High-Order Stabilized Finite Elements on Dynamic Meshes

W. Kyle Anderson*

NASA Langley Research Center, Hampton, VA 23681, USA

James C. Newman III †

University of Tennessee at Chattanooga, Chattanooga, TN 37419, USA

The development of dynamic mesh capability for turbulent flow simulations using the Streamlined Upwind Petrov-Galerkin (SUPG) discretization is described. The current work extends previous research to include high-order spatial accuracy, including the satisfaction of the discrete geometric conservation law (GCL) on curved elements. Two closely-related schemes are described and the ability of these schemes to satisfy the GCL, while also maintaining temporal accuracy and conservation is assessed. Studies indicate that although one scheme discretizes the time derivative in conservative form, both schemes exhibit temporal conservation errors that decrease according to the expected design order of accuracy. The source of the temporal conservation errors is examined, and it is demonstrated that many finite-volume and finite-element schemes can also be expected to have difficulty strictly satisfying conservation in time. The effects on conservation are examined and, while present in the simulations, are seen to be negligible for the problems considered.

I.   Introduction

In Ref. 1, computational results demonstrate that a stabilized finite-element discretization can provide notable accuracy benefits over a commonly used finite-volume discretization, especially on triangular and tetrahedral meshes. While numerous research results have been reported for time-dependent applications using stabilized finite elements,[2–6] results with high-order spatial discretizations on dynamically evolving meshes have not been demonstrated for Streamline-Upwind-Petrov-Galerkin (SUPG) discretizations. In Ref. 7, the current authors presented results for high-order time discretization for viscous, turbulent flows, using the same stabilized finite-element methodology presented in Refs. 1, 8–10. Here, several blended-multistep methods, including the Modified-Extended BDF (MEBDF),[11,12] Two Implicit Advanced Step Point (TIAS),[13] and Singly Diagonally Implicit Runge-Kutta (SDIRK)[14] schemes have all been extended to accomodate for mass matrices that depend on the flow variables, which occur in stabilized finite-element schemes. In that reference, temporal accuracy up to fourth order has been verified using the Method of Manufactured Solution (MMS)[15] and has been demonstrated for time-accurate flows on moving domains. However, the mesh movement in Ref. 7 is limited to rigid-body rotation and, therefore, does not address the Geometric Conservation Law (GCL), which is necessary when the cell volumes are deforming.

The addition of dynamic mesh capability is a natural extension in the evolving development of this technology for simulating flows at high Reynolds numbers. As with other formulations, a desirable feature of the discretization is that freestream flow can be maintained, regardless of whether the mesh is moving or stationary. Although stabilized finite-elements have been used for simulations on dynamic meshes that satisfy the discrete geometric conservation law for algorithms not based on space-time discretizations,[16] these simulations have been restricted to linear elements with linear deformations assumed between time steps. Because the long-range plans for this technology include spatially adaptive meshes, as well as adaptation to locally modify the spatial order of accuracy, a missing component for moving forward is the ability to satisfy the discrete geometric conservation law on high-order elements with likely curving surfaces.

---

*Senior Research Scientist, Computational AeroSciences Branch, AIAA Associate Fellow.
†Professor, Mechanical Engineering Department, AIAA Associate Fellow.

American Institute of Aeronautics and Astronautics

The purpose of the current work is to continue the development of stabilized finite elements for high Reynolds number aerodynamic flows by extending the applicability presented in Ref. 1 and Ref. 7 to include dynamically deforming meshes. Although high-order temporal accuracy has been previously developed and demonstrated in Ref. 7, in the present work, only second-order temporal differencing is considered for simplicity. The emphasis here is on developing methodologies that satisfy the geometric conservation with high-order spatial discretizations and curved meshes, while also maintaining second-order temporal accuracy.

II. Governing Equations

The governing equations are the compressible, Reynolds-Averaged Navier-Stokes equations augmented with the one-equation Spalart-Allmaras turbulence model[17] that has been modified from the original model[18] to allow for negative values of the turbulence model working variable and will subsequently be denoted as the negative SA turbulence model. The equations can be expressed in the following conservative form:

$$\frac{\partial \mathbf{Q}(\mathbf{x},t)}{\partial t} + \nabla \cdot (\mathbf{F}_e(\mathbf{Q}) - \mathbf{F}_v(\mathbf{Q}, \nabla \mathbf{Q})) = \mathbf{S}(\mathbf{Q}, \nabla \mathbf{Q}) \quad \text{in} \ \ \Omega \tag{1}$$

where Ω is a bounded domain. The vector of conservative flow variables $\mathbf{Q}$, the inviscid and viscous Cartesian flux vectors, $\mathbf{F}_e$ and $\mathbf{F}_v$, are defined by:

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \\ \rho \tilde{\nu} \end{bmatrix}, \ \ \mathbf{F}_e^x = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u w \\ (\rho E + p)u \\ \rho u \tilde{\nu} \end{bmatrix}, \ \ \mathbf{F}_e^y = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ \rho v w \\ (\rho E + p)v \\ \rho v \tilde{\nu} \end{bmatrix}, \ \ \mathbf{F}_e^z = \begin{bmatrix} \rho w \\ \rho u w \\ \rho v w \\ \rho w^2 + p \\ (\rho E + p)w \\ \rho w \tilde{\nu} \end{bmatrix}, \tag{2}$$

$$\mathbf{F}_v^x = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} + \kappa \frac{\partial T}{\partial x} \\ \frac{1}{\sigma}\rho(\nu + \tilde{\nu} f_n)\frac{\partial \tilde{\nu}}{\partial x} \end{bmatrix}, \ \ \mathbf{F}_v^y = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{yz} \\ u\tau_{xy} + v\tau_{yy} + w\tau_{yz} + \kappa \frac{\partial T}{\partial y} \\ \frac{1}{\sigma}\rho(\nu + \tilde{\nu} f_n)\frac{\partial \tilde{\nu}}{\partial y} \end{bmatrix},$$

$$\mathbf{F}_v^z = \begin{bmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\tau_{xz} + v\tau_{yz} + w\tau_{zz} + \kappa \frac{\partial T}{\partial z} \\ \frac{1}{\sigma}\rho(\nu + \tilde{\nu} f_n)\frac{\partial \tilde{\nu}}{\partial z} \end{bmatrix}. \tag{3}$$

Here, ρ, p, and E denote the fluid density, pressure, and specific total energy per unit mass, respectively, $\mathbf{u} = (u, v, w)$ represents the Cartesian velocity vector, and $\tilde{\nu}$ represents the turbulence working variable in the negative SA model. The pressure p is determined by the equation of state for an ideal gas,

$$p = (\gamma - 1)\left(\rho E - \frac{1}{2}\rho(u^2 + v^2 + w^2)\right) \tag{4}$$

where γ is the ratio of specific heats, which is 1.4 for air. The subscripts on τ represent the components of the viscous stress tensor, which is defined for a Newtonian fluid as,

$$\tau_{ij} = (\mu + \mu_T)\left(\frac{\partial \mathbf{u}_i}{\partial \mathbf{x}_j} + \frac{\partial \mathbf{u}_j}{\partial \mathbf{x}_i} - \frac{2}{3}\frac{\partial \mathbf{u}_k}{\partial \mathbf{x}_k}\delta_{ij}\right) \tag{5}$$

American Institute of Aeronautics and Astronautics

where δ_{ij} is the Kronecker delta and subscripts i, j, k refer to the Cartesian coordinate components for $\mathbf{x} = (x, y, z)$. μ refers to the fluid dynamic viscosity and is obtained via Sutherland's law.[19] In Eq. (5), μ_T denotes the turbulence eddy viscosity, which is obtained from the negative SA model by:

$$\mu_T = \begin{cases} \rho\tilde{\nu}f_{v1} & \text{if } \tilde{\nu} \geq 0 \\ 0 & \text{if } \tilde{\nu} < 0. \end{cases} \tag{6}$$

The source term, $\mathbf{S}$, in Eq. (1) is given by $\mathbf{S} = [0, 0, 0, 0, 0, S_t]^T$, where the components for the continuity, momentum and energy equations are zero. The source term corresponding to the turbulence model equation takes the following form:[17]

$$S_t = P - D + \frac{1}{\sigma}c_{b2}\rho\nabla\tilde{\nu} \cdot \nabla\tilde{\nu} - \frac{1}{\sigma}(\nu + \tilde{\nu}f_n)\nabla\rho \cdot \nabla\tilde{\nu} \tag{7}$$

where the production term is given as

$$P = \begin{cases} c_{b1}\rho(1 - f_{t2})\tilde{S}\tilde{\nu} & \text{if } \tilde{\nu} \geq 0 \\ c_{b1}\rho(1 - c_{t3})S\tilde{\nu} & \text{if } \tilde{\nu} < 0 \end{cases} \tag{8}$$

and the destruction term is defined as

$$D = \begin{cases} \rho(c_{w1}f_w - \frac{c_{b1}}{\kappa_t^2}f_{t2})(\frac{\tilde{\nu}}{d})^2 & \text{if } \tilde{\nu} \geq 0 \\ -\rho c_{w1}(\frac{\tilde{\nu}}{d})^2 & \text{if } \tilde{\nu} < 0. \end{cases} \tag{9}$$

In Eq. (7), (8), and (9), ν denotes kinematic viscosity that is the ratio of dynamic viscosity to density, μ/ρ. Additional definitions associated with the production and destruction terms are given as:[17]

$$\tilde{S} = \begin{cases} S + \hat{S} & \text{if } \hat{S} \geq -c_{v2}S \\ S + \frac{S(c_{v2}^2 + c_{v3}\hat{S})}{(c_{v3} - 2c_{v2})S - \hat{S}} & \text{if } \hat{S} < -c_{v2}S \end{cases} \tag{10}$$

$$S = \sqrt{\vec{\omega} \cdot \vec{\omega}}, \quad \hat{S} = \frac{\tilde{\nu}}{\kappa_t^2 d^2}f_{v2}, \quad f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3}, \quad f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}}, \quad f_{t2} = c_{t3}e^{-c_{t4}\chi^2} \tag{11}$$

and

$$\chi = \frac{\tilde{\nu}}{\nu}, \quad r = \min(\frac{\tilde{\nu}}{\tilde{S}\kappa_t^2 d^2}, r_{\lim}), \quad g = r + c_{w2}(r^6 - r), \quad f_w = g\left(\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6}\right)^{1/6} \tag{12}$$

where the vorticity vector is given by, $\vec{\omega} = \nabla \times \mathbf{u}$ and d represents the distance to the nearest wall.

The constants in the negative SA model are given as: $c_{b1} = 0.1355$, $\sigma = 2/3$, $c_{b2} = 0.622$, $c_{t3} = 1.2$, $c_{t4} = 0.5$, $\kappa_t = 0.41$, $c_{w1} = c_{b1}/\kappa_t^2 + (1 + c_{b2})/\sigma$, $c_{w2} = 0.3$, $c_{w3} = 2$ $c_{v1} = 7.1$, $c_{v2} = 0.7$ and $c_{v3} = 0.9$. κ and T denote the thermal conductivity and temperature, respectively, and are related to the total energy and velocity as,

$$\kappa T = \gamma(\frac{\mu}{P_r} + \frac{\mu_T}{P_{r_T}}) \left(E - \frac{1}{2}(u^2 + v^2 + w^2) \right) \tag{13}$$

where P_r and P_{r_T} are the Prandtl and turbulent Prandtl number that are set to 0.72 and 0.9, respectively. In the case of laminar flow, the governing equations reduce to the compressible Navier-Stokes equations, where the turbulence model equation is deactivated and the turbulence eddy viscosity, μ_T, in the fluid viscous stress tensor and the thermal conduction term vanishes.

For the purpose of the spatial discretization, the Cartesian viscous fluxes are rewritten in the following equivalent form:

$$\mathbf{F}_v^x = \mathbf{G}_{1j}\frac{\partial \mathbf{Q}}{\partial \mathbf{x}_j}, \quad \mathbf{F}_v^y = \mathbf{G}_{2j}\frac{\partial \mathbf{Q}}{\partial \mathbf{x}_j}, \quad \mathbf{F}_v^z = \mathbf{G}_{3j}\frac{\partial \mathbf{Q}}{\partial \mathbf{x}_j} \tag{14}$$

where the matrices $\mathbf{G}_{ij}(\mathbf{Q})$ are determined by $\mathbf{G}_{ij} = \partial \mathbf{F}_v^{x_i}/\partial(\partial \mathbf{Q}/\partial \mathbf{x}_j)$ for $i, j = 1, 2, 3$.

III.   SUPG Discretization

Before extending the SUPG scheme for dynamic meshes, the methodology is first described on a static mesh for later reference. Here, the SUPG finite-element scheme is formulated as a weighted residual method, which can be cast in the form shown below

$$\int_\Omega (N + P) \left(\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbf{F} - \mathbf{S} \right) d\Omega = 0 \tag{15}$$

where N and P are weighting functions described further below. The domain of interest is discretized into a series of nonoverlapping elements, and the field variables are assumed continuous across element boundaries. Single-valued data is stored at the vertices of the elements and the solution is assumed to vary within each element according to a linear combination of polynomial basis functions

$$\mathbf{Q}_h = \sum_{i=1}^{n} N_i \mathbf{Q}_i \tag{16}$$

Here, $\mathbf{Q}_h$ represents the dependent variables approximated within each element, $\mathbf{Q}_i$ is the corresponding data at the nodes of the element, and each N_i represents a basis function. The weighting function, N, is composed of a linear combination of the same basis functions used in Eq. (16) for defining the variables within the element. The second contribution to the weighting function, P, is a stabilizing term that provides dissipation along preferential directions to eliminate odd-even point decoupling that often occurs with a standard Galerkin scheme, which is obtained if the second contribution is neglected. In the present work, the Streamlined Upwind Petrov-Galerkin (SUPG) method is used in defining the weighting function.[20]

$$N[I] + [P] = N[I] + \left(\frac{\partial N}{\partial x}[A] + \frac{\partial N}{\partial y}[B] + \frac{\partial N}{\partial z}[C] \right) [\tau] \tag{17}$$

$$N = \sum_{i=1}^{n} N_i c_i \tag{18}$$

where c_i are arbitrary constants and $[\tau]$ can be obtained using the following definitions[21]

$$[\tau]^{-1} = \sum_{j=1}^{M} \left(\left| \frac{\partial N_j}{\partial x_i}[\mathbf{A}_i] \right| + \frac{\partial N_j}{\partial x_i}[\mathbf{G}_{ik}]\frac{\partial N_j}{\partial x_k} \right) \tag{19}$$

$$\left| \frac{\partial N_j}{\partial x_i}[\mathbf{A}_i] \right| = [\mathbf{T}] |\Lambda| [\mathbf{T}]^{-1} \tag{20}$$

Here, M corresponds to the number of basis functions within the element and the repeated index, i,j, and k imply summation over all the values $(i, j, k = 1, 2, 3)$, and the definitions of $[\mathbf{G}_{ik}]$ corresponds to those given in Eq. (14).

The matrices $[\Lambda]$ and $[T]$ are the eigenvalues and right eigenvectors, respectively, of the matrix on the left side of Eq. (20) whereas the inverse of $[T]$ is given by $[T]^{-1}$. The resulting weak statement may be written as

$$\int_\Omega (N\frac{\partial \mathbf{Q}}{\partial t} - \mathbf{F} \cdot \nabla N - N\mathbf{S})d\Omega + \int_\Omega [P]\left\{\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbf{F} - \mathbf{S}\right\}d\Omega + \int_\Gamma N\mathbf{F} \cdot nd\Gamma = 0 \tag{21}$$

In evaluating the volume and surface integrals, Gaussian quadrature rules are used where for polynomial representations of the dependent variables of order p, the volume integrals are evaluated using quadrature formulas appropriate for integrating polynomials of order 2p. Similarly, surface integrals are integrated using formulas for integrating polynomials of order 2p + 1.[22] Note that because the field variables are assumed to be continuous in the interior of the domain, the surface integral typically vanishes on the boundaries of the interior elements and need only be evaluated on the boundaries of the domain where appropriate boundary conditions are applied. In this regard, on solid surfaces for inviscid flows, the boundary conditions are weakly enforced by assuming the normal velocity is zero in the flux computations. For viscous flows, strong boundary conditions are currently used to enforce the velocities on the surface of moving bodies. Using this boundary condition it is easy to verify that the implementation is correct. In the far-field, an approximate Riemann solver is used.[23]

IV.   Arbitrary Lagranian-Eulerian (ALE) formulation

The equations, as provided in Eq. (1), describe the conservation of mass, momentum, and energy for viscous flow in a stationary, Eulerian coordinate system. For a dynamically moving mesh, additional terms arise in the equations, whose origin is rooted in the transformation of the temporal derivative terms from a fixed coordinate system to a dynamically evolving one. Because these terms do not affect viscous contributions, the derivations and solution methodology can be illustrated, discretized, demonstrated, and verified using the inviscid subset of equations obtained by neglecting viscous terms. As such, for brevity, the derivations, and many of the verification experiments, are restricted to the inviscid subset of equations given in differential form as

$$\frac{\partial \mathbf{Q}(\mathbf{x}, t)}{\partial t} + \nabla \cdot \mathbf{F}_e(\mathbf{Q}) = 0 \quad \text{in} \quad \Omega \tag{22}$$

The transformation from a fixed coordinate system to a dynamic one is facilitated by use of the Reynolds transport theorem[24]

$$\frac{\partial}{\partial t}\int_{\Omega(t)}\mathbf{Q}d\Omega = \int_{\Omega(t)}\frac{\partial \mathbf{Q}}{\partial t}d\Omega + \int_{\Omega(t)}\nabla \cdot (\mathbf{Q}\mathbf{V}_g)d\Omega \tag{23}$$

Here, the first integral is evaluated over a time evolving control volume, and $\mathbf{V}_g$ is the mesh velocity.

Integrating Eq. (22) over the control volume and substituting Eq. (23), provides the Euler equations expressed in terms of a time-varying control volume as

$$\frac{\partial}{\partial t}\int_{\Omega(t)}\mathbf{Q}d\Omega + \int_{\Omega(t)}\nabla \cdot (\mathbf{F} - \mathbf{Q}\mathbf{V}_g)d\Omega = 0 \tag{24}$$

For finite-volume schemes the above equation is used as the starting point for discretization, where the domain of interest is divided into discrete control volumes, and the second volume integral is then converted to a surface integral and integrated numerically. For later reference, note that if constant freestream conditions are assumed in the integrand of Eq. (24), the following relationship is obtained

$$\frac{\partial V}{\partial t} = \int_\Gamma \mathbf{V}_g \cdot \mathbf{n}d\Gamma \tag{25}$$

where V is the integration volume. While this equation provides an identity relating the change in volume to the expansion or contraction of its surface, upon discretization, this equation will not necessarily be satisfied

American Institute of Aeronautics and Astronautics

under general mesh movements. The result is that in numerical simulations, freestream conditions may not be maintained. The desired preservation of freestream values in the presence of a deforming mesh is known as the Geometric Conservation Law (GCL).

For deriving a finite-element method, one approach is to first derive a Galerkin scheme, and then subsequently augment the Galerkin contribution with stabilization. The Galerkin scheme is first derived from the viewpoint of a weighted residual scheme as

$$\int_\Omega N \left(\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbf{F} \right) d\Omega = 0 \tag{26}$$

As with the finite-volume approach, the Reynolds transport theorem can again be applied after replacing the integrand in Eq. (23) with the product of the basis function and the dependent variable, $N\mathbf{Q}$.

$$\frac{\partial}{\partial t} \int_{\Omega(t)} \mathbf{Q} N d\Omega = \int_{\Omega(t)} N \frac{\partial \mathbf{Q}}{\partial t} d\Omega + \int_{\Omega(t)} N \nabla \cdot (\mathbf{Q} \mathbf{V}_g) d\Omega \tag{27}$$

Note that in the manipulation of the Reynolds transport theorem into this form two assumptions have been made.[25] Specifically,

$$\frac{\partial(\mathbf{Q}N)}{\partial t} = \mathbf{Q} \frac{\partial N}{\partial t} + N \frac{\partial \mathbf{Q}}{\partial t} \tag{28}$$

and

$$\frac{\partial N}{\partial t} + \mathbf{V}_g \cdot \nabla N = 0 \tag{29}$$

This term is the substantial derivative of the weighting function, and is analytically satisfied because these functions are independent of time in the Lagrangian coordinates. However, as with Eq. (25), these identities are not generally satisfied once discretized.

Upon substitution into Eq. (26), the following form is obtained

$$\frac{\partial}{\partial t} \int_{\Omega(t)} N \frac{\partial \mathbf{Q}}{\partial t} d\Omega + \int_{\Omega(t)} N \nabla \cdot (\mathbf{F} - \mathbf{Q} \mathbf{V}_g) d\Omega = 0 \tag{30}$$

Because a Galerkin formulation lacks sufficient dissipation to provide a robust algorithm, a suitable stabilization term can then be explicitly added, leading to the following form after subsequent integration of the Galerkin flux contribution by parts:

$$\frac{\partial}{\partial t} \int_{\Omega(t)} (N + P) \mathbf{Q} d\Omega - \int_{\Omega(t)} (\mathbf{F} - \mathbf{Q} \mathbf{V}_g)) \cdot \nabla(N) d\Omega + \int_{\Omega(t)} P \nabla \cdot (\mathbf{F} - \mathbf{Q} \mathbf{V}_g) d\Omega + \int_\Gamma N \mathbf{F} \cdot \mathbf{n} d\Gamma = 0 \tag{31}$$

Note that in the third term on the left-hand side of the equation, mesh speeds have been included to provide symmetry to the stabilization. Without the mesh speeds, the stabilization term would represent a strictly Eulerian point of view. Based on the velocities used for $\mathbf{V}_g$, the Galerkin contribution can represent a point of view ranging from a pure Eulerian to pure Lagrangian formulation. For a given relative oncoming velocity, the inclusion of the mesh speeds in this term allows consistent results to be obtained independent of whether the geometry is fixed, or translating.

In deriving the Galerkin formulation, the weighting function, N, does not have dependency on the flow variables. Attempting to formally follow a similar approach using the weighting function, P, starting from Eq. (15), would not lead to simple relationships such as Eq. (29) thereby rendering this approach to be somewhat ad hoc. Because of the development of the scheme presented in Eq. (31) is not particularly rigorous, an alternate approach is taken in the current work.

Returning to Eq. (24) and transforming to a fixed computational domain, the time derivative can be pulled inside the integral and includes the differentiation of the Jacobian to account for the time variation of the volume in physical space. A weighting function that includes both the Galerkin and Petrov-Galerkin contributions can then be included, and integration can be performed over the element in the reference space,

$$\int_{\Omega_o} (N + P) \left[\frac{\partial(\mathbf{Q}J)}{\partial t} + \nabla \cdot (\mathbf{F} - \mathbf{Q} \mathbf{V}_g) J \right] d\Omega = 0 \tag{32}$$

American Institute of Aeronautics and Astronautics

Two slightly different approaches for treating the time derivative are considered. In the first approach, subsequently referred to as "method 1", the time derivative is re-expressed using the chain rule as

$$\frac{\partial(\mathbf{Q}J)}{\partial t} = \mathbf{Q}\frac{\partial J}{\partial t} + J\frac{\partial \mathbf{Q}}{\partial t} = \mathbf{Q}J\nabla \cdot \mathbf{V}_g + J\frac{\partial \mathbf{Q}}{\partial t} \tag{33}$$

where Euler's expansion formula,[26–28] from continuum mechanics, has been used to relate the time rate of change of the Jacobian to the divergence of the mesh velocities:

$$\frac{\partial J}{\partial t} = J\nabla \cdot \mathbf{V}_g \tag{34}$$

After transforming back to physical space, the final equation for discretization is given as

$$\int_{\Omega(t)} (N + P)\left[\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot (\mathbf{F} - \mathbf{Q}\mathbf{V}_g) + \mathbf{Q}\nabla \cdot \mathbf{V}_g\right] d\Omega = 0 \tag{35}$$

In deriving Eq. (35), the time derivative of $(\mathbf{Q}J)$ has been rewritten in a nonconservative form. While common,[28–30] this approach requires investigation to quantify potentially undesirable effects. Additionally, this relationship is similar to Eqs. (25) and (29); it is valid at the continuous level, but when evaluated using discrete formulas, its accuracy should be suspect.

A second approach, henceforth referred to as "method 2", discretizes the conservative form of the time derivative without first using the chain rule. Utilizing a first-order accurate temporal discretization for simplicity, the time derivative can be approximated as[31, 32]

$$\frac{\partial(\mathbf{Q}J)}{\partial t} \approx \frac{\mathbf{Q}^{n+1}J^{n+1} - \mathbf{Q}^n J^n}{\Delta t} = \frac{\mathbf{Q}^{n+1}J^{n+1} - \mathbf{Q}^n J^n}{\Delta t} + \frac{\mathbf{Q}^n J^{n+1} - \mathbf{Q}^n J^{n+1}}{\Delta t} \tag{36}$$

The last term in this equation is identically zero, but allows the equation to be re-written as

$$\frac{\partial(\mathbf{Q}J)}{\partial t} \approx \mathbf{Q}^n \frac{J^{n+1} - J^n}{\Delta t} + J^{n+1}\frac{\mathbf{Q}^{n+1} - \mathbf{Q}^n}{\Delta t}. \tag{37}$$

Using the discrete form of Eq. (34), the time derivative may be approximated as

$$\frac{\partial(\mathbf{Q}J)}{\partial t} \approx \mathbf{Q}^n J^{n+1}\nabla \cdot \mathbf{V}_g + J^{n+1}\frac{\mathbf{Q}^{n+1} - \mathbf{Q}^n}{\Delta t}. \tag{38}$$

For first-order accurate differencing, this expression clearly represents an obvious discretization of Eq. (33), with the exception being that for method 1, $\mathbf{Q}$ is evaluated at time level $n+1$, whereas for method 2, it is evaluated at time level n. The extension to second-order accurate time differencing is straight forward,[31, 32] but leads to a slightly more opaque differencing of the product of the Jacobian and the derivative of the dependent variables.

$$\frac{\partial(\mathbf{Q}J)}{\partial t} \approx \mathbf{Q}^n J^{n+1}\nabla \cdot \mathbf{V}_g + J^{n+1}\left[1.5\left(\frac{\mathbf{Q}^{n+1} - \mathbf{Q}^n}{\Delta t}\right) - 0.5\frac{J^{n-1}}{J^{n+1}}\left(\frac{\mathbf{Q}^n - \mathbf{Q}^{n-1}}{\Delta t}\right)\right]. \tag{39}$$

Although in method 2 the time derivative is initially discretized in conservative form, the discrete form of Eq. (34) is again used. As with finite-volume schemes that invoke the discrete form of Eq. (25), and finite-element formulations that use Eq. (29), the consequences of using Eq. (34) requires further evaluation, which will be provided in a subsequent section.

V. Time Advancement

While high-order temporal differencing has previously been described in Ref. 7, the primary objective of the current work is to establish that the discrete geometric conservation law can be satisfied on dynamic high-order meshes, and to investigate subtle consequences related to achieving this objective. Therefore, to advance the solution at each time step, first-order (BDF1) and second-order (BDF2) backward time stepping schemes are used. A pseudotime term is added so that local CFL numbers can be used, with the intent of being able to robustly take large physical time steps. The scheme is illustrated in Eq. (40) for a BDF1 scheme for simplicity, with obvious extensions for second-order differencing.

$$\frac{\mathbf{Q}^{(n+1,i+1)} - \mathbf{Q}^{n+1,i}}{\Delta\tau} + \frac{\mathbf{Q}^{(n+1,i+1)} - \mathbf{Q}^{n}}{\Delta t} + R_s(\mathbf{Q}^{n+1,i+1}, \mathbf{Q}^{n}) = R(\mathbf{Q}^{n+1,i+1}, \mathbf{Q}^{n+1,i}, \mathbf{Q}^{n}) \qquad (40)$$

Here, the solution is sought at time level $n + 1$ whereas i is used to indicate the current solution at each subiteration within the time step. The pseudotime step, given by $\Delta\tau$, is determined locally at each mesh point based on a specified CFL number. R_s represents the steady-state contribution to the overall residual whereas R is the complete time-dependent residual for the current time, which will be driven to zero using a subiterative procedure. At each subiteration, the density, velocity components, temperature, and the turbulence-model working variable are tightly coupled and updated using a Newton-type algorithm. At each nonlinear subiteration, the linear system is approximately solved using the generalized minimal residual (GMRES) algorithm[33] with a preconditioner based on an incomplete lower upper (LU) decomposition.[34]

For the Newton-type algorithm, an initial update to the flow variables at the time step is computed using the locally varying time-step parameter, $\Delta\tau$, that is then multiplied by the current CFL number, which is systematically adjusted within each subiteration of each time step. Initially, the full update of the variables is added to the current estimate of the solution. The L_2 norm of the unsteady residual, including the pseudotime term, is compared to its value at the beginning of the subiteration. If the L_2 norm after the update is less than one half of the original value, the the CFL number is increased by a multiplicative factor, which is an input parameter that can be used for examining the effects on robustness and efficiency. If the L_2 norm reduction target is not met, a line search is conducted using four locations along the search direction to determine an appropriate relaxation factor. Using the four L_2 norm values of the residual, the optimal relaxation factor is determined by locating the minimum of a cubic polynomial curve fit through the samples. The solution is then updated using the relaxation factor, and the CFL number is neither increased nor decreased.

VI.   Temporal Conservation

The desire for spatial conservation is well known because of its importance in obtaining correct shock positions for steady flows. For unsteady flows, temporal conservation can be equally important for obtaining correct shock propagation speed. For steady flows, finite-volume schemes are conservative by construction because the flux computed on each face of the control volumes is added to the residual on one side of the interface, whereas it is subtracted from the residual on the other side. The spatial conservative property of the SUPG scheme has been previously established in Ref. 35 by writing the discretization in the form of a finite-volume scheme. An alternate means of establishing conservation is to sum the columns in the matrix representing the linearization of the steady residual. Because each entry in a column represents the change of a residual with respect to perturbations in the variable associated with the column, in a conservative scheme, some residuals in the column will increase, while others will decrease, such that the column sum is zero. This concept is verified in Figs. 1(a) and (b) for steady inviscid flow over an NACA 0012 airfoil at a freestream Mach number of 0.63 and an angle of attack of $2.0°$. The mesh used for this calculation is comprised of approximately 36,000 triangles with 18,000 nodes, and linear elements are used in the simulation. Figure 1(b) presents a scatter plot of the column sums associated with perturbations in each of the dependent variables. Note that only every 50th column is plotted for clarity, but that the column sums not depicted in the figure have similar values. As expected, the column sums are zero to machine precision.

The extension for time-dependent flows is straight-forward,[36] as demonstrated below for a second-order backward time differencing scheme. Here, the linearized residual entries for several time steps are presented in a single matrix to illustrate the relationships between the linearization of the residuals at the time levels. In Eq. (41), the matrix entries for the last three time steps are explicitly depicted, whereas previous time steps are implied using the diagonal dots.
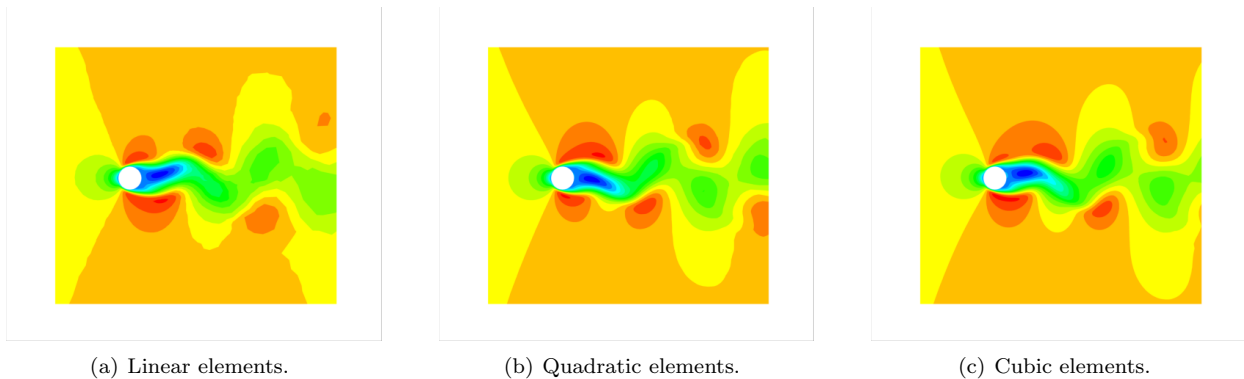
(a) Pressure distribution.

(b) Column sums.

Figure 1. Pressure distribution and column sums for verifying spatial conservation.

$$A = \begin{bmatrix} \begin{bmatrix} \ddots \\ \ddots \\ \ddots \end{bmatrix} & & & & \text{sum} & \\ & \begin{bmatrix} \ddots \\ \ddots \end{bmatrix} & & & \Downarrow & \\ & & \begin{bmatrix} \ddots \end{bmatrix} & & & \\ \left[\frac{\partial R^{N-2}}{\partial \mathbf{Q}^{N-4}}\right] & \left[\frac{\partial R^{N-2}}{\partial \mathbf{Q}^{N-1}}\right] & \left[\frac{\partial R^{N-2}}{\partial \mathbf{Q}^{N-2}}\right] & & & \\ & \left[\frac{\partial R^{N-1}}{\partial \mathbf{Q}^{N-3}}\right] & \left[\frac{\partial R^{N-1}}{\partial \mathbf{Q}^{N-2}}\right] & \left[\frac{\partial R^{N-1}}{\partial \mathbf{Q}^{N-1}}\right] & & \\ & & \left[\frac{\partial R^{N}}{\partial \mathbf{Q}^{N-2}}\right] & \left[\frac{\partial R^{N}}{\partial \mathbf{Q}^{N-1}}\right] & \left[\frac{\partial R^{N}}{\partial \mathbf{Q}^{N}}\right] \end{bmatrix}. \tag{41}$$

To verify the validity of the approach, column sums for the column indicated in Eq. (41) are computed for unsteady laminar flow over a circular cylinder at a freestream Mach number of 0.2, an angle of attack of 0°, and a Reynolds number of 120 based on the diameter of the circle. In Fig. 2, velocity contours at a random point during the time history are shown for solutions obtained with linear, quadratic, and cubic basis functions, all obtained on a stationary mesh. The maximum column sums obtained using the last three time steps are shown in Table 1. Note that many of the column sums for the block entry corresponding to $\left[\frac{\partial R^{N-2}}{\partial \mathbf{Q}^{N-2}}\right]$ are on the order of 1×10^3. Therefore, the sums from the blocks in the column are of similar size but cumulatively of opposite sign, thereby cancelling thus the column sums are zero to machine precision. Also note that the columns sums assocated with boundary nodes, as well as nodes connecting to boundary nodes, are not included in the summation. The reason is due, in part, because of the use of strong enforcement of the velocity boundary conditions, which subsequently affects all of these columns. An important aspect of this particular example is that the mesh is stationary. The effects of moving the mesh, and the implication for many schemes on dynamic meshes will be studied in a subsequent section.

Table 1. Maximum temporal conservation errors for impulsively started flow over circular cylinder on a static mesh.

Element Type	Continuity	x-momentum	y-momentum	Energy
Linear	0.3183231456E-11	0.1364242053E-11	0.1534772309E-11	0.9094947018E-12
Quadratic	0.9663381206E-12	0.7958078641E-12	0.6536993169E-12	0.6252776075E-12
Cubic	0.5258016245E-12	0.3694822226E-12	0.3410605132E-12	0.2842170943E-12

American Institute of Aeronautics and Astronautics

(a) Linear elements.　　　　　　　(b) Quadratic elements.　　　　　　　(c) Cubic elements.

Figure 2.　Velocity contours for impulsively started cylinder: $M_\infty = 0.2$, $\alpha = 0.0°$, $Re = 120$.

VII.　Results

Results are presented to demonstrate that the present scheme satisfies the discrete geometric conservation law, while also maintaining the temporal design order of accuracy. Temporal conservation errors are then examined in detail by simulating the flow in a shock tube, where conservation errors are quantitatively assessed and the effects on the accuracy of the simulated flowfield are examined. Finally, initial three-dimensional results for turbulent flow over an oscillating wing are also demonstrated and shown to agree well with an established finite-volume solver.

Examples are first given to demonstrate that the stabilized finite-element scheme has been extended to time-dependent simulations on dynamic meshes, that the GCL is satisfied, and that temporal design order of accuracy is maintained. For this set of results, inviscid flow is considered because, as mentioned previously, the addition of viscous terms does not affect the ability of a particular formulation to satisfy GCL. However, the current implementation of the finite-element scheme can simulate turbulent flows, as demonstrated in Refs. 1, 7–10, and turbulent results will be subsequently presented.

The first result is simply a demonstration that the flow field obtained at a uniform freestream Mach number can be reproduced using a lower Mach number in addition to translating the airfoil in the opposite direction. The NACA 0012 airfoil considered earlier for demonstrating spatial conservation is used. For the stationary airfoil, the freestream Mach number is 0.63 and the airfoil has been inclined $2°$. For the translating airfoil, the freestream Mach number is reduced to 0.315 and translation is imposed in the opposite direction so that the combined relative Mach number remains 0.63. The residuals for the stationary airfoil case have been converged to the order of 1×10^{-15}. For the translating case, the solution is converged so that the initial residual on the last time step is on the order of 1×10^{-14}. Note that on the first subiteration at a given time step, the initial estimate of the solution variable for the time step is initialized to that of the previous time step, such that the computed residual is the same as a steady-state case. At convergence, and after rescaling the results to account for the fact that the Mach number is used in computing the coefficients and pressure distributions, the difference in lift coefficient and drag coefficients between the stationary and dynamic mesh computations are within 1×10^{-10} of each other. As seen in Fig. 3, the pressure distributions for the stationary and dynamic mesh simulations are indistinguishable. The ability to reproduce a steady simulation with the translating airfoil is due to the presence of the mesh speeds in the stabilization term. Without these terms, the Galerkin contribution to the residual would suitably adjust to account for the moving mesh, but the stabilization term would correspond to that of a steady flow at the reduced Mach number.

To verify that the GCL is satisfied, and to confirm that the temporal order of accuracy is maintained, the next case uses a freestream Mach number of 0.2, a mean angle of attack of $0.0°$, an unsteady pitching amplitude of $1.0°$, and a reduced frequency of unity. During the pitching cycle, the motion of the airfoil surface is specified, while the mesh points in the interior of the domain are repositioned using linear elasticity. The baseline mesh and density contours at two intermediate positions during the fourth pitching cycle are shown in Fig. 4. The solutions have been obtained using quadratic elements, and the contours are generated by enriching each triangle in the baseline mesh with additional mesh points. The solution is then evaluated at the new points using the basis functions, and results are saved in a format suitable for use in standard visualization software.

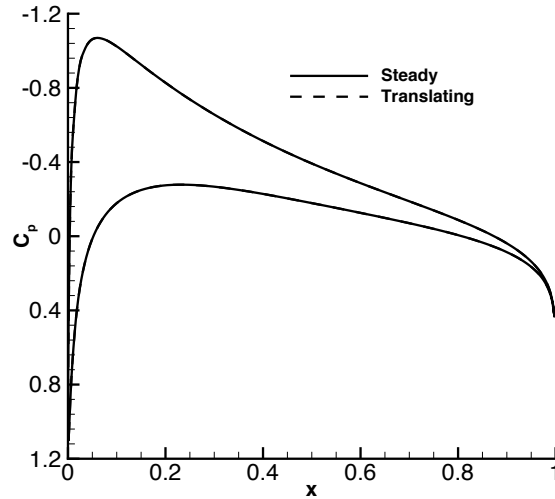American Institute of Aeronautics and Astronautics

Figure 3. Comparison of pressure distribution obtained on static and translating Mesh.

(a) Baseline mesh.

(b) Density contours at $0.0°$ ↑.
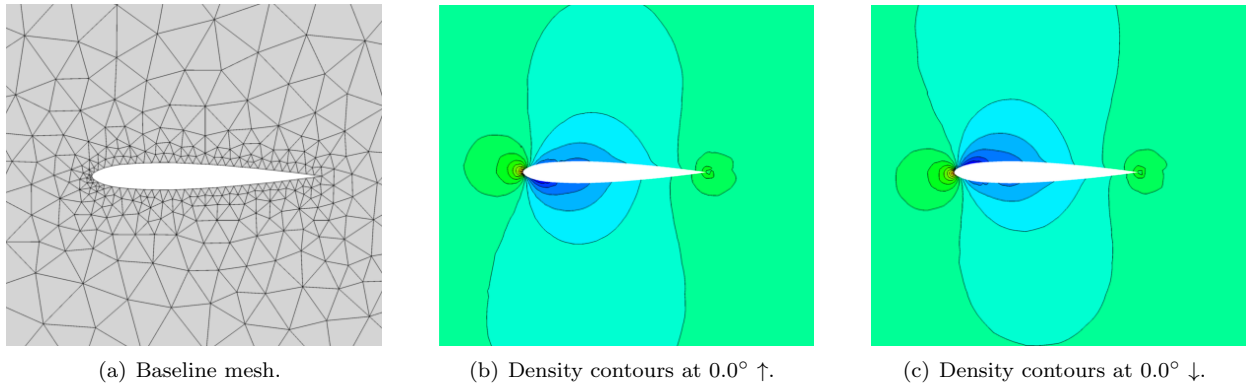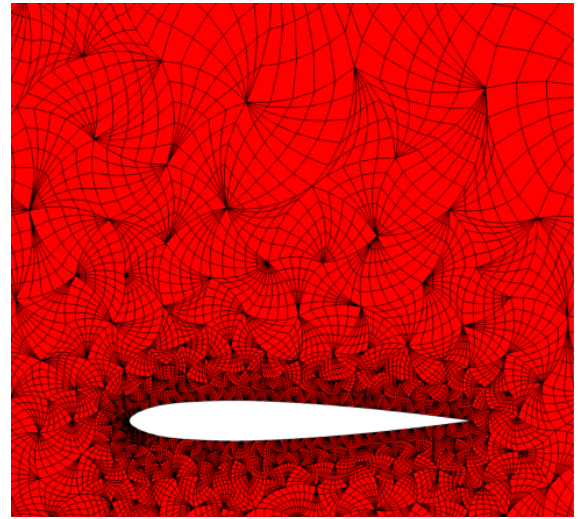
(c) Density contours at $0.0°$ ↓.

Figure 4. Mesh and density contours of oscillating NACA 0012 airfoil.

As is well known, the purpose of the GCL is to preserve freestream flow when the mesh is dynamically deformed. To demonstrate that the present formulation satisfies the GCL, a solution is initialized with freestream values, and the variables on the surface of the airfoil and on the far field boundary are held constant for all subsequent time steps. For these simulations, quadratic elements are again used, and all the edges in the mesh are sinusoidally oscillated to introduce both deformation and curvature into the simulation. Figure 5(a) shows density contours on the mesh with the edges in the undeformed configuration, whereas Fig. 5(b) shows similar contours at a point in the cycle where the edge deformations are at a maximum. Note that the enriched mesh, described above, is depicted in the figure. The density throughout the field is maintained at freestream values, despite the oscillations imposed on the mesh. In the figure, the constant-color density contours visually demonstrate that the freestream values are indeed maintained. Although not shown, the computed residuals throughout the oscillation cycle are maintained at machine zero, and all variables retain their freestream values, further verifying the satisfaction of the GCL. While this demonstration has been performed using quadratic elements on a continually curving mesh, similar results have been obtained using linear elements and cubic elements. Also note that only results obtained using method 1 have been shown, as methods 1 and 2 are identical for freestream flow.

To verify that the correct temporal order of accuracy is maintained, eight cycles of oscillation are computed beginning from an initially converged steady-state solution at the mean angle of attack. To determine the order of accuracy, simulations are computed using various steps per cycle ranging from 20 to 640. The lift coefficient at the end of the eighth cycle is used for computing errors. Here, tests with both linear and quadratic elements are conducted, although only the results for linear elements are shown as the results for quadratic elements are very similar. Note that in determining the temporal order of accuracy, it is assumed that because the spatial discretizations do not change within each test, that any variation in the final lift

American Institute of Aeronautics and Astronautics
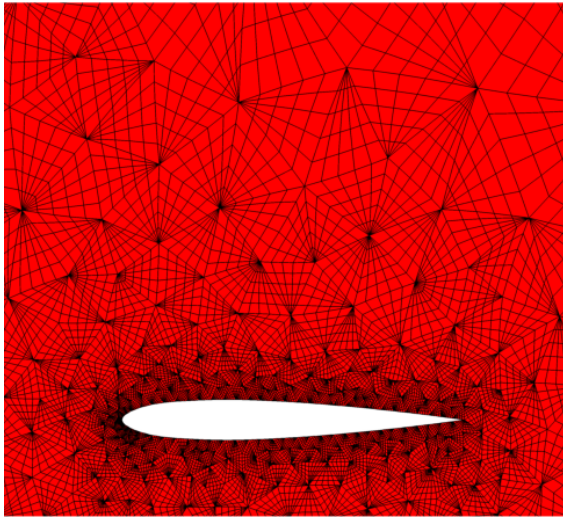
| (a) Mesh with undeformed edges. | (b) Mesh at maximum edge curvature. |

Figure 5. Mesh with oscillating edge curvature.

coefficient is attributable to temporal integration.[31] The achieved order of accuracy for each of the simulations is shown in Table 2. As observed in the table, the solution obtained using BDF1 temporal integration achieves first-order accuracy as expected, whereas the solutions obtained using BDF2 are verified to maintain second-order accuracy.

Table 2. Order of accuracy.

	Method 1		Method 2	
Steps/Cycle	Lift Slope BDF1	Lift Slope BDF2	Lift Slope BDF1	Lift Slope BDF2
20/40/80	0.4574345947E+00	0.1998040136E+01	0.4561396771E+00	0.1997634326E+01
40/80/160	0.8029611833E+00	0.2013662794E+01	0.8025580920E+00	0.2013601823E+01
80/160/320	0.9148943698E+00	0.2010366998E+01	0.9147298035E+00	0.2010362937E+01
160/320/640	0.9604253007E+00	0.2005713830E+01	0.9603535821E+00	0.2005734845E+01

American Institute of Aeronautics and Astronautics

Two approaches, for developing a SUPG discretization have been presented and verified to satisfy the geometric conservation law. However, as previously discussed, satisfying the GCL is unrelated to achieving temporal conservation, and only indicates whether freestream conditions can be maintained in the presence of a deforming mesh. The two methods described earlier are very similar, and differ only in the treatment of the time derivative. In the first method, the chain rule is used at the continuous level, and Eq. (34) is used to replace the time derivative of the Jacobian with the product of the Jacobian and the divergence of the mesh velocities. In the second approach, the time derivative of the product of the dependent variables and the Jacobian is discretized in conservative form, and the use of Eq. (34) is invoked after discretization. To this end, it is demonstrated that while both approaches satisfy GCL and maintain design order of accuracy, neither approach is strictly conservative in time, although the errors satisfy an order property and appear to have no substantial impact on solution quality. Furthermore, it is asserted that essentially all approaches for satisfying GCL on unstructured meshes likely do not strictly satisfy temporal conservation because of assumptions made early in the derivations.

To quantify the level of temporal conservation errors, and to examine the effects on the accuracy of the solution, a simulation of a shock tube is used to assess whether the velocities and strengths of the shock, the expansion, and the contact discontinuity are properly captured. For the simulation, the domain extends from $-2 \leq x \leq 3$ with a height of 0.5. The initial discontinuity is located midway through the domain with a pressure ratio of 5 on either side of the shock, and second-order temporally accurate simulations are obtained using a nondimensional time step of 0.05 and ran for 340 time steps. While the time step is held fixed for the initial simulations, subsequent simulations will investigate the effect of the time step on conservation errors. A depiction of the domain, including the initial and final distribution of density, is shown in Fig. 6

(a) Initial density distribution.

(b) Final density distribution.

Figure 6. Shock tube $p_{\mathrm{L}}/p_{\mathrm{R}} = 5$.

A dynamic mesh is achieved by introducing "displacement circles" at 5 locations within the mesh, where a displacement circle is a region in which the mesh is sinusoidally displaced along radial lines with zero displacement at the center of the circle, reaching a maximum value at the outer radius. Specifically, the displacements are given as

$$\overrightarrow{d} = A \sin(\omega t) \tag{42}$$

where the direction of the displacement vector is radial. Table 3 provides a summary of the locations, (x_c, y_c), the radius of influence, r, the amplitude, A, and the frequency, ω, for each circle. Note that k is a somewhat arbitrary parameter, currently set to 75, that serves as a baseline reduced frequency. The locations of the deformation circles are illustrated in Fig. 7(a), while a depiction of their relative extent and magnitude at a particular point in the simulation is shown on the actual mesh in Fig. 7(b). In addition to the mesh deformations within the circles, linear elasticity is used in the remainder of the domain and, therefore, the entire domain is ultimately deformed at every time step.

Table 3. Parameters for deformation dircles

Circle	x_c	y_c	r	A	ω
1	-1.50	0.22	0.14	0.02	$0.9kt$
2	-0.25	0.30	0.10	0.13	$0.75kt$
3	0.60	0.25	0.15	0.02	kt
4	1.50	0.35	0.07	0.01	$1.5kt$
5	2.25	0.20	0.15	0.02	$1.3kt$

Note that observing effects caused by temporal conservation errors is somewhat conflated with the effects caused by dispersion errors. This is due to the fact that both phase velocities and group velocities are directly effected by dispersion, thereby making it impossible to unambiguously assign any observed differences in the simulations to either conservation or dispersion. However, using the method discussed above, the magnitude of conservation errors can be quantified by summing the columns of the linearized residuals.
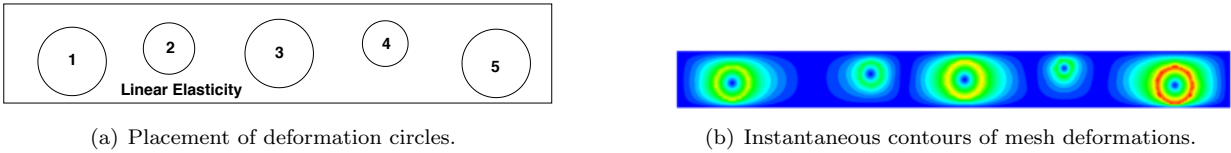
American Institute of Aeronautics and Astronautics

(a) Placement of deformation circles.

(b) Instantaneous contours of mesh deformations.

Figure 7. Mesh deformation circles for shock tube.

A qualitative comparison of density solutions at the final time, obtained on both static and dynamic meshes, is shown in Fig. 8. In this figure, method 1 is used for the computations and, as seen, no noticeable differences are seen in the solutions obtained on the static or dynamic meshes. Solutions obtained using method 2 are indistinguishable from those shown above and, therefore, not shown.

(a) Static mesh.

(b) Pulsating mesh.

Figure 8. Comparison of final solution on static and pulsating mesh for shock tube $p_{\mathrm{L}}/p_{\mathrm{R}} = 5$.

Quantitative comparisons are shown in Fig. 9(a), which depicts the density along the length of the shock tube at the final time, whereas Fig. 9(b) shows the solution in the vicinity of the shock and contact discontinuity. Results obtained with linear and quadratic elements, on both static and dynamic meshes, are compared to the exact solution. As seen, all the simulation results agree well with the exact solution in that the location of the shock and contact discontinuities are well captured, as are the details of the expansion fan. The oscillations at the discontinuity are attributed to the fact that no shock smoothing is used in the simulations to avoid introducing unnecessary complicating factors.

(a) Far-field view of density profiles for shock tube.

(b) Density profiles near shock and expansion.
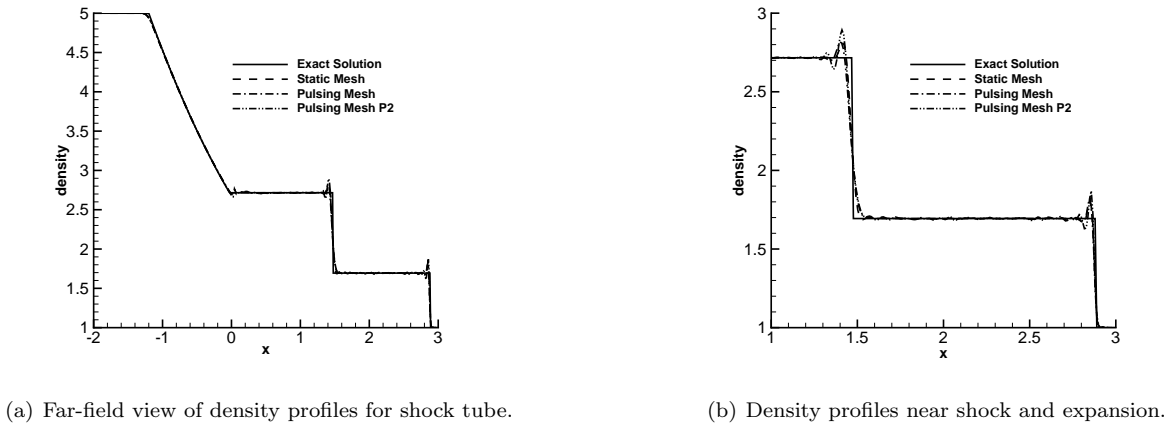
Figure 9. Density profiles for shock tube at $t = 1.7$.

Using column sums, the errors in temporal conservation have been verified to be zero for the static mesh, but, as seen in Fig. 10, the maximum conservation errors in the field are clearly nonzero for the simulations on the deforming mesh. Here, the conservation errors are computed at an arbitrary time in the simulation using time steps ranging from 0.01 to 0.00015625. Note that the largest time step of 0.01 is smaller by a factor of 5 than that used in the simulations shown previously. One would expect that the maximal errors in those simulations are somewhat larger than depicted here, however, these errors do not appear to adversely affect the solution. Although it is unsurprising that the results obtained with method 1 have conservation errors, it is noted that method 2 also has conservation errors, even though the temporal discretization originated from a conservative form of the equations. While the errors with method 2 are initially lower than those of method 1, these differences are eventually eliminated as the mesh is refined and both methods obtain an asymptotic convergence rate of 2.0, as expected.

Additionally, depicted in Fig. 10, are temporal conservation errors obtained with an alternate method for evaluating the time derivative as shown in Eq. (43).

American Institute of Aeronautics and Astronautics

$$\frac{\partial \mathbf{Q}}{\partial t} = \frac{\partial \mathbf{Q}}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial t} \tag{43}$$

where $\mathbf{q}$ represent primative variables of density, velocity, and temperature.

Analytically, the left-hand side and right-hand side of Eq. (43) are equivalent, but as with Eqs. (34), (25) and (29), these relationships are not satisfied upon discretization. Therefore, for reference purposes, results obtained using this discretization are referred to as being obtained with the "inconsistent" method. As seen in Fig.10, the conservation errors obtained using this formulation of the time derivative do not exhibit an order property as the time step is systematically reduced. The effect of the lack of conservation, and more specifically, the lack of an order property, is demonstrated in Figure 11. This figure is similar to that shown before, but the solutions obtained with quadratic basis functions have been replaced by the solution obtained with the alternate time discretization. The results obtained with the inconsistent time discretization exhibits a slightly slower shock propagation speed than the other solutions, or the exact solution. The same discretization of the time term has been demonstrated using nonlinear Burger's equation to also yield incorrect shock speeds[37] where the term "inconsistent" is used in reference to the discretization being inconsistent with the weak form of the equations.
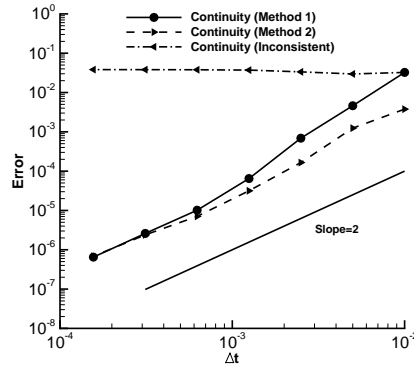
Figure 10. Errors in time conservation for shock tube.
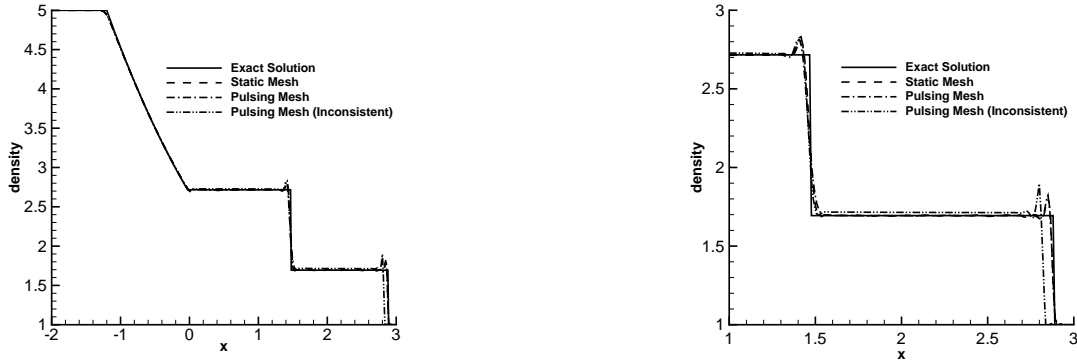
(a) Far-field view density profiles for shock tube.

(b) Density profiles near shock and expansion.

Figure 11. Density profiles for shock tube at $t = 1.7$.

Although the discretization described as method 2 originates from a conservative form of the time derivative, temporal conservation errors are clearly present. As shown below, the source of the errors lies in the fact that Eq. (34) is not identically satisfied after discretization. Similar results should be expected for any method that explicitly, or implicitly, relies on Eqs. (25) or (29).

American Institute of Aeronautics and Astronautics

For illustrative purposes, it is helpful to examine a finite-volume scheme with first-order backward time differencing, because the scheme is somewhat transparent for analysis. Referring to Eq. (44), which is similar to Eq. (41) but modified to reflect first-order time differencing, the derivatives in the column indicated with the arrow are required for computing the column sums.

$$
A = \begin{bmatrix}
\begin{bmatrix} \ddots & \\ & \ddots \end{bmatrix} & & & \text{sum} & \\
& \begin{bmatrix} \ddots \end{bmatrix} & & \Downarrow & \\
& \left[\frac{\partial R^{N-2}}{\partial \mathbf{Q}^{N-1}}\right] & \left[\frac{\partial R^{N-2}}{\partial \mathbf{Q}^{N-2}}\right] & & \\
& & \left[\frac{\partial R^{N-1}}{\partial \mathbf{Q}^{N-2}}\right] & \left[\frac{\partial R^{N-1}}{\partial \mathbf{Q}^{N-1}}\right] & \\
& & & \left[\frac{\partial R^{N}}{\partial \mathbf{Q}^{N-1}}\right] & \left[\frac{\partial R^{N}}{\partial \mathbf{Q}^{N}}\right]
\end{bmatrix}.
\tag{44}
$$

For a finite-volume scheme, a notional residual computation at any given time step can be represented using Eq. (45)

$$
R(\mathbf{Q}^{n+1}) = \sum_{i=1}^{N_f} \mathbf{F}(\mathbf{Q}^{n+1}) \cdot n - \sum_{i=1}^{N_f} \mathbf{Q}^{n+1} \mathbf{V}_g \cdot n + \frac{\mathbf{Q}^{n+1} V^{n+1} - \mathbf{Q}^n V^n}{\Delta t}.
\tag{45}
$$

The first two terms represent a generic discretization of the surface integral over each control volume in Eq. (24). Note that in this equation, the mesh speeds have been separated from the flux vector normally associated with the Eulerian frame of reference. The reason is that for freestream flow, the columns associated with the flux vector automatically sum to zero because they satisfy the governing equations and are evaluated with geometric metric terms without specificity as to whether the mesh is moving or not. The terms required for evaluating the matrix entry for the derivative of the residual at time-level $N-1$ can now be expressed as

$$
\frac{\partial R^{N-1}}{\partial \mathbf{Q}^{N-1}} = - \sum_{i=1}^{N_f} \mathbf{V}_g \cdot n + \frac{V^{N-1}}{\Delta t}.
\tag{46}
$$

Similarly, the contribution to the column originating from the linearization of the residual at time N with respect to the variables at time $N-1$ is given as

$$
\frac{\partial R^{N}}{\partial \mathbf{Q}^{N-1}} = - \frac{V^{N}}{\Delta t}.
\tag{47}
$$

Using Eqs. (46) and (47), the column sum, shown below in Eq. (48) is therefore the discrete version of finite-volume GCL.

$$
\frac{\partial R^{N-1}}{\partial \mathbf{Q}^{N-1}} + \frac{\partial R^{N}}{\partial \mathbf{Q}^{N-1}} = \frac{V^N - V^{N-1}}{\Delta t} - \sum_{i=1}^{N_f} \mathbf{V}_g \cdot n \approx \frac{\partial V}{\partial t} - \int_{\Gamma} \mathbf{V}_g \cdot n d\Gamma
\tag{48}
$$

Equation (48) is a discrete analog of (25) and demonstrates that even when the time derivative, $\frac{\partial (\mathbf{Q}V)}{\partial t}$, is approximated in conservative form, temporal conservation errors still arise because of the assumptions made early in the derivation that Eq. (25) is discretely valid. As stated previously, finite-volume and finite-element schemes typically utilize similar assumptions embodied in discrete forms of Eqs. (25), (29) and (34), and all should be expected to contribute to temporal conservation errors.

For finite-element approaches, an alternate approach has been proposed in Refs. 38 and 26 that may alleviate these errors. In this formulation, Eq. (34) is viewed as an ordinary differential equation and is evolved in time using numerical techniques, thereby discretely satisfying this relationship. A tradeoff, however, exists in that for the case of an initially dynamic mesh that is eventually frozen, the Jacobians that can be accurately computed on the static mesh will necessarily differ by truncation error from those computed via time evolution.

The most accurate and consistent method for computing flows on time-dependent meshes are space-time formulations, which, unfortunately, can still be very expensive for engineering computations. While not dispositive on the issue, the results presented above indicate that the temporal conservation errors for many applications in aerospace engineering are minimal, or at least do not add noticeable errors above dispersion errors already present in the simulations.

The discretization and time-advancement algorithm described previously has been extended for three-dimensional turbulent flow and is operational as a statically linked library within FUN3D.[1] For initial verification purposes, a simulation has been conducted for the Baseline Supercritical Wing (BSCW).[39, 40] While this configuration is intended for validation studies for aeroelastic prediction methods, the current results are only used for a forced oscillation at a subsonic Mach number and comparisons are made with the finite-volume capability in FUN3D. The wing, and the surface mesh, are depicted in Fig. 12. For these calculations, a tetrahedral mesh is used, which is comprised of $17,463,974$ elements and $2,964,728$ nodes. The chord is 16 inches, the span is 32 inches, and the normal spacing at the wall is 0.914×10^{-4} inches.
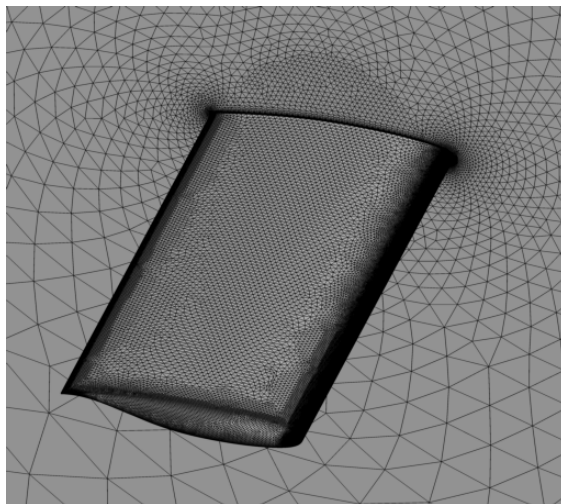
Figure 12. Mesh for sinusoidally oscillating wing: $M_\infty = 0.3$, $\alpha_s = 3.0°$, $\alpha_u = 1.0°$ $Re = 4.56 \times 10^3$, $k = 0.108$.

Qualitative comparisons between the finite-element solution and the finite-volume solution are given in Figs. 13 and 14. Here, Fig. 13 depicts pressure contours on the second oscillatory cycle as the wing is descending through zero degrees angle of attack, whereas Fig. 14 is at a subsequent time when the wing is ascending through zero. As seen, the solutions qualitatively agree very well.
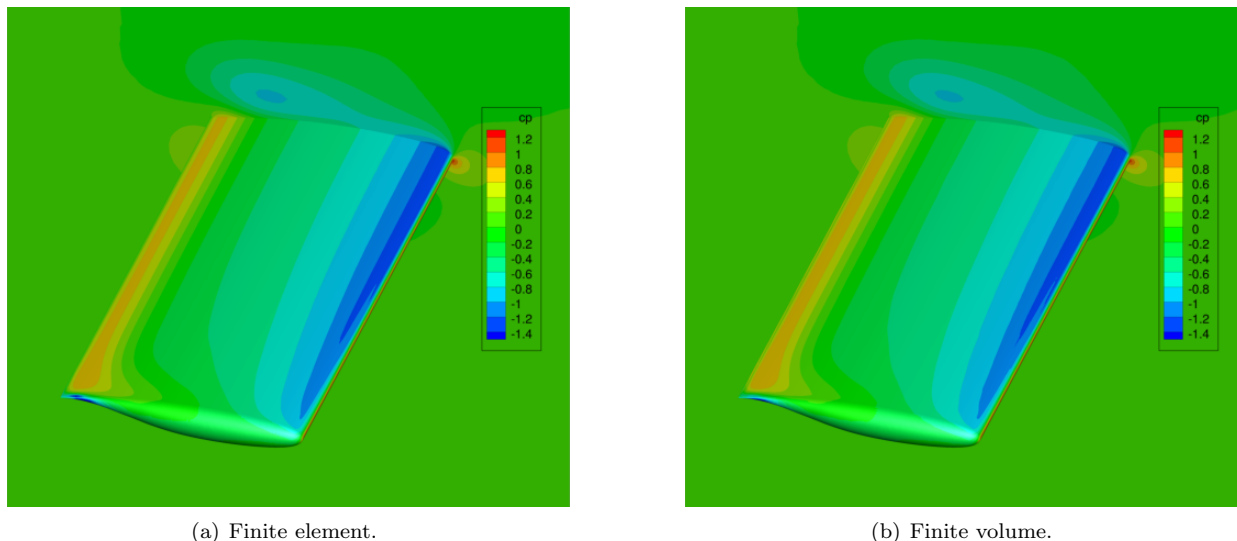

(a) Finite element.

(b) Finite volume.

Figure 13. Pressure contours for sinusoidally oscillating wing at $0.0°$ ↓: $M_\infty = 0.3$, $\alpha_s = 3.0°$, $\alpha_u = 1.0°$ $Re = 4.56 \times 10^3$, $k = 0.108$.

Quantitative comparisons between the solutions are shown at the corresponding times in Fig. 15 and Fig. 16, respectively. In each figure, comparisons are shown at $\eta = 0.25$, $\eta = 0.5$, and $\eta = 0.94$, where η is the relative spanwise position on the wing. As seen, the finite-element and finite-volume solutions agree very well with each other at both times during the simulation for all spanwise stations.
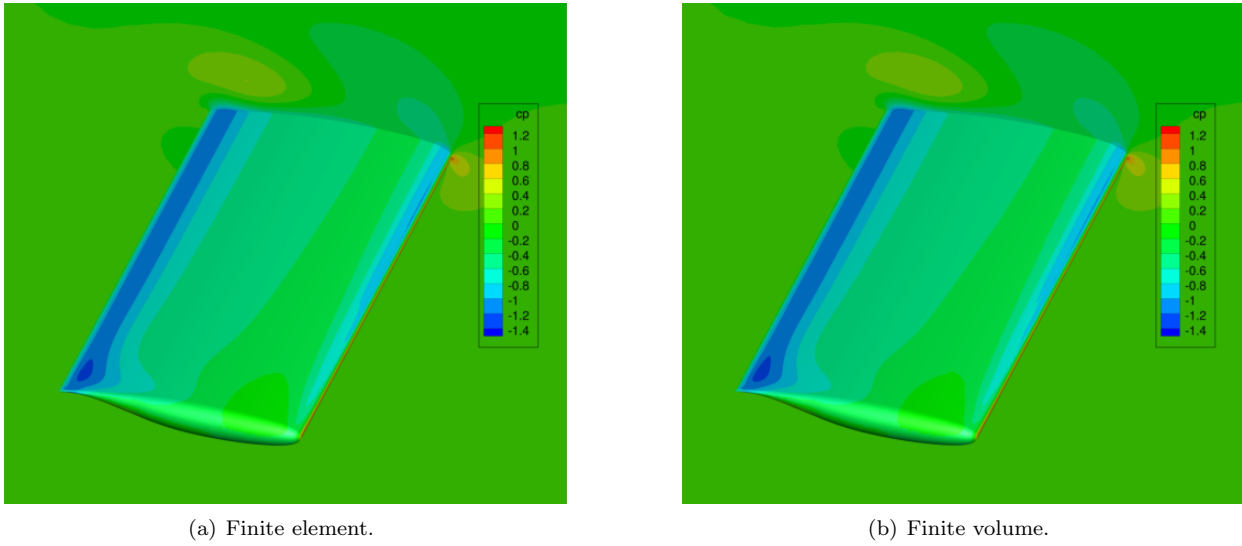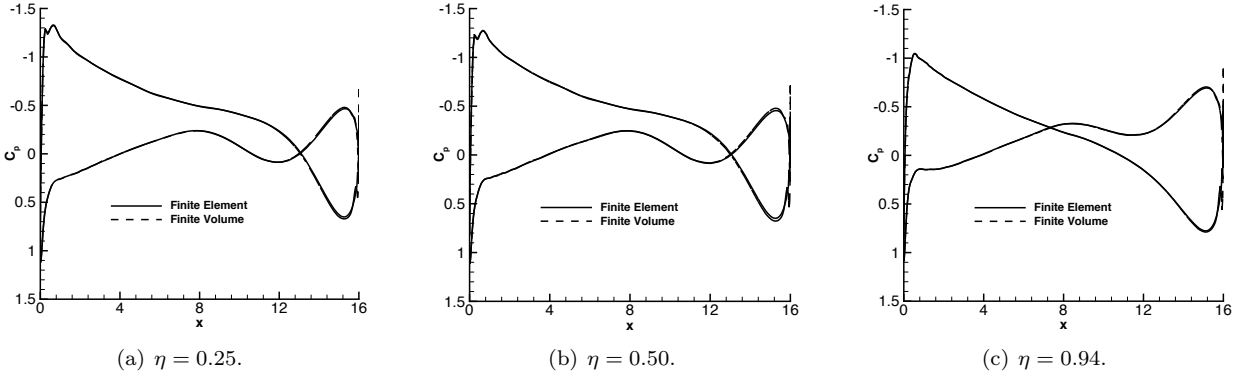
American Institute of Aeronautics and Astronautics

(a) Finite element.

(b) Finite volume.

Figure 14. Pressure contours for sinusoidally oscillating wing at $0.0°$ ↑**:** $M_\infty = 0.3$, $\alpha_s = 3.0°$, $\alpha_u = 1.0°$ $Re = 4.56 \times 10^3$, $k = 0.108$.

(a) $\eta = 0.25$.

(b) $\eta = 0.50$.

(c) $\eta = 0.94$.

Figure 15. Pressure distribution comparisons for sinusoidally oscillating wing at $0.0°$ ↓**:** $M_\infty = 0.3$, $\alpha_s = 3.0°$, $\alpha_u = 1.0°$ $Re = 4.56 \times 10^3$, $k = 0.108$.

As described in the section on time advancement, the current algorithm uses a Newton-type iterative scheme to iteratively converge the solution at each time step. Furthermore, within the time step, a line search algorithm is used to adjust the CFL number in a pseudotime scheme to improve the robustness. Numerical experiments to date indicate that for many inviscid flows, such as presented earlier, the initial CFL number can be essentially set to infinity without detrimental effects on robustness. For turbulent flows, this is not the case and the CFL controller is essential for robust simulations. The importance of the adjustable CFL number stems from the fact that many cells in the mesh are very small, so that at a specified time step, very large CFL numbers may be locally present. For the particular example under consideration, a time step of 1.6598 has been used, which results in 100 steps per oscillation cycle. However, with this time step, the maximum CFL number in the field is over 339,000 so that inclusion of the pseudotime term has proven necessary and effective for increasing robustness. At each time step during the solution process, several subiterations are used to drive the unsteady residual to zero. As shown in Fig.17, the current scheme is very effective at reducing the residual to machine tolerance, requiring approximately 20 subiterations for each time step.
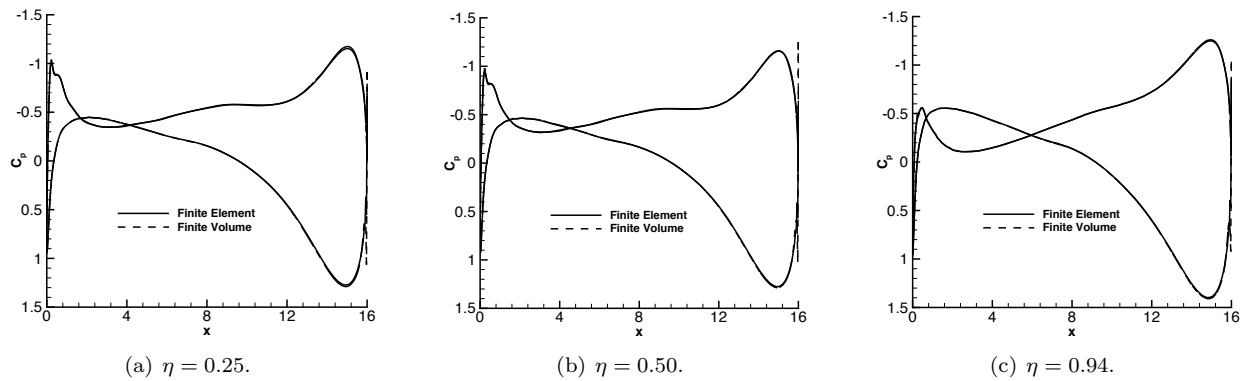
American Institute of Aeronautics and Astronautics

(a) $\eta = 0.25$.   (b) $\eta = 0.50$.   (c) $\eta = 0.94$.

Figure 16. Pressure distribution comparisons for sinusoidally oscillating wing at $0.0^\circ \uparrow$: $M_\infty = 0.3$, $\alpha_s = 3.0^\circ$, $\alpha_u = 1.0^\circ$ $Re = 4.56 \times 10^3$, $k = 0.108$.
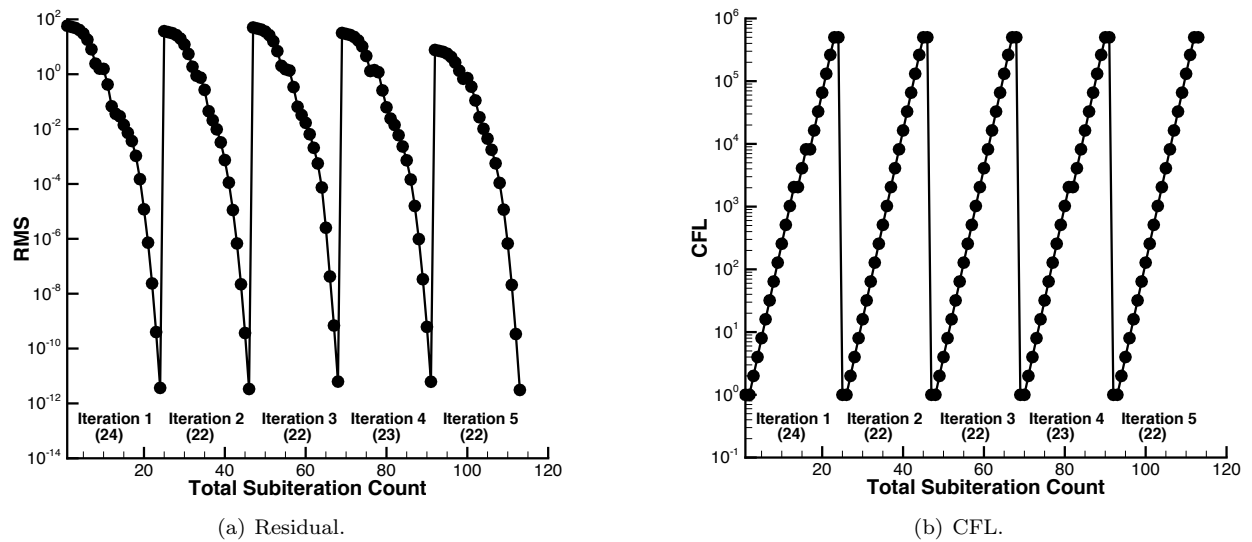

(a) Residual.   (b) CFL.

Figure 17. Residual and CFL history for sinusoidally oscillating wing.

VIII.   Summary

A stabilized finite-element scheme for computing turbulent flows on unstructured meshes has been extended for applicability to unsteady flows on dynamic meshes with curved elements. The formulations are demonstrated to satisfy the geometric conservation law and maintain the temporal order of accuracy of the underlying time-integration schemes. In deriving schemes to satisfy the above requirements, several options are available, and two closely related methods are presented. Temporal conservation errors are quantitatively examined and the underlying source of the errors is identified. For the simulations considered here, the temporal conservation errors have no noticeable effects. The paper demonstrates that the source of conservation errors stems from the fact that geometric relationships that are valid at the continuous level, are only valid to truncation error upon discretization. Because these assumptions are commonly made throughout the research community, it is quite likely that many methods in wide use today unknowingly possess the same source of errors. Finally, an initial verification simulation is presented for three-dimensional turbulent flow over a sinusoidally oscillating wing.

IX.   Acknowledgments

References

[1] Anderson, W. K. and Newman III, J. C., "Stabilized Finite-Elements in FUN3D," AIAA Paper 2017–0017, 2017.

[2] Rasquin, M., Smith, C., Chitale, K., Seol, E. S., Matthews, B., Martin, J., Sahni, O., Loy, R., Shephard, M. S., and Jansen, K. E., "Scalable Fully Implicit Finite Element Flow Solver with Application to High-Fidelity Flow Control Simulations on a Realistic Wing Design," *Computing in Science and Engineering*, Vol. 16, No. 6, 2014, pp. 13–21.

[3] Lesoinne, M. and Farhat, C., "Geometric Conservation Laws for Flow Problems with Moving Boundaries and Deformable Meshes, and their Impact on Aeroelastic Computations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 134, No. 1-2, 1996, pp. 71–90.

[4] Tezduyar, T. E., "Finite Element Methods for Flow Problems with Moving Boundaries and Interfaces," *Archives of Computational Methods in Engineering*, Vol. 8, No. 2, 2001, pp. 83.

[5] Viale, M. and Nigro, N., "Stabilized Finite Element Method with an ALE Strategy to Solve Moving Boundary Problems," www.fceia.unr.edu.ar/fceia1/mecanica/Automotores/CFD-ALE-Engine-Viale.pdf, Accessed: 2017-5-28.

[6] Venkatasubban, C. S., "A New Finite Element Formulation for ALE (Arbitrary Lagrangian Eulerian) Compressible Fluid Mechanics," *International Journal of Engineering Science*, Vol. 33, No. 12, 1995, pp. 1743–1762.

[7] Newman, J. C. and Anderson, W. K., "Investigation of Unstructured Higher-Order Methods for Unsteady Flow and Moving Domains," AIAA Paper 2015-2917, 2015.

[8] Anderson, W. K., Wang, L., Kapadia, S., Tanis, C., and Hilbert, B., "Petrov–Galerkin and Discontinuous-Galerkin Methods for Time-Domain and Frequency-Domain Electromagnetic Simulations," *Journal of Computational Physics*, Vol. 230, No. 23, 2011, pp. 8360–8385.

[9] Anderson, W. K., Ahrabi, B. R., and Newman, J. C., "Finite-Element Solutions for Turbulent Flow over the NACA 0012 Airfoil," *AIAA Journal*, Vol. 54, No. 9, Sept. 2016, pp. 2688–2704.

[10] Wang, L., Anderson, W. K., Erwin, J. T., and Kapadia, S., "Discontinuous-Galerkin and Petrov-Galerkin Methods for Compressible Viscous Flows," *Computers and Fluids*, Vol. 100, No. 1, 2014, pp. 13–29.

[11] Cash, J., "The Integration of Stiff Initial Value Problems in ODEs Using Modified Extended Backward Differentiation Formulae," *Computers & Mathematics with Applications*, Vol. 9, No. 5, 1983, pp. 645–657.

[12] Cash, J., "Modified Extended Backward Differentiation Formulae for the Numerical Solution of Stiff Initial Value Problems in ODEs and DAEs," *Journal of Computational and Applied Mathematics*, Vol. 125, No. 1, 2000, pp. 117–130.

[13] Psihoyios, G., "A General Formula for the Stability Functions of a Group of Implicit Advanced Step-Point (IAS) Methods," *Mathematical and Computer Modelling*, Vol. 46, No. 1, 2007, pp. 214–224.

[14] Hairer, E. and Wanner, G., *Solving Ordinary Differential Equations II*, Springer-Verlag, 1993.

[15] Eça, L., Hoekstra, M., et al., "Code Verification of Unsteady Flow Solvers with the Method of Manufactured Solutions," *The Seventeenth International Offshore and Polar Engineering Conference*, International Society of Offshore and Polar Engineers, 2007.

[16] Lesoinne, M. and Farhat, C., "Geometric Conservation Laws for Flow Problems with Moving and Deformable Meshes, and their Impact on Aeroelastic Computations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 134, No. 1-2, 1996, pp. 71–90.

[17] Allmaras, S. R., Johnson, F. T., and Spalart, P. R., "Modifications and Clarifications for the Implementation of the Spalart-Allmaras Turbulence Model," *Seventh International Conference on Computational Fluid Dynamics (ICCFD7)*, 2012.

[18] Spalart, P. R. and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *La Recherche Aérospatiale*, Vol. 1, 1994, pp. 5–21.

[19] White, F. M. and Corfield, I., *Viscous Fluid Flow*, Vol. 3, McGraw-Hill New York, 2006.

[20] Bonhaus, D. L., *A Higher Order Accurate Finite Element Method for Viscous Compressible Flows*, Ph.D. thesis, Virginia Polytechnic Institute and State University, 1998.

[21] Barth, T. J., "Numerical Methods for Gasdynamic Systems on Unstructured Grids," *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws*, edited by D. Kroner, M. Ohlberger, and C. Rohde, Lecture Notes in Computational Science and Engineering, Springer, 1998, pp. 195–285.

[22] Cockburn, B., Hou, S., and Shu, C.-W., "The Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws. IV. The Multidimensional Case," *Mathematics of Computation*, Vol. 54, No. 190, 1990, pp. 545–581.

[23] Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics*, Vol. 43, No. 2, Oct. 1981, pp. 357–372.

[24] Currie, I. G., *Fundamental Mechanics of Fluids*, McGraw-Hill Book Company, New York, 1974.

[25] Mavriplis, D. J. and Nastase, C. R., "On the Geometric Conservation Law for High-Order Discontinuous Galerkin Discretizations on Dynamically Deforming Meshes," AIAA Paper 2008–778, 2008.

[26] Nobile, F. and Formaggia, L., "A Stability Analysis for the Arbitrary Lagrangian : Eulerian Formulation with Finite Elements," *East-West Journal of Numerical Mathematics*, Vol. 7, No. 2, 1999, pp. 105–132.

[27] Segel, L. A., *Mathematics Applied to Continuum Mechanics*, Dover Publications, 1961.

[28] Förster, C., Wall, W. A., and Ramm, E., "On the Geometric Conservation Law in Transient Flow Calculations on Deforming Domains," *International Journal for Numerical Methods in Fluids*, Vol. 50, No. 12, 2006, pp. 1369–1379.

[29] Chang, X., Ma, R., Zhang, L., He, X., and Li, M., "Further Study on the Geometric Conservation Law for Finite-Volume Method on Dynamic Unstructured Mesh," *Computers and Fluids*, Vol. 120, No. 4, Feb. 2015, pp. 98–110.

[30] Visbal, M. R. and Gaitonde, D. V., "On the Use of Higher-Order Finite-Difference Schemes on Curvilinear and Deforming Methods," *Journal of Computational Physics*, Vol. 181, No. 1, 2002, pp. 155–185.

[31] Biedron, R. T. and Thomas, J. L., "Recent Enhancements to the FUN3D Flow Solver for Moving-Mesh Applications," AIAA Paper 2009–1360, 2009.

[32]Morton, S. A., Melville, R. B., and Visbal, M. R., "Accuracy and Coupling Issues of Aeroelastic Navier-Stokes Solutions on Deforming Meshes," AIAA Paper 1997–1085, 1997.

[33]Saad, Y. and Schultz, M. H., "GMRES: A Generalized Minimum Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal of Scientific and Statistical Computing*, Vol. 7, 1986, pp. 856–869.

[34]Saad, Y., *Iterative Methods for Sparse Linear Systems*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd ed., 2003.

[35]Venkatakrishnan, V., Allmaras, S. R., Johnson, F. T., and Kamenetskii, D. S., "Higher Order Schemes for the Compressible Navier-Stokes Equations," AIAA Paper 2003–3987, 2003.

[36]Barth, T. J., "Analysis of Implicit Local Linearization Techniques for Upwind and TVD Algorithms," AIAA Paper 1987–0595, 1987.

[37]Couchman, B. L., Darmofal, D. L., Allmaras, S. R., and Galbraith, M. C., "On the Convergence of Higher-Order Finite Element Methods to Weak Solutions," AIAA Paper 2017–4274, 2017.

[38]Persson, P.-O., Bonet, J., and Peraire, J., "Discontinuous Galerkin Solution of the Navier-Stokes Equations on Deformable Meshes," *Computer Methods in Applied Mechanics and Engineering*, Vol. 198, No. 17–20, April 2009, pp. 1585–1595.

[39]Chwalowski, P. and Heeg, J., "FUN3D Analysis in Support of the Second Aeroelastic Prediction Workshop," AIAA Paper 2016–3122, 2016.

[40]Chwalowski, P., Heeg, J., and Biedron, R. T., "Numerical Investigations of the Benchmark Supercritical Wing in Transonic Flow," AIAA Paper 2017–0190, 2017.