Ulam Quarterly — Volume 1, Number 3, 1992

Extensions of Lax–Wendroff to Multicolor Schemes

Joseph Kolibal

CASE
University of Florida
3950 RCA Blvd. Suite 5003
Palm Beach Gardens, FL 33410

Introduction

The structure of the Lax-Wendroff cell vertex finite volume method is examined and a family of numerical algorithms based on using explicit, pseudo-unsteady, spatially centered formulations are developed. These are based on a "multicolor" update technique, based on split grids on regular, structured meshes. A comparison is made with the standard Lax–Wendroff algorithm, and applications to obtaining steady state solutions to the equations of inviscid, compressible gas dynamics are presented.

1. The problem

The interest is in introducing algorithmic modifications into the cell vertex finite volume formulation used for solving problems of the form

$$\mathbf{w}_t + \mathbf{f}_x(\mathbf{w}) + \mathbf{g}_y(\mathbf{w}) = \mathbf{0}, \tag{1}$$

where $\mathbf{w}$ is the vector of dependent values, and $\mathbf{f}$, and $\mathbf{g}$ are flux vectors defined on a domain $(x, y) \times t$. This hyperbolic pde models convective transport without viscous damping terms. In this paper a multicolor variation (designated as MC) of the standard cell vertex update based on Lax–Wendroff (LW) is introduced and discussed. This has some interesting advantages, notably: (1) neutral stability, thus additional dissipation is not required to stabilize the scheme; (2) internal grids which are uncoupled to artificial boundaries; (3) larger effective computational stencil, thereby yielding larger effective CFL numbers; (4) a choice of second order corrections; and, (5) it is easily implemented into any standard cell vertex code. On the other hand, its difficulties must be understood. These include: (6) the presence of a transient spurious mode; and, (7) its algorithm cost.

The cell vertex method expresses the the dependence of $\mathbf{w}$ on t in terms of the spatially dependent fluxes by using a Lax-Wendroff type expansion:

$$\begin{aligned}&\delta\mathbf{w}(t+\Delta t)\\ &\quad = -\Delta t(\mathbf{f}_x+\mathbf{g}_y)+\frac{1}{2}\Delta t^2\left[\frac{\partial}{\partial x}A(\mathbf{f}_x+\mathbf{g}_y)+\frac{\partial}{\partial y}B(\mathbf{f}_x+\mathbf{g}_y)\right].\end{aligned} \tag{2}$$

The first term on the right hand side of (2) which is linear in Δt is denoted as the *first order term*, and to the second term which depends on Δt^2 is denoted as the *second order term*. The combination of first order and second order terms in (2) form the basis for the time iterative algorithms considered. The approach will be to consider separately each finite volume mesh associated with each update term, in contrasts with the usual emphasis only on the mesh associated with the dependent variables.

The finite volume method has been successfully applied to the solution of inviscid, compressible flow problems represented by (1). In choosing between the cell vertex and cell volume finite volume methods, motivation is derived from technical arguments which appear to favor the cell vertex method [4, 5, 6], and by its flexibility which opens the possibilities for constructing interesting modifications of the basic scheme. Much of this follows on the work of Morton and Paisley [3], and their investigations of the application of the cell vertex methods to solving the Euler equations.

2. Finite volume formulation

It is desirable to impose to conditions on the development of finite volume methods: (1) the structure and connectivity of the meshes must be arranged so as to assure conservation, and (2) the computational stencil is kept minimal. Taking the computational stencil too large means that the estimates of the partial derivatives become less localized and introducing additional points into the stencil can introduce undesirable spurious solutions.

The finite volume formulation is constructed from the integral form of (1) using

$$\frac{d}{dt}\int_{\Omega_l}\mathbf{w}\,d\Omega=-\int_{\partial\Omega_l}\mathbf{f}(w)dy-\mathbf{g}(w)dx=\boldsymbol{\varphi}_l. \tag{3}$$

In the cell vertex method, the discretization is carried out with respect to the control volume, or cell Ω_l. In computing steady state solutions it is required that $\boldsymbol{\varphi}_l=0$ for each computational cell.

3. Mesh structure

The computational cell $\Omega_l\in\mathcal{D}^h$ is a convex, polygonal region whose boundary contains vertices $i(l)$, $i=1\ldots N_l$ located at $\mathbf{x}_i$. The domain is

subdivided into cells Ω_l such that they form a partition of $\mathcal{D}^h$, that is

$$\bigcup_{l=1}^{N_D} \Omega_l = \mathcal{D}^h \tag{4}$$

constructed such that,

$$\Omega_k \bigcap \Omega_l = \partial\Omega_k \bigcap \partial\Omega_l \qquad \forall\, k \neq l. \tag{5}$$

A further degree of complexity is introduced in the description of the finite volume construction, in that the necessary connectivity (with suitable relaxation at the boundaries) admits unions over disjoint partitions of $\mathcal{D}^h$. That is, families of sets of cells are allowed, each of which satisfy the requirements of (4) and (5). To avoid notational difficulties, the index i is reserved to denote the locations of the dependent variables associated with the cell vertices, and the index l to denote the numbering of the cells in $\mathcal{D}^h$, and hence the variables associated with cell dependent quantities.

The boundary integral in (3) is discretized over the cell Ω_l by applying the trapezium rule along each edge on the boundary of Ω_l. The discrete cell residual $\mathbf{R}_l$ for Ω_l is

$$\mathbf{R}_l = \frac{1}{V_l} \sum_{i=1}^{N_l} \frac{1}{2}(\mathbf{f}_i + \mathbf{f}_{i+1})(y_{i+1} - y_i) - \frac{1}{2}(\mathbf{g}_i + \mathbf{g}_{i+1})(x_{i+1} - x_i), \tag{6}$$

where the index i is taken modulo $N_l + 1$. The notation is used that $\mathbf{f}_i = \mathbf{f}(\mathbf{W}_i)$ where $\mathbf{W}_i$ is the discrete vector of dependent variables corresponding to $\mathbf{w}_i$ at the vertex $\mathbf{x}_i$.

Equation (6) conveniently defines a map $\mathcal{T}_1$ from the dependent variables, $\mathbf{W}_i$, to the cell residual, $\mathbf{R}_l$. There is no completely satisfactory way of relating these residuals back to the dependent variables at the vertices of the cell. There are several ways of closing the loop, that is, of constructing a map $\mathcal{T}_2$ relating the cell dependent values back to the vertices:

- By looking at an ensemble of cells each of which contains a given vertex, the boundary integral, and hence the residual for this ensemble, can be related to the update at this vertex using a linear combination of the cell residuals.
- By arranging the connectivity of the mesh so that the mapping from the cell residual to each vertex is necessarily unique.
- By assigning the residual, or fraction of the residual, to a vertex based on a decomposition following, for example, the characteristic directions.

The first two possibilities are examined, in which case the structure of $\mathcal{T}_2$ depends entirely on the connectivity of the mesh. This will allow the construction of $\mathcal{T} = \mathcal{T}_2 \circ \mathcal{T}_1$, giving the iteration $\mathbf{W}^{n+1} = \mathcal{T}(\mathbf{W}^n)$.

In either case it is natural to define an element $\Omega_{L(i)}$ as the control volume associated with the vertex $\mathbf{x}_i$ and $\mathcal{T}_2$ as the bijection between fluxes into $\Omega_{L(i)}$ and the update at the vertex $\mathbf{x}_i$. In general, in the interior of $\mathcal{D}^h$, $\Omega_{L(i)}$ contains $\mathbf{x}_i$ wholly within its interior, that is $\mathbf{x}_i \in \Omega_{L(i)}$ and $\mathbf{x}_i \notin \partial\Omega_{L(i)}$. This is modified in an obvious way to account for the boundary of $\mathcal{D}^h$. It is useful to define $l(i)$ to be the set of all cell indices l such that $\mathbf{x}_i \in \Omega_l$. The element $\Omega_{L(i)}$ is defined with volume V_L containing $\mathbf{x}_i$ as

$$\Omega_{L(i)} = \bigcup_{l(i)} \Omega_l, \qquad V_L = \sum_{l(i)} V_l. \tag{7}$$

This is the smallest control volume which is centered on $\mathbf{x}_i$.

3.1. Regular meshes

On a regular mesh in the interior of the domain the number of vertices associated with each cell Ω_l is constant, and the number of cells which compose each element Ω_L is constant. On a regular mesh, the vertices may be partitioned into equivalence classes, defined by the cancellation of the fluxes along the boundaries of the elements in the interior of $\mathcal{D}^h$.

For example, on a regular quadrilateral mesh, every vertex can be written in terms of its (i, j) coordinate. Using this the domain can be partitioned into elements whose center vertices are equivalent to the canonical vertices located at (i, j), $(i+1, j)$, $(i, j+1)$ and $(i+1, j+1)$, yielding four equivalence classes. Using parenthetic subscript notation to indicate a particular vertex which belongs to a given equivalence class, that is, $\mathbf{x}_{i,j} \in \mathcal{D}^h_{(\cdot,\cdot)}$ if $i(\mathrm{mod}\ 2) = 0$ and $j(\mathrm{mod}\ 2) = 0$, $\mathbf{x}_{i,j+1} \in \mathcal{D}^h_{(\cdot,+)}$ if $i(mod\,2) = 0$ and $j + 1(mod\,2) = 1$, and so on, the vertices in mesh $\mathcal{G}_1$ can be written as the disjoint union,

$$\mathcal{D}^h_* = \mathcal{D}^h_{(\cdot,\cdot)} \bigcup \mathcal{D}^h_{(+,\cdot)} \bigcup \mathcal{D}^h_{(\cdot,+)} \bigcup \mathcal{D}^h_{(+,+)}. \tag{8}$$

Alternatively, by using sets of overlapping meshes (two families) a finite volume mesh can be constructed in which the element is equivalent with a cell. This mesh is referred to as $\mathcal{G}_2$. On these meshes the vertices from each class lie inside the cells from the other equivalence class. Consequently, except for some difficulties at boundaries, there is a unique cell flux associated with the update of each vertex. The vertices in this mesh can be expressed as the disjoint union

$$\mathcal{D}^h_* = \mathcal{D}^h_{(\cdot,\cdot)} \bigcup \mathcal{D}^h_{(+,+)}. \tag{9}$$

3.2. The update for LW

In the application of the Lax–Wendroff update, the approach is to use $\mathcal{G}_1$ to form a simultaneous linear combination of the cell residuals to update each vertex. This can easily be extended to meshes of the type $\mathcal{G}_2$.

The first order update $\delta\mathbf{W}_l$ for a cell Ω_l is given by

$$\delta\mathbf{W}_l = -\Delta t_l \mathbf{R}_l, \tag{10}$$

where the time Δt_l depends on the cell Ω_l and the local wave speed. By combining the residuals, $\delta\mathbf{W}_i$ is constructed for each cell containing $\mathbf{x}_i$ using volume weighting [1], in which case

$$\delta\mathbf{W}_i^{n+1} = -\frac{1}{V_L}\sum_{l(i)} \delta\mathbf{W}_l^n V_l = -\frac{\Delta t}{V_L}\sum_{l(i)} \mathbf{R}_l^n V_l. \tag{11}$$

To accommodate the second order update in a conservative fashion A subsidiary mesh, $\tilde{\mathcal{D}}^h$, is introduced. The structure of $\tilde{\mathcal{D}}^h$ is dependent on $\mathcal{D}^h$ which is used to construct the first order update.

The second order term is given by the flux of the Jacobians times the fluxes, that is

$$\frac{d}{dt}\int_{\tilde{\Omega}_i} \mathbf{w}_t \, d\Omega = \int_{\partial\tilde{\Omega}_i} A(\mathbf{w})(\mathbf{f}_x + \mathbf{g}_y)dy + B(\mathbf{w})(\mathbf{f}_x + \mathbf{g}_y)dx, \tag{12}$$

where $\tilde{\Omega}_i$ is the control volume containing the vertex $\mathbf{x}_i$. The discrete second order update can be interpreted based on the flux through the boundaries of this element $\tilde{\Omega}_i$ formed by the vertices at $\mathbf{x}_l$ contained in each cell Ω_l. The second order residual $\tilde{\mathbf{R}}_i$ associated with the cell $\tilde{\Omega}_i$ is given by

$$\begin{aligned}\tilde{\mathbf{R}}_i = \frac{1}{V_i}\sum_{l=1}^{N} &\frac{1}{2}(A_l\mathbf{R}_l + A_{l+1}\mathbf{R}_{l+1})(y_{l+1} - y_l) \\ &- \frac{1}{2}(B_l\mathbf{R}_l + B_{l+1}\mathbf{R}_{l+1})(x_{l+1} - x_l),\end{aligned} \tag{13}$$

where A_l and B_l are located at $\mathbf{x}_l$, V_i is the volume of $\tilde{\Omega}_i$ and N is the number of cells surrounding the vertex V_i. Following from (2), the second order correction at time level $n+1$ is then given by $(\Delta t^2/2)\tilde{\mathbf{R}}_i^n$.

3.3. Constructing the update for MC

The LW iteration consists of solving for the fluxes and then combining these fluxes to obtain the update of the dependent variable at each vertex, $\mathbf{x}_i$, that is,

$$\begin{aligned}\delta\mathbf{W}_i^{n+1} &= -\Delta t\left(\mathbf{R}_L^n - \frac{\Delta t}{2}\tilde{\mathbf{R}}_i^n\right) \\ &= -\Delta t\left(\frac{1}{V_L}\sum_{l(i)}\mathbf{R}_l^n V_l - \frac{\Delta t}{2}\tilde{\mathbf{R}}_i^n\right).\end{aligned} \tag{14}$$

Thus each vertex is updated by a linear combination of the residuals occurring over Ω_L and $\tilde{\Omega}_i$. In formulating multicolor variants of LW, a discrete, conservative, nondissipative, stable operator is desired. The dissipation is introduced to achieve improved convergence, rather than as a means of stabilizing the iteration, as in Lax–Wendroff.

3.4. The first order term for MC

As a consequence of splitting the mesh into equivalence classes, successive updating can be accomplished on $\mathcal{G}_1$ if the dependent variables are updated in each equivalence class at a new time step. On $\mathcal{G}_1$:

$$\delta \mathbf{W}_{i+1,j+1}^{n+4} = -\frac{\Delta t}{V_{i+1,j+1}} \sum_{l(i+1,j+1)} V_l \mathbf{R}_l^{n+3} \tag{15}$$

$$\delta \mathbf{W}_{i,j+1}^{n+3} = -\frac{\Delta t}{V_{i,j+1}} \sum_{l(i,j+1)} V_l \mathbf{R}_l^{n+2} \tag{16}$$

$$\delta \mathbf{W}_{i+1,j}^{n+2} = -\frac{\Delta t}{V_{i+1,j}} \sum_{l(i+1,j)} V_l \mathbf{R}_l^{n+1} \tag{17}$$

$$\delta \mathbf{W}_{i,j}^{n+1} = -\frac{\Delta t}{V_{i,j}} \sum_{l(i,j)} V_l \mathbf{R}_l^{n}, \tag{18}$$

where the index $l(i,j)$ indicates the cells which contribute to the update of the vertex at $\mathbf{x}_{i,j}$.

The reason for looking at meshes in terms of contiguous equivalence classes is that conservation can be strictly maintained on each mesh. In solving systems of equations using iterative methods, Jacobi or Gauss–Seidel iteration can be used. In Jacobi iteration, newly updated components of the solution vector are known but not used in computing the remaining components. The new components are introduced at the end of each cycle, hence the updates are done simultaneously. In Gauss–Seidel iteration the updates are done successively as they become known. In solving a problem using explicit time stepping those vertices on which conservation can be maintained, that is, those vertices within an equivalence class, are updated simultaneously, and those in different classes are updates successively.

3.5. The second order term for MC

The explicit introduction of the second order correction is examined. Having evaluated (15)–(18), a second order correction, similar to that in LW, is easily constructed:

$$\delta \mathbf{W}_{i,j}^{n+*} = \frac{\Delta t^2}{V_{i,j}} \tilde{\mathbf{R}}_{i,j}^{*}. \tag{19}$$

where $*$ indicates several possibilities for the time level. Consider, for example, the residual $\mathbf{R}^*_{i,j}$ associated with element $\Omega_{i,j}$, consisting of the Jacobian weighted flux around the vertices $\mathbf{x}_{i+1,j}$, $\mathbf{x}_{i+1,j+1}$, $\mathbf{x}_{i,j+1}$, and $\mathbf{x}_{i,j}$. In computing $\tilde{\mathbf{R}}_{i,j}$, all of the required cell dependent quantities can be evaluated uniformly over the mesh. This is equivalent to introducing a new time level $n+5$ for the second order correction, that is,

$$\begin{aligned}\mathbf{R}^*_{i,j} =& \frac{1}{2}\,[\,((A\mathbf{R})^{n+5}_{i+1,j} - (A\mathbf{R})^{n+5}_{i,j+1})(y_{i+1,j+1} - y_{i,j})\\ &+ ((A\mathbf{R})^{n+5}_{i+1,j+1} - (A\mathbf{R})^{n+5}_{i,j})(y_{i,j+1} - y_{i+1,j})\\ &- ((B\mathbf{R})^{n+5}_{i+1,j} - (B\mathbf{R})^{n+5}_{i,j+1})(x_{i+1,j+1} - x_{i,j})\\ &- ((B\mathbf{R})^{n+5}_{i+1,j+1} - (B\mathbf{R})^{n+5}_{i,j})(x_{i,j+1} - x_{i+1,j})\,]\,.\end{aligned} \tag{20}$$

If, instead, these are used from the previously available time steps, then

$$\begin{aligned}\mathbf{R}^*_{i,j} =& \frac{1}{2}\,[\,((A\mathbf{R})^{n}_{i+1,j} - (A\mathbf{R})^{n+2}_{i,j+1})(y_{i+1,j+1} - y_{i,j})\\ &+ ((A\mathbf{R})^{n+1}_{i+1,j+1} - (A\mathbf{R})^{n+3}_{i,j})(y_{i,j+1} - y_{i+1,j})\\ &- ((B\mathbf{R})^{n}_{i+1,j} - (B\mathbf{R})^{n+2}_{i,j+1})(x_{i+1,j+1} - x_{i,j})\\ &- ((B\mathbf{R})^{n+1}_{i+1,j+1} - (B\mathbf{R})^{n+3}_{i,j})(x_{i,j+1} - x_{i+1,j})\,]\,.\end{aligned} \tag{21}$$

This is significantly more efficient in terms of numerical operations.

3.6 Total update for MC

The total update for MC is formed in the same manner as for LW, except that the first and second order updates occur at different time levels.

There are several further possibilities in applying the second order correction to the first order update. For example, apply several first order corrections before applying the second order term. Denote by c_1, c_2, c_3, c_4 the four convective steps which constitute one complete convective update cycle on a regular two-dimensional mesh.. Denote by d the application of a Jacobian based damping cycle and by s the application of additional smoothing. Then sequences such as,

$$\begin{array}{ll} & \text{Sequence}\\ C^1ds & (c_1,c_2,c_3,c_4),d,s,(c_1,c_2,c_3,c_4),d,s,\ldots\\ C^2ds & (c_1,c_2,c_3,c_4)^2,d,s,(c_1,c_2,c_3,c_4)^2,d,s\ldots\\ \vdots & \\ C^nds & (c_1,c_2,c_3,c_4)^n,d,s,(c_1,c_2,c_3,c_4)^n,d,s,\ldots\end{array} \tag{22}$$

can be constructed; or, even more damped systems such as

$$\begin{array}{ll} C^{\frac{1}{2}}d)s & (c_1,c_2,d,c_3,c_4,d),s\ldots\\ C^{\frac{1}{4}}d)s & (c_1,d,c_2,d,c_3,d,c_4,d),s\ldots\end{array} \tag{23}$$

Indeed, because the damping is so important it is preferable to refer to the damped versions as just MC (i.e, without any qualifiers).

4. One-dimensional analysis of MC

Consider the construction of a discrete solution in one dimension for the convection equation

$$w_t + aw_x = 0,$$

using the finite volume method with successive updating of the dependent variables. Let $x_0, x_1, x_2, \ldots x_N$ be a uniform partition of the domain, $[a, b]$. In one dimension there are two equivalence classes of meshes. Let $0 \leq j \leq N$ be the index of any vertex. Designate by $(\cdot)$ those vertices such that $j(\text{mod } 2) = 0$ and by $(\_)$ those such that $j(\text{mod } 2) = 1$. The minus sign indicates that the shift is to $j-1$ from j. Discretize (24) on the vertices at $(\cdot)$ and $(\_)$ by

$$W^{n+1}_{(\cdot)} - W^{n-1}_{(\cdot)} + \frac{\Delta t}{2}\left(R^{n-\frac{1}{2}}_{(\cdot)-\frac{1}{2}} + R^{n-\frac{1}{2}}_{(\cdot)+\frac{1}{2}}\right) = 0, \tag{25}$$

$$W^{n+2}_{(\_)} - W^{n}_{(\_)} + \frac{\Delta t}{2}\left(R^{n+\frac{1}{2}}_{(\_)+\frac{1}{2}} + R^{n+\frac{1}{2}}_{(\_)-\frac{1}{2}}\right) = 0, \tag{26}$$

and rewrite this using $\nu = a\Delta t/2\Delta x$, as the finite difference formulation

$$W^{n+1}_{(\cdot)} - W^{n-1}_{(\cdot)} + \nu\left(W^{n}_{(\_)+2} - W^{n}_{(\_)}\right) = 0, \tag{27}$$

$$W^{n+2}_{(\_)} - W^{n}_{(\_)} + \nu\left(W^{n+1}_{(\cdot)} - W^{n+1}_{(\cdot)-2}\right) = 0. \tag{28}$$

Since $(\cdot) \pm 1 \equiv (\_)$ and $(\_) \pm 1 \equiv (\cdot)$, an explicit representation of the coupling between equivalence classes is obtained, which is inherent in this system of equations.

4.1. Stability analysis

This is rewritten as a one step scheme. Substituting $W^{n+1}_{(\_)}$ in place of $W^{n+2}_{(\_)}$ and $W^{n}_{(\cdot)}$ in place of $W^{n-1}_{(\cdot)}$ in (27)–(28), A more compact expression for the system of difference equations is obtained,

$$W^{n+1}_{(\cdot)} - W^{n}_{(\cdot)} + \nu\left(W^{n}_{(\_)+2} - W^{n}_{(\_)}\right) = 0, \tag{29}$$

$$W^{n+1}_{(\_)} - W^{n}_{(\_)} + \nu\left(W^{n+1}_{(\cdot)} - W^{n+1}_{(\cdot)-2}\right) = 0. \tag{30}$$

In effect, the two meshes are equated onto the same time level, instead of having a gap at alternate time levels on alternate meshes. This also

amounts to a redefinition of ν in (29)–(30). Defining the shift operator $S_\pm = \exp(\pm i\xi\Delta x)$, then (29)–(30) can be written as

$$\begin{bmatrix} \widehat{W}_{(\cdot)}^{n+1} \\ \widehat{W}_{(-)}^{n+1} \end{bmatrix} = G \begin{bmatrix} \widehat{W}_{(\cdot)}^{n} \\ \widehat{W}_{(-)}^{n} \end{bmatrix}, \tag{31}$$

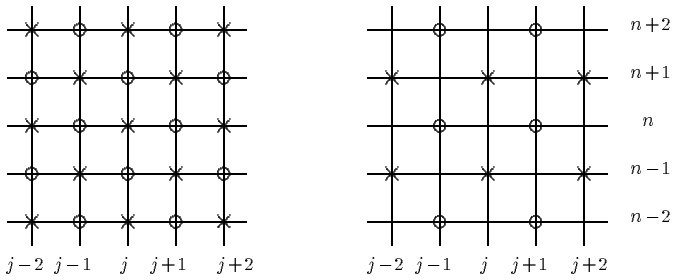
where the amplification matrix G is

$$G = \begin{pmatrix} 1 & -\nu S_+(S_+ - S_-) \\ -\nu S_-(S_+ - S_-) & 1 + \nu^2(S_+ - S_-)^2. \end{pmatrix} \tag{32}$$

Substituting $\exp(\pm i\xi h)$ in place of $S_\pm$ and making use of the usual trigonometric identities

$$G = \begin{pmatrix} 1 & 2\nu\sin(\xi h)[\sin(\xi h) - i\cos(\xi h)] \\ -2\nu\sin(\xi h)[\sin(\xi h) + i\cos(\xi h)] & 1 - 4\nu^2\sin^2(\xi h) \end{pmatrix}. \tag{33}$$

Solving for the roots of the characteristic polynomial of $G - \lambda I$, $|\lambda| = 1$ if $|\nu| \leq 1$. Note that after every node has been updated, two time steps have been taken instead of one. This gives an effective cell based CFL number $\nu' = a\Delta t/\Delta x = 2$.



a) Leapfrog (LF) b) Staggered leapfrog (SLF)

Centered discretizations for the first order update in one dimension

The discretization of the convection equation using the finite volume method obtained by splitting the first order update in one dimension has a spectrum which is recognized to be the same as that of staggered leapfrog method (SLF). Looking at the computational stencil for SLF, expanding the usual half step representation onto full steps, it can be seen that SLF is closely related to the leapfrog method which is also nondissipative. The computational stencil for leapfrog (LF) is given by

$$W_{(\cdot)}^{n+1} - W_{(\cdot)}^{n-1} + \nu\left(W_{(-)+2}^{n} - W_{(-)}^{n}\right) = 0, \tag{34}$$

$$W_{(-)}^{n+1} - W_{(-)}^{n-1} + \nu\left(W_{(\cdot)}^{n} - W_{(\cdot)-2}^{n}\right) = 0. \tag{35}$$

In leapfrog there are two computational grids which which can be colored red ($\circ$) and black ($\times$). At each time level the flux computed on the red grid, that is by differencing across two adjacent red points, yields the update on the red grid and the flux computed on the black grid yields the update on the black grid. Thus in terms of von Neumann stability analysis, there is only one grid to consider. In staggered leapfrog, unlike leapfrog, the red and black grids are directly coupled at alternate time levels. At each time level there is only one grid. The computation of the flux on the red grid determines the update on the black grid, and the computation of the flux on the black grid determines the update on the red grid. Thus the restriction of MC to their one-dimensional analogs on $\mathcal{G}_1$ and $\mathcal{G}_2$ is related to staggered leapfrog.

4.2. MC with damping

Consider the introduction of a dissipative operator into the dispersive scheme introduced in (29)–(30). For the one-dimensional convection problem which is being considered, a familiar smoother is given by the second order term of the Lax-Wendroff update. Since this applies uniformly to each vertex (i.e., no equivalence classes) the parenthetic notation is dropped to obtain

$$W_j^{n+1} - W_j^n - \frac{\Delta t^2}{2}\left(\frac{1}{\Delta x}\left(aR_{j+\frac{1}{2}}^n - aR_{j-\frac{1}{2}}^n\right)\right) = 0. \tag{36}$$

Using the definitions of the cell residual and defining $\gamma = (a\Delta t/\Delta x)^2$, this is written as a two step scheme,

$$W_{(\cdot)}^{n+1} - W_{(\cdot)}^n - \frac{\gamma}{2}\left(W_{(-)+2}^n - 2W_{(\cdot)}^n + W_{(-)}^n\right) = 0, \tag{37}$$

$$W_{(-)}^{n+1} - W_{(-)}^n - \frac{\gamma}{2}\left(W_{(\cdot)}^n - 2W_{(-)}^n + W_{(\cdot)-2}^n\right) = 0. \tag{38}$$

The amplification matrix is obtained for (36),

$$G_s = \begin{pmatrix} 1-\gamma & \frac{\gamma}{2}S_+(S_+ + S_-) \\ \frac{\gamma}{2}S_-(S_+ + S_-) & 1-\gamma \end{pmatrix} \tag{39}$$

Making the substitutions $\exp(\pm i\xi h) = S_\pm$ with $h \equiv \Delta x$, and using suitable trigonometric relationships,

$$G_s = \begin{pmatrix} 1-\gamma & \gamma\cos(\xi h)[\cos(\xi h)+i\sin(\xi h)] \\ \gamma\cos(\xi h)[\cos(\xi h)-i\sin(\xi h)] & 1-\gamma \end{pmatrix}. \tag{40}$$

The eigenvalues of (40) are given by $\lambda = 1-\gamma(1\pm\cos\xi h)$. The intention is to smooth the dispersive SLF update. In the one-dimensional case,

(29)–(30) can be applied followed by an application (36). This amounts to composing G_s with G such that $\widehat{\mathbf{W}}^{n+1} = G_s G \widehat{\mathbf{W}}^n$,

$$\begin{bmatrix} \widehat{W}^{n+1}_{(\cdot)} \\ \widehat{W}^{n+1}_{(-)} \end{bmatrix} = G_s G \begin{bmatrix} \widehat{W}^{n}_{(\cdot)} \\ \widehat{W}^{n}_{(-)} \end{bmatrix} \tag{41}$$

A second order update is done after the first order update has been completed for each equivalence class. Computing $(G_s G)$ explicitly,

$$\begin{aligned}
(G_s G)_{11} &= 1-\gamma-2i\gamma\nu\sin(\xi h)\cos(\xi h), \\
(G_s G)_{12} &= [4\gamma\nu^2\cos^2(\xi h)+2\nu(\gamma-1)+\gamma(1-4\nu^2)]\cos^2(\xi h)+2\nu(1-\gamma) \\
&\quad +i\left(4\gamma\nu^2\cos^2(\xi h)+[2\nu(\gamma-1)+\gamma(1-4\nu^2)]\right)\sin(\xi h)\cos(\xi h), \\
(G_s G)_{21} &= (2\nu(1-\gamma)+\gamma)\cos^2(\xi h)+2\nu(\gamma-1) \\
&\quad -i(2\nu(1-\gamma)+\gamma)\sin(\xi h)\cos(\xi h), \\
(G_s G)_{22} &= (4\nu^2(1-\gamma))\cos^2(\xi h)+(4\nu^2-1)(\gamma-1)-2i\gamma\nu\sin(\xi h)\cos(\xi h).
\end{aligned} \tag{42}$$

Instead of examining the eigenvalues of this matrix which are analytically unwieldy, the eigenvalues of the matrix for a limiting case are constructed. The stability limit on the smoothing step occurs when $\gamma = 1$. Substituting this into (42) gives $(G_s G) =$

$$\begin{pmatrix} -2i\nu\sin(\xi h)\cos(\xi h) & [1-4\nu^2\sin^2(\xi h)]\cos^2(\xi h) \\ & +i[1-4\nu^2\sin^2(\xi h)]\sin(\xi h)\cos(\xi h) \\ \cos^2(\xi h)-\sin(\xi h)\cos(\xi h) & -2i\nu\sin(\xi h)\cos(\xi h) \end{pmatrix} \tag{43}$$

Clearly, it is desirable like to be able to take ν as large as possible. Computing the eigenvalues of (43) gives,

$$\lambda_{\pm} = \pm\cos(\xi h)\sqrt{1-4\nu^2\sin^2(\xi h)} - 2i\nu\sin(\xi h)\cos(\xi h). \tag{44}$$

If the maximum stability limit on the dissipative step is met, the convective step is limited to a CFL number which is less than the maximum permitted for ν.

Comparing (44) as a function of ξh and ν' shows the existence of two regions of growth for some values of ξh if $\nu' > 1.48$ with $\gamma = 1.0$. For $\nu' = 2\nu \leq 1$, the amplification factor given by (44) is given by $|\lambda| = |\cos(\xi h)|$. The change in the amplification factor given by $G_s G$ varies as a function of ν' for values of γ, showing a smooth transition from $\gamma = 0$, which corresponds to no damping, to $\gamma = 1$ which corresponds to maximum damping. If $\gamma > 1$, the amplification factor $|\lambda| > 1$ for all values of ν', hence the scheme is unstable.

There are a number of advantages to doing this smoothing separately from the convection operator as opposed to Lax–Wendroff where because of stability considerations the two are linked together on the same step. In

[8], Sod uses a second order dissipative update after a number of nondissipative leapfrog steps to maintain the coupling between the two grids of the leapfrog method. In MC the grids do not decouple, however damping is required to enhance convergence and to facilitate the removal of transient spurious modes. In any case, the lesson is that the damping and dispersive CFL numbers cannot be taken independently. In addition, the existence of undamped, high frequency modes [2] in two dimensions means that damped MC is a necessity in two dimensions.

4.3. Transient spurious modes

In using SLF to solve the convection equation, the system of equations is modeled by

$$u_t + a v_x = 0, \tag{45}$$

$$v_t + a u_x = 0. \tag{46}$$

This is the wave equation,

$$u_{tt} - a^2 u_{xx} = 0 \tag{47}$$

written as a system of first order differential equations. The two grids in SLF arise in considering the solution of the equation in u and v on two separate grids coupled at alternate time levels. In applying SLF to the convection equation in one dimension, a single equation is discretized in one dependent variable as two finite difference equations in two dependent variables, that is, a dependent variable is associated with each equivalence class on the grid. If (45)–(46) are diagonalized by adding and subtracting the two equations, the system of equations

$$\left(\frac{u+v}{2}\right)_t + a\left(\frac{u+v}{2}\right)_x = 0, \tag{48}$$

$$\left(\frac{u-v}{2}\right)_t - a\left(\frac{u-v}{2}\right)_x = 0. \tag{49}$$

is obtained.

There are two waves traveling at a characteristic velocities of a and $-a$. Thus SLF allows us to correctly approximate the behavior of the solutions of the wave equation. The waves will travel undamped at the correct speed in both directions. However, for the unsteady convection equation, a spurious solution traveling at $-a$ is introduced. Significantly, under suitable restrictions, the steady state solutions to systems of conservation laws using LW or MC are the same. That is,

Theorem 1. *Let* $\mathbf{W}$ *be the steady state solution obtained for the initial boundary value problem* $\mathbf{w}_t + \mathbf{f}_x = \mathbf{b}$ *using MC. If the residuals decouple at steady state (i.e., the net residual in each element is zero because each individual cell residual is zero), then there is no mode associated with spurious eigenvalues; that is, the solution is consistent with that which would be obtained using the LW method.*

Proof. At steady state $\mathbf{W}_{(j)}$ is a fixed point of the iteration since the flux across the cells $(j-1)$ and $(j+1)$ is zero. Similarly, the spurious mode $\mathbf{W}_{(j+1)}$ is a fixed point since the flux across the cells (j) to $(j+2)$ is zero. If the residuals decouple at steady state, it also follows that the flux in the interval (j) to $(j+1)$ is zero. This corresponds to the steady state solution obtained: hence, there is no spurious mode at convergence. □

Note that the boundary conditions play a signifincant role in assuring that the residuals decouple. In one dimension this will always happen [3]. In modeling the one dimensional convection equation, undamped MC admits a spurious mode associated with a wave traveling in the wrong direction. If no additional boundary conditions are imposed on the internal mesh, the solution $u_{(+)} = c$, where c is an arbitrary constant, is admissible on these alternate mesh locations, even if the solution has been found at locations $u_{(\cdot)}$. This spurious mode is of low frequency, having a wavelength equal to the length of the domain. More significantly, it is unstable with respect to perturbations: the solutions $u_{(+)}$ can persist only if the flux on the mesh $u_{(\cdot)}$ vanishes exactly. This is important since the decoupling of the residuals in more than one dimension cannot always be assured, increasing the importance of the numerical viscosity in eliminating these metastable modes.

While the spurious solution can be eliminated by imposing an additional boundary condition, say $u - v = 0$, this can really only be done effectively in the linear, one dimensional case. As it stands the spurious solution introduced by the alternate, internal mesh is unstable and does not affect the steady state result. There is an advantage in not specifying any additional boundary conditions. Except for linear problems, the imposition of correct boundary conditions for hyperbolic systems is difficult, inducing spurious reflections at the boundary. In effect the meshes in MC which are internal to the domain avoid this.

5. One dimensional applications

To obtain a one-dimensional model,the conservation laws are integrated over the cross sectional area S and neglect variations across S. Then the pressure at the nozzle boundary adds a momentum source term pdS/dx to give

$$\mathbf{w}_t + \mathbf{f}_x(\mathbf{w}) = \mathbf{b}(x), \tag{50}$$

where

$$\mathbf{w} = \begin{bmatrix} \rho \\ \rho u \\ \rho e \end{bmatrix}, \mathbf{f}(\mathbf{w}) = \begin{bmatrix} \rho u \\ (p + \rho u^2) \\ u(p + \rho e) \end{bmatrix}, \mathbf{b}(x) = \begin{bmatrix} 0 \\ pS_x/S \end{bmatrix}. \tag{51}$$

having substituted ρ for $S\rho^*$ and p for Sp^*, where ρ^* and p^* are the nondimensionalized density and pressure. This is a model of the Laval nozzle, containing a source term in place of the y component of momentum.

5.1. Discretization

Let $-1 = x_1 < x_2 < \ldots < x_{N-1} < x_N = 1$ define a partition of $\mathcal{D}^h$. The cell vertex finite volume discretization of equations (50)–(51) is straightforward, however, the source term must be taken into account. Defining the residuals as

$$R(\rho)_{j-1/2} = \frac{(\rho u)_j - (\rho u)_{j-1}}{\Delta x_{j-1/2}} \tag{52}$$

$$R(\rho u)_{j-1/2} = \frac{(p + \rho u^2)_j - (p + \rho u^2)_{j-1}}{\Delta x_{j-1/2}} - \frac{S_x}{2}\left(\frac{p_j}{S_j} + \frac{p_{j-1}}{S_{j-1}}\right), \tag{53}$$

$$R(\rho e)_{j-1/2} = \frac{(up + \rho ue)_j - (up + \rho ue)_{j-1}}{\Delta x_{j-1/2}} \tag{54}$$

The first order update is obtained by combining the residuals as

$$-\Delta t \frac{R_{j+1/2}\Delta x_{j+1/2} + R_{j-1/2}\Delta x_{j-1/2}}{\Delta x_{j+1/2} + \Delta x_{j-1/2}} \tag{55}$$

for each component of the system. The second order mesh is obtained by using the partition of $[-1, 1]$ obtained from the cells formed from the midpoints of each first order cell, $(x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}})$. The Jacobian matrices for this system are given by

$$A = \begin{pmatrix} 0 & 1 & 0 \\ \frac{1}{2}(\gamma - 3)u^2 & (3 - \gamma) & \gamma - 1 \\ -\gamma ue + u^3(\gamma - 1) & \gamma e - \frac{3}{2}(\gamma - 1)u^2 & \gamma u, \end{pmatrix} \tag{56}$$

$$b_w = \frac{S_x}{S}\begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{2}(\gamma - 1)u^2 & (1 - \gamma)u & (\gamma - 1) \\ 0 & 0 & 0, \end{pmatrix} \tag{57}$$

where b_w is the Jacobian of the source term.

In a similar fashion for the isenthalpic system, the definition of the residuals given by (52)–(53) with the first order update given again by

(55), is obtained. The Jacobian matrices are instead given by

$$A = \begin{pmatrix} 0 & 1 \\ \frac{1}{\gamma} - \frac{(1+\gamma)}{2\gamma}u^2 & \frac{\gamma+1}{\gamma}u \end{pmatrix}, \tag{58}$$

$$b_w = \frac{S_x}{S}\begin{pmatrix} 0 & 0 \\ \frac{1}{\gamma} + \frac{1}{2\gamma}(\gamma-1)u^2 & \frac{(1-\gamma)}{\gamma}u \end{pmatrix}. \tag{59}$$

Numerically, the Jacobian matrix for the flux is determined at the cell centers, that is at $j-1/2$, while the Jacobian matrix of the source term is evaluated at the nodal values j. Thus the update of the second order term is given by

$$\frac{1}{2}\Delta t^2 \left(\frac{A_{j+1/2}R_{j+1/2} - A_{j-1/2}R_{j-1/2}}{x_{j+1/2} - x_{j-1/2}} + \frac{1}{2}(b_w)_j (R_{j+1/2} + R_{j-1/2}) \right), \tag{60}$$

with the total update given by the sum of the first order and second order update for the Lax–Wendroff method.

In using MC, the update is given by the first order term taken at alternate time steps on the alternate meshes, followed by a second order correction taken after the first order update is completed. In the case of undamped MC there is no second order correction. The CFL number associated with the first order update is denoted ν_1 and that associated with the second order update as ν_2. References to ν are taken to mean that $\nu_1 = \nu_2$. Unless otherwise noted, a Jacobian-type second order correction is used in MC.

5.2. Numerical studies using MC

Computational studies confirm the predications on the stability limits derived in Sec.4.1. The undamped MC remained stable for the Laval nozzle problem to a CFL number of 1.97, consistent with the CFL limit of 2. If damping is included, then as the CFL number ν_1 is increased, stability considerations require that ν_2 be decreased: Consequently for large values of ν_1, the damping, especially as the residuals become small, does emerge as a factor contributing to the convergence rate. The comparison between LW and MC with a Lax–Wendroff type second order update show that the number of iterations required to obtain the same norm of the residual is 1.4 times less. Note that taking $\nu_1 > 1.48$ with $\nu_2 = 0.97$ was found to be the numerical limit of stability, in agreement with the results predicted in Sec.4.2.

In these comparisons, the second order term which is applied is of the same form as the second order correction (60) for LW. The second order correction is applied after one complete cycle of the first order update, that is, after each vertex on each mesh had been updated. Since there are

two such meshes in one dimension, effectively two first order updates are performed on each mesh, followed by the second order update. The use of computationally simpler second order corrections, such as those based on simpler Laplacian type operators, yielded unexceptional results. This can be interpreted in terms of the dual role which the Jacobian matrix in the second order term plays in damping and *upwinding* the fluxes.

Consider the numerical solution of the subsonic Laval nozzle problem using undamped MC, that is, setting $\nu_2 = 0$. Numerical computations confirm that undamped MC yields a high quality, steady state solution for the nozzle problem in which the transient spurious mode dies out as convergence is approached. The Mach number plot for LW is indistinguishable from the same plot using MC, and both agree well with the exact solution. The difference between these methods, as far as the steady state solution is concerned, is discernible only in the entropy deviation which is extremely sensitive to small deviations in the solution.

The entropy deviation is similar using LW or undamped MC; however, there are some high frequency oscillations present in the entropy plot using undamped MC, which are not present using Lax–Wendroff. The magnitude of these oscillations, however, are extremely small compared to the overall entropy deviation. Taking $\nu_2 > 0$ in MC eliminates these oscillations, and brings the entropy curve into agreement with that obtained using LW. These confirm the existence of the transient spurious mode which was predicted in Sec.4.3. Additionally it confirms the prediction that the the two boundary conditions in one dimension assure that no spurious modes can exist at convergence.

The substantial difference between LW and undamped MC is the time required to achieve a reasonable level of convergence. In the case of Lax–Wendroff, to reduce the residual to less than about 10^{-5} requires about 800 iterations on this mesh with a CFL number taken near the maximum, i.e. $\nu = 0.97$. For undamped MC with $\nu_1 = 1.97$ and $\nu_2 = 0$, this same convergence criterion requires over 5300 iterations. The important point is this: damping does have an important role in determining the convergence rates.

The computational costs of MC depend on the implementation. In one dimension on a uniformly spaced mesh, the cancellation of the central node in the central difference scheme means that it suffices to compute the flux, and consequently the residual across the cell from $j-1$ to $j+1$. Thus there is no inefficiency introduced by splitting the computation of the residual. If the residual is not recomputed in order to perform the second order update, there is no loss in efficiency in using split meshes. Fortunately, the convergence rate shows no appreciable degradation in applying this economy.

One of the advantages of MC as compared to LW for steady state solutions of the Euler equations, is the existence of internal meshes which do not require a boundary condition to be applied directly. Consequently, the

internal meshes do not see the boundary errors which are introduced even by a nonreflective boundary treatment. A convincing demonstration of this effect is made if the convergence of LW and MC using nonreflective boundary conditions is compared to using a boundary condition, such as pressure at the outlet [7]. In computer simulations, comparing the effects of using Riemann invariant, nonreflective boundary conditions, the observed convergence rates of LW and MC are the same. Using the pressure boundary conditions, the difference between these two methods becomes apparent, with the convergence of MC being about 1.5 times better than LW. MC can be expected to converge better than LW, if the boundary conditions are a large source of errors.

6. Conclusion

By considering a regular structured grid, a family of updates based on the second order accurate Lax-Wendroff update can be constructed using overlapping grids defined for the first order update. Using one dimensional stability analysis, this produced a neutrally stable algorithm into which a second order update is introduced to improve performance rather than maintain stability.

Numerical studies using the one-dimensional Laval nozzle problem confirm the predicted properties of the algorithm, including the stability limits in the damped and undamped cases. The presence of transient spurious modes is demonstrated in one dimension, and shown to be no problem in either the damped or undamped cases. Indeed, convergence for problems limited by the ability to correctly specify nonreflective boundary conditions show improvement in using a multicolor variant of LW.

References

[1] M. G. Hall, *Cell-vertex multigrid schemes for solution of the Euler equations,* in K. W. Morton and M. J. Baines, editors, *Proceedings of the Conference on Numerical Methods for Fluid Dynamics, University of Reading,* pages 303–345, Oxford, 1985. Oxford University Press.

[2] K. W. Morton and M. F. Paisley, *A finite volume scheme with shock fitting for the steady Euler equations,* Number 87/6, Oxford University Computing Laboratory, Oxford, July 1987.

[3] K. W. Morton and M. F. Paisley, *A finite volume scheme with shock fitting for the steady Euler equations,* Journal of Computational Physics 80 (1): 168–203, January 1989.

[4] M. F. Paisley, *A comparison of cell centre and cell vertex schemes for the steady Euler equtions,* Number 86/1, Oxford University Computing Laboratory, Oxford, 1986.

[5] M. F. Paisley, *Finite Volume Methods for the Steady Euler Equations,* Ph.D. Thesis, University of Oxford, Oxford, September 1986.

[6] C. Rossow, *Comparison of cell centered and cell vertex finite volume schemes, in M. Deville, editor, Proceedings of the Seventh GAMM-Conference on Numerical Methods in Fluid Mechanics,* pages 327–334, Braunschweig, 1988. Freid. Vieweg & Sohn. Notes on Numerical Fluid Mechanics, Vol. 20.

[7] D. H. Rudy and J. C. Strikwerda, *A nonreflecting outflow boundary condition for subsonic Navier-Stokes calcualtions,* Journal of Computational Physics, 36: 53–70, 1980.

[8] G. A. Sod, *Numerical Methods in Fluid Dynamics,* Cambridge University Press, Cambridge, 1985.