Toward a Third-Order Accurate, Second-Derivative-Free, Shock-Capturing Finite-Volume Method for Hypersonic Flows on Tetrahedral Grids

Hiroaki Nishikawa*
National Institute of Aerospace, Hampton, VA 23666, USA

Jeffery A. White† and Matthew D. O'Connell‡
NASA Langley Research Center, Hampton VA 23681, USA

In this paper, we report progress in the development of a third-order accurate, second-derivative-free, shock-capturing finite-volume solver for three-dimensional unstructured grids. The method is economical in the sense that the computation and storage of second derivatives are not required for third-order accuracy. It is based on point-valued numerical solutions stored at cells, gradients computed and stored at nodes, and an efficient projected-derivative formula that eliminates the need for second derivatives in a quadratic solution interpolation. The projected-derivative formula is also used to eliminate second derivatives from a high-order flux quadrature formula, so that it can be implemented conveniently in the form of a numerical flux at a face center plus a correction term. Similarly, a high-order source quadrature formula can also be implemented in the form of a cell-center point evaluation plus a similar correction term. These features make it relatively straightforward to extend an existing second-order finite-volume code to third-order. This paper reports progress of implementing the method in the NASA VULCAN-CFD code and discusses the implementation of a high-order accurate limiter for shock capturing.

1. Introduction

Recently, a unique and economical third-order cell-centered finite-volume (CCFV) discretization method has been developed for tetrahedral grids based on an efficient quadratic solution interpolation scheme utilizing nodal gradients [1]. This method is very economical compared with conventional third-order CCFV unstructured-grid discretization methods [2–9] in that there is no need to compute and store second derivatives of solution variables and it also only requires a single numerical flux per face. Also, since we store numerical solutions as point values at cell centroids as in the QUICK scheme [10,11] or deconvolution finite-volume methods [12–14], there is no need to compute and store geometric moments as typically required for a quadratic solution reconstruction in finite-volume methods with cell averaged solutions. In this third-order method, we only need a quadratic solution interpolation.

This third-order method is constructed as an extension of the face-averaged nodal-gradient (FANG) CCFV method [15–17], where solution gradients are computed and stored at nodes rather than cells. Having gradients at nodes, we can compute second derivatives in a cell by applying the Green-Gauss theorem. Such second derivatives can be expressed in terms of the nodal gradients, and therefore, a quadratic solution interpolation within a cell can be expressed entirely by the gradients stored at its nodes and the solution value stored at the cell, eliminating the need to compute and store second derivatives. This technique can be considered as an extension of the linear solution interpolation formula proposed in Refs. [18,19], where

*Research Fellow, AIAA Associate Fellow
†Senior Research Scientist, Computational AeroSciences Branch, AIAA Associate Fellow
‡Research Scientist, Computational AeroSciences Branch, AIAA Member

a solution is linearly interpolated at a face centroid of a tetrahedral cell using the solution values recovered at nodes from cell center solutions. These economical interpolation schemes are based on a general efficient projected-derivative formula that gives a projected gradient of a function along a direction from a cell center to an arbitrary point. This projected-derivative formula is expressed in terms of only function values available at nodes, not requiring the gradient. Applying the general efficient projected-derivative formula to the gradient term in a linear interpolation, we eliminate the gradient and obtain the efficient linear interpolation scheme in Refs. [18, 19]. Similarly, applying it to the curvature term in a quadratic interpolation formula, we can express all the second derivatives in terms of gradients stored at nodes and obtain the efficient quadratic solution interpolation scheme [1]. The resulting third-order discretization method is especially attractive for practical second-order unstructured-grid computational fluid dynamics (CFD) codes. It is relatively simple to implement in such codes, allowing the use of many existing algorithmic components (e.g., least-squares methods, linear and nonlinear solvers). It also achieves third-order accuracy with linear tetrahedral grids even for curved geometries, similarly to other economical third-order methods [20, 21]. Moreover, as we will show, the third-order discretization can be solved robustly by an implicit defect-correction solver, which is widely employed and available in practical second-order CFD solvers. Therefore, the development of a new nonlinear solver is not a prerequisite for this new third-order discretization.

In this paper, we report progress in the development of the economical third-order CCFV solver for tetrahedral grids with an emphasis for application toward hypersonic-flow simulations and fully automated simulations utilizing anisotropic tetrahedral grid adaptation [22–24]. To this end, we implemented the third-order discretization in the NASA VULCAN-CFD code [17], which is a fully-parallelized structured/unstructured-grid hypersonic viscous-flow solver developed at the NASA Langley Research Center. Due to our interest in hypersonic chemically reacting flow applications, it is necessary to develop a robust shock capturing approach. For third-order accuracy, a limiter function must also be designed to preserve third-order accuracy in smooth regions. In a recent paper [25], a family of high-order differentiable limiter functions that can preserve up to fifth-order accuracy was derived, as an extension of Venkatakrishnan's limiter [26]. In this work, we employ Nishikawa's R_5 limiter function that preserves up to fifth-order accuracy, which is expected to serve as a sufficiently accurate and less dissipative limiter for the third-order scheme than those preserving third-order accuracy [25]. This paper discusses the implementation of the high-order limiter and presents preliminary results.

2. Target Governing Equations

In this paper, we consider the Euler equations:

$$\int_V \frac{\partial \mathbf{u}}{\partial t}\, dV + \oint_{\partial V} \mathcal{F}\hat{\mathbf{n}} ds = \int_V \mathbf{s}\, dV, \tag{2.1}$$

where V and ∂V denote a control volume and its boundary, respectively, $\mathbf{s}$ is a vector of source terms, $\hat{\mathbf{n}}$ is a unit outward normal vector of the infinitesimal surface area ds,

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho\mathbf{v} \\ \rho E \end{bmatrix}, \quad \mathcal{F} = \begin{bmatrix} \rho\mathbf{v} \\ \rho\mathbf{v}\otimes\mathbf{v} + p\mathbf{I} \\ \rho\mathbf{v}H \end{bmatrix}, \tag{2.2}$$

where $\otimes$ denotes the dyadic product, $\mathbf{I}$ is the 3×3 identity matrix, ρ is the density, $\mathbf{v} = (u, v, w)^t$ is the column vector of the velocity (the superscript t denotes transpose) with Cartesian components u, v, and w, p is the static pressure, and $H = \gamma p/\{\rho(\gamma-1)\} + \mathbf{v}\cdot\mathbf{v}/2$ is the specific total enthalpy with $\gamma = 1.4$ (air), E is the specific total energy, $E = H - p/\rho$, and $\mathbf{s}$ is a source term vector. The system is nondimensionalized by free stream values in a similar manner as described in Ref. [27] and closed by the nondimensionalized ideal gas law:

$$p = \rho T/\gamma, \tag{2.3}$$

where T is the static temperature.

3. Economical Third-Order Cell-Centered Finite-Volume Method for Tetrahedral Grids

3.1. Discretization

At a cell j on a tetrahedral grid, the economical third-order CCFV discretization of the Euler equations can be written as [1]

$$V_j \frac{\partial \mathbf{u}_j}{\partial t} + \sum_{k \in \{k_j\}} \Phi_{jk}(\mathbf{w}_L, \mathbf{w}_R)|\mathbf{n}_T| - \mathbf{s}_j V_j = \delta \mathbf{f}_j + \delta \mathbf{s}'_j, \tag{3.1}$$

where $\delta \mathbf{f}_j$ and $\delta \mathbf{s}'_j$ are given by

$$\delta \mathbf{f}_j = -\frac{1}{24} \sum_{k \in \{k_j\}} \sum_{i_k=1}^{3} \left(\frac{\partial \mathbf{f}}{\partial \mathbf{w}} \right)_{i_k} \nabla \mathbf{w}_{i_k} (\mathbf{x}_{i_k} - \mathbf{x}_T)|\mathbf{n}_T|, \quad \delta \mathbf{s}'_j = \frac{1}{40} \sum_{i=1}^{4} \nabla \left[\mathbf{s}_i - \left(\frac{\partial \mathbf{u}_i}{\partial t} \right) \right] (\mathbf{x}_i - \mathbf{x}_j) V_j. \tag{3.2}$$

Here, $\mathbf{u}_j$ and $\mathbf{s}_j$ are point values of the conservative variable vector and the source term at the cell centroid $\mathbf{x}_j$, Φ_{jk} is a numerical flux at the centroid $\mathbf{x}_T$ of the triangular face T between the cell j and its face neighbor cell $k \in \{k_j\}$, V_j is the volume of the cell j, $\{k_j\}$ is a set of neighbor cells, $\mathbf{f} = \mathcal{F}\mathbf{n}_T/|\mathbf{n}_T|$ is a flux projected along the face normal direction, $\mathbf{n}_T$ is an outward normal vector of the face T, $|\mathbf{n}_T|$ is the area of the face, $\mathbf{w}$ denotes the vector of primitive variables $\mathbf{w} = (\rho, \mathbf{v}, p)$, i_k denotes a node of the face T, and $\mathbf{x}_{i_k}$ is the position of the node i_k, and i in $\delta \mathbf{s}'_j$ denotes a node of the cell j. For the numerical flux $\Phi_{jk}(\mathbf{w}_L, \mathbf{w}_R)$, we employ the HLLC flux [28, 29], and the left and right states, $\mathbf{w}_L$ and $\mathbf{w}_R$, are computed by an efficient quadratic interpolation scheme discussed in the next section.

The left hand side of Eq. (3.1) is a typical second-order finite-volume discretization, except that a quadratic solution interpolation is used here for computing the left and right values that are passed to the numerical flux. The first term on the right hand side, $\delta \mathbf{f}_j$, is a flux correction term required for quadratic exactness of the flux integral, and the second term, $\delta \mathbf{s}'_j$, consists of correction terms for the source and time-derivative terms. These forms of high-order surface and volume integration formulas have been derived in Ref. [1], by following the k-exact finite-volume discretization approach [9,30–32]. They are useful because we can upgrade an existing second-order code to third-order accuracy just by adding an efficient quadratic interpolation scheme to the construction of the numerical flux and adding these correction terms. Note that these correction terms do not require second derivatives, which naturally appear when a quadratic polynomial is integrated over a cell but can be eliminated completely by utilizing the efficient projected-gradient formula derived in Ref. [1], as described in the next section.

3.2. Quadratic Solution Interpolation without Second Derivatives

In Ref. [18], Frink introduced a simple formula for linearly interpolating a solution at a face centroid in a tetrahedron: e.g., for a scalar variable g,

$$g_T = g_j + \frac{1}{4} \left(\frac{g_2 + g_3 + g_4}{3} - g_j \right), \tag{3.3}$$

where g_T denotes the linearly interpolated value of g at the centroid $\mathbf{x}_T$ of the face opposite to the node 1, g_2, g_3, g_4 are the values of g at the node computed from cell-center solutions, 2, 3, and 4, respectively, as shown in Figure 1. This formula is equivalent to the linear interpolation with the Green-Gauss gradient:

$$g_T = g_j + \nabla g_j^{GG} \cdot (\mathbf{x}_T - \mathbf{x}_j), \tag{3.4}$$

where ∇g_j^{GG} is the Green-Gauss gradient expressed in terms of the nodal values. As discussed in Ref. [1], the projected derivative $\nabla g_j^{GG} \cdot (\mathbf{x}_T - \mathbf{x}_j)$ can be written as

$$\nabla g_j^{GG} \cdot (\mathbf{x}_T - \mathbf{x}_j) = \overline{g}_T - \overline{g}_j, \tag{3.5}$$

where $\overline{g}_T$ and $\overline{g}_j$ are the function values linearly interpolated at the geometric centroids of the triangular face T and the tetrahedron j, respectively:

$$\overline{g}_T = \frac{g_2 + g_3 + g_4}{3}, \quad \overline{g}_j = \frac{g_1 + g_2 + g_3 + g_4}{4}, \tag{3.6}$$

where the node 1 is a node located opposite to the face T as in Figure 1. It is then generalized in Ref. [1] as

$$\nabla g_j^{GG} \cdot (\mathbf{x} - \mathbf{x}_j) = g - \overline{g}_j, \tag{3.7}$$

where g is a value computed from the nodal values in a linearly exact manner at an arbitrary point $\mathbf{x}$. A similar formula can be derived for a triangular face T:

$$\nabla g_T^{GG} \cdot (\mathbf{x} - \mathbf{x}_T) = g - \overline{g}_T, \tag{3.8}$$

where ∇g_T^{GG} is the Green-Gauss gradient defined over the triangular face T. Observe that these formulas suggest how we can eliminate the gradient from the projected derivatives, expressing them in terms of the nodal values only. In the economical third-order method, we apply these formulas to eliminate second derivatives from a curvature term.

To accomplish this elimination of the second derivatives, consider a quadratic interpolation for the density ρ:

$$\rho_L = \rho_j + \overline{\nabla\rho}_j \cdot (\mathbf{x}_T - \mathbf{x}_j) + \frac{1}{2}(\mathbf{x}_T - \mathbf{x}_j)^t \nabla^2 \rho_j (\mathbf{x}_T - \mathbf{x}_j), \tag{3.9}$$

where ρ_L denotes the value quadratically interpolated from the centroid of the cell j to the centroid of the face T, $\overline{\nabla\rho}_j$ is the arithmetic average of the gradients stored at nodes of the cell j. In the FANG CCFV method, we compute and store gradients of the primitive variables at the nodes [15–17]. For third-order accuracy, we compute the gradient of the primitive variables by a quadratic least-squares method using an augmented stencil of cells around each node. See Ref. [1] for further details. By applying the projected-derivative formula (3.7) to the gradient, i.e., $g \leftarrow \nabla\rho$, we obtain

$$\rho_L = \rho_j + \overline{\nabla\rho}_j \cdot (\mathbf{x}_T - \mathbf{x}_j) + \frac{1}{2}\left(\overline{\nabla\rho}_T - \overline{\nabla\rho}_j\right) \cdot (\mathbf{x}_T - \mathbf{x}_j), \tag{3.10}$$

where $\overline{\nabla\rho}_T$ is the arithmetic average of the gradients available at nodes of the face T, and therefore,

$$\rho_L = \rho_j + \frac{1}{2}\left(\overline{\nabla\rho}_T + \overline{\nabla\rho}_j\right) \cdot (\mathbf{x}_T - \mathbf{x}_j). \tag{3.11}$$

This is the efficient quadratic interpolation formula, called nodal-gradient quadratic interpolation (NGQI) that does not require second derivatives [1]. A similar formula can be applied to the neighbor cell k to obtain the right state ρ_R. Applying these formulas, we obtain the left and right states, $\mathbf{w}_L$ and $\mathbf{w}_R$, required for the numerical flux. A similar formula is applied also to quadratically interpolate the value at a node i_k from a cell:

$$\rho_{i_k}^L = \rho_j + \frac{1}{2}\left(\overline{\nabla\rho}_{i_k} + \overline{\nabla\rho}_j\right) \cdot (\mathbf{x}_{i_k} - \mathbf{x}_j), \tag{3.12}$$

and

$$\rho_{i_k}^R = \rho_k + \frac{1}{2}\left(\overline{\nabla\rho}_{i_k} + \overline{\nabla\rho}_k\right) \cdot (\mathbf{x}_{i_k} - \mathbf{x}_k), \tag{3.13}$$

and their average is used to evaluate the flux Jacobian in the flux correction $\delta\mathbf{f}_j$. In a similar manner, second derivatives arising from high-order flux and source quadrature formulas can be eliminated by using the efficient projected-derivative formulas (3.7)and (3.8). See [1] for details.

4. Shock Capturing and High-Order Slope Limiter

For shock capturing, a mechanism for robustly producing non-oscillatory solutions is necessary. In this work, we will employ the unstructured-grid slope-limiter framework originally developed by Barth and Jespersen [33] and later improved by Venkatakrishnan [26], Michalak and Ollivier-Gooch [34], and more recently by Nishikawa [25]. For the density ρ, we implement the limiter in the form:

$$\rho_L = \rho_j + \frac{\phi_j^{(\rho)}}{2}\left(\overline{\nabla\rho}_T + \overline{\nabla\rho}_j\right) \cdot (\mathbf{x}_T - \mathbf{x}_j), \tag{4.1}$$

which is the limited version of NGQI in Eq. (3.11), $\phi_j^{(\rho)}$ is a limiter function for the density, which can have a value between 0 and 1, computed by Nishikawa's R_5 limiter [25]:

$$\phi_j^{(\rho)} = \min_{T\in\{k_j\}} \begin{cases} \dfrac{a^5+\epsilon^5+aS_5}{a^5+\epsilon^5+b(a^4+S_5)}, & \text{if } a > 2b, \\ 1, & \text{otherwise,} \end{cases} \tag{4.2}$$

where T is a face between the cell j and a face-neighbor $k \in \{k_j\}$,

$$S_5 = 8b^2[a^2 - 2b(a-b)], \quad a = |\Delta_+|, \quad b = |\Delta_-| = |\rho_L - \rho_j|, \quad \epsilon^5 = \left\{K\overline{\Delta x}\right\}^6 \overline{\rho}^5, \tag{4.3}$$

$$\Delta_+ = \begin{cases} \rho_{max} - \rho_j, & \rho_L \geq \rho_j, \\ \rho_{min} - \rho_j, & \rho_L < \rho_j, \end{cases} \tag{4.4}$$

ρ_{min} and ρ_{max} are found from a set of face neighbor cells, and here K is an adjustable constant and $\overline{\Delta x}$ is a local mesh spacing defined in a three-dimensional tetrahedral cell as

$$\overline{\Delta x} = \left(6\sqrt{2}\,V_j\right)^{1/3}, \tag{4.5}$$

which corresponds to the side length of a regular tetrahedron having the volume V_j. Following Ref. [35], we have introduced the scaling factor $\overline{\rho}$ in the definition of ϵ for dimensional consistency, using a local value stored at the cell j; but for the x-velocity u, we set $\max(|u_j|, a_j)$ with the local speed of sound a_j, and similarly for the other velocity components v and w. As discussed and demonstrated in a recent paper [25], Venkatakrishnan's limiter is useful for second-order schemes but cannot preserve the design order of accuracy of third- and higher-order schemes. On the other hand, the R_5 limiter is designed to preserve up to fifth-order accuracy and serve as a less dissipative limiter for second-, third-, and fourth-order schemes [25]. To deal with strong shocks, we employ Gnoffo's pressure limiter ϕ_j^{press} [36], which smoothly switches from 0 to 1 depending on the ratio of the maximum to the minimum pressure in a set of neighbor cells at a cell j, and apply it to the limiters for all the variables:

$$\phi_j^{(\rho)} \leftarrow \phi_j^{\text{press}}\phi_j^{(\rho)}, \quad \phi_j^{(u)} \leftarrow \phi_j^{\text{press}}\phi_j^{(u)}, \quad \phi_j^{(v)} \leftarrow \phi_j^{\text{press}}\phi_j^{(v)}, \quad \phi_j^{(w)} \leftarrow \phi_j^{\text{press}}\phi_j^{(w)}, \quad \phi_j^{(p)} \leftarrow \phi_j^{\text{press}}\phi_j^{(p)}. \tag{4.6}$$

If there exists a large pressure ratio in the set of neighbor cells, we have $\phi_j^{\text{press}} = 0$ and replace all limiters. For unsteady problems, it is necessary to apply a limiter to the mass matrix coupling the time derivatives with neighbors, which arises due to the source correction term $\delta\mathbf{s}_j$. Later, we will present some preliminary results obtained for the linear advection equation in one dimension.

For the flux correction term, we apply the limiter to both the solution interpolation to a node and the projection of the node primitive variable gradients to the cell face geometric center:

$$\mathbf{w}_{i_k}^L = \mathbf{w}_j + \frac{1}{2}\mathbf{D}_\phi \left\{ \left(\overline{\nabla \mathbf{w}}_{i_k} + \overline{\nabla \mathbf{w}}_j \right) (\mathbf{x}_{i_k} - \mathbf{x}_j) \right\}, \tag{4.7}$$

and

$$\mathbf{w}_{i_k}^R = \mathbf{w}_k + \frac{1}{2}\mathbf{D}_\phi \left\{ \left(\overline{\nabla \mathbf{w}}_{i_k} + \overline{\nabla \mathbf{w}}_k \right) (\mathbf{x}_{i_k} - \mathbf{x}_k) \right\}, \tag{4.8}$$

$$\delta\mathbf{f}_j = -\frac{1}{24} \sum_{k\in\{k_j\}} \sum_{i_k=1}^{3} \left(\frac{\partial \mathbf{f}}{\partial \mathbf{w}} \right)_{i_k} \mathbf{D}_\phi \left\{ \nabla \mathbf{w}_{i_k} (\mathbf{x}_{i_k} - \mathbf{x}_T) \right\} |\mathbf{n}_T|, \tag{4.9}$$

where the flux Jacobian $\left(\frac{\partial \mathbf{f}}{\partial \mathbf{w}}\right)_{i_k}$ is evaluated at $(\mathbf{w}_{i_k}^L + \mathbf{w}_{i_k}^R)/2$, and

$$\mathbf{D}_\phi = \text{diag}\left(\phi_{i_k}^{(\rho)}, \phi_{i_k}^{(u)}, \phi_{i_k}^{(v)}, \phi_{i_k}^{(w)}, \phi_{i_k}^{(p)} \right), \tag{4.10}$$

$$\phi_{i_k}^{(\rho)} = \min\left(\phi_j^{(\rho)}, \phi_k^{(\rho)}\right), \ \phi_{i_k}^{(\rho)} = \min\left(\phi_j^{(u)}, \phi_k^{(u)}\right), \ \phi_{i_k}^{(\rho)} = \min\left(\phi_j^{(v)}, \phi_k^{(v)}\right), \tag{4.11}$$

$$\phi_{i_k}^{(\rho)} = \min\left(\phi_j^{(w)}, \phi_k^{(w)}\right), \ \phi_{i_k}^{(\rho)} = \min\left(\phi_j^{(p)}, \phi_k^{(p)}\right). \tag{4.12}$$

The limiting strategy described in the above is just one possible approach, and there is certainly room for improvements. For example, we are exploring a way of eliminating ϵ by detecting smooth extrema and constant regions in some other way along the same lines of the methods in Refs. [37,38]. We are also exploring a robust way of triggering a limiter freeze for deeper iterative convergence. For unsteady problems, the time derivative terms are coupled with neighbor values through a mass matrix and it has to be limited as well. The mass-matrix limiting is discussed later for linear advection, and it will require a careful extension to the Euler equations because the limiters are computed for the primitive variables but the time derivatives have the conservative variables.

5. Results

In this section, we present preliminary results obtained with the third-order method implemented in the VULCAN-CFD code, and also a study on the mass-matrix limiting for the linear advection equation in one dimension. For steady problems, we solve a global system of nonlinear residual equations by an implicit defect-correction solver with the exact residual Jacobian of a first-order accurate residual and the multi-color Gauss-Seidel relaxation scheme is applied to relax the linearized system (see Refs. [39] for example for further details).

5.1. Accuracy Verification

First, we consider an accuracy verification test using the method of manufactured solutions [40]. We define the following functions:

$$\mathbf{w}^{exact} = \mathbf{w}_0 + \mathbf{w}_1 \exp(0.5(x + y + z)), \tag{5.1}$$

where $\mathbf{w}_0 = (1.0, 0.3, 0.2, 0.1, 1.0)$ and $\mathbf{w}_1 = (0, 1, 0.1, 0.1, 0.1, 0.1)$, and make them the exact solutions by introducing the source term $\mathbf{s}$ defined by

$$\mathbf{s} = \mathrm{div}\mathcal{F}(\mathbf{w}^{exact}), \tag{5.2}$$

which we compute numerically at each cell centroid and then add to the correction term (3.2), computing the gradient at nodes by the quadratic least-squares method. For the source correction, we compute the gradient of $\mathbf{s}$ numerically by the quadratic least-squares method that we already have for the discretization. To determine the order of discretization errors, we solve the residual equations for the steady Euler equations in a unit cube domain with a weak Dirichlet condition, where the right state is given by the exact solution (5.1), using a set of consistently refined perturbed tetrahedral grids of 3,072, 10,368, 24,576, 48,000, and the 118,098 cells. See Figure 2(a) for the coarsest grid and contours of the exact solution for the pressure. L_1 error convergence results are shown in Figure 2(b). Here, the effective mesh spacing h_{eff} has been computed as the average of the cubic root of the cell volumes. As can be seen, third-order accuracy is observed for all variables. Similar results have been obtained in the L_∞ norm also. The results verify the implementation of the third-order accurate inviscid discretization including the high-order source quadrature in VULCAN-CFD.

5.2. Smooth Bump

Next, we consider compressible flows over a smooth bump through a duct, where a smooth bump defined by

$$z_b = 0.05 \sin^4\left[\frac{\pi(10x - 3)}{9}\right], \quad x \in [0.3, 1.2], \tag{5.3}$$

is placed at the bottom of a rectangular domain $(x, y, z) = [-0.25, 1.75] \times [-0.5, 0.5] \times [0, 1]$. We solve the steady residual equations without source terms over three levels of irregular tetrahedral grids with 9,216, 31,104, and 367,416 tetrahedra (see Figure 3(a) for the coarsest grid).

We first consider a calorically perfect subsonic flow at $M_\infty = 0.3$, and verify third-order accuracy using the entropy error defined as $|\gamma p/\rho^\gamma - 1|$. For this problem, a slip-wall condition has been implemented with the boundary face normal vector defined as the arithmetic average of the exact wall normal vectors analytically computed at boundary nodes. The slip boundary condition is applied at the bottom boundary including the bump and also at the top boundary at $z = z_{max}$, a characteristic condition is applied at the inflow boundary located at $x = x_{min}$, a constant static pressure condition is applied at the outflow located at $x = x_{max}$, and a periodic condition is applied at the remaining boundaries at $y = y_{min}$ and $y = y_{max}$. As discussed and demonstrated in Ref. [1], this boundary condition gives third-order accuracy with linear tetrahedral grids for curved geometries. L_1 entropy error convergence results are shown in Figure 3(c). Here, we compare four schemes: NGLI(1) is a linear interpolation scheme with the gradient given by the arithmetic average of linear least-squares gradients at nodes, NGLI(2) is the same linear interpolation scheme with quadratic least-squares gradients, NGQI(2) is the efficient quadratic interpolation scheme but without the flux correction, NGQI(2)+FC is the third-order scheme. As expected, NGQI(2)+FC achieves third-order accuracy. NGQI(2) gives similar error convergence but begins to deteriorate on the finest grid. Other schemes are second-order accurate as expected.

Next, we consider a hypersonic flow over the same bump at a free stream Mach number 6. For comparison, we perform the computation with a conventional second-order scheme with Venkatakrishnan's limiter. The same defect-correction solver is used to solve both the second- and third-order discretization residuals, with the limiters frozen after 400 iterations. For both schemes, we use the value of $K = 5$ and have been able to converge the residuals by more than 10 orders of magnitude down to a machine-zero level. Residual convergence is shown in Figure 4(c) for the second-order scheme and Figure 4(d) for the third-order scheme. As can be seen, the defect-correction with a first-order Jacobian, which has been employed in many practical second-order CFD codes, is equally effective for the third-order discretization. This is expected because the third-order discretization is not very different from the baseline second-order discretization. Therefore, there is no need to develop a more advanced solver for the third-order scheme. However, we expect a stronger solver such as a Jacobian-free Newton-Krylov solver would be necessary to solve problems with anisotropically adapted tetrahedral grids. Then, the current defect-correction solver is expected to serve as an effective preconditioner. Figures 4(c) and 4(d) show pressure line contours plotted over color contours of the limiter values for the pressure, for the second- and third-order schemes, respectively. As can be seen, the limiter is less active in the third-order solution. This is expected because the R_5 limiter is less dissipative as demonstrated in Ref. [25] and also because a linear interpolation tends to create greater overshoots/undershoots than a quadratic interpolation Ref. [38].

5.3. A Hypersonic Flow over a Sphere

As a robustness test, we consider a more difficult case involving a stronger shock and a highly-stretched grid: a hypersonic flow over a sphere at a free stream Mach number of 6 on an adaptive tetrahedral grid. Figure 5 shows an adapted grid with 285,227 tetrahedra generated by the refine package (`https://github.com/nasa/refine`) developed at NASA Langley Research Center. A far-field boundary condition is applied at the far-field boundary, a third-order extrapolation is applied a the x_{max} outflow boundary, the slip condition is applied at the sphere surface, and a symmetry condition is applied a the y_{min} boundary. As in the previous case, we compare the third-order method with the R_5 limiter and a conventional second-order scheme with Venkatakrishnan's limiter. For this problem, the limiters are not frozen and the calculations are performed until the residuals are reduced at least two orders of magnitude. Figures 6(c) and 6(d) show the results for a relatively small value of K, $K = 1.5$: pressure line contours and limiter contours. As can be seen, the limiter is activated in a wider range of the domain in the second-order solution than in the third-order solution. Also, the pressure line contours look slightly smoother, especially over the sphere surface near the outflow. Increasing K to $K = 5.0$, we obtained the results shown in Figures 6(c) and 6(d). As expected, the limiters are much less activated in both cases, but we still observe that the limiter is activated in a wider range of the domain in the second-order solution than in the third-order solution. Slight oscillations are observed around the bow shock, but this is expected for a large K. Nevertheless, these results demonstrate that the defect-correction solver, which is originally designed and widely employed for a second-order method, works robustly for the third-order method.

5.4. Unsteady Advection in One Dimension

So far, we have only considered steady problems. For unsteady problems, it is necessary to apply some limiting to the time derivatives for both explicit and implicit schemes, or oscillations will occur because the time derivatives are discretized in space by a high-order quadrature formula, and therefore, the scheme will not reduce entirely to a first-order scheme when a limiter is zero unless the spatial discretization of the time derivatives is suitably modified. Since the high-order quadrature formula results in a global coupling of the time derivative of a cell with its neighbors through a mass matrix, we consider limiting the mass matrix for monotonicity. Here, we present a basic study for linear advection in one dimension, which is sufficient for demonstrating the impact of the mass matrix limiting.

Consider the linear advection equation,

$$\partial_t u + c\,\partial_x u = 0, \tag{5.4}$$

where $c = 2$. The third-order spatial discretization on a uniform grid of spacing h is straightforward:

$$\int_{x_{j-1/2}}^{x_{j+1/2}} \frac{du}{dt} dx + [\Phi_{j+1/2} - \Phi_{j-1/2}] = 0, \tag{5.5}$$

where the numerical flux $\Phi_{j+1/2}$ is computed by the upwind flux evaluated with the one-dimensional version of the efficient quadratic interpolation scheme:

$$\Phi_{j+1/2} = cu_L = c\left[u_j + \frac{1}{2}\left\{(u_x)_{j+1/2} + \overline{(u_x)}_j\right\}\frac{h}{2}\right], \tag{5.6}$$

where $(u_x)_{j+1/2}$ is a nodal gradient at $j+1/2$ and $\overline{(u_x)}_j$ is the arithmetic average of the nodal gradients at $j-1/2$ and $j+1/2$:

$$(u_x)_{j-1/2} = \frac{u_j - u_{j-1}}{h}, \quad (u_x)_{j+1/2} = \frac{u_{j+1} - u_j}{h}, \quad \overline{(u_x)}_j = \frac{(u_x)_{j-1/2} + (u_x)_{j+1/2}}{2}. \tag{5.7}$$

Note that the flux correction is not needed in one dimension since the flux integration is exact. However, the time derivative needs to be integrated to higher-order:

$$\begin{aligned}
\int_{x_{j-1/2}}^{x_{j+1/2}} \frac{du}{dt} dx &= \int_{x_{j-1/2}}^{x_{j+1/2}} \left[\frac{du_j}{dt} + \frac{d(u_x)_j}{dt}(x - x_j) + \frac{1}{2}\frac{d(u_{xx})_j}{dt}(x - x_j)^2\right] dx \\
&= h\frac{du_j}{dt} + \frac{h^2}{24}\frac{d(u_{xx})_j}{dt} \\
&= h\frac{du_j}{dt} + \frac{h^2}{24}\left[\frac{d(u_x)_{j+1/2}}{dt} - \frac{d(u_x)_{j-1/2}}{dt}\right] \\
&= h\frac{du_j}{dt} + \frac{h}{24}\left[\frac{du_{j-1}}{dt} - 2\frac{du_j}{dt} + \frac{du_{j+1}}{dt}\right] \\
&= \frac{h}{24}\left[\frac{du_{j-1}}{dt} + 22\frac{du_j}{dt} + \frac{du_{j+1}}{dt}\right].
\end{aligned} \tag{5.8}$$

Therefore, the third-order scheme is given by

$$\frac{h}{24}\left[\frac{du_{j-1}}{dt} + 22\frac{du_j}{dt} + \frac{du_{j+1}}{dt}\right] + [\Phi_{j+1/2} - \Phi_{j-1/2}] = 0, \tag{5.9}$$

or in the global form:

$$\mathbf{M}\frac{d\mathbf{U}}{dt} + \mathbf{Res} = \mathbf{0}, \tag{5.10}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{U}$ is a global vector of the numerical solutions, and $\mathbf{Res}$ is a global residual vector. The spatial part of the resulting scheme is essentially the QUICK scheme [10], and the time-derivative

term with the mass matrix is equivalent to the coupled formulation of the QUICK scheme as described in Ref. [11]. This means that the economical third-order scheme can be considered as a unique extension of the QUICK scheme to unstructured grids based on point-valued solutions at cells and gradients computed and stored at nodes.

The time derivative is integrated in time by the third-order strong stability-preserving Runge-Kutta scheme [41], where the mass matrix is directly inverted at each stage. For comparison, we consider a second-order scheme defined by the following linear interpolation scheme:

$$\Phi_{j+1/2} = cu_L = c\left[u_j + \overline{(u_x)}_j \frac{h}{2}\right], \tag{5.11}$$

without the mass matrix $\mathbf{M}$. For limiting, we employ the R_5 limiter [25] with $\overline{\Delta x} = h$ and apply it in the form:

$$\Phi_{j+1/2} = c\left[u_j + \phi_j(u_L - u_j)\right], \tag{5.12}$$

where ϕ_j is the limiter computed at the cell j with the minimum and maximum values taken over the cell j and its two neighbor cells. The above limiting is applied in the same way to both the third- and second-order schemes. For the parameter K, we consider two different values: $K = 0.1$ and $K = 5.0$.

If the limiter is zero, the scheme is expected to reduce to a first-order scheme. The spatial discretization has been designed to achieve this, and the time derivative term needs to be designed similarly, such that the mass matrix will reduce to the identity matrix. In this study, we consider the following method of limiting the mass matrix:

$$\frac{h}{24}\left[\phi_{j-1/2}\frac{du_{j-1}}{dt} + (24 - \phi_{j-1/2} - \phi_{j+1/2})\frac{du_j}{dt} + \phi_{j+1/2}\frac{du_{j+1}}{dt}\right] + [\Phi_{j+1/2} - \Phi_{j-1/2}] = 0, \tag{5.13}$$

where

$$\phi_{j-1/2} = \frac{\phi_{j-1} + \phi_j}{2}, \quad \phi_{j+1/2} = \frac{\phi_j + \phi_{j+1}}{2}. \tag{5.14}$$

Note that the need to modify the mass matrix for shock capturing is well known in finite-element methods. See, for example, Refs. [42, 43].

Consider the initial solution profile as shown in Figure 7(a). We perform the time integration to the final time $t = 0.5$ with a periodic boundary condition (i.e., no boundaries): the initial profile will, therefore, come back to the initial position at $t = 0.5$. First, we compare unlimited schemes. Figure 7(b) shows the result of the second-order scheme (5.11). Figures 7(c) and 7(d) show results of the unlimited third-order scheme without the mass matrix and with the mass matrix, respectively. As can be seen, the third-order scheme preserves the symmetry of the square profile much better with the mass matrix, compared with the other two schemes. This is an expected feature of any third-order scheme because it is the leading dispersive error that is eliminated from a second order scheme [44]. The results indicate that the mass matrix is essential for the third-order scheme and at the same time the mass matrix will have to be limited for shock capturing.

Figure 8 shows the results obtained with the limiter. The parameter K is set to $K = 0.1$. Results of the second-order scheme and the third-order scheme without the mass matrix are shown in Figures 8(a) and 8(b), respectively. The wave forms are shown to be monotone, but the symmetry of each profile (square or sine) is not preserved very well, in comparison with the third-order solutions in Figures 8(c) and 8(d). Note finally that the solution exhibits slight overshoot and undershoot in Figure 8(c), which is a solution obtained with the original mass matrix. On the other hand, the solution is monotone in the case of the limited mass matrix, as can be seen in Figure 8(d).

As can be seen, the results in Figure 8 are dissipative at the top of the sine wave, which is a smooth extremum. To preserve the design order of accuracy at smooth extrema, we can increase the parameter K, but will have to expect some oscillations at discontinuities [26]. We performed additional calculations with $K = 2.0$, and obtained the results shown in Figure 9. Comparing the solutions in Figure 8, we can see that the solution is more accurate at the top of the sine wave, but oscillations appear near discontinuities. Small oscillations may be tolerated in some flow problems, but a strict monotonicity would be required for robust hypersonic-flow simulations. In this regard, Gnoffo's pressure limiter can be useful because it allows us to use a relatively large value of K while maintaining monotonicity at shock waves. However, it does not detect

contact discontinuities and can generate oscillations there. It is also difficult to determine the value of K for a given problem before running simulations, and moreover as originally formulated ϵ is not dimensionally consistent although a more sensible and dimensionally consistent formula has been proposed in Ref. [35]. To address these issues, we are currently developing some techniques for potentially eliminating the parameter ϵ and thus K, following the work presented in Refs. [37, 38].

6. Conclusion

In this paper, we reported progress in developing an economical third-order finite-volume scheme for tetrahedral grids, focusing on its implementation in a practical code and shock capturing. We have successfully implemented the third-order scheme in the NASA VULCAN-CFD code and verified the implementation by accuracy verification tests. Also, we implemented a limiter and performed computations of high-Mach-number flows. The R_5 limiter, which preserves up to fifth-order accuracy, has been implemented and tested for shock capturing. Numerical results to date show that the limited third-order solver is robust enough to produce accurate solutions for a hypersonic flow over a smooth bump using a regular tetrahedral grid and the much more challenging problem of a hypersonic flow over a sphere using a highly-stretched adapted tetrahedral grid. Finally, we derived a one-dimensional version of the proposed third-order scheme and tested it for a linear advection problem. Results indicate that limiting the mass matrix is essential to preserving the monotonicity.

Currently, our effort is focused on developing a more sophisticated limiting strategy that potentially eliminates the need for adjusting the parameter K in the limiter function. To achieve this, we will have to detect smooth extrema and constant regions, in a way similar to the techniques discussed in Refs. [37, 38]. Also, extensions to other cell types, which are possible, are left as important future work.

Acknowledgments

The authors gratefully acknowledge support by the Hypersonic Technology Project, through the Hypersonic Airbreathing Propulsion Branch of the NASA Langley Research Center. The first author was funded under Contract No. 80LARC17C0004.

References

[1]H. Nishikawa and J. A. White. An economical third-order cell-centered finite-volume method for tetrahedral grids: Conservative differential form, nodal gradients, and efficient quadratic reconstruction. 2022, in preparation.

[2]T. J. Barth and P. O. Frederickson. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. AIAA Paper 90-0013, 1990.

[3]M. Delanaye and J. A. Essers. Quadratic-reconstruction finite volume scheme for compressible flows on unstructured adaptive grids. *AIAA J.*, 35(4):631–639, 1997.

[4]C. Ollivier Gooch and M. Van Altena. A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation. *J. Comput. Phys.*, 181:729–752, 2002.

[5]D. Caraeni and D. C. Hill. Unstructured-grid third-order finite volume discretization using a multistep quadratic data-reconstruction method. *AIAA J.*, 48(4):808–817, 2010.

[6]F. Haider, P. Brenner, B. Courbet, and J.-P. Croisille. Efficient implementation of high order reconstruction in finite volume methods. In *Proc. of Finite Volumes for Complex Applications VI Problems & Perspectives, FVCA 6, International Symposium*, pages 553–560, Prague, June 2011.

[7]M. R. J. Charest, C. P. T. Groth, and P. Q. Gauthier and. High-order ceno finite-volume scheme for low-speed viscous flows on three-dimensional unstructured mesh. In *Proc. of Seventh International Conference on Computational Fluid Dynamics*, Big Island, Hawaii, 2012.

[8]A. Jalali and C. Ollivier-Gooch. Higher-order unstructured finite volume RANS solution of turbulent compressible flows. *Comput. Fluids*, 143:32–47, 2017.

[9]G. Pont, P. Brenner, P. Cinnella, B. Maugars, and J.-C. Robinet. Multiple-correction hybrid k-exact schemes for high-order compressible RANS-LES simulations on fully unstructured grids. *J. Comput. Phys.*, 350:45–83, 2017.

[10]B. P. Leonard. A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering*, 19:59–98, 1979.

[11]H. Nishikawa. The QUICK scheme is a third-order finite-volume scheme with point-valued numerical solutions. *Int. J. Numer. Meth. Fluids*, 93:2311–2388, April 2021.

[12]D. de Felice and F. M. Denaro C. Moela. Multidimensional single-step vector upwind schemes for highly convective transport problems. *Numerical Heat Transfer, Part B: Fundamentals: An International Journal of Computation and Methodology*, 23(4):425–460, 1993.

[13]F. M. Denaro. Towards a new model-free simulation of high-Reynolds-flows: Local average direct numerical. *Int. J. Numer. Meth. Fluids*, 23:125–142, 1996.

[14]A. Aprovitola, P. D'Ambra, F. M. Denaro, D. di Serafino, and S. Filippone. SParC-LES: Enabling large eddy simulations with parallel sparse matrix computation tools. *Computer Methods in Applied Mechanics and Engineering*, 70:2688–2700, 2015.

[15]H. Nishikawa and J. A. White. An efficient cell-centered finite-volume method with face-averaged nodal-gradients for triangular grids. *J. Comput. Phys.*, 411:109423, 2020.

[16]J. White, H. Nishikawa, and R. Baurle. A 3-D nodal-averaged gradient approach for unstructured-grid cell-centered finite-volume methods for application to turbulent hypersonic flow. In *SciTech 2020 Forum*, AIAA Paper 2020-0652, Orlando, FL, 2020.

[17]J. A. White, H. Nishikawa, and M. O'Connell. F-ANG+: A 3-D augmented-stencil face-averaged nodal- gradient cell-centered finite-volume method for hypersonic flows. In *SciTech 2022 Forum*, AIAA Paper 2022-1848, San Diego, CA, 2022.

[18]N. T. Frink. Tetrahedral unstructured Navier-Stokes method for turbulent flows. *AIAA J.*, 36(11):1975–1982, 1998.

[19]M. J. Pandya, B. Diskin, J. L. Thomas, and N. T. Frink. Improved convergence and robustness of USM3D solutions on mixed element grids. *AIAA J.*, 54(9):2589–2610, September 2016.

[20]D. Caraeni and L. Fuchs. Compact third-order multidimensional upwind discretization for steady and unsteady flow simulations. *Comput. Fluids*, 34:419–441, 2005.

[21]Y. Liu and H. Nishikawa. Third-order inviscid and second-order hyperbolic Navier-Stokes solvers for three-dimensional inviscid and viscous flows. In *46th AIAA Fluid Dynamics Conference*, AIAA Paper 2016-3969, Washington, D.C., 2016.

[22]M. A. Park and D. L. Darmofal. Parallel anisotropic tetrahedral adaptation. In *Proc. of 46th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA Paper 2008-917, Reno, NV, 2008.

[23]F. Alauzet, A. Dervieux, and A. Loseille. Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations. *J. Comput. Phys.*, 229(8):2866–2897, 2010.

[24]W. L. Kleb, M. A. Park, W. A. Wood, K. L. Bibb, K. B. Thompson, and R. J. Gomez. Sketch-to-solution: An exploration of viscous CFD with automatic grids. In *24th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2019-2948, Dallas, TX, 2019.

[25]H. Nishikawa. New unstructured-grid limiter functions. In *SciTech 2022 Forum*, AIAA Paper 2022-1374, San Diego, CA, 2022.

[26]V. Venkatakrishnan. Convergence to steady state solutions of the euler equations on unstructured grids with limiters. *J. Comput. Phys.*, 118:120–130, 1995.

[27]S. L. Krist, R. T. Biedron, and C. L. Rumsey. CFL3D User's Manual (Version 5.0). *NASA/TM-1998-208444*, June 1998.

[28]E. F. Toro, M. Spruce, and W. Speares. Restoration of the contact surface in the HLL-Riemann solver. *Shock Waves*, 4:25–34, 1994.

[29]P. B. Batten, N. Clarke, C. Lambert, and D. M. Causon. On the choice of wavespeeds for the HLLC riemann solver. *SIAM J. Sci. Comput.*, 18(6):1553–1570, 1997.

[30]P. Brenner. Towards rather comprehensive methods for finite volume methods. October 2010. doi:10.13140/RG.2.2.29412.53127.

[31]F. Setzwein, P. Ess, and P. Gerlinger. An implicit high-order k-exact finite-volume approach on vertex-centered unstructured grids for incompressible flows. *J. Comput. Phys.*, 446:110629, 2021.

[32]H. Nishikawa. A flux correction for finite-volume discretizations: Achieving second-order accuracy on arbitrary polyhedral grids. *J. Comput. Phys.*, 468:111481, 2022.

[33]T. J. Barth and D. C. Jesperson. The design and application of upwind schemes on unstructured meshes. In *Proc. of 27th AIAA Aerospace Sciences Meeting*, AIAA Paper 89-0366, Reno, NV, 1989.

[34]C. Michalak and C. Ollivier-Gooch. Accuracy preserving limiter for the high-order accurate solution of the Euler equations. *J. Comput. Phys.*, 228:8693–8711, 2009.

[35]E. Luke. On robust and accurate arbitrary polytope CFD solvers. In *Proc. of 18th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2007-3956, Miami, 2007.

[36]P. A. Gnoffo. Updates to multi-dimensional flux reconstruction for hypersonic simulations on tetrahedral grids. AIAA Paper 2010-1271, January 2010.

[37]J. S. Park and C. Kim. Hierarchical multi-dimensional limiting strategy for correction procedure via reconstruction. *J. Comput. Phys.*, 308:57–80, 2016.

[38]A. Sresh and H/ T. Huynh. Accurate monotonicity-preserving schemes with runge-kutta time stepping. *J. Comput. Phys.*, 136:83–99, 1997.

[39]J. A. White, R. Baurle, B. J. Passe, S. C. Spiegel, and H. Nishikawa. Geometrically flexible and efficient flow analysis of high speed vehicles via domain decomposition, part 1, unstructured-grid solver for high speed flows. In *JANNAF 48th Combustion 36th Airbreathing Propulsion, 36th Exhaust Plume and Signatures, 30th Propulsion Systems Hazards, Joint Subcommittee Meeting, Programmatic and Industrial Base Meeting*, Newport News, VA, 2017.

[40]W. L. Oberkampf and C. J. Roy. *Verification and Validation in Scientific Computing*. Cambridge University Press, 2010.

[41]S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Rev.*, 43(1):89–112, 2001.

[42]D. Kuzmin and S. Turek. Flux correction tools for finite elements. *J. Comput. Phys.*, 175:525–558, 2002.

[43]J. Donea and A. Huerta. *Finite Element Methods for Flow Probelms*. John Wiley & Sons, seventh edition, 2003.

[44]H. Nishikawa and Y. Liu. Third-order edge-based scheme for unsteady problems. In *AIAA 2018 Fluid Dynamics Conference*, AIAA Paper 2018-4166, Atlanta, Georgia, 2018.

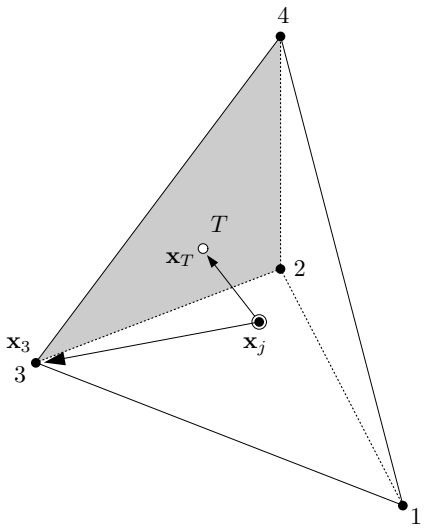
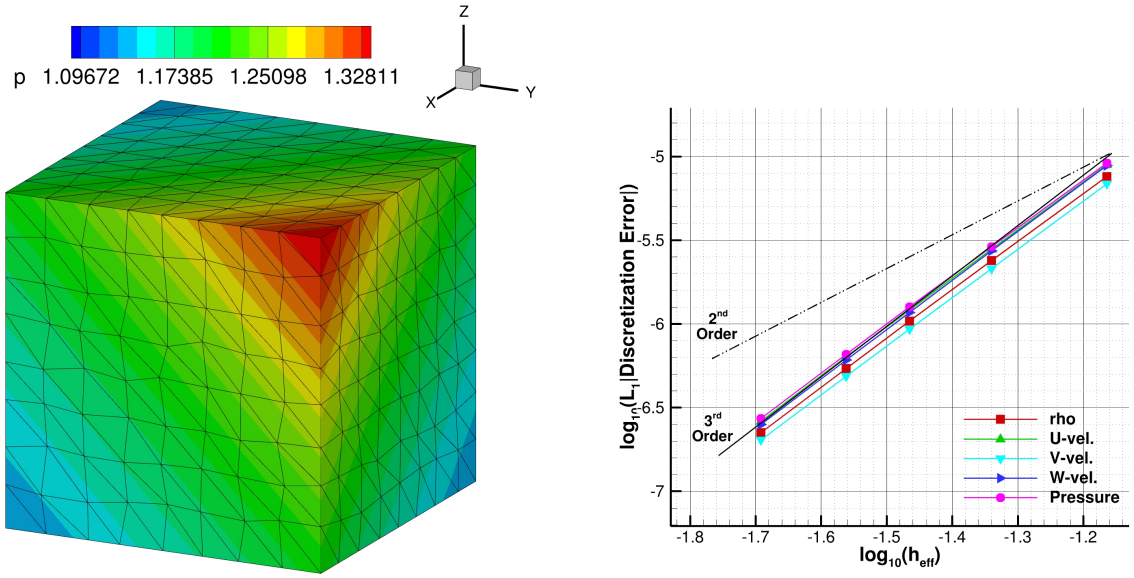


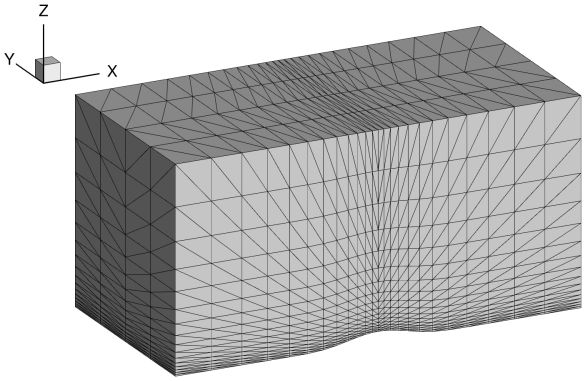
Figure 1. Interpolations to a face center $\mathbf{x}_T$ and a node 3 from the cell centroid in a tetrahedral cell. Function values are assumed to be available at the cell centroid $\mathbf{x}_j$ and at the nodes.



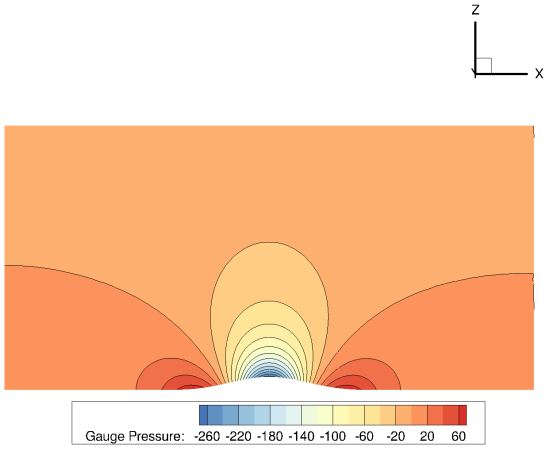
(a) The coarsest grid with pressure contours.

(b) L_1 error convergence.

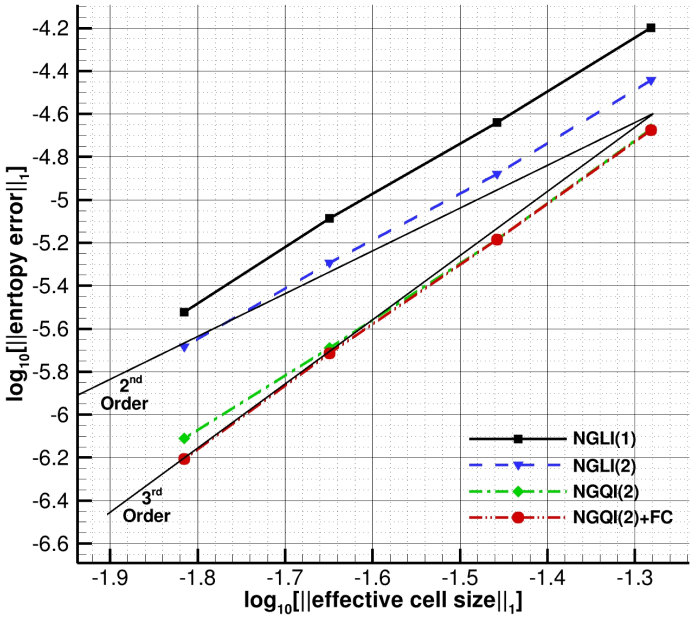
Figure 2. Error convergence results for the accuracy verification test case.



(a) The coarsest grid.

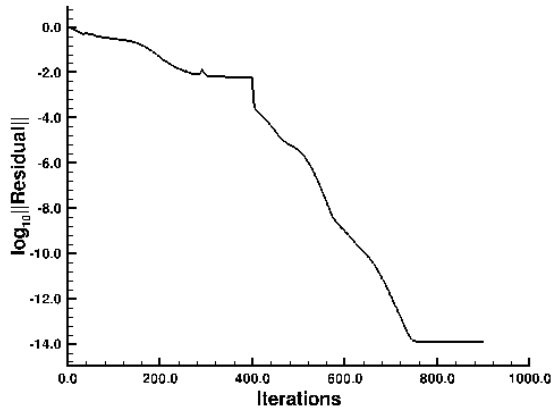


(b) Third-order accurate solution on the finest grid.

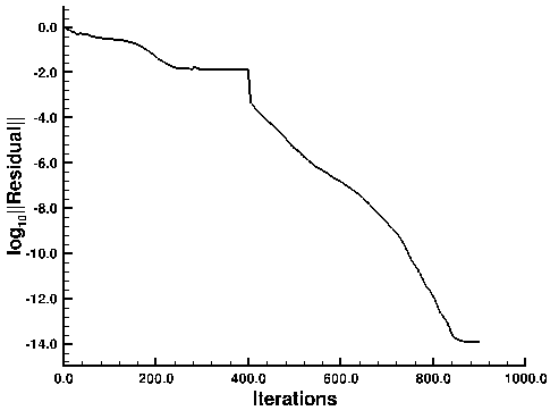


(c) L_1 error convergence.

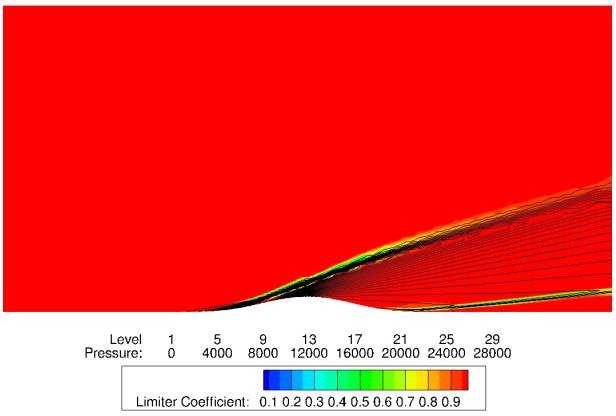
Figure 3. Error convergence results for the subsonic bump test case.



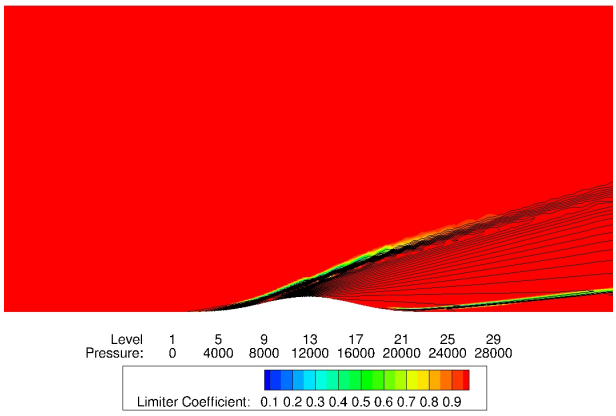
(a) Residual convergence for 2nd-order.



(b) Residual convergence for 3rd-order.



(c) Second-order solution with Venkatakrishan's limiter.



(d) Third-order solution with the R_5 limiter.

Figure 4. Pressure line contours and limiter color contours for the hypersonic bump test case.

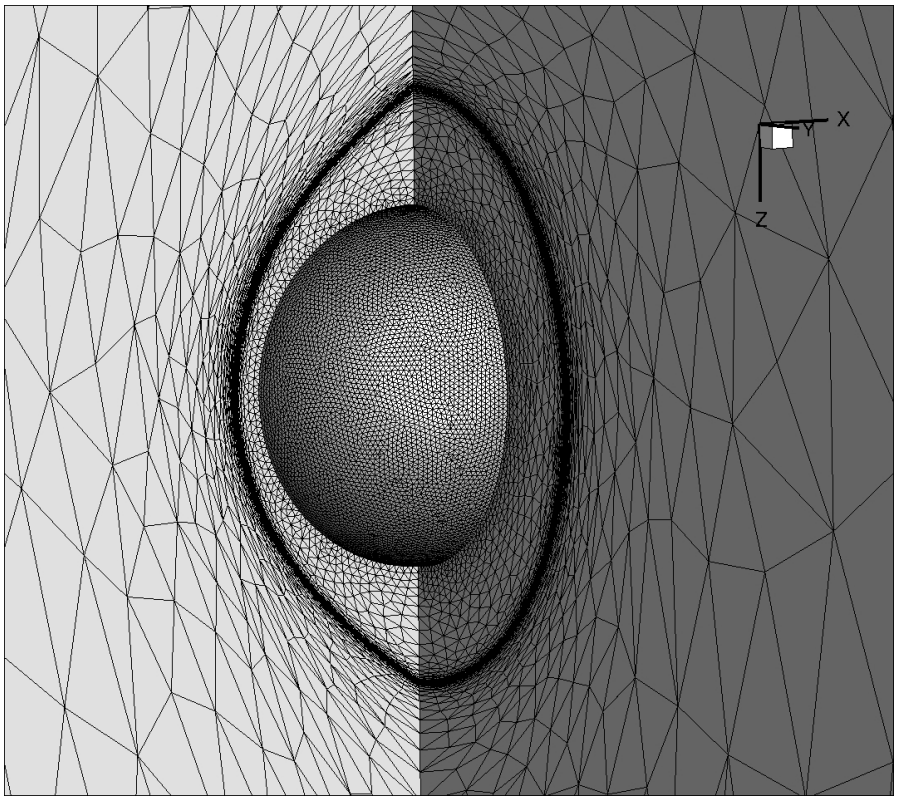
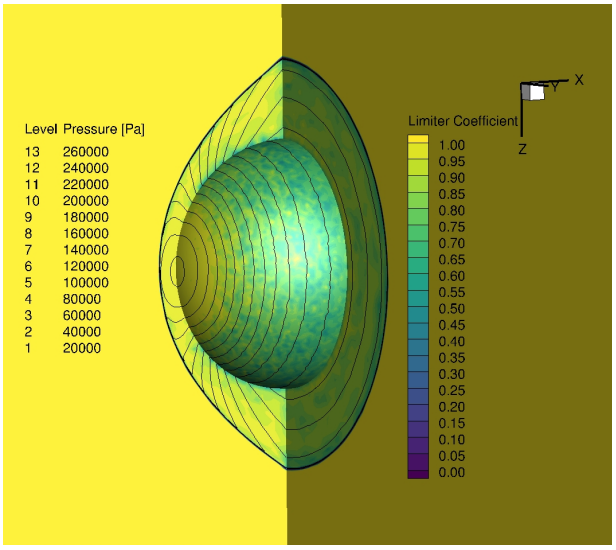
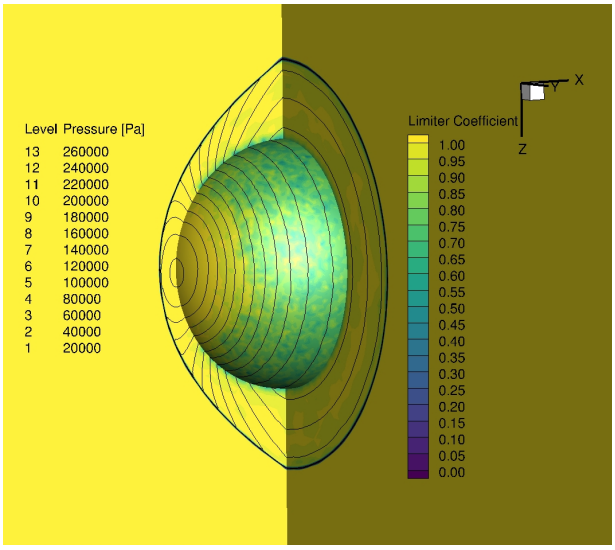


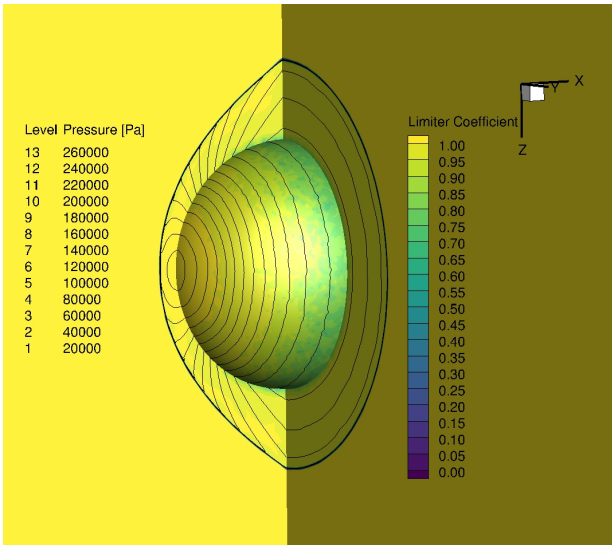
Figure 5. Adaptive tetrahedral grid over a sphere.



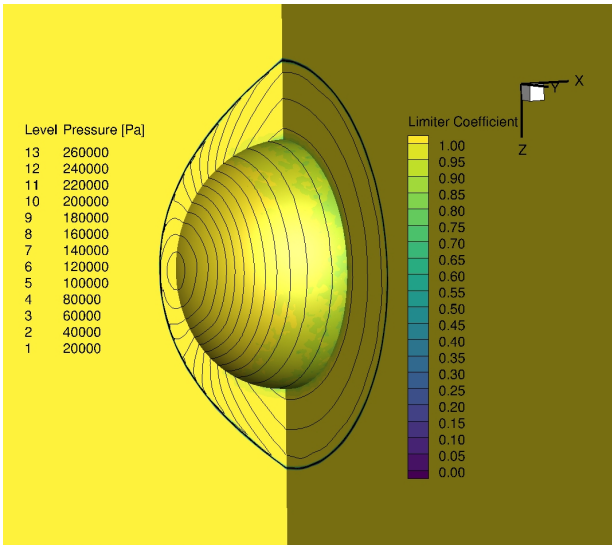
(a) K=1.5: second-order solution.



(b) $K = 1.5$: third-order solution.

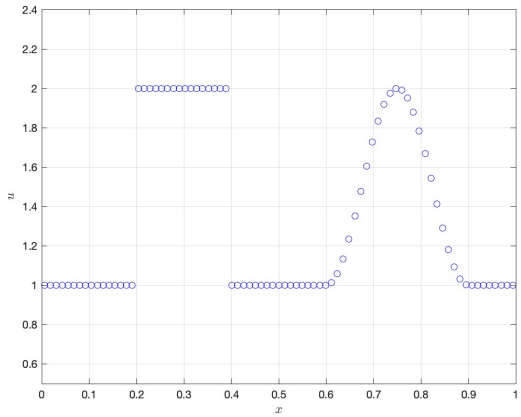


(c) $K = 5.0$: second-order solution.

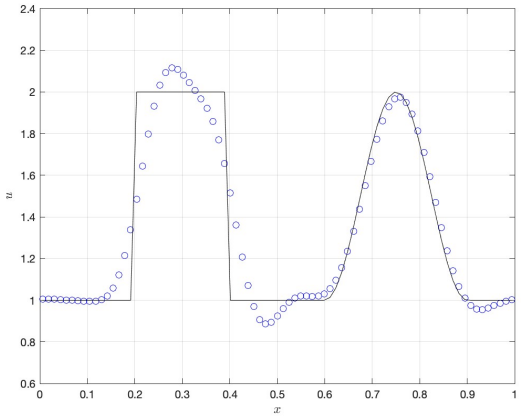


(d) $K = 5.0$: third-order solution.

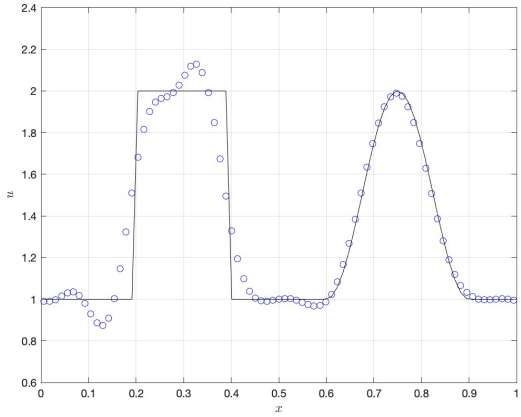
Figure 6. Pressure line contours and limiter color contours for the hypersonic sphere test case. Second and third order solutions are obtained with Venkatakrishan's limiter and the R_5 limiter, respectively.



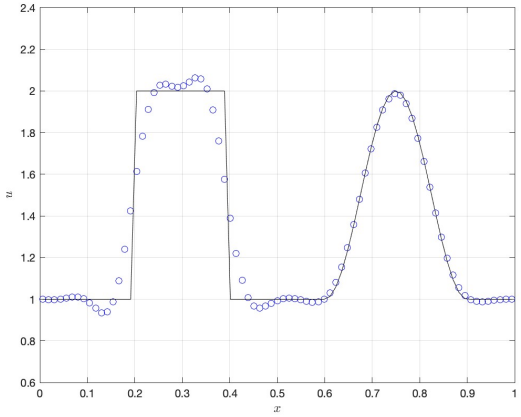
(a) Initial profile.



(b) Second-order scheme.

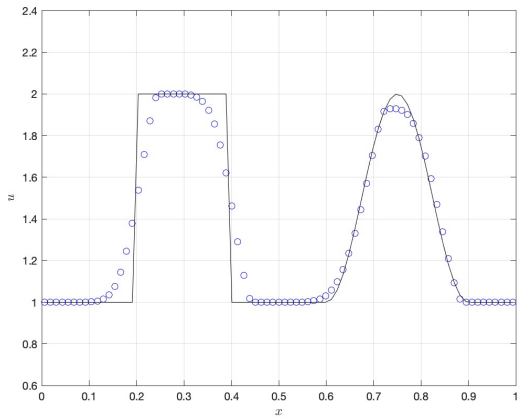


(c) Third-order scheme without mass matrix.

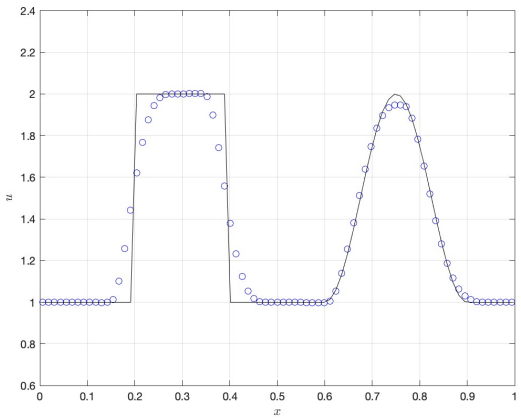


(d) Third-order scheme with mass matrix.

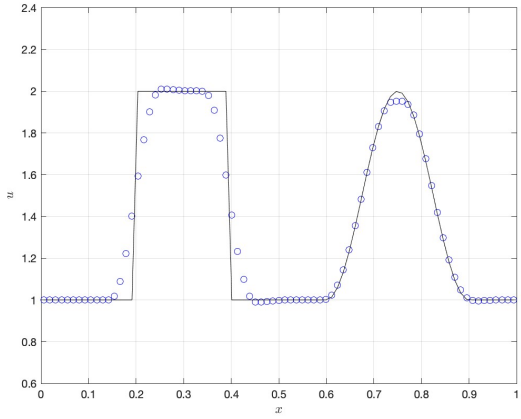
Figure 7. Initial profile and numerical solutions obtained with unlimited schemes for linear advection test.



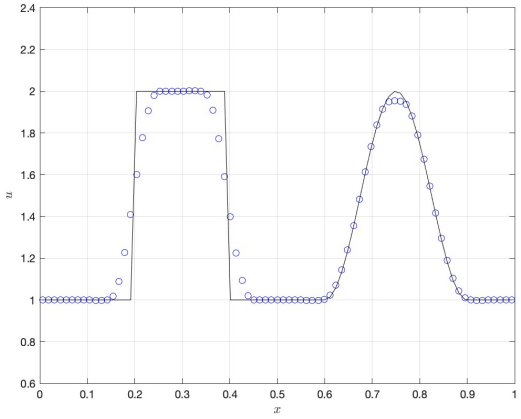
(a) Second-order scheme.



(b) Third-order scheme without the mass matrix.

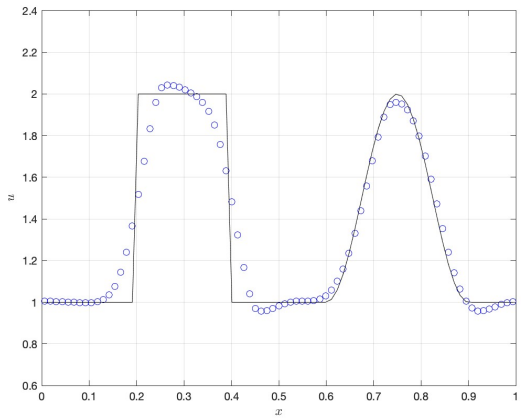


(c) Third-order scheme with the original mass matrix.

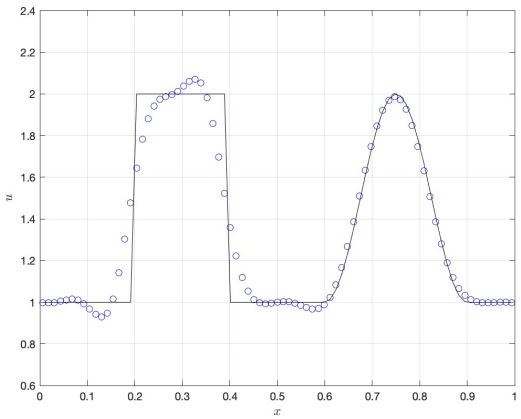


(d) Third-order scheme with the limited mass matrix.

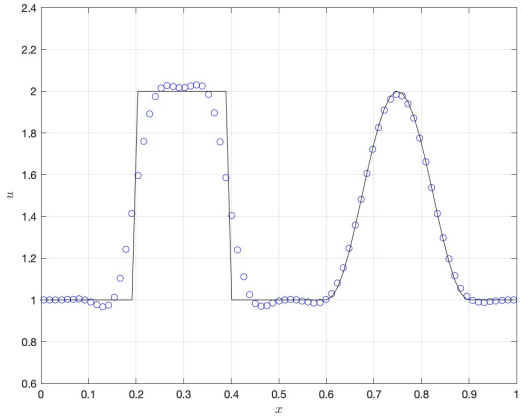
Figure 8. Numerical solutions obtained with limited schemes for linear advection test ($K = 0.1$).



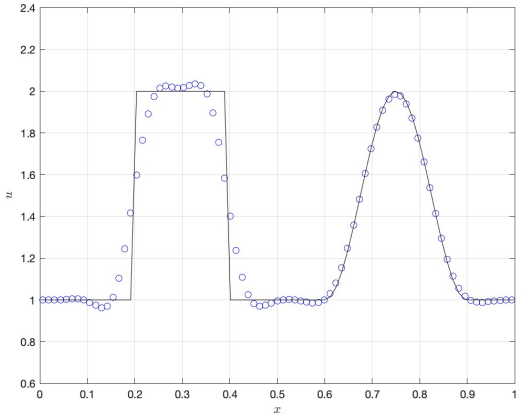
(a) Second-order scheme.



(b) Third-order scheme without the mass matrix.



(c) Third-order scheme with the original mass matrix.



(d) Third-order scheme with the limited mass matrix.

Figure 9. Numerical solutions obtained with limited schemes for linear advection test ($K = 2.0$).