JOURNAL OF COMPUTATIONAL PHYSICS 104, 99–109 (1993)

Application of a Primitive Variable Newton's Method for the Calculation of an Axisymmetric Laminar Diffusion Flame

YUENONG XU AND MITCHELL D. SMOOKE

Department of Mechanical Engineering, Yale University, P.O. Box 2157, Yale Station, New Haven, Connecticut 06520-2157

Received April 12, 1991; revised March 12, 1992

In this paper we present a primitive variable Newton-based solution method with a block-line linear equation solver for the calculation of reacting flows. The present approach is compared with the stream function–vorticity Newton's method and the SIMPLER algorithm on the calculation of a system of fully elliptic equations governing an axisymmetric methane–air laminar diffusion flame. The chemical reaction is modeled by the flame sheet approximation. The numerical solution agrees well with experimental data in the major chemical species. The comparison of three sets of numerical results indicates that the stream function–vorticity solution using the approximate boundary conditions reported in the previous calculations predicts a longer flame length and a broader flame shape. With a new set of modified vorticity boundary conditions, we obtain agreement between the primitive variable and stream function–vorticity solutions. The primitive variable Newton's method converges much faster than the other two methods. Because of much less computer memory required for the block-line tridiagonal solver compared to a direct solver, the present approach makes it possible to calculate multidimensional flames with detailed reaction mechanisms. The SIMPLER algorithm shows a slow convergence rate compared to the other two methods in the present calculation. © 1993 Academic Press, Inc.

1. INTRODUCTION

Many gas turbines and commercial burners employ diffusion flames as their primary flame type. The ability to predict the coupled effects of transport phenomena with complex chemical process in these systems is critical in improving engine efficiency and in understanding the emission process. The solution of laminar flames can also be employed in the calculation of turbulent reacting flows using the laminar flamelet model, which interprets the turbulent flame as being comprised of an ensemble of laminar flamelets.

The equations governing laminar flames are strongly coupled together as a result of the interaction between the fluid transport, the heat transfer, and the chemical processes. The equations are also characterized by the presence of stiff source terms [13]. The solution of multidimensional laminar diffusion flames requires the calculation of a large system of highly nonlinear elliptic equations. Newton's method has been proven to be a robust and efficient approach in the numerical calculation of such a system [17]. Recently, a few numerical solutions of axisymmetric laminar diffusion flames with different levels of chemical kinetic models have been reported [12, 17]. The numerical approaches used in these calculations were limited to the stream function–vorticity formulation. Such a formulation was adopted since the determination of the pressure in these systems is difficult due to its first-order nature in the momentum equations and its absence in the continuity equation.

One advantage of the stream function–vorticity formulation is that it eliminates the pressure as a dependent variable from the governing equations. However, this reduction brings with it some side effects. The correct boundary conditions for the vorticity are difficult to determine in the case of complex flows [14]. The vorticity boundary conditions specified from the stream function, e.g., at a solid wall, often cause trouble in obtaining a converged solution. The uncertainty of the vorticity boundary conditions at the inlet of the computational domain may result in a rough approximation of the true solution, thus altering the solution of the stream function whose derivatives are used in forming the velocity. In diffusion flames, combustion is controlled primarily by the rate at which the fuel and oxidizer are brought together in stoichiometric proportion. Thus, an accurate representation of the flow field is a precondition for the overall accuracy of the solution. In addition, the eliminated pressure may be desired as a solution during the iteration to evaluate thermodynamic properties, transport coefficients, and reaction rates (e.g., Lindemann fall-off effect) in some cases. Furthermore, in many problems, it is easier to estimate a reasonable velocity and pressure field rather than a stream function and vorticity field. Finally, a stream function–vorticity approach intrinsically limits the modeling of reacting flows to two-dimensional configurations including axisymmetric problems. While a vector–

potential vorticity model can be formulated for three-dimensional flows, they introduce additional dependent variables [1].

The application of Newton's method to a primitive variable formulation of two-dimensional non-reacting viscous flows has been reported previously [22–24]. In these studies, the linear Newton equations were solved by a direct matrix solver. The reason for this is that some zero diagonal elements were present in the Jacobian matrix as a result of the absence of the pressure in the continuity equation. This prevented the use of a block-line iterative solver without pivoting in the diagonal blocks. Computations similar to those reported in [22–24] require a great amount of main memory and are possible only on a supercomputer. In the numerical modeling of multidimensional reacting flows, the computer memory size and time demanded by Newton's method, even with an iterative linear equation solver, is already near the computational limits of some supercomputers. For example, the number of nonzero storage locations in the Jacobian matrix formed in a Newton iteration in a nine-point finite difference scheme is given by $9 \times N \times N \times NODE$, where N is the number of unknowns and $NODE$ is the number of mesh points in the computational domain. It is clear from this discussion that it is essentially infeasible to apply the Newton direct solver approach in modeling multidimensional reacting flows even with a simplified chemistry model. The difficulties described so far can be resolved by introducing Newton's method with a primitive variable formulation in such a way that the Jacobian matrix can still be solved by a block-line tridiagonal iterative method. The advantages of this numerical approach are: (1) Newton's method is particularly robust in handling the nonlinearity and the coupled effects among the equations of reacting flows; (2) the primitive variable formulation is suitable for complex flows and three-dimensional problems; (3) the use of a block-line tridiagonal iterative method for the Jacobian matrix greatly reduces the requirement on the computer memory and thus makes it possible for large scale computations, e.g., three-dimensional problems.

In this paper we employ a primitive variable Newton's method to calculate a fully coupled elliptic model of an axisymmetric, methane–air, laminar diffusion flame in a confined co-flowing jet (see Fig. 1). The chemistry is approximated by a flame sheet model [2, 17]. The transport coefficients are evaluated with simple empirical formulae. The purpose of using a flame sheet model in the present work is twofold: the performance of the present numerical method is compared with Newton's method in the stream function–vorticity formulation and the SIMPLER algorithm [12] on the basis of this model; the solution of the flame sheet model can be used as an initial estimate for calculations using detailed chemistry models [11, 17, 25]. The Patankar–Spalding SIMPLER algorithm solves the primitive variable equations and has been quite successful in solving non-reacting flows, but it often fails or becomes prohibitively slow in calculating reacting flows associated with high nonlinearity in the source terms. In the next section, the flame sheet problem is formulated using the primitive variables and the stream function-vorticity approach. The numerical methods are presented in Section 3 and the numerical results are discussed in Section 4. Some final comments are contained in Section 5.

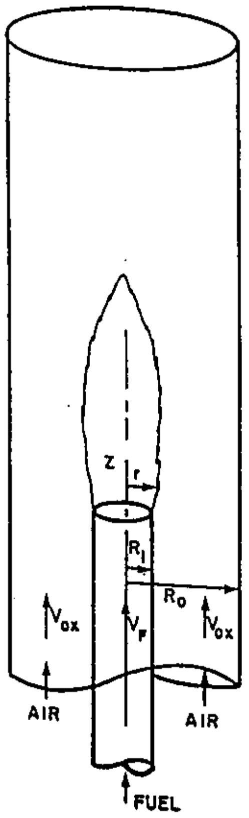


FIG. 1. Schematic of a diffusion flame in a confined co-flowing burner.

2. Problem Formulation

The laminar diffusion flame considered in the present work is produced by an axisymmetric co-flowing jet in which the fuel flows in the center jet with the air stream from the surrounding jet (see Fig. 1). The jet is confined by a shield wall. A series of extensive experimental studies have been made on this burner [12, 13], which makes it a good test case for the calculation of two-dimensional laminar diffusion flames. The previous numerical modeling of similar burners ranged from a flame sheet model to a complicated chemical reaction mechanism involving 16 chemical species equations [12, 17]. All previous elliptic solutions for this burner configuration were obtained using the stream function-vorticity formulation.

Our purpose in the present work is to demonstrate that Newton's method in the primitive variable setting is computationally feasible and can perform well against Newton's method when applied to the stream function–vorticity formulation and the SIMPLER algorithm. In the chemical reaction model that follows, a flame sheet approximation is employed.

2.1. *Primitive Variable Formulation of the Flame Sheet Model*

The flame sheet concept was proposed by Burke and Schumann [2]. The flame sheet model of a laminar diffusion flame describes the chemical reaction with a one-step global irreversible reaction. The reaction is assumed infinitely fast and limited to a very thin exothermic reaction zone which separates the fuel and oxidizer. In the reaction zone, the fuel and oxidizer react in stoichiometric proportion. With such an approximation, no fuel is present on the oxidizer side and vice versa.

In the presence of an inert gas (N), the reaction of fuel (F), and oxidizer (X) to form product (P) can be written in the form

$$\nu_F F + \nu_X X + N \rightarrow \nu_P P + N, \tag{2.1}$$

where ν_F, ν_X, and ν_P are the stoichiometric coefficients of the reaction. To further simplify the governing equations [17], we assume that (1) thermal diffusion is negligible, (2) the specific heats, c_p and c_{p_k} ($k = F, X, P, N$), are constant, (3) the ordinary mass diffusion velocities obey Fick's law, and (4) the Lewis numbers of all the species are equal to unity.

With these approximations, the energy equation and the species equations become similar mathematically. By introducing Shvab–Zeldovich variables, the solutions of the temperature and the species can be formulated from the conserved scalar solution of a source-free convective-diffusive equation [17]. The primitive variable governing equations for a flame sheet model of an axisymmetric, laminar diffusion flame in cylindrical coordinates are

Continuity,

$$\frac{\partial}{\partial r}(r\rho v) + \frac{\partial}{\partial z}(r\rho u) = 0; \tag{2.2}$$

Axial momentum,

$$\frac{\partial}{\partial z}(r\rho uu) + \frac{\partial}{\partial r}(r\rho vu) - \frac{4}{3}\frac{\partial}{\partial z}\left(r\mu\frac{\partial u}{\partial z}\right) - \frac{\partial}{\partial r}\left(r\mu\frac{\partial u}{\partial r}\right) = -r\frac{\partial p}{\partial z} + r\rho g + \left[-\frac{2}{3}\frac{\partial}{\partial z}\left(\mu\frac{\partial(rv)}{\partial r}\right) + \frac{\partial}{\partial r}\left(r\mu\frac{\partial v}{\partial z}\right)\right]; \tag{2.3}$$

Radial momentum,

$$\frac{\partial}{\partial z}(r\rho uv) + \frac{\partial}{\partial r}(r\rho vv) - \frac{\partial}{\partial z}\left(r\mu\frac{\partial v}{\partial z}\right) - \frac{4}{3}\frac{\partial}{\partial r}\left(r\mu\frac{\partial v}{\partial r}\right) = -r\frac{\partial p}{\partial r} + \left[-\frac{4}{3}\frac{\mu v}{r} - \frac{2}{3}v\frac{\partial \mu}{\partial r}\right] + \left[-\frac{2}{3}r\frac{\partial}{\partial r}\left(\mu\frac{\partial u}{\partial z}\right) + r\frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial r}\right)\right]; \tag{2.4}$$

Conserved scalar,

$$\left[r\rho v\frac{\partial S}{\partial r} + r\rho u\frac{\partial S}{\partial z}\right] - \frac{\partial}{\partial r}\left(r\rho D\frac{\partial S}{\partial r}\right) - \frac{\partial}{\partial z}\left(r\rho D\frac{\partial S}{\partial z}\right) = 0; \tag{2.5}$$

Equation of state,

$$\rho = \frac{p\bar{W}}{RT}, \quad \text{where} \quad \frac{1}{\bar{W}} = \sum_{k=1}^{4}\frac{Y_k}{W_k}, \quad k = F, X, P, N. \tag{2.6}$$

Of critical importance to the flame sheet model is the ability to recover the temperature and the major species profiles from the conserved scalar solution. If we denote variables at the flame front with the subscript f, then it can be shown that the location of the flame front at the axial coordinate z can be obtained using

$$S(r_f)|_{\text{fixed } z} = S_f = Y_{X_O}\bigg/\left(Y_{X_O} + \frac{W_X \nu_X}{W_F \nu_F} Y_{F_I}\right), \tag{2.7}$$

where the subscripts I and O refer to the inner jet and the outer jet, respectively.

Using the result of Eq. (2.7) we can generate expressions for the temperature and species on the fuel and oxidizer sides of the flame as follows. On the fuel side we have

$$T = T_I S + \left[T_O + Y_{X_O}\frac{Q}{c_p}\frac{W_F \nu_F}{W_X \nu_X}\right](1 - S), \tag{2.8}$$

$$Y_F = Y_{F_I} S + Y_{X_O}\frac{W_F \nu_F}{W_X \nu_X}(S - 1), \tag{2.9}$$

$$Y_X = 0, \tag{2.10}$$

$$Y_P = Y_{X_O}\frac{W_P \nu_P}{W_X \nu_X}(1 - S), \tag{2.11}$$

$$Y_N = Y_{N_O}(1 - S) + Y_{N_I} S. \tag{2.12}$$

On the oxidizer side we have

$$T = T_O(1-S) + \left[\frac{Q}{c_p} Y_{F_I} + T_I\right] S, \qquad (2.13)$$

$$Y_F = 0, \qquad (2.14)$$

$$Y_X = Y_{X_O}(1-S) - Y_{F_I}\frac{W_X \nu_X}{W_F \nu_F} S, \qquad (2.15)$$

$$Y_P = \frac{W_P \nu_P}{W_F \nu_F} Y_{F_I} S, \qquad (2.16)$$

where $\mathrm{Pr}_{\mathrm{ref}}$ is a reference Prandtl number and λ is the thermal conductivity of the mixture. Hence, determination of ρD is reduced to the specification of a transport relation for the viscosity, which is approximated by the power law

$$\mu = \mu_0 \left(\frac{T}{T_0}\right)^r, \qquad (2.23)$$

where $\mathrm{Pr}_{\mathrm{ref}} = 0.75$, $r = 0.7$, $T_0 = 298\,K$, and $\mu_0 = 1.85 \times 10^{-4}$ gm/cm are reference values for air [9]. The heat release parameter Q/c_p in Eq. (2.8) can be

where the *max* operator is equivalent to the intrinsic function *AMAX* 1 in FORTRAN.

3.2. *Newton's Method with Adaptive Time Steps*

The system of discretized equations (see [25] for detail) including an unsteady term is given by

$$D\frac{\partial \phi}{\partial t} + F(\phi) = 0, \qquad (3.2)$$

where D is a scaling matrix and F is the steady-state residual function evaluated at the solution ϕ. If the unsteady term is implicitly differenced with a backward Euler method, we have

$$\bar{F}(\phi^n) = F(\phi^n) + D\frac{(\phi^n - \phi^{n-1})}{\Delta t^n} = 0, \qquad (3.3)$$

where ϕ^n and ϕ^{n-1} are the solutions at step n and $n-1$, respectively, and $\Delta t^n = t^n - t^{n-1}$. The above system of equations can be solved efficiently using a damped and modified Newton iteration [17, 25],

$$J(\phi^k)(\phi^{k+1} - \phi^k) = -\lambda^k \bar{F}(\phi^k), \qquad k = 0, 1, 2, \ldots, \qquad (3.4)$$

where λ^k is the damping parameter at the kth Newton iteration. $J(\phi^k)$ is the Jacobian matrix,

$$J(\phi^k) = \frac{\partial \bar{F}(\phi^k)}{\partial \phi^k}. \qquad (3.5)$$

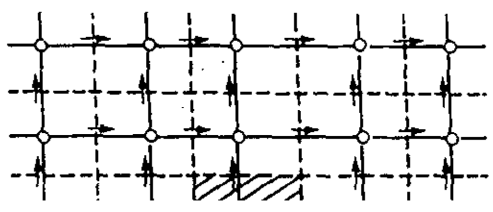
With the spatial discretizations described in Section 3.1, the Jacobian matrix in (3.5) can be written in block nine-diagonal form. For problems involving complex transport and chemistry models, it is often more efficient to evaluate the Jacobian matrix numerically as opposed to analytically [3, 14]. Once the Jacobian matrix is formed, the Newton equations (3.4) can be solved by a matrix solver. Given an initial solution estimate, we follow the transient solution (Eq. (3.3)) until the Euclidean norm of the residual, $\|F(\phi^n)\|_2$, is small enough so that Newton's method can be

the new grid to generate the initial solution estimate for the calculation on the new mesh. The grid refinement may be applied several times depending on the particular problem.

3.3. *Primitive Variable Formulation with Newton's Method*

The difficulties with the primitive variable formulation lie in the treatment of the pressure terms in the momentum equations and the velocity components in the continuity equation. If the first-order derivatives of the pressure in the momentum equations are discretized by a central difference operator on a regular grid, a zigzag pressure field will be regarded as realistic by the momentum equations [15]. A similar kind of difficulty arises in discretizing the continuity equation. However, these difficulties can be resolved by employing the staggered grid technique. In the staggered grid, the axial velocity $u_{i,j}$ is shifted half way from the regular grid point to the point $(i+\frac{1}{2}, j)$; the radial velocity $v_{i,j}$ is similarly moved to the point $(i, j+\frac{1}{2})$, and the rest of the dependent variables are still kept at the regular point (i, j) (see Fig. 2). The structure of the present staggered grid requires specification of the pressure boundary condition at the outlet of the reactor, which reflects the physical characteristics of internal flows. It is worth mentioning that the shifting of grids for the velocity only moderately increases the programming complexity on a rectangular mesh. A by-product of the approach is that it increases the accuracy of the discretization for some of the first-order derivatives. For example, the continuity equation can be discretized using central differences at adjacent points. In addition, it enhances the diagonal dominance of the Jacobian matrix due to the compact points at each grid cell.

As we discussed in Section 1, a direct matrix solver is still infeasible for the calculation of reacting flows. Therefore, a key factor in the primitive variable Newton's method is the



successful application of a block-line iterative method for the solution of the linear Newton equations (3.4). The structure of the nine-strip blocks in the Jacobian matrix should be kept so that the block-line tridiagonal iteration method can be applied efficiently. While a direct matrix solver can be applied to any nonsingular matrix, a block-line tridiagonal iterative method without pivoting can only be applied to a nonsingular matrix without zero diagonal elements. Since the pressure does not explicitly appear in the continuity equation, some zero diagonal elements are present in the Jacobian matrix. As a result, we employ a block-line tridiagonal iterative method with partial pivoting [8]. In this way we only require that the overall matrix and each diagonal block be nonsingular.

3.4. *Stream Function–Vorticity Formulation with Newton's Method*

Newton's method applied to the stream function–vorticity formulation is presented in detail in [17]. In the present work, we solve for the vorticity w as one of the dependent variables instead of w/r as in [12, 17]. This gives the advantage of utilizing Dirichlet boundary conditions for the vorticity at the symmetric line and thus avoids the undefined value of w/r at the same boundary.

3.5. *The SIMPLER Algorithm*

The SIMPLER algorithm described in [15] is modified to calculate compressible reacting flows. Due to the large variation of the density occurring in reacting flows, the pressure and pressure correction equations must be solved exactly to ensure the conservation of mass before calculating the other equations, e.g., the conserved scalar equation (Eq. (2.5)). We initially encountered difficulties in calculating the pressure and pressure correction equations when using a simple tridiagonal method. The time step had to be kept very small in order for the iteration to converge. Due to the relatively small size, the very sparse nature, and the same structure of the pressure and pressure correction equations, a direct sparse matrix solver, YSMP [5], was employed to solve these two equations. As a result, mass is conserved over each grid cell after the velocity correction is made.

4. NUMERICAL RESULTS AND DISCUSSION

The overall reaction for a methane–air system with an inert gas (N_2) is

$$CH_4 + 2O_2 + N_2 \rightarrow 2H_2O + CO_2 + N_2. \tag{4.1}$$

The reactor configuration is such that the radius of the inner fuel jet $R_I = 0.635$ cm, the radius of the outer oxidizer jet $R_O = 2.54$ cm, and the length of the tubular pyrex shield $Z = 30.0$ cm. The flow conditions at the inlet ($z = 0$) are given by

$r \leqslant R_I$,

$$v_r = 0.0 \text{ cm/s}, \quad v_z = 4.50 \text{ cm/s}, \quad T = 298 \text{ K}, \quad Y_{CH_4} = 1.0, \quad Y_{O_2} = Y_{H_2O} = Y_{CO_2} = Y_{N_2} = 0; \tag{4.2}$$

$R_I < r < R_O$,

$$v_r = 0.0 \text{ cm/s}, \quad v_z = 9.88 \text{ cm/s}, \quad T = 298 \text{ K}, \quad Y_{O_2} = 0.232, \quad Y_{N_2} = 0.768, \quad Y_{CH_4} = Y_{H_2O} = Y_{CO_2} = 0. \tag{4.3}$$

The pressure at the exit is $p = 1$ atm. The shield temperature is approximated at 298 K. The peak burning temperature is estimated to be 2050 K from the experimental data [12]. For simplicity, we denote *p-u-v* as the primitive variable solution and ψ-ω as the stream function–vorticity solution with Newton's method in the following discussion.

The numerical solutions are first computed on a 55×50 nonuniform grid with all three methods discussed in the previous section. This initial grid is constructed by using algebraic expressions [1] to cluster grid points in the regions of high spatial activity (e.g., near the solid wall, the fuel and air jet boundaries). The converged solutions are used to refine the grid. The grid refinement is applied two times to produce grids of 65×65 and finally 75×75 points. The solution on the new mesh is easily obtained with a small

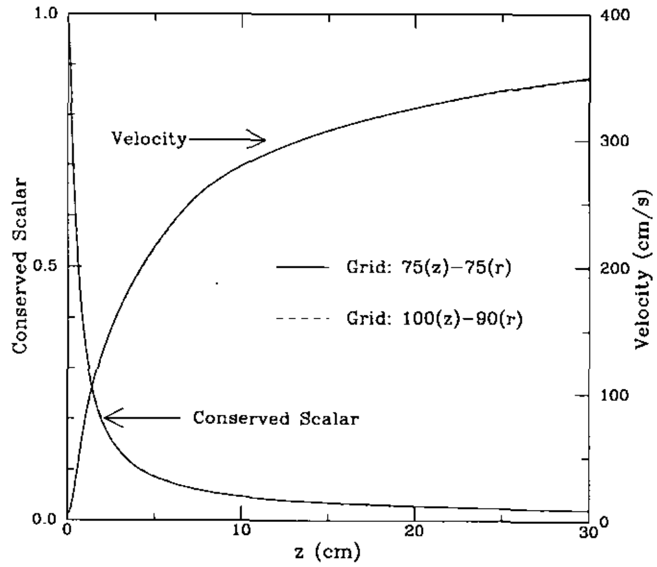


FIG. 3. Computed axial profiles of the conserved scalar and the axial velocity at the centerline ($r = 0$) from two sets of grids, $75(z) \times 75(r)$ grid (solid line), $100(z) \times 90(r)$ grid (dashed line). (The lines are overlapped.)