Abstract. Numerical weighted essentially non-oscillatory (WENO) schemes have historically been formulated as a method of lines (MOL) or Taylor (Lax-Wendroff) method. In the MOL viewpoint, the partial differential equation (PDE) is treated as a large system of ordinary differential equations, to which an appropriate time-integrator is applied. In contrast, Lax-Wendroff discretizations immediately convert Taylor series in time to discrete spatial derivatives. We propose a Picard integral formulation that introduces new possibilities for combining space and time derivatives. In particular, we present a new class of methods by applying the WENO reconstruction procedure to the so-called “time-averaged” fluxes in place of the semi-discrete flux that would arise from a typical MOL formulation. Given that the classical WENO reconstruction coupled with forward Euler time stepping conserves mass, the Picard integral formulation is automatically conservative under any temporal discretization, and therefore lends itself to a multitude of options for further investigation, including Taylor, Runge-Kutta and linear-multistep discretizations. At present, our focus is on the discretization of the time-averaged fluxes with Taylor methods. We describe this new vantage by combining it with classical finite difference WENO schemes and apply it to hyperbolic conservation laws in one- and two-dimensions. Its effectiveness is demonstrated in a series of test cases, and the results are in agreement with current state of the art methods.

1. Introduction. Numerical methods often treat spatial and temporal discretizations as separate operations. In a method of lines (MOL) formulation, one first discretizes in space, and then applies an appropriate time integrator to the problem. Popular high-order methods for hyperbolic conservation laws include finite difference and finite volume weighted essentially non-oscillatory (WENO) methods, discontinuous- and Petrov-Galerkin as well as piecewise parabolic methods (PPMs). This defines a large system of ordinary differential equations (ODEs), that is then integrated in time by a suitable ODE solver, which is usually an explicit Runge-Kutta method. This type of formulation misses out on the potential to couple the space- and time-discretizations.

The large exception to this space-time separation include high-order Taylor (Lax-Wendroff) temporal discretizations. The basic procedure is to first discretize in time, and then immediately convert the temporal derivatives to discrete spatial derivatives via the Cauchy-Kowalewski procedure. The original Lax-Wendroff method [1] is likely the first numerical method based on this procedure, where second-order accuracy was obtained by including the evaluation of flux Jacobians. The coupling of arbitrarily high order Lax-Wendroff discretizations to finite volume methods has been thoroughly investigated since the early 2000s, and it consists of the so-called ‘Arbitrary DERivative (ADER) methods [2–9]. Recent work also includes coupling the Lax-Wendroff discretization to high-order finite difference WENO [refs] and discontinuous Galerkin [refs] methods. Other recent work includes coupling the Lax-Wendroff procedure to
high-order finite difference WENO \cite{10-13} and discontinuous Galerkin \cite{14-18} methods. All high-order Lax-Wendroff discretizations rely on the evaluation of Jacobians, Hessians and higher derivatives that inevitably show up for all high-order single-step methods. The finite difference WENO methods that fall into this category typically compute the first derivative with a single non-linear WENO reconstruction, and then use (centered) finite differences to compute higher derivatives. The finite volume flavored methods including the discontinuous Galerkin and ADER methods \cite{18, 19} rely on high-order solutions to generalized Riemann problems \cite{7}. The common thread in these methods is that the time dependence is immediately removed in the process of converting temporal to spatial derivatives. When compared with Runge-Kutta methods, there are no degrees of freedom to work with because each coefficient in a Taylor series expansion is unique.

The purpose of this work is to define the Picard integral formulation of a hyperbolic conservation law for the purposes of investigating the interplay between temporal and spatial discretizations. We construct conservative finite difference WENO schemes from this new vantage.

2. The 1D Picard integral formulation. We begin our discussion with a 1D system of conservation laws defined by

\[ q_t + f(q)_x = 0, \quad (2.1) \]

where \( q(t, x) : \mathbb{R}^+ \times \mathbb{R} \rightarrow \mathbb{R}^m \) is the unknown vector of \( m \) conserved quantities and \( f(q) : \mathbb{R}^m \rightarrow \mathbb{R}^m \) is a prescribed flux function.

Formal integration of (2.1) over a single interval \( t \in [t^n, t^{n+1}] \) defines the Picard integral formulation of the 1D conservation law

\[ q(t^{n+1}, x) = q(t^n, x) - \Delta t F^n(x)_x, \quad (2.2a) \]

where the time-averaged flux is defined as

\[ F^n(x) := \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} f(q(t, x)) \, dt. \quad (2.2b) \]

We stop to point out that many finite volume formulations including the original Godunov scheme, high-order variations of the scheme \cite{20-24}, and modern ADER methods \cite{5, 6, 8, 9} also work with time-averaged fluxes. In these formulations, one takes the further step of integrating (2.2a) over a control element and then applying the divergence theorem. This defines an evolution equation for the unknown cell averages, which is based on integrating (2.2b) over the boundaries of each control element. The differences in the methods primarily come from the choice in discretization of the time-averaged fluxes, as well as how each Riemann problem is solved. For example, in 1959 Godunov discretized the time-average fluxes with constants; his insight was to leverage exact Riemann solvers for piecewise constant functions. In 1979 Van Leer \cite{20} introduced piecewise linear approximations to the solution, and therefore was able to define a second-order version of Godunov’s method. A decade later, Harten, Engquist, Osher and Chakravarthy \cite{22} defined an essentially non-oscillatory method that used piecewise polynomial reconstructions of the solution from cell averages, and

\footnote{The Picard integral formulation would normally use \( \int_{t^n}^{t^{n+1}} f(q(t, x)) \, dt \). We replace this with the averaged value in order to carry the same units as the physical flux.}
therefore were able to extend Godunov’s method to arbitrary order. Shortly thereafter, the now classical ADER formulation was defined \cite{2}, where additional Arbitrary DERivative Riemann problems were introduced in order to define an arbitrary order, single-step finite volume method.

Our focus is not on finite volume methods, nor their discontinuous Galerkin cousins, but on conservative finite difference methods. In particular, we offer \cite{22} as the starting point for defining new conservative finite difference methods, where we propose the application of the following two step process:

1. Approximate the time-averaged fluxes in \eqref{2.2a} with any temporal discretization;
2. Insert the result into the finite difference WENO reconstruction procedure.

This step provides a conservative high-order approximation to the spatial derivative in \eqref{2.2b}.

The combination of these two steps will be called the Picard integral formulation of WENO (PIF-WENO).

A variety of conservative finite difference schemes can be constructed from the vantage the Picard integral formulation presents. To the authors’ knowledge, the first conservative finite difference scheme of this type is the 1D semi-Lagrangian solver created by J-M Qiu and Shu in 2011 \cite{25}. There, the authors construct 2D incompressible flow solutions by combining 1D advection equation solvers for \( q_t + (a(t,x)q)_x \) with operator splitting techniques. They make use of a time-averaged version of the usual discrete implicit flux function and rely on semi-Lagrangian techniques to convert the temporal integration to spatial integration. Their application of the PIF-WENO formulation culminated with scalar advection problems.

Our contribution is the application of time-averaged fluxes to multidimensional systems of hyperbolic conservation laws. We make use of time-averaged discrete implicit flux functions in order to retain mass conservation, which will be described in the next two sections. We refer the reader to the references for details on the essentially non-oscillatory method \cite{22,26,27}, as well as the weighted essentially non-oscillatory method \cite{28–31}. The conservative finite difference reconstruction procedure can be carried out independent of the temporal discretization of the time-averaged fluxes, and therefore we delay a description of how this can be accomplished until \S 2.3 where we describe an example Taylor discretization.

### 2.1. 1D conservative finite differences with time-averaged fluxes.

Consider a uniform grid of \( m_x \) equidistant points in \( \Omega = [a,b] \):

\[
    x_i = a + \left( i - \frac{1}{2} \right) \Delta x, \quad \Delta x = \frac{b-a}{m_x}, \quad i \in \{1,\ldots,m_x\}.
\]  

In a finite difference method, one enforces some approximation to \eqref{2.1} to hold on a finite set of points \( x_i \), and seeks point value approximations \( q^n_i \approx q(t^n, x_i) \) to hold at discrete time levels \( t^n \).

The classical conservative finite difference WENO method implicitly defines a function \( h(q(t,x)) \), whose sliding average agrees with \( f(q(t,x)) \) via

\[
    \tilde{h}(q(t,x)) := \frac{1}{\Delta x} \int_{x-\Delta x/2}^{x+\Delta x/2} h(q(t,s)) \, ds = f(q(t,x)).
\]

We consider a time-averaged version of this implicit function, defined by

\[
    H^n(x) := \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} h(q(t,x)) \, dt.
\]
This allows us to rewrite \((2.4)\) as
\[
\tilde{H}^n(x) := \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \tilde{h}(t,x) \, dt = \frac{1}{\Delta x} \int_{x_{n+1/2}}^{x_{n-1/2}} \tilde{H}^n(s) \, ds = F^n(x). \tag{2.6}
\]

We remark that \((2.6)\) is similar to the usual averaging property found in finite difference methods, but the primary departure is that we have chosen to integrate both the flux and the implicitly defined function in time. These time-averaged functions are different than what is typically done in a method of lines (MOL) formulated finite difference WENO method, where the reconstruction procedure is typically carried out at discrete “frozen in time” levels. We now point out two important properties concerning the implicitly defined function. These share commonality with the classical MOL formulation.

- High-order reconstruction algorithms can produce high-order interface values \(H^n(x_{i-1/2})\) by identifying cell averages as known point values: \(\tilde{H}^n(x_i) = F^n_i\), where \(F^n_i := F^n(x_i)\), and therefore \(H^n\) need never be computed.
- High-order derivatives of \(F^n\) can be computed from interpolated values of \(H^n\) by a simple application of the Fundamental Theorem of Calculus \([30, 31]\)
\[
\frac{dF^n}{dx}(x_i) = \frac{1}{\Delta x} \left[ H^n(x_{i+1/2}) - H^n(x_{i-1/2}) \right]. \tag{2.7}
\]

Once \((2.7)\) has been defined, the complete finite difference PIF-WENO method is given by inserting the result into \((2.2)\), which results in a single-step update
\[
q_{i}^{n+1} = q_i^n - \frac{\Delta t}{\Delta x} \left( \hat{F}_i^{n+1/2} - \hat{F}_i^{n-1/2} \right), \tag{2.8}
\]
where \(\hat{F}_i^{n-1/2} := H^n(x_{i-1/2})\). Before providing the description of how the \(H^n_{i-1/2}\) are constructed, we point out a single remark followed by an important theorem.

**Remark 1.** In the case where the time-averaged fluxes are approximated as constants through \(F^n(x) \approx f(q(t^n, x))\), the PIF-WENO formulation reduces to a single Euler step of a classical WENO method.

**Theorem 1.** No matter how the quantities \(F^n_i\) are approximated, on an infinite or periodic domain, the update defined by \((2.8)\) satisfies the conservation property
\[
\sum_i q_i^{n+1} = \sum_i q_i^n. \tag{2.9}
\]

**Proof.** Sum equation \((2.8)\) over all \(i\). 

This theorem is a consequence of the deliberate choice of the order of operations used to carry out \((2.2)\). Its significance means that the time-averaged fluxes can be discretized in any manner without violating mass conservation. Under the assumption that the \(F^n_i\) have been approximated at each grid point, we are now ready to describe the finite difference WENO procedure for systems. A Taylor discretization will be presented in \(\S 2.3\).

**2.2. The finite difference WENO method with time-averaged fluxes.**

1. Compute average values of \(q\) at the half grid points
\[
q_{i-1/2}^* = \frac{1}{2} (q_i + q_{i-1}). \tag{2.10}
\]

Roe averages \([32]\) that satisfy \(f(q_i) - f(q_{i-1}) = f'(q_{i-1}^*) (q_i - q_{i-1})\) may be used in place of \((2.10)\). We use the simple arithmetic average.
2. Compute the left and right eigenvalue decomposition of $f'(q) = RAR^{-1}$ at
the half-grid points

$$R_{i-1/2} = R(q_i^{*}-1/2), \quad R_{i-1/2}^{-1} = R^{-1}(q_i^{*}-1/2).$$

Compute the fastest local wave speed

$$\alpha_{i-1/2}^* := \max \left\{ \left| \min \left\{ \Lambda(q_{i-1}), \Lambda(q_{i-1}^{-1/2}) \right\} \right|, \left| \min \left\{ \Lambda(q_{i-1}^{*}), \Lambda(q_{i}) \right\} \right| \right\}.$$ 

Our definition of $\alpha_{i-1/2}^*$ is nominally different than other definitions that
have been used for local wave speeds in the finite difference WENO method
[33, 34]. For stability, we follow the common practice of increasing this speed
by exactly 10% and define $\alpha_{i-1/2} := 1.1 \cdot \alpha_{i-1/2}^*$.

3. For each $i$, determine the weighted ENO stencil $\{i + r\}$ surrounding $x_{i-1/2}$.
In fifth-order WENO, the full stencil is given by $r \in \{-3, -2, -1, 0, 1, 2\}$. Project each $q_{i+r}$ and each time-averaged flux value $F_{i+r}$ onto the characteristic variables using the linear mapping $R_{i-1/2}^{-1}$

$$w_{i+r} = R_{i-1/2}^{-1} \cdot q_{i+r}, \quad (2.11a)$$

$$z_{i+r} = R_{i-1/2}^{-1} \cdot F_{i+r}. \quad (2.11b)$$

Apply a (local) Lax-Friedrichs flux splitting on $q_{i+r}$

$$z_{i+r}^\pm = \frac{1}{2} (z_{i+r} \pm z_{i-1/2} w_{i+r}) \cdot (2.12)$$

As an alternative, one could use a global wave speed.

4. Perform a WENO reconstruction on each of the characteristic variables separately. Use the stencil that uses an extra point on the upwind direction for defining $z^+$:

$$\hat{z}_{i-1/2}^+ = WENO5^+ \left[ z_{i-3}^+, z_{i-2}^+, z_{i-1}^+, z_i^+, z_{i+1}^+ \right],$$

$$\hat{z}_{i-1/2}^- = WENO5^- \left[ z_{i-2}^-, z_{i-1}^-, z_i^-, z_{i+1}^-, z_{i+2}^- \right].$$

See Appendix A for a description of this function. Define $\hat{z}_{i-1/2} := \hat{z}_{i-1/2}^+ + \hat{z}_{i-1/2}^-$.

5. Using the same projection matrix, $R_{i-1/2}$, project characteristic variables
back onto the conserved variables

$$\hat{R}_{i-1/2} := R_{i-1/2} \cdot \hat{z}_{i-1/2}.$$

To summarize, given any modified flux function $F^n$, we compute $\hat{R}_{i-1/2}$ with
WENO reconstructions applied to flux-split characteristic decompositions. The result
is inserted into (2.8) to update the solution. This procedure can be carried out
independent of the choice of how the temporal integrals are discretized. As pointed
out in Remark 1, this formulation reduces to classical WENO with forward Euler
time-discretization if the time-averaged fluxes are approximated with constants. We
now point out one method of increasing the temporal accuracy: Taylor series.
2.3. A Taylor series discretization of the 1D Picard integral formulation. Many options of defining a semi-discrete modified flux in Eqn. (2.2b) exist, including Runge-Kutta, Taylor, linear multi-step and multiderivative methods [35, 36]. To illustrate the basic idea of the Picard integral formulation, we present Taylor methods coupled with the finite difference WENO method, and we will report on other discretizations in a follow-up work.

Consider a Taylor expansion of \( f \) centered at \( t = t^n \), and define an approximation to the time-averaged flux as
\[
F^n_T(x) := f(q(t^n, x)) + \frac{\Delta t}{2!} \frac{df}{dt}(q(t^n, x)) + \frac{\Delta t^2}{3!} \frac{d^2f}{dt^2}(q(t^n, x)) = F^n + \mathcal{O}(\Delta t^3). \tag{2.13}
\]
These temporal derivatives can be computed via the Cauchy-Kowalewski procedure. For example, in 1D the first two derivatives are
\[
\frac{df}{dt} = \frac{\partial f}{\partial q} \cdot q_t = -\frac{\partial f}{\partial q} \cdot f_x \tag{2.14}
\]
\[
\frac{d^2f}{dt^2} = \frac{\partial^2 f}{\partial q^2} \cdot (f_x, f_x) + \frac{\partial f}{\partial q} \left( \frac{\partial f}{\partial q} \cdot f_x \right)_x \tag{2.15}
\]
where \( \frac{\partial f}{\partial q} \in \mathbb{R}^{m \times m} \) is the Jacobian matrix of \( f \), and \( \frac{\partial^2 f}{\partial q^2} \in \mathbb{R}^{m \times m \times m} \) is the Hessian tensor\(^2\). Further derivatives produce tensors that grow dramatically in size. We compute point values \( F^n_i := F^n_T(x_i) \) using the following finite difference formulas on a single 5-point stencil:
\[
u_{i,x} := \frac{1}{12\Delta x} (u_{i-2} - 8u_{i-1} + 8u_{i+1} - u_{i+2}) = u_{i,x}(x_i) + \mathcal{O}(\Delta x^4) \tag{2.16a}
\]
\[
u_{i,xx} := \frac{1}{12\Delta x^2} (-u_{i-2} + 16u_{i-1} - 30u_i + 16u_{i+1} - u_{i+2}) = u_{i,xx}(x_i) + \mathcal{O}(\Delta x^4). \tag{2.16b}
\]
Higher derivatives can be included using the same stencil. They will start to lose single orders of accuracy, but because those derivatives are multiplied by increasing powers of \( \Delta t = \mathcal{O}(\Delta x) \) the overall order of accuracy is retained. This is consistent with previous results in the literature [10, 13, 14, 36]. Additionally, observe that this defines a single value for each \( F^n_i \), and therefore any stencil can be used without violating mass conservation.

---

\(^2\)Given two vectors \( u, v \in \mathbb{R}^m \), the Hessian product is defined as the vector whose \( i^{th}\)-component is
\[
\left[ \frac{\partial^2 f}{\partial q^2} \cdot (u, v) \right]_i := \sum_{j=1}^{m} \sum_{k=1}^{m} \frac{\partial^2 f_i}{\partial q_j \partial q_k} u_j v_k.
\]
One important departure from previous finite difference Lax-Wendroff WENO methods \[10,13\] is that this particular method applies the non-linear WENO reconstruction after the linear finite differences have been computed, and therefore serves as a non-linear filter to the higher order terms. In previous works, all the authors rely on \(O(\Delta t, \Delta x)\) factors on linear finite differences to suppress spurious oscillations. We believe there is an advantage to be gained by applying the non-linear reconstruction after these terms have been computed in order to correctly select the smoothest sub-stencil.

This completes the description of the 1D finite difference WENO method based on Taylor approximations to the time-averaged fluxes. We now describe the extension of this method to the two-dimensional case.

### 3. The 2D Picard integral formulation

A 2D conservation law is given by

\[
q_{,t} + (f(q))_{,x} + (g(q))_{,y} = 0.
\]

(3.1)

Similar to the 1D case, we define the 2D Picard integral formulation as

\[
q^{n+1} = q^n - \Delta t (F^n(x,y))_{,x} - \Delta t (G^n(x,y))_{,y},
\]

(3.2a)

where the time-averaged fluxes are defined by

\[
F^n(x,y) := \frac{1}{\Delta t} \int_t^{t+1} f(q(t,x,y)) \, dt, \quad G^n(x,y) := \frac{1}{\Delta t} \int_t^{t+1} g(q(t,x,y)) \, dt.
\]

(3.2b)

We again propose an application of the finite difference WENO method to the time-averaged fluxes in place of the usual “frozen in time” approximation to the physical fluxes. We repeat that many options for discretizing (3.2b) exist. For now, we proceed with describing how the 2D finite difference WENO method can be applied under the assumption that one such approximation has been made.

#### 3.1. 2D conservative finite differences with time-averaged fluxes

We start with a discretization of the domain \(\Omega = [a_x, b_x] \times [a_y, b_y] \subset \mathbb{R}^2\) given by

\[
x_i = a_x + \left( i - \frac{1}{2} \right) \Delta x, \quad \Delta x = \frac{b_x - a_x}{m_x}, \quad i \in \{1, \ldots, m_x\},
\]

(3.3a)

\[
y_j = a_y + \left( j - \frac{1}{2} \right) \Delta y, \quad \Delta y = \frac{b_y - a_y}{m_y}, \quad j \in \{1, \ldots, m_y\}.
\]

(3.3b)

The finite difference method seeks pointwise approximations \(q_{ij}^n \approx q(t^n, x_i, y_j)\) to hold at each of these grid points.

In the hyperbolic 2D case, one has two eigen-decompositions for \(f'(q)\) and \(g'(q)\), to which the procedure described in \[24\] is applied on a dimension by dimension fashion. We propose using \(F_{ij}^n\) and \(G_{ij}^n\) in place of the typical \(f(t^n, x_i, y_j)\) and \(g(t^n, x_i, y_j)\) in order to define a single-step update of the form

\[
q_{ij}^{n+1} = q_{ij}^n - \frac{\Delta t}{\Delta x} \left( \hat{F}_{i+1/2,j}^n - \hat{F}_{i-1/2,j}^n \right) - \frac{\Delta t}{\Delta y} \left( \hat{G}_{i,j+1/2}^n - \hat{G}_{i,j-1/2}^n \right).
\]

(3.4)

Similar to Remark 1, we point out the following special case.

**Remark 2.** If \(F_{ij}^n = f(q(t^n, x_i, y_j))\) and \(G_{ij}^n = f(q(t^n, x_i, y_j))\), then (3.4) is identical to a single Euler step of a classical RK-WENO method. It is the inclusion
of higher-order terms in the approximation of the time-averaged fluxes that increases the temporal order of accuracy.

Similar to Theorem 1, this method will automatically be mass conservative given any approximation to the time-averaged fluxes.

We repeat that many options exist for constructing the time-averaged fluxes. At present, we offer Taylor methods; additional methods are currently under investigation, and will be presented in the future.

3.2. A Taylor series discretization of the 2D Picard integral formulation. We define Taylor discretization of the time-averaged fluxes with

$$F_{T}^{n}(x,y) := f(q(t^{n},x,y)) + \frac{\Delta t}{2!} \frac{df}{dt} (q(t^{n},x,y)) + \frac{\Delta t^{2}}{3!} \frac{d^{2}f}{dt^{2}} (q(t^{n},x,y)) = F^{n} + O(\Delta t^{3});$$

$$G_{T}^{n}(x,y) := g(q(t^{n},x,y)) + \frac{\Delta t}{2!} \frac{dg}{dt} (q(t^{n},x,y)) + \frac{\Delta t^{2}}{3!} \frac{d^{2}g}{dt^{2}} (q(t^{n},x,y)) = G^{n} + O(\Delta t^{3}).$$

Similar to the 1D case, analytical temporal derivatives can be computed via the Cauchy-Kowalewski procedure. Derivatives of the first component in the flux function $f$ are given by

$$\frac{df}{dt} = - \frac{\partial f}{\partial q} \cdot (f_{x} + g_{y}),\quad (3.6)$$

$$\frac{d^{2}f}{dt^{2}} = \frac{\partial^{2}f}{\partial q^{2}} \cdot (f_{x} + g_{y}, f_{x} + g_{y}) + \frac{\partial f}{\partial q} \cdot (f_{x} + g_{y}),\quad (3.7)$$

where an expansion of the final time derivative is given by

$$(f_{x} + g_{y}),\quad (3.8)$$

Note that third-order accuracy requires an approximation to the mixed derivatives $f_{xy}$ and $g_{xy}$. Similarly, the temporal derivatives of $g$ can be expressed with

$$\frac{dg}{dt} = - \frac{\partial g}{\partial q} \cdot (f_{x} + g_{y}),\quad (3.9)$$

$$\frac{d^{2}g}{dt^{2}} = \frac{\partial^{2}g}{\partial q^{2}} \cdot (f_{x} + g_{y}, f_{x} + g_{y}) + \frac{\partial g}{\partial q} \cdot (f_{x} + g_{y}),\quad (3.10)$$

We compute the first- and second-derivatives of $f, g$ and $q$ on a dimension by dimension fashion, with the coefficients prescribed by equations (2.16a)-(2.16b). The singular departure from the 1D case is that we require an approximation for the cross-derivative terms. These could be computed through a composition of finite differences or inclusion of off diagonal terms in the stencil. In order to retain a compact stencil, we choose the latter approach, to which we apply the finite difference formula

$$u_{,xy}(x_{i},y_{j}) \approx u_{ij,xy} := \frac{1}{4\Delta x\Delta y} (u_{i+1,j+1} - u_{i-1,j+1} - u_{i+1,j-1} + u_{i-1,j-1}). \quad (3.11)$$
Given that they are eventually multiplied by a factor of $\Delta t^3 = O(\Delta x^3)$, we only compute these to second-order. In the past, these cross-derivatives have been computed by applying a single finite difference stencil multiple times [10], which results in a large effective stencil.

The complete scheme is given by inserting

$$F^n_{ij} := F^n_T(x_i, y_j) \quad \text{and} \quad G^n_{ij} := G^n_T(x_i, y_j)$$ (3.12)

into the conservative finite difference reconstruction procedure.

This current proposal provides an additional departure from previous finite difference Lax-Wendroff works [10, 13], because we use a single inexpensive stencil to compute the time-averaged fluxes everywhere in the domain, and then follow that with a single WENO reconstruction. The overall effective stencil is presented in Figure 3.1 and is much smaller than Qiu and Shu’s original method [10], and similar in size but different in shape to Jiang Shu and Zhang’s recent proposal [13]. The shaded cells indicate the size of the stencil used in the first pass, and the red ‘X’ terms indicate the stencil required for the WENO reconstruction. An implementation that uses a single loop over the domain would require access to every circled element to update the solution at one point.

3.3. A brief note on implementation. One simple implementation of the proposed method that requires minimal modification to existing codes uses two loops over the computational domain:

1. Compute the time-averaged fluxes $F^n_{ij}$ and $G^n_{ij}$ with finite differences on the conserved variables. The benefit is that no characteristic projections or nonlinear reconstructions are necessary. However, Taylor methods in particular require additional algebraic operations and therefore the computational expense comes from the need to evaluate high-derivatives using Jacobians and Hessians that increase in size.

2. Use the time-averaged fluxes $F^n_{ij}$ and $G^n_{ij}$ to perform the usual WENO reconstruction with an existing sub-routine. The cost of this step is equivalent to that of a single WENO reconstruction.

We remark that the storage required for this particular implementation is competitive with low-storage Runge-Kutta methods [37–40]. In particular, any three-stage Runge-Kutta method applied to the classical MOL formulation will require a total of three WENO reconstructions, whereas we make a first pass with an inexpensive linear finite difference stencil, and then follow that up with a single WENO reconstruction. In addition, because it is a single-step Taylor method, this method is amenable to adding adaptive mesh refinement.

4. Numerical Results.

4.1. 1D Burger’s equation. We begin our numerical results section with a convergence study on smooth solutions to inviscid Burger’s equation:

$$q_t + \left( \frac{1}{2} q^2 \right)_x = 0.$$ (4.1)

We take the computational domain to be $[0, 2]$ and apply periodic boundary conditions. The initial conditions are prescribed by $q(t = 0, x) = 0.5 + \sin(\pi x)$. For our convergence study, we choose a final time of $t = 0.5\pi^{-1}$ before any shocks develop.
Effective stencil. Shown here are the collection of all elements required to define a single point: $q_{ij}^{n+1}$. Each red ‘X’ comes from the seven point fifth-order WENO stencil used in each direction, where a 13-point stencil is used to define each required $F_{i+3,j}^n$ and $G_{i,j+3}^n$. For example, the elements required to define $F_{i+3,j}^n$ have been shaded. Note that the cross-derivative terms have been explicitly included in the stencil as opposed to relying on many compositions of the WENO stencil to produce these.

Exact solutions to this problem can be found implicitly by applying the method of characteristics [41]. With initial conditions defined by $q_0(s) = q(0, s)$, the exact solution is

$$q(t, x) = q_0(x - t \cdot q_0(\xi)),$$

where $\xi$ is an implicit solution to the equation $\xi = x - t \cdot q_0(\xi)$. For the WENO schemes, we solve (4.2) at each grid point using Newton iteration with a tolerance of $10^{-15}$.

In 1D, we use relative errors with the $L^1$ norm defined by point-wise values

$$\text{Error} := \frac{\Delta x \sum_{i=1}^{m_x} |q^n_i - q(t^n, x_i)|}{\Delta x \sum_{i=1}^{m_x} |q(t^n, x_i)|},$$

and we define the CFL number as the dimensionless quantity

$$\nu := \frac{\Delta t}{\Delta x} \max_i \left\{ \alpha^{*}_{i-1/2} \right\}.$$ (4.4)

In Jiang, Shu and Zhang [13], the authors compare results of their finite difference WENO method in a side-by-side table against the older Qiu and Shu method [10]. However, the new simulations use a smaller CFL number of $\nu = 0.3$ as opposed to the value of $\nu = 0.5$ reported in the earlier paper. For completeness, in Table 4.1 we present results of our method for both CFL numbers. We make two brief observations:

- Large CFL numbers bring out the temporal error faster, and therefore higher-order in time is likely more important to reduce error if a user wishes to take large time steps.
- For small meshes, the difference in errors between the two CFL numbers is quite small, indicating that in coarse grids, high-order spatial resolution is more important than a high temporal order of accuracy.

Similar to past investigations, for small meshes the spatial error dominates, and therefore the algebraic order is higher than expected. This practice is commonplace in the WENO literature, where authors often use lower order time-integrators such as SSP-RK3 [26, 42], and present results indicating fifth or higher order convergence rates. In this work, we refine our mesh until the temporal error is observable.


Convergence study for Burger’s equation. The number of mesh elements \( m_x \) is given in the first column, and errors are defined in (4.3). Two different CFL numbers of \( \nu = 0.3 \) and \( \nu = 0.5 \) as defined in (4.4) have been chosen in order to offer a comparison with previous Lax-Wendroff based finite difference WENO methods [10, 13]. The “Order” columns refer to the algebraic order of convergence, computed as the base-2 logarithm of the ratio of two successive error norms.

| Mesh | Error (\( \nu = 0.3 \)) | Order | Error (\( \nu = 0.5 \)) | Order |
|------|----------------------|-------|----------------------|-------|
| 10   | 1.94 × 10^{-02}      | —     | 2.05 × 10^{-02}      | —     |
| 20   | 3.40 × 10^{-03}      | 2.52  | 3.60 × 10^{-03}      | 2.51  |
| 40   | 3.68 × 10^{-04}      | 3.21  | 3.92 × 10^{-04}      | 3.20  |
| 80   | 1.86 × 10^{-05}      | 4.31  | 2.12 × 10^{-05}      | 4.21  |
| 160  | 8.41 × 10^{-07}      | 4.46  | 1.47 × 10^{-06}      | 3.86  |
| 320  | 5.54 × 10^{-08}      | 3.92  | 1.41 × 10^{-07}      | 3.37  |
| 640  | 4.75 × 10^{-09}      | 3.54  | 1.56 × 10^{-08}      | 3.18  |
| 1280 | 4.82 × 10^{-10}      | 3.30  | 1.86 × 10^{-09}      | 3.07  |
| 2560 | 5.34 × 10^{-11}      | 3.17  | 2.28 × 10^{-10}      | 3.03  |
| 5120 | 6.28 × 10^{-12}      | 3.09  | 2.81 × 10^{-11}      | 3.02  |

4.2. The 1D Euler equations. The Euler equations describe the evolution of density \( \rho \), momentum \( \rho u \) and energy \( E \) of an ideal gas:

\[
\begin{pmatrix}
\frac{\partial \rho}{\partial t} \\
\frac{\partial \rho u}{\partial t} \\
\frac{\partial E}{\partial t}
\end{pmatrix}
+ \begin{pmatrix}
\rho u \\
\rho u^2 + p \\
(E + p)u
\end{pmatrix}
= 0,
\]

(4.5)

where \( p \) is the pressure. The energy \( E \) is related to the primitive variables \( \rho, u \) and \( p \) by \( E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho u^2 \), and \( \gamma \) is the ratio of specific heats. For all of our simulations, we take \( \gamma = 1.4 \).

4.2.1. The 1D Euler equations: a Riemann problem. We present a classic test case of a difficult shock tube, which is commonly referred to as the Lax shock tube. However, the initial conditions are those defined by Harten [22, 43]:

\[
(\rho, \rho u, E)^T = \begin{cases} 
(0.445, 0.3111, 8.928)^T , & \text{if } x \leq 0.5, \\
(0.5, 0, 1.4275)^T, & \text{otherwise}.
\end{cases}
\]

(4.6)

We select \( t = 0.16 \) for the final time of our simulation [10], and we use a computational domain of \([0, 1]\) with outflow boundary conditions. Results for this problem are presented in Fig. 4.1.

4.2.2. The 1D Euler equations: shock entropy. Our final 1D test case is another problem that is popular in the literature [27]. The initial conditions are

\[
(\rho, u, p) = \begin{cases} 
(3.857143, 2.629369, 10.3333), & x < -4, \\
(1 + \epsilon \sin(5x), 0, 1), & x \geq -4,
\end{cases}
\]

with a computational domain of \([-5, 5]\). The final time for this simulation is \( t = 1.8 \). With \( \epsilon = 0 \), this is a pure Mach 3 shock moving to the right. We follow the common practice of setting \( \epsilon = 0.2 \). Results for the new method are presented in Fig. 4.2.
4.3. The 2D Euler equations. In 2D the Euler equations become

\[
\begin{pmatrix}
\rho \\
\rho u \\
\rho v \\
\mathcal{E}
\end{pmatrix},_t + \begin{pmatrix}
\rho u \\
\rho u^2 + p \\
\rho uv \\
(\mathcal{E} + p)u
\end{pmatrix},_x + \begin{pmatrix}
\rho v \\
\rho uv \\
\rho v^2 + p, \\
(\mathcal{E} + p)v
\end{pmatrix},_y = 0,
\]

where \(u\) is the \(x\)-component of velocity, and \(v\) is the \(y\)-component of velocity. Now, the total energy relies on both components of the velocity: \(\mathcal{E} = \frac{\rho}{\gamma - 1} + \frac{1}{2} \rho (u^2 + v^2)\).

4.3.1. The 2D Euler equations: a smooth solution. We present convergence results for a smooth solution proposed elsewhere in the literature \[10,14\]. The initial conditions are prescribed by

\[(\rho, u, v, p) = (1 + 0.2 \sin(\pi(x + y)), 0.7, 0.3, 1.0),\]

and we compute the solution on a periodic domain \(\Omega = [0, 2] \times [0, 2]\). The exact solution has an evolving density, \(\rho(t, x, y) = 1 + 0.2 \sin(\pi(x + y - (u + v)t))\), and constant velocities \(u = 0.7, v = 0.3\) and pressure \(p = 1.0\). A convergence study for this 2D problem is presented in Table 4.2. In 2D, we the relative \(L^1\) error is defined by

\[
\text{Error} := \frac{\Delta x \Delta y \sum_{i=1}^{m_x} \sum_{j=1}^{m_y} |q_n^i - q(t^n, x_i, y_j)|}{\Delta x \Delta y \sum_{i=1}^{m_x} \sum_{j=1}^{m_y} |q(t^n, x_i, y_j)|}
\]
Table 4.2

Euler equations: smooth solutions. Shown here is a convergence study for smooth solutions to 2D Euler equations as presented in §4.3.1. We use a uniform grid with \( m_x = m_y \) grid points reported in the first column. Errors are defined as relative errors as in (4.8). We use a CFL number of \( \nu = 0.4 \) as defined in (4.9). For comparison, in the last two columns, we report data for a classical finite difference WENO simulation run with the SSP-RK3 method described in Gottlieb and Shu [42], and remark that the errors are on par with each other. Again, we refine until the third-order overall accuracy (for each method) is observable, and note that we obtain similar errors to the popular time integrator we compared against.

| Mesh | PIF-WENO | Order | SSP-RK3 | Order |
|------|----------|-------|---------|-------|
| 50   | 5.4101 \times 10^{-16} | — | 5.4514 \times 10^{-16} | — |
| 100  | 1.9177 \times 10^{-10} | 4.818 | 1.9289 \times 10^{-10} | 4.821 |
| 200  | 8.8529 \times 10^{-10} | 4.437 | 8.8841 \times 10^{-10} | 4.440 |
| 400  | 6.3477 \times 10^{-10} | 3.802 | 6.3557 \times 10^{-10} | 3.805 |
| 800  | 6.4601 \times 10^{-11} | 3.297 | 6.4636 \times 10^{-11} | 3.298 |
| 1600 | 7.6148 \times 10^{-12} | 3.085 | 7.6448 \times 10^{-12} | 3.080 |

and the CFL number is defined by

\[
\nu := \Delta t \max_{i,j} \left\{ \frac{\alpha^*_{i-1/2,j}}{\Delta x}, \frac{\alpha^*_{i,j-1/2}}{\Delta y} \right\}.
\] (4.9)

4.3.2. The 2D Euler equations: double mach reflection. We now present results for the so-called double mach reflection originally proposed by Woodward and Colella [44] that was intended to serve as a test problem to compare numerical methods. This problem has since become ubiquitous in the literature [5, 29, 33, 45, 46]. The initial conditions describe a Mach-10 shock incident upon a single wedge (c.f. Fig. 4 in [45]). The computational domain is tilted, so that the wedge is positioned along the bottom of the grid. The shock forms an oblique angle with the mesh, where it starts at the front of the wedge located at \((x, y) = (1/6, 0)\), and continues up to the top of the computational domain located at \(y = 1\). The initial conditions describe two constant values, to the left and right of the shock

\[
(p, u, v, p) = \begin{cases} 
(1.4, 0, 0, 1.0)^T, & \text{if } x < \frac{1}{6} + \frac{y}{\sqrt{3}}, \\
(8.0, \frac{8.25\sqrt{3}}{2}, -\frac{8.25}{2}, 116.5)^T, & \text{otherwise.}
\end{cases}
\] (4.10)

Reflective boundary conditions are applied along the bottom edge when \(x > 1/6\), and the exact pre- and post-shock values are padded everywhere else. In order to pad the correct boundary conditions along the top, we require the exact location \(x_s\) of the shock at time \(t\) along the line \(y = 1\), which is given by \(x_s(t) = 1/6 + (20t + 1)/\sqrt{3}\). Results for this problem are presented in Fig. 4.3 Given that a consensus concerning what contour lines to plot has not been reached, we plot the 30 equally spaced contours from \(\rho = 1.728\) to \(\rho = 20.74\), as reported in one of the many simulations presented by the original authors [44].

5. Conclusions. We have formulated and presented results for the Picard integral formulation of the finite difference WENO method. The new formulation allows us to step back from the classical method of lines (MOL) formulations and include the spatial discretization as part of the temporal discretizations. We have demonstrated how Taylor methods can be developed from this new vantage, and have introduced
results for the proposed formulation in one- and two-dimensions, that indicate the new method competes with current state of the art technology. Future work will include investigating positivity-preserving limiters, as well as other temporal discretizations of the time-averaged fluxes including Runge-Kutta and multiderivative methods.

Acknowledgments One of the authors (D.C. Seal) would like to thank Gwendolyn Miller Seal for creating one of the figures used in this manuscript, and Nathan Collins for aide with manipulating the symbolic expressions used to generate source code for the simulations.

Appendix A. WENO reconstruction.

For completeness, we present the coefficients required to reproduce the results in this paper. We restrict our attention to the fifth-order case, which uses a five point stencil shifted to the left or right of each cell interface:

$$\begin{align*}
    u_{i+1/2}^+ & := WENO5^+[\bar{u}_{i-2}, \bar{u}_{i-1}, \bar{u}_{i}, \bar{u}_{i+1}, \bar{u}_{i+2}], \\
    u_{i+1/2}^- & := WENO5^-[\bar{u}_{i-1}, \bar{u}_{i}, \bar{u}_{i+1}, \bar{u}_{i+2}, \bar{u}_{i+3}].
\end{align*}$$

(A.1)

Here, the known values are the cell averages

$$\bar{u}_j = \frac{1}{\Delta x} \int_{x_j - \Delta x/2}^{x_j + \Delta x/2} u(x) dx$$

for some scalar function $u$. The purpose of having a “+” and “−” value is to define upwind stencils that are numerically stable [11]. We define coefficients for the function $WENO5^+$, and by symmetry, we define

$$WENO5^-[\bar{u}_{i-1}, \bar{u}_{i}, \ldots \bar{u}_{i+3}] := WENO5^+[\bar{u}_{i+3}, \bar{u}_{i+2}, \ldots \bar{u}_{i-1}].$$

(A.1)

Three sub-stencils define quadratic polynomials that offer competing third-order ac-
curate values for $u(x_{i+1/2})$:

\[
\begin{align*}
   u_{i+1/2}^{(0)} &= \frac{1}{3} \bar{u}_{i-2} - \frac{7}{6} \bar{u}_{i-1} + \frac{11}{6} \bar{u}_i, \\
   u_{i+1/2}^{(1)} &= -\frac{1}{6} \bar{u}_{i-1} + \frac{5}{6} \bar{u}_i + \frac{1}{3} \bar{u}_{i+1}, \\
   u_{i+1/2}^{(2)} &= \frac{1}{3} \bar{u}_i + \frac{5}{6} \bar{u}_{i+1} - \frac{1}{6} \bar{u}_{i+2}.
\end{align*}
\] (A.2a)

A linear combination of (A.2a)-(A.2c) yields a fifth-order accurate approximation $u(x_{i+1/2}) \approx \gamma_0 u_{i+1/2}^{(0)} + \gamma_1 u_{i+1/2}^{(1)} + \gamma_2 u_{i+1/2}^{(2)}$ with linear weights $\gamma_j \in \{1/10, 3/5, 3/10\}$. The WENO procedure replaces the linear weights $\gamma_j$ with nonlinear weights $\omega_j$ based on the smoothness indicators $\beta_j$. In fifth-order WENO, the indicators are

\[
\begin{align*}
   \beta_0 &= \frac{13}{12} (\bar{u}_{i-2} - 2 \bar{u}_{i-1} + \bar{u}_i)^2 + \frac{1}{4} (\bar{u}_{i-2} - 4 \bar{u}_{i-1} + 3 \bar{u}_i)^2, \\
   \beta_1 &= \frac{13}{12} (\bar{u}_{i-1} - 2 \bar{u}_i + \bar{u}_{i+1})^2 + \frac{1}{4} (\bar{u}_{i-1} - \bar{u}_{i+1})^2, \\
   \beta_2 &= \frac{13}{12} (\bar{u}_i - 2 \bar{u}_{i+1} + \bar{u}_{i+2})^2 + \frac{1}{4} (3 \bar{u}_i - 4 \bar{u}_{i+1} + \bar{u}_{i+2})^2.
\end{align*}
\] (A.3)

The Jiang and Shu weights \cite{29} are defined by

\[
\omega_k = \frac{\hat{\omega}_k}{\sum_{l=0}^{2} \hat{\omega}_l}, \quad \hat{\omega}_k = \frac{\gamma_k}{(\beta_k + \varepsilon)^p}.
\] (A.4)

We use the power parameter $p = 2$ and regularization parameter $\varepsilon = 10^{-12}$ for all of our simulations. With these definitions in place, the final reconstructed value is defined as

\[
\text{WENO5}^+[\bar{u}_{i-2}, \ldots, \bar{u}_{i+2}] := \omega_0 u_{i+1/2}^{(0)} + \omega_1 u_{i+1/2}^{(1)} + \omega_2 u_{i+1/2}^{(2)}.
\] (A.5)

**References.**

[1] P. Lax and B. Wendroff, “Systems of conservation laws,” *Comm. Pure Appl. Math.*, vol. 13, pp. 217–237, 1960.

[2] E. F. Toro and V. A. Titarev, “ADER: Arbitrary high order Godunov approach,” *Journal of Scientific Computing*, vol. 17, no. 1-4, pp. 609–618, 2002.

[3] E. F. Toro and V. A. Titarev, “Solution of the generalized Riemann problem for advection-reaction equations,” *R. Soc. Lond. Proc. Ser. A Math. Phys. Eng. Sci.*, vol. 458, no. 2018, pp. 271–281, 2002.

[4] E. F. Toro and V. A. Titarev, “TVD fluxes for the high-order ADER schemes,” *J. Sci. Comput.*, vol. 24, no. 3, pp. 285–309, 2005.

[5] V. A. Titarev and E. F. Toro, “ADER schemes for three-dimensional non-linear hyperbolic systems,” *J. Comput. Phys.*, vol. 204, no. 2, pp. 715–736, 2005.

[6] M. Dumbser and M. Käser, “Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems,” *J. Comput. Phys.*, vol. 221, no. 2, pp. 693–723, 2007.

[7] E. F. Toro and V. A. Titarev, “Derivative Riemann solvers for systems of conservation laws and ADER methods,” *J. Comput. Phys.*, vol. 212, no. 1, pp. 150–165, 2006.

[8] M. R. Norman and H. Finkel, “Multi-moment ADER-Taylor methods for systems of conservation laws with source terms in one dimension,” *J. Comput. Phys.*, vol. 231, no. 20, pp. 6622–6642, 2012.
[9] D. S. Balsara, C. Meyer, M. Dumbser, H. Du, and Z. Xu, “Efficient implementation of ADER schemes for Euler and magnetohydrodynamical flows on structured meshes—speed comparisons with Runge-Kutta methods,” *J. Comput. Phys.*, vol. 235, pp. 934–969, 2013.

[10] J. Qiu and C.-W. Shu, “Finite difference WENO schemes with Lax-Wendroff-type time discretizations,” *SIAM J. Sci. Comput.*, vol. 24, no. 6, pp. 2185–2198, 2003.

[11] W. Liu, J. Cheng, and C.-W. Shu, “High order conservative Lagrangian schemes with Lax-Wendroff type time discretization for the compressible Euler equations,” *J. Comput. Phys.*, vol. 228, no. 23, pp. 8872–8891, 2009.

[12] C. Lu and J. Qiu, “Simulations of shallow water equations with finite difference Lax-Wendroff weighted essentially non-oscillatory schemes,” *J. Sci. Comput.*, vol. 47, no. 3, pp. 281–302, 2011.

[13] Y. Jiang, C.-W. Shu, and M. Zhang, “An alternative formulation of finite difference weighted ENO schemes with Lax-Wendroff time discretization for conservation laws,” *SIAM J. Sci. Comput.*, vol. 35, no. 2, pp. A1137–A1160, 2013.

[14] J. Qiu, M. Dumbser, and C.-W. Shu, “The discontinuous Galerkin method with Lax-Wendroff type time discretizations,” *Comput. Methods Appl. Mech. Eng.*, vol. 194, no. 42-44, pp. 4528–4543, 2005.

[15] M. Dumbser and C.-D. Munz, “Building blocks for arbitrary high order discontinuous Galerkin schemes,” *J. Sci. Comput.*, vol. 27, no. 1-3, pp. 215–230, 2006.

[16] A. Taube, M. Dumbser, D. S. Balsara, and C.-D. Munz, “Arbitrary high-order discontinuous Galerkin schemes for the magnetohydrodynamic equations,” *J. Sci. Comput.*, vol. 30, no. 3, pp. 441–464, 2007.

[17] J. Qiu, “A numerical comparison of the Lax-Wendroff discontinuous Galerkin method based on different numerical fluxes,” *J. Sci. Comput.*, vol. 30, no. 3, pp. 345–367, 2007.

[18] G. Gassner, M. Dumbser, F. Hindenlang, and C.-D. Munz, “Explicit one-step time discretizations for discontinuous Galerkin and finite volume schemes based on local predictors,” *J. Comput. Phys.*, vol. 230, no. 11, pp. 4232–4247, 2011.

[19] M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz, “A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes,” *J. Comput. Phys.*, vol. 227, no. 18, pp. 8209–8253, 2008.

[20] B. van Leer, “Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov’s method,” *J. Comput. Phys.*, vol. 32, no. 1, pp. 101–136, 1979.

[21] A. Rodinov, “Methods of increasing the accuracy in Godunov’s scheme,” *USSR Computational Mathematics and Mathematical Physics*, vol. 27, no. 6, pp. 164 – 169, 1987.

[22] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy, “Uniformly high-order accurate essentially nonoscillatory schemes. III,” *J. Comput. Phys.*, vol. 71, no. 2, pp. 231–303, 1987.

[23] J. B. Bell, P. Colella, and J. A. Trangenstein, “Higher order Godunov methods for general systems of hyperbolic conservation laws,” *J. Comput. Phys.*, vol. 82, no. 2, pp. 362–397, 1989.

[24] I. Men’shov, “Increasing the order of approximation of Godunov’s scheme using solutions of the generalized Riemann problem,” *USSR Computational Mathematics and Mathematical Physics*, vol. 30, no. 5, pp. 54 – 65, 1990.

[25] J.-M. Qiu and C.-W. Shu, “Conservative high order semi-Lagrangian finite dif-
ference WENO methods for advection in incompressible flow,” *J. Comput. Phys.*, vol. 230, no. 4, pp. 863–889, 2011.

[26] C.-W. Shu and S. Osher, “Efficient implementation of essentially nonoscillatory shock-capturing schemes,” *J. Comput. Phys.*, vol. 77, no. 2, pp. 439–471, 1988.

[27] C.-W. Shu and S. Osher, “Efficient implementation of essentially nonoscillatory shock-capturing schemes. II,” *J. Comput. Phys.*, vol. 83, no. 1, pp. 32–78, 1989.

[28] X.-D. Liu, S. Osher, and T. Chan, “Weighted essentially non-oscillatory schemes,” *J. Comput. Phys.*, vol. 115, no. 1, pp. 200–212, 1994.

[29] G.-S. Jiang and C.-W. Shu, “Efficient implementation of weighted ENO schemes,” *J. Comput. Phys.*, vol. 126, no. 1, pp. 202–228, 1996.

[30] C.-W. Shu, “Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws,” in *Advanced numerical approximation of nonlinear hyperbolic equations* (Cetraro, 1997), vol. 1697 of *Lecture Notes in Math.*, pp. 325–432, Berlin: Springer, 1998.

[31] C.-W. Shu, “High order weighted essentially nonoscillatory schemes for convection dominated problems,” *SIAM Rev.*, vol. 51, no. 1, pp. 82–126, 2009.

[32] P. L. Roe, “Approximate Riemann solvers, parameter vectors, and difference schemes,” *J. Comput. Phys.*, vol. 43, no. 2, pp. 357–372, 1981.

[33] D. S. Balsara and C.-W. Shu, “Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy,” *J. Comput. Phys.*, vol. 160, no. 2, pp. 405–452, 2000.

[34] S. Vukovic and L. Sopota, “ENO and WENO schemes with the exact conservation property for one-dimensional shallow water equations,” *J. Comput. Phys.*, vol. 179, no. 2, pp. 593–621, 2002.

[35] E. Hairer and G. Wanner, “Multistep-multistage-multiderivative methods of ordinary differential equations,” *Computing (Arch. Elektron. Rechnen)*, vol. 11, no. 3, pp. 287–303, 1973.

[36] D. C. Seal, Y. Güçlü, and A. J. Christlieb, “High-order multiderivative time integrators for hyperbolic conservation laws,” *J. Sci. Comput.*, pp. 1–40, 2013.

[37] J. H. Williamson, “Low-storage Runge-Kutta schemes,” *J. Comput. Phys.*, vol. 35, no. 1, pp. 48–56, 1980.

[38] D. I. Ketcheson, “Highly efficient strong stability-preserving Runge-Kutta methods with low-storage implementations,” *SIAM J. Sci. Comput.*, vol. 30, no. 4, pp. 2113–2136, 2008.

[39] D. I. Ketcheson, “Runge-Kutta methods with minimum storage implementations,” *J. Comput. Phys.*, vol. 229, no. 5, pp. 1763–1773, 2010.

[40] J. Niegemann, R. Diehl, and K. Busch, “Efficient low-storage Runge-Kutta schemes with optimized stability regions,” *J. Comput. Phys.*, vol. 231, no. 2, pp. 364–372, 2012.

[41] R. LeVeque, *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002.

[42] S. Gottlieb and C.-W. Shu, “Total variation diminishing Runge-Kutta schemes,” *Math. Comp.*, vol. 67, no. 221, pp. 73–85, 1998.

[43] A. Harten, “The artificial compression method for computation of shocks and contact discontinuities. III. Self-adjusting hybrid schemes,” *Math. Comp.*, vol. 32, no. 142, pp. 363–389, 1978.

[44] P. Woodward and P. Colella, “The numerical simulation of two-dimensional fluid flow with strong shocks,” *J. Comput. Phys.*, vol. 54, no. 1, pp. 115–173, 1984.

[45] J.-F. Remacle, J. E. Flaherty, and M. S. Shephard, “An adaptive discontinu-
ous Galerkin technique with an orthogonal basis applied to compressible flow problems,” *SIAM Rev.*, vol. 45, no. 1, pp. 53–72 (electronic), 2003.

[46] J. Shi, Y.-T. Zhang, and C.-W. Shu, “Resolution of high order WENO schemes for complicated flow structures,” *J. Comput. Phys.*, vol. 186, no. 2, pp. 690–696, 2003.