



A Central Compact-Reconstruction WENO Method for Hyperbolic Conservation Laws

Kilian Cooley* and Dr. James Baeder†
University of Maryland, College Park, Maryland, 20740

A central WENO scheme is presented which requires no Riemann solver. The WENO reconstructions are compact in the sense that unknowns in adjacent cells are coupled. This coupling provides enough degrees of freedom to design WENO schemes for which the coefficient matrices for the subcell-average and point-value reconstructions are the same, so that only one linear system must be solved per time step. Numerical tests demonstrate that the scheme achieves fifth-order accuracy on smooth solutions and is largely non-oscillatory at discontinuities.

I. Introduction

INVISCID, compressible flows can contain both fine-scale features (turbulent eddies) and discontinuities (shocks), and are described by the Euler equations which are a system of hyperbolic PDEs. Linear schemes of the high order required to efficiently resolve turbulence produce spurious oscillations at shocks. Therefore a nonlinear scheme that is high-order accurate in smooth regions and non-oscillatory at shocks is required. The Weighted Essentially Non-Oscillatory (WENO) schemes [1] fulfill these requirements by adapting the reconstruction stencil in response to the local smoothness of the solution. The Compact-Reconstruction WENO (CRWENO) scheme [2] uses the adaptive stenciling process to automatically couple unknowns in smooth regions, resulting in improved resolution of small-scale features.

Both WENO and CRWENO are generalizations of the first-order Godunov scheme: two values are reconstructed at each interface and a Riemann solver computes a numerical flux for that interface from those values. This process requires the reconstruction to be performed in local characteristic variables so that the inputs to the Riemann solver are biased entirely in the upwind or downwind directions. In contrast, central schemes extend the first-order Lax-Friedrichs scheme and avoid the need for any Riemann solver by evaluating the flux function only at points where the reconstruction is continuous. In principle, these schemes also require no transformation to characteristic variables.

The original central scheme [3] used a piecewise-linear MUSCL-like reconstruction. Levy et al. developed a series of central WENO schemes ([4], [5], [6], [7]) culminating in a fourth-order scheme for multidimensional systems of conservation laws [8]. Whereas these schemes formed the reconstruction by a weighted combination of polynomials, Qiu and Shu developed fifth- and ninth-order central WENO schemes using finite-volume formulas without explicitly forming a reconstruction polynomial [9]. They found that oscillations still arose when the characteristic decomposition was not used, but could be alleviated by resorting to characteristic variables only in the subcell reconstruction.

The present work develops a central compact-reconstruction WENO (CCRWENO) scheme to produce a non-oscillatory scheme that does not require a characteristic decomposition or a Riemann solver, extends componentwise to systems, and efficiently resolves small-scale features. The paper is organized as follows. Section II reviews WENO, CRWENO, and central schemes. Section III develops the one-dimensional CCRWENO scheme which is extended to two dimensions in Section IV. Section V describes the modifications required at boundaries. We present numerical tests in Section VI in comparison to the schemes of Levy et al. in [4] and [8] and the CWENO5 scheme in [9]. We conclude with remarks on future improvements to CCRWENO.

II. WENO, CRWENO, and Central Schemes

A. WENO Schemes

Rather than apply a single scheme at every solution point, WENO schemes employ a convex combination of subschemes:

*Graduate Assistant, Applied Mathematics and Scientific Computing, 4176 Campus Drive, AIAA Student Member

†Professor, Department of Aerospace Engineering, 4298 Campus Drive, AIAA Associate Fellow

$$u_{j+1/2} = \sum_k \omega_k u_{j+1/2}^{(k)} \quad (1)$$

where the candidate interface values $u^{(k)}$ reconstructed from cell-averages \bar{u} by linear schemes, e.g. the third-order schemes of [10]:

$$\begin{aligned} u_{j+1/2}^{(1)} &= \frac{1}{3}\bar{u}_{j-2} - \frac{7}{6}\bar{u}_{j-1} + \frac{11}{6}\bar{u}_j \\ u_{j+1/2}^{(2)} &= -\frac{1}{6}\bar{u}_{j-1} + \frac{5}{6}\bar{u}_j + \frac{1}{3}\bar{u}_{j+1} \\ u_{j+1/2}^{(3)} &= \frac{1}{3}\bar{u}_j + \frac{5}{6}\bar{u}_{j+1} - \frac{1}{6}\bar{u}_{j+2} \end{aligned} \quad (2)$$

The weights ω_k adapt to the local smoothness of the solution. Un-normalized, non-convex weights are first calculated by scaling a set of ideal weights $\bar{\omega}_k$. Those ideal weights are chosen to cancel the leading-order error terms in Eq. (2) so that

$$\sum_k \bar{\omega}_k u_{j+1/2}^{(k)} = u(x_{j+1/2}) + O(\Delta x^5) \quad (3)$$

The ideal weights are then scaled:

$$\alpha_k = \frac{\bar{\omega}_k}{(\epsilon + \beta_k)^2} \quad (4)$$

The smoothness indicators β_k are the scaled sums of squared L^2 norms of the derivatives of the reconstruction polynomial formed over each stencil (see [10]), which when written explicitly are:

$$\begin{aligned} \beta_1 &= \frac{13}{12}(u_{j-2} - 2u_{j-1} + u_j)^2 + \frac{1}{4}(u_{j-2} - 4u_{j-1} + 3u_j)^2 \\ \beta_2 &= \frac{13}{12}(u_{j-1} - 2u_j + u_{j+1})^2 + \frac{1}{4}(u_{j-1} - u_{j+1})^2 \\ \beta_3 &= \frac{13}{12}(u_j - 2u_{j+1} + u_{j+2})^2 + \frac{1}{4}(3u_j - 4u_{j+1} + u_{j+2})^2 \end{aligned} \quad (5)$$

The parameter ϵ is a small number introduced to prevent division by zero, and is set to 10^{-6} for all computations presented here. Finally, the convex weights $\bar{\omega}_k$ are computed by normalizing the non-convex weights α_k :

$$\omega_k = \frac{\alpha_k}{\sum_m \alpha_m} \quad (6)$$

If a discontinuity exists within any of the cells whose averages determine $u_{j+1/2}^{(k)}$, then the corresponding β_k will be large and the corresponding α_k will be small leading to a small ω_k . As a result, the reconstructions that would lead to spurious oscillations are automatically assigned small influence in the final reconstruction Eq. (1). It can be shown (see [10]) that the weights assigned to discontinuous stencils are of size $O(\Delta x^2)$, so that as the grid is refined the amplitude of oscillations caused by the discontinuous stencils decreases. This behavior almost completely prevents the Gibbs phenomenon.

If, on the other hand, if the solution is smooth in all stencils then the β_k are all similar in magnitude so that the ω_k approximate their ideal values $\bar{\omega}_k$. The convex combination Eq. (1) then approaches the high-order linear scheme Eq. (3) as the grid is refined, maintaining fifth-order accuracy as long as $\omega_k - \bar{\omega}_k = O(\Delta x^2)$ for each k . The solution-dependent nature of the weights therefore yields high accuracy away from shocks yet prevents oscillations at shocks. For systems of equations, the smoothness indicators can be calculated for each component independently or a single indicator can be used for all components. In the latter case, which is used for all computations presented in Section VI, that single indicator is simply the sum of the indicators for the individual components.

B. CRWENO Schemes

Ghosh and Baeder [2] modified the WENO idea to use compact (i.e. spatially implicit) stencils in which the unknown interface values are coupled. The candidate schemes in Eq. (2) are replaced by compact third-order schemes:

$$\begin{aligned} \frac{2}{3}u_{j-1/2} + \frac{1}{3}u_{j+1/2} &= \frac{1}{6}\bar{u}_{j-1} + \frac{5}{6}\bar{u}_j \\ \frac{1}{3}u_{j-1/2} + \frac{2}{3}u_{j+1/2} &= \frac{5}{6}\bar{u}_j + \frac{1}{6}\bar{u}_{j+1} \\ \frac{2}{3}u_{j+1/2} + \frac{1}{3}u_{j+3/2} &= \frac{1}{6}\bar{u}_j + \frac{5}{6}\bar{u}_{j+1} \end{aligned} \quad (7)$$

with ideal weights $\bar{\omega}_{1,2,3} = \frac{1}{5}, \frac{1}{2}, \frac{3}{10}$. This method inherits the improved accuracy and spectral resolution of compact finite-difference schemes [11] at the price of greater computational expense, now that a system of equations must be solved. In the one-dimensional scalar case, this Compact-Reconstruction WENO (CRWENO) scheme allows a given level of accuracy to be obtained with a coarser grid compared to what the WENO scheme requires. In the case of systems, however, stability concerns require that the reconstruction step be applied to characteristic variables instead of to the conserved variables directly. Performing that change of variables requires the eigenvectors of the flux jacobian which might be costly to compute. Once those eigenvectors are found, instead of D tridiagonal systems (one for each component of a system) one must solve a single block-tridiagonal system with blocks of size $D \times D$. This necessary change destroys the advantage of CRWENO over WENO - the same accuracy can be obtained at less expense by using WENO on a finer grid. If the need to use characteristic variables could be avoided, however, then the resultant componentwise CRWENO scheme would be superior to the componentwise WENO scheme. Moreover, the components can be solved in parallel if the system can be treated componentwise whereas parallelization becomes difficult when characteristic variables are required.

The decomposition to characteristic variables ultimately arises from the upwind nature of the full WENO scheme, since solving the Riemann problem at interfaces requires a left-biased and a right-biased input where the bias refers to the local characteristic directions. Therefore the solution to the problem at hand requires an alternative framework to that of the upwind schemes.

C. Central Schemes

Consider a hyperbolic conservation law in one space dimension:

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} f(u) = 0 \quad (8)$$

Like the upwind schemes, central schemes begin with a piecewise continuous reconstruction of the solution from cell-averages, but avoid expensive operations at interfaces by introducing an auxiliary grid of staggered cells centered on those interfaces as in Fig. 1.

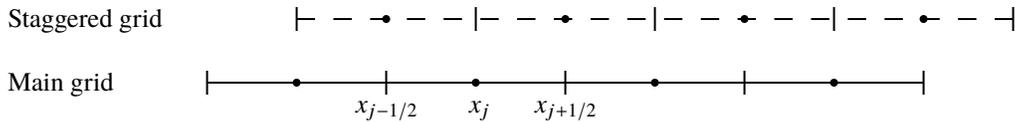


Fig. 1 The cells of the staggered grid are centered on the interfaces of the main grid.

Consider the staggered cell with endpoints x_j, x_{j+1} , with the averages of the solution $u(x)$ over this cell given by $\bar{u}_{j+1/2}^n, \bar{u}_{j+1/2}^{n+1}$ at time indices n and $n+1$ respectively. We introduce integrated quantities into the discretization by integrating Eq. (8) over the space-time region $[x_j, x_{j+1}] \times [t^n, t^{n+1}]$:

$$\int_{x_j}^{x_{j+1}} u^{n+1}(x) - u^n(x) dx + \int_{t^n}^{t^{n+1}} f(u(x_{j+1})) - f(u(x_j)) dx = 0 \quad (9)$$

The staggered cell-average at the next time index can then be expressed in terms of its current value according to:

$$\bar{u}_{j+1/2}^{n+1} = \bar{u}_{j+1/2}^n - \frac{1}{\Delta x} \int_{t^n}^{t^{n+1}} f(u(x_{j+1})) - f(u(x_j)) dx \quad (10)$$

Remarks

- 1) Repeating the process with staggered-cell averages as the input returns averages over cells aligned with the main grid.
- 2) For sufficiently small time steps, the waves emanating from the discontinuities at interfaces stay within the staggered cells so the solution at midpoints x_j of the original cells remains smooth. Any discontinuities emanating from the interface do not reach the midpoint within one time step. The time integrals in Eq. (10) are therefore smooth, and can be approximated by any quadrature formula.
- 3) The need for a Riemann solver is avoided because there is no need to assign a numerical flux at a point where the reconstructed solution is discontinuous. The physical flux function is only ever evaluated at main-grid cell midpoints, where the reconstructed solution remains smooth.
- 4) Evaluating the time integrals requires only the point values of the solution at cell midpoints. These quantities can be evolved using the differential form of the conservation law, Eq. (8), without any consideration to cell averages.
- 5) To maintain conservation when moving from the main grid to the staggered grid, the reconstructed quantities are actually the averages over subcells of the main grid, e.g. $[x_{j-1/2}, x_j]$ for the left subcells, then the right-subcell averages are computed to be consistent with the averages over the main cells. The averages over staggered cells are then computed by joining right-subcells with left-subcells from neighbors to the right in the main grid.
- 6) Since no Riemann solver is required and only the physical flux function is ever evaluated, central schemes can be applied as a black-box method to problems for which no Riemann solver is available.

D. Motivation

The Levy scheme [4] forms a single reconstruction polynomial in each cell. Its explicit use of polynomials limits its accuracy to fourth order but allows one set of weights to be used for both the subcell-average and point-value reconstructions. On the other hand, [9] eschew the polynomial approach in favor of finite-difference-type formulas. This choice allows greater accuracy on the same stencil at the cost of requiring different weights for the two reconstruction steps. That the weights are different implies that there can be no single polynomial underlying the reconstructions in [9]. Although that scheme performs reasonably well, its reconstructions are inconsistent in this sense and it is possible that this inconsistency explains why Qiu and Shu found a characteristic decomposition necessary whereas Levy et al. in [4] did not. We aim to test this conjecture by developing a central WENO scheme that has fifth-order accuracy but uses the same weights for the reconstructions and keeps the same five-cell stencil. Doing so requires compact reconstructions since the only non-compact fifth-order reconstructions on the five-cell stencil are those in [9]. Compact stencils allow additional degrees of freedom that can be used to design the resulting scheme to have desirable properties. The motivations for the present work are now threefold:

- 1) Restore the advantage of compact reconstructions for systems by removing the need for characteristic variables.
- 2) Improve robustness compared to CRWENO by avoiding a confounding step (the Riemann solver).
- 3) Test the preceding conjecture on the reason for requiring a characteristic decomposition in [9].

III. The Central CRWENO Scheme in One Dimension

One time step of a central scheme proceeds as follows:

- 1) From cell averages on the main grid at time index n , reconstruct the left-subcell averages.
- 2) Compute the right-subcell averages by conservation.
- 3) Compute the averages over staggered cells.
- 4) Reconstruct the point values of the solution at main-grid cell midpoints (staggered-cell interfaces).
- 5) Compute the flux and flux derivatives at main-grid cell midpoints.
- 6) Repeat step 4 as needed (see Section C) to calculate the time integrals in Eq. (10).
- 7) Compute the staggered cell-averages at time index $n + 1$ according to Eq. (10).

Thus there are three quantities that need to be reconstructed: subcell averages from cell averages, midpoint values from cell averages, and flux derivatives from point values of the flux. The CCRWENO idea is to perform each of these reconstructions using a compact WENO reconstruction. Each reconstruction therefore requires three subschemes and a set of ideal weights. In the next sections we design those subschemes.

A. The Subcell-Average and Point-Value Reconstructions

The central WENO scheme of [4] explicitly forms a reconstruction polynomial in each cell as a weighted combination of parabolic reconstruction polynomials. From this reconstruction both the subcell averages and point values are computed with the same non-oscillatory weights. The use of polynomials, however, limits the central WENO scheme to fourth-order accuracy whereas we aim for CCRWENO to be at least fifth-order accurate. Fortunately, it turns out to be possible to design subschemes for the point-value reconstruction that use the same ideal weights as the subcell-average reconstruction and meet the fifth-order accuracy requirement. Such schemes eliminate the need to calculate a new set of non-oscillatory weights for the point-value reconstruction.

Experimentation with these schemes showed that oscillations and numerical instability can arise if certain conditions are not met. Together with the accuracy and ideal weight conditions, the design conditions for the subschemes are as follows:

- 1) The left-hand-side (LHS) coefficient matrix must be diagonally dominant for all possible combinations of the non-oscillatory weights, for both the subcell-average and point-value reconstructions.
- 2) The subcell-average and point-value reconstructions must use the same ideal weights.
- 3) The resulting CCRWENO scheme must achieve at least 5th-order accuracy in all reconstructions.
- 4) The LHS coefficients must sum to 1.
- 5) The ideal weights must be symmetric, i.e. $\bar{\omega}_1 = \bar{\omega}_3$.
- 6) The LHS of the combined scheme with ideal weights must be symmetric with respect to j .

The first condition is equivalent to requiring each individual subscheme to be diagonally dominant, in the sense that the coefficient of the unknown at point j is greater in absolute value than the sum of all other coefficients of unknowns. The symmetry conditions on the ideal weights and combined LHS coefficients ensure that the same subcell reconstructions produce the same result regardless of which subcell is reconstructed and which is calculated by conservation. Although it is possible to treat negative ideal weights that arise by the method of [12], to do so requires additional computational expense so we aim for one additional condition:

- 7) The ideal weights should be positive and sum to 1.

Considerable computational expense can be avoided if, in addition to matching the ideal weights for the two reconstructions, the LHS coefficients of the individual subschemes can also be matched. This would cause the coefficient matrices for the two reconstructions to be identical, with only the RHS differing to produce the reconstructed subcell averages and point values. The benefit to this constraint is so great that we include it as the final condition:

- 8) The LHS coefficients of each subcell-average subscheme must match those of the corresponding point-value subscheme.

A family of schemes generated by the parameter $d_1 \geq 0$ exists that satisfy all of these conditions:

$$d_2 = \frac{5 + 8d_1}{17 + 20d_1} \quad (11)$$

$$\begin{aligned} \bar{\omega}_1 = \frac{1 - \bar{\omega}_2}{2} : \quad & \frac{1 - d_1}{2} \bar{u}_{j-1}^L + \frac{1 + d_1}{2} \bar{u}_j^L = \frac{-d_1}{16} \bar{u}_{j-2} + \frac{3 - d_1}{8} \bar{u}_{j-1} + \frac{3d_1 + 2}{16} \bar{u}_j \\ \bar{\omega}_2 = \frac{d_1 + \frac{1}{4}}{d_1 + d_2} : \quad & \frac{1 - d_2}{4} \bar{u}_{j-1}^L + \frac{1 + d_2}{2} \bar{u}_j^L + \frac{1 - d_2}{4} \bar{u}_{j+1}^L = \frac{3 - 2d_2}{16} \bar{u}_{j-1} + \frac{1 + d_2}{4} \bar{u}_j + \frac{1 - 2d_2}{16} \bar{u}_{j+1} \end{aligned} \quad (12)$$

$$\begin{aligned} \bar{\omega}_3 = \frac{1 - \bar{\omega}_2}{2} : \quad & \frac{1 + d_1}{2} \bar{u}_j^L + \frac{1 - d_1}{2} \bar{u}_{j+1}^L = \frac{5d_1 + 6}{16} \bar{u}_j + \frac{1 - 3d_1}{8} \bar{u}_{j+1} + \frac{d_1}{16} \bar{u}_{j+2} \\ \bar{\omega}_1 = \frac{1 - \bar{\omega}_2}{2} : \quad & \frac{1 - d_1}{2} u_{j-1} + \frac{1 + d_1}{2} u_j = \frac{-1}{24} \bar{u}_{j-2} + \frac{7 - 6d_1}{12} \bar{u}_{j-1} + \frac{11 + 12d_1}{24} \bar{u}_j \\ \bar{\omega}_2 = \frac{d_1 + \frac{1}{4}}{d_1 + d_2} : \quad & \frac{1 - d_2}{4} u_{j-1} + \frac{1 + d_2}{2} u_j + \frac{1 - d_2}{4} u_{j+1} = \frac{5 - 6d_2}{24} \bar{u}_{j-1} + \frac{7 + 6d_2}{12} \bar{u}_j + \frac{5 - 6d_2}{24} \bar{u}_{j+1} \quad (13) \\ \bar{\omega}_3 = \frac{1 - \bar{\omega}_2}{2} : \quad & \frac{1 + d_1}{2} u_j + \frac{1 - d_1}{2} u_{j+1} = \frac{11 + 12d_1}{24} \bar{u}_j + \frac{7 - 6d_1}{12} \bar{u}_{j+1} + \frac{-1}{24} \bar{u}_{j+2} \end{aligned}$$

Because the average over the right subcell is calculated by conservation, it is convenient to reconstruct the half-averages $\bar{u}_j^L = \frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_j} u(x) dx$ for the left subcells and obtain the half-average over the right subcell by subtraction. The averages over staggered cells can then be easily found by adding half-averages of a left subcell and right subcell that share an interface. In this way all integrated quantities are normalized by the volume of a cell.

The combined point-value scheme with ideal weights is unstable for $0 \leq d_1 \leq 1.2$. Although this instability does not noticeably affect the solution in some cases, in general a value of $d_1 \geq 1.3$ appears to confer stability even for solutions with high-frequency content. All results presented here used $d_1 = 1.3$.

B. Flux Derivative Reconstruction

The flux derivatives at each cell midpoint must be calculated once per Runge-Kutta stage (see Section C). It is not possible to reuse the coefficient matrix from the subcell-average and point-value reconstructions because either the ideal weights or the LHS coefficients in the subschemes must be different in order to satisfy the accuracy condition. Furthermore, no symmetric and diagonally dominant compact scheme with positive weights exceeds the fourth-order accuracy that is easily attained by the non-compact scheme:

$$\begin{aligned}\bar{\omega}_1 = \frac{1}{6} : \quad \Delta x(u_x)_j &= \frac{1}{2}u_{j-2} - 2u_{j-1} + \frac{3}{2}u_j \\ \bar{\omega}_2 = \frac{2}{3} : \quad \Delta x(u_x)_j &= -\frac{1}{2}u_{j-1} + \frac{1}{2}u_{j+1} \\ \bar{\omega}_3 = \frac{1}{6} : \quad \Delta x(u_x)_j &= -\frac{3}{2}u_j + 2u_{j+1} - \frac{1}{2}u_{j+2}\end{aligned}\tag{14}$$

Since the time integral in Eq. (10) is approximated to fourth order anyway a fourth-order flux derivative reconstruction is sufficient, and since the compact reconstruction would be more expensive without improving the order of accuracy, we opt to use the non-compact scheme Eq. (14) for the flux derivative reconstructions.

C. Time Advancement

Advancing to the next time step requires approximation of a time integral. The quadrature of this integral requires evaluating the integrand at points between t^n and t^{n+1} . However, the evolution of the flux values is performed by a Runge-Kutta procedure applied to Eq. (8), i.e. to:

$$\frac{du_j}{dt} = -\left. \frac{\partial f(u)}{\partial x} \right|_{x_j}\tag{15}$$

which after discretization becomes:

$$\frac{du_j}{dt} = F_j(t, u)\tag{16}$$

where the right-hand side is calculated in a WENO process defined by Eq. (14). The Runge-Kutta process in general does not return values that can be interpreted as located at the desired quadrature points, and performing a separate Runge-Kutta process for each quadrature point would be computationally expensive. Instead, we employ a natural continuous extension [13] of the fourth-order Runge-Kutta scheme:

$$\begin{aligned}u^{n+1} &= u^n + \Delta t \sum_{k=1}^4 b_k G_k \\ G_1 &= F(u^n) \\ G_2 &= F\left(t^n + c_2 \Delta t, u^n + \frac{\Delta t}{2} G_1\right) \\ G_3 &= F\left(t^n + c_3 \Delta t, u^n + \frac{\Delta t}{2} G_2\right) \\ G_4 &= F\left(t^n + c_4 \Delta t, u^n + \Delta t G_3\right) \\ c_2 &= \frac{1}{2}, \quad c_3 = \frac{1}{2}, \quad c_4 = 1 \\ b_1 &= \frac{1}{6}, \quad b_2 = \frac{1}{3}, \quad b_3 = \frac{1}{3}, \quad b_4 = \frac{1}{6}\end{aligned}\tag{17}$$

The natural continuous extension replaces the coefficients b_i with polynomials $b_i(\theta)$ to construct a polynomial $z(t)$ such that:

$$z(t^n + \theta \Delta t) = y^n + \Delta t \sum_{k=1}^4 b_k(\theta) G_k, \quad 0 \leq \theta \leq 1$$

$$z(t^n) = u^n \text{ and } z(t^{n+1}) = u^{n+1}$$

$$\max_{t^n \leq t \leq t^{n+1}} |u^{(r)}(t) - z^{(r)}(t)| = \mathcal{O}(\Delta t^{4-r}), 0 \leq r \leq 4$$

where $u(t)$ is the exact solution to Eq. (16). The polynomials $b_k(\theta)$ are:

$$b_1(\theta) = 2(1 - 4b_1)\theta^3 + 3(3b_1 - 1)\theta^2 + \theta$$

$$b_k(\theta) = 4(3c_k - 2)b_k\theta^3 + 3(3 - 4c_k)b_k\theta^2, \quad k = 2, 3, 4$$

Since $z(t)$ uniformly approximates the solution to Eq. (15) within $0 \leq \theta \leq 1$ it can be evaluated at the quadrature points in order to accurately approximate the time integrals in Eq. (10). If the quadrature points for the time integrals are known in advance, then the values of $b_i(\theta)$ can be precalculated. All results presented here were obtained by using Simpson's rule on the natural continuous extensions performed at each time step.

Remark: Upwind WENO and CRWENO schemes are usually paired with a strong-stability-preserving (SSP) Runge-Kutta scheme [14] to enhance their non-oscillatory properties. However, the typical third-order SSPRK scheme, used in [2] for example, does not have a natural continuous extension that allows it to be used in the present framework. Furthermore, the fourth-order RK scheme Eq. (17) is not SSP. We find that Eq. (17) with its natural continuous extension produces sufficiently non-oscillatory solutions anyway.

IV. Extension to Two Space Dimensions

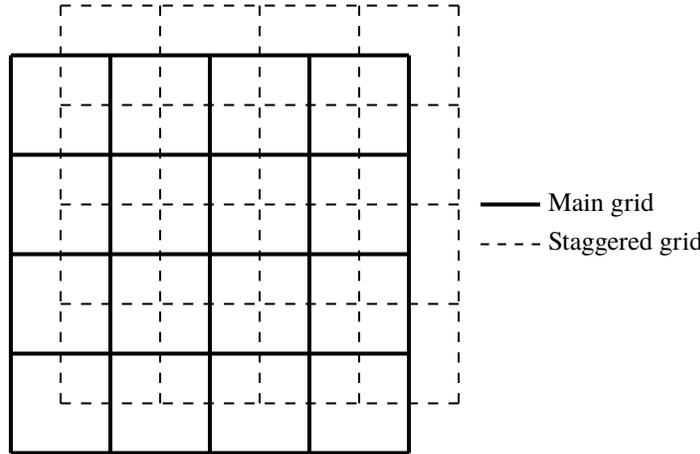


Fig. 2 The staggered-grid framework extends to two dimensions.

The staggered-grid framework extends straightforwardly to two space dimensions, with the staggered cells being centered at the vertices of the main cells as in Fig. 2. Instead of two subcells that are intervals, we have four subcells that are rectangles. The analog of Eq. (10) is:

$$\bar{u}_{i+1/2, j+1/2}^{n+1} = \bar{u}_{i+1/2, j+1/2}^n - \frac{1}{\Delta x \Delta y} \int_{t^n}^{t^{n+1}} \left(\int_{x_i}^{x_{i+1}} g(u(x, y_{j+1})) - g(u(x, y_j)) dx + \int_{y_j}^{y_{j+1}} f(u(x_{i+1}, y)) - f(u(x_i, y)) dy \right) dt \quad (18)$$

The space integrals on the right-hand side were not present in the one-dimensional version and require special treatment. Though it is possible to evaluate these integrals by a WENO-like process of combining candidate values depending on local smoothness, as in [8] we find that it is enough to approximate them by a simple quadrature rule:

$$\frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} u(x) dx = \frac{-1}{12} u_{i-1} + \frac{13}{12} u_i + \frac{13}{12} u_{i+1} - \frac{1}{12} u_{i+2} + \mathcal{O}(\Delta x^4) \quad (19)$$

Alternatively, a rule with sixth-order accuracy is:

$$\frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} u(x) dx = \frac{11}{140} u_{i-2} - \frac{31}{480} u_{i-1} + \frac{401}{720} u_i + \frac{401}{720} u_{i+1} - \frac{31}{480} u_{i+2} + \frac{11}{140} u_{i+3} + \mathcal{O}(\Delta x^6) \quad (20)$$

In either case, the integral is approximated by a linear combination of values of the integrand at midpoints, which will be computed by a Runge-Kutta method with its natural continuous extension as in the one-dimensional case. For convenience, we exchange the order of integration in Eq. (18), calculating first the time integrals of the fluxes at each cell center and then using Eq. (20) to approximate the spatial integrals.

A. Subcell-Average and Point-Value Reconstructions

The one-dimensional subschemes can be easily promoted to two-dimensional subschemes by making use of the fact that averaging in the x direction followed by averaging in the y direction is equivalent to averaging over a rectangular region:

$$\frac{1}{d-c} \int_c^d \left(\frac{1}{b-a} \int_a^b u(x, y) dx \right) dy = \frac{1}{(b-a)(d-c)} \iint_{[a,b] \times [c,d]} u(x, y) dA \quad (21)$$

It follows that from two one-dimensional schemes for the half-average over the left subcell \bar{u}^L :

$$\begin{aligned} \sum_{i=-1}^1 L_i^{(1)} \bar{u}_i^L &= \sum_{i=-2}^2 R_i^{(1)} \bar{u}_i + \mathcal{O}(\Delta x^p) \\ \sum_{j=-1}^1 L_j^{(2)} \bar{u}_j^L &= \sum_{j=-2}^2 R_j^{(2)} \bar{u}_j + \mathcal{O}(\Delta y^q) \end{aligned} \quad (22)$$

a two-dimensional scheme can be formed as an outer product of the two one-dimensional subschemes:

$$\sum_{i,j=-1}^1 L_i^{(1)} L_j^{(2)} \bar{u}_{i,j}^{LL} = \sum_{i,j=-2}^2 R_i^{(1)} R_j^{(2)} \bar{u}_{i,j} + \mathcal{O}(\Delta x^p, \Delta y^q) \quad (23)$$

where $\bar{u}_{i,j}^{LL}$ denotes the quarter-average over the lower-left subcell of cell i, j . Schemes for the other subcells can be obtained by using subschemes for \bar{u}^R in place of those in Eq. (22), and schemes for the point value can be obtained by replacing them with subschemes from Eq. (13).

In this manner one can produce nine two-dimensional subschemes for the subcell averages and point values. The corresponding ideal weights are simply the corresponding product of the one-dimensional ideal weights. That is, if a two-dimensional subscheme is formed from subschemes i and j with ideal weights $\bar{\omega}_i$ and $\bar{\omega}_j$ respectively, then the ideal weight of the two-dimensional subscheme is $\bar{\omega}_{i,j} = \bar{\omega}_i \bar{\omega}_j$. These weights are still positive and still sum to unity if the one-dimensional weights do.

Schemes constructed in this way retain the key cost-saving property of the one-dimensional schemes. First, because the one-dimensional subschemes have symmetric ideal weights, the two-dimensional ideal weights are symmetric, i.e. $\bar{\omega}_{4-i,j} = \bar{\omega}_{i,j} = \bar{\omega}_{i,4-j}$. Note that the left-hand sides of the one-dimensional subschemes for subcell averages Eq. (12) are symmetric: reflecting the coefficients about the center of cell j does not change the second scheme but interchanges the first and third. Since the first and third ideal weights are equal and the smoothness indicators are not changed, however, this interchange does not affect the final LHS coefficients after the subschemes are combined. The only effect of the reflection is to convert the left-subcell subscheme into a right-subcell subscheme and to change the RHS coefficients. Because the final LHS coefficients are the same for left- and right-subcell subschemes and the two-dimensional subschemes are built from those one-dimensional versions, the final LHS coefficients for all four subcells is the same. Because the one-dimensional subschemes were designed to have the same LHS coefficients for subcell averages and point values, the two-dimensional subschemes built from them inherit that property as well. Ultimately, all of the subcell-average reconstructions and the point-value reconstruction use the same LHS coefficient matrix. In addition, the construction of the two-dimensional subschemes by an outer product guarantees that the two-dimensional CCRWENO method reduces to the one-dimensional version in the case where the solution is constant along a coordinate direction.

B. Time Advancement

The approach of using a Runge-Kutta method with its natural continuous extension does not change. The second spatial dimension adds another flux term so that the analog of Eq. (16) is:

$$\frac{du_j}{dt} = \left(-\frac{\partial f(u)}{\partial x} - \frac{\partial g(u)}{\partial y} \right) \Bigg|_{(x_i, y_j)} = F_j(t, u) \quad (24)$$

Each spatial derivative is calculated independently using the same subschemes Eq. (14) as before, changing the direction of the stencils appropriately to produce derivatives in the x or y directions.

We close this section by remarking that a similar process can extend the one-dimensional scheme to any number of space dimensions. The surface flux integrals can be discretized by versions of the quadrature rules constructed from Eq. (19) or Eq. (20) by outer products, and outer products also enable construction of the multidimensional schemes for the subcell averages and point values. Furthermore, regardless of the number of dimensions only a single coefficient matrix is required.

V. Boundary Treatment

Because the scheme defined by Eq. (12)-Eq. (13) couples the unknowns in three cells it cannot be used for the cells adjacent to the boundary. Ghosh and Baeder in [2] solved the equivalent problem by reverting to the non-compact WENO scheme of [10] in those cells. That approach has a serious drawback in the present framework, because although the two-dimensional non-compact subschemes that would be required can still be expressed as outer products of non-compact one-dimensional subschemes, the resulting subschemes have different ideal weights for the subcell-average and point-value reconstructions. This fact would mean that a second set of non-oscillatory weights would need to be calculated and a second coefficient matrix formed. Fortunately, this consequence can be avoided by the use of bcompact subschemes in which the coupled values are those in the cell adjacent to the boundary and its neighbor opposite the boundary. The resulting subschemes are:

$$c_1 = \frac{389c_3 + 140}{444c_3 + 165}, \quad c_2 = \frac{437c_3 + 170}{738c_3 + 405} \quad (25)$$

$$\begin{aligned} \bar{\omega}_1 &= \frac{148c_3 + 55}{640c_3 + 400} : \frac{1+c_1}{2} \bar{u}_j^L + \frac{1-c_1}{2} \bar{u}_{j+1}^L = \frac{2-3c_1}{16} \bar{u}_{j-2} + \frac{5c_1-3}{8} \bar{u}_{j-1} + \frac{12-7c_1}{16} \bar{u}_j \\ \bar{\omega}_2 &= \frac{246c_3 + 135}{320c_3 + 200} : \frac{1+c_2}{2} \bar{u}_j^L + \frac{1-c_2}{2} \bar{u}_{j+1}^L = \frac{c_2}{16} \bar{u}_{j-1} + \frac{3+c_2}{8} \bar{u}_j + \frac{2-3c_2}{16} \bar{u}_{j+1} \end{aligned} \quad (26)$$

$$\bar{\omega}_3 = 1 - \bar{\omega}_1 - \bar{\omega}_2 : \frac{1+c_3}{2} \bar{u}_j^L + \frac{1-c_3}{2} \bar{u}_{j+1}^L = \frac{5c_3+6}{16} \bar{u}_j + \frac{1-3c_3}{8} \bar{u}_{j+1} + \frac{c_3}{16} \bar{u}_{j+2}$$

$$\begin{aligned} \bar{\omega}_1 &= \frac{148c_3 + 55}{640c_3 + 400} : \frac{1+c_1}{2} u_j + \frac{1-c_1}{2} u_{j+1} = \frac{11-12c_1}{24} \bar{u}_{j-2} + \frac{18c_1-17}{12} \bar{u}_{j-1} + \frac{47-24c_1}{24} \bar{u}_j \\ \bar{\omega}_2 &= \frac{246c_3 + 135}{320c_3 + 200} : \frac{1+c_2}{2} u_j + \frac{1-c_2}{2} u_{j+1} = \frac{-1}{24} \bar{u}_{j-1} + \frac{7+6c_2}{12} \bar{u}_j + \frac{11-12c_2}{24} \bar{u}_{j+1} \end{aligned} \quad (27)$$

$$\bar{\omega}_3 = 1 - \bar{\omega}_1 - \bar{\omega}_2 : \frac{1+c_3}{2} u_j + \frac{1-c_3}{2} u_{j+1} = \frac{11+12c_3}{24} \bar{u}_j + \frac{7-6c_3}{12} \bar{u}_{j+1} + \frac{-1}{24} \bar{u}_{j+2}$$

The weights are no longer symmetric, but it is still true that the central weight $\bar{\omega}_2 > \bar{\omega}_1$ and $\bar{\omega}_2 > \bar{\omega}_3$ and that all three ideal weights are positive for $0 \leq c_3 \leq 1$. Subschemes for the right subcell can be obtained using the fact that $\bar{u}_j^L = \bar{u}_j - \bar{u}_j^R$. The resulting subschemes preserve the weights and LHS coefficients. Subschemes for each boundary segment and corner can be formed by an outer product of appropriate subschemes in the same manner as for the interior subschemes. With these subschemes, quantities in the interior are only coupled with other interior quantities, although it is still necessary to place ghost cells outside the physical domain.

The staggered-grid system complicates the imposition of boundary conditions because the boundary of the staggered grid does not coincide with the physical boundary. This problem is of no consequence for periodic boundary conditions, and if the solution near the boundary is known to be constant in the direction normal to the boundary then a condition of zero gradient in the normal direction can be easily imposed without affecting the behavior of the solution in the interior. More sophisticated boundary conditions will be the subject of future research. For the present work, these two types of boundary conditions that can be straightforwardly implemented still allow a range of test cases that adequately demonstrates the behavior of the CCRWENO method.

Table 1 CCRWENO errors in sinusoid advection

M	N	L^∞ error	L^∞ error order	L^1 error	L^1 error order
20	20	3.26×10^{-4}	-	7.87×10^{-5}	-
40	40	1.04×10^{-5}	4.96	2.27×10^{-6}	5.12
80	80	3.27×10^{-7}	5.00	6.92×10^{-8}	5.03
160	160	9.48×10^{-9}	5.11	2.13×10^{-9}	5.02
320	320	2.62×10^{-10}	5.18	6.63×10^{-11}	5.00

VI. Numerical Tests

A. Linear Advection of a Sinusoid

A sinusoid advects through a unit square with periodic boundary, with constant velocity oriented along a diagonal of the square. The initial condition is:

$$u(x, y, 0) = \sin^2(\pi x) \sin^2(\pi y) \quad (28)$$

The governing equation is:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0 \quad (29)$$

The CCRWENO stencil parameter $d_1 = 1.3$ and the mesh ratio $\lambda = \Delta t / \Delta x = 0.35$. Time steps are taken until $t = 1$ and the solution is resolved on the main grid. Table 1 shows the error behavior for CCRWENO applied to this problem.

Since the solution is smooth and an exact solution is available, we can compare the efficiency of CCRWENO to those of the Levy scheme and the outer-product extension of CWENO5 (see [9]) by considering the computation time required for each to achieve a given error, as in Fig. 3. The compact reconstructions in CCRWENO avoid the need to calculate nine candidates for each subcell average, causing CCRWENO to use less time than the other two methods on the coarsest grids. This speed advantage disappears as the grid is refined due to the expense of solving a linear system, but the error incurred by CCRWENO is sufficiently smaller than those produced by the other schemes that it remains the most efficient choice.

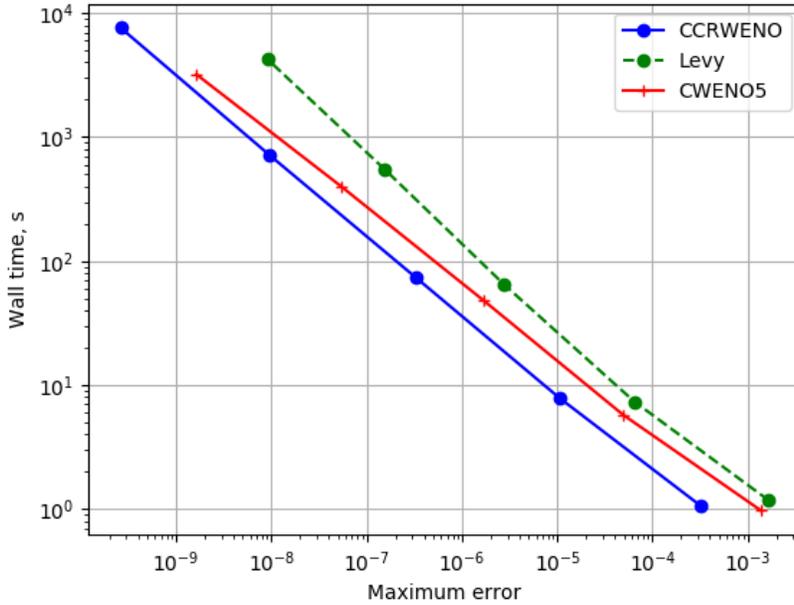


Fig. 3 Efficiency of CCRWENO and Levy schemes for linear wave advection. Solid line: CCRWENO. Dashed line: Levy. Dotted line: CWENO5. Markers correspond to the grids in Table 1.

Table 2 CCRWENO density errors in vortex advection

M	N	L^∞ error	L^∞ error order	L^1 error	L^1 error order
40	40	3.77×10^{-2}	-	1.94×10^{-1}	-
80	80	7.92×10^{-4}	5.57	5.61×10^{-3}	5.11
120	120	9.90×10^{-5}	5.13	8.31×10^{-4}	4.71
160	160	2.27×10^{-5}	5.12	2.10×10^{-4}	4.79
200	200	7.23×10^{-6}	5.12	7.06×10^{-5}	4.87

B. Isentropic Vortex Advection

The isentropic vortex is a smooth exact solution of the two-dimensional Euler equations for an ideal gas. The domain is a square of side length 10, with the vortex initially centered. The freestream velocity at which the vortex travels is $(u_\infty, v_\infty) = (1, 5.5)$. Note that the vortex does not travel in a coordinate direction.

$$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} \rho_\infty \left(1 - \frac{(\gamma-1)w^2}{2\gamma}\right)^{\frac{1}{\gamma-1}} \\ u_\infty - (y-5)w \\ v_\infty + (x-5)w \\ \rho^\gamma \end{pmatrix} \quad (30)$$

$$w = \frac{b}{2\pi} \exp\left(\frac{1-r^2}{2}\right), \quad r = \sqrt{(x-5)^2 + (y-5)^2}$$

The ratio of specific heats is set to its value for air, $\gamma = 1.4$ and the freestream density is $\rho_\infty = 1$. The vortex strength is $b = 5$. The mesh ratio is $\lambda = 0.05$. Table 2 shows the error achieved at time $t = 10$ and confirms fifth-order convergence. Using the exact solution, we can again compare the computational efficiency yielding the results in Fig. 4. CCRWENO is again more efficient than the Levy scheme and CWENO5, except on the coarsest grid.

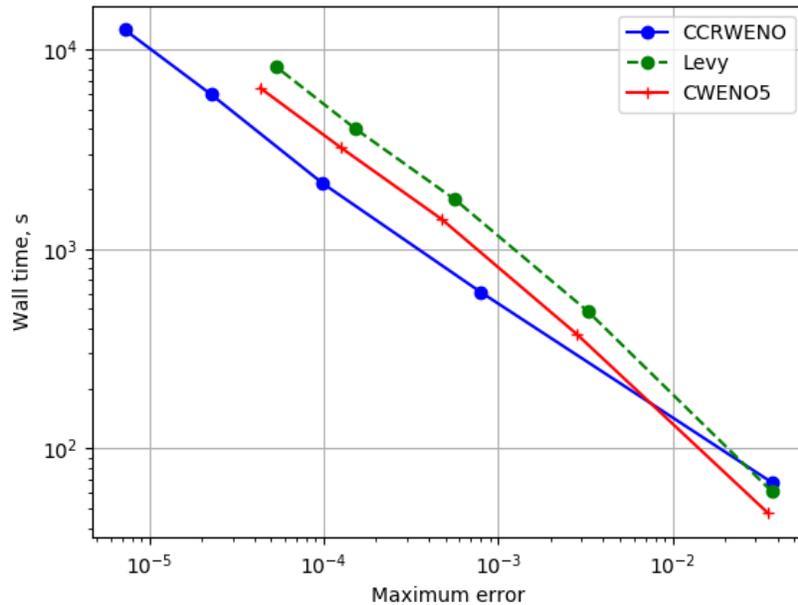


Fig. 4 Efficiency of CCRWENO and Levy schemes for isentropic vortex advection. Solid line: CCRWENO. Dashed line: Levy. Dotted line: CWENO5. Markers correspond to the grids in Table 2.

C. The Lax Problem

Qiu and Shu in [9] use the Lax problem to test the non-oscillatory behavior, and it was the results of the tests with this problem that led them to suggest incorporating a characteristic decomposition into the subcell reconstruction. We repeat their tests here with the CCRWENO scheme. The initial conditions are:

$$(\rho, u, P) = \begin{cases} (0.445, 0.698, 3.528) & x \leq 0 \\ (0.5, 0, 0.571) & x > 0 \end{cases} \quad (31)$$

Figure 5 shows part of the solutions obtained with $\lambda = 0.038$ at time $t = 0.16$. Both the Levy and CCRWENO results show oscillations of similar magnitude near the contact discontinuity, with the CCRWENO scheme producing steeper discontinuities. The oscillations decrease in amplitude as the grid is refined. Compared with Fig. 3 in [9], CCRWENO produces oscillations of similar magnitude and frequency to CWENO5.

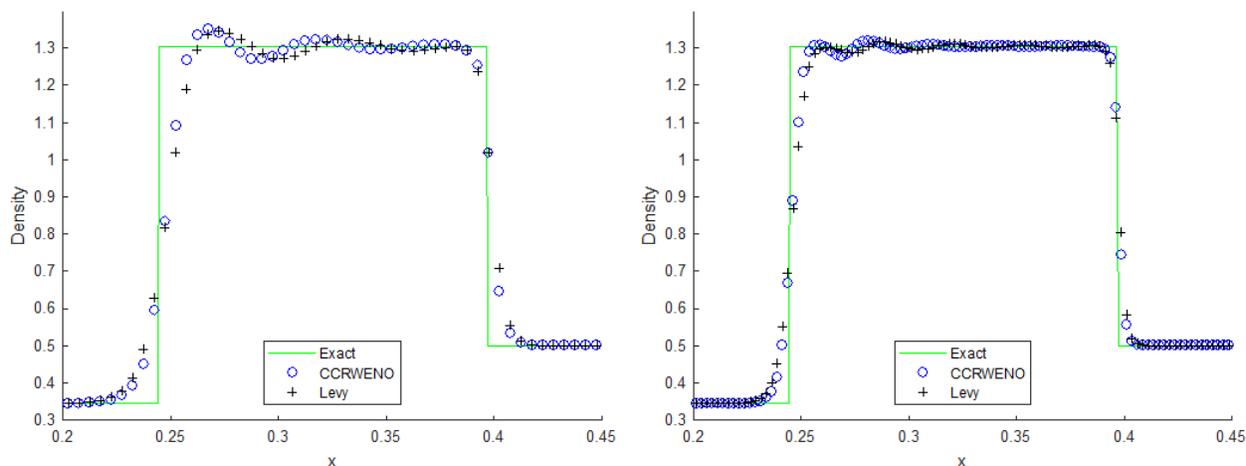


Fig. 5 Solutions to the Lax problem near discontinuities. Left: 200 cells. Right: 400 cells. Circles: CCRWENO solution. Crosses: Solution by the Levy scheme. Solid line: Exact solution

D. The Shu-Osher Problem

A high-order scheme should resolve small-scale features in addition to being non-oscillatory. In the Shu-Osher problem, a shock encounters a density wave producing a wave train behind the shock. Thus the problem contains both a discontinuity and high-frequency waves in close proximity. The initial conditions are:

$$(\rho, u, P) = \begin{cases} \left(\frac{27}{7}, \frac{4\sqrt{35}}{9}, \frac{31}{3}\right) & x \leq 1 \\ (1 + 0.2 \sin(5x), 0, 1) & x > 1 \end{cases} \quad (32)$$

Figure 6 shows the solutions obtained with $\lambda = 0.038$ at time $t = 1.8$ in the region containing the small-scale features. Both the Levy and CCRWENO schemes fail to resolve the oscillations on 200 cells, but detect them on 400 cells with CCRWENO matching the amplitudes much more closely. No spurious oscillations appear due to the shock in either case.

E. Two-Dimensional Riemann Problems

The next two problems demonstrate the ability of the CCRWENO scheme to handle discontinuous solutions without a Riemann solver. In the first case, the initial condition is piecewise constant in the four quadrants of the unit square domain with the values chosen to produce contact discontinuities at the interfaces (Configuration 5 in [8] and [15]):

The solution obtained on a 200×200 grid with $\lambda = 0.1247$ is shown at $t = 0.23$ in Fig. 8. The discontinuities have little smearing and the two rolled-up contact discontinuities are clearly resolved (cf. Fig. 4.9 in [8]). Some oscillations can be seen leaving the domain in the lower-right quadrant. In the second case, the values are chosen

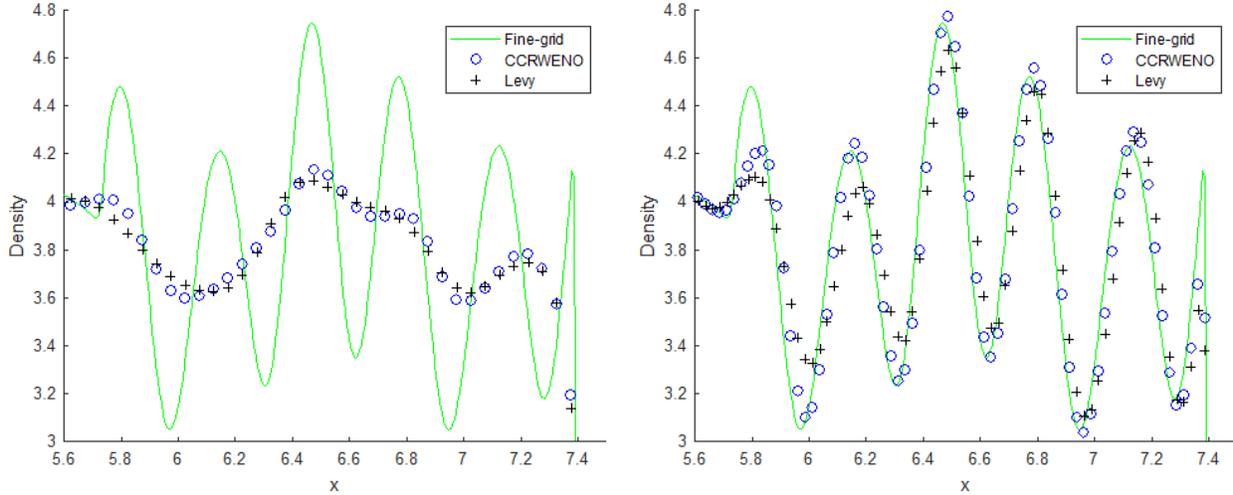


Fig. 6 Solutions to the Shu-Osher problem near discontinuities. Left: 200 cells. Right: 400 cells. Circles: CCRWENO solution. Crosses: Solution by the Levy scheme. Solid line: CRWENO solution on 1600 cells

$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 2 \\ -0.75 \\ 0.5 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 1 \\ -0.75 \\ -0.5 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 1.0222 \\ -0.6179 \\ 0.1 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 0.5313 \\ 0.1 \\ 0.1 \\ 0.4 \end{pmatrix}$
$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 1 \\ 0.75 \\ 0.5 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 3 \\ 0.75 \\ -0.5 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 0.8 \\ 0.1 \\ 0.1 \\ 1 \end{pmatrix}$	$\begin{pmatrix} \rho \\ u \\ v \\ P \end{pmatrix} = \begin{pmatrix} 1 \\ 0.1 \\ 0.8276 \\ 1 \end{pmatrix}$

Fig. 7 Initial conditions for the two-dimensional Riemann problems. Left: Configuration 5. Right: Configuration 16.

to produce a rarefaction, a shock, and two contact discontinuities (Configuration 16 in [8] and [15]). The solution obtained on a 400×400 grid with $\lambda = 0.1247$ is shown at $t = 0.2$ in Fig. 9. As the shock near $y = 0.8$ moves in the $+y$ direction, it generates oscillations whose sizes are not apparent from the contour plot. A cross-section through these oscillations shows their small amplitude - most of the oscillations indicated by the contours are barely visible if at all in the cross-section. The two-dimensional extension of CWENO5 produces a similar train of oscillations, shown in Fig. 10, which suggests that they are consequences of the higher order of accuracy common to CCRWENO and CWENO5.

F. Advection of High-Frequency Waves

Based on the nature of CRWENO, we expect the compact reconstructions used in CCRWENO to improve the resolution of small-scale features. As a test we consider advection of sinusoidal density waves of increasingly high frequency on a fixed 150×150 grid. The initial condition is:

$$\begin{aligned} \rho(x, y) &= 2 + \sin(2\pi kx) \sin(2\pi ky) \\ u &= 1 \\ v &= 0 \\ P &= 1 \end{aligned} \tag{33}$$

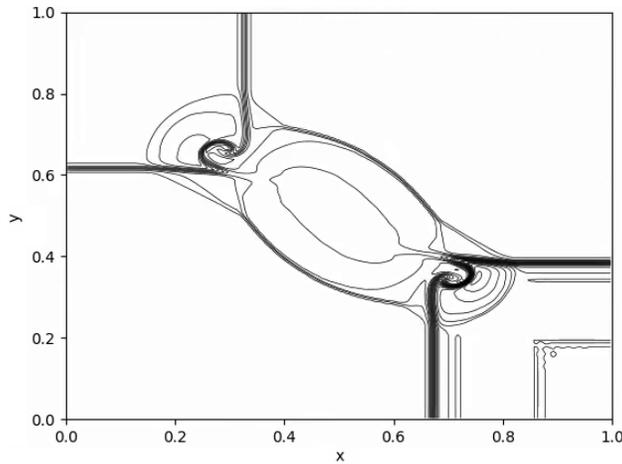


Fig. 8 Density contours of the CCRWENO solution of Configuration 5

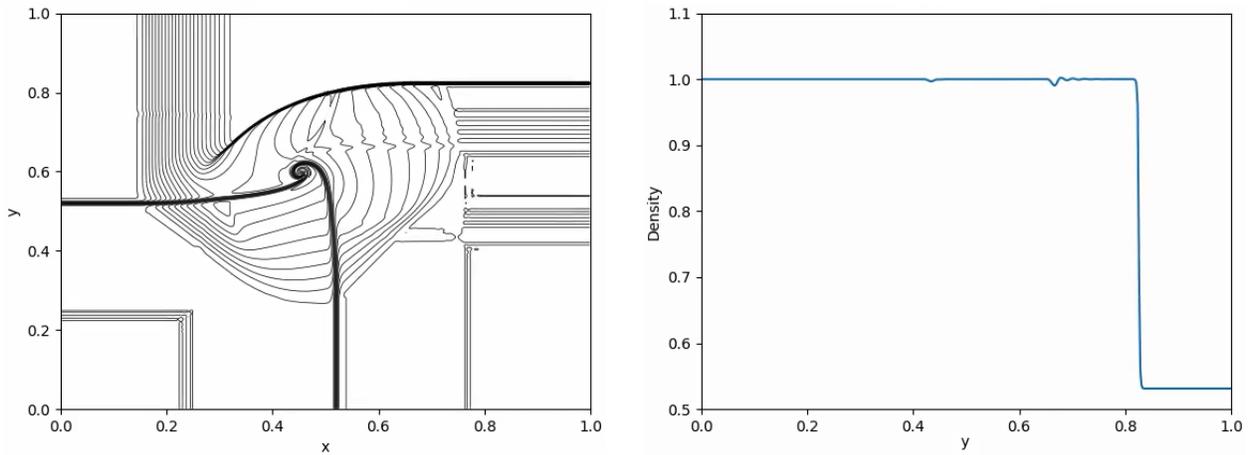


Fig. 9 Left: Density contours of the CCRWENO solution of Configuration 16. Right: Cross-section of density at $x = 0.9$.

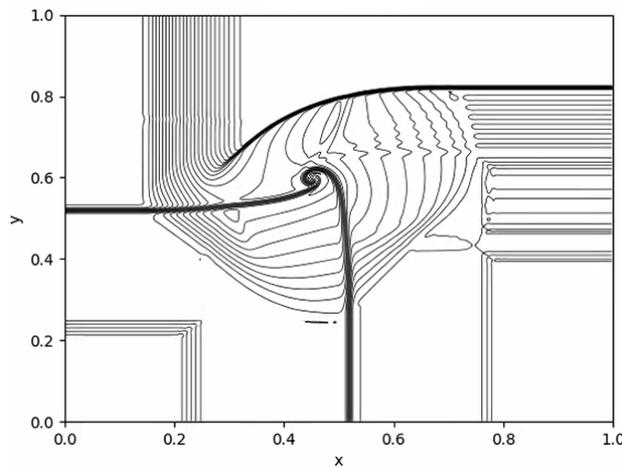


Fig. 10 Density contours of the CWENO5 solution of Configuration 16.

which evolves according to the Euler equations. The domain is a square of side length 1 and the mesh ratio is $\lambda = 0.35$. Figure 11 shows the errors incurred for each wavenumber k with CCRWENO, the Levy scheme, and CWENO5. Sudden dips in the error occur at wavenumbers of $k = 30, 50,$ and 60 with each method, corresponding to cases where either the five-point stencil of the combined scheme or a three-point substencil contain an integer number of wavelengths. In such cases the combined point-value reconstruction becomes exact for a sinusoidal solution so less error accumulates per time step. CCRWENO either matches or considerably outperforms the other two schemes at each wavenumber.

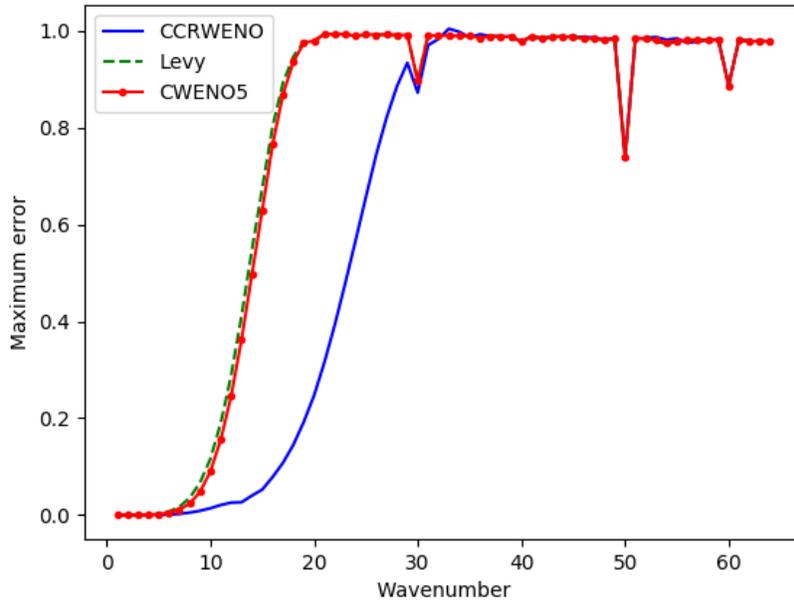


Fig. 11 Maximum error by wavenumber. Solid line: error incurred with CCRWENO. Dashed line: Levy scheme. Dotted line: CWENO5.

VII. Conclusion

We developed a central compact-reconstruction WENO method which avoids the need for either a Riemann solver or reconstruction in characteristic variables. The compact nature of the subschemes allowed weights to be reused between the subcell-average and point-value reconstructions, without compromising the desired fifth-order accuracy as demonstrated in numerical tests. Furthermore, the subschemes were designed so that the systems of equations that arise in the two reconstructions have the same coefficient matrices. This choice saves enough computational expense that the CCRWENO scheme is more efficient than the non-compact scheme of Levy et al. [4] and CWENO5 which use the same cells. The compact reconstructions also improve resolution of waves over a broad range of wavenumbers. Numerical tests demonstrated the ability of the CCRWENO scheme to capture behavior of two-dimensional Riemann problems without any Riemann solver.

The CCRWENO scheme is moderately successful at addressing the three motivations for this investigation. First, the need for characteristic variables seems to be only somewhat mitigated. The CCRWENO results on the Lax problem are about as oscillatory as those from the Levy scheme and slightly more so than those from CWENO5. That test problem is the most challenging, however, and CCRWENO produces acceptable results on the Shu-Osher problem and the two-dimensional Riemann problems. If one deems the characteristic transformation unnecessary, then CCRWENO is more efficient than CWENO5. Secondly, the need for a Riemann solver is thoroughly eliminated as demonstrated by the ability of CCRWENO to effortlessly capture two-dimensional physics in the two-dimensional Riemann problems. Disregarding the oscillations, its results on the Lax problem are clearly reminiscent of the true solution in that the rarefaction and discontinuities appear at the correct locations and have nearly the correct magnitudes. Finally, the suspicion that inconsistency between the subcell-average and point-value reconstructions in CWENO5 forces a characteristic decomposition is dispelled. The behavior of CCRWENO on the Lax problem is qualitatively the same as that which led Qiu and Shu to employ characteristic variables. The oscillations appearing in the two-dimensional Riemann problems also raise skepticism as to whether CCRWENO really can do without that transformation. On the

other hand, the Levy scheme is hardly better on the Lax problem yet shows no oscillations in the two-dimensional Riemann problems [8]. That CWENO5 produces similar oscillations strongly suggests that the smaller dissipation attendant with high-order accuracy is indeed responsible.

The outer-product approach for extending a one-dimensional scheme to multiple dimensions applies to any schemes for the subcell averages and point values. Therefore, arbitrary compact and non-compact schemes can be combined to modify the coupling pattern between the unknowns. Judicious choice of the schemes can yield CCRWENO variants that, for example, benefit more from parallelization. Such variants may lose the advantage of identical coefficient matrices, however. We reiterate that once an acceptable set of one-dimensional subschemes is found, the extension to any number of dimensions is immediate and preserves the order of accuracy.

The boundary treatment described here is limited to boundary conditions that arise infrequently in applications. This situation could be remedied by replacing the staggered grid with an expanded grid that overlays the main grid and includes one layer of subcells outside the physical domain. The alternation between main and staggered grids would then be replaced by alternation between the main grid, which coincides with the physical domain, and the expanded grid. This approach would require the boundary closure to supply subcell averages for the subcells just outside the physical domain and point values on its boundary.

Strictly speaking, we did not ensure that the underlying reconstruction polynomial is the same in the two reconstruction steps, only that the weights coincide which is a necessary but perhaps not sufficient condition. Each subscheme in Eq. (12) and Eq. (13) is satisfied by infinitely many cubic polynomials that also match the relevant cell averages. It may be possible to choose one such polynomial by some criterion (e.g. minimum total variation, minimum maximum norm), then combine them in a WENO-like manner to design central WENO schemes with desirable properties.

Acknowledgments

This work was supported under the Vertical Lift Research Center of Excellence Grant at the University of Maryland with Dr. Mahendra Bhagwat as Technical Mentor.

References

- [1] Liu, X.-D., Osher, S., and Chan, T., "Weighted essentially non-oscillatory schemes," *Journal of Computational Physics*, Vol. 115, No. 1, 1994, pp. 200–212. doi:10.1006/jcph.1994.1187.
- [2] Ghosh, D., and Baeder, J. D., "Compact reconstruction schemes with weighted ENO limiting for hyperbolic conservation laws," *SIAM Journal on Scientific Computing*, Vol. 34, No. 3, 2012, pp. A1678–A1706. doi:10.1137/110857659.
- [3] Nessyahu, H., and Tadmor, E., "Non-oscillatory central differencing for hyperbolic conservation laws," *Journal of Computational Physics*, Vol. 87, No. 2, 1990, pp. 408–463. doi:10.1016/0021-9991(90)90260-8.
- [4] Levy, D., Puppo, G., and Russo, G., "Central WENO schemes for hyperbolic systems of conservation laws," *ESAIM: Mathematical Modelling and Numerical Analysis*, Vol. 33, No. 3, 1999, pp. 547–571. doi:10.1051/m2an:1999152.
- [5] Bianco, F., Puppo, G., and Russo, G., "High-order central schemes for hyperbolic systems of conservation laws," *SIAM Journal on Scientific Computing*, Vol. 21, No. 1, 1999, pp. 294–322. doi:10.1137/S1064827597324998.
- [6] Levy, D., Puppo, G., and Russo, G., "A third order central WENO scheme for 2D conservation laws," *Applied Numerical Mathematics*, Vol. 33, No. 1-4, 2000, pp. 415–421. doi:10.1016/S0168-9274(99)00108-7.
- [7] Levy, D., Puppo, G., and Russo, G., "Compact central WENO schemes for multidimensional conservation laws," *SIAM Journal on Scientific Computing*, Vol. 22, No. 2, 2000, pp. 656–672. doi:10.1137/S1064827599359461.
- [8] Levy, D., Puppo, G., and Russo, G., "A fourth-order central WENO scheme for multidimensional hyperbolic systems of conservation laws," *SIAM Journal on Scientific Computing*, Vol. 24, No. 2, 2002, pp. 480–506. doi:10.1137/S1064827501385852.
- [9] Qiu, J., and Shu, C.-W., "On the construction, comparison, and local characteristic decomposition for high-order central WENO schemes," *Journal of Computational Physics*, Vol. 183, No. 1, 2002, pp. 187–209. doi:10.1006/jcph.2002.7191.
- [10] Jiang, G.-S., and Shu, C.-W., "Efficient Implementation of Weighted ENO Schemes," *Journal of Computational Physics*, Vol. 126, 1996, pp. 202–228. doi:10.1006/jcph.1996.0130.
- [11] Lele, S. K., "Compact finite difference schemes with spectral-like resolution," *Journal of Computational Physics*, Vol. 103, No. 1, 1992, pp. 16–42. doi:10.1016/0021-9991(92)90324-R.

- [12] Shi, J., Hu, C., and Shu, C.-W., "A technique of treating negative weights in WENO schemes," *Journal of Computational Physics*, Vol. 175, No. 1, 2002, pp. 108–127. doi:10.1006/jcph.2001.6892.
- [13] Zennaro, M., "Natural continuous extensions of Runge-Kutta methods," *Mathematics of Computation*, Vol. 46, No. 173, 1986, pp. 119–133. doi:10.1090/S0025-5718-1986-0815835-1.
- [14] Shu, C.-W., "A survey of strong stability preserving high order time discretizations," *Collected Lectures on the Preservation of Stability Under Discretization*, Vol. 109, 2002, pp. 51–65.
- [15] Lax, P. D., and Liu, X.-D., "Solution of two-dimensional Riemann problems of gas dynamics by positive schemes," *SIAM Journal on Scientific Computing*, Vol. 19, No. 2, 1998, pp. 319–340. doi:10.1137/S1064827595291819.