N. Ghia, C.T. Shin, High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method, J. Comput. Phys. 48 (1982) 387–411.
[40] D.S. Kershaw, Differencing of the diffusion equation in Lagrangian hydrodynamic codes, J. Comput. Phys. 76 (1981) 375–395.
[41] J.E. Morel, J.E. Dendy, M.L. Hall, S.W. White, A cell-centered Lagrangian-mesh diffusion differencing scheme, J. Comput. Phys. 103 (1992) 286–299.
[42] M. Shashkov, Conservative Finite-Difference Methods on General Grids, CRC Press, Boca Raton, FL, 1992.
[43] C. Arcoumanis, H.G. Green, J.H. Whitelaw, The application of laser Reyleigh scattering to a reciprocating model engine, SAE Paper No. 840376, 1984.
[44] A.P. Morse, J.H. Whitelaw, M. Yianneskis, Turbulent flow measurements by laser-Doppler anemometry in motored piston–cylinder assemblies, Trans. ASME 101 (1979) 208–216.
[45] D.C. Haworth, S.B. Pope, A PDF modeling study of self-similar turbulent free shear flows, Phys. Fluids 30 (1987) 1026–1044.
[46] S.H. El Tahry, The k–ϵ equations in compressible flows, J. Energy 7 (1983) 345–353.