SIAM J. SCI. COMPUT.
Vol. 24, No. 6, pp. 2185–2198

FINITE DIFFERENCE WENO SCHEMES WITH LAX–WENDROFF-TYPE TIME DISCRETIZATIONS*

JIANXIAN QIU† AND CHI-WANG SHU‡

Abstract. In this paper we develop a Lax–Wendroff time discretization procedure for high order finite difference weighted essentially nonoscillatory schemes to solve hyperbolic conservation laws. This is an alternative method for time discretization to the popular TVD Runge–Kutta time discretizations. We explore the possibility in avoiding the local characteristic decompositions or even the nonlinear weights for part of the procedure, hence reducing the cost but still maintaining nonoscillatory properties for problems with strong shocks. As a result, the Lax–Wendroff time discretization procedure is more cost effective than the Runge–Kutta time discretizations for certain problems including two-dimensional Euler systems of compressible gas dynamics.

Key words. weighted essentially nonoscillatory method, Lax–Wendroff-type time discretization, high order accuracy

AMS subject classification. 65M06

DOI. 10.1137/S1064827502412504

1. Introduction.

In this paper, we study an alternative method for time discretization, namely the Lax–Wendroff-type time discretization [7], to the popular TVD Runge–Kutta time discretization in [16], for the high order finite difference WENO (weighted essentially nonoscillatory) methods [6], [1] in solving nonlinear hyperbolic conservation law systems

$$\begin{cases} u_t + \nabla \cdot f(u) = 0, \\ u(x,0) = u_0(x). \end{cases} \tag{1.1}$$

WENO finite difference methods have been developed in recent years as a class of high order methods for conservation laws (1.1) which gives sharp, nonoscillatory discontinuity transitions and at the same time provides high order accurate resolutions for the smooth part of the solution. The first WENO scheme is constructed in [9] for a third order finite volume version in one space dimension. In [6], third and fifth order finite difference WENO schemes in multispace dimensions are constructed, with a general framework for the design of the smoothness indicators and nonlinear weights. Finite difference WENO schemes of higher orders (seventh to eleventh order) are constructed in [1]. WENO schemes are designed based on the successful ENO schemes in [3, 16, 17]. Both ENO and WENO schemes use the idea of adaptive stencils in the reconstruction procedure based on the local smoothness of the numerical solution to automatically achieve high order accuracy and a nonoscillatory property near

*Received by the editors August 5, 2002; accepted for publication (in revised form) December 26, 2002; published electronically July 11, 2003.
http://www.siam.org/journals/sisc/24-6/41250.html

†Department of Mathematics, University of Science and Technology of China, Hefei, Anhui 230026, People's Republic of China (jxqiu@ustc.edu.cn). The research of this author was supported in part by NNSFC grant 10028103.

‡Division of Applied Mathematics, Brown University, Providence, RI 02912 (shu@dam.brown.edu). The research of this author was supported by NNSFC grant 10028103 while he was in residence at the Department of Mathematics, University of Science and Technology of China, Hefei, Anhui 230026, People's Republic of China. Additional support was provided by ARO grant DAAD19-00-1-0405 and NSF grants DMS-9804985 and DMS-0207451.

discontinuities. ENO uses just one (optimal in some sense) out of many candidate stencils when doing the reconstruction, while WENO uses a convex combination of all the candidate stencils, each being assigned a nonlinear weight which depends on the local smoothness of the numerical solution based on that stencil. WENO improves upon ENO in robustness, better smoothness of fluxes, better steady state convergence, better provable convergence properties, and more efficiency. Finite volume WENO schemes on unstructured meshes and central-type WENO schemes have also been constructed; see, e.g., [4, 5, 8, 11]. For a detailed review of ENO and WENO schemes, we refer to the lecture notes [15].

WENO is a spatial discretization procedure; namely, it is a procedure to approximate the spatial derivative terms in (1.1). The time derivative term there must also be discretized. There are mainly two different approaches to approximate the time derivative. The first approach is to use an ODE solver, such as a Runge–Kutta or a multistep method, to solve the method of lines ODE obtained after spatial discretization. The second approach is a Lax–Wendroff-type time discretization procedure, which is also called the Taylor type, referring to a Taylor expansion in time, or the Cauchy–Kowalewski type, referring to the similar Cauchy–Kowalewski procedure in the PDE. This approach is based on the idea of the classical Lax–Wendroff scheme [7], and it relies on converting all the time derivatives in a temporal Taylor expansion into spatial derivatives by repeatedly using the PDE and its differentiated versions. The spatial derivatives are then discretized by, e.g., the WENO approximations.

The first approach, namely the method of lines plus an ODE solver, has the advantage of simplicity, both in concept and in coding. It also enjoys good stability properties when the TVD-type Runge–Kutta or multistep methods are used [16, 14]. Thus the majority of the WENO codes are using this type of time discretization. To be on the safe side, a TVD time discretization [16, 14] is preferred. There is, however, an order barrier for TVD Runge–Kutta methods with positive coefficients: they cannot be higher than fourth order accurate [12]. In many practical implementations reported, when the solution is not smooth, a third order TVD Runge–Kutta method from [16] has been used, e.g., even for the very high order WENO methods in [1]. We should, however, remark that in practice it seems that for many problems spatial accuracy is more crucial than temporal accuracy; hence a third order TVD Runge–Kutta method together with a higher order spatial discretization often gives satisfactory results.

The second approach, the Lax–Wendroff-type time discretization, usually produces the same high order accuracy with a smaller effective stencil than that of the first approach, and it uses more extensively the original PDE. However, the formulation and coding of this procedure could be quite complicated, especially for multidimensional systems. The original finite volume ENO schemes in [3] used this approach for the time discretization. More recently, Titarev and Toro [18] and Schwartzkopff, Munz, and Toro [13] used this approach to construct a class of high order schemes. They termed such schemes ADER (arbitrary high order schemes utilizing higher order derivatives), which stem from the modified generalized Riemann problem (MGRP) scheme by Toro [19], a simplification of the GRP scheme by Ben-Artzi and Falcovitz [2]. The ADER approach is an explicit two-level, finite-volume scheme which uses, among other things, the Lax–Wendroff time discretization procedure to convert time derivatives to spatial derivatives using the PDE before discretizing them. In Lukácová-Medvid'ová and Warnecke [10], a Lax–Wendroff-type second order evolution Galerkin method for multidimensional hyperbolic systems is discussed.

The Lax–Wendroff-type time discretization is easier to formulate for the finite volume than for the finite difference spatial discretizations. The existing schemes using the Lax–Wendroff-type time discretization procedure in the literature, such as those mentioned above, are mostly of finite volume (or finite element) type.

In this paper we explore the Lax–Wendroff-type time discretization procedure for the high order finite difference WENO spatial discretizations in [6] and [1]. The same procedure certainly also applies to other types of high order conservative finite difference methods. The resulting conservative schemes are more complex to formulate and to code than those using the TVD Runge–Kutta time discretization [16]; hence unless there is an advantage in CPU timing for the same accuracy, they will not be competitive. Fortunately, after exploring the possibility in avoiding the local characteristic decompositions or even the nonlinear weights for part of the procedure, hence reducing the cost but still maintaining nonoscillatory properties for problems with strong shocks, we demonstrate numerically that the Lax–Wendroff time discretization procedure adopted in this paper is more cost effective than the Runge–Kutta time discretizations for certain problems including two-dimensional Euler systems of compressible gas dynamics.

In [11], we have demonstrated the necessity of using a local characteristic decomposition in the WENO reconstruction procedure to avoid or to significantly reduce the spurious oscillations, especially for higher order schemes, although for lower order WENO schemes such local characteristic decompositions can sometimes be avoided. A fact more relevant to this paper is that after extensive numerical experiments, we found in [11] we need to perform only the costly local characteristic decomposition for some part of reconstruction, and use componentwise WENO in other parts of reconstruction, and still maintain an ENO shock transition. Similarly, in this paper we develop the fifth order finite difference WENO schemes with a Lax–Wendroff time discretization procedure by performing only the local characteristic decomposition and WENO approximation for the reconstruction of the fluxes to the first order time derivative, and using the inexpensive central difference approximations for the reconstruction of the higher order time derivatives, as these terms are multiplied by $O(\Delta t)$ or its higher powers. We demonstrate through extensive numerical examples that the procedure is robust, ENO, accurate, and CPU time efficient when compared with the fifth order finite difference WENO schemes with Runge–Kutta time discretizations [6] for certain types of problems including the two-dimensional Euler equations of compressible gas dynamics. Of course, this procedure becomes progressively more complicated with more complicated PDEs and/or higher order time accuracy; hence we do not expect it to be always cost effective relative to the standard Runge–Kutta time discretizations.

In this paper we do not address the important issue of time discretization for PDEs with diffusion terms and/or with stiff source terms, which calls for hybrid explicit/implicit time discretization. There are good Runge–Kutta methods to easily achieve this; see, e.g., [21]. The Lax–Wendroff procedure in this paper can also be adapted for such a purpose, through careful Taylor expansions.

The organization of the paper is as follows. In section 2, we describe in detail the construction and implementation of the high order (we use the fifth order as an example) WENO finite difference schemes with a Lax–Wendroff-type time discretization for one- and two-dimensional scalar and system equations (1.1). In section 3, we provide extensive numerical examples to demonstrate the behavior of the schemes and to perform a comparison with the fifth order finite difference WENO schemes with Runge–Kutta time discretizations [6]. Concluding remarks are given in section 4.

2. Construction and implementation of the scheme. In this section we describe in detail the construction and implementation of the high order (using the fifth order as an example) WENO finite difference schemes with a Lax–Wendroff-type time discretization for one- and two-dimensional scalar and system conservation laws.

2.1. One-dimensional scalar case. Consider the one-dimensional scalar conservation laws:

$$\begin{cases} u_t + f(u)_x = 0, \\ u(x,0) = u_0(x). \end{cases} \tag{2.1}$$

For simplicity, we assume that the grid points $\{x_i\}$ are uniform with $x_{i+1} - x_i = \Delta x$, and we denote the cells by $I_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$. Let Δt be the time step, $t^{n+1} = t^n + \Delta t$, and let u_i^n be the approximation of the point values $u(x_i, t^n)$. We denote by $u^{(r)}$ the rth order time derivative of u, namely $\frac{\partial^r u}{\partial t^r}$. We also use u', u'', and u''' to denote the first three time derivatives of u. By a temporal Taylor expansion we obtain

$$u(x, t+\Delta t) = u(x,t) + \Delta t u' + \frac{\Delta t^2}{2} u'' + \frac{\Delta t^3}{6} u''' + \frac{\Delta t^4}{24} u^{(4)} + \cdots . \tag{2.2}$$

If we would like to obtain kth order accuracy in time, we would need to approximate the first k time derivatives: $u', \ldots, u^{(k)}$. We will proceed up to fourth order in time in this paper, although the procedure can be naturally extended to any higher order.

After extensive numerical tests, we have found the following Lax–Wendroff procedure, which produces the best balance between cost reduction and ensuring ENO properties.

Step 1. The reconstruction of the first time derivative $u' = -f(u)_x$ is obtained by the regular conservative $(2r-1)$th order WENO finite difference procedure as described in detail in [6] and [1]; see also [15]. For example, $r = 3$ would produce a fifth order approximation.

Step 2. The reconstruction of the second time derivative $u'' = -(f'(u)u')_x$ is obtained as follows. Notice that we will need only an approximation of order $(2r-2)$, one order lower than before, because of the extra Δt factor. Let $g_i = f'(u_i)\, u_i'$, where u_i and u_i' are the point values of u and u' at the point (x_i, t^n) computed in Step 1 described above. *We can use a simple* $(2r-2)$*th order central difference formula to approximate* u'' *at the point* (x_i, t^n). For example, when $r = 3$, we use the following fourth order central difference approximation:

$$u_i'' \approx -\frac{1}{12\Delta x}(g_{i-2} - 8g_{i-1} + 8g_{i+1} - g_{i+2}). \tag{2.3}$$

Notice that this approximation is conservative; namely, it can be written as a flux differenced form. It seems that a more costly WENO approximation is *not* needed here to control spurious oscillations, presumably because this term is multiplied by an extra Δt anyway.

Step 3. The reconstruction of the third time derivative $u''' = -(f'(u)u'' + f''(u)(u')^2)_x$ is obtained as follows. Let $g_i = f'(u_i)u_i'' + f''(u_i)(u_i')^2$; here u_i' and u_i'' are the point values of u' and u'' at the point (x_i, t^n) computed in Step 1 and Step 2 above. Then we repeat Step 2 to get the approximation of u''' using a central difference approximation of order $(2r-2)$. In fact, we need only an approximation of order $(2r-3)$, because of the extra Δt^2 factor, but we would like to use simple central differences which are all of even order. Again, it seems that a more costly WENO approximation is *not* needed here to control spurious oscillations.

Step 4. The reconstruction of the fourth time derivative $u^{(4)} = -(f'(u)u''' + 3f''(u)u'u'' + f'''(u)(u')^3)_x$ is obtained in a similar fashion. Let $g_i = f'(u_i)u_i''' + 3f''(u_i)u_i'u_i'' + f'''(u_i)(u_i')^3$, where u_i', u_i'', and u_i''' are the point values of u', u'', and u''' at the point (x_i, t^n) computed in Step 1, Step 2, and Step 3 above. In order to get a $(2r-1)$th order scheme, we need to use only a $(2r-4)$th order central difference approximation to $u^{(4)}$ at the point (x_i, t^n), because of the extra Δt^3 factor. For example, when $r = 3$, we can use the following second order approximation:

$$u_i^{(4)} \approx -\frac{1}{2\Delta x}(g_{i+1} - g_{i-1}), \tag{2.4}$$

which is again a conservative approximation.

If we require higher order accuracy, in time this procedure can be continued in a similar fashion. The final conservative approximation at the next time step is then given by

$$u(x_i, t^{n+1}) \approx u_i + \Delta t u_i' + \frac{\Delta t^2}{2}u_i'' + \frac{\Delta t^3}{6}u_i''' + \frac{\Delta t^4}{24}u_i^{(4)} + \cdots + \frac{\Delta t^k}{k!}u_i^{(k)}. \tag{2.5}$$

In the following we denote the schemes with $(2r-1)$th order in space and kth order in time, obtained with this Lax–Wendroff-type time discretization, as WENO$(2r-1)$-LWk. We will concentrate our attention on WENO5-LW3 and WENO5-LW4. As a comparison we denote the fifth order finite difference WENO schemes with time discretization by third and fourth order Runge–Kutta methods in [6] as WENO5-RK3 and WENO5-RK4.

We remark that the Lax–Wendroff-type time discretization adopted in this paper yields a final stencil which is narrower than the corresponding scheme with a Runge–Kutta time discretization. Hence the scheme is more compact, which is a feature shared by all Lax–Wendroff-type time discretization based schemes. For example, for the WENO5 reconstruction, 7 points are needed in the most general case with a Lax–Friedrich flux splitting [6]. For the schemes WENO5-RK3 and WENO5-RK4, 3 and 4 inner stages are needed, respectively, to march one time step; hence 3 or 4 WENO5 reconstructions are needed to compute solution at the next time step, resulting in a final stencil consisting of 19 and 25 points, respectively, for the WENO5-RK3 and WENO5-RK4 schemes. On the other hand, the total number of points in the final stencil for the WENO5-LW3 and WENO5-LW4 schemes are 15 and 17, respectively, which consist of 7 points in Step 1, 4 additional points in each of Steps 2 and 3, and 2 additional points in Step 4.

2.2. One-dimensional systems. For systems of conservation laws (2.1), $u(x,t) = (u^1(x,t), \ldots, u^m(x,t))^T$ is a vector and $f(u) = (f^1(u^1, \ldots, u^m), \ldots, f^m(u^1, \ldots, u^m))^T$ is a vector function of u. As before, the time derivatives in (2.2) are replaced by the spatial derivatives using the PDE. For the first time derivative $u' = -f(u)_x$, we again use the regular conservative WENO procedure of $(2r-1)$th order accuracy to approximate $-f(u)_x$. This approximation should be performed in the local characteristic directions to avoid spurious oscillations. See [6] and [11] for details. For the second and higher time derivatives, however, it seems that central approximations component by component of adequate order of accuracy, as in the scalar case, is enough to ensure the ENO property. Neither a local characteristic decomposition nor a WENO approximation is needed for these terms. On the other hand, it seems that WENO schemes with TVD Runge–Kutta time discretizations [6] would need a local characteristic decomposition for all inner stages to guarantee a nonoscillatory solution. If

TABLE 1
CPU time (in seconds) for the WENO5-LW and WENO5-RK schemes to compute the double Mach reflection problem in Example 3.9.

Schemes	240×59	Ratio	480×119	Ratio
WENO5-LW3	104.36		1290.97	
WENO5-RK3	210.48	1:2.02	2805.65	1:2.17
WENO5-LW4	152.80		2111.94	
WENO5-RK4	281.54	1:1.84	3780.80	1:1.75

the local characteristic decomposition is performed only for the first inner stage, the result becomes oscillatory, according to our numerical experiments. We note that the second and higher order time derivatives, when converted to spatial derivatives as before, involve expressions like $f'(u)$, which is a matrix (the Jacobian), $f''(u)$, which is a three-dimensional "matrix" (tensor), etc., which could become very complicated. A symbolic manipulator such as MAPLE would be helpful to avoid mistakes. The code is also quite long and messy compared with codes using Runge–Kutta time discretizations. However, we will see in the next section that one can save CPU time by this approach for certain problems.

2.3. Two-dimensional cases. Consider the two-dimensional conservation laws:

$$\begin{cases} u_t + f(u)_x + g(u)_y = 0, \\ u(x, y, 0) = u_0(x, y). \end{cases} \tag{2.6}$$

By a temporal Taylor expansion we obtain

$$u(x, y, t + \Delta t) = u(x, y, t) + \Delta t u' + \frac{\Delta t^2}{2} u'' + \frac{\Delta t^3}{6} u''' + \cdots .$$

For example, for third order accuracy in time we would need to reconstruct three time derivatives: u', u'', u'''.

We again use the PDE (2.6) to replace time derivatives by spatial derivatives. The first time derivative $u' = -f(u)_x - g(u)_y$ is approximated by the regular conservative finite difference procedure in a dimension-by-dimension fashion, including a local characteristic decomposition for the system case; see [6]. On the other hand, as in the one-dimensional situation, the second order time derivative $u'' = -(f'(u)u')_x - (g'(u)u')_y$, the third order time derivative $u''' = -(f''(u)(u')^2 + f'(u)u'')_x - (g''(u)(u')^2 + g'(u)u'')_y$, and the fourth order time derivative $u^{(4)} = -(f'''(u)(u')^3 + 3f''(u)u'u'' + f'(u)u''')_x - (g'''(u)(u')^3 + 3g''(u)u'u'' + g'(u)u''')_y$, etc., can be approximated by simple central differences of suitable orders of accuracy, again in a dimension-by-dimension fashion. For the system case, no local characteristic decompositions are needed for these approximations. It is again helpful to use a symbolic manipulator to obtain the complicated time derivative terms for the system case.

3. Numerical results. In this section we present the results of our numerical experiments for WENO5-LW3/LW4 schemes developed in the previous section and compare them with the finite difference WENO schemes using Runge–Kutta time discretizations in [6]. Uniform meshes are used, and the CFL number is taken as 0.5.

We first remark on the important issue of CPU timing and relevant efficiency of WENO5-LW schemes compared with WENO-RK schemes. In general the WENO5-LW schemes have smaller CPU costs for the same mesh and same order of accuracy in our implementation. For example, in Table 1, we provide a CPU time comparison between WENO5-LW and WENO5-RK schemes for the two-dimensional Euler

TABLE 2
$u_t+(\frac{u^2}{2})_x = 0$. $u(x,0) = 0.5+\sin(\pi x)$. *WENO5-LW4 using N equally spaced points.* $t = 0.5/\pi$. L_1 *and* L_∞ *errors.*

N	L_1 error	L_1 order	L_∞ error	L_∞ order
10	3.65E-02		6.85E-02	
20	4.44E-03	3.04	1.26E-02	2.45
40	2.55E-04	4.12	1.06E-03	3.57
80	9.99E-06	4.67	5.11E-05	4.38
160	3.76E-07	4.73	1.69E-06	4.92
320	1.15E-08	5.03	7.62E-08	4.47
640	2.96E-10	5.28	1.46E-09	5.70

equation, double Mach reflection test case in Example 3.9. The computer we used is an IBM PC with a Pentium-4 processor of 1.4G cpu and a 128M ram. We can see that the WENO5-LW schemes cost about half the CPU time compared with the WENO5-RK schemes for this example. Of course, this is closely related to the fact that the costly local characteristic decomposition and WENO approximation are used only once per time step in computing the first order time derivative, and the much less costly central difference approximations are used for all higher order time derivatives. However, in all the numerical experiments we have performed, nonoscillatory results can be obtained by this approach. We should remark, however, that there are also possibilities for reducing costs in Runge–Kutta methods, such as the possibility of freezing the nonlinear smoothness indicators for the inner stages, etc., which could potentially change the comparison in Table 1 but are not addressed in this paper.

3.1. Accuracy tests. We first test the accuracy of the schemes on linear scalar problems (results not shown to save space), nonlinear scalar problems, and nonlinear systems. Fourth order accuracy in time is used for this subsection. It seems that for all the test cases here, the spatial errors are dominating over the time errors; hence we actually observe close to fifth order accuracy, even though the time accuracy is only fourth order, for the meshes we have used.

Example 3.1. We solve the following nonlinear scalar Burgers equation:

$$u_t + \left(\frac{u^2}{2}\right)_x = 0 \tag{3.1}$$

with the initial condition $u(x,0) = 0.5+\sin(\pi x)$ and a 2-periodic boundary condition. When $t = 0.5/\pi$ the solution is still smooth, and the errors and numerical orders of accuracy by the WENO5-LW4 scheme are shown in Table 2. For comparison, errors and numerical orders of accuracy by the WENO5-RK4 scheme are shown in Table 3. We can see that both WENO5-LW4 and WENO5-RK4 schemes achieve their designed order of accuracy, and they produce similar errors and orders of accuracy.

Example 3.2. We solve the following two-dimensional nonlinear scalar Burgers equation:

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0 \tag{3.2}$$

with the initial condition $u(x,y,0) = 0.5+\sin(\pi(x+y)/2)$ and a 4-periodic boundary condition. To avoid any special error cancelations due to the symmetry axis being in the diagonals of cells, we use different mesh sizes in the x- and y-directions. When $t = 0.5/\pi$ the solution is still smooth, and the errors and numerical orders of accuracy

TABLE 3
$u_t+(\frac{u^2}{2})_x=0$. $u(x,0)=0.5+\sin(\pi x)$. *WENO5-RK4 using N equally spaced points.* $t=0.5/\pi$. L_1 *and* L_∞ *errors.*

N	L_1 error	L_1 order	L_∞ error	L_∞ order
10	4.10E-02		7.43E-02	
20	4.31E-03	3.25	1.23E-02	2.59
40	2.49E-04	4.12	1.05E-03	3.55
80	8.79E-06	4.82	4.78E-05	4.46
160	3.27E-07	4.75	1.40E-06	5.09
320	9.60E-09	5.09	7.27E-08	4.27
640	2.37E-10	5.34	1.20E-09	5.92

TABLE 4
$u_t+(\frac{u^2}{2})_x+(\frac{u^2}{2})_y=0, u_0(x)=0.5+\sin(\pi(x+y)/2)$. $[0,4]\times[0,4]$, $t=0.5/\pi$. *WENO5-LW4 using* $N_x\times N_y$ *equally spaced points.* L_1 *and* L_∞ *errors.*

$N_x\times N_y$	L_1 error	L_1 order	L_∞ error	L_∞ order
8* 12	2.23E-02		6.44E-02	
16* 24	2.98E-03	2.90	1.45E-02	2.15
32* 48	2.90E-04	3.36	1.65E-03	3.13
64* 96	8.58E-06	5.08	8.08E-05	4.36
128* 192	3.27E-07	4.71	2.57E-06	4.97
256* 384	1.01E-08	5.01	1.33E-07	4.28

by the WENO5-LW4 scheme are shown in Table 4. We can see that the scheme achieves close to its designed order of accuracy.

Example 3.3. We solve the following nonlinear system of Euler equations:

$$\xi_t+f(\xi)_x+g(\xi)_y=0 \tag{3.3}$$

with

$$\xi=(\rho,\rho u,\rho v,E)^T,\qquad f(\xi)=(\rho u,\rho u^2+p,\rho uv,u(E+p))^T,$$
$$g(\xi)=(\rho v,\rho uv,\rho v^2,v(E+p))^T.$$

Here ρ is the density, (u,v) is the velocity, E is the total energy, and p is the pressure, which is related to the total energy by $E=\frac{p}{\gamma-1}+\frac{1}{2}\rho(u^2+v^2)$ with $\gamma=1.4$. The initial condition is set to be $\rho(x,y,0)=1+0.2\sin(\pi(x+y))$, $u(x,y,0)=0.7$, $v(x,y,0)=0.3$, $p(x,y,0)=1$, with a 2-periodic boundary condition. The exact solution is $\rho(x,y,t)=1+0.2\sin(\pi(x+y-(u+v)t))$, $u=0.7$, $v=0.3$, $p=1$. We compute the solution up to $t=2$. The errors and numerical orders of accuracy of the density ρ for WENO5-LW4 are shown in Table 5. For comparison, the errors and numerical orders of accuracy by the WENO5-RK4 scheme are shown in Table 6. We can see that both WENO5-LW4 and WENO5-RK4 schemes achieve their designed order of accuracy, and they produce similar errors and orders of accuracy. Again, to avoid any special error cancelations due to the symmetry axis being in the diagonals of cells, we use different mesh sizes in the x- and y-directions.

3.2. Test cases with shocks.

Example 3.4. We solve the same nonlinear Burgers equation (3.1) as in Example 3.1 with the same initial condition $u(x,0)=0.5+\sin(\pi x)$, except that we now plot the results at $t=1.5/\pi$ when a shock has already appeared in the solution. In Figure 1, the solutions of WENO5-LW and the comparison schemes WENO5-RK are shown. The left figure compares WENO5-LW3 with WENO5-RK3, and the right

TABLE 5
Euler equations. $\rho(x,y,0) = 1+0.2\sin(\pi(x+y))$, $u(x,y,0) = 0.7$, $v(x,y,0) = 0.3$, $p(x,y,0) = 1$. *WENO5-LW4 using* $N_x \times N_y$ *equally spaced points.* $t = 2$. L_1 *and* L_∞ *errors of the density* ρ.

$N_x \times N_y$	L_1 error	L_1 order	L_∞ error	L_∞ order
8* 12	1.14E-02		1.84E-02	
16* 24	6.78E-04	4.07	1.18E-03	3.96
32* 48	2.19E-05	4.95	4.38E-05	4.75
64* 96	6.78E-07	5.01	1.36E-06	5.00
128* 192	2.09E-08	5.02	4.07E-08	5.07
256* 384	6.44E-10	5.02	1.14E-09	5.16

TABLE 6
Euler equations. $\rho(x,y,0) = 1+0.2\sin(\pi(x+y))$, $u(x,y,0) = 0.7$, $v(x,y,0) = 0.3$, $p(x,y,0) = 1$. *WENO5-RK4 using* $N_x \times N_y$ *equally spaced points.* $t = 2$. L_1 *and* L_∞ *errors of the density* ρ.

$N_x \times N_y$	L_1 error	L_1 order	L_∞ error	L_∞ order
8* 12	1.70E-02		2.65E-02	
16* 24	1.03E-03	4.05	1.69E-03	3.97
32* 48	3.41E-05	4.91	6.51E-05	4.70
64* 96	1.06E-06	5.00	2.09E-06	4.96
128* 192	3.31E-08	5.01	6.31E-08	5.05
256* 384	1.02E-09	5.02	1.87E-09	5.08

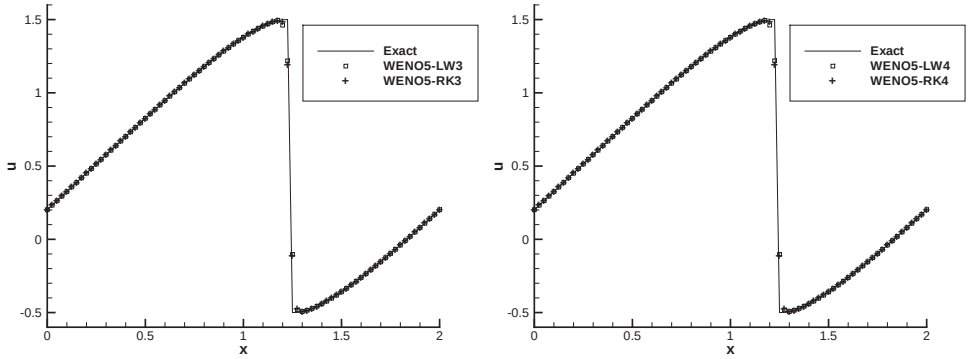

FIG. 1. *The Burgers equation.* $u(x,0) = 0.5 + \sin(\pi x)$. $t = 1.5/\pi$. $N = 80$ *points. Left: WENO5-LW3 and WENO5-RK3; right: WENO5-LW4 and WENO5-RK4.*

figure compares WENO5-LW4 with WENO5-RK4. $N = 80$ grid points are used. We can see that both schemes give equally good nonoscillatory shock transitions for this problem. Notice that the central difference approximations for the higher order time derivatives in the WENO5-LW schemes do not lead to spurious oscillations.

Example 3.5. We solve the nonlinear nonconvex scalar Buckley–Leverett problem

$$u_t + \left(\frac{4u^2}{4u^2 + (1-u)^2}\right)_x = 0 \tag{3.4}$$

with the initial data $u = 1$ when $-\frac{1}{2} \le x \le 0$ and $u = 0$ elsewhere. The solution is computed up to $t = 0.4$. The exact solution is a shock-rarefaction-contact discontinuity mixture. We remark that some high order schemes may fail to converge to the correct entropy solution for this problem. In Figure 2, the solutions of WENO5-LW3 and WENO5-RK3 schemes (left) and WENO5-LW4 and WENO-RK4 schemes (right) with $N = 160$ grid points are shown. The solid line is the exact solution. We can

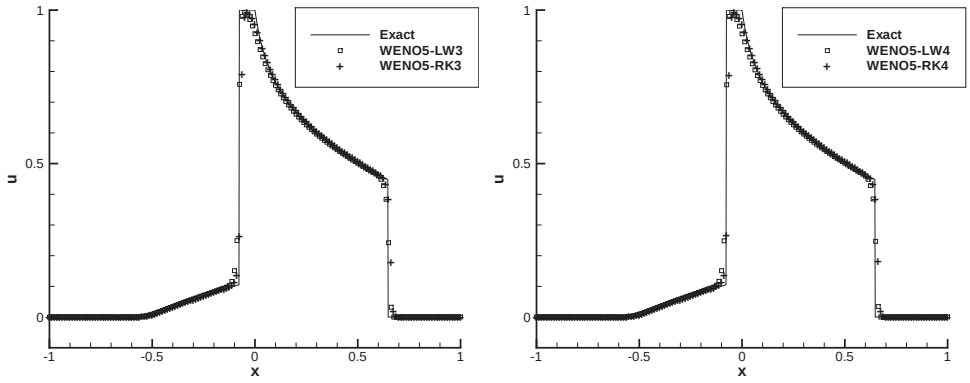

FIG. 2. *The Buckley–Leverett problem.* $t = 0.4$. $N = 160$ *points. Left: WENO*5*-LW*3 *and WENO*5*-RK*3*; right: WENO*5*-LW*4 *and WENO*5*-RK*4.

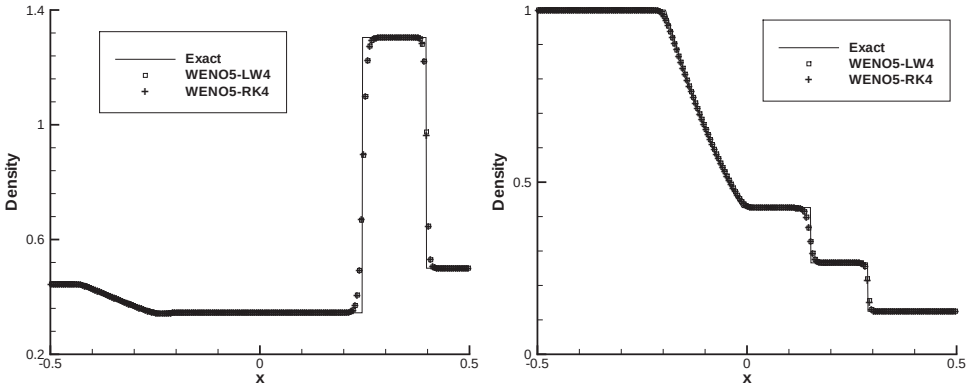

FIG. 3. *WENO*5*-LW*4 *comparing with WENO*5*-RK*4. $N = 200$ *points. Density. Left: the Lax problem at* $t = 0.16$*; right: the Sod problem at* $t = 0.1644$.

see that all the schemes give good nonoscillatory resolutions to the correct entropy solution for this problem.

Example 3.6. We solve the Euler equations (3.3) in one dimension with a Riemann initial condition for the Lax Problem:

$$(\rho, v, p) = (0.445, 0.698, 3.528) \text{ for } x \le 0; \qquad (\rho, v, p) = (0.5, 0, 0.571) \text{ for } x > 0.$$

And for the Sod Problem,

$$(\rho, v, p) = (1.0, 0, 1.0) \text{ for } x \le 0; \qquad (\rho, v, p) = (0.125, 0, 0.1) \text{ for } x > 0.$$

The computed density ρ is plotted at $t = 0.16$ for the Lax Problem and at $t = 0.1644$ for the Sod Problem against the exact solution. In Figure 3, we plot the densities by the WENO5-LW4 scheme, together with those by the WENO5-RK4 scheme, with $N = 200$ uniformly spaced grid points. We can see that the results by both schemes are nonoscillatory and comparable.

Example 3.7. The previous examples contain only shocks and simple smooth region solutions (almost piecewise linear) for which shock resolution is the main concern and usually a good second order nonoscillatory scheme would give satisfactory results.

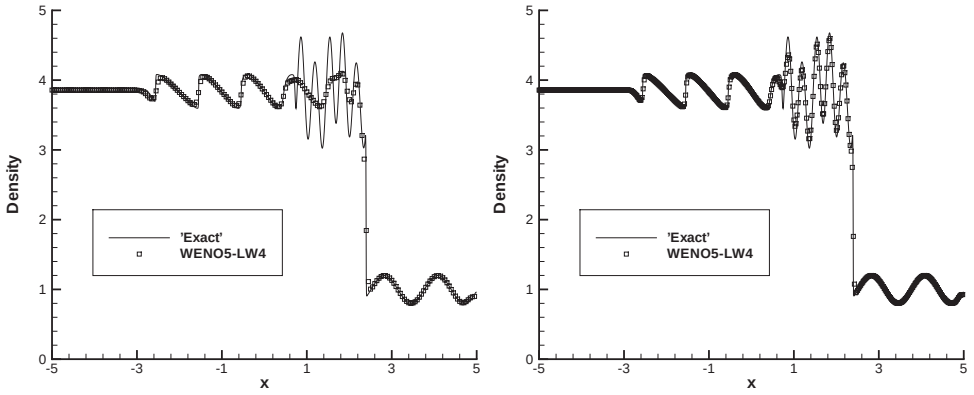

FIG. 4. *The shock density wave interaction problem.* $t = 1.8$. *by WENO5-LW4. Left:* $N = 200$ *grid points; right:* $N = 400$ *grid points.*

There is little advantage in using higher order schemes for such cases. We have been using them in the numerical experiments mainly to demonstrate the nonoscillatory properties of the high order schemes. A higher order scheme would show its advantage when the solution contains both shocks and complex smooth region structures. A typical example for this is the problem of shock interaction with entropy waves [17]. We solve the Euler equations (3.3) in one dimension with a moving Mach=3 shock interacting with sine waves in density, i.e., initially

$$(\rho, v, p) = (3.857143, 2.629369, 10.333333) \text{ for } x < -4;$$

$$(\rho, v, p) = (1 + \varepsilon \sin(5x), 0, 1) \text{ for } x \geq -4.$$

Here we take $\varepsilon = 0.2$. The computed density ρ is plotted at $t = 1.8$ against the reference solution, which is a converged solution computed by the fifth order finite difference WENO scheme [6] with 2000 grid points.

In Figure 4 we show the results of the WENO5-LW4 scheme with $N = 200$ (left) and $N = 400$ (right) grid points.

Example 3.8. We solve the same nonlinear Burgers equation (3.2) as in Example 3.2 with the same initial condition $u(x, y, 0) = 0.5 + \sin(\pi(x + y)/2)$, except that we now plot the results at $t = 1.5/\pi$ when a shock has already appeared in the solution. In Figure 5, the solutions of WENO5-LW4 with 80×80 grid points are shown. On the left we are showing a one-dimensional cut at $x = y$, and on the right we are showing the surface of the computed solution. We can see that the scheme gives nonoscillatory shock transitions for this problem.

Example 3.9. Double Mach reflection. This problem is originally from [20]. The computational domain for this problem is $[0, 4] \times [0, 1]$. The reflecting wall lies at the bottom, starting from $x = \frac{1}{6}$. Initially a right-moving Mach 10 shock is positioned at $x = \frac{1}{6}, y = 0$ and makes a 60^o angle with the x-axis. For the bottom boundary, the exact postshock condition is imposed for the part from $x = 0$ to $x = \frac{1}{6}$ and a reflective boundary condition is used for the rest. At the top boundary, the flow values are set to describe the exact motion of a Mach 10 shock. We compute the solution up to $t = 0.2$. In Figure 6 we show 30 equally spaced density contours from 1.5 to 22.7 computed by the WENO5-LW4 scheme. A "zoomed-in" graph is provided in Figure 7. Uniform meshes with grid points 960×239 and 1920×479 are used.

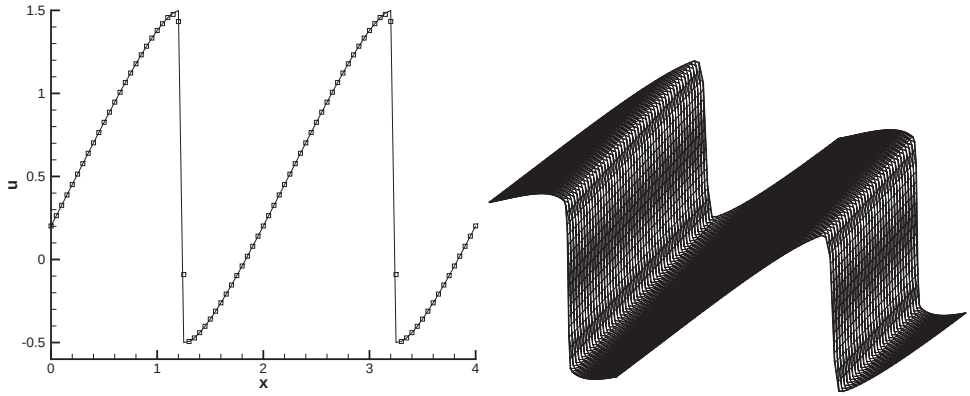

FIG. 5. *The two-dimensional Burgers equation.* $u(x,0) = 0.5 + \sin(\pi(x+y)/2)$. $t = 1.5/\pi$. 80×80 *grid points, by WENO*5*-LW*4. *Left: a cut of the solution at* $x = y$, *where the solid line is the exact solution and the squares are the computed solution; right: the surface of the solution.*

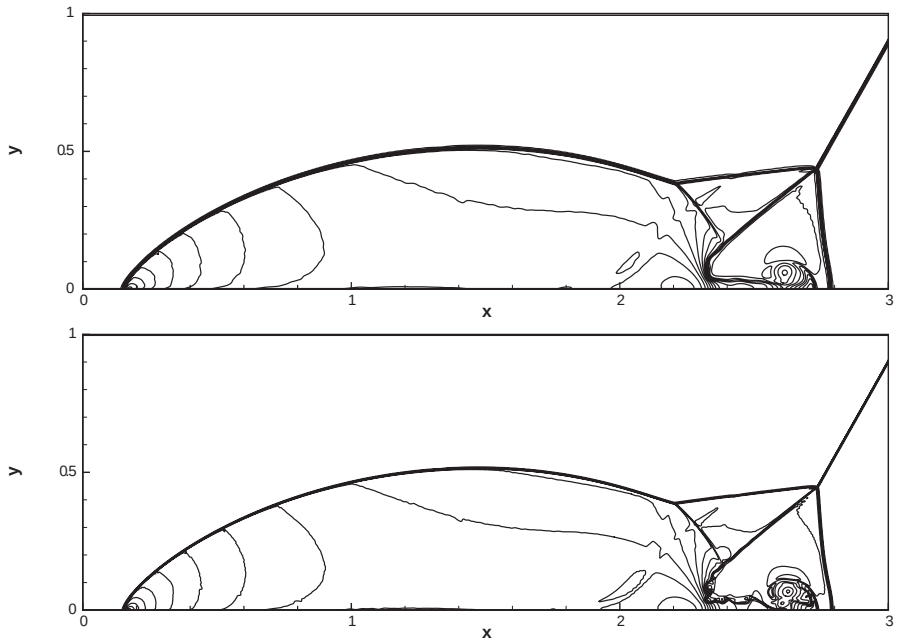

FIG. 6. *Double Mach reflection problem by WENO*5*-LW*4 *with* 960×239 *(top) and* 1920×479 *(bottom) grid points.* 30 *equally spaced contours from* 1.5 *to* 22.7 *for the density* ρ.

4. Concluding remarks. We have developed and implemented a Lax–Wendroff-type time discretization procedure for high order finite difference WENO schemes. After exploring a balance between reduction of cost and maintaining the nonoscillatory property, we have found that it suffices to use the regular WENO procedure, with the local characteristic decomposition for systems, only on the treatment of the first order time derivative terms. The higher order time derivative terms, when converted to spatial derivatives using the PDE, can be approximated by the inexpensive central difference formulas of suitable orders of accuracy. Extensive numerical experi-

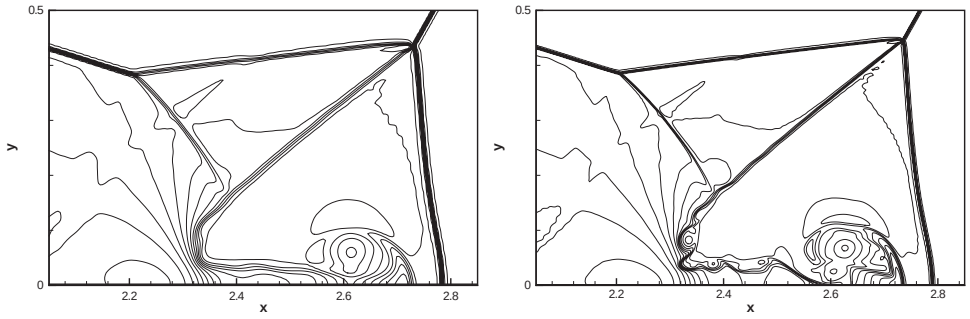

FIG. 7. *Zoomed-in figures. Double Mach reflection problem by WENO5-LW4 with* 960×239 *(left) and* 1920×479 *(right) grid points.* 30 *equally spaced contours from* 1.5 *to* 22.7 *for the density* ρ.

ments are performed to verify the accuracy and nonoscillatory shock resolution of this approach. A comparison with WENO schemes using Runge–Kutta time discretizations indicates that the schemes based on this Lax–Wendroff-type time discretization saves CPU time for certain problems, including the two-dimensional Euler systems of compressible gas dynamics. Of course, this procedure becomes progressively more complicated with more complicated PDEs and/or higher order time accuracy; hence we do not expect it to always be cost effective relative to the standard Runge–Kutta time discretizations.

REFERENCES

[1] D. BALSARA AND C.-W. SHU, *Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy*, J. Comput. Phys., 160 (2000), pp. 405–452.

[2] M. BEN-ARTZI AND J. FALCOVITZ, *A second-order Godunov-type scheme for compressible fluid dynamics*, J. Comput. Phys., 55 (1984), pp. 1–32.

[3] A. HARTEN, B. ENGQUIST, S. OSHER, AND S. CHAKRAVATHY, *Uniformly high order accurate essentially non-oscillatory schemes*, III, J. Comput. Phys., 71 (1987), pp. 231–303.

[4] O. FRIEDRICHS, *Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids*, J. Comput. Phys., 144 (1998), pp. 194–212.

[5] C. HU AND C.-W. SHU, *Weighted essentially non-oscillatory schemes on triangular meshes*, J. Comput. Phys., 150 (1999), pp. 97–127.

[6] G. JIANG AND C.-W. SHU, *Efficient implementation of weighted ENO schemes*, J. Comput. Phys., 126 (1996), pp. 202–228.

[7] P. D. LAX AND B. WENDROFF, *Systems of conservation laws*, Comm. Pure Appl. Math., 13 (1960), pp. 217–237.

[8] D. LEVY, G. PUPPO, AND G. RUSSO, *Central WENO schemes for hyperbolic systems of conservation laws*, M2AN Math. Model. Numer. Anal., 33 (1999), pp. 547–571.

[9] X. LIU, S. OSHER, AND T. CHAN, *Weighted essentially non-oscillatory schemes*, J. Comput. Phys., 115 (1994), pp. 200–212.

[10] M. LUKÁCOVÁ-MEDVID'OVÁ AND G. WARNECKE, *Lax-Wendroff type second order evolution Galerkin methods for multidimensional hyperbolic systems*, East-West J. Numer. Math., 8 (2000), pp. 127–152.

[11] J. QIU AND C.-W. SHU, *On the construction, comparison, and local characteristic decomposition for high order central WENO schemes*, J. Comput. Phys., 183 (2002), pp. 187–209.

[12] S. J. RUUTH AND R. J. SPITERI, *Two barriers on strong-stability-preserving time discretization methods*, J. Sci. Comput., 17 (2002), pp. 211–220.

[13] T. SCHWARTZKOPFF, C. D. MUNZ, AND E. F. TORO, *ADER: A high-order approach for linear hyperbolic systems in* $2D$, J. Sci. Comput., 17 (2002), pp. 231–240.

[14] C.-W. SHU, *Total-variation-diminishing time discretizations*, SIAM J. Sci. Statist. Comput., 9 (1988), pp. 1073–1084.

[15] C.-W. Shu, *Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws*, in Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, B. Cockburn, C. Johnson, C.-W. Shu, and E. Tadmor, A. Quarteroni, ed., Lecture Notes in Math. 1697, Springer-Verlag, Berlin, 1998, pp. 325–432.

[16] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock-capturing schemes*, J. Comput. Phys., 77 (1988), pp. 439–471.

[17] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schemes* II, J. Comput. Phys., 83 (1989), pp. 32–78.

[18] V. A. Titarev and E. F. Toro, *ADER: Arbitrary high order Godunov approach*, J. Sci. Comput., 17 (2002), pp. 609–618.

[19] E. F. Toro, *Primitive, conservative and adaptive schemes for hyperbolic conservation laws*, in Numerical Methods for Wave Propagation, E. F. Toro and J. F. Clarke, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998, pp. 323–385.

[20] P. Woodward and P. Colella, *The numerical simulation of two-dimensional fluid flow with strong shocks*, J. Comput. Phys., 54 (1984), pp. 115–173.

[21] X. Zhong, *Additive semi-implicit Runge-Kutta methods for computing high-speed nonequilibrium reactive flows*, J. Comput. Phys., 128 (1996), pp. 19–31.