INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **57**:265–293
Published online 25 October 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1635

Development and validation of a SUPG finite element scheme for the compressible Navier–Stokes equations using a modified inviscid flux discretization

Benjamin S. Kirk[1, 2, *, †] and Graham F. Carey[1, 2]

[1] *NASA Lyndon B. Johnson Space Center, Houston, TX 77058, U.S.A.*
[2] *The University of Texas at Austin, Austin, TX 78712, U.S.A.*

SUMMARY

This paper considers the streamline-upwind Petrov–Galerkin (SUPG) method applied to the unsteady compressible Navier–Stokes equations in conservation-variable form. The spatial discretization, including a modified approach for interpolating the inviscid flux terms in the SUPG finite element formulation, and the second-order accurate time discretization are presented. The numerical method is discussed in detail. The performance of the algorithm is then investigated by considering inviscid flow past a circular cylinder. Validation of the finite element formulation *via* comparisons with experimental data for high-Mach number perfect gas laminar flows is presented, with a specific focus on comparisons with experimentally measured skin friction and convective heat transfer on a 15° compression ramp. Copyright © 2007 John Wiley & Sons, Ltd.

Received 8 July 2007; Revised 4 September 2007; Accepted 6 September 2007

KEY WORDS: SUPG; finite element; compressible flows; shock capturing; inviscid flux discretization; validation

1. INTRODUCTION

Compressible flows encompass a wide range of applications that are of particular interest in the design and analysis of atmospheric flight and re-entry vehicles. This paper begins with the presentation of the compressible Navier–Stokes equations, which are used to model this class of flows. A finite element formulation (which is suitable to arbitrary unstructured discretizations) is then developed and implemented on top of the `libMesh` parallel adaptive finite element library [1, 2]. A fully implicit algorithm is used to preclude explicit stability restrictions that are dependent on mesh size. A nonstandard treatment of the inviscid flux term is used and is found to enhance the

*Correspondence to: Benjamin S. Kirk, NASA Lyndon B. Johnson Space Center, Mail Code EG3, 2101 NASA Parkway, Houston, TX 77058, U.S.A.
†E-mail: benjamin.kirk@nasa.gov

stability of the streamline-upwind Petrov–Galerkin (SUPG) scheme as applied to the conservation variables. Since the primary goal of this paper is to assess the suitability of the modified SUPG scheme, the problems considered here are restricted to laminar, perfect gas flows. Of particular interest are the stability, consistency, and convergence properties of the current approach. The time discretization and nonlinear solution techniques used in the computational algorithm are also described in detail.

In the present work, the focus is on some specific refinements to finite element methodology and algorithms that are designed for the compressible aerodynamic flows mentioned in the opening paragraph. Most specifically, the novel aspects concern the inviscid flux treatment, group-variable approach, and extended SUPG stabilization strategy for the applications class and within the parallel scheme developed here. Moreover, the present work will form a baseline for future studies exploring adaptive mesh refinement techniques for this problem class, as well as modeling error relative to other candidate models, discretization error, and other errors such as uncertainty in data in flow field behavior.

The remainder of this paper is outlined as follows. Section 2 reviews the compressible Navier–Stokes equations for a laminar, calorically perfect gas (and associated transport properties) which describe the problem class. Section 3 then presents the stabilized weak form of the governing equations and describes the associated finite element discretization. The parallel solution methodology is described in Section 5, and the performance of the algorithm is then investigated with numerical experiments and validation cases in Section 6. Finally, some general observations are drawn and areas for future research are discussed in Section 7.

2. MATHEMATICAL MODEL

The compressible Navier–Stokes equations describe the conservation of mass, momentum, and energy for this class of flows. This section summarizes the Navier–Stokes system of equations, relevant state equations and transport property models for air, and the nondimensionalization scheme used in the present work.

2.1. Governing equations

The conservation of mass, momentum, and energy for a compressible fluid may be expressed as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{1}$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) = -\nabla P + \nabla \cdot \boldsymbol{\tau} \tag{2}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot (\rho E \mathbf{u}) = -\nabla \cdot \mathbf{q} - \nabla \cdot (P\mathbf{u}) + \nabla \cdot (\boldsymbol{\tau}\mathbf{u}) \tag{3}$$

where ρ is the density, $\mathbf{u}$ is the velocity, E is the total energy per unit mass, and P is the pressure. The total energy per unit mass, E, in Equation (3) may be decomposed into internal and kinetic

DOI: 10.1002/fld

contributions: $E = e + (\mathbf{u}\cdot\mathbf{u})/2$. The viscous stress tensor $\boldsymbol{\tau}$ and the heat flux vector $\mathbf{q}$ are defined as

$$\boldsymbol{\tau} = \mu(\nabla\mathbf{u} + \nabla^T\mathbf{u}) - \tfrac{2}{3}\mu(\nabla\cdot\mathbf{u})\mathbf{I} \tag{4}$$

$$\mathbf{q} = -k\nabla T \tag{5}$$

where μ is the dynamic viscosity, k is the thermal conductivity, T is the fluid temperature, and $\mathbf{I}$ denotes the identity matrix.

2.2. *Equations of state*

In three dimensions, Equations (1)–(3) provide a system of five coupled partial differential equations in the seven unknowns $\rho, \mathbf{u}, e, P, T$, provided that the transport properties μ and k may be related to the unknown thermodynamic properties. Clearly, two additional equations are required to close the system. These additional equations are equations of state that relate the thermodynamic variables ρ, e, P, T. Assuming that the fluid is in thermodynamic equilibrium, its state is fixed by any two independent thermodynamic variables. Thus, by choosing ρ and e to be the independent variables, state equations for $P = P(\rho, e)$ and $T = T(\rho, e)$ may be obtained as follows:

$$P = (\gamma - 1)\rho e \tag{6}$$

$$T = \frac{(\gamma - 1)e}{R} \tag{7}$$

where γ is the ratio of specific heats and R is the ideal gas constant. Equations (6)–(7) are valid for a calorically perfect gas.

2.3. *Transport properties*

The remaining coefficients of viscosity and thermal conductivity may be related to the thermodynamic variables using kinetic theory [3]. For air over a wide range of temperatures, $\mu = \mu(T)$ and is given by Sutherland's law [4]

$$\mu = \mu_{\text{ref}} \frac{T^{3/2}}{T + T_{\text{ref}}} \tag{8}$$

where μ_{ref} and T_{ref} are reference values. (For air $\mu_{\text{ref}} = 1.458 \times 10^{-6}\,\text{Pa s}$ and $T_{\text{ref}} = 110.4\,\text{K}$.) With the viscosity given by Equation (8) for a given temperature, it is convenient to determine the thermal conductivity k assuming a constant Prandtl number. The Prandtl number, which defines the ratio of the fluid's viscous to thermal diffusivity, is defined as

$$Pr = \frac{\mu c_p}{k} \tag{9}$$

and $Pr = 0.71$ for air at standard conditions.

DOI: 10.1002/fld

2.4. Nondimensionalization

The governing equations (1)–(3) can be nondimensionalized in a number of ways. The present work employs the Reynolds number based on reference length L ($Re_L = \rho_\infty U_\infty L/\mu_\infty$) as the basis for the nondimensionalization of the independent variables as follows:

$$\hat{\mathbf{x}} = \frac{\mathbf{x}}{L} \tag{10}$$

$$\hat{\mathbf{u}} = \frac{\mathbf{u}}{U_\infty} \tag{11}$$

$$\hat{t} = \frac{t}{L/U_\infty} \tag{12}$$

$$\hat{\rho} = \frac{\rho}{\rho_\infty} \tag{13}$$

$$\hat{p} = \frac{p}{\rho_\infty U_\infty^2} \tag{14}$$

$$\hat{T} = \frac{T}{T_\infty} \tag{15}$$

$$\hat{e} = \frac{e}{U_\infty^2} \tag{16}$$

$$\hat{\mu} = \frac{\mu}{Re_L \, \mu_\infty} \tag{17}$$

where $()_\infty$ denotes freestream values. Substituting (10)–(17) into (1)–(3) yields the nondimensional set of equations which have an analogous form to that in (1)–(3). Since these forms have the same structure, they can be applied for both dimensional and nondimensional simulations [5].

2.5. System of equations

Equations (1)–(3) may be expressed in conservative system form as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_i}{\partial x_i} = \frac{\partial \mathbf{G}_i}{\partial x_i} \tag{18}$$

where the vector $\mathbf{U}$ consists of the so-called conservation variables, $\mathbf{F}_i$ and $\mathbf{G}_i$ are the inviscid and viscous fluxes in the ith direction, respectively. The conservation variables $\mathbf{U} = [\rho, \rho u, \rho v, \rho w, \rho E]^{\mathrm{T}}$

DOI: 10.1002/fld

correspond to the fluid density, Cartesian components of momentum per unit volume, and total energy per unit volume, respectively. The inviscid and viscous fluxes in (18) are given by

$$\mathbf{F}_i = \begin{bmatrix} \rho u_i \\ \rho u_i u_1 + \delta_{i1} P \\ \rho u_i u_2 + \delta_{i2} P \\ \rho u_i u_3 + \delta_{i3} P \\ \rho u_i \left(E + \dfrac{P}{\rho} \right) \end{bmatrix} \tag{19}$$

$$\mathbf{G}_i = \begin{bmatrix} 0 \\ \tau_{i1} \\ \tau_{i2} \\ \tau_{i3} \\ -q_i + \tau_{ik} u_k \end{bmatrix} \tag{20}$$

where δ_{ij} is the Kronecker delta satisfying $\delta_{ij} = 0$ when $i \neq j$ and is of unit value otherwise. The second term on the left-hand side of (18) is the divergence of the inviscid flux vector, $\partial \mathbf{F}_i / \partial x_i$, and may be expressed in terms of the unknowns $\mathbf{U}$ as

$$\frac{\partial \mathbf{F}_i}{\partial x_i} = \frac{\partial \mathbf{F}_i}{\partial \mathbf{U}} \frac{\partial \mathbf{U}}{\partial x_i} = \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial x_i} \tag{21}$$

where $\mathbf{A}_i = \partial \mathbf{F}_i / \partial \mathbf{U}$ is the inviscid flux Jacobian. Similarly, the viscous flux vector $\mathbf{G}_i$ may be expressed as

$$\frac{\partial \mathbf{G}_i}{\partial x_i} = \frac{\partial}{\partial x_i} \left(\mathbf{K}_{ij} \frac{\partial \mathbf{U}}{\partial x_j} \right) \tag{22}$$

where $\mathbf{K}_{ij}$ is a diffusivity matrix. Matrices $\mathbf{A}_i$ and $\mathbf{K}_{ij}$ are both functions of the independent variables $\mathbf{U}$ and are listed explicitly in Reference [2]. Using (21) and (22) in (18) yields the second-order system

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial x_i} = \frac{\partial}{\partial x_i} \left(\mathbf{K}_{ij} \frac{\partial \mathbf{U}}{\partial x_j} \right) \tag{23}$$

which will be the basis for developing a weak formulation in Section 3. In the limit of vanishing viscosity the right-hand side of Equation (23) is identically zero, resulting in the first-order, hyperbolic Euler equations.

DOI: 10.1002/fld

3. WEAK FORMULATION

3.1. Galerkin weak statement

The corresponding weak form of the governing system of Equations (23) may be constructed in the standard way by first multiplying with an appropriate set of test functions $\mathbf{W}$ and integrating over the domain Ω. Integrating the viscous term by parts yields the weak statement: Find $\mathbf{U}$ satisfying the essential boundary and initial conditions such that

$$\int_\Omega \left[\mathbf{W}\cdot\left(\frac{\partial \mathbf{U}}{\partial t}+\mathbf{A}_i\frac{\partial \mathbf{U}}{\partial x_i}\right)+\frac{\partial \mathbf{W}}{\partial x_i}\cdot\left(\mathbf{K}_{ij}\frac{\partial \mathbf{U}}{\partial x_j}\right)\right]\mathrm{d}\Omega-\oint_\Gamma \mathbf{W}\cdot\mathbf{g}\,\mathrm{d}\Gamma=0 \tag{24}$$

for all $\mathbf{W}$ in an appropriate function space. In the last term $\mathbf{g}=\mathbf{G}\cdot\hat{\mathbf{n}}$ is the normal component of the viscous flux on the boundary Γ with unit normal $\hat{\mathbf{n}}$.

3.2. Stabilized formulation

A standard Galerkin finite element formulation as presented in (24) (or similar finite difference or finite volume strategies) is unstable in the sense that it may produce nonphysical oscillations in regions of steep solution gradients or strong convection. Even when viscous effects are included as in (24) standard Galerkin calculations may produce nonphysical oscillations for convection-dominated flows. This well-known phenomenon results because the standard Galerkin formulation (or equivalently central differencing on a structured grid) produces a difference stencil whose solution admits oscillatory behavior [6–8].

For some classes of flow and transport this instability can be directly related to inadequate spatial resolution in the grid. In these cases the Galerkin discretization on a sufficiently refined mesh will produce stable results. This is typically the case for low-speed incompressible flows for which there is an approximate balance between the convective and diffusive length scales. This balance is described by the cell Reynolds (or Peclet) number, which is defined as

$$Re_c \equiv \frac{\rho U h_{\mathrm{ref}}}{\mu} \tag{25}$$

where h_{ref} is the cell reference length and the other properties are evaluated locally. When the local flow properties and mesh spacing are such that $Re_c<2$ the standard Galerkin formulation will yield nonoscillatory results. Unfortunately, such a balance is rarely achieved for compressible flows in aerospace applications. Indeed, the Euler equations are devoid of any diffusion; hence, a standard Galerkin discretization such as in Equation (24) will always exhibit stability issues, regardless of mesh resolution.

Several techniques have been proposed to address the stability issue of the Galerkin formulation. The familiar Lax–Wendroff finite difference scheme produces the Taylor–Galerkin scheme in the context of finite elements. The Taylor–Galerkin scheme employs a second-order Taylor series in time and an interchange of spatial and temporal differentiation in the discretization of (18). This yields a second-order term in the discrete form that can be interpreted as a stabilizing diffusion. Recently the Taylor–Galerkin scheme has been applied to hypersonic flowfields in chemical and thermal nonequilibrium [9], illustrating its applicability to the class of problems considered in the present work.

A different approach is pursued by Carey *et al.* in the least-squares finite element method. In the least-squares approach the test function $\mathbf{W}$ in (24) is replaced by the variation of the residual

DOI: 10.1002/fld

of the governing equations [10, 11]. Conceptually this is equivalent to minimizing the residual in a least-squares sense. A detailed analysis of this formulation reveals a stabilizing mechanism similar to the Taylor–Galerkin scheme. This least-squares idea can be combined with the Galerkin statement to yield the so-called Galerkin/least-squares scheme [12].

The stabilization introduced *via* numerical dissipation in upwind differencing can be achieved in the finite element setting when an upwind bias is added to the test function **W**. This idea, and the need to reduce cross-wind dissipation in two or three dimensions, led to the development of the directed SUPG formulation as another stabilizing mechanism for convection-dominated flows [13]. For the system of equations (23) a suitably upstream-biased test function can be defined by augmenting the standard Galerkin test function **W** with the convective operator acting on the test function:

$$\hat{\mathbf{W}}=\mathbf{W}+\boldsymbol{\tau}_{\mathrm{SUPG}}\mathbf{A}_i\frac{\partial \mathbf{W}}{\partial x_i} \tag{26}$$

The stabilization matrix $\boldsymbol{\tau}_{\mathrm{SUPG}}$ plays an important role in the SUPG formulation in that it seeks to introduce the minimal amount of diffusion necessary to stabilize the scheme. In this work $\boldsymbol{\tau}_{\mathrm{SUPG}}$ is adapted from previous work by Shakib *et al.* [14] in the context of entropy variables and later used by Aliabadi with the conservation variables [15, 16]. Specifically, in three dimensions

$$\boldsymbol{\tau}_{\mathrm{SUPG}}=\mathrm{diag}(\tau_{\mathrm{c}},\tau_{\mathrm{m}},\tau_{\mathrm{m}},\tau_{\mathrm{m}},\tau_{\mathrm{e}}) \tag{27}$$

where τ_{c}, τ_{m}, and τ_{e} are scalar stabilization parameters for the continuity, momentum, and energy equations, respectively, and are given by

$$\tau_{\mathrm{c}}=\left[\left(\frac{2}{\Delta t}\right)^2+\left(\frac{2\,(\|\mathbf{u}\|+c)}{h_{\mathbf{u}}}\right)^2\right]^{-1/2}$$

$$\tau_{\mathrm{m}}=\left[\left(\frac{2}{\Delta t}\right)^2+\left(\frac{2\,(\|\mathbf{u}\|+c)}{h_{\mathbf{u}}}\right)^2+\left(\frac{4\mu}{\rho h_{\mathbf{u}}^2}\right)^2\right]^{-1/2}$$

$$\tau_{\mathrm{e}}=\left[\left(\frac{2}{\Delta t}\right)^2+\left(\frac{2\,(\|\mathbf{u}\|+c)}{h_{\mathbf{u}}}\right)^2+\left(\frac{4k}{\rho c_p h_{\mathbf{u}}^2}\right)^2\right]^{-1/2}$$

and are designed to transition smoothly between convective, diffusive, and transient-dominated flow regimes. The flow-aligned element length scale, $h_{\mathbf{u}}$, is defined as

$$h_{\mathbf{u}}=2\left(\sum_{k=1}^{\mathrm{NN}}|\hat{\mathbf{u}}\cdot\nabla\phi_k|\right)^{-1}$$

where NN is the number of nodes in the element, $\{\nabla\phi\}$ are the element shape function gradients, and $\hat{\mathbf{u}}=\mathbf{u}/\|\mathbf{u}\|$ is the flow-aligned unit vector.

It is important to note that all of the schemes discussed previously address instabilities induced by strong convection. For supersonic problems involving strong shock waves another form of stabilization is required. More specifically, a local regularization scheme using a shock-capturing function is required to eliminate nonphysical over and under-shoots induced by strong gradients.

DOI: 10.1002/fld

The regularized SUPG weak statement then follows by multiplying (23) with (26) and integrating by parts as before:

$$\begin{aligned}\int_\Omega \left[\mathbf{W}\cdot\left(\frac{\partial \mathbf{U}}{\partial t}+\mathbf{A}_i\frac{\partial \mathbf{U}}{\partial x_i}\right)+\frac{\partial \mathbf{W}}{\partial x_i}\cdot\left(\mathbf{K}_{ij}\frac{\partial \mathbf{U}}{\partial x_j}\right)\right]\mathrm{d}\Omega \\ +\sum_{e=1}^{n_{el}}\int_{\Omega_e}\boldsymbol{\tau}_{\text{SUPG}}\frac{\partial \mathbf{W}}{\partial x_k}\cdot\mathbf{A}_k\left[\frac{\partial \mathbf{U}}{\partial t}+\mathbf{A}_i\frac{\partial \mathbf{U}}{\partial x_i}-\frac{\partial}{\partial x_i}\left(\mathbf{K}_{ij}\frac{\partial \mathbf{U}}{\partial x_j}\right)\right]\mathrm{d}\Omega \\ +\sum_{e=1}^{n_{el}}\int_{\Omega_e}\delta\left(\frac{\partial \mathbf{W}}{\partial x_i}\cdot\frac{\partial \mathbf{U}}{\partial x_i}\right)\mathrm{d}\Omega-\oint_\Gamma \mathbf{W}\cdot\mathbf{g}\,\mathrm{d}\Gamma=0\end{aligned} \tag{28}$$

The shock-capturing function is local and essentially regularizes the problem by selectively introducing isotropic artificial diffusion. This added local dissipation captures shocks approximately across a few mesh cells.

The shock-capturing function was adapted for a system of conservation variables by Aliabadi [15] Aliabadi and Tezduyar [16], and LeBeau [17] from the original definition employed by Hughes *et al.* for the case of entropy variables [14, 18], and is given by

$$\delta_{\text{orig}}=\left[\frac{\left\|\mathbf{A}_i\dfrac{\partial \mathbf{U}}{\partial x_i}\right\|_{\mathbf{A}_0^{-1}}}{\|\boldsymbol{\nabla}\xi\cdot\boldsymbol{\nabla}\mathbf{U}\|_{\mathbf{A}_0^{-1}}+\|\boldsymbol{\nabla}\eta\cdot\boldsymbol{\nabla}\mathbf{U}\|_{\mathbf{A}_0^{-1}}+\|\boldsymbol{\nabla}\zeta\cdot\boldsymbol{\nabla}\mathbf{U}\|_{\mathbf{A}_0^{-1}}}\right]^{1/2} \tag{29}$$

where (ξ,η,ζ) are the canonical reference element coordinates and $\mathbf{A}_0^{-1}$ is the mapping from conservation to entropy variables. The physical-domain to reference-domain element transformation terms $(\boldsymbol{\nabla}\xi,\boldsymbol{\nabla}\eta,\boldsymbol{\nabla}\zeta)$ are $\mathcal{O}(1/h)$, hence δ is proportional to h. Thus, in regions of appreciable δ, (28) reduces to an $\mathcal{O}(h)$ approximation of (18) for a piecewise-linear finite element approximation.

Note that in (29) the numerator corresponds to the residual of the steady Euler equations, hence (28) is consistent with (18) only for this special case. A modified form is employed in the present work and is defined as

$$\delta=\left[\frac{\left\|\dfrac{\partial \mathbf{U}}{\partial t}+\mathbf{A}_i\dfrac{\partial \mathbf{U}}{\partial x_i}-\dfrac{\partial}{\partial x_i}\left(\mathbf{K}_{ij}\dfrac{\partial \mathbf{U}}{\partial x_j}\right)\right\|_{\mathbf{A}_0^{-1}}}{\|\boldsymbol{\nabla}\xi\cdot\boldsymbol{\nabla}\mathbf{U}\|_{\mathbf{A}_0^{-1}}+\|\boldsymbol{\nabla}\eta\cdot\boldsymbol{\nabla}\mathbf{U}\|_{\mathbf{A}_0^{-1}}+\|\boldsymbol{\nabla}\zeta\cdot\boldsymbol{\nabla}\mathbf{U}\|_{\mathbf{A}_0^{-1}}}\right]^{1/2} \tag{30}$$

The time derivative term was absent in the original formulations and has been added here for use in time-accurate simulations. Additionally, the diffusive term in the numerator is included so that consistency with (23) is maintained. That is, this form of the shock-capturing parameter will vanish when the discrete solution satisfies (23).

Note that the combination of streamline upwinding and shock capturing required to obtain stable solutions with the finite element method is similar to the upwinding and limiting which is characteristic of total-variation-diminishing (TVD) finite difference and finite volume schemes. TVD schemes typically employ an upwind treatment of the inviscid flux terms which is sufficient

DOI: 10.1002/fld

to stabilize convective-dominated flows. However, flux or slope limiters, which are designed to restore monotonicity, are required in the presence of strong shock waves. The shock-capturing function used in the present scheme is similar to the use of limiters in that it attempts to restore monotonicity in regions of large gradients such as shock waves. (In general, monotonicity can only be guaranteed for the one-dimensional case.) Both TVD finite volume schemes and the current finite element schemes reduce to first order at shock waves in an attempt to restore monotonicity of the solution.

3.3. Boundary conditions

Supersonic and hypersonic viscous and inviscid flows are considered in the subsequent numerical studies. For this class of flows the Navier–Stokes equations form a mixed parabolic–hyperbolic set of partial differential equations. Three classes of boundary conditions relevant to the problem class of interest follow.

3.3.1. Supersonic inflow. At supersonic inflow boundaries the characteristics of the system are all directed into the domain, and hence each component of the system may be specified as an essential boundary condition. In general, for aerothermodynamic applications the freestream density, velocity, and temperature are usually prescribed. With these primitive variables specified the conservation variables may be determined.

3.3.2. Solid body. Inviscid flows: The Euler equations are a first-order system of partial differential equations, which is in contrast to the second-order Navier–Stokes equations. One consequence of this is that the Euler equations admit one less boundary condition at solid walls. The familiar no-slip condition for viscous flows degenerates to the no-penetration condition for the Euler condition, requiring only that the normal component of the velocity vanish on solid walls. That is

$$\mathbf{u}\cdot\hat{\mathbf{n}}=0 \quad \text{on } \Gamma_s \tag{31}$$

The proper way to impose this boundary condition has been discussed at length in the literature and several options have been proposed. One approach is to impose an explicit correction step in a time marching scheme to remove any normal component of velocity at no-penetration boundaries [5]. This approach is not used in this work because it is critical that the boundary condition be implemented in a fully implicit manner if the convergence properties of an implicit formulation are to be retained. Another approach is to transform the Cartesian coordinate axes $(\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}})$ into a normal-tangential set $(\hat{\boldsymbol{\xi}}, \hat{\boldsymbol{\eta}}, \hat{\mathbf{n}})$ and then impose an essential boundary condition on the normal velocity component [15, 17]. This approach has the benefit of imposing the boundary condition implicitly, but it requires the definition of a unique normal $\hat{\mathbf{n}}$ for nodes on the boundary. For the faceted boundary description that results from discretizing a smooth body with a mesh the normal is not defined at the nodes of elements, and produces local error in the solution, particularly at sharp corners.

In this work an alternate approach is taken in which the boundary condition is implemented through manipulation of the weak statement (28). To obtain the weak form of the boundary condition it is necessary to integrate the convective term in the first integral of Equation (28)

DOI: 10.1002/fld

by parts, yielding

$$\begin{aligned}\int_{\Omega}\left(\mathbf{W}\cdot\frac{\partial \mathbf{U}}{\partial t}-\frac{\partial \mathbf{W}}{\partial x_i}\mathbf{A}_i\mathbf{U}\right)\mathrm{d}\Omega \\ +\sum_{e=1}^{n_{el}}\int_{\Omega_e}\tau_{\text{SUPG}}\frac{\partial \mathbf{W}}{\partial x_k}\cdot\mathbf{A}_k\left(\frac{\partial \mathbf{U}}{\partial t}+\mathbf{A}_i\frac{\partial \mathbf{U}}{\partial x_i}\right)\mathrm{d}\Omega \\ +\sum_{e=1}^{n_{el}}\int_{\Omega_e}\delta\left(\frac{\partial \mathbf{W}}{\partial x_i}\cdot\frac{\partial \mathbf{U}}{\partial x_i}\right)\mathrm{d}\Omega+\int_{\Gamma}\mathbf{W}\cdot\mathbf{f}\,\mathrm{d}\Gamma=0\end{aligned} \tag{32}$$

where the homogeneity of $\mathbf{F}_i(\mathbf{U})$ has been invoked by recognizing $\mathbf{F}_i(\mathbf{U})=(\partial\mathbf{F}_i/\partial\mathbf{U})\mathbf{U}$. In (32) $\mathbf{f}=\mathbf{F}\cdot\hat{\mathbf{n}}$ is the normal component of the inviscid flux $\mathbf{F}$ on the boundary Γ and for three-dimensional flows is

$$\mathbf{F}\cdot\hat{\mathbf{n}}=\begin{bmatrix}\rho\mathbf{u}\cdot\hat{\mathbf{n}}\\(\rho\mathbf{u}\cdot\hat{\mathbf{n}})u+Pn_x\\(\rho\mathbf{u}\cdot\hat{\mathbf{n}})v+Pn_y\\(\rho\mathbf{u}\cdot\hat{\mathbf{n}})w+Pn_z\\(\rho\mathbf{u}\cdot\hat{\mathbf{n}})H\end{bmatrix}=\begin{bmatrix}0\\Pn_x\\Pn_y\\Pn_z\\0\end{bmatrix}\quad\text{on }\Gamma_s \tag{33}$$

where $H=E+P/\rho$ and $\hat{\mathbf{n}}=n_x\hat{\mathbf{i}}+n_y\hat{\mathbf{j}}+n_z\hat{\mathbf{k}}$. The implicit contribution for this boundary term follows directly from invoking the homogeneity of the normal component of the inviscid flux:

$$\mathbf{F}\cdot\hat{\mathbf{n}}=(\mathbf{A}_1n_x+\mathbf{A}_2n_y+\mathbf{A}_3n_z)\mathbf{U}=\mathbf{A}_n\mathbf{U} \tag{34}$$

Using (34) in (32) gives

$$\begin{aligned}\int_{\Omega}\left(\mathbf{W}\cdot\frac{\partial \mathbf{U}}{\partial t}-\frac{\partial \mathbf{W}}{\partial x_i}\mathbf{A}_i\mathbf{U}\right)\mathrm{d}\Omega \\ +\sum_{e=1}^{n_{el}}\int_{\Omega_e}\tau_{\text{SUPG}}\frac{\partial \mathbf{W}}{\partial x_k}\cdot\mathbf{A}_k\left(\frac{\partial \mathbf{U}}{\partial t}+\mathbf{A}_i\frac{\partial \mathbf{U}}{\partial x_i}\right)\mathrm{d}\Omega \\ +\sum_{e=1}^{n_{el}}\int_{\Omega_e}\delta\left(\frac{\partial \mathbf{W}}{\partial x_i}\cdot\frac{\partial \mathbf{U}}{\partial x_i}\right)\mathrm{d}\Omega+\int_{\Gamma}\mathbf{W}\cdot(\mathbf{A}_n\mathbf{U})\,\mathrm{d}\Gamma=0\end{aligned} \tag{35}$$

This formulation requires the normal direction for each element residing on the no-penetration surface *on the boundary face*, which is well defined even for faceted discretizations. In numerical calculations the boundary flux is computed using the well-defined normal for each element segment coincident with the boundary.

Viscous flows: At the surface of a body in a viscous flow the no-slip, isothermal boundary condition is applied. The no-slip condition is implemented simply by specifying appropriate essential boundary conditions for the momentum components of the equation system. The isothermal boundary condition is implemented as an essential condition on the total energy per unit volume, ρE. At a no-slip wall we have

$$\rho E = \rho\left(e + \frac{\mathbf{u}\cdot\mathbf{u}}{2}\right) = \rho e = \rho c_v T_w$$

which is implemented as the essential, implicit boundary condition $\rho E - \rho c_v T_w = 0$.

3.3.3. Supersonic outflow. At supersonic outflow boundaries the state is defined entirely by the internal conditions. However, as pointed out by Hauke and Hughes, it is important to include the viscous boundary terms that result from the integration by parts performed in Equation (28) [19]. These boundary term contributions are computed at viscous supersonic outflow boundaries and are included in the system matrix.

4. FINITE ELEMENT FORMULATION

Upon introducing a finite element discretization and corresponding basis to define the approximate solution $\mathbf{U}_h$ and test functions $\mathbf{W}_h$, and substituting into (28), the corresponding approximate finite element formulation has the form: Find $\mathbf{U}_h$ satisfying the essential boundary and initial conditions such that

$$\begin{aligned}&\int_\Omega \left[\mathbf{W}_h \cdot \left(\frac{\partial \mathbf{U}_h}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}_h}{\partial x_i}\right) + \frac{\partial \mathbf{W}_h}{\partial x_i} \cdot \left(\mathbf{K}_{ij} \frac{\partial \mathbf{U}_h}{\partial x_j}\right)\right] \mathrm{d}\Omega \\ &\quad + \sum_{e=1}^{n_{el}} \int_{\Omega_e} \tau_{\mathrm{SUPG}} \frac{\partial \mathbf{W}_h}{\partial x_k} \cdot \mathbf{A}_k \left[\frac{\partial \mathbf{U}_h}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}_h}{\partial x_i} - \frac{\partial}{\partial x_i}\left(\mathbf{K}_{ij} \frac{\partial \mathbf{U}_h}{\partial x_j}\right)\right] \mathrm{d}\Omega \\ &\quad + \sum_{e=1}^{n_{el}} \int_{\Omega_e} \delta \left(\frac{\partial \mathbf{W}_h}{\partial x_i} \cdot \frac{\partial \mathbf{U}_h}{\partial x_i}\right) \mathrm{d}\Omega - \oint_\Gamma \mathbf{W}_h \cdot \mathbf{g}_h \, \mathrm{d}\Gamma = 0 \end{aligned} \tag{36}$$

for all admissible test functions $\mathbf{W}_h$.

More specifically, let us expand $\mathbf{U}_h(\mathbf{x},t)$ and $\mathbf{F}_i(\mathbf{x},t)$ in terms of the finite element basis functions:

$$\mathbf{U}_h(\mathbf{x},t) = \sum_j \phi_j(\mathbf{x}) \mathbf{U}_h(\mathbf{x}_j,t) \tag{37}$$

$$\mathbf{F}_i(\mathbf{x},t) = \sum_j \phi_j(\mathbf{x}) \mathbf{F}_i(\mathbf{x}_j,t) \tag{38}$$

where $\mathbf{U}_h(\mathbf{x}_j,t)$ and $\mathbf{F}_i(\mathbf{x}_j,t) = \mathbf{A}_i(\mathbf{U}_h(\mathbf{x}_j,t))\mathbf{U}_h(\mathbf{x}_j,t)$ are the nodal solution values and nodal inviscid flux components at time t, respectively. In this work a standard piecewise-linear Lagrange basis is chosen for $\{\phi\}$, which yields a nominally second-order accurate scheme. Since the focus here is on supersonic flows that exhibit shock waves no attempt has been made to achieve higher-order spatial discretizations. (As discussed in Section 3.2, the scheme is locally first-order accurate in the vicinity of shocks.) However, previous work with a similar formulation for the compressible Navier–Stokes equations suggests that the current scheme could easily be extended to higher order for flows without shocks simply by using a higher-order finite element basis [20].

Note the particular discretization chosen in Equation (38) for the inviscid flux term. This approach is motivated by results that show that for the model Burger's equation this grouped discretization yields slightly higher accuracy than the ungrouped scheme [21]. This approach is one of the several alternatives presented by Morgan and Peraire for the Galerkin finite element method with the explicit addition of diffusion [22]. Recently this approach has received renewed attention in flux-corrected transport discretizations for multidimensional conservation laws [23, 24]. Applied in the current work, this approach is in contrast to previous SUPG discretizations for compressible flows in which the inviscid flux terms are evaluated using the interpolated discrete solution (e.g. as in [17, 15, 19, 25]). To illustrate the difference consider the expansion of the steady analog to (38) using (21)

$$\mathbf{F}_i(\mathbf{x}) = \sum_j \phi_j(\mathbf{x})\mathbf{F}_i(\mathbf{x}_j) = \sum_j \phi_j(\mathbf{x})\mathbf{A}_i(\mathbf{U}(\mathbf{x}_j))\mathbf{U}(\mathbf{x}_j) \tag{39}$$

in contrast to the typical approach in which

$$\mathbf{F}_i(\mathbf{x}) = \mathbf{A}_i(\mathbf{U}(\mathbf{x}))\mathbf{U}(\mathbf{x}) \tag{40}$$

where $\mathbf{U}(\mathbf{x})$ is interpolated from nodal values as in (37).

Numerical experiments suggest that this choice of inviscid flux discretization improves the stability of the numerical scheme. One possible explanation for this behavior may be that in (39) the inviscid flux is only computed at the nodes $\mathbf{x}_j$ of an element and interpolated in the interior, while (40) evaluates the inviscid flux directly in the interior using the interpolated values $\mathbf{U}(\mathbf{x})$.

Figure 1 examines why this procedure may enhance the stability of the formulation. The figure considers a one-dimensional, steady, inviscid, normal shock at Mach 5. For this simple case the governing equations reduce to

$$\frac{\partial}{\partial x}(\rho u) = \frac{\partial}{\partial x}(\rho u^2 + P) = \frac{\partial}{\partial x}(\rho u H) \equiv 0 \tag{41}$$

which implies that ρu, $\rho u^2 + P$, and $\rho u H$ are all constant.

Figure 1(a) presents the scenario in which the exact solution is captured on three piecewise-linear finite elements of unit length. The $(\rho, \rho u, \rho E)$ lines in the figure depict the conserved variables for the nodally exact solution interpolated linearly in the finite element basis. The (P, u, T) lines are the reconstructed primitive variables, which are highly nonlinear as they are in general rational functions of the conserved variables.

Figure 1(b) plots the inviscid flux vector components for both the traditional approach and the discretization given by (39). Note that for the traditional approach Equation (41) is not satisfied within the element containing the shock; hence, this scheme is incapable of representing the nodally exact solution. By contrast, for the alternate choice of (39) Equation (41) is satisfied exactly.

Recalling Equation (30), the inability to represent a nodally exact solution implies that the shock-capturing operator will always be active in the traditional approach. The approach proposed here can satisfy the nodally exact solution and, therefore, is capable of converging to solutions in which δ vanishes throughout the domain. Further, the distribution of $(\rho u^2 + P)$ and $\rho u H$ interior to the element containing the shock are of high order for the traditional approach. This is important because, in practice, the integrals in the finite element weak statement (36) are approximated using numerical quadrature, and this high-order behavior will not be evaluated exactly.

The fact that both schemes recover identical inviscid flux values at the nodes is also important. This suggests that for a nodal quadrature rule both schemes should exhibit similar performance.

DOI: 10.1002/fld

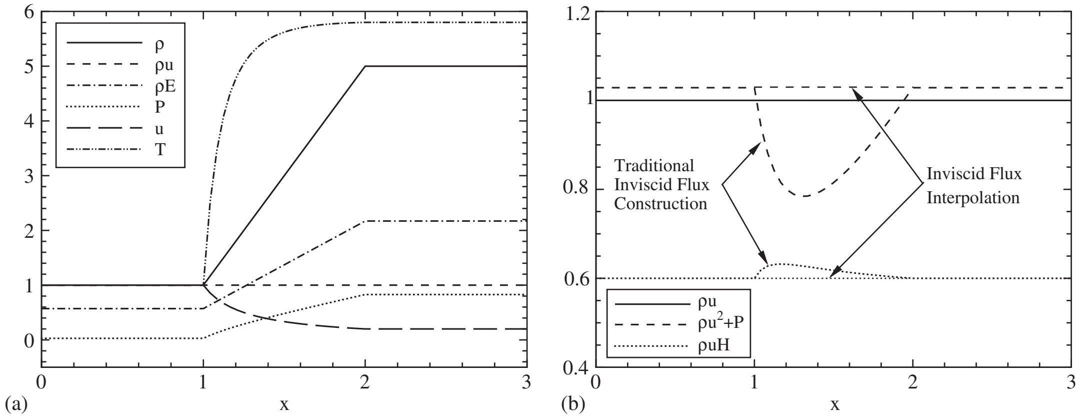

Figure 1. A steady normal shock at Mach 5 spanning three notional elements: (a) linearly interpolated conservation variables and reconstructed primitive variables and (b) linearly interpolated and reconstructed inviscid flux vector components.

This conjecture is supported by recent work in which Kessler and Awruch consider an explicit Taylor–Galerkin finite element method for the Navier–Stokes equations in thermochemical nonequilibrium [9]. In this work the authors evaluate the element integrals *a priori* in closed form using Gauss–Lobatto quadrature so that at each explicit time step costly numerical integration is avoided. Given the behavior shown in Figure 1(b) their approach may have benefited from this enhanced stability by sampling the inviscid flux only at the element nodes.

5. SOLUTION METHODOLOGY

Equations (36) form a transient, tightly coupled nonlinear system for the unknown nodal values $\mathbf{U}_h(\mathbf{x}_j, t)$. Even when a steady solution to the governing equations is sought Equations (36) are often solved with a pseudo-time continuation strategy. That is, even for steady problems, the unsteady equations are often integrated in time until steady state is reached. This is especially the case for compressible flows containing shock waves because strong gradients that occur in the flow imply an extremely small zone of attraction for nonlinear implicit solution schemes such as Newton's method [26, 27]. Algorithms for solving this type of transient system fall broadly into two categories: explicit and implicit.

Since the present work seeks to use adaptive meshing techniques to locally resolve fine features of the flow (thus decreasing h), the h dependence of Δt for explicit schemes is particularly unattractive [5]. The cost for this increased stability is the need to solve (at least approximately) a nonlinear implicit system at each time step of the solution. Preconditioned Krylov subspace iterative methods provide a suitable choice of solvers that are amenable to parallel solution and are efficient for the problems of interest here [28].

The remainder of this section describes (1) the domain decomposition approach used to achieve parallelism, (2) the time integration scheme, and (3) linearization strategies used for both steady-

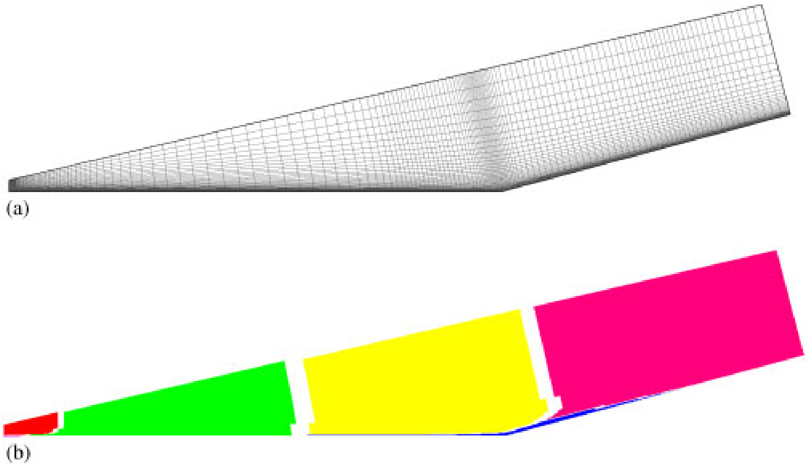

Figure 2. Computational mesh and resulting parallel-domain decomposition for a compression ramp: (a) computational mesh (every other point shown) and (b) parallel-domain decomposition.

state and time-accurate flows. The iterative techniques used to solve the resulting linear systems will also be briefly discussed.

5.1. Domain decomposition

A standard nonoverlapping domain decomposition scheme is used in which a unique set of elements is assigned to each processor used in the simulation (see Reference [1] and references therein). The METIS unstructured graph partitioning library [29] is used to create a weighted partition that attempts to balance the computational load incurred for a hybrid-element unstructured mesh. An example of decomposition into six subdomains is shown in Figure 2 for the case of a two-dimensional compression ramp. Each subdomain contains approximately the same number of elements. Owing to the fine mesh spacing at the leading edge of the cylinder, the two forward-most subdomains are barely visible in the figure.

The domain decomposition approach allows element contributions to the global implicit system to be calculated in parallel. That is, each processor will form the system matrix contributions only for its local elements. These contributions are then accumulated into a distributed sparse matrix data structure, which is ultimately used in an iterative Krylov subspace technique to approximately solve the linear system [1, 2, 30].

5.2. Time integration

As mentioned previously, steady solutions are often found by time marching the transient governing equations to steady state. In this sense the initial condition is taken at time $t=0$ and the solution is marched in time until $\partial U/\partial t \to 0$. In this way time is essentially a continuation parameter which defines a sequence $(n=1,2,\ldots)$ of solutions $\mathbf{U}_n$ which converge to the steady solution $\mathbf{U}$.

DOI: 10.1002/fld

Table I. First- and second-order accurate time discretization coefficients.

p	α_t	β_t	γ_t
1	$\dfrac{1}{\Delta t_{n+1}}$	$\dfrac{-1}{\Delta t_{n+1}}$	0
2	$-\beta_t-\gamma_t$	$-\left[\dfrac{1}{\Delta t_{n+1}}+\dfrac{1}{\Delta t_n}\right]$	$\dfrac{\Delta t_{n+1}}{\Delta t_n(\Delta t_{n+1}+\Delta t_n)}$

The semidiscrete weak form in Equation (36) is discretized in time using backwards finite difference schemes. Both first- and second-order accurate in time schemes may be derived from Taylor series expansions in time about $\mathbf{U}_h(t_{n+1})=\mathbf{U}_{n+1}$:

$$\mathbf{U}_n=\mathbf{U}_{n+1}+\frac{\partial\mathbf{U}_{n+1}}{\partial t}(t_n-t_{n+1})+\frac{\partial^2\mathbf{U}_{n+1}}{\partial t^2}\frac{(t_n-t_{n+1})^2}{2}+\mathcal{O}((t_n-t_{n+1})^3)$$

$$\mathbf{U}_{n-1}=\mathbf{U}_{n+1}+\frac{\partial\mathbf{U}_{n+1}}{\partial t}(t_{n-1}-t_{n+1})+\frac{\partial^2\mathbf{U}_{n+1}}{\partial t^2}\frac{(t_{n-1}-t_{n+1})^2}{2}+\mathcal{O}((t_{n-1}-t_{n+1})^3)$$

which, upon substituting $t_{n+1}-t_n\equiv\Delta t_{n+1}$ and $t_{n+1}-t_{n-1}=\Delta t_{n+1}+\Delta t_n$, can be rewritten for $\partial\mathbf{U}_{n+1}/\partial t$ as

$$\frac{\partial\mathbf{U}_{n+1}}{\partial t}=\frac{\mathbf{U}_{n+1}}{\Delta t_{n+1}}-\frac{\mathbf{U}_n}{\Delta t_{n+1}}+\frac{\partial^2\mathbf{U}_{n+1}}{\partial t^2}\frac{\Delta t_{n+1}}{2}-\mathcal{O}(\Delta t_{n+1}^2) \tag{42}$$

$$\frac{\partial\mathbf{U}_{n+1}}{\partial t}=\frac{\mathbf{U}_{n+1}}{\Delta t_{n+1}+\Delta t_n}-\frac{\mathbf{U}_{n-1}}{\Delta t_{n+1}+\Delta t_n}+\frac{\partial^2\mathbf{U}_{n+1}}{\partial t^2}\frac{(\Delta t_{n+1}+\Delta t_n)}{2}-\mathcal{O}((\Delta t_{n+1}+\Delta t_n)^2) \tag{43}$$

The familiar backward Euler time discretization follows directly from (42) by recognizing

$$\frac{\partial\mathbf{U}_{n+1}}{\partial t}=\frac{\mathbf{U}_{n+1}}{\Delta t_{n+1}}-\frac{\mathbf{U}_n}{\Delta t_{n+1}}+\mathcal{O}(\Delta t_{n+1}) \tag{44}$$

which provides a first order in time approximation upon neglecting the $\mathcal{O}(\Delta t_{n+1})$ term. As such, this scheme yields a fully implicit problem for $\mathbf{U}_{n+1}$ which may be used when time accuracy is not required. Alternatively, a linear combination of (42) and (43) can be used to annihilate the leading $\partial^2\mathbf{U}_{n+1}/\partial t^2$ term and create a backward, second-order accurate approximation to $\partial\mathbf{U}_{n+1}/\partial t$. This approximation, along with (44), can be generalized in the form

$$\frac{\partial\mathbf{U}_{n+1}}{\partial t}=\alpha_t\mathbf{U}_{n+1}+\beta_t\mathbf{U}_n+\gamma_t\mathbf{U}_{n-1}+\mathcal{O}(\Delta t_{n+1}^p) \tag{45}$$

to yield either a first- or second-order accurate scheme. The weights α_t, β_t, and γ_t are given for $p=1$ and 2 in Table I. Since this second-order scheme requires *two* levels of solution history it is not self-starting. In practice five backward Euler steps are taken to develop the required solution history and to allow rapid transients to subside before applying the second-order scheme.

DOI: 10.1002/fld

5.3. Linearization

After time discretization using either (44) or (45), Equation (36) can be written in residual form for the unknown nodal values $\mathbf{U}_{n+1} \equiv \mathbf{U}_h(t_{n+1})$ as the nonlinear algebraic system:

$$\mathbf{R}(\mathbf{U}_{n+1}) = 0 \tag{46}$$

The goal of this section is to define a sequence of linear problems that, when solved, converge to obtain the solution $\mathbf{U}_{n+1}$ of the nonlinear system (46). Expanding (46) with a Taylor series about iterate $\mathbf{U}^l_{n+1}$ gives

$$\mathbf{R}(\mathbf{U}^{l+1}_{n+1}) = \mathbf{R}(\mathbf{U}^l_{n+1}) + \left[\frac{\partial \mathbf{R}(\mathbf{U}^l_{n+1})}{\partial \mathbf{U}_{n+1}}\right] \delta\mathbf{U}^{l+1}_{n+1} + \mathcal{O}((\delta\mathbf{U}^{l+1}_{n+1})^2) \tag{47}$$

where $\partial\mathbf{R}/\partial\mathbf{U}$ is the Jacobian matrix for the nonlinear system and $\delta\mathbf{U}^{l+1}_{n+1} = \mathbf{U}^{l+1}_{n+1} - \mathbf{U}^l_{n+1}$. Truncating this expansion and setting $\mathbf{R}(\mathbf{U}^{l+1}_{n+1}) = 0$ yields Newton's method:

$$\begin{gathered} 0 = \mathbf{R}(\mathbf{U}^l_{n+1}) + \left[\frac{\partial \mathbf{R}(\mathbf{U}^l_{n+1})}{\partial \mathbf{U}_{n+1}}\right] \delta\mathbf{U}^{l+1}_{n+1} \\ \left[\frac{\partial \mathbf{R}(\mathbf{U}^l_{n+1})}{\partial \mathbf{U}_{n+1}}\right] \delta\mathbf{U}^{l+1}_{n+1} = -\mathbf{R}(\mathbf{U}^l_{n+1}) \end{gathered} \tag{48}$$

which results in an implicit linear system for $\delta\mathbf{U}^{l+1}_{n+1}$ and a sequence of iterates $(l = 0, 1, \ldots)$ which converges to $\mathbf{U}_{n+1}$. It is important to recall that Newton's method exhibits second-order *conditional* convergence. That is, the magnitude of $\mathbf{R}(\mathbf{U}^{l+1}_{n+1})$ decreases quadratically at successive iterates provided that the initial guess $\mathbf{U}^0_{n+1}$ is 'sufficiently close' to the unknown $\mathbf{U}_{n+1}$ [31, 32].

While the full-Newton scheme is conceptually simple the implementation is complicated by the nonlinear dependence of the transport properties on the unknowns (see Equation (8)) and the highly nonlinear nature of the convective terms themselves. In practice, implementing the full-Newton scheme is computationally intensive and, in the case of supersonic flows exhibiting shock waves, is often only of modest benefit. That is, due to the conditional convergence restriction of the method and the sharp gradients or discontinuities that are present in the flowfield, the asymptotic quadratic convergence rate may not be achieved [33]. The implementation of an approximate Newton–Krylov technique to address these issues will be discussed further in the following sections.

5.4. Linear system solution scheme

The Newton scheme results in a series of sparse linear problems of the form

$$\mathbf{K}\delta\mathbf{U}_{n+1} = \mathbf{f} \tag{49}$$

which must be solved to obtain $\mathbf{U}_{n+1}$. For the discretization presented in Section 4 using standard piecewise-linear elements $\mathbf{K}$ is a sparse, nonsymmetric, nonsingular matrix. Given the size and sparseness of $\mathbf{K}$ it is natural to use preconditioned Krylov subspace iterative techniques to approximate $\delta\mathbf{U}_{n+1}$ [34, 35]. The essential kernel of these techniques is the computation of the matrix–vector product $\mathbf{y} = \mathbf{K}\mathbf{x}$. Two techniques for providing this kernel will be discussed, the first stores the sparse matrix and computes the matrix–vector product explicitly; the second computes the action of the matrix–vector product in a 'matrix-free' sense.

5.4.1. Sparse matrix approach. One straightforward technique for solving (49) is to build the system matrix **K** and right-hand side vector **f**. Since the matrix is large yet sparse care must be taken to store it efficiently. In the present work the parallel sparse matrix format implemented in the PETSc toolkit is used, as are the PETSc iterative solvers [30]. When the system matrix is constructed explicitly it may then be copied and modified to serve as a preconditioner as well. In the current work a standard parallel block-Jacobi ILU-0 preconditioner is used [34, 35]. Once the system matrix and preconditioner are formed the required matrix–vector products are computed directly.

5.4.2. Matrix-free approach. Recall from Equation (48) the particular form of the implicit system to be solved:

$$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right]\delta\mathbf{U}=-\mathbf{R}(\mathbf{U})$$

For this special case the action of the matrix–vector product $[\partial\mathbf{R}/\partial\mathbf{U}]\delta\mathbf{U}$ is nothing more than the derivative of **R** in the direction specified by $\delta\mathbf{U}$, and may be approximated within $\mathcal{O}(\varepsilon)$ for finite ε as

$$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right]\delta\mathbf{U}\approx\frac{\mathbf{R}(\mathbf{U}+\varepsilon\delta\mathbf{U})-\mathbf{R}(\mathbf{U})}{\varepsilon} \tag{50}$$

From Equation (50) it is clear that the required matrix–vector product may be approximated by differencing successive residual evaluations. It is in this sense that the scheme is matrix free: the actual system matrix need not be explicitly formed. All that is required is the capability to evaluate the discrete residual $\mathbf{R}(\mathbf{U})$. Of course, for practical applications some form of preconditioning must be applied to the linear system. Depending on the implementation of this preconditioning, the composite scheme may store some approximation of the system matrix. Still, one attractive feature of the matrix-free approach is that it can require substantially less memory than the sparse matrix approach.

Perhaps the most compelling reason to use the matrix-free approach is that it directly yields a quasi-Newton formulation. That is, the finite difference approximation properly accounts for *all* the nonlinearities in the system. This is especially attractive from an algorithm development perspective. For example, alternate shock-capturing terms, SUPG weighting functions, equations of state, and transport property definitions can all be implemented simply by defining their contribution to the discrete residual. Their contribution to the quasi-Newton iteration simply falls out through the approximate matrix–vector product (50).

5.5. Software implementation

All computations employ the PETSc toolkit from Argonne National Laboratory [30] to solve the parallel implicit linear systems using the generalized minimum residual Krylov subspace technique [36] with preconditioning. The preconditioner is of parallel block Jacobi-type where each processor sub-block uses an overlapping additive Schwartz method with an incomplete lower–upper factorization at the sub-block level with no fill (ILU-0). Spatial integration is performed with Gauss quadrature rules sufficient to integrate third–order polynomials exactly.

6. APPLICATIONS

This section presents two applications used to validate the finite element algorithm described in Section 4. Supersonic inviscid and hypersonic, laminar viscous flows in two dimensions are considered here.

6.1. Inviscid flow over a cylinder

6.1.1. Geometry and flow conditions. Two-dimensional inviscid Mach 3 flow over a circular cylinder is an established benchmark problem [37, 38] and is introduced here to investigate the performance of the implicit formulation. Because of the hyperbolic nature of the problem, the exterior flow problem is posed on a finite subdomain shown in Figure 3 with uniform far-field data prescribed on the inflow boundary. The computational grid for this case is mapped from the unit square $[0,1]\times[0,1]$ in the (ξ,η) plane by [38]

$$x(\xi,\eta)=(R_x-(R_x-R_\mathrm{c})\xi)\cos(\theta(2\eta-1)) \tag{51}$$

$$y(\xi,\eta)=(R_y-(R_y-R_\mathrm{c})\xi)\sin(\theta(2\eta-1)) \tag{52}$$

where the cylinder radius $R_\mathrm{c}=0.5$, the upstream boundary of the computational domain is given by $R_x=1.5, R_y=3$, and $\theta=5\pi/12$. A coarse mesh is shown in Figure 3 with $n_\xi\times n_\eta=30\times 40$ elements in the normal and circumferential directions, respectively. Mesh convergence will be considered in Section 6.1.3. The particular form of this mapping has been used in high-order finite difference discretizations as it yields a smooth, differentiable mapping from computational to physical space.

This example is a prototype for convection-dominated supersonic and hypersonic problems that arise in aerospace engineering. The performance of the finite element algorithm presented in the previous sections will be examined in detail for this example. The results of the numerical experiments performed in this section will be generalized and applied to more physically complicated flow phenomena in later application studies.

The simulation is initialized with uniform freestream values and marched in time until steady state is reached. A supersonic boundary condition is imposed on the upstream boundary in which the conserved variables $[\rho,\rho u,\rho v,\rho E]^\mathrm{T}$ are specified as essential boundary conditions. At the outflow boundary the flow is supersonic, and hence no outflow boundary conditions are specified for this inviscid flow. The no-penetration boundary condition $\mathbf{u}\cdot\hat{\mathbf{n}}=0$ holds on the cylinder surface and is enforced as a natural boundary condition through the boundary integral in the weak statement as described in Section 3.3.

6.1.2. Flowfield and stagnation line properties. Figure 4 illustrates the steady-state flowfield for this case. For this inviscid case the governing Euler equations (32) are hyperbolic and admit discontinuous solutions. As expected, the cylinder produces a strong bow shock across which the density, velocity, and pressure jump, in accordance with the Rankine–Hugoniot equations.

DOI: 10.1002/fld

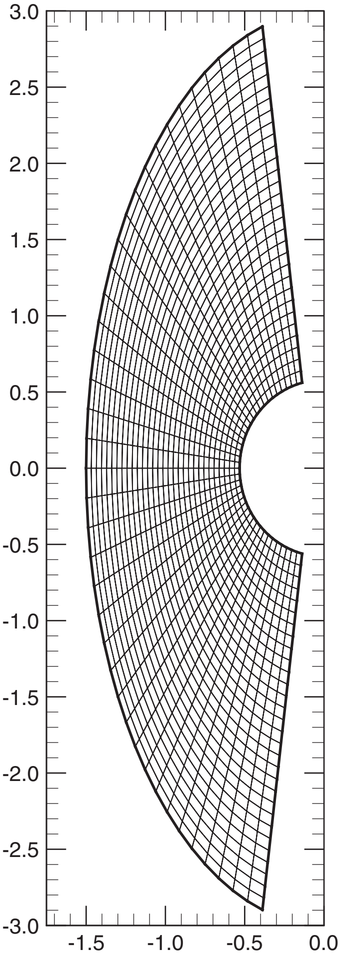

Figure 3. Coarse computational grid for Mach 3 flow over a cylinder.

Figure 5 shows the flowfield properties along the stagnation line *versus* nondimensional distance x/R_N, where R_N is the cylinder radius. It is apparent from the figure that the bow shock is located at approximately $0.7\,R_\mathrm{N}$ upstream from the stagnation point, which agrees well with experimentally measured values [39]. As expected, the pressure, density, and temperature all increase across the shock wave while the Mach number decreases. The computed jumps are in excellent agreement with theoretical predictions as evident in Table II. This indicates that the numerical scheme is properly reproducing the shock jump conditions, which is expected for any viable formulation based on the conservation form of the governing equations (1)–(3). Note that other formulations that are not based on the conservation form of the governing equations are possible and may provide simpler discretizations [40]. In general, however, these formulations *will not* converge to a solution that satisfies the Rankine–Hugoniot equations. Therefore, such schemes would not produce proper jump conditions for this case.

DOI: 10.1002/fld

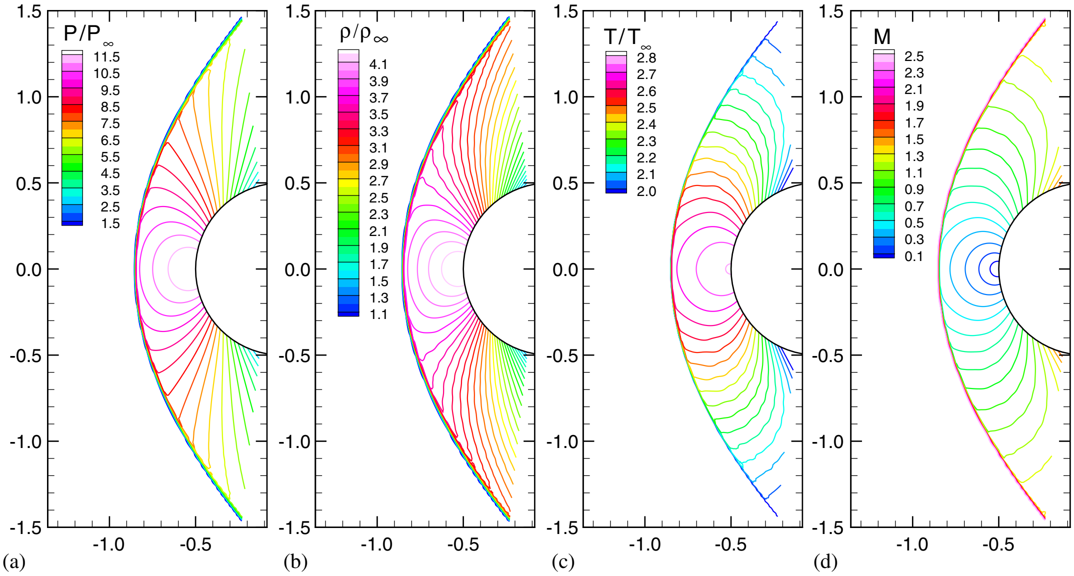

Figure 4. Illustration of flowfield for Mach 3 flow over a cylinder: (a) pressure; (b) density; (c) temperature; and (d) Mach number.

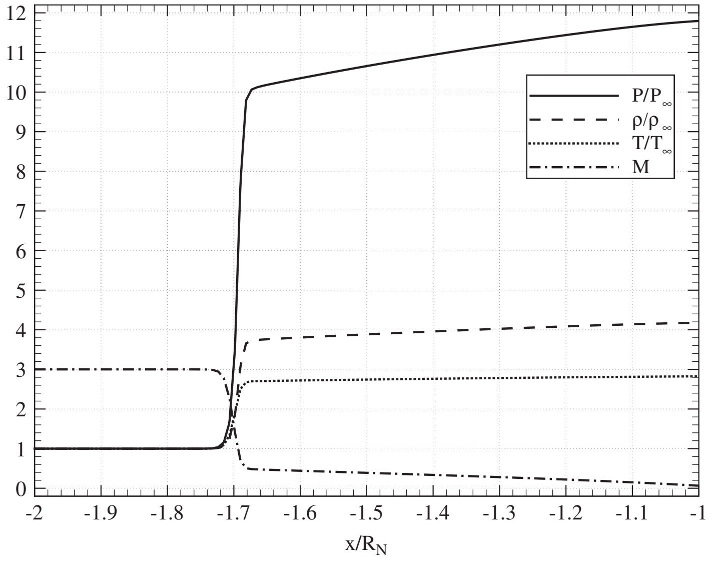

Figure 5. Stagnation line profile for Mach 3 flow over a cylinder.

This example also serves as a good test case for assessing the shock–capturing operator δ. The stagnation line profiles depicted in Figure 5 show that the shock is captured over 2–3 elements without spurious oscillations when the shock is essentially aligned with the grid.

DOI: 10.1002/fld

Table II. Computed and theoretical jump values for a Mach 3 normal shock.

	P/P_∞	ρ/ρ_∞	T/T_∞
Theory [41]	10.33	3.857	2.679
Computation	10.34	3.854	2.683

6.1.3. Convergence. In order to better characterize both the transient and nonlinear discretization schemes described in Section 5, a series of numerical experiments were conducted which varied (1) the order of the time discretization and (2) the number of subiterations used to solve the discrete, nonlinear, implicit problem that results at each time step. For these higher fidelity numerical experiments, a mesh of $n_\xi \times n_\eta = 120\times 160$ elements was used. A discussion of mesh convergence will be presented following the algorithmic performance investigation.

Temporal convergence to steady state: The absence of viscosity-induced separation, wake flow, and shock interaction produces a fairly simple, steady flowfield. It is therefore expected that the numerical scheme will converge to a steady state, which is assumed to be reached when the discrete unsteady residual, $\Delta\mathbf{U}/\Delta t$, falls below a user-specified tolerance ε_{ss} in the maximum norm. That is, steady state is assumed when

$$\mathscr{R}_n \equiv \left\| \frac{\Delta\mathbf{U}_n}{\Delta t_n} \right\|_\infty < \varepsilon_{\text{ss}} \tag{53}$$

where ε_{ss} is the steady-state solution tolerance and was taken as $\varepsilon_{\text{ss}} = 10^{-12}$.

The time convergence history for this case is shown in Figure 6(a). The simulation begins with the domain initialized to freestream conditions everywhere and a user-specified initial time step Δt_0 is used to advance the solution, which was taken here as 2×10^{-3}. The time step is allowed to grow geometrically with the relative change in the unsteady residual measured over k time steps. Explicitly

$$\Delta\bar{t}_{n+1} = \Delta t_{n-k}\left[\frac{\mathscr{R}_{n-k}}{\mathscr{R}_n}\right]^r \tag{54}$$

$$\Delta t_{n+1} = \min(\Delta\bar{t}_{n+1}, \Delta t_{\max}) \tag{55}$$

where r is the geometric time step growth rate, which was fixed at 1.2 in this case. The time step size is updated every $k=5$ time steps and the maximum allowable time step $\Delta t_{\max}=1$ corresponds to the amount of time required for a fictitious point in the freestream to be convected one cylinder diameter.

One immediate observation from the numerical experiments is that the first- and second-order time discretizations have similar transient convergence behavior. The convergence history exhibits two distinct phases: (1) the pre-asymptotic phase in which the bow shock develops and travels upstream to its steady location and (2) the asymptotic phase where large-scale changes in the flowfield have subsided and the remaining transient behavior is damped out.

In the pre-asymptotic phase the time discretization order of accuracy has little influence on the convergence rate. This is consistent with the observation that, during this highly nonlinear process, the time step size must be limited to achieve convergence of the nonlinear subproblem. In the asymptotic phase the convergence rate of the two schemes is again comparable. Since the only

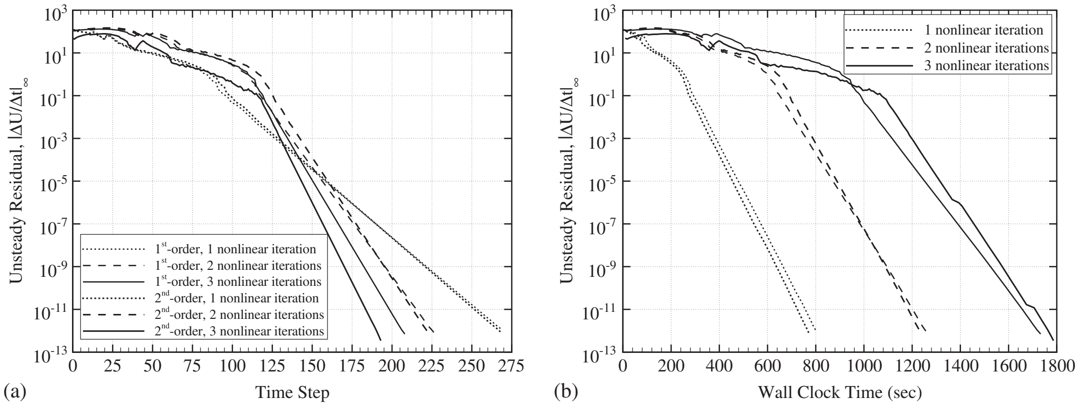

Figure 6. Convergence history for Mach 3 flow over a cylinder for a range of nonlinear solver subiterations and time discretizations: (a) time step and (b) wall clock.

added cost associated with the second-order scheme is the storage of an additional solution vector, there seems to be no compelling motivation to use the first-order scheme. Additionally, using the second-order scheme will more accurately capture any unsteady flow phenomena that might occur for a given configuration.

It is interesting that the current finite element scheme does not exhibit the nonlinear residual convergence stagnation noted by Catabriga and Coutinho when using a very similar SUPG finite element scheme for the conservation variables [25]. This difference must be due to either (i) the inviscid flux treatment used in the present scheme or (ii) the integration by parts performed on the inviscid flux terms since the remainder of the discretized weak form is identical.

Nonlinear solver accuracy: A first glance at Figure 6(a) might suggest that the algorithm performs better when specifying a larger number of nonlinear iterations per time step. In this case the figure is misleading because, in the current implementation, the computational cost of each time step is proportional to the number of nonlinear iterations used.

Figure 6(b) shows the unsteady residual *versus* wall-clock time for the cases previously mentioned. Note that the trend observed in the previous figure is now reversed. The wall-clock time is seen to increase directly with the number of nonlinear iterates. Thus, even though the case of three nonlinear iterations per time step converges in the shortest number of physical time steps it clearly is the most expensive in terms of wall-clock time.

This study supports the common practice of performing only one nonlinear solution iterate per time step when considering steady flows. It must be emphasized that this truncated nonlinear problem is essentially a pseudo-time continuation procedure and, that for cases where transient behavior is of interest, the nonlinear problem at each time step must be solved to an accuracy commensurate with other aspects of the algorithm.

While the wall-clock time required increases with the number of nonlinear solver subiterations, it is interesting that it does not increase linearly. This is because at each subsequent nonlinear iteration the underlying linear system is solved with an iterative Krylov subspace method which benefits from the accuracy of the initial iterate. The linear solver for each subsequent nonlinear iteration in general converges more rapidly than the one before; hence, the overall scaling is not linear with the number of nonlinear subiterations.

DOI: 10.1002/fld

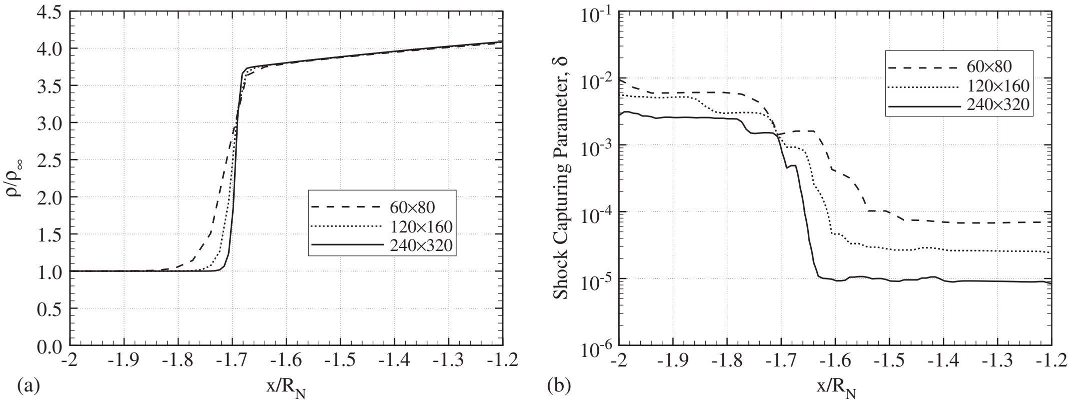

Figure 7. Stagnation line profiles for Mach 3 flow over a cylinder at a series of mesh resolutions: (a) density and (b) shock-capturing parameter.

Note that in this work the preconditioning matrix used in the linear solver is assembled and factored for each linear solve. Follow-on work could consider the performance trade-off between recomputing the preconditioning matrix *versus* fixing the preconditioner for some number of iterations and accepting a less accurate approximate inverse matrix. Similar techniques were investigated by Barth [42] for incompressible non-Newtonian fluids and show promise for reducing the computational effort required per time step.

Mesh convergence: A series of nested meshes consisting of $n_\xi \times n_\eta = 60\times 80$, 120×160, and 240×320 elements was used. Typical results for the stagnation line density profile are shown in Figure 7(a). The figure shows the density jump that occurs across the bow shock along the stagnation line. For the series of nested meshes both location and strength (indicated by the density jump across the shock) of the bow shock are consistent for all the three simulations. Interestingly, the location of the trailing edge of the shock wave is more consistent across the series of meshes than the leading edge of the shock. One may conjecture that this is associated with the behavior of the shock-capturing function. Figure 7(b) shows the shock-capturing function, δ, along the stagnation line for the three nested meshes. Interestingly, in all cases the shock-capturing parameter peaks *upstream* of the bow shock in the uniform freestream. Recall from (30) that $\delta \propto 1/\nabla \mathbf{U}$; hence, this behavior is not surprising. Since the flow is uniform in this region the artificial diffusion term weighted by δ is inconsequential; hence this behavior, albeit unsettling, does not adversely affect the quality of the solution.

The parameter decreases nearly monotonically through the bow shock and reaches a steady, low value in the post-shock stagnation region. The post-shock value of the shock-capturing parameter decreases from approximately 10^{-4} to 10^{-5} with two levels of uniform mesh refinement. Since this corresponds to a factor of four reductions in the mesh spacing h, for this case δ appears to decrease superlinearly with h. For this case $\delta \propto h^{1.5}$.

6.2. *Viscous hypersonic flow over a compression ramp*

This case considers laminar hypersonic flow over a 15° compression ramp. The freestream Mach number is 11.68, temperature is 64.6 K, and unit Reynolds number is 558 0001/m. Figure 8

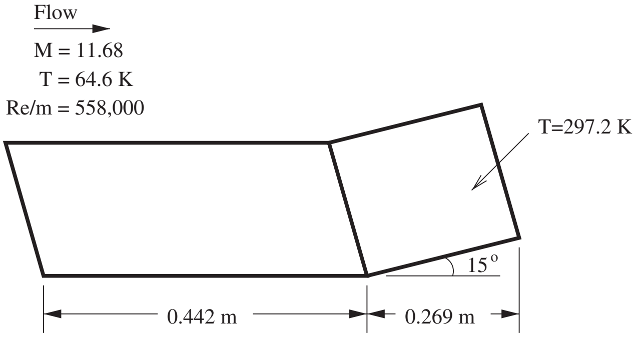

Figure 8. Illustration of geometry and boundary conditions for hypersonic shock ramp problem.

illustrates the ramp geometry and boundary conditions. The Reynolds number based on the flat plate length is $Re_L = 246\,636$ [43–45].

6.2.1. Motivation. Supersonic flow over a compression ramp is of interest in aerodynamic applications because it is analogous to a control surface deflecting into a supersonic flow. For this case a weak shock will develop at the leading edge of the plate due to displacement thickness effects from the viscous boundary layer. The boundary layer thickness will grow relatively quickly down the plate length due to the high edge Mach number. The compression ramp will produce an additional weak shock which is required to deflect the incoming supersonic flow. This weak shock causes a pressure increase on the compression ramp surface which can feed upstream in the subsonic portion of the boundary layer. This adverse pressure gradient, in turn, will affect the laminar boundary layer itself and can induce separation. For the case of control surface deflection the resulting pressure distribution on the compression ramp is of interest because it will dictate the performance of the control surface itself. Additionally, the heat transfer in this interaction region is also of interest because localized peaks can occur due to the laminar shock/boundary layer interaction, and these effects must be accounted for in the control surface design.

6.2.2. Computational mesh. A single structured grid was used to discretize the domain and was shown previously in Figure 2(a). The outer boundary of the grid was created with a straight segment from the leading edge of the plate to the outflow boundary. The height of the outflow boundary was chosen such that the weak shock and subsequent Mach wave produced by the upstream portion would be fully contained within the flow domain. The left and upper boundaries are specified as freestream with essential boundary conditions. The plate itself is modeled as an isothermal no-slip wall. While not visible in the image, there is a very small region upstream of the plate leading edge, which is modeled with a symmetry boundary condition. Thus, there is a slip/stick boundary condition on the velocity at the leading edge of the plate. The baseline nonadapted mesh used in the simulation contains 46 680 quadrilateral elements with 47 190 nodes, yielding a discrete problem with 188 760 degrees of freedom. The mesh was partitioned into six subdomains and the simulation was run in parallel on a group of desktop-class machines. The partitioned mesh is shown in Figure 2(b). Note that due to the fine streamwise and normal mesh resolution used at the leading edge of the plate, one of the subdomains is so physically small as to be barely visible in the figure, although each subdomain contains roughly the same number of elements.

6.2.3. Experimental validation. Figure 9 depicts the global flowfield for this case. The adverse pressure gradient induced by the compression ramp is evident in Figure 9(b). This pressure gradient causes the boundary layer to separate upstream of the compression ramp.

Figure 10(a) and (b) compares the computed skin friction coefficient and Stanton number distribution with measurements made by Holden [43]. The experimental data were obtained in the Calspan 48-in shock tunnel. The test article was instrumented with thin-film heat transfer gages, pressure transducers, and skin friction transducers along the centerline. A range of plate widths were tested to ensure that the centerline data were not adversely affected by three-dimensional expansion effects [43].

The surface shear is an excellent indicator of the onset of separation that occurs upstream of the compression ramp corner. At the separation point the surface shear vanishes. The attached upstream flow produces a positive shear while the flow in the recirculation region produces a negative shear. Figure 10(a) plots the nondimensional skin friction coefficient *versus* the nondimensional distance from the leading edge of the plate. The skin friction coefficient is defined as

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho_\infty U_\infty^2} \tag{56}$$

where τ_w is the shear stress that is nondimensionalized with the freestream dynamic pressure. The experimental and computed values are in general agreement, and the magnitude of the shear is

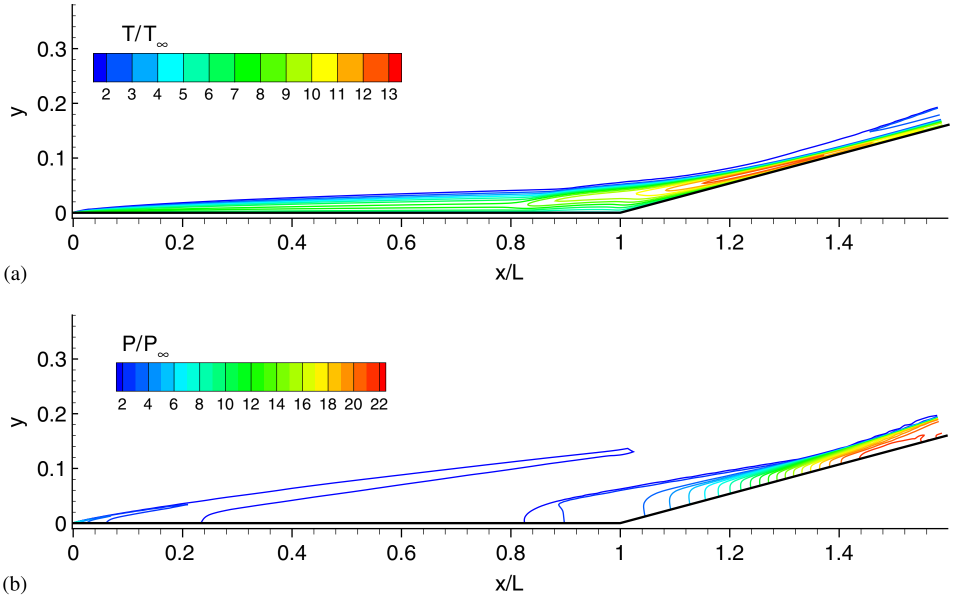

Figure 9. Illustration of flowfield for hypersonic shock ramp problem: (a) temperature contours and (b) pressure contours.

DOI: 10.1002/fld

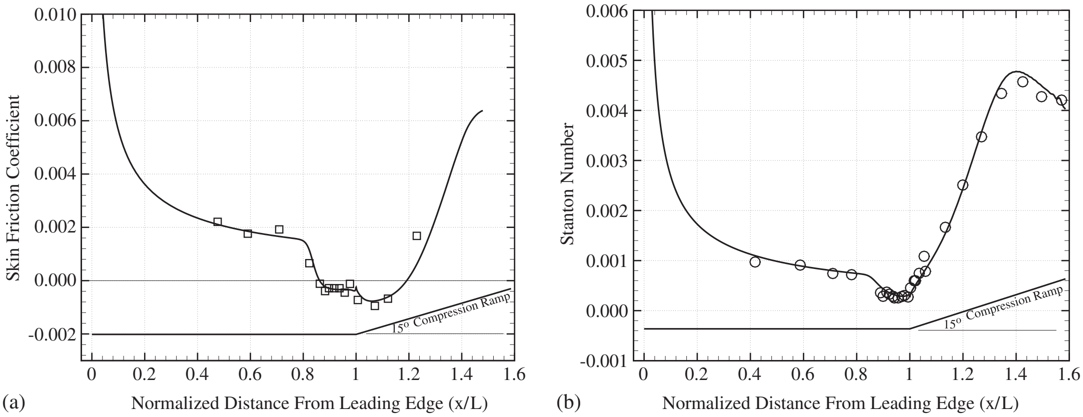

Figure 10. Surface data comparison with experimental data: (a) skin friction coefficient and (b) Stanton number.

in excellent agreement in the recirculation region (and hence the strength of the recirculation). Similar results were reported by Lillard and Dries with a completely different flow solver [46].

The surface heat transfer is critically important because of the severe heating that can occur in the reattachment region on the compression ramp. In this region the edge of the boundary layer is subject to a compression fan which markedly thins the boundary layer. The resulting surface heat transfer obtains a local maximum. As previously discussed, the compression ramp serves as a conceptual model for a control surface deflected into a hypersonic stream. In this application it is critically important to understand the magnitude of the reattachment heating because it provides the design environment for the thermal protection system on the control surface.

In Figure 10(b) the computed and measured heat transfers are compared. The wall heat transfer, q_{wall}, is nondimensionalized by means of the Stanton number

$$St = \frac{q_{\text{wall}}}{\rho_\infty U_\infty c_p (T_0 - T_w)} \tag{57}$$

where T_0 is the freestream total temperature, T_w is the surface temperature of the model, ρ_∞ and U_∞ are the freestream density and velocity, and c_p is the freestream specific heat at constant pressure.

7. CONCLUSIONS

A modified finite element formulation is developed to simulate high-Reynolds number flows. The scheme is an extension of the SUPG family augmented by a modified shock-capturing operator which is required to eliminate spurious oscillations in the vicinity of shock waves. The main features of this study concern improvements in numerical methodology for compressible Navier–Stokes simulation supported by accompanying verification simulations and an experimental validation study.

The verification test results for Mach 3 flow over a cylinder serve as a good test case for the effectiveness of the modified shock-capturing operator (e.g. computed and theoretical jump values

DOI: 10.1002/fld

are in excellent agreement). The performance of the associated transient, nonlinear, and mesh convergence was investigated. The method was then validated by comparison with experimentally measured quantities of interest such as surface pressure and heat transfer distributions.

The method is applicable to arbitrary unstructured discretizations, but the results shown here employ high-quality, structured grids. The performance of the algorithm on unstructured meshes, including the influence of mesh quality on solution accuracy, is of interest and will be considered in future work. This is a particularly important question as the ability to use hybrid-element unstructured meshes can greatly simplify the mesh generation process. Additional work will also examine how the specific choice of inviscid flux discretization (Equation (38)) enhances the numerical stability of the method.

While only laminar, calorically perfect gases are considered in this work, the approach is expected to generalize directly to the case of turbulent and/or reacting flows. Future work will extend the range of applicability of the finite element model by including state equations for gases in thermal equilibrium. The effects of turbulence may be included through the typical Reynolds-Averaged Navier–Stokes approach by implementing suitable turbulence models. Additionally, the highly localized shock waves and boundary layers that occur in this class of flows are well suited for simulation with adaptive mesh refinement techniques, and such simulations will be the focus of future research.

REFERENCES

1. Kirk BS, Peterson JW, Stogner RH, Carey GF. `libMesh`: a C++ library for parallel adaptive mesh refinement/coarsening simulations. *Engineering with Computers* 2006; **22**(3):237–254.
2. Kirk BS. Adaptive finite element simulation of flow and transport applications on parallel computers. *Ph.D. Thesis*, The University of Texas at Austin, May 2007.
3. Vincenti WG, Kruger CH. *Introduction to Physical Gas Dynamics*. Krieger: New York, 1965.
4. White FM. *Viscous Fluid Flow* (2nd edn). McGraw-Hill: New York, 1991.
5. Tannehill JC, Anderson DA, Pletcher RH. *Computational Fluid Mechanics and Heat Transfer* (2nd edn). Taylor & Francis: Washington, DC, 1997.
6. Christie I, Griffiths DF, Mitchell AR, Zienkiewicz OC. Finite element methods for second order differential equations with significant first derivatives. *International Journal for Numerical Methods in Engineering* 1976; **10**:1389–1396.
7. Carey GF, Oden JT. *Finite Elements*: *VI*, *Special Problems in Fluid Mechanics*. Prentice-Hall: Englewood Cliffs, 1986.
8. Fries T-P, Matthies HG. A review of Petrov–Galerkin stabilization approaches and an extension to meshfree methods. *Technical Report 2004-01*, Institute of Scientific Computing, Technical University Braunschweig, March 2004.
9. Kessler MP, Awruch AM. Analysis of hypersonic flows using finite elements with Taylor–Galerkin scheme. *International Journal for Numerical Methods in Fluids* 2004; **44**:1355–1376.
10. Jiang BN, Carey GF. A stable least-squares finite element method for nonlinear hyperbolic problems. *International Journal for Numerical Methods in Fluids* 1988; **8**:933–942.
11. Jiang BN, Carey GF. Least-squares finite element methods for compressible Euler equations. *International Journal for Numerical Methods in Fluids* 1990; **10**:557–568.
12. Hughes TJR, Franca LP, Hullbert GM. A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method advective–diffusive equations. *Computer Methods in Applied Mechanics and Engineering* 1989; **73**:173–189.
13. Hughes TJR, Mallet M. A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective–diffusive systems. *Computer Methods in Applied Mechanics and Engineering* 1986; **58**:305–328.
14. Shakib F, Hughes TJR, Johan Z. A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1991; **89**:141–219.

DOI: 10.1002/fld

15. Aliabadi SK. Parallel finite element computations in aerospace applications. *Ph.D. Thesis*, The University of Minnesota, 1994.
16. Aliabadi SK, Tezduyar TE. Parallel fluid dynamics computations in aerospace applications. *International Journal for Numerical Methods in Fluids* 1995; **21**:783–805.
17. LeBeau GJ. The finite element computation of compressible flows. *Master's Thesis*, The University of Minnesota, 1990.
18. Hughes TJR, Mallet M. A new finite element formulation for computational fluid dynamics: IV. A discontinuity operator for multidimensional advective–diffusive systems. *Computer Methods in Applied Mechanics and Engineering* 1986; **58**:329–336.
19. Hauke G, Hughes TJR. A comparative study of different sets of variables for solving compressible and incompressible flows. *Computer Methods in Applied Mechanics and Engineering* 1998; **153**:1–44.
20. Bonhaus DL. A higher order accurate finite element method for viscous compressible flows. *Ph.D. Thesis*, Virginia Polytechnic Institute and State University, Blacksburg, VA, November 1998.
21. Fletcher CAJ. The group finite element formulation. *Computer Methods in Applied Mechanics and Engineering* 1983; **37**:225–243.
22. Morgan K, Peraire J. Unstructured grid finite element methods for fluid mechanics. *Inst. Phys. Rev.*, 1998; **61**(6):569–638.
23. Kuzmin D, Möller M, Turek S. High-resolution FEM-FCT schemes for multidimensional conservation laws. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**:4915–4946.
24. Kuzmin D, Turek S. High-resolution FEM-TVD schemes based on a fully multidimensional flux limiter. *Journal of Computational Physics* 2004; **198**:131–158.
25. Catabriga L, Coutinho ALGA. Improving convergence to steady state of implicit SUPG solution of Euler equations. *Communications in Numerical Methods in Engineering* 2002; **18**(5):345–353.
26. Gropp WD, Kaushik DK, Keyes DE, Smith BF. High performance parallel implicit CFD. *Journal of Parallel Computing* 2001; **27**:337–362.
27. Hovland PD, McInnes LC. Parallel simulation of compressible flow using automatic differentiation and PETSc. *Parallel Computing*, 2000 (Special issue of Parallel Computing on 'Parallel Computing in Aerospace').
28. Kaushik DK, Keyes DE, Smith BF. Newton–Krylov–Schwarz methods for aerodynamic problems: compressible and incompressible flows on unstructured grids. In *Proceedings of the 11th International Conference on Domain Decomposition Methods*, Lai C-H *et al.* (eds). Domain Decomposition Press: Bergen, 1999.
29. Karypis G, Kumar V. METIS unstructured graph partitioning and sparse matrix order. *Technical Report*, Department of Computer Science, University of Minnesota, August 1995.
30. Balay S, Buschelman K, Eijkhout V, Gropp WD, Kaushik D, Knepley MG, McInnes LC, Smith BF, Zhang H. *PETSc Users Manual. Technical Report ANL-95/11—Revision 2.3.0*, Argonne National Laboratory, April 2004.
31. Iserles A. *A First Course in the Numerical Analysis of Differential Equations*. Cambridge University Press: Cambridge, 1996.
32. Greenberg MD. *Foundations of Applied Mathematics*. Prentice-Hall: Englewood Cliffs, NJ, 1978.
33. Johan Z, Hughes TJR, Shakib F. A globally convergent matrix-free algorithm for implicit time-marching schemes arising in finite element analysis in fluids. *Computer Methods in Applied Mechanics and Engineering* 1991; **87**:281–304.
34. Barrett R, Berry M, Chan TF, Demmel J, Donato JM, Dongarra J, Eijkhout V, Pozo R, Romine C, der Vorst HV. *Templates for the Solution of Linear Systems*: *Building Blocks for Iterative Methods*. Society for Industrial and Applied Mathematics: Philadelphia, 1994 (Also available as postscript file on http://www.netlib.orgtemplatesTemplates.html).
35. Golub GH, Van Loan CF. *Matrix Computations* (3rd edn). Johns Hopkins University Press: Baltimore, MD, 1996.
36. Saad Y, Schultz MH. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1986; **7**(3):856–869.
37. Jiang G, Shu C-W. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics* 1996; **126**:202–228.
38. Shu C-W. High order finite difference and finite volume WENO schemes and discontinuous Galerkin methods for CFD. *Technical Report ICASE Report No. 2001-11*, Brown University, Providence, RI, May 2001.
39. Ambrosio A, Wortman A. Stagnation point shock detachment distance for flow around spheres and cylinders. *American Rocket Society* 1962; **32**(2):285–287.
40. Capon PJ. Adaptive stable finite element methods for the compressible Navier–Stokes equations. *Ph.D. Thesis*, The University of Leeds, 1995.

41. Ames Research Staff. Equations, tables, and charts for compressible flow. *Technical Report NACA 1135*, National Advisory Committee for Aeronautics, 1953.
42. Barth WL. Simulation of non-Newtonian fluids on workstation clusters. *Ph.D. Thesis*, The University of Texas at Austin, 2004.
43. Holden MS. A study of flow separation in regions of shock wave-boundary layer interaction in hypersonic flow. *Eleventh Fluid and Plasma Dynamics Conference*, Seattle, Washington, AIAA Paper 1978-1169, July 1978.
44. Holden MS. Historical review of experimental studies and prediction methods to describe laminar and turbulent shock wave/boundary layer interactions in hypersonic flows. *Forty-fourth AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, AIAA Paper 2006-494, January 2006.
45. Abgrall R, Désidéri J-A, Glowinski R, Mallet M, Périaux J. *Hypersonic Flows for Reentry Problems*, vol. 3. Springer: Berlin, 1991.
46. Lillard RP, Dries KM. Laminar heating validation of the OVERFLOW code. *Forty-third AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, AIAA Paper 2005-685, January 2005.

DOI: 10.1002/fld