Journal of Computational Physics **178,** 210–251 (2002)
doi:10.1006/jcph.2002.7041, available online at http://www.idealibrary.com on IDEAL®

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

Basic Formulation

Z. J. Wang

Department of Mechanical Engineering, Michigan State University, 2328B Engineering Building, East Lansing, Michigan 48824
E-mail: zjw@egr.msu.edu

Received May 1, 2001; revised February 13, 2002

A high-order, conservative, yet efficient method named the spectral volume (SV) method is developed for conservation laws on unstructured grids. The concept of a "spectral volume" is introduced to achieve high-order accuracy in an efficient manner similar to spectral element and multidomain spectral methods. Each spectral volume is further subdivided into control volumes, and cell-averaged data from these control volumes are used to reconstruct a high-order approximation in the spectral volume. Then Riemann solvers are used to compute the fluxes at spectral volume boundaries. Cell-averaged state variables in the control volumes are updated independently. Furthermore, total variation diminishing and total variation bounded limiters are introduced in the SV method to remove/reduce spurious oscillations near discontinuities. Unlike spectral element and multidomain spectral methods, the SV method can be applied to fully unstructured grids. A very desirable feature of the SV method is that the reconstruction is carried out analytically, and the reconstruction stencil is always nonsingular, in contrast to the memory and CPU-intensive reconstruction in a high-order k-exact finite volume method. Fundamental properties of the SV method are studied and high-order accuracy is demonstrated for several model problems with and without discontinuities.

Key Words: high-order; unstructured grid; finite volume; spectral volume; conservation laws.

1. INTRODUCTION

Numerical algorithms for conservation laws have been extensively researched in the past two-and-a-half decades. One of the most successful algorithms for conservation laws is the Godunov method [17], which laid a solid foundation for the development of modern

0021-9991/02 $35.00

upwind schemes [33] including MUSCL [43–44], total variation diminishing (TVD) [19], piecewise parabolic method [14], and essentially nonoscillatory (ENO) [20] schemes. Van Leer extended the first-order Godunov scheme to second-order in a series of papers [43, 44] with the introduction of a limiter to remove spurious numerical oscillations near steep gradients. Other techniques to reduce or eliminate spurious numerical oscillations were also developed, for example, by Boris *et al*. [6, 7]. Meanwhile, the exact Riemann solver used in the Godunov scheme was sometimes replaced by approximate "Riemann solvers" or flux-splitting procedures [29, 34, 36, 42, 45, 48] for better efficiency.

In 1983, Harten developed the nonoscillatory, high-resolution TVD scheme in [19]. In this paper, the total variation was first used to measure the degree of oscillations, and sufficient conditions for TVD schemes were presented. Following Harten's TVD scheme, a variety of other TVD schemes were developed [10, 47, 50]. These schemes showed excellent shock-capturing ability, with no numerical oscillations. An undesirable property of TVD schemes is that near every extrema, even if it is a smooth extremum, the schemes degrade to first-order accuracy to suppress any spurious numerical oscillations, resulting in the clipping of smooth extrema. To remedy this drawback, Harten *et al*. further developed the ENO method [20]. ENO schemes are high-order accurate and essentially nonoscillatory; i.e., numerical oscillations, if any, decay with the order of the truncation error. The basic idea of ENO is adaptive "stenciling," i.e., to use the "smoothest" possible data stencil in the reconstruction. By doing that, stencils containing a discontinuity are usually avoided. Following the original ENO paper, research on ENO schemes has been very active [9, 21]. Other ideas to extend and improve ENO schemes have been published. For example, weighted ENO (WENO) schemes were developed to further improve the accuracy and smoothness of ENO schemes [3, 24, 30, 39].

Another disadvantage of TVD schemes is that multidimensional TVD schemes are at most first-order accurate [18]. Therefore, the extension of second-order TVD schemes to multidimensions is carried out in a dimension-by-dimension manner; i.e., one-dimensional TVD schemes are used in each coordinate direction. The extension of ENO schemes to multidimensions can be accomplished in two different fashions. One approach is based on cell-averaged state variables, and the other is based on state variables at grid points. It was shown in [25] that the extension using node-based state variables can be accomplished in a dimension-by-dimension fashion and is several times more efficient than using cell-averaged variables. However, the computational grid must be smooth to maintain the formal order of accuracy. For complex geometries, it is usually difficult to satisfy this requirement.

The difficulty in generating smooth structured grids for complex geometries has prompted intensive research and development in unstructured grid algorithms in the past one-and-a-half decades, e.g., [22, 26, 31, 46, 49]. 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 in [4], an ENO scheme for unstructured grids was developed by Abgrall in [1], and WENO schemes for unstructured grids were developed by Friedrich [16] and Hu and Shu [24]. An ENO scheme with a nonpolynomial basis was developed by Sonar [41]. Although arbitrarily high-order accurate finite volume schemes can be obtained theoretically for an arbitrary 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 (nonsingular) stencils, and the enormous memory required to store the coefficients

used in the reconstruction. For each control volume, the reconstruction stencil is unique for an unstructured grid. A data reconstruction must be performed at each iteration for each control volume. This reconstruction step is the most memory and time consuming in higher than second-order schemes. In a recent implementation of a third-order finite volume (FV) scheme with a quadratic reconstruction in three dimensions by Delanaye and Liu [15], the average size of the reconstruction stencils is 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. It seems that only second-order FV schemes are practical in three dimensions.

Recently, another high-order conservative algorithm called the discontinuous Galerkin (DG) method was developed by Cockburn *et al*. in a series of papers [11–13]. 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. The residual is then minimized with a Galerkin approach. Due to the use of Riemann fluxes cross 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. For example, a kth order DG scheme (with an order $k-1$ polynomial reconstruction) requires a $2k$th order quadrature formula for the surface integrals, and a $2k-1$th order quadrature formula for the volume integrals. Another high-order conservative scheme for unstructured quadrilateral grids is the multidomain spectral method on a staggered grid developed by Kopriva and Kolias [27, 28]. The multidomain spectral method is similar to the spectral element method by Patera [35]. However, the spectral element method is not conservative. Although a very high order of accuracy was achievable with both methods, the methods are difficult to extend to other cell types such as triangles or tetrahedral cells.

In this paper, a new conservative high-order numerical method is developed for conservation laws on unstructured grids. In the following section, Section 2, we first present the basic framework of the spectral volume (SV) method for unstructured grids, and discussions are made concerning why the SV method is far more efficient than the k-exact high-order finite volume method. In Section 3, the basic one-dimensional scheme is presented. In Section 4, the second-order SV scheme is analyzed to understand the basic characteristics of the method. In particular, the second-order SV scheme is compared with the second-order DG scheme. In Section 5, convergent reconstructions for high-order SV schemes are discussed, and the choice of control volumes (CVs) is shown to strongly affect the convergence of the method. Section 6 discusses issues related to discontinuity-capturing and TVD and total variation bounded (TVB) limiters. In Section 7, 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 8.

2. THE SPECTRAL VOLUME METHOD

To present the basic idea, we consider the following multidimensional scalar conservation laws

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

on domain Ω with the following 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 denotes time and $t \in [0, T]$, u is a state variable, and f and g are fluxes in x and y directions respectively. Domain Ω is discretized into N nonoverlapping cells, which are called spectral volumes; i.e.,

$$\Omega = \bigcup_{i=1}^{N} S_i. \tag{2.2}$$

The reason the cells are called SVs will be clear later. Integrating (2.1) on a SV S_i, we obtain

$$\int_{S_i} \frac{\partial u}{\partial t}\, dV + \oint_{\partial S_i} (F \cdot \boldsymbol{n})\, dA = 0, \tag{2.3}$$

where $F = (f, g)$, and $\mathbf{n}$ is the unit outward normal of ∂S_i. Define the cell-averaged state variable for S_i as

$$\bar{u}_i = \frac{\int_{S_i} u\, dV}{V_i}, \tag{2.4}$$

where V_i is the volume of S_i. Then (2.3) becomes

$$\frac{d\bar{u}_i}{dt} + \frac{1}{V_i} \sum_{r=1}^{L} \int_{A_r} (F \cdot \boldsymbol{n})\, dA = 0, \tag{2.5}$$

where L is the number of faces in S_i, and A_r represents the rth face. The surface integration on each face can be performed with a kth order accurate Gauss quadrature formula; i.e.,

$$\int_{A_r} (F \cdot \boldsymbol{n})\, dA = \sum_{j=1}^{J} w_{rj} F(u(x_{rj}, y_{rj})) \cdot \boldsymbol{n}_r A_r + O(A_r h^k), \tag{2.6}$$

where w_{rj} are the Gauss quadrature weights, (x_{rj}, y_{rj}) are the Gauss quadrature points, and h is the maximum span of all the SVs in x and y directions. Time t is omitted whenever there is no confusion. If $F =$ constant the following identity exists:

$$\sum_{r=1}^{L} \int_{A_r} (F \cdot \boldsymbol{n})\, dA = 0. \tag{2.7}$$

Therefore, we will gain an extra order if we sum up the surface integrals for all faces of S_i; i.e.,

$$\sum_{r=1}^{L} \int_{A_r} (F \cdot \boldsymbol{n})\, dA = \sum_{r=1}^{L} \sum_{j=1}^{J} w_{rj} F(u(x_{rj}, y_{rj})) \cdot \boldsymbol{n}_r A_r + O(A_r h^{k+1}). \tag{2.8}$$

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

$$\frac{1}{V_i} \sum_{r=1}^{L} \oint_{A_r} (F \cdot \boldsymbol{n})\, dA = \frac{1}{V_i} \sum_{r=1}^{L} \sum_{j=1}^{J} w_{rj} F(u(x_{rj}, y_{rj})) \cdot \boldsymbol{n}_r A_r + O(h^k). \tag{2.9}$$

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

$$p_i(x, y) = u(x, y) + O(h^k), \quad (x, y) \in S_i. \tag{2.10}$$

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 of order $k-1$ 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 then carried out using an exact Riemann solver or one of the Lipschitz continuous approximate Riemann solvers or flux slitting procedures; i.e.,

$$\begin{aligned} F(u(x_{rj}, y_{rj})) \cdot \boldsymbol{n}_r = {} & F_{Riemann}(p_i(x_{rj}, y_{rj}), p_{i,r}(x_{rj}, y_{rj}), \boldsymbol{n}_r) \\ & + O(p_i(x_{rj}, y_{rj}) - p_{i,r}(x_{rj}, y_{rj})). \end{aligned} \tag{2.11}$$

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

$$p_i(x_{rj}, y_{rj}) = u(x_{rj}, y_{rj}) + O(h^k) \tag{2.12a}$$

$$p_{i,r}(x_{rj}, y_{rj}) = u(x_{rj}, y_{rj}) + O(h^k). \tag{2.12b}$$

Therefore

$$F(u(x_{rj}, y_{rj})) \cdot \boldsymbol{n} = F_{Riemann}(p_i(x_{rj}, y_{rj}), p_{i,r}(x_{rj}, y_{rj}), \boldsymbol{n}_r) + O(h^k). \tag{2.13}$$

Substituting (2.13) into (2.6), we obtain

$$\int_{A_r} (F \cdot \boldsymbol{n})\, dA = \sum_{j=1}^{J} w_{rj} F_{Riemann}(p_i(x_{rj}, y_{rj}), p_{i,r}(x_{rj}, y_{rj}), \boldsymbol{n}_r) A_r + O(A_r h^k). \tag{2.14}$$

Summarizing (2.5)–(2.14), we obtain the following semidiscrete, kth order accurate scheme on S_i for the conservation laws (2.1)

$$\frac{d\bar{u}_i}{dt} + \frac{1}{V_i} \sum_{r=1}^{L} \sum_{j=1}^{J} w_{rj} F_{Riemann}(p_i(x_{rj}, y_{rj}), p_{i,r}(x_{rj}, y_{rj}), \boldsymbol{n}_r) A_r = O(h^k). \tag{2.15}$$

What we have done so far follows exactly the finite volume doctrine. We, however, omitted a vital detail, i.e., how we build the high-order reconstruction polynomial given just the cell-averaged state variables for the SVs. Here is where the new method departs from the traditional FV scheme. In a FV method, a stencil (a group of neighboring cells and the cell under consideration) is used to build a high-order polynomial approximation to the state variable on the cell under consideration. Instead of using a large stencil of neighboring cells to perform the reconstruction, the SV is partitioned into subcells called *control volumes*. The order of accuracy of the reconstruction determines the number of CVs to be generated in each SV. For example, for a linear reconstruction on a triangle, the triangle is divided into three CVs as shown in Fig. 1a, and cell-averaged state variables are defined on the CVs. Figures 1b and 1c give some possible CV subdivisions for quadratic and cubic data reconstructions. The number of CVs in Fig. 1 is the minimum required for these polynomial reconstructions.

With any of these high-order reconstructions, Eq. (2.15) can then be used to update the cell-averaged state variable on the SVs: i.e., the cell-averaged state variable $\bar{u}_i$ for S_i at a new time level $n+1$ (i.e., $\bar{u}_i^{n+1}$) can be obtained with an appropriate time integration scheme based on the solution at time level n with $t = n\Delta t$, where Δt is the time step. However, to use the same high-order reconstruction at time level $n+1$, it is necessary to "scatter" the update $\Delta\bar{u}_i \equiv \bar{u}_i^{n+1} - \bar{u}_i^n$ back to the cell-averaged state variables at all the CVs in S_i. However, there is only one obvious relationship between the cell-averaged variable for S_i and the cell-averaged variables for the CVs in S_i; i.e.,

$$\bar{u}_i = \frac{\sum_{j=1}^{K} \bar{u}_{i,j} V_{i,j}}{V_i}. \tag{2.16}$$

Here $\bar{u}_{i,j}$ is the cell-averaged state variable of the jth control volume of S_i, $V_{i,j}$ is the volume of this control volume, and K is the number of CVs in S_i. It is obvious that

$$V_i = \sum_{j=1}^{K} V_{i,j}. \tag{2.17}$$

Therefore, we need some distribution scheme to "scatter" the total cell-averaged state variable from the SV to the CVs. To ensure conservation, the updates of the state variable on the CVs must satisfy

$$\Delta\bar{u}_i = \frac{\sum_{j=1}^{K} \Delta\bar{u}_{i,j} V_{i,j}}{V_i}. \tag{2.18}$$

For a second-order SV scheme based on a linear reconstruction, we can use the fluctuation splitting approach [37] to "scatter" the updates from the SVs to the CVs.

Note that it is difficult to derive a scattering scheme for higher order reconstructions; i.e., for $k > 2$. The FV doctrine is again our saving grace. Each CV inside a SV is treated separately *as if it is independent* to update the cell-averaged state variable for the CV. Note that the subtle difference between a FV and 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

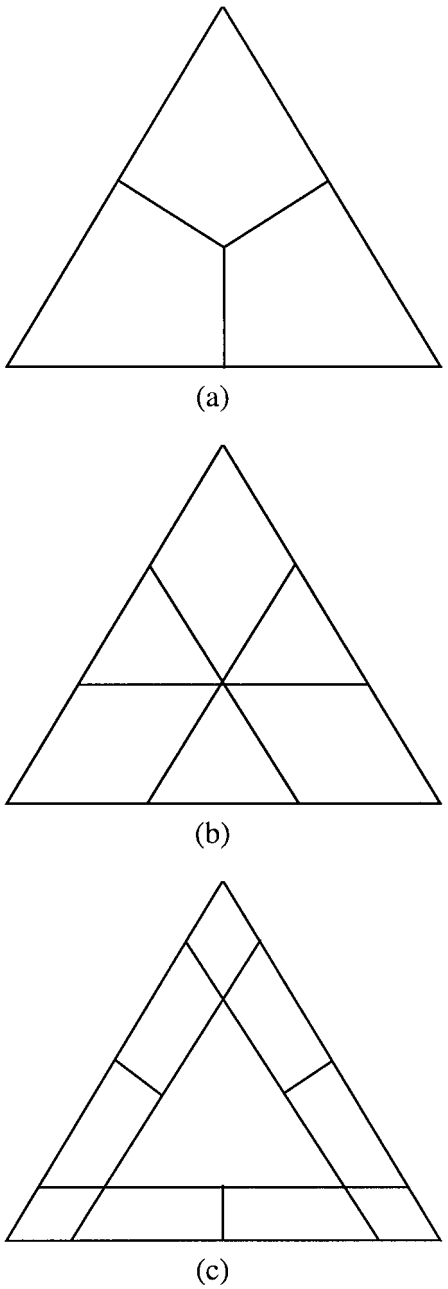

FIG. 1. Control volumes in a triangular spectral volume (a) linear; (b) quadratic; (c) cubic.

variable is continuous across the interior CV boundaries. Riemann fluxes are only necessary at the boundaries of the SV. To maintain a high-order accuracy, Gauss quadrature formulas are again used, not only for the Riemann fluxes through the SV boundaries, but also for the analytical fluxes through interior CV boundaries. The most significant advantage of the SV method, as compared to 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 Fig. 1) is exactly the

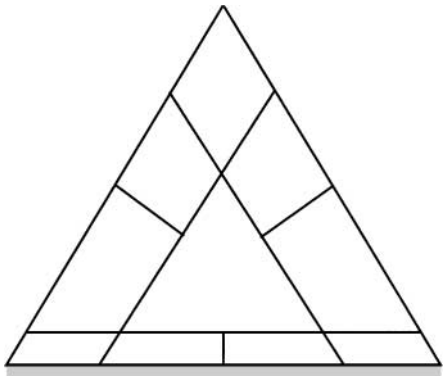

FIG. 2. Control volumes in a cubic triangular spectral volume to resolve a viscous boundary layer.

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 in again significant savings because the Riemann flux is usually several times more expensive to compute than the exact flux.

Another advantage of the SV method is that grid clustering necessary to resolve a particular flow feature (e.g., a viscous boundary layer) can be easily accommodated when CVs are generated for a SV. For example, to resolve a viscous boundary layer near a solid wall, control volumes can be easily clustered near the wall, as shown in Fig. 2. *It is not necessary to regenerate another grid with the appropriate grid clustering for a particular Reynolds number*. Therefore the same grid can be used for different physical flow problems. A user only needs to change one input parameter to obtain the desired grid clustering, significantly reducing the time and cost in grid generation.

The idea can of course be extended to other cell types such as quadrilaterals, tetrahedra, hexahedra, prisms, etc. 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 find.

The SV method shares many advantages with the DG method [11–13] in that it is compact, which is suitable for parallel computing, high-order accurate, conservative, and capable of handling complex geometries. We believe the SV method is more efficient than the DG method and has higher resolution than the DG method for discontinuities because of the availability of local cell-averaged state variables at the control volumes. Although a quadrature-free implementation of the DG method is possible [2], the implementation is more efficient than the usual DG method with quadratures if the expansion for the flux function can be accomplished with only a few extra terms than the expansion for the state variables. For complex nonlinear flux involving discontinuities, the expansion must be expanded to significantly more terms, e.g., $k(2k+1)$ versus $k(k+1)/2$ terms (k being the order of accuracy), to achieve the formal order of accuracy. In this case, the quadrature-free implementation may be slower than the usual DG method. Therefore, we exclude the quadrature-free implementation from our comparison, and we only compare the main differences between the SV method and the usual DG method with quadrature formulas.

The main steps in a kth order SV method (with an order $k-1$ polynomial reconstruction) are:

1. Compute the state variables at the quadrature points;
2. Use a kth order accurate quadrature formula (exact for a polynomial of order $k-1$) and a Riemann solver to compute the surface flux integrals at the spectral volume boundaries, and use analytical fluxes for interior control volume boundaries because the state variable is continuous across the interior CV boundaries;
3. Use a TVD Runge–Kutta scheme for time integration.

The main steps in a kth order DG method are:

1. Compute the state variables at the quadrature points;
2. Use a Riemann solver and a $2k$th order quadrature formula to compute the surface flux integrals;
3. Use a $(2k-1)$th order quadrature formula to compute the volume integrals;
4. Left multiply the residual by the inverse of the mass matrix if the mass matrix is not diagonal;
5. Use a TVD Runge–Kutta scheme for time integration.

The SV method requires more flux computations, while the DG method needs a volume integral, and a higher order surface integral quadrature formula may require an extra matrix–vector multiplication if the mass matrix is not diagonal. Note that with a special choice of basis functions, it is possible to achieve a diagonal mass matrix in the DG method, e.g., [8]. However, the basis functions are not symmetric with respect to the symmetries of a triangle because of a mapping from a triangle to a square. A detailed comparison of the SV and DG methods is now under way and will be reported in a future publication.

3. THE BASIC ONE-DIMENSIONAL SV METHOD

To analyze the characteristics of the SV method, we will concentrate on the one-dimensional scheme from here on. The extension to multiple dimensions will be carried out elsewhere. Therefore, we consider the following one-dimensional conservation laws

$$\frac{\partial u(x,t)}{\partial t} + \frac{\partial f(u(x,t))}{\partial x} = 0, \qquad \text{in } [a,b]x[0,T] \tag{3.1a}$$

$$u(x,0) = u_0(x), \quad \forall x \in [a,b] \tag{3.1b}$$

and periodic boundary conditions. Other types of boundary conditions will be dealt with later. Given a partition of the domain $[a,b]$, $\{x_{i+1/2}\}_{i=0}^{N}$, the domain is then divided into N nonoverlapping spectral volumes

$$[a,b] = \bigcup_{i=1}^{N} S_i, \quad S_i = \left[x_{i-1/2}, x_{i+1/2}\right]. \tag{3.2}$$

Let $h_i = x_{i+1/2} - x_{i-1/2}$, and denote the quantity $\max_{1\le i\le N} h_i$ by h. Given a desired numerical order of accuracy k for (3.1), each spectral volume S_i is subdivided into k control volumes using the following partitioning $\{x_{i,j+1/2}\}_{j=0}^{k}$ with $x_{i,1/2} = x_{i-1/2}$ and $x_{i,k+1/2} = x_{i+1/2}$. The jth CV of S_i is then

$$C_{i,j} = \left(x_{i,j-1/2}, x_{i,j+1/2}\right), \quad j = 1, \ldots, k. \tag{3.3}$$

Denote $h_{i,j} = x_{i,j+1/2} - x_{i,j-1/2}$. The cell-averaged state variable at time t for control volume $C_{i,j}$ is defined as

$$\bar{u}_{i,j}(t) = \frac{\int_{x_{i,j-1/2}}^{x_{i,j+1/2}} u(x,t)\,dx}{h_{i,j}}, \quad j = 1,\ldots,k, \quad i = 1,\ldots,N. \tag{3.4}$$

Here is the reconstruction problem in the SV method. Given the cell-averaged state variables for all the CVs in S_i, construct a polynomial $p_i(x)$, of degree at most $k-1$, for S_i, such that it is a kth order accurate approximation to the function $u(x)$ inside S_i:

$$p_i(x) = u(x) + O(h^k), \quad x \in S_i, \quad i = 1,\ldots,N. \tag{3.5}$$

In particular, this gives approximations to the function $u(x)$ at the CV boundaries

$$u_{i,j+1/2} \equiv p_i\left(x_{i,j+1/2}\right) = u\left(x_{i,j+1/2}\right) + O(h^k), \quad j = 0,\ldots,k, \quad i = 1,\ldots,N. \tag{3.6}$$

This reconstruction can be solved in many ways. Here we use a method presented in [39]. There is a unique polynomial of degree of at most $k-1$, whose cell average in each of the CVs in S_i agrees with that of $u(x)$,

$$\frac{\int_{x_{i,j-1/2}}^{x_{i,j+1/2}} p_i(x)\,dx}{h_{i,j}} = \bar{u}_{i,j}; \quad j = 1,\ldots,k. \tag{3.7}$$

This polynomial $p_i(x)$ is the kth order approximation we are looking for as long as the function $u(x)$ is smooth in the region covered by S_i. Ultimately, we need to compute the approximate solutions of $u(x)$ at the CV boundaries to update the state variables at the next time level. Since the mappings from the given cell averages $\bar{u}_{i,j}$ to the CV boundary values are linear, there exist coefficients c_{jl} which depend on the order of accuracy k, and on the mesh size $h_{i,j}$ in S_i, but not on the function u itself, such that

$$u_{i,j+1/2} = \sum_{l=1}^{k} c_{jl}\bar{u}_{i,l}, \quad j = 0,\ldots,k. \tag{3.8}$$

We summarize this as follows: given the k cell averages for the CVs in S_i

$$\bar{u}_{i,1},\ldots,\bar{u}_{i,k}.$$

There exist constants c_{jl} such that the reconstructed values computed according to (3.8) at the CV boundaries $x_{i,j+1/2}$ is kth order accurate:

$$u_{i,j+1/2} = u\left(x_{i,j+1/2}\right) + O(h^k), \quad j = 0,\ldots,k. \tag{3.9}$$

To obtain the coefficients $\{c_{jl}\}$, and to understand the accuracy property, we look at the primitive function of $u(x)$,

$$U(x) \equiv \int_{x_{i,1/2}}^{x} u(\xi)\,d\xi, \quad x \in S_i. \tag{3.10}$$

It is obvious that $U(x_{i,j+1/2})$ can be expressed exactly by the cell averages of $u(x)$ at the control volume boundaries of S_i,

$$U\left(x_{i,j+1/2}\right) = \sum_{l=1}^{j} \int_{x_{i,l-1/2}}^{x_{i,l+1/2}} u(\xi)\, d\xi = \sum_{l=1}^{j} \bar{u}_{i,l} h_{i,l}, \quad \text{for } j = 1, \ldots, k. \tag{3.11a}$$

Note that by definition, we have

$$U\left(x_{i,1/2}\right) = 0. \tag{3.11b}$$

Thus with the cell-averaged variables for the CVs, we know the primitive function $U(x)$ at the CV boundaries exactly. If we denote the unique polynomial of degree at most k, which interpolates $U(x_{i,j+1/2})$ at the following $k+1$ points

$$x_{i,1/2}, \ldots, x_{i,k+1/2}$$

by $P_i(x)$ and denotes its derivative by $p_i(x)$, i.e.,

$$p_i(x) \equiv P_i'(x),$$

then it is easy to verify

$$\begin{aligned}
\frac{1}{h_{i,j}} \int_{x_{i,j-1/2}}^{x_{i,j+1/2}} p_i(x)\, dx &= \frac{1}{h_{i,j}} \int_{x_{i,j-1/2}}^{x_{i,j+1/2}} P_i'(x)\, dx = \frac{1}{h_{i,j}} \left[P\left(x_{i,j+1/2}\right) - P\left(x_{i,j-1/2}\right)\right] \\
&= \frac{1}{h_{i,j}} \left[U\left(x_{i,j+1/2}\right) - U\left(x_{i,j-1/2}\right)\right] \\
&= \frac{1}{h_{i,j}} \left(\int_{x_{i,1/2}}^{x_{i,j+1/2}} u(x)\, dx - \int_{x_{i,1/2}}^{x_{i,j-1/2}} u(x)\, dx \right) \\
&= \frac{1}{h_{i,j}} \int_{x_{i,j-1/2}}^{x_{i,j+1/2}} u(x)\, dx = \bar{u}_{i,j}, \quad j = 1, \ldots, k.
\end{aligned} \tag{3.12}$$

This implies that $p_i(x)$ is what we are looking for. Standard approximation theory tells us that

$$P_i'(x) = U'(x) + O(h^k), \quad x \in S_i; \tag{3.13a}$$

i.e.,

$$p_i(x) = u(x) + O(h^k), \quad x \in S_i. \tag{3.13b}$$

This is actually the accuracy requirement. Next we look at how to obtain the constants $\{c_{jl}\}$. For this we use the Lagrange interpolation polynomial

$$P_i(x) = \sum_{r=0}^{k} U_{i,r+1/2} \varphi_{i,r+1/2}(x) \quad x \in S_i, \tag{3.14a}$$

where $\varphi_{i,r+1/2}(x)$ are the Lagrange interpolation coefficients, $U_{i,r+1/2} = U(x_{i,r+1/2})$, and

$$\varphi_{i,r+1/2}(x) = \prod_{l=0,l\neq r}^{k} \frac{x - x_{i,l+1/2}}{x_{i,r+1/2} - x_{i,l+1/2}}. \tag{3.14b}$$

By introducing

$$\varpi(x) = \left(x - x_{i,1/2}\right)\left(x - x_{i,3/2}\right)\cdots\left(x - x_{i,k+1/2}\right), \tag{3.15}$$

we have

$$\varphi_{i,r+1/2} = \frac{\varpi(x)}{\left(x - x_{i,r+1/2}\right)\varpi'\left(x_{i,r+1/2}\right)}. \tag{3.16}$$

In our reconstruction based on cell-averaged variables, it is the following order $k-1$ polynomial, which we are looking for

$$p_i(x) = P_i'(x) = \sum_{r=0}^{k} U_{i,r+1/2}\varphi'_{i,r+1/2}(x) = \sum_{r=1}^{k} U_{i,r+1/2}\varphi'_{i,r+1/2}(x), \tag{3.17}$$

where

$$\begin{aligned}\varphi'_{i,r+1/2}(x) &= \frac{\varpi'(x)}{\left(x - x_{i,r+1/2}\right)\varpi'\left(x_{i,r+1/2}\right)} - \frac{\varpi(x)}{\left(x - x_{i,r+1/2}\right)^2\varpi'\left(x_{i,r+1/2}\right)} \\ &= \frac{1}{\varpi'\left(x_{i,r+1/2}\right)} \sum_{l=0,l\neq r}^{k} \frac{\varpi(x)}{\left(x - x_{i,r+1/2}\right)\left(x - x_{i,l+1/2}\right)}.\end{aligned} \tag{3.18}$$

Substitute (3.11) into (3.17), we obtain the following $(k-1)$th order interpolation polynomial based on the cell-averaged variables

$$p_i(x) = \sum_{r=1}^{k}\sum_{l=1}^{r} \bar{u}_{i,l}h_{i,l}\varphi'_{i,r+1/2}(x) = \sum_{l=1}^{k}\left(\sum_{r=l}^{k}\varphi'_{i,r+1/2}(x)h_{i,l}\right)\bar{u}_{i,l}. \tag{3.19}$$

Let

$$\begin{aligned}\phi_l(x) &= h_{i,l}\sum_{r=l}^{k}\varphi'_{i,r+1/2}(x) = h_{i,l}\sum_{r=l}^{k}\frac{1}{\varpi'\left(x_{i,r+1/2}\right)}\sum_{m=0,m\neq r}^{k}\frac{\varpi(x)}{\left(x - x_{i,r+1/2}\right)\left(x - x_{i,m+1/2}\right)} \\ &= h_{i,l}\sum_{r=l}^{k}\frac{1}{\varpi'\left(x_{i,r+1/2}\right)}\sum_{\substack{m=0\\m\neq r}}^{k}\prod_{\substack{q=0\\q\neq r,m}}^{k}\left(x - x_{i,q+1/2}\right).\end{aligned} \tag{3.20}$$

Then

$$p_i(x) = \sum_{l=1}^{k}\phi_l(x)\bar{u}_{i,l}. \tag{3.21}$$

Therefore,

$$c_{jl} = \phi_l\left(x_{i,j+1/2}\right) = h_{i,l} \sum_{r=l}^{k} \frac{1}{\varpi'\left(x_{i,r+1/2}\right)} \sum_{\substack{m=0 \\ m\neq r}}^{k} \prod_{\substack{q=0 \\ q\neq r,m}}^{k} \left(x_{i,j+1/2} - x_{i,q+1/2}\right). \tag{3.22}$$

This equation can be used to compute the interpolation constants for any point distribution (CV subdivision in a SV). If the CVs are of equal size, then

$$c_{jl} = \sum_{r=l}^{k} \frac{1}{\prod_{q=0,q\neq r}^{k}(r-q)} \sum_{\substack{m=0 \\ m\neq r}}^{k} \prod_{\substack{q=0 \\ q\neq r,m}}^{k} (j-q). \tag{3.23}$$

With the reconstructed state variables at the CV boundaries for all SVs, we can then update each control volume as if it is independent; i.e.,

$$\frac{d\bar{u}_{i,j}}{dt} h_{i,j} + \left(f_{i,j+1/2} - f_{i,j-1/2}\right) = 0, \tag{3.24}$$

where $f_{i,j+1/2}$ is a numerical flux at $x_{i,j+1/2}$. At the interior CV boundaries in a SV, the constructed state variable is continuous. Therefore, the analytical flux function can be used directly to compute the numerical flux; i.e.,

$$f_{i,j+1/2} = f\left(u_{i,j+1/2}\right), \quad j = 1, \ldots, k-1. \tag{3.25}$$

At the two SV boundaries $x_{i-1/2}$ and $x_{i+1/2}$ (or $x_{i,1/2}$ and $x_{i,k+1/2}$), there are two constructed values of the state variable. For example, at $x_{i-1/2}$, one constructed value is due to the reconstruction in S_i, i.e., $u_{i,1/2}$, and the other is due to the reconstruction in S_{i-1}, i.e., $u_{i-1,k+1/2}$. Therefore, the numerical flux at $x_{i-1/2}$ is computed with a Riemann solver or a flux-splitting technique;

$$f_{i,1/2} = f_{Riemann}\left(u_{i-1,k+1/2}, u_{i,1/2}\right). \tag{3.26}$$

Similarly

$$f_{i,k+1/2} = f_{Riemann}\left(u_{i,k+1/2}, u_{i+1,1/2}\right). \tag{3.27}$$

It is obvious that

$$f_{i,1/2} = f_{i-1,k+1/2} = f_{Riemann}\left(u_{i-1,k+1/2}, u_{i,1/2}\right) \tag{3.28}$$

and

$$f_{i,k+1/2} = f_{i+1,1/2} = f_{Riemann}\left(u_{i,k+1/2}, u_{i+1,1/2}\right). \tag{3.29}$$

Therefore the scheme is fully conservative. The Riemann flux should satisfy the following conditions to ensure that the first-order version is monotonic:

1. Locally Lipschitz and consistent with the flux function $f(u)$; i.e.,

$$f_{Riemann}(u, u) = f(u). \tag{3.30}$$

2. A nondecreasing function of its first argument, and
3. A nonincreasing function of its second argument.

The two examples of numerical fluxes satisfying the above properties are the following:

1. The Lax–Friedrichs flux:

$$\begin{aligned} f_{LF}(x, y) &= \frac{1}{2}[f(x) + f(y) - \alpha(y - x)] \\ \alpha &= \max_u |f'(u)|. \end{aligned} \tag{3.31}$$

2. The Roe flux with "entropy fix":

$$f_R(x, y) = \begin{cases} f(x), & \text{if } f'(u) \geq 0 \text{ for } u \in [\min(x, y), \max(x, y)] \\ f(y), & \text{if } f'(u) \leq 0 \text{ for } u \in [\min(x, y), \max(x, y)] \\ f_{LF}(x, y), & \text{otherwise.} \end{cases} \tag{3.32}$$

Generally speaking, the Roe flux provides the smaller amount of artificial viscosity than the Lax–Friedrichs flux.

Based on the accuracy argument in the last section, the scheme is kth order accurate in space. For time integration of (3.24), we employ the TVD Runge–Kutta time discretization introduced in [38]. Given a time step Δt, then $t = n\Delta t$, where n is the time level integer index, and denote $\bar{u}_{ij}^n = \bar{u}_{i,j}(n\Delta t)$. The semidiscrete numerical scheme (3.24) for all the CVs can be written as

$$\frac{d\bar{u}}{dt} = L(\bar{u}), \tag{3.33a}$$

where

$$\bar{u} = \begin{bmatrix} \bar{u}_{1,1} \\ \cdots \\ \bar{u}_{i,j} \\ \cdots \\ \bar{u}_{N,k} \end{bmatrix}, \quad L(\bar{u}) = \begin{bmatrix} L_{1,1}(\bar{u}) \\ \cdots \\ L_{i,j}(\bar{u}) \\ \cdots \\ L_{N,k}(\bar{u}) \end{bmatrix} \tag{3.33b}$$

$$L_{i,j} = -\tfrac{1}{h_{i,j}}\big(f_{i,j+1/2}(\bar{u}) - f_{i,j-1/2}(\bar{u})\big). \tag{3.33c}$$

Let $\bar{u}^{(0)} = \bar{u}^n$, and define $\bar{u}^{n+1} = \bar{u}^{(k)}$. The second-order scheme is then

$$\begin{aligned} \bar{u}^{(1)} &= \bar{u}^{(0)} + \Delta t L\big(\bar{u}^{(0)}\big), \\ \bar{u}^{(2)} &= \frac{1}{2}\big[\bar{u}^{(0)} + \bar{u}^{(1)} + \Delta t L\big(\bar{u}^{(0)}\big)\big] \end{aligned} \tag{3.34}$$

The third-order scheme is

$$\begin{aligned} \bar{u}^{(1)} &= \bar{u}^{(0)} + \Delta t L\big(\bar{u}^{(0)}\big), \\ \bar{u}^{(2)} &= \frac{3}{4}\bar{u}^{(0)} + \frac{1}{4}\bar{u}^{(1)} + \frac{1}{4}\Delta t L\big(\bar{u}^{(1)}\big), \\ \bar{u}^{(3)} &= \frac{1}{3}\bar{u}^{(0)} + \frac{2}{3}\bar{u}^{(2)} + \frac{2}{3}\Delta t L\big(\bar{u}^{(2)}\big). \end{aligned} \tag{3.35}$$

4. ANALYSIS OF THE SECOND-ORDER SV SCHEME

In this section, we analyze the second-order SV scheme on a uniform grid for the scalar model wave equation; i.e., $f = cu$, with $c > 0$ in (3.1). Therefore, we assume that the spectral volumes have the same size, and in addition, the control volumes also have the same size:

$$h_i = (b-a)/N, \qquad i = 1, \ldots, N$$
$$\Delta \equiv h_{i,j} = h_i/k = \frac{(b-a)}{kN}, \quad i = 1, \ldots, N \quad j = 1, \ldots, k.$$

The semidiscrete second-order SV schemes for the two control volumes in S_i can be easily shown to be

$$\frac{d\bar{u}_{i,1}}{dt} = -\frac{c}{\Delta}\left(\frac{\bar{u}_{i,1} + \bar{u}_{i,2} - 3\bar{u}_{i-1,2} + \bar{u}_{i-1,1}}{2}\right) \tag{4.1a}$$

$$\frac{d\bar{u}_{i,2}}{dt} = -\frac{c}{\Delta}(\bar{u}_{i,2} - \bar{u}_{i,1}). \tag{4.1b}$$

Note that the cell-averaged value on a CV can be approximated by the value at the cell center with a second-order accuracy; i.e.,

$$\bar{u}_{i,j} = u_{i,j} + O(\Delta^2), \quad j = 1, 2, \tag{4.2}$$

where $u_{i,j}$ is the approximate numerical solution at the CV center. Therefore (4.1) can be expressed as node-based finite-difference schemes without loss of accuracy,

$$\frac{du_{i,1}}{dt} = -\frac{c}{\Delta}\left(\frac{u_{i,1} + u_{i,2} - 3u_{i-1,2} + u_{i-1,1}}{2}\right) \tag{4.3a}$$

$$\frac{du_{i,2}}{dt} = -\frac{c}{\Delta}(u_{i,2} - u_{i,1}). \tag{4.3b}$$

It is interesting that while the overall SV scheme is second-order accuracy in space, neither of the two schemes (4.3a) and (4.3b) is second-order accurate if used alone because it is obvious that (4.3b) is the well-known *first-order* upwind scheme! Therefore, it seems that some sort of error canceling is happening here. We can easily derive the modified equations for (4.3a) and (4.3b) to be

$$\frac{\partial u_{i,1}}{\partial t} + c\frac{\partial u_{i,1}}{\partial x} = -c\frac{\Delta}{2}\frac{\partial^2 u_{i,1}}{\partial x^2} + O(\Delta^2) \tag{4.4a}$$

$$\frac{\partial u_{i,2}}{\partial t} + c\frac{\partial u_{i,2}}{\partial x} = +c\frac{\Delta}{2}\frac{\partial^2 u_{i,2}}{\partial x^2} + O(\Delta^2). \tag{4.4b}$$

Note that scheme (4.3a) has a first-order negative artificial dissipation term if used throughout the computational domain, while (4.3b) has a positive first-order dissipation term. Without the finite-volume argument on the order of accuracy presented earlier, it is difficult to believe the combined use of (4.3a) and (4.3b) would yield a second-order accurate scheme. Adding (4.3a) and (4.3b) together, we obtain

$$\frac{d(u_{i,1} + u_{i,2})}{dt} = -\frac{c}{\Delta}\left(\frac{-u_{i,1} + 3u_{i,2} - 3u_{i-1,2} + u_{i-1,1}}{2}\right). \tag{4.5}$$

Expanding (4.5) at the center of S_i, we obtain

$$\frac{\partial u_i}{\partial t} + c\frac{\partial u_i}{\partial x} = O(\Delta^2), \tag{4.6}$$

where u_i represents the state variable at the center of S_i. That is to say that the cell-averaged state variable in the *spectral volume* S_i satisfies the second-order accuracy requirement. It is clear that the semidiscrete second-order SV scheme is consistent. However, the stability of the SV scheme is not obvious. To analyze the stability condition, we first discretize the time derivative with the second-order TVD Runge–Kutta scheme. Given the solutions at time level n, compute

$$u_{i,1}^* = u_{i,1}^n - \sigma\left(\frac{u_{i,1}^n + u_{i,2}^n - 3u_{i-1,2}^n + u_{i-1,1}^n}{2}\right), \tag{4.7a}$$

$$u_{i,2}^* = u_{i,2}^n - \sigma\left(u_{i,2}^n - u_{i,2}^n\right) \quad \text{for } i = 1, \ldots, N, \tag{4.7b}$$

where $\sigma = \frac{c\Delta t}{\Delta}$. Then

$$u_{i,1}^{n+1} = \frac{1}{2}u_{i,1}^n + \frac{1}{2}\left[u_{i,1}^* - \sigma\left(\frac{u_{i,1}^* + u_{i,2}^* - 3u_{i-1,2}^* + u_{i-1,1}^*}{2}\right)\right] \tag{4.8a}$$

$$u_{i,2}^{n+1} = \frac{1}{2}u_{i,2}^n + \frac{1}{2}[u_{i,2}^* - \sigma(u_{i,2}^* - u_{i,1}^*)]. \tag{4.8b}$$

Let

$$U_i^n = \begin{bmatrix} u_{i,1}^n \\ u_{i,2}^n \end{bmatrix}. \tag{4.9}$$

Then the two-step scheme can be written as

$$U_i^{n+1} = \begin{bmatrix} 1 - \sigma/2 - \sigma^2/8 & -\sigma/2 + 3\sigma^2/8 \\ \sigma - 3\sigma^2/4 & 1 - \sigma + \sigma^2/4 \end{bmatrix} U_i^n + \begin{bmatrix} -\sigma/2 + \sigma^2 & 3\sigma/2 - \sigma^2 \\ -\sigma^2/4 & 3\sigma^2/4 \end{bmatrix} U_{i-1}^n + \begin{bmatrix} \sigma^2/8 & -3\sigma^2/8 \\ 0 & 0 \end{bmatrix} U_{i-2}^n. \tag{4.10}$$

Using a Fourier analysis, we can prove that the scheme is stable when

$$\sigma = \frac{c\Delta t}{\Delta} \le 1. \tag{4.11}$$

This stability condition has also been numerically verified. This condition compares favorably with the stability condition of the second-order DG scheme, which is

$$\sigma' = \frac{c\Delta t}{2\Delta} \le 1/3; \quad \text{i.e., } \sigma \le 2/3. \tag{4.12}$$

It is quite interesting to see how the second-order SV scheme differs from the second-order DG scheme. We therefore derive the DG method using the same notations of the SV scheme.

The element in the DG method is chosen to be the spectral volume in the SV method. The following linear distribution is used in element i:

$$u_i(x) = \frac{1}{2}(\bar{u}_{i,1} + \bar{u}_{i,2}) + \frac{(\bar{u}_{i,2} - \bar{u}_{i,1})}{\Delta}(x - x_i). \tag{4.13}$$

Denote

$$\begin{aligned} \bar{u}_i &= \frac{1}{2}(\bar{u}_{i,1} + \bar{u}_{i,2}) \\ \tilde{u}_i &= \bar{u}_{i,2} - \bar{u}_{i,1}. \end{aligned} \tag{4.14}$$

Then the linear distribution can be viewed as an expansion on the basis set $\{\Phi_1, \Phi_2\}$ with expansion coefficients $\{\bar{u}_i, \tilde{u}_i\}$, where $\Phi_1 = 1$, $\Phi_2 = (x - x_i)/\Delta$. Integrating

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \left(\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x}\right)\Phi_j \, dx = 0, \quad j = 1, 2 \tag{4.15}$$

and using a Riemann solver to obtain the fluxes at element boundaries, we obtain

$$\frac{d\bar{u}_i}{dt} = -\frac{c}{2\Delta}(\bar{u}_i + \tilde{u}_i - \bar{u}_{i-1} - \tilde{u}_{i-1}) \tag{4.16a}$$

$$\frac{d\tilde{u}_i}{dt} = -\frac{3c}{2\Delta}(-\bar{u}_i + \tilde{u}_i + \bar{u}_{i-1} + \tilde{u}_{i-1}). \tag{4.16b}$$

Substituting (4.14) into (4.16), we obtain the following schemes

$$\frac{d\bar{u}_{i,1}}{dt} = -\frac{c}{8\Delta}(3\bar{u}_{i,2} + 7\bar{u}_{i,1} - 15\bar{u}_{i-1,2} + 5\bar{u}_{i-1,1}) \tag{4.17a}$$

$$\frac{d\bar{u}_{i,2}}{dt} = -\frac{c}{8\Delta}(9\bar{u}_{i,2} - 11\bar{u}_{i,1} + 3\bar{u}_{i-1,2} - \bar{u}_{i-1,1}). \tag{4.17b}$$

Again, we can drop the overbars from (4.17a) and (4.17b) so that the cell-averaged variables are replaced with the variables at the CV centers. Then the modified equations for (4.17a) and (4.17b) are

$$\frac{\partial u_{i,1}}{\partial t} + c\frac{\partial u_{i,1}}{\partial x} = -c\frac{\Delta}{2}\frac{\partial^2 u_{i,1}}{\partial x^2} + O(\Delta^2) \tag{4.18a}$$

$$\frac{\partial u_{i,2}}{\partial t} + c\frac{\partial u_{i,2}}{\partial x} = +c\frac{\Delta}{2}\frac{\partial^2 u_{i,2}}{\partial x^2} + O(\Delta^2). \tag{4.18b}$$

Comparing (4.18) and (4.4), we can see that the second-order DG scheme has the same modified equations as the second-order SV scheme although they are quite different in form! Adding (4.17a) and (4.17b) together and dropping the overbars, we obtain

$$\frac{d(u_{i,1} + u_{i,2})}{dt} = -\frac{c}{\Delta}\left(\frac{-u_{i,1} + 3u_{i,2} - 3u_{i-1,2} + u_{i-1,1}}{2}\right), \tag{4.19}$$

which is identical to (4.5). This also shows that the above DG scheme is indeed second-order in space. Note that although the second-order SV and DG schemes have the same

accuracy, the second-order SV scheme has a less restrictive stability limit. In fact, numerical tests suggest that the third-order SV method with a third-order TVD Runge–Kutta time-integration scheme is stable when $\sigma' = \frac{c\Delta t}{3\Delta} \leq 1/3$, comparing with a stability limit of $1/5$ for the third-order DG scheme.

5. HIGH-ORDER RECONSTRUCTIONS IN A SPECTRAL VOLUME

It is well known in CFD that the solution error for a given problem is strongly dependent on the smoothness of the computational grid. Uniform or smoothly varying grids always yield smaller solution errors than nonuniform or nonsmooth grids. Many second-order structured grid CFD algorithms often degenerate into first-order on nonsmooth grids. A carefully implemented unstructured CFD solver with a linearity-preserving reconstruction scheme can achieve the formal second-order accuracy on arbitrary grids [49]. However, the magnitude of the solution error is still strongly dependent on the smoothness of the computational grid. In generating CVs inside a SV, one would naturally produce uniform CVs or grids, extrapolating experiences gained in the FV method. As a matter of fact, using uniform CVs for a high-order ($k > 3$) SV scheme is disastrous because the reconstruction is highly oscillatory near the two SV boundary grid points. To understand why, let us look at the difference between the FV and SV reconstructions. In a high-order finite-volume method, a large stencil consisting of many neighboring cells is used in the reconstruction. However, this reconstruction is only *used* for the cell under consideration. In the SV method, the reconstruction is based on data at all the CVs inside a SV, and this reconstruction is *used* in all the CVs of a SV. It is this subtle difference that produces the different behavior in the reconstruction. Basic numerical analysis books, e.g., [23], tell us that high-order Lagrange polynomials for a nonpolynomial function based on nodal data at equidistant points are highly oscillatory. It is not generally true that higher degree interpolation polynomials yield more accurate approximations. In fact, for equidistant points of interpolation, one should use polynomials of relatively low order. A well-known example is that the interpolation polynomial sequences generated by increasing the polynomial order for function $f(x) = \frac{1}{1+x^2}$ (considered by Runge) actually diverge; i.e., the error goes to infinity as the order of the interpolation polynomial is raised. On the other hand, if nonuniform grid points corresponding to the zeros of the Chebyshev polynomial are used to generate a Lagrange interpolation polynomial, the interpolation error is uniformly distributed all over the region covered by the grid points.

If a reconstruction is used in a small interval in the middle of the region covered by the grid points (such as in the case of a centered high-order finite volume reconstruction), the interpolation polynomial using equidistant points is far more accurate than the interpolation polynomial using the zeros of the Chebyshev polynomial. In fact, the error can be estimated using (p. 268 of [23])

$$\frac{\text{Error}_{\max,Ch}}{\text{Error}_{\max,Eq}} \approx \frac{1}{2}\left(\frac{e}{2}\right)^{\gamma} = \frac{1}{2}(1.3591\ldots)^{\gamma},$$

where γ is the order of the polynomial and $\text{Error}_{\max,Ch}$ is the maximum error of the Lagrangian polynomial using zeros of Chebyshev polynomial as the interpolation points, while $\text{Error}_{\max,Eq}$ is the maximum error using equidistant interpolation points. This estimate explains why uniform grids in a FV algorithm produce the least error. Another

observation from the above discussion is that one-sided high-order reconstructions on equidistant points must be used with care because the interpolation polynomial can be highly oscillatory.

To further study the characteristics of the polynomial reconstruction based on cell-averaged variables, we consider the standard interval $[-1, 1]$ as the SV. The reconstruction polynomial given the cell-averaged variables on the CVs is then given in (3.21). It is clear that the reconstruction polynomial can be interpreted as an expansion with respect to a basis set $\{\phi_l(x)\}$ with expansion coefficients $\{\bar{u}_{i,l}\}$. The polynomials in the basis set, are order $k-1$ polynomials, which are dependent only on the distribution of the $k+1$ "collocations" points $\{x_{i,j+1/2}\}$. The basis set $\{\phi_l\}$ for $k=3$ and $k=6$ using equidistant grid points are plotted in Fig. 3. Note that for $k=6$, the basis polynomials near the two boundary grid points are indeed quite oscillatory. The values of the basis set at the two boundary points range from -4 to 4, making the interpolations at the two end grid points potentially very oscillatory.

For comparison purposes, we also plot the Lagrange interpolation coefficients for node-value-based reconstructions. In this case, we assume state variables are known at the CV centers; i.e., $u_{i,j}$ are defined at the position $x_{i,j}=(x_{i,j-1/2}+x_{i,j+1/2})/2$. The Lagrange interpolation polynomial is then

$$\tilde{p}_i(x)=\sum_{l=1}^{k}\varphi_l(x)u_{i,l}, \tag{5.1a}$$

where

$$\varphi_l(x)=\prod_{q=0,q\neq l}^{k}\frac{x-x_{i,q}}{x_{i,l}-x_{i,q}}. \tag{5.1b}$$

The basis set $\{\varphi_l\}$ for $k=3$ and $k=6$ with equidistant points are plotted in Fig. 4. Note that the node-value-based interpolation polynomials are comparable to the cell-average-based interpolation polynomials in the case of $k=3$, but are much more oscillatory in the case of $k=6$ near the two end grid points.

If the grid points in the standard interval $[-1, 1]$ are the Gauss–Lobatto points defined by

$$x_{i,j+1/2}=-\cos\left(\frac{j\pi}{k}\right),\quad j=0,\ldots,k, \tag{5.2}$$

the basis polynomials $\{\phi_l\}, \{\varphi_l\}$ are much less oscillatory as shown in Figs. 5 and 6, which show the cell-average-based polynomial basis set, and the node-value-based polynomial basis set respectively. Note that all the basis polynomials are bounded between -1 and 1.7.

It is clear that it is necessary to cluster grid points near the boundaries to make the interpolation polynomial bounded. Another natural choice of "clustering" function is the hyperbolic tangent defined by

$$x_{i,j+1/2}=\frac{\tanh\left(\frac{2\mu j}{k}-\mu\right)}{\tanh(\mu)},\quad j=0,\ldots,k, \tag{5.3}$$

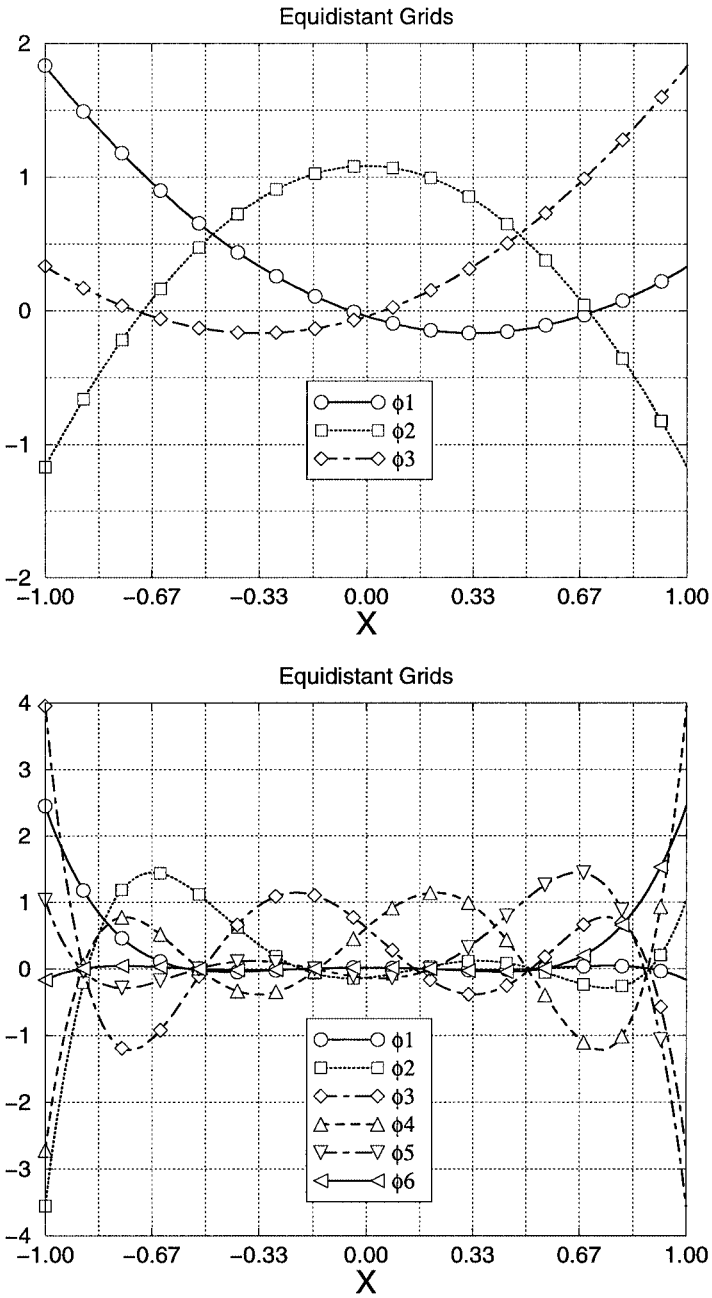

FIG. 3. The basis polynomials $\{\phi_l\}$ with known cell-averaged variables for $k = 3$ and 6 using equidistant grid points.

where μ is a constant, which controls the degree of grid clustering near the two end grid points. The larger the value of μ is, the stronger the grid clustering. The cell-average-based interpolation polynomials with three different values of μ are plotted in Fig. 7 for $k = 6$. The corresponding grid points are also shown. Note that when $\mu = 0.6$, the grid points are still quite even. As a result, the basis polynomials are highly oscillatory near the two end grid

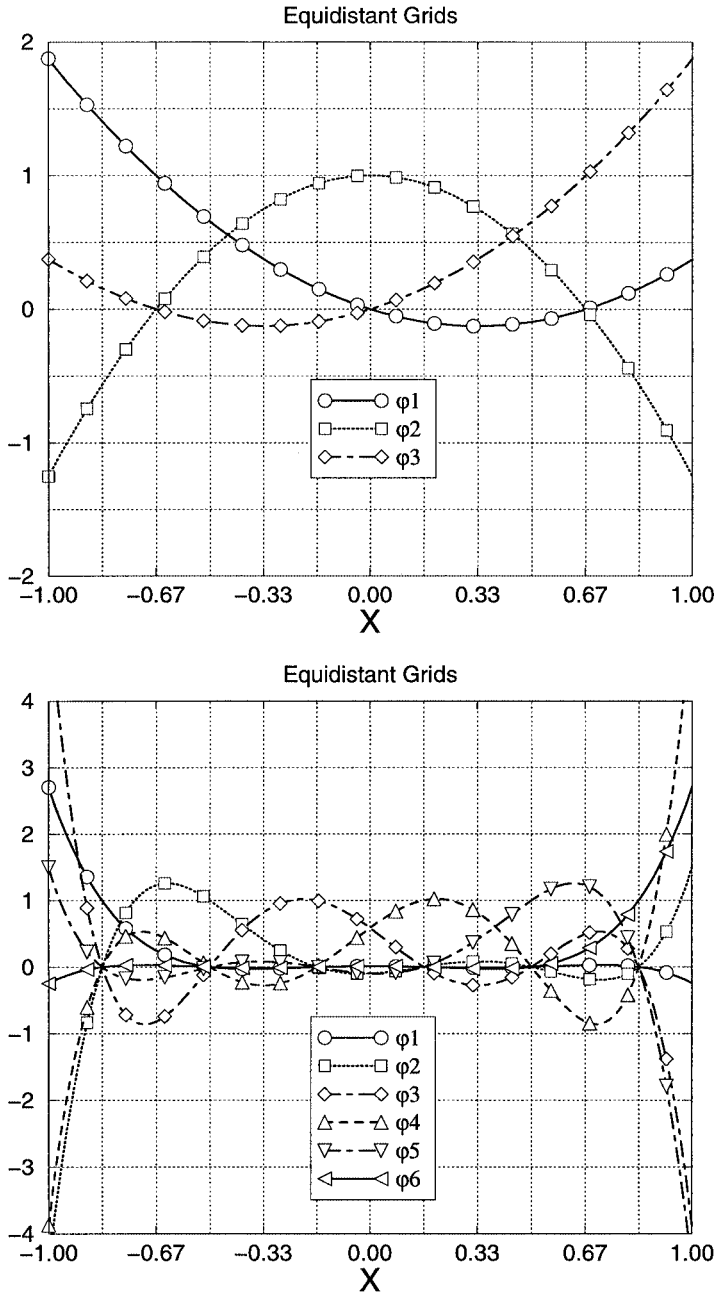

FIG. 4. The basis polynomials $\{\varphi_l\}$ with known node-based variables for $k = 3$ and 6 using equidistant grid points.

points. The value of $\mu = 1.6$ produces a grid similar to the Gauss–Lobatto grid. Therefore, the basis polynomials are also quite similar. When $\mu = 2.6$, the grid points are strongly clustered near the boundaries. Even so, the basis polynomials are nicely bounded between -0.6 and 1.5. This set of grid points is expected to be quite useful when a viscous boundary layer near a wall needs to be resolved.

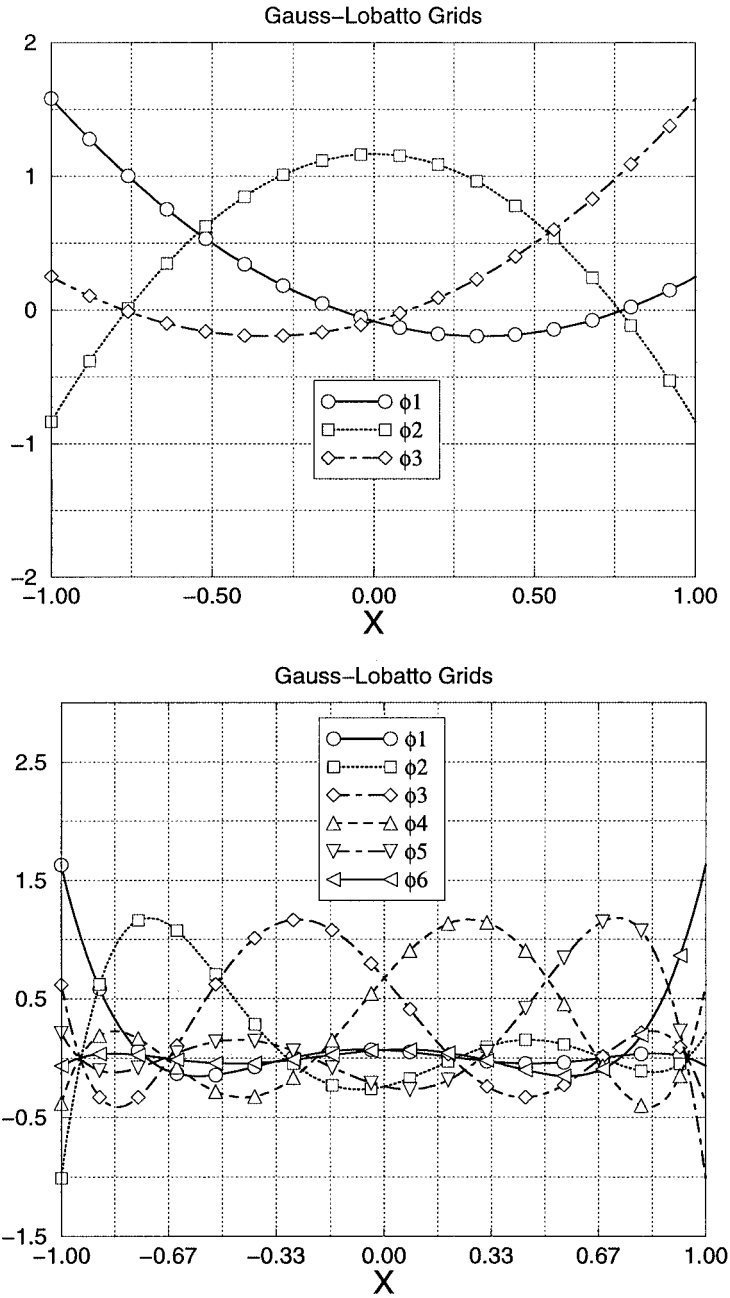

FIG. 5. The basis polynomials $\{\phi_l\}$ with known cell-averaged variables for $k = 3$ and 6 using Gauss–Lobatto grid points.

6. TVD AND TVB LIMITERS

The Gibbs phenomenon associated with high-order schemes in the presence of discontinuities causes loss of monotinicity in the solution of hyperbolic conservation laws. Godunov [17] first proved that there are no linear second- or higher order schemes which guarantee monotonicity. Therefore high-order monotonic schemes, if they exist, must be nonlinear.

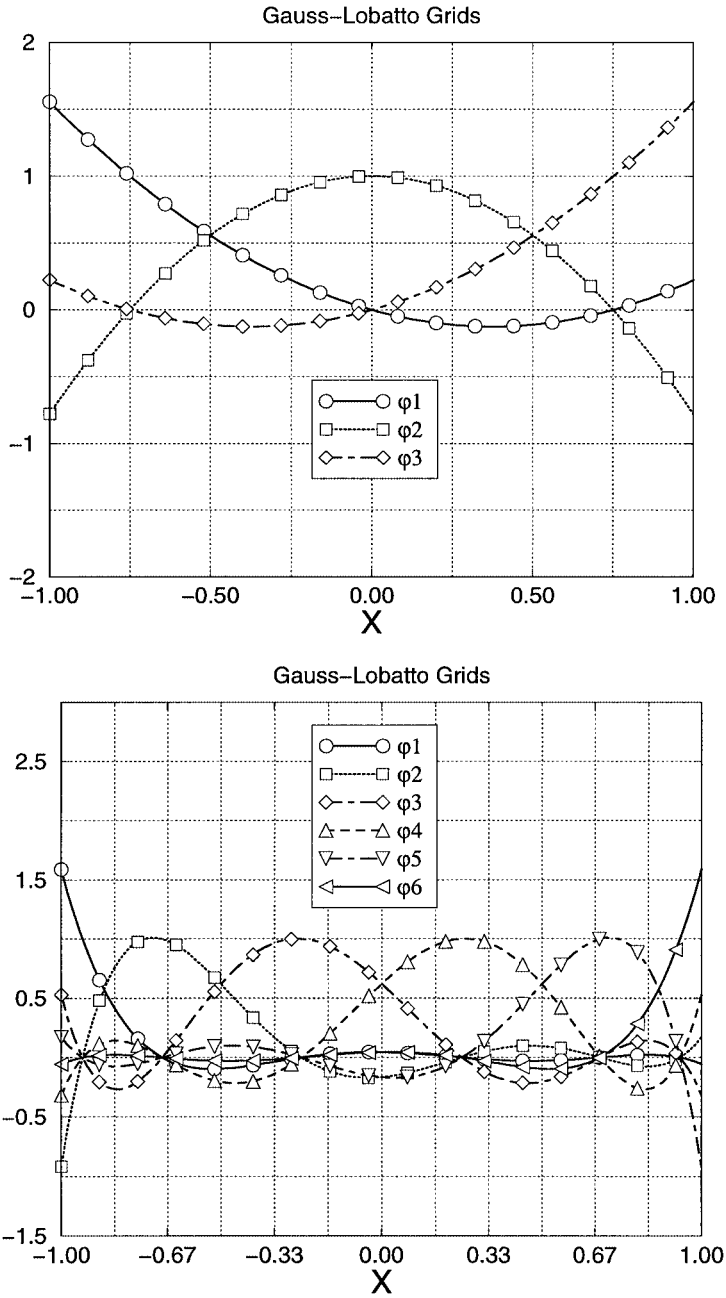

FIG. 6. The basis polynomials $\{\varphi_l\}$ with known node-based variables for $k = 3$ and $k = 6$ using Gauss-Lobatto grid points.

There are two popular, but different limiting approaches to achieve monotonicity. One approach, represented by the FCT algorithm of Boris *et al*. [6, 7], is to limit the antidiffusive flux so that the solution at the next time level remains monotonic. The other approach, originally developed by van Leer [43, 44], is to limit the reconstruction so that the reconstructed solution is monotonic. Although both approaches have been used very successfully over the

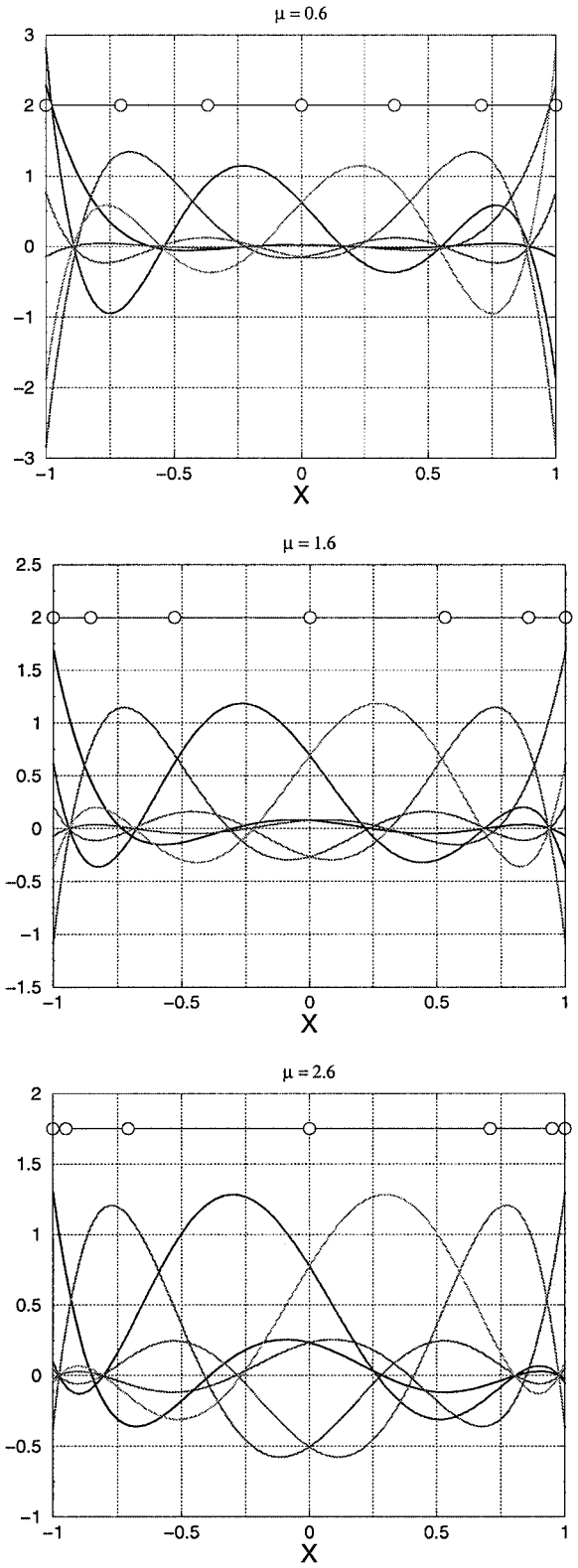

FIG. 7. The basis polynomials $\{\phi_l\}$ with known cell-averaged variables for $k = 6$ using grid points defined with a hyperbolic tangent function with various degrees of grid clustering near the two end points.

past several decades, we choose to use a limiter similar to the latter. That is to say that the reconstructed solution is limited. Other limiters, such as the problem independent limiter developed in [8], will be considered in a future publication.

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

$$\int_{x_{i,j-1/2}}^{x_{i,j+1/2}} p_i(x)\,dx = \bar{u}_{i,j}h_{i,j}, \quad j = 1, \ldots, k. \tag{6.1}$$

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

$$\frac{d\bar{u}_{i,j}}{dt}h_{i,j} + \left(f_{i,j+1/2} - f_{i,j-1/2}\right) = 0, \tag{6.2a}$$

where

$$f_{i,j+1/2} = f_{Riemann}\left(u^-_{i,j+1/2}, u^+_{i,j+1/2}\right). \tag{6.2b}$$

If $f = cu$, with $c > 0$, then we have

$$f_{i,j+1/2} = cu^-_{i,j+1/2}. \tag{6.2c}$$

Scheme (6.2a) then becomes

$$\frac{d\bar{u}_{i,j}}{dt} = -c\frac{\left(u^-_{i,j+1/2} - u^-_{i,j-1/2}\right)}{h_{i,j}}. \tag{6.3}$$

If the reconstruction is written in the following form,

$$\begin{aligned} u^-_{i,j+1/2} &= \bar{u}_{i,j} + \Delta u^+_{i,j} \\ u^+_{i,j-1/2} &= \bar{u}_{i,j} - \Delta u^-_{i,j}, \end{aligned} \tag{6.4}$$

Eq. (6.3) can be written as

$$\begin{aligned} \frac{d\bar{u}_{i,j}}{dt} &= -\frac{c}{h_{i,j}}(\bar{u}_{i,j} + \Delta u^+_{i,j} - \bar{u}_{i,j-1} - \Delta u^+_{i,j-1}) \\ &= -\frac{c}{h_{i,j}}\left(1 + \frac{\Delta u^+_{i,j}}{\bar{u}_{i,j} - \bar{u}_{i,j-1}} - \frac{\Delta u^+_{i,j-1}}{\bar{u}_{i,j} - \bar{u}_{i,j-1}}\right)(\bar{u}_{i,j} - \bar{u}_{i,j-1}). \end{aligned} \tag{6.5}$$

Following [21], a sufficient condition for (6.5) to be TVD in the cell-averaged means (TVDM) is

$$1 + \frac{\Delta u^+_{i,j}}{\bar{u}_{i,j} - \bar{u}_{i,j-1}} - \frac{\Delta u^+_{i,j-1}}{\bar{u}_{i,j} - \bar{u}_{i,j-1}} \geq 0. \tag{6.6}$$

If a wave is traveling in the negative x direction—i.e., $c < 0$—then the following sufficient condition for (6.2) to be TVD is obtained:

$$1 - \frac{\Delta u^-_{i,j+1}}{\bar{u}_{i,j+1} - \bar{u}_{i,j}} + \frac{\Delta u^-_{i,j}}{\bar{u}_{i,j+1} - \bar{u}_{i,j}} \geq 0. \tag{6.7}$$

Generally speaking, (6.6) and (6.7) are not satisfied by a high-order reconstruction near a discontinuity. Therefore, we limit the values of $\Delta u^+_{i,j}$ and $\Delta u^-_{i,j}$ to be $\Delta \tilde{u}^+_{i,j}$ and $\Delta \tilde{u}^-_{i,j}$ so that (6.6) and (6.7) are satisfied everywhere:

$$1 + \frac{\Delta \tilde{u}^+_{i,j}}{\bar{u}_{i,j} - \bar{u}_{i,j-1}} - \frac{\Delta \tilde{u}^+_{i,j-1}}{\bar{u}_{i,j} - \bar{u}_{i,j-1}} \geq 0 \tag{6.8}$$

$$1 - \frac{\Delta \tilde{u}^-_{i,j+1}}{\bar{u}_{i,j+1} - \bar{u}_{i,j}} + \frac{\Delta \tilde{u}^-_{i,j}}{\bar{u}_{i,j+1} - \bar{u}_{i,j}} \geq 0. \tag{6.9}$$

It can be easily verified that the limiter below satisfies (6.8) and (6.9)

$$\Delta \tilde{u}^+_{i,j} = \min \operatorname{mod}(\Delta u^+_{i,j}, \bar{u}_{i,j+1} - \bar{u}_{i,j}, \bar{u}_{i,j} - \bar{u}_{i,j-1}) \tag{6.10}$$

$$\Delta \tilde{u}^-_{i,j} = \min \operatorname{mod}(\Delta u^-_{i,j}, \bar{u}_{i,j+1} - \bar{u}_{i,j}, \bar{u}_{i,j} - \bar{u}_{i,j-1}), \tag{6.11}$$

where the minmod function is defined as

$$\min \operatorname{mod}(x, y, z) = \begin{cases} s \min(|x|, |y|, |z|), & \text{if } s = \operatorname{sign}(x) = \operatorname{sign}(y) = \operatorname{sign}(z) \\ 0 & \text{otherwise.} \end{cases} \tag{6.12}$$

Note that this TVD limiter is similar to the $\Lambda\Pi^1_h$ limiter in the DG method [11–13], but applied to the cell-averaged variable for the CVs, rather than for the "elements." We will name this limiter CVTVDM (CV-wise TVDM) limiter.

Unfortunately, any TVD scheme is only first-order accurate at extrema, even if the extrema is smooth. To restore the full accuracy at smooth extrema, we follow the TVB [40] idea developed by Shu, and used in the DG method [11–13]. Instead of the limiter given in (6.10) and (6.11), we use the limiter

$$\Delta \tilde{u}^+_{i,j} = \begin{cases} \Delta u^+_{i,j} & \text{if } |\Delta u^+_{i,j}| \leq M h^2_{i,j} \\ \min \operatorname{mod}(\Delta u^+_{i,j}, \bar{u}_{i,j+1} - \bar{u}_{i,j}, \bar{u}_{i,j} - \bar{u}_{i,j-1}) & \text{otherwise} \end{cases}, \tag{6.13}$$

$$\Delta \tilde{u}^-_{i,j} = \begin{cases} \Delta u^-_{i,j} & \text{if } |\Delta u^-_{i,j}| \leq M h^2_{i,j} \\ \min \operatorname{mod}(\Delta u^-_{i,j}, \bar{u}_{i,j+1} - \bar{u}_{i,j}, \bar{u}_{i,j} - \bar{u}_{i,j-1}) & \text{otherwise} \end{cases}, \tag{6.14}$$

where M represents some measure of the second derivative of the function. In this paper, we select M to be the maximum absolute value of the second derivative over the computational domain. This TVB limiter is similarly named the CVTVBM limiter.

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 better resolutions for discontinuities than the DG method. For comparison purposes, we mimic the $\Lambda\Pi^1_h$ limiter developed in the DG method [11–13]. An exact implementation of the $\Lambda\Pi^1_h$

limiter is not cost-effective because the so-called P^1 component of the reconstruction is not available. This limiter is therefore formulated in terms of the SV notations approximately. This limiter is named SVTVDM (SV-wise TVDM) limiter here. Therefore, given $\{\bar{u}_{i,j}\}$, we need to find $\{\bar{v}_{i,j}\}$, such that

$$\{\bar{v}_{i,j}\} = \text{SVTVDM}(\{\bar{u}_{i,j}\}). \tag{6.15}$$

Here are the steps in the SV method to obtain SVTVDM:

- First compute the cell-averaged variables for the SVs according to

$$\bar{u}_i = \frac{\sum_{j=1}^{k} \bar{u}_{i,j} h_{i,j}}{\sum_{j=1}^{k} h_{i,j}}, \quad i = 1, \ldots, N. \tag{6.16}$$

- Compute the limited solutions at the SV boundaries using the reconstructed solutions at the boundaries,

$$u^{+}_{i,1/2} = \bar{u}_i - \text{min mod}\left[\bar{u}_i - p_i\left(x_{i,1/2}\right), \bar{u}_i - \bar{u}_{i-1}, \bar{u}_{i+1} - \bar{u}_i\right] \tag{6.17a}$$

$$u^{-}_{i,k+1/2} = \bar{u}_i + \text{min mod}\left[p_i\left(x_{i,k+1/2}\right) - \bar{u}_i, \bar{u}_i - \bar{u}_{i-1}, \bar{u}_{i+1} - \bar{u}_i\right]. \tag{6.17b}$$

- If $u^{+}_{i,1/2} = p_i(x_{i,1/2})$ and $u^{-}_{i,k+1/2} = p_i(x_{i,k+1/2})$, then set

$$\bar{v}_{i,j} = \bar{u}_{i,j}, \quad j = 1, \ldots, k. \tag{6.18}$$

Otherwise, a linear distribution is assumed for S_i; i.e.,

$$q_i(x) = \bar{u}_i + \lambda(x - x_i), \tag{6.19}$$

where

$$\lambda = \text{min mod}\left[\frac{2(\bar{u}_i - \bar{u}_{i-1})}{h_i}, \frac{2(\bar{u}_{i+1} - \bar{u}_i)}{h_i}\right]. \tag{6.20}$$

Then with this linear distribution, the cell-averaged variables on the CVs can be computed according to

$$\bar{v}_{i,j}(x) = \bar{u}_i + \lambda(x_{i,j} - x_i). \tag{6.21}$$

Similarly we can also define a SV-wise TVBM limiter by changing the limiting used in (6.17) to the following:

$$u^{+}_{i,1/2} = \begin{cases} \bar{u}_i - p_i\left(x_{i,1/2}\right) & \text{if } \left|\bar{u}_i - p_i\left(x_{i,1/2}\right)\right| \le Mh_i^2 \\ \bar{u}_i - \text{min mod}\left[\bar{u}_i - p_i\left(x_{i,1/2}\right), \bar{u}_i - \bar{u}_{i-1}, \bar{u}_{i+1} - \bar{u}_i\right] & \text{otherwise} \end{cases} \tag{6.22a}$$

$$u^{-}_{i,k+1/2} = \begin{cases} p_i\left(x_{i,k+1/2}\right) - \bar{u}_i & \text{if } \left|\bar{u}_i - p_i\left(x_{i,k+1/2}\right)\right| \le Mh_i^2 \\ \bar{u}_i + \text{min mod}\left[p_i\left(x_{i,k+1/2}\right) - \bar{u}_i, \bar{u}_i - \bar{u}_{i-1}, \bar{u}_{i+1} - \bar{u}_i\right] & \text{otherwise} \end{cases}. \tag{6.22b}$$

This limiter is called SVTVBM limiter. A variety of test cases will be presented to evaluate the performances of the above limiters.

7. NUMERICAL TESTS

In this section we test the SV method on several model problems, both linear and nonlinear. We numerically verify the order of accuracy claims made earlier for schemes up to sixth-order accurate. For all our computations, we use the Lax–Frientrichs flux because of its smoothness.

7.1. *Accuracy Study with Linear Wave Equation*

In this case, we test the accuracy of the SV method on the linear equation:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0, \qquad -1 \le x \le 1$$
$$u(x, 0) = u_0(x), \quad \text{periodic boundary condition.}$$

No limiters were used in the accuracy study. In Table I, we show the L_1 and L_∞ errors produced using SV schemes of various orders of accuracy on equidistant control volumes at time $t = 1$, given the initial condition $u_0(x) = \sin(\pi x)$. The third-order TVD Runge–Kutta time-integration scheme was used for all the computations presented here. Although the order of the time-marching scheme is only third-order accurate, the errors presented in the table are time-step independent because the time step Δt was made small enough

TABLE I
Accuracy on $u_t + u_x = 0$, with $u_0(x) = \sin(\pi x)$ with Equidistant Control Volumes

Order of accuracy	NDOF	L_∞ error	L_∞ order	L_1 error	L_1 order
2	20	5.32e-2	—	3.28e-2	—
	40	1.41e-2	1.92	8.71e-3	1.91
	80	3.56e-3	1.99	2.24e-3	1.96
	160	8.94e-4	1.99	5.65e-4	1.99
	320	2.24e-4	2.00	1.42e-4	1.99
	640	5.59e-5	2.00	3.55e-5	2.00
3	30	4.12e-3	—	2.40e-3	—
	60	5.31e-4	2.96	3.08e-4	2.96
	120	6.66e-5	3.00	3.90e-5	2.98
	240	8.33e-6	3.00	4.90e-6	2.99
	480	1.01e-6	3.00	6.12e-7	3.00
	960	1.27e-7	3.03	7.45e-8	3.04
4	20	3.23e-3	—	1.63e-3	—
	40	1.91e-4	4.08	1.13e-4	3.85
	80	1.47e-5	3.70	7.04e-6	4.01
	160	8.44e-7	4.12	4.30e-7	4.03
	320	5.39e-8	3.97	2.71e-8	4.00
	640	4.97e-9	3.44	1.75e-9	3.95
5	20	1.27e-3	—	4.40e-4	—
	40	7.54e-5	4.33	2.19e-5	4.07
	80	8.15e-6	2.46	3.98e-6	3.21
	160	1.06e-5	Negative	4.57e-6	Negative
	320	4.58e-4	Negative	1.79e-4	Negative

TABLE II

Accuracy on $u_t + u_x = 0$, with $u_0(x) = \sin(\pi x)$ with Gauss–Lobatto Points to Subdivide Control Volumes

Order of accuracy	NDOF	L_∞ error	L_∞ order	L_1 error	L_1 order
3	30	2.67e-3	—	1.24e-3	—
	60	3.65e-4	2.87	1.61e-4	2.95
	120	4.67e-5	2.97	2.05e-5	2.97
	240	5.90e-6	2.98	2.59e-6	2.98
	480	7.41e-7	2.99	3.23e-7	3.00
4	20	2.26e-3	—	7.30e-4	—
	40	1.60e-4	3.82	5.07e-5	3.85
	80	9.72e-6	4.04	3.18e-6	3.99
	160	6.15e-7	3.98	2.00e-7	3.99
	320	3.85e-8	4.00	1.26e-8	3.99
	640	2.41e-9	4.00	7.87e-10	4.00
5	20	5.30e-4	—	1.46e-4	—
	40	1.96e-5	4.76	4.58e-6	4.99
	80	6.50e-7	4.91	1.49e-7	4.94
	160	2.13e-8	4.93	4.91e-9	4.92
	320	6.13e-10	5.12	1.57e-10	4.97
6	30	1.28e-5	—	2.57e-6	—
	60	1.88e-7	6.09	4.08e-8	5.98
	120	2.98e-9	5.98	6.49-10	5.97
	240	4.51e-11	6.05	1.04e-11	5.96

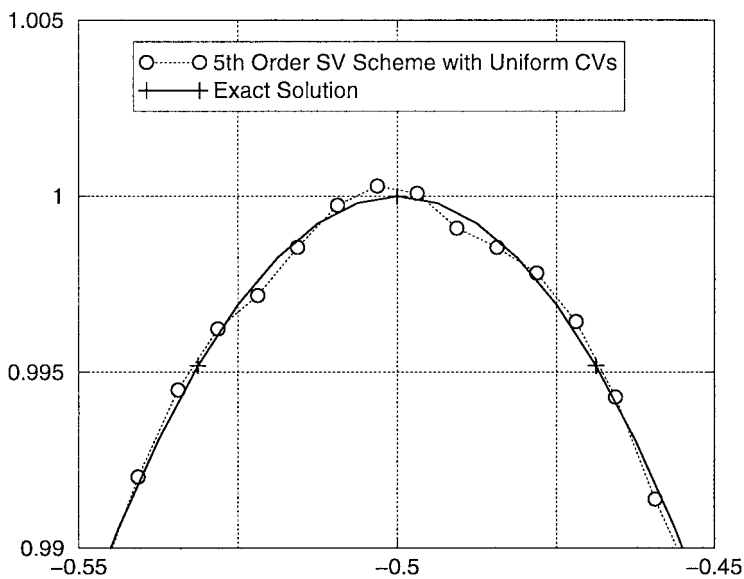

FIG. 8. A close-up view of the exact solution and the computed CV-averaged means using a fifth-order SV scheme on 64 spectral volumes with uniform control volumes. The "+" signs denote SV boundaries.

TABLE III

Accuracy on $u_t + u_x = 0$, with $u_0(x) = \sin^4(\pi x)$ with Gauss–Lobatto Points to Subdivide Control Volumes

Order of accuracy	NDOF	L_∞ error	L_∞ order	L_1 error	L_1 order
2	20	2.63e-1	—	1.07e-1	—
	40	1.15e-1	1.19	5.26e-2	1.02
	80	3.47e-2	1.73	1.63e-2	1.69
	160	9.21e-3	1.91	4.29e-3	1.93
	320	2.34e-3	1.98	1.09e-3	1.98
	640	5.89e-4	1.99	2.75e-4	1.99
3	30	4.81e-2	—	2.80e-2	—
	60	7.38e-3	2.70	3.70e-3	2.92
	120	9.30e-4	2.99	4.56e-4	3.02
	240	1.22e-4	2.93	5.76e-5	2.98
	480	1.57e-5	2.96	7.25e-6	2.99
	960	1.98e-6	2.99	9.09e-7	3.00
4	20	6.76e-2	—	3.26e-2	—
	40	6.21e-3	3.44	3.17e-3	3.36
	80	5.77e-4	3.43	2.27e-4	3.80
	160	3.70e-5	3.96	1.49e-5	3.93
	320	2.37e-6	3.96	9.53e-7	3.97
	640	1.49e-7	3.99	5.98e-8	3.99
5	20	4.41e-2	—	1.78e-2	—
	40	2.41e-3	4.19	1.09e-3	4.03
	80	1.08e-4	4.48	3.69e-5	4.88
	160	3.60e-6	4.91	1.24e-6	4.90
	320	1.19e-7	4.92	3.87e-8	5.00
	640	3.03e-9	5.30	1.13e-9	5.10
6	30	6.21e-3	—	1.95e-3	—
	60	1.19e-4	5.71	3.51e-5	5.80
	120	2.04e-6	5.87	5.80e-7	5.92
	240	3.28e-8	5.96	9.27e-9	5.97
	480	5.08e-10	6.01	1.46e-10	5.99

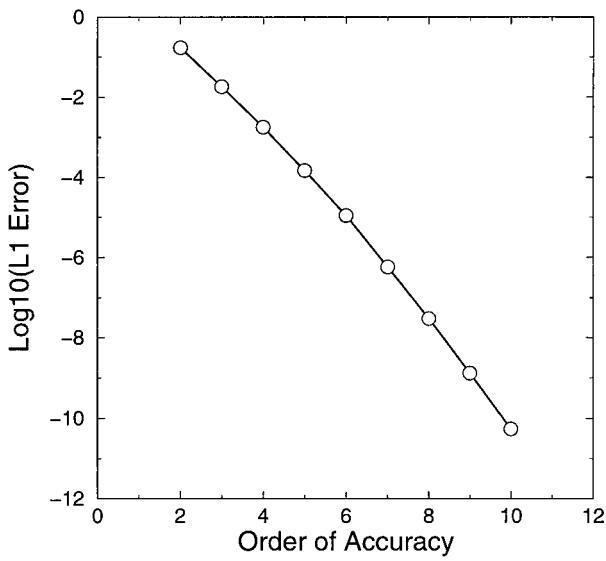

FIG. 9. Demonstration of spectral accuracy with the linear wave equation on four spectral volumes.

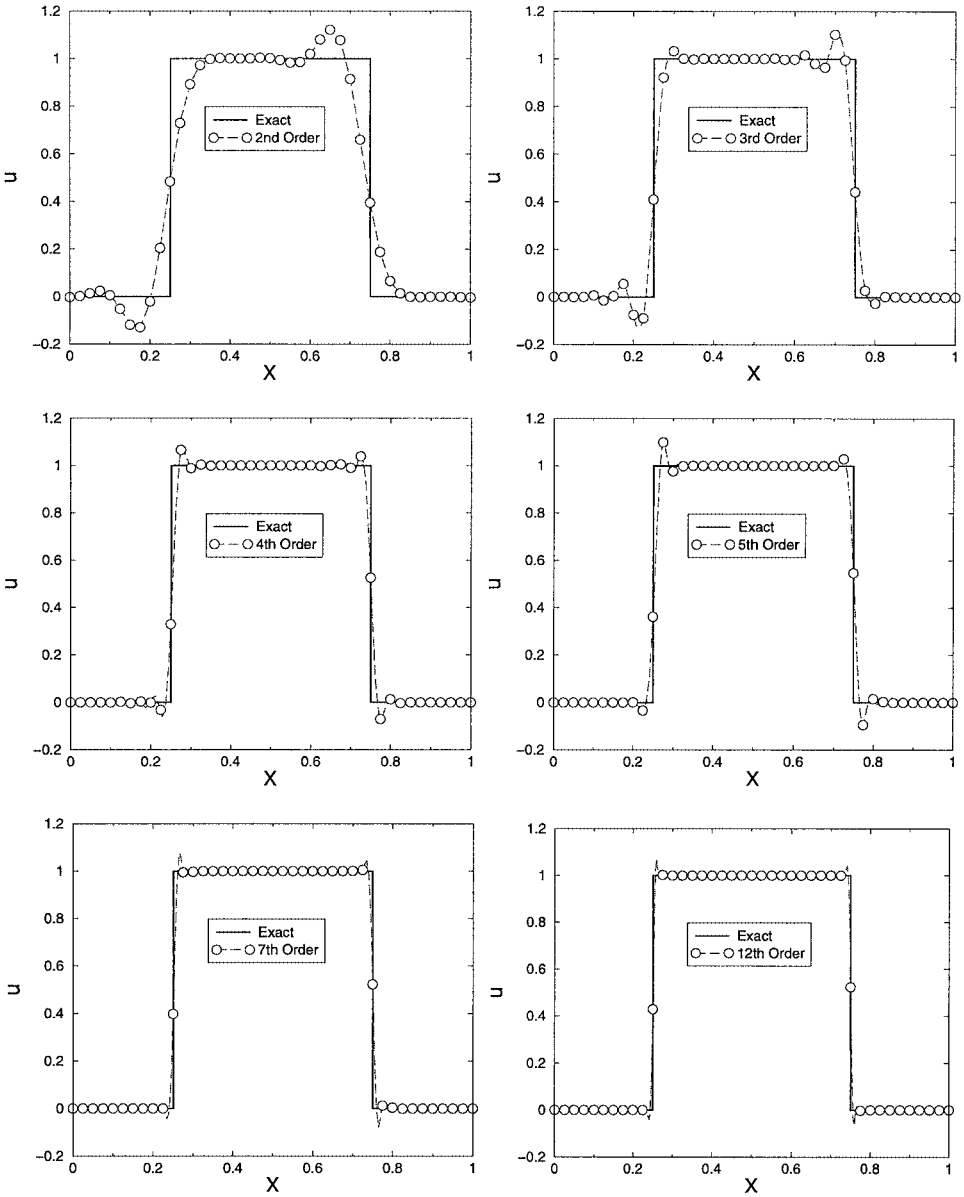

FIG. 10. Simulation of a traveling square wave with SV schemes of various accuracy without limiters at $t = 1$.

so that the errors are dominated by the spatial discretization. The parameter NDOF means the number of degrees of freedom and is defined as $N \times k$, i.e., the total number of SVs times the number of CVs in each SV. It is obvious that the second- and third-order SV schemes with equidistant CVs behave properly and are capable of achieving the expected formal order of accuracy on quite coarse grids. However, the performance of the fourth-order scheme is sporadic. When the grid is relatively coarse, the method is able to achieve the formal fourth-order accuracy. However, there is a clear loss of accuracy when the grid is refined in the L_∞ norm. This may suggest that some local oscillations due to equidistant distributions of the CVs are developing. The fifth-order SV scheme with equidistant-CVs could never

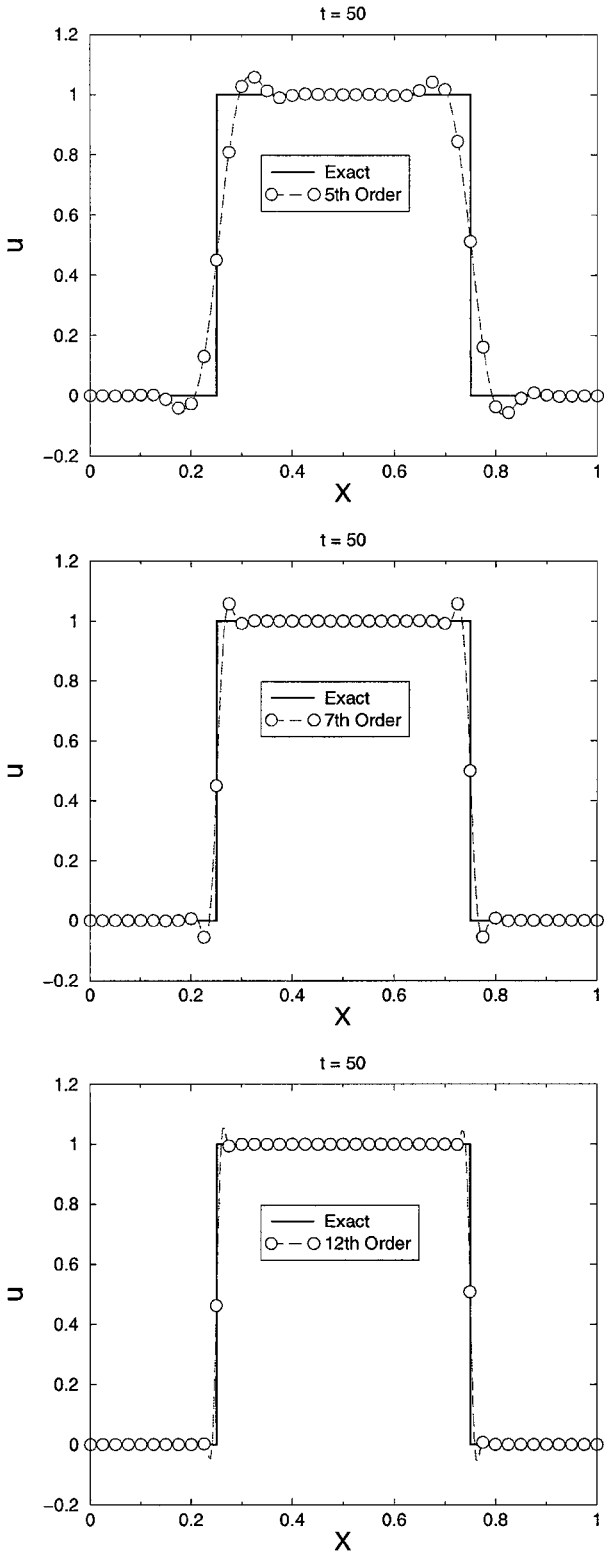

FIG. 11. Simulation of a traveling square wave with SV schemes of various accuracy without limiters at $t = 50$.

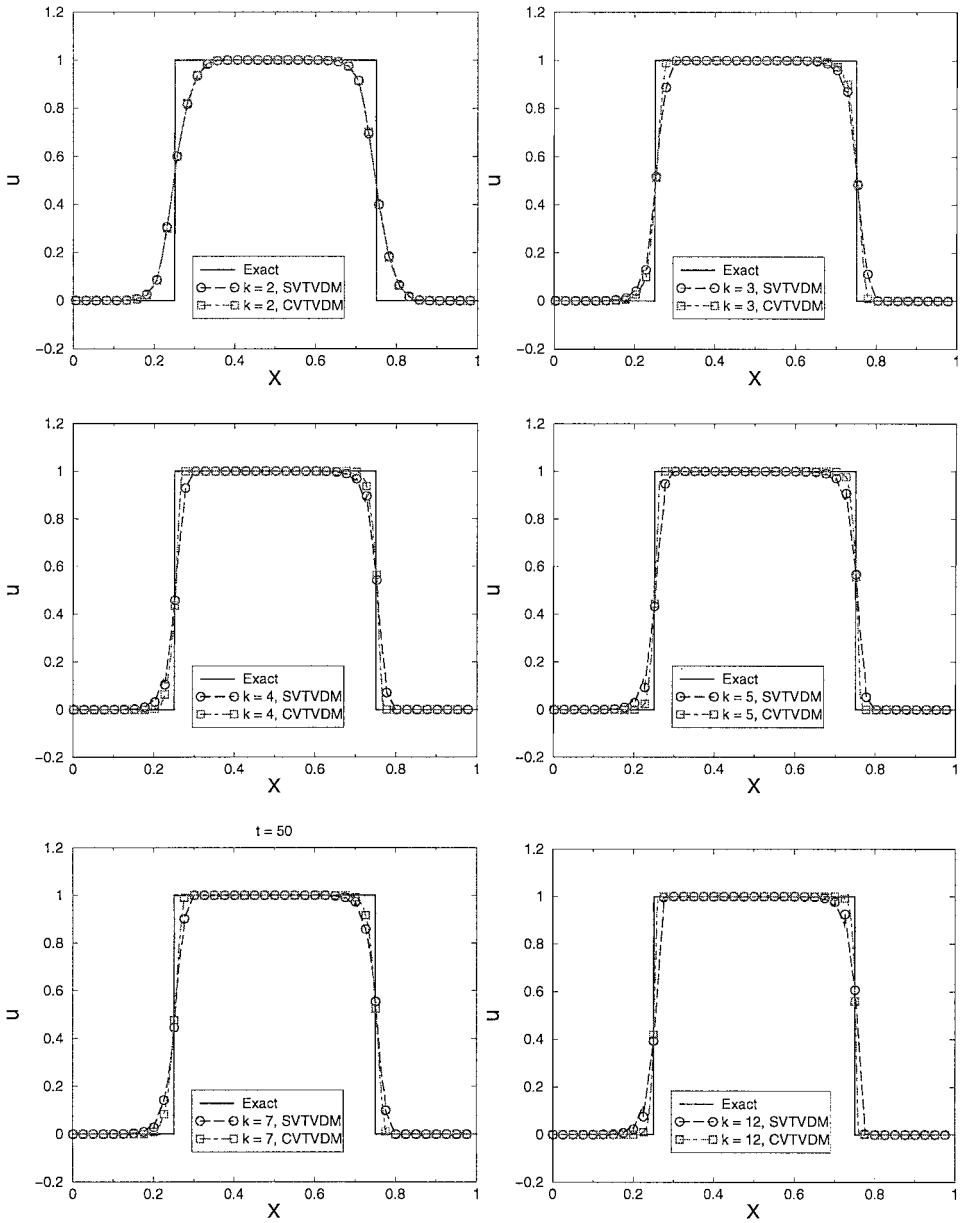

FIG. 12. Simulation of a traveling square wave at $t = 1$ with SV schemes of various orders of accuracy using SVTVDM and CVTVDM limiters.

achieve the formal fifth-order of accuracy, and as the grid is refined, the solution errors actually increase in both the L_1 and L_∞ norms, giving a negative order of accuracy. This may indicate that the Lagrange interpolation is very oscillatory when the order of the interpolation polynomial reaches 4. As a matter of fact, this is confirmed in Fig. 8, which compares the exact solution and the numerically computed CV-averaged means on the finest grid with 64 SVs. It should be pointed out the numerical solution is independent of the time step used. Therefore, the local oscillations shown in Fig. 8 are caused solely by the space discretization.

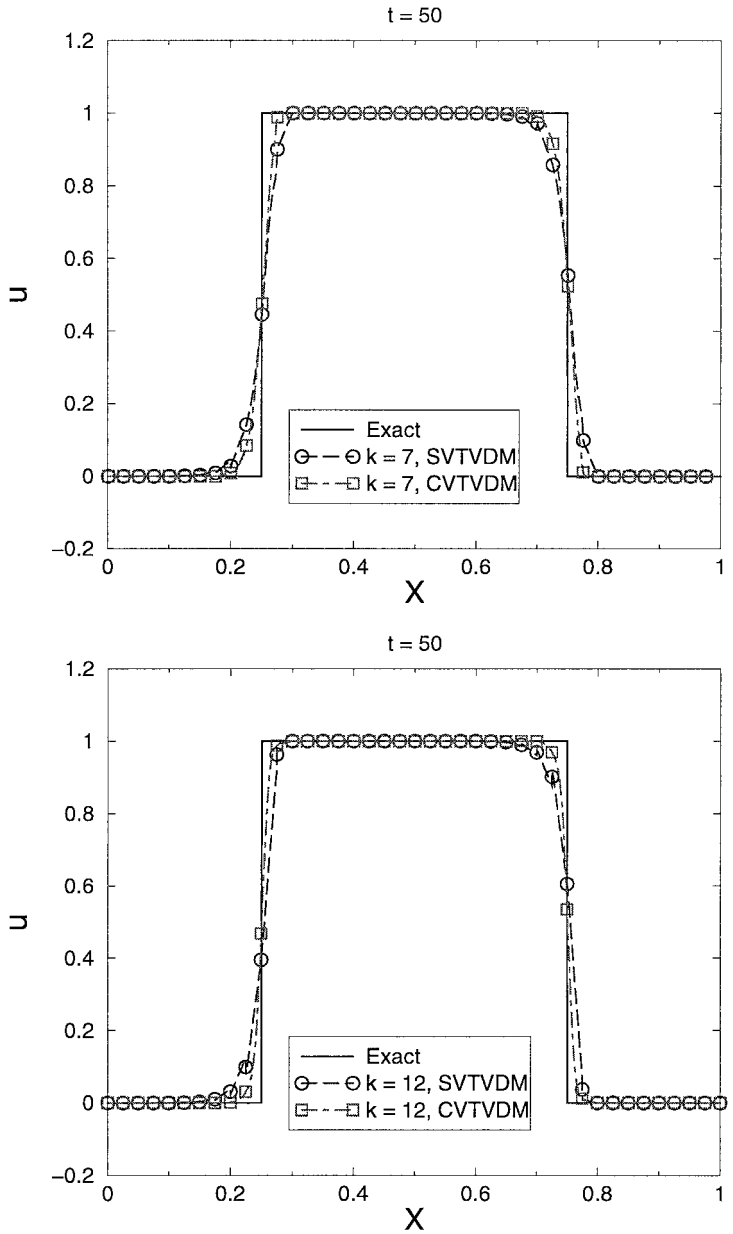

FIG. 13. Simulation of a traveling square wave at $t = 50$ with SV schemes of seventh- and twelfth-order of accuracy using CVTVDM and SVTVDM limiters.

Therefore in the next test, we use the Gauss–Lobatto points to partition the SVs. All the other factors remain exactly the same. Table II shows the L_1 and L_∞ errors at time $t = 1$ computed with the SV schemes from third- to sixth-order accurate. Note that for a second-order SV scheme, the Gauss–Lobatto point is in the middle of the domain, resulting in equidistant CVs. It is clear from Table II that the SV schemes of various orders of accuracy using the Gauss–Lobatto points are capable of achieving the formal order of accuracy without any difficulty in both the L_1 and L_∞ norms. Comparing Tables II with I, we notice

TABLE IV
Accuracy on $u_t + uu_x = 0$, with $u_0(x) = 1 + \frac{1}{2}\sin(\pi x)$ at $t = 0.3$ with Gauss–Lobatto Points to Subdivide Control Volumes

Order of accuracy	NDOF	L_∞ error	L_∞ order	L_1 error	L_1 order
2	20	2.74e-2	—	1.16e-2	—
	40	9.97e-3	1.46	3.07e-3	1.92
	80	3.09e-3	1.69	7.82e-4	1.97
	160	8.11e-4	1.93	1.95e-4	2.00
	320	2.08e-4	1.96	4.88e-5	2.00
	640	5.24e-5	1.99	1.22e-5	2.00
3	30	1.17e-2	—	1.44e-3	—
	60	2.11e-3	2.47	1.92e-4	2.91
	120	3.90e-4	2.44	2.70e-5	2.83
	240	5.77e-5	2.76	3.66e-6	2.88
	480	7.72e-6	2.90	4.80e-7	2.93
	960	9.91e-7	2.96	6.15e-8	2.96
4	20	1.64e-2	—	2.90e-3	—
	40	7.81e-4	4.39	1.09e-4	4.73
	80	2.59e-4	1.59	1.25e-5	3.12
	160	1.67e-5	3.96	7.17e-7	4.12
	320	1.11e-6	3.91	4.35e-8	4.04
	640	7.01e-8	3.99	2.71e-9	4.00
5	20	1.45e-2	—	2.09e-3	—
	40	8.15e-4	4.15	6.45e-5	5.02
	80	9.10e-5	3.16	3.45e-6	4.22
	160	4.56e-6	4.32	9.54e-8	5.18
	320	1.89e-7	4.59	3.78e-9	4.66
	640	6.73e-9	4.81	1.32e-10	4.84
6	30	2.96e-3	—	2.74e-4	—
	60	5.97e-5	5.63	3.66e-6	6.23
	120	5.48e-6	3.45	1.45e-7	4.66
	240	8.76e-8	5.97	1.77e-9	6.36
	480	1.55e-9	5.82	2.53e-11	6.13

that the Gauss–Lobatto points also produce smaller errors in both the L_1 and L_∞ norms than equidistant CVs for both the third- and fourth-order accurate SV schemes. We can therefore conclude that CVs partitioned with the Gauss–Lobatto points not only produce convergent SV schemes, but also more accurate schemes than equidistant CVs.

As another demonstration of the "spectral" accuracy of the SV schemes, a finite-element-type p-refinement study is also performed. In this study, the number of SVs are fixed, while the order of accuracy is increased in the SVs. Shown in Fig. 9 is the L_1 errors versus the order of accuracy on four SVs. Notice that the logarithm of the L_1 error decreases linearly with the order of accuracy.

Comparing Tables I and II, it is obvious that a second-order SV scheme with 640 degrees of freedom (DOF) is not as accurate as a sixth-order scheme with 30 DOF. If we extrapolate this estimate to three dimensions, then a sixth-order scheme with 1000 DOF will be more accurate than a second-order SV scheme with about $1000(\frac{640}{30})^3 = 9{,}700{,}000$ DOF. With a sixth-order SV scheme, one may never have to use a grid with millions of cells in three

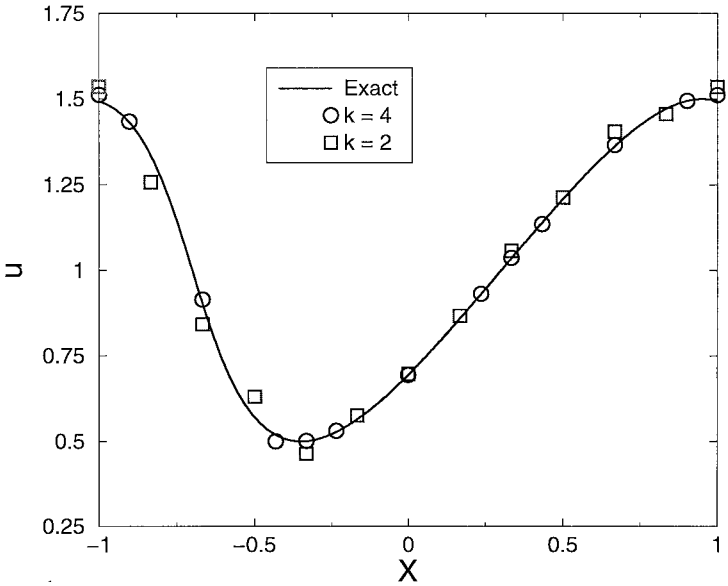

FIG. 14. Computed solutions to the Burger's equation at $t = 0.3$ using second-order and fourth-order SV schemes with 12 degrees of freedom without limiters.

dimensions anymore. The potential of the SV method for 3D problems with complex geometry is tremendous.

In Table III, we show the L_1 and L_∞ errors for the initial condition $u_0(x) = \sin^4(\pi x)$. This is an example for which ENO schemes lose their accuracy [39]. Note that the SV method was able to achieve the formal order of accuracy in both the L_1 and L_∞ norms.

7.2. *Propagation of a Square Wave—Discontinuity Capturing*

In this test, the following square wave on domain [0, 1]

$$u_0(x) = \begin{cases} 1 & \frac{1}{4} \le x \le \frac{3}{4} \\ 0 & \text{otherwise} \end{cases}$$

is propagated at a constant speed of one subject to periodic boundary conditions. In all the following computations, we exclusively use the Gauss–Lobatto points to subdivide CVs. Forty spectral volumes are used in all the computations. First the performance of the SV schemes without limiters are tested. Figure 10 shows the computed solutions at $t = 1$ with SV schemes of various orders of accuracy. The circular symbols indicate the reconstructed solutions at the spectral volume boundaries. Note the obvious oscillations near the jump discontinuities (the "Gibbs" phenomenon). For higher order schemes (order >3), the oscillations are confined to a narrow region near the discontinuities. To study the long-term performance of the SV method, the computed solutions at $t = 50$ for the fifth-, seventh-, and twelfth-order schemes (i.e., after the square wave crosses the computational domain 50 times) are shown in Fig. 11. It is obvious that the fifth-order scheme dissipated the square wave much more than the seventh- and twelfth-order schemes. In fact, the seventh- and twelfth-order schemes preserve the square wave quite well after 50 periods.

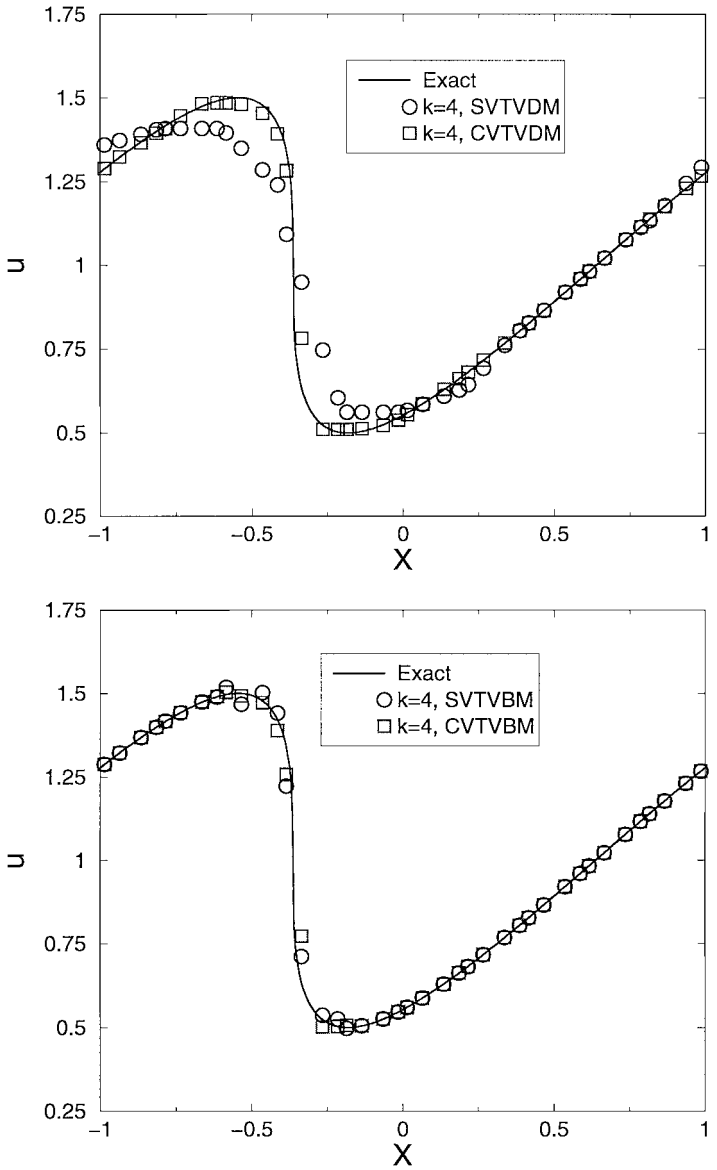

FIG. 15. Computed solutions to the Burger's equation at $t = 2/\pi$ using a fourth-order SV scheme with CVTVDM, SVTVDM, and CVTVBM, SVTVBM limiters on 10 SVs.

Next the performance of the various limiters is evaluated. Figure 12 shows the computed solution at $t = 1$ with SV schemes of various orders of accuracy and the CVTVDM and SVTVDM limiters. Note first that the solutions are indeed nonoscillatory, and the discontinuity was captured with higher resolution using higher order schemes. It is also obvious that the CVTVDM limiter performs better than the SVTVDM limiter when the order of the numerical scheme is higher because of the availability of local data at the CVs.

Finally the seventh- and twelfth-order SV schemes with both CVTVDM and SVTVDM limiters are used to carry out a "long-term" simulation until $t = 50$, when the square wave has crossed the computational domain 50 times. Both the seventh-order and twelfth-order

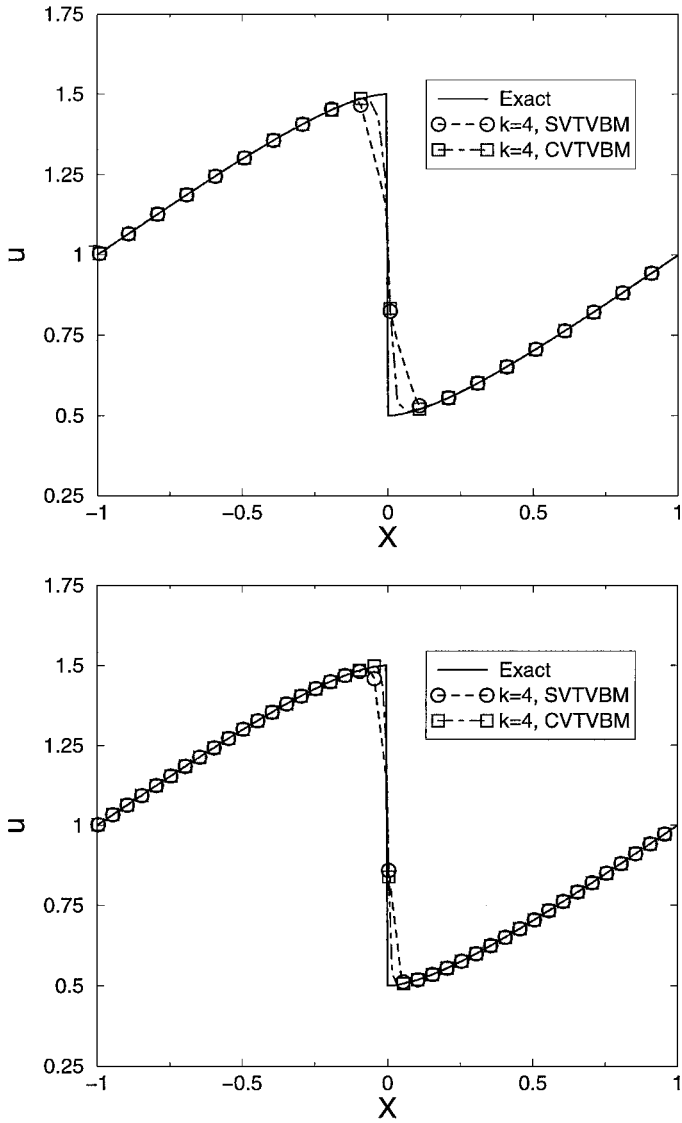

FIG. 16. Computed solutions to the Burger's equations at $t = 1$ with CVTVBM and SVTVBM limiters using 20 and 40 spectral volumes.

schemes did a good job in preserving the resolution of the discontinuity, and the twelfth-order scheme gave a visibly better solution, as shown in Fig. 13. Once again, the CVTVDM limiter yielded a sharper resolution of the discontinuity than the SVTVDM limiter.

7.3. *Test with Burger's Equation*

In this test, we solve the nonlinear Burger's equation with a periodic boundary condition:

$$\frac{\partial u}{\partial t} + \frac{\partial u^2/2}{\partial x} = 0, \quad -1 \le x \le 1$$

$$u(x, 0) = u_0(x) = 1 + \frac{1}{2}\sin(\pi x).$$

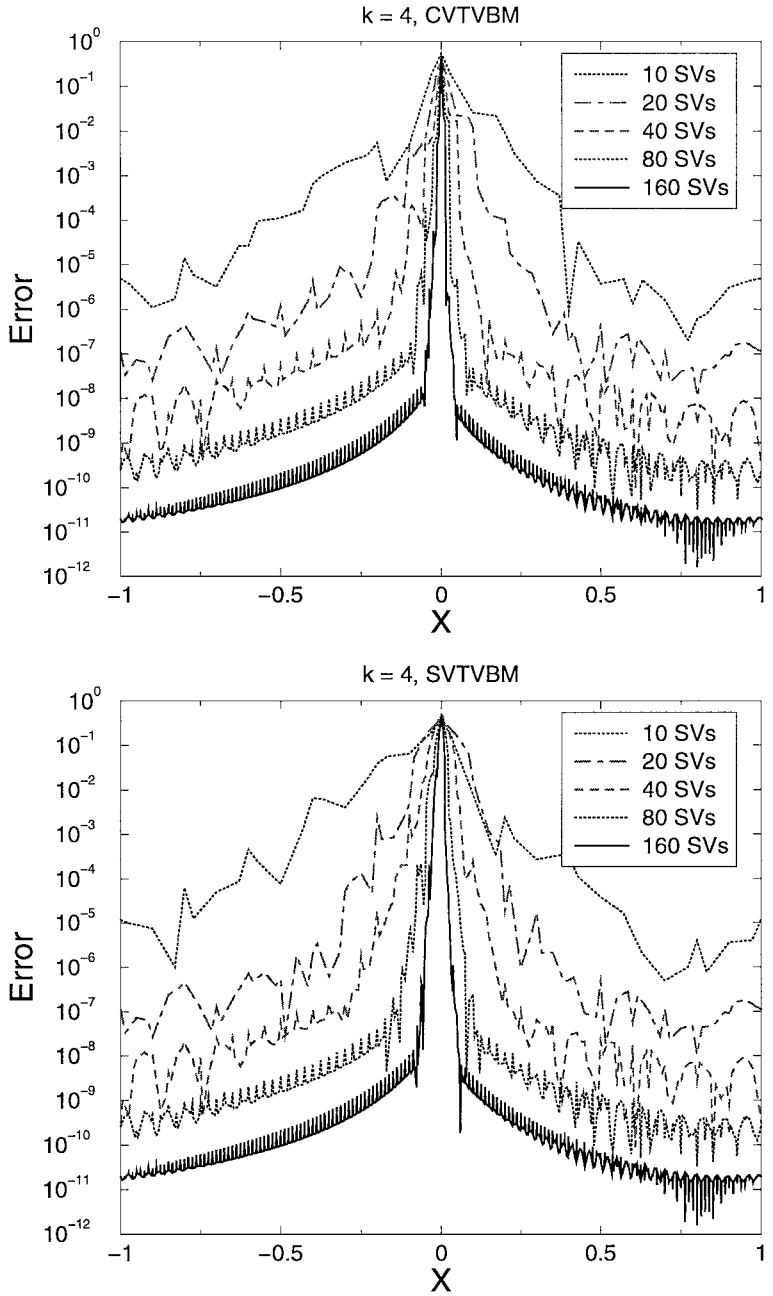

FIG. 17. Local error of computed solutions of the Burger's equation at $t = 1$ with a fourth-order SV scheme and CVTVBM and SVTVBM limiters.

The exact solution is smooth up to $t = 2/\pi$, then it develops a moving shock, which interacts with rarefaction waves. For details, see [20]. At $t = 0.3$, the solution is still smooth. SV schemes from second- to sixth-order of accuracy are tested, and the L_1 and L_∞ errors are listed in Table IV, together with the numerical order of accuracy. Note that the expected formal orders of accuracy for all the tested schemes are achieved in both the L_1 and L_∞ norms. The computed solution with a second-order SV scheme on six SVs is compared

with the solution with a fourth-order SV scheme on three SVs in Fig. 14. The numerical solutions therefore have the same number of degrees of freedom. Note that the fourth-order scheme gave a visibly better solution than the second-order scheme.

At $t = 2/\pi$, a shock starts to form in the solution. The numerical solution would be oscillatory without limiters. Figure 15 displays the computed solutions with a fourth-order SV scheme on 20 SVs using various limiters. Note that the SVTVDM limiter strongly dissipated the numerical solution, while the CVTVDM limiter gave a much better solution. Both TVBM limiters with $M = 20$ gave reasonable results, with the CVTVBM limiter showing a slightly more accurate prediction. The solution with the SVTVBM limiter is slightly oscillatory.

At $t = 1$, a shockwave has formed in the solution. The numerical solutions computed with a fourth-order SV scheme on 20 and 40 SVs using both TVBM limiters are presented in Fig. 16. Note that the shockwave is generally captured in one spectral volume, and the CVTVBM limiter once again produced a solution with a better resolution for the shockwave.

Finally to see whether TVBM limiters affect the solution accuracy away from the shockwave, the local solution errors with a fourth-order SV scheme on different grids are plotted in Fig. 17. It is shown clearly in the figure that the expected order of accuracy (fourth-order) is retained away from the shockwave with both the CVTVBM and SVTVBM limiters. The CVTVBM limiter is shown again to have a better resolution for the shock wave.

CONCLUSIONS

A high-order spectral volume method is developed in this study. The concept of "spectral volume" is introduced to achieve high-order accuracy in a very efficient manner. The SV method is much more efficient in terms of both memory and CPU requirements than a high-order k-exact finite volume method because the reconstruction for a particular grid type is solved analytically and never explicitly carried out. Furthermore, the "reconstruction stencils" are never singular. It is interesting to note that the stability condition for the second-order SV scheme (CFL $\leq 1/2$) is better than the second-order DG scheme (CFL $\leq 1/3$). Numerical tests strongly suggest that the third-order SV scheme is stable for CFL $\leq 1/3$, versus the condition of CFL $\leq 1/5$ in the third-order DG scheme. Control-volume-wise and spectral-volume-wise TVDM and TVBM limiters are developed to remove spurious oscillations near discontinuities. It has been shown that CV-wise limiters perform better than SV-wise limiters. Because of the availability of local data, the SV method is expected to produce sharper discontinuity profiles than the DG method.

Accuracy studies with 1D linear and nonlinear scalar conservation laws have been carried out, and the order of accuracy claim has been numerically verified. It has been found that control volumes generated with Gauss–Lobatto points in a spectral volume not only yield convergent numerical schemes, but also more accurate schemes. The TVBM limiters were found to maintain uniformly high-order accuracy away from discontinuities. The extension of the method to two dimensions and one-dimensional systems is now under way and will be reported in future publications.

ACKNOWLEDGMENTS

The author gratefully acknowledges the start-up funding provided by the Department of Mechanical Engineering, College of Engineering of Michigan State University. He also thanks Dr. Yen Liu of NASA Ames Research Center for helpful discussions.

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 P. O. Frederickson, *High-Order Solution of the Euler Equations on Unstructured Grids using Quadratic Reconstruction*, AIAA Paper No. 90-0013 (1990).
5. F. Bassi and S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, *J. Comput. Phys.* **138**, 251 (1997).
6. J. P. Boris and D. L. Book, Flux-corrected transport I: SHASTA, a fluid transport algorithm that works, *J. Comput. Phys.* **11**, 38 (1973).
7. D. L. Book, J. P. Boris, and K. Hain, Flux-corrected transport II: Generalization of the method, *J. Comput. Phys.* **18**, 248 (1975).
8. A. Burbeau, P. Sagaut, and Ch.-H. Bruneau, A problem independent limiter for high order Runge–Kutta discontinuous Galerkin methods, *J. Comput. Phys.* **169**, 111 (2001).
9. J. Casper and H. L. Atkins, A Finite volume high-order ENO scheme for two-dimensional hyperbolic systems, *J. Comput. Phys.* **106**, 62 (1993).
10. S. R. Chakravarthy and S. Osher, *A New Class of High Accuracy TVD Schemes for Hyperbolic Conservation Laws*, AIAA Paper No. 85-0363 (1985).
11. 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).
12. 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).
13. 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).
14. P. Colella and P. Woodward, The piecewise parabolic method for gas-dynamical simulations, *J. Comput. Phys.* **54**, 174 (1984).
15. M. Delanaye and Y. Liu, *Quadratic Reconstruction Finite Volume Schemes on 3D Arbitrary Unstructured Polyhedral Grids*, AIAA Paper No. 99-3259-CP (1999).
16. O. Friedrich, Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids, *J. Comput. Phys.* **144**, 194 (1998).
17. S. K. Godunov, A finite-difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics, *Mat. Sb.* **47**, 271 (1959).
18. J. B. Goodman and R. J. LeVeque, On the accuracy of stable schemes for 2D scalar conservation laws, *Math. Comput.* **45**, 15 (1985).
19. A. Harten, High resolution schemes for hyperbolic conservation laws, *J. Comput. Phys.* **49**, 357 (1983).
20. A. Harten, B. Engquist, S. Osher, and S. Chakravarthy, Uniformly high order essentially non-oscillatory schemes III, *J. Comput. Phys.* **71**, 231 (1987).
21. A. Harten, ENO schemes with subcell resolution, *J. Comput. Phys.* **83**, 148 (1989).
22. O. Hassan, K. Morgan, and J. Peraire, *An Implicit Finite Element Method for High Speed Flows*, AIAA Paper No. 90-0402 (January 1990).
23. E. Isaacson and H. B. Keller, *Analysis of Numerical Methods* (Dover, New York, 1993).
24. C. Hu and C.-W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, *J. Comput. Phys.* **150**, 97 (1999).
25. G. Jiang and C.-W. Shu, Efficient implementation of weighted ENO schemes, *J. Comput. Phys.* **126**, 202 (1996).
26. Y. Kallinderis, A. Khawaja, and H. McMorris, Hybrid prismatic/tetrahedral grid generation for complex geometries, *AIAA J.* **34**, 291 (1996).

27. D. A. Kopriva, Multidomain spectral solutions of the Euler gas-dynamics equations, *J. Comput. Phys.* **96**, 428 (1991).
28. D. A. Kopriva and J. H. Kolias, A conservative staggered-grid Chebyshev multidomain method for compressible flows, *J. Comput. Phys.* **125**, 244 (1996).
29. M.-S. Liu, Mass flux schemes and connection to shock instability, *J. Comput. Phys.* **160**, 623 (2000).
30. X. D. Liu, S. Osher, and T. Chan, Weighted essentially non-oscillatory schemes, *J. Comput. Phys.* **115**, 2000 (1994).
31. H. Luo, D. Sharov, J. D. Baum, and R. Lohner, *On the Computation of Compressible Turbulent Flows on Unstructured Grids*, AIAA Paper No. 2000-0927 (January 2000).
32. D. J. Mavriplis and A. Jameson, Multigrid solution of the Navier–Stokes equations on triangular meshes, *AIAA J.* **28**, 1415 (1990).
33. S. Osher and S. Chakravarthy, Upwind schemes and boundary conditions with applications to Euler equation in general geometries, *J. Comput. Phys.* **50**, 447 (1983).
34. S. Osher, Riemann solvers, the entropy condition, and difference approximations, *SIAM J. Numer. Anal.* **21**, 217 (1984).
35. A. T. Patera, A Spectral element method for fluid dynamics: Laminar flow in a channel expansion, *J. Comput. Phys.* **54**, 468 (1984).
36. P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* **43**, 357 (1981).
37. P. L. Roe, *Optimum Upwind Advection on a Triangular Mesh*, ICASE Report 90-75 (1990).
38. C.-W. Shu, Total-Variation-Diminishing time discretizations, *SIAM J. Sci. Statist. Comput.* **9**, 1073 (1988).
39. C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, edited by A. Quarteroni, Lecture Notes in Mathematics (Springer-Verlag, Berlin/New York, 1998.), Vol. 1697, p. 325.
40. C.-W. Shu, TVB uniformly high-order schemes for conservation laws, *Math. Comput.* **49**, 105 (1987).
41. T. Sonar, On families of pointwise optimal finite volume ENO approximations, *SIAM J. Numer. Anal.* **35**, No. 6, 2350 (1998).
42. 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).
43. B. van Leer, Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second-order scheme, *J. Comput. Phys.* **14**, 361 (1974).
44. B. van Leer, Towards the ultimate conservative difference scheme V. A second order sequel to Godunov's method, *J. Comput. Phys.* **32**, 101 (1979).
45. B. van Leer, Flux-vector splitting for the Euler equations, *Lect. Notes Phys.* **170**, 507 (1982).
46. V. Venkatakrishnan and D. J. Mavriplis, Implicit solvers for unstructured meshes, *J. Comput. Phys.* **105**, No. 1, 83 (1993).
47. Z. J. Wang and B. E. Richards, High-resolution schemes for steady flow computation, *J. Comput. Phys.* **97**, 53 (1991).
48. Z. J. Wang, A fast flux-splitting for all speed flow, in *Proceedings of Fifteen International Conference on Numerical Methods in Fluid Dynamics* (1997), p. 141.
49. Z. J. Wang and R. F. Chen, *Anisotropic Cartesian Grid Method for Viscous Turbulent Flow*, AIAA Paper No. 2000-0395 (2000).
50. H. C. Yee, Construction of explicit and implicit symmetric TVD schemes and their applications, *J. Comput. Phys.* **68**, 151 (1987).