AIAA 2013-2433

Parallel Implicit Adaptive Mesh Refinement Scheme for Unsteady Fully-Compressible Reactive Flows

S. A. Northrup, * and C. P. T. Groth†
University of Toronto Institute for Aerospace Studies
4925 Dufferin Street, Toronto, Ontario, Canada, M3H 5T6

An accurate and robust parallel implicit adaptive mesh refinement (AMR) algorithm is proposed and described for the prediction of unsteady behaviour of laminar flames. The scheme is applied to the solution of the system of the partial-differential equations governing time-dependent, three-dimensional, compressible laminar flows for reactive thermally perfect gaseous mixtures. A high-resolution finite-volume spatial discretization procedure is used to solve the conservation form of these equations on body-fitted multi-block hexahedral mesh. A local preconditioning technique is used to remove numerical stiffness and maintain solution accuracy for low-Mach-number, nearly incompressible flows. A flexible block-based octree data structure has been developed and is used to facilitate automatic solution-directed mesh adaptation according to physics-based refinement criteria. The data structure also enables an efficient and scalable parallel implementation via domain decomposition. The parallel implicit formulation makes use of a dual-time-stepping like approach with an implicit second-order backward discretization of the physical time, in which a Jacobian-free inexact Newton method with a preconditioned generalized minimal residual (GMRES) algorithm is used to solve the system of nonlinear algebraic equations arising from the temporal and spatial discretization procedures. An additive Schwarz global preconditioner is used in conjunction with block incomplete LU type local preconditioners for each sub-domain. The Schwarz preconditioning and block-based data structure readily allow efficient and scalable parallel implementations of the implicit AMR approach on distributed-memory multi-processor architectures. Numerical results for steady and unsteady laminar co-flow diffusion and premixed methane-air flames demonstrate the capabilities of the proposed approach for a range of reactive-flow applications. The scheme is shown to accurately predict key characteristics of the diffusion flames. For a premixed flame under terrestrially gravity, the scheme is also shown to accurately predict the frequency of the natural buoyancy induced oscillations.

I. Introduction and Motivation

Numerical methods have become an essential tool for investigating a wide range of combustion phenomena. However despite the significant advances in numerical methods and computer hardware, obtaining accurate and reliable solutions can still place severe demands on available computational resources. Many approaches have been taken to reduce the computational costs of simulating combusting flows. One successful approach is to make use of solution-directed mesh adaptation, such as the adaptive mesh refinement refinement (AMR) algorithms developed for aerospace applications.[1–6] Computational grids that automatically adapt to the solution of the governing equations are very effective in treating problems with disparate length scales, providing the required spatial resolution while minimizing memory and storage requirements. A second approach for coping with the computational cost of reacting flow prediction is to apply a domain decomposition procedure and solve the problem in a parallel fashion using multiple processors. Large massively parallel distributed-memory computers have the potential to provide many fold increases in processing power and memory resources beyond those of conventional single-processor computers and would therefore

*PhD Candidate, UTIAS, northrup@utias.utoronto.ca.
†Professor, UTIAS, groth@utias.utoronto.ca, Senior Member AIAA.

provide an obvious avenue for greatly reducing the time required to obtain numerical solutions of combusting flows, as a number of studies have already shown.

Northrup and Groth,[7] Gao and Groth,[8–11] and Gao *et al.*[12] have combined these two numerical approaches, producing a parallel AMR method for steady non-premixed laminar and turbulent combusting flows having both two-dimensional (planar and axisymmetric) and fully three-dimensional geometries. For time-invariant flows, a preconditioned nonlinear multigrid algorithm with multi-stage semi-implicit time marching scheme as a smoother was proposed by Gao *et al.*[12] to obtain converged steady-state solutions to the governing partial differential equations. Although accurate solutions were obtained, the approach was found not to be optimal, as in many cases, a large number of multigrid cycles and solution residual evaluations were required to obtain steady-state flame solutions. Moreover, because of these difficulties and other issues, it was felt that a multigrid approach would not be well suited for unsteady reactive flow applications.

In order to reduce the time to achieve a solution and deal with the numerical stiffness of unsteady, three-dimensional, reactive flow problems, a parallel implicit AMR scheme is proposed and developed herein in which a block-based AMR approach is combined with Newton's method. In the proposed approach, which is based on extensions of the earlier work by Northrup and Groth[13] for two-dimensional laminar reactive flows, Newton's method is used to solve the system of nonlinear algebraic equations arising from a cell-centred, density-based, finite-volume, spatial discretization procedure coupled with an implicit time-marching scheme applied to the integral form of the governing equations for three-dimensional, fully-compressible, laminar, reactive flows. The spatial discretization is applied to each hexahedral computational cell of a multi-block body-fitted mesh, with the latter allowing solution-directed mesh adaptation via a block-based AMR technique.[10,11] A Riemann-solver based flux-function[14] is combined with a local low-Mach number preconditioning technique[15] to provide both an accurate and robust upwind discretization, as well as alleviate the numerical difficulties that arise with the low-Mach-number, nearly incompressible flow regimes that characterize the laminar reactive flows of interest. A preconditioned generalized minimal residual (GMRES) method is used to solve the resulting system of linear equations at each step arising from the application of Newton's method. An additive Schwarz preconditioner is used in combination with local block incomplete lower-upper (BILU) preconditioning to improve performance of this linear iterative solver so as to make it of practical use. The unconditional stability of the implicit time-marching procedure leads to a robust scheme for which time step selection is dictated solely by accuracy considerations and not those of stability. Moreover, the Schwarz preconditioning and block-based AMR readily allow efficient and scalable parallel implementations of the implicit time-marching approach on distributed-memory multi-processor computers. The algorithm targets solution of large-scale problems using 10,000-30,000 processor cores.

In what follows, the proposed parallel implicit AMR scheme is described in detail. This includes a discussion of the mathematical and physical modelling used to describe the reactive laminar flows of interest, details of the proposed finite-volume method and parallel implicit time-marching scheme, a discussion of the parallel performance of the overall solution method, as well as a description of the results of the application of the methodology to several laminar flames. The proposed parallel implicit AMR method is shown to be particularly well suited for predicting, in a reliable and efficient fashion, complex unsteady reactive flows containing widely disparate spatial and temporal scales as encountered in many combustion processes and demonstration of these capabilities is achieved here by through the application of the proposed scheme to both unsteady laminar non-premixed and premixed flames. In particular, the numerical results demonstrate the promise of the proposed scheme for application to the prediction of combustion instabilities and thermo-acoustic phenomena in practical combustor configurations.

II. Mathematical Description of Laminar Reactive Flows

A. Navier-Stokes Equations for a Compressible Reactive Mixture

For the laminar reactive flows of interest here, the Navier-Stokes equations for a thermally-perfect, compressible, gaseous, reactive mixture, which can be expressed using tensor notation as

$$\frac{\partial}{\partial t}(\rho) + \frac{\partial}{\partial x_i}(\rho u_i) = 0\,, \tag{1}$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j + \delta_{ij} p - \tau_{ij}) = \rho g_i\,, \tag{2}$$

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x_i}\left[(\rho E + p)\,u_i - \tau_{ij}u_j + q_i\right] = \rho u_i g_i \tag{3}$$

$$\frac{\partial}{\partial t}(\rho Y_k) + \frac{\partial}{\partial x_i}(\rho Y_k u_i + \mathcal{J}_{k,i}) = \dot{\omega}_k\,, \tag{4}$$

are used to represent the transport and combustion of the gaseous fuels and oxidizers. Equations (1)–(3) reflect the conservation of mass, momentum, and energy for the mixture, ρ is the mixture density, u_i is the mixture velocity, E is the total specific energy of the mixture given by $E = e + \frac{1}{2}u_i u_i$, e is the specific internal energy, p is the mixture pressure, τ_{ij} is the fluid stress tensor for the mixture, q_i is the heat flux vector, and g_i is the acceleration due to gravitational forces. Equation (4) is the mass concentration equation for species k, where Y_k is the species mass fraction, $\check{\mathcal{J}}_{k,i}$, is the mass flux of species k due to diffusion processes, and $\dot{\omega}_k$ is the time rate of change of the species concentration due to finite-rate chemistry. For a thermally perfect mixture, it follows that $e = \sum_{k=1}^{N} Y_k(h_k + \Delta h^0_{f,k}) - p/\rho$, where h_k and $\Delta h^0_{f,k}$ are the sensible enthalpy and heat of formation for species k, respectively, N is the number of species, and the ideal gas equation of state for the mixture is given by $p = \sum_{k=1}^{N} \rho Y_k R_k T$, where R_k is the species gas constant and T is the mixture temperature.

Note that most solution techniques to date for reactive flows are based on low-Mach number forms of the Navier-Stokes equations. While accounting for the variable-density nature of such flows, they are not capable of capturing transient variations in the pressure field associated with acoustic phenomena. The use of the compressible form of the Navier-Stokes equations readily allows for the often large density variations associated with combusting flows as well as the accurate prediction of both high-speed combustion and thermo-acoustic phenomena. Nevertheless, care must be taken to deal properly with the low-Mach-number limit for low-speed flows.

B. Weak Conservation Form of Governing Equations

The Navier-Stokes equations given by Eqs. (1)–(4) above can be re-expressed in the following general weak conservation form using matrix-vector notation:

$$\frac{\partial \mathbf{U}}{\partial t} + \vec{\nabla}\cdot\vec{\mathbf{F}} = \frac{\partial \mathbf{U}}{\partial t} + \vec{\nabla}\cdot\vec{\mathbf{F}}_{\mathrm{H}}(\mathbf{U}) + \vec{\nabla}\cdot\vec{\mathbf{F}}_{\mathrm{E}}\left(\mathbf{U}, \vec{\nabla}\mathbf{U}\right) = \mathbf{S} \tag{5}$$

where $\mathbf{U}$ is the vector of conserved solution variables given by

$$\mathbf{U} = \left[\ \rho,\quad \rho u_i,\quad \rho E,\quad \rho Y_k\ \right]^{\mathrm{T}}, \tag{6}$$

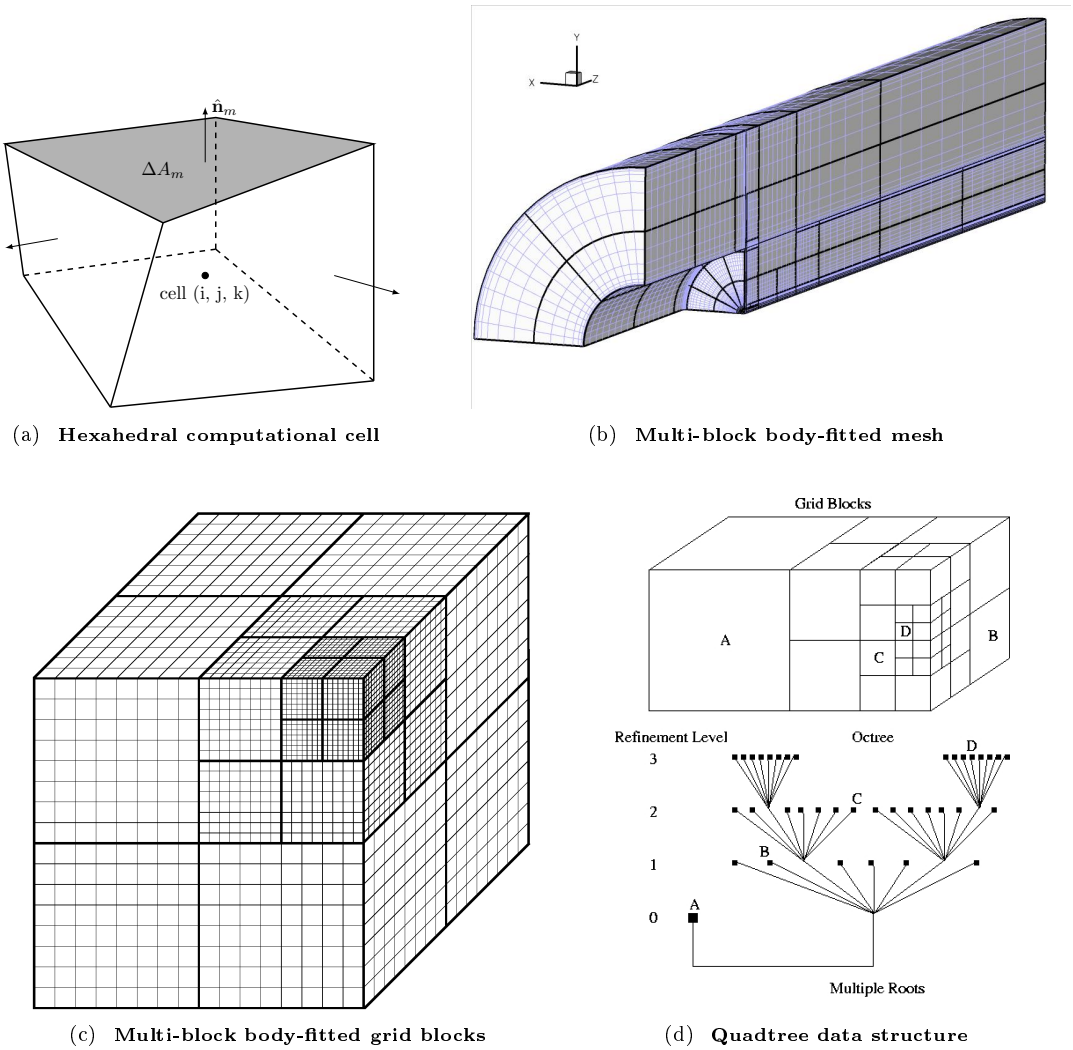
and $\vec{\mathbf{F}}$ is the solution flux dyad. The flux dyad can be decomposed into two components and written as $\vec{\mathbf{F}} = \vec{\mathbf{F}}_{\mathrm{H}} + \vec{\mathbf{F}}_{\mathrm{E}}$ where $\vec{\mathbf{F}}_{\mathrm{H}} = \vec{\mathbf{F}}_{\mathrm{H}}(\mathbf{U})$ contains the hyperbolic or inviscid components of the solution fluxes $\vec{\mathbf{F}}_{\mathrm{E}} = \vec{\mathbf{F}}_{\mathrm{E}}(\mathbf{U}, \vec{\nabla}\mathbf{U})$ contains the elliptic or viscous components of the fluxes. The latter depend on both the solution and its gradient.

C. Thermodynamic and Transport Properties and Chemical Kinetics

For the numerical results presented herein, thermodynamic and molecular transport properties of each mixture component needed to complete the mathematical description of a reactive mixture are prescribed using the database compiled by Gordon and McBride.[16,17] Perfect mixture rules are used to determine mixture thermodynamic properties. In the case of the mixture transport coefficients, Wilke's mixture rule[18] is adopted to evaluate the mixture viscosity. Similarly, the thermal conductivity for the mixture is found using Mason and Saxena's[19] mixture rule.

For the purposed of this numerical study, attention is restricted to the combustion of gaseous fuels and methane is used as the representative fuel. Although full or detailed chemical reaction mechanisms are available for describing methane-air (CH_4-air) fuel-oxidizer combustion processes (e.g., refer to the detailed GRI-Mech 3.0 chemical kinetic mechanism[20]), further computational convenience is achieved by employing a the simplified one-step chemical reaction mechanism of Westbrook and Dryer[21] to represent the finite-rate reaction processes. This 1-step, 5-species model represents the combustion of methane by means of a single forward reaction given by

$$CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O \tag{7}$$



(a) **Hexahedral computational cell**

(b) **Multi-block body-fitted mesh**

(c) **Multi-block body-fitted grid blocks**

(d) **Quadtree data structure**

Figure 1. Block-based AMR scheme for multi-block body-fitted mesh showing: (a) hexahedral computational cell; (b) example of a multi-block body-fitted mesh; (c) multi-block body-fitted grid blocks following three levels of refinement; and (d) corresponding grid blocks and octree data structure.

and just five species are tracked: methane (CH_4), oxygen (O_2), carbon dioxide (CO_2), water(H_2O), and nitrogen (N_2). Nitrogen is assumed to be inert. The one-step mechanism obviously significantly reduces the complexity of the reactive flow modelling yet retains many computational features of the more complete problem. Additionally, many key physical features of methane combustion, such as the premixed laminar flame speed, can be described reasonably accurately with a well-tuned model. Future follow-on studies will consider the use of more detailed chemical kinetic mechanisms for a wider range of both gaseous and liquid fuels.

III. Parallel Implicit Finite-Volume Scheme with AMR

A. Finite-Volume Spatial Discretization

Numerical solutions of Eq. (5) are sought here by applying a cell-centred, finite-volume, spatial discretization procedure. A semi-discrete form of the governing equations can be derived from the application of the finite-volume method to the integral form of Eq. (5) for cell (i, j, k) of a three-dimensional multi-block mesh composed of hexahedral computational cells as shown in Figs. 1(a) and 1(b). The resulting semi-discrete

form can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\left[\mathbf{U}_{i,j,k}(t)\right] = -\frac{1}{V_{i,j,k}}\sum_{l=1}^{N_f}\left(\vec{\mathbf{F}}\cdot\vec{n}\,\Delta A\right)_{i,j,k,l} + \mathbf{S}_{i,j,k} = -\mathbf{R}_{i,j,k}(\mathbf{U})\,, \tag{8}$$

where each cell has $N_f = 6$ faces and standard mid-point rule quadrature is used to evaluate the solutions fluxes through each face, providing second-order spatial accuracy. The variables ΔA and $\vec{n}$ represent the surface areas and unit vectors normal to the cell faces, respectively. Formally, $\mathbf{U}_{i,j,k}$, is the average value of the conserved solution vector for cell (i,j,k) at a given instance in time as defined by the integral expression

$$\mathbf{U}_{i,j}(t) = \frac{1}{V_{i,j,k}}\iiint_{V_{i,j,k}} \mathbf{U}(\vec{X},t)\;\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z\,, \tag{9}$$

where $\vec{X} = [x,y,z]$ and $V_{i,j}$ is the volume of the hexahedral cell of interest. Similarly, $\mathbf{S}_{i,j,k}$ is given by

$$\mathbf{S}_{i,j,k}(t) = \frac{1}{V_{i,j,k}}\iiint_{V_{i,j,k}} \mathbf{S}(\vec{X},t)\;\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z\,, \tag{10}$$

and approximated to second order using mid-point quadrature as $\mathbf{S}_{i,j,k} \approx \mathbf{S}(\mathbf{U}_{i,j,k})$. The vector, $\mathbf{R}_{i,j,k}$, is referred to here as the solution residual vector.

B. Low-Mach-Number Local Preconditioning

As noted previously, the use of fully-compressible form of the Navier-Stokes equations allows for the automatic treatment of density variations, high-speed flows, and transient combustion phenomena with thermo-acoustic coupling. However, the application of a fully-compressible flow solver to a nearly incompressible flow can be computational problematic. For many laminar flames, the flow Mach number, M, can be very small with M $\approx$ 0.001–0.003 and this value tends to decrease with increasing pressure. Hence, $|\vec{u}| + a \gg |\vec{u}|$ and the resulting coupled non-linear system of ordinary differential equations (ODEs) of the form given by Eq. (8) arising from an upwind-based finite-volume discretization procedure is both ill-conditioned and can introduce excessive numerical dissipation. The local preconditioning technique proposed by Weiss and Smith[15] is applied here to alleviate these numerical difficulties in the low-Mach-number limit for nearly incompressible flows.

The low-Mach-number local preconditioning technique of Weiss and Smith[15] is applied to the conservation form of the governing Navier-Stokes equations by introducing a pseudo time, τ, and the preconditioning matrix, $\mathbf{\Gamma}$, and re-writing the equations as

$$\mathbf{\Gamma}\frac{\partial \mathbf{U}}{\partial \tau} + \frac{\partial \mathbf{U}}{\partial t} + \vec{\nabla}\cdot\vec{\mathbf{F}} = \mathbf{S}\,. \tag{11}$$

The corresponding semi-discrete form is then taken to be

$$\mathbf{\Gamma}_{i,j,k}\frac{\mathrm{d}}{\mathrm{d}\tau}\left[\mathbf{U}_{i,j,k}\right] + \frac{\mathrm{d}}{\mathrm{d}t}\left[\mathbf{U}_{i,j,k}\right] = -\mathbf{R}_{i,j,k}(\mathbf{U})\,, \tag{12}$$

where $\mathbf{\Gamma}_{i,j,k}$ is the preconditioning matrix that is applied locally to hexahedral cell (i,j,k). Steady-state solutions in pseudo time of Eq. (12) are then sought for which $\partial\mathbf{U}/\partial\tau = 0$. Appropriate choices of the preconditioning matrix, $\mathbf{\Gamma}$, do not affect the desired steady-state solution satisfying $\partial\mathbf{U}/\partial t + \vec{\nabla}\cdot\vec{\mathbf{F}} = \mathbf{S}$ but alter the eigenstructure of the governing partial differential equations such that the numerical stiffness and required dissipation can be controlled, making the reactive flow problem more tractable. Refer to the original paper of Weiss and Smith[15] for the choice and details of the preconditioning matrix.

C. Numerical Flux Evaluation

The solution of Eq. (8) requires the evaluation of the moment fluxes at the mid-points of each cell face. Upwind values for the hyperbolic fluxes, $\vec{\mathbf{F}}_{\mathrm{H}}\cdot\vec{\mathbf{n}}$, for each cell face, l, are determined from the solution of a

Riemann problem. Given the left and right values of the solution vectors, $\mathbf{U}_l$ and $\mathbf{U}_r$, in the cells just to the left and right of the face quadrature point, the hyperbolic flux is given by

$$\vec{\mathbf{F}}_{\mathrm{H}} \cdot \vec{n} = \mathcal{F}_{\mathrm{H}}(\mathbf{U}_l, \mathbf{U}_r, \vec{n}) \,, \tag{13}$$

where the numerical flux function $\mathcal{F}_{\mathrm{H}}$ is evaluated by solving a Riemann problem in a direction defined by the normal to the cell face with initial data $\mathbf{U}_l$ and $\mathbf{U}_r$. The approximate linearized Riemann solver of Roe[14] is used here with an extension to account for mixture composition. This flux function is further modified by applying the preconditioning matrix to the upwind dissipation term such that the numerical flux takes the form

$$\mathbf{F}_{\mathrm{H}}\left(\mathbf{U}_{\mathrm{L}}, \mathbf{U}_{\mathrm{R}}\right) = \frac{1}{2}\left(\mathbf{F}_{\mathrm{H,R}} + \mathbf{F}_{\mathrm{H,L}}\right) - \frac{1}{2}\boldsymbol{\Gamma}(\mathbf{U}_*) \; \left|\mathbf{A}_{\Gamma}(\mathbf{U}_*)\right| \; \left(\mathbf{U}_{\mathrm{R}} - \mathbf{U}_{\mathrm{L}}\right) \,, \tag{14}$$

where $\mathbf{A}_{\Gamma}(\mathbf{U}) = \boldsymbol{\Gamma}^{-1}\mathbf{A}(\mathbf{U})$ and $\mathbf{A}(\mathbf{U})$ is the Jacobian of the inviscid flux with respect to the conserved solution variables. This preconditioning of the Roe flux provides an improved scaling of the dissipation terms in the low-Mach-number limit and avoids the introduction of excessive numerical dissipation. The left and right solution vectors, $\mathbf{U}_l$ and $\mathbf{U}_r$, of Eqs. (13) and (14) are determined via a least-squares, piece-wise, limited, linear solution reconstruction procedure in conjunction with the limiter of Venkatakrishnan.[22] The latter provides a second-order-accurate spatial discretization for smooth solutions.

The elliptic fluxes arising from thermal diffusion are evaluated using a second-order centrally weighted scheme of the form

$$\vec{\mathbf{F}}_{\mathrm{E}} \cdot \vec{n} = \mathcal{F}_{\mathrm{E}}(\mathbf{U}_m, \vec{\nabla}\mathbf{U}_m, \vec{n}) = \vec{\mathbf{F}}_{\mathrm{E}}(\mathbf{U}_m, \vec{\nabla}\mathbf{U}_m) \cdot \vec{n} \,, \tag{15}$$

where $\mathbf{U}_m$ is the interpolated value of the solution vector at the mid-point of the face or edge and the face gradient, $\vec{\nabla}\mathbf{U}_m$, is evaluated a weighted cell-face gradient approach of Mathur and Murthy.[23]

D. Dual-Time-Stepping-Like Approach and Implicit Time-Marching Scheme

The semi-discrete form of the governing equations given in Eq. (8) form a coupled set of non-linear ODEs. For unsteady flows, time-dependent solutions are obtained by employing a dual-time-stepping-like procedure.[24] In the proposed approach a modified residual is defined by

$$\mathbf{R}^*(\mathbf{U}) = \frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} + \mathbf{R}(\mathbf{U}) \,. \tag{16}$$

and steady-state solutions in pseudo time are sought to the ODEs

$$\boldsymbol{\Gamma}\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}\tau} + \mathbf{R}^*(\mathbf{U}) = 0 \,, \tag{17}$$

satisfying $\mathbf{R}^*(\mathbf{U}) = 0$. Application of an implicit second-order backward difference time-marching scheme with time step, Δt, to the physical time derivative yields

$$\mathbf{R}^*(\mathbf{U}^{(n+1)}) = \frac{3\mathbf{U}^{(n+1)} - 4\mathbf{U}^{(n)} + \mathbf{U}^{(n-1)}}{2\triangle t} + \mathbf{R}(\mathbf{U}^{(n+1)}) = 0 \,, \tag{18}$$

The preceding non-linear algebraic equations define the solution $\mathbf{U}^{(n+1)}$ given $\mathbf{U}^{(n)}$ and $\mathbf{U}^{(n-1)}$ at the previous time steps.

E. Block-Based Adaptive Mesh Refinement

Solution of Eq. (18) yields area-averaged solution quantities defined within hexahedral computational cells. These cells are embedded in structured, body-fitted, grid blocks consisting of $N_{\mathrm{cells}} = N_i \times N_j \times N_k$ cells as depicted in Fig. 1, where N_i, N_j and N_k are integers representing the number of cells in each logical coordinate direction of the grid block. The multi-block mesh and spatial discretization procedure described above readily allows for the application of solution-directed block-based AMR.[7,10–12] The block-based AMR algorithm developed here builds on the previous work by Gao *et al.*[9,11,12,25] for three space dimensions. In this previous work, the focus was primarily on refinement and the application was restricted to steady-state flow problems. Assumptions concerning grid block connectivity were made that restricted the generality of the approach, specifically the approach used for determining block connectivity was not sufficiently general

to allow for both refinement and coarsening dynamically in unsteady flow applications. A more generalized block connectivity and data structure has been developed here that allows for both refinement and coarsening as required for unsteady flows with a corresponding wider range of multi-block mesh topologies.

In the proposed AMR scheme, local refinement and coarsening of the mesh is carried out by division and merging of solution blocks, respectively, as depicted in Figs. 1(b) and 1(c). Standard restriction and prolongation operators are used to evaluate the solution on all blocks created by the coarsening and division processes. The solution-directed adaptation of the multi-block, body-fitted, hexahedral meshes is based on physics-based refinement criteria, such as the gradients of scalar quantities and the divergence and curl of vector quantities (blocks containing high and low values of the selected criteria are flagged for either refinement or coarsening). Furthermore, changes in mesh resolution at grid block interfaces are restricted to at most one level. A flexible block-based hierarchical octree data structure has been developed and is used to keep track of the mesh adaptation and connectivity between grid blocks, an example of which is given in Figs. 1(c) and 1(d). The octree data structure allows for general unstructured connectivity of the root grid blocks. Solution information is shared between adjacent blocks through the use of "ghost" or "halo" cells and the flux conservation properties of the finite-volume scheme are preserved by using the interface fluxes computed on refined blocks to correct the interface fluxes computed on adjacent coarser blocks. Refer to the recent papers by Northrup and Groth[7] and Gao and Groth,[10,11] and Gao *et al.*[12] for further details of the parallel AMR scheme.

F. Newton's Method

Newton's method is used for the solution of the coupled non-linear algebraic equations of Eq. 18 as defined by the solutions quantities in each computational cell of the grid. This requires the solution of the following linear system of equations

$$\left[\left(\frac{3}{2\triangle t}\right)\mathbf{I}+\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right]\triangle \mathbf{U}^{(n+1,k)}=\mathbf{J}\Delta \mathbf{U}^{(n+1,k)}=-\mathbf{R}^{*}(\mathbf{U}^{(n+1,k)})\,, \tag{19}$$

for the solution change $\Delta \mathbf{U}^{(n+1)}=\mathbf{U}^{(n+1)}-\mathbf{U}^{(n)}$ at time level n, at each Newton step or iteration level, k. Using the previous time step as the initial estimate, $\mathbf{U}^{(n+1,k=0)}=\mathbf{U}^{(n)}$, successively improved estimates for the solution, $\mathbf{U}^{(n+1,k)}$, are obtained by solving Eq. 19 at each step, k, of the Newton method, where $\mathbf{J}$ is the modified residual Jacobian. The improved approximation for the solution is then given by $\mathbf{U}^{n+1}=\mathbf{U}^{n}+\Delta \mathbf{U}^{n}$. The iterative procedure is repeated until an appropriate norm of the solution residual is sufficiently small, i.e., $||\mathbf{R}^{*}(\mathbf{U}^{(n+1,k+1)})||_2<\epsilon||\mathbf{R}^{*}(\mathbf{U}^{(n+1,k)})||_2$ where ϵ is some small parameter (typically, $\epsilon\approx 10^{-2}$–10^{-3}).

As noted above, each step of Newton's method requires the solution of a system of linear algebraic equations of the form

$$\mathbf{J}\mathbf{x}=\mathbf{b}\,. \tag{20}$$

This system is large, sparse, and non-symmetric and a preconditioned GMRES method[26,27] is used for its solution. In particular, a restarted version of the flexible GMRES algorithm, GMRES(m), is used, where m is the number of steps after which the GMRES algorithm is restarted. Application of this iterative technique leads to an overall solution algorithm with iterations within iterations: the "inner loop" iterations involving the solution of the linear system and the "outer loop" iterations associated with the solution of the nonlinear problem. An inexact Newton method is adopted here in which the inner iterations are not fully converged at each Newton step. The inner iterations are carried out only until $||\mathbf{R}^{*}+\mathbf{J}\Delta \mathbf{U}||_2\leq \zeta||\mathbf{R}^{*}||_2$, where ζ is typically in the range 0.01–0.5. As discussed by Dembo *et al.*,[28] an exact solution of the linear system is not necessary for rapid convergence of Newton's method.

Preconditioning is required for the GMRES algorithm to be effective. Right preconditioning of the form

$$(\mathbf{J}\mathbf{M}^{-1})(\mathbf{M}\mathbf{x})=\mathbf{b}\,, \tag{21}$$

is used here where $\mathbf{M}$ is the preconditioning matrix. An additive Schwarz global preconditioner with variable overlap[27,29,30] is used in conjunction with local BILU preconditioners for each sub-domain. The local preconditioner is based on a block ILU(f) or BILU(f) factorization of an approximate Jacobian for each subdomain. Here, f is the level of fill. This combination of preconditioning fits well with the block-based AMR described previously and is very compatible with domain decomposition methods, readily enabling parallel implementation of the overall Newton method. Rather efficient parallel implementations of implicit

algorithms via Schwarz preconditioning have been developed by Keyes and co-researchers and successfully applied to the prediction of transonic full potential, low-Mach-number compressible combusting, and three-dimensional inviscid flows.[29,31,32]

As the GMRES algorithm does not explicitly require the evaluation of the global Jacobian matrix, $\mathbf{J}$, a so-called "matrix-free" or "Jacobian-free" approach can be adopted and is used here. Numerical differentiation based on Fréchet derivatives is used to approximate the matrix-vector product $\mathbf{JM}^{-1}\mathbf{x}$ as follows:

$$\mathbf{JM}^{-1}\mathbf{x} \approx \frac{\mathbf{R}(\mathbf{U}+\varepsilon\mathbf{M}^{-1}\mathbf{x}) - \mathbf{R}(\mathbf{U})}{\varepsilon} + \frac{3\mathbf{M}^{-1}\mathbf{x}}{2\triangle t}, \tag{22}$$

where $\mathbf{R}(\mathbf{U}+\varepsilon\mathbf{M}^{-1}\mathbf{x})$ is the residual vector evaluated at some perturbed solution state and ε is a small scalar quantity. Although the performance of the Jacobian-free method is sensitive to the choice of ε, Neilsen *et al.*[33] have found that $\varepsilon = \varepsilon_\circ/||\mathbf{x}||_2^{1/2}$ seems to work well, with $\varepsilon_\circ \approx 10^{-8}$–$10^{-7}$.

G. Parallel Implementation

As indicated above, the multi-block, body-fitted, AMR scheme and and additive Schwarz global preconditioning technique used in the GMRES algorithm are well suited to the parallel implementation of the parallel implicit finite-volume AMR scheme on distributed-memory multi-processor architectures via domain decomposition. Because of the self-similar nature of the grid blocks, Domain decomposition is a achieved here by simply distributing the the blocks making up the computational mesh equally among available processors and/or processor cores, with more than one block permitted per core. A Morton ordering space filling curve is used to provide nearest-neighbour ordering of the solution blocks in the multi-block hexahedral AMR mesh for more efficient load balancing.[34] Parallel implementation of the overall scheme has been carried out using the MPI (message passing interface) library.[35,36] Message passing of information between processors is largely limited to the asynchronous communication of ghost-cell solution values as well as communication associated with the evaluation of norms of global residual vectors in the Newton method.

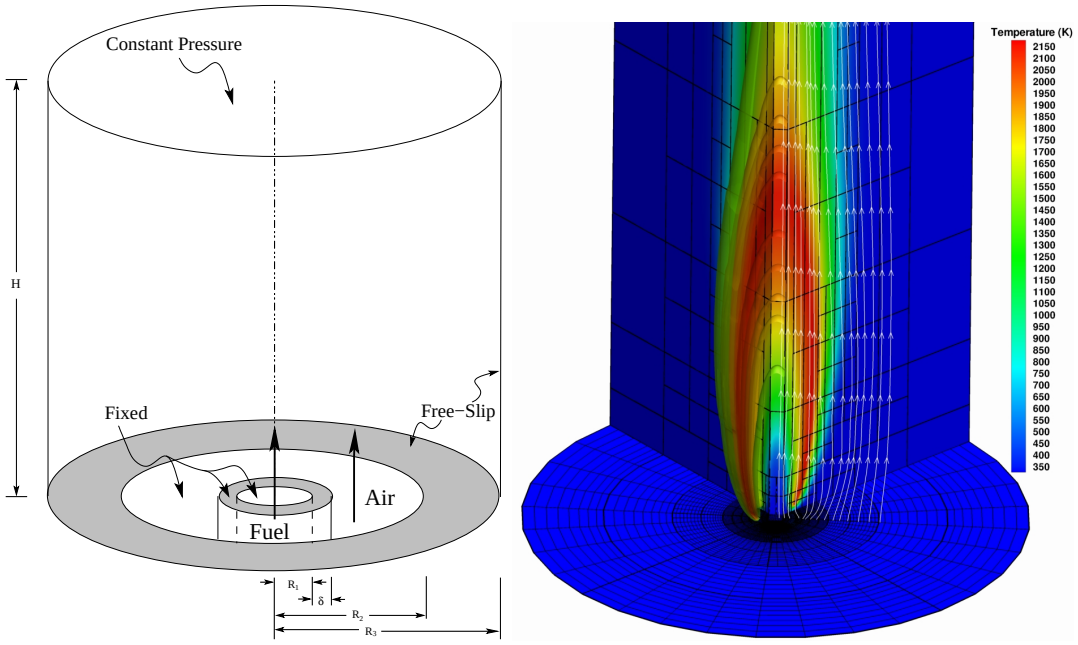
All of computations presented herein were performed on a high performance parallel cluster consisting of 3,780 Intel Xeon E5540 nodes with two quad-core 2.53 GHz Intel Xeon processors and 16GB RAM main memory per node. The cluster is connected with a high-speed low-latency InfiniBand switched fabric communications link.

IV. Numerical Results

A. Unsteady Laminar Diffusion Flame

The proposed parallel implicit AMR algorithm has been applied to the solution of a forced, time-dependent, methane-air co-flow laminar diffusion flame in three space dimensions. In particular, the solution of the flame studied by Day and Bell,[37] and Dworkin *et al.*[38] is considered. The flame boundary and initial conditions are the same as those used in the previous studies and are illustrated along with the burner geometry in Figure 2(a).

The computational domain is cylindrical in shape with a radius of $r = 5$ cm and height of $H = 10$ cm. The far-field or outer wall boundary is taken to be a free-slip boundary along which inviscid reflection boundary data is specified. The top or outlet of the flow domain is open to a stagnant reservoir at atmospheric pressure and temperature and Neumann-type boundary conditions are applied to all properties except pressure which is held constant. The bottom or inlet is subdivided into four regions. The innermost region ($r = 0$ to $r = 2$ mm) is the fuel inlet or jet, which injects a nitrogen diluted methane fuel mixture ($c_{\mathrm{CH_4}} = 0.5149$, $c_{\mathrm{N_2}} = 0.4851$, $c_{\mathrm{O_2}} = 0$, $c_{\mathrm{CO_2}} = 0$, $c_{\mathrm{CO}} = 0$, and $c_{\mathrm{H_2O}} = 0$) at 298 K with a parabolic axial velocity profile. The time-variation in the flame is produced by imposing a sinusoidal axial velocity fluctuation across the fuel jet, $v_z = 70(1 - r^2/R)(1 + \alpha \sin \omega t)$ cm/s where α is the velocity amplitude and ω is the frequency of oscillation. Matching the experiments of Dworkin *et al.*,[38] the velocity amplitude was set at 50% at a frequency of 20Hz. The next region ($\delta = 0.38$ mm) is a small gap associated with the annular wall separating the fuel and oxidizer. The third region ($r = 2.38$ mm to $r = 2.5$ cm) is the co-flowing oxidizer, in this case air at 298K ($c_{\mathrm{O_2}} = 0.232$, $c_{\mathrm{N_2}} = 0.768$, $c_{\mathrm{CH_4}} = 0$, $c_{\mathrm{CO_2}} = 0$, $c_{\mathrm{CO}} = 0$, and $c_{\mathrm{H_2O}} = 0$), with a uniform velocity profile of 0.35 m/s. The final outer region of the lower boundary ($r = 2.5$ cm to $r = 5$ cm) is again a far-field boundary along which free-slip boundary conditions are applied.



(a) Laminar co-flow flame geometry (b) Steady laminar co-flow flame

Figure 2. (a) Schematic showing the co-flow laminar diffusion flame domain geometry and boundary conditions; and (b) initial steady 3D methane-air co-flow laminar diffusion flame computational grid with 3 levels of refinement.

The solution domain is initialized with a uniform solution state corresponding to quiescent air at 298K, except for a thin region across the fuel and oxidizer inlets, which is taken to be air at 1500 K so as to ignite the flame. Additional details concerning the setup for this diffusion flame can be found in the papers by Day and Bell,[37] and Dworkin *et al.*[38]

The flame was calculated by first letting the steady-state flame develop and using 3 levels of mesh refinement (4 mesh levels) as shown in Figure 2(b), with the fuel flow velocity held constant at 70 cm/s, then four-full periods of the flame oscillation were calculated to avoid any non-periodic oscillations created during start-up. Newton's method was used with a GMRES tolerance of 0.1, ILU(0) preconditioner and at each time step the Newton iterations were converged two orders of magnitude, to a maximum of 10 Newton steps, where 5-7 is typical. The approximate Jacobian preconditioner was only updated for the first Newton step of each time-step, unless the number of GMRES iterations required increased. A fixed time-step of 0.05 ms was used during the unsteady calculation and mesh refinement was carried out every 2.5 ms based on the gradient of temperature. Figure 3 shows the resulting cross-section isotherms and adapting mesh at three times through the flames periodic fluctuation of 50 ms.

The predicted flames shape and structure matches well with the previous experimental results of Mohammed *et al.*[39] and numerical results of Dworkin *et al.*[38] The most significant difference is in the over prediction of temperature, however this is related to the use of the simplified non-reversible one-step reaction mechanism used to model the methane-air chemistry in the present work, whereas a more detailed mechanism was used in the previous studies.

B. Unsteady Premixed Diffusion Flame

The previous combustion case studied the well characterized and predictable behaviour of non-premixed flames where the fuel and oxidizer are mixed through diffusion. In a premixed flame the fuel and oxidizer are first mixed outside the combustion chamber and the resulting mixture is injected into the combustion chamber. Conical methane-air premixed laminar flames, unlike diffusion flames, do not typically form a steady flame, but instead have an oscillation or flickering of the flame tip. Buoyancy-induced interactions between the hot products and the cold environment produces velocity fluctuations in the reactants which

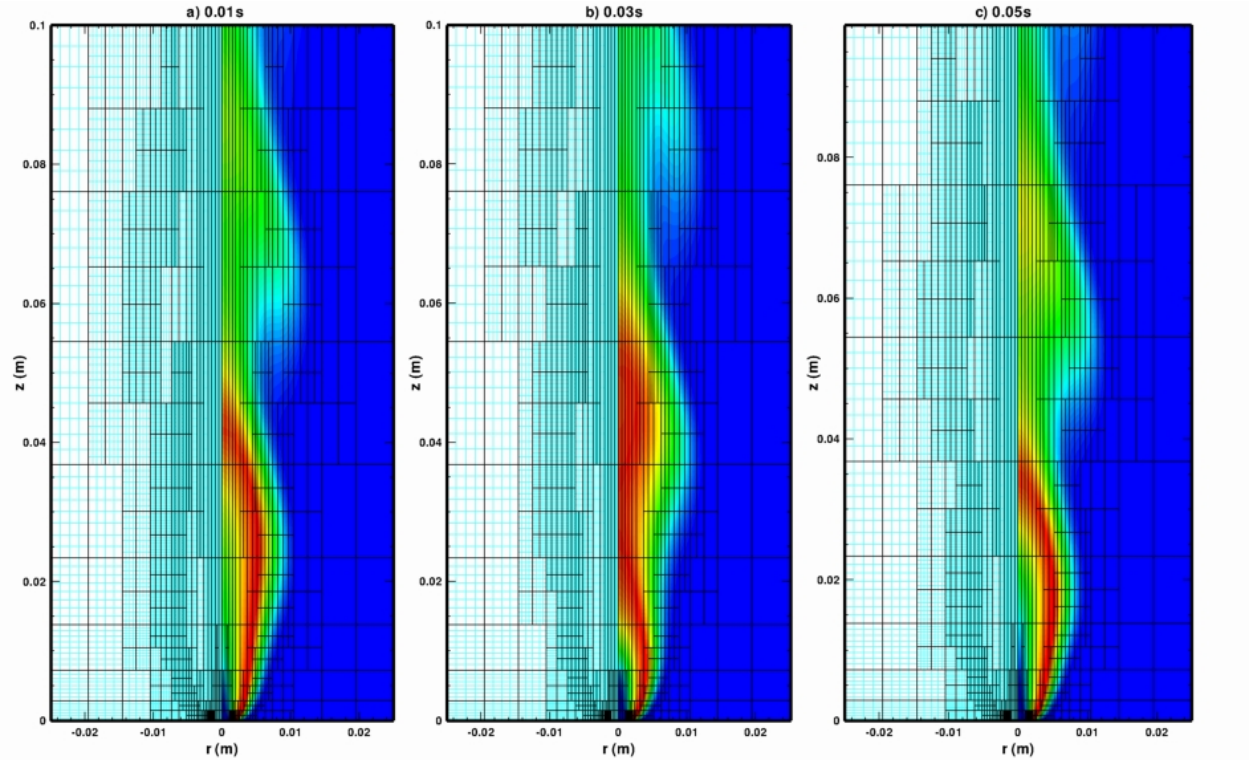


Figure 3. Time-varying methane-air co-flow laminar diffusion flame cross-section isotherms at three intervals calculated each with 3 levels of refinement.

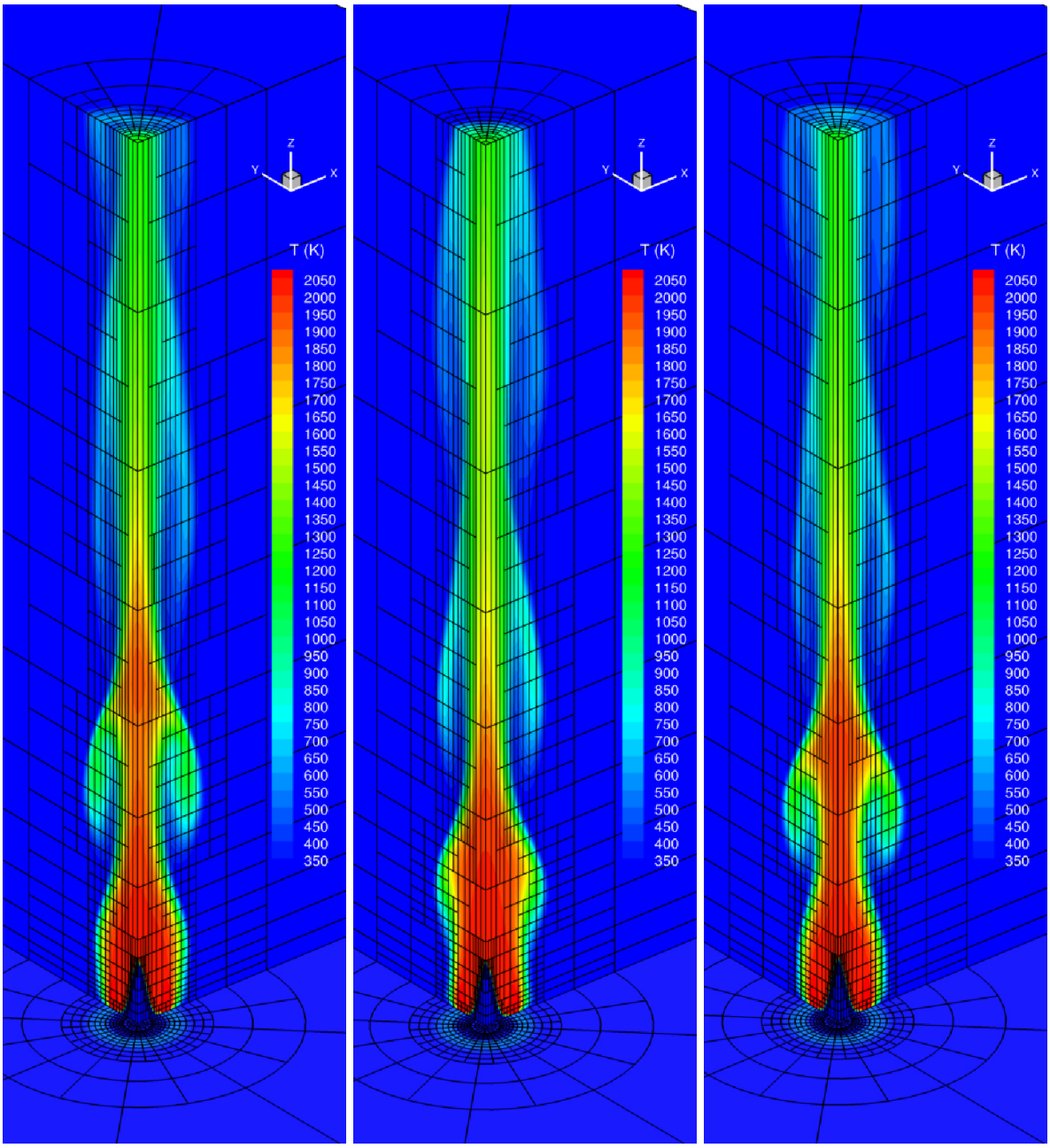
results in flickering of the flame tip.

A number of aspects of the flame/buoyancy coupling in conical and V-shaped premixed flames have been investigated experimentally such as the frequency as a function of flow velocity, pressure, and gravitational levels.[40,41] Kostiuk and Cheng[42,43] have also shown that premixed flame oscillations with characteristic frequencies of 10–20 Hz can be correlated to a wide range of system parameters. Recent experimental investigations into flame-intrinsic Kelvin–Helmholtz instabilities in premixed flames by Guahk *et al.*[44] in inverted conical flames showed that the Strouhal number, dimensionless frequency, can be correlated to the Richardson number.

Experimental and numerical comparisons were performed by Shepherd *et al.*[45] using the numerical approach developed Day and Bell[37] for laminar diffusion flames. In this research they found a quasi-periodic flame flicker could be developed given the correct conditions namely the correct balance of equivalence ratio and inlet mass flow. The premixed methane-air flame studied herein attempts to reproduce the quasi-periodic flame oscillations using the proposed time-accurate parallel implicit AMR algorithm.

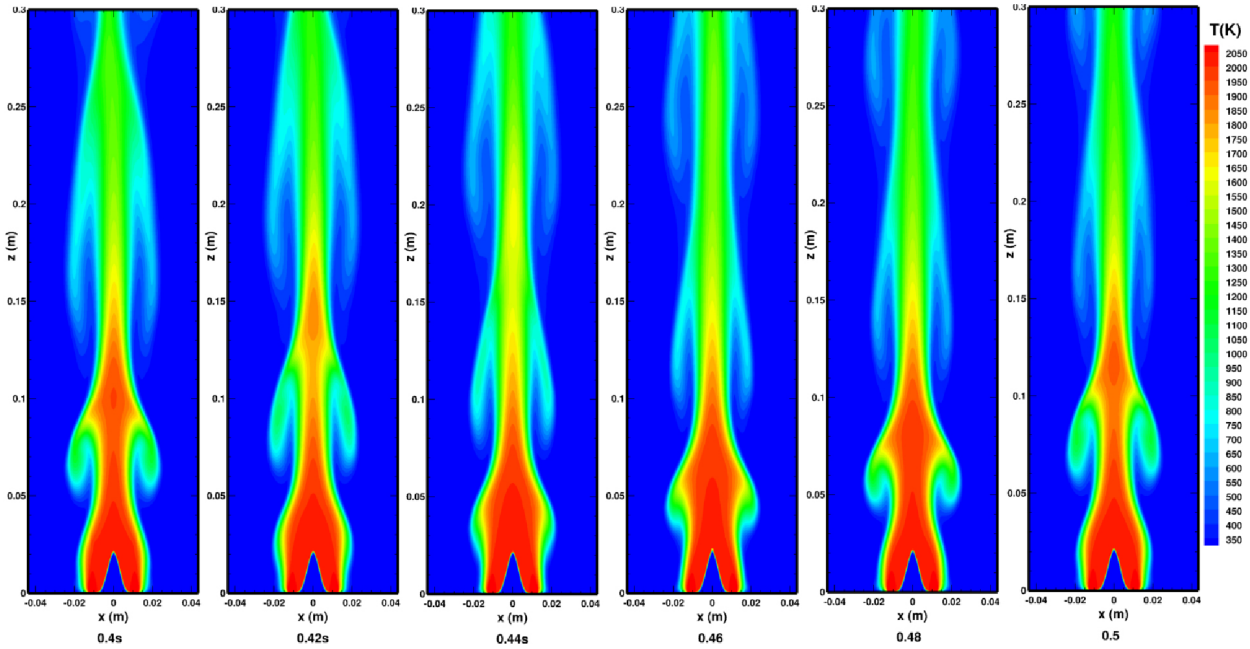
The flame configuration is very similar to that used for the laminar diffusion flame cases, however this time the fuel being injected through the inlet is replaced with a premixed methane-air mixture at a fixed equivalence ratio. The same initial and boundary conditions described previously and shown in Figure 2(a) are adopted, however the size of the computational domain used is much larger with a height of 0.3m and outer radius of 0.1m to avoid any boundary condition affects on the solution. The premixed reactants with a $\phi = 0.8$ are injected into the domain through a 0.025 m diameter inlet with a parabolic velocity profile with a peak velocity of 0.73 m/s. The gas mixture in the domain is quiescent standard atmospheric air and there is no co-flow.

The time-accurate parallel implicit algorithm was used to predict the behaviour of the unsteady premixed flame with a GMRES tolerance of 0.05 and ILU(0) preconditioning, with each time step of the Newton iteration converged two orders of magnitude, to a maximum of 10 Newton steps. The approximate Jacobian preconditioner was only updated for the first Newton step of each time-step, unless the number of GMRES iterations required increased. A fixed time-step of 0.01 ms was used during the unsteady calculation and mesh refinement was carried out every 0.1 ms, 10 steps, based on the gradient of temperature, and 3 levels of mesh refinement (4 mesh levels) were used. The calculation was run for 0.75 s to allow the solution to evolve

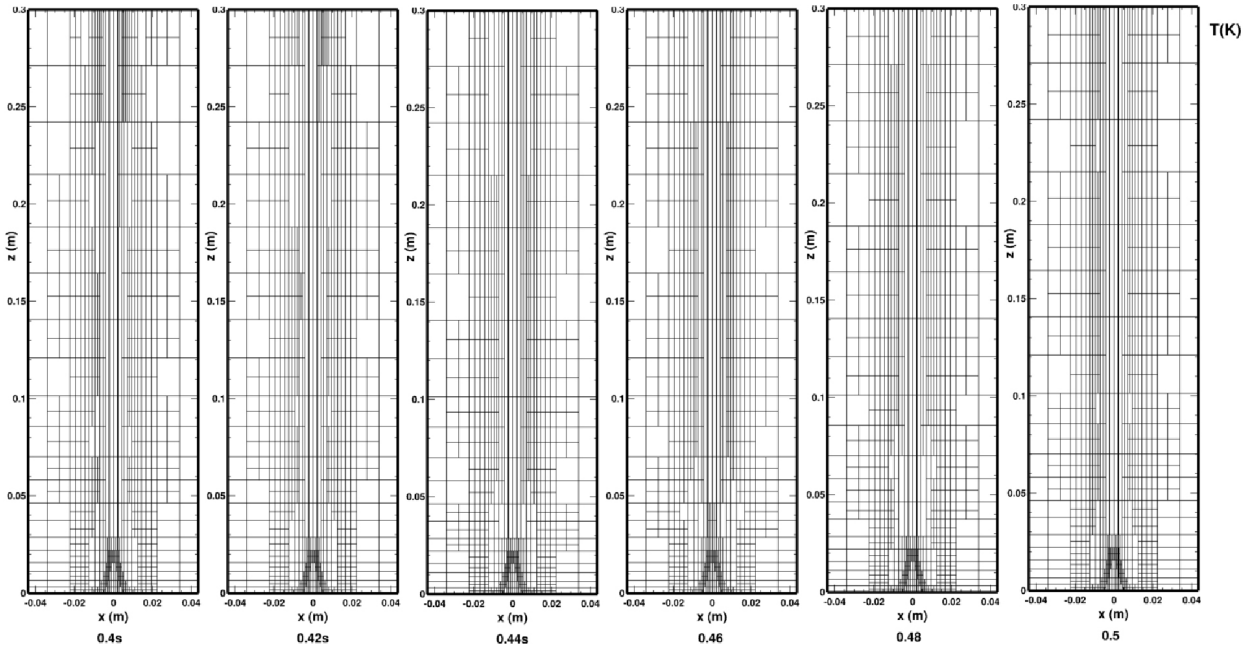


(a) 0.4s (b) 0.44s (c) 0.48s

Figure 4. 3D solution of a time-varying methane-air co-flow laminar premixed flame at 3 time intervals during its approximately 10Hz periodic cycle, with a) 12,707 b) 12,539 and c) 12,553 (8×8×8 cell) blocks each with 4 levels of refinement.



(a) Temperature Isotherms



(b) Block boundaries

Figure 5. 3D solution xz cross-section (y=0) of a methane-air premixed flame showing the (a) computed isotherms and (b) block boundaries with four levels of refinement at 6 time intervals showing the approximately 10 Hz buoyancy driven oscillations.

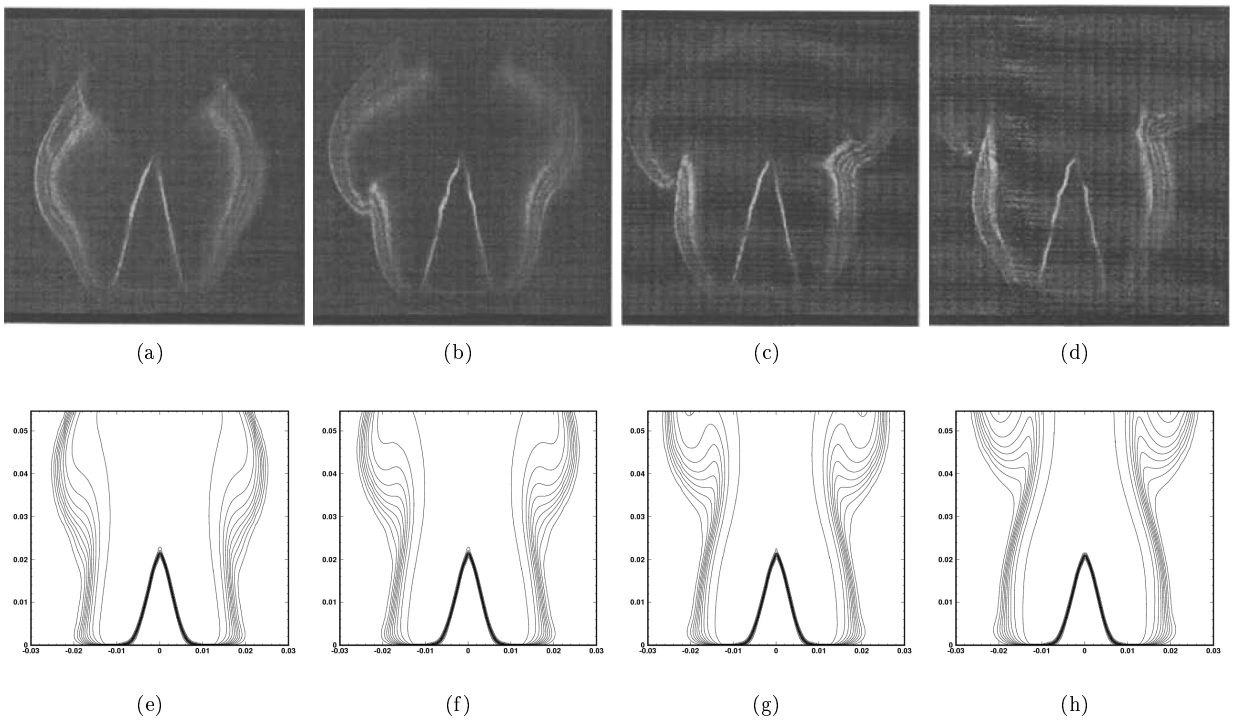


Figure 6. Comparison of (a-d) Schlieren images of experiments by Kostiuk *et al.*[41] and (e-h) numerical 2D cross-section density contours of an unsteady laminar premixed methane air flame with gravity.

toward a quasi-periodic solution in which the influences of the initial conditions and startup transients are effectively eliminated.

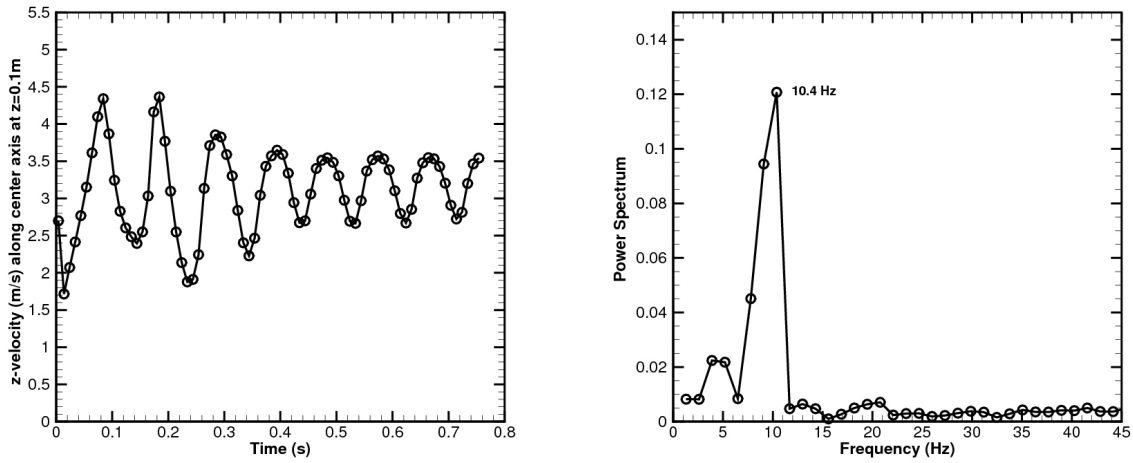
The resulting solution, as shown in Figure 4, matches well with the 2D axisymmetric results of Shepherd *et al.*[45] in terms of temperature, flame front size, and centerline velocities. When compared with the Schlieren imagery of the Kostiuk and Cheng[41] experiments, the 2D density distribution cross-sections at y=0 as given in Figure 6 also match reasonably well, showing the buoyancy interaction of the hot products and cold ambient air that produces the unsteady oscillations. Figure 5(a) shows six predicted isotherm cross-sections in the y=0 plane over 0.1 s time period showing the formation and evolution of the flame edge vortices. The adaptive mesh refinement indicated by the block boundaries in Figure 5(b) would seem to accurately track and resolve the vortices as they are shed and propagate downstream of the flame. The predicted flame tip height is nominally 22 mm which matches well with Shepherd's *et al.*[45] experimental estimates of the measured flame height that was reported to be about 20 mm. The oscillation frequency can be determined by tracking the centerline axial (z) velocity, Figure 7(a), verses time. When plotted in frequency space, shown in Figure 7(b), a dominant peek at 10.4Hz is observed which is the the periodic frequency of the flame oscillations. This agrees well with Shepherd's *et al*[45] measurement of 10.2 Hz and consistent with Kostiuk and Cheng[42]'s characterization of premixed methane flame oscillations in the 10–20Hz range.

Overall the combination of the time-accurate parallel implicit algorithm with dynamic AMR does a very good job at resolving the buoyancy driven instability. The adaptive mesh refinement, as mentioned, was required to provide significant resolution to resolve the thin flame front and allow the buoyancy induced oscillations to develop naturally. The agreement with previous numerical and experimental studies is quite good considering the limitations of the reduced methane-air chemical mechanisms.

V. Algorithm Performance

A. Solution Performance Comparison

To investigate the performance of the proposed time-accurate parallel implicit algorithm, referred herein as BDF2-NKS, two contemporary time-marching schemes were used for comparison. An explicit two-step, Runge-Kutta, time-marching scheme, RK2, without the temporal low-Mach-number preconditioning applied



(a) Axial Velocity (b) Frequency

Figure 7. Premixed methane-air (a) centerline flow velocity fluctuations at an axial height of z=0.1 m as a function of time and after (b) spectrum analysis shows a dominant frequency of 10.4 Hz.

as that would break time accuracy, as well as the same BDF2 implicit time scheme implemented within a dual-time stepping procedure using an explicit multi-stage optimally-smoothing scheme as the pseudo-time stepping method, referred herein as BDF2-DTS. The time-step of the Runge-Kutta scheme is limited by the CFL condition, whereas the implicit BDF2 schemes are not. Nevertheless, the solution accuracy of the implicit methods are affected by inner loop convergence tolerance.

Unlike for steady problems where a direct comparison of convergence histories shows the algorithms performance for time-accurate problems the only fair comparison is to compare overall solution CPU time at an equivalent solution accuracy for the same simulation or problem. The methodology used herein for comparison is similar to that used by Tabesh *et al.*[46] when comparing computational costs of various time-marching methods.

The performance is investigated using the solution of the 3D laminar diffusion flame discussed previously. As the solution is periodic it provides a rather good test case for comparing the time-marching schemes. The accuracy of the solutions is assessed by comparing the solution of the axial component of the center-line velocity, w, over one full oscillation, which for this case is 0.05 s. As there is no analytical solution for this case, the solution error, w_{error}, is calculated with reference to a computed solution, $w_{i,ref}$ shown in Figure 8, calculated with a very small time step. Equation (23) shows the error calculation where N is the total number of time steps.

$$w_{error} = \sqrt{\frac{\sum_{i=0}^{N} \left(w_i - w_{i,ref}\right)^2}{N}} \tag{23}$$

A 126 (8×8×8 cell) block mesh with 64,512 computational cells was used without any dynamic adaption to ensure consistent solutions with the different time-marching schemes. Initially the problem was run for 10 full periods (0.5 s) to remove any initial condition hysteresis affects and ensure a period solution was achieved. The BDF2 schemes were run with physical time steps, Δt, of 0.1, 0.05, 0.025, 0.01, and 0.005 ms corresponding to 500, 1000, 2000, 5000, and 10000 steps per period respectively. The RK2 solutions were run with CFL's of 0.3, 0.2 and 0.1, as 0.3 was found to be the largest stable value for this method. For the NKS-BDF2 an Newton tolerance of 0.01 with a GMRES tolerance of 0.01 was used.

The performance results depicted in Figure 9 provide a comparison of the various algorithms solution error, w_{error} verses the associated computational time. It is clear that the proposed BDF2-NKS algorithm outperforms the other methods requiring considerably less computational time to achieve the same solution

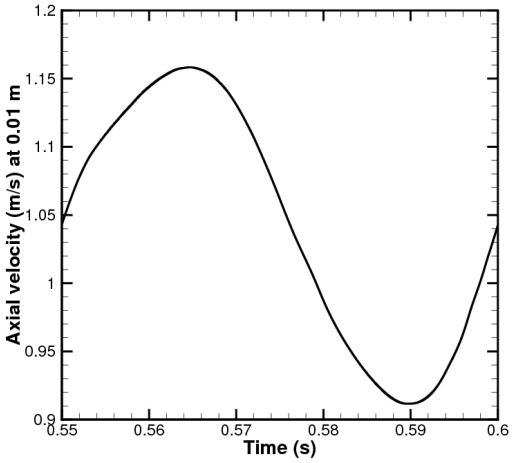


Figure 8. Time-accurate driven 3D laminar diffusion flame centerline velocity w at z=0.1 m for one period at 20 Hz oscillation (0.05 s).

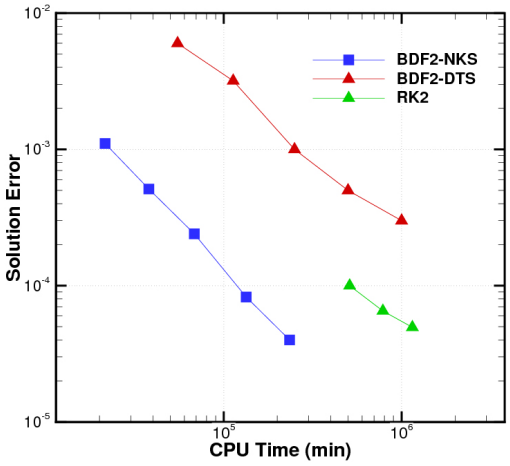


Figure 9. Unsteady algorithm performance comparisons of BDF2-NKS, BDF2-DTS, and RK2 for the solution of a time-accurate 3D driven laminar diffusion flame.

accuracy. The RK2 scheme is considerably handicapped by having to take very small time steps to maintain stability, and also not being able to utilize the low-Mach-number preconditioning of the numerical flux to maintain solution accuracy. The BDF2-DTS scheme does benefit from the greater stability of the implicit time scheme allowing it to use much large time-steps; however, it requires a far larger number of inner loop iterations of the multi-stage optimally smoothing scheme which completely offsets any possible gains in computational savings. Much like the performance of the steady solutions, the unsteady NKS solver outperforms the other methods by significantly reducing the number of residual evaluations that are required. For the reactive flow cases, the cost of the residual evaluation is very high due to the thermally perfect gas relationships and the chemical source terms. Thus the residual evaluation cost dominates over the extra costs associated with the NKS, i.e. preconditioner and GMRES overhead, resulting in the very significant performance gains in overall solution time.

B. Parallel Performance

Estimates of the parallel performance and scalability of the proposed parallel implicit AMR algorithm for the laminar diffusion flame problem are shown in Figure 10. The performance evaluation was carried out on a cluster of 2-way 4-core Intel Xeon E5540 CPU's connected with DDR Infiniband. The figure depicts the strong scaling of the algorithm. In particular, the relative parallel speed-up, S_p, given by $S_p = (t_1/t_p)p$, and the relative parallel efficiency, E_p, defined by $E_p = S_p/p$, for a fixed-size problem (fixed total computational work) as a function of the number of processors, p, are both provided where t_p is the total processor time required to solve the problem using p processors, and t_1 is the processor time required to solve the problem using a single processor.

The performance curves of Figure 10 are compared to the idealized results for up to 6400 processors for a methane/air diffusion flame calculation on a mesh consisting of 6400 $8\times 8\times 8$ cell solution blocks (3,276,800 cells). Although, there are some inefficiencies in the Schwarz preconditioning for this problem, it is quite evident that the speed-up is nearly linear and the efficiency remains above 80% for up to 6400 processors, the maximum number considered in this performance assessment.

C. Effects of Additive Schwarz Preconditioning

The strong parallel scaling simulation results described previously were performed using a fixed number of iterations and a fixed domain decomposition, thus the number of total blocks were held constant, regardless the number of processor cores being used. These results portray accurately the algorithms ability to scale well with the increases in communication; however, they do not assess or include the expected degradation

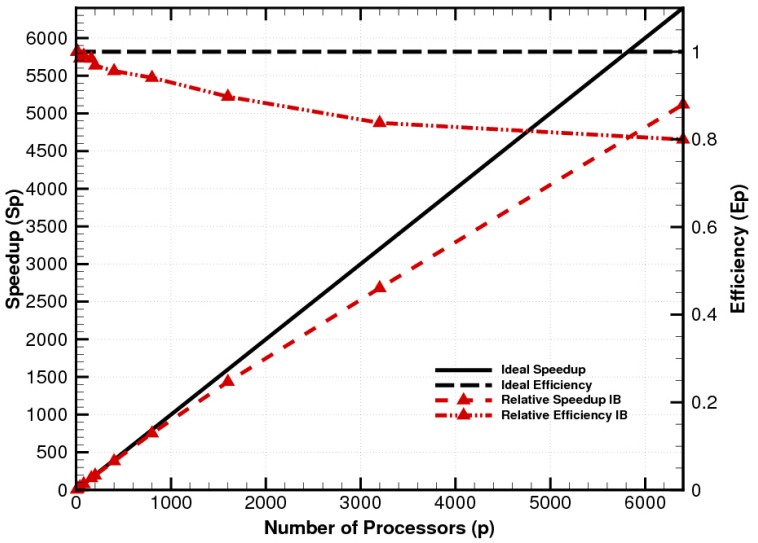


Figure 10. Parallel performance of the Newton algorithm for laminar diffusion flame calculation on a mesh consisting of 6400 $8\times 8\times 8$ cell solution blocks (3,276,800 cells) showing the relative parallel speed-up (strong scaling), $S_p = (t_1/t_p)p$, and parallel efficiency, E_p, as a function of the number of processors, p, used in the calculation.

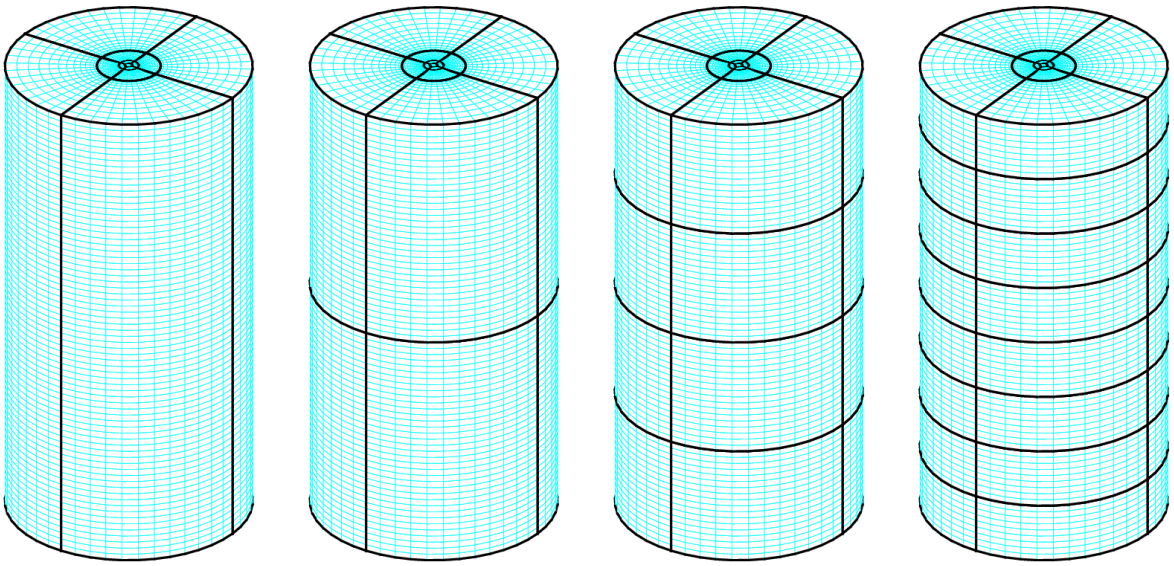
in performance of the parallel implicit algorithm caused by the domain decomposition procedure and the Schwarz preconditioning. Typically the decomposition changes as the number of processors are increased as further subdivisions of the domain are required. For explicit calculations this typically does not change the underlying algorithm or convergence rates, it just adds the increased overhead involved with communication at the boundaries. However, in the proposed parallel implicit algorithm, where the domain decomposition procedure is also used as the global Schwarz preconditioner, the algorithms convergence can be deleteriously affected by the decomposition. As more blocks are used, the less accurate the approximate inverse that results from the Schwarz preconditioner. As a result, typically a greater number of GMRES iterations to converge the problem are required, resulting in a higher computational cost. This was well illustrated by Groth *et al.*[47] for 2D inviscid flow problems.

As the Schwarz preconditioner is tied to the domain decomposition in this implementation, the effect of the preconditioner can be investigated by adjusting the partitioning (i.e. the number of blocks) for a mesh with a fixed number of cells. As the number of blocks increases the number of cells per block decreases and thus the global Schwarz preconditioner subdivides the global linear system into smaller and smaller local systems.

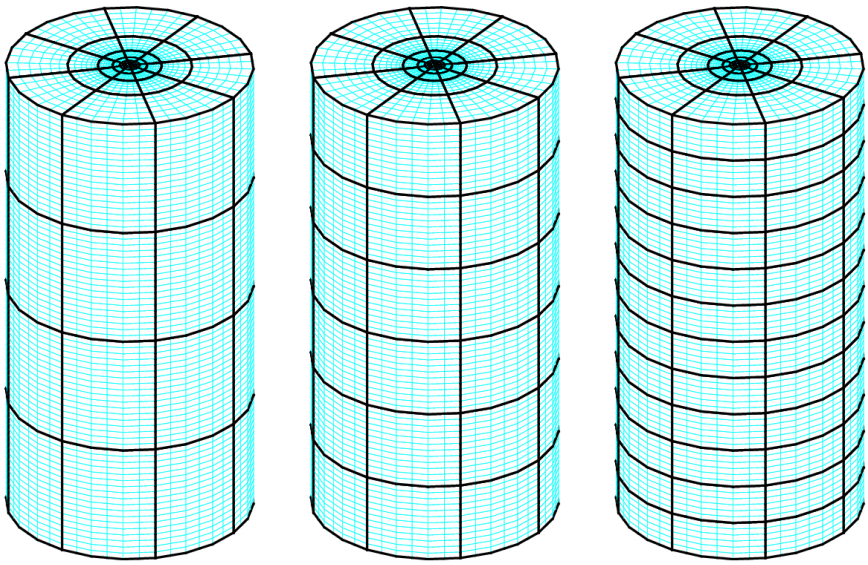
The influence of the Schwarz preconditioning and the resulting degradation in overall performance is investigated using the solution of a steady three-dimensional co-flow methane-air laminar diffusion flame. The overall solution was converged four orders of magnitude using a GMRES tolerance of 0.01 and ILU(0). An initial 13 ($12\times12\times72$ cell) block grid with 134,784 cells total is subdivided into 26, 52, 104, 208, 312, and 624 blocks as shown in Figure 11 with the 624 block portioning having only $6\times6\times6$ cells per block. All other algorithm and grid parameters, beside the block partitioning, are held constant.

Figure 12(a) shows the convergence histories for all 7 cases computed using 1 block per processor core. As is expected the 13 block case, with the least amount of Schwarz preconditioning, converges fastest and the 624 block slowest. The spread however is only about 20%, which is much better than 50% seen in the previous two-dimensional inviscid cases by Groth *et al.*[47] When looking at the solver statistics given in Table 1, the overall NKS iterations are relatively constant; however the total number of GMRES iterations increases by about 20%. This is consistent with what is expected as the global GMRES problem becomes less well conditioned as it is subdivided, requiring a greater number of iterations to achieve a specified tolerance. Figure 12(b) shows the strong parallel scaling for this case and even with the combined effects of the Schwarz preconditioner and parallel communication overhead parallel efficiency at 624 cores is just under 80%.

In summary, it would seem that overall the Schwarz preconditioner has a less of a detrimental effect on solution performance for the 3D reactive case than for the 2D inviscid cases of Groth *et al.*[47] Also the 3D

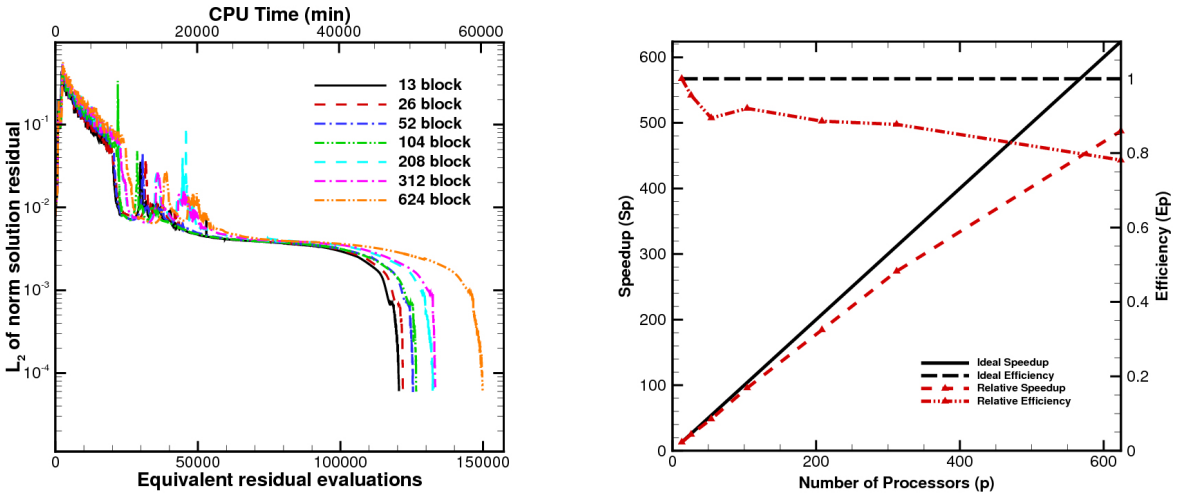


(a) 13 blocks (b) 26 blocks (c) 52 blocks (d) 104 blocks



(e) 208 blocks (f) 312 blocks (g) 624 blocks

Figure 11. 3D laminar diffusion flame 134,784 cells mesh partitioned (a) 13, (b) 26, (c) 52, (d) 104, (e) 208 , (f) 312, and (g) 624 blocks.



(a) Convergence histories

(b) Strong scaling

Figure 12. 3D Steady Laminar Diffusion flame computed with 134,784 cells and partitioned with 13, 26, 52, 104, 208, 312, and 624 blocks showing (a) the effect of the Schwarz preconditioner on convergence rate and (b) the associated strong parallel scaling.

reactive flow case has a much more expensive residual evaluation enabling greater parallel efficiency, due to the increased computation to communication ratio, ultimately reducing the overall time to achieve a solution.

Blocks	NKS Iterations	GMRES Iterations	CPU time (min)	Wall time (min)
13	4,501	37,155	48,446	3,726
26	4,521	37,548	48,988	1,884
52	4,573	37,622	50,421	969
104	4,552	38,105	50,871	489
208	4,596	41,313	53,186	254
312	4,567	42,625	53,572	171
624	4,640	46,527	60,236	96

Table 1. Summary of NKS algorithm solution statistics for the same 134,784 cells mesh with varying levels of Schwarz preconditioning determined by the number of blocks.

VI. Conclusions

A 3D parallel implicit AMR scheme has been developed for solving unsteady laminar reacting flows. The combination of finite-volume discretization procedure, parallel block-based AMR strategy, low-Mach-number preconditioning, dual-time-stepping-like approach, and Newton method solution procedure has resulted in a powerful and highly scalable computational tool for predicting a wide range of unsteady laminar reactive flows. Future research will involve the consideration of anisotropic mesh refinement strategies, the investigation of high-order temporal and spatial discretization procedures for reactive flows, and the application of the computational tool to the prediction of both turbulent reactive flows as well as combustion instabilities and thermo-acoustic phenomena.

Acknowledgments

Computational resources for performing all of the calculations reported herein were provided by the SciNet High Performance Computing Consortium at the University of Toronto and Compute/Calcul Canada through funding from the Canada Foundation for Innovation (CFI) and the Province of Ontario, Canada.

References

[1]Berger, M. J., "Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations," *J. Comput. Phys.*, Vol. 53, 1984, pp. 484–512.

[2]Berger, M. J. and Colella, P., "Local Adaptive Mesh Refinement for Shock Hydrodynamics," *J. Comput. Phys.*, Vol. 82, 1989, pp. 67–84.

[3]De Zeeuw, D. and Powell, K. G., "An Adaptively Refined Cartesian Mesh Solver for the Euler Equations," *J. Comput. Phys.*, Vol. 104, 1993, pp. 56–68.

[4]Aftosmis, M. J., Berger, M. J., and Melton, J. E., "Robust and Efficient Cartesian Mesh Generation for Component-Base Geometry," *AIAA J.*, Vol. 36, No. 6, 1998, pp. 952–960.

[5]Groth, C. P. T., Zeeuw, D. L. D., Powell, K. G., Gombosi, T. I., and Stout, Q. F., "A Parallel Solution-Adaptive Scheme for Ideal Magnetohydrodynamics," Paper 99-3273, AIAA, June 1999.

[6]Sachdev, J. S., Groth, C. P. T., and Gottlieb, J. J., "A Parallel Solution-Adaptive Scheme for Predicting Multi-Phase Core Flows in Solid Propellant Rocket Motors," *Int. J. Comput. Fluid Dyn.*, Vol. 19, No. 2, 2005, pp. 157–175.

[7]Northrup, S. A. and Groth, C. P. T., "Solution of Laminar Diffusion Flames Using a Parallel Adaptive Mesh Refinement Algorithm," Paper 2005–0547, AIAA, January 2005.

[8]Gao, X. and Groth, C. P. T., "Parallel Adaptive Mesh Refinement Scheme for Turbulent Non-Premixed Combusting Flow Prediction," Paper 2006-1448, AIAA, January 2006.

[9]Gao, X. and Groth, C. P. T., "A Parallel Adaptive Mesh Refinement Algorithm for Predicting Turbulent Non-Premixed Combusting Flows," *Int. J. Comput. Fluid Dyn.*, Vol. 20, No. 5, 2006, pp. 349–357.

[10]Gao, X. and Groth, C. P. T., "Parallel Adaptive Mesh Refinement Scheme for Three-Dimensional Turbulent Non-Premixed Combustion," Paper 2008-1017, AIAA, January 2008.

[11]Gao, X. and Groth, C. P. T., "A Parallel Solution-Adaptive Method for Three-Dimensional Turbulent Non-Premixed Combusting Flows," *J. Comput. Phys.*, Vol. 229, No. 5, 2010, pp. 3250–3275.

[12]Gao, X., Northrup, S. A., and Groth, C. P. T., "Parallel Solution-Adaptive Method for Two-Dimensional Non-Premixed Combusting Flows," *Prog. Comput. Fluid Dyn.*, Vol. 11, No. 2, 2011, pp. 76–95.

[13]Northrup, S. A. and Groth, C. P. T., "Parallel Implicit Adaptive Mesh Refinement Algorithm for Solution of Laminar Combusting Flows," 14th Annual Conference of the CFD Society of Canada, Kingston, Ontario, Canada, July 16–18, 2006, July 2006.

[14]Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *J. Comput. Phys.*, Vol. 43, 1981, pp. 357–372.

[15]Weiss, J. M. and Smith, W. A., "Preconditioning Applied to Variable and Constant Density Flows," *AIAA J.*, Vol. 33, No. 11, 1995, pp. 2050–2057.

[16]Gordon, S. and McBride, B. J., "Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications I. Analysis," Reference Publication 1311, NASA, 1994.

[17]McBride, B. J. and Gordon, S., "Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications II. Users Manual and Program Description," Reference Publication 1311, NASA, 1996.

[18]Wilke, C. R., "A Viscosity Equation for Gas Mixtures," *J. Chem. Phys.*, Vol. 18, 1950, pp. 517–519.

[19]Dixon-Lewis, G., "Computer Modeling of Combustion Reactions in Flowing Systems with Transport," *Combustion Chemistry*, edited by W. C. Gardiner, Springer-Verlag, New York, 1984, pp. 21–126.

[20]Smith, G. P., Golden, D. M., Frenklach, M., Moriarty, N. W., Eiteneer, B., Goldenberg, M., Bowman, C. T., Hanson, R. K., Song, S., Gardiner, W. C., Lissianski, V. V., and Qin, Z., "GRI-Mech 3.0," http://www.me.berkeley.edu/gri_mech/.

[21]Westbrook, C. K. and Dryer, F. L., "Simplified Reaction Mechanisms for the Oxidation of Hydrocarbon Fuels in Flames," *Combust. Sci. Tech.*, Vol. 27, 1981, pp. 31–43.

[22]Venkatakrishnan, V., "On the Accuracy of Limiters and Convergence to Steady State Solutions," Paper 93-0880, AIAA, January 1993.

[23]Mathur, S. R. and Murthy, J. Y., "A Pressure-Based Method for Unstructured Meshes," Vol. 31, 1997, pp. 191–215.

[24]Jameson, A., "Time Dependent Calculations using Multigrid, with Applications to Unsteady Flows Past Airfoils and Wings," Paper 1991-1596, AIAA, June 1991.

[25]Gao, X., *A Parallel Solution-Adaptive Method for Turbulent Non-Premixed Combusting Flows*, Ph.D. thesis, University of Toronto, September 2008.

[26]Saad, Y. and Schultz, M. H., "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Equations," *SIAM J. Sci. Stat. Comput.*, Vol. 7, No. 3, 1986, pp. 856–869.

[27]Saad, Y., *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996.

[28]Dembo, R. S., Eisenstat, S. C., and Steihaug, T., "Inexact Newton Methods," *SIAM J. Numer. Anal.*, Vol. 19, No. 2, 1982, pp. 400–408.

[29]Gropp, W. D., Kaushik, D. K., Keyes, D. E., and Smith, B. F., "High-Performance Parallel Implicit CFD," *Parallel Computing*, Vol. 27, 2001, pp. 337–362.

[30]Knoll, D. A. and Keyes, D. E., "Jacobian-Free Newton-Krylov Methods: A Survey of Approaches and Applications," *J. Comput. Phys.*, Vol. 193, 2004, pp. 357–397.

[31]Knoll, D. A., McHugh, P. R., and Keyes, D. E., "Newton-Krylov Methods for Low-Mach-Number Compressible Combustion," *AIAA J.*, Vol. 34, No. 5, 1996, pp. 961–967.

[32]Cai, X.-C., Gropp, W. D., Keyes, D. E., Melvin, R. G., and Young, D. P., "Parallel Newton-Krylov-Schwarz Algorithms for the Transonic Full Potential Equations," *SIAM J. Sci. Comput.*, Vol. 19, No. 1, 1998, pp. 246–265.

[33]Nielsen, E. J., Anderson, W. K., Walters, R. W., and Keyes, D. E., "Application of Newton-Krylov Methodology to a Three-Dimensional Unstructured Euler Code," Paper 95-1733-CP, AIAA, June 1995.

[34]Aftosmis, M. J., Berger, M. J., and Murman, S. M., "Applications of Space-Filling Curves to Cartesian Methods for CFD," Paper 2004-1232, AIAA, January 2004.

[35]Gropp, W., Lusk, E., and Skjellum, A., *Using MPI*, MIT Press, Cambridge, Massachussets, 1999.

[36]Gropp, W., Lusk, E., and Thakur, R., *Using MPI-2*, MIT Press, Cambridge, Massachussets, 1999.

[37]Day, M. S. and Bell, J. B., "Numerical Simulation of Laminar Reacting Flows with Complex Chemistry," *Combust. Theory Modelling*, Vol. 4, No. 4, 2000, pp. 535–556.

[38]Dworkin, S. B., Connelly, B. C., Schaffer, A. M., Bennett, B. A. V., Long, M. B., Smooke, M. D., Puccio, M. P., McAndrews, B., and Miller, J. H., "Computational and experimental study of a forced, time-dependent, methane-air coflow diffusion flame," *Thirty First Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, 2007, pp. 971–978.

[39]Mohammed, R. K., Tanoff, M. A., Smooke, M. D., Schaffer, A. M., and Long, M. B., "Computational and Experimental Study of a Forced, Time-Varying, Axisymmetric, Laminar Diffusion Flame," *Twenty-Seventh Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, 1998, pp. 693–702.

[40]Durox, D., Baillot, F., Scoufflaire, P., and Prudhomme, R., "Some effects of gravity on the behaviour of premixed flames," *Combust. Flame*, Vol. 82, 1990, pp. 66–74.

[41]Kostiuk, L. W. and Cheng, R. K., "Imaging of premixed flames in microgravity," Vol. 18, 1994, pp. 59–68.

[42]Kostiuk, L. W. and Cheng, R. K., "The coupling of conical wrinkled laminar flames with gravity," *Combust. Flame*, Vol. 103, 1995, pp. 27–40.

[43]Cheng, R. K., Bedat, B., and Kostiuk, L. W., "Effects of buoyancy on lean premixed V-flames Part I: laminar and turbulent flame structures," *Combust. Flame*, Vol. 116, 1999, pp. 360–375.

[44]Guahk, Y. T., Lee, D. K., Oh, K. C., and Shin, H. D., "Flame-Intrinsic Kelvin–Helmholtz Instability of Flickering Premixed Flames," Vol. 23, 2009, pp. 3875–3884.

[45]Shepherd, I. G., Day, M. S., and Cheng, R. K., "The Dynamics of Flame Flicker in Conical Premixed Flames: An Experimental and Numerical Study," LBNL Report LBNL-59249, 2005.

[46]Tabesh, M. and Zingg, D. W., "Efficient Implict Time Marching Methods Using a Newton-Krylov Algorithm," Paper 2009-164, AIAA, January 2009.

[47]Groth, C. P. T. and Northrup, S. A., "Parallel Implicit Adaptive Mesh Refinement Scheme for Body-Fitted Multi-Block Mesh," Paper 2005-5333, AIAA, June 2005.