42nd AIAA Joint Propulsion Conference and Exhibit AIAA 2006-4780

Time-Accurate, Unstructured-Mesh Navier-Stokes Computations with the Space-Time CESE Method

Chau-Lyan Chang*
NASA Langley Research Center, Hampton, VA23681

Application of the newly emerged space-time conservation element solution element (CESE) method to compressible Navier-Stokes equations is studied. In contrast to Euler equations solvers, several issues such as boundary conditions, numerical dissipation, and grid stiffness warrant systematic investigations and validations. Non-reflecting boundary conditions applied at the truncated boundary are also investigated from the stand point of acoustic wave propagation. Validations of the numerical solutions are performed by comparing with exact solutions for steady–state as well as time-accurate viscous flow problems. The test cases cover a broad speed regime for problems ranging from acoustic wave propagation to 3D hypersonic configurations. Model problems pertinent to hypersonic configurations demonstrate the effectiveness of the CESE method in treating flows with shocks, unsteady waves, and separations. Good agreement with exact solutions suggests that the space-time CESE method provides a viable alternative for time-accurate Navier-Stokes calculations of a broad range of problems.

I. Introduction

Computational methods based on unstructured meshes offer many advantages over their more traditional counterpart based on structured meshes. In many aerodynamics and acoustic applications, unstructured mesh methods are increasingly being used to better handle complex configurations. Despite complaints about difficulties in generating viscous mesh and implementing multi-grid acceleration, unstructured methods are still the choice of many new developments. Numerical methods devised for unstructured solvers for fluid dynamics applications can be roughly divided into two major categories: finite element and fine volume approaches. Notwithstanding its great success for solid mechanics applications, the finite-element approach only enjoys moderate popularity in fluid dynamics applications, mostly for incompressible or low subsonic flows. The elliptic nature of the governing equations at low speed renders the derivation of finite element equations via the Galerkin or other least-square methods relatively straightforward (e.g. refs. [1-2]). The finite-element based spectral element method uses spectral approximation or spectral collocation approach for dependent variables within each element in order to achieve high accuracy (e.g. [3]). Without special treatment for possible flow discontinuities, most finite element methods suffer from applicability to general compressible viscous flows due to the mixed hyperbolic/elliptic characters of the governing equations. To address these important issues, the discontinuous Galerkin (DG) method introduces Riemann solvers at the element interfaces to uniquely define the discontinuous flux. Combined with the use of local solution polynomials as the test function of the weak form integrals, the DG method offers a local high-order numerical framework that can be used for general compressible flows (see refs.[4-6]). However, application of the DG method for Navier-Stokes equations is still in its infancy and several competing approaches to treat the viscous terms are still under fundamental investigations [7-8]. More recently, collocation based spectral difference methods are also available [9-10]. In a large part, these local high-order methods are attractive due to the existing rigorous error analysis inherited from the finite-element approach and their potential for *p*-multigrid refinement [11]. Many believe that a grid refinement technique without increasing the stencil or mesh relocation holds the key for future high-fidelity computations such as large eddy simulations (LES) or direct numerical simulations (DNS) of complex geometries.

Finite-volume based unstructured methods have also been used extensively to solve compressible Navier-Stokes equations (e.g., refs.[12-13]). In contrast to the finite-element based methods, the finite volume approach is

* Research Scientist, Computational Aeroscience Branch, email: Chau-Lyan.Chang@nasa.gov,
Senior member, AIAA

constructed on the original integral form of the flux conservation equations. Numerical integration of the conservative flux equations is carried out around discretized control volumes. These control volumes are small regions surrounding the mesh points. To account for possible flow discontinuities across the control volumes, the flux vector at the control volume interface requires special attention. An upwind scheme with approximated Riemann solvers and weight averages must be used to reconstruct the interface flux vectors in order to maintain higher than first order accuracy. Point or block Jacobi implicit treatment is often used to allow larger time-step and thus improve convergence rate to steady state. Sub-iterations are required to achieve time accuracy. NASA's FUN3D code has been developed based on the unstructured finite-volume framework [14]. The FUN3D code is the product of a continuous team developmental work. In addition to the unstructured Navier-Stokes flow solver, it offers a suite of tools including adjoint–based design methods, adjoint-based error estimation, adaptive grid refinement capabilities, and detailed hypersonic reacting flow chemistry. The finite-volume based unstructured method has been successfully applied to solve a large class of flow problems. It is in general easy to implement but also suffers from accuracy constraints. Higher accuracy of the finite-volume method can only be achieved by increasing the grid stencil which is non-trivial in an unstructured mesh. Moreover, the upwind difference and approximate Riemann solver used for interface flux integration is known to cause problems for a detached bow shock. In some hypersonic applications, it introduces anomalies in the solutions near the stagnation point region [15].

The space-time CESE method was introduced by S.-C. Chang in the early 1990's[16]. This new method is constructed based on distinctly different philosophies in several regards. Unlike most traditional schemes, the space and time derivatives of the conservation laws are treated in exactly the same manor. Integral forms of the governing equations are numerically integrated over discretized conservation elements (CE) constructed on the space-time domain. Solutions are assumed to vary piecewise linearly within a prescribed (usually the elements directly from grid generation) element called solution element (SE). To avoid multiply defined solutions at the interfaces of the integration volume, CE and SE are staggered. The faces of a CE only cut across a uniquely defined region that belongs to a unique SE. In this way, flux reconstruction at the integration volume interfaces does not enter the picture. Consequently, the Riemann solver used to uniquely define the flux at an interface in finite-volume as well as other finite-element based methods is never needed. Discontinuities in the solution domain are handled by applying weighted average of the derivatives of the dependent variables without resorting to procedures to redefine flux vectors, which in turn implicitly alters the conservation properties of the governing equations.

Unlike the finite-element based methods, the CESE approach uses the original integral form of the governing conservation laws. It can be proved mathematically that the summation of numerical integrals over all discretized CE in the domain of interest also conserves the flux [16]. This nice conservation property over space and time enables the CESE method to handle time accuracy, shock capturing, and boundary conditions in a much simpler manor than conventional methods. From the outset, the CESE method is built upon a numerical dissipation free a-scheme. When applied to the scalar wave equation, it can be proved that the CESE framework is space-time invariant, i.e., time integration can be carried out reversibly [17]. Marching backward in time recovers the answers from forward marching. The construction of the dissipation free a-scheme also provides a baseline solution for augmenting numerical dissipation pertinent to maintaining stability for nonlinear governing equations. The other feature of the method is that derivatives are part of the unknowns in addition to the dependent variables. Chang proved that treating derivatives as unknowns is a necessary condition to preserve space-time invariance [17]. Numerical framework for the CESE method was initially constructed for triangular elements [18] in 2D and tetrahedrons in 3D [19]. Later, it was extended for quadrilateral and hexahedral elements [20].

The CESE method has been successfully applied to a broad spectrum of flow problems including computational aero-acoustics (CAA), general flow solutions from very low to hypersonic speeds, and electromagnetic wave propagations [21-23]. Preliminary investigations on applying the method for Navier-Stokes equations are also available [21, 24]. Many of these results indicate that the CESE method offers high solution accuracy for flows containing both shocks and acoustic or vortical waves. In this paper, we apply the CESE method for a class of viscous problems using the compressible Navier-Stokes equations. The objective is to address the issues related to Navier-Stokes calculations for this new method and in the mean time, assess the numerical framework as a viable unstructured mesh method for time-accurate viscous simulations of a variety of flow problems. In the following sections, a brief discussion about the numerical formulation is given followed by discussions of validations and applications of the CESE method.
.

II. Numerical Method

The Space-time CESE method

Numerical formulation of the space-time CESE method has been discussed in full details in the literature [16-20]. In addition to highlighting the distinct features of the scheme, we focus on two interrelated issues in this section: viscous flow calculations using Navier-Stokes equations and numerical dissipation control. The three-dimensional compressible Navier-Stokes equations in conservative form can be written as:

$$\frac{\partial Q}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}+\frac{\partial H}{\partial z}=I+\frac{\partial F_v}{\partial x}+\frac{\partial G_v}{\partial y}+\frac{\partial H_v}{\partial z} \qquad (1)$$

where the dependent variable vector is defined as $Q = (\rho, \rho u, \rho v, \rho w, e)$ and $x, y, z,$ and t represent spatial coordinates and time, respectively. Flow variables $\rho, u, v, w,$ and e represent density, three velocity components, and total energy, respectively. Definitions of the inviscid flux vectors F, G, H and viscous flux vectors F_v, G_v, H_v can be found in standard text books (e.g. Hirsch [25]) and will not be included here. The source vector I contains all external forcing or other energy-related source terms. For axisymmetric flows, $H = H_v = 0$ and I contains extra terms associated with the cylindrical coordinate system. Euler equations are recovered by neglecting all viscous and source terms on the right hand side of the equation. To close the system, the perfect gas relation, $p=\rho RT$ is used in conjunction with eq. (1).

In contrast to conventional CFD algorithms, the space time CESE method is formulated by treating both temporal and spatial directions with synergy. To this end, eq. (1) is re-written in the following integral form

$$\oint_{S(V)} \vec{h}\cdot d\vec{s}+\int_{V} I dV=0 \qquad (2)$$

where the space-time flux vector is integrated over the surface S of an arbitrary space-time volume V. The flux vector $\vec{h}$ is defined as

$$\vec{h}=(F-F_v, G-G_v, H-H_v, Q) \qquad (3)$$

The surface normal vector is defined by $\vec{s}=\vec{n}\,d\sigma$, where $d\sigma$ is the area increment on S and $\vec{n}$ is the outward unit normal vector. For an arbitrary unstructured mesh with triangular or quadrilateral elements for 2D and tetrahedral or hexahedral elements for 3D, the dependent variable vector is assumed to vary linearly within each element:

$$Q(x,y,z,t)=Q_0+Q_t(t-t_0)+Q_x(x-x_0)+Q_y(y-y_0)+Q_z(z-z_0) \qquad (4)$$

where Q_0 is the solution vector at the solution point (x_0, y_0, z_0, t_0) defined as the geometry center of the polygons formed by all surrounding conservation elements to be described later. Coefficients of the polynomial, Q_t, Q_x, Q_y, and Q_z.are part of the solutions and need to be determined within each element. Chang has shown that to preserve the space time invariant property, derivatives of dependent variables must be treated as unknowns [17]. High-order extension of the CESE method is currently under investigation by Chang [26]. Numerical formulations for types of elements other than those mentioned above are also possible. Here, we only consider the first-order Taylor series expansion for the above four types of elements. This results in a formally second-order accurate scheme. In the CESE method, a discretized space-time volume within which eq. (4) is valid is referred to as a solution element (SE). Solution elements are naturally constructed on each individual element in the mesh.

In traditional finite-volume unstructured mesh methods, flux integration for the discretized volumes is either carried out on a control volume that coincides with the solution element as in the cell-centered schemes or on a control volume that surrounds a given vertex point as in the cell-vertex schemes. In either case, the flux vector must be "reconstructed" on the faces of the control volume because at these interfaces, the solution vector Q cannot be

uniquely determined. A simple averaging or more complex Riemann solver is usually employed at the interface to evaluate the flux vectors needed in the numerical integration. Flux conservation over the entire domain is guaranteed by the unique flux on each interface. However, the degree of conservation within each control volume is still dictated by the flux construction procedure adopted. Any averaging procedure with an approximated Riemann solver based on one-dimensional formulation is likely to introduce errors in flux conservation locally. We note here that many of the finite-element based methods also employ the concept of flux reconstruction at the element interface as in the finite-volume methods.

To circumvent the above shortcomings of conventional unstructured mesh schemes, the space-time integration of the CESE method is carried out on discretized volumes called conservation elements (CE). These conservation elements are constructed independently from the solution elements. Chang [16] demonstrates that by using a space-time staggered mesh, each face of the conservation element is located within the solution element and no flux reconstruction is needed for numerical integration. Flux balance is reinforced at each conservation element without introducing any ad-hoc averaging procedure. The overall flux conservation is naturally satisfied by the integration of all conservation elements involved in the domain. The use of the original integral form of equations in the CESE method prohibits a rigorous error analysis based on the minimization procedure as in the finite-element based schemes. Nonetheless, it provides mechanisms for ensuring crucial flux conservation for hyperbolic conservation laws, especially, when flow discontinuity is present.

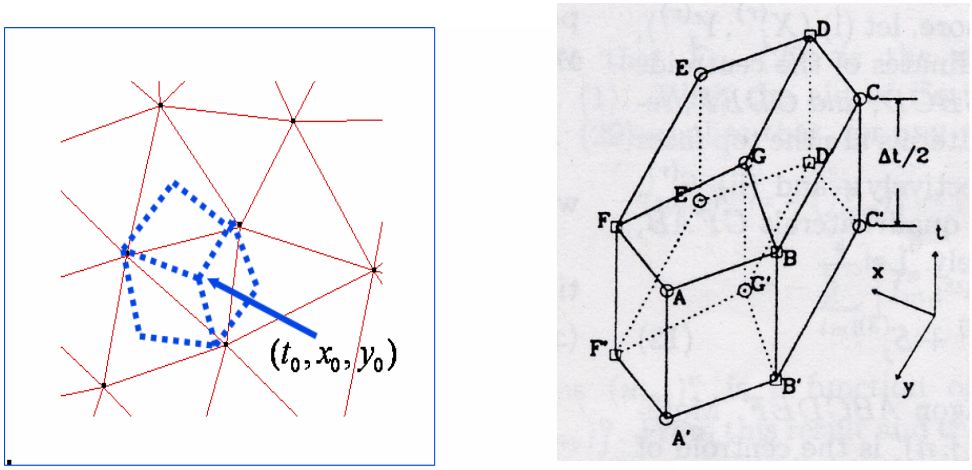
We briefly outline the CESE method for a 2D triangular mesh depicted in Figure 1. For a triangular solution element centered at point P0 with coordinates (x_0, y_0, t_0), three quadrilateral prism CE are constructed around it. The top faces of these prisms are at the new time level to be evaluated. Applying eq. (2) for all three CE requires solutions at the old time level on the bottom and side prism faces as well as solutions at the new time level on the top and inner CE faces. Note that all CE boundaries lie in the vicinity of an SE and thus no interpolation or extrapolation is needed for flux calculations. By selecting the geometric center of all three surrounding CE as the solution point of the SE under consideration, the summation of the flux integration equations for all three CE can be represented by the unknown solution vector Q_0 at the new time level and known solutions at the old time level. This equation can be easily solved to obtain Q_0 at the new time level. We still need to solve for the derivatives in eq. (4). To this end, recall that applying eq.(2) for each CE provides three conservation equations. Disregard the summation equation already used to obtain the solution vector Q_0, we still have two equations left for two more unknowns, Q_x and Q_y, respectively. These unknowns form a block matrix equation for each element of interest and can be solved locally. The remaining time derivative can be solved by assuming

$$Q_t + AQ_x + BQ_y = I \qquad (5)$$

where A and B are Jacobian matrices of the corresponding flux vectors. The above procedure forms the basis of the a-scheme. A-scheme is numerical dissipation free and only exists if the number of surrounding CE matches the unknowns of derivatives (i.e., only for 2D triangular and 3D tetrahedral elements). For Euler equations, the a-scheme is only neutrally stable. To maintain numerical stability, artificial dissipation must be added into the discretized equations. Nevertheless, the construction of a dissipation free scheme serves as a base line for better control of numerical dissipations.

One alternative way to calculate derivatives is to apply finite difference differentiation using the neighboring solutions. For the triangular element shown in Fig. 1, let Q^1, Q^2, and Q^3 represent three solution vectors evaluated at three neighboring solution points P1, P2, and P3, respectively. Three finite-differenced derivatives may be obtained by solving polynomials formed by three groups of points: (Q_0 , Q^1, Q^2), (Q_0, Q^2, Q^3), and (Q_0, Q^3, Q^1), respectively. Derivatives of each dependent variable at each SE are thus determined by weighted average of all the above finite-difference derivatives. These alternative derivatives represent solutions that are different from the a-scheme. The departure from a-scheme implies that numerical dissipation is being added to the discretized system. The amount of dissipation added depends on how weighted averaged derivatives are evaluated. The weighted averaging also serves as a means to handle flow discontinuities when solution vectors exhibit a large gradient across the element interfaces. This weighted averaging procedure in conjunction with the algebraic equation for the solution vector at the new time level is referred to as the c-scheme [16]. The amount of numerical dissipation is related to a parameter α that refers to the exponent used in the weighted averaging procedure. A larger value of α implies more attenuation near the sharp discontinuity. C-scheme can be constructed for all four types of elements mentioned above. In fact,

for a quad or hex mesh, no a-scheme is available because the number of equations is more than unknowns. Compared to a-scheme, c-scheme is much more efficient because no additional matrices need to be solved. In general, c-scheme can be constructed for any arbitrary mesh provided a numerically stable weighted averaging procedure can be devised.



(a) (b)

Figure 1. Conservation elements for a 2D triangular mesh: (a) top view (b) space time CE volume

Despite the effectiveness of the c-scheme, it can be shown that numerical dissipation increases as the local CFL number decreases [16]. While this may not be a major shortcoming for a relatively uniform mesh, it does pose problems when a time accurate solution for a highly non-uniform mesh is of interest. A numerically stable (the CFL number less than 1) time step at the smallest element could mean very small CFL number at coarse mesh. As a result, numerical solutions could become very diffusive at some part of the flowfield. To remedy this issue, Chang introduces a CFL number insensitive (CNI) scheme [27]. In a CNI scheme, the neighboring solutions used for finite difference evaluations of derivatives are not necessarily the neighboring solution points P1, P2, or P3. Instead, the locations of the neighboring points are dynamically adjusted between those neighboring solution points and each CE center. For example, let M_i denote the geometric center of the *i-th* CE and P_i the associated solution point at the neighboring SE. The point N_i for derivative evaluation is controlled by a parameter σ,

$$N_i = M_i + \sigma(P_i - M_i) \qquad (6)$$

In one-dimensional cases, the point M_i is the mid-point between the two neighboring solution elements. For multi-dimensional cases, numerical experiments indicate that using the mid point between two neighboring solution points or the CE center does not make significant differences in the solution although the CE center should be used in theory. Chang [27] shows mathematically that as the time step approaches zero (CFL goes to zero), derivatives evaluated at the mid point give numerical solutions that asymptotically approach the a-scheme. Therefore, an effective way to reduce numerical dissipation is to use a smaller value for σ in eq. (6). For 1D problems, it can be shown that by reducing the value of σ, numerical dissipation for very small CFL number is comparable to that for a large CFL number close to 1. Ideally, the parameter σ should be automatically adjusted based on the local CFL number. Note that the c-scheme is nothing but a special case with a σ value of 1. Although eq. (6) is used in the CNI scheme to reduce dissipation, it can also be used to *increase* numerical dissipation by assigning a σ value greater than 1. Flexible use of this parameter enables better control over numerical dissipation in Euler as well as Navier-Stokes calculations. For high-fidelity computations such as those in CAA or direct numerical simulations, better control of numerical dissipation can greatly increase the solution accuracy.

Boundary conditions are implemented via an image element at the boundary face. For inviscid walls, reflected boundary conditions are imposed by constructing a mirror image of the boundary element velocity vector at the image element solution point. Tangential conditions at the inviscid wall are met because the summation of two

mirrored velocity vectors must be parallel to the wall boundary. Temperature and density values of the image element are set to be identical to the corresponding boundary element. Inflow and outflow conditions are imposed at the image cells. In general, no characteristic splitting is employed at the boundary. Implementation of a non-reflected boundary condition is relatively easy for the CESE method. In conventional CFD methods, characteristic variables (Riemann invariants) in the direction normal to the boundary are used to manually limit the traveling directions of characteristic information. For multi-dimensional problems, this one-dimensional characteristic splitting is only an approximation and introduces errors that may cause boundary reflections. A buffered domain or sponge layer near the boundary is usually introduced to further alleviate boundary reflections. In the CESE method, a non-reflected boundary condition is implemented simply by imposing the boundary element values at the image element, i.e.

$$Q^i = Q^b, \qquad Q_x^i = Q_x^b, \qquad Q_y^i = Q_y^b, \qquad Q_z^i = Q_z^b \tag{7}$$

where superscripts *i* and *b* represent image and boundary elements, respectively. Eq. (7) works fine for many steady-state and time-accurate applications in which no visible boundary reflection is present in the solution. The main reason that such simple extrapolation non-reflecting boundary conditions would work for the CESE method is attributed to the enforcement of flux conservation in both time and space. Global flux formulation along all directions allows information to propagate in time and exit the domain without incurring significant boundary reflections even with a simple steady-state boundary condition. In fact, it can be shown that propagation of erroneous information associated with inappropriate boundary conditions is confined to the close vicinity of the source. For very small-amplitude perturbations, noticeable reflection at the boundary may still be observed. However, it can be eliminated by a relatively small buffered domain thanks to the "local" nature of boundary reflections. In the following section, we will discuss in more details about the effectiveness of the non-reflecting boundary condition.

Given one more degree of freedom in time, there are many possible ways to integrate eq. (2) over the domain of interest. For the present CESE scheme, the conservation elements are constructed by an element interface parallel to the temporal axis. In theory, CE can take any arbitrary shape in the space-time domain. However, this would add one more dimension (in time) to the mesh generation process. In contrast to the current schemes, an arbitrary space-time mesh alters the equation for the solution vector at the new time level. Coupled equations for the unknowns of the neighboring solution vectors would form a large banded matrix instead of a separate algebraic equation for each element in the present schemes. Since direct solvers are not practical for 3D problems, a localized strategy must be employed to improve numerical efficiency in solving this coupled system of equations. The other problem is that an arbitrary space-time mesh for a 3D mesh is topologically non-trivial for the mesh generation process and may require some sort of projection methods.

Navier-Stokes Formulation

Numerical formulations of the CESE method for the Euler equations are directly applicable to the Navier-Stokes equations provided that several important issues are properly addressed. Firstly, viscous terms must be added to the flux vectors. These viscous flux terms involve first derivatives of dependent variables, which are assumed to be constants within the SE. To account for the variations of viscous derivatives within each SE, higher order derivatives must be introduced and evaluated. As a first approximation and also to avoid introducing more complex numerical treatments, we use the constant derivatives at the CE interface for the evaluation of viscous terms. More accurate discretization for viscous derivative will be one of the subjects of a high-order CESE method [26].

No slip conditions are applied at the wall for Navier-Stokes calculations. Instead of using the unified inviscid/viscous flux treatment suggested by Chang [28], we apply the no-slip boundary condition directly. The viscous boundary conditions are illustrated in Figure 2 for a triangular element. Extension to other type of elements is straightforward. Let O denote the solution point of a wall boundary SE and O' stands for its image point with respect to the wall boundary. The distance from O' to the wall is denoted by *d*. The values for both dependent variables and their derivatives at each time (iterative) step must be prescribed for the image element. There are many ways to determine these values at O' because the image element is nothing but a numerical artifact. One

important concept is that the velocity derivative at the wall is part of the solution and cannot be arbitrarily assigned. Note that when $d \to 0$, point O' coincides with the wall and the image element degenerates into a line segment on the viscous wall. The velocity vector at O' thus takes the no-slip wall value, $\vec{V}_w$ and the velocity derivatives must also be equal to those at the neighboring boundary element, i.e.,

$$\vec{V}_{O'} = \vec{V}_w, \quad \left(\frac{\partial \vec{V}}{\partial x}\right)_{O'} = \left(\frac{\partial \vec{V}}{\partial x}\right)_O, \quad \left(\frac{\partial \vec{V}}{\partial y}\right)_{O'} = \left(\frac{\partial \vec{V}}{\partial y}\right)_O \tag{8}$$

Thermal boundary conditions are used to determine the temperature. For constant wall temperature, $T_{O'} = T_w$; and for adiabatic wall, $(\partial T / \partial n)_{O'} = 0$. The latter condition can be approximated by setting $T_{O'} = T_O$. The simplified normal momentum equation at the no-slip wall is normally used as the last condition to close the problem. We use the zero normal pressure gradient condition, $(\partial p / \partial n)_{O'} = 0$ instead.

One alternative way to determine the derivatives lies on the fact that any points on the degenerated image element must satisfied the no-slip conditions. Let *A* and *B* denote two vertices of the wall face. It follows that,

$$\vec{V}_A = \vec{V}_w, \ \vec{V}_B = \vec{V}_w \tag{9}$$

These two conditions in conjunction with the first order Taylor series expansion of the velocity vector within the wall boundary element can be used to determine two unknowns, $\partial \vec{V} / \partial x$ and $\partial \vec{V} / \partial y$. These two computed derivatives are valid for both the wall and image elements. Density and total energy at the image element are determined as described above. The main difference in the above two approaches lie in the fact that the velocity derivatives in the wall element are determined solely based on no-slip conditions and no weighted averaging procedure is applied. Numerical experiments indicate that both of the above implementations of the no-slip conditions give very similar numerical results. It remains to be seen how the differences in these two no-slip conditions affect the turbulent velocity profiles that demand higher accuracy of the wall shear stress.

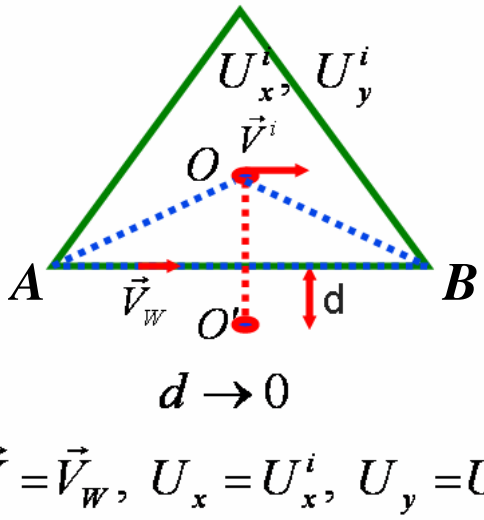


Figure 2. Illustration of the no-slip viscous boundary conditions.

The second issue related to viscous calculations concerns the near wall grid stretching. Without proper grid clustering near the solid wall boundary, effects of viscosity cannot be properly accounted for. High aspect ratio grid near the solid wall may slow down convergence rate and significantly affect the robustness of the N-S solver. For

steady-state problems, a constant CFL number (local time-stepping) is usually used to accelerate convergence. For unsteady calculations with a constant time step for all elements, the local CFL number varies from 1 (maximum value to maintain numerical stability) near the wall to a very small value in the coarse mesh region. As mentioned before, small CFL number implies large numerical dissipation for the original c-scheme. The disparity in CFL number thus may reduce the solution accuracy. Solution accuracy and robustness are thus two main issues regarding Navier-Stokes calculations. For steady-state calculations, local time-stepping allows a large but stable CFL number to be imposed throughout the whole domain to keep numerical dissipation in check. On the other hand, to avoid compromises of solution accuracy at the coarse grid region for time-accurate calculations, the CNI scheme must be used for viscous calculations. A smaller value of σ in eq. (6) should be used for coarse meshes to reduce numerical dissipation. In the viscous region, the local CFL number is large enough and σ can be closer to 1. Chang [27] suggests a simple way to determine σ based on the local CFL number and robust results for 1D test cases were obtained. From our numerical experiments, σ can be reduced to a rather small value (e.g. 10^{-3} or even smaller) if a relatively uniform mesh is involved and a small CFL number is imposed at all elements. However, for a highly non-uniform mesh, σ would vary several orders of magnitudes in the domain and this may lead to divergent results. A minimum allowable σ is usually enforced during time marching to maintain numerical stability. This minimum value is problem dependent and lies somewhere between 0.1 and 0.4. Trial and error is necessary for a new problem with highly non-uniform mesh. Optimal σ value distribution universally applicable for multi-dimensional CESE calculations is in need. The main goal is to come up with a functional relationship between the optimal value of σ and the local cell CFL number.

While determining an optimal σ value in the near wall region is not an issue, the high aspect ratio mesh does pose a robustness problem. Robustness of highly-stretched viscous mesh is probably a common problem for many unstructured mesh flow solvers. This issue is particularly unsparing for the CESE method due to the fact that derivatives are part of the unknowns. The geometry skewness introduced by a very large aspect ratio mesh has a first order effect on the derivative calculations. A near singular triangle or quadrilaterals, for instance, may result in a large derivative value that could cause numerical problems during flux integration. In fact, a quad mesh behaves better than a triangle because of the paring of neighbors. Symmetry (two pairs of neighbors) around a quad mesh results in a near cancellation of large derivatives during the weighted average calculation for the derivatives. Triangular and tetrahedral meshes lack the symmetry pairing of the neighbors and therefore are more likely to have singular derivatives.

One way to mitigate the grid skewness is to use a large σ value in eq. (6) for derivative calculations. Effectively, this is equivalent to pulling away two very close neighboring points. By reducing the proximity, the skewness of the element is significantly reduced such that anomalies in derivative calculations might be avoided. For a moderate to large aspect ratio mesh around 500, a σ value of about 2-3 would stabilize the calculation. For an even higher aspect ratio mesh in the order of 1000, it would need a σ of 10-50 or even 100. The major side effect of using a large σ value is increasing numerical dissipation. As a result, there is a trade-off between stabilizing the time-marching procedure and the solution accuracy. A value of 50 or more for σ should be avoided. For laminar flow calculations, the largest aspect ratio is normally less than 500. In contrast, turbulent calculations do require a much larger aspect ratio meshes. Nonetheless, the presence of large eddy viscosity near the wall would help stabilize the iteration procedure. Possible remedies to this problem are to incorporate the geometry skewness into the weighted average procedure or to increase the order of approximations for large aspect ratio elements.

Code Development

An unstructured mesh compressible Reynolds Averaged Navier-Stokes code (named *ez4d*) based on the space-time CESE method has been developed at NASA Langley Research Center. Turbulence models are still under validation, here we only present laminar solutions. The main goal is to facilitate time-accurate viscous flow calculations for general compressible flows using unstructured meshes with large eddy simulations or direct numerical simulations as longer term objectives. Unlike most research codes, the *ez4d* code has been designed for long-term development. The code is written completely in the C++ programming language using object-oriented and generic programming techniques. The standard template library (STL) of C++ offers a suite of data structure and algorithm software that is especially suited for numerical computations. In particular, several frequently encountered coding elements in a typical unstructured computation code such as index matching, element search, and sorting can be easily programmed with the STL. The “template” language construct in C++ also offers a new dimension in programming numerical computations. For example, it allows easy implementation of built-in or derived type substitution,

compile time polymorphism without using class inheritance, and compile time loop unrolling. The template meta-programming can also be used for code optimization. The *ez4d* code utilizes many of these programming features offered by the C++ language for developmental work to improve the overall modular independency and easier code maintenance and extensions.

The core solver is designed for an unstructured topology using triangular or quadrilateral mesh for 2D flows and tetrahedral or hexahedral mesh for 3D flows. Input interfaces to process a 2D or 3D blocked structured mesh are also included. A 2D/3D multi-block structured mesh can also be handled by the code. Domain decomposition based on the popular Metis library is used to partition an unstructured mesh for parallel processing. Both multi-thread and Messaging Passing Interface (MPI) are implemented in ez4d to facilitate parallel processing of very large meshes. Multi-threading can better take the advantages of the becoming popular dual core CPU architecture.

III. Results and Discussion

Instead of discussing detailed flow physics of applications using the present numerical method, we focus on several important validation cases encountered during the course of algorithmic and code development process. Other papers focus more on the applications [22]. In the following discussion, code validation is performed by comparing with exact solutions or predictions from other existing CFD codes. Both structured and unstructured meshes are used in these cases even though the core solver is an unstructured. Speed regimes range from incompressible to hypersonic flows. Some acoustic wave propagation examples are investigated to assess the non-reflecting boundary conditions. For hypersonic flows, we emphasize on several advantages the present method offers when comparing with existing upwind based schemes. Case studies discussed here are aimed at providing building blocks for future high-fidelity simulations using the space-time CESE method.

Mach 3 Flat-Plate Boundary Layer

A flat plate boundary layer serves as a good validation case for compressible Navier-Stokes codes. A Mach 3 flow over an adiabatic flat plate with an 81x91 quad mesh (91 in the wall-normal direction) is computed. To focus on the viscous effect, the rectangular domain height is only about 2-3 boundary-layer thickness. A mild stretching is applied in the wall-normal direction. The mesh along with the computed streamwise velocity contours are shown in Fig. 3(a). The converged streamwise velocity profile using the c-scheme is compared with solutions from a compressible boundary layer code [36] in Figs. 3(b)-3(d). The overall agreement is quite good at this grid resolution. The small differences in the temperature profile and the departure of wall-normal velocity near the boundary layer edge are related to the small domain used in the Navier-Stokes calculations for which it is more difficult to hold the free-stream conditions. The very small discrepancies in the streamwise velocity profile can be further improved by using a CNI scheme (see below).

The above example validates the viscous terms treatment. To further assess numerical dissipation from the stand point of physical diffusion, we use a slightly coarser quad mesh 81x61 with 61 points in the wall-normal direction for a series of calculations using different numerical dissipation. All calculations were done by local time stepping (constant CFL for every element). The same σ value is applied to the entire flowfield for the results of the CNI scheme and α is 2 for the c-scheme. Figure 4 shows the comparisons with the exact boundary layer solutions for c-scheme and the CNI scheme with three different values of σ (as defined in eq. (6)). The solution by using the c-scheme does not agree with the boundary layer solution as well as that shown in Fig. 3(b) because of the coarser grid near the boundary layer edge and a slightly stronger stretching near the wall. Among all the CESE results, the case with $\sigma = 0.4$ gives the most accurate solution, as expected. Pushing the derivative points toward the mid point effectively reduces the numerical dissipation. On the contrary, by moving the derivative point away from the neighboring solution point (where $\sigma = 1$), numerical dissipation increases. The case of σ = 3.2 reveals notable departure from the exact solution and for a value of 40, the physical diffusion is already overwhelmed by the added numerical dissipation. The use of large σ value thus must be used with caution.

To benchmark effects of numerical dissipation against a triangular mesh, the quad mesh is triangulated along a consistent diagonal cut. Results by using the c-scheme and CNI scheme with $\sigma = 0.4$ are shown in Fig. 5. Both

calculations show very good agreement with the exact solutions. Compared with the quad mesh results using the c-scheme, it appears that numerical dissipation is intrinsically lower for the triangular mesh. Note that the triangular mesh has twice as many elements as the quad mesh. However, the increased grid resolution is mainly along the streamwise direction and the wall-normal direction has the same resolution for both quad and triangular meshes. A calculation for the quad mesh was repeated by doubling the streamwise grid from 81 to 161 in order to make it more compatible with the triangular mesh. The results show that the velocity profile remains unchanged. It is thus confirmed that more accurate solution by using the triangular mesh is unlikely to be related to the increased number of elements in the streamwise direction. In the above calculations, only a small number of grid points are used in the wall-normal direction. Numerical dissipation thus plays an important role in the solution. Given enough points as those shown in Fig. 3, the c-scheme would also give very good agreement with the exact solution. In a typical Navier-Stokes calculation, however, the viscous boundary layer is not necessarily well-resolved. In that case, controlling numerical dissipation is the key to ensure accurate solutions inside the boundary layer.

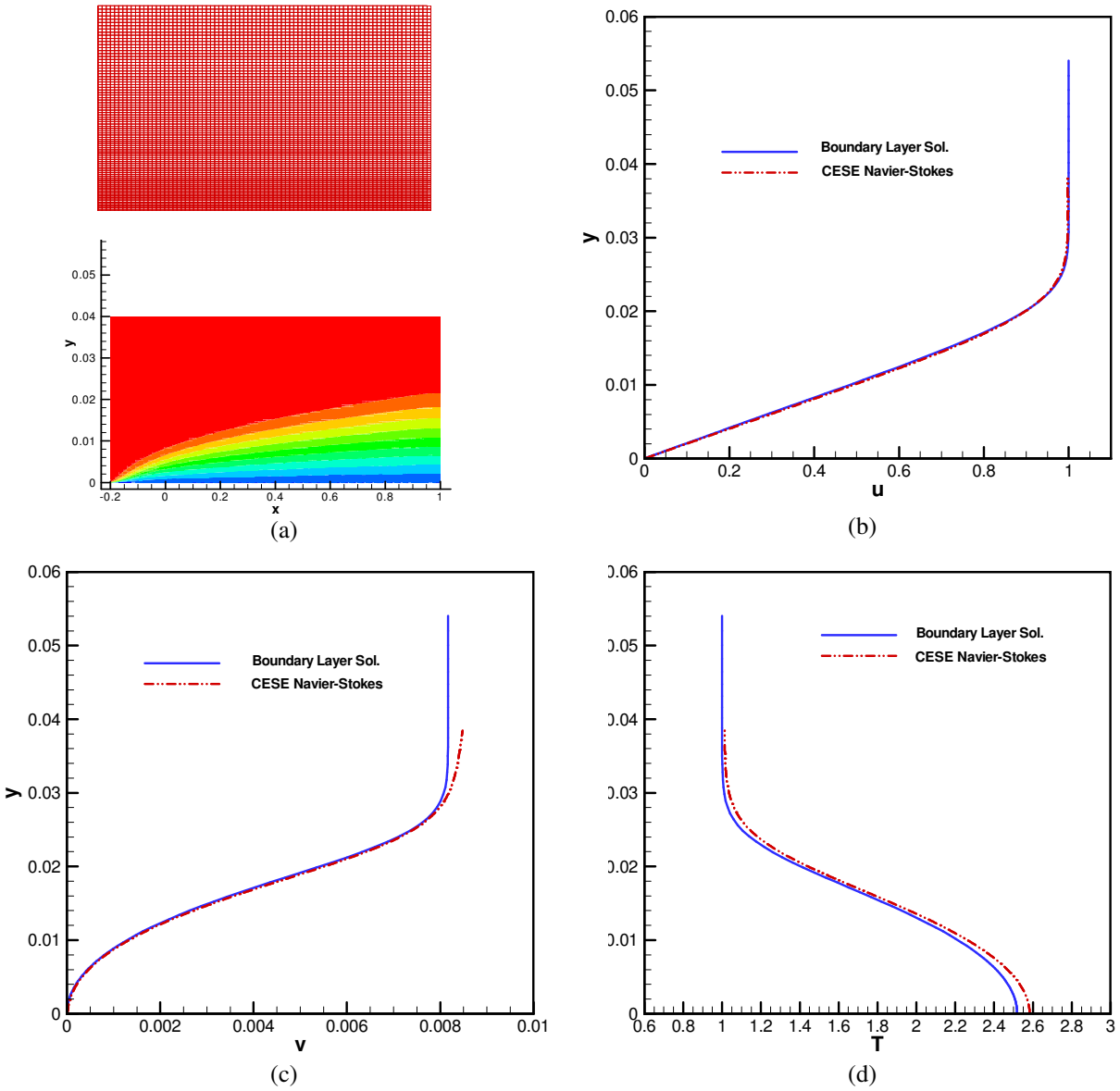


Figure 3. Mach 3 flow over an adiabatic flat plate: (a) quad mesh and streamwise velocity contours (b) streamwise velocity (c) wall-normal velocity (d) temperature profiles at $\mathrm{Re}_x = 1\times10^5$.

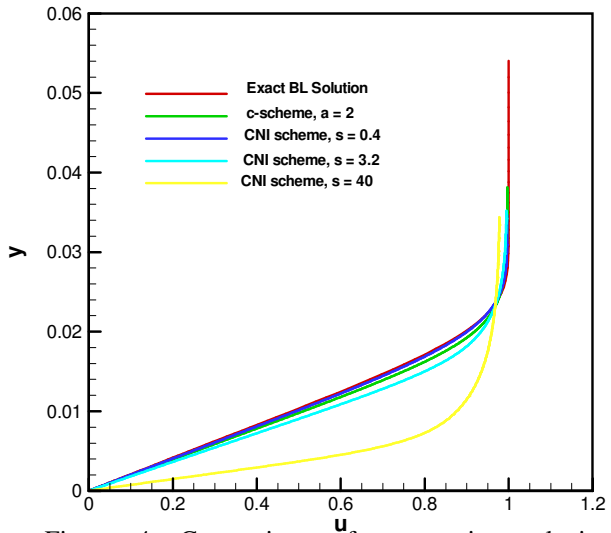


Figure 4. Comparison of streamwise velocity profiles of the Mach 3 flat plate boundary solutions obtained by a quad mesh using various numerical dissipation controls.

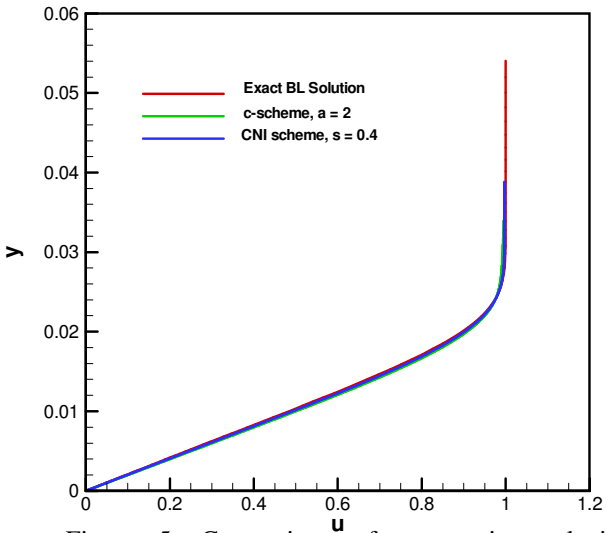


Figure 5. Comparison of streamwise velocity profiles of the Mach 3 flat plate boundary solutions obtained by a triangular mesh using two numerical dissipation controls.

Controlling numerical dissipation also benefits the inviscid solutions. The effectiveness of the CNI scheme for resolving flow discontinuities is evident in the following case where a Mach 3 flow over a 5^0 ramp is calculated. The mesh and computed density contours are shown in Figs. 6(a)-6(d). Strong clustering is used in the wall-normal direction to mimic viscous simulations even though the results shown here are from Euler solutions only. Figure 6(b) shows density contours obtained by using the c-scheme with a constant time step (CFL number is 0.9). The shock thickness increases downstream as numerical dissipation increases with the decreasing local CFL number due to the growing mesh. With a variable time step (local time-stepping), the results in Fig. 6(c) shows better shock resolution. If time-accurate solutions are of interest and local time-stepping cannot be used, CNI scheme with a σ value less than 1 can be used to reduce numerical dissipation (Fig. 6(d)).

For highly non-uniform meshes frequently encountered in real world problems, the use of local time-stepping and the CNI scheme plays an important role in controlling numerical dissipation for the CESE method. Local time stepping is used for steady-state problems and the CNI scheme with a small value of σ is used to control numerical dissipation for time-accurate solutions. In cases when very high-aspect ratio meshes are present, a σ value larger than 1 is used to maintain numerical stability. How to devise an automatic, adaptive numerical dissipation control for the CNI scheme is still an open topic for future research. The outcome would significantly increase the robustness of the CESE method, especially for viscous flow applications.

Low Speed Unsteady Viscous Flows Validation

We focus on time-accurate solutions in this section. The Stoke's second problem [37] serves as a good test case for low-speed time accurate viscous flow validation. There are two ways to mimic the problem: by vibrating the wall or the free-stream. A 6x101 quad mesh with 101 in the wall-normal direction is used for calculations. A mild stretching is applied in the wall-normal direction to better resolve the near wall viscous layer. Non-reflecting boundary conditions described in the previous section are used at both ends of the plate. Figure 7 shows the instantaneous velocity profiles at several physical times calculated by vibrating the plate. Except the initial transient, the computed profiles compare very well with exact solutions. The same problem is computed by imposing an acoustic wave at the inlet of the domain with a specified frequency. An 81x101 (with 41 points on the wall and 101 in the wall-normal direction) is used for calculations. The computed Fourier transformed velocity profiles are compared with exact solutions at several plate locations in Fig.8.

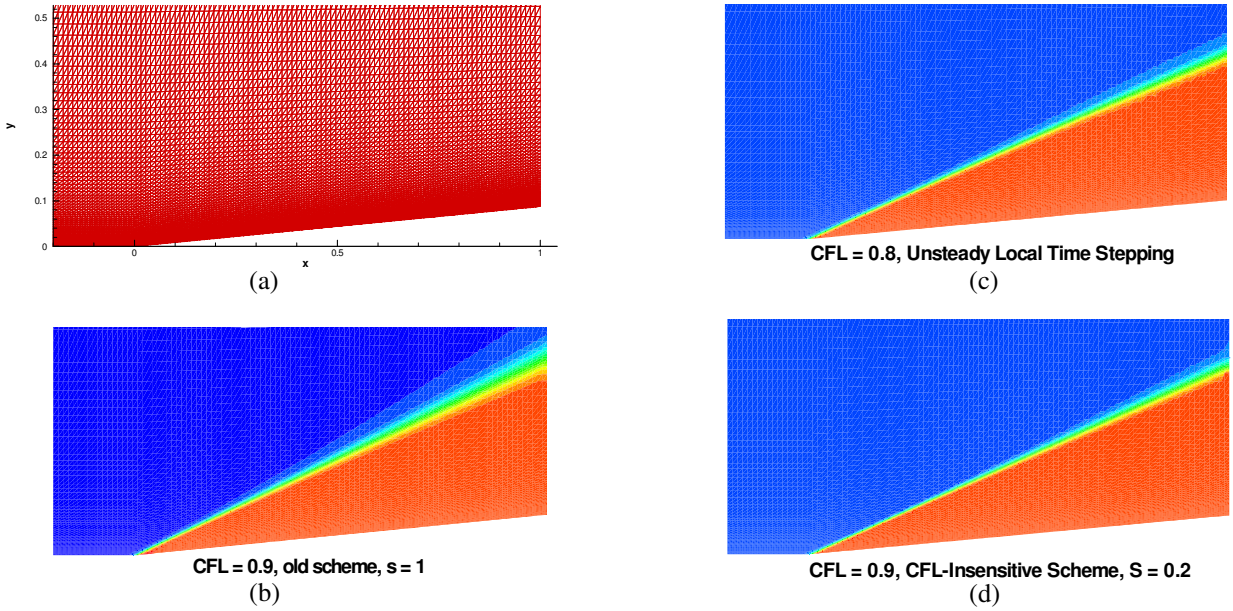


Figure 6. Mesh and density contours for a Mach 3 flow over a 5^0 wedge: (a) mesh (b) c-scheme, constant time step (c) c-scheme, local time-stepping (d) CNI scheme, constant time step $\sigma = 0.2$.

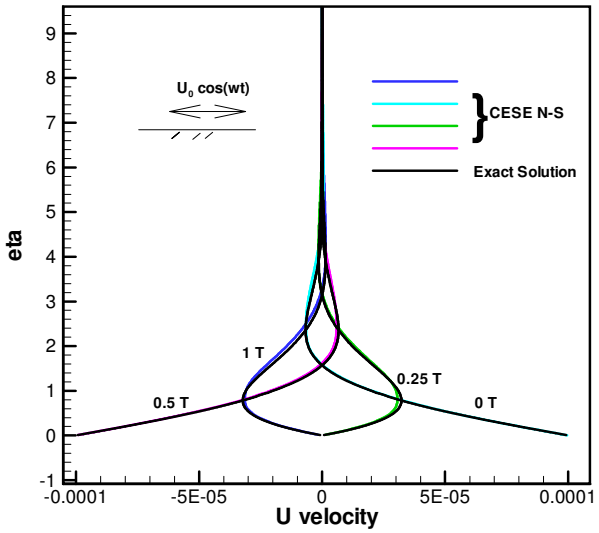


Figure 7. Instantaneous velocity profiles compared with exact solutions for Stoke's second problem computed by vibrating the plate.

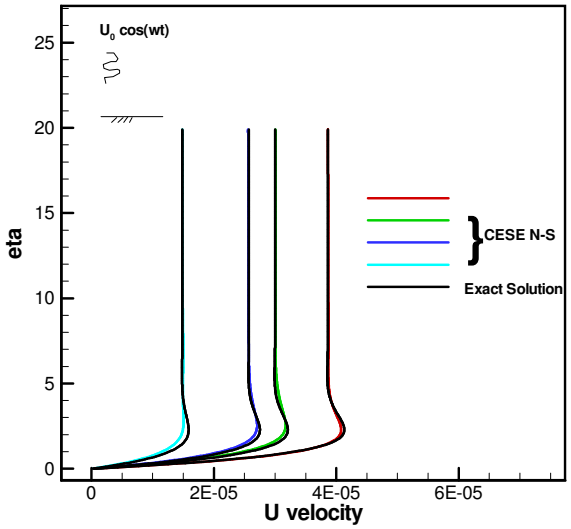


Figure 8. Transformed Fourier velocity profiles compared with exact solutions for Stoke's second problem computed by vibrating the plate.

These validation cases verified the temporal accuracy in conjunction with viscous terms treatment in the CESE Navier-Stokes calculations. Both incompressible and supersonic cases are investigated. In the next section, we will discuss non-reflecting boundary conditions for acoustic wave propagation.

Non-reflecting Boundary Conditions for Acoustic Wave Propagation

The non-reflecting boundary conditions (eq. (7)) discussed in the previous section have been used extensively for a broad range of problems with success (e.g. [21-24, 29]). One advantage the CESE method offers is that the far-field conditions need not to be located very far away from the body. For instance, a computational domain that extends about 5 cylinder radii is sufficient for calculations of a subsonic flow over a cylinder. Figures 9(a) and 9(b) show the o-grid used and the computed pressure contours, respectively. The free-stream Mach number is 0.2 and Euler calculations were performed. Even for such a small domain, only minor reflections are observed near the boundaries. The near field core flow structure appears to be almost unaffected by the presence of an artificial numerical boundary. Conventional schemes are much more sensitive to the proximity of numerical boundaries. As mentioned previously, the strong flux conservation property in space-time allows the CESE method to fulfill the hyperbolic characteristics in space-time and avoid error accumulation at the boundaries. Another example of the non-reflecting boundary conditions at work can be observed in Fig. 10 where the mesh and density contours are plotted for a two-dimensional plane acoustic wave propagating through a straight duct. With the non-reflecting boundary conditions, the wave exits the domain without any visible reflection. Other successful stories of the non-reflecting boundary conditions in computational aeroacoustic applications are also available (e.g. Yen[21]).

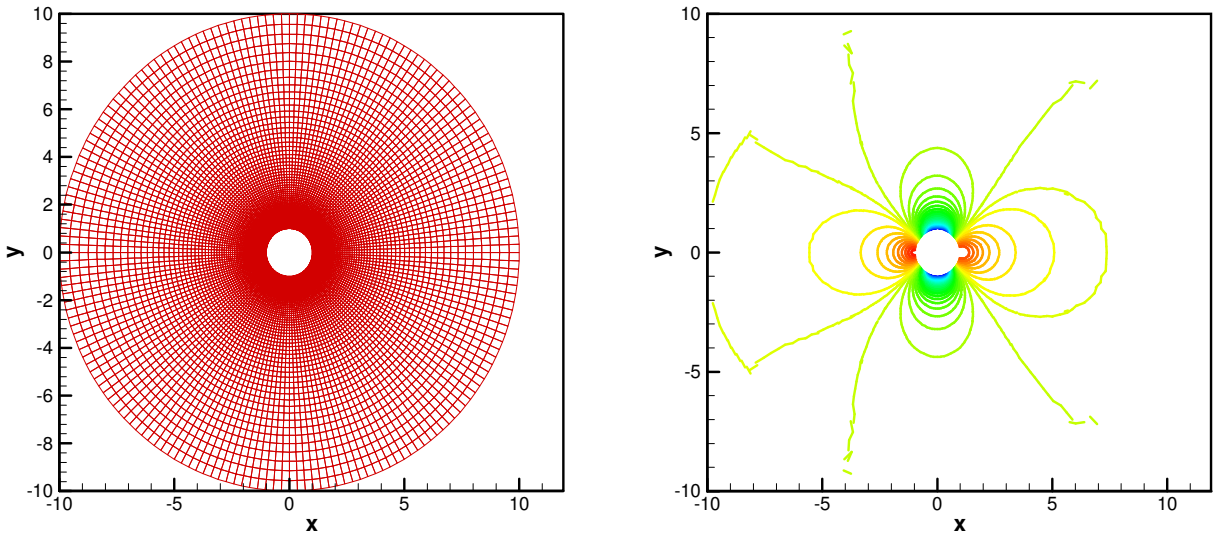


Figure 9. Mach 0.2 flow over a cylinder: (a) o-grid (b) computed pressure contours

To further investigate the effectiveness and limitations of the non-reflecting boundary conditions, we study again, the propagation of a plane acoustic wave in a rectangular domain. However, unlike the previous example shown in Fig. 10, the wave makes a 45^0 incident angle on the domain of interest. It is believed that a non-zero incident angle represents a harsher test for computational aeroacoustic problems. Acoustic wave boundary conditions are prescribed at the left and bottom faces of the rectangular domain with a 512x512 uniform mesh. The acoustic wave frequency is 1 kHz and Navier-Stokes equations are used for the computation even though no significant viscous effects are present. Analytical dispersion relation of the acoustic wave in a uniform mean flow is derived and imposed at the left and bottom boundaries [30]. Non-reflecting boundary conditions are applied at the remaining two boundaries. For a uniform mean flow, analytical solutions exist for fast and slow acoustic modes, vorticity wave, and the entropy wave. In a practical application, a combination of these waves can be applied at the boundaries. Here, we only impose the fast acoustic mode for the purpose of illustrating the effect of non-reflecting boundary conditions. When no flow is present, the computed density contours are shown in Fig. 11. Significant reflections exist after the wave hit the upper and right boundaries. It appears that additional waves with a comparable wave length are generated at the non-reflecting boundaries and propagate into the domain. It is interesting to note that the

reflected waves do not penetrate all the way to the left and bottom boundaries and roughly reach only one half of the domain. By and large, the spurious wave generated has an almost identical 45^0 wave angle and appears to be out of phase with the primary wave. A more detailed Fourier spectrum analysis, which is beyond the scope of the present paper, may shed some light of the nature of these spurious waves.

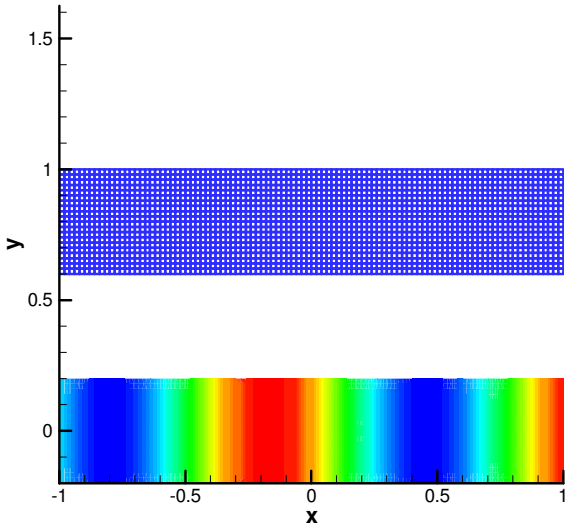


Figure 10. 2D plane acoustic wave exiting a rectangular duct without reflection.

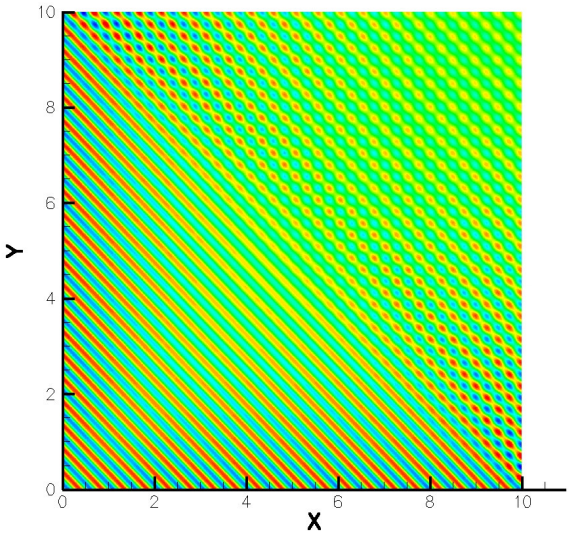
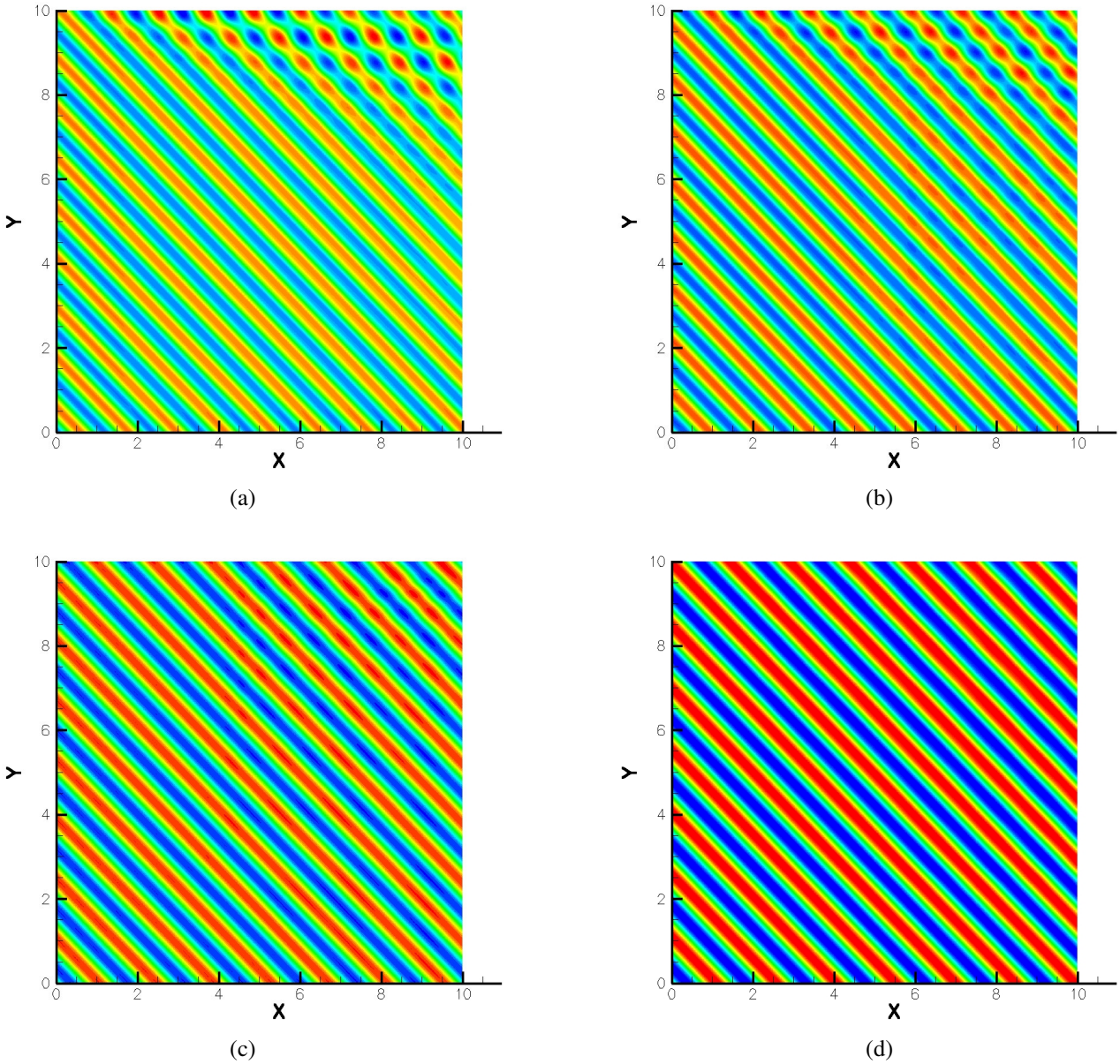


Figure 11. 2D plane acoustic wave passing through a rectangular domain from left and bottom at a 45^0 incident angle; showing boundary reflection

The same acoustic wave is immersed in a uniform mean flow at Mach 2. To study the effect of mean flow speed on the non-reflecting boundaries, we vary the mean flow angle from 0^0 (with respect to the x-axis) to 45^0. The resulting effective Mach numbers in the x direction are 2, 1.99, 1.81, and 1.41 for four cases presented in Figs. 12(a)-12(d), respectively. The corresponding Mach numbers in the y direction are 0, 0.17, 0.84, and 1.41, respectively. Boundary reflections are clearly present for the first two small flow angles. For the 25^0 case, only very minor reflections can be observed near the upper right corner. For the last case with a mean flow angle of 45^0, all boundary reflections completely disappear. It is evident that the boundary reflection of acoustic wave is strongly influenced by the mean flow Mach number. When the mean flow is convecting at a supersonic speed, all characteristics are going out of the domain along the space-time and no wave reflection occurs. On the contrary, for subsonic convecting Mach number, noticeable reflection occurs. Lower subsonic Mach numbers seem to have stronger wave reflection. It appears that the non-reflecting boundary conditions, eq. (7), still allows acoustic waves to be reflected at the boundaries if the wave makes a non-zero incident angle on the boundary face and some of the characteristics of the hyperbolic conservation equations are traveling into the domain (subsonic mean flow).

In summary, the simple non-reflecting boundary conditions normally used in the CESE method work well for allowing large scale numerical oscillations to exit the domain without causing far-reaching effects of boundary reflections. For small-scale acoustic waves, no reflection can be observed if the wave front is parallel to the boundary face. When there is an incident angle, the local convecting Mach number determines whether reflection takes place or not. For a subsonic mean flow, waves reflect and the generated spurious, out of phase, numerical perturbations with a comparable frequency (and thus wave length) are present. On the other hand, no wave reflection occurs if the local mean flow is supersonic. It is worth noting that the strong conservation nature of the CESE method does offer one benefit in that the artificial boundary wave reflection has a finite domain of influence. It is illustrated that for all the unsuccessful cases shown above, the extent of contamination is finite and concentrates in the vicinity of the truncated boundaries. No accumulated errors due to long time integration are observed in all these cases. To completely eliminate small-scale reflection, a small buffer domain is necessary near the truncated

boundaries. Alternatively, it is also possible to construct an improved non-reflecting boundary condition by locally altering the mean flow speed and characteristics similar to what is done in conventional non-reflecting boundary conditions based on the upwind differencing concept.



(a) (b)

(c) (d)

Figure 12. 2D plane acoustic wave propagating through a rectangular domain with a uniform Mach 2 mean flow and different flow angles: (a) 0^0 (b) 5^0 (c) 25^0 (d) 45^0

Time-Accurate Hypersonic Viscous Flow Calculations

With the recent NASA space exploration program, hypersonic flow simulation becomes one of the important research topics in aeronautics research. The focus on the design of crew exploration vehicles (CEV) prompts interests in computing hypersonic flows over blunt bodies to simulate entry, descent, and reentry of the CEV type vehicles. Next example illustrates one such case for a wind-tunnel model of a cone-shape body submerged in a

supersonic free-stream as shown in Fig. 13(a). The free-stream Mach number is 3.5. Triangular mesh with strong clustering near the model surface is generated for simulation. There are a total of about 110,000 vertices and 370,000 elements. Constant wall temperature and adiabatic wall conditions are applied at the cone surface and the sting, respectively. Symmetric conditions are applied at the center line. Axisymmetric option in the *ez4d* code is used for the calculations. A snapshot of he computed Mach number contours is shown in Fig. 13(b). The flowfield is rather complex. Several important features frequently encountered in hypersonic flow simulations are present in this example. Firstly, the detached bow show ahead of the cone is captured correctly without any carbuncle phenomenon [31-33]. The famous carbuncle phenomenon is a numerical instability associated with the upwind scheme that takes place at the detached bow shock near the stagnation point region where one of the eigenvalues is nearly zero. For the CESE method, no such phenomenon has ever been observed. Secondly, flow separation also occurs behind the leading edge of the cone base. The shear layer instability thus causes flow unsteadiness that may persist all the way till the end of the model. Also present in the flowfield is the shock-shock interaction after the bow shock reflects from the wind tunnel wall. The reflected shock then impinges on the sting surface to cause shock-boundary layer interaction and local flow separation.

All of the above flow features requires a time-accurate Navier-Stokes code that can capture the bow shock well without causing any numerical instability near the blunt leading edge and yet, retain sufficient numerical accuracy to resolve the shock-boundary layer interaction. The complex shock-shock interaction also requires a high-resolution and stable shock capturing scheme that can resolve the entropy gradient and possible unsteady shear layer instabilities. The space-time CESE method offers one such numerical framework for hypersonic flow calculations pertinent to the CEV type applications. More details of the capsule flow physics simulations are discussed in a separate paper [34]. In addition, we note here that real gas effects and turbulence models also play an important part in hypersonic flow simulations and must be addressed in future development.

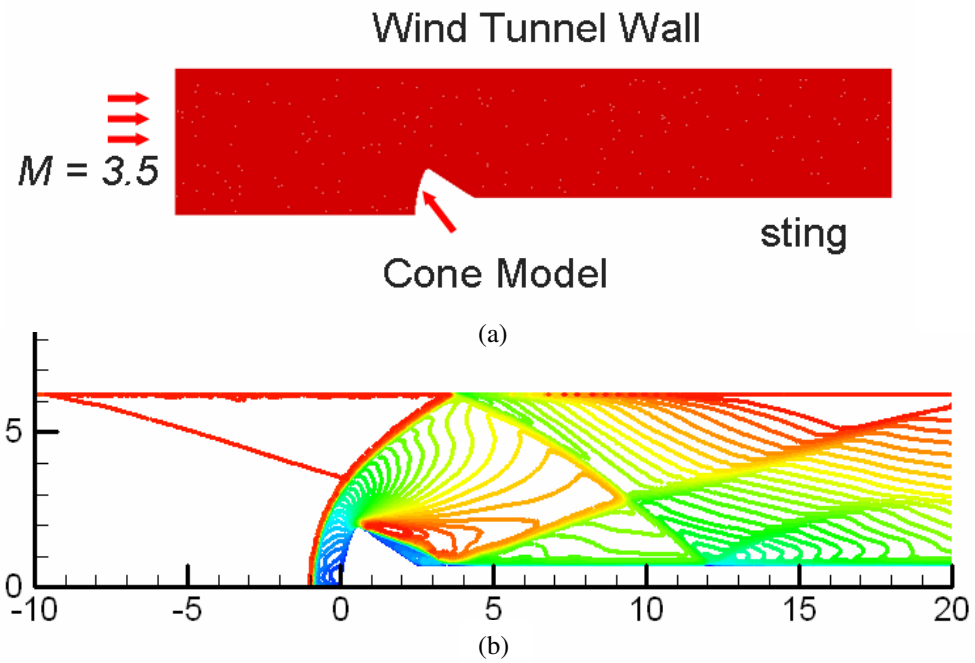


Figure 13. Mach 3.5 flow over a capsule wind-tunnel model with sting: (a) triangular mesh (b) computed Mach number contours.

3D Navier-Stokes Validations

The *ez4d* code has been written for both two- and three-dimensional Navier-Stokes equations. There is no separate 2D or 3D version. The 2D option also includes axisymmetric capabilities. All the C++ class objects and routines are shared by both 2D and 3D calculations. In all class definitions, the dimension number (2 for 2D and 3 for 3D) is a template parameter. In fact, the 3D path of the code works automatically after a simpler 2D path has been thoroughly debugged. The use of a template parameter to determine 2D/3D calculations avoids many common coding mistakes that are caused by cut and paste of the 2D routines for the 3D path. To validate the 3D path, a few 2D cases were made three-dimensional by stacking up replicated 2D surfaces. The resulting "3D" configurations should reproduce the two-dimensional results.

For validation purpose, a supersonic flow over a cylinder is calculated by using the 3D hex mesh shown in Figs. 14(a)-14(d). The computed density contours in Fig. 14(c) and surface pressure distributions in Fig. 14(d) were plotted by stacking up all spanwise planes. The results clearly show no spanwise variation at all. Again, there is no carbuncle phenomenon in the bow shock region. The surface pressure distribution computed is very clean and shows no sign of contamination from the numerical entropy generation near the shock as reported by the upwind finite-volume based unstructured method [15]. Typical shortcomings of the upwind unstructured method are associated with the numerical instability of the bow shock near the center line. A smaller CFL number must be used to maintain numerical stability and avoid carbuncle phenomenon. However, the small CFL number solution tends to vary in the spanwise direction and unsteady spurious disturbances are also generated near the bow shock. The same test case was also repeated by using the CESE method with a tetrahedral mesh and similar results were obtained. From the algorithm stand point, the CESE method appears to be free of all the numerical troubles associated with the hypersonic blunt body problems.

Application of the *ez4d* code for three-dimensional configurations is the subject of other papers [34-35]. Here, we demonstrate one example by showing the results of the capsule configuration (shown in Fig. 13) with a 10^0 angle of attack. Figure 15(a) shows the tetrahedral mesh used. The computed Mach number contours are shown in Fig. 15(b). Extension of the CESE method to 3D calculations is straightforward numerically. However, the high aspect ratio tetrahedral mesh near the viscous wall may cause more numerical problems than its 2D counterpart. The other observation is that the c-scheme for 3D tetrahedral meshes appears to be less diffusive and robustness becomes an issue for highly non-uniform grids. More research is underway to tackle all these issues.

IV. Summary

The newly immerged space-time CESE method is investigated for three-dimensional Navier-Stokes equations. Several issues regarding the computation of viscous flows are discussed in details. The no-slip boundary condition implementation within the CESE framework is discussed with validations against exact solutions for steady-state as well as time-accurate solutions. Numerical accuracies of the method and boundary conditions used are verified via good agreements with exact solutions. Effectiveness and issues of the non-reflecting boundary conditions normally used for the CESE method are illustrated and discussed for acoustic wave propagation in a truncated domain. It was found that boundary reflection does occur when a small scale wave is impinging on the boundary with an incident angle. The amount of reflection appears to depend on the local convective mean flow speed. Low subsonic mean flow tends to give more boundary reflection. It was also shown that the boundary reflection is confined to a finite region near the truncated boundary. This observation may imply that a compact buffered domain could be easily constructed to eliminate upstream propagation.

The stiffness issues for a very high aspect ratio mesh near the viscous wall are discussed. A high order framework in conjunction with an optimal numerical diffusion control based on the CFL insensitive scheme may help alleviate the robustness issues associated with the high aspect ratio mesh. The current CESE method is an explicit scheme. To facilitate time-accurate calculations of very large 3D configurations or LES and DNS, an implicit CESE method is probably necessary. It was shown in this paper that for hypersonic flow applications, the CESE method offers several advantages in treating the rich flow physics involved. Bow shocks encountered near the blunt leading edges are resolved by the CESE method without any problem. Unlike most finite-volume based unstructured methods, the strong flux conservation in conjunction with a set of carefully design conservation elements for flux integration in

the CESE method provides a viable flow solver for hypersonic applications. This time-accurate, unstructured mesh 3D Navier-Stokes solver based on this relatively new method is currently being used for investigating a broad range of problems.

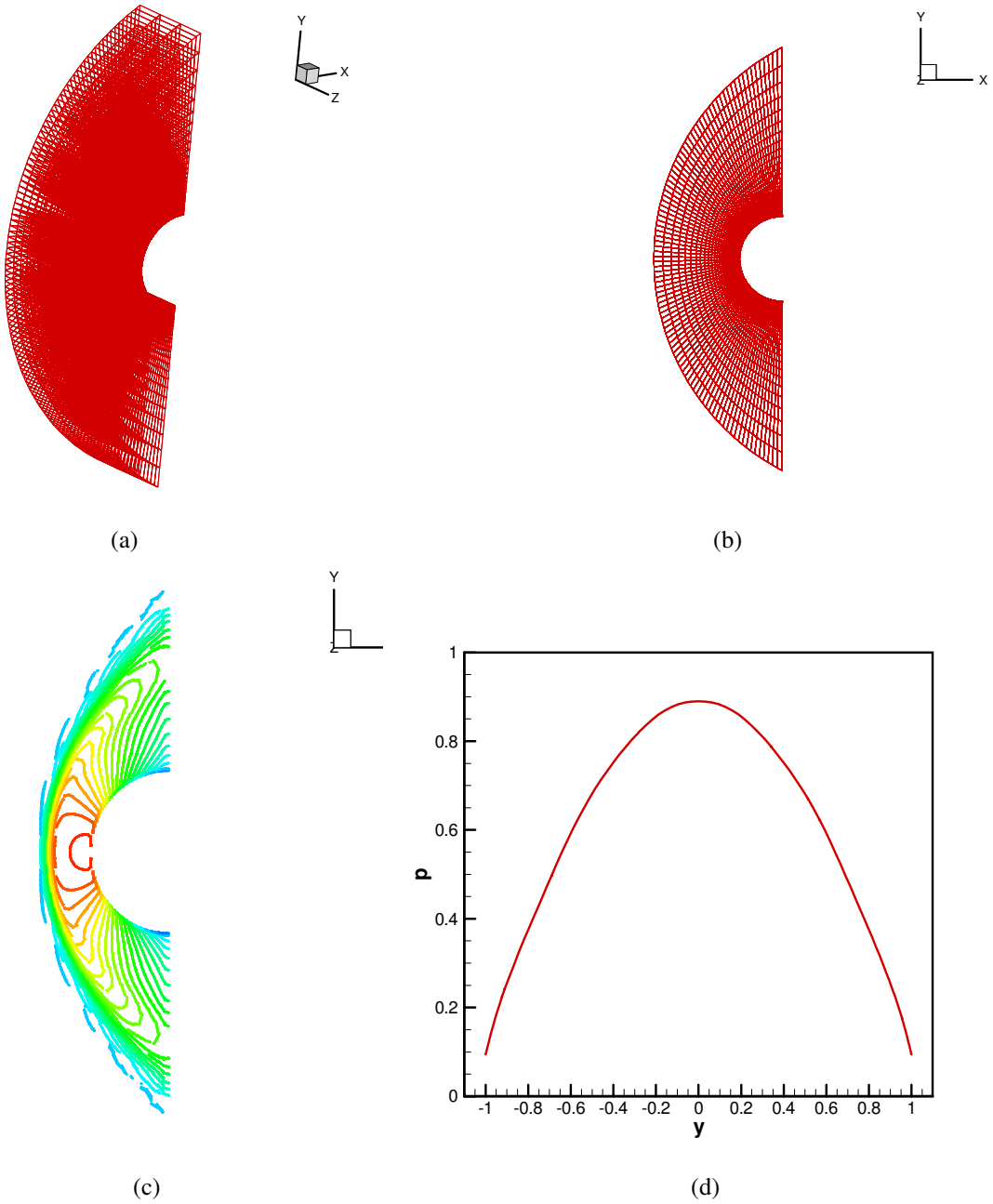
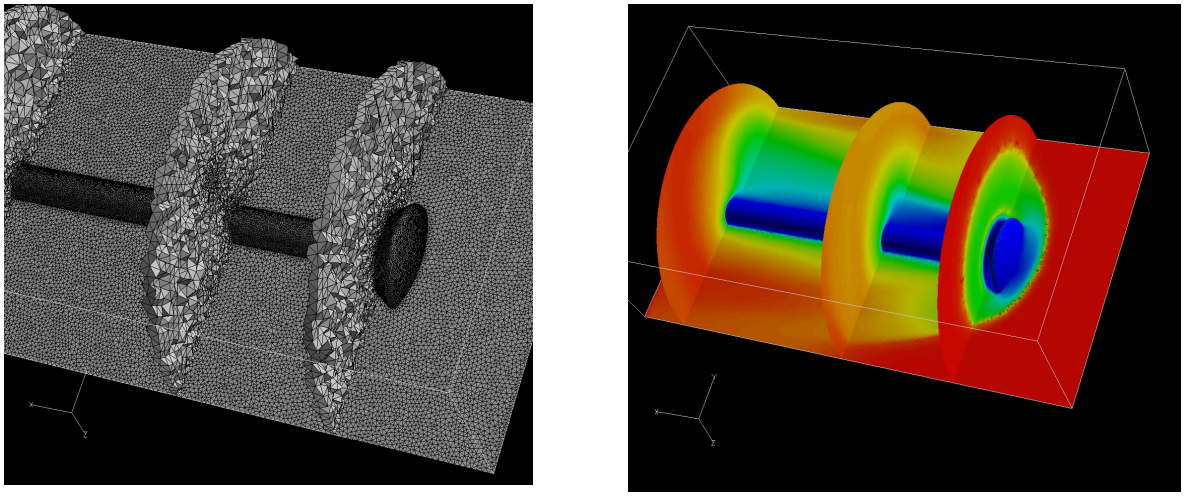


Figure 14. Mach 5 flow over a cylinder computed with a 3D hex mesh: (a) 3D mesh (b) mesh projected on the *x*-*y* plane (c) computed density contours (d) computed surface pressure distribution



(a) (b)

Figure 15. Mach 3.5 flow over a wind tunnel capsule mode: (a) 3D tetrahedral mesh (b) Mach number contours for a 10^0 angle of attack.

Acknowledgments

The author is very grateful to many inspiring conversations with Dr. Sin-Chung Chang of NASA Glenn Research Center. Technical discussions with Dr. I-Shih Chang of Aerospace Corporation, Prof. Sheng-Tao Yu of Ohio State, Dr. Joseph Yen of Jacobs Engineering, and Prof. Gary Cheng of University of Alabama at Birmingham are also acknowledged. The author would like to thank the help of Dr. Fei Li of National Institute of Aerospace for implementing the oblique acoustic wave boundary conditions.

References

1. Zienkiewicz, O. C, Taylor, R. L., and Nithiarasu, P, "The Finite Element Method for Fluid Dynamics," *Elsevier B. V.,* 2005.
2. Chung, T. J, "Computational Fluid Dynamics," *Cambridge University Press*, 2002.
3. Fischer, P. F., Kruse, G. W., and Loth, F., "Spectral Element Methods for Transitional Flows in Complex Geometries," *J. Scientific Computing*, Vol. 17, Nos. 1-4, December 2002.
4. Cockburn, B. and Shu, C.-W., "TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws II: General Framework," *Mathematics of Computation*, Vol. 52, No. 186, pp. 411-435, 1989.
5. Cockburn, B., Lin, S. Y., and Shu, C.-W., "TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws III: One Dimensional Systems," *J. Comput. Physics*, Vol. 84, No. 1, pp. 90-113, 1989.
6. Atkins, H. L. and Shu, C.-W., "Quadrature-Free Implementation of Discontinuous Galerkin Method for Hyperbolic Equations," *AIAA J.*, Vol. 36, No. 5, pp. 775-782, 1998.
7. Landmann, B., Kessler, M., Wagner, S., and Krämer, E., "A Parallel Discontinuous Galerkin Code for the Navier-Stokes Equations," AIAA Paper 2006-111, 2006.
8. Darmofal, D, "Towards a High-Order Discontinuous Galerkin Method for Aerospace Applications," NASA Langley presentation, Feb. 2006.
9. Yen, L., Vinokur, M., and Wang, Z. J., "Multi-Dimensional Spectral Difference Method for Unstructured Grids," AIAA Paper 2005-320, 2005.

10. May, G. and Jameson, A., "A Spectral Difference Method for the Euler and Navier-Stokes Equations on Unstructured Meshes," AIAA Paper 2006-304, 2006.
11. Helenbrook, B. T. and Atkins, H. L., "Application of p-Multigrid to Discontinuous Galerkin Formulations of the Poisson Equation," *AIAA J.*, Vol. 44, No. 3, pp. 566-575, 2006.
12. Anderson, W. K. and Bonhaus, D. L., "An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grid," *Computers and Fluids*, Vol. 23, No. 1, pp. 1-21, 1994.
13. Barth, T. J., "Numerical Aspects of Computing High-Reynolds Number Flows on Unstructured Meshes," AIAA Paper 91-0721, 1991.
14. http://fun3d.larc.nasa.gov/
15. Gnoffo, P. A. and White, J. A., "Computational Aerothermodynamics Simulation Issues on Unstructured Grids," AIAA Paper 2004-2371, 2004.
16. Chang, S.-C., "The Method of Space-Time Conservation Element and Solution Element–A New Approach for Solving the Navier-Stokes can Euler Equations," *J. Comput. Physics*, Vol. 119, pp.295-324, 1995.
17. Chang, S.-C., "The Space-Time Conservation Element and Solution Element Method: Motivating Ideas, Basic Schemes and Its Recent Development," Workshop on *Large-Scale Multi-Physics Unstructured Mesh Simulation with the Space-Time Conservation Element-Solution Element Method*, National Cheng-Kung University, Tainan, Taiwan, December, 2005.
18. Wang, X.-Y. and Chang, S.-C., "A 2D Non-Splitting Unstructured Triangular Mesh Euler Solver Based on the Space-Time Conservation Element and Solution Element Method," *J. Computational Fluid Dynamics*, Vol. 8, No. 2, pp. 309-325, 1999.
19. Wang, X.-Y. and Chang, S.-C., "A 3D Non-Splitting Structured/Unstructured Euler Solver Based on the Space-Time Conservation Element and Solution Element Method," AIAA Paper 99-3278, 1999.
20. Zhang, Z. C., Yu, S.-T., and Chang, S.-C., "A Space-Time Conservation Element and Solution Element Method for Solving the Euler Equations Using Quadrilateral and Hexahedral Meshes," *J. Comput. Physics*, Vo. 175, pp. 168-199, 2002.
21. Yen, J. C. and Wagner, D. A., "Computational Aeroacoustics Using a Simplified Courant Number Insensitive CE/SE Method," AIAA Paper 2005-2820, 2005.
22. Chang, I.-S., Chang, C.-L., and Chang, S.-C., "Unsteady Navier-Stokes Rocket Nozzle Flows," AIAA Paper 2005-4353, 2005.
23. Zhang, M., S.-C. Chang, H. Lin, S. T. J. Yu, "Application of the Space-Time CE/SE Method to the Ideal Magnetohydrodynamic Equations," AIAA paper 2002-3888, 2002.
24. Venkatachari, B. S. and Cheng, G. C., and Chang, S.-C., "Courant Number Insensitive Transient Viscous Flow Solver Based on CE/SE Framework," AIAA Paper 2005-0093, 2005.
25. Hirsch, C., "Numerical Computation of Internal and External Flows," *John Wiley & Sons*, 1991.
26. Chang, S.-C., "High-Order Space-Time CESE Method," private communication.
27. Chang, S.-C. and Wang, X.-Y., "Multi-Dimensional Courant Number Insensitive CE/SE Euler Solvers for Applications Involving Highly Non-uniform Meshes," AIAA Paper 2003-5285, 2003.
28. Chang, S.-C., Zhang, Z.-C., Yu, S.T.J., and Jorgenson, P. C. E., "A Unified Wall Boundary Treatment for Viscous and Inviscid Flows in the CE/SE Method," NASA /TM-2000-210517.
29. Loh, C., Hultgren, L. S., and Chang, S.-C., "Wave Computation in Compressible Flow Using Space–Time Conservation Element and Solution Element Method," *AIAA J.* Vol. 39, No. 5, pp. 794-801, 2001.
30. Li, F., private communication.
31. Pandolfi, M. and Ambrosio, D. D., "Numerical Instabilities in Upwind Methods: Analysis and Cures for the "Carbuncle" Phenomenon," *J. Comput. Physics*, Vo. 166, pp. 271-301, 2001.
32. Robinet, J.-Ch., Gressier, J., Casalis, G., and J.-M. Moschetta, "Shock Wave Instability and the Carbuncle Phenomenon: Same Intrinsic Origin?" *J. Fluid Mech*. Vo. 417, pp. 237-263, 2000.
33. Kim, S.-S., Kim, C, Rho, O.-J., and Hong, S. K., "Cures for the Shock Instability: Development of a Shock-Stable Roe Scheme," *J. Comput. Physics*, Vo. 185, pp. 342-374, 2003.
34. Cheng, G., Chang, C.-L., and Venkatachari, B., "Unsteady Aerothermodynamics Analysis of Hypersonic Flows over Reentry Capsules," AIAA Paper 2006-4776, 2006.
35. Chang, I.-S., Chang, C.-L., and Chang, S.-C., "3D Unsteady Navier-Stokes Rocket Nozzle Flows," AIAA Paper 2006-4775, 2006.
36. Harris, J. E. and Blanchard, D. K., "Computer Program for Solving Laminar, Transitional, or Turbulent Compressible Boundary-Layer Equations for Two-Dimensional and Axisymmetric Flow," NASA Technical Memorandum 83207, 1982.
37. Schlichting, H., "Boundary Layer Theory," McGraw-Hill, New York (7^{th} ed. 1979).