ICES REPORT 17-20

August 2017

A Multilevel-WENO Technique for Solving Nonlinear Conservation Laws

by

Todd Arbogast, Chieh-Sen Huang, and Xikai Zhao

The Institute for Computational Engineering and Sciences
The University of Texas at Austin
Austin, Texas 78712

Reference: Todd Arbogast, Chieh-Sen Huang, and Xikai Zhao, "A Multilevel-WENO Technique for Solving Nonlinear Conservation Laws," ICES REPORT 17-20, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, August 2017.

A MULTILEVEL-WENO TECHNIQUE FOR SOLVING NONLINEAR CONSERVATION LAWS*

TODD ARBOGAST†, CHIEH-SEN HUANG‡, AND XIKAI ZHAO§

Abstract. We present a multi-level WENO polynomial reconstruction technique. Polynomials of various orders defined over various size and shifted stencils may be combined by using novel weighting strategy. Numerical results show that the technique provides some improvement over standard WENO reconstructions.

Key words. Weighted essentially non-oscillatory, polynomial reconstruction, hyperbolic conservation law

AMS subject classifications. 65M08, 65M06, 76M12, 76M20

1. Introduction. We consider finite volume and finite difference methods for systems of nonlinear, hyperbolic conservation laws of the form

$$u_t + f(x)_x = 0, \quad t > 0,\ x \in \mathbb{R},\ u \in \mathbb{R}^d,\ d \geq 1. \tag{1.1}$$

(For simplicity, we do not treat boundary conditions, although we might also consider a bounded space domain if we impose periodic boundary conditions.) Weighted essentially non-oscillatory (WENO) schemes are a popular choice for solving such systems of equations. They allow one to reconstruct a high order version of the solution merely from approximations of cell averages (in finite volume schemes) or point values (in finite difference schemes). The key is to average approximations defined on various stencils, and to weight them so as to avoid stencils containing a shock discontinuity in the solution.

We present a novel multi-level WENO polynomial reconstruction technique. Polynomials of various orders defined over various size and shifted stencils may be combined by using novel weighting strategy. Numerical results in 1-D for scalar problems and Euler systems show that the technique provides some improvement over standard WENO reconstructions.

2. Standard WENO framework. To establish the context for our new multilevel WENO technique and to fix notation, we review the standard theory. For simplicity of exposition, we consider the finite volume framework. The finite difference framework is similar.

Partition time as $0 = t^0 < t^1 < \cdots < t^n$. In the domain of (1.1), take arbitrary grid points $\cdots < x_{-1} < x_0 < x_1 < \cdots$, and define the cell $I_i = [x_i, x_{i+1}]$, and the length of the cell $\Delta x_i = x_{i+1} - x_i$. The mid point of I_i is denoted as $x_{i+\frac{1}{2}} = \frac{1}{2}(x_i + x_{i+1})$. Let $\bar{u}_i^n$ be the numerically approximated average of u on the cell I_i at

*This work was supported by the U.S. National Science Foundation under grant DMS-1418752.

†Department of Mathematics, University of Texas, 2515 Speedway, C1200, Austin, TX 78712-1202 and Institute for Computational Engineering and Sciences, University of Texas, 201 East 24th St., C0200, Austin, TX 78712-1229 (arbogast@ices.utexas.edu)

‡ Department of Applied Mathematics and National Center for Theoretical Sciences, National Sun Yat-sen University, Kaohsiung 804, Taiwan, R.O.C. (huangcs@math.nsysu.edu.tw)

§Department of Mathematics, University of Texas, 2515 Speedway, C1200, Austin, TX 78712-1202 and Institute for Computational Engineering and Sciences, University of Texas, 201 East 24th St., C0200, Austin, TX 78712-1229 (xzhao@math.utexas.edu)

time t^n, so

$$\bar{u}_i^n \approx \frac{1}{\Delta x_i} \int_{I_i} u(x, t^n)\, dx.$$

Now assume u is smooth. For a kth order approximation of u on the given cell I_i, we consider the ordered stencil S_i, which contains r cells to the left and s cells to the right of I_i with $r + s + 1 = k$, i.e.,

$$S_i = \{I_{i-r},\ \dots\ , I_i\ ,\ \dots\ , I_{i+s}\}, \quad r + s + 1 = k, \tag{2.1}$$

from which we obtain the kth order reconstruction polynomial P_i by imposing the interpolation conditions

$$\frac{1}{\Delta x_m} \int_{I_m} P_i(x)\, dx = \bar{u}_m, \quad \text{for all } I_m \in S_i.$$

2.1. Background on WENO Reconstruction. Given the cell I_i, suppose we are interested in a $(2k-1)st$ general WENO reconstruction. From now and for the remainder of the paper, we fix a value of i and drop it from the notation. First consider all the small stencils S_j^k containing I_i and with k cells ($k \geq 2$)

$$S_j^k = \{I_{i-k+j+1},\ \dots\ , I_{i+j}\}, \quad 0 \leq j \leq k-1.$$

From each S_j^k we can reconstruct P_j^{k-1} of degree $k-1$. Moreover, we can define the large stencil $S^{2k-1} = \bigcup_j S_j^k$ and reconstruct a higher order polynomial P^{2k-2} of degree $(2k-2)$. At a fixed point x, the polynomial P^{2k-2} can usually be written as a convex combination of P_j^{k-1}, so

$$P^{2k-2}(x) = \sum_j \alpha_j P_j^{k-1}(x), \quad \sum_j \alpha_j = 1. \tag{2.2}$$

where we drop the subscript i. We refer to α_j as a *linear weight*.

When there are discontinuities in the data over stencil S_i, we want to make use of the relatively small stencils on which u is smooth in order to achieve the essentially non-oscillatory property. The standard WENO reconstruction is

$$R(x) = \sum_j \tilde{\alpha}_j P_j^{k-1}(x), \quad \sum_j \tilde{\alpha}_j = 1, \tag{2.3}$$

where

$$\tilde{\alpha}_j = \frac{\hat{\alpha}_j}{\sum_m \hat{\alpha}_m}, \quad \hat{\alpha}_j = \frac{\alpha_j}{(\epsilon + \sigma_{P_j^{k-1}})^\tau}, \quad \text{for all } j, \tag{2.4}$$

for some smoothness indicator $\sigma_{P_j^{k-1}}$ and exponent τ (usually $\tau = 2$ and $\epsilon = 10^{-6} > 0$ avoids any possibility of division by zero). We refer to $\tilde{\alpha}_j$ as a *nonlinear weight*.

The smoothness indicator σ defined by Jiang and Shu in [1] is normally used to measure the smoothness of the reconstruction polynomials on the cell I_i. It is given by

$$\sigma_{P_j^k} = \sum_{m=1}^{k} \int_{x_i}^{x_{i+1}} (\Delta x_i)^{2m-1} \left(\frac{d^m P_j^k(x)}{dx^m} \right)^2 dx, \tag{2.5}$$

where again k is the degree of the polynomial P_j^k. If the grid is uniform in the cell size $\Delta x = h$, then in the regions where u is smooth, the Taylor expansion of (2.5) gives

$$\sigma_{P_j^k} = (u'h)^2(1 + \mathcal{O}(h^2)),$$

which implies $\tilde{\alpha}_j = \alpha_j + \mathcal{O}(h^2)$. If there are discontinuities in u within some of the stencils S_j^k, then $\sigma_{P_j^k} = \mathcal{O}(1)$. So the weights of the smooth and non-smooth stencils are $\mathcal{O}(1)$ and $\mathcal{O}(h^{2\tau})$, respectively. This leads to an oder of accuracy of the reconstruction R decreasing to k.

2.2. Levy-Puppo-Russo Reconstructions. In [3], Levy, Puppo and Russo describe a third order compact, central WENO scheme. They use a somewhat different WENO reconstruction than the standard one (2.3), because the linear weights in (2.2) fail to exist when $x = x_{i+\frac{1}{2}}$. For the given cell I_i, reconstruct the *optimal* quadratic polynomial P^2 using the large stencil $S^3 = \{I_{i-1}, I_i, I_{i+1}\}$ and two linear polynomials P_0^1 and P_1^1 from the stencils $S_0^2 = \{I_{i-1}, I_i\}$ and $S_1^2 = \{I_i, I_{i+1}\}$, respectively. The idea is to introduce a centered quadratic polynomial $p_{\rm c}$ such that

$$P^2 = \alpha_0 P_0^1 + \alpha_1 P_1^1 + \alpha_{\rm c} p_{\rm c}, \quad \sum_j \alpha_j = 1,\ \alpha_j \geq 0,\ j \in \{0, {\rm c}, 1\},$$

$$p_c = \frac{1}{\alpha_c}(P^2 - \alpha_0 P_0^1 - \alpha_1 P_1^1),$$

where α_j are the linear weights, subject only to the symmetry $\alpha_0 = \alpha_1$. Then the WENO reconstruction is

$$R = \tilde{\alpha}_0 P_0^1 + \tilde{\alpha}_1 P_1^1 + \tilde{\alpha}_{\rm c} p_{\rm c}, \quad \tilde{\alpha}_j = 1,\ \tilde{\alpha}_j \geq 0,\ j \in \{0, {\rm c}, 1\},$$

and the nonlinear weights $\tilde{\alpha}_j$ are computed by (2.4) with $\tau = 2$ and $\epsilon = 10^{-2}$. Here ϵ needs to satisfy: 1) $\epsilon \gg \sigma$ in smooth region, 2) $\epsilon \ll \sigma$ near discontinuities.

3. Multilevel-WENO. In this section, we present the multilevel-WENO scheme in one space dimension.

We proceed as if u were smooth. For the kth order reconstruction ($k \geq 3$), consider the full stencil S_i as in (2.1). We will have hierarchically smaller stencils. Consider the small stencils at *level* ℓ, where $1 \leq \ell \leq k-1$. We can construct an $(\ell+1)st$ order accurate reconstruction using the stencils $S_j^{\ell+1}$ with $(\ell+1)$ cells contained in S_i. Define

$$S_j^{\ell+1} = \{I_{i-\ell+j}, ..., I_{i+j}\} \subset S, \quad \max(0, \ell - r) \leq j \leq \min(\ell, s),$$

from which we obtain the $(\ell+1)$st order reconstruction polynomial P_j^ℓ.

Then the kth order multilevel-WENO reconstruction is given by the following algorithm, where the nonlinear weights are computed by (2.4) with corresponding linear weights and smoothness indicators, to be defined after presenting the algorithm.

Step 1. For $\ell = 1$, reconstruct P_0^1 from the stencil S_0^2, and P_1^1 from the stencil S_1^2. Then combine these two polynomials to obtain the second order reconstruction R^1

$$R^1 = \tilde{\gamma}_0^1 P_0^1 + \tilde{\gamma}_1^1 P_1^1, \quad \gamma_0^1 + \gamma_1^1 = 1,\ \gamma_0^1, \gamma_0^1 \geq 0,$$

with linear weights γ_j^1 and smoothness indicators $\sigma_{P_j^1}$. Perhaps the unbiased choice $\gamma_0^1 = \gamma_1^1 = \frac{1}{2}$ is most reasonable.

Step 2. For $\ell \geq 2$, first construct the $(\ell+1)$th order polynomial P_j^ℓ from the stencil $S_j^{\ell+1}$ for all j. Then following Levy-Puppo-Russo's idea [3], the reconstruction R_j^ℓ will be a convex combination of $R^{\ell-1}$ and the centered polynomial

$$p_{j,\mathrm{c}}^\ell = \frac{1}{\beta_j^\ell}(P_j^\ell - \alpha_j^\ell R^{\ell-1}),$$

that is,

$$R_j^\ell = \tilde{\alpha}_j^\ell R^{\ell-1} + \tilde{\beta}_j^\ell p_{j,\mathrm{c}}^\ell, \quad \alpha_j^\ell + \beta_j^\ell = 1,\ \alpha_j^\ell,\ \beta_j^\ell \geq 0,\ \text{all } j,$$

with linear weights α_j^ℓ, β_j^ℓ and using the modified smoothness indicators given below. We generally take $\alpha_j^\ell = \frac{1}{5}$ and $\beta_j^\ell = \frac{4}{5}$.

Step 3. If $\ell = k-1$, then S_0^k is the same as the entire stencil S_i, so we stop here. If $\ell < k-1$, then again, using the WENO methodology, we combine all the reconstructions R_j^ℓ to get the final $(\ell+1)$th reconstruction R^ℓ at level ℓ:

$$R^\ell = \sum_j \tilde{\gamma}_j^\ell R_j^\ell, \quad \sum_j \gamma_j^\ell = 1,\ \gamma_j^\ell \geq 0,\ \text{all } j,$$

with linear weights γ_j and smoothness indicators $\sigma_{P_j^\ell}$.

3.1. Smoothness indicators. We now define the smoothness indicators

1. For $2 \leq \ell \leq k-2$.
   (a) If $j = \max(0, \ell - r)$, then

$$\begin{cases} \sigma_{j,R}^\ell = \sqrt{\sigma_{R^{\ell-1}}\sigma_{P_j^\ell}}\sigma_{P_{j+1}^\ell - P_j^\ell}, \\ \sigma_{j,p}^\ell = \sigma_{P_j^\ell}\sigma_{P_{j+1}^\ell - P_j^\ell}. \end{cases} \tag{3.1}$$

   (b) If $\max(0, \ell - r) < j < \min(\ell, s)$, then

$$\begin{cases} \sigma_{j,R}^\ell = \sqrt{\sigma_{R^{\ell-1}}\sigma_{P_j^\ell}}\sqrt{\sigma_{P_j^\ell - P_{j-1}^\ell}\sigma_{P_{j+1}^\ell - P_j^\ell}}, \\ \sigma_{j,p}^\ell = \sigma_{P_j^\ell}\sqrt{\sigma_{P_j^\ell - P_{j-1}^\ell}\sigma_{P_{j+1}^\ell - P_j^\ell}}. \end{cases} \tag{3.2}$$

   (c) If $j = \min(\ell, s)$, then

$$\begin{cases} \sigma_{j,R}^\ell = \sqrt{\sigma_{R^{\ell-1}}\sigma_{P_j^\ell}}\sigma_{P_j^\ell - P_{j-1}^\ell}, \\ \sigma_{j,p}^\ell = \sigma_{P_j^\ell}\sigma_{P_j^\ell - P_{j-1}^\ell}. \end{cases} \tag{3.3}$$

2. For $\ell = k-1$.

$$\begin{cases} \sigma_{0,R}^{k-1} = \sqrt{\sigma_{R^{k-2}}\sigma_{P_0^{k-1}}}, \\ \sigma_{0,p}^{k-1} = \sigma_{P_0^{k-1}}. \end{cases} \tag{3.4}$$

3.2. Level 0. Notice that for the above algorithm, we started from $\ell = 1$. So to add in the level $\ell = 0$, we can first obtain the reconstruction R^0 and then implement Step 2 to Step 5 recursively starting from $\ell = 1$. We will describe the method for reconstructing R^0 below.

When $\ell = 0$, we only have one stencil $S_0^1 = \{I_i\}$ from which we can obtain a first order reconstruction polynomial $P_0^0 = \bar{u}_i$. We will reconstruct R^0 in two steps.

1. First, reconstruct P_0^1 from the stencil S_0^2 and P_1^1 from the stencil S_1^2. Then combine P_0^0 and P_j^1 to get R_j^0

$$R_j^0 = \tilde{\alpha}_j^0 P_0^0 + \tilde{\beta}_j^0 P_j^1,\ j = 0, 1,$$

where the nonlinear weights are defined by (2.4) with smoothness indicators $\sigma_{P_0^0}$ and $\sigma_{P_j^1}$. If we use (2.5) to compute the smoothness indicator $\sigma_{P_0^0}$, then $\sigma_{P_0^0} = 0$. This will result in the nonlinear weights being highly biased to $\tilde{\alpha}_j^0$, and therefore we lose accuracy. We want to define $\sigma_{P_0^0}$ such that it has order $\mathcal{O}(h^2)$ just like the other smoothness indicators in the smooth region, while maintaining the higher order polynomials when the data fits. So define

$$\sigma_{P_0^0} = \frac{1}{4}\sigma_{P_0^1+P_1^1} = \frac{1}{4}(\bar{u}_{i+1} - \bar{u}_{i-1})^2. \tag{3.5}$$

2. Next, combine the reconstructions R_0^0 and R_1^0 to get

$$R^0 = \tilde{\gamma}_0^0 R_0^0 + \tilde{\gamma}_1^1 R_1^0,$$

and the nonlinear weights are computed by (2.4) with linear weight γ_j and smoothness indicator $\sigma_{P_j^1}$.

Remark. Suppose instead of using the above method, we let $R^0 = P_0^0$. Then in Step 2 we have

$$R_j^1 = \tilde{\alpha}_j^1 R^0 + \tilde{\beta}_j^1 p_j^1, \quad j = 0, 1,$$

where we use the smoothness indicator $\sigma_{R^0} = \sigma_{P_0^0}$ given by (3.5) and $\sigma_{P_j^1}$ to compute nonlinear weights. Near the shock, in theory we expect this modified method to cause oscillations. For instance, when there are shocks on one side of the cell I_i, say $\sigma_{P_0^1} = \mathcal{O}(h^2)$, $\sigma_{P_1^1} = \mathcal{O}(1)$, then σ_{R^0} may be much larger than $\sigma_{P_0^1}$ so that the nonlinear weights bias to $\tilde{\beta}_j^1$ and $R^0 \approx p_j^1$. This will cause overshooting or undershooting near the shocks. However, this method seems to give slightly better results than the original method given above for the examples we tested in this paper.

3.3. Reduction. When implement the algorithm, we can choose which levels to use rather than using all the levels. For example, for the fifth order scheme, we can choose level 2 to level 4 simply by setting $\alpha_j^2 = 0$, for all j, or only use level 2 and level 4 by setting $\tilde{\alpha}_j^3 = 1$ and $\tilde{\beta}_j^3 = 0$.

4. Numerical results in One Dimensional Space. In our one dimensional tests, the L^1 and L^∞ errors are computed by the formulas

$$\sum_i \left| \frac{1}{\Delta x_i} \int_{I_i} u(x, t^n)\, dx - \bar{u}_i^n \right| \Delta x_i \quad (L^1\text{error}), \tag{4.1}$$

$$\max_i \left| \frac{1}{\Delta x_i} \int_{I_i} u(x, t^n)\, dx - \bar{u}_i^n \right| \quad (L^\infty\text{error}). \tag{4.2}$$

In all of the tests in this paper, for both third order and fifth order schemes, we use $\alpha_j^\ell = \frac{1}{5}$, $\beta_j^\ell = \frac{4}{5}$ for all ℓ and $\gamma_0^1 = \gamma_1^1 = \frac{1}{2}$. For the fifth order scheme, use $\gamma_0^2 = \gamma_2^2 = \frac{1}{4}$ and $\gamma_1^2 = \frac{1}{2}$. We will not include level 0 in our tests.

4.1. Linear advection. We first test $u_t + u_x = 0$ with periodic boundary conditions on $[0, 2]$ and initial condition $u_0(x) = \sin(\pi x)$, at time $T = 10$ on both uniform and nonuniform grids using the third order ML-WENO scheme with $\epsilon = 10^{-2}$, see Tables 4.1–4.2, and the fifth order ML-WENO scheme with $\epsilon = 10^{-6}$, see Tables 4.3–4.4.

We show the results of *Shu's linear test* in Figures 4.1–4.2.

Finally, we test the initial condition with a sawtooth pattern on a uniform grid on $(0, 1)$ with 10 cells; that is, $\bar{u}_i^0 = 0$ if i is odd and $\bar{0}_i^0 = 1$ if i is even. The results are shown in Figure 4.3.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.308927	——	0.154047	——
40	0.025662	3.5896	0.011913	3.6928
80	0.0032286	2.9906	0.0012765	3.2222
160	0.00040404	2.9984	0.00015872	3.0076
320	5.0501e-05	3.0001	1.9836e-05	3.0003
640	6.3122e-06	3.0001	2.4791e-06	3.0003
1280	7.8898e-07	3.0001	3.0985e-07	3.0002

Table 4.1: Linear advection. ML-WENO3 error and convergence rate on the uniform grid at $T = 10$, with $\epsilon = 10^{-2}$, $\Delta t = 0.5\Delta x$.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.332156	——	0.163285	——
40	0.026169	3.6659	0.013576	3.5882
80	0.0033471	2.9669	0.0013484	3.3318
160	0.00041517	3.0111	0.00016337	3.045
320	5.2211e-05	2.9913	2.0557e-05	2.9904
640	6.4707e-06	3.0124	2.544e-06	3.0145
1280	8.0959e-07	2.9987	3.1823e-07	2.999

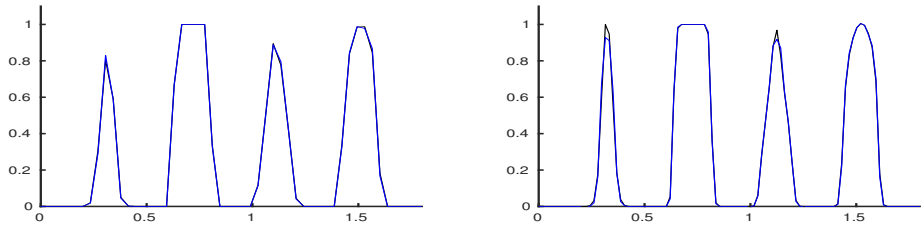
Table 4.2: Linear advection. ML-WENO3 error and convergence rate on the nonuniform grid at $T = 10$, with $\epsilon = 10^{-2}$.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.00198609	——	0.00161003	——
40	6.3369e-05	4.97	4.9828e-05	5.014
80	1.9902e-06	4.9928	1.5632e-06	4.9944
160	6.2287e-08	4.9979	4.8901e-08	4.9985
320	1.9479e-09	4.9989	1.5295e-09	4.9987
640	6.1026e-11	4.9964	4.7962e-11	4.9951
1280	1.9237e-12	4.9875	1.6108e-12	4.896

Table 4.3: Linear advection. ML-WENO5 error and convergence rate on the uniform grid at $T = 10$, with $\epsilon = 10^{-6}$, $\Delta t = 0.2\Delta x$.

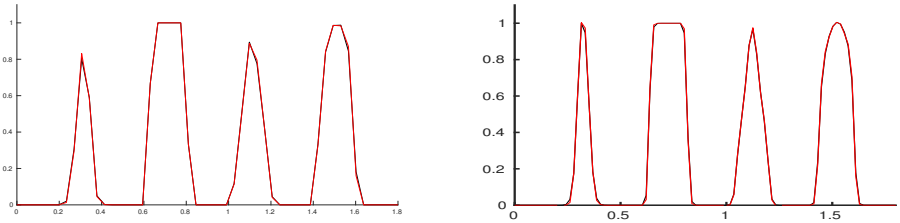
N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.00200827	——	0.00201253	——
40	6.4563e-05	4.9591	5.0791e-05	5.3083
80	2.0459e-06	4.9799	1.6069e-06	4.9822
160	6.3382e-08	5.0125	4.9769e-08	5.0129
320	1.9865e-09	4.9958	1.5602e-09	4.9954
640	6.2455e-11	4.9912	4.9099e-11	4.9899
1280	1.9726e-12	4.9846	1.559e-12	4.977

Table 4.4: Linear advection. ML-WENO5 error and convergence rate on the nonuniform grid at $T = 10$, with $\epsilon = 10^{-6}$.



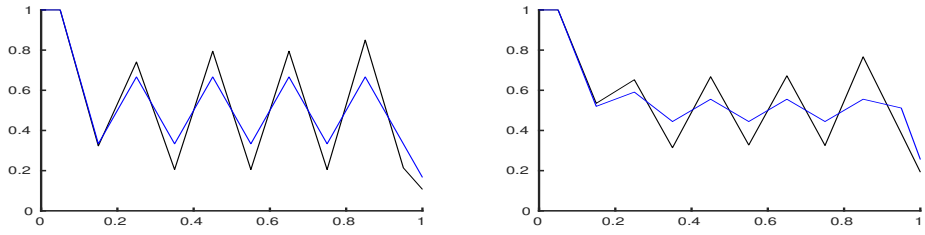
(a) 1 time step. $N = 50$. (b) 4 time steps. $N = 100$.

Fig. 4.1: Shu's linear test. $\Delta t = \Delta x/3$. The plots are WENO5-JS (black line) and ML-WENO5 (blue line).



(a) 1 time step. $N = 50$. (b) 4 time steps. $N = 100$.

Fig. 4.2: Shu's linear test. $\Delta t = \Delta x/3$. The plots are WENO5-JS (black line) and ML-WENO5 with levels 2-4 (red line).



(a) 1 time step. (b) 2 time steps.

Fig. 4.3: Sawtooth. $\Delta t = \Delta x/3$. The plots are WENO5-JS (black line) and ML-WENO5 (blue line).

4.2. Burger's equation. We solve the Burger's equation

$$u_t + (u^2/2)_x = 0$$

with the initial condition $u_0(x) = 0.75 + 0.25\sin(\pi x)$ on $[0, 2]$ at time $T = 1$. The results are shown in Tables 4.5-4.8.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.027721	——	0.034421	——
40	0.0097809	1.5029	0.018005	0.93489
80	0.0021051	2.2161	0.005806	1.6328
160	0.00035519	2.5672	0.0016446	1.8198
320	4.9815e-05	2.8339	0.00027398	2.5856
640	6.466e-06	2.9456	3.7233e-05	2.8794
1280	8.1516e-07	2.9877	4.731e-06	2.9764

Table 4.5: Burger's equation. ML-WENO3 error and convergence rate on the uniform grid at $T = 1$, with $\epsilon = 10^{-2}$, $\Delta t = 0.2\Delta x$.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.029534	——	0.034901	——
40	0.0092092	1.6812	0.016954	1.0416
80	0.0020155	2.1919	0.0058789	1.528
160	0.00038675	2.3817	0.0017418	1.755
320	4.8858e-05	2.9847	0.00028498	2.6116
640	6.6661e-06	2.8737	4.028e-05	2.8227
1280	8.3898e-07	2.9901	5.0546e-06	2.9944

Table 4.6: Burger's equation. ML-WENO3 error and convergence rate on the nonuniform grid at $T = 1$, with $\epsilon = 10^{-2}$.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.0102223	——	0.0136383	——
40	0.0040052	1.3518	0.0094211	0.5337
80	0.00054117	2.8877	0.0021382	2.1395
160	4.1833e-05	3.6934	0.00023803	3.1672
320	1.8758e-06	4.4791	1.6293e-05	3.8688
640	6.4662e-08	4.8584	6.0797e-07	4.7441
1280	2.069e-09	4.9659	1.9755e-08	4.9437

Table 4.7: Burger's equation. ML-WENO5 error and convergence rate on the uniform grid at $T = 1$, with $\epsilon = 10^{-6}$, $\Delta t = 0.2\Delta x$.

4.3. Buckley-Leverett equation. In this example we test the Buckley-Leverett equation, which has the nonconvex flux

$$f(u) = \frac{u^2}{u^2 + (1-u)^2}.$$

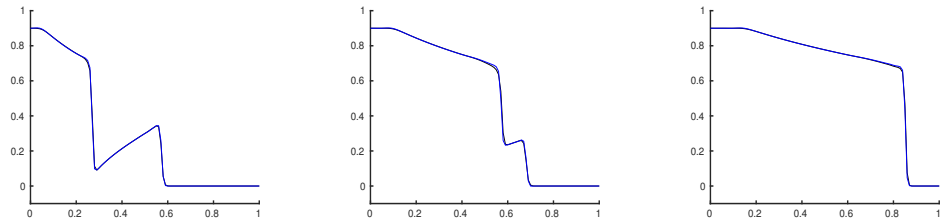
N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.00935745	———	0.012636	———
40	0.004098	1.1912	0.0083887	0.59101
80	0.00060747	2.754	0.0020472	2.0348
160	3.7498e-05	4.0179	0.00023405	3.1287
320	1.9878e-06	4.2375	1.7059e-05	3.7782
640	6.8505e-08	4.8589	6.3076e-07	4.7573
1280	2.074e-09	5.0457	1.9914e-08	4.9852

Table 4.8: Burger's equation. ML-WENO5 error and convergence rate on the nonuniform grid at $T = 1$, with $\epsilon = 10^{-6}$.

We use the initial condition

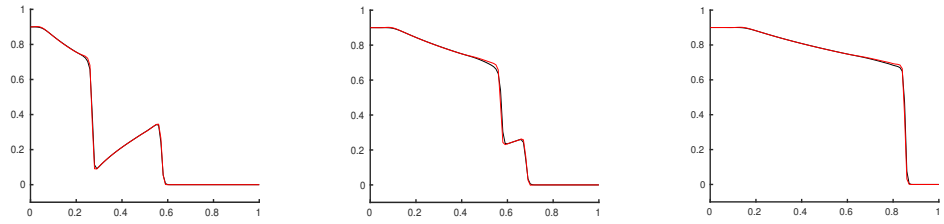
$$u_0(x) = \begin{cases} 1 - 20x, & \text{for } 0 \le x \le 0.05, \\ 0.5, & \text{for } 0.25 \le x \le 0.4, \\ 0, & \text{otherwise.} \end{cases}$$

The results are shown in Figures 4.4 and 4.5 on uniform grids using $N = 100$ cells at $t = 0.2$, $t = 0.4$, $t = 0.6$ and $dt = 0.1\Delta x$. We take $\epsilon = 10^{-6}$ for both of the tests.



(a) t = 0.2. (b) t = 0.4. (c) t = 0.6.

Fig. 4.4: Buckley-Leverett equation. $N = 100$, $\Delta t = 0.1\Delta x$. The plots are WENO5-JS (black line) and ML-WENO5 (blue line).



(a) t = 0.2. (b) t = 0.4. (c) t = 0.6.

Fig. 4.5: Buckley-Leverett equation. $N = 100$, $\Delta t = 0.1\Delta x$. The plots are WENO5-JS (black line) and ML-WENO5 with levels 2-4 (red line).

4.4. 1D Euler system. The one dimensional Euler system of gas dynamics is given by

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ m\\ E\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}m\\ \rho u^2+p\\ u(E+p)\end{pmatrix}=0, \tag{4.3}$$

where $m=\rho u$, $E=p/(\gamma-1)+\rho u^2/2$ and $\rho,\ u,\ m,\ p,\ E$ are the density, velocity, momentum, pressure and energy, respectively, and $\gamma=1.4$. For all the tests below, we implemented ML-WENO3 with $\epsilon=10^{-6}$, ML-WENO5 with $\epsilon=10^{-10}$ and WENO3-JS with $\epsilon=10^{-6}$, WENO5-JS with $\epsilon=10^{-6}$.

4.4.1. Accuracy test. We first test the order of convergence on the Euler equations on a smooth problem. The initial condition is given by

$$\rho(x,0)=1+0.2\,\sin(\pi x),\ u(x,0)=1,\ p(x,0)=1,$$

with periodic boundary conditions. The exact solution is

$$\rho(x,t)=1+0.2\sin(\pi x-\pi t),\ u=1,\ p=1.$$

See Tables 4.9–4.11.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.00126714	——	0.00113825	——
40	0.0001553	3.0284	0.00013052	3.1244
80	1.9382e-05	3.0023	1.7405e-05	2.9067
160	2.4082e-06	3.0087	2.1246e-06	3.0343
320	3.0811e-07	2.9664	2.7545e-07	2.9473
640	3.8353e-08	3.006	3.3719e-08	3.0302
1280	4.7796e-09	3.0044	4.25e-09	2.988

Table 4.9: Euler's equations. ML-WENO3 error and convergence rate on the nonuniform grid at $T=0.5$ and $\epsilon=10^{-2}$.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.0109309	——	0.0136255	——
40	0.0025514	2.0991	0.0049705	1.4549
80	0.00021705	3.5552	0.00077401	2.683
160	5.1174e-06	5.4065	2.512e-05	4.9455
320	3.0611e-07	4.0633	4.8434e-07	5.6966
640	3.8225e-08	3.0015	3.6405e-08	3.7338
1280	4.7546e-09	3.0071	4.2174e-09	3.1097

Table 4.10: Euler's equations. ML-WENO3 error and convergence rate on the nonuniform grid at $T=0.5$ and $\epsilon=10^{-6}$.

N	L^1 error	L^1 order	L^∞ error	L^∞ order
20	0.0167892	——	0.00989689	——
40	2.4422e-06	12.747	7.5993e-07	13.669
80	7.6389e-08	4.9987	2.5682e-08	4.887
160	2.4489e-09	4.9631	8.3405e-10	4.9445
320	7.6266e-11	5.005	2.6277e-11	4.9883
640	2.3736e-12	5.0059	7.9314e-13	5.0501

Table 4.11: Euler's equations. ML-WENO5 error and convergence rate on the nonuniform grid at $T = 0.5$.

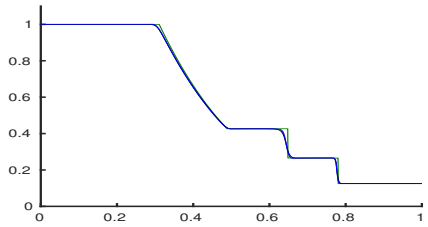
4.4.2. Shock tube test with Sod's and Lax's initial conditions. Sod's initial condition is

$$(\rho, m, E) = \begin{cases} \rho_l = 1,\ m_l = 0,\ E_l = 2.5, & \text{for } x < 0.5, \\ \rho_r = 0.125,\ m_r = 0,\ E_r = 0.25, & \text{for } x > 0.5, \end{cases}$$

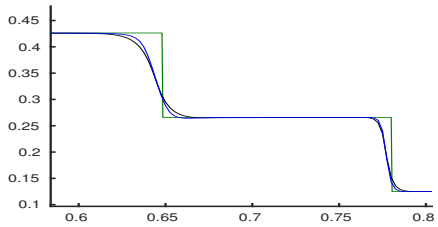
and Lax's initial condition is

$$(\rho, m, E) = \begin{cases} \rho_l = 0.445,\ m_l = 0.331,\ E_l = 8.928, & \text{for } x < 0.5, \\ \rho_r = 0.5,\ m_r = 0,\ E_r = 1.4275. & \text{for } x > 0.5. \end{cases}$$

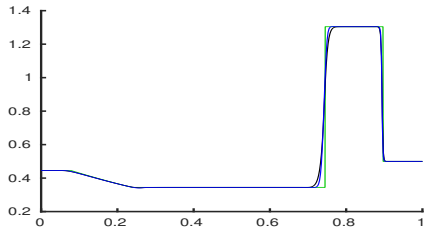
The results are shown in Figures 4.6–4.8 using 400 cells and $\Delta t = 0.1\Delta x$. The ML-WENO5 result is somewhat better than WENO5-JS.



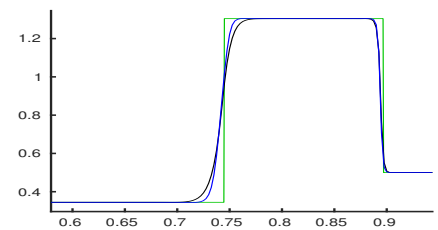
(a) Sod. $N = 400$. Third order scheme.



(b) Sod. $N = 400$. Zoom.

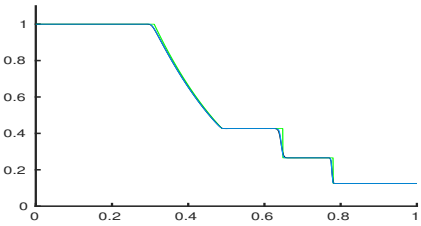


(c) Lax. $N = 400$. Third order scheme.

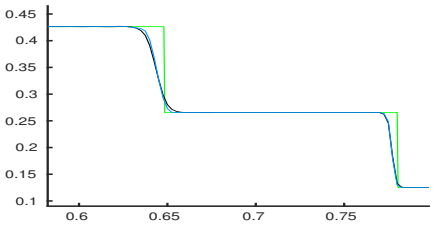


(d) Lax. $N = 400$. Zoom.

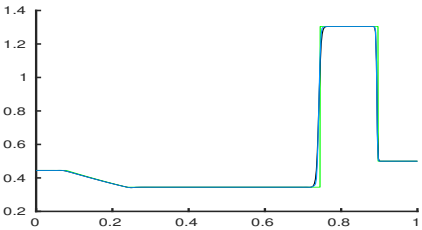
Fig. 4.6: Sod and Lax density at $T = 0.16$ using $N = 400$ cells. The plots are the reference solution (green), WENO3-JS (black) and ML-WENO3 (blue).



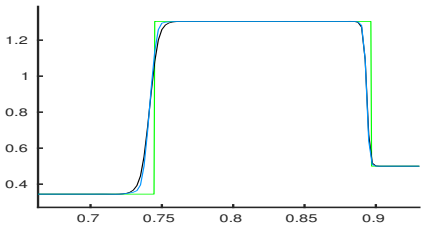
(a) Sod. $N = 400$. Fifth order scheme.



(b) Sod. $N = 400$. Zoom.

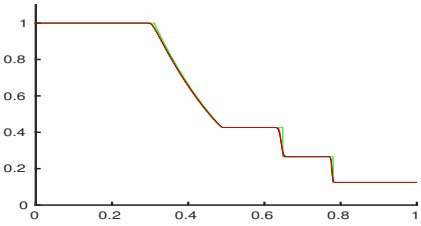


(c) Lax. $N = 400$. Fifth order scheme.

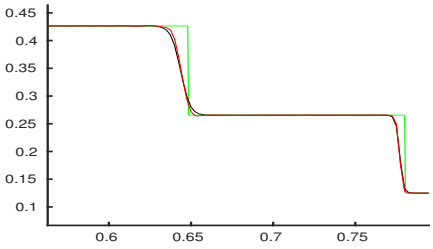


(d) Lax. $N = 400$. Zoom.

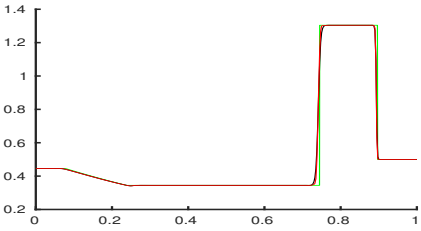
Fig. 4.7: Sod and Lax density at $T = 0.16$ using $N = 400$ cells. The plots are the reference solution (green), WENO5-JS (black) and ML-WENO5 (blue).



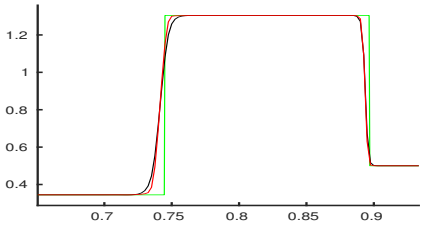
(a) Sod. $N = 400$. Fifth order scheme.



(b) Sod. $N = 400$. Zoom.



(c) Lax. $N = 400$. Fifth order scheme.



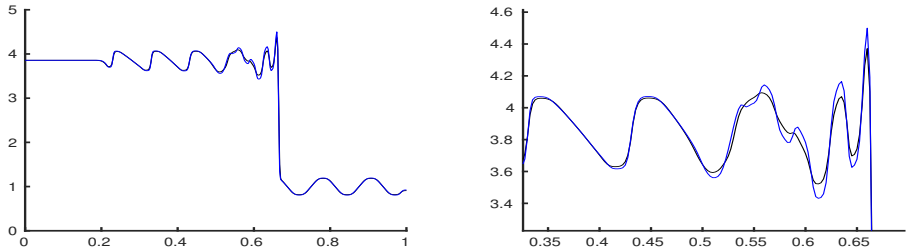
(d) Lax. $N = 400$. Zoom.

Fig. 4.8: Sod and Lax density at $T = 0.16$ using $N = 400$ cells. The plots are the reference solution (green), WENO5-JS (black) and ML-WENO5 with levels 2-4 (red).

4.4.3. Shu and Osher's shock interaction with entropy waves. Next we solve the shock interaction with entropy waves problem given in [4], which has a moving Mach 3 shock interacting with sine waves in density. The initial condition is

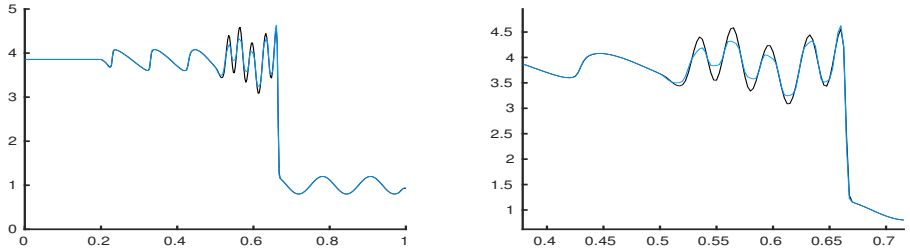
$$(\rho, u, p) = \begin{cases} \rho_l = 3.857143,\ u_l = 2.629396,\ p_l = 10.333333, & \text{for } 0 < x < 0.1, \\ \rho_r = 1 + 0.2\sin(5(10x - 5)),\ u_r = 0,\ p_r = 1, & \text{for } 0.1 < x < 1. \end{cases}$$

We compute the density at $T = 0.16$ using $\Delta t = 0.1\Delta x$ and $N = 400$ cells. The results are shown in Figures 4.9–4.11.



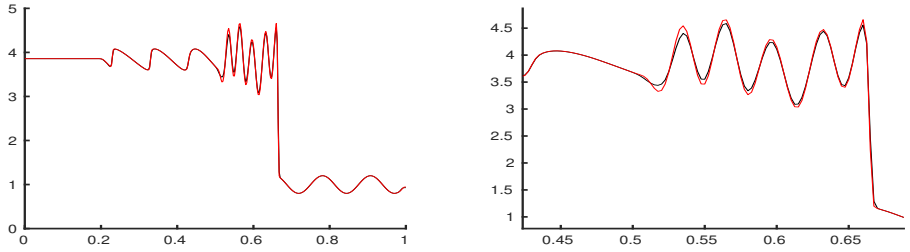
(a) $N = 400$. Third order scheme. (b) $N = 400$. Zoom.

Fig. 4.9: Shu and Osher's shock interaction with entropy waves density at $T = 0.16$ using $N = 400$ cells. The plots are WENO3-JS (black) and ML-WENO3 (blue).



(a) $N = 400$. Fifth order scheme. (b) $N = 400$. Zoom.

Fig. 4.10: Shu and Osher's shock interaction with entropy waves density at $T = 0.16$ using $N = 400$ cells. The plots are WENO5-JS (black) and ML-WENO5 (blue).



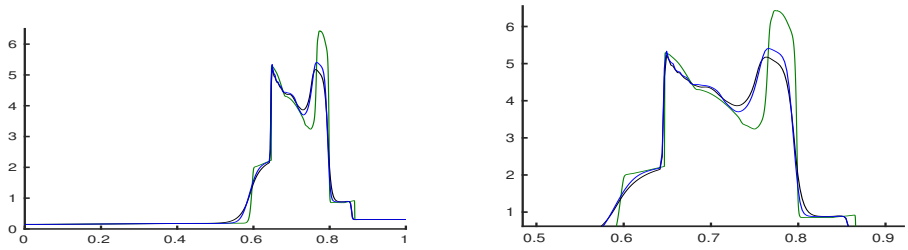
(a) $N = 400$. Fifth order scheme. (b) $N = 400$. Zoom.

Fig. 4.11: Shu and Osher's shock interaction with entropy waves density at $T = 0.16$ using $N = 400$ cells. The plots are WENO5-JS (black) and ML-WENO5 with levels 2-4 (red).

4.4.4. Woodward and Colella's double blast test. This test uses the initial condition

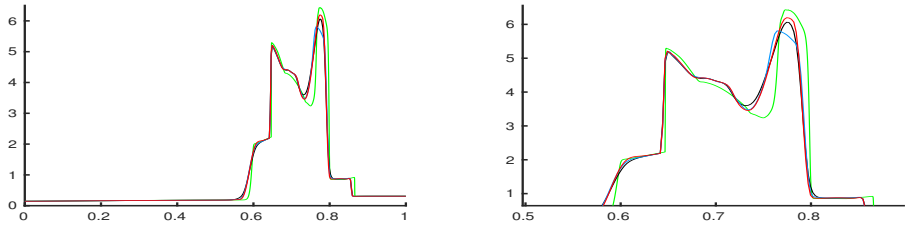
$$(\rho, m, E) = \begin{cases} \rho_l = 1,\ m_l = 0,\ E_l = 1000/(\gamma-1), & \text{for } 0 < x < 0.1, \\ \rho_m = 1,\ m_m = 0,\ E_m = 0.01/(\gamma-1), & \text{for } 0.1 < x < 0.9, \\ \rho_r = 1,\ m_r = 0,\ E_r = 100/(\gamma-1), & \text{for } 0.9 < x < 1. \end{cases}$$

See Figures 4.12 and 4.13.



(a) $N = 399$. Third order scheme. (b) $N = 399$. Zoom

Fig. 4.12: Woodward and Colellas double blast test density at $T = 0.038$ using $N = 399$ cells. The plots are the reference solution (green), WENO3-JS (black) and ML-WENO3 (blue).



(a) $N = 399$. Fifth order scheme. (b) $N = 399$. Zoom

Fig. 4.13: Woodward and Colellas double blast test density at $T = 0.038$ using $N = 399$ cells. The plots are the reference solution (green), WENO5-JS (black), ML-WENO5 (blue) and ML-WENO5 with levels 2-4 (red).

REFERENCES

[1] G.-S. Jiang and C.-W. Shu, *Efficient implementation of weighted ENO schemes*, J. Comput. Phys., 126 (1996), pp. 202–228.

[2] R. J. Leveque, *High-resolution conservative algorithms for advection in incompressible flow*, SIAM J. Numer. Anal., 33 (1996), pp. 627–665.

[3] D. Levy, G. Puppo, and G. Russo, *Compact central WENO schemes for multidimensional conservation laws*, SIAM J. Sci. Comput., 22 (2000), pp. 656–672.

[4] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock-capturing schemes, ii*, Journal of Computational Physics, 83 (1989), pp. 32 – 78.