A.2 27

A PROJECTION METHOD FOR LOW MACH NUMBER FAST CHEMISTRY REACTING FLOW

James Hilditch * and Phillip Colella †
Department of Mechanical Engineering
University of California
Berkeley, CA 94720

Abstract

We present a numerical technique for treating unsteady, low Mach number reacting flow. For fast chemistry nonpremixed combustion, the reaction effects are generated using a conserved scalar, the mixture fraction. Such an approach yields species concentration information via post-processing rather than through the integration of species equations. Limits on the range of the conserved scalar are enforced using a conservative redistribution of overshoots rather than through traditional slope limiting methods. An approximate projection method is employed to enforce the non-zero divergence constraint. The method is second order in space and time. We examine a gaussian conserved scalar field advected in a developed axisymmetric pipe flow.

1 Introduction

Nonpremixed flames are found in many practical combustion systems. In these systems, fuel and oxidant are initially separated and the combustion process depends on effectively mixing the two reactant streams. In cases where the chemical reaction rate is much higher than the mixing rate, an interesting limiting case exists. It is here that we can more clearly examine the fluid mechanics in the presence of large density variations due to the expanding gases.

Numerical studies of combustion systems at the fast chemistry limit are aided by the use of a conserved scalar field to provide information about the reaction progress [14] [17]. In this approach, the system of differential equations that describe the full reacting system can be greatly simplified. Assuming equal mass diffusivities for all species, for example, results in the collapse of the numerous species evolution equations to a single evolution equation for a conserved scalar such as the mixture fraction. Species concentrations can be generated at any time through knowledge of the conserved scalar without the need for detailed integration of each species evolution equation.

Furthermore, for flows where the thermal diffusivity is equal to the mass diffusivity, the energy equation can be related to the conserved scalar. Here, temperature is known as a function of the mixture fraction. The advantage is greater for turbulent flows as the computational cost of solving for reaction rates becomes more prohibitive. Such an approach has been utilized successfully in a number of areas including studies of transition to turbulence in reacting jets [12].

For many low Mach number nonpremixed systems the effects of acoustic waves can be neglected. Examples include boilers, furnaces, jets and fires. Utilizing a mathematical model that removes the effect allows computation to proceed at time steps limited by the bulk flow of the gas rather than the much more restrictive time step of pressure waves [8] [11].

The aim of the present research is to develop a finite difference method for solving time dependent nonpremixed reacting systems accurately and efficiently. By formalizing the method, we provide the foundation for future investigations of more complex problems. The low Mach number equations are solved using an approximate projection method developed for premixed reacting flows [7]. A conserved scalar, the mixture fraction, is used to model chemical reaction effects. We allow for temperature dependent viscosity and diffussion coefficients while computing at unity Lewis number, $Le = \alpha/D \equiv 1$. The accuracy of the method is demonstrated by examining the burning of a gaseous droplet in an axisymmetric pipe flow.

*Ph.D. Candidate, Member AIAA
†Professor, Member AIAA

Copyright © 1997 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

2 Low Mach Number Model

The system of equations treated in this work is based on a low Mach number combustion model. For low speed flows, $M < 0.3$, one can perform asymptotic expansions, in M, of the primitive variables, U, T, ρ, and P. The result of such an analysis is a system in which the effect of acoustic waves can be neglected for open domains. The pressure can then be expressed as the sum of a spatially and temporally constant part P_{amb} and a dynamic part π,

$$P(r, z, t) = P_{amb} + \pi(r, z, t), \tag{2.1}$$

where $\pi/P_{amb} = O(M^2)$. The momentum equation can be expressed as

$$\rho(U_t + (U \cdot \nabla)U)) = -\nabla\pi + \nabla\cdot\tau. \tag{2.2}$$

We can obtain a constraint on the divergence of the velocity field from the continuity equation:

$$\nabla\cdot U = -\tfrac{1}{\rho}\tfrac{D\rho}{Dt} \equiv S. \tag{2.3}$$

3 Conserved Scalar Chemistry

It has been noted that for many non-premixed combustion applications, chemical effects can be well modeled through knowledge of a conserved scalar field. In particular, if it is assumed that the molecular diffusivities are equal, a significant simplification can be made in the species and energy evolution equations. For example, consider a system on which we make the following assumptions:

(1) Soret and Dufour effects can be neglected.

(2) Molecular diffusivities are equal, $\mathcal{D}_i = \mathcal{D}$.

(3) Radiation loss can be neglected.

(4) Body forces can be neglected.

We can derive an evolution equation for a conserved scalar, f, that can be used to calculate reaction rates for use in the energy equation and the divergence constraint.

$$\rho\tfrac{Df}{Dt} = \nabla\cdot(\rho\mathcal{D}\nabla f). \tag{3.1}$$

For a two stream mixing problem, such as a reacting jet, a useful conserved scalar to consider is the mixture fraction:

$$f = \frac{\beta - \beta_{ox}}{\beta_{fuel} - \beta_{ox}} \tag{3.2}$$

where β is any extensive property of the mixture free of sources and sinks.

Assuming equilibrium chemistry, we can determine species mass fractions from knowledge of the conserved scalar field, $Y_i = Y_{i,eq}(f)$. Further, we can obtain the reaction rate by applying the chain rule to the species equation:

$$\begin{aligned} \rho\dot{Y}_i\frac{Df}{Dt} &= \omega_i + \dot{Y}_i\nabla\cdot(\rho\mathcal{D}\nabla f) \\ &\quad + \rho\mathcal{D}(\nabla f\cdot\nabla f)\ddot{Y}_i. \end{aligned} \tag{3.3}$$

This reduces to

$$\omega_i = -\rho\mathcal{D}(\nabla f\cdot\nabla f)\ddot{Y}_{i,eq}(f). \tag{3.4}$$

For an ideal gas, the temperature, density and divergence constraint can now be expressed:

$$\begin{aligned} \rho c_p\frac{DT}{Dt} &= \nabla\cdot(\kappa\nabla T) - \sum_{i=1}^{N}\omega_i\Delta h^o_{f,i} \\ \rho &= \frac{P_{amb}}{RT} \\ \nabla\cdot U &= \frac{\gamma-1}{\gamma P_{amb}}(\nabla\cdot(\kappa\nabla T) - \sum_{i=1}^{N}\omega_i\Delta h^o_{f,i}) \\ &\quad + \frac{R}{\dot{R}}\nabla\cdot(\rho\mathcal{D}\nabla f). \end{aligned} \tag{3.5}$$

The ideal gas constant is a function of mixture fraction only, $R = R(f)$, and the transport coefficients, κ and $\mathcal{D}$, and the specific heat, c_p, are functions of mixture fraction and temperature. Furthermore, since each species concentration is a function of mixture fraction only, the detailed chemical composition can be determined as a post-processing step.

For the purpose of demonstrating the salient features of the proposed computational method, we simplify the system further by making the additional assumptions:

(1) The thermal conductivity and diffusivities are equal, i.e. $Le = 1$.

(2) Viscosity, μ, and the species diffusion coefficient, $\mathcal{D}$, are related to temperature as [19]:

$$\begin{aligned} \mu &= \mu_{ref}(\tfrac{T}{T_{ref}})^{0.7} \\ \mathcal{D} &= \mathcal{D}_{ref}(\tfrac{T}{T_{ref}})^{1.7}. \end{aligned} \tag{3.6}$$

In this case, the evolution equations for species and enthalpy are identical allowing us to express temperature and density as functions of mixture fraction only. This results in the following system of equations:

$$
\begin{aligned}
\rho(U_t + (U\cdot\nabla)U) &= -\nabla\pi + \nabla\cdot\tau \\
\nabla\cdot U &= \frac{1}{\rho}\left(\frac{\dot{T}}{T} + \frac{\dot{R}}{R}\right)\nabla\cdot(\rho\mathcal{D}\nabla f) \\
f_t + (U\cdot\nabla)f &= \frac{1}{\rho}\nabla\cdot(\rho\mathcal{D}\nabla f) \\
T = T_{eq}(f); \rho &= \rho_{eq}(f)
\end{aligned} \tag{3.7}
$$

4 Numerical Algorithm

In this section, we present the details of the numerical algorithm for solving the above equations. Computations take place on a uniform grid of rectangular cells of dimension Δr by Δz. These cells are indexed by $i, [0...M]$ in the r direction and $j, [0...N]$ in the z direction. We assume that all variables are cell-centered except where explicitly noted.

The strategy of the method in advancing the solution from time t^n to time $t^n + \Delta t = t^{n+1}$ is a predictor-corrector. We solve evolution equations of the form: $\varphi_t + (U\cdot\nabla)\varphi = L(\varphi) + Q$ for $\varphi = U$, f, where $L(\varphi)$ is a diffusive operator and Q is some source term.

In the predictor, we estimate values of the flow quantities at $t^{n+\frac{1}{2}}$ for use in constructing the nonlinear convective derivatives. An explicit second-order Godunov method given by [2], [5], and [10] is used for this purpose. We use implicit Crank-Nicolson differencing to evaluate the mixture fraction and to estimate the new velocity field:

$$
\begin{aligned}
\varphi^* &= \varphi^n + \Delta t\Big(-[(U\cdot\nabla)\varphi]^{n+\frac{1}{2}} \\
&+ \frac{1}{2}\left(L(\varphi^n) + L(\varphi^*)\right) + Q^{old}\Big).
\end{aligned} \tag{4.1}
$$

Q^{old} is an estimate of Q obtained from previous time step data.

In the corrector, we improve our velocity field estimate by employing time centered values of the source terms in an approximate projection to satisfy the divergence constraint. Conceptually, this can be written:

$$
\begin{aligned}
\varphi^{n+1} &= \varphi^n + \Delta t\Big(-[(U\cdot\nabla)\varphi]^{n+\frac{1}{2}} \\
&+ \frac{1}{2}\left(L(\varphi^n) + L(\varphi^*)\right) + Q^{n+\frac{1}{2}}\Big).
\end{aligned} \tag{4.2}
$$

The source term, $Q^{n+\frac{1}{2}}$, is a time centered estimate obtained using φ^n and φ^*. Finally, the velocity field is filtered to remove nonphysical modes left by the projection. In the remainder of this section we provide the details for the above outline.

4.1 Diffusion Operators

Diffusion operators need to be defined in order to advance the flow quantities in time. For velocity, we need to approximate the axisymmetric viscous stress tensor:

$$
\begin{aligned}
(\nabla\cdot\tau)^r &= \frac{1}{r}\frac{\partial}{\partial r}\left[r\mu\left(2\frac{\partial u}{\partial r}\right)\right] - \frac{2\mu u}{r^2} + \frac{2}{3r}\mu(\nabla\cdot U) \\
&\quad -\frac{2}{3r}\frac{\partial}{\partial r}(r\mu(\nabla\cdot U)) \\
&\quad +\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial v}{\partial r} + \frac{\partial u}{\partial z}\right)\right] \\
(\nabla\cdot\tau)^z &= \frac{1}{r}\frac{\partial}{\partial r}\left[r\mu\left(\frac{\partial v}{\partial r} + \frac{\partial u}{\partial z}\right)\right] + \frac{\partial}{\partial z}\left[\mu\left(2\frac{\partial v}{\partial z}\right)\right] \\
&\quad -\frac{2}{3}\frac{\partial}{\partial z}(\mu(\nabla\cdot U)).
\end{aligned} \tag{4.3}
$$

We construct this operator by defining gradients and a divergence in the following way:

$$
\begin{aligned}
(G^r\varphi)_{i+\frac{1}{2},j} &= \frac{\varphi_{i+1,j} - \varphi_{i,j}}{\Delta r} \\
(G^z\varphi)_{i,j+\frac{1}{2}} &= \frac{\varphi_{i,j+1} - \varphi_{i,j}}{\Delta z} \\
(G^r_t\varphi)_{i,j+\frac{1}{2}} &= \frac{\varphi_{i+1,j+1} - \varphi_{i-1,j+1} + \varphi_{i+1,j} - \varphi_{i-1,j}}{2\Delta r} \\
(G^z_t\varphi)_{i+\frac{1}{2},j} &= \frac{\varphi_{i+1,j+1} - \varphi_{i+1,j-1} + \varphi_{i,j+1} - \varphi_{i,j-1}}{2\Delta z} \\
D(F)_{i,j} &= \frac{2r_{i+\frac{1}{2},j}F^r_{i+\frac{1}{2},j} - 2r_{i-\frac{1}{2},j}F^r_{i-\frac{1}{2},j}}{r^2_{i+\frac{1}{2},j} - r^2_{i-\frac{1}{2},j}} \\
&\quad + \frac{F^z_{i,j+\frac{1}{2}} - F^z_{i,j-\frac{1}{2}}}{\Delta z}
\end{aligned} \tag{4.4}
$$

where $r_{i+\frac{1}{2},j}$ is the radial distance to the cell edge indexed by $(i+\frac{1}{2}, j)$. The viscous operator is formed:

$$
\begin{aligned}
(L^h_\mu U)^r_{i,j} &= D(\bar{F}^1) - \frac{2(\mu u)_{i,j}}{r^2_{i,j}} - \frac{2\mu_{i,j}}{3r_{i,j}}S_{i,j} \\
&\quad -\frac{2}{3}\frac{(rS)_{i+\frac{1}{2},j} - (rS)_{i-\frac{1}{2},j}}{r^2_{i+\frac{1}{2},j} - r^2_{i-\frac{1}{2},j}} \\
(L^h_\mu U)^z_{i,j} &= D(\bar{F}^2) - \frac{2}{3}\frac{S_{i,j+1} - S_{i,j-1}}{\Delta z}
\end{aligned} \tag{4.5}
$$

where

$$
\begin{aligned}
F^1_{i+\frac{1}{2},j} &= 2\mu G^r(u) \\
F^1_{i,j+\frac{1}{2}} &= \mu\Big(G^r_t(v) + G^z(u)\Big) \\
F^2_{i+\frac{1}{2},j} &= \mu\Big(G^r(v) + G^z_t(u)\Big) \\
F^2_{i,j+\frac{1}{2}} &= 2\mu G^z(v).
\end{aligned} \tag{4.6}
$$

The flux functions, $\bar{F}^1$ and $\bar{F}^2$, are edge centered vectors with individual components defined above. Where needed, μ is averaged to cell edges.

Boundary conditions are required for the viscous operator. Along the axis of symmetry and at outflow , a homogeneous Neumann condition on the velocity field is set using a copy condition , e.g. $\varphi_{-1} = \varphi_0$. At walls, a first-order no-slip condition is enforced on the gradients where necessary:

$$\begin{aligned}
(G^r\varphi)_{M+\frac{1}{2},j} &= \frac{1}{\Delta r}(-3\varphi_{M,j} + \frac{1}{3}\varphi_{M-1,j}) \\
(G^r_t v)_{M,j+\frac{1}{2}} &= \frac{1}{2\Delta r}(-v_{M,j+1} + \frac{1}{3}v_{M-1,j+1} \\
&\quad -v_{M,j} + \frac{1}{3}v_{M-1,j}) \\
(G^z_t u)_{M+\frac{1}{2},j} &= 0 \qquad (4.7)
\end{aligned}$$

for $\varphi = u, v$. The inflow is set using the prescribed inflow condition.

The diffusion operator for the mixture fraction is computed:

$$\begin{aligned}
(F^{f,r}f)_{i+\frac{1}{2},j} &= (\rho\mathcal{D}G^r f)_{i+\frac{1}{2},j} \\
(F^{f,z}f)_{i,j+\frac{1}{2}} &= (\rho\mathcal{D}G^z f)_{i,j+\frac{1}{2}} \\
(L^h_{\mathcal{D}}f)_{i,j} &= (DFf)_{i,j}. \qquad (4.8)
\end{aligned}$$

We obtain ρ on cell edges by harmonic averaging of the cell centered density. Boundary conditions are set using a copy condition on f for solid walls, the outflow boundary and the axis of symmetry. The inflow boundary is set using the prescribed profile.

4.2 Reaction Term

For cases where $Le = 1$, the reaction term is obtained through knowledge of the mixture fraction field:

$$(\nabla\cdot U)_{i,j} = \frac{1}{\rho}(\frac{\dot{T}}{T} + \frac{\dot{R}}{R})(L^h_{\mathcal{D}}f)_{i,j} \equiv S^n_{i,j} \qquad (4.9)$$

where T, $\dot{T}$, R, and $\dot{R}$ are functions of the mixture fraction alone and are centered at (i, j).

4.3 Predictor

4.3.1 Edge Velocities

The edge velocities, $(u,v)^{n+\frac{1}{2}}_{i+\frac{1}{2},j} \equiv U^{n+\frac{1}{2}}_{i+\frac{1}{2},j}$ and $(u,v)^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} \equiv U^{n+\frac{1}{2}}_{i,j+\frac{1}{2}}$, are computed using a variant of the higher order Godunov procedure in [5]. The procedure consists of two steps:

(1) Extrapolate in time and space to obtain $U^{n+\frac{1}{2},L}_{i+\frac{1}{2},j}$ and $U^{n+\frac{1}{2},R}_{i+\frac{1}{2},j}$ at all r-cell edges and $U^{n+\frac{1}{2},B}_{i,j+\frac{1}{2}}$ and $U^{n+\frac{1}{2},T}_{i,j+\frac{1}{2}}$ at all z-cell edges.

(2) Use an upwinding procedure to uniquely determine $U^{n+\frac{1}{2}}_{i+\frac{1}{2},j}$ at r-cell edges and $U^{n+\frac{1}{2}}_{i,j+\frac{1}{2}}$ at z-cell edges.

In the extrapolation step, we use a Taylor series to approximate cell edge values at $t^{n+\frac{1}{2}}$. The temporal derivative is replaced using the momentum equation. For edge $(i+\frac{1}{2}, j)$ we get,

$$\begin{aligned}
U^{n+\frac{1}{2},L}_{i+\frac{1}{2},j} &= U^n_{i,j} \\
&+ \frac{1}{2}(1 - max(u^n_{i,j}, 0)\frac{\Delta t}{\Delta r})\delta U_{r,i,j} \\
&- \frac{\Delta t}{2}(v[U])_{z,i,j}\frac{1}{\Delta z} \\
&+ \frac{\Delta t}{2\rho_{i,j}}(L^h_\mu U)_{i,j} \\
U^{n+\frac{1}{2},R}_{i+\frac{1}{2},j} &= U^n_{i,j} \\
&+ \frac{1}{2}(-1 - min(u^n_{i,j}, 0)\frac{\Delta t}{\Delta r})\delta U_{r,i+1,j} \\
&- \frac{\Delta t}{2}(v[U])_{z,i+1,j}\frac{1}{\Delta z} \\
&+ \frac{\Delta t}{2\rho_{i+1,j}}(L^h_\mu U)_{i+1,j} \qquad (4.10)
\end{aligned}$$

We define values at other cell edges in a similar manner. For edge cells where either the left or right extrapolation is undefined, we apply boundary conditions. For $(i+\frac{1}{2}, j)$ cells, the r-component of velocity is set to zero. The z-component of velocity is determined from the properly defined edge value, e.g. the left extrapolation is defined at the right edge of the domain. For $(i, j+\frac{1}{2})$ cells, the prescribed profile is set for both components of velocity at the inflow edge. The properly defined edge value is used at outflow.

The slopes, $\delta U_{r,i,j}$, are computed using a standard centered difference stencil. Where needed, we set boundary conditions on the velocity field. For u, we use a homogeneous Dirichlet condition at the axis of symmetry and walls. At inflow and outflow, we use linear extrapolation, i.e. $u_{i,-1} = 2u_{i,0} - u_{i,1}$. For v, we apply a homogeneous Dirichlet condition at walls and a homogeneous Neumann condition at the axis of symmetry. At inflow we apply the prescribed inflow profile. At outflow we use linear extrapolation.

The transverse derivatives are defined as in [10]:

$$[U]_{z,i+1,j} = \begin{cases} \hat{U_C} - \hat{U_L} & \text{if } v^n_{i,j} > 0 \\ \hat{U_R} - \hat{U_C} & \text{otherwise} \end{cases} \tag{4.11}$$

where

$$\begin{aligned} \hat{U_L} &= U_{i,j-1} + \frac{\Delta t}{2\rho_{i,j-1}}(L^h_\mu U)_{i,j-1} \\ \hat{U_C} &= U_{i,j} + \frac{\Delta t}{2\rho_{i,j}}(L^h_\mu U)_{i,j} \\ \hat{U_R} &= U_{i,j+1} + \frac{\Delta t}{2\rho_{i,j+1}}(L^h_\mu U)_{i,j+1}. \end{aligned} \tag{4.12}$$

Note that this procedure requires values for $L^h_\mu U$ that lie outside the physical domain of the problem. We want the boundary conditions for $L^h_\mu U$ to be related to those set on the velocity field, U. For $L^{h,r}_\mu(u)$, we copy values across the boundary for inflow and outflow, i.e. $(L^{h,r}_\mu u)_{i,-1} = (L^{h,r}_\mu u)_{i,0}$. At the axis of symmetry and walls, we apply an inverse copy: $(L^{h,r}_\mu u)_{-1,j} = -(L^{h,r}_\mu u)_{0,j}$. For $L^{h,z}_\mu(v)$, we copy values at the axis of symmetry and at outflow. We apply an inverse copy at inflow and at the wall.

We obtain $U^{n+\frac{1}{2}}_{i+\frac{1}{2},j}$ through the following upwinding procedure:

$$U^{n+\frac{1}{2}}_{i+\frac{1}{2},j} = \begin{cases} U^{n+\frac{1}{2},L}_{i+\frac{1}{2},j} & \text{if } (u^n_{i+1,j} + u^n_{i+1,j}) > 0 \\ U^{n+\frac{1}{2},R}_{i+\frac{1}{2},j} & \text{otherwise} \end{cases} \tag{4.13}$$

4.3.2 Projection of Edge Velocities

In this step we use a MAC projection to enforce the divergence constraint on the edge velocites [3].

We solve

$$\begin{aligned} (D\frac{1}{\rho^n}G\phi)_{i,j} &= (DU^{n+\frac{1}{2}})_{i,j} - S^n_{i,j} \\ &\equiv L^h_\rho \phi \end{aligned} \tag{4.14}$$

for ϕ, where $S^n_{i,j}$ is given by (4.9), D, G^r and G^z are given in (4.4).

A set of boundary conditions are required for the projection. At the axis of symmetry and the wall, homogeneous Neumann conditions are used on ϕ. The prescribed inflow profile is used at the inflow edge. The boundary condition at the outflow edge is found by applying a one-dimensional version of the differential equation at that edge. Solve

$$\begin{aligned} D^{1D}\tfrac{1}{\rho}G\phi^{1D} &= D^{1D}U^{n+\frac{1}{2}} - S \\ \phi^{1D} &= 0 \quad \text{at wall,} \end{aligned} \tag{4.15}$$

where D^{1D} is the divergence defined in (4.4) with all z-direction differences omitted. ϕ^{1D} is applied at the outflow edge in the same way as the prescribed inflow profile is set at the inflow edge. This has the effect of setting $\partial v/\partial z = 0$ while still allowing for the existence of velocity variations in the r-direction.

The velocities are corrected as

$$\begin{aligned} u^{n+\frac{1}{2}}_{i+\frac{1}{2},j} &:= u^{n+\frac{1}{2}}_{i+\frac{1}{2},j} - (G^{MAC}\phi)^r_{i+\frac{1}{2},j} \\ v^{n+\frac{1}{2}}_{i+\frac{1}{2},j} &:= v^{n+\frac{1}{2}}_{i+\frac{1}{2},j} - (G^{MAC}\phi)^z_{i+\frac{1}{2},j}. \end{aligned} \tag{4.16}$$

The MAC gradients are defined:

$$\begin{aligned} (G^{MAC}\phi)^r_{i+\frac{1}{2},j} &= \frac{\phi_{i+1,j} - \phi_{i,j}}{\Delta r} \\ (G^{MAC}\phi)^z_{i+\frac{1}{2},j} &= \frac{1}{4\Delta z}(\phi_{i+1,j+1} + \phi_{i,j+1} \\ &\quad -\phi_{i+1,j-1} - \phi_{i,j-1}) \end{aligned} \tag{4.17}$$

We define the velocity corrections and gradients similarly for $(i, j+\frac{1}{2})$ edges. Homogeneous Neumann boundary conditions are set on ϕ for inflow, the axis of symmetry and the wall. The condition found in (4.15) is used at the outflow boundary.

Finally, the radial velocity on $j = 1/2$ edges is extrapolated:

$$u^{n+\frac{1}{2}}_{i,1/2} = (4.u^{n+\frac{1}{2}}_{i,3/2} - u^{n+\frac{1}{2}}_{i,5/2})/3. \tag{4.18}$$

This correction insures that the problem is not overdetermined at inflow. That is, it may be inappropriate to specify both components of the velocity field and the divergence at a single point.

4.3.3 Edge Centered Mixture Fraction

The procedure for finding $f^{n+\frac{1}{2}}_{i+\frac{1}{2},j}$ and $f^{n+\frac{1}{2}}_{i,j+\frac{1}{2}}$ is similar to the calculation of edge centered velocities with two minor differences.

First, boundary conditions are set on the mixture fraction field, for the slope computation. Homogeneous Neumann conditions are applied at the axis of symmetry and walls. The prescribed profile is set at inflow. At outflow, linear extrapolation is used.

It should be noted that use of a centered difference slope without a limiter can result in values of the mixture fraction that exceed the mathematical limits of that variable, $f \in [0, 1]$. There are many limiters available in the literature that produce monotone profiles ([18] for a summary). Additionally, there are smooth limiters that produce second accurate solutions for the advected variable such as those suggested in [15] and [16].

The current problem requires a limiter that is suitably accurate, monotone, and smooth for calculations of both f and $L^h_{\mathcal{D}} f$. We were unable to develop a limiter that satisfied all of these criteria. Monotone limiters invariably involve some clipping of extrema that result in disastrous errors in the calculation of the second divided difference. As a result, the limits on the mixture fraction will be enforced using a smooth conservative redistribution scheme discussed later.

The second difference is in the upwinding procedure. Here we use the relevant edge velocity to choose an upwind state. At edges where the extrapolation is defined from only one direction, we use that value for the upwind state. For example at the right edge of the domain, the left extrapolation to the $(i+\frac{1}{2}, j)$ edge is used.

4.3.4 Computation of Advective terms

The final step in the predictor is the computation of the nonlinear convective derivatives. We obtain $(U \cdot \nabla \cdot)\varphi$ for $\varphi = u, v, f$ as:

$$\begin{aligned} [(U \cdot \nabla)\varphi]^{n+\frac{1}{2}} &= \qquad (4.19) \\ \frac{1}{2\Delta r}(u^{n+\frac{1}{2}}_{i+\frac{1}{2},j} &+ u^{n+\frac{1}{2}}_{i-\frac{1}{2},j})(\varphi^{n+\frac{1}{2}}_{i+\frac{1}{2},j} - \varphi^{n+\frac{1}{2}}_{i-\frac{1}{2},j}) \\ +\frac{1}{2\Delta z}(v^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} &+ v^{n+\frac{1}{2}}_{i,j-\frac{1}{2}})(\varphi^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} - \varphi^{n+\frac{1}{2}}_{i,j-\frac{1}{2}}) \end{aligned}$$

4.3.5 Scalar Update

An estimate of density and the source term, S, at time $t^{n+\frac{1}{2}}$ is needed. We advance the mixture fraction field to t^{n+1} using Crank-Nicolson differencing:

$$\begin{aligned} f^{n+1} &= f^n + \Delta t[-[U \cdot \nabla f]^{n+\frac{1}{2}} \\ &\quad + \frac{1}{\rho} L^h_{\mathcal{D}}(\frac{f^{n+1} + f^n}{2})]. \qquad (4.20) \end{aligned}$$

where the density is evaluated at t^n. Homogeneous Neumann conditions are applied at the axis of symmetry, wall, and outflow edges. The prescribed inflow profile is used at the inflow edge.

Due to the unlimited slopes in the advection routine, the new mixture fraction field may contain values, $f \notin [0, 1]$. We enforce the limits on the mixture fraction by redistributing the overshoot to the neighboring cells in a manner similar to that in [6].

For example, if $f_{i,j} > 1$, we define the overshoot, $\delta f \equiv f_{i,j} - 1$. Weights are calculated to determine how much of the overshoot to distribute in each cell:

$$\xi_{i+k,j+m} = max(0, 1 - f_{i+k,j+m}) \qquad (4.21)$$

for $k, m = -1, 0, 1$, where cell (i, j) is excluded. The overshoot is then redistributed as:

$$f_{i+k,j+m} = f_{i+k,j+m} + \frac{\xi_{i+k,j+m}\delta f}{\Sigma\xi} \qquad (4.22)$$

where the sum occurs over the values of k and m specified above.

With the new mixture fraction in bounds, an estimate for the mixture fraction at $t^{n+\frac{1}{2}}$ is found by averaging the old and new time values. Temperature, density, and the transport properties are found in a look-up table using the time centered mixture fraction. The divergence source term, $S^{n+\frac{1}{2}}$ is evaluated according to (4.9) using values centered at $t^{n+\frac{1}{2}}$.

4.4 Corrector

4.4.1 Time Centering of Advective Terms

The advective derivatives that were computed previously were formed using a lagged source term in the MAC Projection (4.14). In order to maintain second-order accuracy in time, we need to improve that calculation using a time-centered source:

$$L^h_\rho \phi = (DU^{n+\frac{1}{2}})_{i,j} - S^{n+\frac{1}{2}}_{i,j}. \qquad (4.23)$$

We complete the MAC projection and recompute the advective derivatives. Finally, these values and $\rho^{n+\frac{1}{2}}$ are used in the scalar update (4.20) to improve the estimate of the mixture fraction at t^{n+1}.

4.4.2 Velocity Update

The velocity field is updated using:

$$\begin{aligned} U^* &= U^n + \Delta t[-[U \cdot \nabla U]^{n+\frac{1}{2}} - \frac{1}{\rho^{n+\frac{1}{2}}} G\pi^{n-\frac{1}{2}} \\ &\quad + \frac{1}{\rho^{n+\frac{1}{2}}} L^h_\mu(\frac{U^* + U^n}{2})] \qquad (4.24) \end{aligned}$$

Physical boundary conditions are used for this update as described for computing $L^h_\mu U$.

4.4.3 Velocity Projection

The update performed in (4.24) uses a lagged pressure gradient. In order to achieve second-order accuracy in time, an expression for U^{n+1} is required with the pressure gradient properly centered in time.

$$\begin{aligned} U^{n+1} &= U^n + \Delta t[-[U \cdot \nabla U]^{n+\frac{1}{2}} - \frac{1}{\rho^{n+\frac{1}{2}}} G\pi^{n+\frac{1}{2}} \\ &\quad + \frac{1}{\rho^{n+\frac{1}{2}}} L^h_\mu(\frac{U^* + U^n}{2})] \qquad (4.25) \end{aligned}$$

We obtain the time centered pressure gradient by solving a Poisson equation for the pressure:

$$\begin{aligned} L_\rho^h \pi^{n+\frac{1}{2}} &= D_0\left(\frac{U^* - U^n}{\Delta t} + \frac{1}{\rho^{n+\frac{1}{2}}} G\pi^{n-\frac{1}{2}}\right) \\ &- \left(\frac{S^{n+1} - S^n}{\Delta t}\right) \\ D_0 U &= D(F(U)) \\ (F^r U)_{i+\frac{1}{2},j} &= \frac{u_{i,j} + u_{i+1,j}}{2} \\ (F^z U)_{i,j+\frac{1}{2}} &= \frac{v_{i,j} + v_{i,j+1}}{2} \end{aligned} \tag{4.26}$$

where D is defined in 4.4. Boundary conditions are similar to those applied in the MAC projection, (4.15), with an appropriately modified right hand side to the outflow boundary condition equation:

$$D^{1D}\frac{1}{\rho}G\pi^{1D} = D^{1D}\left(\frac{U^* - U^n}{\Delta t} + \frac{1}{\rho^{n+\frac{1}{2}}}G\pi^{n-\frac{1}{2}}\right) - \frac{S^{n+1} - S^n}{\Delta t} \tag{4.27}$$

The projected velocity is then found from:

$$\begin{aligned} U^{n+1} &= U^* - \frac{\Delta t}{\rho^{n+\frac{1}{2}}}(G_0\pi^{n-\frac{1}{2}} - G_0\pi^{n+\frac{1}{2}}) \\ G_0^r\varphi &= \frac{\varphi_{i+1,j} - \varphi_{i-1,j}}{\Delta r} \\ G_0^z\varphi &= \frac{\varphi_{i,j+1} - \varphi_{i,j-1}}{\Delta z}. \end{aligned} \tag{4.28}$$

A second order extrapolation is used to set boundary conditions on δ for computing the gradients:

$$\begin{aligned} \delta_{i,-1} &= 3(\delta_{i,0} - \delta_{i,1}) + \delta_{i,2} \\ \delta_{-1,j} &= 3(\delta_{0,j} - \delta_{1,j}) + \delta_{2,j} \end{aligned} \tag{4.29}$$

4.4.4 Filter

The need for a filter when using an approximate projection was recognized in [6] and [7]. Several modifications have been made since that work. For this reason, we have devoted a full section to discussion of the filter.

5 Filter

The approximate projection operator used in this paper allows a non-physical oscillatory error to remain after the projection: $U = \epsilon(-1)^{(i+j)}$. This mode can be removed from the velocity field by applying a filter to the output of the projection.

The filter consists of a series of point-Jacobi iterations using a divergence centered alternately on $(i+\frac{1}{2}, j)$ and $(i, j+\frac{1}{2})$. We isolate the approximately divergence free part of the velocity field by solving:

$$\begin{aligned} \Delta\Psi &= S \\ U_d &= U - G\Psi \end{aligned} \tag{5.1}$$

with homogeneous Neumann boundary conditions on all edges. We damp the oscillatory mode by performing several point relaxation passes (currently 8) on the velocity field with operators alternately defined on $(i+\frac{1}{2}, j)$ and $(i, j+\frac{1}{2})$ edges.

$$\begin{aligned} U_d &:= U_d + G_{ss}\lambda_{ss}D_{ss}(U_d) \\ U_d &:= U_d + G_{tb}\lambda_{tb}D_{tb}(U_d). \end{aligned} \tag{5.2}$$

The operators on $(i+\frac{1}{2}, j)$ edges, G_{ss}, D_{ss} are defined:

$$\begin{aligned} (G_{ss}^r\varphi)_{i,j} &= \frac{\varphi_{i+\frac{1}{2},j} - \varphi_{i-\frac{1}{2},j}}{\Delta r} \\ (G_{ss}^z\varphi)_{i,j} &= \frac{1}{4\Delta z}(\varphi_{i+\frac{1}{2},j+1} + \varphi_{i-\frac{1}{2},j+1} \\ &\quad -\varphi_{i+\frac{1}{2},j-1} - \varphi_{i-\frac{1}{2},j-1}) \\ (D_{ss}U)_{i+\frac{1}{2},j} &= \frac{(2ru)_{i+1,j} - (2ru)_{i,j}}{r_{i+1}^2 - r_i^2} \\ &\quad + \frac{v_{i,j+1} + v_{i+1,j+1} - v_{i,j-1} - v_{i+1,j-1}}{4\Delta z} \end{aligned} \tag{5.3}$$

G_{tb} and D_{tb} are defined:

$$\begin{aligned} (G_{tb}^r\varphi)_{i,j} &= \frac{1}{4\Delta r}(\varphi_{i+1,j+\frac{1}{2}} + \varphi_{i+1,j-\frac{1}{2}} \\ &\quad -\varphi_{i-1,j+\frac{1}{2}} - \varphi_{i-1,j-\frac{1}{2}}) \\ (G_{tb}^z\varphi)_{i,j} &= \frac{\varphi_{i,j+\frac{1}{2}} - \varphi_{i,j-\frac{1}{2}}}{\Delta z} \\ (D_{tb}U)_{i,j+\frac{1}{2}} &= \frac{1}{4(r_{i+1}^2 - r_{i-1}^2)} \\ &\quad *(2r_{i+1,j}(u_{i+1,j+1} + u_{i+1,j}) \\ &\quad -2r_{i-1,j}(u_{i-1,j+1} + u_{i-1,j})) \\ &\quad + \frac{v_{i,j+1} - v_{i,j}}{\Delta z} \end{aligned} \tag{5.4}$$

The relaxation parameters, λ_{ss} and λ_{tb}, in (5.2) are chosen to insure a stable update of the velocity field. While a traditional stability analysis cannot

be performed for axisymmetric geometry, the results for the rectangular grids are easily calculated:

$$\begin{aligned}(\lambda_{ss})_{i,j} &= -\frac{\Delta r^2}{4} \\ (\lambda_{tb})_{i,j} &= -\frac{\Delta z^2}{4}. \end{aligned} \tag{5.5}$$

These values appear to remain stable for all r.

The stencil for the filter requires values of D_{tb} that extend one cell beyond the physical boundaries in the r-direction and fall on the boundaries in the z-direction. Likewise, values of D_{ss} are needed in the cells outside the domain in the z-direction and on the boundaries in the r-direction. In order to avoid ambiguities in evaluating the divergence at inflow and outflow edges, we apply the filter for all (i, j) for $i \in [0..M], j \in [1..N-1]$.

First consider D_{tb}. For the cells inside the domain at the axis of symmetry and at the wall, we make use of the no flow boundary condition on u. Accordingly, the divergence is calculated near the axis, for example, by:

$$\begin{aligned}(D_{tb}U)_{0,j+\frac{1}{2}} &= \frac{r_{\frac{1}{2}} u_{\frac{1}{2},j+\frac{1}{2}} - 0}{4(r^2_{\frac{1}{2}} - 0)} \\ &+ \frac{v_{0,j+1} - v_{0,j}}{\Delta z} \\ u_{\frac{1}{2},j+\frac{1}{2}} &= u_{0,j+1} + u_{1,j+1} + u_{0,j} + u_{1,j} \end{aligned} \tag{5.6}$$

where we have taken advantage of the fact that no flux passes through $r = 0$.

Next, boundary conditions are applied to the divergence operator to get values outside the domain in the r-direction. The choice of these boundary conditions appears to be arbitrary. Best results are achieved when we simply copy values in the r-direction, $(D_{tb}U)_{-1,j+\frac{1}{2}} = (D_{tb}U)_{0,j+\frac{1}{2}}$.

Physical boundary conditions are also used to find D_{ss} on the boundary. At the axis of symmetry, we estimate the axial velocity at $r = 0$ using a linear extrapolation, $v_{-1/2,j} = 2v_{1/2,j} - v_{3/2,j}$. The divergence is calculated:

$$\begin{aligned}(D_{ss}U)_{-\frac{1}{2},j} &= \frac{2r_0 u_{0,j} - 0}{r_0^2} \\ &+ \frac{v_{-\frac{1}{2},j+1} - v_{-\frac{1}{2},j-1}}{2\Delta z^2}. \end{aligned} \tag{5.7}$$

At the wall, we set no-slip boundary conditions on the velocity field, $U_{M+1,j} = -U_{M,j}$, and apply the regular divergence stencil. At inflow, we copy the divergence from the interior where needed, $(D_{ss}U)_{i+1/2,1} = (D_{ss}U)_{i+1/2,2}$.

6 Numerical Results

6.1 Accuracy

We demonstrate the accuracy of this method by examining a gaseous droplet burning in an axisymmetric pipe flow. Equilibrium methane chemistry is used to generate derived quantities such as temperature and density. The physical domain is $0 \le r \le 1$, $0 \le z \le 4$. We avoid nonphysical effects caused by the presence of a source on the outflow boundary by centering a mixture fraction profile at $[0, 1.5]$:

$$f = 0.5 + 0.25e^{-12r^2}. \tag{6.1}$$

The mixture fraction takes on values, $0.5 \le f \le 0.75$, in order to avoid steep temperature gradients found near stoichiometric values, $f_{st} = 0.055$. For the grid sizes used in this study, these regions would be not be fully resolved resulting in a loss of accuracy.

The mixture fraction field is smoothed by advancing the solution without reaction to $t = 0.001$ using $\mathcal{D} = 1$. This helps to insure smoothness in the second derivative of the field, needed to estimate the source term.

We present results for both an inviscid flow and a $Re = 100$ flow. For the inviscid case, we use a plug flow velocity profile, $v = 1$, $u = 0$. For the viscous case, the initial velocity field is an axisymmetric pipe flow,

$$\begin{aligned} v &= 1 - r^2 \\ u &= 0. \end{aligned} \tag{6.2}$$

There is some difficulty generating an initial velocity field that is in agreement with the divergence constraint implied by the chemical reaction while still satisfying the viscous boundary condition at the wall. For this reason, the reaction is slowly switched on over $t = 0.3$. We multiply the right hand side of (4.9) by a factor:

$$\Gamma = \frac{1}{2}(tanh(32(t - 0.225)) + 1). \tag{6.3}$$

This allows the output of the inviscid projection to account for viscous effects at the outer wall. Currently, we are examining a $Le = 1$, $Sc = 1$, flow with variable viscosity and diffusivity. The flow is computed to $t = 0.5$.

A convergence rate can be found by estimating the error on successively refined grids. Thus, we compute

$$\frac{\left\|\varphi^{avg}_{h/2} - \varphi_h\right\|_{L_1}}{\left\|\varphi_h\right\|_{L_1}} \tag{6.4}$$

Case	16-32	Rate	32-64	Rate	64-128
u	5.26e-3	1.86	1.45e-3	1.61	4.75e-4
v	1.92e-6	1.60	6.33e-7	1.58	2.11e-7
f	2.58e-5	1.98	6.54e-6	1.93	1.71e-6
S	9.92e-3	1.95	2.56e-3	1.93	6.70e-4
T	8.71e-6	1.98	2.20e-6	1.94	5.75e-7

Table 1: L_1 Convergence rates - Euler, $t = 0.5$

Case	16-32	Rate	32-64	Rate	64-128
u	2.37e-1	2.09	5.55e-2	2.06	1.38e-2
v	6.03e-4	2.17	1.34e-4	2.03	3.28e-5
f	1.01e-3	1.99	2.54e-4	2.00	6.35e-5
S	5.04e-1	1.99	1.27e-1	2.00	3.18e-2
T	3.74e-4	2.00	9.38e-5	2.00	2.34e-5

Table 2: L_1 Convergence rates - $Re = 100$, $t = 0.5$

for $\varphi = u, v, f, T$ on each grid. Asymptotically, these differences are proportional to the error on the coarser grid. These values, as well as the numerical rates of convergence are presented in Tables 1 and 2.

While the convergence rates for the viscous case are second order, the rates for the inviscid test show difficulties with the velocity field. Better results are generated when using a pressure correction form of the projection for inviscid problems. In this formulation, we replace equation (4.26) with:

$$\begin{aligned} L_\rho^h \delta^{n+\frac{1}{2}} &= D_0 \left(\frac{U^* - U^n}{\Delta t} \right) \\ &- \left(\frac{S^{n+1} - S^n}{\Delta t} \right) \\ \delta &= \pi^{n+\frac{1}{2}} - \pi^{n-\frac{1}{2}}. \end{aligned} \tag{6.5}$$

These results, and others found while investigating an all-speed algorithm, have suggested that the current formulations do not appropriately treat the potential part of the velocity field. Further work is necessary to correct this problem.

6.2 Methane in Air

In Figures 1 and 2, we present a calculation of methane burning in an air stream, $Re = 100$. The problem is not well resolved on this grid size (128x512) due to the thinness of the reaction zone and the proximity of the viscous boundary. However, all fields show better than first order convergence in L_1. We present the results to illustrate the method in a more realisitic setting.

The mixture fraction field takes on the full range of values, $0 \leq f \leq 1$, with a gaussian profile similar to that in (6.8). As a result, we get temperatures that range from that of ambient air to the adiabatic flame temperature: $300K \leq T \leq 2225K$. Expansion velocities are 30% of the free stream value in the axial direction and 8% of the free stream value in the radial direction. The unconstrained outflow boundary results in most of the expansion occurring in the $+z$-direction.

In Figure 2, we show equilibrium species mass fraction fields obtained through post-processing the mixture fraction. CO_2, a product, is found near the reaction zone and in the rich portion of the field, while the inert N_2 has diffused from the air stream into the core of the fuel rich region. Though NO_x concentrations are not well represented by equilibrium assumptions, the NO field is also shown. This field is high near the flame and very small elsewhere.

7 Conclusions

We have constructed a numerical method for computing unsteady non-premixed reacting flows that is second order accurate in space and time. The algorithm takes full advantage of a fast chemistry assumption to provide reaction rate data and species concentrations by means of look-up tables. A projection method is employed to enforce the non-zero divergence constraint. This work provides the foundation for future investigations of more challenging problems.

8 Acknowledgements

This research is supported by the following two grants:

"Adaptive Numerical Methods for Partial Differential Equations", US Department of Energy, Mathematical, Computing, and Information Sciences Division, Grant DE-FG03-94ER25205.

"Computational Fluid Dynamics and Combustion Dynamics", US Department of Energy High-Performance Computing and Communications Grand Challenge Program, Grant DE-FG03-92ER25140.

The first author was supported by a National Science Foundation Fellowship.

References

[1] A. Almgren, J. Bell, and W. Szymczak, "A numerical method for the incompressible Navier-Stokes equations based on an approximate projection," *SIAM J. Sci. Comput.*, vol. 17, 1996, pp.358-369.

[2] J. B. Bell, P. Colella, and H. M. Glaz, "A second-order projection method for the incompressible Navier-Stokes equations," *J. Comput. Phy.*, vol. 85, 1989, pp. 257-283.

[3] J. B. Bell, P. Colella, and L. H. Howell, "An efficient second-order projection method for viscous incompressible flow," *Proceedings, AIAA 10th Computational Fluid Dynamics Conference, Honolulu, HI, June 24-26, 1991*, pp. 360-367.

[4] J. B. Bell, J. Solomon, and W. Szymczak, "A second-order projection method for the incompressible Navier-Stokes equations on quadrilateral grids," *Proceedings, AIAA 9th Computational Fluid Dynamics Conference, Buffalo, NY, June 14-16, 1989.*

[5] P. Colella, "Multidimensional upwind methods for hyperbolic conservation laws," *J. Comput. Phy.*, vol. 87, 1990, pp. 171-200.

[6] M. Lai and P. Colella, "An approximate projection method for the incompressible Navier-Stokes equations," *to appear.*

[7] M. Lai and P. Colella, "A projection method for combustion in the zero Mach number limit," *Proceedings, AIAA 11th Computational Fluid Dynamics Conference, Orlando, FL, July 6-9, 1993.*

[8] A. Majda and J. A. Sethian, "The derivation and numerical solution of the equations for zero Mach number combustion," *Combust. Sci. Tech*, vol. 42, 1985, pp. 185-205.

[9] B. Van Leer, "Multidimensional explicit difference schemes for hyperbolic conservation laws," in *Computing Methods in Applied Sciences and Engineering, VI*, p. 493, Elsevier Science, Amsterdam, 1984.

[10] M. Minion, "A note on the stability of Godunov projection methods," *Courant Mathematics and Computing Lab*, #95-002, New York, NY, Jan 1995.

[11] R. G. Rehm and H. R. Baum, "The equations of motion for thermally driven, bouyant flows," *Journal of Research of the National Bureau of Standards*, vol. 83 (3), 1978, pp. 297-308.

[12] H. Yamashita, etal., "A numerical study of the transition of jet diffusion flames," *Proc. R. Soc. Lond. A*, vol. 431, 1990, pp. 301-314.

[13] W. Briggs, *A Multigrid Tutorial*, SIAM, Philadelphia, PA, 1987.

[14] R. W. Bilger, "Turbulent Diffusion Flames," *Ann. Rev. Fluid Mech.*, vol. 21, 1989, pp. 101-135.

[15] G. D. van Albada, etal., "A comparitive study of computational methods in cosmic gas dynamics," *Astron. Astrophys.*, vol.108, 1982, pp.76-84.

[16] V. Venkatakrishnan, "Convergence to steady state solutions of the Euler equations on unstructured grid with limiters," *J. Comput. Phys.*, vol. 118, 1995, pp. 120-130.

[17] F. A. Williams, *Combustion Theory*, Benjamin-Cummings, Menlo Park, CA, 1985, 2nd ed.

[18] P. K. Sweby, "High resolution schemes using flux limiters for hyperbolic conservation laws," *SIAM J. Numer. Anal.*, vol. 21, No. 5, 1984, pp. 995-1011.

[19] R. A. Strehlow, *Combustion Fundamentals*, McGraw Hill, New York, NY, 1984.

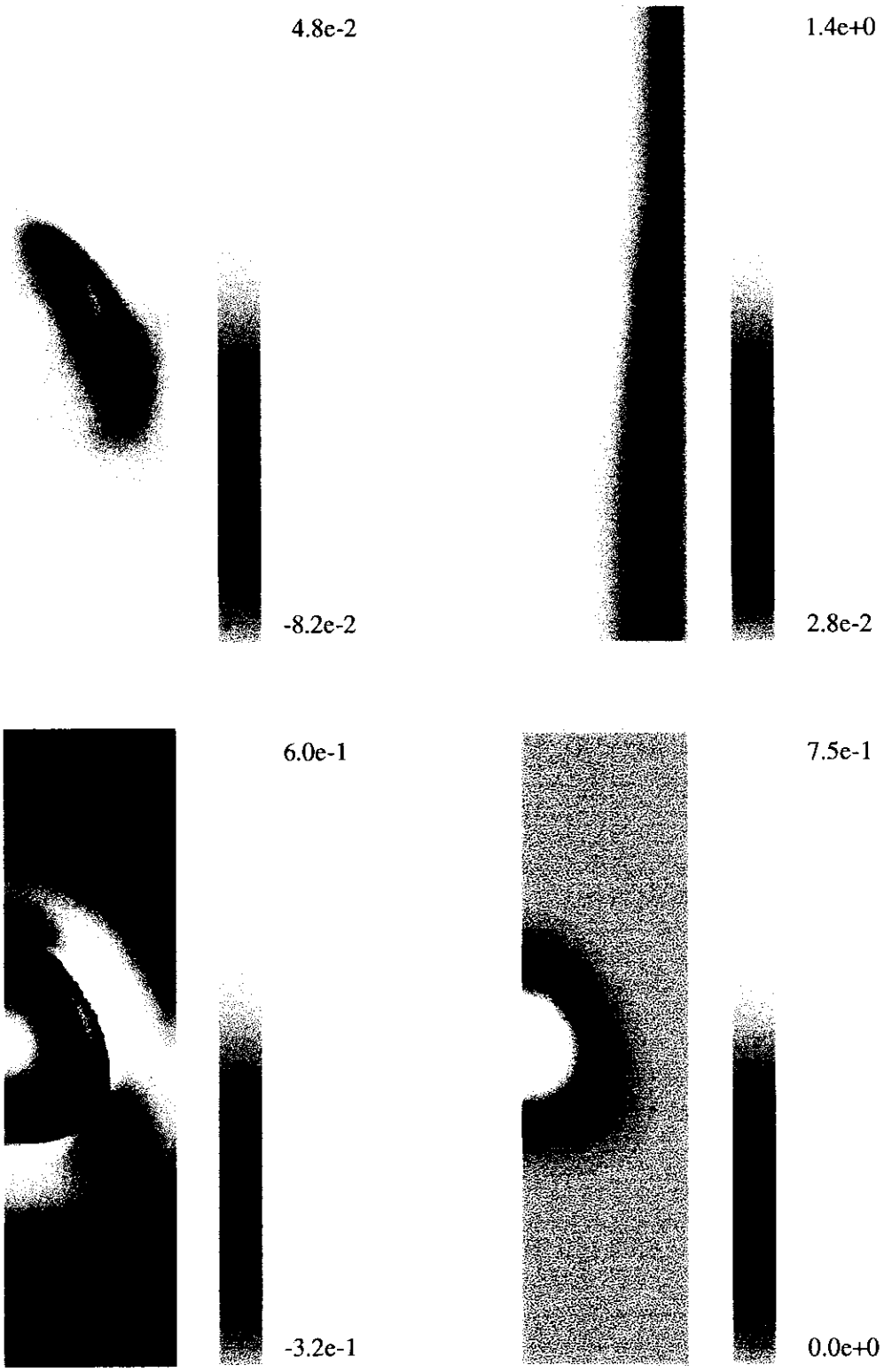

Figure 1: Methane in air results. Fluid flows from bottom to top, axis of symmetry on left, wall on right. Top row, left to right: u, v. Bottom row: S, f.

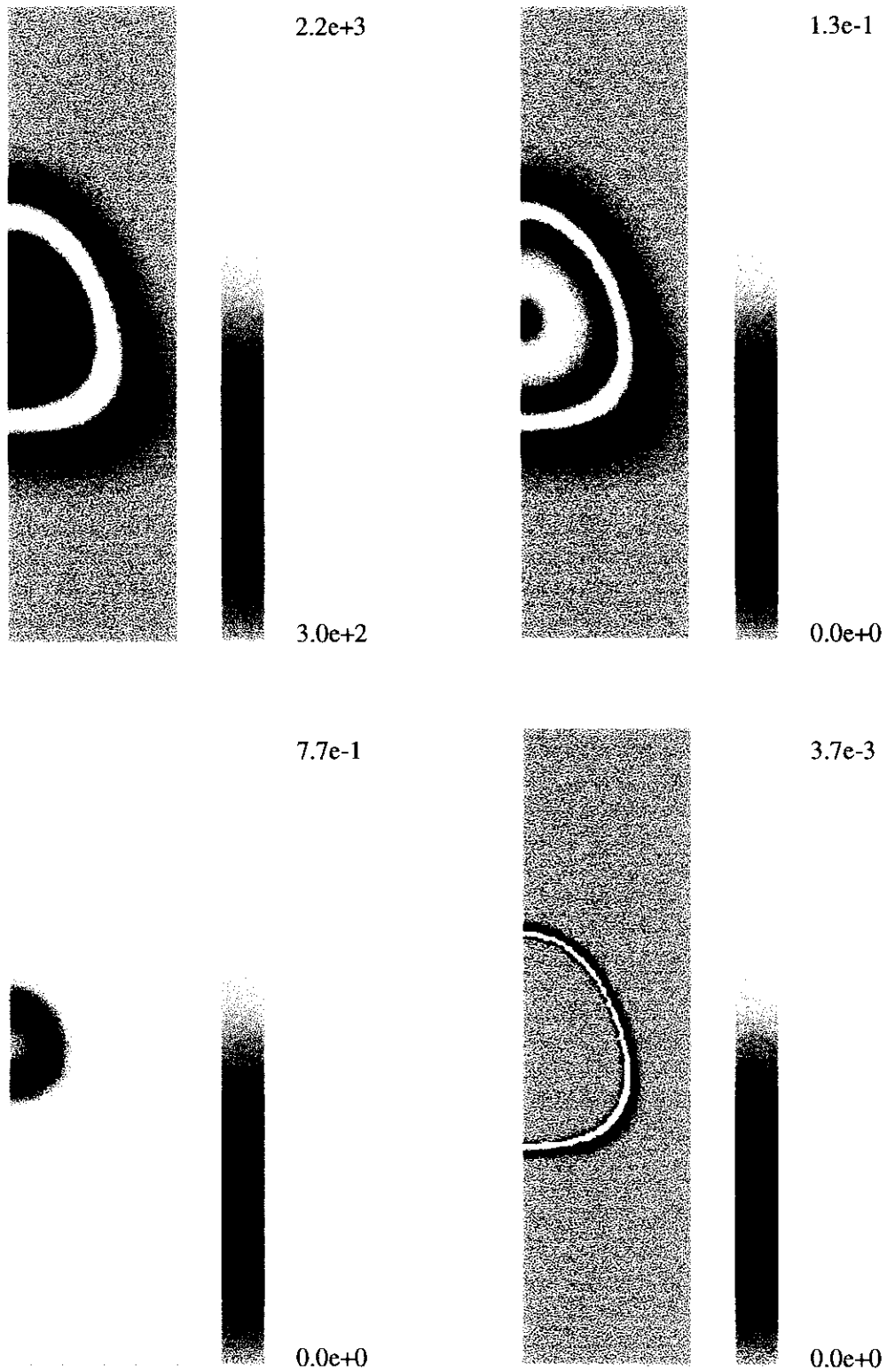

Figure 2: Methane in airled results. Top row, left to right: T, Y_{CO_2}. Bottom row: Y_{N_2}, Y_{NO}.