Available online at www.sciencedirect.com
ScienceDirect
Mathematics and Computers in Simulation 207 (2023) 97–110
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original articles

An adaptive numerical method for simulating diffusion flame jets

Priscila C. Calegari[a,*], Alexandre M. Roma[b], Luis C.C. Santos[b], Guenther C. Krieger Filho[c]

[a] *Department of Computing, Federal University of Santa Catarina, Araranguá, Brazil*
[b] *Department of Applied Mathematics, University of São Paulo, São Paulo, Brazil*
[c] *Department of Mechanical Engineering, University of São Paulo, São Paulo, Brazil*

Received 14 September 2021; received in revised form 4 May 2022; accepted 26 December 2022
Available online 30 December 2022

Abstract

In this paper, we present an adaptive numerical methodology for a reacting three-dimensional low-Mach number flow. This computational strategy combines an adaptive mesh refinement (AMR), an implicit–explicit time stepping strategy (IMEX), an extension of increment-pressure projection method, and a mixture fraction to model the chemistry combustion dynamics. To accurately resolve sharp gradients, vorticity shedding, and localized small length scale flow features, dynamic adaptive mesh refinement (given by hierarchical nested Cartesian grid patches) is employed. That spatial dynamic adaptation is used in conjunction with a variable time step, second-order, linearly implicit time integration scheme. The capabilities of the present numerical method are demonstrated by numerical verification and simulation of two classical diffusion flame examples from literature.
© 2022 International Association for Mathematics and Computers in Simulation (IMACS). Published by Elsevier B.V. All rights reserved.

Keywords: Adaptive mesh refinement; Low-mach number flow; Mixture fraction; Projection method

1. Introduction

Combustion is an important research area in Fluid Dynamics which involves many phenomena such as high speed flows, heat transfer, chemical reactions, and turbulence — among others. The understanding of these phenomena is of significant technological and scientific interest since they are present in a variety of industrial and natural processes. Computer simulation is one of the most fundamental tools employed to that end. As an example, one may mention non-premixed flames [8,17,37,41] which are being studied in different scenarios. Computer simulations of a three-dimensional unsteady reacting low-Mach number flows can be computationally quite expensive. Grid resolution and detailed chemical mechanism demand huge computational resources due to the total number of chemical species and to time-step restrictions because of the stiffness present in the reaction kinetics.

In this context, adaptivity is an important component to be considered for efficient numerical solutions of the partial differential equations involved. Many techniques appeared through the years and, nowadays, the term "adaptive mesh refinement" (or simply AMR) embraces an entire collection of approaches which spread to a variety

* Corresponding author.
E-mail address: priscila.calegari@ufsc.br (P.C. Calegari).

https://doi.org/10.1016/j.matcom.2022.12.021
0378-4754/© 2022 International Association for Mathematics and Computers in Simulation (IMACS). Published by Elsevier B.V. All rights reserved.

of different application fields [31]. However, the key idea behind all those approaches is still the same: to concentrate computational power where it is most needed by increasing the resolution in space on regions of special interest in the computational domain. Several refinement criteria may be used to identify these regions. For example, one needs more grid resolution where there are immersed interfaces/bodies to describe accurately intricate geometry details, where there is high vorticity/turbulence, where boundary layers develop, and, more generally, where there are flow features of special interest (with highly localized span) [3,9,13,14,27,36].

There are several studies which have introduced numerical methods for simulating non-premixed laminar flames. Hilditch and Colella [18] have proposed a numerical technique that includes fast chemistry. The reaction effects are generated using a conserved scalar, the mixture fraction. According to Bilger [7] the mixture fraction has been an important concept to understand many phenomena, as mixing and combustion in non-premixed turbulent flows. Pember et al. [27] have presented an adaptive projection method for low-Mach number combustion with a reduced kinetics mechanism. Mohammed et al. [23] have investigated computationally and experimentally the structure of a forced, time-varying, axisymmetric laminar methane–air diffusion flame. Day and Bell [13] have presented a generalization of the numerical methodology proposed in [27] that incorporates complex chemistry for simulating premixed and non-premixed flames described in [23].

The purpose of this work is to use a simplified chemistry model where a conserved scalar is responsible for all reaction effects in conjunction with an adaptive mesh refinement strategy in order to reduce the computational cost. To this end, we combine here methods and techniques discussed and developed in several studies [1,9,18,27] to perform computer simulations of reacting low-Mach number flows. The main ingredients of this combination are an implicit–explicit time integration scheme (IMEX [1,40]), a fast chemistry non-premixed combustion model [18,32], and an extension of a projection method [13,27] to solve the resulting equations. The present work extends the numerical methods introduced previously in different contexts [9–11,14,26,30,36] to reacting flow problems. It includes a verification analysis of the adaptive numerical technique proposed and its application to perform simulations of three-dimensional non-premixed flame jets.

Next section presents a mathematical formulation for the unsteady low-Mach number reacting flow. Section 3 describes the numerical methods, including the time–space discretization, and an extension of projection method to be used. Section 4 presents the numerical experiments which include, besides a numerical verification of the overall algorithm, diffusion flame jet simulations. Concluding remarks appear in Section 5.

2. Mathematical formulation

The mathematical formulation is based on a low-Mach number combustion model proposed by [22,34]. In the low-Mach number limit [13,27], we may write the pressure

$$p(\mathbf{x}, t) = p_0 + \widetilde{p}(\mathbf{x}, t), \tag{1}$$

where p_0 is the constant thermodynamic pressure and $\widetilde{p}$ is the dynamic pressure. Thus, the equations expressing the mass and linear momentum balances, and a conserved scalar transport (for chemical species and temperature) are given by

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{2}$$

$$\frac{\partial (\rho z)}{\partial t} + \nabla \cdot (\rho \mathbf{u} z) = \nabla \cdot (\rho D \nabla z), \tag{3}$$

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla \widetilde{p} + \nabla \cdot \left[\mu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T - \frac{2}{3} \mathbf{I} \nabla \cdot \mathbf{u} \right) \right] + \mathbf{f}, \tag{4}$$

where ρ is the density, $\mathbf{u}$ the velocity, μ the viscosity, $\mathbf{I}$ is the identity tensor, $\mathbf{f}$ represents the sum of all external forces acting on the fluid, z is the conserved scalar, and D the diffusion coefficient (the same for all chemical species). The set of governing equations is closed if we assume valid the equation of state for an ideal gas law,

$$p_0 = \frac{\rho R_u T}{W}, \tag{5}$$

where T is temperature distribution, $W = (\sum_k Y_k / W_k)^{-1}$ is the mixture-averaged molecular weight, and R_u is the universal gas constant. The terms Y_k and W_k are, respectively, the mass fraction and the molecular weight of the

kth chemical species. By differentiating (5) as in [13,18,27] with respect to time and by using Eq. (2), we obtain the constraint on the divergence of the velocity,

$$\nabla \cdot \mathbf{u} = \frac{1}{T}\frac{DT}{Dt} + W \sum_k \frac{1}{W_k}\frac{DY_k}{Dt} \equiv S, \tag{6}$$

where the differential operator D/Dt is the material derivative and $S \neq 0$.

We have considered a fast chemistry model (also called Burke–Schumann flame structure) for a non-premixed flame. In other words, we assume a one step infinitely fast and irreversible chemistry reaction, where the Lewis number is unity [32,35,39]. In this case, fuel and oxidizer coexist only in a thin reaction zone. The chemical reaction involves only fuel (F), oxidizer (O), and products (P): $\upsilon_F F + \upsilon_O O \longrightarrow \upsilon_P P$, where υ_k are the stoichiometric coefficients of species k with $k = F, O, P$.

The combustion process is described using a conserved scalar variable z which involves chemical species and temperature. Following Poinsot and Veynante [32], we obtain (3) from the conservation equation for the kth mass fraction and temperature which is given by

$$\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot (\rho \mathbf{u} Y_k) = \nabla \cdot (\rho D \nabla Y_k) + \dot{\omega}_k, \quad \text{with} \quad k = F, O, \tag{7}$$

$$\frac{\partial \rho T}{\partial t} + \nabla \cdot (\rho \mathbf{u} T) = \nabla \cdot \left(\frac{\kappa}{c_p}\nabla T\right) + \dot{\omega}_T, \tag{8}$$

where $\dot{\omega}_k$ is the production rate for ρY_k, $\dot{\omega}_T$ the heat release due to chemical reactions, κ thermal conductivity, and c_p the constant pressure specific heat capacity.

The product mass fraction Y_P is given by the mass conservation law $Y_P = 1 - (Y_O + Y_F)$. Moreover, as the Lewis number is unity, we obtain $\rho D = \kappa/c_p$ which means that all chemical species and temperature diffuse in the same way. In stoichiometric chemical reactions, fuel and oxidizer react completely, combining instantly into reaction products. Thus, the oxidizer consumption rate satisfies $\dot{\omega}_O = s\dot{\omega}_F$ and the heat release rate is $\dot{\omega}_T = -\Delta h_F^* \dot{\omega}_F$, where $s = (\upsilon_O W_O)/(\upsilon_F W_F)$ is the mass stoichiometric ratio, and Δh_F^* is the heat of formation of Y_F. The stoichiometric value is given by $z_{st} = 1/(1+\Phi)$, where this quantity Φ is the ratio of fuel to oxidizer in the injected streams with respect to the ratio corresponding to stoichiometric conditions, given by $\Phi = s(Y_F^\star/Y_O^\star)$, where $Y_O^\star$ is the stream oxidizer mass fraction and $Y_F^\star$ is the stream fuel mass fraction. Therefore, combining Eqs. (7)–(8), we can define three conserved scalars,

$$Z_1 = sY_F - Y_O, \quad Z_2 = \frac{c_p T}{\Delta h_F^*} + Y_F, \quad Z_3 = \frac{s c_p T}{\Delta h_F^*} + Y_O. \tag{9}$$

The scalar Z_1 is called *mixture fraction*. Each scalar Z_i, $i = 1, 2, 3$, satisfies (3). In non-premixed flames, they assume two values in the reaction zone inlet: $Z_{i,F}$ (inlet fuel) and $Z_{i,O}$ (inlet oxidizer). Therefore, we can define the normalized scalar, $z = (Z_i - Z_{i,O})/(Z_{i,F} - Z_{i,O})$. Thus, when $Z_i = Z_{i,F}$ we have $z = 1$ (fuel indication) and when $Z_i = Z_{i,O}$, $z = 0$ (oxidizer indication).

The Burke–Schumann flame structure [32,35] connects temperature and chemical species with conserved scalar z assuming that fuel and oxidizer do not coexist. If the mixture is *lean* ($z \leq z_{st}$) then $Y_F(z) = 0$ and

$$\begin{aligned} Y_O(z) &= Y_O^\star \left(1 - \frac{z}{z_{st}}\right), \\ T(z) &= zT_F^\star + (1-z)T_O^\star + \frac{\Delta h_F^* Y_F^\star}{c_p} z, \end{aligned} \tag{10}$$

where $T_F^\star$ and $T_O^\star$ are the fuel and oxidizer temperature stream. If the mixture is *rich* ($z \geq z_{st}$) then $Y_O(z) = 0$ and

$$\begin{aligned} Y_F(z) &= Y_F^\star \frac{z - z_{st}}{1 - z_{st}}, \\ T(z) &= zT_F^\star + (1-z)T_O^\star + \frac{\Delta h_F^* Y_F^\star}{c_p} z_{st} \frac{1-z}{1-z_{st}}. \end{aligned} \tag{11}$$

The flame position occurs when Y_F and Y_O are both null, i.e. $z = z_{st}$. This means that fuel and oxidizer became reaction products and the temperature of the flame gets the maximum value. Fig. 1 presents the possible states of a flame.

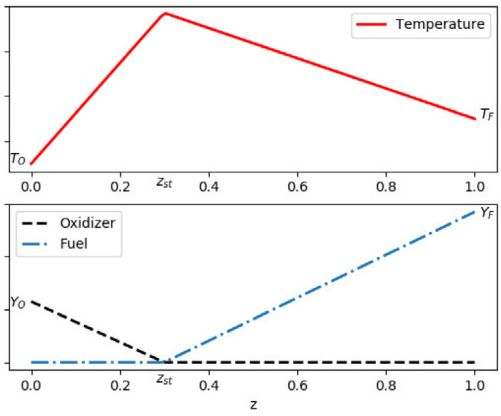

Fig. 1. Example of diffusion flame structure in the conserved scalar z.

Chemical reactions may contain inert species, such as the Nitrogen (N_2) [32,39]. In this case, the inert mass fraction is obtained by $Y_I(z) = Y_{I,O}(1 - z) + Y_{I,F}z$, where $Y_{I,O}$ is the inert mass fraction in the oxidizer stream and $Y_{I,F}$ is the inert mass fraction in the fuel stream. In this case, the product mass fraction is given by $Y_P = 1 - (Y_O + Y_F + Y_I)$.

In summary, our mathematical model is given by (2)–(4), with temperature, oxidizer, and fuel mass fraction given by (10)–(11). The mathematical model requires boundary and initial conditions which will be detailed in Section 4. Following Pember et al. [27] the coefficients, μ and D, are given, respectively, by $\mu = \mu_O(T/T_O)^{0.7}$ and $\rho D = \mu/Pr$, with $\mu_O = 1.85 \times 10^{-5}$kg/m.sec, $T_O = 298$ K, and $Pr = 0.75$. Next, we present the adaptive numerical method.

3. Numerical method

We present in this section the combination of an adaptive spatial discretization and a variable-step time integration with an extended version of an increment-pressure projection method, geared by a multilevel-multigrid method to solve linear systems.

3.1. Adaptive spatial discretization

In the discretization of the computational domain, we follow closely the adaptive mesh refinement technique (AMR) first proposed by Berger and Colella [4–6]. The technique is based on a grid structure given by a set of nested, Cartesian grid patches which form a level hierarchy, usually referred to as *composite grid*. These rectangular grid patches obey certain rules, targeting for easiness in their construction and efficiency in their use:

1. a fine grid starts and ends at the corner of a cell in the next coarser grid, and
2. all fine grid cells at level l must be surrounded either level l cells or by level $l - 1$ cells except when it touches the border of the physical domain.

In a composite grid a level l is given by the union of the grid patches with the same computational cell size, i.e., $G_l = \bigcup_{k=1}^{n_l} G_{l,k}$, where $G_{l,k}$ is the kth grid patch and n_l is the number of patches in level l. Two different grid patches in the same level do not overlap, i.e., $G_{l,i} \cap G_{l,j} = \emptyset$, if $i \neq j$. The composite grid is given by $G = \bigcup_{l=0}^{L} G_l$, where L is the number of levels and G_l is a set of grid patches in level l. Fig. 2 shows an example of two-dimensional composite grid.

Based on techniques common in computer vision and pattern recognition, the mesh generation is performed by the *point clustering algorithm* proposed by Berger and Rigoutsos [6]. The number of refinement levels is an input data. First, we define the refinement criteria which is responsible for flagging grid points needing more resolution (e.g. regions with some special feature such as high temperature, high vorticity, high turbulence, and so on — see Section 4). Afterwards, the point clustering algorithm returns an optimal set of blocks enclosing all the flagged

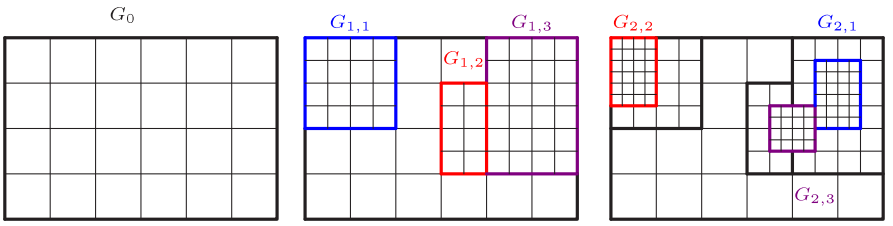

Fig. 2. A sequence of nested grid patches.

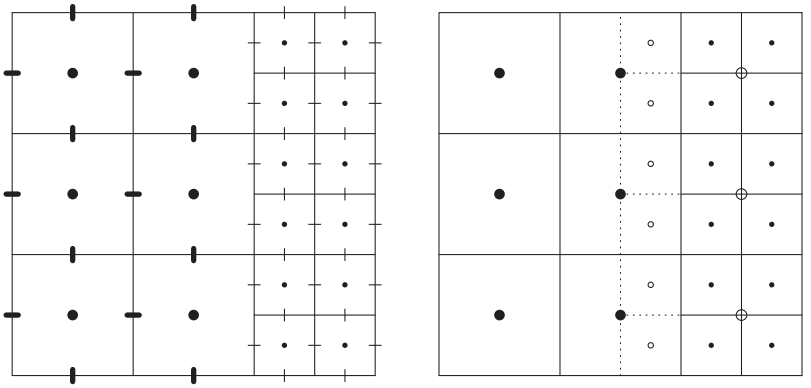

Fig. 3. (a) MAC fashion variable: • scalars and - vector field and (b) Computational cells: • visible cells and ○ auxiliary cells (*ghosts* and covered).

points to generate the new composite grid. For efficiency, our simulations use composite grids for which at least 85% of cells have been flagged.

We place the variables in a "marker-and-cell" (MAC) fashion: scalars at the cell centers (e.g. density, pressure, temperature, conserved scalar and viscosity), and vectors have their components on cell faces. It is important to highlight that the existence of computational *ghost cells*, which form layers around each grid patch $G_{l,k}$, is an essential part of this technique. Ghost cells furnish boundary conditions from polynomial interpolations of values at neighboring coarse/fine levels. Fig. 3(a) displays variable location in a MAC fashion and Fig. 3(b) displays auxiliary ghost and covered cells for a sample two-dimensional grid.

We use standard second-order finite difference operators for the discretizations of the gradient and divergent differential operators as well as for the stress tensor [9,36]. Such discretizations need only one layer of ghost cells along the interface between patches. In the conservation of mass and conserved scalar equations, (2) and (3) respectively, we use a third-order Quadratic Upstream Interpolation for Convective Kinematics (QUICK) for the advective terms [16,39]. They need two ghost cells along the interfaces between patches.

3.2. Temporal discretization

We perform a variable-step time discretization using a linearly-implicit integration scheme based on a second-order, two-step IMEX (Implicit–Explicit) method [1,40]. Nonlinearities in the diffusive term (e.g. coming from the turbulence modeling) are handled by rewriting (4) as

$$\frac{\partial \mathbf{u}}{\partial t} = \frac{1}{\rho}\left(\lambda \Delta \mathbf{u} - \nabla \widetilde{p} + \widetilde{\mathbf{f}}\right), \tag{12}$$

where $\widetilde{\mathbf{f}} = \nabla \cdot \left[\mu \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T - \frac{2}{3} \nabla \cdot \mathbf{u} \right) \right] - \rho \mathbf{u} \cdot \nabla \mathbf{u} - \lambda \Delta \mathbf{u} + \mathbf{f}$ and λ is an *ad hoc* constant chosen through numerical experimentation. The resulting time integration scheme is

$$\begin{aligned} \frac{\alpha_2 \mathbf{u}^{n+1} + \alpha_1 \mathbf{u}^n + \alpha_0 \mathbf{u}^{n-1}}{\Delta t^{n+1}} = & \frac{\theta_2}{\rho^{n+1}} (\lambda \Delta \mathbf{u}^{n+1} - \nabla \widetilde{p}^{n+1}) + \frac{\theta_1}{\rho^n} (\lambda \Delta \mathbf{u}^n - \nabla \widetilde{p})^n + \\ & \frac{\theta_0}{\rho^{n-1}} (\lambda \Delta \mathbf{u}^{n-1} - \nabla \widetilde{p}^{n-1}) + \frac{\beta_1 \widetilde{\mathbf{f}}^n}{\rho^n} + \frac{\beta_0 \widetilde{\mathbf{f}}^{n-1}}{\rho^{n-1}}, \end{aligned} \tag{13}$$

where Δt^{n+1} is the time-step size in t^{n+1}, the parameters α_i, β_i and θ_i are given by

$$\alpha_2 = \frac{\Delta t^n + 2\gamma \Delta t^{n+1}}{\Delta t^n + \Delta t^{n+1}}, \ \alpha_1 = \frac{\Delta t^{n+1} - \Delta t^n - 2\gamma \Delta t^{n+1}}{\Delta t^n}, \alpha_0 = -\alpha_2 - \alpha_1, \tag{14}$$

$$\beta_1 = \frac{\Delta t^n + \gamma \Delta t^{n+1}}{\Delta t^n}, \ \beta_0 = -\frac{\gamma \Delta t^{n+1}}{\Delta t^n}, \tag{15}$$

$$\theta_2 = \gamma + c \frac{\Delta t^{n+1}}{\Delta t^n + \Delta t^{n+1}}, \ \theta_1 = 1 - \gamma - c \frac{\Delta t^{n+1}}{\Delta t^n}, \theta_0 = c \left(\frac{\Delta t^{n+1}}{\Delta t^n} - \frac{\Delta t^{n+1}}{\Delta t^n + \Delta t^{n+1}} \right), \tag{16}$$

and the parameters c and γ define the chosen method. In (14),(15), and (16), we may select methods such as a Crank–Nicolson/Adams–Bashforth (CNAB), Modified CNAB (MCNAB), Gear, and Crank–Nicolson/ Leap-Frog (CNLF), which we obtain by specifying, respectively, the values (c, γ): $(0, 1/2)$, $(1/8, 1/2)$, $(0, 1)$, and $(1, 0)$. We have used Crank–Nicolson method to approximate the first time step. Note that, in (13), we need an iterative process to split the solution into two parts, one for the velocity and one for the pressure-increment. The number of iterations per time-step is fixed.

We use the same scheme to approximate (2) and (3) in time. In (2), we add a numerical artificial diffusion term in the lines of the classical approach of *artificial viscosity* for advection-dominated problems [38]. To complete the description, the only stability constraint left comes from the explicit discretization of the advection term. The time-step restriction due to the CFL condition, $\Delta t = C \Delta x / U_{max}$, with $C < 1$, Δx the width of a cell of the finest level, and U_{max} the maximum velocity component anywhere in the domain.

3.3. Pressure–velocity coupling and projection method

We decouple the pressure from the velocity in (4) by an extended version of the pressure-increment method [3,9,27,36]. Using (13), we compute a "preliminary" velocity $\mathbf{u}^\star$ field which is given by

$$\begin{aligned} \frac{\alpha_2 \mathbf{u}^{\star,k} + \alpha_1 \mathbf{u}^n + \alpha_0 \mathbf{u}^{n-1}}{\Delta t^{n+1}} = & \frac{\theta_2}{\rho^{n+1,k}} (\lambda \Delta \mathbf{u}^{\star,k} - \nabla \widetilde{p}^{n+1,k-1}) + \frac{\theta_1}{\rho^n} (\lambda \Delta \mathbf{u}^n - \nabla \widetilde{p}^n) \\ & + \frac{\theta_0}{\rho^{n-1}} (\lambda \Delta \mathbf{u}^{n-1} - \nabla \widetilde{p}^{n-1}) + \frac{\beta_1 \widetilde{\mathbf{f}}^n}{\rho^n} + \frac{\beta_0 \widetilde{\mathbf{f}}^{n-1}}{\rho^{n-1}}. \end{aligned} \tag{17}$$

This approximation $\mathbf{u}^{\star,k}$ does not satisfy the constraint $\nabla \cdot \mathbf{u} = S^{n+1}$ where S^{n+1} is given by (6) (note that $S^{n+1} \neq 0$ because the flow is not incompressible). To satisfy this constraint, we decompose the velocity field $\mathbf{u}^{\star,k}$ as

$$\mathbf{u}^{\star,k} = \mathbf{u}^{n+1,k} + \frac{\Delta t^{n+1} \theta_2 \lambda}{\alpha_2 \rho^{n+1,k}} \nabla q, \tag{18}$$

where q is the *pressure increment*. Therefore $\mathbf{u}^{n+1,k}$ satisfies the constraint (6). Applying a differential operator in (18) we obtain an elliptic equation for q,

$$\nabla \cdot \left(\frac{1}{\rho^{n+1,k}} \nabla q \right) = \frac{\alpha_2}{\lambda \theta_2 \Delta t^{n+1}} \left(\nabla \cdot \mathbf{u}^{\star,k} - S^{n+1} \right). \tag{19}$$

Solving (19), q is determined and we can update the pressure and velocity field by

$$\begin{aligned} \widetilde{p}^{n+1,k} &= \widetilde{p}^{n+1,k-1} + q, \\ \mathbf{u}^{n+1,k} &= \mathbf{u}^{\star,k} - \frac{\Delta t^{n+1} \theta_2 \lambda}{\alpha_2 \rho^{n+1,k}} \nabla q. \end{aligned} \tag{20}$$

In our numerical experiments, two iterations (that is, $k = 1, 2$), were enough to get a second-order accuracy in time, with $\widetilde{p}^{n+1,0} = \widetilde{p}^{n}$ being the initial approximation taken for the pressure.

3.4. Multilevel-multigrid method

Once we have introduced the discretization in space and in time, the numerical method requires the solution of twelve linear systems per time step (six for $k = 1$ and six for $k = 2$): ten of them come from parabolic equations (for the velocity components, mass density, and for the conserved scalar) and two of them come from elliptic equations (for the pressure-increment). To solve these linear systems, we use a multilevel-multigrid method [9]. The block-structured grid used suggests the use of a geometric multigrid strategy. Here multilevel refers to the fact that refinement levels belonging to the grid structure are also considered as multigrid levels. These refinement levels differ from usual multigrid levels because they do not necessarily cover the entire computational domain. Our multilevel-multigrid is based on V-cycles with red-black Gauss–Seidel relaxations, *fine-to-coarse transfers* given by simple average, and *coarse-to-fine transfers* by trilinear interpolation [9,38]. Cycling stops when the maximum norm of the residual vector is $O(\Delta x^2)$, which proved to be enough to maintain overall second-order accuracy.

We summarize the algorithm below. One time step corresponds to an IMEX step combined with a projection method that satisfy the velocity divergence constraint. Given ρ^n, z^n, $\widetilde{p}^n$, and $\mathbf{u}^n$ at time t^n, we obtain their updates in time by:

Algorithm 1: A time step solution procedure summarized

Compute Δt^{n+1};
Compute α, β and γ using (14), (15), and (16);
while $k \leq kmax$ **do**
  Update the density equation (2) to obtain $\rho^{n+1,k}$;
  Compute $\mu^{n+1,k}$ e $D^{n+1,k}$, see Section 2.
  Update the mixture fraction equation (3) to obtain $z^{n+1,k}$;
  Compute $T^{n+1,k}$, $Y_O^{n+1,k}$, and $Y_F^{n+1,k}$, using (10) and (11);
  Compute S^{n+1} using equation (6);
  Update the velocity $\mathbf{u}^{\star,k}$ using equation(17);
  Solve the elliptic equation (19) for q;
  Update $\mathbf{u}^{n+1,k}$ and $\widetilde{p}^{n+1,k}$ using (20).
end

4. Numerical examples

We present in this section numerical results to verify the convergence and the accuracy properties displayed by the described methodology for smooth problems ("manufactured solution" approach). Afterwards, we apply it to two flame-jet problems found in the literature to validate the modeling approach: in the first problem, we present full numerical results for a laminar methane diffusion flame and, in the second problem, we have investigated the performance of the numerical methodology in a turbulent, piloted methane/air diffusion flame.

4.1. Numerical verification: manufactured solution approach

In this test, we use a manufactured solution to investigate the convergence order of the adaptive numerical method [9,24]. The computational domain is $[a_1, b_1] \times [a_2, b_2] \times [a_3, b_3]$ with $a_i = 0$, $b_i = 1$, and $i = 1, 2, 3$. The manufactured solutions for the velocity components are given by $u_{1,e} = \sin^2[2\pi(x_1 + x_2 + x_3) + t]$, $u_{2,e} = \cos^2[2\pi(x_1+x_2+x_3)+t]$, and $u_{3,e} = \sin[2\pi(x_1+x_2+x_3)+t]$, for the pressure, $\widetilde{p}_e = \cos[2\pi(x_1+x_2+x_3)+t]$, and for the density $\rho_e = \rho_0+\sin^2[2\pi(x_1+x_2+x_3)+t]$, with $\rho_0 = 0.5$. In this test, the equation of state is $p_0 = \rho z$, therefore $z_e = p_0/\rho_e$, with $p_0 = 1$. In (2)–(4) and (6) are included source terms to keep the equations balanced. Other parameters are given by $\mu = \mu_0 + \mu_1 \cos^2[2\pi(x_1 + x_2 + x_3) + t]$, $\mu_0 = 1$, $\mu_1 = 0.5$, and $\rho D = \mu/0.75$.

The composite grid is given by a single block with N^3 computational cells in the base level, G_0, on the top of which we add one refinement level. For the spatial–temporal convergence rate test, the composite grid patches and

Table 1
Errors in L_2 norm and observed convergence order.

N	$\|u_{1,e} - u_1\|_2$	Order	$\|u_{2,e} - u_2\|_2$	Order	$\|u_{3,e} - u_3\|_2$	Order
16	$1.7730 \cdot 10^{-2}$	–	$1.8726 \cdot 10^{-2}$	–	$3.4106 \cdot 10^{-3}$	–
32	$4.4185 \cdot 10^{-3}$	2.00	$4.6506 \cdot 10^{-3}$	2.00	$9.1857 \cdot 10^{-4}$	1.89
64	$1.1145 \cdot 10^{-3}$	1.98	$1.1676 \cdot 10^{-3}$	1.99	$2.5448 \cdot 10^{-4}$	1.85
128	$2.8220 \cdot 10^{-4}$	1.98	$2.9866 \cdot 10^{-4}$	1.96	$7.0944 \cdot 10^{-5}$	1.84
N	$\|\widetilde{p}_e - \widetilde{p}\|_2$	Order	$\|\rho_e - \rho\|_2$	Order	$\|z_e - z\|_2$	Order
16	$7.3144 \cdot 10^{-1}$	–	$4.0612 \cdot 10^{-3}$	–	$3.4434 \cdot 10^{-3}$	–
32	$2.0906 \cdot 10^{-1}$	1.80	$9.8890 \cdot 10^{-4}$	2.03	$8.3116 \cdot 10^{-4}$	2.08
64	$5.8046 \cdot 10^{-2}$	1.84	$2.4180 \cdot 10^{-4}$	2.03	$2.0215 \cdot 10^{-4}$	2.01
128	$2.0189 \cdot 10^{-2}$	1.52	$5.9304 \cdot 10^{-5}$	2.02	$5.0875 \cdot 10^{-5}$	1.99

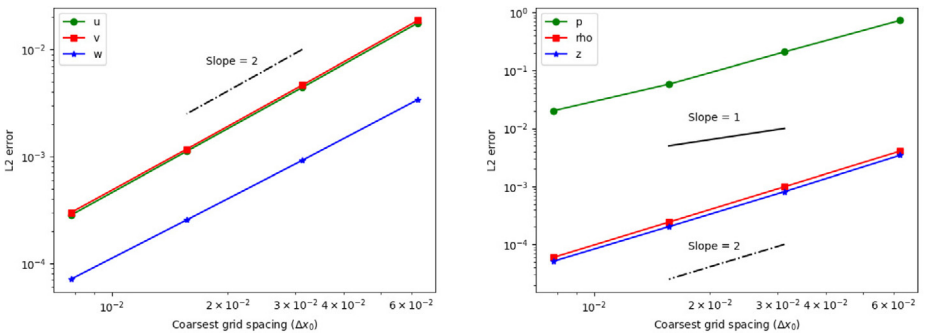

Fig. 4. Errors in L_2 norm at $t = 0.5$.

final time are kept fixed for all the runs. The number of grid cells in the base level are taken to be $N \times N \times N$, with $N = 16, 32, 64, 128$. Note that $\Delta x_0 = (b_i - a_i)/N$ is the coarsest grid spacing. The refinement level is composed by two patches: $G_{1,1}$ which covers the subdomain $[0, 0.2] \times [0, 0.15] \times [0, 0.5]$ and $G_{1,2}$ which covers $[0, 0.2] \times [0.15, 0.35] \times [0.4, 0.5]$. Note that the refined region is always the same, kept fixed in all the runs. The goal is to identify eventual loss of accuracy or the introduction of non-physical features in the numerical solution due to the interpolation schemes involving ghost/covered cells. We supply Dirichlet boundary conditions for the velocity $\mathbf{u} = (u_1, u_2, u_3)$, for the density ρ, and for the conserved scalar z on the plane x_1x_3 in $x_2 = 0$ (see the Cartesian axis in Fig. 5) and Neumann boundary conditions on the other boundaries. For the dynamic pressure, we adopt Neumann boundary conditions on the plane x_1x_3 in $x_2 = 0$ and Dirichlet conditions on the other boundaries.

Table 1 presents the observed convergence order and the errors in the L_2 norm for the velocity components, pressure, density, and for the conserved scalar at $t = 0.5$. The plots in Fig. 4 present these errors as functions of the coarsest grid spacing, Δx_0. By comparing their slopes with slopes equal to one and equal to two (which indicate first and second order of convergence, respectively), we conclude, as expected, that the methodology is second-order for the velocity components, density, and conserved scalar, and order better than one for the dynamic pressure [12]. Thus interpolation schemes are working properly.

4.2. Laminar diffusion flame

The first numerical simulation we perform is based on the experimental studies of a laminar diffusion flame presented by Mohammed et al. [23] and by Dworkin et al. [15]. There are plenty of articles which present a variety of combustion models and numerical methods to perform simulations of such a flame [13,23,25]. Moreover, recent studies used this experiment to help to understand the role of pressure and the fuel dilution on the flame structure [8,15].

In this numerical experiment, the computational flow domain is given by the Cartesian product $[0, b_1] \times [0, b_2] \times [0, b_3]$ with $b_1 = b_2 = b_3 = 0.1$ m. Initially, the fluid is at rest with null dynamic pressure (see (1)), the inflow

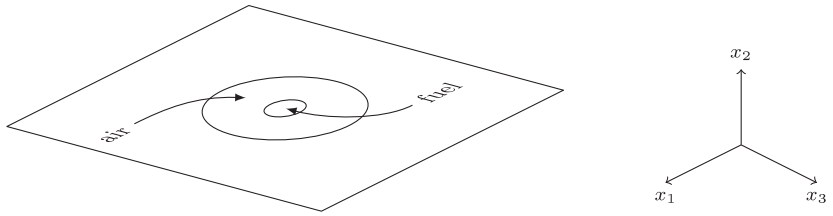

Fig. 5. Schematic diagram of the inlet computational domain and Cartesian axis.

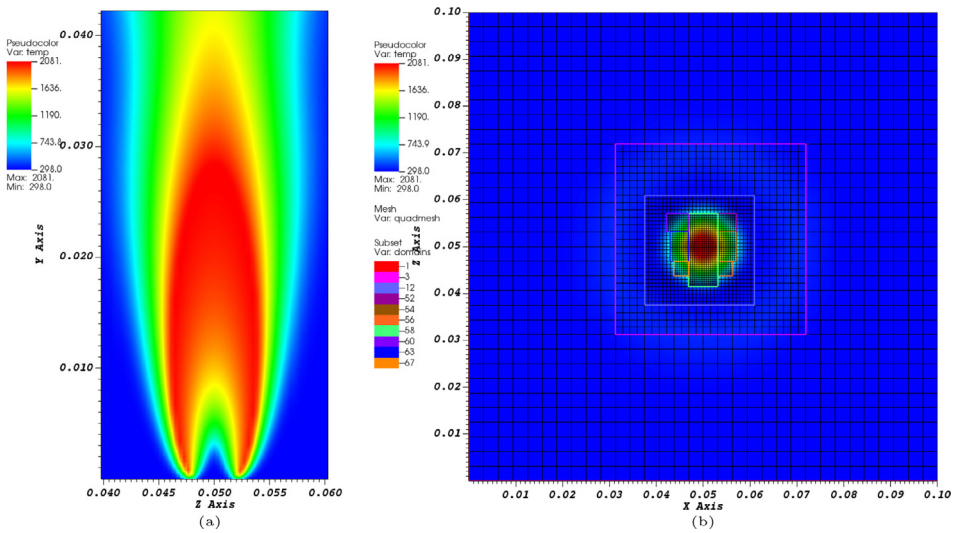

Fig. 6. Laminar diffusion flame temperature: (a) in the cross section plane $x_1 = 0.05$ m and (b) in the cross section plane $x_2 = 0.03$ m — note the adaptive mesh refinement patches in this cross section.

boundary ($x_2 = 0$) contains in its center a main jet with diameter $d_0 = 4$ mm, and the velocity components are given by $u_{1,in} = u_{3,in} = 0$ m/s, and

$$u_{2,in} = v_F \left[1 - \frac{(x_1 - x_{1c})^2 + (x_3 - x_{3c})^2}{r_0} \right], \tag{21}$$

where $v_F = 0.7$ m/s, $x_{ic} = b_i/2$, $r_0 = d_0/2$. The nozzle injects the fuel (methane diluted with nitrogen, $Y_{CH_4} = 0.5149$ and $Y_{N_2} = 0.4851$) at 298 K and the conserved scalar is given by $z = 1$. The oxidizer, air at 298 K ($Y_{O_2} = 0.232$ and $Y_{N_2} = 0.768$), is injected through the annular region between d_0 and $d_1 = 0.05$ m with a uniform velocity profile of 0.35 m/s and $z = 0$. At the inflow boundary, homogeneous Neumann boundary condition is adopted for the pressure increment. On the other computational boundaries, we have homogeneous Neumann boundary conditions for the velocity, for the conserved scalar, and for the density, and Dirichlet boundary conditions for the pressure increment. Fig. 5 shows a schematic diagram of the inlet computational domain and Cartesian axis.

The computation employs a three-dimensional block-structured grid having a base level with $32 \times 32 \times 32$ computational cells and three refinement levels (four levels in total). We refine regions of the flow where the temperature satisfies $\|T\| > 800$ K, constrained to 85% efficiency (i.e. the ratio between the number of cells in need of refinement over the total number of cells that end up being refined to get a grid patch). In the finest level, we have $\Delta x \approx 3.9 \cdot 10^{-2}$ cm and the time step size is $O(\Delta x)$. Fig. 6 presents the temperature distribution and composite grid in two cross sections.

Mohammed et al. [23] compute the solution in a two-dimensional non-uniform grid with 91×82 cells in radial and axial directions, respectively. Day and Bell [13] compute the solution in a two-dimensional AMR grid where the finest level has $\Delta x = 2 \cdot 10^{-2}$ cm ("base case") and $\Delta x = 5 \cdot 10^{-3}$ cm ("refined case"). We compare our results

Table 2
Comparison of flame characteristics.

	Maximum temperature	Flame length
Present work	2081 K (at 2.77 cm)	2.77 cm
Day and Bell [13]	2029 K (at 3.08 cm)	2.81 cm
Mohammed et al. [23]	2025 K (at 3.48 cm)	2.82 cm
Northup and Groth [25]	2080 K	3.30 cm

to Day and Bell [13] "base case" results since they use grid resolutions similar to ours. Both, Mohammed et al. [23] and Day and Bell [13] use complex chemical kinetics mechanism with 26 species and 83 reactions. Northrup and Groth [25] use an AMR grid and a two-step reaction mechanism.

Despite of the simpler chemical model and of the smaller spatial resolution, our numerical results keep up with literature [13,23,25]. The maximum centerline temperature is 2081 K compared to 2029 K [13], 2025 K [23], and 2080 K [25], which represent a relative difference of 2.56%, 2.57%, and 0.05%, respectively. The maximum centerline temperature is attained at 2.77 cm compared to 3.08 cm [13] and 3.48 cm [23], which represent a relative difference of 10.06% and 20.40%, respectively. This measure is not reported in [25]. Mohammed et al. [23] have defined the flame length as the difference between the location of the maximum temperature at the centerline and the lift-off height. Our data predict a flame length of 2.77 cm compared with 2.81 cm [13], 2.82 cm [23], and 3.3 cm [25], which represents a relative difference of 1.42%, 1.77%, and 16.06%, respectively. Table 2 summarizes the measured flame quantities.

It is important to notice that with less grid resolution and with a reduced chemical model, we have obtained results that agree with previous studies. Moreover, when we compare our results with the "refined case" [13], we should include two more refinement levels. In this case, the relative differences for the centerline location where the maximum temperature is reached, and for the flame length, are 12.62% and 5.78%, respectively.

4.3. Turbulent diffusion flame

Next, we present a numerical simulation of a turbulent diffusion flame based on experimental data [2] and on numerical results [19–21,29,41]. The fuel is a mixture made up of 25% of CH_4 and of 75% air (by volume). The fuel nozzle diameter is $d_0 = 7.2$ mm and is enclosed by an annular pilot nozzle with a diameter of $d_1 = 18.2$ mm. The pilot stream is a lean premixed gas mixture of C_2H_2, H_2, CO_2, N_2, corresponding to a mixture fraction of $z = 0.271$ [19]. The computational domain is $[a_1, b_1] \times [a_2, b_2] \times [a_3, b_3]$, with $a_1 = a_2 = a_3 = 0$, $b_1 = b_3 = 40d_0$, and $b_2 = 80d_0$. The velocity profile is $u_{1,in} = u_{3,in} = 0$ m/s and $u_{2,in} = u_{d_0} + u_{d_1}$ at the inflow boundary, with

$$
\begin{aligned}
u_{d_1} &= \frac{v_1 + v_0}{2} - \left(\frac{v_1 - v_0}{2}\right) \tanh\left[25\left(\frac{r_j}{r_1} - \frac{r_1}{r_j}\right)\right], \\
u_{d_0} &= \frac{v_2 + v_0}{2} - \left(\frac{v_2 - v_0}{2}\right) \tanh\left[25\left(\frac{r_j}{r_0} - \frac{r_0}{r_j}\right)\right],
\end{aligned}
\tag{22}
$$

where $r_j = \sqrt{(x_1 - x_{1c})^2 + (x_3 - x_{3c})^2}$, $v_0 = 0.9$ m/s is the coflow velocity, $v_1 = 11.4$ m/s the pilot gas velocity, and $v_2 = 49.6$ m/s the fuel exit velocity; r_1 and r_0 are, respectively, the pilot and fuel nozzle radii, and $x_{1c} = x_{3c} = 20d_0$. Moreover, we add a uniform noise to $u_{2,in}$ which is given by $0.01 \cdot \omega \cdot u_{2,in}$, where ω is a random number normally distributed in $[-0.5, 0.5]$. We adopt homogeneous Neumann boundary condition for the pressure increment. In the other computational boundaries, we impose homogeneous Neumann boundary conditions for the velocity and Dirichlet boundary conditions for the pressure increment. The Reynolds number is $Re = 2.24 \times 10^4$ [21,29]. Even for turbulent reacting flows, the sound speed is in the range of 300 m/s to 600 m/s, using the values of our example, $Ma = 0.16 < 0.3$ and the low-Mach approximation is still valid (see [27]). The mixture fraction profile inlet is prescribed as a step function with experimental value $z = 1$ at fuel nozzle, $z = 0.271$ in the pilot stream, and $z = 0$ in the coflow [20]. The stoichiometric fraction mixture is $z_{st} = 0.351$ [19,29].

We use the Large Eddy Simulation (LES) approach to model the turbulence and solve the Favre filtered transport Eqs. (2)–(3) with the subgrid scale stresses determined by the Smagorinsky model. In that context, the eddy viscosity is $\mu_t = C\Delta^2|\mathbf{S}|$ and the turbulent diffusivity $D_t = \mu_t/\sigma_t$, where C is the Smagorinsky constant, Δ the filter size,

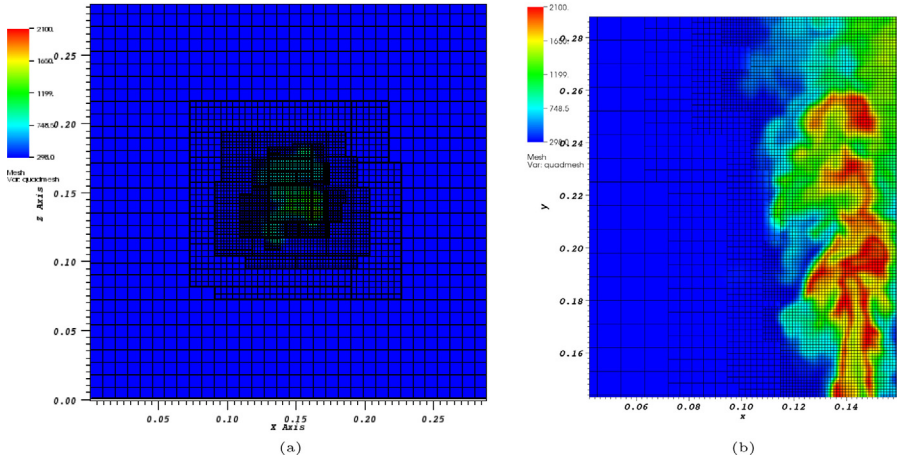

Fig. 7. Instantaneous temperature distribution and adaptive mesh refinement: (a) cross section in $x_2 = 40d_0$ and (b) cutting section in $x_1 = 20d_0$.

$|\mathbf{S}|$ the characteristic filtered rate of strain and σ_t is turbulent Schmidt-number [28,33]. The filtered species mass fraction and temperature are evaluated by a presumed β-PDF approach of the conserved scalar z [28,29].

The block-structured grid has $32 \times 64 \times 32$ cells in the base level G_0 and three refinement levels. The finest level has $\Delta x = 1.125 \times 10^{-3}$ m. This Δx size retains the time-step size in our simulation at $O(10^{-6})$. Ihme and Pitsch [19] have reported that $\Delta x_{min} = O(10^{-4})$ m and $\Delta x_{max} = O(10^{-3})$ m. The refinement criteria are based on the maximum norms of the turbulent viscosity, of the vorticity, and of the temperature, constrained to 85% of efficiency.

For turbulent flows, it is desirable to wait for the numerical simulations to reach a statistically stationary state. Kemenov and Pope [20] and Ihme and Pitsch [19] report that, for this turbulent diffusion flame, this should occur after running over 10 flow-through times. While aware of the importance of this fact, the results we report here have not statistically reached the steady state due to the limitation of the hardware at our disposal.[1] When the flame is fully developed, the processing time spent per time step is approximately 15 min in average. The bottleneck is the numerical solution of the elliptic equation which takes about 25 V-cycles to be solved, representing almost 80% of the time spent in one time step. The other linear systems take 5 to 7 cycles to be solved. At these later simulation times, composite grids often have about 3 million cells (plus ghost and covered cells), resulting in about 3 million unknown variables in each of the six linear systems that must be solved twice. For the given conditions, a typical run may take easily about three months to run for this case. However, despite the simplified chemical reaction model and the relatively low spatial resolution that were adopted, our numerical results capture surprisingly well qualitatively important characteristics of the fluid dynamics such as the temperature and mixture fraction distributions, thus exhibiting, even in this more stringent example, the prediction capabilities of the adaptive mesh refinement in conjunction with the fast chemistry approach proposed.

Fig. 7 shows instantaneous temperature distribution and the adaptive mesh refinement in two cutting sections. Fig. 7(a) presents a cross section in flow direction (top view) with an adaptive mesh. Fig. 7(b) shows an enlarged view of the temperature with some turbulent structures. Fig. 8 shows a snapshot of, respectively, instantaneous conserved scalar (a) and (c), and temperature distribution (b) and (d), in two instants of time. The comparison between two time steps shows the dynamics of turbulent structures.

We point out that the maximum temperature occurs where the conserved scalar assumes the stoichiometric value. Fuel and oxidizer are represented by the value one and zero, respectively, of the conserved scalar. Both have the same value 298 K. The intermediate values of conserved scalar indicate the region where the mixing and reaction occurs.

Our instantaneous temperature distribution results shown in Figs. 8(b) and 8(d) capture the turbulent flame dynamics and are in agreement with results presented by Ihme and Pitsch ([19], Fig. 2(a)), by Yang et al. ([41],

[1] Serial code running on core 2 Quad 2.4 GHz processors, 16 Gb of RAM.

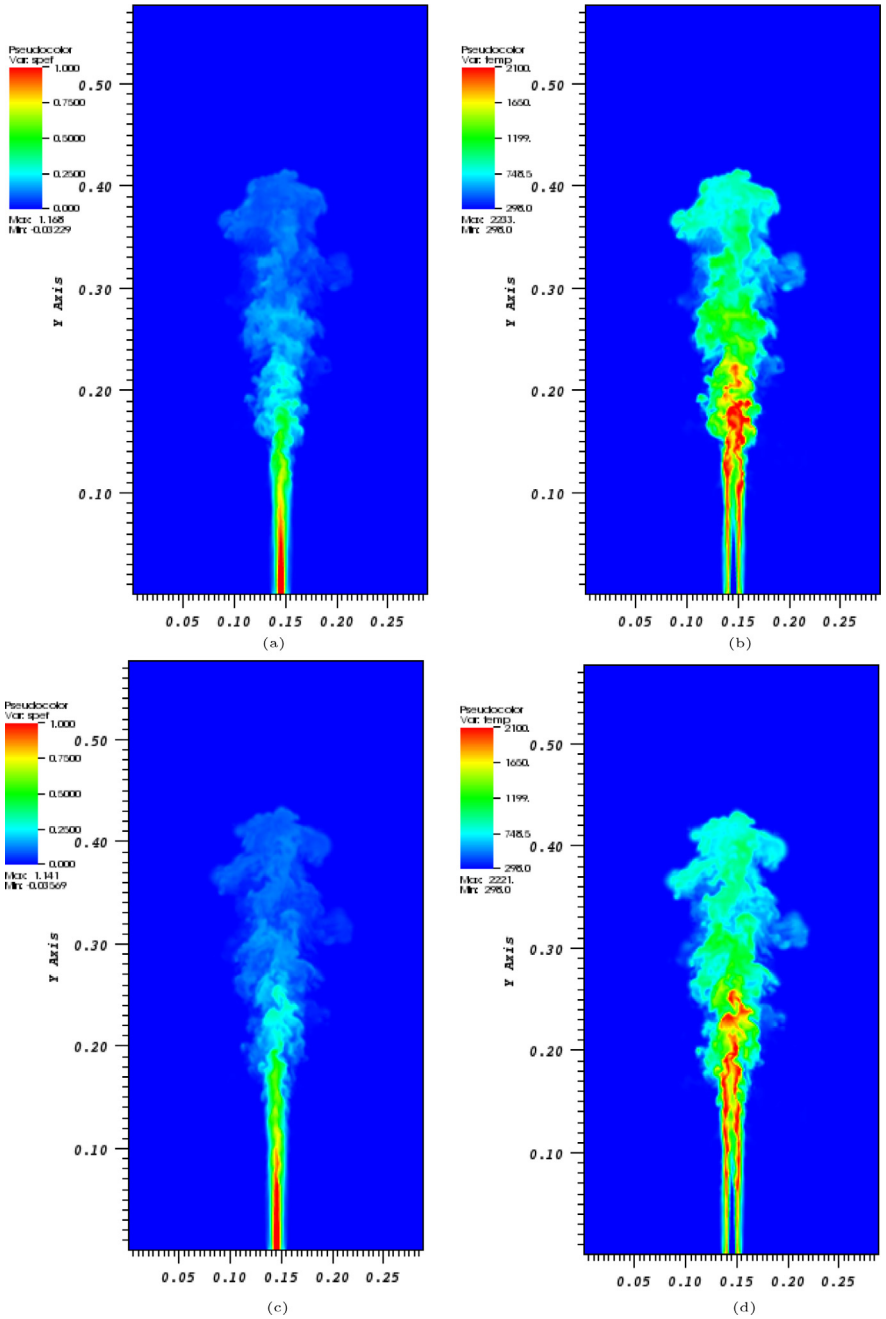

Fig. 8. Turbulent diffusion flame in two time steps: instantaneous (a)–(c) conserved scalar, and (b)–(d) temperature distribution in $x_1 = 20d_0$.

Fig. 4), and by Pitsch and Steiner ([29], Figs. 5–6) which, and we highlight that fact, were produced with more complex mechanism and parallel codes. These results indicate that our strategy may very well be suited to perform simulations of certain turbulent flame dynamics, although this has to be investigated further.

5. Concluding remarks

In this paper, we present an adaptive numerical methodology for simulating low-Mach number reacting flows. The overall approach combines an adaptive mesh refinement technique with an IMEX time integration scheme, a

projection method, and a fast chemistry approach. This combination extends the capabilities of our methodologies introduced in previous works [9–11,14,26,30,36]. The numerical verification performed using the manufactured solution strategy indicates that the numerical method has second-order convergence accuracy for problems with smooth solutions. We illustrate the performance of the numerical method through simulations of diffusion flame jets. For a laminar diffusion flame, the strategy used to model the fast chemistry presents good quantitative results at a lower cost when compared to other more expensive approaches found in the literature. Performance tests reveal that the adaptive feature of the current approach makes it about 150 times faster, initially, than the same approach on uniform meshes for the reported problems, and about 5 times faster towards the end of the simulation. In the more stringent example of a turbulent diffusion flame, our approach is capable of capturing some important fluid dynamic characteristics such as temperature and conserved scalar distributions in time and space, performing surprisingly well on regular hardware with a serial code.

Acknowledgments

The authors thank João M. Vedovoto, Millena M. Villar, Rudimar L. Nós, Catalina M. R. Alvarez, and Álvaro J. P. Franco for their helpful insights, and also the reviewers for their valuable comments and suggestions which helped to improve this published version. Financial support was provided in part by CAPES, and by CNPq (142071/2007-2, 307348/2008-3, 309433/2011-8), and PQ-2 (303514.2014-0), Brazil. Serial simulations were performed using Core 2 Quad 2.4G Hz processors, with 16 Gb RAM, at the Applied Mathematics Laboratory (LabMAP) at IME-USP.

References

[1] U.M. Ascher, S.J. Ruuth, B.T.R. Wetton, Implicit-explicit methods for time-dependent partial differential equations, SIAM J. Numer. Anal. 32 (3) (1995) 797–823, http://dx.doi.org/10.1137/0732037.
[2] R. Barlow, J. Frank, Effects of turbulence on species mass fractions in methane/air jet flames, in: Symposium (International) on Combustion, Vol. 27, Elsevier, 1998, pp. 1087–1095, http://dx.doi.org/10.1016/S0082-0784(98)80510-9.
[3] J. Bell, AMR for low Mach number reacting flow, in: Adaptive Mesh Refinement-Theory and Applications, Springer, 2003, pp. 203–221, http://dx.doi.org/10.1007/3-540-27039-6_14.
[4] M.J. Berger, P. Colella, et al., Local adaptive mesh refinement for shock hydrodynamics, J. Comput. Phys. 82 (1) (1989) 64–84, http://dx.doi.org/10.1016/0021-9991(89)90035-1.
[5] M.J. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, J. Comput. Phys. 53 (3) (1984) 484–512, http://dx.doi.org/10.1016/0021-9991(84)90073-1.
[6] M. Berger, I. Rigoutsos, An algorithm for point clustering and grid generation, IEEE Trans. Syst. Man Cybern. 21 (5) (1991) 1278–1286, http://dx.doi.org/10.1109/21.120081.
[7] R.W. Bilger, A mixture fraction framework for the theory and modeling of droplets and sprays, Combust. Flame 158 (2) (2011) 191–202, http://dx.doi.org/10.1016/j.combustflame.2010.08.008.
[8] S. Cao, B. Davi, D. Giassi, B.A.V. Bennett, M.B. Long, M.D. Smooke, Effects of pressure and fuel dilution on coflow laminar methane-air diffusion flames: A computational and experimental study, Combust. Theory Model. 22 (2) (2018) 316–337, http://dx.doi.org/10.1080/13647830.2017.1403051.
[9] H.D. Ceniceros, R.L. Nós, A.M. Roma, Three-dimensional, fully adaptive simulations of phase-field fluid models, J. Comput. Phys. 229 (17) (2010) 6135–6155, http://dx.doi.org/10.1016/j.jcp.2010.04.045.
[10] H.D. Ceniceros, A.M. Roma, A nonstiff, adaptive mesh refinement-based method for the Cahn–Hilliard equation, J. Comput. Phys. 225 (2) (2007) 1849–1862, http://dx.doi.org/10.1016/j.jcp.2007.02.019.
[11] H.D. Ceniceros, A.M. Roma, A. Silveira-Neto, M.M. Villar, et al., A robust, fully adaptive hybrid level-set/front-tracking method for two-phase flows with an accurate surface tension computation, Commun. Comput. Phys. 8 (1) (2010) 51–94, http://dx.doi.org/10.4208/cicp.050509.141009a.
[12] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comp. 22 (104) (1968) 745–762, http://dx.doi.org/10.2307/2004575.
[13] M.S. Day, J.B. Bell, Numerical simulation of laminar reacting flows with complex chemistry, Combust. Theory Model. 4 (4) (2000) 535–556, http://dx.doi.org/10.1088/1364-7830/4/4/309.
[14] W.C. de Jesus, A.M. Roma, M.R. Pivello, M.M. Villar, A. da Silveira-Neto, A 3D front-tracking approach for simulation of a two-phase fluid with insoluble surfactant, J. Comput. Phys. 281 (2015) 403–420, http://dx.doi.org/10.1016/j.jcp.2014.10.021.
[15] S. Dworkin, B. Conelly, A. Schaffer, B. Bennett, M. Long, M. Smooke, M. Puccio, B. McAndrews, J. Miller, Computational and experimental study of a forced, time-dependent, methane-air coflow diffusion flame, Proc. Combust. Inst. 31 (2007) 971–978, http://dx.doi.org/10.1016/j.proci.2006.08.109.
[16] J.H. Ferziger, M. Perić, R.L. Street, Computational Methods for Fluid Dynamics, Vol. 3, Springer, 2002.
[17] K.K. Foo, Z. Sun, P.R. Medwell, Z.T. Alwahabi, G.J. Nathan, B.B. Dally, Influence of nozzle diameter on soot evolution in acoustically forced laminar non-premixed flames, Combust. Flame 194 (2018) 376–386, http://dx.doi.org/10.1016/j.combustflame.2018.05.026.
[18] J. Hilditch, P. Colella, A projection method for low Mach number fast chemistry reacting flow, in: 35th Aerospace Sciences Meeting and Exhibit, 1997, p. 263, http://dx.doi.org/10.2514/6.1997-263.

[19] M. Ihme, H. Pitsch, Prediction of extinction and reignition in nonpremixed turbulent flames using a flamelet/progress variable model 2. Application in LES of Sandia flames D and E, Combust. Flame 155 (2008) 90–107, http://dx.doi.org/10.1016/j.combustflame.2008.04.015.
[20] K.A. Kemenov, S.B. Pope, Molecular diffusion effects in LES of a piloted methane-air flame, Combust. Flame 158 (2011) 240–254, http://dx.doi.org/10.1016/j.combustflame.2010.08.014.
[21] A. Kempf, F. Flemming, J. Janicka, Investigation of lenghtscales, scalar dissipation, and flame orientation in a piloted diffusion flame by LES, Proc. Combust. Inst. 30 (2005) 557–565, http://dx.doi.org/10.1016/j.proci.2004.08.182.
[22] A. Majda, J. Sethian, The derivation and numerical solution of the equations for zero Mach number combustion, Combust. Sci. Technol. 42 (3–4) (1985) 185–205, http://dx.doi.org/10.1080/00102208508960376.
[23] R.K. Mohammed, M.A. Tanoff, M.D. Smooke, A.M. Schaffer, M.B. Long, Computational and experimental study of a forced, timevarying, axisymmetric, laminar diffusion flame, in: Symposium (International) on Combustion, Vol. 27, Elsevier, 1998, pp. 693–702, http://dx.doi.org/10.1016/S0082-0784(98)80462-1.
[24] N. Nangia, B.E. Griffith, N.A. Patankar, A.P.S. Bhalla, A robust incompressible Navier–Stokes solver for high density ratio multiphase flows, J. Comput. Phys. 390 (2019) 548–594, http://dx.doi.org/10.1016/j.jcp.2019.03.042.
[25] S. Northrup, C. Groth, Solution of laminar combusting flows using a parallel implicit adaptive mesh refinement algorithm, in: Computational Fluid Dynamics 2006, Springer, 2009, pp. 341–346, http://dx.doi.org/10.1007/978-3-540-92779-2_52.
[26] R.L. Nós, A.M. Roma, C.J. García-Cervera, H.D. Ceniceros, Three-dimensional coarsening dynamics of a conserved, nematic liquid crystal-isotropic fluid mixture, J. Non-Newton. Fluid Mech. 248 (2017) 62–73, http://dx.doi.org/10.1016/j.jnnfm.2017.08.009.
[27] R.B. Pember, L.H. Howell, J.B. Bell, P. Colella, W.Y. Crutchfield, W.A. Fiveland, J.P. Jessee, An adaptive projection method for unsteady, low-Mach number combustion, Combust. Sci. Technol. 140 (1–6) (1998) 123–168, http://dx.doi.org/10.1080/00102209808915770.
[28] C.D. Pierce, P. Moin, Progress-variable approach for large-eddy simulation of non-premixed turbulent combustion, J. Fluid Mech. 504 (2004) 73–97, http://dx.doi.org/10.1017/S0022112004008213.
[29] H. Pitsch, H. Steiner, Large-eddy simulation of a turbulent piloted methane/air diffusion flame (Sandia flame D), Phys. Fluids 12 (10) (2000) 2541–2554, http://dx.doi.org/10.1063/1.1288493.
[30] M.R. Pivello, M. Villar, R. Serfaty, A. Roma, A. Silveira-Neto, A fully adaptive front tracking method for the simulation of two phase flows, Int. J. Multiph. Flow. 58 (2014) 72–82, http://dx.doi.org/10.1016/j.ijmultiphaseflow.2013.08.009.
[31] T. Plewa, T. Linde, V.G. Weirs, et al., Adaptive Mesh Refinement-Theory and Applications, Springer, 2003, http://dx.doi.org/10.1007/b138538.
[32] T. Poinsot, D. Veynante, Theoretical and Numerical Combustion, RT Edwards, Inc., 2005.
[33] S.B. Pope, Turbulent Flows, Cambridge University Press, 2000.
[34] R. Rehm, H. Baum, The equations of motion for thermally driven, buoyant flows, J. Res. Natl. Bur. Stand. 83 (3) (1978) 297–308.
[35] P.-H. Renard, D. Thevenin, J.-C. Rolon, S. Candel, Dynamics of flame/vortex interactions, Prog. Energy Combust. Sci. 26 (3) (2000) 225–282, http://dx.doi.org/10.1016/S0360-1285(00)00002-2.
[36] A.M. Roma, C.S. Peskin, M.J. Berger, An adaptive version of the immersed boundary method, J. Comput. Phys. 153 (2) (1999) 509–534, http://dx.doi.org/10.1006/jcph.1999.6293.
[37] M.N. Soloklou, A.A. Golneshan, Effect of CO_2 diluent on the formation of pollutant NOx in the laminar non-premixed methane-air flame, Int. J. Heat Mass Transfer 148 (2020) 119071, http://dx.doi.org/10.1016/j.ijheatmasstransfer.2019.119071.
[38] U. Trottenberg, C.W. Oosterlee, A. Schuller, Multigrid, Elsevier, 2000.
[39] H.K. Versteeg, W. Malalasekera, An Introduction to Computational Fluid Dynamics: The Finite Volume Method, Pearson education, 2007.
[40] D. Wang, S.J. Ruuth, Variable step-size implicit-explicit linear multistep methods for time-dependent partial differential equations, J. Comput. Math 26 (6) (2008) 835–855.
[41] S. Yang, X.W.H. Huo, W. Sun, V. Yang, An efficient finite-rate chemistry model for a preconditioned compressible flow solver and its comparison with the flamelet/progress-variable model, Combust. Flame 210 (2019) 172–182, http://dx.doi.org/10.1016/j.combustflame.2019.08.035.