

# Active Flux Schemes for Systems

Timothy A. Eymann\*

*DoD HPCMP/CREATE Kestrel Team, Eglin AFB, FL 32542*

Philip L. Roe<sup>†</sup>

*Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, 48109*

**We introduce a new formulation of active flux schemes and extend the idea to systems of hyperbolic equations. Active flux schemes treat the edge values, and hence the fluxes, as independent variables, doubling the degrees of freedom available to describe the solution without enlarging the stencil. Schemes up to third order accurate are explored. The limiter employed uses solution characteristics to set the bounds for the edge updates. The process reduces to simply accessing the solution history from memory and ensuring that the updates stay within the bounded range. The limited edge values are then used to construct the fluxes that conservatively update the centroid value. Using data that most closely follows the solution characteristics allows the limiter to better maintain true extrema in the solution. The scheme is used to generate 1-D solutions for the linear advection, Burgers', and Euler equations.**

## I. Introduction

SECOND order computational methods commonly employed in production codes, while sufficient for many applications, lack the accuracy required to perform certain simulations such as those that require high-fidelity vortex tracking and computational aeroacoustics. One method for increasing the order of accuracy of a finite-volume scheme is to expand the computational stencil, generating higher-order reconstructions from which the solution may be interpolated. An early third order scheme using this idea was developed by Warming, Kutler, and Lomax<sup>1</sup> (WKL). Later, essentially non-oscillatory<sup>2</sup> (ENO) and weighted essentially non-oscillatory<sup>3</sup> (WENO) schemes were developed to obtain higher-order accuracy without generating spurious oscillations in the solution. WENO and ENO schemes perform the higher-order reconstructions by searching through a set of candidate stencils and either selecting the smoothest reconstruction or using a weighted average of the possible choices. This approach is most straightforward to apply on a structured mesh, but it is also applicable to unstructured meshes.<sup>4,5</sup> One drawback of increasing accuracy by enlarging the computational stencil is that the solver attempts to build a high order reconstruction using data from regions of the flow that may not be physically relevant. Another is that synchronizing large stencils during parallel computations requires extremely careful implementation to minimize the communication overhead between processes.

The combined need for higher accuracy and increased parallelization has driven massive interest in methods that achieve higher-order accuracy by increasing the internal degrees of freedom within a cell. Popular methods in this class include discontinuous Galerkin<sup>6</sup> (DG), spectral difference<sup>7</sup> (SD), and spectral volume methods<sup>8</sup> (SV). These schemes are well suited for parallel computations on unstructured meshes since they all are capable of generating higher-order reconstructions without extending the computational stencil beyond the neighboring cells. While a great deal of research has been concentrated on these schemes, they still suffer from two primary issues: a lack of robustness<sup>9</sup>/sufficient limiters for nonlinear problems<sup>10</sup> and a high memory requirement.

The issue of generating higher-order solutions from compact stencils was recognized and addressed very early by van Leer.<sup>11</sup> In his 1977 paper, he presented a series of linear and nonlinear advection schemes that evolve not only the cell-average value but also a solution gradient or edge value. He then used the extra degrees of freedom to increase the accuracy of the solution. Other novel attempts at increasing the accuracy of schemes focused not on the formal order, but on reducing the dissipation through a careful representation and interpolation of data on a space-time mesh. Iserles<sup>12</sup> and Roe<sup>13</sup> investigated Upwind Leapfrog (ULF) schemes, which are two-level schemes with zero dissipation due to their symmetric stencils in space and time. The CABARET scheme presented by Karabasov<sup>14</sup> collapses the

\*Research Engineer, Ph.D. Candidate, Member AIAA

<sup>†</sup>Professor, Fellow AIAA

two-level ULF to a single-step scheme by independently updating cell-averages and flux values. The scheme retains all the advantages of the ULF scheme while only using data from one time level.

Active flux schemes encompass some of the advection schemes proposed by van Leer but also use information from the larger space-time mesh to limit spurious oscillations. With these schemes, we attempt to retain the compactness of DG/SD/SV schemes, while lowering the memory overhead by basing the internal cell reconstructions on shared data. The schemes have great potential for the solution of hyperbolic conservation laws. The next section of this paper introduces the main concepts of active flux schemes by concentrating on their application to scalar equations. After introducing families of second and third order schemes, we discuss the limiting strategy before moving on to the extension of the scheme to systems of equations. Finally, key properties of the scheme and limiter are highlighted by presenting solutions from the one-dimensional linear advection equation, Burgers' equation, and the Euler equations.

## II. Scalar active fluxes

Begin by interpreting the data contained within a given cell as a quadratic function defined by Eq. (1), where  $x_j$  is the cell centroid coordinate for cell  $j$ .

$$u(x) = \bar{u}_j^n + \bar{s}_j \left( \frac{x - x_j}{\Delta x} \right) + \bar{c}_j \left[ \left( \frac{x - x_j}{\Delta x} \right)^2 - \frac{1}{12} \right] \quad (1)$$

This function is a combination of Legendre polynomials, ensuring that we recover the cell-average  $\bar{u}_j$  when  $u(x)$  is integrated over each cell and divided by the cell width  $\Delta x$ . Since the terms are orthogonal, each may be multiplied by an independent coefficient. We interpret the coefficients  $\bar{s}_j$  and  $\bar{c}_j$  as the average slope and curvature, which are functions of the vertex values  $u_{j\pm\frac{1}{2}}$ , cell-average, and Courant number. The slope and curvature definitions determine the reconstruction shape and hence the behavior of the scheme.

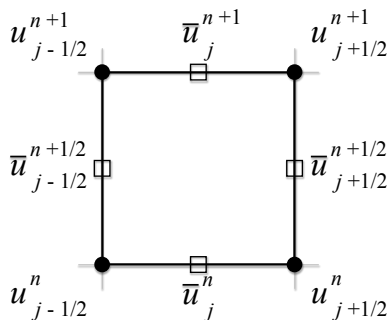


Figure 1. Storage structure for 1-D element

One of the advantages of active flux schemes is that they store the data required to define and update a given reconstruction very economically. Figure 1 illustrates the data storage for a 1-D element. Cell-averages are stored at the centroid of each cell, edge-averages at the face centroids and the vertices hold reconstruction point values. Because cells share data between faces and vertices, the scheme is able to achieve savings over other higher-order approaches. These savings become even more apparent in higher dimensions. In one dimension, active flux schemes require only two degrees of freedom per cell for third order accuracy, but they only require approximately 3.5 degrees of freedom per cell for the same level of accuracy in either two or three dimensions.

### II.A. Linear equations

The key concept of active flux schemes is that the fluxes at the cell edges are evolved as part of the solution rather than simply being calculated from the surrounding data. This does not significantly add to the cost since some flux calculation is required with any choice of scheme. We define the active flux system by specifying an evolution equation for both the vertex value and the edge-average value. The edge-average  $\bar{u}_{j+\frac{1}{2}}^{n+\frac{1}{2}}$  is a time average of the velocity at the interface and is therefore directly related to the flux.

Consider a linear advection case with positive wave speed  $a$ . As illustrated in Fig. 2, the new vertex value is the reconstruction value at a point  $\nu\Delta x$  upwind from the edge, and the edge-average is proportional to the integral of the

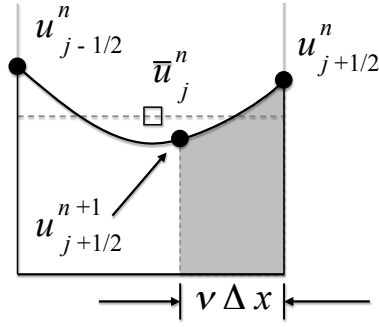


Figure 2. Update quantities

reconstruction function over this interval. The evolution equations as a function of the slope, curvature, and Courant number are:

$$\begin{aligned}
 u_{j+\frac{1}{2}}^{n+1} &= \bar{u}_j^n + \bar{s}_j \frac{1}{2} (1 - 2\nu) + \bar{c}_j \left[ \frac{1}{6} - \nu(1 - \nu) \right] \\
 \bar{u}_{j+\frac{1}{2}}^{n+1} &= \bar{u}_j^n + \bar{s}_j \frac{1}{2} (1 - \nu) + \bar{c}_j \frac{1}{6} (1 - \nu) (1 - 2\nu)
 \end{aligned} \tag{2}$$

The edge-averages can then be used to conservatively update the centroid value, using a standard time-integration scheme such as the Euler explicit form below.

$$\bar{u}_j^{n+1} = \bar{u}_j^n - \frac{\Delta t}{\Delta x} \left[ f \left( \bar{u}_{j+\frac{1}{2}}^{n+1} \right) - f \left( \bar{u}_{j-\frac{1}{2}}^{n+1} \right) \right] \tag{3}$$

Parameterizing the equations using differences between the vertex values and cell-average values greatly simplifies the identification of second and third order accurate schemes within the active flux family. The state differences can be considered fluctuations which generate signals that travel to a target interface. This is a similar approach to Roe's fluctuation-signal approach,<sup>15</sup> but here, the fluctuations are calculated on half-cell intervals rather than between centroids. If we assume the wave speed is positive everywhere, the target interface is the downwind node  $j + \frac{1}{2}$ . The subscripts of the parameterization coefficients indicate whether the fluctuation is adjacent to the target node (near) or a half-cell width away (far). Equation (2) and Eq. (4) can be compared to derive expressions for the slope and curvature as a function of the signal coefficients.

$$\begin{aligned}
 u_{j+\frac{1}{2}}^{n+1} &= u_{j+\frac{1}{2}}^n - \frac{\Delta t}{\Delta x} \left[ a\eta_{\text{far}} \left( \bar{u}_j^n - u_{j-\frac{1}{2}}^n \right) + a\eta_{\text{near}} \left( u_{j+\frac{1}{2}}^n - \bar{u}_j^n \right) \right] \\
 \bar{u}_{j+\frac{1}{2}}^{n+1} &= u_{j+\frac{1}{2}}^n - \frac{\Delta t}{\Delta x} \left[ a\phi_{\text{far}} \left( \bar{u}_j^n - u_{j-\frac{1}{2}}^n \right) + a\phi_{\text{near}} \left( u_{j+\frac{1}{2}}^n - \bar{u}_j^n \right) \right]
 \end{aligned} \tag{4}$$

The updates defined by Eq. (4) are exact by construction for constant data. A family of second order schemes can be defined by requiring that the expressions in Eq. (4) are exact when the cell contains linear data. This is equivalent to specifying  $u(x) = x$ , leading to:

$$\begin{aligned}
 u_{j+\frac{1}{2}}^{n+1} &= -\nu \Delta x \\
 \bar{u}_{j+\frac{1}{2}}^{n+1} &= -\frac{1}{2} (\nu \Delta x)
 \end{aligned} \tag{5}$$

where the expressions have been simplified by transforming to a coordinate frame in which  $x_{j+1/2} = 0$ . After substitution into Eq. (4), we arrive at the coefficient definitions for a family of second order schemes.

$$\begin{aligned}
 \phi_{\text{far}} + \phi_{\text{near}} &= 1 \\
 \eta_{\text{far}} + \eta_{\text{near}} &= 2
 \end{aligned} \tag{6}$$

Third order schemes are obtained by repeating the procedure for quadratic cell data  $u(x) = x^2$ . For third and higher order schemes, it is important to be mindful that the fluctuation-signal formulation contains cell-averages, not the

reconstruction value at the cell centroid. The vertex update and edge-average with a quadratic internal reconstruction are:

$$\begin{aligned} u_{j+\frac{1}{2}}^{n+1} &= (\nu \Delta x)^2 \\ \bar{u}_{j+\frac{1}{2}}^{n+\frac{1}{2}} &= \frac{1}{3} (\nu \Delta x)^2 \end{aligned} \quad (7)$$

resulting in the signal coefficients valid for a family of third order schemes.

$$\begin{aligned} 2\phi_{\text{far}} + \phi_{\text{near}} &= \nu \\ 2\eta_{\text{far}} + \eta_{\text{near}} &= 3\nu \end{aligned} \quad (8)$$

There are a variety of methods that can be used to determine the signal coefficients  $\eta$  and  $\phi$ , some of which were discussed in a previous paper by the authors.<sup>16</sup> By solving the system defined by Eq. (6) and Eq. (8) we obtain the third order scheme introduced by van Leer<sup>11</sup> as Scheme V. Its signal coefficients are listed in Table 1 along with those for the first order upwind (FUP) scheme. Scheme V contains many desirable properties that distinguish it from other members of the active flux family we have investigated. The individual reconstruction within each cell has a clear physical interpretation and is  $C^0$  continuous with reconstructions from neighboring cells. Additionally, the scheme has a stability range from  $0 \leq \nu \leq 1$ , quickly damps spurious modes, and has relatively simple expressions for the signal, slope, and curvature coefficients. For these reasons, much of our testing has centered on Scheme V.

**Table 1. Selected signal coefficients**

	<b>FUP</b>	<b>Scheme V</b>
$\eta_{\text{near}}$	$1/ \nu $	$4 - 3 \nu $
$\eta_{\text{far}}$	0	$3 \nu  - 2$
$\phi_{\text{near}}$	$1/ \nu $	$2 -  \nu $
$\phi_{\text{far}}$	0	$ \nu  - 1$

## II.B. Nonlinear equations

For simplicity, the discussion above assumes a constant, positive wave speed but it is not difficult to formulate the scheme to work in more general cases. When the wave speed is negative, the signal coefficients should be defined as functions of the absolute value of  $\nu$ . The signal target changes with the sign of the wave speed, so if  $a$  is negative everywhere, the fluctuation  $(u_{j+\frac{1}{2}}^n - \bar{u}_j^n)$  must now be multiplied by the “far” values and  $(\bar{u}_j^n - u_{j-\frac{1}{2}}^n)$  by the “near” values. Active fluxes may also be used for nonlinear equations, where the wave speeds can be a mixture of positive and negative values, by simply linearizing the characteristic speed within each cell. This could be done in various ways; at present we are using the linearized speed  $\tilde{\lambda}$ , defined in Eq. (9), to replace constant speed  $a$  in Eq. (4). In cases such as Burgers’ equation, the average wave speed expression may be simplified to avoid division by zero in constant regions of the flow.

$$\tilde{\lambda}_j = \frac{f(u_{j+\frac{1}{2}}^n) - f(u_{j-\frac{1}{2}}^n)}{u_{j+\frac{1}{2}}^n - u_{j-\frac{1}{2}}^n} = \frac{1}{2} (u_{j+\frac{1}{2}}^n + u_{j-\frac{1}{2}}^n) \quad (9)$$

The vertex values and edge-averages are updated by sweeping through each cell, sending the fluctuations to  $j + \frac{1}{2}$  if  $\tilde{\lambda} \geq 0$  and  $j - \frac{1}{2}$  when  $\tilde{\lambda} < 0$ . After all the signals traveling to each node have been accounted for, the interface fluxes can be calculated from the appropriate analytical flux function and the cell centroid updated according to Eq. (3).

## II.C. Limiting

Godunov showed that any linear scheme capable of second or higher order accuracy will generate oscillations.<sup>17</sup> Traditional limiting approaches introduce nonlinearity by examining spatial neighbors and performing some type of activity such as reducing the local gradient. A form of nonlinearity that we have found to be effective examines near

neighbors in both space and time. The strategy makes use of a characteristic coordinate concept illustrated in Fig. 3. If one chooses the update point as the origin, the coordinates of each point in the characteristic frame  $x - \tilde{\lambda}t$  are the functions of  $\nu$  shown in the figure.

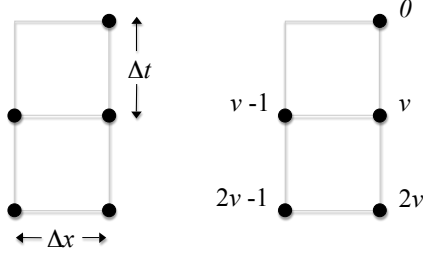


Figure 3. Characteristic coordinates

Once again, momentarily assume  $\tilde{\lambda}$  is positive. Begin by finding those points in the stencil that are closest to the update point  $(j + \frac{1}{2}, n + 1)$  in the characteristic coordinate frame. When  $\nu < \frac{1}{2}$  these are the points at  $(j + \frac{1}{2}, n)$  and  $(j - \frac{1}{2}, n - 1)$ . When  $\nu > \frac{1}{2}$ , the characteristic neighbors are  $(j - \frac{1}{2}, n)$  and  $(j - \frac{1}{2}, n - 1)$ . If the edge update at  $n + 1$  is within the bounds of the appropriate set of characteristic neighbors, it is accepted, otherwise the value closest to this interval is used. By using the value at  $(j - \frac{1}{2}, n - 1)$  rather than the value at  $(j - \frac{1}{2}, n)$ , the limiter is made much milder in its operation. The limiter ensures a version of the TVD property for the nodal updates, but does not strictly enforce it upon the cell-averages. Thus, the method tends to avoid the excessive clipping of legitimate extrema in the solution. The expression in Eq. (10) ensures that the node update does not exceed the maximum of its characteristic neighbors. Similar expressions keep the vertex update above the minimum of the characteristic neighbors.

$$u^* = \begin{cases} \min \left[ u_{j+\frac{1}{2}}^{n+1}, \max \left( u_{j+\frac{1}{2}}^n, u_{j-\frac{1}{2}}^{n-1} \right) \right] & \text{if } 0 \leq \nu \leq \frac{1}{2} \\ \min \left[ u_{j+\frac{1}{2}}^{n+1}, \max \left( u_{j-\frac{1}{2}}^n, u_{j-\frac{1}{2}}^{n-1} \right) \right] & \text{if } \nu > \frac{1}{2} \end{cases} \quad (10)$$

When one relaxes the requirement that the wave speed is positive, the terms in Eq. (10) are simply mirrored across  $j + \frac{1}{2}$ . For linear advection cases, a straightforward implementation of the limiting strategy applied to the vertex and edge-average values sufficiently damped the oscillations. Nonlinear equations introduce options to the strategy that we are currently exploring. For instance, we determine the primary characteristic direction based average of the left and right cell-average speeds, but other methods are certainly possible. Tests have shown that a limiting strategy that produces good results with linear equations does not necessarily work for nonlinear problems. This is a new approach to limiting and we acknowledge that there are still a number of details left to work out.

### III. Active fluxes applied to systems

As with the scalar case, we begin with the simple, linear case and extend the ideas to the more general nonlinear system. The governing equations can be written compactly as Eq. (11).

$$\mathbf{u}_t + \mathbf{A}\mathbf{u}_x = 0 \quad (11)$$

Diagonalizing the Jacobian matrix  $\mathbf{A} = \mathbf{R}\mathbf{\Lambda}\mathbf{R}^{-1} = \mathbf{R}\mathbf{\Lambda}\mathbf{L}$  leads to a system of independent, linear advection equations in the characteristic variables  $\mathbf{w} = \mathbf{L}\mathbf{u}$ .

$$\mathbf{w}_t + \mathbf{\Lambda}\mathbf{w}_x = 0 \quad (12)$$

Equation (4) can be recast as:

$$\begin{aligned} \mathbf{u}_{j+\frac{1}{2}}^{n+1} &= \mathbf{u}_{j+\frac{1}{2}}^n - \frac{\Delta t}{\Delta x} (\delta\mathbf{u}_R + \delta\mathbf{u}_L) \\ \bar{\mathbf{u}}_{j+\frac{1}{2}}^{n+\frac{1}{2}} &= \mathbf{u}_{j+\frac{1}{2}}^n - \frac{\Delta t}{\Delta x} (\delta\bar{\mathbf{u}}_R + \delta\bar{\mathbf{u}}_L) \end{aligned} \quad (13)$$

where the fluctuation term  $\delta\mathbf{u}_{L,R}$  or  $\delta\bar{\mathbf{u}}_{L,R}$  is decomposed as a sum over each of the  $k$  characteristic fields. For example,

$$\begin{aligned}\delta\mathbf{u}_R &= \sum_{i=1}^k \eta\lambda_i \delta w_R^{(i)} \mathbf{r}_i \\ &= \sum_{i=1}^k \eta\lambda_i \left( \mathbf{w}_{j+\frac{1}{2}}^n - \bar{\mathbf{w}}_j^n \right)_i \mathbf{r}_i \\ &= \sum_{i=1}^k \eta\lambda_i \ell_i \left( \mathbf{u}_{j+\frac{1}{2}}^n - \bar{\mathbf{u}}_j^n \right) \mathbf{r}_i\end{aligned}\tag{14}$$

The product of the row vector  $\ell_i$  with the state difference is the strength of the  $i^{\text{th}}$  wave generated by the fluctuation. To update the interface values, the wave strength is modified by the appropriate signal coefficient, chosen based on the sign of  $\lambda_i$ , and the quantity from Eq. (14) is sent to the target interface. Once the average state at each interface is determined, the values can be used in the analytical flux function associated with the governing equations, just as in the scalar case.

$$\mathbf{F}_{j+\frac{1}{2}} = \mathbf{R}(\Lambda\bar{\mathbf{w}}_{j+\frac{1}{2}}^{n+\frac{1}{2}}) = \mathbf{R}(\Lambda\mathbf{L}\bar{\mathbf{u}}_{j+\frac{1}{2}}^{n+\frac{1}{2}}) = \mathbf{A}\bar{\mathbf{u}}_{j+\frac{1}{2}}^{n+\frac{1}{2}} = f(\bar{\mathbf{u}}_{j+\frac{1}{2}}^{n+\frac{1}{2}})\tag{15}$$

Nonlinear systems may be computed by linearizing about an average state. A natural choice is to use Roe's linearization<sup>18</sup> within each cell to find an average state in the cell, which can then be used to calculate a local approximation for the eigenvectors and characteristic speeds (eigenvalues). We currently use vertex values for the linearization, but once again there are other valid options. Once a linearization choice has been made, the limiter from section II.C may be applied to the system provided the bounds checks are performed one field at a time using characteristic variables, not conserved or primitive variables.

## IV. Performance

### IV.A. Linear scalar performance

#### IV.A.1. No limiter, randomized grid

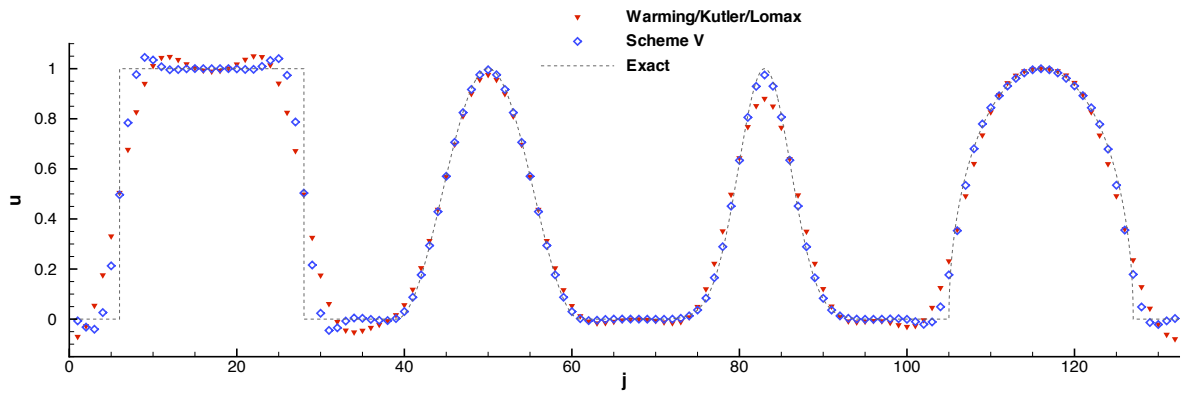
We tested Scheme V on the linear advection equation with a suite of waveforms selected by Zaleszak<sup>19</sup> and others. The cases were run using a mesh of 132 equally spaced cells as well as a mesh with randomized cell spacing. Figure 4 compares Scheme V to the standard WKL cubic-interpolating scheme. The more compact spatial stencil of Scheme V improves the accuracy compared to the cubic-interpolating scheme. On both the uniform and random meshes, Scheme V performs noticeably better on the narrow peak, and generally produces a solution closer to the exact solution. Table 2 highlights Scheme V's accuracy when computing on a random mesh. Scheme V loses only 6.6% of the Gaussian peak after one cycle through the mesh, while WKL reduces the peak by 16.4%. On the smooth cosine curve, Scheme V drops the peak value by just 1.5% compared to a 4.7% loss with the WKL scheme.

Table 2. Peak values without limiting

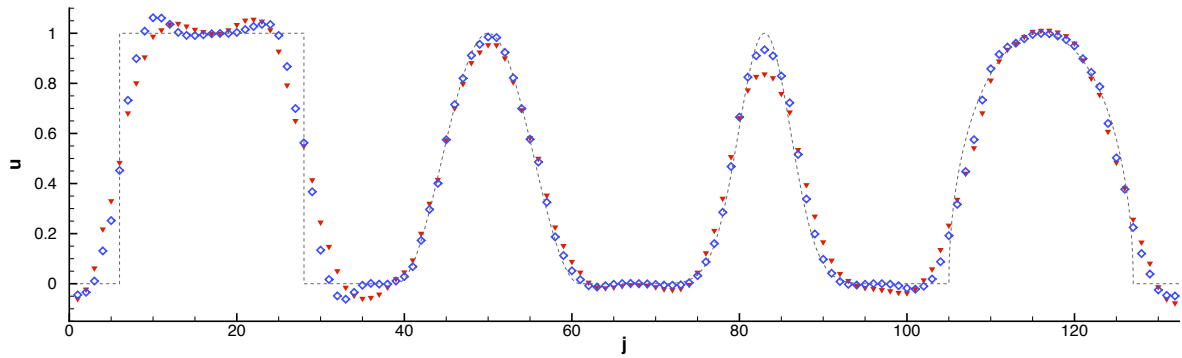
(a) Uniform mesh			(b) Random mesh		
Wave	WKL	Scheme V	Wave	WKL	Scheme V
Square	1.050	1.040	Square	1.055	1.062
Cosine	0.9756	0.9947	Cosine	0.9532	0.9850
Gaussian	0.8801	0.9745	Gaussian	0.8365	0.9340
Ellipse	1.001	1.000	Ellipse	1.011	0.9994

#### IV.A.2. Performance with limiting

Figure 5 compares a limited solution, an unlimited solution and a Lax-Wendroff (LxW) solution that employed the superbee limiter. The results for the limited LxW scheme and Scheme V are comparable for the square wave, but for



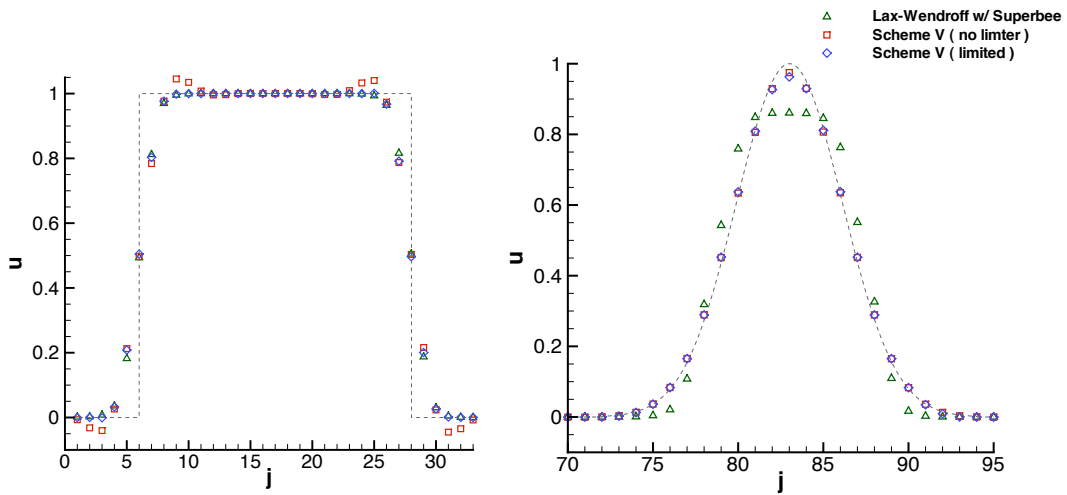
(a) Uniform mesh,  $\nu = 0.44$



(b) Random mesh

Figure 4. Performance of third order schemes with compact and non-compact stencil

the Gaussian profile, the characteristic-based limiter does not suffer from either the excessive clipping or the squaring of the waveform caused by the compressive superbee limiter. Table 3 summarizes the peak values predicted by Scheme V on a uniform mesh using the limiter. The values show that the limiter preserves true extrema in the solution since only the Gaussian peak is reduced by more than 1%. Even this is not significantly higher than the reduction of the peak seen in the unlimited solution (3.7% reduction with limiting versus 2.5% without).



(a) Square wave

(b) Gaussian

Figure 5. Limiter comparison after one cycle;  $\nu = 0.44$

Table 3. Scheme V peak values with limiting

Wave	Scheme V (limited)
Square	1.001
Cosine	0.9903
Gaussian	0.9628
Ellipse	0.9987

#### IV.B. Nonlinear scalar performance

We use Burgers' equation to demonstrate the scheme's performance on nonlinear equations. Scheme V was tested on two cases; a Gaussian pulse and a centered compression wave. The Gaussian pulse starts with the following initial data:

$$u_0(x) = \frac{1}{20} + \frac{19}{20}e^{-50(x-\frac{1}{2})^2} \quad (16)$$

Straightforward analysis shows that a shock forms at time  $t_s = 0.174$  and position  $x_s = 0.709$ . The initial speed of the shock is  $u_s = 0.63$ , but it slows as it interacts with the expansion. Our previous implementation of Scheme V used a fluctuation-signal approach based around the cell-average value and a linearization that included values from neighboring cells. This performed well up to the shock formation, but after that, the scheme began producing excessive overshoots in the conserved variables. A shock detector and explicit call to a Riemann solver were required to remove the overshoots. The new implementation removes the ambiguity associated with basing the edge updates off a cell-average value by basing the updates on the vertex value. Additionally, the linearization is performed entirely within a single cell. As a result, the method no longer requires explicit shock detection to properly capture these features.

Figure 6 shows the solution at  $t_s = 0.174$  (shock formation) and  $t_1 = 0.375$ . Both the limited and unlimited schemes produce similar results prior to the shock forming, but differentiate after that. The results in Fig. 6(b) were produced applying limiting to the just the vertex values. Also limiting edge-averages further reduces the overshoot, but does not eliminate it. Both approaches capture the shock itself well and closely match the exact solution in smooth regions.

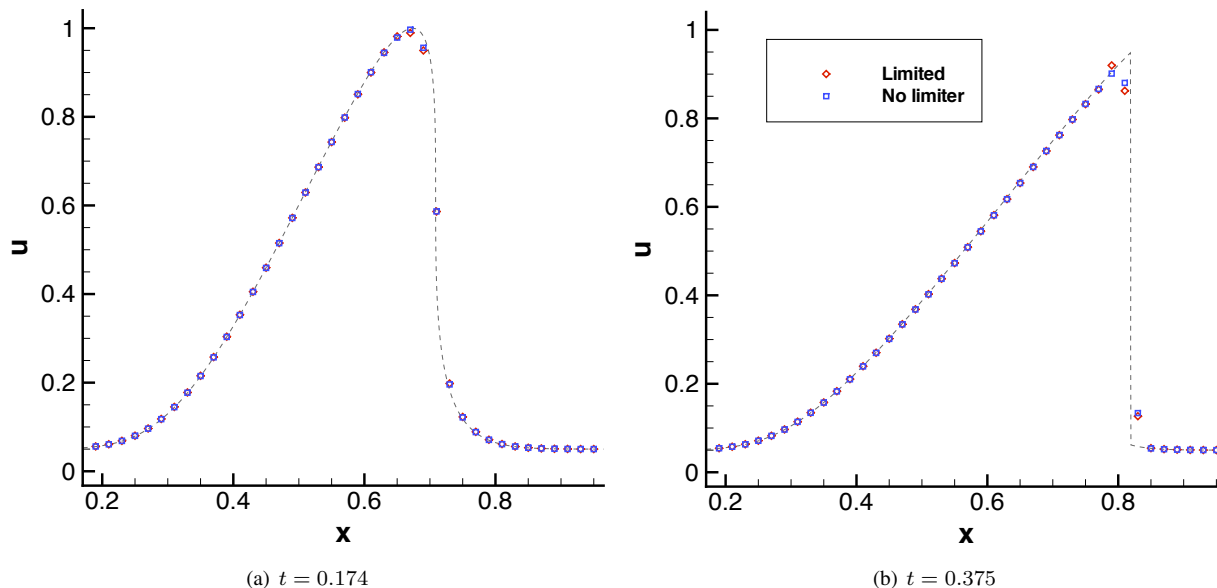


Figure 6. Solution of Gaussian pulse at shock formation (left) and post shock formation (right). Exact solution shown with dotted line.

A steady shock case was also computed to evaluate the scheme's ability to reach machine zero. The solution was initialized with a compression wave in Eq. (17), such that a standing shock forms at  $t = 0.25$  between the cell centroid



and cell edge. Final shock locations directly at the cell edge and centroid were also investigated but did not significantly change the convergence behavior. Figure 7 compares the norm of the residual with and without the limiter. Here, only vertex values are being limited. The figure shows that the residual norm steadily drops to approximately  $1 \times 10^{-13}$  once the shock structure is established. While there is some chatter toward the end of the unlimited residual history, activating the limiter eliminates the oscillations.

$$u_0(x) = \begin{cases} 1 & x \leq 0.26875 \\ 2.075 - 4.0x & 0.26875 < x \leq 0.76875 \\ -1 & x > 0.76875 \end{cases} \quad (17)$$

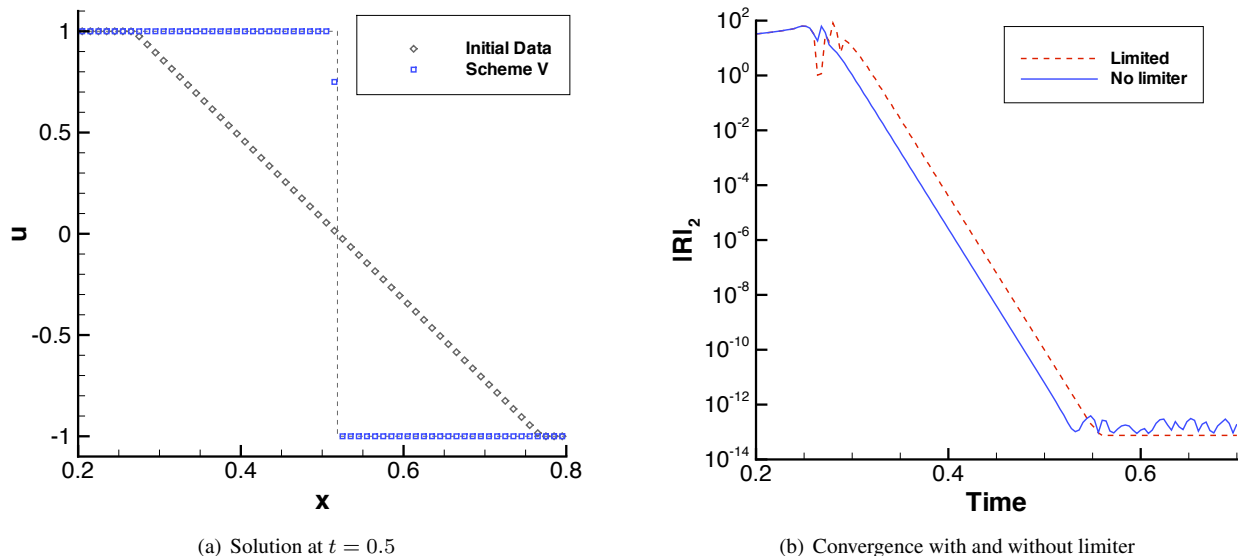


Figure 7. Burgers steady shock case

#### IV.C. Euler results

To test the scheme on a more complicated problem, Scheme V was used to generate solutions for Sod's shocktube problem.<sup>20</sup> The mesh used 100 cells with a uniform spacing of  $\Delta x = 0.01$ . The time step was set at  $\Delta t = 0.002$ . Figure 8 shows the computed density and Fig. 9 illustrates the velocity profile. The exact solution is shown within the figures by the dotted line and exact cell-averages are indicated by filled circles. Each figure compares two limiting options: limiting only the vertex values and limiting both the vertex and edge-average values. Both solutions capture the shock and contact quite sharply, but each also has its drawbacks. The vertex-only solution is more symmetric around the shock and does a better job of limiting the expansion; however, there appears to be some error near the original discontinuity location of  $x = 0.5$ . Limiting the edge-average values along with the vertex values appears to clean up errors near the expansion, but, as with the Burgers' case, this limiting strategy appears to induce a new extremum at the shock that is not present in the unlimited solution. The new velocity peak value is relatively small though, measuring only 1.77% above the exact solution.

### V. Conclusion

Active flux schemes have been demonstrated in one dimension and generated accurate solutions for the linear advection equation, nonlinear Burgers equation, and the Euler equations. It has been shown that randomizing the grid has very little effect on the accuracy. This is because the reconstructions are performed entirely within each cell, and do not depend on surrounding geometry. Additionally, the limiting approach reduces spurious oscillations in the solution without clipping legitimate extrema. The higher accuracy does increase the cost of the schemes over traditional finite-volume approaches; however, the memory savings gained through the data storage strategy and simple, effective

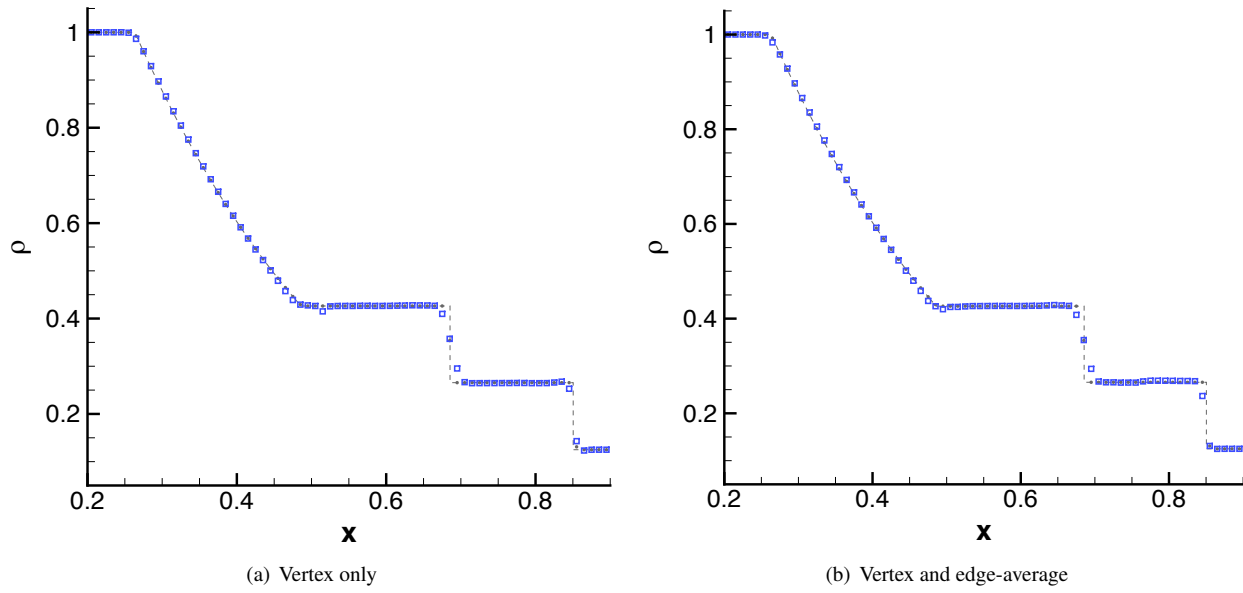


Figure 8. Density at  $t = 0.2$  using two limiting options

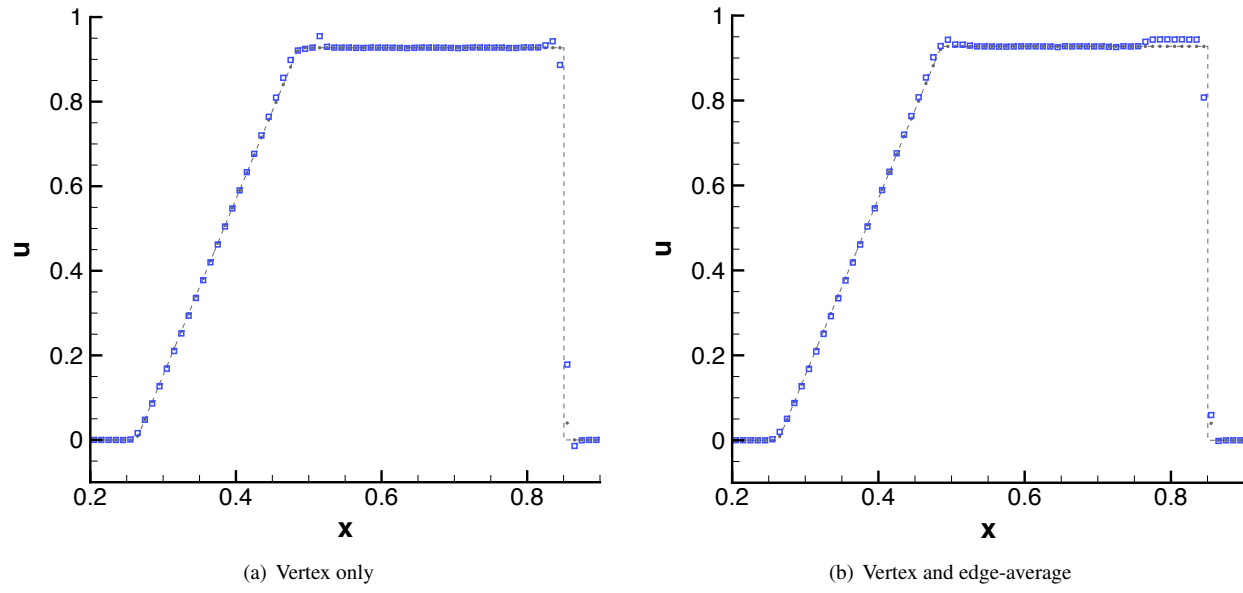


Figure 9. Velocity at  $t = 0.2$  using two limiting options

limiter suggest that the scheme will be less computationally expensive than other high order methods such as DG or WENO schemes, even in higher dimensions.

Future work on active fluxes will concentrate on improving the characteristic-based limiter and extending the approach to multi-dimensional systems. The nonlinear tests conducted so far indicate that the limiting approach is sound, but it requires improvement to more accurately simulate the flow conditions, especially around shocks. While the extension to multi-dimensional systems will not be trivial, there are not any foreseeable limitations with the approach that make it impossible. Six degrees of freedom are required for third order accuracy in 2-D. A seventh may be added as a symmetrical “bubble function” to ensure conservation. A natural location for the conserved variable is at the triangle centroid, leaving six locations at which to store solution data that does not need to be conserved (i.e. fluxes). An obvious choice is to store flux data at the three cell vertices and three edge midpoints. For 3-D tetrahedra, the vertices and edges also provide the correct degrees of freedom. In either the two or three-dimensional case, once a location for the variables is determined, all the 1-D concepts of carefully conserving centroid data while updating flux

data using appropriate upwinding may be directly applied.

There are, of course, other second and third order members of the active flux family that may be robust, useful schemes. An additional area of research will be exploring the various schemes in search of favorable properties. Active flux schemes are targeted for use within the Kestrel flow solver, but other codes are expected to benefit from their generality. Because active flux schemes are an offshoot of the finite-volume approach that forms the basis for many of the CFD codes currently used throughout industry, we anticipate that integrating these new schemes within existing solvers that users are familiar with will be a much simpler task than inserting other high order methods.

## References

- <sup>1</sup>Warming, R., Kutler, P., and Lomax, H., "Second- and Third-Order Noncentered Difference Schemes for Nonlinear Hyperbolic Equations," *AIAA Journal*, Vol. 11, No. 2, 1973, pp. 189–196.
- <sup>2</sup>Harten, A., Engquist, B., Osher, S., and Charkravarthy, S. R., "Uniformly high order accurate essentially non-oscillatory schemes, III," *Journal of Computational Physics*, Vol. 71, 1987, pp. 231–303.
- <sup>3</sup>Liu, X.-D., Osher, S., and Chan, T., "Weighted Essentially Non-oscillatory Schemes," *Journal of Computational Physics*, Vol. 115, 1994, pp. 200–212.
- <sup>4</sup>Ollivier-Gooch, C. F., "Quasi-ENO Schemes for Unstructured Meshes Based on Unlimited Data-Dependent Least-Squares Reconstruction," *Journal of Computational Physics*, Vol. 133, 1997, pp. 6–17.
- <sup>5</sup>Friedrich, O., "Weighted Essentially Non-Oscillatory Schemes for the Interpolation of Mean Values on Unstructured Grids," *Journal of Computational Physics*, Vol. 144, 1998, pp. 194–212.
- <sup>6</sup>Cockburn, B., Lin, S.-Y., and Shu, C.-W., "TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems," *J. Comput. Phys.*, Vol. 84, 1989, pp. 90–113.
- <sup>7</sup>Wang, Z. J. and Liu, Y., "The Spectral Difference Method for the 2D Euler Equations on Unstructured Grids," *17th AIAA Computational Fluid Dynamics Conference*, 2005, AIAA Paper 2005-5112.
- <sup>8</sup>Wang, Z. J., "Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids: Basic Formulation," *Journal of Computational Physics*, Vol. 178, 2002, pp. 210–251.
- <sup>9</sup>Allmaras, S. R., Bussoletti, J. E., Hildes, C. L., Johnson, F. T., Melvin, R. G., Tinoco, E. N., Venkatakrishnan, V., Wigdon, L. B., and Young, D. P., "Algorithm Issues and Challenges Associated with the Development of Robust CFD Codes," *Variational Analysis and Aerospace Engineering*, Springer New York, 2009, pp. 1–19.
- <sup>10</sup>Kuzmin, D., "A vertex-based hierarchical slope limiter for  $p$ -adaptive discontinuous Galerkin methods," *Journal of Computational and Applied Mathematics*, Vol. 233, 2010, pp. 3077–3085.
- <sup>11</sup>van Leer, B., "Towards the Ultimate Conservative Difference Scheme. IV. A New Approach to Numerical Convection," *Journal of Computational Physics*, Vol. 23, No. 3, 1977, pp. 276–299.
- <sup>12</sup>Iserles, A., "Generalized Leapfrog Methods," *IMA Journal of Numerical Analysis*, Vol. 6, 1986, pp. 381–392.
- <sup>13</sup>Roe, P., "Linear Bicharacteristic Schemes Without Dissipation," *SIAM Journal of Scientific Computing*, Vol. 19, No. 5, 1998, pp. 1405–1427.
- <sup>14</sup>Karabasov, S. and Goloviznin, V., "Compact Accurately Boundary-Adjusting high-Resolution Technique for fluid dynamics," *Journal of Computational Physics*, Vol. 228, 2009, pp. 7426–7451.
- <sup>15</sup>Roe, P. L., "Fluctuations and Signals - A Framework for Numerical Evolution Problems," *Numerical Methods for Fluid Dynamics*, 1982, pp. 219–257.
- <sup>16</sup>Eymann, T. A. and Roe, P., "Active Flux Schemes," *49th AIAA Aerospace Sciences Meeting*, 2011, AIAA Paper 2011-382.
- <sup>17</sup>Godunov, S. K., "A Difference Scheme for Numerical Computation of Discontinuous Solution of Hydrodynamic Equations," *Matematicheskii Sbornik*, Vol. 47, 1959, pp. 271–306.
- <sup>18</sup>Roe, P. L., "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.
- <sup>19</sup>Zalesak, S. T., "Fully Multidimensional Flux-Corrected Transport Algorithms for Fluids," *Journal of Computational Physics*, Vol. 31, No. 3, 1979, pp. 335–362.
- <sup>20</sup>Sod, G., "A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws," *Journal of Computational Physics*, Vol. 27, 1978, pp. 1–31.