The Space-Time Conservative Schemes for Large-Scale, Time-Accurate Flow Simulations with Tetrahedral Meshes

Balaji Shankar Venkatachari[*]
National Institute of Aerospace, Hampton, VA 23666

Craig L. Streett,[†] Chau-Lyan Chang[‡]
NASA Langley Research Center, Hampton, VA 23681

David J. Friedlander,[§] Xiao-Yen Wang,[**] and Sin-Chung Chang,[††]
NASA Glenn Research Center, Cleveland, OH 44315

Despite decades of development of unstructured mesh methods, high-fidelity time-accurate simulations are still predominantly carried out on structured, or unstructured hexahedral meshes by using high-order finite-difference, weighted essentially non-oscillatory (WENO), or hybrid schemes formed by their combinations. In this work, the space-time conservation element solution element (CESE) method is used to simulate several flow problems including supersonic jet/shock interaction and its impact on launch vehicle acoustics, and direct numerical simulations of turbulent flows using tetrahedral meshes. This paper provides a status report for the continuing development of the space-time conservation element solution element (CESE) numerical and software framework under the Revolutionary Computational Aerosciences (RCA) project. Solution accuracy and large-scale parallel performance of the numerical framework is assessed with the goal of providing a viable paradigm for future high-fidelity flow physics simulations.

Nomenclature

E	=	turbulent kinetic energy
F, G, H	=	flux vectors for compressible Navier-Stokes equations in three spatial directions
$\vec{h}$	=	flux in the joint space-time domain
k_x, k_y, k_z	=	wave numbers in x, y, and z direction, respectively
k_0	=	the most energetic turbulent wave number
M	=	freestream Mach number
$M_{1,}$, M_2	=	Mach numbers before and after the normal shocks
M_t	=	turbulent Mach number
$\vec{n}$	=	unit surface normal
$\overrightarrow{e_x}$, $\overrightarrow{e_y}$, $\overrightarrow{e_z}$, and $\overrightarrow{e_t}$	=	unit vectors along the x-, y-, z-, and t- directions
p	=	static pressure
Re	=	Reynolds number
s	=	entropy
$\vec{s}$	=	surface normal vector in the joint space-time domain
t	=	time
T	=	temperature normalized by the freestream value
u	=	streamwise velocity

[*] Research Engineer, Email: balaji.s.venkatachari@nasa.gov, Senior member, AIAA

[†] Senior Research Scientist, Computational AeroSciences Branch, Email: craig.l.streett@nasa.gov, Member AIAA.

[‡] Aerospace Technologist, Computational AeroSciences Branch, Email: Chau-Lyan.Chang@nasa.gov, Associate Fellow, AIAA

[§] Aerospace Engineer, Inlet and Nozzles Branch, Email: d.j.friedlander@nasa.gov, Member AIAA

[**] Aerospace Engineer, Thermal Systems Branch, Email: xiao-yen.j.wang@nasa.gov, Member AIAA

[††] Aerospace Engineer, Inlet and Nozzles Branch, Email: sin-chung.chang@nasa.gov, Member AIAA

U	=	dependent solution vector
v	=	wall-normal velocity
V	=	space-time volume
w	=	spanwise velocity
x	=	streamwise coordinate
y	=	wall-normal coordinate
z	=	spanwise coordinate in direction parallel to leading edge
γ	=	ratio of the specific heats
λ	=	turbulent Taylor microscale
ρ	=	density
ω	=	vorticity

Subscripts

x, y, z, t	=	derivatives in spatial and temporal directions
∞	=	freestream conditions

Superscripts

i	=	number index elements

I. Introduction

After several decades of development, computational fluid dynamics (CFD) software is routinely used as a low-cost alternative to wind tunnel experiments. In addition to integrating CFD tools in the design process, great deal of concept validations and pre-/post-experimental analyses have been carried out by computational tools. The demand for high-fidelity, time-accurate computations for intricate flow physics has become the main theme for CFD development for the next few decades.[1] Along this line, NASA has been focusing on the development of next generation CFD algorithms and software under the Revolutionary Computational Aerosciences (RCA) project. Potential numerical algorithms that possess high solution accuracy, robustness, and excellent parallel performance have been investigated vigorously in the past several years. While no single algorithm could satisfy all the requirements, this paper focuses on time-accurate, unstructured-mesh numerical solutions of unsteady waves, their interactions with shocks, and the requirement to have good parallel scaling for large-scale computations. Despite decades of CFD development, it is generally agreed that these issues are still very challenging for many numerical algorithms and CFD software.

The importance of resolving waves and shock interactions goes without saying. For transonic and supersonic flows, acoustic, vorticity, and entropy waves are generated regularly as a result of the environment and surface irregularities. These waves interact with flow discontinuities that arise when supersonic flow adjusts its direction to comply with the surface geometries. In turbulent flows with high enough turbulent Mach numbers, local flow acceleration causes shocklets to develop. Accurately resolving small to medium turbulent structures interacting with shocklets is of fundamental importance to accurately simulating high-speed turbulent flows. The choice of unstructured meshes, in particular, triangular and tetrahedral meshes, stems from the requirement to handle more and more complex geometries in many realistic engineering problems. The capability of local refinement without incurring additional overhead associated with topological constraints makes unstructured mesh generation more appealing than its structured counterpart. Being able to handle corners, bumps, cavities and other geometric irregularities easily is another important reason to employ unstructured meshes for high fidelity computations. For the simulation of turbulent flows, the isotropic nature of the tetrahedral mesh provides a discretized domain that is free of any orientation that could potentially influence the damping of mid to high wave number fluctuations. For the resolved scales in typical large eddy simulations, the isotropic nature of unstructured mesh could also enhance un-biased energy transfer to smaller scales.

Large-scale parallel performance depends on many factors. Implicit schemes require iterative LU decomposition of a relatively large matrix. Dimensional splitting and relaxation schemes are often used to reduce the matrix size. For time accurate solutions of unsteady flows, the time step is often restricted by the physical resolution requirement rather than the convergence rate. Explicit schemes are preferred in this regard because fewer operations are required to march the solution per time step. The stability bound for most explicit schemes is around CFL one, or the acoustic time scale. For transonic or higher speed, this time step is appropriate to resolve instability or acoustic waves. Communication latency is another major factor to gauge parallel performance. Implicit schemes have the advantage of a large amount of computations per communication, especially when the total number of decomposed domains is relatively large. The communication overhead is thus relatively smaller than the explicit schemes if the same number of parallel domains is used, despite the fact that overall computational time for implicit schemes may still be longer. Explicit schemes, on the other hand, can be utilized by using a smaller number of domains with more elements in each domain. GPU or multi-thread acceleration can further enhance the parallel performance. For time-accurate computations, the smallest element in the mesh determines the time step. This implies that a large part of the computational domain has a CFL number much smaller than one. Latency effect is significant in these regions. Dynamic domain decomposition in time[2] can be used to improve such latency effects for elements running at very small CFL numbers. Alternatively, as in the space-time CESE method,[3–5] a time-accurate local time stepping (TALTS) scheme can be employed, with every element in a mesh, with large disparity is mesh size, marching at a uniform physical time step close to CFL number one and yet guaranteeing conservation in space and time. This approach can be implemented without the costly approach of repartitioning large 3D meshes.

In light of the discussion above, efficient simulations for unsteady flows hinge upon tailoring parallel computations with hardware architectures as well as accurate numerical algorithms. With the success of pushing the state-of-the-art for structured-mesh simulations for decades, more work is needed to explore the use of unstructured meshes and the required numerical algorithms to improve accuracy, robustness, and parallel performance. Along this line, this paper investigates numerical simulations of unsteady flows using explicit, strongly conservation property preserving schemes with unstructured, tetrahedral meshes. In particular, the space-time conservation element, solution element (CESE) method[3–8] is used for large-scale unsteady flow simulations with tetrahedral meshes. Time-accurate flow computation is investigated from the perspective of four unique aspects. Firstly, the CESE method is formulated by enforcing the strong space-time unity integral form of the conservation laws.[5–8] A high degree of conservation is in general a key aspect in preserving numerical accuracy for waves and turbulent flow simulations. The flux conservation in time introduced in the CESE method, in particular, could offer appreciable advantages in time-accurate computations. Secondly, as discussed above, the time-accurate local time-stepping (TALTS) scheme[3–5] formulated based on the CESE method is appealing to improve the shortcoming of heavy latency effects for explicit schemes. For flows with eddy shocklets at high turbulent Mach numbers, the range of temporal and spatial scales widens significantly owing to the fact that the energy containing eddies have a much larger length scale than the dissipation structures.[9] Local time-stepping algorithms allow larger time steps to be used for large flow structures away from the wall and in the quiet freestream regions. Thirdly, the CESE method is able to capture waves and flow discontinuities with high fidelity.[4–5,10–11] Finally, due to its space-time conservation formulation, the CESE allows easy and efficient implementation of non-reflecting boundary conditions.[13–15] For turbulent flow simulations, substantial savings and ease of implementation can be achieved without the need for a buffer domain in the outflow/farfield regions. The genuinely multi-dimensional formulation of the CESE method, which is free of approximate Riemann solvers and dimensional splitting, offers a distinctly different way to simulate turbulent flows interacting with shocks.

This paper presents a status report of the continuing development of the CESE numerical and software framework for high fidelity numerical simulations of general conservation laws. In the march towards development of a matured tool for time-accurate simulation of multi-physics problems using unstructured meshes, order of accuracy studies for several benchmark problems, large-scale computations of realistic configurations and canonical problems are discussed in detail. Investigations of these problems are aimed at simulating more complex and realistic problems such as direct numerical simulations of turbulent boundary layer-shock interactions over a compression corner or other more complex geometries.

II. Foundation Aspect of the CESE Numerical Framework—Conservation in space-time

This section describes one of the key foundational ideas that went into the development of the CESE numerical framework—solving the space-time unity integral form of conservation laws. Most of the core design ideas of this numerical method have been expounded in detail in several works (see Refs. 6–8).

It was felt that this particular aspect is substantially different from the mainstream ideas prevalent in the CFD community, and should be explained in a more thorough manner.

CFD as a research field aims to develop the most accurate, robust, and efficient methods for simulating the conservation laws of fluid flows. As such, ideally the methods should be developed from the most fundamental and general form of these conservation laws. Unfortunately, most (if not all) of the established CFD methods are based on some special forms of conservation laws which are derived from the general form with restrictive assumptions. Not only do these assumptions make it impossible for these methods to strictly enforce mass, momentum and energy conservation in unsteady flows, thus causing a serious loss of accuracy in unsteady flow simulations, but they also have a negative impact on the robustness and efficiency of these methods.[16,17] In this section, we attempt to explain why the space-time unity integral form of the conservation laws is the most general form and why it would be better for a numerical scheme to be developed starting with this form rather than the differential form of conservation laws. In Ref. [16], Roe himself talked about the strengths of developing a numerical scheme on the basis of a space-time unity integral form. In other branches of physics (particularly electromagnetics) there have also been efforts[17–18] that utilize the space-time unity integral form of conservation laws as the starting point in the development of a numerical method.

Consider the conservation laws governing 3D unsteady flows. Let (a) x, y, and z be the spatial coordinates, and t be the time coordinate; (b) $x_1 \overset{\text{def}}{=} x, x_2 \overset{\text{def}}{=} y, x_3 \overset{\text{def}}{=} z,$ and $x_4 \overset{\text{def}}{=} t$ be the coordinates of a four-dimensional Euclidean space E$_4$; (c) $\overrightarrow{e_x}$, $\overrightarrow{e_y}$, $\overrightarrow{e_z}$, and $\overrightarrow{e_t}$ be the unit vectors along the x-, y-, z-, and t-directions respectively; and (d) $\vec{h}_m,\ m = 1, 2, 3, 4, 5,$ be the space-time mass, x-momentum, y-momentum, z-momentum, and energy flux density vectors respectively. Then $\vec{h}_m,\ m = 1, 2, 3, 4, 5,$ can be expressed as

$$\vec{h}_m = U_m \overrightarrow{e_t} + F_m \overrightarrow{e_x} + G_m \overrightarrow{e_y} + H_m \overrightarrow{e_z}, m = 1, 2, 3, 4, 5 \tag{1}$$

where (a) (d) U_m, $m = 1, 2, 3, 4, 5,$ are the mass, x-momentum, y-momentum, z-momentum, and energy per unit spatial fluid volume respectively; and (b) each of F_m, G_m, and H_m, $m = 1, 2, 3, 4, 5,$ is a differentiable functions of $U_m,\ m = 1, 2, 3, 4, 5$ for inviscid flows, while it is also a function of the spatial derivatives of $U_m, m = 1, 2, 3, 4, 5$ for viscous flows. Then, the most fundamental and general form of the unsteady conservations laws applied over a space-time flow domain D in E_4 can be cast into the following space-time unity integral form:

$$\oint_{S(V)} \vec{h}_m \cdot \mathrm{d}\vec{s} = 0 \text{ for any space-time subdomain } V \text{ in } D,\ m = 1, 2, 3, 4, 5 \tag{2}$$

Here (a) $S(V)$ is the boundary of the arbitrary space-time subdomain V in D, and (b) $\mathrm{d}\vec{s} = \mathrm{d}\sigma\vec{n}$ with $\mathrm{d}\sigma$ and $\vec{n}$, respectively, being the area and the outward unit normal vector of a surface element on $S(V)$. Note that $\vec{h}_m \cdot \mathrm{d}\vec{s}$ is the space-time flux of $\vec{h}_m$ leaving the space-time subdomain V through the surface element $\mathrm{d}\vec{s}$. Thus, for each subdomain V of the space-time flow domain D, Eq.(2) imposes the conservation condition that the total space-time flux of $\vec{h}_m$ leaving V through its boundary is zero. This form of the conservation law is valid for all 3D unsteady flows, including those with solution discontinuities, such as shocks and contact discontinuities and is the form employed by the space-time CESE schemes.

Let (a) t_1 and t_2 be any two real parameters with $t_1 < t_2$, (b) $\underline{V}$ be a fixed and bounded spatial domain (hereafter referred to as the control volume) in the x-y-z space denoted by E_3, and (c)

$$C(t_1, t_2; \underline{V}) \overset{\text{def}}{=} \{(x, y, z, t) | (x, y, z) \in \underline{V} \text{ and } t_1 \le t \le t_2\} \tag{3}$$

By definition, $C(t_{1,}t_{2+}; \underline{V})$ is a space-time cylinder/hyper-volume E_4 which : (a) is sandwiched between two planes defined by $t = t_1$ and $t = t_2$, respectively; and (b) has the property that $\underline{V}$ is the common (spatial) projection onto E_3 of any cross-section of $C(t_{1,}t_{2+}; \underline{V})$ cut by any $t = t_0$ plane where t_0 is any constant meeting the condition $t_1 \le t_0 \le t_2$. Note that, for the special case $V = C(t_{1,}t_{2+}; \underline{V})$, Eq. (2) reduces to

$$\oint_{S(C(t_1,t_2;\underline{V}))} \vec{h}_m \cdot \mathrm{d}\vec{s} = 0, m = 1, 2, 3, 4, 5 \tag{4}$$

Let

$$S_1\left(C\left(t_1,t_2;\underline{V}\right)\right) \overset{\text{def}}{=} \left\{\left(x,y,z,t\right)\middle|\left(x,y,z\right) \in \underline{V} \text{ and } t = t_1\right\} \tag{5}$$

$$S_2\left(C\left(t_1,t_2;\underline{V}\right)\right) \overset{\text{def}}{=} \left\{\left(x,y,z,t\right)\middle|\left(x,y,z\right) \in \underline{V} \text{ and } t = t_2\right\} \tag{6}$$

$$S_0\left(C\left(t_1,t_2;\underline{V}\right)\right) \overset{\text{def}}{=} \left\{\left(x,y,z,t\right)\middle|\left(x,y,z\right) \in \underline{S}\left(\underline{V}\right) \text{ and } t_1 < t < t_2\right\} \tag{7}$$

where $\underline{S}(\underline{V})$ is the boundary of the control volume $\underline{V}$ in E_4 . The hypersurface $\underline{S}(\underline{V})$ can be defined by

$$\underline{S}\left(\underline{V}\right) \overset{\text{def}}{=} \left\{\left(x,y,z\right)\middle|e\left(x,y,z\right) = 0\right\} \tag{8}$$

where $e(x,y,z)$ is a continuous and piecewise smooth real function defined on a subdomain of E_3, such that $e(x,y,z) = 0$ represents a closed surface in E_3. With the aid of Eq.(8), Eq. (7) implies

$$S_0(C(t_1,t_2;\underline{V})) \overset{\text{def}}{=} \{(x,y,z,t)|e(x,y,z) = 0 \text{ and } t_1 < t < t_2\} \tag{9}$$

As a consequence, it is clear that the hypersurfaces $S_1(C(t_1,t_2;\underline{V}))$, $S_2(C(t_1,t_2;\underline{V}))$, and $S_0(C(t_1,t_2;\underline{V}))$ (see Eqs. (5) - (7)) are respectively the bottom face, top face and side faces of $S(C(t_1,t_2;\underline{V}))$. In fact, their union is the boundary of $C(t_1,t_2;\underline{V})$; i.e.,

$$S(C(t_1,t_2;\underline{V})) = \bigcup_{l=0}^{2} S_l(C(t_1,t_2;\underline{V})) \tag{10}$$

The outward normals for these hypersurfaces can be given as:

$$\begin{aligned} &\text{For } S_1\left(C\left(t_1,t_2;\underline{V}\right)\right): \ \vec{n} = -\vec{e_t} \\ &\text{For } S_2\left(C\left(t_1,t_2;\underline{V}\right)\right): \ \vec{n} = \vec{e_t} \\ &\text{For } S_0\left(C\left(t_1,t_2;\underline{V}\right)\right): \ \vec{n} = \underline{n}\left(x,y,z\right) \end{aligned} \tag{11}$$

Note that in Eq.(11), the unit outward normal vector $\vec{n}$ has no component in (x,y,z) direction for surfaces $S_1(C(t_1,t_2;\underline{V}))$, and $S_2(C(t_1,t_2;\underline{V}))$, while it has no component in the t- direction for the surface $S_0(C(t_1,t_2;\underline{V}))$. With these basics, now consider the mass conservation equation component (m = 1) of Eq.(4). Let (i) ρ be the mass density; (ii) u, v, w be the velocity components in the x-, y-, and z- directions respectively; and (iii) h_{mass} be the mass flux current density vector defined by

$$\vec{h}_{\text{mass}} \overset{\text{def}}{=} \rho\left(u\vec{e_x} + v\vec{e_y} + w\vec{e_z} + \vec{e_t}\right) \tag{12}$$

Using Eq. (12), and Eq(10), Eq. (4) implies,

$$\oint_{S(C(t_1,t_2;\underline{V}))} \vec{h}_{\text{mass}} \cdot \mathrm{d}\vec{s} = 0 \tag{13}$$

and

$$\oint_{S(C(t_1,t_2;\underline{V}))} \vec{h}_{\text{mass}} \cdot \mathrm{d}\vec{s} = \int_{S_0(C(t_1,t_2;\underline{V}))} \vec{h}_{\text{mass}} \cdot \mathrm{d}\vec{s} + \int_{S_1(C(t_1,t_2;\underline{V}))} \vec{h}_{\text{mass}} \cdot \mathrm{d}\vec{s} + \int_{S_2(C(t_1,t_2;\underline{V}))} \vec{h}_{\text{mass}} \cdot \mathrm{d}\vec{s} \tag{14}$$

Utilizing Eq.(11), we have

$$\begin{aligned} &\int_{S_1(C(t_1,t_2;\underline{V}))} \vec{h}_{\text{mass}} \cdot \mathrm{d}\vec{s} = -\oint_{\underline{V}} \rho(x,y,z,t_1)\,\mathrm{d}\underline{v} \\ &\int_{S_2(C(t_1,t_2;\underline{V}))} \vec{h}_{\text{mass}} \cdot \mathrm{d}\vec{s} = \oint_{\underline{V}} \rho(x,y,z,t_2)\,\mathrm{d}\underline{v} \\ &\int_{S_0(C(t_1,t_2;\underline{V}))} \vec{h}_{\text{mass}} \cdot \mathrm{d}\vec{s} = \int_{t_1}^{t_2} \mathrm{d}t \oint_{\underline{S}(\underline{V})} \vec{\underline{h}}_{\text{mass}}(x,y,z,t) \cdot \vec{\underline{n}}\,\mathrm{d}\underline{\sigma} \end{aligned} \tag{15}$$

where (a) d$\underline{v}$ is the volume of a volume element in the spatial domain $\underline{V}$, (b) $\vec{\underline{h}}_{\text{mass}} \overset{\text{def}}{=} \rho\left(u\overrightarrow{e_x} + v\overrightarrow{e_y} + w\overrightarrow{e_z}\right)$, and (c) $\mathrm{d}\underline{\sigma} \overset{\text{def}}{=}$ the area of a spatial surface element on $\underline{S}(\underline{V})$. With the aid of Eqs. (12) - (14), we have:

$$\oint_{\underline{V}} \rho(x,y,z,t_2)\,\mathrm{d}\underline{v} - \oint_{\underline{V}} \rho(x,y,z,t_1)\,\mathrm{d}\underline{v} = -\int_{t_1}^{t_2} \mathrm{d}t \oint_{\underline{S}(\underline{V})} \vec{\underline{h}}_{\text{mass}}(x,y,z,t) \cdot \vec{\underline{n}}\,\mathrm{d}\underline{\sigma} \tag{16}$$

In Eq. (16), the first two terms on the left hand side are the total mass in the control volume $\underline{V}$ at $t = t_2$ and $t = t_1$ respectively. The terms on the right hand side represent the total mass entering the control volume $\underline{V}$ through its boundary $\underline{S}(\underline{V})$ during the time interval (t_1, t_2). Thus Eq. (16) states the obvious mass conservation law. Consider the special case with $t_1 = t$ and $t_2 = t + \mathrm{d}t$. Then Eq.(16) reduces to

$$\oint_{\underline{V}} \rho(x,y,z,t+\mathrm{d}t)\,\mathrm{d}\underline{v} - \oint_{\underline{V}} \rho(x,y,z,t)\,\mathrm{d}\underline{v} = -\mathrm{d}t \oint_{\underline{S}(\underline{V})} \vec{\underline{h}}_{\text{mass}}(x,y,z,t) \cdot \vec{\underline{n}}\,\mathrm{d}\underline{\sigma} \tag{17}$$

If the derivative $\partial\rho(x,y,z,t)/\partial t$ exists for a time t for all the spatial points (x, y, z) in $\underline{V}$, we have

$$\frac{\partial}{\partial t} \oint_{\underline{V}} \rho(x,y,z,t)\,\mathrm{d}\underline{v} = \frac{\oint_{\underline{V}} \rho(x,y,z,t+dt)\,\mathrm{d}\underline{v} - \oint_{\underline{V}} \rho(x,y,z,t)\,\mathrm{d}\underline{v}}{\mathrm{d}t} \tag{18}$$

Then, by defining $\mathrm{d}\vec{\underline{s}} \overset{\text{def}}{=} \vec{\underline{n}}\,\mathrm{d}\underline{\sigma}$, Eq.(17) can be cast in the form:

$$\frac{\partial}{\partial t} \oint_{\underline{V}} \rho(x,y,z,t)\,\mathrm{d}\underline{v} = \oint_{\underline{S}(\underline{V})} \vec{\underline{h}}_{\text{mass}}(x,y,z,t) \cdot \mathrm{d}\vec{\underline{s}} \tag{19}$$

Eq.(19), a mixed differential-integral equation, is often referred to as the integral form of mass conservation law, in finite-volume schemes. Since it only involves a volume and surface integral in E_3, it is easier to visualize than the space-time unity integral form of mass conservation law, Eq.(13), that is defined in E_4. However, Eq.(19) is not applicable for a flow with discontinuities such as shocks and contact discontinuities, since it requires the existence of $\partial\rho / \partial t$ for every point (x, y, z) in $\underline{V}$. Even if $\partial\rho / \partial t$ exists for every point (x, y, z) in $\underline{V}$, Eq.(18) is based on the assumption:

$$\frac{\partial \rho}{\partial t}\left(x,y,z,t\right)\overset{\text{def}}{=}\lim_{\Delta t\to 0}\frac{\rho\left(x,y,z,t+\Delta t\right)-\rho\left(x,y,z,t\right)}{\Delta t} \tag{20}$$

and this assumption can never be fully realized by a numerical simulation, which can only be specified at discrete time levels. As a result, equivalency between Eq.(13) and Eq.(19) can never be fully realized in a numerical solution, even if the physical solution possesses the required analytical properties.

Now, let us additionally assume that the spatial vector field $\vec{\underline{h}}_{\text{mass}}$, introduced in Eq.(13), satisfies the analytical conditions that allow the applicability of Gauss-Divergence theorem in E_3. Then,

$$\oint_{\underline{S}(\underline{V})} \vec{\underline{h}}_{\text{mass}}\cdot \mathrm{d}\vec{\underline{s}} = \oint_{\underline{V}}\left(\vec{\underline{\nabla}}\cdot \vec{\underline{h}}_{\text{mass}}\right)\mathrm{d}\underline{v} \tag{21}$$

where,

$$\vec{\underline{\nabla}}\cdot \vec{\underline{h}}_{\text{mass}} \overset{\text{def}}{=} \frac{\partial\left(\rho u\right)}{\partial x}+\frac{\partial\left(\rho v\right)}{\partial y}+\frac{\partial\left(\rho w\right)}{\partial z} \tag{22}$$

Then, Eq.(17) can be cast into the form:

$$\oint_{\underline{V}}\left[\frac{\partial \rho}{\partial t}+\frac{\partial\left(\rho u\right)}{\partial x}+\frac{\partial\left(\rho v\right)}{\partial y}+\frac{\partial\left(\rho w\right)}{\partial z}\right]\mathrm{d}\underline{v}=0 \tag{23}$$

As $\underline{V}$ can be any subset of E_4, within the interior of the space-time domain E_4, one has,

$$\frac{\partial \rho}{\partial t}+\frac{\partial\left(\rho u\right)}{\partial x}+\frac{\partial\left(\rho v\right)}{\partial y}+\frac{\partial\left(\rho w\right)}{\partial z}=0 \tag{24}$$

Eq.(24) is the differential form of the conservation law. The additional analytical requirements imposed in deriving this form makes this form even more limited in its applicability than the mixed differential-integral form, Eq. (19).

Based on the discussion above, we hope we have elucidated the benefits of starting from the space-time unity form of conservation laws over any form derived with restrictive assumptions. A discrete system evolving from such an approach is expected to be a much richer system and closer to the physical phenomenon being described. For a more thorough exposition on the limitations of starting computational physics from differential equations, the reader is referred to Ref. [17].

III. Results

The CESE numerical framework has been implemented for general conservation laws as the *ez4d* software at NASA Langley Research Center. The *ez4d* software framework has been developed using a combined object-oriented and generic programming paradigm in the C++ programming language. Light-weight object-oriented hierarchy is used in conjunction with heavy use of template classes and functions to allow compile time polymorphism. Different conservation laws can be plugged in with templates that represent physics. Currently, the software supports either triangular/tetrahedral or quadrilateral/hexahedral unstructured meshes. Both multi-thread (based on low-level POSIX thread) and MPI parallel computations are used to facilitate large-scale computations. Each MPI process within a computational node can be executed in multi-thread mode to further enhance parallel performance, especially for a memory bound multi-domain layout. A communication map is used for data transfer among interface zones. For a large mesh in the order ofa billion elements, each unstructured block can be built with its own connectivity and nodes. A global communication map is then used to join all the independent blocks in the parallel computations. This arrangement allows the grid generation process to always have a low memory requirement. The interfaces among blocks can be continuous or discontinuous. A continuous interface mesh ensures better solution accuracy for unsteady flow computations. Both second- and fourth-order CESE numerical schemes are implemented for general conservation laws including Euler and Navier-Stokes equations in the software framework. As discussed above, the TALTS scheme is used to enhance parallel performance for elements running at CFL numbers much smaller than one. Load balance for such runs can be improved via volume-weighted domain decomposition offered in the Metis utility. For Navier-Stokes simulations, the Sparlart-Allmaras and Mentor's SST models[19] are implemented in the *ez4d* framework.

In the next several sub-sections, order of accuracy studies and some recent efforts on exploratory flow physics computations are discussed in detail.

A. Order of Accuracy Studies

High-order accuracy of the CESE method has been established for the linear advection equation, and the linearized and non-linear Euler equations in the literature.[5, 20–21] Good agreements with published results for acoustic waves interacting with shocks have also been established recently.[22] Here, we discuss three additional cases.

a. Transonic Ringleb Flow

This test case corresponds to a smooth inviscid transonic flow through a curved channel, where the exact solution is known through a hodograph transformation.[23] The flow is initialized with the analytical solution and the L_2 norm of the residuals is monitored for convergence to the steady-state solution. The L_2 norm of the entropy error, defined below, is used as an indicator of accuracy.

$$\text{Entropy error } (L_2) = \left[\frac{\sum_{i=1}^{N}\int_{V_i}(s-s_{exact})^2\mathrm{d}V}{\sum_{i=1}^{N}|V_i|}\right]^{1/2} \tag{25}$$

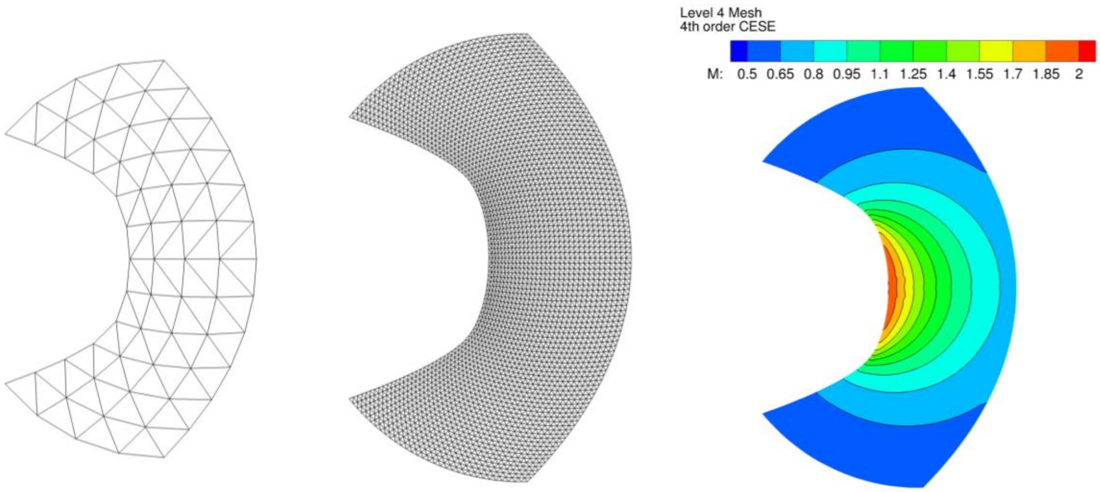
In the above equation, s stands for entropy, N for the number of elements, and V_i is the volume of element i. All the computations were performed in serial on a Linux desktop machine with an Intel Xeon processor X-5560 (Gainestown) and 48GB of RAM. The machine produces a Taubench[24] time that varies between 8.719 - 8.748 seconds, with an average of about 7.349 seconds. The time for performing 100 residual evaluations with degrees of freedom (DOF) = 250,000, for the 2nd-order accurate and 4th-order accurate schemes are summarized in Table 1. The residual evaluation is performed on a mesh with 6144 triangular elements and the times are scaled to 250,000 DOFs. The CESE solver for flows in two spatial dimensions has 1 DOF per element per equation for the 2nd-order accurate scheme and 4 DOFs per element per equation for the 4th-order accurate scheme respectively. The times reported have been normalized by the Taubench time and this unit is henceforth referred to as a work unit.

Table 1. Time taken for 100 residual evaluations with 250,000 DOFs.

Order of Accuracy	Time (work unit)
2	6.06
4	27.79

The domain was discretized into uniform isotropic triangular elements (linear elements) using Pointwise® with four levels of resolution. The standard boundary conditions (BC) to be enforced for this problem are either, subsonic inflow and outflow along with inviscid treatment (slip wall) at the two walls or by imposing the analytical solution at all the four boundaries. The latter procedure is adopted in this order of accuracy study. The mesh contained 96, 384, 1536, and 6144 triangular elements, respectively. The level 1 (96 elements) and level 4 (6144 elements) meshes are shown in Fig. 1, along with the computed

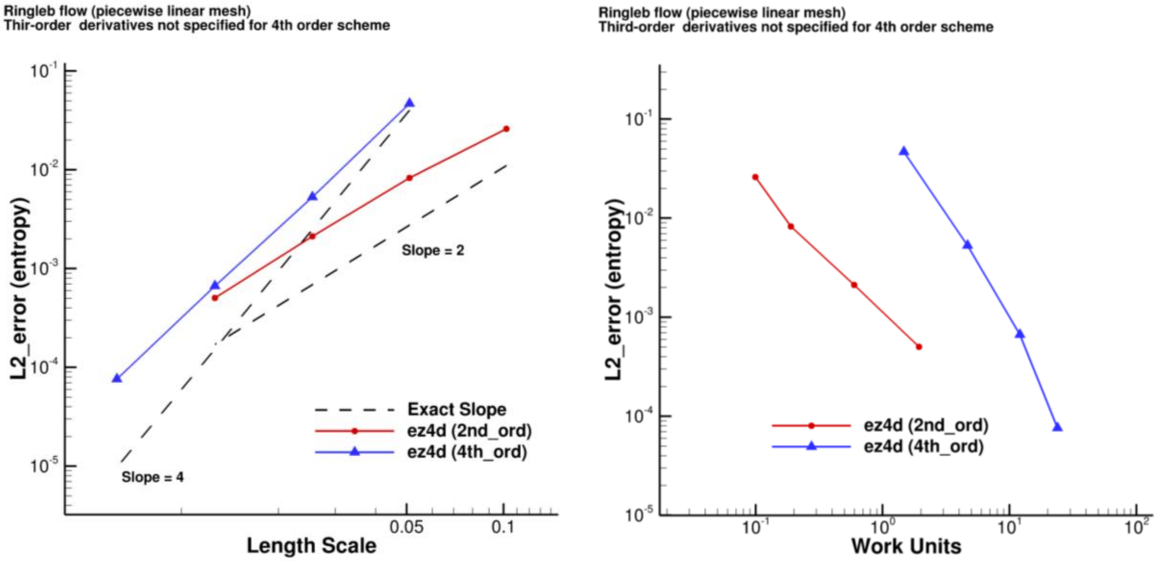
Mach number contours using the Level 4 mesh. Each level of mesh refinement was achieved by halving the average edge lengths of the previous level mesh. In contrast to most high-order-schemes investigations,[25] linear boundary elements were used throughout for both 2nd and 4th-order computations shown here. The piecewise linear boundary elements are widely available in most grid generation tools. Such computations provide a good assessment of accuracy using linear boundary elements.



(a) Level 1 Mesh (b) Level 4 Mesh (c) Mach Contour (Level 4 mesh)

Figure 1. Meshes with two different resolutions used for the Ringleb flow computations.

The error norm convergence obtained is shown in Fig. 2. The length scale utilized is = 1.0 / $\sqrt{DOFs}$. The order of convergence for the 4th order scheme is one-order lower than its design order. This is somewhat better than expected given the fact that only linear elements are used in the entire domain. Other studies[25] have shown that 4th or even higher-order convergence is achievable when higher-order schemes are used in conjunction with high-order elements throughout the entire domain. Another factor that could affect the overall accuracy of a CESE scheme, is that high-order derivatives at boundaries must be specified in the high-order CESE schemes. For the ringleb flow, there is a weak singularity in the third derivatives along the left and right streamline boundaries when the flow accelerates from subsonic to supersonic flows. High-order boundary treatments for the CESE method are currently being investigated and will be discussed in a future paper. The convergence versus work unit is shown in Fig. 2(b) for both 2nd and 4th-order schemes. For this benchmark problem, the high-order code does not appear to gain performance, partially due to the fact that the computer-generated high-order terms in the high-order code are not fully optimized by the compiler due to the length of some of the statements (those containing long expressions).



(a) Entropy error vs. length scale (b) Entropy error vs. work units

Figure 2. L_2 entropy error convergence for the half-domain computations.

b. Convecting Isentropic Vortex

The second test case chosen for assessment of accuracy of the CESE scheme is the horizontal convection of a 2-D isentropic vortex embedded in a uniform free stream. As vortical structures are part of many realistic flows (especially turbulent flows) and also play a critical role in many applications, it is important that the numerical scheme utilized to predict such flows convect and preserve the vortical structures accurately at all speeds.

The conditions used in the set-up of this problem are based on Ref. 26 and are slightly different from thode utilized in the international workshop on higher-order CFD methods.[25] Let the free stream conditions be such that it has a mean flow velocity, u_∞ and v_∞; and temperature $T_{¥}$. As an initial condition, an isentropic vortex with no perturbation in entropy is embedded into the mean flow field. The perturbation values are given by:

$$\delta u = \frac{\beta}{2\pi}(-\bar{y})e^{(1-r^2)/2}$$
$$\delta v = \frac{\beta}{2\pi}(\bar{x})e^{(1-r^2)/2} \qquad (26)$$

$$\delta T = -\frac{(\gamma-1)\beta^2}{8\gamma\pi^2}e^{(1-r^2)/2} \qquad (27)$$

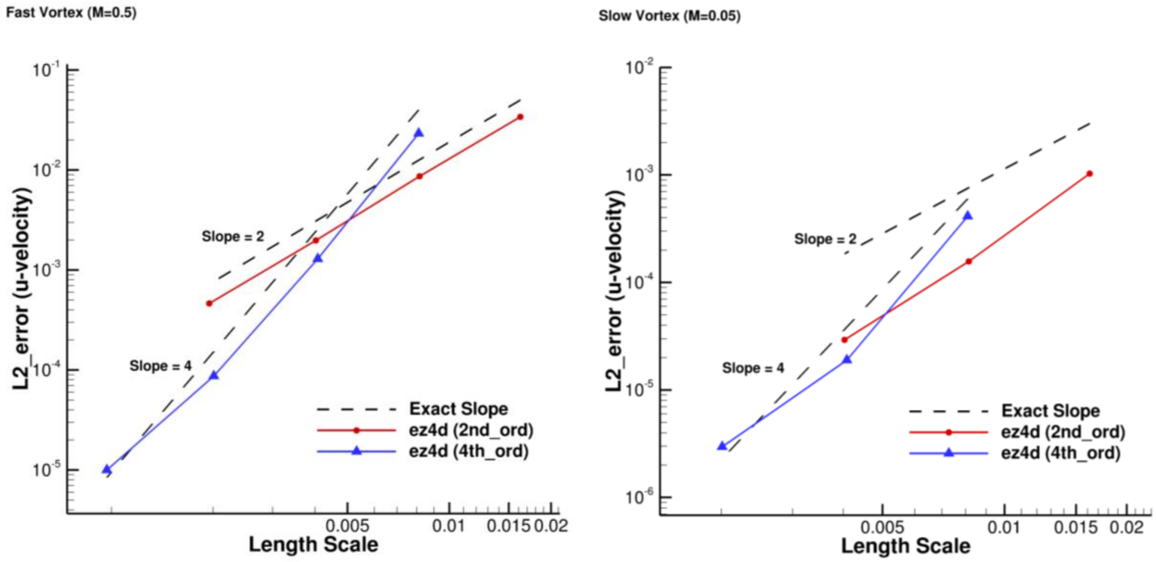
$$(\bar{x},\bar{y}) = \left(x - x_{v_0}, y - y_{v_0}\right);\;\; r^2 = \left(\bar{x}^2 + \bar{y}^2\right) \qquad (28)$$

where β is the vortex strength. $\left(x_{v_0}, y_{v_0}\right)$ are the initial coordinates of the vortex center, and $\gamma = 1.4$. The entire flow field is also required to be isentropic, so for a perfect gas, we have the relation

$$p/\rho^\gamma = \text{constant} \qquad (29)$$

The free-stream conditions (non-dimensionalized) are assumed to be such that $T_\infty = 1$ and $(u_\infty, v_\infty) = (1,0)$. In this study, vortices that convect fast with $M_\infty = 0.8415$, and slowly with $M_\infty = 0.05$ are studied. The computational domain is covered by the region $0 \le x \le 10$ and $-5 \le y \le 5$, with the vortex being placed at location, $(x_{v0}, y_{v0}) = (5,0)$. The vortex strength β is assumed to be 5.0 for the fast vortex and 0.25 for the slow vortex. Non-reflecting boundary conditions are enforced at the top and bottom boundaries, while periodic boundary conditions are enforced at the left and right boundaries respectively.

The computational domain consists of uniform isotropic triangular elements (linear elements) using Pointwise®, with four levels of resolution. The mesh contains 3,804, 15,000, 61,506, and 262,280 triangular elements, respectively. Each level of mesh refinement was achieved by halving the average edge lengths of the previous level mesh. The L_2 norm of the error (with respect to the exact solution) in u-velocity is shown after 5 time periods for the fast vortex and 1 time period for the slow vortex in Figs. 3(a) and (b). The 2nd-order code gives 2nd order convergence for the fast-vortex and slightly better than 2nd order convergence (~2.5) for the slow vortex. The 4th order code gives close to 4th order convergence for the fast and slow vortex, with the order of convergence slightly dipping below 4 (~3.7) for the final mesh in the fast-and slow vortex transport cases.

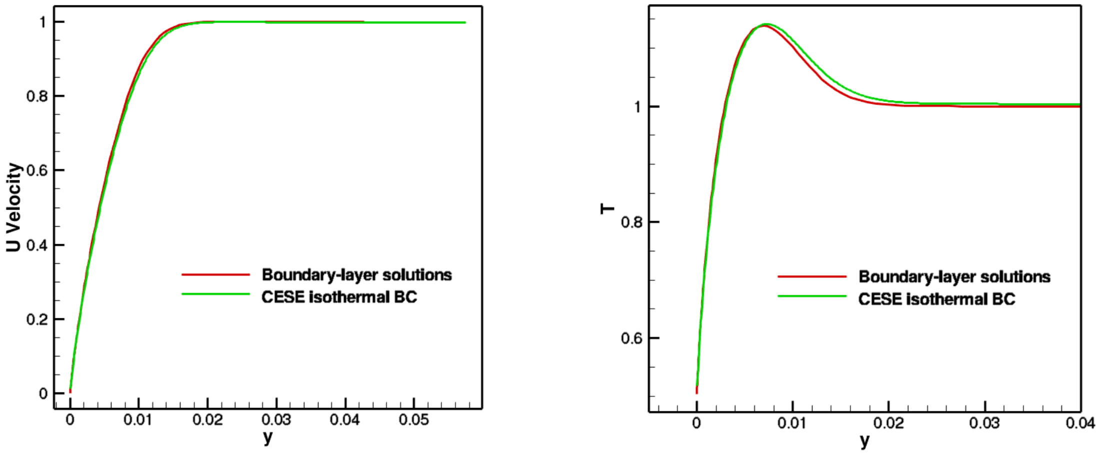


(a) Fast vortex transport **(b) Slow vortex transport**

Figure 3. L^2 error convergence for the vortex transport computations.

c. *Supersonic Isothermal Boundary Layer*

Surface heating prediction has been one of the most important topics in many supersonic or hypersonic computational investigations. Accurately predicting the surface heat flux in a highly cooled boundary layer has important implications in vehicle thermal shield and aerodynamic design. Of interest is the isothermal boundary layer developing behind the leading-edge bow shock of a blunt body. To isolate the boundary layer from shocks for numerical accuracy studies, a Mach 3 flow (with a free-stream temperature of 200 K) over a highly cooled boundary layer with $T_w / T_{adw} = 0.2$ is numerically computed. Due to the presence of the boundary layer, there is a weak leading edge shock that would make the free-stream conditions of the boundary layer deviate slightly from the Mach 3 conditions. The computational domain is a 1.2 $m \times 0.1$ m rectangle (the height is about 5 boundary-layer thicknesses at the exit). A structured quadrilateral mesh with different sizes is sliced to form triangular elements for computations. The computed velocity and temperature profiles (with the 4th-order scheme and a sliced 41×321 quad mesh and geometric stretching factor of 1.04 along the wall-normal direction) at the location of $Re = 10^5$ are compared with compressible similarity boundary layer solutions in Fig. 4. Due to the slight differences in free-stream conditions and boundary-layer assumptions, both the velocity and temperature profiles show some small discrepancies around the boundary-layer edge and the free-stream.



(a) (b)

Figure 4. Comparison of Navier-Stokes solutions with compressible boundary-layer solutions at *Re* = 10^5 for a Mach 3 flate-plate flow with $T_w/T_{adw} = 0.2$: (a) streamwise velocity (b) temperature.

Table 2. Grid convergence study for the Mach 3 isothermal boundary layer

Triangular Mesh (σ = geometric stretching factor)	**Velocity L_2 error norm**	**Temperature L_2 error norm**
1. Sliced $41 \times 81, \sigma = 1.04$ mesh	0.0409296	0.0340573
2. Sliced $41 \times 81, \sigma = 1.05$ mesh	0.00725355	0.00406225
3. Sliced $41 \times 81, \sigma = 1.06$ mesh	0.00508539	0.00272823
4. Sliced $41 \times 161, \sigma = 1.06$ doubled mesh	0.00226095	0.00132477
5. Sliced 161×161 uniform mesh	0.0516967	0.0323901

A sequence of grid refinement studies was used to assess the order of accuracy of the Navier-Stokes solver with this more realistic viscous solution (as opposed to other manufactured solutions). As can be seen from Fig. 4, both the near-wall velocity and temperature gradients agree quite well with the boundary-layer solutions. To better measure the accuracy, the entire profile at *Re* = 10^5 as shown in Fig. 4 is used as exact solutions to compute L_2 error norm with different grids, instead of comparing just the surface heat flux. Table 2 summarizes the accuracy and grid convergence for the 2nd-order scheme. As mentioned previously, the high-order derivative temperature boundary conditions are still under development and convergence of high-order schemes will be reported when the new boundary conditions are ready. In general, it was found that uniform grid refinement does not produce grid converged solutions for this highly cooled isothermal boundary layer. Computed error norms shown in Table 2 indicate that by pushing the grid non-uniformly (with a geometric stretching factor) toward the wall results in better accuracy with the same grid dimensions. There is a big jump in accuracy by going from a stretching factor of 1.04 to 1.05. The convergence rate slows down when σ is increased from 1.05 to 1.06. In the next mesh (denoted mesh 4 in Table 2), mesh 3 is doubled only in the wall-normal direction. The error continues to drop. The error norm ratio between mesh 3 and 4 is about 2.1, slightly faster than second-order. In contrast, a uniform and more refined mesh 5 gives an error that is even higher than the geometrically stretched coarser mesh 1. These results indicate that to resolve surface heat flux in an isothermal boundary layer, the first mesh from the wall plays a crucial role in the solution accuracy. From the standpoint of flux boundary conditions used in the CESE method, this is to be expected because the temperature gradient at the wall determines how much energy is being propagated into the computational domain. A smaller mesh near the wall, apparently, resolves the energy flux imposed at the wall and thus improves the accuracy of the entire thermal boundary layer. This observation is in line with other hypersonic surface heating investigations where designed order of accuracy can only be substantiated by non-uniform, adaptive grid refinement.[27]

B. Supersonic Jet-Shock Interaction Applied to Launch Vehicle Acoustics

As an example of the robustness and ease of application of the *ez4d* code and CESE method, we consider a set of simulations performed in late 2011 in support of the analysis of abort motor aeroacoustics on the Orion/MPCV vehicle. The abort-motor configuration for the Orion vehicle is a tractor type, meaning that the solid rocket abort motor is mounted ahead of the crew module (CM), and its exhaust plume envelops the CM boost-protection cover (BPC) surface, creating very high levels of surface fluctuating pressure (SFP). The internal vibration and acoustics created by this SFP must be accurately predicted in order to establish vibration qualification specifications for equipment which must survive an abort in order to function reliably during later stages of the abort and recovery of the CM. To this end, the Orion program conducted a series of wind tunnel tests and motor test firings, culminating in a complex transonic wind tunnel test in which the abort-motor plumes were simulated using helium jets exhausting from a model instrumented with 237 fluctuating-pressure sensors.[28] A shadowgraph from that test is shown in Fig. 5. The general conclusion from that test was that the plume-induced SFP was the greatest when the plume boundary was distinctly off the surface, e.g., when the plume was not impinging directly on the surface. In general, plume impingement increased with Mach number, when the momentum of the freestream flow deflected the plumes aft, and with altitude, as reduced freestream static pressure caused increased plume expansion. However, it was found that a region of the BPC just aft of a compression corner in the outer mold line (OML) saw significantly increased SFP at low supersonic Mach numbers, in contrast with the general Mach trend. One of the authors had earlier conjectured that interaction of the plume turbulence with the oblique shock created by the compression corner in the OML would cause such a local SFP increase, based on analysis by Ribner.[29] In order to assess the impact of such an effect, a series of simulations were quickly conducted. While a number of other CFD codes are certainly capable of performing such simulations, the robustness and accuracy of the *ez4d* code for time-dependent problems involving strong shocks and large-amplitude wave propagation gave confidence that the exploratory simulations could be conducted rapidly and with minimal difficulty.

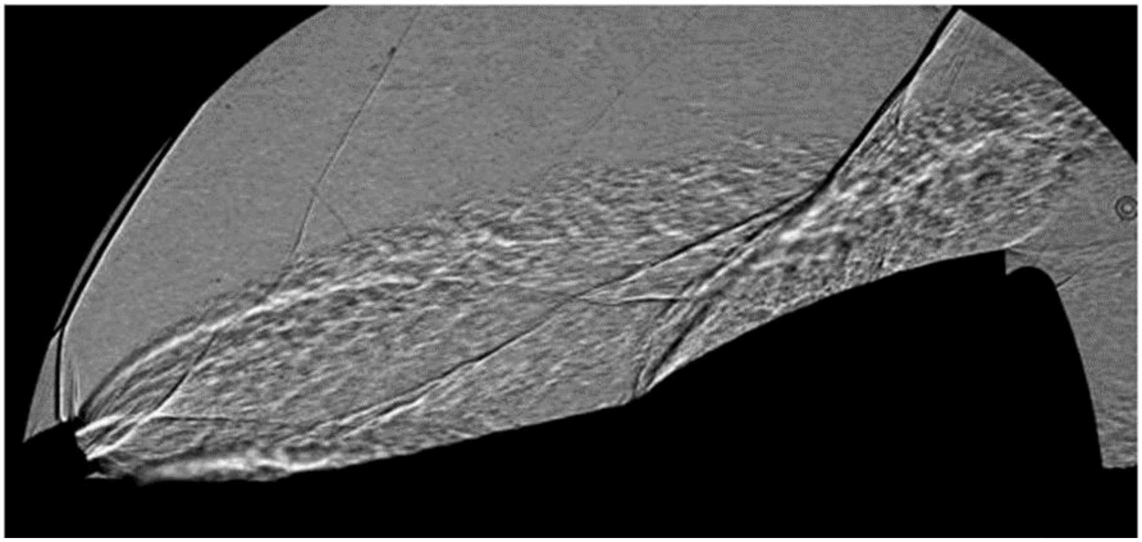


Figure 5. Shadowgraph from Orion hot-helium abort-motor noise wind-tunnel test, showing plume-oblique shock interaction.

The layout of the simulations conducted is shown in Fig. 6. The simulations were conducted using the inviscid Euler equations in two dimensions for very rapid turn-around using a workstation computer, and for more direct linkage with Ribner's theoretical analysis. [29] Once a relatively stationary state was reached using steady boundary conditions (Fig. 7), unsteady perturbations were imposed on the part of the inflow boundary corresponding to the jet; perturbations in normal velocity, temperature, and density were imposed in separate simulations to investigate the strength of effect of each mode. Density and temperature perturbations were defined to be isobaric and satisfied the equation of state, whereas normal-velocity perturbations maintained constant density and temperature. For example, imposition of a simple low-amplitude normal-velocity perturbation at a single frequency gave rise to strong fluctuations in the jet after interaction with the shock; see Fig. 8. Note the occurrence of multiple eddy-shocklets in the post-shock region, exiting cleanly from the simulation domain.

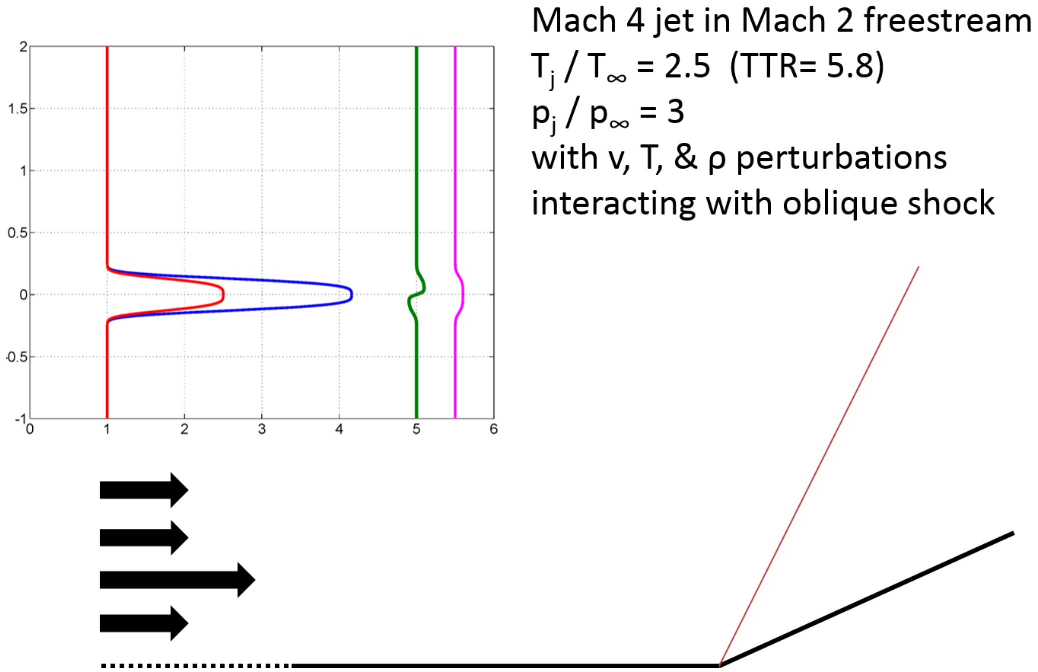

Mach 4 jet in Mach 2 freestream
$T_j / T_\infty = 2.5$ (TTR= 5.8)
$p_j / p_\infty = 3$
with v, T, & ρ perturbations
interacting with oblique shock
2
1.5
1
0.5
0
-0.5
-1
0
1
2
3
4
5
6

Figure 6. Schematic of jet-shock interaction simulations.

Next, simulations were conducted with random spectra of normal velocity, temperature, and density fluctuations imposed in the jet inflow. In order to ensure that the inflow spectrum was broadband rather than containing discrete frequencies, inflow perturbations were defined as a series of dozens of concurrent packets of random length and oscillation frequency pulled from a uniform distribution over predefined ranges. The relative strengths of the responses to normal-velocity and temperature inflow perturbations are illustrated in Fig. 9 in terms of the pressure fluctuations measured at four locations in the simulation. Since the temperature inflow perturbations were defined to be isobaric, the pressure fluctuations measured in the middle of the jet at inflow and just upstream of the shock are very small for that case in comparison to those from the normal-velocity perturbation case (green and magenta traces in Fig. 9), the SFP response is more than 2.5 times larger post shock (red and blue traces). This effect is also seen in the pressure power spectral density (PSD) plots in Fig. 10 (trace colors in Fig. 10 correspond to the same locations as in Fig. 9). Finally, in Fig. 11 pressure-fluctuation histograms for the two cases are compared. The histograms were derived from the pressure histories taken at the downstream-surface location (blue arrow in Fig. 9. Note that the high-level tail for the temperature-perturbation case is considerably fuller and longer than for the normal velocity-perturbation case.

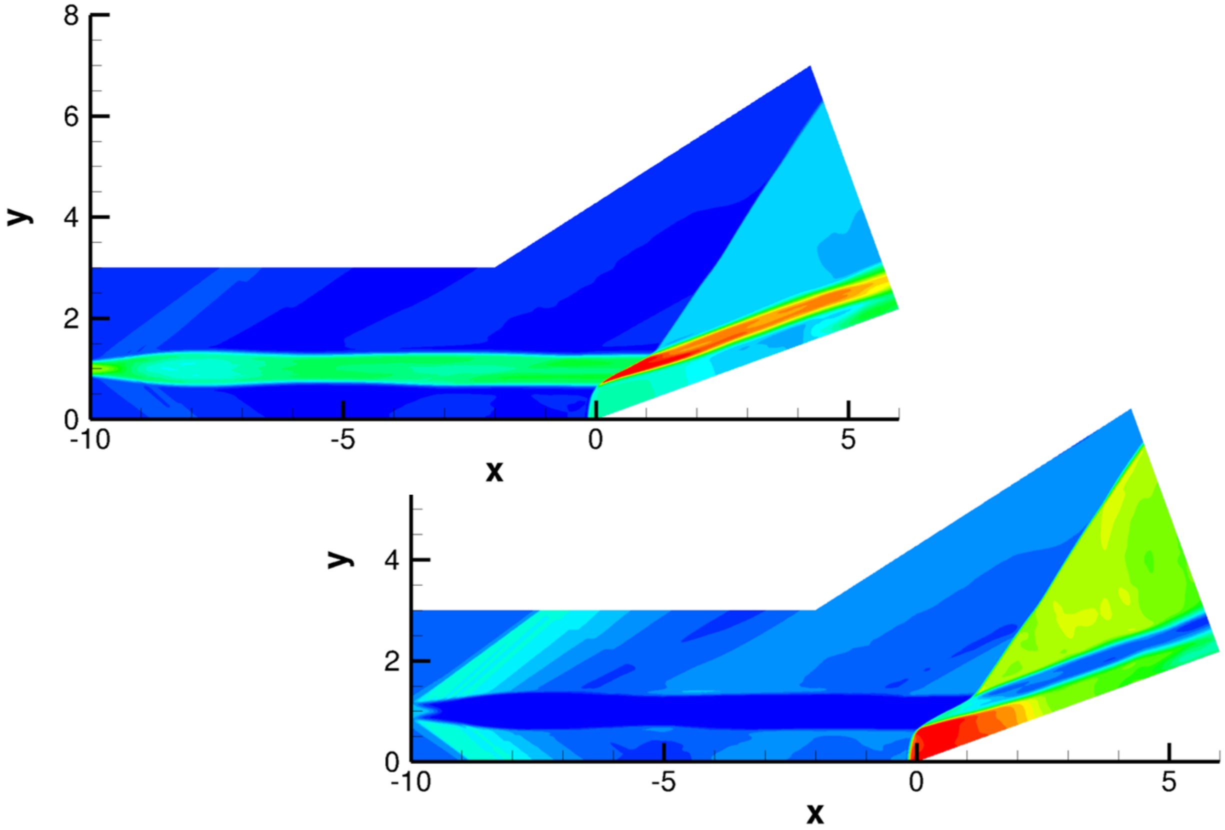


Figure 7. Mean temperature (top) and density (bottom) from jet-shock interaction simulations.

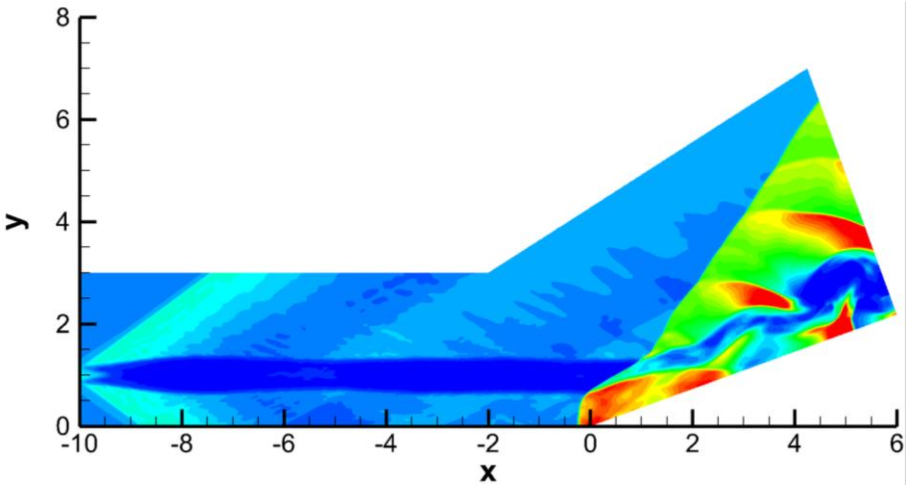


Figure 8. Density contours for simulation with single normal-velocity perturbation imposed at jet inflow.

The conclusions which was drawn from this rapidly-conducted exploratory investigation was that: (i) velocity fluctuations in the form of plume turbulence could indeed give rise to the SFP-augmentation effect, as observed for supersonic cases during the Orion helium-plume abort-motor noise test; and (ii) this effect would likely be significantly magnified in the actual situation of a solid rocket-motor plume interacting with an oblique shock, since the plume of a solid rocket motor contains modest levels of velocity fluctuations but very high levels of temperature fluctuations.

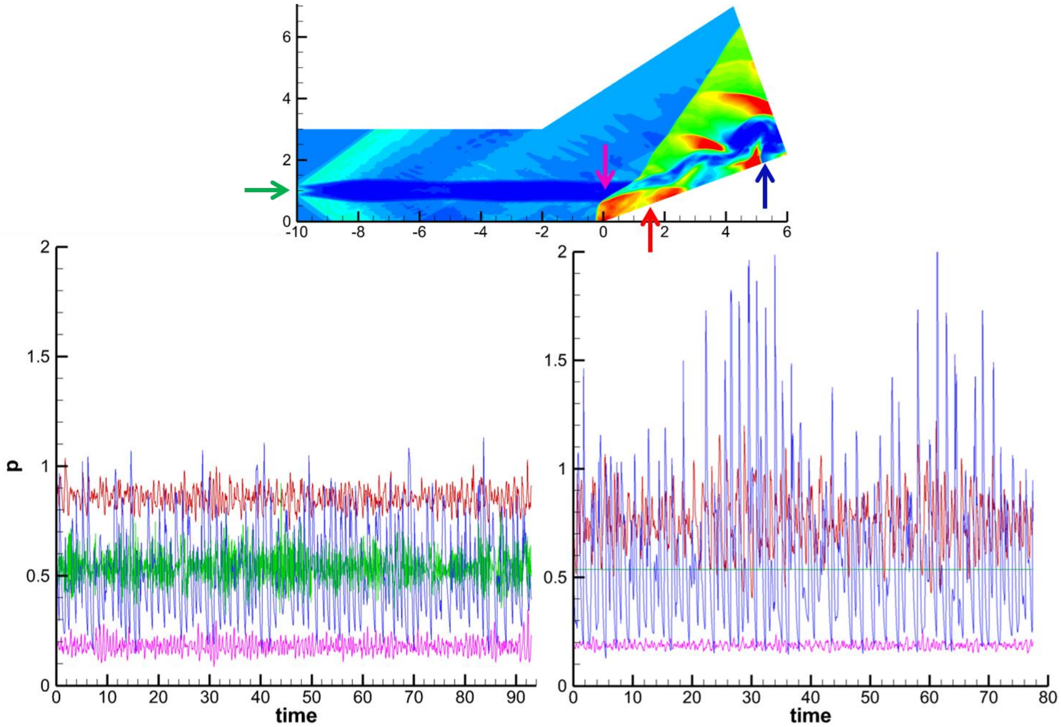


Figure 9. Pressure fluctuation histories, inflow perturbations of normal-velocity (left) and temperature (right).

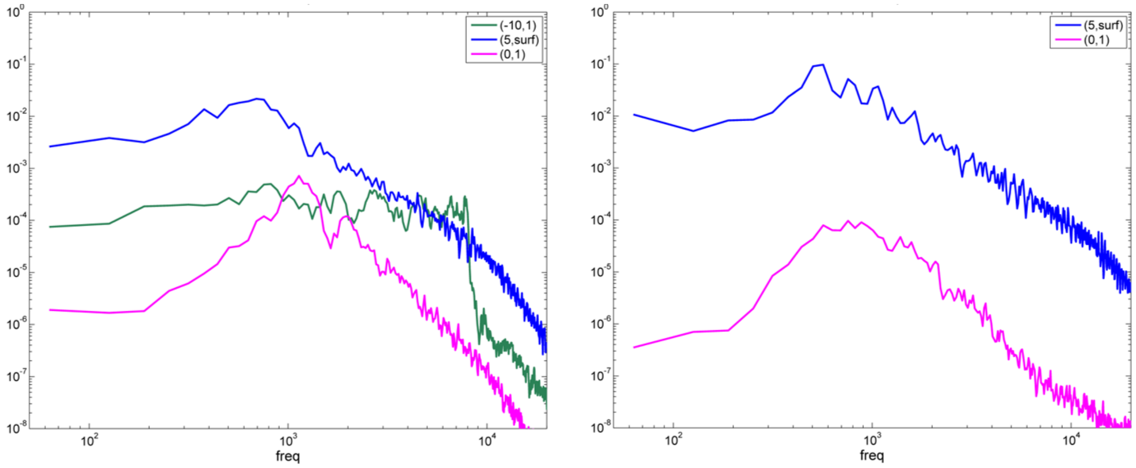


Figure 10. Pressure PSD's, inflow perturbations of normal velocity (left) and temperature (right).

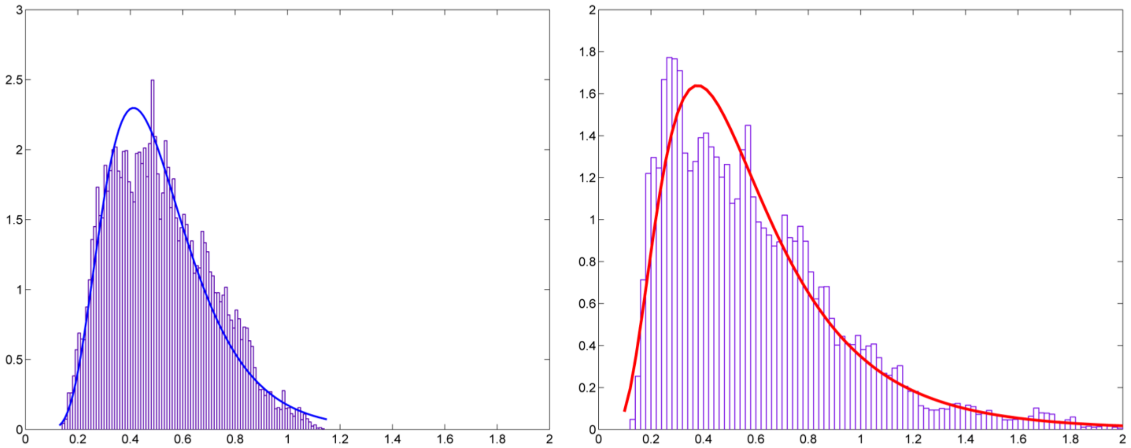


Figure 11. Pressure-fluctuation histograms, inflow perturbations of normal velocity (left) and temperature (right).

C. Canonical Isotropic Turbulence-Normal Shock Interaction

In this study the focus is on the most fundamental problem in shock/turbulence interaction, namely that of isotropic turbulence passing through a nominally normal shock. A cubic domain of length 4π in the streamwise direction and 2π in the transverse directions was used in the computations. The normal shock with M_1 = 1.5 was located at $x = \pi$. Unstructured meshes containing isotropic tetrahedrons generated using a commercial grid generation tool are used for the solutions presented in this section. Several meshes were tested, with all of them having no clustering of mesh points around the shock region. As part of the ongoing grid-sensitivity studies and to assess the actual grid resolution required by the 2nd order CESE scheme to capture the essential features of the flow, an initial set of computations was made with meshes containing ~ 6 Million tetrahedral cells (average edge length, Δs = 2π /64), ~9 Million cells (Δs = 2π /75), and ~33 Million cells (Δs = 2π /128), respectively. Periodic boundary conditions are used at the boundaries on the *y*- and *z*- planes. At the inflow plane, isotropic turbulent structures are superimposed using a procedure outlined in Ref. 30. Two important non-dimensional parameters characterizing the state of the flow are (i) turbulent Mach number, M_t, and (ii) Reynolds number based on the Taylor microscale, Re_λ. They are defined as follows:

$$M_t = \frac{\sqrt{\langle u_i u_i \rangle}}{\langle c \rangle}\ ;\ Re_\lambda = \frac{\langle \rho \rangle u_{rms} \lambda}{\langle \mu \rangle} \qquad (30)$$

where,

$$\boldsymbol{u}_{rms} = \sqrt{\frac{\langle u_i u_i \rangle}{3}}\ ;\ \lambda = \frac{\langle u_1^2 \rangle}{\langle (\partial u_1 / \partial x)^2 \rangle} \qquad (31)$$

In the above equations, the symbol $\langle\ \rangle$ stands for ensemble average and λ represents the Taylor microscale. The flow field is initialized by setting up a random velocity field, $u_{i,0}$, that is solenoidal and also satisfies a chosen energy spectrum and the initial turbulent Mach number:

$$E(k) \sim k^4 \exp(-2(k/k_0)^2);\ \frac{3\boldsymbol{u}_{rms,0}^2}{2} = \frac{\langle u_{i,0} u_{i,0} \rangle}{2} = \int_0^{\propto} E(k)dk \qquad (32)$$

where, k stands for the wavenumber magnitude. k_0, the most energetic wavenumber, is taken as 4. For the chosen energy spectrum, the initial Taylor length scale, λ_0, is $2/k_0$. The initial density and pressure fields are assumed to be constant and the remaining parameters are set based on $M_{t,0} = 0.22$ and $Re_{\lambda,0} = 19.9$. The approach of imposing inflow conditions, is adopted from earlier studies[31–33] that specify the method of generating inflow turbulence for studying spatially evolving turbulence and its interaction with a shock wave. Randomization of the phase factor (used to prescribe the energy spectrum) in time, which is needed to ensure that the signal generated at the inflow is not periodic,[33] has not been done in this study. At the outflow boundary, subsonic conditions with a Mach number of M_2 = 0.7011 are used without a buffer domain. In fact, all the computations performed in this study, unlike previous studies,[31,32,34] did not need any special treatment of the outflow boundary condition (buffer domain or sponge region to damp any acoustic reflection) or clustering of the grid points around the shock to ensure that the shock was stationary and did not convect upstream, because of the features of the CESE method.

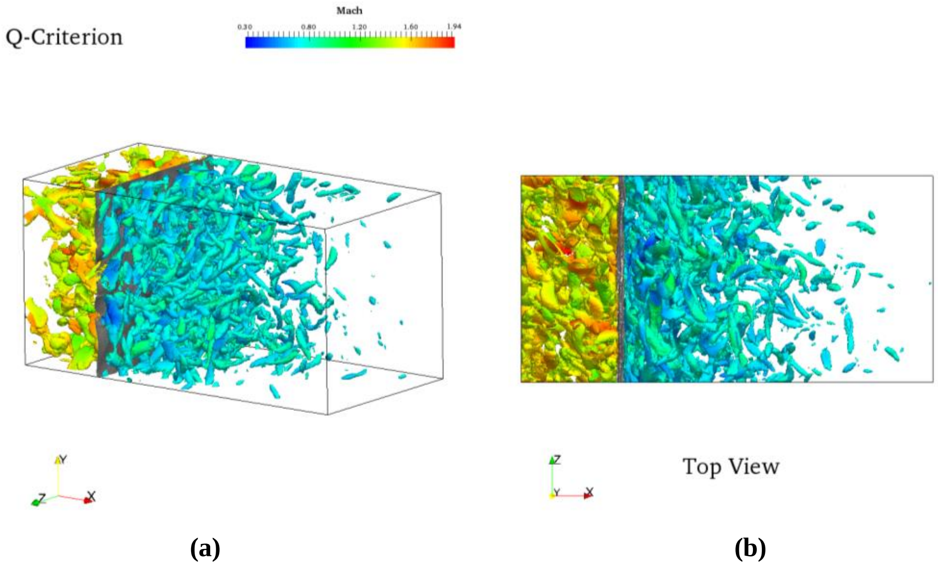
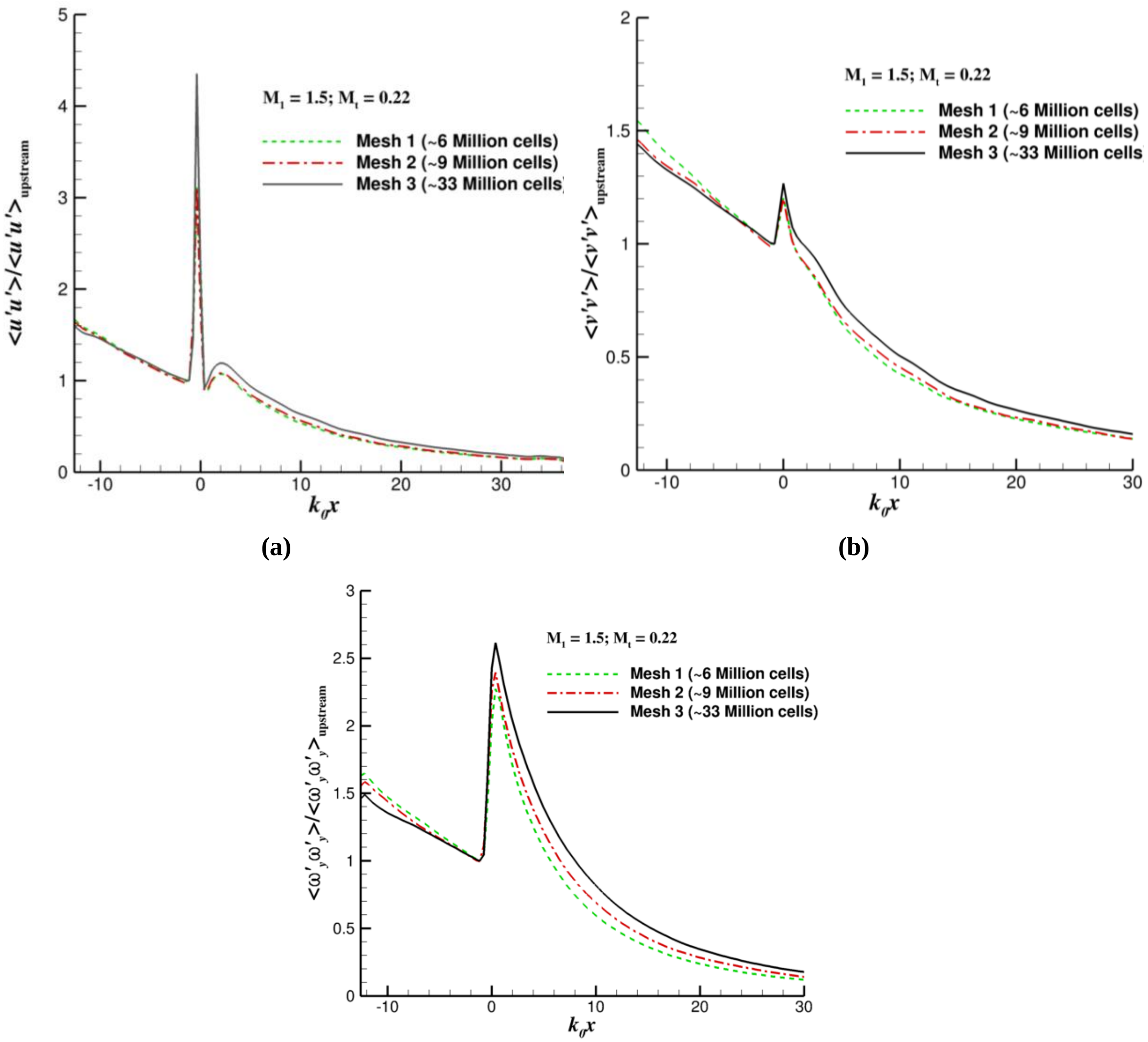


Figure 12. Solutions for interaction of isotropic turbulence (*M*$_t$ = 0.22) with a normal shock (*M*$_1$=1.5), showing isosurfaces of Q-criterion shaded with the local Mach number. Shock surface is represented by isosurface of dilatation. (a) Side view; (b) top view (*x*-*y* plane).

The instantaneous solution after approximately four flow-through times is shown in Fig.12. The Q-criterion isosurfaces have been color shaded by the Mach numbers, showing pre- and post-shock regions with supersonic and subsonic colors, respectively. Isosurfaces of dilatation are used in these plots to help visualize the instantaneous shock surface in the middle of the domain. The isotropic nature of the incoming turbulence is indicated by the random orientation of the vortex cores upstream of the shock surface. The shock compresses the turbulence in the streamwise direction, distorts the vortices, and in the meantime, makes them predominantly align in *y*-*z* plane (see the top view, Fig. 12). For the normal shock conditions studied here, the level of turbulence in the inflow causes a weak distortion of shock surface, putting it under the category of "wrinkled shock regime" (Ref. 34).

The streamwise and transverse components of the Reynolds stresses, along with the transverse vorticity variances (normalized with their values just upstream of the shock) are shown in Figs. 13(a)–(c). In all of the above-mentioned plots, the streamwise coordinate has been non-dimensionalize,d using the wavenumber of peak energy k_0, for facilitating comparison with linear theory[35] when possible. Furthermore the non-dimensionalized streamwise coordinates have been translated such that the shock is located at x = 0. Averaging was performed over transverse direction and time using data obtained over four flow through-times. From Figs. 13(a)–(c), it is evident that the grid resolution for the meshes utilized in the study are still not sufficient for achieving truly grid converged solutions. However, they seem to display the correct trend based on the qualitative behavior of the results.



(a)

(b)

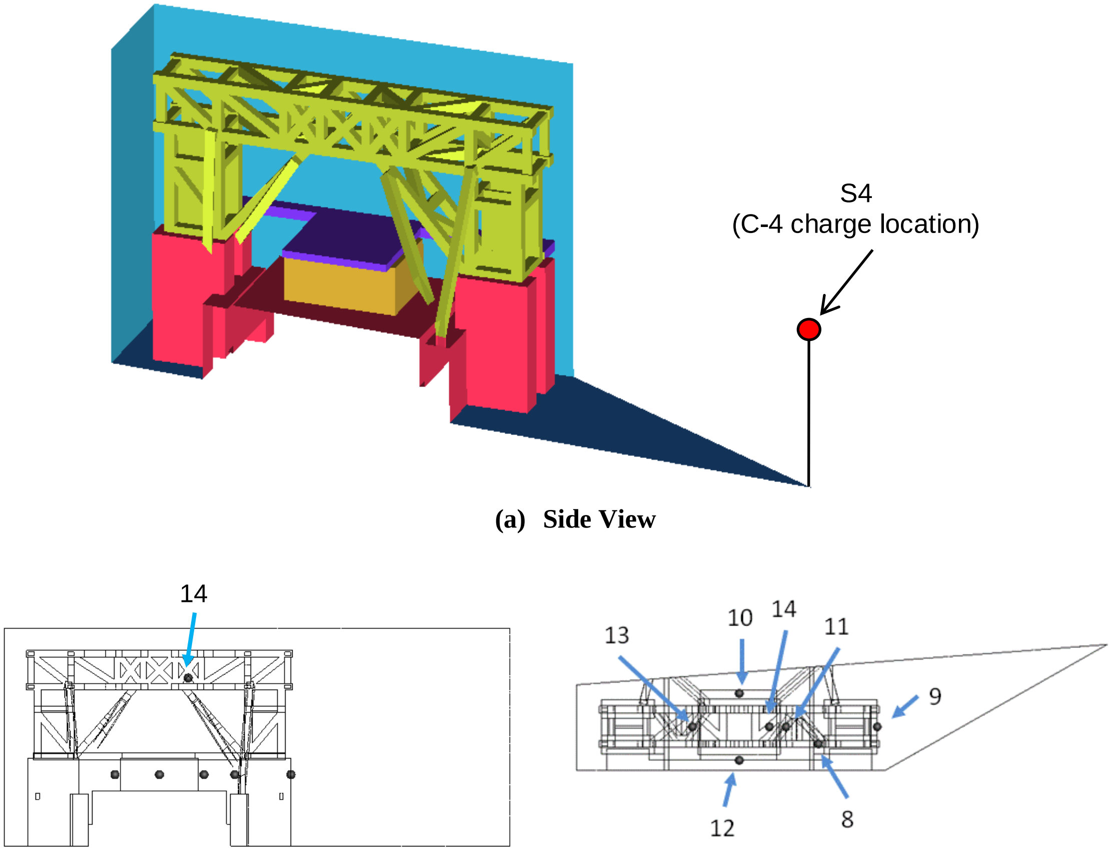
(c)

Figure 13. Evolution of Reynolds stresses and vorticity variances obtained from the computations of normal shock-turbulence interaction, using meshes with different resolutions: (a) Streamwise Reynolds stress component, (b) transverse Reynolds stress component, and (c) transverse vorticity variance.

Both streamwise (Fig.13(a)) and transverse (Fig.13(b)) Reynolds stresses increase at the shock. The transverse stress decays almost monotonically behind the shock, while there is a rapid increase of the streamwise component behind the shock before it begins to decay slowly. The level of fluctuation in the streamwise component just behind the shock is in general larger than the transverse component because of the contribution from the acoustic waves that are generated when the vortical waves interact with the shock. The post-shock increase of the streamwise component of the Reynolds stress is not predicted by the linear theory and can only be explained by nonlinear effects.[31,32,34] The behavior of the Reynolds stresses observed in this study has also been well-investigated in the literature.[31,32,34] From Fig. 13(c), the amplification of the transverse component of the vorticity variance directly behind the shock is evident. Thereafter it begins to decay. The amplification factor of the transverse vorticity is close to that predicted by linear theory[35] (~3.1, see Fig. 5(b) of Ref. 32). It is expected that with increased mesh resolution it should grow and converge to the correct value. Towards this end, a finer grid in the vicinity of the post-shock region, as opposed to the uniform grid spacing that was used in this study, is expected to be advantageous in resolving the dissipative scales in the post-shock turbulence. With use of unstructured tetrahedral elements, this could bring about a substantial saving in the overall grid count for DNS studies, as it is free of any topological constraints when mesh adaptation is required. These issues will be investigated over the next few months and reported in a future work.

D. Blast Wave Propagation in Test Facilities and Parallel Scaling Study

In this section, we assess the scalability of the *ez4d* software framework for large-scale computations. Both strong and weak scaling are investigated. The test problem considered for this study is the numerical simulation of the blast/acoustic wave propagation in NASA Stennis Space Center's E-1 test facility. The work was carried out in support of NASA Engineering and Safety Center's (NESC) independent assessment of the test facility. The E-1 test facility was originally developed as a facility with three test cells for testing rocket components by multiple users simultaneously. With time, the facility grew into a larger one, allowing for testing of engine systems, large thrusters, etc., in various orientations. These changes interfered with the original design and negated many of the design protections that were in place to protect the test cells from blast waves or acoustic disturbances. With demand from the commercial crew/cargo companies for use of the facility towards engine development and the needs of NASA's own Space Launch System (SLS) engine-component development, the E-1 facility's capability to simultaneously accommodate multiple tenants becomes critical. The NESC's assessment study was aimed at identifying tools (analytical/empirical) that can characterize the blast/acoustic environment arising at certain locations surrounding the test-cell areas, determining potential mitigation methods that would be effective in making the environment suitable for clean tests, and minimizing the influence from blast/acoustic effects.



(a) Side View

(b) Probe Locations

Figure 14. Geometry of a portion of the E-1 test facility utilized in the S-4 configuration, along with the location of the probes.

To facilitate the assessment of the computational tools and to determine the optimum grid resolution needed to model such problems, a series of experimental tests were conducted in the E-1 test facility using C-4 charges to create small blasts. However, given the size of the E-1 test facility (125' × 106' × 57.5') for CFD simulations, even a modest grid resolution of 2 inches results in about 600 million tetrahedral cells by conservative estimates. Given the project schedule and limitations on computational resources, it was decided that a smaller domain would be initially investigated to obtain an understanding of the mesh resolution requirements to achieve the desired numerical accuracy, as well as to identify the proper numerical approach by which the initial blasts can be modeled or represented in the CFD computations. As a result, additional experiments (hereafter designated as the S-4 configuration) were designed and carried out using a smaller C-4 charge (0.27 lbm) focusing on a critical region around test cell No. 3. The domain of interest is shown in Fig. 14(a). The C-4 charge was placed at a height of 510 inches above the ground near the corner of the domain (see Fig. 14(a)), coinciding with the height at which the test cell is also located. Overpressure signal data around the test cell of interest were obtained using pressure probes at various locations (see Fig.14 (b)).

Generating the geometry of the S-4 configuration with appropriate simplifications, to ease the mesh generation process, was one of the challenging tasks of the project and the main focus of the initial efforts of NESC. Using the geometry provided by the NESC, the mesh was generated using the commercial software Pointwise®. A good quality initial mesh, comprised of tetrahedrons with approximately 2-inches resolution throughout the domain, was obtained. The overall grid count was about 59 million tetrahedral cells and this mesh will be henceforth referred to as Mesh 1. Subsequently, two additional meshes with finer resolutions were generated to see further improvements in computational results. These contained approximately 81 million tetrahedrons (Mesh 2) and 177 million tetrahedrons (Mesh 3) respectively. For the computation, all physical surfaces (ground, trusses, etc.) were modeled as inviscid slip walls. Remaining surfaces were treated as non-reflecting boundaries. Time-accurate inviscid computations were carried out using the second-order accurate (in both space and time) Euler solver in *ez4d* along with a perfect-gas assumption where the medium is modeled as air. To avoid the issues of modeling the blast itself, the blast was modeled as a pressure source with its initial conditions being provided from a 1-D spherical solution of the blast at different time slices, which was obtained by one of the NESC members using Sandia National Laboratories' Eulerian shock physics code CTH (CHART squared to the three halves).[36]

The pressure-gradient contours obtained from the computation are shown at the plane of y = 510 inch for three different time instances in Fig. 15. As seen from this figure, the blast wave interacts with the various surfaces as it propagates through the domain, resulting in several reflections and causing further distortion of the primary blast wave. Some of these reflected waves linger around the test-cell area for a long time and represent areas of concern, when actual engine tests are carried out. For further quantitative evaluation, the overpressure signatures at two different probes are shown in Fig. 16, along with their comparison against experimental data. As can be seen from Fig. 16(a), data from probe OP9 (that is in the line-of-sight of the blast wave) shows that the computations capture the timing of pressure peaks in the signals quiet well, although the magnitudes are under-predicted. However, at the probe location OP14 (Fig. 16(b)), the solution doesn't identify the secondary peak, which follows the leading peak, as the dominant peak very well. A similar trend was observed at a few other probe locations. The reason for this discrepancy can be attributed to the fact that, by modeling the initial blast wave as an initial condition, only the initial blast wave is permitted into the solution and thereby, the computation lacks information about the C-4 explosion products that trail the initial blast wave. The use of air as the medium also has a role in the discrepancy, as it is incorrect to approximate C-4 blast products as just air.

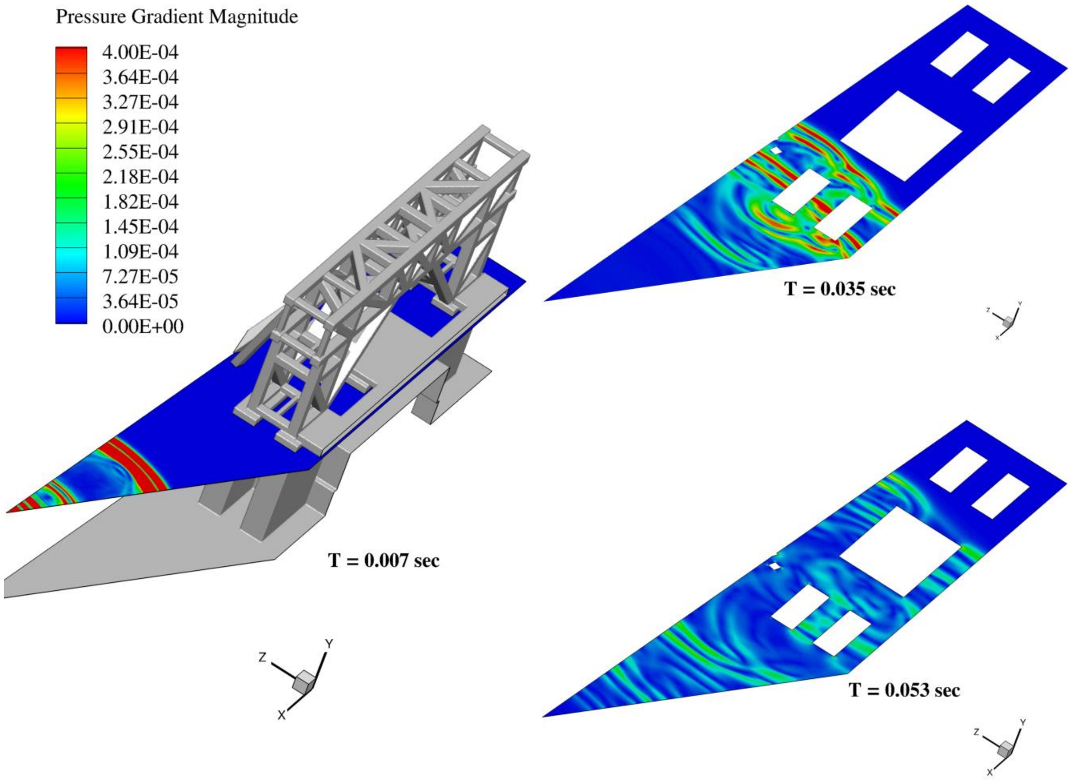
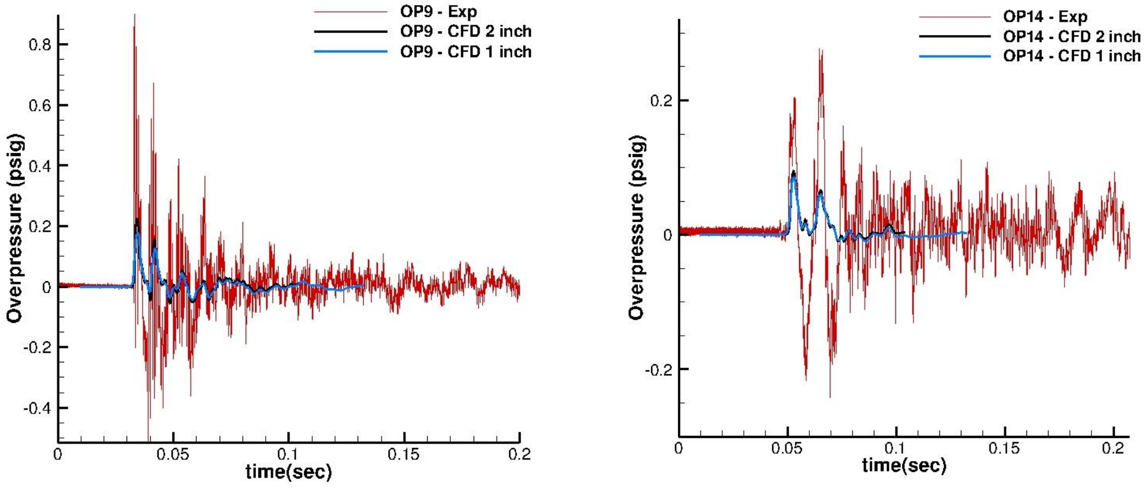


Figure 15. Snapshots of pressure gradient at two different time instances on the *y* = 510 inch plane.



(a) Probe OP9

(b) OP14

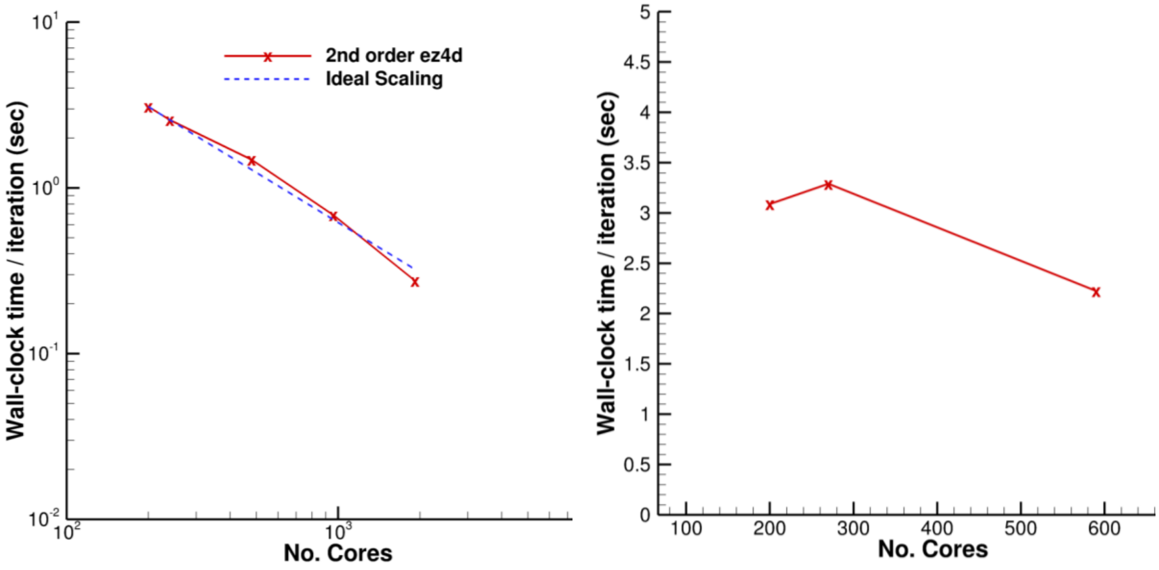
Figure 16. Comparison of the time-history of overpressure signature (CFD) against experimental data for representative probes.

For the strong scalability performance study, the Mesh 1 was utilized, whereas all the three different meshes were utilized in the weak scalability study. The performance study was carried out on NASA's Pleiades cluster using its Ivy Bridge (E5-2680v2) Intel Xeon processors. The Ivy Bridge nodes contain 2 ten-core processors per node with approximately 64 GB memory per node and has a clock speed of 2.6GHz. Towards the strong scalability study, the Mesh 1 was partitioned into various number of blocks with the number of tetrahedral elements per block varying from ~290,000 elements to ~30,000 elements, as indicated in Table 3. For this particular mesh, close to or better than linear scaling was achieved up to 1920 cores, as seen from Table 3 and Fig 17. The total time (wall-clock) to solution when utilizing 1920 cores was a little over two hours, making *ez4d* very efficient for studying such problems. However, as the partition size reduces to less than 10,000 elements per block, the performance is expected to reduce based on previous experience, as communication overheads would start to take precedence over computational efforts.

Table 3: Details of the runs performed for the strong scaling study on Pleiades.

No. of Elements / core	No.of Cores	Wall Clock time / iteration (sec)	Speed-up
290,000	200	3.09	
240,000	240	2.57	1.2
120,000	480	1.48	2.08
60,000	960	0.685	4.51
30,000	1920	0.275	11.23

For the weak scalability study, the number of elements per partition was kept at a constant (~290,000 elements /core), while the problem size was scaled from 59 million elements to 177 million elements. The performance remains close to linear between 200 and 270 cores, but gains efficiency when 590 cores are used. More data points need to be collected to obtain a thorough characterization of the weak scalability. This will be done by running even larger jobs that require many thousands of cores. In this regard, we are expecting to run a simulation using a mesh with ~1.5 billion tetrahedrons, which corresponds to the full E-1 test facility, in the coming months. That particular run will require at least 5000 cores and will also help improve our study of the strong scalability performance.



(a) Strong scalabilty study results

(b) Weak scalability study results

Figure 17. Scaling results for the 2nd order *ez4d* code on Pleiades cluster.

IV. Summary

After decades of computational fluid dynamics research and development, it is generally agreed that future developments should focus on improving solution accuracy, robustness and performance, particularly for time-accurate, large-scale computations with unstructured meshes. Along this line of development, this paper discusses the current status of the NASA in-house space-time CESE numerical and software framework. Continuing order-of-accuracy studies along with several numerical computations of practical engineering or canonical problems are reported. Time accurate simulations using the developed software show promising results in large-scale parallel computing environments. High solution fidelity obtained for flows with small turbulent structures and strong shocks using unstructured meshes helps establish confidence for more complex-geometry turbulent flow simulations in the future. From the numerical stand point, the integral form of conservation equations solved in the CESE numerical framework is shown to provide several nice properties in reducing the need for extensive buffered domain or grid-adaptation around shocks. The consistent numerical formulations closely mimic the physics and allow accurate computations of waves and shocks simultaneously. A potentially high-degree of savings in computational time could come from the smaller domain free of buffer zones, unstructured mesh free of topological constraints, and capabilities of time-accurate local time-stepping as demonstrated in the hot-spot/shock and shock/turbulence interaction problems discussed in this and previous papers. The CESE numerical and software framework under development is a viable tool for next-generation high-fidelity simulations of general conservation laws, and in particular, Navier-Stokes equations.

Acknowledgments

The research funding for this paper has been provided by Revolutionary Computational Aerosciences (RCA) subproject under the Transformational Tools and Technologies (TTT) project of NASA Transformative Aeronautics Concepts Program (TACP). The first author's research is funded through National Institute of Aerospace (NIA) under the cooperative agreement 2A00.

References

[1] Slotnick, J., Khodahoust, A., Alonso, J, Darmofal, D., Gropp, W., and Mavriplis, D., "CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences," NASA/CR-2014-218178, 2014.

[2] Larsson, J., and Wang, Q., "The Prospect of Using LES and DES in Engineering Design, and the Research Required to Get There," *Phil. Trans. R. Soc A*, Vol. 372 (2022), August, 2014. DOI: 10.1098/rsta.2013.0329

[3] Chang, S.-C., Wu, Y., Yang, V., and Wang, X. Y., "Local Time-Stepping Procedures for the Space-Time Conservation Element and Solution Element Method," *Inter. J. of Comput. Fluid Dyn.,* Vol. 19, No. 5, pp. 359–380, 2005.

[4] Yen, J. C., "Demonstration of a Multi-Dimensional Time-Accurate Local Time Stepping CESE Method," AIAA Paper 2011-2755, 2011.

[5] Chang, C.-L., Venkatachari, B., and Cheng, G, "Time-Accurate Local Time Stepping and High-Order Space-Time CESE Methods for Multi-Dimensional Flows with Unstructured Meshes," AIAA Paper 2013-3069, 2013.

[6] Chang, S.-C., "The Method of Space-Time Conservation Element and Solution Element–A New Approach for Solving the Navier-Stokes and Euler Equations," *J. Comput. Physics*, Vol. 119, pp. 295–324, 1995.

[7] Chang, S.-C., "A New Approach for Constructing Highly Stable High Order CESE Schemes," AIAA-2010-543, 2010.

[8] Chang, C.-L., "Three-Dimensional Navier-Stokes Calculations Using the Modified Space-Time CESE Method," AIAA Paper 2007-5818, 2007.

[9] Bermejo-Moreno, I., Bodart, J., Larson, J., Barney, B. M., Nichols, J. W., and Jones, S., "Solving The Compressible Navier-Stokes Equations on up to 1.97 Million Cores and 4.1 Trillion Grid Points," in *Proceedings of the 2013 ACM/IEEE Conference on Supercomputing (SC13)*, 2013.

[10] Chang, C.-L., Choudhari, M. M., and Li, F., "Numerical Computations of Hypersonic Boundary-Layer over Surface Irregularities," AIAA Paper 2010-1572, 2010.

[11] Chang, C.-L., Choudhari, M., Li, F., and Venkatachari, B. S., "Effects of Cavities and Protuberances on Transition over Hypersonic Vehicles," AIAA Paper 2011-3245, 2011.

[12] Wang, X. Y., Chang, S.-C., "A 2D Non-Splitting Unstructured Triangular Mesh Euler Solver Based on the Space-Time Conservation Element and Solution Element Method," *Comp. Fluid Dynamics Journal*, Vol. 8, No. 2, pp. 309–325, 1999.

[13] Loh, C. Y., Chang, S.-C., and Scott, J. R., "Computational Aeroacoustics via the Space-Time Conservation Element/ Solution Element Method," AIAA Paper 96-1687, 1996.

[14] Loh, C. Y., 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.

[15] Chang, S.-C., Himansu, A., Loh, C. Y., Wang, X.-Y, and Yu, S.-T., "Robust and Simple Non-Reflecting Boundary Conditions for the Euler Equations—A New Approach Based on the Space-Time CE/SE Method," NASA/TM-2003-

212495, 2003.

[16] Roe, P.L., "Introduction to Numerical Methods Suitable for the Euler Equations," von Karman Institute for Fluid Dynamics Lecture Series: Introduction to Computational Fluid Dynamics, 1983.

[17] Tonti, E., "Why Starting from Differential Equations for Computational Physics?," *Journal of Computational Physics*, Vol.257, 2014, pp. 126090.

[18] Mattiussi, C., "The geometry of time-stepping," *Progress In Electromagnetics Research*, Vol. 32, 2001, pp.123-149

[19] "Turbulence Modeling Resource," NASA Langley Research Center, URL: http://turbmodels.larc.nasa.gov/

[20] Bilyeu, D. L., "A Higher-Order Conservation Element Solution Element Method for Solving Hyperbolic Differential Equations on Unstructured Meshes," PhD thesis, The Ohio State University, 2013.

[21] Bilyeu, D. L., Yu, S.-T. J., Chen, Y.-Y., and Cambier, J.-L., "A Two-Dimensional Fourth-Order Unstructured – Meshed Euler Solver Based on the CESE Method," *J. Comput. Phys.*, Vol. 257, pp. 981–999, 2014.

[22] Chang, C.-L., Venkatachari, B. S., and Cheng, G. C., "Tetrahedral-Mesh Simulation of Turbulent Flows with the Space-Time Conservative Schemes," AIAA Paper 2015-3084, 2015.

[23] Chiocchia, G. "Exact solutions to transonic and supersonic flows," Technical Report AR-211, AGARD, 1985.

[24] "TauBench - IPACS Benchmarking Code from DLR Germany," URL: http://www.ipacsbenchmark.org/index.php?s=download&unterseite=taubench [cited 1 Nov 2015]

[25] "2nd International Workshop on High-Order CFD Methods," URL: http://www.dlr.de/as/desktopdefault.aspx/tabid-8170/13999_read-35550/ [cited 1 Nov 2015].

[26] Yee, H. C., Sandham, N. D., and Djomehri, M. J., "Low-dissipative High-order Shock-capturing Methods Using Characteristic-based Filters," *Journal of Computational Physics*, Vol. 150, 1999, pp. 199-238.

[27] Burgess, N. K., "An Adaptive Discontinuous Galerkin Solver for Aerodynamic Flows," Ph.D. Thesis, University of Wyoming, Nov 2011.

[28] Panda, J., James, G. H., Burnside, N. J., Fong, R. K., Fogt, V. A., and Ross, J. C., "Use of Heated Helium to Measure Surface Pressure Fluctuations on the Launch Abort Vehicle During Abort Motor Firing," AIAA Paper 2011-2901.

[29] Ribner, H. S., "Shock-Turbulence Interaction and the Generation of Noise," NACA TN-3255, 1954.

[30] Lee, S., Lele, S. K., and Moin, P., "Eddy Shocklets in Decaying Compressible Turbulence," *Phys. Fluids A*, Vol. 3, pp. 657–664, 1991.

[31] Lee, S., Lele, S. K., and Moin, P., "Direct Numerical Simulation of Isotropic Turbulence Interacting with a Weak Shock Wave," *J. Fluid Mech.*, Vol. 251, pp. 533–562, 1993.

[32] Lee, S., Lele, S. K., and Moin, P., "Interaction of Isotropic Turbulence Interacting with Shock Waves: Effect of Shock Strength," *J. Fluid Mech.*, Vol. 340, pp. 225–247, 1997.

[33] Lee, S., Lele, S. K., and Moin, P., "Simulation of Spatially Evolving Turbulence and the Applicability of Taylor's Hypothesis in Compressible Flow," *Phys. Fluids A*, Vol. 4, pp. 1521–1530, 1992.

[34] Larsson, J., and Lele, S. K., "Direct Numerical Simulation of Canonical Shock/Turbulence Interaction," *Phys. Fluids*, Vol. 21, 126101, 2009.

[35] Anyiwo, J. C., and Bishnell, D. M., "Turbulence Amplification in Shock-Wave Boundary Layer Interaction," *AIAA J.*, Vol. 20, No. 7, pp.893–899, 1982.

[36] McGlaun, J., Thompson, S., and Elrick, M., "CTH: A Three-Dimensional Shock-Wave Physics Code," *International Journal of Impact Engineering*, Vol. 10, pp. 351-360, 1990.