ELECTRONIC RESEARCH ARCHIVE
Volume **29**, Number **1**, March **2021**

doi:10.3934/era.2020093
pp. **1819–1839**

A CONSERVATIVE SEMI-LAGRANGIAN FINITE DIFFERENCE WENO SCHEME BASED ON EXPONENTIAL INTEGRATOR FOR ONE-DIMENSIONAL SCALAR NONLINEAR HYPERBOLIC EQUATIONS

Guoliang Zhang and Shaoqin Zheng

School of Mathematical Sciences, Xiamen University
Xiamen, Fujian 361005, China

Tao Xiong*

School of Mathematical Sciences, Xiamen University
Fujian Provincial Key Laboratory of Mathematical Modeling
and High-Performance Scientific Computing
Xiamen, Fujian 361005, China

Special issue "High-order numerical methods for PDEs & applications"

Abstract. In this paper, we propose a conservative semi-Lagrangian finite difference (SLFD) weighted essentially non-oscillatory (WENO) scheme, based on Runge-Kutta exponential integrator (RKEI) method, to solve one-dimensional scalar nonlinear hyperbolic equations. Conservative semi-Lagrangian schemes, under the finite difference framework, usually are designed only for linear or quasilinear conservative hyperbolic equations. Here we combine a conservative SLFD scheme developed in [21], with a high order RKEI method [7], to design conservative SLFD schemes, which can be applied to nonlinear hyperbolic equations. Our new approach will enjoy several good properties as the scheme for the linear or quasilinear case, such as, conservation, high order and large time steps. The new ingredient is that it can be applied to nonlinear hyperbolic equations, e.g., the Burgers' equation. Numerical tests will be performed to illustrate the effectiveness of our proposed schemes.

1. **Introduction.** In this paper, we are interested in semi-Lagrangian (SL) finite difference (FD) schemes for solving nonlinear hyperbolic conservation laws (HCL)

$$\begin{cases} u_t + f(u)_x = 0, \\ u(x,0) = u_0(x). \end{cases} \tag{1}$$

The SL scheme updates the solution by following the characteristics, either forwardly [10, 25], or backwardly [19, 14], where for (1) the characteristic equation is defined as

$$\frac{dx}{dt} = f'(u(x,t)). \tag{2}$$

2020 *Mathematics Subject Classification.* 65M06, 65M25.

Key words and phrases. Semi-Lagrangian, finite difference WENO scheme, scalar nonlinear hyperbolic equations, Runge-Kutta exponential integrator, conservation.

The third author is supported by NSF grant of Fujian Province No. 2019J06002, NSFC Grant No. 11971025, Science Challenge Project No. TZ2016002, NSAF Grant No. U1630247.

* Corresponding author: Tao Xiong.

The classical backward SL scheme traces the characteristics back to a previous time level and then updates the solution with interpolation, or polynomial reconstruction. Based on its solution space, there are SL FD schemes [26, 29], SL finite volume (FV) schemes [9, 12], SL spectral element methods [32, 13], SL discontinuous Galerkin (DG) finite element methods [23, 3], SL particle methods [8], etc.

For SL schemes, mass conservation is an important property for certain applications, such as in kinetic simulations [9, 2], weather forecasting [27, 15], fluid dynamics [29, 12], etc. For finite volume schemes and finite element methods, it is more natural to enforce the discrete mass conservation, by working with the integral form for finite volume schemes [12], or the weak form for finite element methods [2]. For finite difference schemes, it is much more challenging [18]. A conservative SL finite difference scheme based on the flux difference form was proposed in [21]. A maximum principle preserving limiter was applied to ensure the L_∞ stability in [29]. However, the method in [21] only considers the linear or quasilinear advection equation with $f(u) = a(x,t)\,u$ in (1), where $a(x,t)$ is a known function. The extension to high dimensional problems is from Strang splitting. This method has later been generalized to nonlinear hyperbolic problems with the idea from an Eulerian-Lagrangian scheme [16]. On the other hand, a multi-dimensional SLFD scheme without dimensional splitting has been developed based on a predictor-corrector approach [20], and applied to the kinetic Vlasov-Poisson equation. It was later been generalized to the incompressible flow and Guiding-center model problems [30]. Here the predictor-corrector repeatedly utilizes the material derivatives and replaces time derivatives by spatial derivatives from the equation, in a similar idea of a Lax-Wendroff approach. A high order approximation of the characteristic equation is built upon low order predictions of its solution. A conservative correction of the scheme was proposed in [31], which is still based on the flux difference form, and the temporal integrals of the numerical fluxes are approximated by numerical quadrature rules. The scheme has also been used to solve the BGK model of the Boltzmann equation [1]. However, since it traces numerical quadrature points along the temporal integration, the fully discrete conservative scheme in [31, 1] will subject to time step restrictions, which has been analyzed by Fourier analysis in [31].

For SLFD schemes solving nonlinear hyperbolic problems, the main difficulty lies on the characteristic speed depending on the unknown solution, such as in (2), especially for high dimensional problems. Except the predictor-corrector approach in [20], there are also some other methods proposed to achieve high order in time. One is the class of multi-step schemes based on backward differential formulas (BDF), such as the Adams-Moulton and Adams-Bashforth schemes [14]. Another is based on commutator-free Runge-Kutta exponential integrators (RKEI) [5, 6]. However, both methods in literature do not have mass conservation. In this paper, we will adopt the RKEI method and combine it with the SLFD scheme in [21] to develop conservative SLFD-RKEI schemes for solving the nonlinear hyperbolic problems. The RKEI method has also recently been coupled with SLDG scheme for solving the Vlasov equations [2].

The RKEI method was first proposed in [7], which is designed for nonlinear ordinary differential equations (ODEs). The idea is to mimic the exponential solution for linear ODEs, and write the nonlinear function as a coefficient multiplying the unknown solution. Then the coefficient is frozen at previous given time levels, so that the resulting ODE have explicit exponential solutions. Later, the RKEI method

was realized to be equivalent to the semi-Lagrangian scheme for partial differential equations (PDEs) [5, 6]. In this paper, we propose to combine it with the conservative SLFD scheme [21] under the flux difference form. However, different from [31], the temporal integral is transferred to a spatial integral at the previous time level with know solutions, owing to the divergence theorem. WENO interpolations and numerical flux reconstructions [21] are used to ensure essentially non-oscillatory properties. Other methods, such as Hermite WENO schemes (HWENO) [4] may also be applied. However, we would emphasize that in order to achieve large time step condition, we have to combine the WENO interpolation and flux reconstruction, so that the common integration in the left and right fluxes can be canceled exactly [21, 18]. This cancellation typically due to the fact, for the one-dimensional problem the spatial integrals in the reconstructions of the left and right fluxes are along the same line (only one dimension). This is not the case for two and higher dimensions. The extension to high dimensional problems is not straightforward. Although dimensional splittings can still be used, they suffer from low order splitting errors and we do not pursue them here.

The rest of the paper is organized as follows. In Section 2, the model equation will be described. The SLFD scheme in [21] and the RKEI method [7] will first be recalled in Section 3, after that conservative SLFD schemes for solving (1) will be presented. In Section 4, numerical experiments will be performed to demonstrate the effectiveness of our proposed methods. Conclusion remarks will be made in Section 5.

2. **Model equation.**

We start with the hyperbolic equations in the advective form,

$$u_t + V(u)(x,t)\, u_x = 0, \tag{3}$$

where $u = u(x,t)$ and $V(u)(x,t)$ is the advection speed which may depend on the unknown u. The solution of (3) propagates along characteristics, which can be defined as

$$\frac{d}{dt}x(t) = V(u)(x(t),t). \tag{4}$$

When V does not depend on the unknown u, namely, in the linear or quasilinear form, the characteristics of (4) are given and can be computed a prior, so that the solution of (3) can be determined from the characteristics by tracing it back to its initial or boundary values. However, when V does depend on the unknown u, (3) becomes a nonlinear hyperbolic equation, e.g., in (1) $V(u)(x,t) = f'(u)$. Even initially the solution u is smooth, the characteristics are changing with the solution u. At some later time, the characteristics would intersect and discontinuities are formed, such as shock waves. This is well-known for nonlinear hyperbolic equations [17].

However, numerical schemes designed according to the advective form (3) cannot provide a correct shock speed for shock waves. It is important instead using the following conservative form

$$u_t + (a(u)(x,t)u)_x = 0. \tag{5}$$

(5) shares the same characteristic equation as (3) for the same problem. But only in the linear or quasilinear case, a would be the same as V. For nonlinear hyperbolic equations, usually it is not. For example, for the nonlinear Burgers' equation,

$$u_t + \left(\frac{u^2}{2}\right)_x = 0, \tag{6}$$

$V(u)(x,t) = u$ in (3), while $a(u)(x,t) = \frac{u}{2}$ in (5). However u is the right characteristic speed for both of them.

In the following, for our numerical scheme design, we consider (4) as the characteristic equation for (3). For the conservative equation (5), instead of (4), we use a pseudo characteristic equation

$$\frac{d}{dt}x(t) = a(u)(x(t),t). \tag{7}$$

Taking the Burgers' equation as an example, we consider

$$\frac{d}{dt}x(t) = u(x(t),t), \tag{8}$$

for (3), while

$$\frac{d}{dt}x(t) = \frac{1}{2}u(x(t),t), \tag{9}$$

for (5). Similarly for other nonlinear hyperbolic equations.

3. **Numerical scheme.**

In this section, we will describe how to design conservative SLFD WENO schemes, based on the RKEI method. We first recall the conservative SLFD scheme defined for the quasilinear hyperbolic equation in [21], then we will present the framework of RKEI method proposed in [5]. After that, we will show how to get conservative schemes by coupling these two methods.

3.1. **Conservative SLFD scheme.**

The conservative SLFD scheme starts from the conservative equation (5). Assuming we already have the solution at time level t_n, integrating (5) from t_n to t_{n+1}, we get

$$u(x,t_{n+1}) = u(x,t_n) - \left(\int_{t_n}^{t_{n+1}} a(u)(x,t)u dt\right)_x. \tag{10}$$

Denoting

$$H(x) = \int_{t_n}^{t_{n+1}} a(u)(x,t)u dt, \tag{11}$$

and defining

$$H(x) = \frac{1}{\Delta x}\int_{x-\frac{\Delta x}{2}}^{x+\frac{\Delta x}{2}} h(\xi)d\xi, \tag{12}$$

we obtain

$$H_x = \frac{1}{\Delta x}\left(h(x+\frac{\Delta x}{2}) - h(x-\frac{\Delta x}{2})\right), \tag{13}$$

where Δx could be referred as the spatial mesh size.

At a given point $x = x_i$, (10) becomes

$$u(x_i,t_{n+1}) = u(x_i,t_n) - \frac{1}{\Delta x}\left(h(x_i+\frac{\Delta x}{2}) - h(x_i-\frac{\Delta x}{2})\right). \tag{14}$$

A conservative finite difference scheme can be provided as follows

$$u_i^{n+1} = u_i^n - \frac{1}{\Delta x}(\widehat{h}_{i+\frac{1}{2}} - \widehat{h}_{i-\frac{1}{2}}), \tag{15}$$

where $\widehat{h}_{i+\frac{1}{2}}$ is a numerical flux approximating $h(x_i + \frac{\Delta x}{2})$ and u_i^n approximates $u(x_i, t_n)$ accordingly. $\widehat{h}_{i+\frac{1}{2}}$ can be obtained from $\{H_i\}$ via a high order finite difference WENO reconstruction, which will be described afterward. From (11), we have

$$H_i = H(x_i) = \int_{t_n}^{t_{n+1}} a(u)(x_i, t)u(x_i, t)dt. \tag{16}$$

It is important on how to get high order approximations of (16), as we only know the solution at time level t_n. In this paper, a semi-Lagrangian scheme based on solving the characteristic equations (7) is used.

Let's introduce a spatio-temporal region $\Omega_{i,t}$, which is associated to the point x_i and time levels from t_n to t_{n+1}. The region is displayed in Fig. 3.1. The curves ℓ_1 and ℓ_3 are along the time and space directions respectively, while ℓ_2 is the curve defined from the characteristic equation (4). From the definition, we know that the outward normal direction along ℓ_2 is $(-1/a, 1)$.

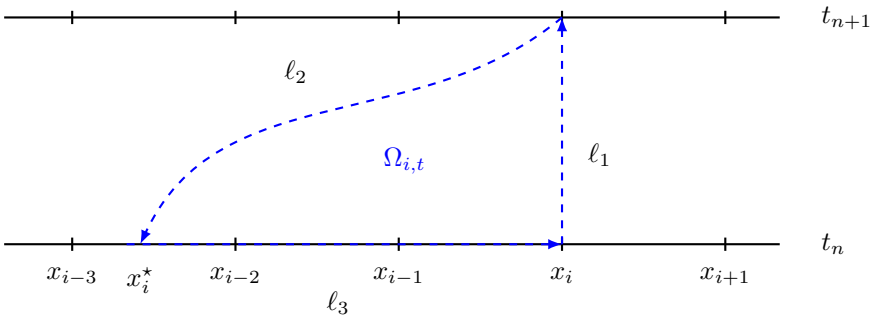

FIGURE 1. The diagram of the spatio-temporal region $\Omega_{i,t}$.

Now if we integrate the conservative equation (5) over the region $\Omega_{i,t}$, that is

$$\int_{\Omega_{i,t}} u_t + (a(u)(x,t)u)_x = 0. \tag{17}$$

Denoting $\nabla_{t,x} = (\partial t, \nabla_x)$, and from the divergence theorem, we get

$$\begin{aligned} 0 &= \int_{\Omega_{i,t}} u_t + (a(u)(x,t)u)_x \\ &= \int_{\Omega_{i,t}} \nabla_{t,x} \cdot (u, a(u)(x,t)u) \\ &= \int_{\partial\Omega_{i,t}} (u, a(u)(x,t)u) \cdot \mathbf{n} \\ &= -\int_{x_i^*}^{x_i} u(x, t_n)dx + \int_{t_n}^{t_{n+1}} a(u)(x_i, t)u(x_i, t)dt, \end{aligned}$$

where $\mathbf{n}$ is the outward unit normal vector along each curve, and this equation yields

$$H_i = \int_{t_n}^{t_{n+1}} a(u)(x_i, t)u(x_i, t)dt = \int_{x_i^*}^{x_i} u(x, t_n)dx. \tag{18}$$

So if we can solve the characteristic equation (7) to find the root x_i^*, we can do the integration in (18) to get the value H_i. After that, we reconstruct $\{\widehat{h}_{i+\frac{1}{2}}\}$ from $\{H_i\}$, the solution u_i^{n+1} is then updated from (15). We denote this conservative updating process as "SLFDc", that is

$$u^{n+1} = \text{SLFDc}(a(u)(x,t), \Delta t)u^n. \tag{19}$$

In [21], only $a(x,t)$ is considered. As $a(x,t)$ is given, classical Runge-Kutta methods can be used to solve (7). When $a(u)(x,t)$ depends on u, since we do not have the values u at intermediate stages, Runge-Kutta methods cannot be directly applied in the SL settings. In [31], a high order predictor-corrector approach is proposed. However, that approach is problem dependent and algebraically a little complicated to extend to higher order (e.g., 4th order and up). Here we will adopt the RKEI method developed in [5], which can be more conveniently extended to high order, and we will present it in the next subsection.

On the other hand, another type of SLFD scheme can base on the nonconservative advective form (3). This scheme is to solve the characteristic equation (4) to get x_i^*, then interpolate the value at x_i^* to update the solution u_i^{n+1} at the next time level, e.g., the approach used in [5]. However, due to interpolating errors, this type SLFD scheme does not have mass conservation. Here we refer this update as "SLFDn", that is

$$u^{n+1} = \text{SLFDn}(V(u)(x,t), \Delta t)u^n. \tag{20}$$

The nonconservative approach cannot get a right shock speed for nonlinear hyperbolic equations. Moreover, it is not appropriate for long time simulations [19]. In our numerical schemes proposed in the following, we could combine "SLFDc" and "SLFDn" to achieve a high order conservative SLFD scheme for the nonlinear hyperbolic equations.

3.2. **RKEI time method.** In this subsection, we first briefly review the RKEI method, which was proposed to solve nonlinear ODEs [7]. For more details, we refer to [7, 5, 2].

We consider a nonlinear initial value ODE problem of size N,

$$\frac{dY(t)}{dt} = C(Y)Y, \quad Y(t=0) = Y_0, \tag{21}$$

where $C(Y)$ is a matrix-value function of size $N \times N$, which may depend on the solution of $Y(t)$. The idea of RKEI method is to freeze the coefficient matrix $C(Y)$ at a given time step, or linear composition of explicitly known values, to obtain a linearized problem, and then apply the exponential integrator method to update the solution from Y^n to Y^{n+1}. Y^n is the numerical solution at time level t_n. For a first order scheme, it is

$$\frac{dY(t)}{dt} = C^n Y, \quad Y^{n+1} = \exp(C^n \Delta t)Y^n, \tag{22}$$

where $C^n = C(Y^n)$ and $\Delta t = t_{n+1} - t_n$ is the time step.

To achieve high order accuracy, a class of commutator-free RKEI methods with multi-stages can be used. For an s-stage RKEI method, the algorithm flow chart is summarized as follows

Algorithm 1: The commutator-free (CF) RKEI method [7].

$p = Y^n$
for $r = 1 : s$ **do**
$Y_r = \exp(\Delta t \sum_k \alpha^k_{rJ^{(r)}} C(Y_k)) \cdots \exp(\Delta t \sum_k \alpha^k_{r1} C(Y_k))p$
end
$Y^{n+1} = \exp(\Delta t \sum_k \beta^k_J C(Y_k)) \cdots \exp(\Delta t \sum_k \beta^k_1 C(Y_k))p$

In the subindex of $\alpha^k_{rJ^{(r)}}$, $J^{(r)}$ represents the number of exponentials one has to take per RK stage. The RKEI method can be represented by the Butcher tableau in Table 1.

$\mathbf{c}$	A
	$\mathbf{b}$

TABLE 1. A Butcher tableau for RKEI method, where $a_{ik} = \sum_{l=1}^{J^{(i)}} \alpha^k_{i,l}$ and $b_k = \sum_{l=1}^{J} \beta^k_l$, which merges $J^{(i)}$ rows into one row at each stage i.

Example 3.1. A 3rd order RKEI method in [5] can be represented by the following Butcher tableau,

0			
$\frac{1}{2}$	$\frac{1}{2}$		
1	-1	2	
	$\frac{1}{12}$	$\frac{1}{3}$	$-\frac{1}{4}$
	$\frac{1}{12}$	$\frac{1}{3}$	$\frac{5}{12}$

TABLE 2. CF3

with which, the RKEI scheme for the nonlinear ODE system (21) reads

$$\begin{aligned}
Y^{(1)} &= Y^n \\
Y^{(2)} &= \exp\left(\frac{1}{2}\Delta t C(Y^{(1)})\right) Y^n \\
Y^{(3)} &= \exp\left(\Delta t(-C(Y^{(1)}) + 2C(Y^{(2)}))\right) Y^n \\
Y^{n+1} &= \exp\left(\Delta t(\frac{1}{12}C(Y^{(1)}) + \frac{1}{3}C(Y^{(2)}) + \frac{5}{12}C(Y^{(3)}))\right) \\
&\quad \exp\left(\Delta t(\frac{1}{12}C(Y^{(1)}) + \frac{1}{3}C(Y^{(2)}) - \frac{1}{4}C(Y^{(3)}))\right) Y^n,
\end{aligned}$$

here $J^{(r)} = 1$ for $r = 1, \ldots, 3$ and $J = 2$.

Example 3.2. A 4th order RKEI method in [5] can be represented by the following Butcher tableau,

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	$\frac{1}{2}$	0	0	
	$-\frac{1}{2}$	0	1	
	$\frac{1}{4}$	$\frac{1}{6}$	$\frac{1}{6}$	$-\frac{1}{12}$
	$-\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{4}$

TABLE 3. CF4

correspondingly, the RKEI scheme for the nonlinear ODE system (21) reads

$$\begin{aligned}
Y^{(1)} =& Y^n \\
Y^{(2)} =& \exp\left(\frac{1}{2}\Delta t C(Y^{(1)})\right) Y^n \\
Y^{(3)} =& \exp\left(\frac{1}{2}\Delta t C(Y^{(2)})\right) Y^n \\
Y^{(4)} =& \exp\left(\Delta t(-\frac{1}{2}C(Y^{(1)}) + C(Y^{(3)}))\right) \exp\left(\frac{1}{2}\Delta t C(Y^{(1)})\right) Y^n \\
Y^{n+1} =& \exp\left(\Delta t(-\frac{1}{12}C(Y^{(1)}) + \frac{1}{6}C(Y^{(2)}) + \frac{1}{6}C(Y^{(3)}) + \frac{1}{4}C(Y^{(4)}))\right) \\
& \exp\left(\Delta t(\frac{1}{4}C(Y^{(1)}) + \frac{1}{6}C(Y^{(2)}) + \frac{1}{6}C(Y^{(3)}) - \frac{1}{12}C(Y^{(4)}))\right) Y^n,
\end{aligned}$$

here $J^{(r)} = 1$ for $r = 1, \ldots, 3$, $J^{(4)} = 2$ and $J = 2$.

3.3. **Conservative SLFD-RKEI scheme.** The RKEI method for linear ODEs is later recognized to be equivalent to an SL scheme for updating the solution for linear transport problems. The method is then coupled with spectral Galerkin method and applied to convection-diffusion problems [7], incompressible Navier-Stokes equations [6]. Very recently, it has been coupled with SLDG methods, and applied to the Vlasov-Poisson system and guiding-center model problems [2].

Here we will couple the RKEI method with the SLFD scheme, to get a high order SLFD-RKEI scheme for solving the nonlinear hyperbolic equations. The difficulty here is that for SL schemes under the finite difference framework, it is not easy to achieve the following properties simultaneously, high order, conservative, large time steps and ability to solve nonlinear hyperbolic equations. Our method proposed here for the one-dimensional hyperbolic equations, will have these properties.

Two types of conservative schemes can be proposed. One is that we only do a conservative correction at the last stage as compared to a nonconservative one. The other is we do it at every stage. Taking the 3rd order RKEI scheme in Table 2 as an example, the first SLFD-RKEI scheme reads as follows

$$\begin{aligned}
u^{(1)} =& u^n \\
u^{(2)} =& SLFDn\left(\frac{1}{2}V(u^{(1)})(x,t_n), \Delta t\right) u^n \\
u^{(3)} =& SLFDn\left(-V(u^{(1)})(x,t_n) + 2V(u^{(2)})(x,t_n+\frac{1}{2}\Delta t), \Delta t\right) u^n \\
u^{n+1} =& SLFDc\Big(\frac{1}{12}a(u^{(1)})(x,t_n) + \frac{1}{3}a(u^{(2)})(x,t_n+\frac{1}{2}\Delta t) \\
& + \frac{5}{12}a(u^{(3)})(x,t_n+\Delta t), \Delta t\Big) SLFDc\Big(\frac{1}{12}a(u^{(1)})(x,t_n) \\
& + \frac{1}{3}a(u^{(2)})(x,t_n+\frac{1}{2}\Delta t) - \frac{1}{4}a(u^{(3)})(x,t_n+\Delta t), \Delta t\Big) u^n. \qquad (23)
\end{aligned}$$

We denote this scheme as "SLFDc1". The other is

$$\begin{aligned}
u^{(1)} =& u^n \\
u^{(2)} =& SLFDc\left(\frac{1}{2}a(u^{(1)})(x,t_n), \Delta t\right) u^n
\end{aligned}$$

$$
\begin{aligned}
u^{(3)} =& SLFDc\left(-a(u^{(1)})(x,t_n)+2a(u^{(2)})(x,t_n+\frac{1}{2}\Delta t),\Delta t\right)u^n\\
u^{n+1} =& SLFDc\Big(\frac{1}{12}a(u^{(1)})(x,t_n)+\frac{1}{3}a(u^{(2)})(x,t_n+\frac{1}{2}\Delta t)\\
&+\frac{5}{12}a(u^{(3)})(x,t_n+\Delta t),\Delta t\Big)SLFDc\Big(\frac{1}{12}a(u^{(1)})(x,t_n)\\
&+\frac{1}{3}a(u^{(2)})(x,t_n+\frac{1}{2}\Delta t)-\frac{1}{4}a(u^{(3)})(x,t_n+\Delta t),\Delta t\Big)u^n. \qquad (24)
\end{aligned}
$$

and we denote it as "SLFDc2". The nonconservative scheme with SLFDn at every stage is still denoted as "SLFDn".

We note that for above schemes at each stage, we freeze $V(u)(x,t)$ or $a(u)(x,t)$ with linear composition of explicitly known values of u at its corresponding time stages, but leaving x as a changing variable. However we only know the values of u at grid points $\{x_i\}$. For x different from the grid points, an interpolation is needed. With the frozen characteristic speed, the characteristic equation (4) or (7), which now is an explicit ODE, can be solved by a high order Runge-Kutta method in a backward way. Let us take the solution to the characteristic equation as $X(t)$, and we introduce $w(X(t))$ as the velocity function. Detailed procedures for a first order scheme and a 3rd order scheme corresponding to (23) are given as follows:

- for a first order RKEI scheme, it has only one stage, the characteristic equation is

$$w(X(t)) = a(u)(X(t),t_n), \qquad (25)$$

$$\begin{cases} \frac{dX(t)}{dt} = w(X(t)), \\ X(t_{n+1}) = x_i. \end{cases} \qquad (26)$$

  It can be solved with an Euler forward scheme, that is, we let $w(x_i) = w(X(t_{n+1})) \approx w(X(t))$, then

$$x_i^* = x_i - w(x_i)\Delta t = x_i - a(u)(x_i,t_n)\Delta t. \qquad (27)$$

  After that we call "SLFDc" to update u^{n+1} from u^n;
- for a 3rd order RKEI scheme, it has three stages. For each stage, we have:
  - **Stage 1:** The characteristic equation is

$$w(X(t)) = \frac{1}{2}V(u)(X(t),t_n), \qquad (28)$$

$$\begin{cases} \frac{dX(t)}{dt} = w(X(t)), \\ X(t_{n+1/2}) = x_i. \end{cases} \qquad (29)$$

    We solve it with a 3rd order Runge-Kutta scheme to get

$$\begin{cases} x_i^{*,(1)} = X(t_n) \\ u_i^{(1)} = \mathcal{I}u(x_i^{*,(1)},t_n) \approx u(x_i, t_{n+\frac{1}{2}}). \end{cases} \qquad (30)$$

  - **Stage 2:** The characteristic equation is

$$w(X(t)) = -V(u)(X(t),t_n) + 2V(u)(X(t),t_n+\frac{1}{2}\Delta t)), \qquad (31)$$

$$\begin{cases} \frac{dX(t)}{dt} = w(X(t)), \\ X(t_{n+1}) = x_i. \end{cases} \qquad (32)$$

We solve it with a 3rd order Runge-Kutta scheme to get

$$\begin{cases} x_i^{*,(2)} = X(t_n) \\ u_i^{(2)} = \mathcal{I}u(x_i^{*,(2)}, t_n) \approx u(x_i, t_{n+1}). \end{cases} \tag{33}$$

– **Stage 3:** This stage has two steps. The first step has the characteristic equation

$$w(X(t)) = \frac{1}{12}\Big(a(u)(X(t), t_n) + 4a(u)(X(t), t_n + \frac{1}{2}\Delta t) - 3a(u)(X(t), t_n + \Delta t)\Big), \tag{34}$$

$$\begin{cases} \frac{dX(t)}{dt} = w(X(t)), \\ X(t_{n+1}) = x_i. \end{cases} \tag{35}$$

We solve it with a 3rd order Runge-Kutta scheme to get

$$x_i^* = X(t_n). \tag{36}$$

Then we call "SLFDc" to obtain $u^{(3)}$ based on u^n. The second step has the characteristic equation

$$w(X(t)) = \frac{1}{12}\Big(a(u)(X(t), t_n) + 4a(u)(X(t), t_n + \frac{1}{2}\Delta t) + 5a(u)(X(t), t_n + \Delta t)\Big), \tag{37}$$

$$\begin{cases} \frac{dX(t)}{dt} = w(X(t)), \\ X(t_{n+1}) = x_i. \end{cases} \tag{38}$$

Similarly, we solve it with a 3rd order Runge-Kutta scheme to get

$$x_i^* = X(t_n), \tag{39}$$

and we call "SLFDc" to update u^{n+1}. Here it is starting from $u^{(3)}$ instead of u^n.

In Stage 1 and Stage 2, $\mathcal{I}u(x_i^{(\ell)}, t_n)$ for $\ell = 1, 2$ is an linear interpolation operator. Corresponding to the 5th order WENO reconstruction in the Appendix A.2, a 6th order interpolation with equal points on both sides of $x_i^{(\ell)}$ is used.

A similar procedure can be applied to other types of RKEI schemes, e.g., the 4th order in Table 3. "SLFDc2" will have every stage as Stage 3 above. Here we all omit them to save space.

4. Numerical test.

In this section, numerical examples for one dimensional linear and nonlinear hyperbolic equations are presented to verify the effectiveness of our algorithms. Three schemes are considered. For the 3rd order RKEI scheme (2), "SLFDc1" corresponds to (23), "SLFDc2" corresponds to (24), and "SLFDn" will have "SLFDn" instead of "SLFDc" in the last stage of (23). The schemes with 4th order RKEI (3) can be defined correspondingly.

The formal 5th order finite difference WENO reconstruction in the Appendix A.2 is used, which is a combination of the interpolation and flux reconstruction, and in this way we will not subject to time step restrictions. We note here this 5th order WENO scheme achieves 5th order when $a(u)(x, t)$ is a constant, but gives 3rd order otherwise, as no optimal linear weights exist. For more details, we refer to

the appendix or [19]. We denote the mesh size as N, the final time T and the time step is taken as

$$\Delta t = \mathrm{CFL}\,\frac{\Delta x}{\max_x |a(u)(x,t)|}. \tag{40}$$

Here "CFL" refers to the CFL number as used in an Eulerian scheme.

4.1. **Linear case.** First we consider the following linear equation with $a(u)(x,t) = 1$

$$u_t + u_x = 0, \quad x \in (0,2). \tag{41}$$

We take the initial condition $u(x,0) = \sin(\pi x)$ and periodic boundary condition. The corresponding exact solution is $u(x,t) = \sin(\pi x - \pi t)$. The final time is set to be $T = 2.5$ and CFL $= 4.5$ is used.

For this example, the characteristic equation (7) can be solved exactly, no temporal errors exist. This example is used to test the spatial orders. In Table 1, we show the errors in L_1 and L_∞ norms and the corresponding orders. We can observe that the 5th orders are obtained as expected.

TABLE 1. Numerical errors and orders for the linear problem (41). $T = 2.5$ and CFL $= 4.5$ in (40).

N	L_1 error	Order	L_∞ error	Order
20	1.25E-4	–	2.07E-4	–
40	3.83E-6	5.03	7.8E-6	4.73
80	1.15E-7	5.06	2.38E-7	5.04
160	3.55E-9	5.00	7.29E-9	5.03
320	1.10E-10	5.01	2.00E-10	5.19
640	3.42E-12	5.01	6.03E-12	5.05

4.2. **Quasilinear case.** In this example, we consider the quasilinear hyperbolic equation with a variable coefficient

$$u_t + (\sin(x)\,u)_x = 0, \quad x \in (0, 2\pi). \tag{42}$$

If we take the initial condition $u(x,0) = 1$ and periodic boundary condition, the exact solution can be given by [21]

$$u(x,t) = \frac{\sin(2\arctan(e^{-t}\tan(\frac{x}{2})))}{\sin(x)}. \tag{43}$$

For this example, the RKEI method for solving the characteristic equation (7) becomes the classic RK method. We consider both the 3rd order RKEI scheme with Butcher tableau (2) and the 4th order RKEI scheme with Butcher tableau (3).

For this quasilinear case, a large CFL number is still allowed. We take CFL $= 4.5$ and compute the solution up to time $T = 1.5$. In Table 2, we show the errors in L_1 and L_∞ norms as mesh refinements, with their corresponding convergence orders. As we can see, with the mesh much refined, 3rd order accuracies are obtained for both 3rd order RKEI and 4th order RKEI schemes, which demonstrate the leading order is 3rd order in space.

For this problem, we also try to test the temporal orders. We take a very refined mesh size $N = 900$. In order to clearly see the temporal orders, we take very large

CFL numbers from 26 to 30 so that the temporal errors dominate. The errors in L_1 and L_∞ norms and the corresponding orders are presented in Table 3. The desired temporal orders are obtained.

TABLE 2. Numerical errors and orders for the quasilinear case (42). $T = 1.5$ and CFL $= 4.5$ in (40).

	3rd order Scheme				4th order scheme			
N	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
80	6.98E-4	–	4.25E-3	–	9.99E-5	–	1.30E-3	–
160	9.11E-5	2.94	6.15E-4	2.79	6.07E-6	4.04	2.00E-4	2.71
320	1.17E-5	2.96	7.92E-5	2.96	2.81E-7	4.43	1.62E-5	3.62
640	1.53E-6	2.94	1.08E-5	2.88	2.28E-8	3.63	1.27E-6	3.67
1280	1.95E-7	2.97	1.69E-6	2.95	3.09E-9	2.88	6.38E-8	4.32
2560	2.47E-8	2.98	1.77E-7	2.98	4.46E-10	2.79	7.59E-9	3.07

TABLE 3. Numerical errors and orders for the quasilinear case (42) with different CFL numbers. $N = 900$ and $T = 1.5$.

CFL	3rd order RKEI (2)				4th order RKEI (3)			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
40	1.54E-5	–	2.39E-5	–	5.95E-8	–	8.80E-8	–
39	1.39E-5	3.02	2.19E-5	2.58	5.13E-8	4.37	7.78E-8	3.64
38	1.25E-5	3.03	1.99E-5	2.73	4.41E-8	4.31	6.85E-8	3.63
37	1.12E-5	3.02	1.74E-5	3.69	3.88E-8	3.52	5.76E-8	4.77
36	1.00E-5	3.00	1.57E-5	2.73	3.29E-8	4.37	5.03E-8	3.59

4.3. **Nonlinear case.** In the third example, we consider the nonlinear hyperbolic equation. We simply choose the Burger's equation

$$u_t + \left(\frac{u^2}{2}\right)_x = 0, \quad x \in (0, 2). \tag{44}$$

We take the initial condition $u(x, 0) = 0.5 + \sin(\pi x)$ and periodic boundary condition. For this example, since it is a nonlinear hyperbolic problem, we only take relatively large CFL numbers up to 3 in the following, although the scheme is designed to be stable for arbitrary large CFL numbers.

We first compute the solution up to $T = \frac{0.5}{\pi}$ when the solution is still smooth [17]. For this nonlinear problem, the RKEI scheme will play its role. First we show the errors and orders for the three schemes "SLFDn", "SLFDc1" and "SLFDc2", with L_1 and L_∞ norms in Table 4 and Table 5 for the 3rd order RKEI scheme (2), and with L_1 and L_∞ norms in Table 6 and Table 7 for the 4th order RKEI scheme (3), respectively. For the nonlinear problem, we take CFL $= 1.5$. At least 3rd order convergence for three schemes are all obtained. However, if we compare the absolute errors, we can find that generally SLFDn has the smallest errors, SLFDc2 has the largest, while SLFDc1 is in between. This is expected, as nonconservative updating involves less interpolation and reconstruction errors than the conservative one. So SLFDc1 is better than SLFDc2 on this aspect.

Then we would like to verify that the three schemes coupled with RKEI, can achieve high order in time for this nonlinear problem. In Table 8, we show the errors and orders for the nonconservative SLFDn scheme with 3rd order RKEI (2) and 4th order RKEI (3), relatively large CFL numbers from 2.6 to 3 are used. We take $N = 800$ and we measure the errors on the grid points $|N - 400| > 100$, as we can find that around the position where $u = 0$, it has large errors. This is due to the fact that at the position where $u = 0$, the RKEI scheme directly gives $x_i^* = x_i$, which means the position $u = 0$ will not move in the nonconservative scheme. If the initial values happened to not contain this point, it still affects the solution close to it. We show another case evolving to shock solutions afterward. We plot the errors for $N = 320$ in Fig. 1, for the 3rd order RKEI scheme and 4th order RKEI scheme respectively. The plots clearly shows large errors around $u = 0$. Similarly for the following conservative schemes. With the measurement away from $u = 0$, we can see 3rd order and 4th order can be obtained.

We now compare the errors and orders for the two conservatives schemes, with 3rd order RKEI (2) and 4th order RKEI (3), in Table 9 and Table 10, respectively. We take CFL numbers from 1.6 to 2 for the 3rd order and 2.6 to 3 for the 4th order. We take $N = 800$ and the errors are measured in the same way as above. We can find that the conservative scheme SLFDc2 with conservative updating at each stage can achieve 3rd order and almost 4th order in time. Note here we only have 3rd order in space, the order will degenerate as CFL getting smaller. The SLFDc1 scheme seems to lose order as compared to SLFDc2. However, we should notice that SLFDc1 has smaller absolute errors as compared to SLFDc2. The lost order might be due to the pollute errors around $u = 0$. But it is still more accurate in time as compared to SLFDc2. From the tables 8-10 and Fig. 1, we can also find that the 4th order RKEI has much better accuracy than the 3rd order RKEI.

Finally we compute the solution to $T = \frac{2}{\pi}$, where the shock wave is already formed. We take CFL $= 1.5$ and $N = 80$. We compare the two conservative schemes with the nonconservative scheme in Fig. 2, where the 3rd order RKEI scheme with (2) and 4th order scheme with (3) are used respectively. We simply take a 2nd order linear interpolation for the nonconservative scheme. We can see that the two conservative schemes can capture the correct shock speed for both cases. However, the nonconservative scheme cannot, which is essentially the limitation of nonconservative schemes for hyperbolic conservation laws [17].

We further consider another initial condition $\sqrt{2}/2+\sin(\pi x)$, and we take $N = 88$. In this case, there is one grid where $u = 0$ initially. We take CFL $= 1.5$ and compute the solution to $T = 2/\pi$ with these three schemes. In Fig. 3, we show the results at $T = 0.7/\pi$, $1.3/\pi$ and $2/\pi$. At $T = 0.7/\pi$, the solution is still smooth, all three schemes work well. Then at $T = 1.3/\pi$, the shock forms, both conservative schemes capture the shock very well, however we can find that the nonconservative scheme keeps a point with value to be 0. As the time further evolves to $T = 2/\pi$, the point with value 0 is still there for the nonconservative scheme, and it causes large oscillations of the solution, also the shock front is not valid. They are not the case for the two conservative schemes. The reason is what we mentioned above, when $u = 0$, the nonconservative scheme cannot evolve this point. These two tests with shock waves clearly show the advantage and robustness of our conservative approach over the nonconservative one.

Remark 1. For nonlinear hyperbolic problems, large time steps might be beneficial when the dominant time scales are much larger than the characteristic-based time

TABLE 4. Numerical L_1 errors and orders for the nonlinear Burgers' equation (44). $T = 0.5/\pi$ and CFL $= 1.5$ in (40). 3rd order RKEI (2) is used.

	SLFDn		SLFDc1		SLFDc2	
N	L_1 error	Order	L_1 error	Order	L_1 error	Order
40	6.40E-5	–	1.36E-4	–	2.24E-4	–
80	7.42E-6	3.11	1.03E-5	3.72	2.30E-5	3.28
160	9.57E-7	2.95	9.29E-7	3.47	2.58E-6	3.16
320	1.35E-7	2.83	1.04E-7	3.16	3.24E-7	2.99
640	1.73E-8	2.97	1.17E-8	3.15	3.93E-8	3.04

scales [11]. However, when discontinuous solutions exist, such as shock waves are formed, large time steps might still be allowed in some circumstances, e.g., if you track the characteristics very accurately to get an entropy satisfying solution and use total variation bounded (TVB) reconstructions [22]. Here we simply take a relatively large CFL $= 1.5$ without pursuing this aspect, and numerical tests show it works in our current settings.

TABLE 5. Numerical L_∞ errors and orders for the nonlinear Burgers' equation (44). $T = 0.5/\pi$ and CFL $= 1.5$ in (40). 3rd order RKEI (2) is used.

	SLFDn		SLFDc1		SLFDc2	
N	L_∞ error	Order	L_∞ error	Order	L_∞ error	Order
40	3.35E-4	–	9.72E-4	–	1.20E-3	–
80	4.75E-5	2.82	7.51E-5	3.69	1.30E-4	3.26
160	6.17E-6	2.95	6.16E-6	3.61	1.59E-5	3.03
320	8.72E-7	2.82	7.09E-7	3.12	2.19E-6	2.86
640	1.10E-7	2.98	7.64E-8	3.22	2.66E-7	3.04

TABLE 6. Numerical L_1 errors and orders for the nonlinear Burgers' equation (44). $T = 0.5/\pi$ and CFL $= 1.5$ in (40). 4th order RKEI (3) is used.

	SLFDn		SLFDc1		SLFDc2	
N	L_1 error	Order	L_1 error	Order	L_1 error	Order
40	5.19E-6	–	1.12E-4	–	1.12E-4	–
80	2.41E-7	4.43	7.04E-6	3.99	6.64E-6	4.08
160	1.09E-8	4.47	4.67E-7	3.91	3.95E-7	4.07
320	5.96E-10	4.19	4.16E-8	3.49	2.97E-8	3.74
640	3.23E-11	4.21	4.06E-9	3.36	2.29E-9	3.69

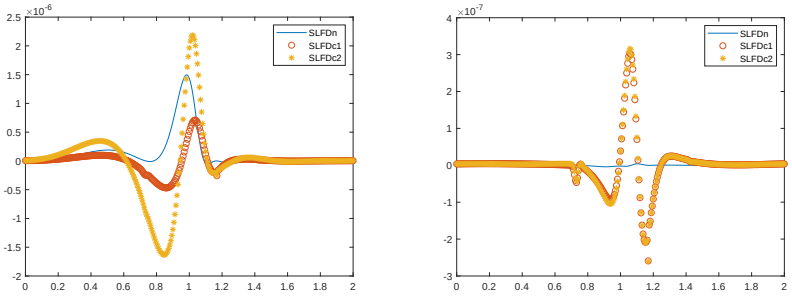

FIGURE 1. Numerical errors as compared to the exact solution for three schemes at $T = 0.5/\pi$ for the nonlinear Burgers' equation (44). The solid line is nonconservative scheme SLFDn; the symbol "o" is the conservative scheme SLFDc1 and the symbol "*" is the conservative scheme SLFDc2. CFL = 1.5 and $N = 320$. Left: 3rd order RKEI (2); right: 4th order RKEI (3).

TABLE 7. Numerical L_∞ errors and orders for the nonlinear Burgers' equation (44). $T = 0.5/\pi$ and CFL = 1.5 in (40). 4th order RKEI (3) is used.

	SLFDn		SLFDc1		SLFDc2	
N	L_∞ error	Order	L_∞ error	Order	L_∞ error	Order
40	5.66E-5	–	9.66E-4	–	1.00E-3	–
80	2.80E-6	4.34	6.27E-5	3.95	6.33E-5	3.99
160	8.62E-8	5.02	3.94E-6	3.99	3.82E-6	4.05
320	4.01E-9	4.42	3.43E-7	3.52	3.16E-7	3.59
640	2.82E-10	3.83	2.79E-8	3.62	2.38E-8	3.73

TABLE 8. Numerical errors and orders for the nonlinear Burgers' equation (44) from different CFL numbers, with SLFDn, 3rd order (2) and 4th order (3) in time, respectively. 4th order linear interpolation is used. $N = 800$ and $T = 0.5/\pi$.

CFL	3rd order SLFDn				4th order SLFDn			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
3	4.51E-8	–	9.66E-8	–	2.33E-10	–	2.01E-9	–
2.9	4.07E-8	3.03	8.72E-8	3.02	2.03E-10	4.07	1.74E-9	4.23
2.8	3.72E-8	2.56	7.93E-8	2.71	1.88E-10	2.19	1.65E-9	1.52
2.7	3.32E-8	3.13	7.09E-8	3.08	1.59E-10	4.61	1.39E-9	4.54
2.6	2.94E-8	3.22	6.30E-8	3.13	1.33E-10	4.73	1.15E-9	5.02

TABLE 9. Numerical errors and orders for the nonlinear Burgers' equation (44) from different CFL numbers, with the two 3rd order RKEI conservative schemes. $N = 800$ and $T = 0.5/\pi$.

CFL	3rd order SLFDc2				3rd order SLFDc1			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
2	4.67E-8	–	3.20E-7	–	3.07E-8	–	1.28E-7	–
1.9	4.08E-8	2.63	2.84E-7	2.33	2.78E-8	1.93	1.16E-7	1.92
1.8	3.50E-8	2.84	2.45E-7	2.73	2.45E-8	2.33	9.83E-8	3.06
1.7	2.95E-8	2.99	2.05E-7	3.12	2.21E-8	1.81	8.98E-8	1.58
1.6	2.44E-8	3.13	1.67E-7	3.38	1.99E-8	1.73	8.21E-8	1.48

TABLE 10. Numerical errors and orders for the nonlinear Burgers' equation (44) from different CFL numbers, with the two 4th order RKEI conservative schemes. $N = 800$ and $T = 0.5/\pi$.

CFL	4th order SLFDc2				4th order SLFDc1			
	L_1 error	Order	L_∞ error	Order	L_1 error	Order	L_∞ error	Order
3	5.46E-10	–	2.15E-9	–	2.53E-10	–	7.94E-10	–
2.9	4.75E-10	4.11	1.70E-9	6.93	2.37E-10	1.92	6.58E-10	5.54
2.8	4.16E-10	3.78	1.52E-9	3.19	2.32E-10	0.61	5.65E-10	4.34
2.7	3.66E-10	3.52	1.29E-9	4.51	2.23E-10	1.09	4.51E-10	6.20
2.6	3.25E-10	3.15	1.08E-9	4.71	2.15E-10	0.97	3.44E-10	7.17

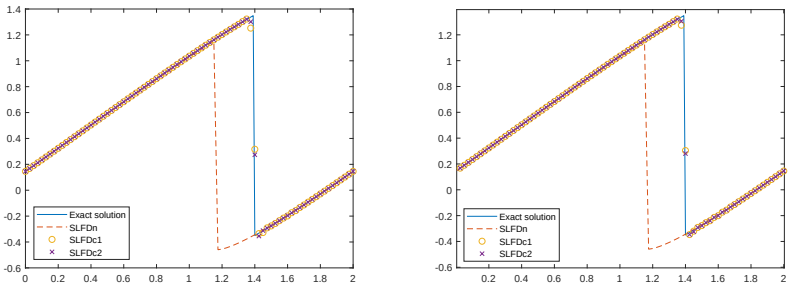

FIGURE 2. The shock wave solution at $T = 2/\pi$ for the nonlinear Burgers' equation (44). The solid line is the exact solution. The dashed line is the 2nd order nonconservative scheme SLFDn; the symbol "o" is the conservative scheme SLFDc1 and the symbol $\times$ is the conservative scheme SLFDc2. CFL = 1.5. Left: 3rd order RKEI (2); right: 4th order RKEI (3).

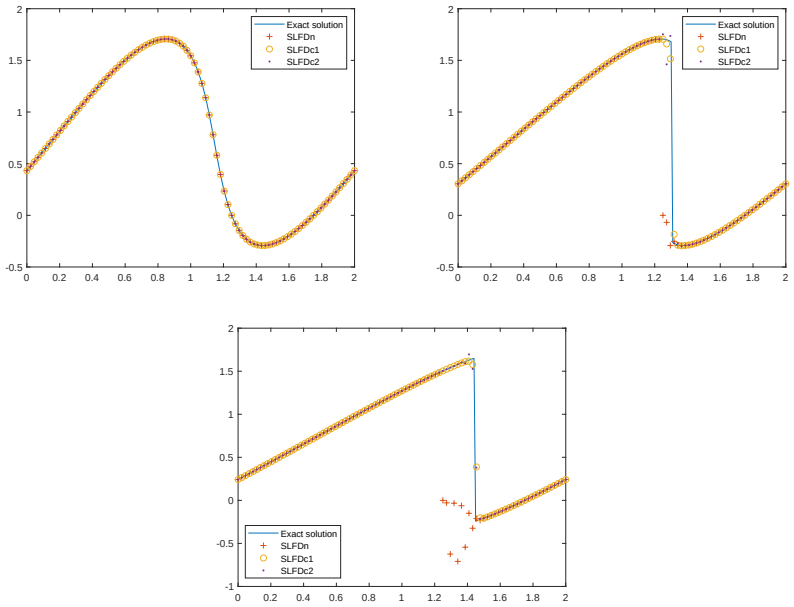

FIGURE 3. Numerical solutions for the nonlinear Burgers' equation (44) with initial condition $\sqrt{2}/2 + \sin(\pi x)$. The solid line is the exact solution. The dashed line is the 2nd order nonconservative scheme SLFDn; the symbol "o" is the conservative scheme SLFDc1 and the symbol "+" is the conservative scheme SLFDc2. CFL = 1.5 and $N = 88$. 3rd order RKEI is used. Top left: $T = 0.7/\pi$; top right: $T = 1.3/\pi$; bottom: $T = 2/\pi$.

5. **Conclusions.** In this paper, we propose two conservative SLFD schemes for nonlinear scalar hyperbolic conservation laws, which is a combination of the conservative SLFD scheme developed for the quasilinear equation and RKEI scheme for nonlinear ODEs. The resulting schemes are conservative, allow large CFL numbers and can deal with the nonlinear equation. In our numerical tests, we have verified that the conservative schemes can achieve the same error levels as the nonconservative one. 3rd and 4th order in time can be obtained. However, for shock wave solution, the conservative schemes are more robust in shock capturing and avoid the actionless point $u = 0$. Besides, the conservative scheme with conservation correction at the last stage shows to be better than the other one with conservation updating at every stage. However, in this work, only one-dimensional problems are considered. The extension to high dimensions is not straightforward, due to the difficulty appeared in the spatial reconstructions. This will be subject to our future work.

Another interesting direction would be applying the conservative SLFD schemes for hyperbolic systems, e.g., the shallow water equations. However, current SL schemes for hyperbolic systems are, e.g., either under the Lagrangian framework which is similar to Lagrangian schemes [28], or updating the Riemann invariants along characteristics [33]. However, the two approaches both are nonconservative and they resort to Eulerian approach for mass conservation. So applying our current

conservative SLFD schemes to hyperbolic system is highly nontrivial, and we leave it to our future investigation.

Appendix A. In this appendix, for completeness, we recall the finite difference WENO reconstruction in [19]. After we solve the characteristic equation (7) to get x_i^*, we need to reconstruct a polynomial of u to do the integration in (18), we denote this procedure as R_1. Then from $\{H_i\}$, we will reconstruct $\{h^{\pm}_{i+\frac{1}{2}}\}$ based on the direction of the characteristics, which is denoted as R_2. The final flux $\{\hat{h}_{i+\frac{1}{2}}\}$ in (15) is obtained by combining these two reconstructions, which we denote as

$$R = R_2 \circ R_1. \tag{A.1}$$

We note that it is important to combine these two reconstructions, especially to achieve large time step without subject to stability restrictions.

For simplicity, we denote the characteristic speed in (7) simply as a.

A.1. **First order reconstruction.** We split it into two cases, $|a|\Delta t \le \Delta x$ and $|a|\Delta t > \Delta x$:

Case 1. $|a|\Delta t \le \Delta x$: if $a \ge 0$, $x_i^* \le x_i$,

$$\hat{h}_{i+\frac{1}{2}} = h^-_{i+\frac{1}{2}} = (x_i - x_i^*)u_i^n \approx H_i = \int_{x_i^*}^{x_i} u(x,t_n)dx; \tag{A.2}$$

if $a < 0$, $x_i^* > x_i$,

$$\hat{h}_{i+\frac{1}{2}} = h^+_{i+\frac{1}{2}} = (x_i - x_i^*)u_{i+1}^n \approx H_i = \int_{x_i^*}^{x_i} u(x,t_n)dx. \tag{A.3}$$

Case 2. $|a|\Delta t > \Delta x$, let i^* be the index satisfying $x_i^* \in (x_{i^*-1}, x_{i^*}]$:

when $a \ge 0$,

$$\hat{h}_{i+\frac{1}{2}} = \sum_{j=i^*+1}^{i} u_j^n \Delta x + (x_{i^*} - x_i^*)u_{i^*}^n \approx H_i = \int_{x_i^*}^{x_i} u(x,t_n)dx; \tag{A.4}$$

when $a < 0$,

$$\hat{h}_{i+\frac{1}{2}} = \sum_{j_{i^*+1}}^{i} u_j^n \Delta x + (x_{i^*} - x_i^*)u_{i^*}^n \approx H_i = \int_{x_{i^*}}^{x_i} u(x,t_n)dx. \tag{A.5}$$

A.2. **Fifth order reconstruction.** We only consider $a > 0$, the case $a < 0$ can be obtained symmetrically.

First, we consider $a\Delta t \le \Delta x$. One chooses three three-point stencils

$$S_1 = \{u_{i-2}^n, u_{i-1}^n, u_i^n\},\ S_2 = \{u_{i-1}^n, u_i^n, u_{i+1}^n\},\ S_3 = \{u_i^n, u_{i+1}^n, u_{i+2}^n\}. \tag{A.6}$$

The reconstruction $R = R_1 \circ R_2$ is as follows:

1. reconstruction $h^{(j)}(x_{i+\frac{1}{2}})$ for $j = 1, 2, 3$:

$$\begin{aligned} h^{(1)}(x_{i+\frac{1}{2}}) = \ & \Delta x \Big((1/3\xi_{-2} + 1/4\xi_{-2}^2 + 1/18\xi_{-2}^3 - 7/24\xi_{-1}^2 - 7/36\xi_{-1}^3 - 11/24\xi_0^2 \\ & + 11/36\xi_0^3)u_{i-2}^n + (-1/3\xi_{-2}^2 - 1/9\xi_{-2}^3 - 7/6\xi_{-1} + 7/18\xi_{-1}^3 \\ & + 11/6\xi_0^2 - 11/18\xi_0^3)u_{i-1}^n + (1/12\xi_{-2}^2 + 1/18\xi_{-2}^3 + 7/24\xi_{-1}^2 \\ & - 7/36\xi_{-1}^3 + 11/6\xi_0 - 11/8\xi_0^2 + 11/36\xi_0^3)u_i^n \Big), \end{aligned}$$

$$\begin{aligned}h^{(2)}(x_{i+\frac{1}{2}}) = \ & \Delta x\Big((-1/6\xi_{-1} - 1/8\xi_{-1}^2 - 1/36\xi_{-1}^3 + 5/24\xi_0^2 + 5/36\xi_0^3 - 1/12\xi_1^2 \\ & + 1/18\xi_1^3)u_{i-1}^n + (1/6\xi_{-1}^2 + 1/18\xi_{-1}^3 + 5/6\xi_0 - 5/18\xi_0^3 + 1/3\xi_1^2 \\ & - 1/9\xi_1^3)u_i^n + (-1/24\xi_{-1}^2 - 1/36\xi_{-1}^3 - 5/24\xi_0^2 + 5/36\xi_0^3 + 1/3\xi_1 \\ & - 1/4\xi_1^2 + 1/18\xi_1^3)u_{i+1}^n\Big),\end{aligned}$$

$$\begin{aligned}h^{(3)}(x_{i+\frac{1}{2}}) = \ & \Delta x\Big((1/3\xi_0 + 1/4\xi_0^2 + 1/18\xi_0^3 + 5/24\xi_1^2 + 5/36\xi_1^3 + 1/24\xi_2^2 \\ & - 1/36\xi_2^3)u_i^n + (-1/3\xi_0^2 - 1/9\xi_0^3 + 5/6\xi_1 - 5/18\xi_1^3 - 1/6\xi_2^2 \\ & + 1/18\xi_2^3)u_{i+1}^n + (1/12\xi_0^2 + 1/18\xi_0^3 - 5/24\xi_1^2 + 5/36\xi_1^3 - 1/6\xi_2 \\ & + 1/8\xi_2^2 - 1/36\xi_2^3)u_{i+2}^n\Big),\end{aligned}$$

where $\xi_j = (x_j - x_j^*)/\Delta x$, $j = i-2, ..., i+2$.

2. compute the linear weights if a is a constant:

$$\begin{aligned}\gamma_1 &= 1/10 + 3/20\xi + 1/20\xi^3, \\ \gamma_2 &= 3/5 + 1/10\xi - 1/10\xi^3, \\ \gamma_3 &= 3/10 + 1/4\xi + 1/20\xi^3,\end{aligned}$$

in this case, all $\{\xi_j\}$ $(j = -2, ..., 2)$ are the same, which we write as ξ.

However when $a(u)(x,t)$ is not a constant, no optimal linear weights to achieve 5th order exist. We take

$$\gamma_1 = 1/6,\ \gamma_2 = 2/3,\ \gamma_3 = 1/6,$$

which are used to ensure the non-oscillatory performance [24].

3. compute the smoothness indicator:

$$\begin{aligned}\beta_1 &= 13/12(u_{i-2}^n - 2u_{i-1}^n + u_i^n)^2 + 1/4(u_{i-2}^n - 4u_{i-1}^n + 3u_i^n)^2, \\ \beta_2 &= 13/12(u_{i-1}^n - 2u_i^n + u_{i+1}^n)^2 + 1/4(u_{i-1}^n - u_{i+1}^n)^2, \\ \beta_3 &= 13/12(u_i^n - 2u_{i+1}^n + u_{i+2}^n)^2 + 1/4(3u_i^n - 4u_{i+1}^n + u_{i+2}^n)^2,\end{aligned}$$

4. compute the nonlinear weights.

$$\widetilde{w}_j = \frac{\gamma_j}{(\varepsilon + \beta_j)^2}, \quad w_j = \frac{\widetilde{w}_j}{\sum_{i=1}^3 \widetilde{w}_i},\ j = 1, 2, 3.$$

5. get the numerical flux.

$$\widehat{h}_{i+\frac{1}{2}} = h_{i+\frac{1}{2}}^- = w_1 h^{(1)}(x_{i+\frac{1}{2}}) + w_2 h^{(2)}(x_{i+\frac{1}{2}}) + w_3 h^{(3)}(x_{i+\frac{1}{2}}).$$

When $a\Delta t > \Delta x$, let i^* be the index such that $x_i^* \in [x_{i^*-1}, x_{i^*}]$, we can write the flux $\widehat{h}_{i+\frac{1}{2}}$ as

$$\begin{aligned}\widehat{h}_{i+\frac{1}{2}} &= \sum_{j=i^*+1}^{i} \triangle x u_j^n + R(u_{i^*-p}, ..., u_{i^*-q}) \\ &= \sum_{j=i^*+1}^{i} \Delta x u_j^n + \widehat{h}_{i^*+\frac{1}{2}},\end{aligned}$$

where $\widehat{h}_{i^*+\frac{1}{2}}$ will be reconstructed as the case $a\Delta t < \Delta x$ with $\{u_{i^*-2}, ..., u_{i^*+2}\}$.

REFERENCES

[1] S. Boscarino, S.-Y. Cho, G. Russo and S.-B. Yun, High order conservative semi-Lagrangian scheme for the BGK model of the Boltzmann equation, submitted (2019), arXiv:1905.03660.

[2] X. Cai, S. Boscarino and J.-M. Qiu, High order semi-Lagrangian discontinuous galerkin method coupled with Runge-Kutta exponential integrators for nonlinear Vlasov dynamics, submitted (2019), arXiv:1911.12229.

[3] X. Cai, W. Guo and J.-M. Qiu, A high order conservative semi-Lagrangian discontinuous Galerkin method for two-dimensional transport simulations, *Journal of Scientific Computing*, **73** (2017), 514–542.

[4] X. Cai, J.-X. Qiu and J.-M. Qiu, A conservative semi-Lagrangian HWENO method for the Vlasov equation, *Journal of Computational Physics*, **323** (2016), 95–114.

[5] E. Celledoni and B. K. Kometa, Semi-Lagrangian Runge-Kutta exponential integrators for convection dominated problems, *Journal of Scientific Computing*, **41** (2009), 139–164.

[6] E. Celledoni, B. K. Kometa and O. Verdier, High order semi-Lagrangian methods for the incompressible Navier–Stokes equations, *Journal of Scientific Computing*, **66** (2016), 91–115.

[7] E. Celledoni, A. Marthinsen and B. Owren, Commutator-free Lie group methods, *Future Generation Computer Systems*, **19** (2003), 341–352.

[8] G.-H. Cottet, J.-M. Etancelin, F. Perignon and C. Picard, High order semi-Lagrangian particles for transport equations: Numerical analysis and implementation issues, *ESIAM: Mathematical Modelling and Numerical Analysis*, **48** (2014), 1029–1060.

[9] N. Crouseilles, M. Mehrenberger and E. Sonnendrücker, Conservative semi-Lagrangian schemes for Vlasov equations, *Journal of Computational Physics*, **229** (2010), 1927–1953.

[10] N. Crouseilles, T. Respaud and E. Sonnendrücke, A forward semi-Lagrangian method for the numerical solution of the Vlasov equation, *Computer Physics Communications*, **180** (2009), 1730–1745.

[11] K. Duraisamy and J. D. Baeder, Implicit scheme for hyperbolic conservation laws using nonoscillatory reconstruction in space and time, *SIAM Journal on Scientific Computing*, **29** (2007), 2607–2620.

[12] A. Efremov, E. Karepova and V. Shaydurov, A conservative semi-Lagrangian method for the advection problem, *Numerical Analysis and its Applications, Lecture Notes in Comput. Sci., Springer, Cham*, *10187* (2017), 325–333.

[13] L. Fatone, D. Funaro and G. Manzini, A semi-Lagrangian spectral method for the Vlasov-Poisson system based on Fourier, Legendre and Hermite polynomials, *Communications on Applied Mathematics and Computation*, **1** (2019), 333–360.

[14] F. Filbet and C. Prouveur, High order time discretization for backward semi-Lagrangian methods, *Journal of Computational and Applied Mathematics*, **303** (2016), 171–188.

[15] W. Guo, R. D. Nair and J.-M. Qiu, A conservative semi-Lagrangian discontinuous Galerkin scheme on the cubed-sphere, *Monthly Weather Review*, **142** (2014), 457–475.

[16] C.-S. Huang, T. Arbogast and C.-H. Hung, A semi-Lagrangian finite difference WENO scheme for scalar nonlinear conservation laws, *Journal of Computational Physics*, **322** (2016), 559–585.

[17] R. J. LeVeque, *Numerical Methods for Conservation Laws* Second edition, Lectures in Mathematics ETH Zürich. Birkhäuser Verlag, Basel, 1992.

[18] J.-M. Qiu, High order mass conservative semi-Lagrangian methods for transport problems, *Handbook of Numerical Methods for Hyperbolic Problems*, **17** (2016), 353–382.

[19] J.-M. Qiu and A. Christlieb, A conservative high order semi-Lagrangian WENO method for the Vlasov equation, *Journal of Computational Physics*, **229** (2010), 1130–1149.

[20] J.-M. Qiu and G. Russo, A high order multi-dimensional characteristic tracing strategy for the Vlasov-Poisson System, *Journal of Scientific Computing*, **71** (2017), 414–434.

[21] J.-M. Qiu and C.-W. Shu, Conservative high order semi-Lagrangian finite difference WENO methods for advection in incompressible flow, *Journal of Computational Physics*, **230** (2011), 863–889.

[22] J.-M. Qiu and C.-W. Shu, Convergence of Godunov-type schemes for scalar conservation laws under large time steps, *SIAM Journal on Numerical Analysis*, **46** (2008), 2211–2237.

[23] J. A. Rossmanith and D. C. Seal, A positivity-preserving high-order semi-Lagrangian discontinuous Galerkin scheme for the Vlasov-Poisson equations, *Journal of Computational Physics*, **230** (2011), 6203–6232.

[24] C.-W. Shu, High order weighted essentially nonoscillatory schemes for convection dominated problems, *SIAM Review*, **51** (2009), 82–126.
[25] D. Sirajuddin and W. N. G. Hitchon, A truly forward semi-Lagrangian WENO scheme for the Vlasov-Poisson system, *Journal of Computational Physics*, **392** (2019), 619–665.
[26] E. Sonnendruüker, J. Roche, P. Bertrand and A. Ghizzo, The semi-Lagrangian method for the numerical resolution of the Vlasov equation, *Journal of Computational Physics*, **149** (1999), 201–220.
[27] A. Staniforth and J. Côté, Semi-Lagrangian integration schemes for atmospheric models: A review, *Monthly Weather Review*, **119** (1991), 2206–2223.
[28] G. Tumolo, L. Bonaventura and M. Restelli, A semi-implicit, semi-Lagrangian, p-adaptive discontinuous Galerkin method for the shallow water equations, *Journal of Computational Physics*, **232** (2013), 46–67.
[29] T. Xiong, J.-M. Qiu, Z. Xu and A. Christlieb, High order maximum principle preserving semi-Lagrangian finite difference WENO schemes for the Vlasov equation, *Journal of Computational Physics*, **273** (2014), 618–639.
[30] T. Xiong, G. Russo and J.-M. Qiu, High order multi-dimensional characteristics tracing for the incompressible Euler equation and the guiding-center Vlasov equation, *Journal of Scientific Computing*, **77** (2018), 263–282.
[31] T. Xiong, G. Russo and J.-M. Qiu, Conservative multi-dimensional semi-Lagrangian finite difference scheme: Stability and applications to the kinetic and fluid simulations, *Journal of Scientific Computing*, **79** (2019), 1241–1270.
[32] D. Xiu and G. E. Karniadakis, A semi-Lagrangian high-order method for Navier-Stokes equations, *Journal of Computational Physics*, **172** (2001), 658–684.
[33] T. Yabe and Y. Ogata, Conservative semi-Lagrangian CIP technique for the shallow water equations, *Computational Mechanics*, **46** (2010), 125–134.

Received March 2020; revised July 2020.

E-mail address: zglmath@stu.xmu.edu.cn
E-mail address: sqzheng@stu.xmu.edu.cn
E-mail address: txiong@xmu.edu.cn