Journal of Computational Physics **179,** 665–697 (2002)
doi:10.1006/jcph.2002.7082

Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids

II. Extension to Two-Dimensional Scalar Equation

Z. J. Wang*,[1] and Yen Liu†

*Michigan State University, 2328B Engineering Building, East Lansing, Michigan 48824; and †NASA Ames Research Center, Mail Stop T27B-1, Moffett Field, California 94035
E-mail: zjw@egr.msu.edu and liu@nas.nasa.gov

Received August 28, 2001; revised April 3, 2002

The framework for constructing a high-order, conservative spectral (finite) volume (SV) method is presented for two-dimensional scalar hyperbolic conservation laws on unstructured triangular grids. Each triangular grid cell forms a spectral volume (SV), and the SV is further subdivided into polygonal control volumes (CVs) to supported high-order data reconstructions. Cell-averaged solutions from these CVs are used to reconstruct a high-order polynomial approximation in the SV. Each CV is then updated independently with a Godunov-type finite volume method and a high-order Runge–Kutta time integration scheme. A universal reconstruction is obtained by partitioning all SVs in a geometrically similar manner. The convergence of the SV method is shown to depend on how a SV is partitioned. A criterion based on the Lebesgue constant has been developed and used successfully to determine the quality of various partitions. Symmetric, stable, and convergent linear, quadratic, and cubic SVs have been obtained, and many different types of partitions have been evaluated. The SV method is tested for both linear and nonlinear model problems with and without discontinuities.

Key Words: high order; unstructured grid; spectral volume; 2D conservation laws.

1. INTRODUCTION

We continue the development of the spectral (finite) volume (SV) method for hyperbolic conservation laws on unstructured grids following the one-dimensional framework presented in [38]. We wish to pursue a numerical method for conservation laws which has all of the following properties: (a) conservative, (b) high-order accuracy, i.e., the order of

[1] Fax: 517-353-1750.

0021-9991/02 $35.00

accuracy is greater than second order, (c) geometrically flexible, i.e., applicable for unstructured grids, and (d) computationally efficient. The SV method is developed, we hope, to satisfy these four requirements, in a relative sense with respect to current state-of-the-art numerical methods such as the high-order k-exact finite volume (FV) method [5, 16], the essentially nonoscillatory (ENO) method [1, 10, 19], the weighted ENO (WENO) method [3, 17, 20, 22, 27], and the discontinuous Galerkin (DG) method [2, 6, 12–14].

One of the most successful algorithms for conservation laws is the Godunov method [18], which laid a solid foundation for the development of modern upwind schemes, including MUSCL [36], PPM [15], ENO [19], and WENO schemes [22, 27]. There are two key components in a Godunov-type method. One is data reconstruction, and the other is the Riemann solver. The original Godunov scheme employed a piecewise constant data reconstruction and the exact Riemann solver, and the resultant scheme was only first-order accurate. Later on, higher order polynomial reconstructions were used to replace the piecewise constant reconstruction, and approximate and more efficient Riemann solvers [21, 25, 29, 31, 35, 37] were employed to replace the exact Riemann solver. In addition, limiters were also introduced to remove spurious numerical oscillations near steep gradients [36] in higher-than-first-order Godunov methods.

Although Godunov-type methods were originally developed for structured grids, they have been successfully extended to unstructured grids, thus achieving greater geometric flexibility. Most of the unstructured grid methods are second-order accurate because they are relatively easy to implement and are quite memory efficient. Several high-order schemes have been developed for unstructured grids. For example, a high-order k-exact finite volume scheme was developed by Barth and Frederickson [5]. An ENO scheme for unstructured grid was developed by Abgrall [1]. Two WENO schemes for unstructured grids were developed by Friedrich [17] and Hu and Shu [20]. Although high-order-accurate finite volume schemes can be obtained theoretically for an unstructured grid by using high-order polynomial data reconstructions, higher-than-linear reconstructions are rarely used in three dimensions in practice. This is mainly because of the difficulty in finding valid high-order (nonsingular) stencils and the enormous memory required to store the coefficients used in the reconstruction. In a k-exact finite volume method, each control volume has a different reconstruction stencil. Therefore, a data reconstruction must be performed at each iteration for each control volume. This reconstruction step uses the most memory and is the most time consuming in higher-than-second-order schemes. In a recent implementation of a third-order FV scheme with a quadratic reconstruction in three dimensions by Delanaye and Liu [16], the average size of the reconstruction stencils was about 50–70. Still there are many singular reconstruction stencils. The size of the reconstruction stencils usually increases nonlinearly with the order of accuracy. For a fourth-order FV scheme, the average stencil size is estimated to be at least 120. It is very memory and CPU intensive to perform the reconstruction.

More recently, another high-order conservative algorithm, called the discontinuous Galerkin method, was developed by Cockburn *et al.* in a series of papers [12–14], and also in [2, 6] on unstructured grids. In the DG method, a high-order data distribution is assumed for each element. As a result, the state variable is usually not continuous across element boundaries. The fluxes through the element boundaries are computed using an approximate Riemann solver, a method similar to FV methods. The residual is then minimized with a Galerkin approach. Due to the use of Riemann fluxes across element boundaries, the DG method is fully conservative. A disadvantage of the DG method is that high-order surface and volume integrals are necessary, which can be expensive to compute. Another

high-order conservative scheme for unstructured quadrilateral grids is the multidomain spectral method on a staggered grid developed by Kopriva and co-workers [23, 24]. The multidomain spectral method is similar to the spectral element method of Patera [30], which is not conservative. Other related methods include the cell-average-based spectral method [9] and the spectral-element-type method [34]. Although a very high-order of accuracy was achievable with these methods, the methods are difficult to extend to other cell types, such as triangles or tetrahedral cells.

In [38], the first paper in the series, a new conservative high-order SV method is developed for conservation laws on one-dimensional unstructured grids. Through the use of spectral volumes (SVs) and control volumes (CVs), the method is not only conservative but very efficient. In this paper, we extend the SV method to two dimensions. In the next section, we first review the basic framework of the SV method on triangular grids. In addition, we present a TVD Runge–Kutta time integration scheme. In Section 3, the reconstruction problem based on CV-averaged solutions is studied, and it is shown that the reconstruction problems on all triangles with a similar partition are identical. In Section 4, convergent reconstructions for high-order SV schemes are discussed, and the partition of a SV is shown to affect the convergence of the method. Section 5 discusses issues related to discontinuity capturing and several TVD and TVB limiters are presented. In Section 6, numerical implementations of the SV method for both linear and nonlinear scalar conservation laws are carried out, and accuracy studies are performed for both linear and nonlinear wave equations to verify the numerical order of accuracy. The shock-capturing capability of the method is also demonstrated with Burger's equation. Finally, conclusions and recommendations for further investigations are summarized in Section 7.

2. REVIEW OF THE SPECTRAL VOLUME METHOD

Consider the multidimensional scalar conservation law

$$\frac{\partial u(x, y, t)}{\partial t} + \frac{\partial f(u(x, y, t))}{\partial x} + \frac{\partial g(u(x, y, t))}{\partial y} = 0 \tag{2.1a}$$

on domain $\Omega \times [0, T]$ and $\Omega \subset R^2$ with the initial condition

$$u(x, y, 0) = u_0(x, y) \tag{2.1b}$$

and appropriate boundary conditions on $\partial\Omega$. In (2.1), x and y are the Cartesian coordinates and $(x, y) \in \Omega$, $t \in [0, T]$ denotes time, u is a state variable, and f and g are fluxes in x and y directions, respectively. Domain Ω is discretized into I nonoverlapping triangular cells. In a k-exact FV method, a data reconstruction is performed for each cell using data from a collection of neighboring cells, collectively known as a reconstruction stencil, as shown in Fig. 1a. A unique flux through each face is then computed, given the reconstructed state variables at both sides of the face using either an exact or approximate Riemann solver. A summation of fluxes through all the faces of a cell is then used to update the cell-averaged state variable.

In the SV method, the triangular cells are called *spectral volumes*, denoted S_i, which are further partitioned into subcells named *control volumes* (CVs), denoted C_{ij}. Volume-averaged state variables on the CVs are used to reconstruct a high-order polynomial inside the SV. To represent the solution as a polynomial of degree m in 2D, we need

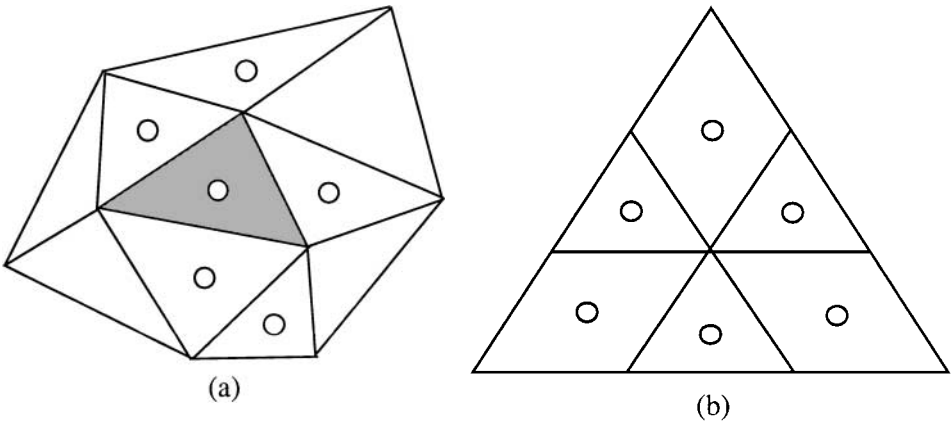


FIG. 1. (a) A possible reconstruction stencil for a quadratic reconstruction in a high-order k-exact finite volume scheme. (b) The partition of a spectral volume into six control volumes supporting a quadratic reconstruction.

$N = (m + 1)(m + 2)/2$ pieces of independent information, or degrees of freedom (DOFs). The DOFs in a SV method are the volume-averaged mean variables at the N CVs. For example, a SV supporting a quadratic data reconstruction is shown in Fig. 1b. Other candidate partitions for linear-to-cubic SVs are shown in Figs. 2–4. The number of CVs in Figs. 2–4 is the minimum required for these polynomial reconstructions. Other CV subdivisions are definitely possible. Integrating (2.1) on C_{ij}, we obtain

$$\int_{C_{i,j}} \frac{\partial u}{\partial t}\, dV + \oint_{\partial C_{i,j}} (F \cdot \boldsymbol{n})\, dA = 0, \tag{2.2}$$

where $F = (f, g)$ and $\mathbf{n}$ is the unit outward normal of $\partial C_{i,j}$, the boundary of C_{ij}. Define the CV-averaged state variable for C_{ij} as

$$\bar{u}_{i,j} = \frac{\int_{c_{i,j}} u dV}{V_{i,j}}, \tag{2.3}$$

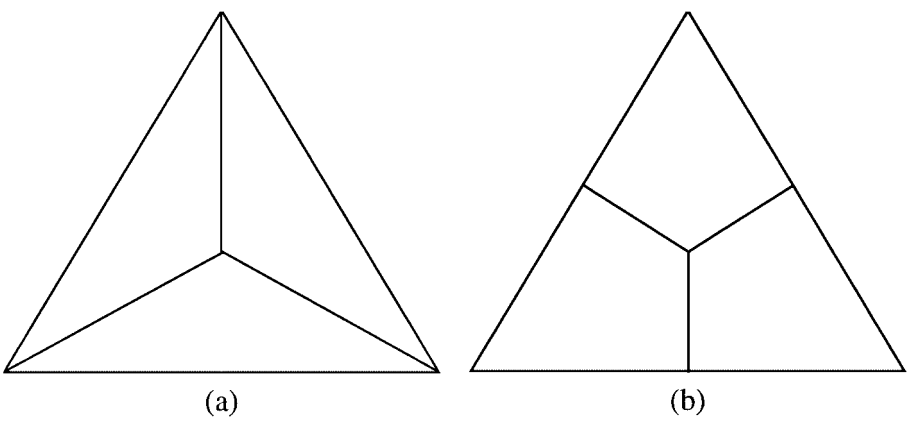


FIG. 2. Control volumes in a triangular linear spectral volume. (a) Type 1, $n_1 = 0, n_3 = 1, n_6 = 0$; (b) Type 2, $n_1 = 0, n_3 = 1, n_6 = 0$.

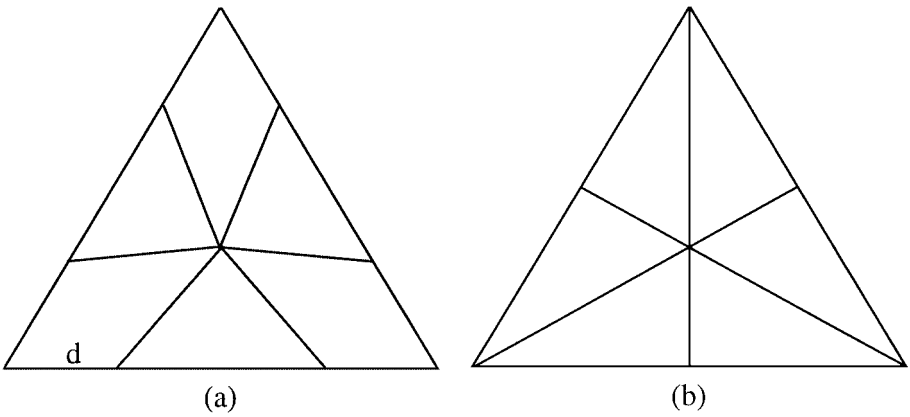


FIG. 3. Possible triangular quadratic spectral volume partitions. (a) Type 1 with $d = 1/3$ and Type 2 with $d = 1/4$; $n_1 = 0, n_3 = 2, n_6 = 0$. (b) A singular partition; $n_1 = 0, n_3 = 0, n_6 = 1$.

where V_{ij} is the volume (area in 2D) of C_{ij}. Then (2.2) becomes

$$\frac{d\bar{u}_{i,j}}{dt} + \frac{1}{V_{i,j}} \sum_{r=1}^{K} \int_{A_r} (F \cdot \boldsymbol{n})\, dA = 0, \tag{2.4}$$

where K is the total number of faces in C_{ij}, and A_r represents the rth face of C_{ij}. The surface

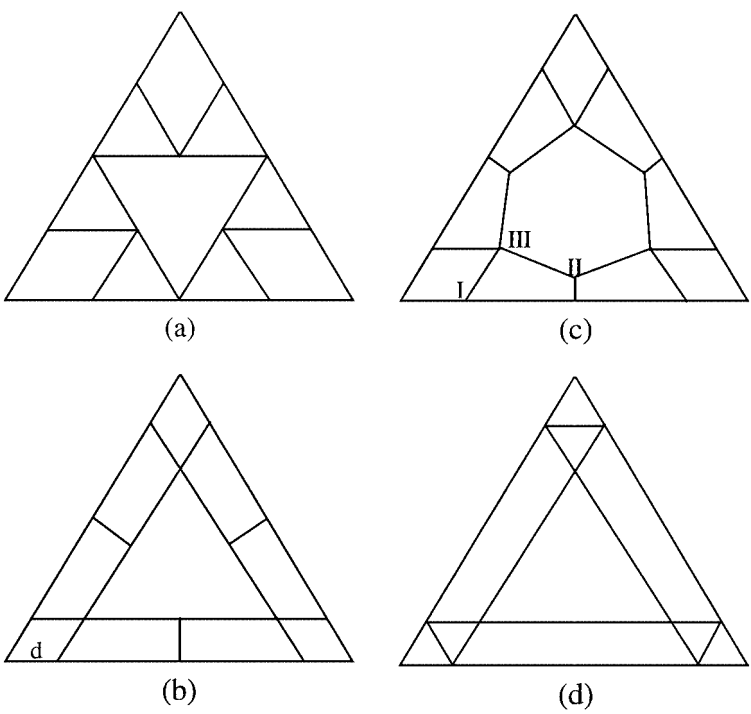


FIG. 4. Possible cubic triangular spectral volumes. (a) Type 1, $n_1 = 1, n_3 = 1, n_6 = 1$. (b) Type 2 with $d = 1/6$ and Type 3 with $d = 1/15$; $n_1 = 1, n_3 = 1, n_6 = 1$. (c) $n_1 = 1, n_3 = 1, n_6 = 1$. (d) $n_1 = 1, n_3 = 3, n_6 = 0$.

integration on each face can be performed with a kth-order-accurate Gauss quadrature formula ($k = m + 1$); i.e.,

$$\int_{A_r} (F \cdot \boldsymbol{n})\, dA = \sum_{q=1}^{J} w_{rq} F(u(x_{rq}, y_{rq})) \cdot \boldsymbol{n}_r A_r + O(A_r h^k), \tag{2.5}$$

where $J = \text{integer}[(k + 1)/2]$ is the number of quadrature points on the rth face, w_{rq} are the Gauss quadrature weights, (x_{rq}, y_{rq}) are the Gauss quadrature points, and h is the maximum edge length of all the CVs. Time t is omitted whenever there is no confusion. If $F = \text{constant}$, the following identity exists:

$$\sum_{r=1}^{K} \int_{A_r} (F \cdot \boldsymbol{n})\, dA = 0. \tag{2.6}$$

Therefore, we will gain an extra order of accuracy if we sum up the surface integrals for the faces of C_{ij}; i.e.,

$$\sum_{r=1}^{K} \int_{A_r} (F \cdot \boldsymbol{n})\, dA = \sum_{r=1}^{K} \sum_{q=1}^{J} w_{rq} F(u(x_{rq}, y_{rq})) \cdot \boldsymbol{n}_r A_r + O(A_r h^{k+1}). \tag{2.7}$$

Since $O(V_i) = O(A_r h)$, we have

$$\frac{1}{V_{i,j}} \sum_{r=1}^{K} \oint_{A_r} (F \cdot \boldsymbol{n})\, dA = \frac{1}{V_{i,j}} \sum_{r=1}^{K} \sum_{q=1}^{J} w_{rq} F(u(x_{rq}, y_{rq})) \cdot \boldsymbol{n}_r A_r + O(h^k). \tag{2.8}$$

Now assume that a multidimensional polynomial in x and y on the order of at most $k - 1$ exists on S_i which is a kth-order approximation to the state variable; i.e.,

$$p_i(x, y) = u(x, y) + O(h^k), \quad (x, y) \in S_i. \tag{2.9}$$

With the polynomial distribution on each SV, the state variable is most likely discontinuous across the SV boundaries, unless the state variable is a polynomial on the order of $k - 1$ or less. Therefore, the flux integration involves two discontinuous state variables just to the left and right of a face of the SV boundary. This flux integration is carried out using an exact Riemann solver or one of the Lipschitz continuous approximate Riemann solvers or flux splitting procedures; i.e.,

$$\begin{aligned} F(u(x_{rq}, y_{rq})) \cdot \boldsymbol{n}_r = {} & F_{\text{Riem}}(p_i(x_{rq}, y_{rq}), p_{i,r}(x_{rq}, y_{rq}), \boldsymbol{n}_r) \\ & + O(p_i(x_{rq}, y_{rq}) - p_{i,r}(x_{rq}, y_{rq})). \end{aligned} \tag{2.10}$$

Here $p_{i,r}$ is the reconstruction polynomial of a neighboring CV $C_{i,j,r}$, which shares the face A_r with C_{ij}. Both p_i and $p_{i,r}$ are kth-order approximations of the exact state variable; i.e.,

$$p_i(x_{rq}, y_{rq}) = u(x_{rq}, y_{rq}) + O(h^k), \tag{2.11a}$$

$$p_{i,r}(x_{rq}, y_{rq}) = u(x_{rq}, y_{rq}) + O(h^k). \tag{2.11b}$$

Therefore

$$F(u(x_{rq}, y_{rq})) \cdot \boldsymbol{n}_r = F_{\text{Riem}}(p_i(x_{rq}, y_{rq}), p_{i,r}(x_{rq}, y_{rq}), \boldsymbol{n}_r) + O(h^k). \tag{2.12}$$

Substituting (2.12) into (2.4), we obtain

$$\int_{A_r} (F \cdot \boldsymbol{n})\, dA = \sum_{q=1}^{J} w_{rq} F_{\text{Riem}}(p_i(x_{rq}, y_{rq}), p_{i,r}(x_{rq}, y_{rq}), \boldsymbol{n}_r) A_r + O(A_r h^k). \tag{2.13}$$

Summarizing (2.4)–(2.13), we obtain the following semidiscrete, kth-order-accurate scheme on C_{ij} for the conservation law (2.1):

$$\frac{d\bar{u}_{i,j}}{dt} + \frac{1}{V_{i,j}} \sum_{r=1}^{K} \sum_{q=1}^{J} w_{rq} F_{\text{Riem}}(p_i(x_{rq}, y_{rq}), p_{i,r}(x_{rq}, y_{rq}), \boldsymbol{n}_r) A_r + O(h^k). \tag{2.14}$$

For time integration, we will use the third-order TVD Runge–Kutta scheme from [32]. We first rewrite (2.14) in a concise ODE form

$$\frac{d\bar{u}}{dt} = R_h(\bar{u}), \tag{2.15a}$$

where

$$\bar{u} = \begin{bmatrix} \bar{u}_{1,1} \\ \cdots \\ \bar{u}_{i,j} \\ \cdots \\ \bar{u}_{I,N} \end{bmatrix}, \quad R_h(\bar{u}) = \begin{bmatrix} R_{1,1}(\bar{u}) \\ \cdots \\ R_{i,j}(\bar{u}) \\ \cdots \\ R_{I,N}(\bar{u}) \end{bmatrix}, \tag{2.15b}$$

and

$$R_{i,j} = -\frac{1}{V_{i,j}} \sum_{r=1}^{K} \sum_{q=1}^{J} w_{rq} F_{\text{Riem}}(p_i(x_{rq}, y_{rq}), p_{i,r}(x_{rq}, y_{rq}), \boldsymbol{n}_r) A_r. \tag{2.15c}$$

Then the third-order TVD Runge–Kutta scheme can be expressed as

$$\begin{aligned} \bar{u}^{(1)} &= \bar{u}^n + \Delta t\, R_h(\bar{u}^n), \\ \bar{u}^{(2)} &= \frac{3}{4}\bar{u}^n + \frac{1}{4}\left[\bar{u}^{(1)} + \Delta t\, R_h\left(\bar{u}^{(1)}\right)\right], \\ \bar{u}^{n+1} &= \frac{1}{3}\bar{u}^n + \frac{2}{3}\left[\bar{u}^{(2)} + \Delta t\, R_h\left(\bar{u}^{(2)}\right)\right]. \end{aligned} \tag{2.16}$$

The SV method idea can of course be easily extended to other cell types, such as quadrilaterals, tetrahedra, hexahedra, prisms, and so forth. For cell types other than triangles and tetrahedra, it seems symmetric CV subdivisions with the minimum number of CVs for a given order of accuracy are difficult to obtain. The development of the SV method for other cell types will be reported elsewhere.

3. THE RECONSTRUCTION PROBLEM IN A SPECTRAL VOLUME

As discussed in the previous section, in order to compute the flux across a surface, one needs to evaluate the state variable u at quadrature points. These evaluations can be achieved by reconstructing the state variable u in terms of some basis functions using the CV-averaged solutions $\bar{u}_{,j}$ within a SV. (For simplicity, we drop the subscript i and use $\bar{u}_{,j}$, $C_{,j}$, and $V_{,j}$ to denote $\bar{u}_{i,j}$, $C_{i,j}$ and $V_{i,j}$, respectively.) In general, one can choose any linearly independent functions as the basis functions. Here we focus only on the reconstruction using polynomials as the basis functions.

Let P_m denote the space of degree m polynomials in two dimensions. Then the dimension of the approximation space is

$$N_m = \binom{m+2}{2} = \frac{(m+1)(m+2)}{2},$$

which is the minimum dimension of the space that allows P_m to be complete. In order to reconstruct u in P_m, we need to partition the SV into a set of N_m nonoverlapping CVs. Let S denote the physical space of a SV and Π_m denote the partition; i.e.,

$$\Pi_m = \{C_{,1}, C_{,2}, \ldots, C_{,N_m}\},$$

where $C_{,j} \subset S$, $j = 1, 2, \ldots, N_m$ (or N if there is no confusion), is the jth CV inside the SV, and

$$S = \bigcup_{j=1}^{N_m} C_{,j}.$$

The reconstruction problem reads as follows: Given a continuous function in S, $u \in \mathcal{C}(S)$ (the space of continuous functions in S), and a partition Π_m of S, find $p_m \in P_m$ such that

$$\int_{C_{,j}} p_m(x, y)\, dV = \int_{C_{,j}} u(x, y)\, dV, \quad j = 1, \ldots, N_m. \tag{3.1}$$

To actually solve the reconstruction problem, we introduce the complete polynomial basis, $e_l(x, y) \in P_m$, where $P_m = \text{span}\{e_l(x, y)\}_{l=1}^{N_m}$. Therefore p_m can be expressed as

$$p_m = \sum_{l=1}^{N_m} a_l e_l(x, y), \tag{3.2a}$$

or in the matrix form

$$p_m = ea, \tag{3.2b}$$

where e is the basis function vector $[e_1, \ldots, e_N]$ and a is the reconstruction coefficient vector $[a_1, \ldots, a_N]^T$. Substituting (3.2a) into (3.1), we then obtain

$$\frac{1}{V_{,j}} \sum_{l=1}^{N_m} a_l \int_{C_{,j}} e_l(x, y)\, dV = \bar{u}_{,j}, \quad j = 1, \ldots, N_m. \tag{3.3}$$

Let $\bar{u}$ denote the column vector $[\bar{u}_{,1}, \ldots, \bar{u}_{,N}]^T$. Equation (3.3) can be rewritten in the matrix form

$$Ra = \bar{u}, \tag{3.4}$$

where the reconstruction matrix

$$R = \begin{bmatrix} \frac{1}{V_1}\int_{C_{,1}} e_1(x, y)\, dV & \cdots & \frac{1}{V_1}\int_{C_{,1}} e_N(x, y)\, dV \\ \vdots & \cdots & \vdots \\ \frac{1}{V_N}\int_{C_{,N}} e_1(x, y)\, dV & \cdots & \frac{1}{V_N}\int_{C_{,N}} e_N(x, y)\, dV \end{bmatrix}. \tag{3.5}$$

The reconstruction coefficients a can be solved as

$$a = R^{-1}\bar{u}, \tag{3.6}$$

provided that the reconstruction matrix R is nonsingular. Substituting Eq. (3.6) into Eq. (3.2), p_m is then expressed in terms of *cardinal* basis functions or shape functions $L = [L, \ldots, L_N]$:

$$p_m = \sum_{j=1}^{N_m} L_j(x, y)\bar{u}_{,j} = L\bar{u}. \tag{3.7}$$

Here L is defined as

$$L \equiv eR^{-1}. \tag{3.8}$$

Equation (3.7) gives the functional representation of the state variable u within the SV. The function value of u at a quadrature point or any point (x_{rq}, y_{rq}) within the SV is thus simply

$$p_m(x_{rq}, y_{rq}) = \sum_{j=1}^{N_m} L_j(x_{rq}, y_{rq})\bar{u}_{,j}. \tag{3.9}$$

The above equation can be viewed as an interpolation of a function value at a point using a set of cell-averaged values with each weight equal to the corresponding cardinal basis functional value evaluated at that point.

Note that once the polynomial basis functions e_l are chosen, the cardinal basis functions L_j are solely determined by the partition Π_m of S. The shape and the partition of S, in general, can be arbitrary as long as the reconstruction matrix R is nonsingular. However, different shapes of spectral volumes can result in the same expression of the cardinal basis functions (in terms of a few geometric parameters) if a geometrically similar partition can be applied to them. In the following, we examine a special case where all SVs are triangular and all CVs are polygons with straight edges. In this case, even though the shapes of the SVs may all be different, as long as they are partitioned in a geometrically similar manner, they all have the same reconstruction, in which the functional values of the cardinal bases at similar grid points are all exactly the same. We defer the discussion of other types of spectral volumes (e.g., quadrilateral, tetrahedron, curved boundaries, etc.) elsewhere.

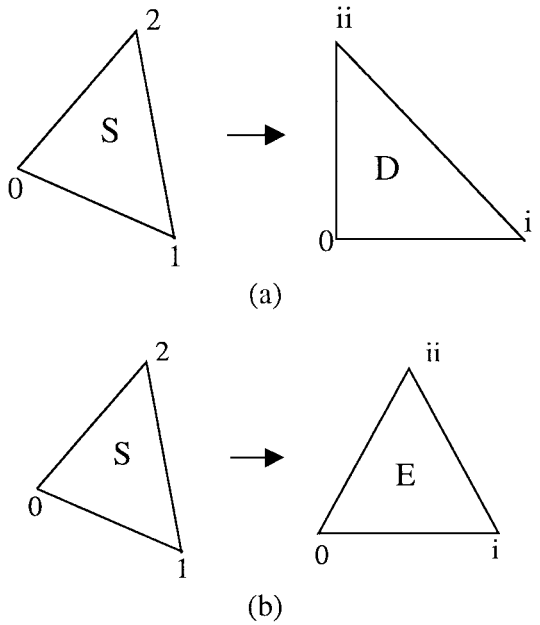


FIG. 5. The schematic of the mapping from the physical triangle to the standard triangle.

We first consider a transformation $\Psi : S \to D$, shown in Fig. 5a, which transforms an arbitrary triangle S to a right triangle D. Another often-used transformation is from an arbitrary triangle to an equilateral triangle E, shown in Fig. 5b. Let us use (x, y) to denote the coordinates in S and (ξ, η) the coordinates in D. For simplicity, we assume one of the nodes is located at the origin $\boldsymbol{r}_0 = (0,0)$ and the other two at $\boldsymbol{r}_1 = (x_1, y_1)$ and $\boldsymbol{r}_2 = (x_2, y_2)$ in S, corresponding to (0,0), (1,0), and (0,1) in D, respectively. Thus, the transformation can be written as

$$\Psi : \boldsymbol{r} = \boldsymbol{r}_1\xi + \boldsymbol{r}_2\eta, \qquad \xi \geq 0, \eta \geq 0, \quad \text{and} \quad \xi + \eta \leq 1. \tag{3.10}$$

Since the transformation is linear, for a complete set of basis functions $e(x, y) \in P_m$, one can easily show that

$$e(x, y) = e(\xi, \eta)T. \tag{3.11}$$

Here T is the transformation matrix containing only the geometric information of the nodal positions of S. For example, if $e(x, y) = [1, x, y, x^2, xy, y^2]$, then

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & x_1 & y_1 & 0 & 0 & 0 \\ 0 & x_2 & y_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1^2 & x_1y_1 & y_1^2 \\ 0 & 0 & 0 & 2x_1x_2 & x_1y_2 + x_2y_1 & 2y_1y_2 \\ 0 & 0 & 0 & x_2^2 & x_2y_2 & y_2^2 \end{bmatrix}.$$

One can also show that

$$dV = dx\, dy = 2V d\xi\, d\eta, \tag{3.12}$$

where $V = \frac{1}{2}|\boldsymbol{r}_1 \times \boldsymbol{r}_2|$ is the volume of S. Substituting Eqs. (3.11) and (3.12) into Eq. (3.8), we obtain

$$L = [e_1(\xi,\eta), \ldots, e_N(\xi,\eta)] \begin{bmatrix} \int_{C_{,1}} e_1(\xi,\eta)\, d\xi\, d\eta & \cdots & \int_{C_{,1}} e_N(\xi,\eta)\, d\xi\, d\eta \\ \vdots & \cdots & \vdots \\ \int_{C_{,N}} e_1(\xi,\eta)\, d\xi\, d\eta & \cdots & \int_{C_{,N}} e_N(\xi,\eta)\, d\xi\, d\eta \end{bmatrix}^{-1} \begin{bmatrix} \frac{V_{,1}}{2V} & & \\ & \cdots & \\ & & \frac{V_{,N}}{2V} \end{bmatrix}. \tag{3.13}$$

From the above equation, the cardinal basis functions can be made independent of the nodal positions of S if each and every $V_{,j}$ is proportional to V. This can be achieved by subdividing the SV into polygonal CVs with straight edges. Therefore, different shapes of triangles have the identical cardinal bases $L_j(\xi,\eta)$ in the transformed space D if they are similarly partitioned into polygons. Transforming back to the physical space S, although the functions $L_j(x, y)$ may be different for different triangles, their functional values $L_j(x_{rq}, y_{rq})$ at similar points (points having the same (ξ,η) in the transformed space D) are exactly the same. We thus have a universal reconstruction formula, Eq. (3.9), for evaluating the state variable u at similar points. This also implies that the reconstruction needs to be carried out only once, and that can be performed using any shape of triangle. Although matrix R may be ill-conditioned, we avoid numerically inverting the matrix by using *Mathematica* [39] to derive the reconstruction coefficients analytically using exact arithmetic. These coefficients are identical for all triangles. The exact integrations of polynomials over arbitrary polygons can be found in [28].

Note that one of the subtle differences between a FV and a SV method is that all the CVs in a SV use the *same* data reconstruction. As a result, it is not necessary to use a Riemann flux or flux splitting for the interior boundaries between the CVs inside a particular SV because the state variable is continuous across the interior CV boundaries. Riemann fluxes are only necessary at the boundaries of the SV. The most significant advantage of the SV method, compared with the FV method, is that the reconstruction for a particular cell type (e.g., triangles) with a certain CV subdivision (e.g., those shown in Figs. 2–4) is exactly the same. Therefore, the memory- and CPU-intensive reconstructions used in a FV method are solved analytically without taking any extra memory in the SV method. Furthermore, exact fluxes rather than Riemann fluxes are used at the interior boundaries of the CVs, resulting again in significant savings because the Riemann flux is usually several times more expensive to compute than the exact flux.

4. CONVERGENT LINEAR, QUADRATIC, AND CUBIC TRIANGULAR SPECTRAL VOLUMES

Based on the discussions in the last section, it is clear that the reconstruction problem is equivalent for all triangles. We therefore focus our attention on the reconstruction problem in an equilateral triangle E, as shown in Fig. 5b. In partitioning E into N nonoverlapping CVs, we further require the CVs to satisfy the following three conditions:

1. The CVs are "symmetric" with respect to all symmetries of the triangle.
2. All CVs are convex.
3. All CVs have straight sides; i.e., the CVs are polygons.

We believe the symmetry and convexity requirement is important for achieving the best possible accuracy and robustness. The requirement of polygons simplifies the formulation of the SV method. To handle curved boundaries, isoparametric SVs will be used. The curved boundary will be represented using high-order polynomials compatible with the polynomial interpolation inside the SV. Then the isoparametric SVs will be transformed to the standard triangle, which will be partitioned in the usual manner. Surface integrals will be performed with respect to the standard triangle. It is obvious that a CV containing the centroid of E must be symmetric with respect to the three edges and vertices, and at most one such CV can exist. This CV, if it exists, is thus said to possess degree 1 symmetry (or 1 symmetry, in short). Similarly, CVs with degree 3 and 6 symmetries can also be defined. For example, if a CV is said to possess degree 3 symmetry, then two other symmetric CVs must exist in the same partition. We denote by n_1, n_3, and n_6 the degree 1, 3, and 6 symmetry groups in a partition with $n_1 = 0$ or 1. Then the total number of CVs in the partition is $n_1 + 3n_3 + 6n_6$. In order to support the unique reconstruction of degree m polynomial, the total number of CVs must be identical to the dimension of the polynomial space; i.e.,

$$n_1 + 3n_3 + 6n_6 = \frac{(m+1)(m+2)}{2}. \tag{4.1}$$

The solutions of (4.1) can be used to guide the partition of E once m is given. For example, for $m = 1, \ldots, 5$, all possible solutions are summarized in Table I. Some possible partitions of the standard triangle corresponding to these solutions for $m = 1, 2, 3$ are shown in Figs. 2–4. Next the question of how these partitions perform in a data reconstruction needs to be answered.

Given any partition, the reconstruction matrix R in (3.5) must be nonsingular. In this case, the expansion coefficients can be solved from (3.6). Note that once the polynomial basis is given, the matrix is solely determined by the partition Π_m of E. For a linear reconstruction using three CVs, it is well-known that the reconstruction is nonsingular as long as the centroids of the CVs are not colinear. Unfortunately no such simple criteria are known for higher order reconstructions in E. We therefore have to compute the determinant of the reconstruction matrix to determine whether it is singular. As a matter of fact, straightforward

TABLE I

Solutions for $m = 1, \ldots, 5$

m	n_1	n_3	n_6
1	0	1	0
2	0	2	0
	0	0	1
3	1	3	0
	1	1	1
4	0	5	0
	0	3	1
	0	1	2
5	0	7	0
	0	5	1
	0	3	2
	0	1	3

computations indicated that several partitions shown in Figs. 3 and 4 are singular. For example, the quadratic SV in Fig. 3b and the cubic SV in Fig. 4d are verified to be singular, and they are excluded from further considerations.

In [38], the first paper on the SV method, it was shown that not all nonsingular reconstructions are convergent. For example, high-order polynomial reconstructions based on equidistant CVs in one dimension are not convergent although the reconstructions are nonsingular. We believe this is the direct consequence of the Runge phenomenon. Therefore some means to quantify the quality of the reconstructions needs to be identified.

Assume that we have a nonsingular partition Π_m of E. For $u \in \mathcal{C}(E)$, we then have

$$p_m(\xi, \eta) = \sum_{j=1}^{N} L_j(\xi, \eta)\bar{u}_{,j}. \tag{4.2}$$

The cardinal basis function has the following property:

$$\frac{1}{V_{,j}} \int_{C_{,j}} L_j(\xi, \eta)\, d\xi\, d\eta = \delta_{i,j}, \quad 1 \le i, j \le N. \tag{4.3}$$

Denote $p_m = \Gamma_\Pi(u)$, where Γ_Π is an operator which maps $\mathcal{C}(E)$ onto $P_m(E)$. It is obvious that Γ_Π is a linear projection operator because

- $\Gamma_\Pi(u + v) = \Gamma_\Pi(u) + \Gamma_\Pi(v), \forall u \in \mathcal{C}(E), v \in \mathcal{C}(E)$;
- $\Gamma_\Pi(cu) = c\Gamma_\Pi(u)$ for any real constant c;
- $\Gamma_\Pi p = p$ for $p \in P_m$.

When both spaces $\mathcal{C}(E)$ and $P_m(E)$ are equipped with the supremum or uniform norm, i.e., $\|\bullet\| = \|\bullet\|_\infty = \max|\bullet|$, the norm of this projection operator can be defined as

$$\|\Gamma_\Pi\| = \sup_{u \neq 0} \frac{\|\Gamma_\Pi u\|}{\|u\|}. \tag{4.4}$$

Because $\bar{u}_{,j} \le \|u\|$, $j = 1, \ldots, N$, it is obvious that

$$\frac{\|\Gamma_\Pi u\|}{\|u\|} = \frac{\left\| \sum_{j=1}^{N} L_j(\xi, \eta)\bar{u}_j \right\|}{\|u\|} \le \frac{\left\| \sum_{j=1}^{N} |L_j(\xi, \eta)\bar{u}_j| \right\|}{\|u\|} \le \frac{\left\| \sum_{j=1}^{N} |L_j(\xi, \eta)| \|u\| \right\|}{\|u\|}$$
$$= \left\| \sum_{j=1}^{N} |L_j(\xi, \eta)| \right\|. \tag{4.5}$$

Therefore we can easily see that

$$\|\Gamma_\Pi\| = \max_{\xi \in E} \sum_{j=1}^{N} |L_j(\xi, \eta)|. \tag{4.6}$$

The function $\lambda(\xi, \eta) = \sum_{j=1}^{N} |L_j(\xi, \eta)|$ is usually referred to as the Lebesgue function of the interpolation, and $\|\Gamma_\Pi\|$ is called the Lebesgue constant, which is of interest for the following two reasons [8]:

(i) If p_m^* is the best uniform approximation to u on E, then

$$\|u - \Gamma_\Pi u\| \le (1 + \|\Gamma_\Pi\|)\|u - p_m^*\|, \tag{4.7}$$

because

$$\begin{aligned}\|u - \Gamma_\Pi u\| &= \|u - p_m^* - (\Gamma_\Pi u - p_m^*)\| \\ &= \|u - p_m^* - \Gamma_\Pi(u - p_m^*)\| \\ &\le \|u - p_m^*\| + \|\Gamma_\Pi(u - p_m^*)\| \\ &\le \|u - p_m^*\| + \|\Gamma_\Pi\|\|(u - p_m^*)\| \\ &= (1 + \|\Gamma_\Pi\|)\|u - p_m^*\|.\end{aligned}$$

(ii) For $p \in P_m$,

$$\|p\| \le \|\Gamma_\Pi\| \max_{1\le j\le N} \bar{u}_{,j}, \tag{4.8}$$

which is obvious from (4.2). Thus $\|\Gamma_\Pi\|$ gives a simple method of bounding the interpolation polynomial. It is obvious from (4.7) that the smaller the Lebesgue constant, the better the interpolation polynomial. Therefore the problem becomes finding the partition with a small Lebesgue constant, if not the smallest one possible. In this paper, our focus is to construct good enough SV partitions so that the interpolation polynomial is convergent when the computational grid is refined. The Lebesgue constant is used as the criterion to judge the quality of the partitions. The optimization of the partitions will be the subject of a future publication.

The problem of partitioning the equilateral triangle E into N symmetric CVs which can support nonsingular polynomial interpolations is not trivial and is much more complex than the problem of determining a set of points in E which support Lagrange interpolations [8, 11]. What we are trying to accomplish in the paper is to identify partitions of E which support linear-to-cubic reconstructions with relatively small Lebesgue constants.

Linear Spectral Volume ($m = 1$)

Based on the solution of (4.1), it is obvious that two partitions are possible, as shown in Figs. 2a and 2b, which are named Type 1 and Type 2 partitions. Since the centroids of the CVs are noncolinear, both partitions are admissible. Note that the CVs in both partitions possess a degree 3 symmetry. The cardinal basis functions $L_j(\xi, \eta)$ are plotted in Fig. 6 for both Type 1 and 2 partitions. Furthermore, the Lebesgue constants are 13/3 (4.3333) and 43/15 (2.8667) for Type 1 and 2 partitions, respectively. Note that the Type 2 SV has a much smaller Lebesgue constant than the Type 1 SV, indicating that the L_∞ error with the Type 2 SV should be smaller than the error with the Type 1 SV.

For a linear reconstruction, only one Gauss quadrature point is required for a surface integral. This quadrature point is located at the center of an edge. Due to the symmetry, we therefore only need to compute and store the functional values of the cardinal bases at two quadrature points, a total of six coefficients. These coefficients are the *same* for all triangles with similar partitions.

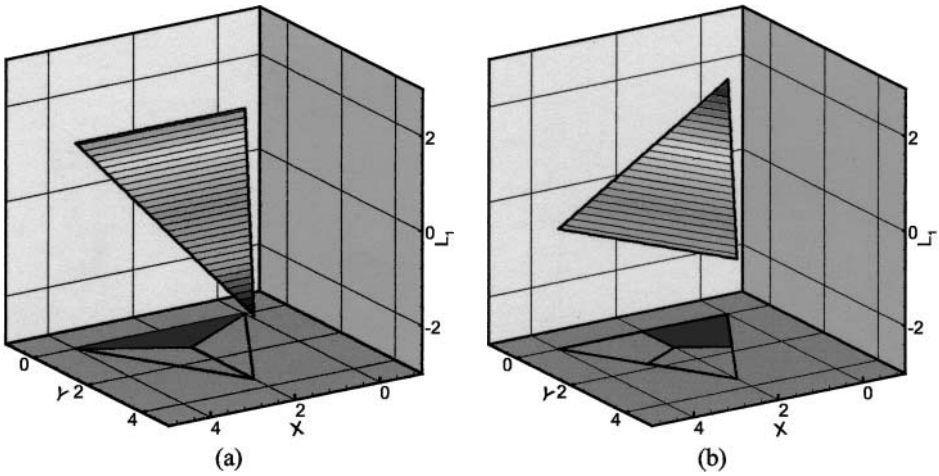


FIG. 6. Shape functions for linear spectral volumes. (a) Type 1; (b) Type 2.

Quadratic Spectral Volume ($m = 2$)

Equation (4.1) has two solutions for $m = 2$, and the possible partitions are shown in Fig. 3. As mentioned earlier, the partition shown in Fig. 3b is singular and is therefore not admissible. The partition presented in Fig. 3a is not unique in the sense that the position of one of the two vertices on an edge of the triangle can change; i.e., the length d shown in Fig. 3a can be any real number in (0, 0.5) assuming the length of the edge is 1. It seems that with any d, the partition is admissible. In our numerical studies, two different values of d were tested, namely $d = 1/3$ and $1/4$ (corresponding to the Gauss–Lobatto points on the edge), which are called Type 1 and 2 partitions, respectively. The Lebesgue constant for the Type 1 partition is 9.3333, and for the Type 2 partition is 8. Therefore, the Type 2 partition is expected to yield more-accurate numerical results. The cardinal basis functions $L_j(\xi, \eta)$ are plotted in Fig. 7 for the Type 2 partition. For a quadratic reconstruction, two

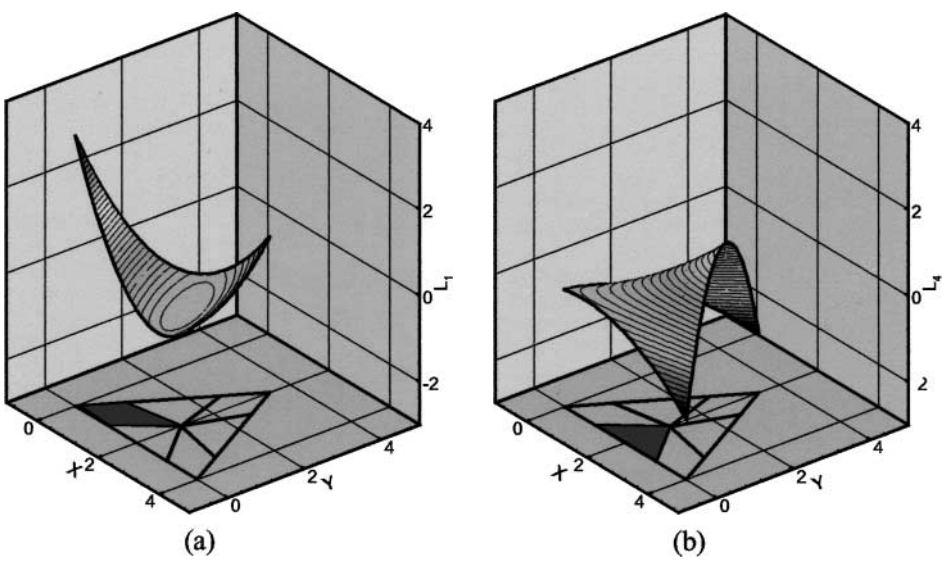


FIG. 7. Shape functions for the Type 2 quadratic spectral volume.

TABLE II
Lebesgue Constants for the Partition Shown in Fig. 4b

d	Lebesgue constant
1/6	8.21499
1/7	6.71178
1/8	5.71904
1/10	4.49231
1/15	3.44485
1/20	3.57595
1/25	3.65981
1/100	3.93353

Gauss quadrature points are required for a surface integral. Due to the symmetry, there are 30 coefficients, corresponding the functional values of the cardinal bases at five quadrature points, which need to be computed and stored.

Cubic Spectral Volume ($m = 3$)

Equation (4.1) has two different solutions for $m = 3$, and several possible partitions are shown in Fig. 4. As mentioned earlier, the partition shown in Fig. 4d is singular and is not admissible. Although the partitions presented in Figs. 4a–4c look quite different, the first two partitions can be viewed as the limiting cases of the partition shown in Fig. 4c. Therefore we can claim that all three partitions (4a–4c) have the same general topology. The partition requires the locations of three vertices, I, II, and III, as shown in Fig. 4c. The optimization of these points will be the topic of a future paper. In this paper, our focus is the partition shown in Fig. 4b, in which one parameter d can be changed to obtain different partitions. In fact, the Lebesgue constants for partitions with a set of d values are presented in Table II. It is interesting to note that among this set of d values, the Lebesgue constant reaches a smallest value of 3.44485 at $d = 1/15$ from a value of 8.21499 at $d = 1/6$. When d is smaller than $1/15$, the Lebesgue constant starts to increase. For presentation purposes, we call the partition shown in Fig. 4a the Type 1 partition. The partition shown in Fig. 4b with $d = 1/6$ is called the Type 2 partition, and with $d = 1/15$ the Type 3 partition. It is expected that the Type 3 partition should give the most accurate numerical solution in the uniform norm. The Lebesgue constant for the Type 1 partition is $167/12$ (13.9), which is significantly larger than those for the Type 2 and 3 partitions. Numerical results to be presented later confirm that the Type 1 partition is not convergent with grid refinement. Several of the cardinal basis functions for the Type 1 partition are plotted in Fig. 8, and for the Type 3 partition are plotted in Fig. 9. For a cubic reconstruction, still two Gauss quadrature points are required for a surface integral. Again, due to the symmetry, we only need to compute and store the cardinal basis functional values at 10 quadrature points, a total of 100 coefficients.

5. MULTIDIMENSIONAL LIMITERS

The Gibbs phenomenon associated with high-order schemes in the presence of discontinuities causes loss of monotonicity in the solution of hyperbolic conservation laws. Godunov

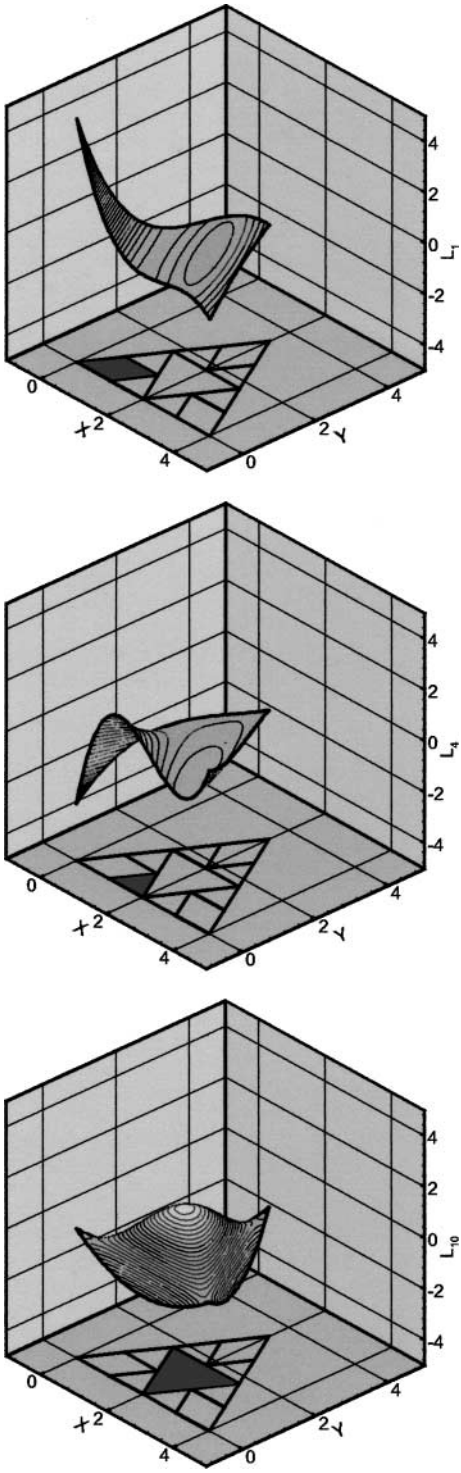


FIG. 8. Shape functions for the Type 1 cubic spectral volume.

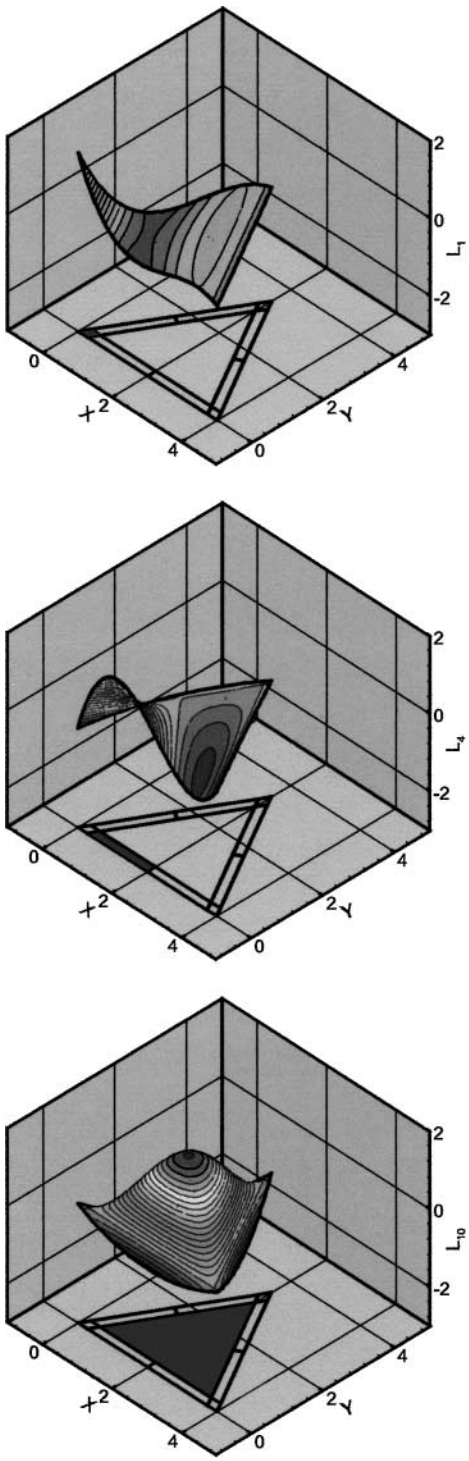


FIG. 9. Shape functions for the Type 3 cubic spectral volume.

[18] first proved that there are no linear second-order or higher schemes which guarantee monotonicity. Therefore high-order monotonic schemes, if they exist, must be nonlinear. One effective approach to achieve monotonicity is to limit the reconstructed solution so that a monotonicity constraint is satisfied. As pointed out by many researchers, strict monotonicity seems to conflict with a uniform high order of accuracy [12]. In order to recover a uniform order of accuracy away from discontinuities, the TVB (total variation bounded) idea [33] is employed here.

We again consider a SV S_i with N CVs. Given cell-averaged state variables for all the CVs $\{\bar{u}_{i,j}\}$, a polynomial reconstruction $p_i(x, y)$ of at most order $k-1$ exists which satisfies

$$\int_{C_{i,j}} p_i(x, y)\, dV = \bar{u}_{i,j} V_{i,j}, \quad j = 1, \ldots, N. \tag{5.1}$$

Recall that this polynomial reconstruction is then used to compute the state variables at the CV boundaries, which are, in turn, used in the update of the solution at the next time level:

$$\frac{d\bar{u}_{i,j}}{dt} + \frac{1}{V_{i,j}} \sum_{r=1}^{K} \sum_{q=1}^{J} w_{rq} F_{\text{Riem}}(p_i(x_{rq}, y_{rq}), p_{i,r}(x_{rq}, y_{rq}), \boldsymbol{n}_r) A_r = 0 \tag{5.2}$$

Denote

$$\Delta u_{rq} = p_i(x_{rq}, y_{rq}) - \bar{u}_{i,j}, \quad r = 1, \ldots, K; q = 1, \ldots, J.$$

Following the TVB idea, if

$$|\Delta u_{rq}| \le 4Mh_{rq}^2, \quad r = 1, \ldots, K; q = 1, \ldots, J, \tag{5.3}$$

it is not necessary to do any data limiting. In (5.3), M represents some measure of the second derivative of the solution, and h_{rq} is the distance from point (x_{rq}, y_{rq}) to the centroid of $C_{i,j}$. It is obvious that this multidimensional TVB limiter degenerates into the one-dimensional TVB limiter. Using the fact that in two dimensions $4h_{rq}^2 \propto V_{i,j}$, we can further simplify (5.3) by replacing $4Mh_{rq}^2$ with $MV_{i,j}$. With the new formula, we do not need to compute or store the distances from the cell centroid to the quadrature points. In this paper, we select M to be close to the maximum absolute value of the second derivative over the computational domain. If for any value of r and q (5.3) is violated, it is assumed that $C_{i,j}$ is near a steep gradient and data limiting is necessary. Instead of using the polynomial $p_i(x, y)$ in $C_{i,j}$, we assume that data is linear in $C_{i,j}$; i.e,

$$u_{i,j}(x, y) = \bar{u}_{i,j} + \nabla u_{i,j} \cdot (\boldsymbol{r} - \boldsymbol{r}_{i,j}), \quad \forall \boldsymbol{r} \in C_{i,j}, \tag{5.4}$$

where $\boldsymbol{r}_{i,j}$ is the position vector of the centroid of $C_{i,j}$. In order to achieve the highest resolution, we need to maximize the magnitude of the solution gradient $\nabla u_{i,j}$ in $C_{i,j}$. At the same time, we require that the reconstructed solutions at the quadrature points of $C_{i,j}$ satisfy the monotonicity constraint

$$\bar{u}_{i,j}^{\min} \le u_{i,j}(x_{rq}, y_{rq}) \le \bar{u}_{i,j}^{\max}; \quad r = 1, \ldots, K, \quad q = 1, \ldots, J, \tag{5.5}$$

where $\bar{u}_{i,j}^{\min}$ and $\bar{u}_{i,j}^{\max}$ are the minimum and maximum cell-averaged solutions among all its neighboring CVs sharing a face with $C_{i,j}$; i.e.,

$$\begin{aligned} \bar{u}_{i,j}^{\max} &= \max\Big(\bar{u}_{i,j}, \max_{1\le r\le K} \bar{u}_{i,j,r}\Big), \\ \bar{u}_{i,j}^{\min} &= \min\Big(\bar{u}_{i,j}, \min_{1\le r\le K} \bar{u}_{i,j,r}\Big), \end{aligned} \tag{5.6}$$

where $\bar{u}_{i,j,r}$ denotes the cell-averaged solution at the neighboring CV of $C_{i,j}$ sharing face r. If the solution is linear, it is obvious that the solution value at the centroid of $C_{i,j}$ is the same as $\bar{u}_{i,j}$, the cell-average solution. Several different approaches are possible in estimating $\nabla u_{i,j}$, depending on the computational complexity. Three approaches are outlined here.

Approach 1. Consider a CV with K faces. Using cell-averaged data at $C_{i,j}$ and its neighbors, we can usually construct K different gradients. For example, the gradient reconstructed from $(\bar{u}_{i,j}, \bar{u}_{i,j,r}, \bar{u}_{i,j,r+1})$ is denoted $\nabla u_{i,j,r}$ with $\bar{u}_{i,j,K+1} = \bar{u}_{i,j,1}$. In addition, another gradient can be constructed through a least-squares reconstruction algorithm using the cell-averaged data at all the face-neighbor cells. This gradient is denoted by $\nabla u_{i,j,K+1}$. Using any of the gradients, the state variable at the quadrature points of $C_{i,j}$ can be computed. If any of the reconstructed variable at the quadrature points is out of the range $[\bar{u}_{i,j}^{\min}, \bar{u}_{i,j}^{\max}]$, the gradient is limited; i.e.,

$$\nabla u_{i,j,r} \Leftarrow \varphi \nabla u_{i,j,r}, \tag{5.7}$$

where $\varphi \in [0, 1]$ is calculated from

$$\varphi = \begin{cases} \min\Big(1, \frac{\Delta u_{rq}}{\bar{u}_{i,j}^{\max} - \bar{u}_{i,j}}\Big) & \text{if } \Delta u_{rq} > 0 \\ \min\Big(1, \frac{\Delta u_{rq}}{\bar{u}_{i,j}^{\min} - \bar{u}_{i,j}}\Big) & \text{if } \Delta u_{rq} < 0 \\ 1 & \text{otherwise.} \end{cases} \tag{5.8}$$

After this limiting step, we have $K + 1$ gradients which all satisfy the monotonicity constraint given in (5.5). Then the gradient with the largest magnitude is selected to maximize the gradient; i.e.,

$$\nabla u_{i,j} = \max_r |\nabla u_{i,j,r}|. \tag{5.9}$$

This limiter is similar to the maximum limited gradient (MLG) limiter developed by Batten *et al.* [7]. It was proven by Liu [26] that the above limiter satisfies a maximum principle for triangular grids. The MLG limiter is an analog of the Superbee limiter in one dimension. This limiter is therefore called **Superbee limiter** in this paper. This limiter has the advantage of minimum numerical dissipation but is expensive to compute. A more efficient limiter is given next.

Approach 2. Only one gradient is computed, i.e., $\nabla u_{i,j,K+1}$, given in approach 1, which is computed with data at all face-neighbor cells using a least-squares linear reconstruction algorithm. Again this gradient is limited using (5.8), so the reconstructed solutions at all the quadrature points satisfy the monotonicity constraint (5.5). This limiter is similar to the minmod limiter in one dimension and is called the **Minmod limiter** here.

Approach 3. Note that in both approaches 1 and 2 the gradients of the solution for each CV must be reconstructed using data in a neighborhood of the CV. This reconstruction can be quite memory and CPU intensive. In this approach, we avoid a separate data reconstruction by reusing the polynomial reconstruction already available for the SV. For each CV, we use the gradient of the reconstructed polynomial at the CV centroid; i.e.,

$$\nabla u_{i,j} = \left(\frac{\partial p_m}{\partial x}, \frac{\partial p_m}{\partial y}\right)\Bigg|_{r_{i,j}} = \left(\frac{\partial p_m}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial p_m}{\partial \eta}\frac{\partial \eta}{\partial x}, \frac{\partial p_m}{\partial \xi}\frac{\partial \xi}{\partial y} + \frac{\partial p_m}{\partial \eta}\frac{\partial \eta}{\partial y}\right)\Bigg|_{r_{i,j}}. \tag{5.10}$$

For the transformation given in (3.10), it is obvious that

$$\begin{pmatrix} \frac{\partial p_m}{\partial x} \\ \frac{\partial p_m}{\partial y} \end{pmatrix} = \begin{pmatrix} \xi_x & \eta_x \\ \xi_y & \eta_y \end{pmatrix} \begin{pmatrix} \frac{\partial p_m}{\partial \xi} \\ \frac{\partial p_m}{\partial \eta} \end{pmatrix} = \frac{1}{x_1 y_2 - x_2 y_1} \begin{pmatrix} y_2 & -y_1 \\ -x_2 & x_1 \end{pmatrix} \begin{pmatrix} \frac{\partial p_m}{\partial \xi} \\ \frac{\partial p_m}{\partial \eta} \end{pmatrix} \tag{5.11}$$

Because

$$p_m(\xi, \eta) = \sum_{j=1}^{N} L_j(\xi, \eta)\bar{u}_{i,j},$$

we have

$$\frac{\partial p_m(\xi, \eta)}{\partial \xi} = \sum_{j=1}^{N} \frac{\partial L_j(\xi, \eta)}{\partial \xi}\bar{u}_{i,j}, \tag{5.12a}$$

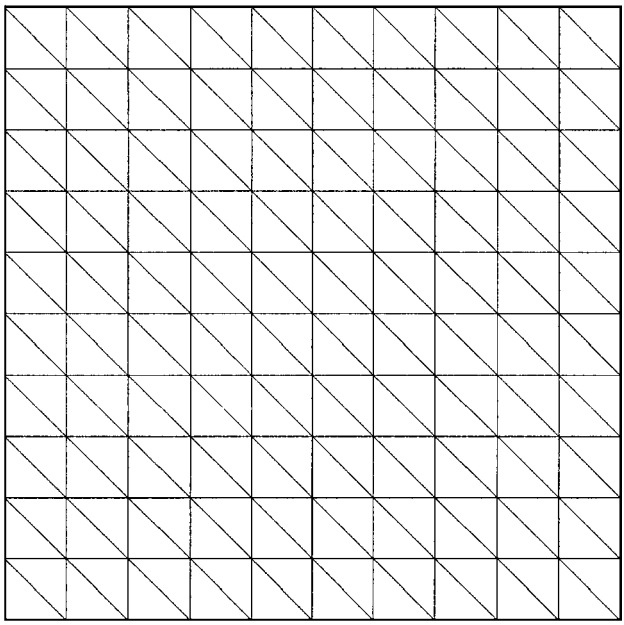
$$\frac{\partial p_m(\xi, \eta)}{\partial \eta} = \sum_{j=1}^{N} \frac{\partial L_j(\xi, \eta)}{\partial \eta}\bar{u}_{i,j}. \tag{5.12b}$$

The first derivatives of the shape function can be obtained analytically. The gradient for each CV is then limited if necessary with the same approach outlined in approach 1. Obviously this is the most efficient among the three limiters. This limiter is named **CV limiter.**

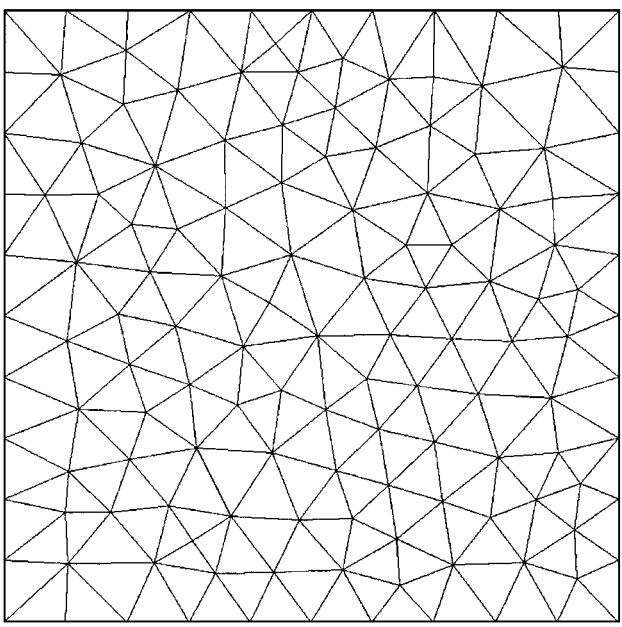
For comparison purposes, we also used another simple limiter, which is called the **Clip limiter.** In this limiter, zero gradient (piecewise constant distribution) is used in CVs wherever limiting is necessary.

Note that if parameter $M = 0$, the TVB limiters are similar to TVD (total variation diminishing) limiters, which strictly enforce monotonicity by sacrificing accuracy near extrema.

The availability of cell-averaged data on the CVs inside a SV makes this CV-based data limiting possible, whereas in the DG method, one can only do an element-based data limiting. Due to the increased local resolution, the SV method is expected to have higher resolutions for discontinuities than the DG method. The improved resolution has been demonstrated in one dimension [38].



(a)



(b)

FIG. 10. Regular and irregular "10 × 10 × 2" computational grids.

6. NUMERICAL TESTS

6.1. Accuracy Study with 2D Linear Wave Equation

Time-Accurate Problem

In this case, we test the accuracy of the SV method on the two-dimensional linear equation

$$\begin{aligned} &\frac{\partial u}{\partial t}+\frac{\partial u}{\partial x}+\frac{\partial u}{\partial y}=0, \quad -1\le x\le 1, \quad -1\le y\le 1,\\ &u(x, y, 0)=u_0(x, y), \quad \text{periodic boundary condition.} \end{aligned} \tag{6.1}$$

The initial condition is $u_0(x, y) = \sin \pi(x + y)$. A fourth-order-accurate Gauss quadrature formula [28] is used to compute the CV-averaged initial solutions. These CV-averaged solutions are then updated at each time step using the third-order TVD Runge–Kutta scheme presented earlier. The numerical simulation is carried until $t = 1$ on two different triangular grids, as shown in Fig. 10. One gird is generated from a uniform Cartesian grid by cutting each Cartesian cell into two triangles and is called the regular grid. The other grid is generated with an unstructured grid generator and is called the irregular grid. The finer irregular grids are generated recursively by cutting each coarser grid cell into four finer grid cells. Note that the cells in the irregular grid have quite different sizes. In Table III, we present the L_1 and L_∞ errors in the CV-averaged solutions produced using second- to fourth-order SV method schemes with SVs shown in Figs. 2–4 on the regular grid. The errors presented in the table are time-step independent because the time step Δt was made small enough so that the errors are dominated by the spatial discretization. In this test, all SVs except the Type 1 cubic SV (shown in Fig. 4a) are convergent with grid refinement on this regular grid. It is obvious that the expected order of accuracy is achieved by all the convergent SVs. It is not surprising that the Type 1 cubic SV is not convergent because of its rather large Lebesgue constant of 13.9. In contrast, the Type 2 and 3 cubic SVs have Lebesgue constants of 8.21 and 3.44, respectively. It is interesting to note that the Type 1 linear SV gives more-accurate results in both the L_1 and L_∞ norms than the Type 2 linear SV even if the Type 1 SV has the larger Lebesgue constant of 4.33 versus that of 2.87 of the Type 2 SV. This indicates that the Lebesgue constant cannot serve as an absolute error estimate but rather is an estimate of the upper bound of the error. For the quadratic and cubic SVs, the partitions with smaller Lebesgue constants do give more-accurate numerical solutions, as shown in Table III.

Next, the L_1 and L_∞ errors in the numerical results computed on the irregular grid using second- to fourth-order SVs are shown in Table IV. This should be a much tougher test case because of the truly unstructured nature of the computational grid. What is striking is that the Type 1 linear SV failed to achieve second-order accuracy on this grid. As a matter of fact, it is only first-order accurate. This may be contributed to the acute angles of the CVs in the partition. Note that both quadratic SVs are convergent and give similar results. Third-order accuracy is achieved by both types of quadratic SVs in the L_1 norm although the numerical order of accuracy in the L_∞ norm is only slightly over second order. We believe this is due to the nonsmoothness of the computational grid. As expected, the Type 1 cubic SV is not convergent on this grid. In addition, the Type 2 cubic SV also showed a nonconvergent behavior in the L_∞ norm on the finest grid. It is nice to see that the Type 3

cubic SV is not only convergent but also fourth-order accurate in both the L_1 and L_∞ norms.

Steady State Problem

One steady state solution for the wave equation (6.1) is $u(x, y) = \sin \pi(x - y)$. In order to test the performance of the SV method for steady state problems, the steady boundary value problem is also studied. Because the wave is traveling in positive x and y directions, inflow boundary conditions are employed at $y = -1$ and $x = -1$, while extrapolation boundary conditions are used at $x = 1$ and $y = 1$. On the inflow boundaries, the exact solutions at the quadrature points are used to evaluate the flux integrals. The simulation is carried out on the

TABLE III

Accuracy of $u_t + u_x + u_y = 0$, with $u_0(x, y) = \sin \pi(x + y)$, at $t = 1$ (Regular Grids)

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
2 (Type 1 SV)	$10 \times 10 \times 2$	3.04e-2	—	4.97e-2	—
	$20 \times 20 \times 2$	7.68e-3	1.98	1.24e-2	2.00
	$40 \times 40 \times 2$	1.92e-3	2.00	3.10e-3	2.00
	$80 \times 80 \times 2$	4.81e-4	2.00	7.75e-4	2.00
	$160 \times 160 \times 2$	1.20e-4	2.00	1.93e-4	2.00
2 (Type 2 SV)	$10 \times 10 \times 2$	4.03e-2	—	6.68e-2	—
	$20 \times 20 \times 2$	1.06e-2	1.93	1.78e-2	1.91
	$40 \times 40 \times 2$	2.71e-3	1.97	4.54e-3	1.97
	$80 \times 80 \times 2$	6.83e-4	1.99	1.14e-3	1.99
	$160 \times 160 \times 2$	1.71e-4	2.00	2.87e-4	1.99
3 (Type 1, $d = 1/3$)	$10 \times 10 \times 2$	4.18e-3	—	7.76e-3	—
	$20 \times 20 \times 2$	5.33e-4	2.97	1.01e-3	2.94
	$40 \times 40 \times 2$	6.73e-5	2.99	1.25e-4	3.01
	$80 \times 80 \times 2$	8.45e-6	2.99	1.55e-5	3.01
	$160 \times 160 \times 2$	1.06e-6	2.99	1.93e-6	3.00
3 (Type 2, $d = 1/4$)	$10 \times 10 \times 2$	4.73e-3	—	7.88e-3	—
	$20 \times 20 \times 2$	4.77e-4	3.31	9.83e-4	3.00
	$40 \times 40 \times 2$	6.04e-5	2.98	1.23e-4	3.00
	$80 \times 80 \times 2$	7.58e-6	2.99	1.53e-5	3.01
	$160 \times 160 \times 2$	9.57e-7	2.99	1.91e-6	3.00
4 (Type 1 SV)	$10 \times 10 \times 2$	1.38e-4	—	4.86e-4	—
	$20 \times 20 \times 2$	8.64e-6	4.00	1.98e-5	4.62
	$40 \times 40 \times 2$	5.47e-7	3.98	1.51e-6	3.71
	$80 \times 80 \times 2$	3.46e-8	3.98	1.17e-7	3.69
	$160 \times 160 \times 2$	4.19e-8	Negative	5.15e-7	Negative
4 (Type 2, $d = 1/6$)	$10 \times 10 \times 2$	9.33e-5	—	3.17e-4	—
	$20 \times 20 \times 2$	5.86e-6	3.99	1.94e-5	4.03
	$40 \times 40 \times 2$	3.70e-7	3.99	1.24e-6	3.95
	$80 \times 80 \times 2$	2.32e-8	4.00	7.78e-8	3.99
	$160 \times 160 \times 2$	1.45e-9	4.00	4.84e-9	4.01
4 (Type 3, $d = 1/15$)	$10 \times 10 \times 2$	7.36e-5	—	2.51e-4	—
	$20 \times 20 \times 2$	4.52e-6	4.03	1.61e-5	3.96
	$40 \times 40 \times 2$	2.81e-7	4.01	1.01e-6	3.99
	$80 \times 80 \times 2$	1.75e-8	4.01	6.30e-8	4.00
	$160 \times 160 \times 2$	1.10e-9	3.99	3.94e-9	4.01

TABLE IV

Accuracy on $u_t + u_x + u_y = 0$, with $u_0(x, y) = \sin \pi(x + y)$, at $t = 1$ (Irregular Grids)

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
2 (Type 1 SV)	$10 \times 10 \times 2$	1.30e-1	—	3.60e-1	—
	$20 \times 20 \times 2$	6.66e-2	0.96	1.91e-1	0.91
	$40 \times 40 \times 2$	3.51e-2	0.92	9.84e-2	0.96
	$80 \times 80 \times 2$	1.85e-2	0.92	4.91e-2	1.00
	$160 \times 160 \times 2$	9.74e-3	0.93	2.86e-2	0.78
2 (Type 2 SV)	$10 \times 10 \times 2$	6.71e-2	—	1.36e-1	—
	$20 \times 20 \times 2$	1.83e-2	1.87	4.42e-2	1.62
	$40 \times 40 \times 2$	4.71e-3	1.96	1.15e-2	1.94
	$80 \times 80 \times 2$	1.19e-3	1.98	2.94e-3	1.97
	$160 \times 160 \times 2$	3.00e-4	1.99	8.85e-4	1.73
3 (Type 1, $d = 1/3$)	$10 \times 10 \times 2$	9.17e-3	—	3.67e-2	—
	$20 \times 20 \times 2$	1.25e-3	2.87	5.28e-3	2.80
	$40 \times 40 \times 2$	1.64e-4	2.93	8.32e-4	2.67
	$80 \times 80 \times 2$	2.15e-5	2.93	1.84e-4	2.18
	$160 \times 160 \times 2$	2.79e-6	2.95	4.05e-5	2.18
3 (Type 2, $d = 1/4$)	$10 \times 10 \times 2$	8.36e-3	—	3.76e-2	—
	$20 \times 20 \times 2$	1.15e-3	2.86	5.63e-3	2.74
	$40 \times 40 \times 2$	1.52e-4	2.92	1.00e-3	2.49
	$80 \times 80 \times 2$	2.01e-5	2.92	2.14e-4	2.22
	$160 \times 180 \times 2$	2.64e-6	2.93	5.31e-5	2.01
4 (Type 1 SV)	$10 \times 10 \times 2$	4.43e-4	—	3.32e-3	—
	$20 \times 20 \times 2$	3.08e-5	3.85	2.56e-4	3.70
	$40 \times 40 \times 2$	2.15e-6	3.84	2.12e-5	3.59
	$80 \times 80 \times 2$	2.48e-7	3.12	5.06e-6	2.07
	$160 \times 160 \times 2$	5.19e-7	Negative	4.49e-5	Negative
4 (Type 2, $d = 1/6$)	$10 \times 10 \times 2$	3.04e-4	—	2.58e-3	—
	$20 \times 20 \times 2$	2.02e-5	3.91	1.73e-4	3.90
	$40 \times 40 \times 2$	1.34e-6	3.91	1.42e-5	3.61
	$80 \times 80 \times 2$	9.61e-8	3.80	1.03e-6	3.79
	$160 \times 160 \times 2$	2.30e-8	2.06	1.23e-6	Negative
4 (Type 3, $d = 1/15$)	$10 \times 10 \times 2$	2.71e-4	—	1.51e-3	—
	$20 \times 20 \times 2$	1.61e-5	4.07	1.14e-4	3.73
	$40 \times 40 \times 2$	9.91e-7	4.02	8.28e-6	3.78
	$80 \times 80 \times 2$	6.17e-8	4.01	5.40e-7	3.94
	$160 \times 160 \times 2$	3.87e-9	3.99	3.79e-8	3.83

irregular grid until a steady state is reached. In all the simulations, the residuals were reduced to machine zero. In Table V, the L_1 and L_∞ errors are presented for second- to fourth-order schemes. It is surprising to see that the Type 1 linear SV gives a more accurate solution in both norms than the Type 2 linear SV on this irregular grid. Recall that in the time-accurate simulation presented earlier on this irregular grid, the Type 1 linear SV failed to achieve second-order accuracy. This may indicate that there is significant error accumulation in the time-accurate simulation with the Type 1 linear SV. Other than that, there are no major surprises. Both quadratic SVs gave reasonable results, while the Type 1 and 2 cubic SVs showed convergence problems on the finest mesh. Once again, the performance of the Type 3 cubic SV is excellent. It is convergent and achieves fourth-order accuracy in both norms.

TABLE V

Accuracy on $u_x + u_y = 0$, with $u(x, y) = \sin \pi(x - y)$ (Irregular Grids)

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
2 (Type 1 SV)	$10 \times 10 \times 2$	8.91e-3	—	4.65e-2	—
	$20 \times 20 \times 2$	2.26e-3	1.98	1.27e-2	1.87
	$40 \times 40 \times 2$	5.78e-4	1.97	3.37e-3	1.91
	$80 \times 80 \times 2$	1.48e-4	1.97	9.03e-4	1.90
	$160 \times 160 \times 2$	3.77e-5	1.97	2.37e-4	1.93
2 (Type 2 SV)	$10 \times 10 \times 2$	1.57e-2	—	6.58e-2	—
	$20 \times 20 \times 2$	3.94e-3	1.99	2.14e-2	1.62
	$40 \times 40 \times 2$	9.99e-4	1.99	5.58e-3	1.94
	$80 \times 80 \times 2$	2.52e-4	1.99	1.47e-3	1.92
	$160 \times 160 \times 2$	6.32e-5	2.00	3.83e-4	1.94
3 (Type 1, $d = 1/3$)	$10 \times 10 \times 2$	1.51e-3	—	1.33e-2	—
	$20 \times 20 \times 2$	2.20e-4	2.78	1.82e-3	2.87
	$40 \times 40 \times 2$	3.07e-5	2.84	2.64e-4	2.75
	$80 \times 80 \times 2$	4.08e-6	2.91	3.93e-5	2.75
	$160 \times 160 \times 2$	5.24e-7	2.96	5.25e-6	2.90
3 (Type 2, $d = 1/4$)	$10 \times 10 \times 2$	1.48e-3	—	1.26e-2	—
	$20 \times 20 \times 2$	2.11e-4	2.81	1.72e-3	2.87
	$40 \times 40 \times 2$	2.88e-5	2.87	2.71e-4	2.67
	$80 \times 80 \times 2$	3.76e-6	2.94	3.83e-5	2.82
	$160 \times 160 \times 2$	4.69e-7	3.00	5.11e-6	2.91
4 (Type 1 SV)	$10 \times 10 \times 2$	9.11e-5	—	1.28e-3	—
	$20 \times 20 \times 2$	6.80e-6	3.74	1.04e-4	3.62
	$40 \times 40 \times 2$	4.21e-7	4.01	7.60e-6	3.77
	$80 \times 80 \times 2$	2.97e-8	3.83	8.75e-7	3.12
	$160 \times 160 \times 2$	3.73e-9	2.99	7.21e-7	0.28
4 (Type 2, $d = 1/6$)	$10 \times 10 \times 2$	9.27e-5	—	1.26e-3	—
	$20 \times 20 \times 2$	6.34e-6	3.87	1.02e-4	3.63
	$40 \times 40 \times 2$	3.93e-7	4.01	7.36e-6	3.79
	$80 \times 80 \times 2$	2.56e-8	3.94	4.83e-7	3.93
	$160 \times 160 \times 2$	1.76e-9	3.86	6.25e-8	2.95
4 (Type 3, $d = 1/15$)	$10 \times 10 \times 2$	1.59e-4	—	9.21e-4	—
	$20 \times 20 \times 2$	9.98e-6	3.99	9.12e-5	3.34
	$40 \times 40 \times 2$	6.26e-7	3.99	6.58e-6	3.79
	$80 \times 80 \times 2$	3.75e-8	4.06	4.49e-7	3.87
	$160 \times 160 \times 2$	2.26e-9	4.05	2.68e-8	4.07

6.2. Accuracy Study with 2D Burgers Equation

In this case, we test the accuracy of the SV method on the two-dimensional nonlinear wave equation:

$$\begin{aligned} &\frac{\partial u}{\partial t} + \frac{\partial u^2/2}{\partial x} + \frac{\partial u^2/2}{\partial y} = 0, \qquad -1 \le x \le 1, \quad -1 \le y \le 1, \\ &u(x, y, 0) = \frac{1}{4} + \frac{1}{2} \sin \pi(x + y), \quad \text{periodic boundary condition.} \end{aligned} \tag{6.2}$$

The initial solution is smooth. Due to the nonlinearity of Burger's equation, discontinuities

TABLE VI

Accuracy on $u_t + uu_x + uu_y = 0$, with $u_0(x, y) = \frac{1}{4} + \frac{1}{2}\sin\pi(x + y)$, at $t = 0.1$ with Regular Grid

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
2 (Type 1 SV)	$10 \times 10 \times 2$	2.05e-3	—	7.20e-3	—
	$20 \times 20 \times 2$	5.31e-4	1.95	2.17e-3	1.73
	$40 \times 40 \times 2$	1.42e-4	1.90	6.63e-4	1.71
	$80 \times 80 \times 2$	3.79e-5	1.91	1.96e-4	1.76
	$160 \times 160 \times 2$	1.03e-5	1.88	5.86e-5	1.74
2 (Type 2 SV)	$10 \times 10 \times 2$	3.35e-3	—	1.18e-2	—
	$20 \times 20 \times 2$	8.07e-4	2.05	3.82e-3	1.63
	$40 \times 40 \times 2$	2.01e-4	2.01	1.02e-3	1.91
	$80 \times 80 \times 2$	5.06e-5	1.99	2.64e-4	1.95
	$160 \times 160 \times 2$	1.27e-5	1.99	6.71e-5	1.98
3 (Type 1, $d = 1/3$)	$10 \times 10 \times 2$	4.44e-4	—	1.93e-3	—
	$20 \times 20 \times 2$	7.81e-5	2.51	4.29e-4	2.17
	$40 \times 40 \times 2$	1.30e-5	2.59	8.84e-5	2.28
	$80 \times 80 \times 2$	2.09e-6	2.64	1.51e-5	2.55
	$160 \times 160 \times 2$	3.24e-7	2.69	2.57e-6	2.55
3 (Type 2, $d = 1/4$)	$10 \times 10 \times 2$	4.31e-4	—	1.91e-3	—
	$20 \times 20 \times 2$	7.44e-5	2.53	4.22e-4	2.18
	$40 \times 40 \times 2$	1.23e-5	2.60	7.93e-5	2.41
	$80 \times 80 \times 2$	1.96e-6	2.65	1.45e-5	2.45
	$160 \times 160 \times 2$	3.00e-7	2.71	2.42e-6	2.58
4 (Type 2, $d = 1/6$)	$10 \times 10 \times 2$	3.26e-5	—	2.40e-4	—
	$20 \times 20 \times 2$	2.22e-6	3.88	2.15e-5	3.48
	$40 \times 40 \times 2$	1.58e-7	3.81	1.74e-6	3.63
	$80 \times 80 \times 2$	1.02e-8	3.95	1.18e-7	3.88
	$160 \times 160 \times 2$	6.54e-10	3.96	7.68e-9	3.94
4 (Type 3, $d = 1/15$)	$10 \times 10 \times 2$	4.10e-5	—	3.12e-4	—
	$20 \times 20 \times 2$	2.69e-6	3.93	2.82e-5	3.47
	$40 \times 40 \times 2$	1.85e-7	3.86	2.23e-6	3.66
	$80 \times 80 \times 2$	1.24e-8	3.90	1.60e-7	3.80
	$160 \times 160 \times 2$	8.21e-10	3.92	1.09e-8	3.88

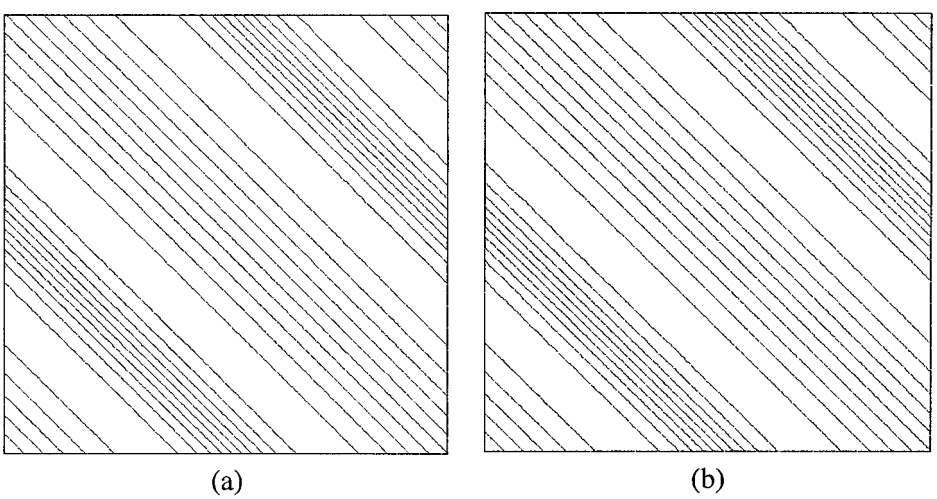


FIG. 11. Exact and computational solutions of Burger's equation at $t = 0.1$. (a) Exact solution at $t = 0.1$; (b) Numerical solution on the $20 \times 20 \times 2$ irregular grid using the Type 1 quadratic SV.

will develop in the solution. Therefore we also test the capability of the SV method to achieve uniform high-order accuracy away from discontinuities. At $t = 0.1$, the exact solution is still smooth, as shown in Fig. 11a. The numerical simulation is therefore carried out until $t = 0.1$ without the use of limiters on both the regular and irregular grids, as shown in Fig. 10. The numerical solution on the $20 \times 20 \times 2$ irregular grid computed with the Type 1 quadratic SV (third-order accurate) is displayed in Fig. 11b. Notice that the visual agreement between the numerical and exact solutions is excellent. In Table VI, we present the L_1 and L_∞ errors produced using various SVs on the regular grid, while in Table VII the errors on the irregular grids are presented. The Type 1 cubic SV is now excluded because it is nonconvergent on any grids. The performance of the SV method on the nonlinear Burger equation is quite similar to the performance on the linear wave equation, although there is a slight loss of accuracy (from 0.1 to 0.6 orders), especially on the irregular grid in the L_∞ norm, probably due to the nonlinear nature of Burger's equation. Once again, the Type 1 linear SV has difficulty in achieving second-order accuracy on the irregular grid in both norms.

TABLE VII

Accuracy on $u_t + uu_x + uu_y = 0$, with $u_0(x, y) = \frac{1}{4} + \frac{1}{2} \sin \pi(x + y)$, at $t = 0.1$ with Irregular Grid

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
2 (Type 1 SV)	$10 \times 10 \times 2$	4.26e-3	—	4.32e-2	—
	$20 \times 20 \times 2$	1.52e-3	1.49	2.34e-2	0.88
	$40 \times 40 \times 2$	5.82e-4	1.38	1.22e-2	0.94
	$80 \times 80 \times 2$	2.42e-4	1.27	6.09e-3	1.00
	$160 \times 160 \times 2$	1.06e-4	1.19	2.99e-3	1.03
2 (Type 2 SV)	$10 \times 10 \times 2$	5.79e-3	—	2.96e-2	—
	$20 \times 20 \times 2$	1.46e-3	1.99	9.15e-3	1.69
	$40 \times 40 \times 2$	3.67e-4	1.99	2.87e-3	1.67
	$80 \times 80 \times 2$	9.40e-5	1.97	8.78e-4	1.71
	$160 \times 160 \times 2$	2.39e-5	1.98	3.54e-4	1.31
3 (Type 1, $d = 1/3$)	$10 \times 10 \times 2$	6.37e-4	—	4.71e-3	—
	$20 \times 20 \times 2$	1.21e-4	2.40	1.26e-3	1.90
	$40 \times 40 \times 2$	2.02e-5	2.58	3.52e-4	1.84
	$80 \times 80 \times 2$	3.22e-6	2.65	8.21e-5	2.10
	$160 \times 160 \times 2$	5.02e-7	2.68	1.66e-5	2.31
3 (Type 2, $d = 1/4$)	$10 \times 10 \times 2$	6.28e-4	—	3.93e-3	—
	$20 \times 20 \times 2$	1.17e-4	2.42	1.09e-3	1.85
	$40 \times 40 \times 2$	1.91e-5	2.61	3.05e-4	1.84
	$80 \times 80 \times 2$	3.01e-6	2.67	7.16e-5	2.09
	$160 \times 160 \times 2$	4.63e-7	2.70	1.43e-5	2.32
4 (Type 2, $d = 1/6$)	$10 \times 10 \times 2$	7.87e-5	—	1.02e-3	—
	$20 \times 20 \times 2$	6.07e-6	3.70	1.00e-4	3.35
	$40 \times 40 \times 2$	4.55e-7	3.74	9.62e-6	3.38
	$80 \times 80 \times 2$	3.44e-8	3.73	8.55e-7	3.49
	$160 \times 160 \times 2$	2.79e-9	3.62	8.75e-8	3.29
4 (Type 3, $d = 1/15$)	$10 \times 10 \times 2$	9.71e-5	—	1.29e-3	—
	$20 \times 20 \times 2$	7.17e-6	3.76	1.24e-4	3.38
	$40 \times 40 \times 2$	5.20e-7	3.79	1.07e-5	3.53
	$80 \times 80 \times 2$	3.79e-8	3.79	9.34e-7	3.52
	$160 \times 160 \times 2$	2.88e-9	3.72	8.34e-8	3.49

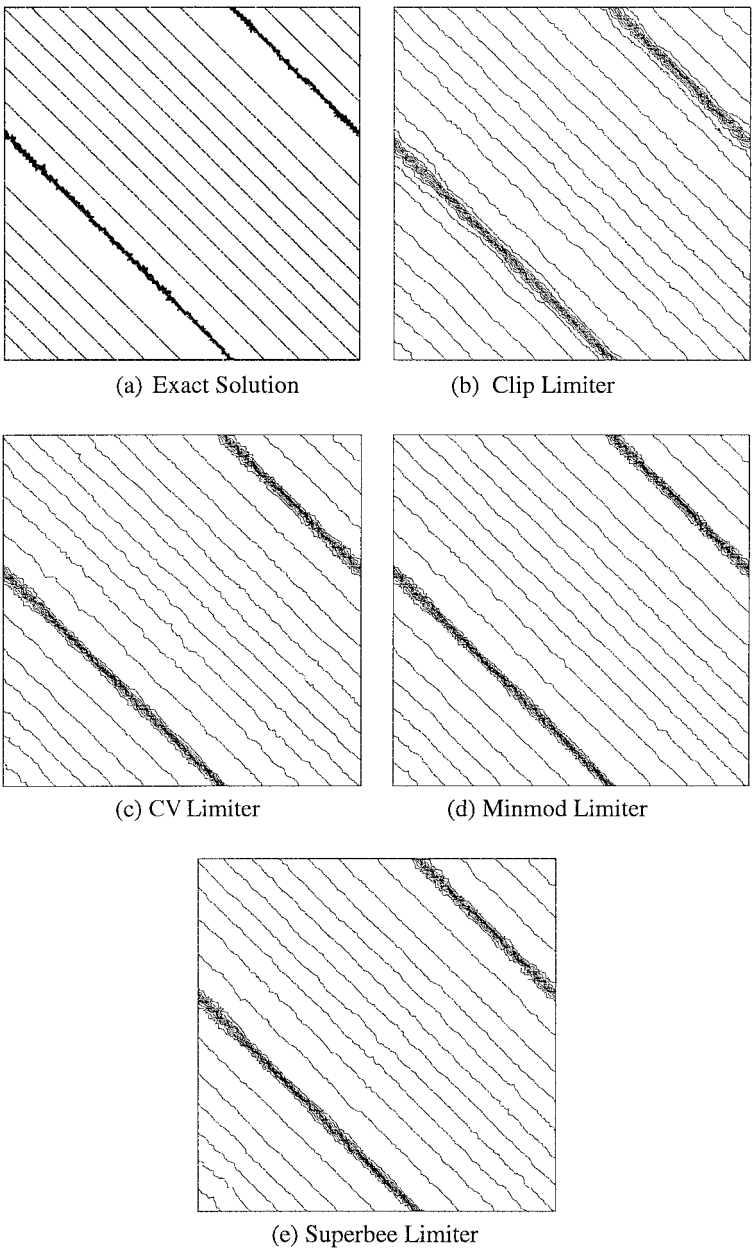


FIG. 12. Exact and computational solutions of Burger's equation at $t = 0.45$ on the $40 \times 40 \times 2$ irregular grid using the Type 2 quadratic SV (third-order accurate), $M = 0$.

At $t = 0.45$, the exact solution has developed two shock waves, as shown in Fig. 12a. Limiters are necessary to handle the discontinuities. All the limiters (Clip, CV, Superbee, and Minmod limiters) are evaluated. Shown in Fig. 12 are the exact solutions, and the computed numerical solutions with the Type 2 quadratic SV (third-order accurate) on the $40 \times 40 \times 2$ irregular grid using all limiters with $M = 0$, i.e., TVD limiters. The use of $M = 0$ was designed to highlight the differences between the limiters. Note that all the TVD limiters gave reasonable solutions, including the Clip limiter. In terms of shock resolution,

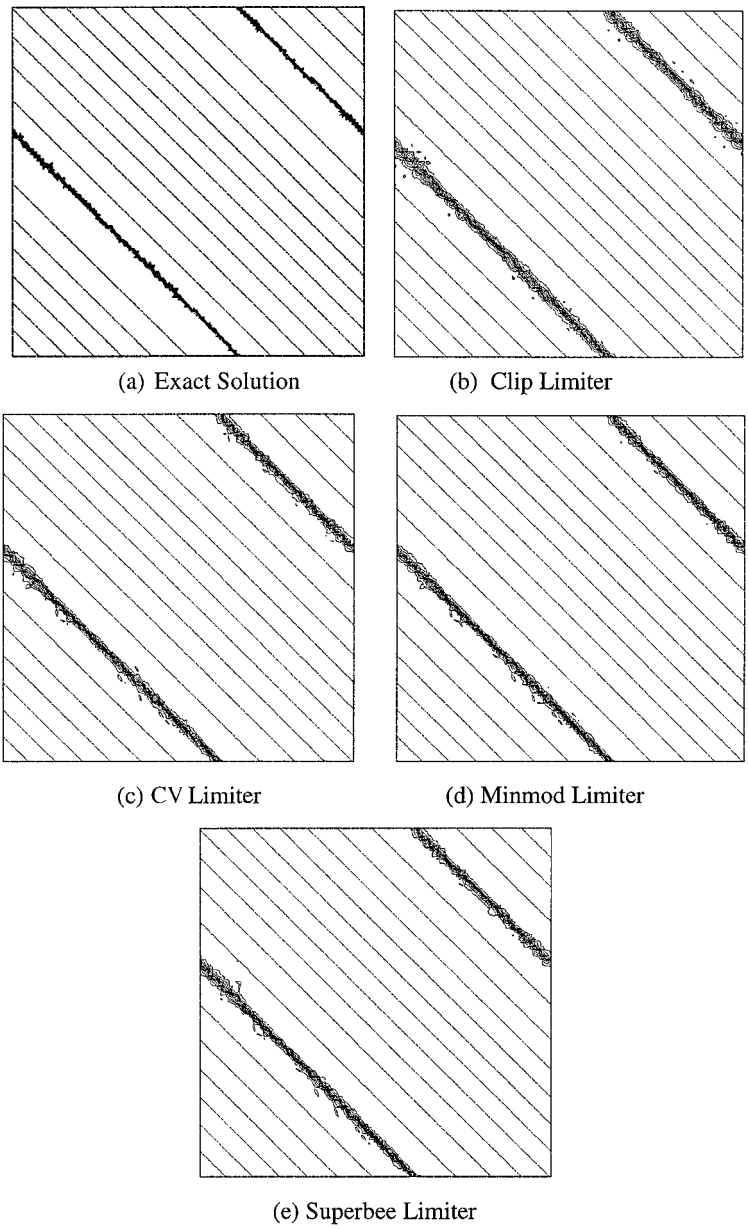


FIG. 13. Exact and computational solutions of Burger's equation at $t = 0.45$ on the $40 \times 40 \times 2$ irregular grid using the Type 2 quadratic SV (third-order accurate), $M = 400$.

the Clip limiter is the most dissipative, followed by the CV limiter. The Minmod and Superbee limiters, which are difficult to distinguish from each other, gave the best solutions. Figure 13 displays the solutions with the same grid and SV scheme with $M = 400$, i.e., TVB limiters. The most striking difference between the results shown in Figs. 12 and 13 is that the solutions away from the shock waves are much smoother with the TVB limiters than with the TVD limiters. This indicates that high-order accuracy is achieved away from the discontinuities with the TVB limiters. However, the price one must pay to achieve this is that some oscillations near the shock waves must be tolerated, as shown in Fig. 13.

TABLE VIII
Accuracy on $u_t + uu_x + uu_y = 0$, with $u_0(x, y) = \frac{1}{4} + \frac{1}{2}\sin\pi(x + y)$, at $t = 0.45$ in $[-0.2, 0.4] \times [-0.2, 0.4]$ on Irregular Grid

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
2 (Type 2 SV)	$10 \times 10 \times 2$	1.68e-4	—	5.33e-3	—
	$20 \times 20 \times 2$	3.92e-5	2.10	1.65e-3	1.69
	$40 \times 40 \times 2$	9.66e-6	2.02	4.83e-4	1.77
	$80 \times 80 \times 2$	2.43e-6	1.99	1.58e-4	1.61
	$160 \times 160 \times 2$	6.01e-7	2.02	3.49e-5	2.18
3 (Type 2 SV)	$10 \times 10 \times 2$	6.23e-5	—	6.57e-3	—
	$20 \times 20 \times 2$	6.25e-6	3.32	5.86e-4	3.49
	$40 \times 40 \times 2$	6.06e-7	3.37	7.21e-5	3.02
	$80 \times 80 \times 2$	7.40e-8	3.03	1.29e-5	2.48
	$160 \times 160 \times 2$	9.47e-9	2.97	2.68e-6	2.27
4 (Type 3 SV)	$10 \times 10 \times 2$	7.81e-5	—	4.39e-2	—
	$20 \times 20 \times 2$	6.78e-7	6.85	1.31e-3	5.07
	$40 \times 40 \times 2$	6.38e-9	6.73	2.97e-6	8.78
	$80 \times 80 \times 2$	3.85e-10	4.05	6.65e-8	5.48
	$160 \times 160 \times 2$	2.84e-11	3.76	4.36e-9	3.93

Note. Minmod limiter with $M = 400$.

In order to estimate the numerical order of accuracy for the solution away from the discontinuities, L_1 and L_∞ errors in the smooth region $[-0.2, 0.4] \times [-0.2, 0.4]$ are computed. Computations were carried out on the irregular grid only with the Minmod limiter. Without the use of the limiter, the solution quickly diverged after shock waves were developed in the solution. The parameter M was set to be 400 in the computation. If M is too small, the accuracy in the smooth region is degraded, probably because limiting was carried out in the smooth region as well as near the shock. The L_1 and L_∞ errors with one type of SV for a given order of accuracy are presented in Table VIII. Obviously, with this choice of M, the designed order of accuracy was achieved away from discontinuities.

7. CONCLUSIONS

The spectral volume method [38] has been successfully extended to two-dimensional scalar conservation laws using unstructured triangular meshes. Each mesh cell forms a spectral volume, and the spectral volume is further partitioned into polygonal control volumes. High-order schemes are then built based on the CV-averaged solutions. It is shown that a universal reconstruction can be obtained if all spectral volumes are partitioned in a similar manner. However, as in the one-dimensional case, the way in which a SV is partitioned into CVs affects the convergence property of the resultant numerical scheme. A criterion based on the Lebesgue constant has been developed and used successfully to determine the quality of various partitions. Symmetric, stable, and convergent linear, quadratic, and cubic SVs have been obtained, and many different types of partitions are evaluated based on the Lebesgue constants and their performance on model test problems.

Accuracy studies with 2D linear and nonlinear scalar conservation laws have been carried out, and the order-of-accuracy claim has been numerically verified on both smooth and nonsmooth triangular grids for convergent SVs. Several TVD and TVB limiters have been

developed for nonoscillatory capturing of discontinuities and found to perform well. The TVB limiters with a properly selected parameter (M) are capable of maintaining uniformly high-order accuracy away from discontinuities. The extension of the method to one- and two-dimensional hyperbolic systems is under way and will be reported in future publications.

ACKNOWLEDGMENT

The first author gratefully acknowledges the start-up funding provided by the Department of Mechanical Engineering, College of Engineering of Michigan State University.

REFERENCES

1. R. Abgrall, On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation, *J. Comput. Phys.* **114**, 45 (1994).
2. H. L. Atkin and C.-W. Shu, Quadrature-free implementation of discontinuous Galerkin method for hyperbolic equations, *AIAA J.* **36**, 775 (1998).
3. D. S. Balsara and C.-W. Shu, Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high-order accuracy, *J. Comput. Phys.* **160**, 405 (2000).
4. T. J. Barth and D. C. Jespersen, *The Design and Application of Upwind Schemes on Unstructured* Meshes, AIAA Paper 89-0366 (1989).
5. T. J. Barth and P. O. Frederickson, *High-Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction*, AIAA Paper 90-0013 (1990).
6. F. Bassi and S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, *J. Comput. Phys.* **138**, 251 (1997).
7. P. Batten, C. Lambert, and D. M. Causon, Positively conservative high-resolution convection schemes for unstructured elements, *Int. J. Numer. Methods Eng.* **39**, 1821 (1996).
8. L. P. Bos, Bounding the Lebesgue functions for Lagrange interpolation in a simplex, *J. Approx. Theory* **39**, 43 (1983).
9. W. Cai, D. Gottlieb, and C.-W. Shu, Essentially nonoscillatory spectral Fourier methods for shock wave calculations, *Math. Comput.* **52**, 389 (1989).
10. J. Casper and H. L. Atkins, A finite volume high-order ENO scheme for two-dimensional hyperbolic systems, *J. Comput. Phys.* **106**, 62 (1993).
11. Q. Chen and I. Babuska, Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle, *Comput. Methods Appl. Mech. Eng* **128**, 405 (1995).
12. B. Cockburn and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II: General framework, *Math. Comput.* **52**, 411 (1989).
13. B. Cockburn, S.-Y. Lin, and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. III: One-dimensional systems, *J. Comput. Phys.* **84**, 90 (1989).
14. B. Cockburn, S. Hou, and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV: The multidimensional case, *Math. Comput.* **54**, 545 (1990).
15. P. Colella and P. Woodward, The piecewise parabolic method for gas-dynamical simulations, *J. Comput. Phys.* **54**, 174 (1984).
16. M. Delanaye and Y. Liu, *Quadratic Reconstruction Finite Volume Schemes on 3D Arbitrary Unstructured Polyhedral Grids*, AIAA Paper 99-3259-CP (1999).
17. O. Friedrich, Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids, *J. Comput. Phys.* **144**, 194 (1998).
18. S. K. Godunov, A finite-difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics, *Mater. Sb.* **47**, 271 (1959).
19. A. Harten, B. Engquist, S. Osher, and S. Chakravarthy, Uniformly high order essentially nonoscillatory schemes, III, *J. Comput. Phys.* **71**, 231 (1987).

20. C. Hu and C.-W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, *J. Comput. Phys.* **150**, 97 (1999).
21. A. Jameson, Analysis and design of numerical schemes for gas dynamics. 2: Artificial diffusion and discrete shock structure, *Int. J. Comput. Fluid Dyn.* **5**, 1 (1995).
22. G. Jiang and C.-W. Shu, Efficient implementation of weighted ENO schemes, *J. Comput. Phys.* **126**, 202 (1996).
23. D. A. Kopriva, Multidomain spectral solutions of the Euler gas-dynamics equations, *J. Comput. Phys.* **96**, 428 (1991).
24. D. A. Kopriva and J. H. Kolias, A conservative staggered-grid Chebyshev multidomain method for compressible flows, *J. Comput. Phys.* **125**, 244 (1996).
25. M.-S. Liou, Mass flux schemes and connection to shock instability, *J. Comput. Phys.* **160**, 623 (2000).
26. X. D. Liu, A maximum principle satisfying modification of triangle based adaptive stencil for the solution of scalar hyperbolic conservation laws, *SIAM J. Numer. Anal.* **30**, 701 (1993).
27. X. D. Liu, S. Osher, and T. Chan, Weighted essentially non-oscillatory schemes, *J. Comput. Phys.* **115**, 200 (1994).
28. Y. Liu and M. Vinokur, Exact integration of polynomials and symmetric quadrature formulas over arbitrary polyhedral grids, *J. Comput. Phys.* **140**, 122 (1998).
29. S. Osher, Riemann solvers, the entropy condition, and difference approximations, *SIAM J. Numer. Anal.* **21**, 217 (1984).
30. A. T. Patera, A spectral element method for fluid dynamics: laminar flow in a channel expansion, *J. Comput. Phys.* **54**, 468 (1984).
31. P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* **43**, 357 (1981).
32. C.-W. Shu, Total-variation-diminishing time discretizations, *SIAM J. Sci. Stat. Comput.* **9**, 1073 (1988).
33. C.-W. Shu, TVB uniformly high-order schemes for conservation laws, *Math. Comput.* **49**, 105 (1987).
34. D. Sidilkover and G. E. Karniadakis, Non-oscillatory spectral element Chebyshev method for shock wave calculations, *J. Comput. Phys.* **107**, 10 (1993).
35. J. L. Steger and R. F. Warming, Flux vector splitting of the inviscid gasdynamics equations with application to finite difference methods, *J. Comput. Phys.* **40**, 263 (1981).
36. B. van Leer, Towards the ultimate conservative difference scheme. V. A second order sequel to Godunov's method, *J. Comput. Phys.* **32**, 101 (1979).
37. B. van Leer, Flux-vector splitting for the Euler equations, *Lecture Notes Phys.* **170**, 507 (1982).
38. Z. J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, *J. Comput. Phys.* **178**, 210 (2002).
39. S. Wolfram, *Mathematica* Book (Wolfram Media and Cambridge Univ. Press, New York, 1999), 4th Ed.