Turkish J. Eng. Env. Sci.
27 (2003) , 275 – 290.

Numerical Simulation of a Confined Methane/Air Laminar Diffusion Flame by the Method of Lines

Tanıl TARHAN and Nevin SELÇUK
Middle East Technical University, Chemical Engineering Department,
06531 Ankara-TURKEY
e-mail: selcuk@metu.edu.tr

Received 21.03.2003

Abstract

A parallel computational fluid dynamics code based on the method of lines approach was developed for the simulation of transient laminar two-dimensional reacting flows. The chemistry was modeled by flame-sheet approximation. The predictive ability and accuracy of the code was tested by applying it to the prediction of a confined methane/air laminar diffusion flame and comparing its predictions with other numerical study and experimental data available in the literature. The predicted velocity, temperature and major species concentrations were found to be in reasonably good agreement with other numerical results and measurements. Execution times resulting from the use of several ordinary differential equation solvers in the code were examined and ROWMAP was found to be superior. A static load balancing strategy was used to improve the performance of the parallel code. A comparison of speed-ups and efficiencies for the runs with and without load balancing showed a considerable decrease in the central processing unit time of the parallel code when load balancing was performed. Finally, the predictive ability of the code for transient combustion was presented.

Key words: Method of lines (MOL), Unsteady combustion, Confined laminar diffusion flame, Parallel computation, Flame-sheet model

Introduction

Practical combustion systems often involve three-dimensional, unsteady, turbulent flames burning in environments of continually varying chemical composition. Non-premixed (or diffusion) combustion occurs in all systems in which the fuel and oxidizer are not perfectly premixed before entering the combustion chamber. As the majority of practical combustion devices operate with non-premixed flames in the presence of turbulent flow, the modeling of non-premixed turbulent combustion has become a central issue in understanding combustion systems. The computational modeling of turbulent reacting flows with today's limited computer resources is complicated by the presence of a broad range of flow and length scales (Williams, 1985) and by a large number of species, and hence the modeling of turbulent non-premixed flames is a scientific challenge, a task that has been the subject of numerous studies (Yamashita *et al.*, 1990; Cuenot and Poinsot, 1994; Pember *et al.*, 1996; Poinsot, 1996; Neuber *et al.*, 1998; Vervisch and Poinsot, 1998; Zhou *et al.*, 2000; Kronenburg and Bilger, 2001; Coelho and Peters, 2001). However, the direct numerical simulation (DNS) of turbulent reacting flames including complex chemistry is still beyond the capabilities of current computational resources. The simulation of time-varying laminar flames as well as steady ones is an intermediate step in the modeling of unsteady turbulent flames, since these time-varying flames enable a detailed understanding of dynamic and complex interactions between the chemistry and fluid flow. Therefore, detailed experimental and numerical investigations of both steady and unsteady laminar diffusion flames are important in understanding the structure

of laminar flames and in predicting more complicated flames. Consequently, many experimental and numerical studies on confined laminar diffusion flames have been performed (Mitchell *et al.*, 1980; Smooke *et al.*, 1989; Xu and Smooke, 1993; Xu *et al.*, 1993; Coelho and Pereira, 1993).

Because a lot of grid points as well as time steps are needed for the direct simulation of reacting flows, both accurate and efficient numerical techniques and high performance computing are still required for simulation in a short computation time. The former can be achieved by increasing the order of the spatial discretization method, resulting in high accuracy with fewer grid points, and using not only a highly accurate but also a stable numerical algorithm for the time integration. The method of lines (MOL) is an alternative approach that meets this requirement for time-dependent problems. The latter requirement is met by either supercomputers or parallel computers, which require efficient parallel algorithms. The MOL involves converting the partial differential equations (PDEs) into an ordinary differential equation (ODE) initial value problem by discretizing the spatial derivatives together with the boundary conditions via finite differences, finite elements, finite volumes or weighted residual techniques, and integrating the resulting ODEs using a sophisticated ODE solver that takes the burden of time discretization and chooses the time steps in such a way that the accuracy and stability of the evolving solution are maintained. The most important advantage of the MOL approach is that it has not only the simplicity of the implementation of the explicit methods used for the evaluation of spatial derivatives but also the superiority of the implicit methods. The advantages of the MOL approach are twofold. First, it is possible to achieve higher-order approximations in the discretization of spatial derivatives without a significant increase in computational complexity except for in the boundaries. Second, comparable orders of accuracy can be obtained in the time steps when highly efficient and reliable initial value ODE solvers are used.

Recently, the numerical simulation of time-dependent 2-D Navier-Stokes equations for non-reacting flows has been carried out by the MOL approach satisfying the requirement of an accurate and efficient numerical algorithm in conjunction with *i*) an intelligent higher-order spatial discretization scheme, *ii*) a parabolic algorithm for the computation of pressure, and *iii*) an elliptic grid generator using a body-fitted curvilinear coordinate system for application to complex cylindrical (Oymak, 1997; Selçuk and Oymak, 1999) and rectangular (Tarhan, 1999; Tarhan and Selçuk, 2001) geometries. The validity and predictive ability of the code were tested by applying it to the prediction of flow fields in both laminar and turbulent flows and comparing its predictions with either measured data or numerical results available in the literature. The predicted flow fields were found to be in perfect agreement with measurements for laminar flow. The success of the code for turbulent flows, however, was found to depend strongly on the number of grid points, which cannot be handled by most present-day computers. The requirement of high-performance computing was also met by developing an efficient parallel algorithm for the code (Erşahin *et al.*, 2003). The performance of the parallel code was assessed by applying it to laminar and turbulent flows and comparing the results of serial and parallel codes. Comparisons showed that the flow fields predicted by parallel codes agreed well with those of serial codes at considerably shorter execution times. The encouraging agreement obtained between the predicted flow fields and measurements for non-reacting flows and the high performance of the parallel code in the prediction of transient flows have led to the development of a parallel code based on the same approach for the prediction of reacting flows.

This paper describes the initial steps in the development of an efficient numerical scheme for the direct simulation of unsteady, multidimensional combustion with stiff detailed chemistry. To the authors' knowledge, a parallel implemented computational fluid dynamics (CFD) code based on MOL for multidimensional reacting flows is not yet available.

Physical Modeling

Unsteady axisymmetric laminar diffusion flames require the solution of a full set of two-dimensional governing equations. As the purpose of the present study is to demonstrate the performance of MOL in reacting flows, a flame-sheet model is considered rather than a finite rate chemistry model. The purpose of using a flame-sheet model is twofold: *i*) A detailed reaction mechanism with a large number of species requires a large number of equations to be solved for each species, whereas the flame-sheet model adds only one field to the hydrodynamics fields that describe the underlying flow and alleviates

the high memory and central processing unit (CPU) time requirements in the computer simulations, *ii*) The solution obtained by the flame-sheet model can be employed as an initial estimate for multidimensional reacting systems in which detailed chemistry models are used.

Flame-sheet model

In diffusion-type flames, the burning rate is controlled by the rate at which the fuel and the oxidizer are brought together in the proper proportions. In the flame-sheet model, the chemical reactions are described with a single one-step irreversible reaction corresponding to infinitely fast conversion of reactants into stable products. This reaction is assumed to be limited to a very thin exothermic reaction zone located at the locus of the stoichiometric mixing of fuel and oxidizer, where temperature and the products of combustion are maximized. This assumption results in an absence of fuel in the oxidizer side and of oxidizer in the fuel side.

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

$$\nu_F F + \nu_X X + N \rightarrow \nu_P P + N, \tag{1}$$

where ν_F, ν_X and ν_P are the stoichiometric coefficients of species i. To further simplify the governing equations, it is assumed that i) thermal diffusion is negligible, ii) the specific heats, C_p and $C_{p,i}$ ($i = F, X, P, N$), are constant, iii) the ordinary mass diffusion velocities obey Fick's law, and iv) the Lewis number of all the species is equal to unity.

With these approximations, the energy equation and the species equations become mathematically similar. By introducing a mixture-fraction approach, one can derive a source-free convective-diffusive equation for a single conserved scalar. The temperature and stable major species' profiles can be obtained from the solution of the conserved scalar equation coupled to the flow fields equations. The location of the spatially distributed reaction zone and its temperature distribution can be adequately predicted by the flame-sheet model for many important fuel-oxidizer combinations and configurations. Flame sheets can be employed to initialize multidimensional diffusion flames.

Governing equations

In primitive variables where r and z denote radial and axial coordinates, respectively, incompressible conservation equations (Desjardin and Frankel, 1997; Becker *et al.*, 1999) for an axisymmetric, laminar diffusion flame in cylindrical coordinates can be written as follows:

continuity

$$\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} + \frac{v}{r} = 0, \tag{2}$$

z-momentum

$$\begin{aligned}\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial z} + v\frac{\partial u}{\partial r} &= -\frac{1}{\rho}\frac{\partial p}{\partial z} \\ &+\frac{\mu}{\rho}\left(\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{\partial^2 u}{\partial z^2}\right) \\ &+\frac{1}{\rho}\left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z}\right)\frac{\partial \mu}{\partial r} + \frac{2}{\rho}\left(\frac{\partial u}{\partial z}\right)\frac{\partial \mu}{\partial z} + g_z,\end{aligned} \tag{3}$$

r-momentum

$$\begin{aligned}\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial z} + v\frac{\partial v}{\partial r} &= -\frac{1}{\rho}\frac{\partial p}{\partial r} \\ &+\frac{\mu}{\rho}\left(\frac{\partial^2 v}{\partial r^2} + \frac{1}{r}\frac{\partial v}{\partial r} - \frac{v}{r^2} + \frac{\partial^2 v}{\partial z^2}\right) \\ &+\frac{2}{\rho}\left(\frac{\partial v}{\partial r}\right)\frac{\partial \mu}{\partial r} + \frac{1}{\rho}\left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z}\right)\frac{\partial \mu}{\partial z},\end{aligned} \tag{4}$$

conserved scalar

$$\begin{aligned}\frac{\partial S}{\partial t} + u\frac{\partial S}{\partial z} + v\frac{\partial S}{\partial r} &= D\left(\frac{\partial^2 S}{\partial r^2} + \frac{1}{r}\frac{\partial S}{\partial r} + \frac{\partial^2 S}{\partial z^2}\right) \\ &+\left(\frac{\partial S}{\partial r}\right)\frac{\partial D}{\partial r} + \left(\frac{\partial S}{\partial z}\right)\frac{\partial D}{\partial z}.\end{aligned} \tag{5}$$

These equations are solved simultaneously for velocity and conserved scalar fields. Then the temperature and major species profiles are recovered from the conserved scalar solution. The location of the flame front, f, at the axial coordinate z can be obtained using

$$S_f = \frac{Y_{X_O}}{Y_{X_O} + \frac{W_X \nu_X}{W_F \nu_F} Y_{F_I}} \tag{6}$$

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

The following expressions are used for the determination of temperature and species on the fuel and oxidizer sides of the flame. On the fuel side:

$$T = T_I S + \left[T_O + Y_{X_O}\frac{Q}{C_p}\frac{W_F \nu_F}{W_X \nu_X}\right](1 - S), \tag{7}$$

$$Y_F = Y_{F_I} S + Y_{X_O}\frac{W_F \nu_F}{W_X \nu_X}(S - 1), \tag{8}$$

$$Y_X = 0, \tag{9}$$

$$Y_P = Y_{X_O}\frac{W_P\nu_P}{W_X\nu_X}(1-S), \tag{10}$$

$$Y_N = Y_{N_O}(1-S) + Y_{N_I}S. \tag{11}$$

On the oxidizer side

$$T = T_O(1-S) + \left[Y_{F_I}\frac{Q}{C_p} + T_I\right]S, \tag{12}$$

$$Y_F = 0, \tag{13}$$

$$Y_X = Y_{X_O}(1-S) - Y_{F_I}\frac{W_X\nu_X}{W_F\nu_F}S, \tag{14}$$

$$Y_P = Y_{F_I}\frac{W_P\nu_P}{W_F\nu_F}S, \tag{15}$$

$$Y_N = Y_{N_O}(1-S) + Y_{N_I}S. \tag{16}$$

In the case of two products, species profiles are determined by

$$Y_{P_1} = \left(\frac{W_{P_1}\nu_{P_1}}{W_{P_1}\nu_{P_1} + W_{P_2}\nu_{P_2}}\right)Y_P, \tag{17}$$

$$Y_{P_2} = \left(\frac{W_{P_2}\nu_{P_2}}{W_{P_1}\nu_{P_1} + W_{P_2}\nu_{P_2}}\right)Y_P. \tag{18}$$

In these equations, ρ is the mass density, u and v the axial and radial velocity of the fluid mixture, S the conserved scalar, p the pressure, g the gravity force, Y_i the mass fraction of the ith species, W_i the molecular weight of the ith species, μ the mixture viscosity, D the mixture diffusivity, T the temperature, Q the heat release per unit mass of the fuel and C_p the specific heat capacity of the mixture.

For a given profile of the conserved scalar, Eqs. (7)-(18) are utilized to obtain solutions for T, Y_F, Y_X, Y_P and Y_N. Transport properties, namely, the viscosity, μ, and the thermal conductivity, λ, are calculated using the TRANSPORT package of Kee *et al.* (1986). Thermodynamic properties, namely, the density, ρ, and the specific heat, C_p, are evaluated using CHEMKIN-III (Kee *et al.*, 1996) and its database. Recalling that all the Lewis numbers are equal to one, mixture diffusivity is calculated by $D = \lambda/\rho C_p$. Following Xu and Smooke (1993), the heat release parameter Q/C_p is determined from the heat of combustion and a representative heat capacity.

Initial and boundary conditions

The governing equations are subject to the initial and boundary conditions given in Table 1.

At the fuel inlet S_{in} is 1, and at the oxidizer inlet it is 0. Before the sudden start of the flow ($t = 0$), a zero velocity field is prescribed in the entire domain filled with room temperature air. The initial conserved scalar field is set to 0 due to the presence of air everywhere. At the fuel and oxidizer inlets, uniform velocity profiles are prescribed.

Treatment of pressure gradient and computation of radial velocity component

The computation of pressure is the most difficult and time-consuming part of the overall solution of the Navier-Stokes equations and there are various pressure correction methods which are applicable to both stationary and time-dependent flow equations. Basically, most of them involve an iterative procedure between the velocity and pressure fields through the solution of a Poisson-type equation for the pressure to satisfy the global mass flow constraint and the divergence-free condition for confined incompressible flows. Therefore, in this paper a non-iterative procedure proposed recently (Oymak and Selçuk, 1996) for the treatment of pressure gradients is applied. Based on this technique, the axial pressure gradient is determined by the following equation:

$$\left(\frac{\partial p}{\partial z}\right)^n_j = \frac{2\pi\rho\int_0^R \Phi^n_{i,j} r dr - \dot{m}}{\pi R^2 \Delta t} \tag{19}$$

where

$$\Phi^n_{i,j} = u^n_{i,j} - \Delta t\Bigg\{u^n_{i,j}\left(\frac{\partial u}{\partial z}\right)^n_{i,j} + v^n_{i,j}\left(\frac{\partial u}{\partial r}\right)^n_{i,j} - \frac{\mu}{\rho}\left[\left(\frac{\partial^2 u}{\partial z^2}\right)^n_{i,j} + \frac{1}{r}\left(\frac{\partial u}{\partial r}\right)^n_{i,j} + \left(\frac{\partial^2 u}{\partial r^2}\right)^n_{i,j}\right]\Bigg\}. \tag{20}$$

Here $\dot{m}$ is the mass flow rate prescribed as the inlet condition and R is the radius of the burner.

The radial velocity component is obtained from the continuity equation (Eq. (2)) by the formula

$$v^n_{i+1,j} = \frac{r_i}{r_{i+1,j}}\left[v^n_{i,j} - (r_{i+1} - r_i)\left(\frac{\partial u}{\partial z}\right)^n_{i,j}\right], \quad i = 1, ..., IR-2, \quad j = 2, ..., JZ, \tag{21}$$

where n represents time level.

Table 1. Initial and boundary conditions.

IC	@ $t = 0$,	$\forall z \wedge \forall r$;	$u = 0$,	$v = 0$,	$S = S_o$,
BC 1	@ center,	$\forall z \wedge \forall t$;	$\partial u/\partial r = 0$,	$v = 0$,	$\partial S/\partial r = 0$,
BC 2	@ wall,	$\forall z \wedge \forall t$;	$u = 0$,	$v = 0$,	$S = 0$,
BC 3	@ inlet,	$\forall r \wedge \forall t$;	$u = u_{in}$,	$v = 0$,	$S = S_{in}$,
BC 4	@ outlet,	$\forall r \wedge \forall t$;	$\partial^2 u/\partial z^2 = 0$,	$\partial^2 v/\partial z^2 = 0$,	$\partial^2 S/\partial z^2 = 0$.

By these approaches, not only are pressure gradient and radial velocity component computed in less CPU time, but also the divergence-free condition for incompressible flows is satisfied automatically.

Numerical Solution Technique

In this study, the governing equations are solved using the numerical MOL (Schiesser, 1991). Many existing numerical algorithms for transient PDEs can be considered MOL algorithms. The most important difference between the MOL approach and conventional methods is that in the MOL approach higher-order, implicit and hence stable numerical algorithms for time integration are used. For the numerical solution of the same PDE system, the MOL approach and the conventional methods, in which lower-order explicit or implicit time-integration methods are used, have the same system of ODEs as a result of spatial discretization. Therefore, the stability of the ODE problem, which can only be achieved by scheme-adaptive spatial discretization of convective terms in a zone of dependence manner, should be satisfied not only for the MOL approach but also for the conventional methods (Oymak and Selçuk, 1997). However, the satisfaction of the stability of the ODE system as a result of spatial discretization does not necessarily mean that the final solution as a result of time-integration will also be stable. Therefore, in order to obtain absolutely stable and accurate solutions, the first condition is to satisfy the ODE problem stability, and the second is to use sophisticated (higher-order and implicit) time-integration methods. In this study, the first is achieved by utilizing a higher-order intelligent spatial discretization scheme (Oymak and Selçuk, 1997; Tarhan, 1999) described briefly in the following section. The second, time integration, is achieved by higher-order and stable schemes embedded in a quality ODE solver. In the present study, several ODE solvers were incorporated into the code and tested for efficiency providing the same order of accuracy. Details are given in the Time Integration and ODE Solvers sections.

Higher-order intelligent spatial discretization scheme

In the MOL, the spatial derivatives in governing equations are approximated by utilizing the general definition of the fourth-order, five-point Lagrange interpolation polynomial

$$y = \sum_{i=1}^{5} \prod_{\substack{j=1 \\ i \neq j}}^{5} \frac{x - x_j}{x_i - x_j} y_i. \tag{22}$$

The discretization procedure is applied on both radial and axial directions after the transformation of the concerned dependent variable, say $\varphi(r, z, t)$, into its pseudo-one-dimensional form, $\varphi(\bar{x}, t)$, in a certain spatial direction $\bar{x}$, for a value of remainder direction, by transforming the two-dimensional array into a one-dimensional array. Here $\bar{x}$ stands for either the radial direction r or the axial direction z, depending on the direction in which discretization of the dependent variable is carried out. Details of the transformation and discretization procedure can be found in Oymak and Selçuk (1996).

The five-point Lagrange interpolation polynomial makes it possible to investigate the solutions of the governing equations by a higher order discretization scheme on both uniform and non-uniform grid topologies. When converting the PDE system into an ODE initial value problem by discretizing the spatial derivatives, the convective terms should be approximated in such a way that the resulting system of ODEs should be stable according to the linear stability theory (Schiesser, 1991). This is achieved by a multidimensional intelligent scheme (scheme-adaptive method) based on the choice of biased-upwind or biased-downwind stencils for the convective derivatives according to the sign of the coefficient of the associated derivative (Oymak and Selçuk, 1997; Tarhan and Selçuk, 2001).

Time integration

Substitution of the approximations of spatial derivatives into the governing equations yields the follow-

ing set of ODEs in time:

$$\frac{d\bar{\phi}}{dt} = f\left(\bar{\phi}\right), \qquad \bar{\phi} = \left(\phi_1, \phi_2, \ldots, \phi_{NEQN}\right), \quad (23)$$

where $NEQN$ denotes the number of equations to be solved. Since the accuracy of the time integration scheme as well as of the spatial discretization scheme is very important in essentially unsteady flows, the resulting system with suitable initial and boundary conditions is integrated using a state-of-the-art ODE solver.

As mentioned before, the most important feature of the MOL approach is that not only does it have the simplicity of the explicit methods but also the superiority of the implicit ones as the higher-order implicit time integration methods are employed in the solution of the resulting system of ODEs. There exist many efficient and reliable stiff ODE solvers in the open literature, but it is important to utilize one of them considering not only the accuracy concept but also the execution time. Therefore, a numerical comparison between six ODE solvers has been carried out in terms of CPU time. The next section describes these ODE solvers.

ODE solvers

The stiff ODE concept is well established and a variety of efficient and quite reliable stiff ODE solvers are available. However, it is very important to select a suitable integrator based on the type and dimension of the physical system and on the desired level of accuracy. In this study, six ODE solvers, namely explicit RKF45 (Schiesser, 1991), semi-implicit ROW-methods of order 4 embedded in the ODE solver ROWMAP (Weiner *et al.*, 1996), implicit Adams-Moulton and BDF methods embedded in the ODE solvers LSODE, LSODES, LSODA (Radhakrishnan and Hindmarsh, 1993) and VODE (Brown *et al.*, 1989) were tested in terms of CPU time providing the same order of accuracy. As these solvers are fully documented in the above references, they are not described here.

Parallel Implementation

Parallel computing is the concept of executing a program on multiprocessors by dividing a large problem into smaller jobs. By allocating n processors, it is expected for the same program to run n times faster than with a single processor. Recently, it was recognized that very large scientific and engineering problems can only be solved in an acceptable time with a reasonable cost by parallel computing (Thevenin *et al.*, 1996; Douglas *et al.*, 1997; Erşahin, 2001; Erşahin *et al.*, 2003). Parallel computing is applicable to many scientific and engineering problems not only in terms of speeding up the program but also in handling large memory requirements in an economical way.

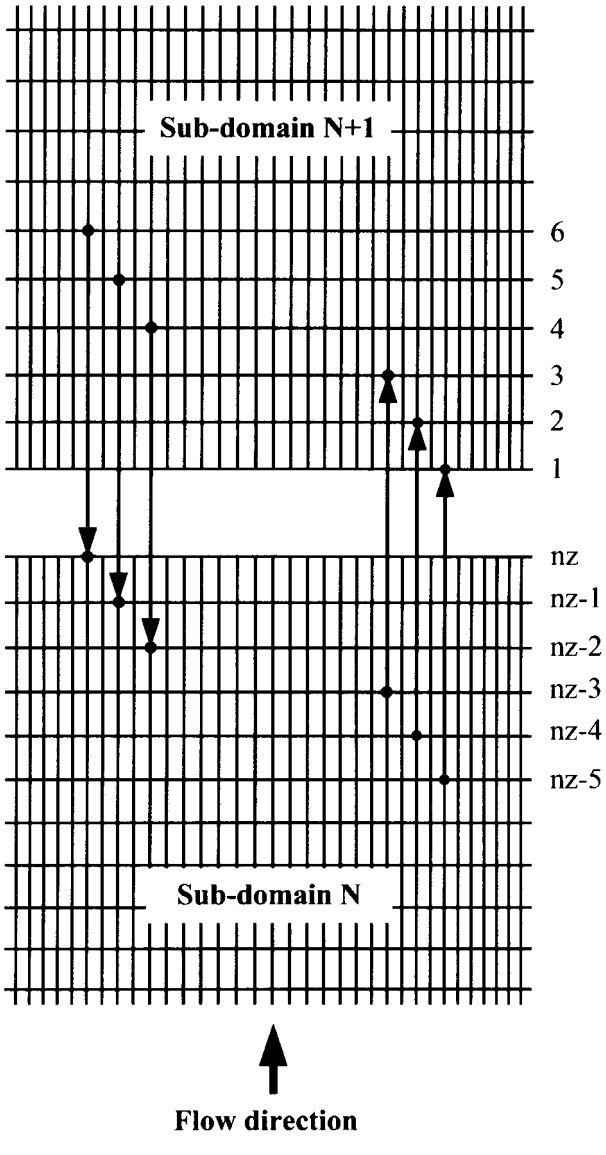

Figure 1. Domain decomposition and boundary exchange.

In this study, for parallel implementation a domain decomposition technique is utilized by means of overlapping boundaries at the intergrid regions

to provide information exchange between the sub-domains (Figure 1). As can be seen from the figure, boundary overlapping includes the exchange of the final three grid lines from sub-domain N to $N+1$ and of the first three grid lines from sub-domain $N+1$ to N in order to preserve the accuracy of the spatial discretization. Because a 5-point discretization scheme based on biased-upwind or biased-downwind stencils is utilized in this study, at least three points from the neighboring sub-domains should be exchanged. The algorithm is based on the master-slave paradigm, where the master process generates the grid structure, sets initial and physical boundary conditions, decomposes the domain into sub-domains, sets the type of each sub-domain (*type*) and sends the related information to the slave processes. The type of sub-domain is determined according to the position of that sub-domain relative to the others. In other words, the leftmost sub-domain is assigned as *type*-1, which includes an inlet boundary condition and an imaginary boundary condition; the rightmost sub-domain is assigned as *type*-3, which includes an imaginary boundary condition and an outlet boundary condition; and the intermediate sub-domains are assigned as *type*-2, which include two imaginary boundary conditions. The slave processes perform the calculations for the sub-domains assigned to them according to the types set by the master process, advance in time and exchange necessary information between each other at user defined time steps (*tp*) and send transient results to the master process for the development of the transient solution until the steady state is reached. Parallel virtual machine (PVM) (IBM, PVM, 1996) message passing software is employed for the information exchange between the sub-domains.

In parallel computing, performance can be defined by speed-up and efficiency. Speed-up is the most frequently used parameter for defining the performance of a parallel program and is the ratio of the total execution time of the serial program to that of the parallel one. Efficiency is the ratio of speed-up to the number of processors. In serial programming, the execution time is equal to the computation time, whereas in parallel programming it includes both the computation time and the communication time. The requirements for obtaining an efficient parallel algorithm are as follows: assigning a different number of grid points to each sub-domain (load balancing), assigning neighboring sub-domains to the same computers in order to reduce communication time during the calculations, and keeping the critical regions where sharp gradients exist completely in one sub-domain to satisfy the stability and high efficiency of the code.

Results

Both sequential and parallel codes were applied to the prediction of an axisymmetric laminar coflowing methane-air confined diffusion flame that has been studied experimentally by Mitchell *et al.* (1980).

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

$$CH_4 + 2O_2 + N_2 \rightarrow 2H_2O + CO_2 + N_2. \qquad (24)$$

Description of a test case

The combustion system, the vertical, cylindrical diffusion flame burner (Mitchell *et al.*, 1980) shown in Figure 2, consists of two concentric tubes of radii R_i and R_o through which the fuel and air issue, respectively. The flow rates of fuel and air were selected to produce a steady diffusion flame. A Pyrex glass cylinder serves to shield the flame and defines the boundaries of the system. The radius of the inner fuel jet, R_i, and the radius of the outer oxidizer jet, R_o, are 0.635 cm and 2.54 cm, respectively. The height of the burner, Z, is 30 cm. The test conditions to be employed are as follows:

1. Inner tube (Fuel Side):

   Inflow axial velocity: $u_F = 4.5$ cm/s

   Inflow radial velocity: $v_F = 0.0$ cm/s

   Temperature: $T_F = 298$ K

2. Outer tube (Air Side):

   Inflow axial velocity: $u_A = 9.88$ cm/s

   Inflow radial velocity: $v_A = 0.0$ cm/s

   Temperature: $T_A = 298$ K

The shield temperature, T_{wall}, is kept constant at 298 K.

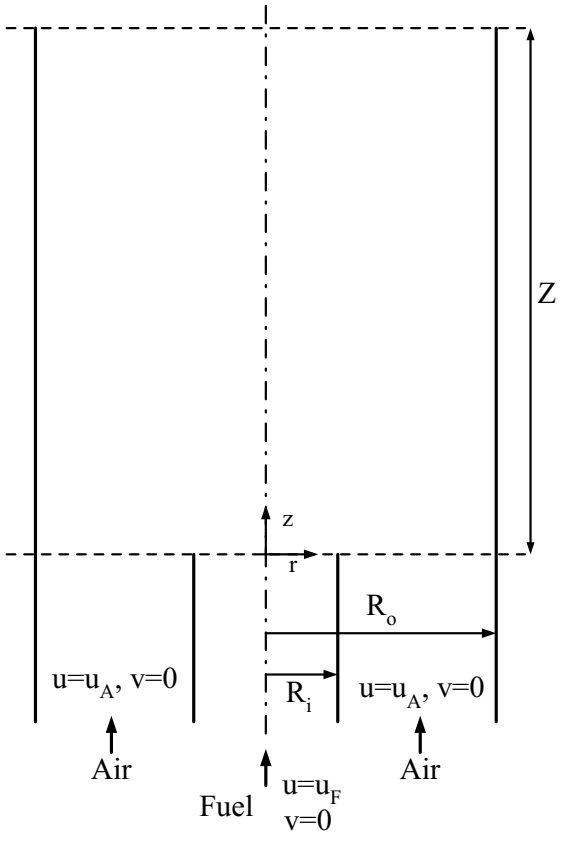

Figure 2. Schematic diagram of confined axisymmetric laminar diffusion flame burner.

First, the grid resolution requirement is determined by running the sequential code for two grid resolutions. Grid points were nonuniformly distributed along the radial and axial directions, allowing a higher number of grid points in the entrance region, along the intersection line of the fuel and air inlet and near the wall, where high gradients are expected. It was found that the results generated by 121×121 are almost identical to those on 151×151 grid points. Furthermore, the CPU time required by coarser resolution is 1.5 times more than that required by the finer grid resolution. Therefore, 121×121 was selected as the grid resolution to be employed.

ODE solver test

All ODE solvers use variable step size control, which requires the choice of a relative error tolerance *rtol* and an absolute error tolerance *atol*. In all ODE solvers except for RKF45, both *rtol* and *atol* were prescribed as 10^{-3}. These tolerances caused RKF45 to fail integration. Therefore, 10^{-4} was used for *rtol* and *atol* in RKF45. All the input parameters required by the ODE solvers are given in Table 2.

Figure 3 shows the comparison of the CPU times of the solvers. ROWMAP shows the best performance with a CPU time 2.5 times faster than RKF45, 4 times faster than LSODE, LSODES and VODE, and 5.5 times faster than LSODA. An examination of the times required by the LSODE family solvers shows that there is not much difference between the performances of LSODE and LSODES, but there is a large gap between them and LSODA. VODE also shows the same performance as LSODE and LSODES, since all are based on BDF methods. After this test, it was decided to use ROWMAP for time integration in the codes. All the results presented below were obtained using ROWMAP.

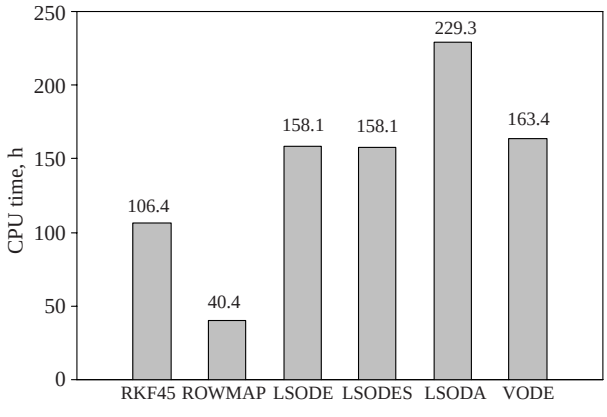

Figure 3. Comparison of CPU times of ODE solvers.

Sequential vs. parallel code performance test

The performance of the sequential and parallel codes was investigated by examining two performance criteria: speed-up and efficiency. For comparative purposes, computations were carried out using the same grids, time steps, final time and user-specified error tolerances required by the ODE solver for both codes. A static load balancing strategy based on a priori knowledge of the runtime behavior of the parallel code was used to improve the performance of the parallel code by minimizing the runtime of the calculation. First, the parallel code was run for an equally distributed number of grid points in each sub-domain, and their CPU times were measured. Then, depending on the CPU times, a different number of grid points was assigned to each sub-domain.

Table 2. Input parameters used by ODE solvers.

Parameters	RKF45	ROWMAP	LSODE	LSODES	LSODA	VODE
itol	-	0	1	1	1	1
rtol	1.0×10^{-4}	1.0×10^{-3}	1.0×10^{-3}	1.0×10^{-3}	1.0×10^{-3}	1.0×10^{-3}
atol	1.0×10^{-4}	1.0×10^{-3}	1.0×10^{-3}	1.0×10^{-3}	1.0×10^{-3}	1.0×10^{-3}
iflag	1	-	-	-	-	-
hs	-	0.0	-	-	-	-
ijacv	-	0	-	-	-	-
ifdt	-	0	-	-	-	-
iout	-	0	-	-	-	-
ifcn	-	0	-	-	-	-
rpar	-	0.0	-	-	-	-
ipar	-	0	-	-	-	-
iopt	-	-	0	0	0	0
itask	-	-	1	1	1	1
istate	-	-	1	1	1	1
mf	-	-	20	20	-	10
jt	-	-	-	-	5	-

All runs were carried out on a dedicated system containing two dual-processor personal computers installed with Linux. One of them has two Pentium III-700 MHz processors with 512 MB RAM, and the other has two Pentium III-1000 MHz processors with 1024 MB RAM. They are on an ethernet network through a 100 Mbps switch.

Figure 4 shows the effect of the number of processors on speed-up. The straight line indicates the ideal case that could be obtained if all the processors were identical and if there were no communication losses during the information exchange between the sub-domains. The circular and square symbols represent the speed-up obtained by running the parallel code with and without load balancing, respectively. As can be seen from the figure, speed-up values for each case increase with the number of processors but with a decreasing slope. The speed-up obtained with load balancing is better than that obtained without.

The variation of efficiency with the number of grid processors is illustrated in Figure 5. Efficiency decreases with an increasing number of processors due to the use of processors with different speeds and the communication delay between the processors. The efficiency obtained with load balancing is higher than that obtained without.

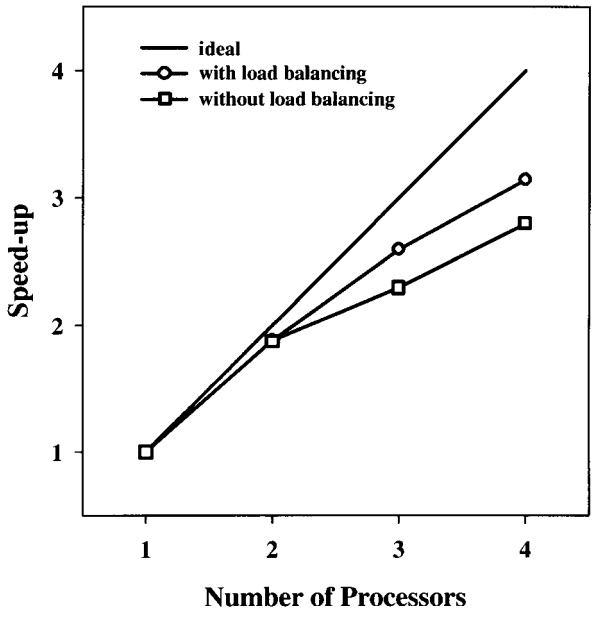

Figure 4. Variation of speed-up with number of processors.

Steady state results

The radial profiles of the species mole fractions predicted by the present study and by Xu and Smooke (1993) are plotted against the experimental data at three axial locations ($z = 1.2, 2.4, 5.0$ cm) in Figure 6. The flame-sheet model predictions of both studies compare favorably with the experimental

data for most of the species. The numerical solutions tend to give somewhat higher CH_4 mole fractions in the fuel side. Because the penetration of oxidizer into the the fuel side is not allowed in the flame-sheet model, the discrepancy of O_2 on the fuel side is due to this approximation. The higher H_2O concentration in the experimental data is caused by the moisture carried by the re-entrant flow from the exit due to recirculation in the burner. A comparison of species profiles obtained by the present study and by Xu and Smooke (1993) showed that the present predictions are in better agreement with the experimental data at the fuel side; however, the predictions of Xu and Smooke (1993) at the oxidizer side agreed more favorably with the data than those of the present study. This discrepancy may be due to the deployment solution of the governing equations in nonconservative and conservative forms and the numerical solution techniques utilized in these two studies.

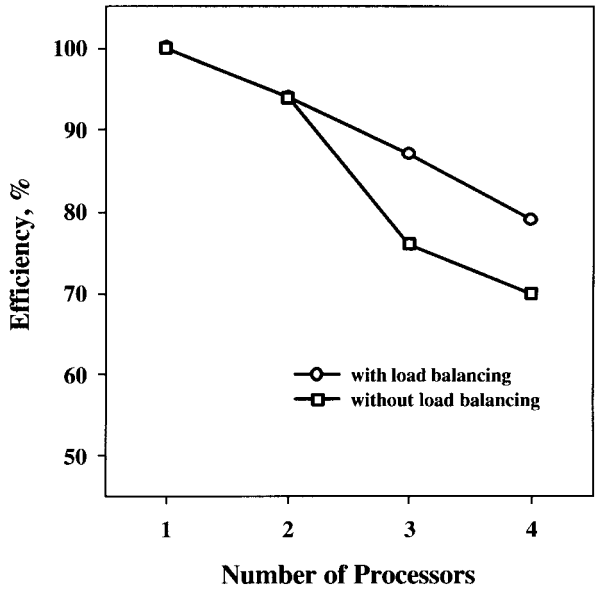

Figure 5. Variation of efficiency with number of processors.

The radial temperature profiles at two axial locations and along the centerline are illustrated in Figure 7. At the first axial location, the numerical solutions of both studies agree reasonably well with the experimental data on the oxidizer side, but they fail to match the experimental temperature on the fuel side. This may be caused by the infinite reaction approximation in the flame-sheet approximation. The

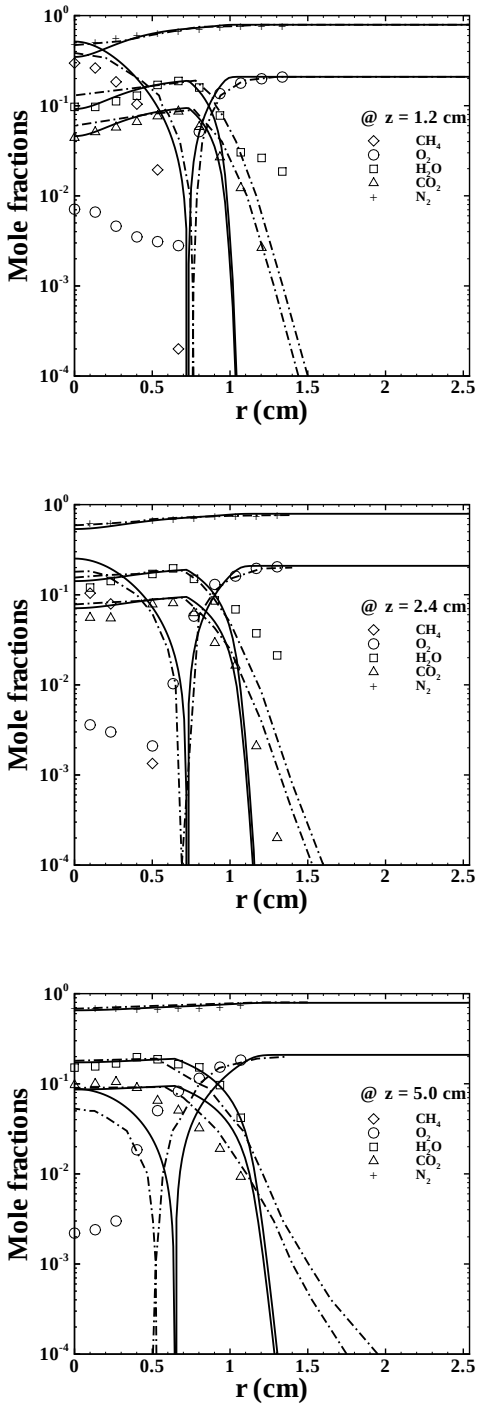

Figure 6. Radial profiles of the species mole fractions at three axial locations.
Symbols: experimental data ([Mitchell *et al.*, 1980]);
−·−·− : predictions by Xu and Smooke (1993);
———— : present study.

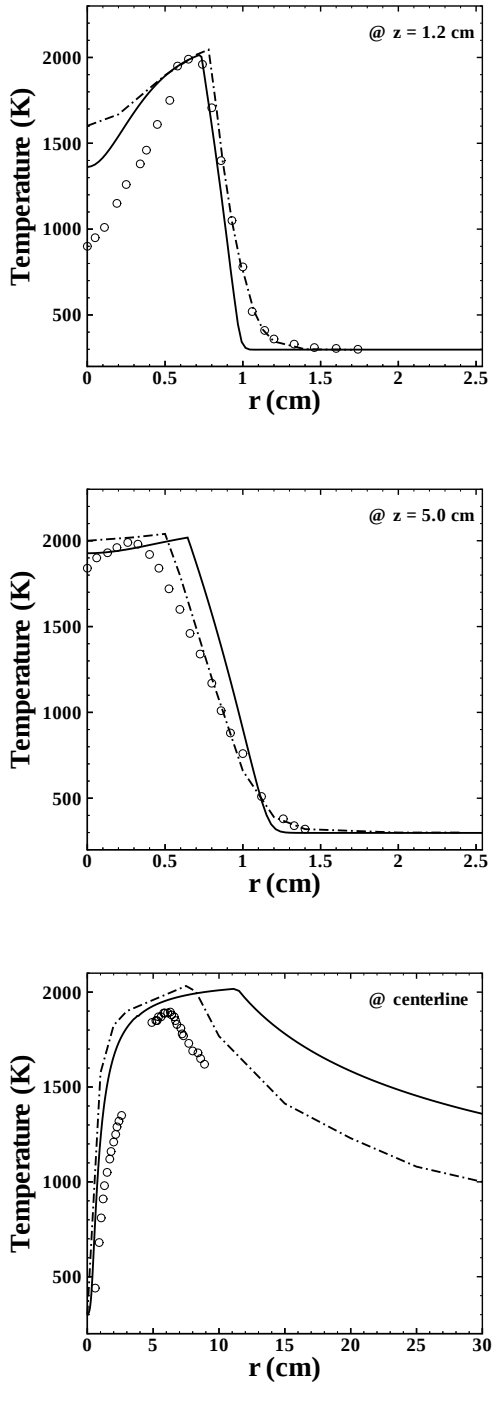

Figure 7. Radial temperature profiles at two axial locations and axial temperature profile along the centerline.
Symbols: experimental data ([Mitchell *et al.*, 1980]);
−·−·− : predictions by Xu and Smooke (1993);
———— : present study.

predictions of the present study agree better with the experimental data than those of Xu and Smooke (1993) at the fuel side and vice versa at the oxidizer side. The same trend can be observed at the other axial location ($z = 5.0$ cm). The last illustration in Figure 7 shows the profile of the temperature along the centerline. There exists disagreement between the computed profiles and experimental data. The numerical solution gives a much higher temperature. The present predictions are closer to the experimental data in the flame; however, the predictions of Xu and Smooke (1993) are in better agreement with the data outside the flame. The flame length, defined as the height of the maximum temperature at the centerline, is predicted as 12 cm and 7.6 cm by the present study and Xu and Smooke (1993), respectively, which are higher than the experimental value of 5.8 cm. To date, the numerical solutions have always overpredicted the temperatures both inside and outside the flame (Mitchell *et al.*, 1980; Smooke *et al.*, 1989; Xu *et al.*, 1993).

The radial profiles of axial velocity for three axial locations are shown in Figure 8. The predicted velocity profiles severely overpredict the experimental results. This disagreement between the numerical and experimental data has been observed by Xu *et al.* (1993) as well, even though they were employing a detailed chemistry as a combustion model. In fact, this overprediction is physically consistent with the higher temperature predictions and a flame length that produces a larger buoyancy force and recirculation zone in the burner.

Unsteady state results

In order to show the predictive ability of the code for the transient flow field, the time development of the velocity, streamline pattern and temperature for the impulsively started flow are exhibited in Figures 9 and 10. The entire domain is filled initially with room temperature air and the velocity is set to zero everywhere. Fuel and air start to enter from their inlets. Figure 9 shows the time development of the axial velocity by color contours and the streamline pattern for 121×121 grid points. As soon as the flow starts, the velocity increases in the inlet region along the centerline due to the increase in temperature and vortices begin to form due to the confined system. As time progresses the flame propagates to the exit of the burner with an accelerating velocity and flow starts to separate downstream yielding two large recirculation cells that are established between

the hot surface and the cooler shield. High velocity gradients appear in the whole domain ranging in axial velocity from −50 cm/s to 400 cm/s. At steady state these recirculation cells take their final forms remaining in the system. Air is entrained into the system at the shield outlet to balance the momentum of the inlet fuel and air streams along with the frictional losses at the shield wall. The presence of these recirculation cells reduces the total area available for the flow of the combustion gases and hence the velocities are increased due to the combined effects of natural convection and a reduced flow area.

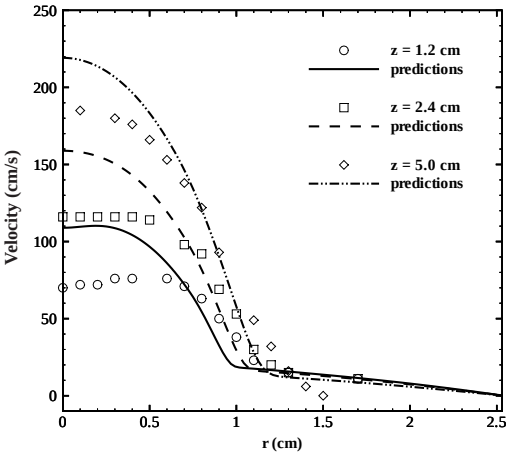

Figure 8. Radial profiles of axial velocity at $z = 1.2$, $z = 4.0$, $z = 5.0$ cm.
Symbols: experimental data ([Mitchell *et al.*, 1980]).

The time development of the temperature field is shown in Figure 10 for 121×121 grid points. As soon as the flow starts, the reaction starts immediately and a high temperature region forms extending from the boundary of the fuel and oxidizer jets to the symmetry axis. Thereafter, as the flow moves downstream, the temperature field broadens to the walls due to recirculations. Recirculations cause the fuel and oxidizer to expand in the direction of the shield. After a distance from the inlet, fuel cannot exist and therefore temperature starts to decrease along the downstream. At steady state it takes its final form yielding extremely high temperature gradients in the system. In the vicinity of the inlet, temperature rises from 298 K to nearly 2000 K in approximately 1 cm. In the radial direction, it is observed that temperature rapidly decreases and ultimately approaches the inlet value.

Preliminary calculations indicate that the flame-sheet model provides reasonable results for the simulation of laminar flames. However, for accurate predictions, the incorporation of a detailed chemical reaction model is required. The flame sheet model can be used to generate an initial solution estimate for detailed chemistry model solutions.

Conclusions

A parallel CFD code based on the MOL approach was developed for the simulation of transient laminar two-dimensional reacting flows. The chemistry was modeled by the flame-sheet approximation. The predictive ability and accuracy of the code was tested by applying it to the prediction of a confined methane/air laminar diffusion flame and comparing its predictions with other numerical study and experimental data available in the literature. On the basis of these numerical simulations the following conclusions have been reached:

- The predicted velocity, temperature and major species concentrations are in reasonably good agreement with other numerical results and measurements.
- Among the several ODE solvers tested in the code, ROWMAP is superior in terms of execution time.
- A considerable decrease in the CPU time of the parallel code can be obtained when static load balancing is employed.
- The code is an accurate and efficient tool for the prediction and understanding of transient combustion systems.

This study constitutes the initial steps in the development of an efficient numerical scheme for the direct simulation of unsteady, multidimensional combustion with stiff detailed chemistry.

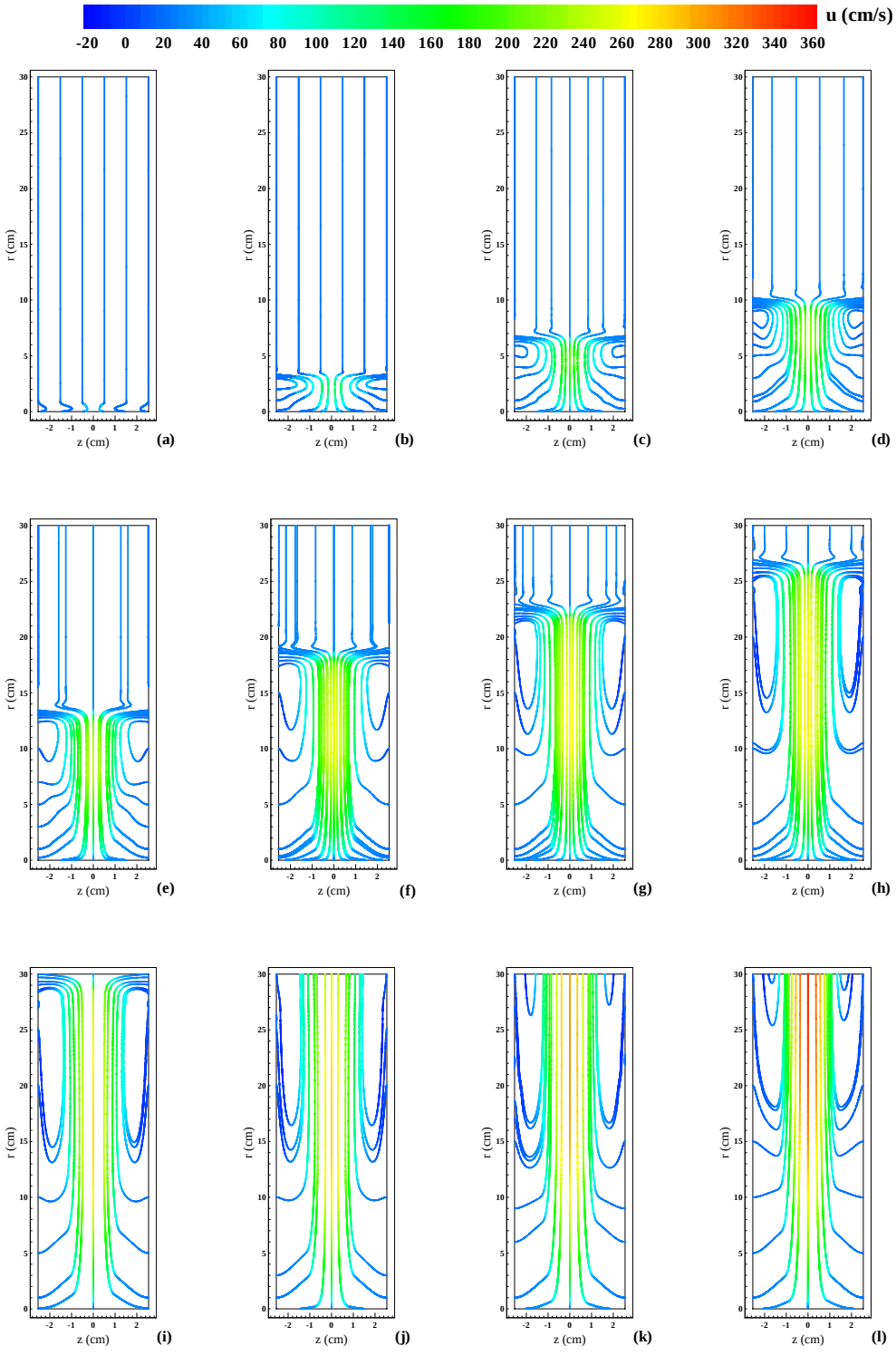

Figure 9. Time development of streamline pattern for 121×121 grid points; (a) @ 10 ms, (b) @ 50 ms, (c) @ 100 ms, (d) @ 150 ms, (e) @ 200 ms, (f) @ 280 ms, (g) @ 340 ms, (h) @ 400 ms, (i) @ 450 ms, (j) @ 500 ms, (k) @ 750 ms, (l) @ steady state.

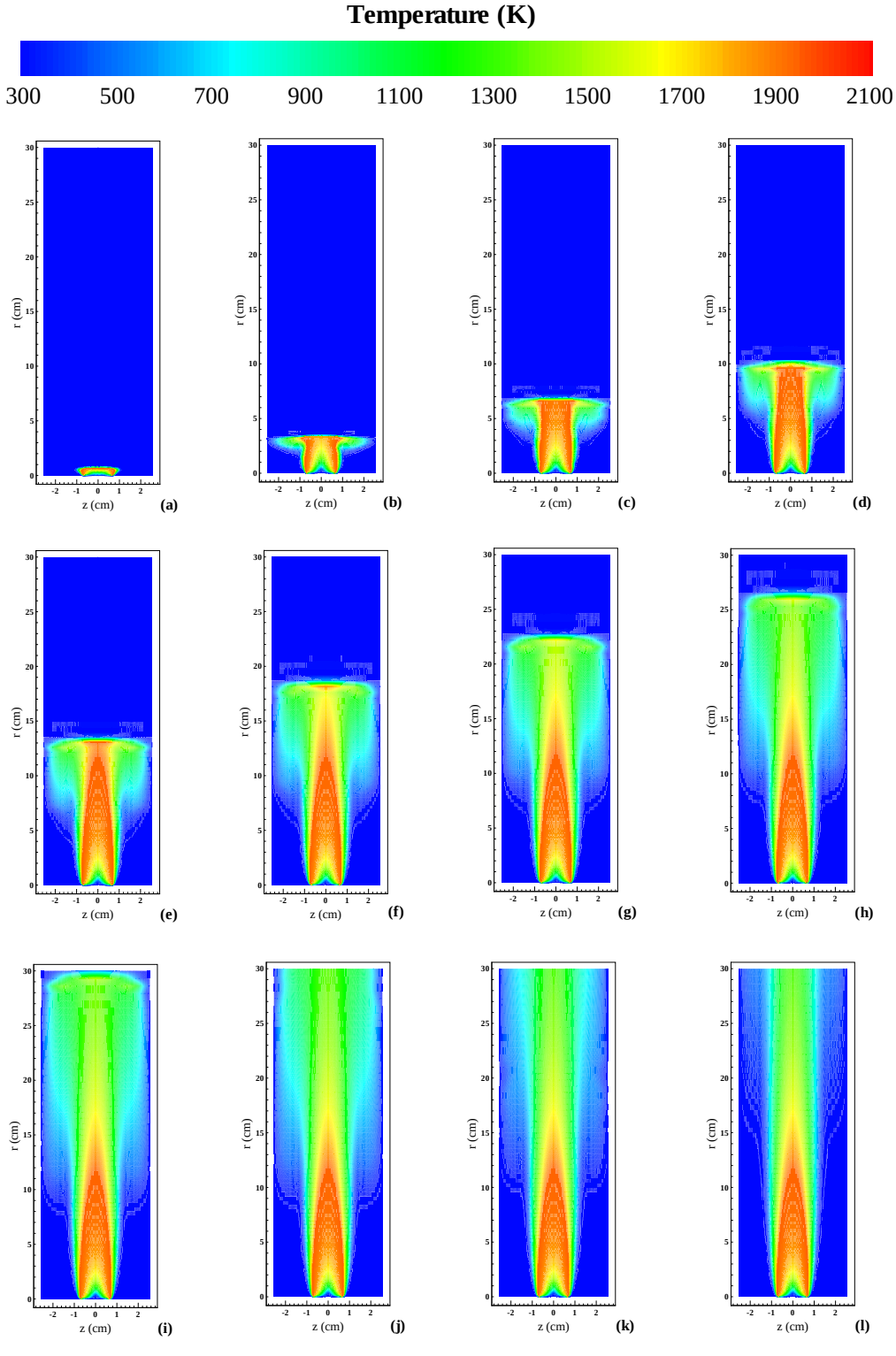

Figure 10. Time development of temperature for 121 × 121 grid points; (a) @ 10 ms, (b) @ 50 ms, (c) @ 100 ms, (d) @ 150 ms, (e) @ 200 ms, (f) @ 280 ms, (g) @ 340 ms, (h) @ 400 ms, (i) @ 450 ms, (j) @ 500 ms, (k) @ 750 ms, (l) @ steady state.

Acknowledgments

This study was performed as part of TÜBİTAK/MİSAG Project MİSAG-CNRS-1 on "Numerical Simulation of Lean Premixed and Prevaporized Turbulent Spray Combustion", and Middle East Technical University research funding project BAP-2002-03-04-02 on "Application of PARMOLS4ME Computational Fluid Dynamics Code to Non-Reacting/Reacting Thermal Flow Problems". All support is gratefully acknowledged.

Nomenclature

A	air
C_p	specific heat
D	mixture diffusivity
f	flame front
F	fuel
g	gravity force
IR	number of grid points in radial direction
JZ	number of grid points in axial direction
$\dot{m}$	mass flowrate
N	inert gas
Q	heat release per unit mass of the fuel
p	pressure
P	product
r	radial coordinate
R	radius
S	conserved scalar
t	time
T	temperature
u	axial velocity
v	radial velocity
W	molecular weight
x	independent variable
X	oxidizer
y	dependent variable
Y	mass fraction
z	axial coordinate
Z	height

Greek Letters

λ	thermal conductivity
μ	mixture viscosity
ν	stoichiometric coefficient
ρ	mass density
ϕ	dependent variable
φ	dependent variable transformed into 1-D array
Δ	macroscopic differential operator

Superscripts

n	index for time-level

Subscripts

i	species index; inner
i, j	index for spatial coordinate of interest
in	inlet
I	inner jet
o	initial; outer
O	outer jet

References

Becker, R., Braack, M. and Rannacher, R., "Numerical Simulation of Laminar Flames at Low Mach Number by Adaptive Finite Elements", Combust. Theory Modelling, 3, 503–534, 1999.

Brown, P.N., Byrne, G.D. and Hindmarsh, A.C., "VODE: A Variable Coefficient ODE Solver", SIAM J. Sci. Stat. Comput., 10, 1038–1051, 1989.

Coelho, P.J. and Pereira, J.C.F., "Calculation of a Confined Axisymmetric Laminar Diffusion Flame Using a Local Grid Refinement Technique", Combust. Sci. and Technol., 92, 243–264, 1993.

Coelho, P.J. and Peters, N., "Unsteady Modelling of a Piloted Methane/Air Jet Flame Based on the Eulerian Particle Flamelet Model", Combust. Flame, 124, 444–465, 2001.

Cuenot, B. and Poinsot, T., "Effects of Curvature and Unsteadiness in Diffusion Flames. Implications for Turbulent Diffusion Combustion", Twenty-Fifth Symp. (Int.) on Combustion, The Combustion Institute, 1383–1390, 1994.

Desjardin, P.E. and Frankel, S.H., "Linear-Eddy Modeling of Nonequilibrium Turbulent Reacting Flows with Nonpremixed Reactant", Combust. Flame, 109, 471–487, 1997.

Douglas, C.C., Ern, A. and Smooke, M.D., "Multigrid Solution of Flame Sheet Problems on Serial and Parallel Computers", Parallel Alg. and Appl., 10, 225–236, 1997.

Erşahin, C., Parallelization of a Transient Navier-Stokes Code Based on Method of Lines Solution, Master's Thesis, Chemical Engineering Department, Middle East Technical University, 2001.

Erşahin, C., Tarhan, T., Tuncer, İ.H. and Selçuk, N., "Parallelization of a Transient Navier-Stokes

Code based on Method of Lines Solution", Int. J. Comput. Fluid Dynamics, (in press).

IBM, PVM, IBM PVMe for AIX User's Guide and Subroutine Reference. Version 2, Release 2, IBM, New York, 2^{nd} Edition, 1996.

Kee, R.J., Dixon-Lewis, G., Warnatz, J., Coltrin, M.E. and Miller, J.A., "A FORTRAN Computer Code Package for the Evaluation of Gas-Phase, Multicomponent Transport Properties", Technical Report SAND86-8246, Sandia National Laboratories, 1986.

Kee, R.J., Rupley, F.M., Meeks, E. and Miller, J.A., "CHEMKIN-III: A FORTRAN Chemical Kinetics Package for the Analysis of Gas-Phase Chemical and Plasma Kinetics", Technical Report SAND96-8216, Sandia National Laboratories, 1996.

Kronenburg, A. and Bilger, R.W., "Modelling Differential Diffusion in Nonpremixed Reacting Turbulent Flow: Model Development", J. Comput. Phys., 166, 195–227, 2001.

Mitchell, R.E., Sarofim, A.F. and Clomburg, L.A., "Experimental and Numerical Investigation of Confined Laminar Diffusion Flames", Combust. Flame, 37, 227–244, 1980.

Neuber, A., Krieger, G., Tacke, M., Hassel, E. and Janicka, J., "Finite Rate Chemistry and *NO* Mole Fraction in Non-Premixed Turbulent Flames, Combust. Flame, 113, 198–211, 1998.

Oymak, O. and Selçuk, N., "Method of Lines Solution of Time-Dependent Two-Dimensional Navier-Stokes Equations", Int. J. Numer. Meth. Fluids, 23, 455–466, 1996.

Oymak O. and Selçuk, N., "Transient Simulation of Internal Separated Flows Using an Intelligent Higher-Order Spatial Discretization Scheme", Int. J. Numer. Meth. Fluids, 24, 759–769, 1997.

Oymak, O., Method of Lines Solution of Time-Dependent 2D Navier-Stokes Equations for Incompressible Separated Flows, PhD Thesis, Chemical Engineering Department, Middle East Technical University, 1997.

Pember, R.B., Almgren, A.S., Bell, J.B., Colella, P., Howell, L.H. and Lai, M.F., "A Higher-Order Projection Method for the Simulation of Unsteady Turbulent Nonpremixed Combustion in an Industrial Burner" Transport Phenomena in Combustion, Taylor & Francis, 1996.

Poinsot, T., "Using Direct Numerical Simulation to Understand Premixed Turbulent Combustion", Twenty-Sixth Symp. (Int.) on Combustion, The Combustion Institute, 219–232, 1996.

Radhakrishnan, K. and Hindmarsh, A.C., "Description and Use of LSODE, the Livermore Solver for Ordinary Differential Equations", Technical Report UCRL-ID-113855, Lawrence Livermore National Laboratory, NASA, 1993.

Schiesser, W.E., The Numerical Method of Lines: Integration of Partial Differential Equations, Academic Press, New York, 1991.

Selçuk, N. and Oymak, O., "A Novel Code for the Prediction of Transient Flow Field in a Gas Turbine Combustor Simulator", NATO/RTO Meeting Proceedings 14, AVT Symposium on Gas Turbine Engine Combustion, Emissions and Alternative Fuels, 11/1-10, 1999.

Smooke, M.D., Mitchell, R.E. and Keyes, D.E., "Numerical Solution of Two-Dimensional Axisymmetric Laminar Diffusion Flames", Combust. Sci. and Technol., 67, 85–122, 1989.

Tarhan, T., Numerical Simulation of Flow through Inlet Flues of Heat Recovery Steam Generators, Master's Thesis, Chemical Engineering Department, Middle East Technical University, 1999.

Tarhan, T. and Selçuk, N., "Method of Lines for Transient Flow Fields", Int. J. Comput. Fluid Dynamics, 15, 309–328, 2001.

Thevenin, D., Behrendt, F., Maas, U., Przywara, B. and Warnatz, J., "Development of a Parallel Direct Simulation Code to Investigate Reactive Flows", Computers Fluids, 25, 485–496, 1996.

Vervisch, L. and Poinsot, T., "Direct Numerical Simulation of Non-Premixed Turbulent Flames", Ann. Rev. Fluid Mech., 30, 655–691, 1998.

Weiner, R., Schmitt, B.A. and Podhaisky, H., "ROWMAP – a ROW-Code with Krylov Techniques for Large Stiff ODEs", Technical Report 39, FB Mathematik und Informatik, Universitate Halle, 1996.

Williams, F.A., Combustion Theory, Benjamin/Cummings, New York, 1985.

Xu, Y. and Smooke, M.D., "Application of a Primitive Variable Newton's Method for the Calculation of an Axisymmetric Laminar Diffusion Flame", J. Comput. Phys., 104, 99–109, 1993.

Xu, Y., Smooke, M.D., Lin, P. and Long, M.B., "Primitive Variable Modeling of Multidimensional Laminar Flames", Combust. Sci. and Technol., 90, 289–313, 1993.

Yamashita, H., Kushida, G. and Takeno, T., "A Numerical Study of the Transition of Jet Diffusion Flames", Proc. R. Soc. Lond. A, 431, 301–314, 1990.

Zhou, X., Sun, Z., Brenner, G. and Durst, F., "Combustion Modeling of Turbulent Jet Diffusion H_2/Air Flame with Detailed Chemistry", Int. J. Heat and Mass Transfer, 43, 2075–2088, 2000.