

PDF Simulations of a Bluff-Body Stabilized Flow

P. Jenny,* M. Muradoglu,† K. Liu,† S. B. Pope,† and D. A. Caughey†

*Chevron Petroleum Technology Company, 6001 Bollinger Canyon Road, San Ramon, California 94583-0932;
and †Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, New York 14853
E-mail: pjnn@chevron.com

Received November 2, 1999; revised October 31, 2000

Three different PDF algorithms have been applied to investigate a constant-density bluff-body stabilized flow using the same turbulence models and the same boundary conditions. The objectives of this paper are to compare the three algorithms in terms of numerical accuracy and efficiency and to demonstrate the ability of PDF methods to calculate this type of flow accurately. While one of the three algorithms is a stand-alone particle-mesh method, the other two are consistent hybrid algorithms, i.e., both are particle methods coupled with finite-volume schemes. The motivation for hybrid algorithms is to reduce the statistical and bias errors. Since the coupling between the finite-volume scheme and the particle method is a major numerical issue, different approaches have been investigated. It is shown that the results obtained from the three numerical algorithms are in good agreement with each other and with the experimental data. © 2001 Academic Press

Key Words: PDF methods; particle methods; Monte Carlo; finite-volume; turbulence modeling; bluff-body stabilized flow.

1. INTRODUCTION

The calculation of complex turbulent flows is of great importance for many engineering applications, and the accuracy of such calculations depends mainly on turbulence models and on the numerical algorithm. The task of turbulence modeling is to make the flow computations feasible without great loss of accuracy. To solve the resulting set of equations accurate numerical algorithms are required. In the context of probability density function (PDF) methods, less attention has been paid to the last point.

Traditional turbulence models, including two-equation models [23, 47] and second-moment closures [22], are based on Reynolds averaging techniques and yield modeled equations for statistical moments. In comparison to these models, PDF methods achieve closure through a modeled transport equation for the one-point, one-time PDF of certain fluid properties in a turbulent flow [11, 32, 33]. The advantage of PDF methods is

that both convection and reaction are represented exactly without modeling assumptions. The tremendous amount of statistical information contained in the PDFs obviously provides a fuller description of turbulent flows than two-equation models or second-moment closures. An overview of turbulence theory and modeling approaches is given in [38]. During the past decade, progress in PDF methods has been made in several areas: adopting a more advanced joint velocity-frequency-composition PDF method which provides a model for the turbulent time scale [39, 44]; and incorporating modeling techniques developed for second-moment closures [14, 34, 35, 45]. These models have been successfully applied in modeling several inert flows [1, 9, 27], reactive flows, and turbulent flames [26, 30, 41].

Different numerical solution algorithms are required for turbulence models of different levels. Moment closures result in a set of partial differential equations. These equations are usually solved numerically by finite-difference or finite-volume methods [20]. In contrast to moment-closure model equations, the modeled PDF transport equation has a completely different structure. It is a high-dimensional scalar transport equation, and it is infeasible to solve it with a finite-volume or a finite-difference method. From early times in the development of PDF methods, Monte Carlo techniques have been employed in which the PDF is represented by an ensemble of particles [31]. Stochastic differential equations (SDEs) are constructed to model the particle properties, e.g., velocity, composition, and frequency, such that the particles exhibit the same PDF as in turbulent flows.

Monte Carlo methods are widely used in computational physics [21] to solve high-dimensional problems since the computational costs increase only linearly with the number of dimensions. Their application in PDF methods has progressed through different stages. In the first method, the particles are located at grid nodes in physical space [31]. Pope [32] then suggested that it is preferable to use a method in which the particles are continuously distributed. Later, a hybrid algorithm was implemented in the code *PDF2DS* in which composition PDFs are calculated by Monte Carlo methods while a finite-volume method is applied to solve for the mean velocity, dissipation, and mean pressure fields [4, 6, 30].

More recently, a stand-alone particle-mesh algorithm has been developed for the joint velocity-frequency-composition PDF model [37]. This method is implemented in the code *PDF2DV* [36]. This is a code to calculate statistically stationary two-dimensional (plane or axi-symmetric) turbulent reactive flows using the joint velocity-frequency-composition PDF method. It has been applied in several published calculations [1, 10, 13, 41, 46]; however, the computational costs are relatively high because of the large number of particles required to decrease the bias error. Motivated by this deficiency, hybrid methods have been developed [19, 29], which are, in contrast to other hybrid methods, fully consistent on the level of the modeled equations. Since the coupling between the finite-volume scheme and the particle method is crucial, two different approaches have been investigated, i.e., a loosely coupled algorithm by Muradoglu *et al.* [29] and a tightly coupled algorithm by Jenny *et al.* [19]. It has been demonstrated [19] that for a nonpremixed piloted jet flame the tightly coupled algorithm which is implemented in *PDF-2D-FV* [19] is about 20 times more efficient than the stand-alone particle-mesh method implemented in *PDF2DV*.

It is not the focus of this paper to compare different turbulence models. The motivation for the current work is to compare the stand-alone particle method and the loosely coupled and the tightly coupled hybrid algorithms in terms of numerical accuracy and efficiency. Therefore, exactly the same model equations are solved, the same grids are used, and the

same boundary conditions are applied. The test case is a cold bluff-body stabilized flow. Besides their practical interest, bluff-body stabilized flows are very challenging test cases for turbulence models and numerical schemes and have been studied experimentally and theoretically [6–8, 24, 25, 40]. It is demonstrated that the converged solutions of the three schemes are in good agreement and that this class of flows can be predicted very well with PDF methods. Finally, asymptotic studies allow a comparison of the efficiency of the three PDF algorithms for given numerical error tolerances.

In the next section the basic idea of PDF modeling is outlined; the applied turbulence models are described in Sections 2.1 and 2.2. In Section 3, the three different solution algorithms are explained. Convergence results are shown in Section 4.

2. PDF MODELS

In this section, the PDF models applied for these studies are outlined. Since only cold flow is considered in this paper, everything is explained for constant density. For all simulations the same joint velocity-frequency PDF model is used which is similar to that used in several other recent studies [1, 13, 41].

Although the flow considered here is constant density, the equations are presented in the variable-density form in which they are used in turbulent combustion calculations. The mass density function (mdf) \mathcal{F} and the one-point, one-time Eulerian mass-weighted joint PDF (JPDF) \tilde{f} of velocity $\mathbf{U}(\mathbf{x}, t)$, and turbulence frequency $\omega(\mathbf{x}, t)$ are defined by

$$\langle \rho \rangle \tilde{f}(\mathbf{V}, \theta; \mathbf{x}, t) = \mathcal{F}(\mathbf{V}, \theta; \mathbf{x}, t) \quad (1)$$

$$\equiv \rho \langle \delta(\mathbf{U} - \mathbf{V}) \delta(\omega - \theta) \rangle, \quad (2)$$

where $\langle \cdot \rangle$ denotes a mean quantity, $\tilde{\cdot}$ denotes a mass-weighted quantity, ρ is the density, and \mathbf{V} and θ are the sample spaces for \mathbf{U} and ω , respectively (for the constant density flow, $\rho = \langle \rho \rangle$ and $\tilde{\mathbf{U}} = \langle \mathbf{U} \rangle$).

Because of the high-dimensionality of \tilde{f} (e.g., for the case of axi-symmetric constant density flow with one species the number of dimensions is 6, but for reacting flow the number of dimensions can be much higher, depending on the number of species involved), it is infeasible to solve the modeled transport equation for \tilde{f} using a finite-difference or finite-volume scheme. Fortunately, the Monte Carlo approach (Lagrangian view; particle method) makes PDF simulations feasible since the computational costs increase only linearly with the number of sample space dimensions.

From a Lagrangian viewpoint, the flow is represented by a set of particles which evolve by stochastic differential equations. This is done in a way such that the particles exhibit the same JPDF as the solution of the modeled JPDF transport equation. Each particle has a set of properties $\{m^*, \mathbf{X}^*, \mathbf{U}^*, \omega^*\}$, where m^* represents the mass of the particle, \mathbf{X}^* is its coordinate, \mathbf{U}^* the velocity, and ω^* the particle's turbulence frequency (the superscript '*' denotes that the quantity is a particle property).

For the evolution of \mathcal{F} , model equations have been developed, using the modeling theories for turbulent reactive flows [32]. Models are required only for the pressure-strain-rate correlation and dissipation. Models for particle velocity and turbulence frequency are described in the following sections.

2.1. Velocity Model

In PDF methods, the fluid particle velocity $\mathbf{U}^+(t)$ is represented by the stochastic particle velocity $\mathbf{U}^*(t)$ and various Langevin models have been developed to model the evolution of the particles in the velocity-sample-space [14, 32, 34, 45]. Here we use the simplest one, the simplified Langevin model (SLM),

$$dU_i^*(t) = -\frac{1}{\langle \rho \rangle} \frac{\partial \langle p \rangle}{\partial x_i} dt - \left(\frac{1}{2} + \frac{3}{4} C_0 \right) \Omega (U_i^*(t) - \tilde{U}_i) dt + (C_0 k \Omega)^{1/2} dW_i \quad (4)$$

where p is the pressure,

$$\Omega \equiv C_\Omega \frac{\langle \rho^* \omega^* \mid \omega^* \geq \tilde{\omega} \rangle}{\langle \rho \rangle} \quad (5)$$

is the conditional Favre averaged turbulence frequency,

$$k = \frac{\widetilde{u_i u_i}}{2} \quad (6)$$

is the turbulent kinetic energy, and C_0 and C_Ω are model constants (Table I). Diffusion in velocity space is represented as a Wiener process $\mathbf{W}(t)$, where $dW_i(t) = W_i(t+dt) - W_i(t)$ is normally distributed with $\langle dW_i(t) \rangle = 0$ and $\langle dW_i(t) dW_j(t) \rangle = dt \delta_{ij}$. The SLM is equivalent to Rotta's model at the second-moment-closure level [35].

In the hybrid algorithms (presented in Section 3), a modeled transport equation for $\tilde{g}(\mathbf{v}, \theta; \mathbf{x}, t)$ is solved. The function \tilde{g} is the JPDF in the fluctuating velocity-frequency space, and \mathbf{v} is the sample variable of the fluctuating (Favre) velocity \mathbf{u} . In place of $\mathbf{U}^*(t)$, the fluctuating part $\mathbf{u}^*(t)$ of the particle velocity becomes a particle property. The modified Langevin equation,

$$du_i^*(t) = \frac{1}{\langle \rho \rangle} \frac{\partial (\langle \rho \rangle \widetilde{u_i u_j})}{\partial x_j} dt - u_j^* \frac{\partial \tilde{U}_i}{\partial x_j} dt - \left(\frac{1}{2} + \frac{3}{4} C_0 \right) \Omega u_i^*(t) dt + (C_0 k \Omega)^{1/2} dW_i, \quad (7)$$

has been derived from (4). In contrast to \tilde{f} , the modified JPDF \tilde{g} contains no information about the mean velocity $\tilde{\mathbf{U}}$, but \tilde{f} and \tilde{g} are otherwise equivalent. However, the use of Eq. (7) instead of Eq. (4) is numerically preferable in hybrid methods [19].

TABLE I
Model Constants

Constant:	Value:	Used in:
C_0	2.1	SLM
C_Ω	0.6893	Definition of Ω
$C_{\omega 1}$	0.65	Turbulence frequency model
$C_{\omega 2}$	0.9	Turbulence frequency model
C_3	1.0	Turbulence frequency model
C_4	0.25	Turbulence frequency model

2.2. Model for the Turbulent Frequency

To close Eq. (7) or (4) a model for the conditional turbulence frequency is needed. In many PDF codes, a model for the mean dissipation (e.g., the $k - \varepsilon$ model) is used to estimate Ω . In the joint velocity-frequency-composition model, the turbulent frequency ω^* is a particle property; and to account for external intermittency effects, Eq. (5) is used to estimate Ω (the advantage of this approach in intermittent flow is that Eq. (5) accounts only for those particles which represent turbulent flow). The model constant C_Ω (Table I) is chosen such that Ω equals $\tilde{\omega}$ for fully developed homogeneous turbulence. Using this approach the Favre averaged turbulence dissipation can be defined as

$$\tilde{\varepsilon} = k\Omega. \quad (8)$$

The stochastic model for $\omega^*(t)$ is

$$d\omega^*(t) = -C_3(\omega^* - \tilde{\omega})\Omega dt - S_\omega\Omega\omega^*(t) dt + (2C_3C_4\tilde{\omega}\Omega\omega^*(t))^{1/2} dW, \quad (9)$$

where C_3 and C_4 are model constants (Table I) [17, 44], and $W(t)$ is a Wiener process, independent of that in the velocity model. In Eq. (9), S_ω is the source of turbulence frequency. Here it is modeled as

$$S_\omega = C_{\omega 2} - C_{\omega 1} \frac{P}{k\Omega}, \quad (10)$$

where P is the turbulence production

$$P = -\widetilde{u_i u_j} \frac{\partial \tilde{U}_i}{\partial x_j}, \quad (11)$$

and $C_{\omega 1}$ and $C_{\omega 2}$ are additional model constants (Table I). It is important to notice that with this turbulence frequency model and the SLM model all equations are closed and no further turbulence model is required.

2.3. Reynolds Averaged Euler (RAE) Equations

In the present hybrid algorithms, a finite-volume scheme is applied to solve the Reynolds averaged Euler (RAE) equations given by

$$\begin{aligned} \frac{\partial}{\partial t} \langle \rho \rangle + \frac{\partial}{\partial x_i} (\langle \rho \rangle \tilde{U}_i) &= 0 \\ \frac{\partial}{\partial t} (\langle \rho \rangle \tilde{U}_i) + \frac{\partial}{\partial x_j} (\langle \rho \rangle \tilde{U}_i \tilde{U}_j + \langle p \rangle \delta_{ij}) &= -\frac{\partial}{\partial x_j} (\langle \rho \rangle \widetilde{u_i u_j}) \\ \frac{\partial}{\partial t} (\langle \rho \rangle \tilde{E}_s) + \frac{\partial}{\partial x_i} (\tilde{U}_i (\langle \rho \rangle \tilde{E}_s + \langle p \rangle)) &= -\frac{\partial}{\partial x_i} \left(\frac{\langle \rho \rangle}{2} \widetilde{u_i u_j u_j} \right) - \frac{\partial}{\partial x_i} (\tilde{U}_j \langle \rho \rangle \widetilde{u_i u_j}), \end{aligned} \quad (12)$$

with the Favre averaged total sensible energy

$$\tilde{E}_s = \tilde{\varepsilon}_s + \frac{1}{2} (\tilde{U}_i \tilde{U}_i + \widetilde{u_i u_i}) \quad (13)$$

and the Favre averaged sensible internal energy $\tilde{\varepsilon}_s$.

Because only high Reynolds number flow remote from walls is considered here, the molecular effects in the conservation equations are neglected (wall effect close to the bluff-body is neglected). The RAE equations are closed by the mean equation of state defined as

$$\langle p \rangle = (\hat{\gamma}' - 1) \left(\langle \rho \rangle \tilde{E}_s - \frac{\langle \rho \rangle}{2} (\tilde{U}_i \tilde{U}_i + \widetilde{u_i u_i}) \right). \quad (14)$$

Here we simply take $\hat{\gamma}' = 1.4$, although this choice does not affect the results.

Notice that the terms on the left-hand side of the RAE equations are in the same form as the compressible Euler equations written in the conservation form. Since the terms on the right-hand side represent turbulent effects and are extracted from the particles, the RAE equations are treated as compressible Euler equations with source terms.

3. NUMERICAL ALGORITHMS

In Section 2, why particle methods are an attractive tool to solve the modeled PDF transport equation was explained. For such algorithms the numerical error of statistically stationary results consists of the spatial discretization error, the bias error, the statistical error, and the particle time-stepping error. The three algorithms used for the studies in this paper are second-order accurate in space [19, 48] so that the spatial discretization error is expected to scale with the number of grid cells (denoted by M^2) as M^{-2} . The bias error is a deterministic error caused by the random fluctuations in the mean fields which are used in the particle equations; and, on the basis of earlier theoretical and numerical studies [19, 37, 48], it is expected to scale with the number of particles per cell (denoted by N_{pc}) as N_{pc}^{-1} . The particle time-stepping error has been found to be negligible [48], if the CFL criterion is fulfilled and therefore it is not further investigated here.

In contrast to the consistent hybrid methods, for the stand-alone particle-mesh method the bias error is a major cause for the relatively high CPU time requirements of PDF simulations. In order to keep it small, N_{pc} has to be chosen sufficiently large and therefore many particles are required to get accurate results. Xu and Pope [48] showed that for a flamelet calculation of a piloted jet flame test case 1400 particles per cell are required to keep the bias error smaller than 5% in all quantities; Jenny *et al.* [19] showed that 55 particles per cell are required for the same test case and the same error tolerance with the consistent hybrid method *PDF-2D-FV*. The statistical error is proportional to $N_{\text{pc}}^{-1/2}$ and can be further reduced by time averaging the results in the statistically stationary state or by performing multiple independent simulations. For the studies in this paper, sufficient time averaging of the statistically stationary results has been applied to reduce the statistical error to a size which allowed asymptotic studies of the bias and of the spatial discretization error.

3.1. Stand-Alone Particle-Mesh Method

A stand-alone particle-method which solves the joint velocity-frequency-composition PDF transport equation has been implemented in the code *PDF2DV* [36], which is discussed in detail by Xu and Pope [48]. The particles evolve according to the SDEs (4) and (9) (the evolution of the particles in the composition sample space is not considered here; the particle velocity \mathbf{U}^* is used to evolve the particles in the physical space) and to close these equations

$\tilde{\mathbf{U}}$, k , Ω , $\langle \rho \rangle$, and $\tilde{\omega}$ are extracted from the particle field and represented at the nodes of a rectangular grid. To estimate these quantities, linear kernel functions are applied [16], and linear splines are used to interpolate the values to the particle positions. The details of this cloud-in-cell (CIC) method can be found in [12, 16]. Additionally, to get the pressure in Eq. (4), a Poisson equation has to be solved. The code *PDF2DV* solves for statistically stationary 2-D flows (plane or axi-symmetric) and uses rectangular grids. It has been applied successfully to a variety of reacting and nonreacting turbulent flow problems [1, 10, 13, 41, 46].

3.2. Consistent Hybrid Algorithms

The consistent hybrid method has been developed to overcome the deficiencies of the stand-alone particle-mesh method, i.e., to reduce the bias error and to avoid the complicated pressure correction algorithm that requires damping and dissipation as implemented in *PDF2DV* [19, 29]. For the consistent hybrid approach, a finite-volume or finite-difference scheme is used to solve the Reynolds averaged Euler Equations (RAE) coupled with the mean equation of state (14), and a particle-mesh method is applied to solve the modeled transport equation of the JPDF for the fluctuating velocity and turbulence frequency (Eqs. (7) and (9)). It is emphasized here that all the equations solved by the hybrid method are directly derived from the modeled transport equation of the JPDF which is the same as solved by the stand-alone particle-mesh method *PDF2DV*. Therefore, unlike some earlier hybrid methods [2, 4, 6], the present hybrid algorithm is completely consistent at the level of the modeled equations. To close the particle evolution and the RAE equations, the finite-volume scheme and the particle method are coupled as follows. The mean velocity field is supplied to the particle method by the finite-volume scheme, which in turn gets the turbulent fluxes from the particle method. This way, the statistical error is substantially reduced in the mean velocity and pressure fields. It has been found that the use of these smooth fields in the particle equations leads also to a dramatic reduction in the deterministic bias error [19, 29]. Notice that the statistical error in the finite-volume fields comes from the fluctuations in the turbulent fluxes which are source terms in the RAE equations.

To evolve the particles in the physical space, the Favre averaged velocity field has to be interpolated from the finite-volume data to the particle positions. The velocity interpolation scheme used here [19] is second-order accurate, and it guarantees that the mean dilatation rate $\nabla \cdot \tilde{\mathbf{U}}$ remains zero everywhere for a constant density flow. It also guarantees that the mean fluxes at the cell interfaces are consistent with those computed by the finite-volume flux solver.

As in *PDF2DV*, it is necessary to extract some fields from the particles to close the finite-volume scheme and the particle evolution equations, and it is done in the same way. The extracted particle quantities which are needed in the particle evolution equations are interpolated from the nodal values of the corresponding cell to the particle locations using linear splines. The spatial derivatives that appear in the particle equations are evaluated at the grid nodes using second-order central differences and interpolated to the particle positions. The kernel estimations, the evaluation of spatial derivatives, and the interpolation schemes are second-order accurate yielding second-order accuracy in space [12, 36].

In the solution process, the finite-volume scheme and the particle method are periodically used to solve the corresponding equations. Each period is called an “outer” iteration and consists of “inner” finite-volume and particle iterations. Different hybrid algorithms are distinguished from each other by the way the finite-volume schemes and the particle methods

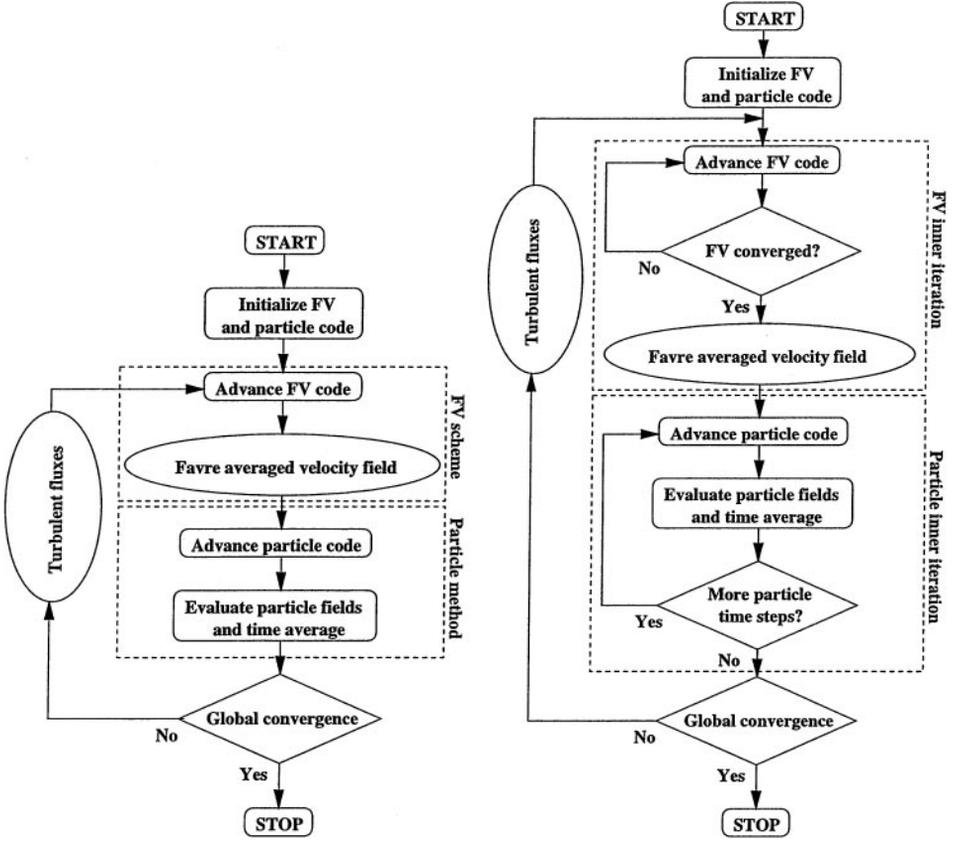


FIG. 1. Flow chart of the tightly (left) and loosely coupled (right) hybrid algorithms.

are coupled. In a strictly loosely coupled algorithm, an outer iteration is completed by running the finite-volume scheme until steady state is reached. Then a specified number of particle method time steps is performed. In a tightly coupled algorithm, on the other hand, only one finite-volume scheme time step and one particle method time step are performed to complete an outer iteration. The flow charts of the loosely and the tightly coupled algorithms are shown in Fig. 1. Notice that the loosely and the tightly coupled algorithms represent two extreme cases and any other coupling strategy between these two extremes may be used instead.

3.2.1. Loosely coupled algorithm. A loosely coupled hybrid algorithm for the solution of the PDF equations has been developed and its numerical features, such as coupling, convergence, statistical error and deterministic error, have been extensively examined by Muradoglu *et al.* [29] in the simpler setting of a 1-D stochastic ideal flow case. In the present study, this loosely coupled algorithm is extended to 2-D (plane or axi-symmetric) flows and implemented in the code *HYB2D*. The particle method in *HYB2D* is a slight modification of *PDF2DV*, and the finite-volume scheme is a modification of that developed by Caughey [3]. The method is based on a cell-centered, finite-volume approximation with added fourth-difference dissipation terms. Several local preconditioning methods including those developed by Turkel [42, 43], by Choi and Merkle [15], and by Muradoglu and

Caughey [28] have been incorporated into the finite-volume scheme in order to remove the well-known eigenvalue stiffness [43] caused by the large disparity between characteristic wave speeds at low Mach numbers. It has been found that Turkel’s incompressible preconditioner [42] outperforms the other preconditioners in terms of robustness and efficiency. Robustness is of essential importance in the context of the hybrid methods because of the random fluctuations in the turbulent fluxes which are extracted from the particle fields. It has also been found that Turkel’s incompressible preconditioner is the most robust in the form of Chorin’s artificial compressibility method [5].

3.2.2. Tightly coupled algorithm. The tightly coupled consistent hybrid algorithm has been developed by Jenny *et al.* [19] and implemented in a newly written code *PDF-2D-FV*. Extensive bias and grid convergence studies (using a piloted jet flame test case) are presented in [19]. The particle method is similar to the one used in *HYB2D*, and the finite-volume scheme solves for the compressible Euler equations. A second-order upwind solver is used to compute the inviscid fluxes, and an explicit Runge–Kutta time stepping method with local CFL number is applied. To overcome the low Mach number stiffness, the preconditioning method by Jenny and Müller [18] is applied and, as for the loosely coupled algorithm, to reduce the statistical error and the bias error moving time averaging of the quantities which are extracted from the particle field is applied [19]. Notice that these time-averaged quantities are used in the finite-volume scheme, in the particle evolution equations, and for the representation of the results.

4. CONVERGENCE RESULTS

The objectives of the studies shown in this section are to compare the three algorithms in terms of accuracy and efficiency and to compare the numerical results with experimental data. For all simulations, the same turbulence models with the same model constants and the same boundary conditions have been applied. Standard model constants have been applied (Table I), but it appeared that the value 0.65 for C_{ω_1} gives better results than $C_{\omega_1} = 0.56$ used by van Slooten and Pope [46]. The choice of optimal model constants is a subject of further studies.

Statistical stationarity of the results is demonstrated and convergence studies of the errors resulting from spatial discretization and bias are presented. To simplify the notation, we introduce an abbreviation scheme to identify the different runs performed. For example, a run with the stand-alone particle-mesh algorithm on a 64×64 grid with $N_{pc} = 200$ is denoted by *S64:200*. Corresponding runs with the loosely coupled and the tightly coupled algorithms are denoted by *L64:200* and *T64:200*, respectively (see also Table II).

TABLE II
Codes, Their Designation, and Their Description

Code	Designation	Description
<i>PDF2DV</i>	S	Stand-alone particle-mesh method [36]
<i>HYB2D</i>	L	Loosely coupled hybrid method [29]
<i>PDF-2D-FV</i>	T	Tightly coupled hybrid method [19]

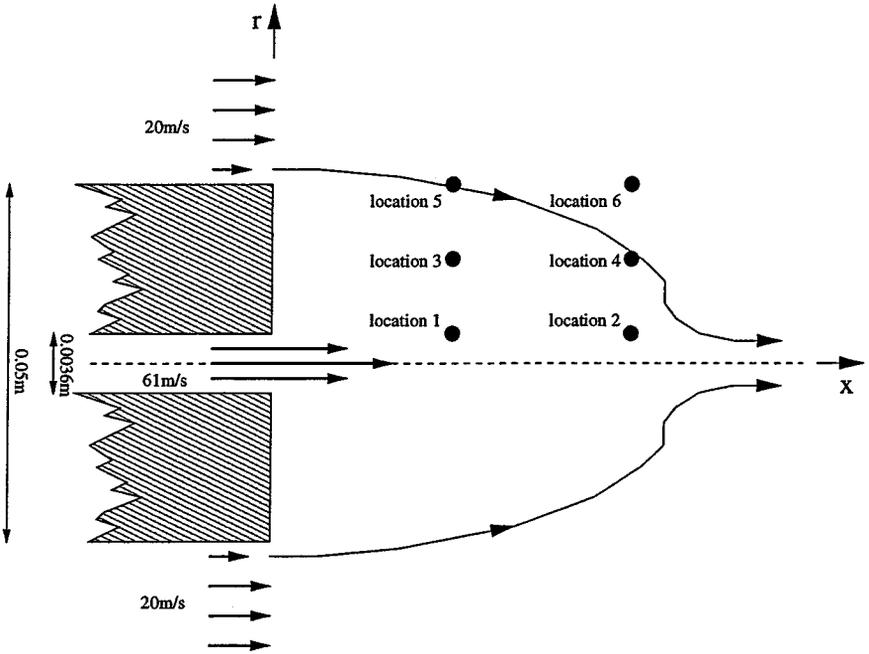


FIG. 2. Sketch of the bluff-body test case with the six selected locations 1–6.

4.1. Test Case: Bluff-Body Stabilized Flow

Bluff-body stabilized flows have been studied experimentally and theoretically [6–8, 24, 25, 40] and in addition to their practical interest they are very challenging test cases for turbulence models and numerical algorithms. For the flow investigated here, the jet diameter ($2R_j$) is 3.6 mm and the bluff-body diameter (D_b) is 50 mm (Fig. 2). Both, the jet and the co-flow consist of air with constant density. The bulk velocity of the jet is 61 m/s and the co-flow velocity is 20 m/s. A more complete description of this test case is found in [8] and experimental data and boundary conditions are found in [24]. For further explanations it is convenient to introduce the six locations 1–6 defined in Table III and shown in Fig. 2.

TABLE III
Six Selected Locations in the Solution Domain

Location	Axial Distance from from the Nozzle	Radial Distance from the Symmetry Axis
1	$D_b/2$	R_j
2	D_b	R_j
3	$D_b/2$	Mid bluff-body = 13.4 mm
4	D_b	Mid bluff-body
5	$D_b/2$	$D_b/2$
6	D_b	$D_b/2$

Note. $D_b = 50$ mm is the Bluff-Body Diameter and $R_j = 1.8$ mm is the Jet radius.

TABLE IV
The Grids Used for the Different Simulations

Grid	Cells in Jet	Cells in Bluff-Body
32×32	3	13
64×64	5	27

4.2. Solution Domain and Grids

The solution domain is $6D_b$ long in the axial direction and extends to $3D_b$ in the radial direction ($D_b = 50$ mm is the bluff-body diameter). Table IV shows the grids used for the simulations. Besides the total number of grid cells, the number of cells (in the radial direction) located in the jet and bluff-body regions is shown.

4.3. Boundary Conditions

Figure 3 shows the solution domain and the boundaries of the bluff-body test case. Making the assumption that this flow is dominated by the large recirculation zones, there is no need to resolve the boundary layer and to apply the Navier–Stokes equations with no-slip boundary conditions. It is far more important to apply accurate in-flow profiles. Therefore, in the bluff-body region of the western boundary and at the southern and northern boundaries, slip boundary conditions are applied. At the eastern boundary, the mean pressure is fixed ($\langle p \rangle = 1$ bar) while the other quantities are extrapolated from the solution domain. In the jet and co-flow regions of the western boundary, the pressure is extrapolated from the computational domain and all other quantities are given, i.e., $\langle \rho \rangle = 1.0 \text{ kg/m}^3$, $\gamma' = 1.4$, $\tilde{U}_r = 0$, and while experimental data is used for \tilde{U}_x [24] in the co-flow region (\tilde{U}_r and

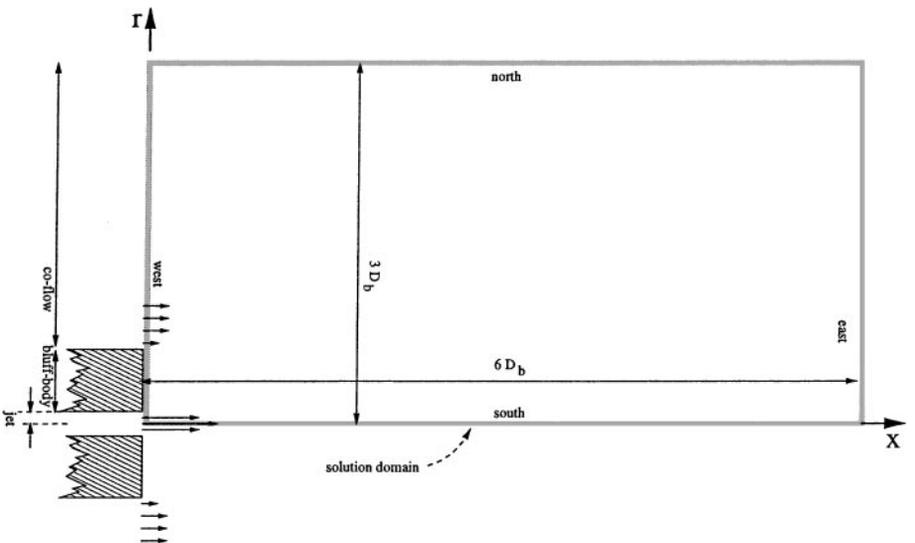


FIG. 3. Solution domain of the bluff-body test case with boundaries.

\tilde{U}_x are the radial and the axial components of the mean velocity, respectively) the function

$$\frac{\tilde{U}_x}{U_{\text{bulk}}} = c_{\text{norm}} \left(1.01 - \frac{r}{R_j} \right)^{1/6} \quad (15)$$

is used to define \tilde{U}_x in the jet region (U_{bulk} is the jet bulk velocity, R_j is the jet radius, and c_{norm} is chosen such that the area averaged velocity over the jet equals U_{bulk}). The profiles of the r.m.s. velocities u'_x and u'_r are obtained from experimental data [24], and for the shear stress $\widetilde{u_x u_r}$ the relation

$$\widetilde{u_x u_r} = \rho_{12} (\widetilde{u_x u_x} \widetilde{u_r u_r})^{1/2} \quad (16)$$

is used with the correlation coefficient $\rho_{12} = -0.4$ in the co-flow region and $\rho_{12} = 0.5(r/R_j)$ in the jet region. The mean turbulence frequency is given by

$$\tilde{\omega} = \frac{P}{k} = \frac{\widetilde{u_x u_r}}{k} \frac{\partial \tilde{U}_x}{\partial r}, \quad (17)$$

which is based on the assumption of equilibrium between production and dissipation.

4.4. Statistical Stationarity

To demonstrate statistical stationarity of the numerical solutions the time histories of $\tilde{U}_x - \tilde{U}_x^{\text{converged}}$ (left plots) and $k - k^{\text{converged}}$ (right plots) from the runs *S48:400*, *L48:25*, and *T48:25* are plotted in Fig. 4 at location 3 (upper plots) and location 4 (lower plots). It can be seen that to reach stationarity it takes about 3000 particle method iterations for the stand-alone particle-mesh method *PDF2DV*, and approximately 6000 and 10,000 iterations for the consistent hybrid algorithms *HYB2D* and *PDF-2D-FV*, respectively.

With none of the methods it is possible to achieve statistical stationarity on much finer grids (e.g., 100×100). The reason for this is discussed in the Appendix.

4.5. Quantification of Errors

It has been found by Xu and Pope [48] that the particle time-stepping error is negligible, and therefore it is not considered in the present study. Furthermore, the statistical error is sufficiently reduced by time averaging to allow asymptotic studies of the bias error and the spatial discretization error. For all three algorithms, *PDF2DV* [48], *HYB2D* [28], and *PDF-2D-FV* [19], it has been shown that the spatial discretization error and the bias error scale as M^{-2} and N_{pc}^{-1} , respectively. Then the error ϵ_Q in the quantity Q can asymptotically be written as

$$\frac{\epsilon_Q}{Q_{\text{ref}}} = \frac{a}{M^2} + \frac{b}{N_{\text{pc}}}, \quad (18)$$

where Q_{ref} is a reference value, M^2 is the number of cells, and N_{pc} is the number of particles per cell. With each code, computations with different values of N_{pc} and M^2 were performed to verify the behavior of the error given by Eq. (18), and to determine the values of a and b (Tables V and VI show a and b for \tilde{U}_x and k at the locations 1–6). Note that the reference

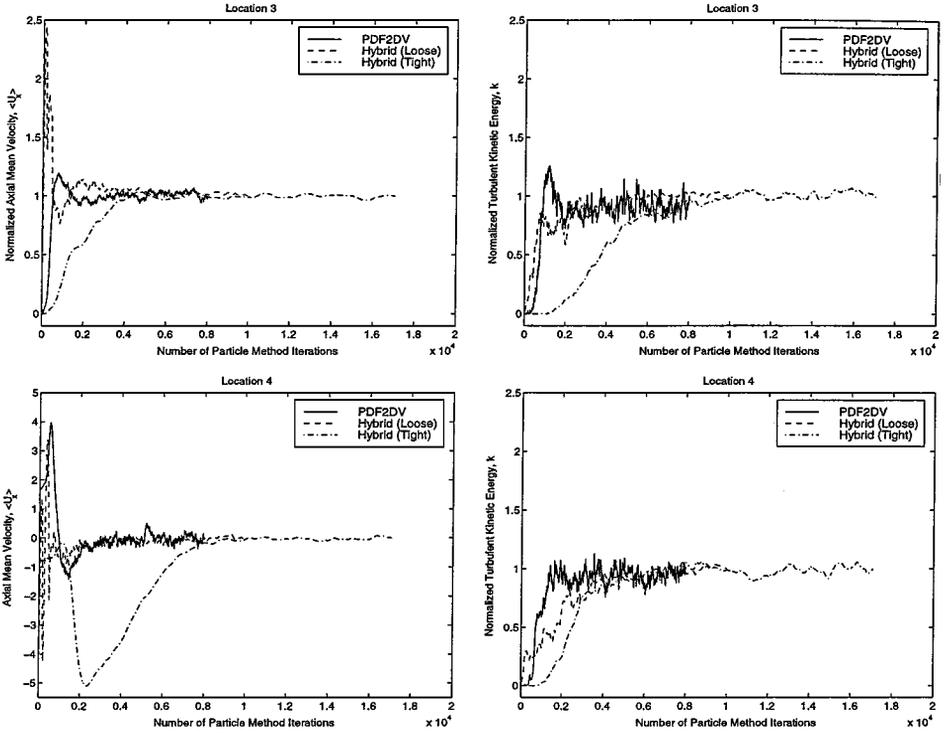


FIG. 4. Time histories of $\tilde{U}_x - \tilde{U}_x^{\text{converged}}$ (left plots) and of $k - k^{\text{converged}}$ (right plots) at location 3 (upper plots) and location 4 (lower plots).

values $\tilde{U}_{x_{\text{ref}}}$ and k_{ref} have been taken as 61 m/s and 127 m²/s² which correspond to the jet bulk velocity and the highest turbulent kinetic energy at the inlet boundary, respectively.

It may be seen from Table V that away from the jet shear layer (locations 3–6) the spatial discretization errors of the three codes are comparable. But in the shear layer (locations 1 and 2), the level of error in *PDF2DV* and *HYB2D* is much larger, whereas that in \tilde{U}_x in *PDF-2D-FV* is the same as elsewhere, and that in k is smaller by about a factor of 5 than in the other codes. The significant difference between the hybrid codes is attributable to the different differencing schemes used in the finite-volume codes; in particular, an unusually large amount of fourth-difference dissipation is required to produce converged results on

TABLE V
The Constant a in Eq. (18) for the Three Algorithms

Location	<i>PDF2DV</i>		<i>HYB2D</i>		<i>PDF-2D-FV</i>	
	\tilde{U}_x	k	\tilde{U}_x	k	\tilde{U}_x	k
1	-383.2	680.8	447.1	1100.6	-20.8	-81.0
2	-389.6	417.7	-247.9	690.7	-21.4	-134.5
3	-1.8	14.1	-21.7	15.1	20.3	-1.9
4	17.5	27.6	98.4	56.5	31.8	-23.2
5	-33.4	11.2	7.2	24.6	-46.4	-18.1
6	-19.4	12.7	-59.3	0.9	-24.3	0.6

TABLE VI
The Constant b in Eq. (18) for the Three Algorithms

Location	<i>PDF2DV</i>		<i>HYB2D</i>		<i>PDF-2D-FV</i>	
	\tilde{U}_x	k	\tilde{U}_x	k	\tilde{U}_x	k
1	6.8	-40.0	-0.06	0.16	0.13	-0.73
2	32.2	-13.3	-0.06	0.06	0.51	0.11
3	-3.4	-5.9	0.16	-0.02	-0.07	-0.04
4	-9.7	-9.4	-0.11	-0.03	-0.07	-0.12
5	-3.3	-2.4	-0.05	0.12	0.01	-0.03
6	3.1	-0.4	0.01	0.06	0.01	-0.01

the finest grids using *HYB2D* (this is conjectured to be due to the instability of the solution rather than of the differencing scheme; see the Appendix).

Table VI confirms that the hybrid algorithms are successful in reducing the bias dramatically. For example, at location 1, to reduce the bias error in k to 5%, the values of N_{pc} required in the three codes is 800 (*PDF2DV*), 3 (*HYB2D*), and 15 (*PDF-2D-FV*).

4.6. Comparison of Extrapolated Results

Here, the first goal was to compare the results of the three algorithms in the limit of $M^{-2} = 0$ and $N_{pc}^{-1} = 0$. It is computationally infeasible to compute such solutions directly, but as shown by Xu and Pope [48], Richardson extrapolation is a useful technique to estimate grid converged and bias free solutions (see also the Appendix). Therefore, three runs have been performed with each algorithm, i.e., with *PDF2DV* *S48:400*, *S48:200*, and *S64:200*; with *HYB2D* *L32:50*, *L32:25*, and *L64:25*; and with *PDF-2D-FV* *T32:50*, *T32:25*, and *T64:25*, and the results of these runs have been extrapolated to $M^{-2} = 0$ and $N_{pc}^{-1} = 0$. The corresponding profiles of \tilde{U}_x , \tilde{U}_r , u'_x , u'_r , and Ω are shown in Figs. 5–9 (lines) at different axial locations.

For the mean axial velocity (Fig. 5) there is excellent agreement between the codes. The largest discrepancies in $\langle U_x \rangle / U_{bulk}$ is 5% on the centerline at $x/D_b = 1.3$. The discrepancy in the radial velocity $\langle U_r \rangle / U_{bulk}$ at $x/D_b = 1$ (Fig. 6) appears to be larger, but is less than 2%. For the r.m.s. velocities, *HYB2D* shows some discrepancy on the axis at $x/D_b = 0.4$, but otherwise the agreement between the codes is good. There is also excellent agreement for the conditional turbulence frequency (Fig. 9).

The agreement between the three codes gives confidence that each is accurate. The remaining discrepancies may be due (at least in part) to the higher-order errors that are not eliminated by the extrapolation scheme.

Since our objective was to compare different numerical algorithms in terms of efficiency and numerical accuracy, the comparison of the computed results with the experimental data was not a major part of our studies, but nevertheless, the agreement is very good considering the challenging test case and results obtained with other models. The numerical results presented in Figs. 5–8 compare very well with the experimental data by Masri [24] which are indicated by markers in the same plots showing that the PDF model applied here predicts this flow accurately. The radial velocity component is much smaller than the axial velocity component, and therefore it is not surprising that the relative differences in the radial velocity

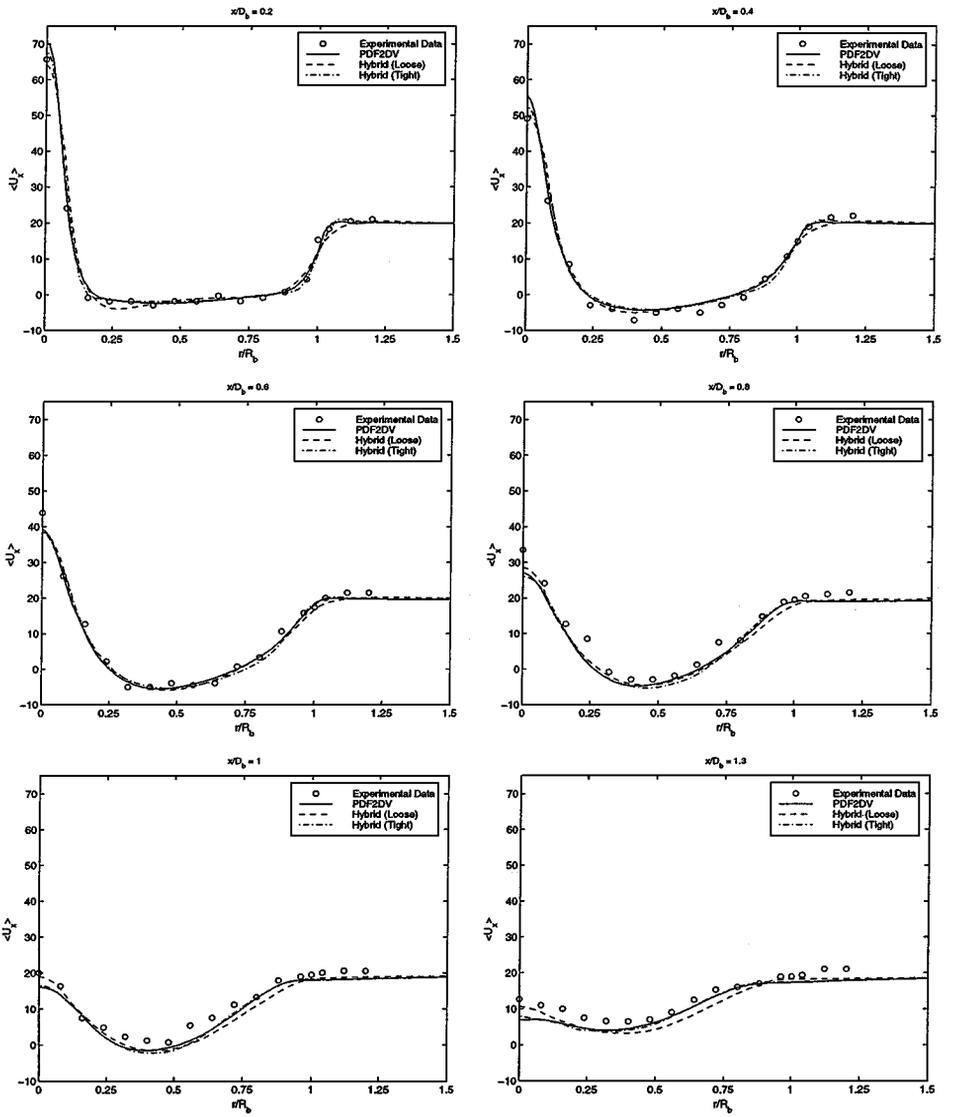


FIG. 5. Asymptotic bias free and grid converged profiles of \tilde{U}_x obtained with the three algorithms (lines) shown together with experimental data (markers) at the axial locations $0.2D_b$, $0.4D_b$, $0.6D_b$, $0.8D_b$, D_b , and $1.3D_b$ downstream of the nozzle.

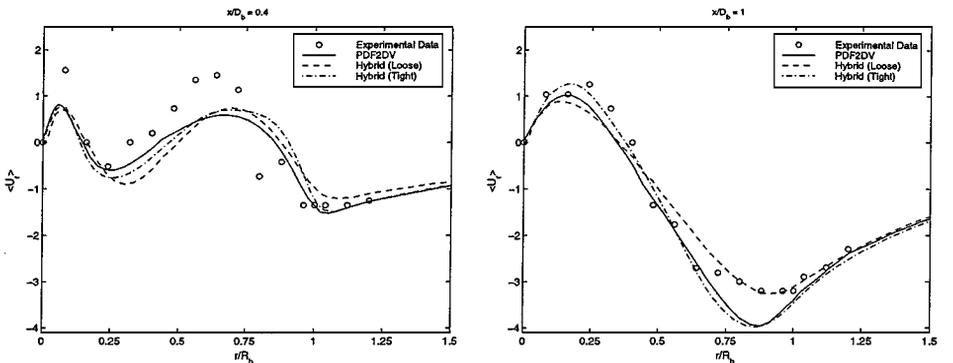


FIG. 6. Asymptotic bias free and grid converged profiles of \tilde{U}_r obtained with the three algorithms (lines) shown together with experimental data (markers) at the axial locations $0.4D_b$ and D_b downstream of the nozzle.

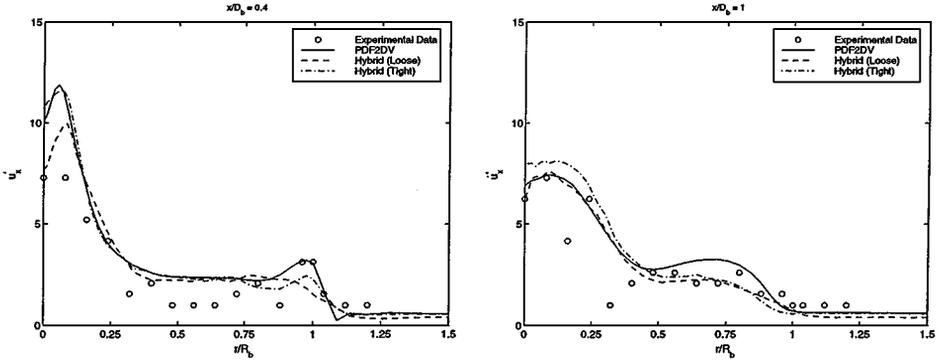


FIG. 7. Axial components of the r.m.s. velocity. Asymptotic bias free and grid converged profiles obtained with the three algorithms (lines) shown together with experimental data (markers) at the axial locations $0.4D_b$ and D_b downstream of the nozzle.

component are much larger. Considering that it is extremely difficult to predict and measure the r.m.s. velocities correctly, the differences in Fig. 7 and 8 are acceptable, if not small. It should be mentioned that the calculations (e.g., the mean centerline velocity) are found to be quite sensitive to the model constant c_{ω_1} , and the value used was chosen with respect to the experimental data.

4.7. Computational Cost

The computational cost of each particle code can be expressed as

$$T_{\text{CPU}} \approx n_{\text{conv}} M^2 (c_{\text{particle}} N_{\text{pc}} + c_{\text{overhead}}) = n_{\text{conv}} M^2 c_{\text{particle}} (N_{\text{pc}} + N_o), \quad (19)$$

where T_{CPU} is the required CPU time, and n_{conv} is the number of time steps (in the particle code) required to reach a statistically stationary state. The quantity c_{particle} is the CPU time per particle per step, while c_{overhead} is the time per step for nonparticle computation. The latter can be re-expressed as an equivalent number of particles $N_o = c_{\text{overhead}}/c_{\text{particle}}$. To determine the constants c_{particle} and c_{overhead} , the CPU times for 100 time steps have been

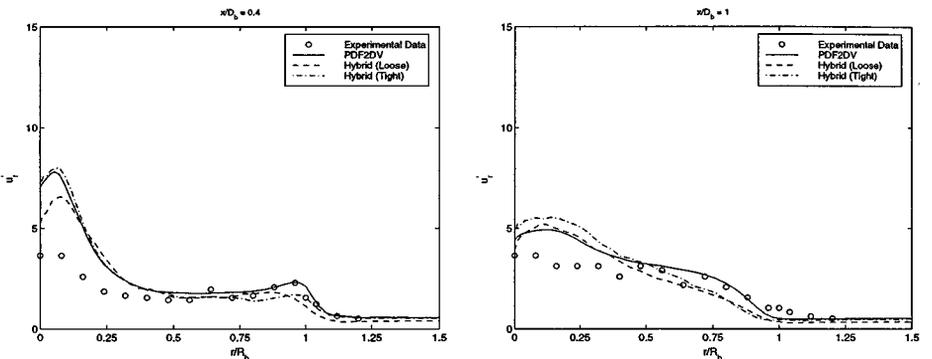


FIG. 8. Radial components of the r.m.s. velocity. Asymptotic bias free and grid converged profiles obtained with the three algorithms (lines) shown together with experimental data (markers) at the axial locations 20 mm and 50 mm downstream of the nozzle.

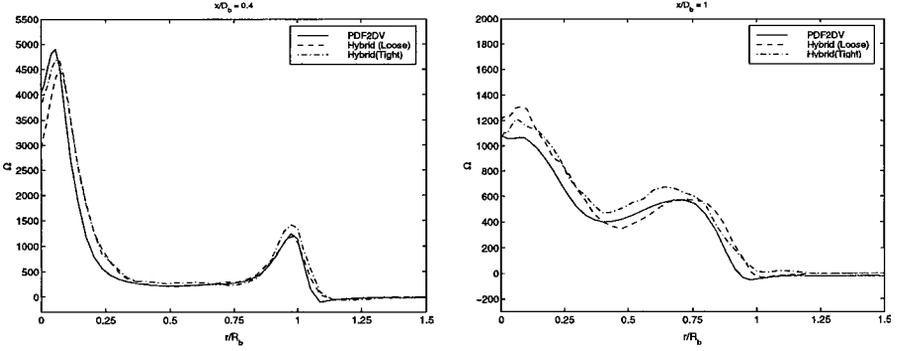


FIG. 9. Conditional turbulence frequency Ω . Asymptotic bias free and grid converged profiles obtained with the three algorithms at the axial locations 20 mm and 50 mm downstream of the nozzle.

measured using a 64×64 grid. With $N_{pc} = 50$ and $N_{pc} = 25$ for each algorithm the times T_1 and T_2 were measured, respectively. Then

$$c_{\text{particle}} = \frac{T_1 - T_2}{64^2 2500} \quad (20)$$

and

$$c_{\text{overhead}} = \frac{2T_2 - T_1}{64^2 100}. \quad (21)$$

The values of c_{particle} , c_{overhead} , and N_0 are shown in Table VII. Basically c_{particle} is the CPU time required per particle per time step and c_{overhead} is the CPU time for the overhead per cell per time step. All simulations have been performed on a 400 MHz Pentium machine. It may be seen that the values of c_{particle} are quite similar for all codes.

Finally, making the assumption that $n_{\text{conv}} \approx c_{\text{conv}} M$, the required CPU time to reach a statistical stationary state can be expressed as

$$T_{\text{CPU}} \approx c_{\text{conv}} M^3 c_{\text{particle}} (N_{pc} + N_0). \quad (22)$$

The assumption that $n_{\text{conv}} \approx c_{\text{conv}} M$ is based on the CFL criterion for particle time stepping. The constant c_{conv} can be obtained from Section 4.4 and is shown for each algorithm in Table VII. It can be seen that for *PDF2DV* and *PDF-2D-FV* the values of c_{particle} are comparable while it is slightly higher for *HYB2D*. For *PDF2DV* the overhead per particle method iteration corresponds to ≈ 15 particle time steps. For *HYB2D* and *PDF-2D-FV*, this ratio is ≈ 19 and ≈ 3 , respectively. The constant c_{conv} is determined to be 63 for *PDF2DV*, 125 for *HYB2D*, and 208 for *PDF-2D-FV*.

TABLE VII
The Constants c_{particle} , c_{overhead} and c_{conv} for the Three Algorithms

Algorithm	c_{particle}	c_{overhead}	$N_0 = \frac{c_{\text{overhead}}}{c_{\text{particle}}}$	c_{conv}
<i>PDF2DV</i>	18.5×10^{-6} s	274.8×10^{-6} s	14.9	≈ 63
<i>HYB2D</i>	25.5×10^{-6} s	473.6×10^{-6} s	18.6	≈ 125
<i>PDF-2D-FV</i>	18.7×10^{-6} s	55.4×10^{-6} s	3.0	≈ 208

4.8. Comparison of Computational Efficiency

As shown above and in [48], Richardson extrapolation is effective in reducing the total CPU time required to reduce the numerical error below a given threshold. But clearly there is advantage in being able to obtain accurate results with a single run.

In this section we use the characterization of the error and CPU times obtained in Section 4.5 and 4.7 to address the computational efficiency of the different codes, i.e., the CPU time required to achieve a given error level in a single run. To compare the

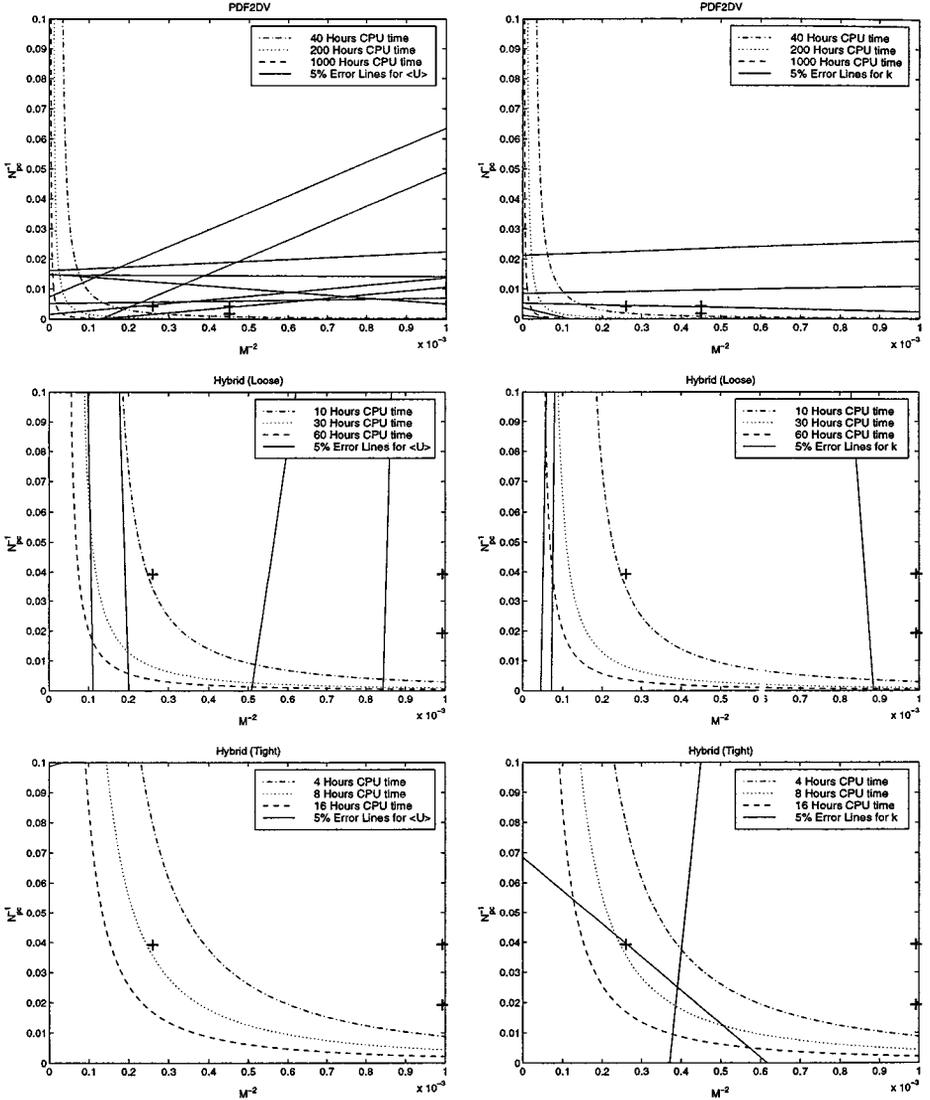


FIG. 10. Locations of simulations in the $M^{-2} - N_{pc}^{-1}$ plane (markers). First row: stand-alone particle-mesh method; second row: loosely coupled hybrid algorithm; third row: tightly coupled hybrid algorithm. The straight lines show where the numerical error is 5% with respect to a reference value (left plots: \tilde{U}_x at location 1–6; right plots: k at location 1–6). The reference value for the velocity is 61 m/s (jet bulk velocity) and for k it is $127 \text{ m}^2/\text{s}^2$ (largest value of the turbulent kinetic energy at the inlet boundary). Along the curved lines T_{CPU} is approximately constant.

TABLE VIII
Efficiency of the Three Different PDF Solution Algorithms
for a Given Numerical Error Tolerance of 5%

Code	T_{CPU}	n_{total}
<i>PDF2DV</i>	2600 h	1.23×10^9
<i>HYB2D</i>	60 h	2.54×10^7
<i>PDF-2D-FV</i>	7 h	2.25×10^7

Note. Shown is CPU time (on a 400 MHz pentium machine) for the worst case from the locations 1–6 of Fig. 10. The values of n_{total} show the total number of particle time steps.

computational efficiency of the three algorithms the results of Section 4.6 and 4.7 are used to quantify T_{CPU} of the three algorithms for a given error tolerance.

The markers in Fig. 10 indicate the locations of the runs in the $M^{-2} - N_{\text{pc}}^{-1}$ plane, i.e., in the first row the runs *S48:400*, *S48:200*, and *S64:200*; in the second row, the runs *L32:50*, *L32:25*, and *L64:25*; and in the third row, the runs *T32:50*, *T32:25*, and *T64:25*. To show how large the numerical error is with respect to the reference values ($\tilde{U}_{x,\text{ref}} = 61$ m/s and $k_{\text{ref}} = 127$) 5% error iso-lines (straight lines) are shown for the quantities \tilde{U}_x (left plots) and k (right plots) at locations 1–6 (here the values of Table V and VI are used). To compare the efficiency of the codes for given accuracy, lines of constant T_{CPU} are shown (curved lines) using Eq. (22) and the values from Table VII. This allows determination of the CPU time required to get statistically stationary results within a specified numerical error tolerance. In Table VIII the worst case of the previous studies is shown for the three algorithms. It can be seen that the bias error is significantly reduced by the hybrid approach. For this test case, the hybrid method *HYB2D* is ≈ 43 times more efficient than *PDF2DV*, and *PDF-2D-FV* is ≈ 370 times more efficient than *PDF2DV* (Table VIII). Since the total number of particle time steps required for the hybrid methods is much smaller (Table VIII) than for *PDF2DV*, this difference could even be more significant for reacting cases with detailed chemistry.

5. CONCLUSION

A constant-density bluff-body stabilized flow test case is used to compare (in terms of numerical accuracy and efficiency) three different PDF solution algorithms, which solve the same modeled JPDF equation. The first algorithm is a stand-alone particle-mesh method, and the other two are consistent hybrid algorithms. We have shown that we obtain converged solutions with all the three algorithms in terms of grid refinement and particle numbers. We could also show that the calculations are in the asymptotic range and therefore that extrapolation can be applied to minimize the numerical error. The extrapolated results with the three different algorithms are in very good agreement with each other, and predict the flow accurately compared to experimental data. This shows that the numerical algorithms converge to the same solution, and the PDF modeling is a powerful approach to simulate this type of flow.

Rigorous asymptotic studies have been performed to compare the three solution algorithms in terms of numerical accuracy and efficiency. It is confirmed that both hybrid

methods are effective at virtually eliminating the bias; and it has been found that the hybrid methods outperform the stand-alone particle-mesh method. For this test case it is shown that, for a single run, the consistent hybrid approach can be ≈ 370 times more efficient than *PDF2DV* in terms of CPU time for a given numerical error tolerance of 5%.

APPENDIX: STATIONARY SOLUTIONS TO THE PDF EQUATION

Needless to say, given the complexity of the modeled PDF evolution equation, little can be said with certainty about the existence, uniqueness, and stability of stationary solutions (in general, or for a particular flow). Nevertheless, from our experience with this flow, a consistent nontrivial picture emerges, which has profound consequences for the grid-convergence studies reported in the text.

To describe this picture, we denote by $f(t, \varepsilon)$ the PDF at time t calculated by one of the numerical methods on a grid with $M^2 = \varepsilon^{-1}$ cells (this is an abbreviated notation, since the PDF also depends on \mathbf{x} and the sample-space variables). The evolution equation for $f(t, \varepsilon)$ is written simply as

$$\frac{\partial}{\partial t} f(t, \varepsilon) = F(f(t, \varepsilon), \varepsilon). \quad (23)$$

If a stationary solution $f(t, \varepsilon) = \bar{f}(\varepsilon)$ exists, then it satisfies

$$F(\bar{f}(\varepsilon), \varepsilon) = 0, \quad (24)$$

and it may be stable or unstable. That is, if the evolution equation, Eq. (23), were solved from an initial condition very close to $\bar{f}(\varepsilon)$, the solution $f(t, \varepsilon)$ would tend to $\bar{f}(\varepsilon)$ if it is a stable stationary solution. If the solution is unstable, however, then $f(t, \varepsilon)$ would not tend to $\bar{f}(\varepsilon)$, but may be instead exhibit chaotic behavior.

Since the numerical methods considered are second-order accurate, for sufficiently small ε , if a stable solution $\bar{f}(\varepsilon)$ exists, then it varies linearly with ε . This is the basis of the Richardson extrapolation used to obtain an approximation \bar{f}_R to the exact stationary solution $\bar{f}(0)$:

$$\bar{f}(0) \approx \bar{f}_R(\varepsilon_1, \varepsilon_2) = \frac{\varepsilon_2 \bar{f}(\varepsilon_1) - \varepsilon_1 \bar{f}(\varepsilon_2)}{\varepsilon_2 - \varepsilon_1}. \quad (25)$$

Note that the different numerical methods are likely to have different dependences on ε , but, if they are consistent, their asymptotic solutions $\bar{f}(0)$ are the same.

With this background we can now state our observations, and the consistent picture that emerges.

1. With each numerical method, stationary solutions are obtained for a range of grids that are not too fine, i.e., for ε greater than some value ε^* .
2. On finer grids, ($\varepsilon < \varepsilon^*$) quasi-periodic or chaotic solutions are observed.
3. Over a range of ε for which stationary solutions are obtained, each method exhibits a linear dependence on ε (to a reasonable approximation).
4. The extrapolated solutions for \bar{f}_R obtained from the different methods are in reasonably good agreement with each other.

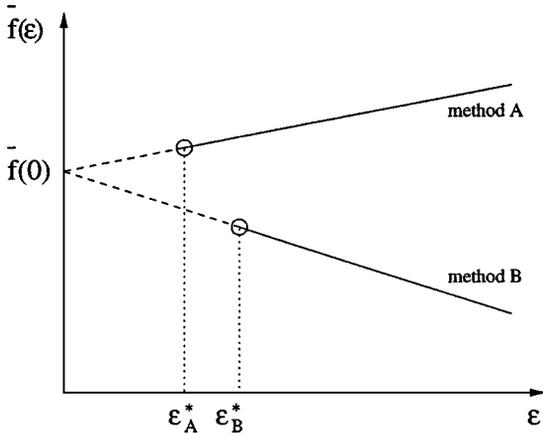


FIG. 11. Sketch of PDF solutions by two methods as a function of the $\epsilon = M^{-2}$ showing stable regions (solid lines) and unstable (dashed lines).

These observations are consistent with the picture shown in Fig. 11: For all ϵ there is a stationary solution, but the solution is unstable below a critical value of ϵ (which depends on the method).

Two important deductions follow. First, even though stationary fine-grid solutions cannot be achieved, the Richardson extrapolation yields a consistent estimate of the error-free (unstable) stationary solution. Second, the large amount of fourth-order dissipation found necessary in *HYB2D* to obtain stationary solutions may well be due to the instability of the solution, not to inherent instabilities in the numerical method.

ACKNOWLEDGMENT

This work has been supported in part by the Department of Energy, Grant DE-FG02-90ER14128.

REFERENCES

1. M. S. Anand, A. T. Hsu, and S. B. Pope, Calculations of swirl combustors using joint velocity-scalar probability density function method, *AIAA J.* **35**, 1143 (1997).
2. M. S. Anand, S. B. Pope, and H. C. Mongia, A PDF method for turbulent recirculating flows, in *Lecture Notes in Engineering*, (Springer-Verlag, Berlin/New York, 1989), pp. 672–693.
3. D. A. Caughey, Diagonal implicit multigrid algorithm for the Euler equations, *AIAA J.* **26**, 841 (1988).
4. G.-C. Chang, *A Monte Carlo PDF/Finite-Volume Study of Turbulent Flames*, Ph.D. thesis (Cornell University, 1996).
5. A. J. Chorin, A numerical method for solving incompressible viscous flow problems, *J. Comput. Phys.* **2**, 12 (1967).
6. S. M. Correa and S. B. Pope, Comparison of a Monte Carlo PDF finite-volume mean flow model with bluff-body Raman data, in *Twenty-Fourth Symposium (Int'l) on Combustion*, (Combustion Institute, Pittsburgh, PA, 1992), p. 279–285.
7. S. M. Correa and S. B. Pope, Measurements and modeling of a bluff-body stabilized flame, *Combust. Flame*, **89**, 195 (1992).

8. B. B. Dally, D. F. Fletcher, and A. R. Masri, Flow and mixing fields of turbulent bluff-body jets and flames, *Combust. Theory Modelling* **2**, 193 (1998).
9. B. J. Delarue and S. B. Pope, Application of PDF methods to compressible turbulent flows, *Phys. Fluids* **9**, 2704 (1997).
10. B. J. Delarue and S. B. Pope, Calculations of subsonic and supersonic turbulent reacting mixing layers using probability density function methods, *Phys. Fluids* **10**, 487 (1998).
11. C. Dopazo, Recent developments in pdf methods, in *Turbulent Reacting Flows*, edited by P. A. Libby and F. A. Williams (Academic Press, London, 1994), Chapter 7, pp. 375-474.
12. T. D. Dreeben and S. B. Pope, Nonparametric estimation of mean fields with application to particle methods for turbulent flows, Unpublished, Technical Report FDA 92-13 (Cornell University, 1992).
13. T. D. Dreeben and S. B. Pope, PDF/Monte Carlo simulation of near-wall turbulent flows, *J. Fluid Mech.* **357**, 141 (1998).
14. D. C. Haworth and S. B. Pope, A generalized Langevin model for turbulent flows, *Phys. Fluids* **29**, 387 (1986).
15. Y. H. Choi and C. L. Merkle, Application of preconditioning in viscous flow, *J. Comput. Phys.* **105**, 207 (1993).
16. R. W. Hockney and J. W. Eastwood, *Computer Simulations using Particles* (Hilger, Bristol, 1988).
17. Jayesh and S. B. Pope, Stochastic model for turbulent frequency, Unpublished, Technical Report FDA 95-05 (Cornell University, 1995).
18. P. Jenny and B. Müller, Convergence acceleration for computing steady state compressible flow at low Mach numbers, *Combust. Flame* **28**, 951(1999).
19. P. Jenny, S. B. Pope, M. Muradoglu, and D. A. Caughey, A hybrid algorithm for the joint pdf equation of turbulent reactive flows, *J. Comput. Phys.*, in press.
20. W. P. Jones, Turbulence modelling and numerical solution methods for variable density and combusting flows, in *Turbulent Reacting Flows*, edited by P. A. Libby and F. A. Williams. (Academic Press, London, 1994), p. 309.
21. M. H. Kalos and P. A. Whitlock, *Monte Carlo Methods* (Wiley, New York, 1986).
22. B. E. Launder, Phenomenological modelling: Present...and future? in *Whither Turbulence? Turbulence at the Crossroads*, Lecture Notes in Physics, edited by J. L. Lumley. Springer-verlag, Berlin, 1990, Vol. 357.
23. B. E. Launder and D. B. Spalding, *Mathematical Models of Turbulence* (Academic Press, London, 1972).
24. A. R. Masri, Technical Report (The University of Sydney), available at <http://www.mech.eng.usyd.edu.au/research/energy/>.
25. A. R. Masri, R. W. Dibble, and R. S. Barlow, The structure of turbulent nonpremixed flames revealed by Raman-Rayleigh-LIF measurements, *Prog. Energy Combust. Sci* **22**, 307 (1996).
26. A. R. Masri and S. B. Pope, PDF calculations of piloted turbulent non-premixed flames of methane, *Combust. Flame* **81**, 13 (1990).
27. J.-P. Minier and J. Pozorski, Analysis of a PDF model in a mixing layer case, in *Proceedings of Tenth Symposium on Turbulent Shear Flows*, 1995, pp. 26.25-26.30.
28. M. Muradoglu and D. A. Caughey, Implicit multigrid solution of the preconditioned multi-dimensional Euler equations, *AIAA J.* **98-0114** (1998).
29. M. Muradoglu, P. Jenny, S. B. Pope, and D. A. Caughey, A consistent hybrid finite-volume/particle method for the pdf equations of turbulent reactive flows, *J. Comput. Phys.* **154**, 342 (1999).
30. P. A. Nooren, H. A. Wouters, T. W. J. Peeters, D. Roekaerts, U. Maas, and D. Schmidt, Monte Carlo PDF simulation of a turbulent natural-gas diffusion flame, in *Twenty-Sixth Symposium (Int'l) on Combustion* (Combustion Institute, Pittsburgh, PA, 1996).
31. S. B. Pope, A Monte Carlo method for the PDF equations of turbulent reactive flow, *Combust. Sci. Technol.* **25**, 159 (1981).
32. S. B. Pope, PDF methods for turbulent reactive flows, *Prog. Energy Combust. Sci.* **11**, 119 (1985).
33. S. B. Pope, Computations of turbulent combustion: progress and challenges, in *Twenty-Third Symposium (Int'l) on Combustion* (Combustion Institute, Pittsburgh, PA, 1990) pp. 591-612.
34. S. B. Pope, Lagrangian PDF methods for turbulent flows, *Ann. Rev. Fluid. Mech.* **26**, 23 (1994).

35. S. B. Pope, On the relationship between stochastic Lagrangian models of turbulence and second-moment closures, *Phys. Fluids* **6**, 973 (1994).
36. S. B. Pope, PDF2DV: A Fortran code to solve the modelled joint PDF equations for two-dimensional recirculating flows. Unpublished (Cornell University, 1994).
37. S. B. Pope, Particle method for turbulent flows: Integration of stochastic model equations, *J. Comput. Phys.* **117**, 332 (1995).
38. S. B. Pope, *Turbulent Flows*, (Cambridge University Press, Cambridge, UK, (2000).
39. S. B. Pope and Y. L. Chen, The velocity-dissipation probability density function model for turbulent flows, *Phys. Fluids A*, **2**, 1437 (1990).
40. W. M. Roquemore, R. L. Britton, and S. S. Sandhu, Investigation of the dynamic behaviour of a bluff-body diffusion flame using flame emission, *AIAA J.* **0178** (1982).
41. V. Saxena and S. B. Pope, PDF calculations of major and minor species in a turbulent piloted jet flame (Combustion Institute, Pittsburgh, PA, 1998), pp. 1081–1086.
42. E. Turkel, Preconditioned methods for solving incompressible and low speed compressible flows, *J. Comput. Phys.* **72**, 277 (1987).
43. E. Turkel, A review of preconditioning methods for fluid dynamics, *Appl. Numer. Math.* **12**, 257 (1993).
44. P. R. Van Slooten, Jayesh, and S. B. Pope, Advances in PDF modeling for inhomogeneous turbulent flows, *Phys. Fluids* **10**, 246, (1998).
45. P. R. Van Slooten and S. B. Pope, PDF modeling of inhomogeneous turbulence with exact representation of rapid distortions, *Phys. Fluids* **9**, 1085 (1997).
46. P. R. Van Slooten and S. B. Pope, Application of PDF modeling to swirling and non-swirling turbulent jets, *Flow, Turbulence and Combust.* **62**, 295 (1999).
47. D. C. Wilcox, *Turbulence Modeling for CFD* (DCW Industries, La Cañada, CA, 1993).
48. J. Xu and S. B. Pope, Assessment of numerical accuracy of PDF/Monte Carlo methods for turbulent reactive flows, *J. Comput. Phys.* **152**, 192 (1999).