Journal of Computational Physics 274 (2014) 413–431

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A velocity divergence constraint for large-eddy simulation of low-Mach flows

Randall J. McDermott

Fire Research Division, National Institute of Standards and Technology, Gaithersburg, MD 20899, USA

ARTICLE INFO

Article history:
Received 29 August 2013
Received in revised form 28 May 2014
Accepted 11 June 2014
Available online 17 June 2014

Keywords:
Large-eddy simulation
LES
Low-Mach flow
Projection scheme
Divergence constraint

ABSTRACT

The velocity divergence (rate of fluid volumetric expansion) is a flow field quantity of fundamental importance in low-Mach flows. It directly affects the local mass density and therefore the local temperature through the equation of state. In this paper, starting from the conservative form of the sensible enthalpy transport equation, we derive a discrete divergence constraint for use in large-eddy simulation (LES) of low-Mach flows. The result accounts for numerical transport of mass and energy, which is difficult to eliminate in relatively coarse, engineering LES calculations when total variation diminishing (TVD) scalar transport schemes are employed. Without the correction terms derived here, unresolved (numerical) mixing of gas species with different heat capacities or molecular weights may lead to erroneous mixture temperatures and ultimately to an imbalance in the energy budget. The new formulation is implemented in a publicly available LES code called the Fire Dynamics Simulator (FDS). Accuracy of the flow solver for transport is demonstrated using the method of manufactured solutions. The conservation properties of the present scheme are demonstrated on two simple energy budget test cases, one involving a small fire in a compartment with natural ventilation and another involving mixing of two gases with different thermal properties.

Published by Elsevier Inc.

1. Introduction

In this paper, starting from the conservative form of the sensible enthalpy transport equation, we derive a numerically consistent velocity divergence constraint for use in large-eddy simulation (LES) of low-Mach flows. The result accounts for numerical transport of mass and energy, which is difficult to eliminate in relatively coarse, engineering LES calculations when total variation diminishing (TVD) scalar transport schemes are employed. Without the correction terms derived here, unresolved (numerical) mixing of gas species with different heat capacities or molecular weights may lead to erroneous mixture temperatures and ultimately to an imbalance in the energy budget.

Equations describing low-speed, variable density flows were derived by Rehm and Baum in 1978 [1] to model thermally-driven buoyant plumes. This system of equations has since been extended to include viscous flows [2,3] and is the basis for many prominent turbulent reacting flow solvers, e.g., [4–8]. Applications suitable to the low-Mach approximation include, but are not limited to, modeling atmospheric pollutant dispersion, gas turbine engines, thermo-nuclear flames, and fire. The author is among the principal developers of a low-Mach LES code called the Fire Dynamics Simulator (FDS) [9], used

E-mail address: randall.mcdermott@nist.gov.

http://dx.doi.org/10.1016/j.jcp.2014.06.019
0021-9991/Published by Elsevier Inc.

primarily for building design and safety analysis by fire protection engineers, but also used in forensic reconstruction of fire events [10].

For chemically reacting flows involving heat release, achieving consistent coupling between the species mass fractions, the mass density, the energy, and the equation of state within a time-marching algorithm is nontrivial. Often, iterative methods are employed to find consistent solutions [4,5,7]. The "EKT" ("echt konservativer transport", "fully conservative transport" in German) algorithm of Kempf [6] is an exception. The present work, which differs from EKT in several important respects, introduces a consistent *explicit* scheme which satisfies the conservative form of the mass, species, and energy transport equations.

The FDS formulation is most closely related to the low-Mach formulation of Bell [4] who uses an approximate projection [11] on a collocated grid to satisfy a thermodynamic velocity divergence constraint. This constraint is derived by factoring the divergence out of the continuity equation, substituting the ideal gas equation of state (EOS) for density, and then differentiating the EOS. Bell then solves a conservative enthalpy equation and iterates until the temperature obtained from the enthalpy and the EOS are consistent. In contrast, FDS employs an exact projection on a staggered grid [12], explicitly solves the continuity equation for mass density, factors the divergence from the enthalpy equation, and obtains temperature from the EOS. At this stage in the FDS algorithm, as the analysis in this paper shows, the temperature may not be consistent with a conservative solution of the enthalpy equation unless careful consideration is given to the effects of numerical mixing.

The mixing corrections derived here for the velocity divergence—which prevent First Law (energy conservation) violations in the FDS algorithm—are reminiscent of flux corrections needed to satisfy the *entropy condition* for compressible finite-volume methods [13–17]. The issue of flux accuracy, however, is separate from the issue addressed in this paper. In any practical combustion simulation we are likely to encounter scalar discontinuities, such as the interface between fuel and oxygen in a diffusion flame (we cannot resolve a flame front in LES), where the flux accuracy must degrade to first order by Godunov's theorem. For a numerical scheme to conserve both mass and energy in such cases, with a consistent velocity field, the divergence corrections derived here must be incorporated, either explicitly into the divergence constraint (which we argue is the most efficient approach) or implicitly through iteration of the mass, momentum, and energy solutions. *The key result of this paper is recognition that the discretization of the state variables (enthalpy and composition) may be inconsistent with the assumed variation of the thermodynamic pressure (the key assumption in the low-Mach formulation) unless the proper corrections are employed (either explicitly or through iteration).* Note that compressible formulations do not suffer this problem because the thermodynamic pressure may vary locally in tight concert with the equation of state.

Furthermore, in low-Mach LES formulations it is common to use kinetic-energy-preserving spatial discretizations for the momentum equation [5,18–21]. In short, central differences are employed for momentum while TVD schemes are typically employed for scalars. This allows kinetic energy to be preserved in the absence of physical viscosity (assuming the time discretization is suitable) and subgrid models for the turbulent stress become wholly responsible for the leading-order physics of transferring kinetic energy from the resolved scales to the unresolved scales of motion (that is, dissipating kinetic energy from the grid). But, regardless of the conservation properties of the spatial discretizations, the time marching scheme—in which the divergence constraint plays a key role for low-Mach flows—is ultimately responsible for maintaining consistency between mass, momentum, (thermal) energy, and the equation of state.

The remainder of this paper is organized as follows. Next, in Section 2, we outline the mathematical formulation drawing attention to the role played by the velocity divergence. In Section 3, we derive the divergence constraint, which is the main focus of this work. Test cases illustrating accuracy of the flow solver, the potential energy imbalance, and the effect of numerical mixing are presented in Section 4. This is followed by a discussion in Section 5 of the correction terms and a viable simplification to the divergence constraint which avoids the need for the corrections. Finally, conclusions are stated in Section 6.

2. Governing equations

Filtering The equations for large-eddy simulation (LES) are derived by applying a low-pass filter, parameterized by a width Δ, to the transport equations for mass, momentum, and energy. For our purposes, it is sufficient to think of the filtered fields in the LES equations as cell means. For example, in one dimension the filtered density, $\bar{\rho}(x,t)$, is

$$\bar{\rho}(x,t) = \frac{1}{\Delta} \int_{x-\Delta/2}^{x+\Delta/2} \rho(r,t)\,\mathrm{d}r. \tag{1}$$

In FDS, the filter width, Δ, is equivalent to the local cell size, Δx, and is a key parameter in the submodels for the turbulent viscosity and the reaction time scale, discussed later. In what follows, the filter formalism is relaxed (the overline notation is suppressed for readability) since no explicit filtering operations are performed in the algorithm.

Low-Mach formulation In the low-Mach formulation, the pressure is decomposed into a background thermodynamic pressure (used in the equation of state) and a fluctuating hydrodynamic pressure (which drives local fluctuations in the flow field),

$$p(\mathbf{x},t) = p_0(t) + \int_0^z \rho_0(z) g_z \, \mathrm{d}z + \tilde{p}(\mathbf{x},t),$$
$$= \underbrace{\bar{p}(z,t)}_{background} + \underbrace{\tilde{p}(\mathbf{x},t)}_{perturbation} . \tag{2}$$

The background pressure is allowed to vary in the vertical coordinate, z, to account for atmospheric stratification; $\rho_0(z)$ is the background density. The background pressure may also vary in time to account for pressurization of the domain, such as a sealed compartment fire scenario.

The equations governing the evolution of a low-Mach, variable density flow are, respectively, continuity, species mass concentration, momentum, energy (sensible enthalpy), and the ideal gas equation of state:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{3}$$
$$\frac{\partial \rho Y_\alpha}{\partial t} + \nabla \cdot (\rho Y_\alpha \mathbf{u}) = -\nabla \cdot \mathbf{J}_\alpha + \dot{m}_\alpha''' \tag{4}$$
$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) = -\nabla \tilde{p} + \nabla \cdot \mathsf{T} + (\rho - \rho_0)\mathbf{g} \tag{5}$$
$$\frac{\partial \rho h_s}{\partial t} + \nabla \cdot (\rho h_s \mathbf{u}) = \frac{\mathrm{D}\bar{p}}{\mathrm{D}t} + \dot{q}''' - \nabla \cdot \dot{\mathbf{q}}'' \tag{6}$$
$$\rho = \frac{\bar{p}\overline{W}}{RT} \tag{7}$$

Here, ρ is mass density, $\mathbf{u}$ is the velocity, Y_α is the mass fraction of species α, $\mathbf{J}_\alpha$ is the diffusive mass flux of α ($\sum_\alpha \mathbf{J}_\alpha = 0$), and $\dot{m}_\alpha'''$ is the reaction source term. In the momentum equation, T is the deviatoric stress tensor and $\mathbf{g}$ is the gravitational acceleration. The sensible enthalpy per unit mass is $h_s = \sum_\alpha Y_\alpha h_{s,\alpha}$, where $h_{s,\alpha} = \int_{T_0}^{T} c_{p,\alpha}(T')\,\mathrm{d}T'$. The heat release per unit volume due to reaction is $\dot{q}'''$, and the heat flux vector (diffusion, conduction, and radiation) is denoted $\dot{\mathbf{q}}''$. The ideal gas equation of state is based on the thermodynamic pressure, $\bar{p}$, the mixture molecular weight, $\overline{W}$, the temperature, T, and the ideal gas constant, R.

In the FDS algorithm, $n_s - 1$ species transport equations are solved, where n_s is the number of chemical species in the system. The diffusive flux, $\mathbf{J}_\alpha$, is modeled with Fick's law using mixture-averaged diffusion coefficients, $\mathbf{J}_\alpha = -\rho D_\alpha \nabla Y_\alpha$ for $\alpha \neq n_s$. The value of the mass fraction for species n_s is obtained from $Y_{n_s} = 1 - \sum_{\alpha=1}^{n_s-1} Y_\alpha$. Thus, the value Y_{n_s} absorbs all errors, both physical and numerical, from the transport model for the other species. (Note that errors are not strictly additive; some may cancel out.) The error in Y_{n_s} due to molecular diffusion is typically considered negligible in LES calculations because the turbulent diffusivity (discussed below)—which supplements molecular diffusion—is the same for all species and is usually one or two orders of magnitude larger than the molecular diffusion coefficient.

To facilitate the numerical solution, the momentum equation is rearranged into the following form:

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{F} + \nabla H), \tag{8}$$
$$\mathbf{F} = \underbrace{\mathbf{u} \cdot \nabla \mathbf{u} - \nabla K}_{-\mathbf{u}\times(\nabla\times\mathbf{u})} - \tilde{p}\nabla(1/\rho) - \frac{1}{\rho}\left[\nabla \cdot \mathsf{T} + (\rho - \rho_0)\mathbf{g}\right], \tag{9}$$
$$H \equiv \tilde{p}/\rho + K, \tag{10}$$

where the resolved kinetic energy per unit mass is $K \equiv \frac{1}{2}|\mathbf{u}|^2$. The Bernoulli integral, H, obeys the Poisson equation

$$\nabla^2 H = -\left[\nabla \cdot \mathbf{F} + \frac{\partial}{\partial t}(\nabla \cdot \mathbf{u})\right]. \tag{11}$$

Eq. (11) finally highlights the main focus of this paper: the velocity divergence constraint, $\nabla \cdot \mathbf{u}$. The time derivative of the divergence shows up as a source term in the Poisson equation and links the momentum equation to the mass and energy equations and the equation of state. Eqs. (8) and (11) are solved with a fractional step method (detailed in Appendix A) which guarantees the velocity field satisfies the divergence constraint. Correct specification of this constraint, which we discuss below in Section 3, is the key to consistency between the mass, energy, and state equations.

Subgrid closures The species, momentum, and sensible enthalpy equations (4)–(6) require modification in LES to account for unresolved turbulent transport. In FDS, the deviatoric stress is closed via gradient diffusion with the isotropic eddy viscosity model of Deardorff [22,23]. The turbulent diffusion of mass and heat are modeled similarly using constant turbulent Schmidt and Prandtl numbers. Details are provided in [9].

Chemical heat release In a typical fire simulation, the heat release per unit volume, $\dot{q}'''$ in (6), is the largest contributor to the velocity divergence. For the test cases considered in this paper, it is sufficient to consider the simplified reaction *fuel* + *air* → *products*. Typically, the kinetics of the heat releasing reactions in a fire are fast compared to the rate of mixing and the appropriate combustion model is therefore the "mixed is burned" approximation. The rate of combustion is obtained from the eddy dissipation concept (EDC) model of Magnussen and Hjertager [24], where the rate of fuel consumption is given by

$$\dot{m}_f''' = -\rho \frac{\min(Y_f, Y_o/s)}{\tau_{mix}}. \tag{12}$$

Here, Y_f is the resolved fuel mass fraction, Y_o is the resolved mass fraction of oxidizer, and s is the mass stoichiometric coefficient. The mixing time τ_{mix} is taken from an algebraic closure based on the eddy viscosity and a constant turbulent Schmidt number. Details are provided in [25]. The heat release per unit volume is obtained from

$$\dot{q}''' = \dot{m}_f''' \Delta h_c, \tag{13}$$

where Δh_c is the heat of combustion of the fuel.

Numerical methods FDS is generally a second-order, explicit code with first-order time splitting for chemistry. Details of the time marching procedure are given in Appendix A. Block structured Cartesian grids are employed and data storage for primitive variables is staggered [26]: scalars live at cell centers and velocity components live at their respective face centers. The Poisson equation (11) is solved using a fast, direct solver [27]. Flow obstructions are treated using a simple direct-forcing immersed boundary method [29].

FDS is an implicitly filtered LES code. That is, the filter width is taken to be equivalent to the mesh spacing, $\Delta = \Delta x$. Note, however, that implicit filtering does *not* necessarily imply the use of dissipative numerical schemes for the momentum equation (as in [30]). FDS employs kinetic-energy-preserving, second-order central differencing schemes for spatial discretization of the momentum equation [19]. The leading order physics of transferring resolved kinetic energy to the subgrid scales relies on physics-based models for the subgrid stress.

To ensure boundedness and realizability of the scalar fields, FDS employs second-order total variation diminishing (TVD) transport schemes [17]. The default flux limiter is the Superbee scheme of Roe [31]; CHARM [32] is also available for highly-resolved calculations. The flux limiter scheme is used for spatial discretization of the density and mass fraction equations, as well as for the correction terms in the divergence expression, which are the main subject of this paper. Implementation details for the flux limiters are provided in Section 3.2.

Summary of solution procedure The basic algorithm is summarized below. Note that here we are showing only one stage of the RK2 scheme (see Appendix A). This stage basically corresponds to a simple Forward Euler step combined with a projection step to enforce the velocity divergence constraint. Given u_i^n, ρ^n, Y_α^n, and T^n at time t^n:

1. Update the density equation (3) to obtain ρ^{n+1}.
2. Update the $n_s - 1$ species equations (4) to obtain Y_α^{n+1}.
3. Update the temperature from the equation of state (7) to obtain T^{n+1}.
4. Compute $(\nabla \cdot \mathbf{u})^{n+1}$, discussed below.
5. Solve the Poisson equation (11) for H^n.
6. Update momentum (8) using the pressure field from Step 5 to obtain u_i^{n+1}.

Notice that the energy equation is not explicitly updated. Instead, the velocity divergence is specified, Step 4, such that (6) is satisfied. In the following sections we will discuss the divergence expression in detail and highlight specific terms which should be retained in order to properly account for numerical mixing of gases with arbitrary specific heats.

It is important to appreciate that Steps 5 and 6 combine to ensure that the discrete divergence of the velocity field computed in Step 6 exactly matches the divergence computed in Step 4. This effectively couples mass, momentum, energy, and the equation of state—all the fields are consistent when starting the next time step. The specification of the divergence in Step 4 forces (6) to be satisfied and furthermore forces the time derivative $\partial(\rho h_s)/\partial t$ to be consistent with the equation of state and the mass updates from Steps 1 and 2.

3. The divergence constraint

The present work stems from attempts to understand and correct an energy budget imbalance which became evident after implementing both temperature-dependent specific heats and TVD scalar transport into FDS. One of the revelations of this work has been that the choice of starting point for deriving the divergence constraint naturally leads to two different forms of the divergence expression. While these forms are mathematically equivalent, they may lead to two completely different—and yet completely plausible—numerical formulations.

From continuity (D1 formulation) Starting from the continuity equation (3), we can factor out the velocity divergence leaving the material derivative of the density:

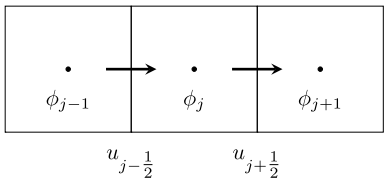

Fig. 1. Staggered grid in 1D. Scalars, ϕ, live at cell centers (indexed by j, $j+1$, etc.), and represent volume averaged values. Fluxes and velocity components live at cell faces (indexed by $j+\frac{1}{2}$, etc.) and represent surface averaged quantities.

$$\nabla \cdot \mathbf{u} = -\frac{1}{\rho}\frac{\mathrm{D}\rho}{\mathrm{D}t}. \tag{14}$$

Using the ideal gas law and differentiating the equation of state leads to

$$\begin{aligned}\nabla \cdot \mathbf{u} = &\left(\frac{1}{\rho c_p T} - \frac{1}{\bar{p}}\right)\frac{\mathrm{D}\bar{p}}{\mathrm{D}t} \\ &+ \frac{1}{\rho c_p T}\left[\dot{q}''' - \nabla \cdot \dot{\mathbf{q}}''\right] \\ &+ \frac{1}{\rho}\sum_\alpha \left(\frac{\overline{W}}{W_\alpha} - \frac{h_{s,\alpha}}{c_p T}\right)\left[\dot{m}''''_\alpha - \nabla \cdot \mathbf{J}_\alpha\right].\end{aligned} \tag{15}$$

From sensible enthalpy (D2 formulation) Alternatively, we may factor the velocity divergence from the sensible enthalpy transport equation (6):

$$\nabla \cdot \mathbf{u} = \frac{1}{\rho h_s}\left[\frac{\mathrm{D}}{\mathrm{D}t}(\bar{p} - \rho h_s) + \dot{q}''' - \nabla \cdot \dot{\mathbf{q}}''\right]. \tag{16}$$

From this starting point, the natural result for the divergence expression is

$$\begin{aligned}\nabla \cdot \mathbf{u} = &\frac{1}{\rho c_p T}\frac{\mathrm{D}\bar{p}}{\mathrm{D}t} - \frac{1}{\bar{p}}\frac{\partial \bar{p}}{\partial t} \\ &+ \frac{1}{\rho c_p T}\left[\dot{q}''' - \nabla \cdot \dot{\mathbf{q}}'' - \mathbf{u} \cdot \nabla(\rho h_s)\right] \\ &+ \frac{1}{\rho}\sum_\alpha \left(\frac{\overline{W}}{W_\alpha} - \frac{h_{s,\alpha}}{c_p T}\right)\left[\dot{m}''''_\alpha - \nabla \cdot \mathbf{J}_\alpha - \mathbf{u} \cdot \nabla(\rho Y_\alpha)\right].\end{aligned} \tag{17}$$

Comparison Notice the subtle differences between the first, second, and third lines of (15) and (17). The first lines differ by $(\mathbf{u} \cdot \nabla \bar{p})/\bar{p}$. In (17), the second and third lines each contain an extra term accounting for advection of enthalpy and mass, respectively, $\mathbf{u} \cdot \nabla(\rho h_s)$ and $\mathbf{u} \cdot \nabla(\rho Y_\alpha)$. Using (3)–(7), it can be shown that (15) and (17) are mathematically equivalent (see Appendix B). However, as will be shown in Section 4, ignoring the discrete forms of the advection terms in (17) may lead to energy conservation errors. These errors are the result of omitting the effects of numerical mixing.

For later comparison, we will refer to the divergence expressions given by (15) and (17) as the D1 and D2 formulations, respectively.

Time advancement issues The advantage of the D1 formulation can be understood by considering Step 4 in the solution procedure discussed above. We require the velocity divergence at the $n+1$ time level, and with Steps 1–3 complete it may be seen that all quantities needed to compute (15) are available. In particular, with D1 note that we *do not* require $\mathbf{u}^{n+1}$, which is yet to be computed in Step 6.

Conversely, with D2 the velocity shows up in the advection terms on the right-hand side. In practice, these velocities are evaluated from the beginning of the subinterval in the RK2 scheme. As shown below, this does not adversely impact the overall accuracy of the scheme. A detailed time marching algorithm is given in Appendix A.

3.1. Discrete operators

Before discussing the numerical formulation, it is necessary to define discrete notation and operators. FDS employs the staggered grid arrangement of Harlow and Welch [26], see Fig. 1. Discrete scalar quantities live at cell centers and physically represent cell volume averages. Velocities and fluxes are stored on or interpolated to cell faces and physically represent cell face surface averages.

We adopt the notation of Morinishi et al. [19], with modifications discussed below. For simplicity, we consider a uniform grid in each coordinate direction i with spacing Δx_i. For a discrete value ϕ, a centered difference about point x_i is denoted by

$$\frac{\delta\phi}{\delta x_i}\Big|_{x_i} \equiv \frac{\phi(x_i + \frac{1}{2}\Delta x_i) - \phi(x_i - \frac{1}{2}\Delta x_i)}{\Delta x_i}. \tag{18}$$

A value linearly interpolated in direction i at point x_i is given by

$$\bar{\phi}^{x_i}\Big|_{x_i} \equiv \frac{1}{2}\phi\left(x_i + \frac{1}{2}\Delta x_i\right) + \frac{1}{2}\phi\left(x_i - \frac{1}{2}\Delta x_i\right). \tag{19}$$

It is also convenient to establish notation for flux limited interpolation and flux corrected differencing operators, $\bar{\phi}^{fl,x_i}$ and $\delta_{fc}(\phi)$, respectively. These are defined in the following section.

3.2. Factoring the discrete flux divergence

Factoring the divergence out of the continuous, conservative form of the sensible enthalpy transport equation (6) to obtain the divergence expression (17) is a straight forward application of the product rule of calculus. Using the Morinishi operator notation on uniform Cartesian grids, the discrete analog of the product rule looks quite similar. There are, however, subtle differences that need to be discussed, especially when flux limiters are used in defining face values for scalar transport.

In this section, we show the numerical decomposition (discrete product rule) of the enthalpy flux divergence for cell j in one dimension (1D) (the directional index i is suppressed in this section to avoid clutter since we are discussing 1D operations). In the staggered grid arrangement, density ρ and sensible enthalpy h_s are stored at cell centers indexed by j, $j+1$, etc. Velocity u is stored at the cell face and indexed by $j+\frac{1}{2}$, etc. (Fig. 1). Here, an overline, $\bar{\phi}^{fl}$, applied to a face value ($j \pm \frac{1}{2}$ suffix) denotes a flux limited interpolation operator. The purpose of the flux limiter is to prevent spurious oscillations in the scalar solution. Such oscillations must be avoided because they may lead to boundedness violations and instability.

In decomposing the flux divergence our goal is to break the term (corresponding the second term on the LHS of (6)) into two parts as follows:

$$\begin{aligned}\left[\frac{\delta(\rho h_s u)}{\delta x}\right]_j &= \frac{\overline{(\rho h_s)}^{fl}_{j+\frac{1}{2}} u_{j+\frac{1}{2}} - \overline{(\rho h_s)}^{fl}_{j-\frac{1}{2}} u_{j-\frac{1}{2}}}{\Delta x}, \\ &= (\rho h_s)_j \underbrace{\frac{u_{j+\frac{1}{2}} - u_{j-\frac{1}{2}}}{\Delta x}}_{\left(\frac{\delta u}{\delta x}\right)_j} + \underbrace{\frac{\frac{1}{2}\delta_{fc}(\rho h_s)_{j+\frac{1}{2}} u_{j+\frac{1}{2}} + \frac{1}{2}\delta_{fc}(\rho h_s)_{j-\frac{1}{2}} u_{j-\frac{1}{2}}}{\Delta x}}_{\left[u\frac{\overline{\delta_{fc}(\rho h_s)}}{\delta x}\right]_j}.\end{aligned} \tag{20}$$

Here, $\delta_{fc}(\rho h_s)_{j+\frac{1}{2}}$ represents a flux corrected difference (also called a slope limiter [17]) of the scalar data (ρh_s in this case) at the face $j+\frac{1}{2}$. The flux corrected differences for cell j are defined such that

$$(\rho h_s)_j + \frac{1}{2}\delta_{fc}(\rho h_s)_{j+\frac{1}{2}} = \overline{(\rho h_s)}^{fl}_{j+\frac{1}{2}}, \tag{21}$$

$$(\rho h_s)_j - \frac{1}{2}\delta_{fc}(\rho h_s)_{j-\frac{1}{2}} = \overline{(\rho h_s)}^{fl}_{j-\frac{1}{2}}. \tag{22}$$

Note that while scalar face values are unique to the face ($\overline{(\rho h_s)}^{fl}_{j+\frac{1}{2}} = \overline{(\rho h_s)}^{fl}_{j+1-\frac{1}{2}}$), the flux corrected differences, in general, are not ($\delta_{fc}(\rho h_s)_{j+\frac{1}{2}} \neq \delta_{fc}(\rho h_s)_{j+1-\frac{1}{2}}$).

To orient the reader, below we provide two examples for factoring the discrete divergence with common, simple differencing schemes: Godunov (first-order upwinding) and central differencing. Then we present the general case for TVD flux limiters.

3.2.1. Example: pure upwinding

Suppose all $u > 0$ in 1D-flow from left to right. For Godunov's scheme (first-order upwinding) the flux corrected differences would be computed as follows:

$$\begin{aligned}\delta_{fc}(\rho h_s)_{j+\frac{1}{2}} &= 2\left[\overline{(\rho h_s)}^{fl}_{j+\frac{1}{2}} - (\rho h_s)_j\right],\\ &= 2\left[(\rho h_s)_j - (\rho h_s)_j\right],\\ &= 0,\end{aligned} \tag{23}$$

$$\begin{aligned}\delta_{fc}(\rho h_s)_{j-\frac{1}{2}} &= 2\left[(\rho h_s)_j - \overline{(\rho h_s)}^{fl}_{j-\frac{1}{2}}\right],\\ &= 2\left[(\rho h_s)_j - (\rho h_s)_{j-1}\right].\end{aligned} \tag{24}$$

Utilizing the second underbrace in (20) together with the Morinishi linear interpolation operator (19), the cell-average advection term becomes

$$\left[\overline{u\frac{\delta_{fc}(\rho h_s)}{\delta x}}\right]_j = u_{j-\frac{1}{2}}\left[\frac{(\rho h_s)_j - (\rho h_s)_{j-1}}{\Delta x}\right]. \tag{25}$$

3.2.2. Example: central differencing

For central differencing the flux corrected differences would be computed as follows:

$$\begin{aligned}\delta_{fc}(\rho h_s)_{j+\frac{1}{2}} &= 2\left[\overline{(\rho h_s)}^{fl}_{j+\frac{1}{2}} - (\rho h_s)_j\right],\\ &= \left[(\rho h_s)_j + (\rho h_s)_{j+1}\right] - 2(\rho h_s)_j,\\ &= (\rho h_s)_{j+1} - (\rho h_s)_j,\end{aligned} \tag{26}$$

$$\begin{aligned}\delta_{fc}(\rho h_s)_{j-\frac{1}{2}} &= 2\left[(\rho h_s)_j - \overline{(\rho h_s)}^{fl}_{j-\frac{1}{2}}\right],\\ &= 2(\rho h_s)_j - \left[(\rho h_s)_{j-1} + (\rho h_s)_j\right],\\ &= (\rho h_s)_j - (\rho h_s)_{j-1}.\end{aligned} \tag{27}$$

Again, using (20), the cell-average advection term is

$$\left[\overline{u\frac{\delta_{fc}(\rho h_s)}{\delta x}}\right]_j = \frac{1}{2}u_{j+\frac{1}{2}}\left[\frac{(\rho h_s)_{j+1} - (\rho h_s)_j}{\Delta x}\right] + \frac{1}{2}u_{j-\frac{1}{2}}\left[\frac{(\rho h_s)_j - (\rho h_s)_{j-1}}{\Delta x}\right]. \tag{28}$$

3.2.3. Flux limiters

In general, we first compute the flux-limited face values and obtain the flux corrected differences from (21) and (22). The cell-average advection term is then computed from the second underbrace in (20).

Flux limiters are implemented as follows. Consider face $j+\frac{1}{2}$ between cells j and $j+1$ and let ϕ denote a general scalar variable. The local (loc) and upstream (up) data variations are

$$\delta\phi_{loc} = \phi_{j+1} - \phi_j, \tag{29}$$

$$\delta\phi_{up} = \begin{cases}\phi_j - \phi_{j-1} & \text{if } u_{j+\frac{1}{2}} > 0,\\ \phi_{j+2} - \phi_{j+1} & \text{if } u_{j+\frac{1}{2}} < 0.\end{cases} \tag{30}$$

The limiter function $B(r)$ depends on the upstream-to-local data ratio, $r = \delta\phi_{up}/\delta\phi_{loc}$. The available limiters in FDS are shown below:

Flux limiter	$B(r)$
None (central differencing)	1
Godunov [17]	0
Superbee [31]	$\max(0, \min(2r, 1), \min(r, 2))$
CHARM [32]	$s(3s+1)/(s+1)^2;\ s = 1/r$

Once the limiter function value has been computed, the scalar face value is computed by

$$\overline{\phi}^{fl}_{j+1/2} = \begin{cases}\phi_j + B(r)\frac{1}{2}(\phi_{j+1} - \phi_j) & \text{if } u_{j+\frac{1}{2}} > 0,\\ \phi_{j+1} + B(r)\frac{1}{2}(\phi_j - \phi_{j+1}) & \text{if } u_{j+\frac{1}{2}} < 0.\end{cases} \tag{31}$$

Table 1
Parameters for Shunn et al. [35] manufactured solution.

Parameter	Value	Units
ρ_0	5	kg/m^3
ρ_1	1	kg/m^3
$k=\omega$	2	1/m, 1/s
$u_f=v_f$	0.5	m/s
$\rho\Gamma=\mu$	0.001	kg/(m s)

3.3. The discrete divergence

3.3.1. Discrete D1 formulation

Let Δ_1 denote the cell volume averaged divergence approximated using the D1 formulation with the differencing operators defined above (note that summation is implied over repeated Roman suffixes):

$$\begin{aligned}\Delta_1 = &\left[\frac{1}{\rho c_p T}-\frac{1}{\bar{p}}\right]\left(\frac{\partial \bar{p}}{\partial t}+w\rho_0 g_z\right)\\ &+\frac{1}{\rho c_p T}\left[\dot{q}'''-\frac{\delta \dot{q}_i''}{\delta x_i}\right]\\ &+\frac{1}{\rho}\sum_\alpha\left(\frac{\overline{W}}{W_\alpha}-\frac{h_{s,\alpha}}{c_p T}\right)\left[\dot{m}_\alpha'''-\frac{\delta J_{\alpha,i}}{\delta x_i}\right].\end{aligned} \tag{32}$$

3.3.2. Discrete D2 formulation

Discretizing the D2 formulation leads to

$$\begin{aligned}\Delta_2 = &\left[\frac{1}{\rho c_p T}-\frac{1}{\bar{p}}\right]\left(\frac{\partial \bar{p}}{\partial t}\right)+\frac{w\rho_0 g_z}{\rho c_p T}\\ &+\frac{1}{\rho c_p T}\left[\dot{q}'''-\frac{\delta \dot{q}_i''}{\delta x_i}-\overline{u_i\frac{\delta_{fc}(\rho h_s)}{\delta x_i}}^{x_i}\right]\\ &+\frac{1}{\rho}\sum_\alpha\left(\frac{\overline{W}}{W_\alpha}-\frac{h_{s,\alpha}}{c_p T}\right)\left[\dot{m}_\alpha'''-\frac{\delta J_{\alpha,i}}{\delta x_i}-\overline{u_i\frac{\delta_{fc}(\rho Y_\alpha)}{\delta x_i}}^{x_i}\right].\end{aligned} \tag{33}$$

4. Test cases

For the purpose of verifying and validating the FDS code for fire dynamics applications, the developers maintain a publicly available suite of test cases with the associated documentation [33,34]. The specific tests presented here are designed to (1) illustrate the overall accuracy of the scheme and (2) highlight the effects of numerical mixing on the energy budget. The first case is a verification case based on the method of manufactured solutions (MMS). The second case is a practical scenario in fire protection engineering; an energy budget error of 5% results if corrections are not employed. The last test case is a toy problem with a simple analytical solution designed to amplify the numerical mixing effects and to provide a platform for investigating the root cause of the energy conservation error.

4.1. Manufactured solution

To demonstrate the transport accuracy of the formulation, here we present numerical results for the manufactured solution proposed by Shunn et al. [35]. The parameters for this problem are given in Table 1. In the 2D sinusoidal solution given below, Z is the mixture fraction, ρ is the density, u and v are velocity components, and $\tilde{p}$ is the hydrodynamic pressure. The solution translates diagonally across the domain with velocity $[u_f, v_f]$. The translated positions are define by $\hat{x}\equiv x-u_f t$ and $\hat{y}\equiv y-v_f t$. The solution is spatially periodic on a square domain extending from -1 m to 1 m ($L=2$ m) on each side and has a time period of 1 s.

$$Z(x,y,t)=\frac{1+\sin(\pi k\hat{x})\sin(\pi k\hat{y})\cos(\pi\omega t)}{(1+\frac{\rho_0}{\rho_1})+(1-\frac{\rho_0}{\rho_1})\sin(\pi k\hat{x})\sin(\pi k\hat{y})\cos(\pi\omega t)} \tag{34}$$

$$\rho(x,y,t)=\left(\frac{Z(x,y,t)}{\rho_1}+\frac{1-Z(x,y,t)}{\rho_0}\right)^{-1} \tag{35}$$

$$u(x,y,t)=u_f+\frac{\rho_1-\rho_0}{\rho(x,y,t)}\left(\frac{-\omega}{4k}\right)\cos(\pi k\hat{x})\sin(\pi k\hat{y})\sin(\pi\omega t) \tag{36}$$

$$v(x,y,t) = v_f + \frac{\rho_1 - \rho_0}{\rho(x,y,t)}\left(\frac{-\omega}{4k}\right)\sin(\pi k\hat{x})\cos(\pi k\hat{y})\sin(\pi\omega t) \tag{37}$$

$$\tilde{p}(x,y,t) = \frac{1}{2}\rho(x,y,t)u(x,y,t)v(x,y,t) \tag{38}$$

Note that (35) is the equation of state (EOS) and may be rearranged for this simple problem to give Z in terms of ρ,

$$Z = \frac{(\rho_0 - \rho)\rho_1}{(\rho_0 - \rho_1)\rho}. \tag{39}$$

The source terms for the manufactured solution, $\dot{Q}_\rho$, etc., are defined as the residuals of the conservative form of the transport equations:

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = \dot{Q}_\rho, \tag{40}$$

$$\frac{\partial(\rho Z)}{\partial t} + \nabla\cdot(\rho Z\mathbf{u}) - \nabla\cdot(\rho\Gamma\nabla Z) = \dot{Q}_Z, \tag{41}$$

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \nabla\cdot(\rho\mathbf{u}\mathbf{u}) + \nabla\tilde{p} - \nabla\cdot\mathsf{T} = \dot{Q}_\mathbf{u}, \tag{42}$$

where the components of the deviatoric stress tensor, T, are given by

$$T_{ij} = 2\mu\left(S_{ij} - \frac{1}{3}S_{kk}\delta_{ij}\right); \quad S_{ij} \equiv \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{43}$$

The source terms may be obtained from [35] or by running the sympy (Symbolic Python) script provided in Appendix D. A feature of this particular solution is that the mass source is zero, $\dot{Q}_\rho = 0$, which provides a more realistic test case and simplifies the MMS implementation.

The divergence expression is obtained by differentiating the EOS:

$$\nabla\cdot\mathbf{u} = \left[\frac{1}{\rho_1} - \frac{1}{\rho_0}\right]\left(\nabla\cdot(\rho\Gamma\nabla Z) + \dot{Q}_Z\right). \tag{44}$$

For this isothermal flow there is no alternative divergence formulation to consider. The RHS of the momentum equation (8) is augmented as follows:

$$\frac{\partial\mathbf{u}}{\partial t} = -(\mathbf{F} + \nabla H) + \dot{Q}_\mathbf{u}/\rho. \tag{45}$$

And the Poisson equation for H becomes

$$\nabla^2 H = -\left[\nabla\cdot(\mathbf{F} - \dot{Q}_\mathbf{u}/\rho) + \frac{\partial}{\partial t}(\nabla\cdot\mathbf{u})\right]. \tag{46}$$

Manufactured solution procedure For the flow to remain isothermal the ideal gas law must be obeyed identically. Since a transport equation is solved for density, the mixture fraction must be obtained from the EOS. Hence, in this problem Z plays the role usually played by temperature in a typical fire problem. For the manufactured solution, the basic FDS algorithm is modified as follows. Given u_i^n, ρ^n, and Z^n at time t^n:

1. Update the density equation (40) to obtain ρ^{n+1}.
2. Obtain Z^{n+1} from the equation of state (39).
3. Compute $(\nabla\cdot\mathbf{u})^{n+1}$ from (44).
4. Solve the Poisson equation (46) for H^n.
5. Update momentum (45) using the pressure field from Step 4 to obtain u_i^{n+1}.

As before, this procedure represents a single stage in the RK2 time integration scheme.

Simulations were performed for $N = \{32, 64, 128, 256, 512\}$, where N is the number of cells in each direction, using an adaptive time step satisfying both convective and diffusive CFL numbers of 0.5. The time step criterion and the simulation parameters in Table 1 were chosen to match Shunn et al. [35]. The density equation (40) is solved using the CHARM flux limiter. Qualitative results for the 256^2 case are shown in Fig. 2. These images may be compared to the images presented in [35]. In Fig. 3, we plot the L2 error at time $t = 1$ s as a function of grid spacing, $\Delta x = L/N$, confirming second-order accuracy of the solutions for density, mixture fraction, and velocity. As shown in Fig. 3, schemes such as ours, where the pressure is set to zero at the beginning of each stage of the integration (in other words, no pressure gradient term shows up in the force on the RHS of the Poisson equation), are known to be first-order accurate for pressure [12]. This is not a severe limitation of the algorithm since the hydrodynamic pressure does not factor into the equation of state for low-Mach flows.

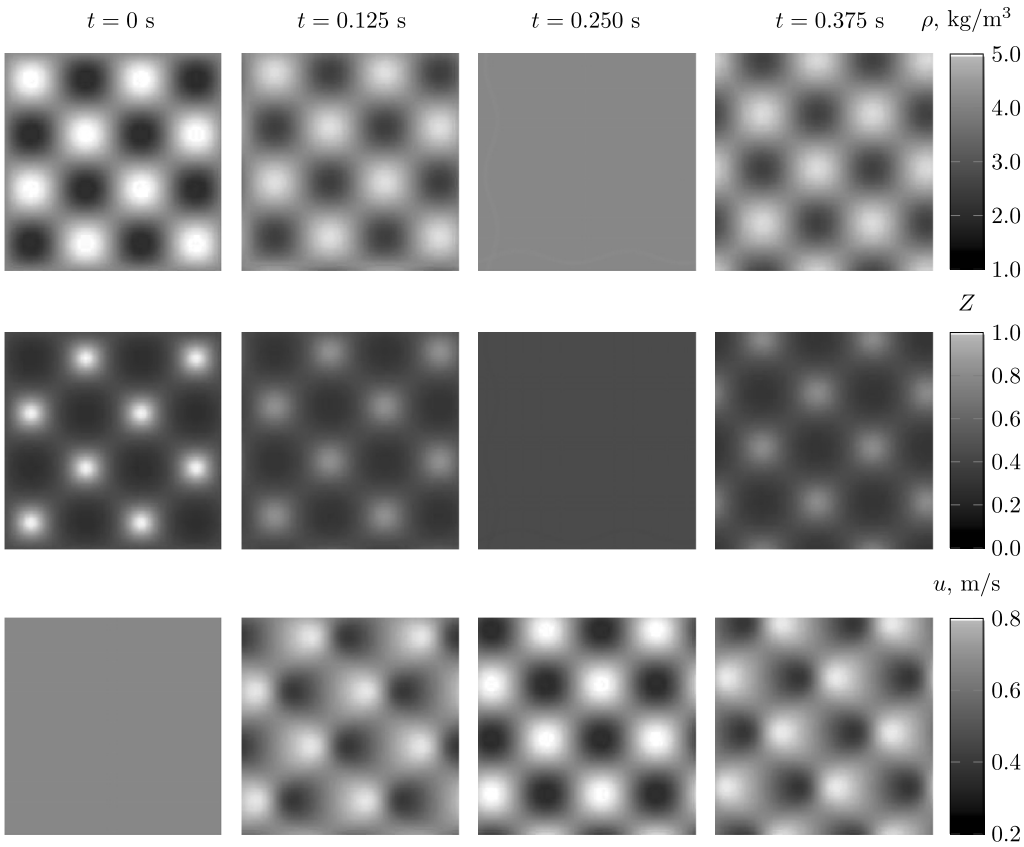

Fig. 2. Evolution of the manufactured solution. From top to bottom, the images show density, ρ, mixture fraction, Z, and the u-velocity component from the 256^2 simulation at the times shown at the top of the columns. These results may be compared to [35] to confirm the validity of the numerical solution.

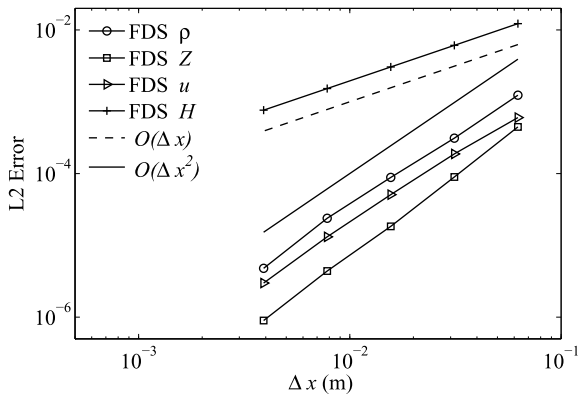

Fig. 3. Convergence for the manufactured solution. The L2 error at time $t = 1$ s is plotted as a function of grid spacing for $N = \{32, 64, 128, 256, 512\}$ points in each direction. The grid spacing is $\Delta x = L/N$, where $L = 2$ m. Calculations were performed with an adaptive time step satisfying both convective and diffusive CFL (Von Neumann) limits of 0.5. These results confirm second-order accuracy of the flow solver for density, mixture fraction, and velocity. As is known for projection schemes like ours, the pressure solution (represented by H) is first-order accurate [12].

4.2. Energy budget

In this problem, which is meant to mimic a realistic fire scenario, a 1200 kW naturally ventilated propane fire is simulated in a 10 m × 10 m × 5 m compartment with a side inlet vent near the floor on one side and a small open outlet vent on the ceiling on the opposite side (see Fig. 4). The grid resolution is 25 cm in each direction. (While this resolution is extremely coarse for LES, it is typical of engineering applications for fire where the computational domain may be the inside of a shopping mall and the fire may be localized in a restaurant kitchen [10].) As discussed above, the chemistry is treated by a simple one-step reaction: *fuel* + *air* → *products*. The *fuel* and *products* are transported as lumped species

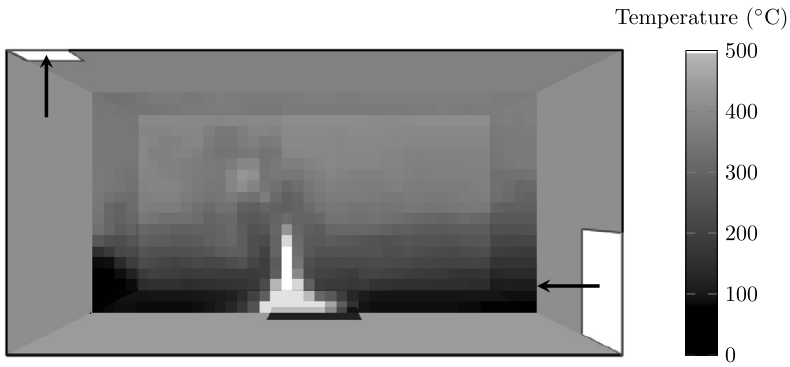

Fig. 4. Energy budget. A 1200 kW propane burner in the center of a compartment with adiabatic walls draws ambient air in from the side vent and discharges hot products out the top vent. A temperature slice through the center of the compartment at quasi-steady state shows the stratified atmosphere and the flame leaning due to the cross ventilation. The pixelation in the temperature contour gives the reader a qualitative sense of the relatively coarse grid resolution (25 cm).

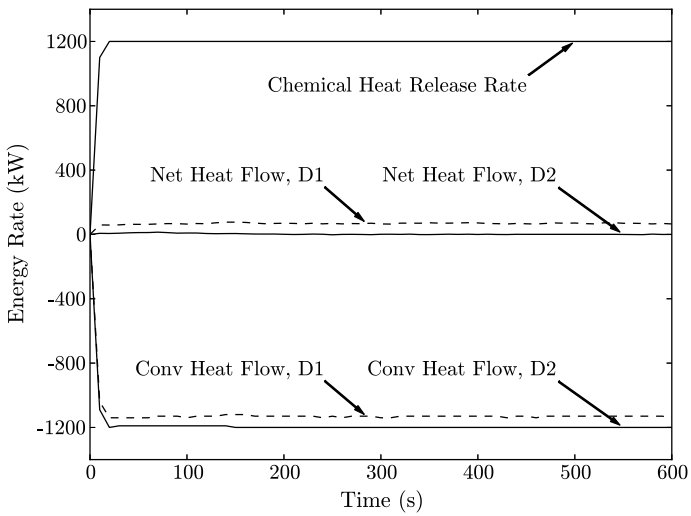

Fig. 5. Energy budget. The D1 scheme under-predicts the convective outflow of energy by 5%. Hence, in the D1 scheme the heat release rate and the outflow energy are not balanced. The net heat flow—the sum of the heat release rate and the convective heat flow—should be zero, as shown by the D2 scheme. In this case, the D1 scheme results in an over-prediction of the mean compartment temperature.

as discussed in [10]. The Superbee flux limiter [31] (FDS default) is employed for scalar transport (although not shown, similar results are found with CHARM). A critical aspect of this test problem is that each lumped species is modeled using temperature-dependent specific heats from the JANAF tables [36] (as discussed later, the D1 and D2 schemes are equivalent under the limiting assumption of constant specific heat ratio). The walls are adiabatic. And, to simplify the case even further, the radiation transport algorithm is turned off.

At a statistically stationary state the 1200 kW generated by the burner is balanced by convective heat flow into and out of the compartment through the vents. The plot in Fig. 5 shows both the chemical heat release rate and net enthalpy flow out of the compartment as a function of time for the D1 and D2 divergence formulations. At steady state, the balance should be zero. Note that the reported heat flows are time averaged over a 10 s interval (which is why the curves appear nearly perfectly smooth). The D1 divergence algorithm predicts the heat flow out of the domain to be low by approximately 5%—the mean compartment temperature is thus over-predicted by the D1 scheme. As can be seen, the D2 scheme nearly perfectly conserves energy.

4.3. Mixing gases with different specific heats

This case dramatically illustrates the problem of using variable specific heats with the discrete D1 divergence expression on coarse grids and thus amplifies the effects shown in the previous test case. The physical fluid properties of diffusivity and thermal conductivity are set to zero, so that *all* mixing is numerical.

In a coarse 2D channel, hot gas (pure species A) at 900 °C with a constant specific heat of $c_p = 1$ kJ/(kg K) enters from the bottom left vent with a flux of 1 kg/(m^2 s) and cold gas (pure species B) at 20 °C with a constant specific heat of

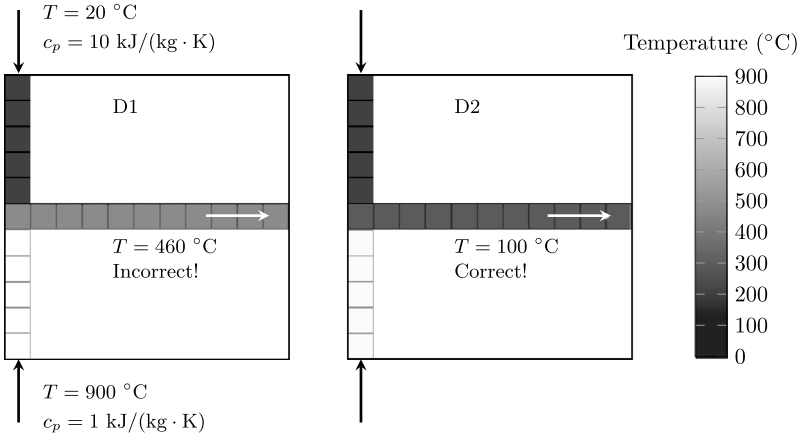

Fig. 6. T-mix problem. Equal mass fluxes of a hot gas with low specific heat (from bottom-left) and a cold gas with a high specific heat (from top-left) mix and exit the domain (center-right). The computational domain in made up of an 11×11 square grid with blocked cells (blank white region) to create one-cell-thick channels for fluid flow. (Left) D1 formulation produces the incorrect mass-weighted temperature of 460 °C. (Right) D2 formulation shows the correct enthalpy-weighted mixture temperature of 100 °C.

$c_p = 10$ kJ/(kg K) enters from a top left vent with the same mass flux. The gases mix (numerically) in a center channel *one cell thick* and exit the domain flowing to the right as depicted in Fig. 6. The resulting mixture outlet temperature depends critically on the correct specification of the velocity divergence in the computational cell at the T-mix junction.

The enthalpy flux, $\rho h_s u$, of the mixture at the outlet is

$$1\ \mathrm{kg}/\left(\mathrm{m}^2\,\mathrm{s}\right) \times 1\ \mathrm{kJ}/(\mathrm{kg\,K}) \times 1173.15\ \mathrm{K} + 1\ \mathrm{kg}/\left(\mathrm{m}^2\,\mathrm{s}\right) \times 10\ \mathrm{kJ}/(\mathrm{kg\,K}) \times 293.15\ \mathrm{K} = 4104.7\ \mathrm{kJ}/\left(\mathrm{m}^2\,\mathrm{s}\right).$$

The mixture heat capacity at the outlet is 5.5 kJ/(kg K) and the flux is 2 kg/(m^2 s). Therefore, the correct outlet temperature is

$$4104.7\ \mathrm{kJ}/\left(\mathrm{m}^2\,\mathrm{s}\right)/2\ \mathrm{kg}/\left(\mathrm{m}^2\,\mathrm{s}\right)/5.5\ \mathrm{kJ}/(\mathrm{kg\,K}) - 273.15\ \mathrm{K} = 100\,^\circ\mathrm{C}.$$

As can be seen from Fig. 6, the D2 scheme accurately computes the outlet mixture temperature, while the D1 scheme produces a mass-weighted mixture temperature

$$(293.15\ \mathrm{K} + 1173.15\ \mathrm{K})/2 - 273.15\ \mathrm{K} = 460\,^\circ\mathrm{C}.$$

5. Discussion

Since the FDS algorithm solves a transport equation for mass density and obtains the divergence from the enthalpy equation, it is reasonable to ask the question: Are the correction terms derived for the D2 scheme valid for algorithms which solve a transport equation for enthalpy and obtain density from the equation of state (e.g., [4])? The answer is *yes*, though the relative importance of these terms may depend on the resolution of the flow field. Well-resolved DNS calculations will likely find these corrections to be negligible. In LES, however, we expect significant kinetic energy content at high wavenumbers on the grid and, especially when dynamic subgrid models [37,38] are employed, we often have locally very low values of the physical transport coefficients with relatively coarse grid resolution. Numerical mixing then becomes an issue.

To prove that the numerical mixing terms are independent of the solution procedure, let us look more closely at the T-mix problem. As mentioned above, the correct outlet mixture temperature depends critically on the correct calculation of the velocity divergence at the T-mix junction. The inlet mass fluxes and temperatures are specified (see Table 2) and the densities may be found from the ideal gas law. The background pressure $\bar{p} = 101\,325$ Pa is constant, the molecular weight is taken to be that of air, $\overline{W} = 29$ kg/kmol. The grid spacing is set to $\Delta x = \Delta y = 1$ m. A mass and energy balance around the T-mix junction cell shown in Fig. 7 leads to the cell face values shown in Table 2. Thus, there is a simple solution for the cell velocity divergence:

$$\Delta = \frac{u_3 - 0}{\Delta x} + \frac{u_2 - u_1}{\Delta y} = -2.037\ \mathrm{s}^{-1}. \tag{47}$$

Now, let us compute the divergence using D1 (32) and D2 (33). Because all physical transport coefficients are zero, and the background pressure is constant, and there is no chemical heat release, D1 trivially returns a value of zero for the divergence. The result within a CFD code (which projects the velocity field to satisfy the divergence constraint) is that the gas is not sufficiently compressed, resulting in a lower outlet density, and a fictitiously high temperature, as shown by the D1 result in Section 4.3.

Table 2
Face values at the T-mix junction. Values on Face 3 in *italic* font are computed from a mass and energy balance at steady state. Other values are specified or computed directly from specified quantities. The background pressure is $\bar{p} = 101\,325$ Pa and the molecular weight is $\overline{W} = 29$ kg/kmol.

Face		1	2	3
Mass fraction A	Y_A	1.000	0.000	*0.500*
Mass fraction B	Y_B	0.000	1.000	*0.500*
Mass flux (kg/(m^2 s))	ρu	1.000	−1.000	*2.000*
Enthalpy flux (kJ/(m^2 s))	$\rho u c_p T$	1173	−2932	*4105*
Specific heat (kJ/(kg K))	c_p	1.000	10.00	*5.500*
Temperature (°C)	T	900.0	20.00	*100.0*
Density (kg/m^3)	ρ	0.3012	1.206	*0.9471*
Velocity (m/s)	u	3.320	−0.8295	*2.112*

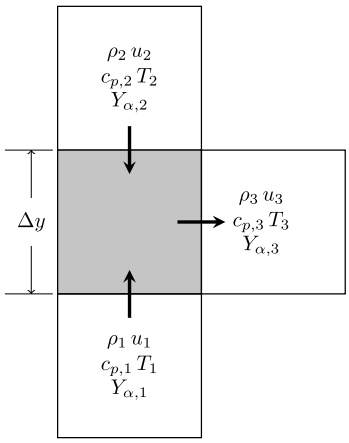

Fig. 7. T-mix junction. The solution is given in Table 2.

We now compute the divergence using D2 from (33). For simplicity, we use Godunov's scheme to compute the average advection term in (20). Note that the mixture values within the junction cell are equivalent to the values on outlet Face 3. Further, note that the flux corrected differences (for both enthalpy and mass fraction) across Face 3 are zero and that the normal component of velocity on the left face of the junction cell (the wall) is also zero—hence, no terms in the x direction survive. The resulting divergence is

$$\begin{aligned}\Delta_2 = &-\frac{1}{\rho_3 c_{p,3} T_3}\left[\frac{(\rho_2 c_{p,2} T_2 - \rho_3 c_{p,3} T_3)u_2 + (\rho_3 c_{p,3} T_3 - \rho_1 c_{p,1} T_1)u_1}{\Delta y}\right] \\ &- \frac{1}{\rho_3}\left(1 - \frac{c_{p,1}}{c_{p,3}}\right)\left[\frac{(\rho_2 Y_{A,2} - \rho_3 Y_{A,3})u_2 + (\rho_3 Y_{A,3} - \rho_1 Y_{A,1})u_1}{\Delta y}\right] \\ &- \frac{1}{\rho_3}\left(1 - \frac{c_{p,2}}{c_{p,3}}\right)\left[\frac{(\rho_2 Y_{B,2} - \rho_3 Y_{B,3})u_2 + (\rho_3 Y_{B,3} - \rho_1 Y_{B,1})u_1}{\Delta y}\right] \\ = &-2.037\ \mathrm{s}^{-1}\end{aligned} \tag{48}$$

Note that the second and third lines of (48) cancel, as they must, since these terms emanate from the time derivative of the enthalpy (see Appendix C) and this is a steady state problem. This may be readily verified by hand since $Y_A + Y_B = 1$ and $c_{p,3} = Y_{A,3} c_{p,1} + Y_{B,3} c_{p,2}$.

Remark. In principle, the D2 scheme does not rely on TVD schemes for scalar transport. As shown in the example above, a central difference is a perfectly valid means of computing the scalar face value and the flux corrected difference needed in (20) is easily constructed (though central differencing leads to dispersion error, making analysis of the T-mix problem more complex). Other transport schemes (e.g., QUICK [39,40]) may also be used.

Simplification For an ideal gas, the heat capacity may be written in terms of the specific heat ratio $\gamma = c_p/c_v$:

$$c_p = \frac{R}{W}\frac{\gamma}{\gamma - 1}. \tag{49}$$

If γ is taken to be constant and equal for all species, then $\rho h_s = \bar{p}\,\gamma/(\gamma - 1)$. Thus, if the background pressure may be assumed constant and uniform, both (15) and (17) simplify to

$$\nabla \cdot \mathbf{u} = \frac{1}{\rho h_s}\left[\dot{q}''' - \nabla \cdot \dot{\mathbf{q}}''\right]. \tag{50}$$

In this case, no corrections for numerical mixing are needed.

6. Conclusions

In this paper, we present a formulation for the velocity divergence for low-Mach flow solvers based on the discrete, conservative form of the sensible enthalpy transport equation. This divergence constraint is required in order to achieve discrete energy conservation when using TVD scalar transport schemes together with temperature-dependent specific heats. Through a practical example, it is shown that neglecting the correction terms for numerical mixing may lead to significant error in the energy budget.

Second-order accuracy of the flow solver for transport of density and velocity is confirmed using the method of manufactured solutions. The time marching procedure is formally first order due to the splitting scheme used for infinitely fast chemistry.

While the correction terms derived in this paper are implemented within the FDS algorithm, it is suggested that these same corrections may be useful in accelerating convergence in algorithms which transport enthalpy instead of density. The formulation is also valid for unstructured solvers, though the scalar transport schemes would need to be altered appropriately.

Acknowledgements

The author gratefully acknowledges the contributions of Kevin McGrattan and the entire FDS development team. The author would also like to thank Boris Stock for his help in developing the energy budget test problem, Jason Floyd for his help developing the T-mix test problem, and Jung-il Choi for his constructive critique of the manuscript.

Appendix A. Time marching algorithm for low-Mach reacting flows

The following procedure gives the full FDS time marching scheme. Considering transport only, the procedure is basically a first-order projection method embedded in a two-stage Runge–Kutta scheme (RK2) [28]. When considering the reaction substep the procedure breaks from a formal RK scheme and is better described as a predictor–corrector method with reaction implemented after the transport corrector. The RK scheme is second-order accurate. Chemistry is time split from transport in order to guarantee discrete conservation and to allow the flux limiter transport schemes to perform optimally. This splitting scheme is formally first-order in time (but strictly mass conservating), which is permissible given the stiffness of combustion chemistry in most practical fire applications.

A.1. RK stage 1 (predictor)

The following equations give the discrete updates performed in the code in procedural order. Superscripts, n, $n+1$, etc., denote discrete time values, $t^{n+1} = t^n + \Delta t$. The superscript * denotes a "predicted" or intermediate value. In general, index notation is used to indicate coordinate directions (summation over repeated Roman suffixes is implied). It is sometimes more conventional, however, to use x, y, z for x_1, x_2, x_3 to indicate direction and u, v, w for u_1, u_2, u_3 to indicate velocity components.

Predictor step 1: transport and equation of state.

$$\rho^* = \rho^n - \Delta t\left[\frac{\delta(\overline{\rho}^{fl,x_i} u_i)^n}{\delta x_i}\right] \tag{A.1}$$

$$(\rho Y_\alpha)^* = (\rho Y_\alpha)^n - \Delta t\left[\frac{\delta(\overline{\rho Y_\alpha}^{fl,x_i} u_i)^n}{\delta x_i} + \frac{\delta J^n_{\alpha,i}}{\delta x_i}\right]; \quad \text{for } \alpha = 1, \ldots, n_s - 1 \tag{A.2}$$

$$Y^*_\alpha = (\rho Y_\alpha)^*/\rho^*; \quad \text{for } \alpha = 1, \ldots, n_s - 1; \quad Y^*_{n_s} = 1 - \sum_{\alpha=1}^{n_s-1} Y^*_\alpha \tag{A.3}$$

$$\overline{W}^* = \left[\sum_\alpha Y^*_\alpha / W_\alpha\right]^{-1} \tag{A.4}$$

$$\bar{p}^* = \bar{p}^n + \Delta t\left(\frac{\partial \bar{p}}{\partial t}\right)^n \tag{A.5}$$

$$T^* = \frac{\bar{p}^* \overline{W}^*}{\rho^* R} \tag{A.6}$$

Predictor step 2: compute the divergence.

$$\Delta^* = \left[\frac{1}{(\rho c_p T)^*} - \frac{1}{\bar{p}^*}\right]\left(\frac{\partial \bar{p}}{\partial t}\right)^* + \frac{w^n \rho_0 g_z}{(\rho c_p T)^*} + \frac{1}{(\rho c_p T)^*}\left[\dot{q}'''^n - \frac{\delta \dot{q}_i''^*}{\delta x_i} - \overline{u_i^n \frac{\delta_{fc}(\rho h_s)^*}{\delta x_i}}^{x_i}\right] + \frac{1}{\rho^*}\sum_\alpha \left(\frac{\overline{W}^*}{W_\alpha} - \frac{h_{s,\alpha}^*}{(c_p T)^*}\right)\left[\dot{m}_\alpha'''^n - \frac{\delta J_{\alpha,i}^*}{\delta x_i} - \overline{u_i^n \frac{\delta_{fc}(\rho Y_\alpha)^*}{\delta x_i}}^{x_i}\right] \tag{A.7}$$

$$= D^* - P^*\left(\frac{\partial \bar{p}}{\partial t}\right)^* \tag{A.8}$$

The discrete divergence is computed in two parts. Eq. (A.7) is rewritten as shown in (A.8). First, D^* is computed. Next, the time derivative of the background pressure is obtained from a discrete version of the following volume integral:

$$\left(\frac{\partial \bar{p}}{\partial t}\right)^* = \left(\int_\Omega D^* \,\mathrm{d}V - \int_{\partial\Omega} \mathbf{u}^n \cdot \mathrm{d}\mathbf{S}\right) \Big/ \int_\Omega P^* \,\mathrm{d}V, \tag{A.9}$$

where the Ω is the computational domain and $\partial\Omega$ is the domain boundary. Generally, the boundary values of the normal component of velocity are known and do not impose time accuracy issues. Once (A.9) is solved, (A.8) may be completed. Note that usually the background pressure is constant in open fire calculations and (A.9) is identically zero for all examples shown in this paper.

Predictor step 3: solve Poisson equation.

$$\frac{\delta}{\delta x_i}\left(\frac{\delta H^n}{\delta x_i}\right) = -\left[\frac{\delta F_i^n}{\delta x_i} + \frac{\Delta^* - \Delta^n}{\Delta t}\right]; \quad \text{note: } \Delta^n = \frac{\delta u_i^n}{\delta x_i} \tag{A.10}$$

The Poisson equation for the Bernoulli integral, H, is formed from taking the divergence of the momentum equation. Below, the divergence values denoted Δ^n, Δ^*, etc., are obtained from the thermodynamic divergence constraint. It is *critical* that these divergence fields exactly match the corresponding discrete divergence formed from the staggered velocity field, $\delta u_i^n/\delta x_i$, for example. This constraint is guaranteed by the time marching procedure.

Predictor step 4: update the velocity field.

$$u_i^* = u_i^n - \Delta t\left[F_i^n + \frac{\delta H^n}{\delta x_i}\right]; \quad \text{note: } \frac{\delta u_i^*}{\delta x_i} = \Delta^* \text{ by construction} \tag{A.11}$$

As noted, taking the discrete divergence of (A.11) recovers (A.10). Thus, since H^n satisfies both (A.10) and (A.11) and it is given that $\delta u_i^n/\delta x_i = \Delta^n$, it is guaranteed that $\delta u_i^*/\delta x_i = \Delta^*$.

A.2. RK stage 2 (corrector), includes chemistry substep

Corrector step 1: transport and equation of state for intermediate composition and temperature.

$$\rho^{n+1} = \frac{1}{2}(\rho^* + \rho^n) - \frac{\Delta t}{2}\left[\frac{\delta(\overline{\rho}^{fl,x_i} u_i)^*}{\delta x_i}\right] \tag{A.12}$$

$$(\rho Y_\alpha)^{**} = \frac{1}{2}\left((\rho Y_\alpha)^* + (\rho Y_\alpha)^n\right) - \Delta t\left[\frac{\delta(\overline{\rho Y_\alpha}^{fl,x_i} u_i)^*}{\delta x_i} + \frac{\delta J_{\alpha,i}^*}{\delta x_i}\right]; \quad \text{for } \alpha = 1, \ldots, n_s - 1 \tag{A.13}$$

$$Y_\alpha^{**} = (\rho Y_\alpha)^{**}/\rho^{n+1}; \quad \text{for } \alpha = 1, \ldots, n_s - 1; \quad Y_{n_s}^{**} = 1 - \sum_{\alpha=1}^{n_s-1} Y_\alpha^{**} \tag{A.14}$$

$$\overline{W}^{**} = \left[\sum_\alpha Y_\alpha^{**}/W_\alpha\right]^{-1} \tag{A.15}$$

$$\bar{p}^{n+1} = \bar{p}^* + \frac{\Delta t}{2}\left[\left(\frac{\partial \bar{p}}{\partial t}\right)^* + \left(\frac{\partial \bar{p}}{\partial t}\right)^n\right] \tag{A.16}$$

$$T^{**} = \frac{\bar{p}^{n+1}\overline{W}^{**}}{\rho^{n+1} R} \tag{A.17}$$

Corrector step 2: chemistry substep, final updates of species, temperature, and chemical heat release.

$$(\rho Y_\alpha)^{n+1} = (\rho Y_\alpha)^{**} + \Delta t\, \dot{m}_\alpha'''^{\,n+1}; \quad \dot{m}_\alpha'''^{\,n+1} \equiv \dot{m}_\alpha'''(\mathbf{Y}^{**}, T^{**}) \tag{A.18}$$

$$Y_\alpha^{n+1} = (\rho Y_\alpha)^{n+1}/\rho^{n+1}; \quad \text{for } \alpha = 1, \ldots, n_s - 1; \quad Y_{n_s}^{n+1} = 1 - \sum_{\alpha=1}^{n_s-1} Y_\alpha^{n+1} \tag{A.19}$$

$$\overline{W}^{n+1} = \left[\sum_\alpha Y_\alpha^{n+1}/W_\alpha\right]^{-1} \tag{A.20}$$

$$T^{n+1} = \frac{\bar{p}^{n+1}\overline{W}^{n+1}}{\rho^{n+1}R} \tag{A.21}$$

$$\dot{q}'''^{\,n+1} = -\sum_\alpha \dot{m}_\alpha'''^{\,n+1} h_{f,\alpha}^0; \quad \text{where } h_{f,\alpha}^0 \text{ are heats of formation} \tag{A.22}$$

Note that, in general, the chemical source term, $\dot{m}_\alpha'''(\mathbf{Y}^{**}, T^{**})$, may be obtained from an Arrhenius rate law or a simple turbulent combustion model like EDC [24].

Corrector step 3: compute divergence.

$$\begin{aligned}\Delta^{n+1} &= \left[\frac{1}{(\rho c_p T)^{n+1}} - \frac{1}{\bar{p}^{n+1}}\right]\left(\frac{\partial \bar{p}}{\partial t}\right)^{n+1} + \frac{w^* \rho_0 g_z}{(\rho c_p T)^{n+1}} \\ &\quad + \frac{1}{(\rho c_p T)^{n+1}}\left[\dot{q}'''^{\,n+1} - \frac{\delta \dot{q}_i''^{\,n+1}}{\delta x_i} - \overline{u_i^* \frac{\delta_{fc}(\rho h_s)^{n+1}}{\delta x_i}}^{x_i}\right] \\ &\quad + \frac{1}{\rho^{n+1}}\sum_\alpha\left(\frac{\overline{W}^{n+1}}{W_\alpha} - \frac{h_{s,\alpha}^{n+1}}{(c_p T)^{n+1}}\right)\left[\dot{m}_\alpha'''^{\,n+1} - \frac{\delta J_{\alpha,i}^{n+1}}{\delta x_i} - \overline{u_i^* \frac{\delta_{fc}(\rho Y_\alpha)^{n+1}}{\delta x_i}}^{x_i}\right]\end{aligned} \tag{A.23}$$

$$= D^{n+1} - P^{n+1}\left(\frac{\partial \bar{p}}{\partial t}\right)^{n+1} \quad \text{where} \tag{A.24}$$

$$\left(\frac{\partial \bar{p}}{\partial t}\right)^{n+1} = \left(\int_\Omega D^{n+1}\,\mathrm{d}V - \int_{\partial\Omega} \mathbf{u}^* \cdot \mathrm{d}\mathbf{S}\right) \Big/ \int_\Omega P^{n+1}\,\mathrm{d}V \tag{A.25}$$

Corrector step 4: solve Poisson equation.

$$\frac{\delta}{\delta x_i}\left(\frac{\delta H^*}{\delta x_i}\right) = -\left[\frac{\delta F_i^*}{\delta x_i} + \frac{\Delta^{n+1} - \frac{1}{2}(\Delta^* + \Delta^n)}{\frac{1}{2}\Delta t}\right] \tag{A.26}$$

Corrector step 5: update velocity field.

$$u_i^{n+1} = \frac{1}{2}(u_i^* + u_i^n) - \frac{\Delta t}{2}\left[F_i^* + \frac{\delta H^*}{\delta x_i}\right]; \quad \text{note: } \frac{\delta u_i^{n+1}}{\delta x_i} = \Delta^{n+1} \text{ by construction} \tag{A.27}$$

Appendix B. Equivalence between divergence expressions

The equivalence between (15) and (17) is apparent based on the following:

$$\begin{aligned}&-\frac{1}{\rho c_p T}\mathbf{u}\cdot\nabla(\rho h_s) - \frac{1}{\rho}\sum_\alpha\left(\frac{\overline{W}}{W_\alpha} - \frac{h_{s,\alpha}}{c_p T}\right)\mathbf{u}\cdot\nabla(\rho Y_\alpha) \\ &= -\frac{1}{\rho c_p T}[\rho\mathbf{u}\cdot\nabla h_s + h_s\mathbf{u}\cdot\nabla\rho] - \frac{1}{\rho}\sum_\alpha\left(\frac{\overline{W}}{W_\alpha} - \frac{h_{s,\alpha}}{c_p T}\right)[\rho\mathbf{u}\cdot\nabla Y_\alpha + Y_\alpha\mathbf{u}\cdot\nabla\rho], \\ &= -\frac{1}{c_p T}\mathbf{u}\cdot\sum_\alpha[Y_\alpha\nabla h_{s,\alpha} + h_{s,\alpha}\nabla Y_\alpha] - \frac{1}{\rho}\mathbf{u}\cdot\nabla\rho - \sum_\alpha\left(\frac{\overline{W}}{W_\alpha} - \frac{h_{s,\alpha}}{c_p T}\right)\mathbf{u}\cdot\nabla Y_\alpha, \\ &= -\frac{1}{c_p T}\mathbf{u}\cdot\sum_\alpha Y_\alpha\nabla h_{s,\alpha} - \frac{1}{\rho}\mathbf{u}\cdot\nabla\rho - \sum_\alpha\frac{\overline{W}}{W_\alpha}\mathbf{u}\cdot\nabla Y_\alpha,\end{aligned}$$

$$
\begin{aligned}
&= -\frac{1}{c_p T}\mathbf{u}\cdot\sum_\alpha Y_\alpha c_{p,\alpha}\nabla T - \frac{1}{\rho}\mathbf{u}\cdot\nabla\rho - \mathbf{u}\cdot\sum_\alpha \overline{W}\nabla(Y_\alpha/W_\alpha),\\
&= -\mathbf{u}\cdot\left[\frac{1}{T}\nabla T + \frac{1}{\rho}\nabla\rho + \overline{W}\nabla(1/\overline{W})\right],\\
&= -\frac{1}{\bar{p}}\mathbf{u}\cdot\nabla\bar{p}.
\end{aligned}
\tag{B.1}
$$

The implication of (B.1) is that care must be taken to ensure that the discretizations of the thermodynamic state variables (in this case, enthalpy and species mass fractions) are consistent with the prescribed variation of the background thermodynamic pressure (the assumption underlying the low-Mach formulation).

Appendix C. Decomposing the time derivative

Using the ideal gas law, the enthalpy time derivative can be decomposed as follows:

$$
\begin{aligned}
\frac{\partial(\rho h_s)}{\partial t} &= \rho\frac{\partial h_s}{\partial t} + h_s\frac{\partial\rho}{\partial t},\\
&= \rho\sum_\alpha\left(Y_\alpha c_{p,\alpha}\frac{\partial T}{\partial t} + h_{s,\alpha}\frac{\partial Y_\alpha}{\partial t}\right) + h_s\frac{\partial\rho}{\partial t},\\
&= \rho c_p\frac{\partial T}{\partial t} + \rho\sum_\alpha h_{s,\alpha}\frac{\partial Y_\alpha}{\partial t} + h_s\frac{\partial\rho}{\partial t},\\
&= \rho c_p T\left[\frac{1}{\bar{p}}\frac{\partial\bar{p}}{\partial t} + \frac{1}{\overline{W}}\frac{\partial\overline{W}}{\partial t} - \frac{1}{\rho}\frac{\partial\rho}{\partial t}\right] + \rho\sum_\alpha h_{s,\alpha}\frac{\partial Y_\alpha}{\partial t} + h_s\frac{\partial\rho}{\partial t},\\
&= \rho c_p T\left[\frac{1}{\bar{p}}\frac{\partial\bar{p}}{\partial t} - \sum_\alpha\frac{\overline{W}}{W_\alpha}\frac{\partial Y_\alpha}{\partial t} - \frac{1}{\rho}\frac{\partial\rho}{\partial t}\right] + \rho\sum_\alpha h_{s,\alpha}\frac{\partial Y_\alpha}{\partial t} + h_s\frac{\partial\rho}{\partial t},\\
&= \frac{\rho c_p T}{\bar{p}}\frac{\partial\bar{p}}{\partial t} + \rho\sum_\alpha\left(h_{s,\alpha} - c_p T\frac{\overline{W}}{W_\alpha}\right)\frac{\partial Y_\alpha}{\partial t} + (h_s - c_p T)\frac{\partial\rho}{\partial t}.
\end{aligned}
\tag{C.1}
$$

The time derivative of the mass fractions, which originates from the species transport equation, is

$$
\frac{\partial Y_\alpha}{\partial t} = \frac{1}{\rho}\left[\dot{m}_\alpha''' - \nabla\cdot\mathbf{J}_\alpha - Y_\alpha\frac{\partial\rho}{\partial t} - \nabla\cdot(\rho Y_\alpha\mathbf{u})\right].
\tag{C.2}
$$

Using (C.2) in (C.1) and summing over species to eliminate the density time derivative we obtain

$$
\frac{\partial(\rho h_s)}{\partial t} = \frac{\rho c_p T}{\bar{p}}\frac{\partial\bar{p}}{\partial t} + \sum_\alpha\left(h_{s,\alpha} - c_p T\frac{\overline{W}}{W_\alpha}\right)\left[\dot{m}_\alpha''' - \nabla\cdot\mathbf{J}_\alpha - \mathbf{u}\cdot\nabla(\rho Y_\alpha) - \rho Y_\alpha\nabla\cdot\mathbf{u}\right].
\tag{C.3}
$$

Finally… plugging (C.3) into (16) yields (17):

$$
\begin{aligned}
\nabla\cdot\mathbf{u} = {} & \frac{1}{\rho c_p T}\frac{\mathrm{D}\bar{p}}{\mathrm{D}t} - \frac{1}{\bar{p}}\frac{\partial\bar{p}}{\partial t}\\
& + \frac{1}{\rho c_p T}\left[\dot{q}''' - \nabla\cdot\dot{\mathbf{q}}'' - \mathbf{u}\cdot\nabla(\rho h_s)\right]\\
& + \frac{1}{\rho}\sum_\alpha\left(\frac{\overline{W}}{W_\alpha} - \frac{h_{s,\alpha}}{c_p T}\right)\left[\dot{m}_\alpha''' - \nabla\cdot\mathbf{J}_\alpha - \mathbf{u}\cdot\nabla(\rho Y_\alpha)\right].
\end{aligned}
\tag{C.4}
$$

Appendix D. Manufactured solution source terms

```
#!/usr/bin/python
#McDermott
#2013-08-20

# L. Shunn, F. Ham, P. Moin, Verification of variable-density flow solvers using manufactured
#    solutions, J. Comput. Phys. 231 (2012) 3801--3827.
#
# Problem 3
```

```
from __future__ import division # make floating point division default, e.g., 1/2=0.5
from mpmath import *
from sympy import *

init_printing(use_unicode=True)

x,y,z,t,r,r0,r1,k,w,uf,vf,f,g,D,p,mu = symbols('x y z t r r0 r1 k w uf vf f g D p mu')

z = ( 1 + sin(pi*k*(x-uf*t))*sin(pi*k*(y-vf*t))*cos(pi*w*t) )/( (1+r0/r1) + \
    (1-r0/r1)*sin(pi*k*(x-uf*t))*sin(pi*k*(y-vf*t))*cos(pi*w*t) ) # mixture fraction
r = 1/( z/r1 + (1-z)/r0 ) # density
u = uf + (r1-r0)/r*(-w/(4*k))*cos(pi*k*(x-uf*t))*sin(pi*k*(y-vf*t))*sin(pi*w*t) # velocity x
v = vf + (r1-r0)/r*(-w/(4*k))*sin(pi*k*(x-uf*t))*cos(pi*k*(y-vf*t))*sin(pi*w*t) # velocity y
p = r*u*v/2 # pressure

f = r*u # mass flux x
g = r*v # mass flux y

Dif = D*(diff(diff(z,x),x)+diff(diff(z,y),y)) # diffusion term in scalar equation
Div = simplify( diff(u,x)+diff(v,y) ) # divergence from velocity solution
T12 = simplify( mu*(diff(u,y)+diff(v,x)) ) # stress tensor components
T11 = simplify( 2*mu*(diff(u,x)-Div/3) )
T22 = simplify( 2*mu*(diff(v,y)-Div/3) )

Q_r = simplify( diff(r,t)+diff(f,x)+diff(g,y) ) # continuity, Q_r=0
Q_z = diff(r*z,t)+diff(f*z,x)+diff(g*z,y)-Dif # mixture fraction
Q_u = diff(r*u,t)+diff(r*u*u,x)+diff(r*u*v,y)+diff(p,x)-(diff(T11,x)+diff(T12,y)) # u momentum
Q_v = diff(r*v,t)+diff(r*v*u,x)+diff(r*v*v,y)+diff(p,y)-(diff(T12,x)+diff(T22,y)) # v momentum

Div_EOS = simplify( (1/r1 - 1/r0)*(Dif + Q_z) ) # divergence from EOS

DD = simplify(Div-Div_EOS) # check EOS divergence formula, DD=0

# print output to screen
print 'Div = ' + str(Div)
print 'Div_EOS = ' + str(Div_EOS)
print 'Div - Div_EOS = ' + str(DD)
print 'Q_r = ' + str(Q_r)
print 'Q_z = ' + str(Q_z)
print 'Q_u = ' + str(Q_u)
print 'Q_v = ' + str(Q_v)

# also write to file
foo=open('shunn_mms.out','w')

foo.write('Div = ' + str(Div) + '\n')
foo.write('\n')
foo.write('Div_EOS = ' + str(Div_EOS) + '\n')
foo.write('\n')
foo.write('DD = ' + str(DD) + '\n')
foo.write('\n')
foo.write('Q_r = ' + str(Q_r) + '\n')
foo.write('\n')
foo.write('Q_z = ' + str(Q_z) + '\n')
foo.write('\n')
foo.write('Q_u = ' + str(Q_u) + '\n')
foo.write('\n')
foo.write('Q_v = ' + str(Q_v) + '\n')

foo.close()
```

References

[1] R.G. Rehm, H.R. Baum, The equations of motion for thermally driven, buoyant flows, J. Res. NBS 83 (1978) 297–308, http://fire.nist.gov/bfrlpubs/fire78/PDF/f78002.pdf.

[2] A. Majda, J. Sethian, The derivation and numerical solution of the equations for zero Mach number combustion, Combust. Sci. Technol. 42 (1985) 185–205, http://dx.doi.org/10.1080/00102208508960376.
[3] B. Müller, Low-Mach-number asymptotics of the Navier–Stokes equations, J. Eng. Math. 34 (1998) 97–109, http://dx.doi.org/10.1023/A:1004349817404.
[4] J.B. Bell, AMR for low Mach number reacting flow, Tech. Rep. LBNL-54351, Lawrence Berkeley National Laboratory, Berkeley, California, 2004, https://ccse.lbl.gov/Publications/bell/amr2003.pdf.
[5] O. Desjardins, G. Blanquart, G. Balarac, H. Pitsch, High order conservative finite-difference schemes for variable density low-Mach number turbulent flows, J. Comput. Phys. 227 (15) (2008) 7125–7159, http://dx.doi.org/10.1016/j.jcp.2008.03.027.
[6] A. Kempf, Large-eddy simulation of non-premixed turbulent flames, Ph.D. thesis, Technische Universität Darmstadt, 2004.
[7] C.D. Pierce, Progress-variable approach for large-eddy simulation of turbulent combustion, Ph.D. thesis, Stanford University, 2001, http://ctr.stanford.edu/pdf/charles_pierce_thesis.pdf.
[8] J. Spinti, J. Thornock, E. Eddings, P. Smith, A. Sarofim, Heat transfer to objects in pool fires, in: M. Faghri, B. Senden (Eds.), Transport Phenomena in Fires, in: Dev. Heat Transf., vol. 20, WIT Press, Southampton, UK, 2008.
[9] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, K. Overholt, Fire Dynamics Simulator, Technical Reference Guide, Mathematical Model, vol. 1, National Institute of Standards and Technology, Gaithersburg, Maryland, USA, 2013, and VTT Technical Research, Centre of Finland, Espoo, Finland, sixth edn., https://docs.google.com/folder/d/0B_wB1pJL2bFQaDJaOFNnUDR4LXM/edit?docId=0B_wB1pJL2bFQUEVMX3hGQlNWVDQ.
[10] K. McGrattan, R. McDermott, J. Floyd, S. Hostikka, G. Forney, H. Baum, Computational fluid dynamics modeling of fire, Int. J. Comput. Fluid Dyn. 26 (2012) 349–361, http://dx.doi.org/10.1080/10618562.2012.659663.
[11] A.S. Almgren, J.B. Bell, W.G. Szymczak, A numerical method for the incompressible Navier–Stokes equations based on an approximate projection, SIAM J. Sci. Comput. 17 (2) (1996) 358–369, http://dx.doi.org/10.1137/S1064827593244213.
[12] S. Armfield, R. Street, An analysis and comparison of the time accuracy of fractional-step methods for Navier–Stokes equations on staggered grids, Int. J. Numer. Methods Fluids 38 (2001) 255–282, http://dx.doi.org/10.1002/fld.217.
[13] R.J. LeVeque, Numerical Methods for Conservation Laws, Springer, Basel AG, 1992.
[14] B. Thornber, D. Drikakis, R.J.R. Williams, D. Youngs, On entropy generation and dissipation of kinetic energy in high-resolution shock-capturing schemes, J. Comput. Phys. 227 (2008) 4853–4872, http://dx.doi.org/10.1016/j.jcp.2007.06.030.
[15] B. Thornber, A. Mosedale, D. Drikakis, D. Youngs, R.J.R. Willams, An improved reconstruction method for compressible flows with low Mach number features, J. Comput. Phys. 227 (2008) 4873–4894, http://dx.doi.org/10.1016/j.jcp.2007.01.036.
[16] P.T. Tsoutsanis, A.F. Antoniadis, D. Drikakis, WENO schemes on arbitrary unstructured meshes for laminar, transitional and turbulent flows, J. Comput. Phys. 256 (2014) 254–276, http://dx.doi.org/10.1016/j.jcp.2013.09.002.
[17] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction, second edn., Springer, 1999.
[18] R. Mittal, P. Moin, Suitability of upwind-biased finite difference schemes for large-eddy simulation of turbulent flows, AIAA J. 35 (1997) 1415–1417, http://dx.doi.org/10.2514/2.253.
[19] Y. Morinishi, T.S. Lund, O.V. Vasilyev, P. Moin, Fully conservative high order finite difference schemes for incompressible flow, J. Comput. Phys. 143 (1998) 90–124, http://dx.doi.org/10.1006/jcph.1998.5962.
[20] F. Nicoud, Conservative high-order finite-difference schemes for low-Mach number flows, J. Comput. Phys. 158 (2000) 71–97, http://dx.doi.org/10.1006/jcph.1999.6408.
[21] F. Ham, F.S. Lien, A.B. Strong, A fully conservative second-order finite-difference scheme for incompressible flows on nonuniform grids, J. Comput. Phys. 177 (2002) 117–133, http://dx.doi.org/10.1006/jcph.2002.7006.
[22] J.W. Deardorff, Stratocumulus-capped mixed layers derived from a three-dimensional model, Bound.-Layer Meteorol. 18 (1980) 495–527, http://dx.doi.org/10.1007/BF00119502.
[23] S.B. Pope, Turbulent Flows, Cambridge University Press, 2000.
[24] B.F. Magnussen, B.H. Hjertager, On mathematical modeling of turbulent combustion with special emphasis on soot formation and combustion, in: Proceedings of the Sixteenth Symposium (International) on Combustion, Combustion Institute, Pittsburgh, Pennsylvania, 1977, pp. 719–729, http://dx.doi.org/10.1016/S0082-0784(77)80366-4.
[25] R. McDermott, K. McGrattan, J. Floyd, A simple reaction time scale for under-resolved fire dynamics, in: 10th International Association of Fire Safety Science Symposium, University of Maryland, 2011, pp. 809–820, http://dx.doi.org/10.3801/IAFSS.FSS.10-809.
[26] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface, Phys. Fluids 8 (1965) 2182, http://dx.doi.org/10.1063/1.1761178.
[27] R.A. Sweet, Direct methods for the solution of Poisson's equation on a staggered grid, J. Comput. Phys. 12 (1973) 422–428, http://dx.doi.org/10.1016/0021-9991(73)90164-2.
[28] S. Gottlieb, C. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Rev. 43 (1) (2001) 89–112, http://dx.doi.org/10.1137/S003614450036757X.
[29] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35–60, http://dx.doi.org/10.1006/jcph.2000.6484.
[30] J.P. Boris, F.F. Grinstein, E.S. Oran, R.L. Kolbe, New insights into large eddy simulation, Fluid Dyn. Res. 10 (1992) 199–228, http://dx.doi.org/10.1016/0169-5983(92)90023-P.
[31] P.L. Roe, Characteristics-based schemes for the Euler equations, Annu. Rev. Fluid Mech. 18 (1986) 337, http://dx.doi.org/10.1146/annurev.fl.18.010186.002005.
[32] G. Zhou, Numerical simulations of physical discontinuities in single and multi-fluid flows for arbitrary Mach numbers, Ph.D. thesis, Chalmers Univ. of Tech., Goteborg, Sweden, 1995.
[33] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, K. Overholt, Fire Dynamics Simulator, Technical Reference Guide, Verification, vol. 2, National Institute of Standards and Technology, Gaithersburg, Maryland, USA, 2013, and VTT Technical Research, Centre of Finland, Espoo, Finland, sixth edn., https://docs.google.com/folder/d/0B_wB1pJL2bFQaDJaOFNnUDR4LXM/edit?docId=0B_wB1pJL2bFQVDdfY3ZkZnh0Rms.
[34] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, K. Overholt, Fire Dynamics Simulator, Technical Reference Guide, Validation, vol. 3, National Institute of Standards and Technology, Gaithersburg, Maryland, USA, 2013, and VTT Technical Research, Centre of Finland, Espoo, Finland, sixth edn., https://docs.google.com/folder/d/0B_wB1pJL2bFQaDJaOFNnUDR4LXM/edit?docId=0B_wB1pJL2bFQVDJNOXFPanVxMDA.
[35] L. Shunn, F. Ham, P. Moin, Verification of variable-density flow solvers using manufactured solutions, J. Comput. Phys. 231 (2012) 3801–3827, http://dx.doi.org/10.1016/j.jcp.2012.01.027.
[36] M.W. Chase Jr., NIST-JANAF thermochemical tables, fourth edition, J. Phys. Chem. Ref. Data Monograph 9 (1998), http://www.nist.gov/data/PDFfiles/jpcrdM9.pdf, http://kinetics.nist.gov/janaf.
[37] M. Germano, U. Piomelli, P. Moin, W. Cabot, A dynamic subgrid-scale eddy viscosity model, Phys. Fluids A 3 (7) (1991) 1760–1765, http://link.aip.org/link/doi/10.1063/1.857955.
[38] P. Moin, K. Squires, W. Cabot, S. Lee, A dynamic subgrid-scale model for compressible turbulence and scalar transport, Phys. Fluids A 3 (11) (1991) 2746–2757, http://link.aip.org/link/doi/10.1063/1.858164.
[39] J.H. Ferziger, M. Perić, Computational Methods for Fluid Dynamics, third edn., Springer, 2002.
[40] B.P. Leonard, A stable and accurate convection modelling procedure based on quadratic upstream interpolation, Comput. Methods Appl. Mech. Eng. 19 (1979) 59–98, http://dx.doi.org/10.1016/0045-7825(79)90034-3.