



AENO: a Novel Reconstruction Method in Conjunction with ADER Schemes for Hyperbolic Equations

Eleuterio F. Toro¹ · Andrea Santacà² · Gino I. Montecinos³ · Morena Celant² · Lucas O. Müller²

Received: 30 November 2020 / Revised: 31 May 2021 / Accepted: 9 June 2021 /
Published online: 3 September 2021
© The Author(s) 2021

Abstract

In this paper, we present a novel spatial reconstruction scheme, called **AENO**, that results from a special averaging of the ENO polynomial and its closest neighbour, while retaining the stencil direction decided by the ENO choice. A variant of the scheme, called m-AENO, results from averaging the modified ENO (m-ENO) polynomial and its closest neighbour. The concept is thoroughly assessed for the one-dimensional linear advection equation and for a one-dimensional non-linear hyperbolic system, in conjunction with the fully discrete, high-order ADER approach implemented up to fifth order of accuracy in both space and time. The results, as compared to the conventional ENO, m-ENO and WENO schemes, are very encouraging. Surprisingly, our results show that the L_1 -errors of the novel AENO approach are the smallest for most cases considered. Crucially, for a chosen error size, AENO turns out to be the most efficient method of all five methods tested.

Keywords Hyperbolic equations · High-order ADER · ENO/m-ENO/WENO · Novel reconstruction technique AENO/m-AENO

Mathematics Subject Classification 35L50 · 65M08

1 Introduction

Circumventing Godunov's theorem [8] is a 60-year old challenge to numerical mathematicians concerned with the development of high-order methods for solving hyperbolic equations. The invention of total variation diminishing (TVD) methods in the 1970s and 1980s

✉ Lucas O. Müller
lucas.muller@unitn.it
Eleuterio F. Toro
eleuterio.toro@unitn.it

¹ Laboratory of Applied Mathematics, DICAM, University of Trento, Trento, Italy

² Department of Mathematics, University of Trento, Trento, Italy

³ Department of Natural Sciences and Technology, Universidad de Aysén, Obispo Vielmo 62, Coyhaique, Chile

[11, 16, 23, 29, 43] represented an important step forward in facing the challenge. However such methods, even if very effective in practice, are limited to, at most, second-order accuracy for smooth solutions. Currently, the alternative approaches relying on non-linear spatial reconstruction pioneered by Harten and others [12, 13], such as ENO, m-ENO and WENO [14, 15, 26–28] have become the dominant approach. Unlike TVD methods, however, ENO, m-ENO and WENO based methods cannot guarantee the absence of spurious oscillations for scalar problems. However, these methods guarantee that such oscillations vanish in the limit of mesh refinement. This is not the case for linear methods, for which, no matter how fine the mesh is, the amplitude of the oscillations will remain unchanged. The initially popular ENO method has, in recent years, been largely discarded by most researchers and practitioners. The reconstruction scenery is currently dominated by the WENO approach, in a variety of forms. See for example [5, 31] and [6].

In this paper, we propose a novel spatial reconstruction method that is akin to both ENO and WENO. The method, called **AENO**, results from averaging two polynomials, the classical ENO polynomial and its closest neighbour, while the search for the stencil remains commanded by ENO. A variant of the scheme, called **m-AENO**, results from averaging the modified ENO (m-ENO) polynomial of Shu [26] and its closest neighbour; **m-AENO** is equivalent to **AENO** for 2nd and 3rd order of accuracy. Here, the reconstruction schemes are applied in conjunction with the fully discrete high-order ADER approach [4, 7, 30, 34, 40]. See [36] for an up-to-date review of ADER and references to the many contributions to its development. Schemes of up to 5th order of accuracy in space and time are implemented and tested, first for the linear advection equation and then for a non-linear hyperbolic system, namely the blood flow equations. For both problems, we first carry out experiments to compare numerical solutions with exact solutions for short and long evolution times. Results for three reconstruction methods are compared, namely ENO, m-ENO, WENO and the novel AENO and m-AENO schemes. Then we carry out a convergence rate study for both types of problems, the linear advection equation and the blood flow equations. Overall, the results are encouraging. For most problems, all five reconstruction methods give comparable results. Surprisingly, our results show that the L_1 -errors of the novel AENO/m-AENO schemes are the smallest, for most cases considered.

The rest of this paper is structured as follows. In Sect. 2 we review the fully discrete ADER methodology to solve hyperbolic equations, first for the linear scalar case and then for non-linear systems. In Sect. 3 we briefly introduce necessary background on the ENO reconstruction method. In Sect. 4 we describe the new spatial reconstruction technique AENO and its variant m-AENO; in Sect. 5 we show results for the linear advection equation. In Sect. 6 we apply the methods to the blood flow equations and in Sect. 7 we draw conclusions.

2 Review of the ADER Methodology for Hyperbolic Equations

Here, we briefly review the ADER approach to construct one-step, fully discrete high-order Godunov methods for solving hyperbolic balance laws, whose building block is the **generalized Riemann problem**, a piece-wise smooth data Cauchy problem including stiff or non-stiff source terms, rather than the conventional piece-wise constant data homogeneous Riemann problem. In this section we review the ADER methodology as applied to one-dimensional systems of hyperbolic balance laws. We first deal with the scalar linear case.

2.1 ADER for the Scalar Linear Case

Before proceeding, we recall the finite volume framework for a general scalar balance law.

2.1.1 The Finite Volume Framework

Consider the one-dimensional balance law

$$\partial_t q + \partial_x f(q) = s(q), \quad (1)$$

solved by a finite volume method, with space-time volumes

$$V = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [t_n, t_{n+1}]. \quad (2)$$

Exact integration of (1) on V in (2) gives

$$q_i^{n+1} = q_i^n - \frac{\Delta t}{\Delta x} [f_{i+\frac{1}{2}} - f_{i-\frac{1}{2}}] + \Delta t s_i, \quad (3)$$

with definitions

$$\left. \begin{aligned} q_i^n &= \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q(x, t_n) dx, \\ f_{i+\frac{1}{2}} &= \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} f(q(x_{i+\frac{1}{2}}, t)) dt, \\ s_i &= \frac{1}{\Delta t} \frac{1}{\Delta x} \int_{t_n}^{t_{n+1}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} s(q(x, t)) dx dt \end{aligned} \right\} \quad (4)$$

and

$$\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}, \quad \Delta t = t_{n+1} - t_n, \quad x_i = \frac{1}{2}(x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}}). \quad (5)$$

2.1.2 ADER for Linear Advection-Reaction

The ADER method [38] departs from the finite volume formula (3), understood as resulting from approximations to the integrals in (4). The one-step updating formula (3) updates the cell average q_i^n to q_i^{n+1} , for which definitions for the **numerical flux** $f_{i+\frac{1}{2}}$ and the **numerical source** s_i are required. The rest of the presentation of the ADER approach will be carried out with (1) as the linear advection-reaction equation, in which $f(q) = \lambda q$ and $s(q) = \beta q$, with λ a constant wave propagation speed and β constant reaction rate satisfying $\beta \leq 0$.

2.1.3 The ADER Flux and the Generalized Riemann Problem

To determine the numerical flux $f_{i+\frac{1}{2}}$ in (3) we compute a high-order accurate approximation to the flux integral in (4), which in turn requires a high-order approximation to the integrand $q(x_{i+\frac{1}{2}}, t)$. This is achieved by solving, approximately, the following **generalized Riemann problem**

$$\left. \begin{aligned} \text{PDE: } & \partial_t q + \lambda \partial_x q = \beta q, \\ \text{IC: } & q(x, 0) = \begin{cases} p_i(x) & \text{if } x < 0, \\ p_{i+1}(x) & \text{if } x > 0, \end{cases} \end{aligned} \right\} \tag{6}$$

in which the polynomials $p_i(x)$ and $p_{i+1}(x)$ are **reconstruction** polynomials of arbitrary degree K . To solve (6) we adopt here the Toro-Titarev [41] method, which starts from the LeFloch-Raviart expansion [18] for the approximate solution

$$q_{i+\frac{1}{2}}(\tau) = q(0, 0_+) + \sum_{k=1}^K \left[\partial_t^{(k)} q(0, 0_+) \right] \frac{\tau^k}{k!}. \tag{7}$$

The leading term of the expansion. The leading term $q(0, 0_+)$ in (7) results from solving the **conventional** piece-wise-constant data Riemann problem

$$\left. \begin{aligned} \text{PDE: } & \partial_t q + \lambda \partial_x q = 0, \\ \text{IC: } & q(x, 0) = \begin{cases} p_i(0) & \text{if } x < 0, \\ p_{i+1}(0) & \text{if } x > 0, \end{cases} \end{aligned} \right\} \tag{8}$$

in which the initial condition consists of the boundary extrapolated values $p_i(0)$ from the left and $p_{i+1}(0)$ from the right. The position of the interface $x_{i+\frac{1}{2}}$ corresponds to 0, in local coordinates.

The higher-order terms of the expansion. To compute the higher-order terms in (7) one requires the determination of the time derivatives $\partial_t^{(k)} q(0, 0_+)$. In the first version of ADER [38] it was proposed to apply the Cauchy-Kovalevskaya procedure to the governing equation in (6). Just for a moment let us assume the linear homogenous case in (6) ($\beta = 0$). Then, the Cauchy-Kovalevskaya procedure gives

$$\partial_t q + \lambda \partial_x q = 0, \quad \partial_t q = -\lambda \partial_x q, \quad \partial_t^{(2)} q = \lambda^2 \partial_x^{(2)} q. \tag{9}$$

In fact, it is easily shown that for an arbitrary positive integer k ,

$$\partial_t^{(k)} q = (-1)^k \lambda^k \partial_x^{(k)} q. \tag{10}$$

Similarly, for the inhomogeneous case with non-zero source term in (6), it can also be shown that the time derivatives can still be expressed as functions of spacial derivatives, as follows:

$$\partial_t^{(k)} q(x, t) = \sum_{n=0}^k \left[\frac{k!}{(k-n)!n!} \beta^n (-\lambda)^{k-n} \right] \partial_x^{(k-n)} q(x, t). \tag{11}$$

Now the pending problem is that of determining $\partial_x^{(l)} q(0, 0_+)$ in (7), for which a useful step is the construction of **evolution equations for the spatial derivatives**. For the homogenous case, it can be easily shown that $\partial_x^{(l)} q(x, t)$ obeys the same evolution equation as $q(x, t)$, that is

$$\partial_t(\partial_x^{(l)}q(x, t)) + \lambda\partial_x(\partial_x^{(l)}q(x, t)) = 0. \tag{12}$$

The next problem is to explore the possibility of formulating initial value problems for (12). In fact this is possible if we find suitable initial conditions. If for each computational cell we have a spatial polynomial representation $p_i(x)$ for $q(x, t_n)$, via a spatial reconstruction technique for example, then it is possible to pose **classical Riemann problems** for spatial derivatives as follows:

$$\left. \begin{aligned} \text{PDE: } & \partial_t(\partial_x^{(l)}q(x, t)) + \lambda\partial_x(\partial_x^{(l)}q(x, t)) = 0, \\ \text{IC: } & \partial_x^{(l)}q(x, 0) = \begin{cases} \partial_x^{(l)}p_i(0) & \text{if } x < 0, \\ \partial_x^{(l)}p_{i+1}(0) & \text{if } x > 0. \end{cases} \end{aligned} \right\} \tag{13}$$

The initial condition for these classical Riemann problems (piece-wise constant data) consists of intercell-boundary extrapolated spatial derivatives of the reconstruction polynomials, $p_i(x)$ from the left and $p_{i+1}(x)$ from the right. Solutions of classical Riemann problems for the spatial derivatives, as in (13), will give the spatial derivatives $\partial_x^{(l)}q(0, 0_+)$ in (7), of any order l . This in turn determines the time partial derivatives in (11) and hence in (7). We have therefore found the complete solution of the homogeneous version of the generalized Riemann problem (6), which is given as

$$q_{i+\frac{1}{2}}(\tau) = \begin{cases} p_i(0) + \sum_{k=1}^K ((-1)^k \lambda^k \partial_x^{(k)} p_i(0)) \frac{(\tau)^k}{(k+1)!} & \text{if } \lambda > 0, \\ p_{i+1}(0) + \sum_{k=1}^K ((-1)^k \lambda^k \partial_x^{(k)} p_{i+1}(0)) \frac{(\tau)^k}{(k+1)!} & \text{if } \lambda < 0. \end{cases} \tag{14}$$

The solution for the inhomogeneous case is similar. The numerical flux results from an approximation to the flux integral in (4), which is now obtained as

$$f_{i+\frac{1}{2}} = \frac{1}{\Delta t} \int_0^{\Delta t} f(q_{i+\frac{1}{2}}(\tau)) d\tau. \tag{15}$$

Exact integration gives

$$f_{i+\frac{1}{2}} = \begin{cases} \lambda \left[p_i(0) + \sum_{k=1}^K ((-1)^k \lambda^k \partial_x^{(k)} p_i(0)) \frac{(\Delta t)^k}{(k+1)!} \right] & \text{if } \lambda > 0, \\ \lambda \left[p_{i+1}(0) + \sum_{k=1}^K ((-1)^k \lambda^k \partial_x^{(k)} p_{i+1}(0)) \frac{(\Delta t)^k}{(k+1)!} \right] & \text{if } \lambda < 0. \end{cases} \tag{16}$$

Generally, the numerical flux in (15) will require numerical integration of the appropriate order. Inserting the numerical flux (16) into the conservative finite volume formula (3) will give a numerical approximation of order $K + 1$ in both space and time. In the presence of

source terms in the governing equations, one must include the source terms when performing the Cauchy-Kovalevskaya procedure, as these enter in the calculation of the numerical flux. In addition, one must compute the numerical source to the appropriate order of accuracy. This requires an approximation of the source integral in (4) in the entire control volume V in (2), for which a high-order approximation to $q(x, t)$ within V is needed. Such approximation is found in a manner similar to that of the generalized Riemann problem.

2.2 ADER for Non-linear Systems

Consider a general one-dimensional system of hyperbolic balance laws

$$\partial_t \mathbf{Q}(x, t) + \partial_x \mathbf{F}(\mathbf{Q}(x, t)) = \mathbf{S}(\mathbf{Q}(x, t)). \tag{17}$$

The vectors \mathbf{Q} , $\mathbf{F}(\mathbf{Q})$ and $\mathbf{S}(\mathbf{Q})$ define conserved variables, fluxes and sources, respectively. Assuming the space/time domain to be discretised by finite volumes V in (2), exact integration of (17) in V gives

$$\mathbf{Q}_i^{n+1} = \mathbf{Q}_i^n - \frac{\Delta t}{\Delta x} (\mathbf{F}_{i+\frac{1}{2}} - \mathbf{F}_{i-\frac{1}{2}}) + \Delta t \mathbf{S}_i, \tag{18}$$

with

$$\left. \begin{aligned} \mathbf{Q}_i^n &= \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{Q}(x, t^n) dx, & \mathbf{F}_{i+\frac{1}{2}} &= \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{Q}(x_{i+\frac{1}{2}}, t)) dt, \\ \mathbf{S}_i &= \frac{1}{\Delta t \Delta x} \int_{t^n}^{t^{n+1}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{S}(\mathbf{Q}(x, t)) dx dt. \end{aligned} \right\} \tag{19}$$

Numerically, (18)–(19) motivate the construction of finite volume methods to solve (17), in which suitable approximations to the integrals (19) determine the numerical flux $\mathbf{F}_{i+\frac{1}{2}}$ and the numerical source \mathbf{S}_i in the approximating formula (18). See Sect. 2.1.2.

2.2.1 ADER and the Godunov Method

The classical Godunov method [8] results from (18)–(19) by assuming $\mathbf{S}(\mathbf{Q}) = \mathbf{0}$ and $\mathbf{Q}(x_{i+\frac{1}{2}}, t) = \tilde{\mathbf{Q}}_{i+\frac{1}{2}}(x/t)$ to be the similarity solution of the piece-wise constant data, homogeneous, Riemann problem

$$\left. \begin{aligned} \text{PDEs:} & \quad \partial_t \mathbf{Q} + \partial_x \mathbf{F}(\mathbf{Q}) = \mathbf{0}, \quad x \in (-\infty, \infty), \quad t > 0, \\ \text{ICs:} & \quad \mathbf{Q}(x, 0) = \begin{cases} \mathbf{Q}_L = \mathbf{Q}_i^n & \text{if } x < 0, \\ \mathbf{Q}_R = \mathbf{Q}_{i+1}^n & \text{if } x > 0. \end{cases} \end{aligned} \right\} \tag{20}$$

Figure 1, top frame, depicts the initial condition (left) and the structure of the solution (right) of (20). In this simplified case the numerical flux $\mathbf{F}_{i+\frac{1}{2}}$ in (18) results from the exact integration in (19), namely

$$\mathbf{F}_{i+\frac{1}{2}}^{\text{God}} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \mathbf{F}(\tilde{\mathbf{Q}}_{i+\frac{1}{2}}(0)) dt = \mathbf{F}(\tilde{\mathbf{Q}}_{i+\frac{1}{2}}(0)). \tag{21}$$

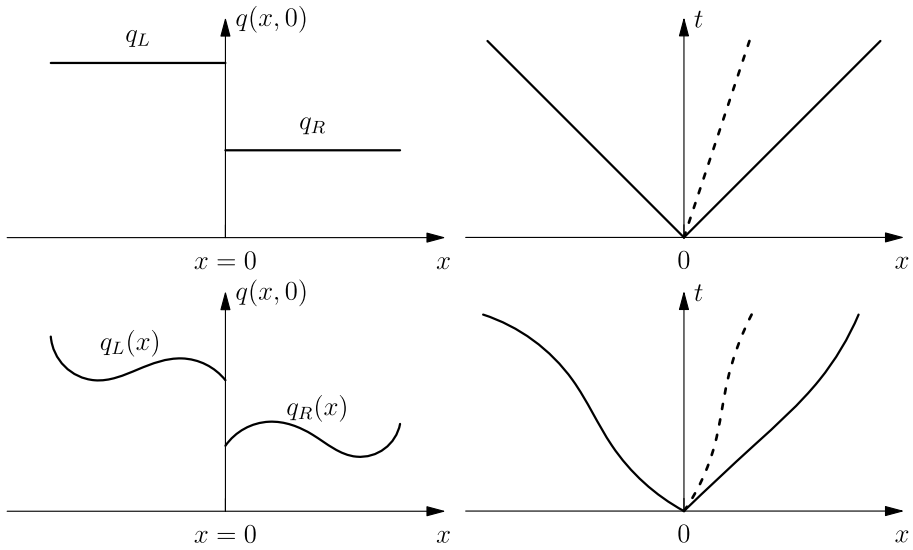


Fig. 1 Riemann problems. Top: classical Riemann problem for the first-order Godunov method. Bottom: generalised Riemann problem for the ADER method. Left-hand side pictures depict initial conditions for the classical (top) and generalized (bottom) Riemann problems, while right-hand side pictures depict the respective, emerging wave structures in the x - t plane

For exact and approximate Riemann solvers for the first-order Godunov method see [35].

The ADER extension of Godunov’s method [38] includes three steps.

- i) Replacing piece-wise constant data by piece-wise smooth data, as resulting from reconstruction procedures, for example; see Fig. 1, bottom frame. This leads naturally to the generalized Riemann problem

$$\left. \begin{aligned}
 \text{PDEs: } & \partial_t \mathbf{Q} + \partial_x \mathbf{F}(\mathbf{Q}) = \mathbf{S}(\mathbf{Q}), \quad x \in (-\infty, \infty), \quad t > 0, \\
 \text{ICs: } & \mathbf{Q}(x, 0) = \begin{cases} \mathbf{Q}_L(x) & \text{if } x < 0, \\ \mathbf{Q}_R(x) & \text{if } x > 0. \end{cases}
 \end{aligned} \right\} \tag{22}$$

- ii) Computing $\mathbf{Q}_{LR}(\tau)$, the time-dependent solution of (22) at the interface, followed by evaluation of the numerical flux

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{Q}_{LR}(\tau)) dt. \tag{23}$$

- iii) Defining a function $\mathbf{Q}_i(x, t)$ within each volume V , followed by evaluation of the numerical source

$$\mathbf{S}_i = \frac{1}{\Delta t \Delta x} \int_{t^n}^{t^{n+1}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{S}(\mathbf{Q}_i(x, t)) dx dt. \tag{24}$$

Early communications on the ADER methodology include [38] and [25]. For an introduction to ADER methods see Chaps. 19 and 20 of [35]. The next section deals with solvers for the generalised Riemann problem (22).

2.2.2 Solvers for the Generalized Riemann Problem

Several methods for solving the generalised Riemann problem are currently available. Here we limit ourselves to brief descriptions for two of them and give appropriate references for other solvers.

The Toro-Titarev solver for the generalized Riemann problem

This solver [41] seeks an approximate solution $\mathbf{Q}_{LR}(\tau)$ of (22) expressed as a truncated series expansion

$$\mathbf{Q}_{LR}(\tau) = \mathbf{Q}(0, 0_+) + \sum_{k=1}^K \left[\partial_t^{(k)} \mathbf{Q}(0, 0_+) \right] \frac{\tau^k}{k!}, \tag{25}$$

following [19]. The leading term is defined as

$$\mathbf{Q}(0, 0_+) = \lim_{t \rightarrow 0_+} \mathbf{Q}(0, t) \tag{26}$$

and is computed by solving a conventional Riemann problem, see (8). The challenge is to determine the coefficients $\partial_t^{(k)} \mathbf{Q}(0, 0_+)$ in (25) for the higher-order terms.

A precursor to the Toro-Titarev solver [41] results from a modification of the Ben-Artzi/Falcovitx second-order GRP solver [1] that departs from $\mathbf{Q}_{LR}(\tau) = \mathbf{Q}(0, 0_+) + \tau \partial_t \mathbf{Q}(0, 0_+)$, with $\mathbf{Q}(0, 0_+)$ defined as in (26). See Chap. 14 of the 1997 edition of [35]. The Cauchy-Kovalevskaya procedure is then used to determine $\partial_t \mathbf{Q}(0, 0_+)$, that is

$$\partial_t \mathbf{Q} + \partial_x \mathbf{F}(\mathbf{Q}) = \mathbf{S}(\mathbf{Q}) \rightarrow \partial_t \mathbf{Q} = -\mathbf{A}(\mathbf{Q}) \partial_x \mathbf{Q} + \mathbf{S}(\mathbf{Q}) \tag{27}$$

and the solution then reads

$$\mathbf{Q}_{LR}(\tau) = \mathbf{Q}(0, 0_+) + \tau [-\mathbf{A}(\mathbf{Q}(0, 0_+)) \partial_x \mathbf{Q}(0, 0_+) + \mathbf{S}(\mathbf{Q}(0, 0_+))]. \tag{28}$$

The pending problem in (28) is to compute the spatial derivative $\partial_x \mathbf{Q}(0, 0_+)$. First it is noted that for the linear homogeneous case

$$\partial_t \mathbf{Q} + \mathbf{A} \partial_x \mathbf{Q} = \mathbf{0} \tag{29}$$

the following evolution equations for all spatial derivatives are valid:

$$\partial_t (\partial_x^{(k)} \mathbf{Q}) + \mathbf{A} \partial_x (\partial_x^{(k)} \mathbf{Q}) = \mathbf{0}. \tag{30}$$

One can then define the classical Riemann problem for (30), with $k = 1$, to find $\partial_x \mathbf{Q}(0, 0_+)$ in (28). The numerical flux (23) then follows.

The Toro-Titarev solver for the GRP_k [35] departs from the LeFloch-Raviart expansion (25) and the Cauchy-Kovalevskaya procedure, leading to

$$\partial_t^{(k)} \mathbf{Q}(x, t) = \mathbf{P}^{(k)}(\partial_x^{(0)} \mathbf{Q}, \partial_x^{(1)} \mathbf{Q}, \dots, \partial_x^{(k)} \mathbf{Q}). \tag{31}$$

The functionals $\mathbf{P}^{(k)}$ are specific to the particular system (17); their arguments are spatial derivatives, yet to be found. To determine these, *evolution equations* are invoked

$$\partial_t(\partial_x^{(k)}\mathbf{Q}(x, t)) + \mathbf{A}(\mathbf{Q})\partial_x(\partial_x^{(k)}\mathbf{Q}(x, t)) = \mathbf{H}^{(k)}. \tag{32}$$

Then, simplified *classical Riemann problems for spatial derivatives* are posed

$$\left. \begin{aligned} \text{PDEs: } & \partial_t(\partial_x^{(k)}\mathbf{Q}(x, t)) + \mathbf{A}_{LR}^{(0)}\partial_x(\partial_x^{(k)}\mathbf{Q}(x, t)) = \mathbf{0}, \\ \text{ICs: } & \partial_x^{(k)}\mathbf{Q}(x, 0) = \begin{cases} \mathbf{Q}_L^{(k)}(0) & \text{if } x < 0, \\ \mathbf{Q}_R^{(k)}(0) & \text{if } x > 0. \end{cases} \end{aligned} \right\} \tag{33}$$

Strong simplifications have been made to the evolution equations in order to arrive at (33); the source terms $\mathbf{H}^{(k)}$ have been neglected and the advection term has been linearised around the leading term of the expansion (25). The coefficient matrix $\mathbf{A}_{LR}^{(0)}$ results from evaluating the Jacobian at the leading term of the expansion. The complete solution (25) has then been determined and the numerical flux follows from (23). A similar but simpler procedure is used to determine the source term in (24), see [41].

The Montecinos-Toro implicit GRP solver

This solver [21, 40] is implicit and can deal with stiff source terms. The key step is the following lemma.

Lemma 1 *Let $\mathbf{Q}(x, \tau)$ be an analytical function of τ . Then*

$$\mathbf{Q}(x, \tau) = \mathbf{Q}(x, 0_+) - \sum_{k=1}^{\infty} \frac{(-\tau)^k}{k!} \partial_t^{(k)}\mathbf{Q}(x, \tau). \tag{34}$$

This expression is the implicit counterpart of the explicit Raviart-LeFloch expansion (25). Now the **implicit** Cauchy-Kovalevskaya procedure leads to

$$\partial_t^{(k)}\mathbf{Q}(0, \tau) = \mathbf{P}^{(k)}(\mathbf{Q}(0, \tau), \dots, \partial_x^{(k)}\mathbf{Q}(0, \tau)). \tag{35}$$

As for the Toro-Titarev solver, a key step is to determine the arguments of functionals $\mathbf{P}^{(k)}$ in (35). For details see [40] for the general case.

Example 1 Second-order scheme. In this special case, see [21] for details, we have

$$\partial_t\mathbf{Q} + \partial_x\mathbf{F}(\mathbf{Q}) = \mathbf{S}(\mathbf{Q}), \quad \mathbf{Q}(0, \tau) = \mathbf{Q}(0, 0_+) + \tau\partial_t\mathbf{Q}(0, \tau). \tag{36}$$

The **implicit** Cauchy-Kovalevskaya procedure gives

$$\mathbf{Q}(0, \tau) = \mathbf{Q}(0, 0_+) + \tau[-\mathbf{A}(\mathbf{Q}(0, \tau))\partial_x\mathbf{Q}(0, \tau) + \mathbf{S}(\mathbf{Q}(0, \tau))]. \tag{37}$$

Then, applying the implicit expansion (34) we have

$$\partial_x\mathbf{Q}(0, \tau) = \partial_x\mathbf{Q}(0, 0_+) + \tau\partial_t(\partial_x\mathbf{Q}(0, \tau)) \tag{38}$$

and

$$\partial_x\mathbf{Q}(0, \tau) = \partial_x\mathbf{Q}(0, 0_+) + \tau\partial_x[-\mathbf{A}(\mathbf{Q}(0, \tau))\partial_x\mathbf{Q}(0, \tau) + \mathbf{S}(\mathbf{Q}(0, \tau))]. \tag{39}$$

Therefore

$$\partial_x\mathbf{Q}(0, \tau) = [\mathbf{I} - \tau\mathbf{B}(\mathbf{Q}(0, \tau))]^{-1}\partial_x\mathbf{Q}(0, 0_+). \tag{40}$$

Finally, the solution $\mathbf{Q}(0, \tau) = \mathbf{Q}^*(\tau)$ satisfies the non-linear algebraic system

$$\mathbf{Q}^* = \mathbf{Q}(0, 0_+) + \tau [-\mathbf{A}(\mathbf{Q}^*)[\mathbf{I} - \tau \mathbf{B}(\mathbf{Q}^*)]^{-1} \partial_x \mathbf{Q}(0, 0_+) + \mathbf{S}(\mathbf{Q}^*)] \quad (41)$$

and the numerical flux follows. This gives a second-order, locally implicit method suitable for balance laws with stiff source terms.

Other GRP_k solvers

We have described two schemes to solve the generalized Riemann problem. The first solver is explicit, while the second one is implicit, leading to a **locally implicit** ADER method. The implicit method can deal with stiff source terms and the corresponding ADER methods are then able to reconcile stiffness and high accuracy, for smooth solutions. These two methods seek the solution right at the interface and extend the (explicit) Ben-Artzi/Falcovitzi second-order method. Another class of solvers emerges from a re-interpretation of the high-order method of Harten et al. [13] that extends the MUSCL-Hancock second-order method. In this approach, the initial data is evolved in time and interactions at the interface are sought at selected time-integration points via classical Riemann problems [2]. The already well-established method of Dumbser et al. [4] is a numerical version of the Harten method, see also [2], whereby the data is evolved numerically via an implicit space-time discontinuous Galerkin approach. The resulting ADER scheme is suitable for stiff source terms, with the additional benefit of avoiding the Cauchy-Kovalevskaya procedure. Other recent solvers for the generalized Riemann problem are due to Götz and Iske [10], Götz and Dumbser [9], and Dematté et al. [3].

We have reviewed the ADER approach to extend Godunov's method to high order of accuracy. Very high-order methods are essential for the following reasons: (i) there is a growing trend to use PDEs to understand the physics they embody; (ii) only very accurate solutions of PDEs will achieve this and also reveal limitations of mathematical models (PDEs themselves) and uncertainty on parameters of the problem; (iii) efficiency: given an error deemed acceptable, then high-order methods attain that error, if small, much more efficiently than low-order methods on fine meshes, by orders of magnitude. For very long time evolution simulations high-order methods are essential.

In the presentation of ADER we assumed that the solution at a time t_n in any computing cell was represented by a polynomial resulting from some spatial reconstruction procedure. In principle, it can be any kind of reconstruction procedure, centred, biased, linear, non-linear. In what follows we review a non-linear reconstruction procedure.

3 Review of ENO Polynomial Reconstruction Methodology

To circumvent Godunov's theorem when constructing high-order methods, it is necessary to resort non-linear schemes and hence avoid or reduce spurious oscillations in the vicinity of large gradients of the solution. A popular method to do so is the essentially non-oscillatory (ENO) reconstruction method of Harten and collaborators [12], which we briefly review here. The presentation is succinct and we follow [17] to construct the polynomial with the required properties. More popular than ENO is the WENO method, which we also use in this paper for comparison, but a review of WENO is omitted as the reader can consult the classical references [15, 27, 28]; see also [31]. There is a variant of ENO due to Shu [26], called here m-ENO. We omit all details of m-ENO here but encourage the reader to consult the original reference.

3.1 ENO Polynomial of Arbitrary Degree

We assume a set of cell integral averages of the unknown $q(x, t)$ denoted as $\{q_j\}_{j=0}^i$ and adopt the **primitive-function** approach [17, 45] to construct an ENO polynomial with samples $q(x_{i+1/2})$ at the intercell boundaries $x_{i+1/2}$. Recall that a function $q(x)$ is the primitive function of $f(x)$, or antiderivative, if

$$f(x) = \frac{d}{dx}q(x), \quad \text{or} \quad q(x) = \int_{x_{-1/2}}^x f(y)dy, \quad (42)$$

where $x_{-1/2}$ is the left most interface. Then, the samples are defined as

$$q(x_{i+1/2}) = \sum_{j=0}^i \int_{x_{j-1/2}}^{x_{j+1/2}} f(y)dy = \sum_{j=0}^i \Delta x_j q_j. \quad (43)$$

That is, the partial sums of cell averages are the samples of the primitive function $q(x)$.

Now the task is to construct the interpolating polynomial $P(x)$ of degree $N + 1$ passing through $(x_{i+1/2}, q(x_{i+1/2}))$, of one degree higher than the sought reconstruction polynomial $p(x)$, with

$$p(x) = \frac{d}{dx}P(x). \quad (44)$$

For convenience, the final sought polynomial $p(x)$ of degree N is written in Taylor form as follows:

$$p(x) = a_0 + 2a_1(x - x_i) + \cdots + (N + 1)a_N(x - x_i)^N. \quad (45)$$

The coefficients a_0, a_1, \dots, a_N depend on the stencil of the polynomial. Next, we find the Newton divided differences. Recalling from (43)

$$q(x_{i+\frac{1}{2}}) = \sum_{j=0}^i \Delta x_j q_j \quad (46)$$

we have, for example,

$$\left. \begin{aligned} q[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] &= \frac{q(x_{i+\frac{1}{2}}) - q(x_{i-\frac{1}{2}})}{x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}} = q_i, \\ q[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}, x_{i+\frac{3}{2}}] &= \frac{q[x_{i+\frac{1}{2}}, x_{i+\frac{3}{2}}] - q[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]}{x_{i+\frac{3}{2}} - x_{i-\frac{1}{2}}} = \frac{q_{i+1} - q_i}{2\Delta x}, \\ q[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}, x_{i+\frac{3}{2}}, x_{i+\frac{5}{2}}] &= \frac{q_{i+2} - 2q_{i+1} + q_i}{6\Delta x^2}. \end{aligned} \right\} \quad (47)$$

Consider the generic cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ with intercell boundaries $x_{i-\frac{1}{2}}$ (left) and $x_{i+\frac{1}{2}}$ (right). The polynomial is built by adding one point at the time, either left or right, to the current stencil; each time one adds one degree to the sought polynomial of degree $N + 1$. The new

point added, to left or to the right, depending on the relative size of divided differences in absolute value.

The general algorithm is as follows. Start with the two points $l_0(i) = i + 1/2$ at the cell interface $x_{i+\frac{1}{2}}$ and $l_1(i) = i - \frac{1}{2}$ at the cell interface $x_{i-\frac{1}{2}}$. The most left point is $l_1(i) = i - \frac{1}{2}$. Then the next point in the sequence is $l_2(i)$. In general the new point to be added is chosen recursively as follows.

For $m = 1, \dots, N$,

$$l_{m+1}(i) = \begin{cases} l_m(i) & \text{if } |q[x_{l_m(i)}, \dots, x_{l_m(i)+m+1}]| \leq |q[x_{l_m(i)-1}, \dots, x_{l_m(i)+m}]|, \\ l_m(i) - 1 & \text{if } |q[x_{l_m(i)-1}, \dots, x_{l_m(i)+m}]| > |q[x_{l_m(i)}, \dots, x_{l_m(i)+m+1}]|. \end{cases} \tag{48}$$

The reconstruction polynomial $P(x)$ of degree $N + 1$ is

$$P(x) = \sum_{j=0}^{N+1} a_j(x - x_i)^j. \tag{49}$$

The coefficients a_j of the polynomial (45) are as follows:

$$a_j = \sum_{k=0}^{N-j+1} d_{kj} q[x_{l_{N+1}(i)}, \dots, x_{l_{N+1}(i)+j+k}], \quad j = 1, \dots, N + 1, \tag{50}$$

with

$$\left. \begin{aligned} d_{0j} &= 1, \\ d_{k0} &= (x_i - x_{l_{N+1}(i)+k-1})d_{k-1,0}, \\ d_{kj} &= d_{k,j-1} + (x_i - x_{l_{N+1}(i)+k+j-1})d_{k-1,j}. \end{aligned} \right\} \tag{51}$$

Therefore the sought polynomial (45) has been determined. For details see for example [17].

In what follows we give details for ENO polynomials of second to fourth degree.

3.2 Piecewise-Linear ENO Reconstruction

The reconstruction polynomial of degree 1 on the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ is

$$p_1(x) = a_0 + 2a_1(x - x_i) \tag{52}$$

or

$$p_1(x) = q_i + \frac{q_{j+1} - q_j}{\Delta x}(x - x_i) \tag{53}$$

on the stencils

$$S_0 = (i - 1, i), \quad S_1 = (i, i + 1). \tag{54}$$

Here

$$j = \begin{cases} i & \text{if } |q_{i+1} - q_i| \leq |q_i - q_{i-1}|, \\ i - 1 & \text{otherwise} \end{cases} \quad (55)$$

and the coefficients are

$$a_1 = \frac{q_{j+1} - q_j}{2\Delta x}, \quad a_0 = q_j + 2(x_i - x_j)a_1 = q_i. \quad (56)$$

As stated in Sect. 2, the ADER method requires the value of the polynomial, as well as its spatial derivatives, at the cell interfaces. The sought values of the polynomial at interface $x_{i+1/2}$ are

$$p_1(x_{i+1/2}) = a_0 + a_1 \Delta x, \quad \frac{d}{dx} p_1(x_{i+1/2}) = 2a_1. \quad (57)$$

3.3 Piecewise-Quadratic ENO Reconstruction

The reconstruction on the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ reads

$$p_2(x) = a_0 + 2a_1(x - x_i) + 3a_2(x - x_i)^2 \quad (58)$$

and the relevant stencils are

$$S_0 = (i - 2, i - 1, i), \quad S_1 = (i - 1, i, i + 1), \quad S_2 = (i, i + 1, i + 2). \quad (59)$$

The choice of the indexes is as follows:

$$\left. \begin{aligned} j &= \begin{cases} i & \text{if } |q_{i+1} - q_i| \leq |q_i - q_{i-1}|, \\ i - 1 & \text{otherwise.} \end{cases} \\ k &= \begin{cases} j & \text{if } |q_{j+2} - 2q_{j+1} + q_j| \leq |q_{j+1} - 2q_j + q_{j-1}|, \\ j - 1 & \text{otherwise.} \end{cases} \end{aligned} \right\} \quad (60)$$

Then

$$\left. \begin{aligned} a_2 &= \frac{q_{k+2} - 2q_{k+1} + q_k}{6\Delta x^2}, \\ a_1 &= \frac{q_{k+1} - q_k}{2\Delta x} + \frac{q_{k+2} - 2q_{k+1} + q_k}{2\Delta x} \left(i - k - \frac{1}{2} \right), \\ a_0 &= q_i - \frac{1}{24}(q_{k+2} - 2q_{k+1} + q_k). \end{aligned} \right\} \quad (61)$$

Depending on the chosen indexes we can get three different polynomials. The values at the interface are

$$\left. \begin{aligned} p_2(x_{i+1/2}) &= a_0 + a_1 \Delta x + \frac{3}{4} a_2 \Delta x^2, \\ \frac{d}{dx} p_2(x_{i+1/2}) &= 2a_1 + 3a_2 \Delta x, \\ \frac{d^2}{dx^2} p_2(x_{i+1/2}) &= 6a_2. \end{aligned} \right\} \tag{62}$$

3.4 Piecewise-Cubic ENO Reconstruction

The reconstruction on the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ is

$$p_3(x) = a_0 + 2a_1(x - x_i) + 3a_2(x - x_i)^2 + 4a_3(x - x_i)^3. \tag{63}$$

The stencils are

$$\left. \begin{aligned} S_0 &= (i - 3, i - 2, i - 1, i), & S_1 &= (i - 2, i - 1, i, i + 1), \\ S_2 &= (i - 1, i, i + 1, i + 2), & S_3 &= (i, i + 1, i + 2, i + 3). \end{aligned} \right\} \tag{64}$$

The ENO choice of indexes is as follows:

$$\left. \begin{aligned} j &= \begin{cases} i & \text{if } |q_{i+1} - q_i| \leq |q_i - q_{i-1}|, \\ i - 1 & \text{otherwise,} \end{cases} \\ k &= \begin{cases} j & \text{if } |q_{j+2} - 2q_{j+1} + q_j| \leq |q_{j+1} - 2q_j + q_{j-1}|, \\ j - 1 & \text{otherwise,} \end{cases} \\ h &= \begin{cases} j & \text{if } |q_{k+3} - 3q_{k+2} + 3q_{k+1} - q_k| \\ & \leq |q_{k+2} - 3q_{k+1} + 3q_k - q_{k-1}|, \\ j - 1 & \text{otherwise.} \end{cases} \end{aligned} \right\} \tag{65}$$

The coefficients read

$$\left. \begin{aligned} a_3 &= \frac{q_{h+3} - 3q_{h+2} + 3q_{h+1} - q_h}{24\Delta x^3}, \\ a_2 &= \frac{q_{h+2} - 2q_{h+1} + q_h}{6\Delta x^2} + \frac{q_{h+3} - 3q_{h+2} + 3q_{h+1} - q_h}{6\Delta x^2} (i - h - 1), \\ a_1 &= \frac{q_{h+1} - q_h}{2\Delta x} + \frac{q_{h+2} - 2q_{h+1} + q_h}{2\Delta x} \left(i - h - \frac{1}{2} \right) \\ &\quad + d_{22} \frac{q_{h+3} - 3q_{h+2} + 3q_{h+1} - q_h}{24\Delta x^3}, \\ a_0 &= q_i - \frac{a_2}{4} \Delta x^2. \end{aligned} \right\} \tag{66}$$

The polynomial values at the interface $x_{i+\frac{1}{2}}$ are

$$\left. \begin{aligned} p_3(x_{i+1/2}) &= a_0 + a_1 \Delta x + \frac{3}{4} a_2 \Delta x^2 + \frac{1}{2} a_3 \Delta x^3, \\ \frac{d}{dx} p_3(x_{i+1/2}) &= 2a_1 + 3a_2 \Delta x + 3a_3 \Delta x^2, \\ \frac{d^2}{dx^2} p_3(x_{i+1/2}) &= 6a_2 + 12a_3 \Delta x, \\ \frac{d^3}{dx^3} p_3(x_{i+1/2}) &= 24a_3. \end{aligned} \right\} \quad (67)$$

3.5 Piecewise-Quartic ENO Reconstruction

The reconstruction on the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ is

$$p_4(x) = a_0 + 2a_1(x - x_i) + 3a_2(x - x_i)^2 + 4a_3(x - x_i)^3 + 5a_4(x - x_i)^4. \quad (68)$$

The stencils are

$$\left. \begin{aligned} S_0 &= (i-4, i-3, i-2, i-1, i), & S_1 &= (i-3, i-2, i-1, i, i+1), \\ S_2 &= (i-2, i-1, i, i+1, i+2), & S_3 &= (i-1, i, i+1, i+2, i+3), \\ S_4 &= (i, i+1, i+2, i+3, i+4). \end{aligned} \right\} \quad (69)$$

The ENO choice of indexes is

$$\left. \begin{aligned} j &= \begin{cases} i & \text{if } |q_{i+1} - q_i| \leq |q_i - q_{i-1}|, \\ i-1 & \text{otherwise,} \end{cases} \\ k &= \begin{cases} j & \text{if } |q_{j+2} - 2q_{j+1} + q_j| \leq |q_{j+1} - 2q_j + q_{j-1}|, \\ j-1 & \text{otherwise,} \end{cases} \\ h &= \begin{cases} j & \text{if } |q_{k+3} - 3q_{k+2} + 3q_{k+1} - q_k| \\ & \leq |q_{k+2} - 3q_{k+1} + 3q_k - q_{k-1}|, \\ j-1 & \text{otherwise,} \end{cases} \\ s &= \begin{cases} h & \text{if } |q_{h+4} - 4q_{h+3} + 6q_{h+2} - 4q_{h+1} + q_h| \\ & \leq |q_{h+3} - 4q_{h+2} + 6q_{h+1} - 4q_h + q_{h-1}|, \\ h-1 & \text{otherwise.} \end{cases} \end{aligned} \right\} \quad (70)$$

To determine the coefficients we first set

$$\left. \begin{aligned}
 d_{23} &= \Delta x^2 \left[\left(i - s - \frac{1}{2} \right) \left(i - s + \frac{1}{2} \right) + 2 \left(i - s - \frac{3}{2} \right) (i - s) \right. \\
 &\quad \left. + 3 \left(i - s - \frac{5}{2} \right) \left(i - s - \frac{1}{2} \right) + 4 \left(i - s - \frac{7}{2} \right) (i - s - 1) \right], \\
 d_{22} &= \Delta x^2 \left[\left(i - s - \frac{1}{2} \right) \left(i - s + \frac{1}{2} \right) + 2 \left(i - s - \frac{3}{2} \right) (i - s) \right. \\
 &\quad \left. + 3 \left(i - s - \frac{5}{2} \right) \left(i - s - \frac{1}{2} \right) \right], \\
 d_{31} &= \Delta x^3 \left[\left(i - s - \frac{3}{2} \right) \left(i - s - \frac{1}{2} \right) \left(i - s + \frac{1}{2} \right) \right. \\
 &\quad \left. + \left(i - s - \frac{5}{2} \right) \left(i - s - \frac{1}{2} \right) \left(i - s + \frac{1}{2} \right) \right. \\
 &\quad \left. + 2 \left(i - s - \frac{5}{2} \right) \left(i - s - \frac{3}{2} \right) (i - s) \right], \\
 d_{32} &= d_{31} + \Delta x \left(i - s - \frac{7}{2} \right) d_{22}.
 \end{aligned} \right\} \tag{71}$$

Then the coefficients are

$$\left. \begin{aligned}
 a_4 &= \frac{q_{s+4} - 4q_{s+3} + 6q_{s+2} - 4q_{s+1} + q_s}{120\Delta x^4}, \\
 a_3 &= \frac{q_{s+3} - 3q_{s+2} + 3q_{s+1} - q_s}{24\Delta x^3} \\
 &\quad + 5\Delta x \left(i - s - \frac{3}{2} \right) \frac{q_{s+4} - 4q_{s+3} + 6q_{s+2} - 4q_{s+1} + q_s}{120\Delta x^4}, \\
 a_2 &= \frac{q_{s+2} - 2q_{s+1} + q_s}{6\Delta x^2} + \frac{q_{s+3} - 3q_{s+2} + 3q_{s+1} - q_s}{6\Delta x^2} (i - s - 1) \\
 &\quad + d_{23} \frac{q_{s+4} - 4q_{s+3} + 6q_{s+2} - 4q_{s+1} + q_s}{120\Delta x^4}, \\
 a_1 &= \frac{q_{s+1} - q_s}{2\Delta x} + \frac{q_{s+2} - 2q_{s+1} + q_s}{2\Delta x} \left(i - s - \frac{1}{2} \right) \\
 &\quad + d_{22} \frac{q_{s+3} - 3q_{s+2} + 3q_{s+1} - q_s}{24\Delta x^3} \\
 &\quad + d_{32} \frac{q_{s+4} - 4q_{s+3} + 6q_{s+2} - 4q_{s+1} + q_s}{120\Delta x^4}, \\
 a_0 &= q_i - \frac{a_2}{4} \Delta x^2 - \frac{a_4}{16} \Delta x^4.
 \end{aligned} \right\} \tag{72}$$

The polynomial values at interface $x_{i+\frac{1}{2}}$ are

$$\left. \begin{aligned} p_4(x_{i+\frac{1}{2}}) &= a_0 + a_1 \Delta x + \frac{3}{4} a_2 \Delta x^2 + \frac{1}{2} a_3 \Delta x^3 + \frac{5}{16} a_4 \Delta x^4, \\ \frac{d}{dx} p_4(x_{i+\frac{1}{2}}) &= 2a_1 + 3a_2 \Delta x + 3a_3 \Delta x^2 + \frac{5}{2} a_4 \Delta x^3, \\ \frac{d^2}{dx^2} p_4(x_{i+\frac{1}{2}}) &= 6a_2 + 12a_3 \Delta x + 15a_4 \Delta x^2, \\ \frac{d^3}{dx^3} p_4(x_{i+\frac{1}{2}}) &= 24a_3 + 60a_4 \Delta x, \\ \frac{d^4}{dx^4} p_4(x_{i+\frac{1}{2}}) &= 120a_4. \end{aligned} \right\} \quad (73)$$

As already pointed out, the evaluation of the polynomial and its derivatives at the cell interface $x_{i+\frac{1}{2}}$, from the left and the right, is needed for solving the generalized Riemann problem to determine the **numerical flux**. To determine the **numerical source** we must evaluate the volume integral (24), for which one makes use of the polynomial $p(x)$ and its derivatives in cell i , at nodes x_l of the integration formula used, e.g., a Gauss-Legendre quadrature.

We have reviewed the ENO method as implemented in the fully discrete high-order ADER methodology. The ENO reconstruction method has been criticised for a number of reasons. One of them is the abrupt change of stencil simply due to small changes in the divided differences that determine the coefficients of the ENO polynomial. In fact, even round-off errors may decide the choice of stencil. Shu [26] addressed this issue by proposing a modified ENO method, called m-ENO here. In the next section, we propose a simple averaged ENO-type method, in which the classical ENO polynomial is averaged with its closest neighbour. A variant m-AENO results from substituting ENO by m-ENO, for schemes of 4th and greater order of accuracy. The results are encouraging.

4 Novel Polynomial Reconstruction: AENO

In this section, we present a non-linear polynomial reconstruction procedure that is akin to both the existing ENO and WENO procedures [12–15, 27, 28]. We first present an example that motivates this section.

4.1 A Motivating Example: Second-Order ADER Method

We begin by considering the simplest, first-degree, polynomial reconstruction

$$p_i(x) = q_i^n + (x - x_i) \Delta_i \quad (74)$$

within the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, at time $t = t_n$. Here $x_i = \frac{1}{2}(x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}})$ is the cell centre and Δ_i is the slope of $p_i(x)$, still to be determined. Figure 2 depicts two polynomials of the form (74) and the results are respectively determined by choosing the slope thus

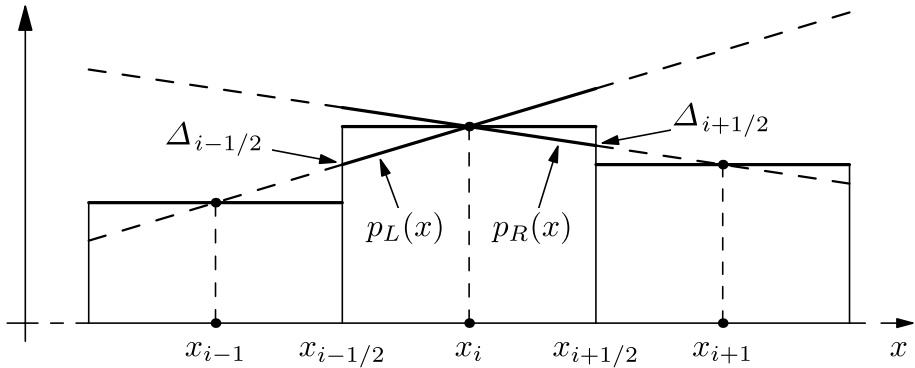


Fig. 2 Two possible first-degree reconstruction polynomials $p_L(x)$ (left) and $p_R(x)$ (right) for cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$. These result from choosing the slope as in (75)

$$\Delta_i \equiv \Delta_{i-\frac{1}{2}} = \frac{q_i^n - q_{i-1}^n}{\Delta x} ; \quad \Delta_i \equiv \Delta_{i+\frac{1}{2}} = \frac{q_{i+1}^n - q_i^n}{\Delta x}. \tag{75}$$

Assume that the slope Δ_i is chosen as the weighted average

$$\Delta_i = \frac{1}{2}(1 + \omega)\Delta_{i-\frac{1}{2}} + \frac{1}{2}(1 - \omega)\Delta_{i+\frac{1}{2}}, \quad \omega \in [-1, 1]. \tag{76}$$

It is easy to show that the second-order ADER method with the linear reconstruction (74) reproduces the classical second-order schemes of Lax-Wendroff, Warming-Beam and Fromm if ω in (76) is chosen as follows:

$$\omega = \begin{cases} -1 & : \text{Lax-Wendroff method,} \\ 0 & : \text{Fromm method,} \\ 1 & : \text{Warming-Beam method.} \end{cases} \tag{77}$$

The classical ENO [12] second-order method chooses

$$\Delta_i = \begin{cases} \Delta_{i-\frac{1}{2}} & \text{if } |\Delta_{i-\frac{1}{2}}| \leq |\Delta_{i+\frac{1}{2}}|, \\ \Delta_{i+\frac{1}{2}} & \text{if } |\Delta_{i-\frac{1}{2}}| > |\Delta_{i+\frac{1}{2}}|. \end{cases} \tag{78}$$

In other words, the classical ENO second-order method switches non-linearly between the Lax-Wendroff and the Warming-Beam methods, depending on the relative sizes in the absolute value of the slopes $\Delta_{i-\frac{1}{2}}$ and $\Delta_{i+\frac{1}{2}}$.

A simple third-order method. It is interesting to note that, still in the frame of the second-order ADER method, if the weight parameter ω is chosen as

$$\omega = \frac{1}{3}(2c - \text{sign}(c)), \tag{79}$$

then, a third-order accurate method results, in both space and time. Here c is the Courant number. This scheme was first presented in [33], see Eq. (13.39), and also in [37],

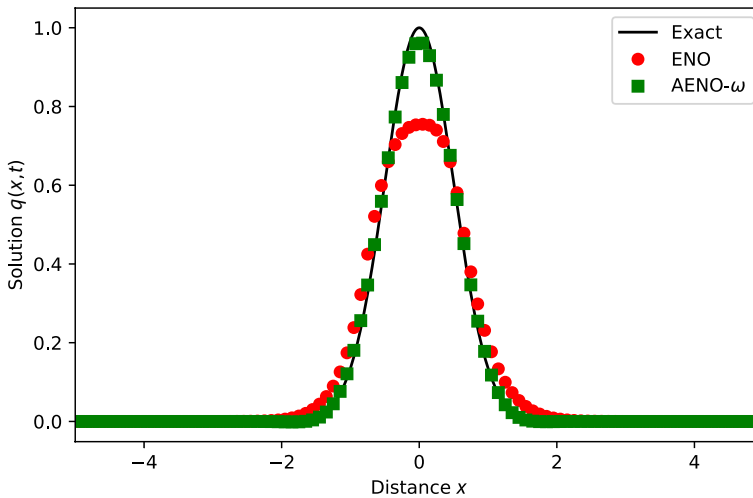


Fig. 3 Numerical results for the linear advection equation for a Gaussian profile test, comparing ADER2 with ENO (circles) and ADER2 with AENO slope (76) (squares), with $\omega = \frac{1}{3}(2c - \text{sign}(c))$. $T_{\text{out}} = 100$ units, mesh $M = 100$ cells and $C_{\text{eff}} = 0.9$

Table 1 Empirical convergence rates for ADER2 with the particular choice of slope averaging $\omega = \frac{1}{3}(2c - \text{sign}(c))$ in (79)

| M | L_1 error | L_1 order | L_2 error | L_2 order | L_∞ error | L_∞ order |
|-----|-------------|-------------|-------------|-------------|------------------|------------------|
| 25 | 1.7E-02 | – | 3.5E-02 | – | 1.2E-01 | – |
| 50 | 3.5E-03 | 2.3 | 7.4E-03 | 2.2 | 2.7E-02 | 2.1 |
| 100 | 4.8E-04 | 2.9 | 1.0E-03 | 2.8 | 4.0E-03 | 2.8 |
| 200 | 6.2E-05 | 3.0 | 1.3E-04 | 3.0 | 5.3E-04 | 2.9 |
| 400 | 7.7E-06 | 3.0 | 1.7E-05 | 3.0 | 6.6E-05 | 3.0 |

in the setting of the MUSCL-Hancock method [44]. However, it has been shown that for the homogeneous case, the MUSCL-Hancock method is entirely equivalent to the ADER method in second-order mode (ADER2), see [20]. We have again verified this in the setting of ADER methods. The details of the analysis are omitted.

Figure 3 displays computed results for the linear advection equation, for which a Gaussian profile is used as initial condition, where comparison is made with ADER2 in conjunction with ENO reconstruction (74), (75) and (78). Table 1 shows empirical convergence rates for the linear advection equation in which it is verified that the ADER2 (formally second-order) scheme with reconstruction (74) with (76) and ω given by (79) is third-order accurate in both space and time.

The AENO reconstruction method presented in this paper, in fact emerges from averaging of two polynomials in the frame of the ENO scheme to choose the reconstruction stencil. The construction of ADER schemes of up to fifth order in space and time, based on averages of the type (76) were constructed by MSc student Andrea Santacá in his Master thesis [24]. Additionally, in the present paper we include a variant of AENO, called m-AENO, to be explained later.

4.2 A Joining Function for Averaging Two Polynomials

Motivated by the example of the previous Sect. 4.1 we seek an averaging procedure that can be applied to two polynomials of arbitrary degree. First we define the function

$$\omega(x) = \frac{-x}{\sqrt{\epsilon^2 + x^2}}, \quad x \in \mathcal{R}, \tag{80}$$

and $\epsilon \in \mathcal{R}$, a constant, with $|\epsilon| < 1$. Consider now the *averaging function* $J(\omega(x))$ joining smoothly two constant, real states q_L and q_R , as follows:

$$J(\omega(x)) = \frac{1}{2}[1 + \omega(x)]q_L + \frac{1}{2}[1 - \omega(x)]q_R. \tag{81}$$

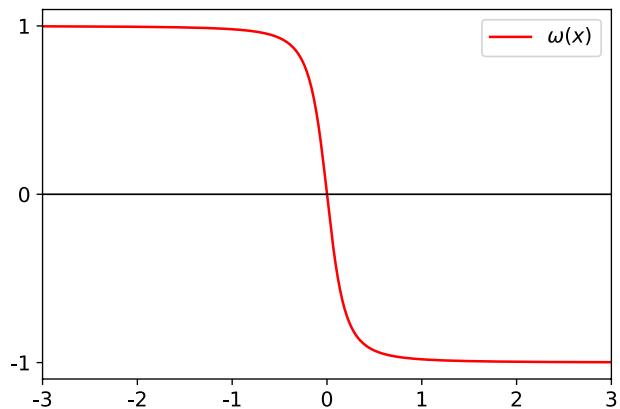
Note the analogy between the weighted average (76) and (81). While in (76) the weight ω is constant, in (81) $\omega(x)$ is obviously variable.

Figure 4 shows the weight function $\omega(x)$ for $\epsilon^2 = 0.2$. For $x < 0$ the function $\omega(x)$ approaches 1 rapidly, while for $x > 0$, $\omega(x)$ approaches -1 . Note the following properties of $\omega(x)$:

$$\begin{cases} \omega(0) = 0, \\ \omega(-\epsilon) = \frac{1}{2}\sqrt{2} \approx 0.72, \\ \omega(+\epsilon) = -\frac{1}{2}\sqrt{2} \approx -0.72. \end{cases} \tag{82}$$

Note also the associated properties of $J(\omega(x))$:

Fig. 4 Weight function $\omega(x)$ defined by (80) for $\epsilon^2 = 0.2$



$$\begin{cases} J(\omega(0)) = \frac{1}{2}(q_L + q_R), \\ J(\omega(-\epsilon)) = \frac{1}{2}\left(1 + \frac{1}{2}\sqrt{2}\right)q_L + \frac{1}{2}\left(1 - \frac{1}{2}\sqrt{2}\right)q_R, \\ J(\omega(+\epsilon)) = \frac{1}{2}\left(1 - \frac{1}{2}\sqrt{2}\right)q_L + \frac{1}{2}\left(1 + \frac{1}{2}\sqrt{2}\right)q_R. \end{cases} \tag{83}$$

Figure 5 shows the distribution of $J(\omega(x))$ for $q_L = 3, q_R = -2$ and $\epsilon^2 = 0.2$. It is seen that $J(\omega(x))$ approaches $q_L = 3$ for $x < 0$, while $J(\omega(x))$ tends to $q_R = -2$ for $x > 0$. Note that $J(\omega(x))$ is monotone, as

$$J'(\omega(x)) = \frac{1}{2}(q_L - q_R)\omega'(x) \begin{cases} < 0 & \text{if } q_L > q_R, \\ > 0 & \text{if } q_L < q_R \end{cases} \tag{84}$$

and

$$\omega'(x) = \frac{-\epsilon^2}{(\epsilon^2 + x^2)^{\frac{3}{2}}} < 0, \forall x. \tag{85}$$

Note also that $J'(x) \rightarrow 0$ as $|\omega(x)| \rightarrow \infty$. That is the constant states q_L and q_R are approached asymptotically by $J(x)$. However we note that these states are approached very rapidly. Consider a positive integer N , then

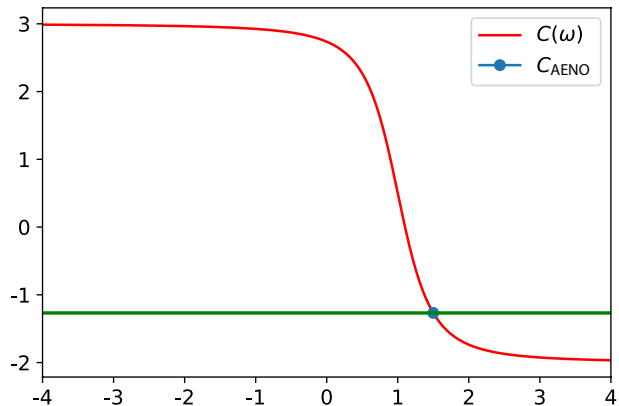
$$\omega(N\epsilon) = \frac{-N\epsilon}{\sqrt{\epsilon^2 + N^2\epsilon^2}} = -\frac{N}{\sqrt{1 + N^2}}\text{sign}(\epsilon). \tag{86}$$

For large N ,

$$\omega(N\epsilon) \approx -\text{sign}(\epsilon) = -1. \tag{87}$$

For example, for $N = 10, \frac{N}{\sqrt{1+N^2}} \approx 0.995$. That is, at a distance of 10ϵ from the origin, $J(x)$ attains the constant states with an error of less than 1%. Finally, we note that the use of ϵ^2 in

Fig. 5 AENO coefficient function $C(\omega)$ derived from the joining function J , for coefficients $C_L = 3$ (left) and $C_R = -2$ (right) and $\epsilon^2 = 0.5$. The resulting AENO coefficient C_{AENO} is closer to the min



the definition of $\omega(x)$ is convenient, as it permits a precise evaluation of $J(x)$ at a distance ϵ from the origin. In particular applications, a translation might be necessary. In this paper we use $J(x)$ for averaging two polynomials, through the averaging of their coefficients.

When applying the above averaging function (81) to two states q_L and q_R we proceed as follows. First we define

$$s = \frac{|q_L|}{|q_R| + \text{TOL}}, \tag{88}$$

where TOL is a small positive quantity chosen to avoid division by zero, e.g., $\text{TOL} = 10^{-6}$. Then we redefine

$$\omega(s) = \frac{-(s - 1)}{\sqrt{\epsilon^2 + (s - 1)^2}}. \tag{89}$$

It is easy to see that for the joining function (81) we have

$$\left. \begin{aligned} |q_L| \leq |q_R| &\iff 0 \leq s \leq 1 \implies \frac{1}{2}[1 + \omega(s)]q_L \text{ is given greater weight,} \\ |q_L| \geq |q_R| &\iff s \geq 1 \implies \frac{1}{2}[1 - \omega(s)]q_R \text{ is given greater weight.} \end{aligned} \right\} \tag{90}$$

In other words, in the joining function (81) acting on two states q_L and q_R , the smaller argument in absolute value receives the larger weight and the larger argument in absolute value receives the smaller weight. This averaging procedure will be applied to two polynomials, the ENO polynomial and that closest to it, by applying it to each pair of respective coefficients. The coefficients of the ENO polynomial will receive the larger weight, as we shall see in the next section.

4.3 AENO: a Two-Polynomial Average Method

In Sect. 3 we reviewed the ENO spatial reconstruction method. In Sect. 4.1 we showed that averaging of the ENO polynomial with its closest neighbour is capable of producing encouraging results. In particular in Sect. 4.1, we showed that for the formally second-order ADER method there is an averaging for which such formally second-order ADER method is actually third-order accurate in both space and time.

In this section, we propose a reconstruction method based on a two-polynomial averaging procedure. Two schemes are proposed. Scheme 1, called AENO, results from a weighted average of the classical ENO polynomial and its closest neighbour. Scheme 2, called m-AENO, follows a similar approach and results from a weighted average of the classical m-ENO polynomial of Shu [26] and its closest neighbour. We first describe AENO.

Here we introduce a general averaging procedure as applied to two *neighbouring* polynomials of arbitrary degree. Assume an N th degree polynomial $p(x)$ on cell $[x_{i-1/2}, x_{i+1/2}]$ of the form

$$p(x) = a_0 + 2a_1(x - x_i) + \dots + (N + 1)a_N(x - x_i)^N \tag{91}$$

that has been obtained by differentiating a polynomial $P(x)$ of degree $N + 1$. By redefining the coefficients as

$$\hat{a}_1 = 2a_1, \dots, \hat{a}_N = (N + 1)a_N \tag{92}$$

we write

$$p(x) = a_0 + \hat{a}_1(x - x_i) + \dots + \hat{a}_N(x - x_i)^N. \tag{93}$$

For the sake of simplicity, in what follows the hat is omitted.

Recall that with a reconstruction polynomial of degree N it is possible to construct an ADER numerical method of order $(N + 1)$ in both space and time. Then in the context of the ENO method, we need $N + 1$ stencils and following the ENO approach we find a single, best polynomial, the ENO polynomial. In the AENO approach, instead of choosing just the single ENO polynomial we consider another polynomial, the **closest** to the ENO polynomial and take an average of these two. Moreover, the search of the stencils for the two polynomials is commanded by the conventional ENO approach and the coefficients of the ENO polynomial take the largest weight.

Example 2 In the case of a third-order method we just need second-degree polynomials, for which we have three possible candidates, namely

$$\left. \begin{aligned} p_L(x) &= a_{0L} + a_{1L}(x - x_i) + a_{2L}(x - x_i)^2, \\ p_M(x) &= a_{0M} + a_{1M}(x - x_i) + a_{2M}(x - x_i)^2, \\ p_R(x) &= a_{0R} + a_{1R}(x - x_i) + a_{2R}(x - x_i)^2. \end{aligned} \right\} \tag{94}$$

In the ENO context we choose one of these polynomials according to the ENO algorithm presented earlier. In this example, we need two stages. In the AENO case we perform just one step of the algorithm; basically, we choose the direction based on the first-order slope, which is either left or right. If the direction is left then we take an average of the coefficients of $p_L(x)$ and $p_M(x)$; otherwise we take an average of the coefficients of $p_M(x)$ and $p_R(x)$. Suppose the direction is the left one, then the sought polynomial is

$$\tilde{p}(x) = \tilde{a}_0 + \tilde{a}_1(x - x_i) + \tilde{a}_2(x - x_i)^2, \tag{95}$$

with

$$\left. \begin{aligned} \tilde{a}_2 &= \frac{1}{2}(1 + \omega)a_{2L} + \frac{1}{2}(1 - \omega)a_{2M}, \\ \omega = \omega(s) &= \frac{1 - s}{\sqrt{\epsilon^2 + (1 - s)^2}}, \quad s = \frac{|a_{2L}|}{|a_{2M}| + \text{TOL}} \end{aligned} \right\} \tag{96}$$

and

$$\left. \begin{aligned} \tilde{a}_1 &= \frac{1}{2}(1 + \omega)a_{1L} + \frac{1}{2}(1 - \omega)a_{1M}, \\ \omega &= \frac{1 - s}{\sqrt{\epsilon^2 + (1 - s)^2}}, \quad s = \frac{|a_{1L}|}{|a_{1M}| + \text{TOL}}. \end{aligned} \right\} \tag{97}$$

The coefficient \tilde{a}_0 is chosen so as to comply with the conservation property, that is

$$\tilde{a}_0 = q_i^n - \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} [\tilde{a}_1(x - x_i) + \tilde{a}_2(x - x_i)^2] dx. \tag{98}$$

The parameter ϵ^2 is a positive constant, while TOL is a small positive tolerance to avoid division by zero. In practice we take $\text{TOL} = 10^{-6}$.

Generalization: The AENO procedure is then generalized as follows.

- **Step 1 ENO-determined stencil.** Perform $N - 1$ steps of the algorithm to choose the stencils: for $m = 1, \dots, N - 1$,

$$l_{m+1}(i) = \begin{cases} l_m(i) & \text{if } |q[x_{l_m(i)}, \dots, x_{l_m(i)+m+1}]| \\ & \leq |q[x_{l_m(i)-1}, \dots, x_{l_m(i)+m}]|, \\ l_m(i) - 1 & \text{otherwise.} \end{cases} \tag{99}$$

This step results in two polynomials $p_L(x)$ and $p_R(x)$, with coefficients a_{kL} and a_{kR} , respectively.

- **Step 2 averaging.** Take the average of the coefficients of the polynomials $p_L(x)$ and $p_R(x)$ to obtain the coefficients of the AENO polynomial, namely

$$\tilde{a}_k = \frac{1}{2}(1 + \omega)a_{kL} + \frac{1}{2}(1 - \omega)a_{kR}, \tag{100}$$

with

$$\omega = \frac{1 - s}{\sqrt{\epsilon^2 + (1 - s)^2}}, \quad s = \frac{|a_{kL}|}{|a_{kR}| + \text{TOL}} \tag{101}$$

and

$$\tilde{a}_0 = q_i^n - \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} (\tilde{a}_1(x - x_i) + \dots + \tilde{a}_N(x - x_i)^N) dx. \tag{102}$$

- **Step 3 the AENO polynomial.** The resulting polynomial is

$$\tilde{p}(x) = \tilde{a}_0 + \tilde{a}_1(x - x_i) + \dots + \tilde{a}_N(x - x_i)^N. \tag{103}$$

The m-AENO scheme follows a similar approach to the AENO scheme presented above. The difference is that, instead of relying on the classical ENO polynomial we rely on the classical m-ENO polynomial of Shu [26]. Then the m-AENO scheme results from a weighted averaged of m-ENO polynomial and its closest neighbour. We omit the details. For background of the m-ENO polynomial the reader is referred to the original paper of Shu [26].

In the next section, we carry out a thorough assessment of the newly proposed spatial reconstruction procedure, as applied to the linear advection equation.

5 Results for the Linear Advection Equation

Here we assess the performance of ADER with the newly proposed **AENO** reconstruction procedure. For the assessment we compare numerical solutions against the exact solutions and against numerical solutions with ENO, m-ENO and WENO, also in conjunction with ADER. To this end we consider four test problems, namely IVP1, IVP2, IVP3 and IVP4 given below

$$\left. \begin{aligned} \text{PDE: } & \partial_t q + \lambda \partial_x q = 0, x \in [0, 30], \\ \text{IC: } & q(x, 0) = h(x) = \sin\left(\frac{2}{5}\pi x\right). \end{aligned} \right\} \text{ IVP1} \tag{104}$$

$$\left. \begin{aligned} \text{PDE: } & \partial_t q + \lambda \partial_x q = 0, x \in [0, 1], \\ \text{IC: } & q(x, 0) = h(x) = \sin(2\pi x). \end{aligned} \right\} \text{ IVP2} \tag{105}$$

$$\left. \begin{aligned} \text{PDE: } & \partial_t q + \lambda \partial_x q = 0, x \in [0, 2], \\ \text{IC: } & q(x, 0) = h(x) = \sin^4(\pi x). \end{aligned} \right\} \text{ IVP3} \tag{106}$$

$$\left. \begin{aligned} \text{PDE: } & \partial_t q + \lambda \partial_x q = 0, x \in [-1, 1], \\ \text{IC: } & q(x, 0) = h(x) = \begin{cases} \exp\left(\frac{\ln 2(x+0.7)^2}{0.0009}\right), & -0.8 \leq x \leq -0.6, \\ 1, & -0.4 \leq x \leq -0.2, \\ 1 - |10x - 1|, & 0.0 \leq x \leq 0.2, \\ (1 - 100(x - 0.5)^2)^{1/2}, & 0.4 \leq x \leq 0.6, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \right\} \text{ IVP4} \tag{107}$$

The exact solution in each IVP is given as

$$q(x, t) = h(x - \lambda t). \tag{108}$$

5.1 Solution Profiles for Long-Time Evolution

The purpose of this section is to illustrate the performance of the new schemes for linear IVPs for long-time evolution, which is known to be particularly challenging for low accuracy schemes. We compare results to exact solutions and to numerical solutions obtained with established reconstruction methods, namely ENO, m-ENO and WENO. To this end, we use IVP1 with smooth initial conditions and IVP4 with a combination of smooth and discontinuous parts. The second problem IVP4 is a more realistic representation of

practical problems for hyperbolic equations that involve both smooth parts as well as discontinuities.

Figures 6, 7, 8, 9 and 10 show computed results (symbols) for IVP1 (104) compared to the exact solution (line) for the ADER high-order scheme with various reconstruction methods, from 2nd to 5th orders of accuracy in both space and time. For the new methods AENO and m-AENO we used $\epsilon^2 = \frac{1}{2}$. Obviously, for all methods, increasing the order of accuracy progressively improves the agreement between the numerical solution and the exact solution. The reconstruction method used has a visible effect on the computed solution; this is patently obvious for 2nd and 3rd orders of accuracy but it is still visible for higher orders of accuracy. Long-time evolution also contributes to differentiate between methods in terms of accuracy. Judging from the figures, AENO is clearly more accurate than both ENO and m-ENO; this is especially clear for 2nd, 3rd and 4th orders of accuracy. In fact, the new AENO schemes of this paper compare well with the sophisticated WENO; it is even reasonable to state that the new AENO schemes have a small advantage over WENO; compare carefully the profiles for 3rd and 4th orders of accuracy. For second order, WENO is superior to all other schemes.

Figures 11, 12, 13, 14 and 15 show computed results for the multiple wave test IVP4 for a long evolution time $T_{\text{out}} = 2000$ units and a coarse mesh of just $M = 100$ cells. This is an exceedingly demanding test problem, as it contains the conflicting requirements of high accuracy for smooth parts and high resolution without spurious oscillations in the vicinity of discontinuities. Figure 11 shows results for ENO in conjunction with ADER for schemes of 2nd to 5th order in space and time. Clearly, for the long chosen output time and the coarse mesh used, even the 5th order scheme gives large errors; even higher order of accuracy would be required to obtain acceptable results. The challenges of this test are representative of practical computational problems involving long time evolution, such as in acoustics, seismic waves and tsunamic waves. Figure 12 shows results for m-ENO in conjunction with ADER for schemes of 2nd to 5th order in space and time. Results are comparable to those of ENO in Fig. 11.

Figure 13 shows results for the new AENO scheme. Compared to the classical ENO and m-ENO shown in Figs. 11 and 12, the new AENO scheme is clearly superior; this is more evident for 3rd and 4th orders of accuracy. Surprisingly, this observation is also true for WENO, comparing Figs. 13 and 15. Only for the 5th order case, AENO and WENO are, roughly, comparable; WENO displays some more visible undershoots and a spurious oscillation on the right-hand side. Results for the second version of our averaged ENO, m-AENO, are shown in Fig. 14. For 2nd and 3rd order accuracy, these results are identical to those of AENO. In fact, the schemes ENO and m-AENO are identical in these two cases. For 4th and 5th order cases the performance of m-AENO is visibly inferior to that of AENO. In fact, m-AENO is also inferior to the classical ENO, m-ENO and WENO schemes for this test problem and for the 4th and 5th order cases. Table 2 shows the errors in the computed solutions for the multi-wave test problem.

For the multi-wave test IVP4, it is the discontinuous components of the profile that pose the most severe challenge to the numerical methods. This is a recurrent feature of high-order methods for hyperbolic equations. To highlight this feature, we carried out computations by keeping the square wave only, in the initial condition of IVP4. Results are shown in Figs. 16 and 17. Figure 16 shows four frames corresponding to ADER schemes of orders 2, 3, 4 and 5. For each order we plot results for all reconstruction

Fig. 6 IVP1: long-time evolution test for IVP1 (104). ADER scheme with ENO reconstruction for the linear advection equation with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 500$ and $C_{\text{eff}} = 0.9$

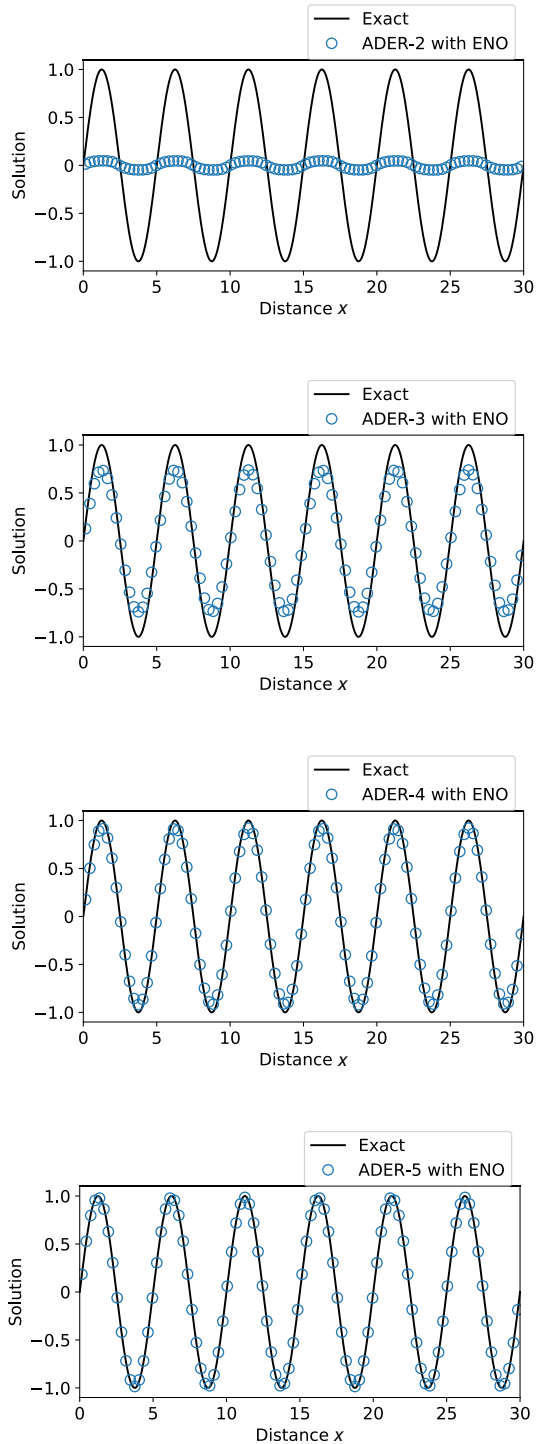


Fig. 7 IVP1: long-time evolution test for IVP1 (104). ADER scheme with m-ENO reconstruction for the linear advection equation with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 500$ and $C_{\text{eff}} = 0.9$

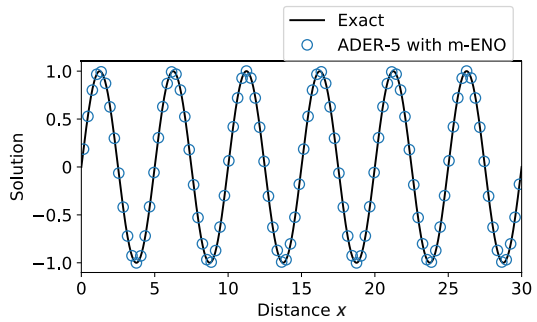
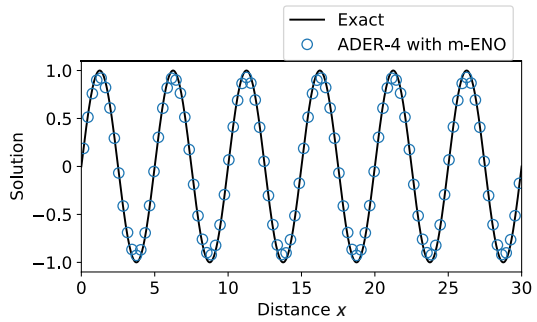
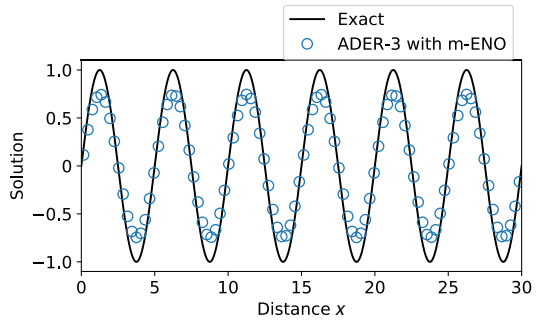
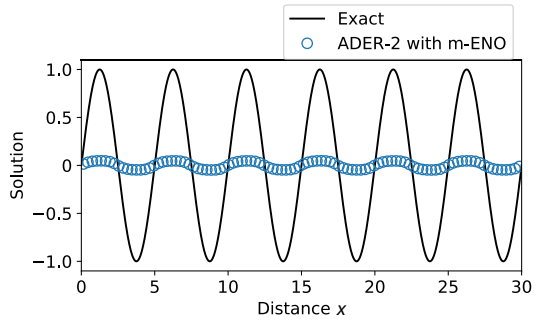


Fig. 8 IVP1: long-time evolution test for IVP1 (104). ADER scheme with AENO reconstruction for the linear advection equation with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 500$ and $C_{\text{eff}} = 0.9$

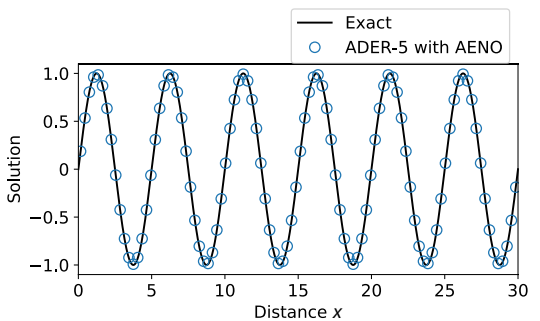
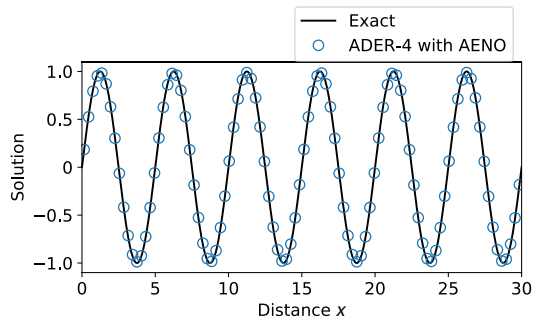
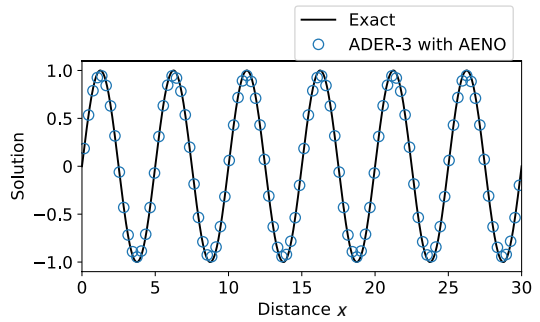
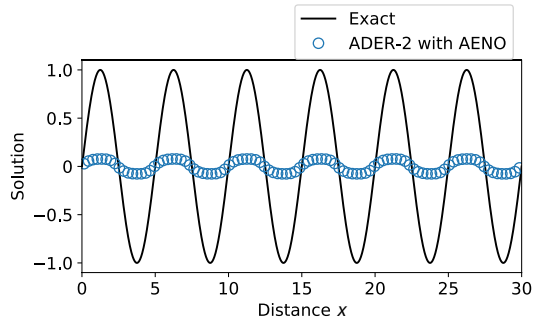


Fig. 9 IVP1: long-time evolution test for IVP1 (104). ADER scheme with m-AENO reconstruction for the linear advection equation with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 500$ and $C_{\text{eff}} = 0.9$

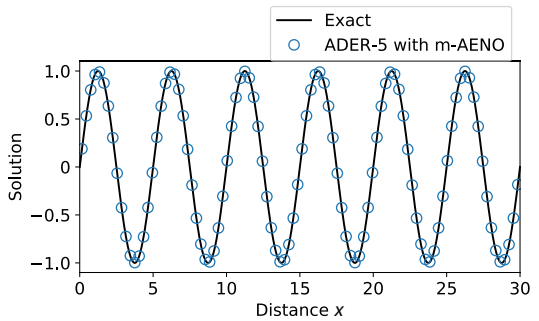
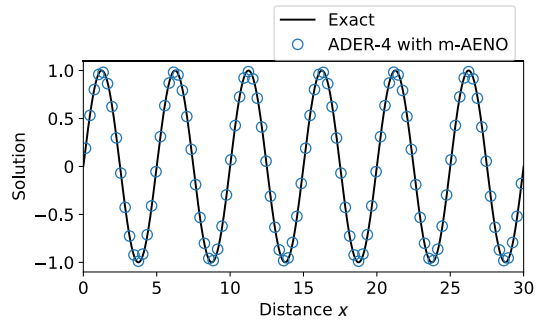
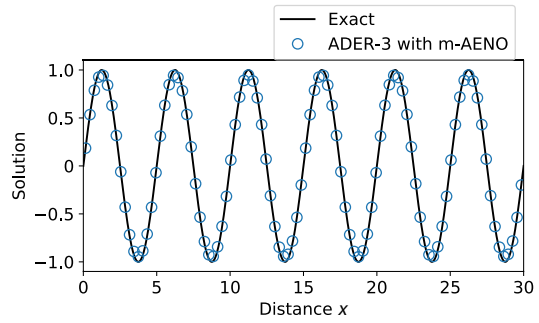
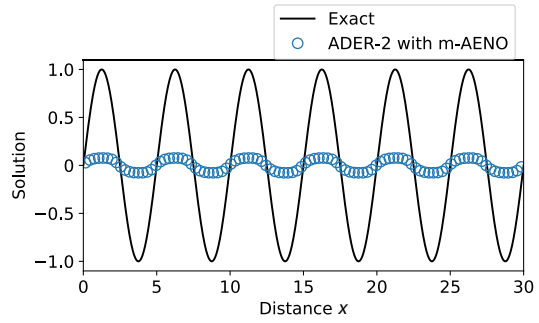


Fig. 10 IVP1: long-time evolution test for IVP1 (104). ADER scheme with WENO reconstruction for the linear advection equation with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 500$ and $C_{\text{eff}} = 0.9$

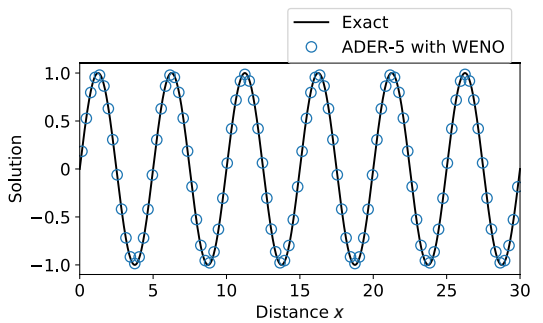
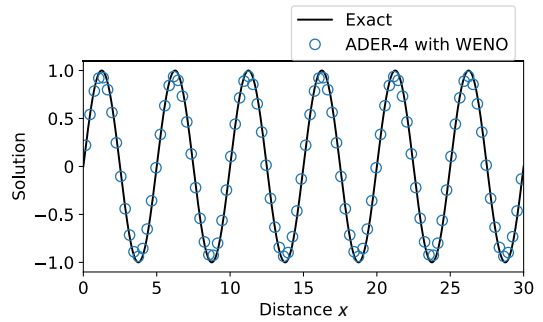
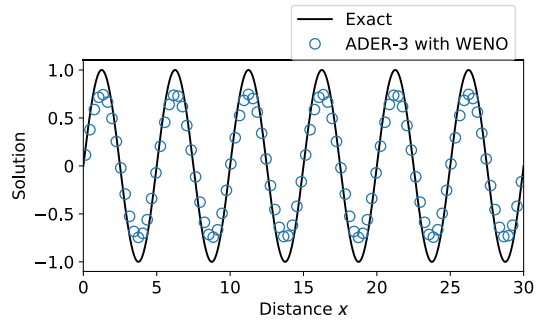
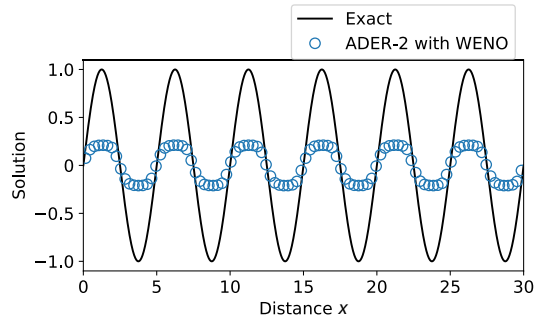


Fig. 11 ENO results for IVP4 (107): ADER scheme with the ENO reconstruction for the linear advection with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 2000$ and $C_{\text{eff}} = 0.9$

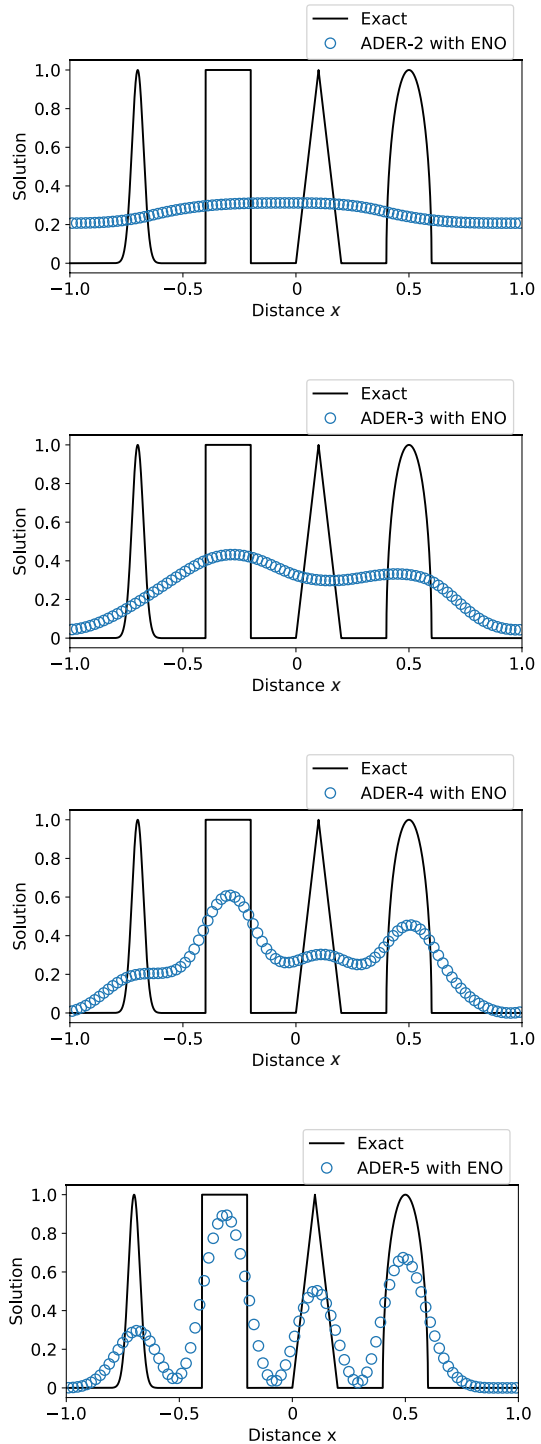


Fig. 12 m-ENO results for IVP4 (107): ADER scheme with the m-ENO reconstruction for the linear advection with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 2000$ and $C_{\text{eff}} = 0.9$

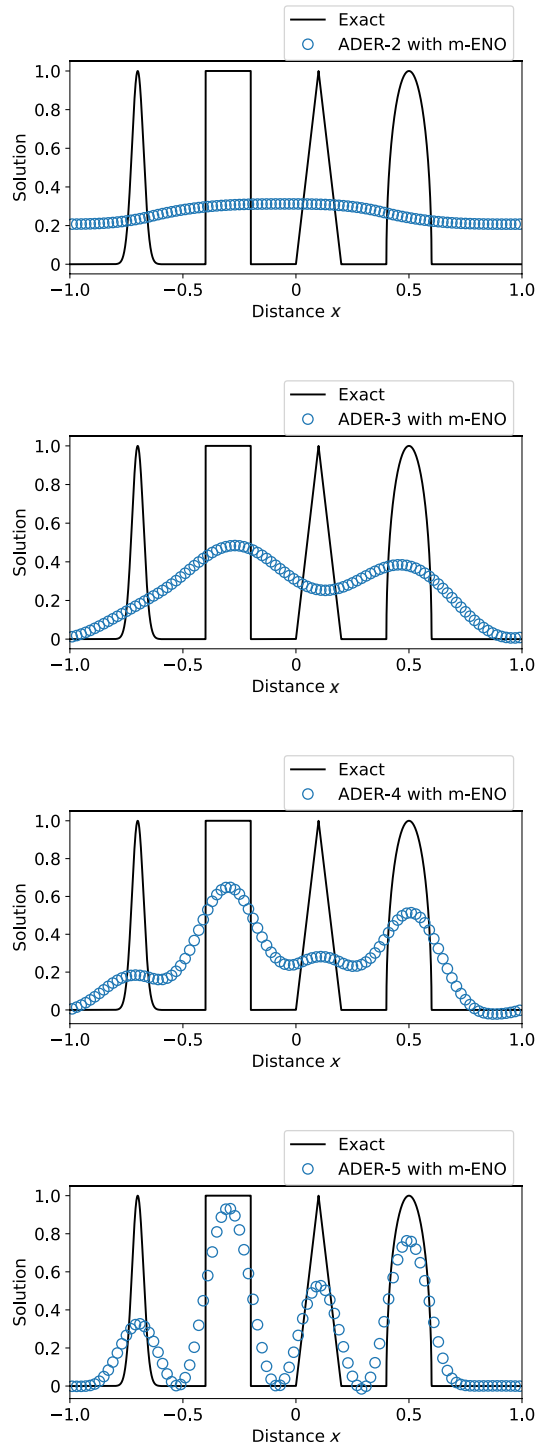


Fig. 13 AENO results for IVP4 (107): ADER scheme with the AENO reconstruction for the linear advection with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 2000$ and $C_{\text{eff}} = 0.9$

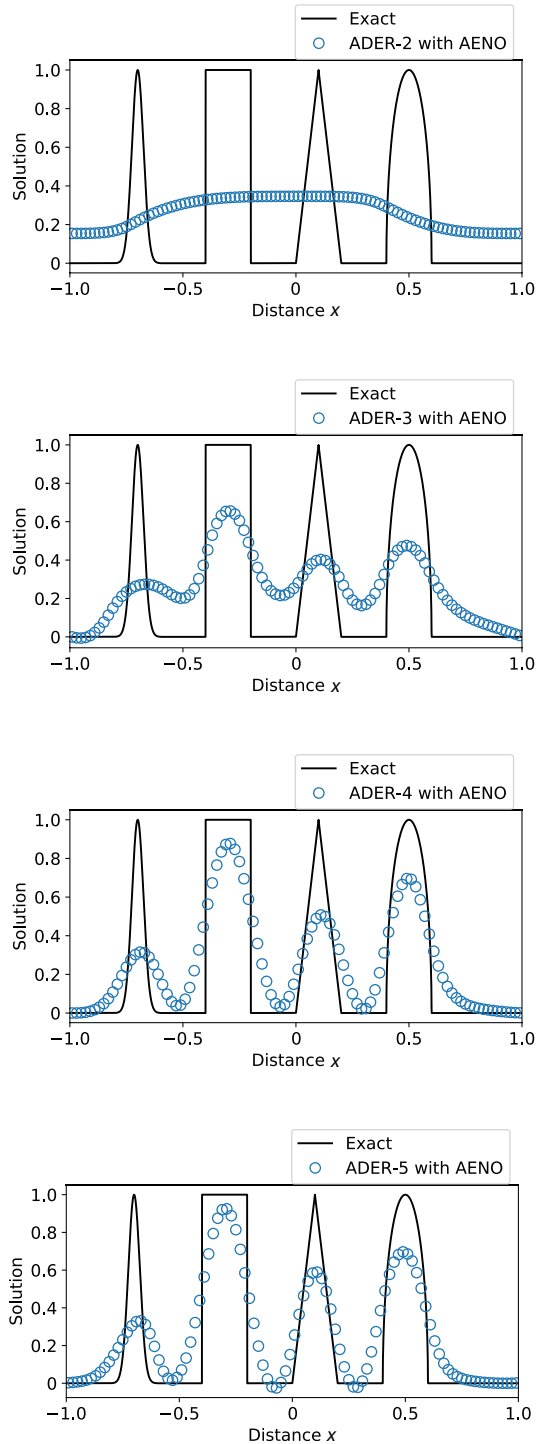


Fig. 14 m-AENO results for IVP4 (107): ADER scheme with the m-AENO reconstruction for the linear advection with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 2000$ and $C_{\text{eff}} = 0.9$

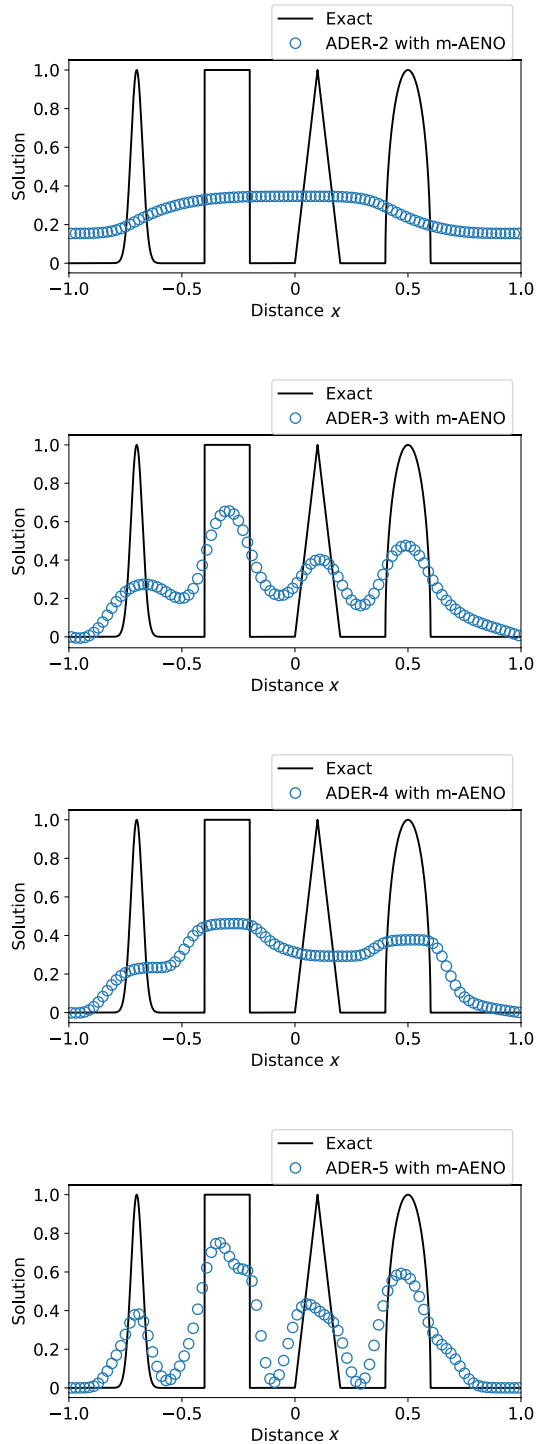
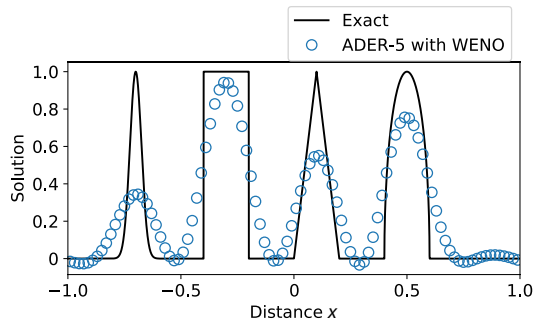
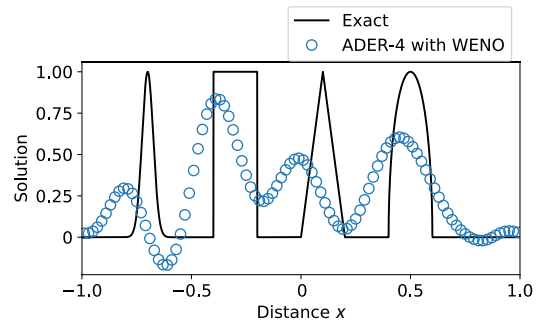
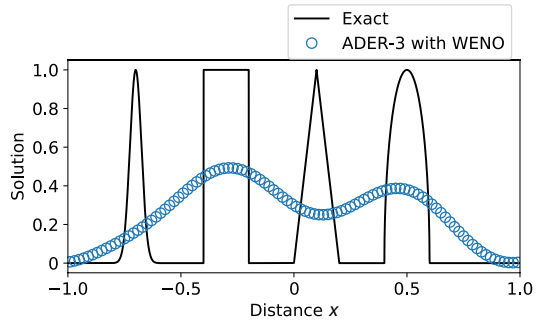
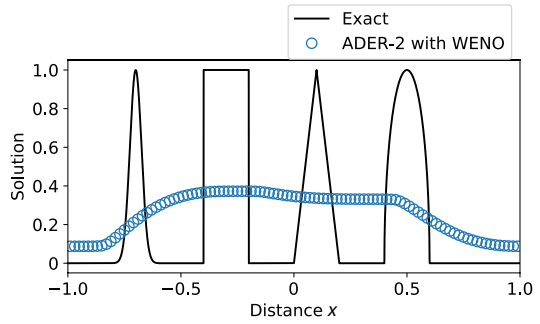


Fig. 15 WENO results for IVP4 (107): ADER scheme with the WENO reconstruction for the linear advection with $\lambda = 1$, $M = 100$, $T_{\text{out}} = 2000$ and $C_{\text{eff}} = 0.9$



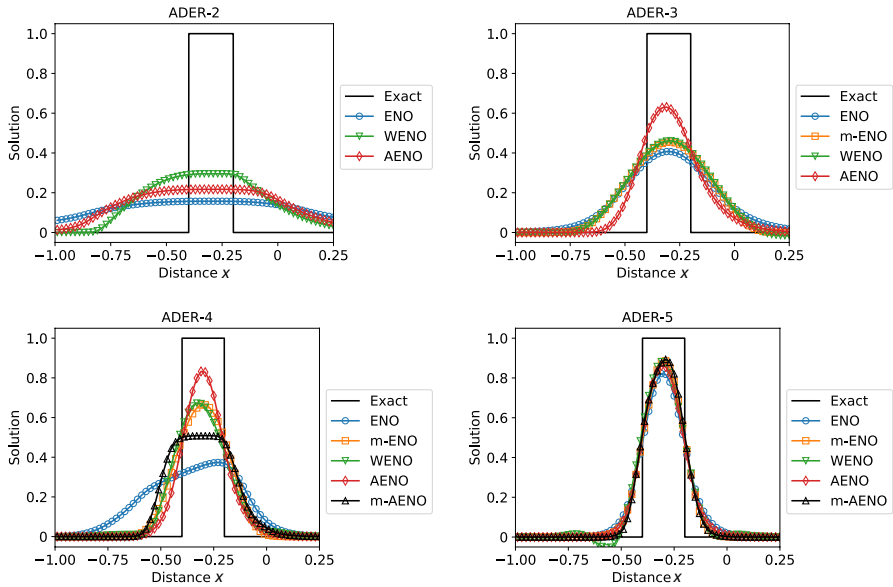


Fig. 16 Square wave results: ADER schemes from 2nd to 5th order with ENO, m-ENO, WENO, AENO and m-AENO reconstructions for the linear advection with $\lambda = 1$, $M = 100$, $T_{out} = 2\,000$ and $C_{eff} = 0.9$

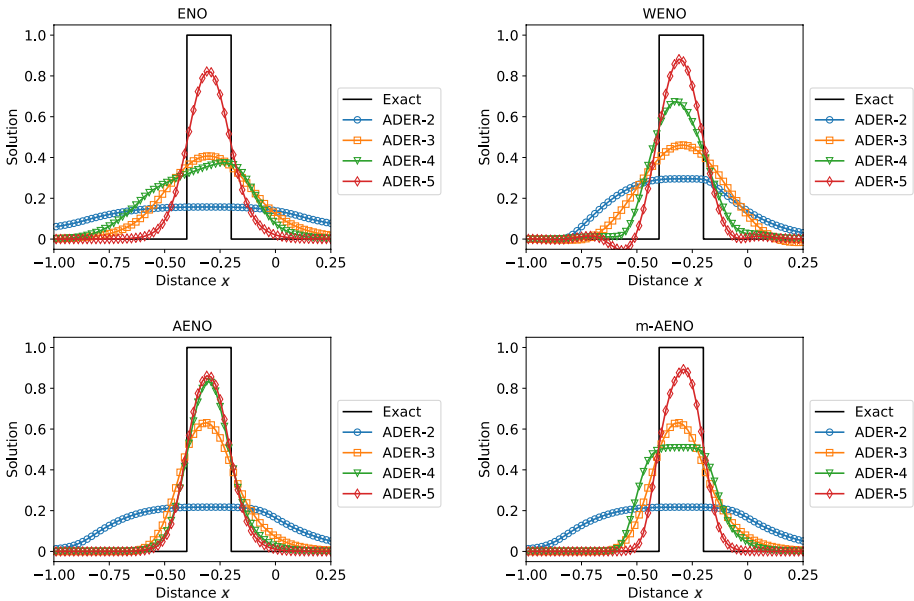


Fig. 17 Square wave results: ADER scheme with ENO, WENO, AENO and m-AENO reconstructions for the linear advection with $\lambda = 1$, $M = 100$, $T_{out} = 2\,000$ and $C_{eff} = 0.9$

Table 2 L_1 -errors for multi-wave test IVP4 (107) from the ADER scheme from 2nd to 5th order, in conjunction with ENO, m-ENO, WENO, AENO and m-AENO reconstructions, as applied the linear advection equation with $\lambda = 1, M = 100, T_{\text{out}} = 2000$ and $C_{\text{eff}} = 0.9$

| | ENO | m-ENO | WENO | AENO | m-AENO |
|--------|----------|----------|----------|----------|----------|
| ADER-2 | 6.64E-01 | 6.64E-01 | 6.13E-01 | 6.46E-01 | 6.46E-01 |
| ADER-3 | 5.94E-01 | 5.73E-01 | 5.72E-01 | 4.58E-01 | 4.58E-01 |
| ADER-4 | 5.00E-01 | 4.80E-01 | 5.46E-01 | 3.06E-01 | 5.55E-01 |
| ADER-5 | 3.09E-01 | 2.65E-01 | 2.66E-01 | 2.76E-01 | 3.87E-01 |

schemes, noting that for orders 4 and 5 we have five distinct reconstruction schemes. For second-order ADER (ADER-2), WENO gives the best results, whereas for all higher-order ADER schemes, AENO gives the best results, by an appreciable margin, most evident for 3rd and 4th orders of accuracy. Note the undershoots in the WENO results for the 5th order scheme. Figure 17 shows four frames corresponding to four reconstruction schemes, including the two new ones of this paper, namely AENO and m-AENO. For each reconstruction schemes we see the effect of increasing the order of accuracy, from 2 to 5. Note the peculiar behaviour of m-AENO for the 4th order scheme. At this stage we note, however, that when it comes to smooth solutions, even with large derivatives, the modified AENO scheme, namely m-AENO based on m-ENO, performs very well indeed, as we shall see in the convergence rates results in Sect. 5.2.

5.2 Convergence Rate Study for the Linear Advection Equation

In this section, we carry out a convergence rate study for the linear advection equation through IVP2 (105) and IVP3 (106), for the schemes of 2nd to 5th order of accuracy in space and time. We compare the newly presented schemes AENO and m-AENO, with the established ENO, m-ENO and WENO reconstruction methods, all of them used with the fully discrete ADER approach. IVP2 (105) could be described as a smooth test of the kind commonly used in the literature to assess convergence rates. IVP3 (106) also serves the same purpose but it is recognised as an exceedingly severe test problem, for which many commonly used high-order schemes fail to give the expected rates.

Results for IVP2. Results are shown in Tables 3, 4, 5, 6, 7, 8, 9, 10, 11 and 12, where errors and corresponding orders of accuracy are shown for the three norms L_1, L_2 and L_∞ . Results for IVP2 (105) are shown in Tables 3, 4, 5, 6 and 7. All schemes attain the theoretically expected convergence rates, for all orders, in the L_1 norm. WENO, with the exception of the second-order scheme, attains the theoretically expected convergence rates also in the L_2 and L_∞ norms. The remaining schemes show poor performance in these norms, especially in the L_∞ norm. Generally, ENO and m-ENO show comparable performance, perhaps with small advantage to ENO. The new AENO scheme does not perform satisfactory in the L_2 and L_∞ norms; as a matter of fact in these norms AENO is inferior to ENO and m-ENO. The new m-AENO scheme improves with respect to AENO for the 5th order scheme.

Figure 18 shows computed L_1 -errors for IVP2 (105), as the mesh is refined, for the ADER scheme with all reconstruction schemes of this paper: ENO, m-ENO, WENO, AENO and m-AENO. With the exception of the second-order case, for which WENO is the

Fig. 18 Computed L_1 -errors for IVP2 (105), as the mesh is refined, for the ADER scheme with reconstructions: ENO, m-ENO, WENO, AENO and m-AENO, as applied to the linear advection with $\lambda = 1$, $T_{\text{out}} = 0.5$ and $C_{\text{eff}} = 0.9$

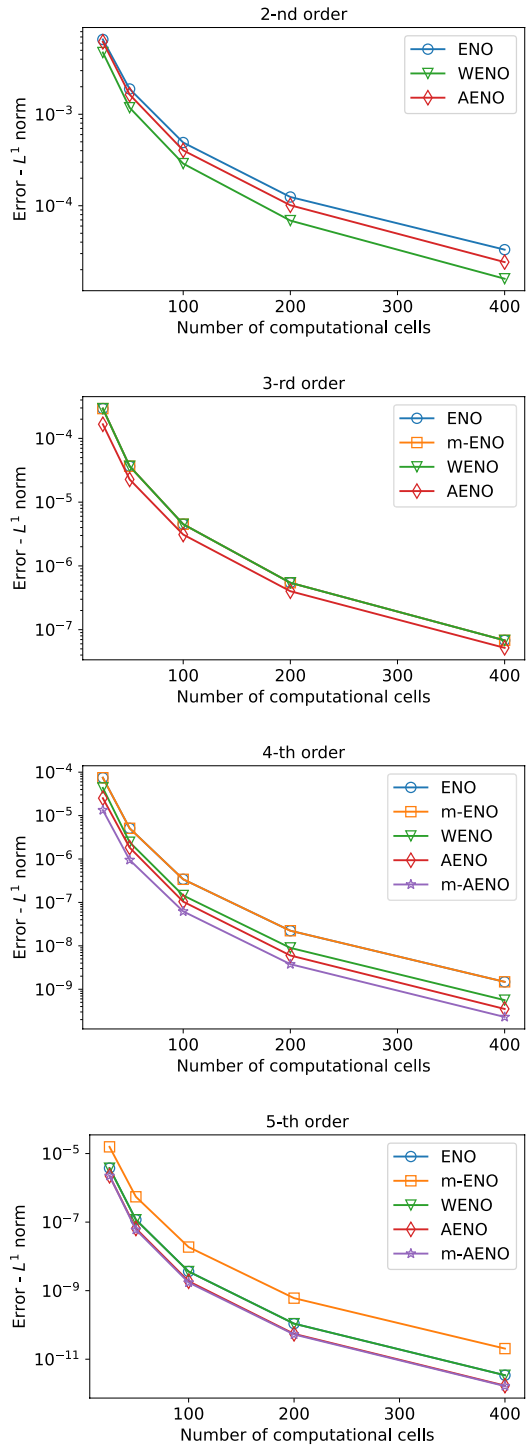


Fig. 19 Computed L_1 -errors for IVP3 (106), as the mesh is refined, for the ADER scheme with reconstructions: ENO, m-ENO, WENO, AENO and m-AENO, as applied to the linear advection with $\lambda = 1$, $T_{\text{out}} = 4$ and $C_{\text{eff}} = 0.9$

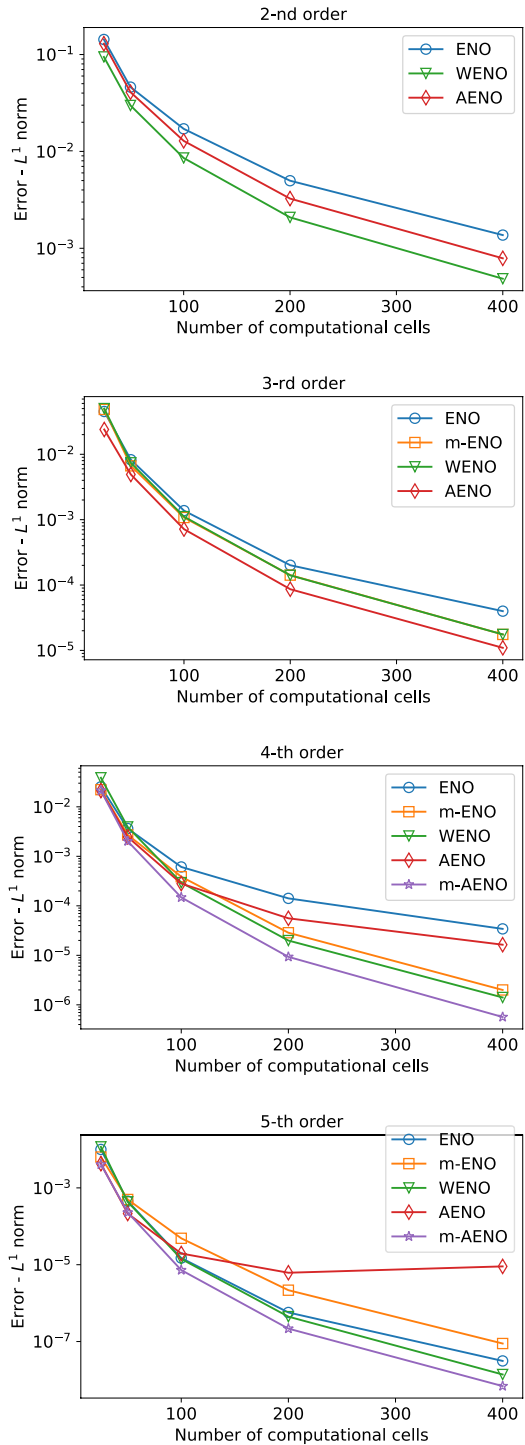


Table 3 ENO results for IVP2 (105): convergence-rate study for the ADER scheme with the ENO reconstruction for the linear advection with $\lambda = 1$, $T_{\text{out}} = 0.5$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 25 | 6.6E-03 | – | 9.1E-03 | – | 2.5E-02 | – | 0 |
| 50 | 1.9E-03 | 1.8 | 2.9E-03 | 1.6 | 1.0E-02 | 1.3 | 0.1 |
| 100 | 4.9E-04 | 1.9 | 9.3E-04 | 1.7 | 4.0E-03 | 1.3 | 0.2 |
| 200 | 1.2E-04 | 2.0 | 2.9E-04 | 1.7 | 1.6E-03 | 1.3 | 0.8 |
| 400 | 3.3E-05 | 1.9 | 9.4E-05 | 1.6 | 6.8E-04 | 1.3 | 3.3 |
| ADER-3 | | | | | | | |
| 25 | 2.9E-04 | – | 3.3E-04 | – | 5.5E-04 | – | 0 |
| 50 | 3.7E-05 | 3.0 | 4.1E-05 | 3.0 | 6.3E-05 | 3.1 | 0.2 |
| 100 | 4.5E-06 | 3.0 | 5.0E-06 | 3.0 | 7.6E-06 | 3.0 | 0.7 |
| 200 | 5.4E-07 | 3.0 | 6.0E-07 | 3.1 | 8.8E-07 | 3.1 | 2.7 |
| 400 | 6.8E-08 | 3.0 | 7.6E-08 | 3.0 | 1.1E-07 | 3.0 | 11.4 |
| ADER-4 | | | | | | | |
| 25 | 7.5E-05 | – | 1.1E-04 | – | 3.2E-04 | – | 0.1 |
| 50 | 5.1E-06 | 3.9 | 9.0E-06 | 3.6 | 3.4E-05 | 3.2 | 0.5 |
| 100 | 3.4E-07 | 3.9 | 7.3E-07 | 3.6 | 3.7E-06 | 3.2 | 2.0 |
| 200 | 2.2E-08 | 3.9 | 6.0E-08 | 3.6 | 4.1E-07 | 3.1 | 8.2 |
| 400 | 1.5E-09 | 3.9 | 5.0E-09 | 3.6 | 4.6E-08 | 3.2 | 31.3 |
| ADER-5 | | | | | | | |
| 25 | 3.7E-06 | – | 4.2E-06 | – | 6.7E-06 | – | 0.2 |
| 50 | 1.2E-07 | 5.0 | 1.3E-07 | 5.0 | 2.0E-07 | 5.0 | 0.7 |
| 100 | 3.6E-09 | 5.0 | 4.0E-09 | 5.0 | 6.2E-09 | 5.0 | 3.1 |
| 200 | 1.1E-10 | 5.1 | 1.2E-10 | 5.1 | 1.8E-10 | 5.1 | 12.5 |
| 400 | 3.4E-12 | 5.0 | 3.8E-12 | 5.0 | 5.5E-12 | 5.0 | 49.5 |

most accurate reconstruction method, the AENO schemes of this paper outperform all the other reconstruction methods.

Results for IVP3. As stated previously IVP3 is an exceedingly severe test problem, for which many of the commonly used high-order schemes fail to give the expected convergence rates. Results for the schemes of this paper are shown in Tables 8, 9, 10, 11 and 12, where errors and corresponding orders of accuracy are shown for the three norms L_1 , L_2 and L_∞ .

For the ENO scheme, results are shown in Table 8. The scheme attains the expected convergence rates sub-optimally for the second and third order cases in the L_1 norm, while failing in the L_2 and L_∞ norms. The 4th order ENO scheme fails to attain the expected rates in all three norms, while for the 5th order case computed rates are close to the expected ones, but are sub-optimal. In summary, ENO fails the convergence rate test for IVP3. For the modified ENO (m-ENO) scheme of Shu [26] we show the computed results in Table 9,

Table 4 m-ENO results for IVP2 (105): convergence-rate study for the ADER scheme with the m-ENO reconstruction for the linear advection with $\lambda = 1, T_{\text{out}} = 0.5$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 25 | 6.6E-03 | – | 9.1E-03 | – | 2.5E-02 | – | 0 |
| 50 | 1.9E-03 | 1.8 | 2.9E-03 | 1.6 | 1.0E-02 | 1.3 | 0.1 |
| 100 | 4.9E-04 | 1.9 | 9.3E-04 | 1.7 | 4.0E-03 | 1.3 | 0.2 |
| 200 | 1.2E-04 | 2.0 | 2.9E-04 | 1.7 | 1.6E-03 | 1.3 | 0.8 |
| 400 | 3.3E-05 | 1.9 | 9.4E-05 | 1.6 | 6.8E-04 | 1.3 | 3.3 |
| ADER-3 | | | | | | | |
| 25 | 2.9E-04 | – | 3.3E-04 | – | 4.6E-04 | – | 0 |
| 50 | 3.7E-05 | 3.0 | 4.1E-05 | 3.0 | 5.7E-05 | 3.0 | 0.2 |
| 100 | 4.5E-06 | 3.0 | 5.0E-06 | 3.0 | 7.1E-06 | 3.0 | 0.7 |
| 200 | 5.4E-07 | 3.0 | 6.0E-07 | 3.0 | 8.5E-07 | 3.0 | 2.9 |
| 400 | 6.8E-08 | 3.0 | 7.6E-08 | 3.0 | 1.1E-07 | 3.0 | 11.4 |
| ADER-4 | | | | | | | |
| 25 | 7.5E-05 | – | 1.1E-04 | – | 3.2E-04 | – | 0.1 |
| 50 | 5.1E-06 | 3.9 | 9.0E-06 | 3.6 | 3.4E-05 | 3.2 | 0.5 |
| 100 | 3.4E-07 | 3.9 | 7.3E-07 | 3.6 | 3.7E-06 | 3.2 | 2.1 |
| 200 | 2.2E-08 | 3.9 | 6.0E-08 | 3.6 | 4.1E-07 | 3.1 | 7.9 |
| 400 | 1.5E-09 | 3.9 | 5.0E-09 | 3.6 | 4.6E-08 | 3.2 | 33.7 |
| ADER-5 | | | | | | | |
| 25 | 1.6E-05 | – | 2.0E-05 | – | 4.7E-05 | – | 0.2 |
| 50 | 5.5E-07 | 4.8 | 8.7E-07 | 4.5 | 2.3E-06 | 4.4 | 0.8 |
| 100 | 1.9E-08 | 4.9 | 3.6E-08 | 4.6 | 1.4E-07 | 4.0 | 3.1 |
| 200 | 6.0E-10 | 5.0 | 1.5E-09 | 4.6 | 8.3E-09 | 4.1 | 11.8 |
| 400 | 2.0E-11 | 4.9 | 6.3E-11 | 4.5 | 5.0E-10 | 4.1 | 49.2 |

where significant improvements with respect to ENO are seen; however, computed rates are sub-optimal and failure is seen in some cases for the L_2 and L_∞ norms.

For the new AENO scheme of this paper, results are shown in Table 10. For the lower-order cases, 2nd and 3rd order, results are satisfactory, but the scheme fails to attain the expected convergence rates for the higher-order schemes, 4th and 5th order. For the new variant of AENO, called m-AENO, results are shown in Table 11. Note that m-AENO is identical to AENO for the second- and third-order schemes, see Table 10 and corresponding comments. The performance of m-AENO for the 4th and 5th order schemes is very satisfactory, attaining the expected rates in the L_1 and L_2 norms, even if sub-optimally for the L_2 norm. In the 4th order case the scheme does not attain the expected rate in the L_∞ norm. In summary, m-AENO constitutes a significant improvement over AENO, for the higher-order range.

Table 5 AENO results for IVP2 (105): convergence-rate study for the ADER schemes with the AENO reconstruction with $\epsilon^2 = 0.5$ for the linear advection for $\lambda = 1$, $T_{\text{out}} = 0.5$, $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------------------------------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 6.2E-03 | – | 8.9E-03 | – | 2.4E-02 | – | 0 |
| 50 | 1.6E-03 | 1.9 | 2.8E-03 | 1.7 | 9.4E-03 | 1.3 | 0.1 |
| 100 | 4.0E-04 | 2.0 | 8.9E-04 | 1.7 | 3.6E-03 | 1.4 | 0.5 |
| 200 | 1.0E-04 | 2.0 | 2.7E-04 | 1.7 | 1.4E-03 | 1.3 | 1.8 |
| 400 | 2.4E-05 | 2.1 | 8.5E-05 | 1.7 | 5.7E-04 | 1.3 | 7.6 |
| ADER-3 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 1.7E-04 | – | 1.9E-04 | – | 3.5E-04 | – | 0.1 |
| 50 | 2.3E-05 | 2.9 | 2.7E-05 | 2.9 | 5.2E-05 | 2.7 | 0.5 |
| 100 | 3.1E-06 | 2.9 | 3.9E-06 | 2.8 | 9.7E-06 | 2.4 | 1.7 |
| 200 | 4.0E-07 | 3.0 | 6.1E-07 | 2.7 | 2.3E-06 | 2.1 | 5.1 |
| 400 | 5.2E-08 | 3.0 | 1.0E-07 | 2.6 | 5.9E-07 | 2.0 | 21.1 |
| ADER-4 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 2.5E-05 | – | 3.8E-05 | – | 1.3E-04 | – | 0.2 |
| 50 | 1.8E-06 | 3.8 | 3.4E-06 | 3.5 | 1.2E-05 | 3.4 | 0.9 |
| 100 | 1.0E-07 | 4.1 | 2.4E-07 | 3.8 | 1.2E-06 | 3.3 | 3.8 |
| 200 | 6.0E-09 | 4.1 | 2.0E-08 | 3.6 | 1.5E-07 | 3.0 | 14.5 |
| 400 | 3.5E-10 | 4.1 | 1.5E-09 | 3.7 | 1.4E-08 | 3.4 | 59.2 |
| ADER-5 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 2.2E-06 | – | 2.8E-06 | – | 6.6E-06 | – | 0.3 |
| 50 | 6.5E-08 | 5.1 | 9.9E-08 | 4.8 | 3.6E-07 | 4.2 | 1.2 |
| 100 | 1.9E-09 | 5.1 | 3.8E-09 | 4.7 | 1.9E-08 | 4.2 | 4.8 |
| 200 | 5.5E-11 | 5.1 | 1.5E-10 | 4.6 | 1.2E-09 | 4.0 | 19.4 |
| 400 | 1.7E-12 | 5.0 | 6.2E-12 | 4.6 | 6.3E-11 | 4.2 | 82.2 |

Results for the classical WENO scheme in conjunction with the fully discrete ADER approach are shown in Table 10. Generally, the WENO results are satisfactory for this severe test problem. For the second-order case convergence in the L_1 norm is attained, being sub-optimal for the L_2 norm and failing for the L_∞ norm. For the higher-order schemes, convergence is attained in all norms, except for the 4th order scheme, where convergence is sub-optimal in the L_1 norm and fails in the L_2 and L_∞ norms. In summary, out of the five reconstruction schemes, WENO gives the best convergence rate performance for this demanding IVP3 test problem.

Figure 19 shows computed L_1 -errors for IVP3 (106), as the mesh is refined, for the ADER scheme with all five reconstructions methods: ENO, m-ENO, WENO, AENO and m-AENO. For the second-order scheme, WENO has the smallest errors, followed by AENO and then by ENO. For the third-order schemes, AENO has the smallest error, followed by WENO, m-ENO and ENO. For the 4th and 5th order schemes, m-AENO has

Table 6 m-AENO results for IVP2 (105): convergence-rate study for the ADER schemes with the m-AENO reconstruction with $\epsilon^2 = 0.5$ for the linear advection for $\lambda = 1, T_{\text{out}} = 0.5, C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------------------------------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 6.2E-03 | – | 8.9E-03 | – | 2.4E-02 | – | 0 |
| 50 | 1.6E-03 | 1.9 | 2.8E-03 | 1.7 | 9.4E-03 | 1.3 | 0.1 |
| 100 | 4.0E-04 | 2.0 | 8.9E-04 | 1.7 | 3.6E-03 | 1.4 | 0.5 |
| 200 | 1.0E-04 | 2.0 | 2.7E-04 | 1.7 | 1.4E-03 | 1.3 | 1.8 |
| 400 | 2.4E-05 | 2.1 | 8.5E-05 | 1.7 | 5.7E-04 | 1.3 | 7.6 |
| ADER-3 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 1.7E-04 | – | 1.9E-04 | – | 3.5E-04 | – | 0.1 |
| 50 | 2.3E-05 | 2.9 | 2.7E-05 | 2.9 | 5.2E-05 | 2.7 | 0.5 |
| 100 | 3.1E-06 | 2.9 | 3.9E-06 | 2.8 | 9.7E-06 | 2.4 | 1.7 |
| 200 | 4.0E-07 | 3.0 | 6.1E-07 | 2.7 | 2.3E-06 | 2.1 | 5.1 |
| 400 | 5.2E-08 | 3.0 | 1.0E-07 | 2.6 | 5.9E-07 | 2.0 | 21.1 |
| ADER-4 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 1.3E-05 | – | 1.6E-05 | – | 3.9E-05 | – | 0.2 |
| 50 | 9.5E-07 | 3.8 | 1.5E-06 | 3.5 | 5.4E-06 | 2.8 | 0.9 |
| 100 | 6.2E-08 | 3.9 | 1.2E-07 | 3.7 | 5.3E-07 | 3.4 | 3.5 |
| 200 | 3.8E-09 | 4.0 | 9.4E-09 | 3.6 | 6.5E-08 | 3.0 | 15.7 |
| 400 | 2.3E-10 | 4.0 | 7.4E-10 | 3.7 | 6.9E-09 | 3.2 | 61.5 |
| ADER-5 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 2.2E-06 | – | 3.1E-06 | – | 8.5E-06 | – | 0.4 |
| 50 | 6.0E-08 | 5.2 | 7.6E-08 | 5.3 | 2.1E-07 | 5.3 | 1.4 |
| 100 | 1.7E-09 | 5.1 | 2.0E-09 | 5.2 | 7.2E-09 | 4.9 | 5.4 |
| 200 | 5.3E-11 | 5.0 | 6.0E-11 | 5.1 | 2.0E-10 | 5.2 | 20.4 |
| 400 | 1.6E-12 | 5.0 | 1.9E-12 | 5.0 | 8.7E-12 | 4.5 | 82.6 |

the smallest error, while AENO’s performance deteriorates visibly as the order of accuracy increases. This observation is consistent with the convergence rate study for IVP3 discussed above.

In the next section, we assess the new AENO reconstruction method as applied to a non-linear hyperbolic system.

6 Results for the Blood Flow Equations for Arteries

In this section, we assess the methods as applied to a non-linear hyperbolic system. The ADER method is implemented with the Toro-Titarev solver [41] for the generalized Riemann problem, as described in Sect. 2.2.2. All three reconstruction methods are implemented in terms of characteristic variables, for all orders of accuracy.

Table 7 WENO results for IVP2 (105): convergence-rate study for ADER schemes with the WENO reconstruction for the linear advection with $\lambda = 1$, $T_{\text{out}} = 0.5$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 25 | 4.7E-03 | – | 7.3E-03 | – | 2.0E-02 | – | 0 |
| 50 | 1.2E-03 | 2.0 | 2.3E-03 | 1.7 | 7.4E-03 | 1.4 | 0.2 |
| 100 | 2.9E-04 | 2.0 | 7.0E-04 | 1.7 | 2.8E-03 | 1.4 | 0.6 |
| 200 | 6.9E-05 | 2.1 | 2.0E-04 | 1.8 | 1.1E-03 | 1.4 | 2.4 |
| 400 | 1.6E-05 | 2.1 | 6.0E-05 | 1.8 | 4.2E-04 | 1.3 | 10.3 |
| ADER-3 | | | | | | | |
| 25 | 2.9E-04 | – | 3.3E-04 | – | 4.6E-04 | – | 0.1 |
| 50 | 3.7E-05 | 3.0 | 4.1E-05 | 3.0 | 5.7E-05 | 3.0 | 0.5 |
| 100 | 4.5E-06 | 3.0 | 5.0E-06 | 3.0 | 7.1E-06 | 3.0 | 1.3 |
| 200 | 5.4E-07 | 3.0 | 6.0E-07 | 3.0 | 8.5E-07 | 3.0 | 5.2 |
| 400 | 6.8E-08 | 3.0 | 7.6E-08 | 3.0 | 1.1E-07 | 3.0 | 20.2 |
| ADER-4 | | | | | | | |
| 25 | 4.4E-05 | – | 4.8E-05 | – | 6.8E-05 | – | 0.2 |
| 50 | 2.4E-06 | 4.2 | 2.7E-06 | 4.2 | 3.8E-06 | 4.2 | 1.2 |
| 100 | 1.4E-07 | 4.1 | 1.6E-07 | 4.1 | 2.3E-07 | 4.1 | 3.8 |
| 200 | 9.0E-09 | 4.0 | 1.0E-08 | 4.0 | 1.4E-08 | 4.0 | 15.4 |
| 400 | 5.6E-10 | 4.0 | 6.2E-10 | 4.0 | 8.8E-10 | 4.0 | 60.2 |
| ADER-5 | | | | | | | |
| 25 | 3.7E-06 | – | 4.2E-06 | – | 5.9E-06 | – | 0.3 |
| 50 | 1.2E-07 | 5.0 | 1.3E-07 | 5.0 | 1.8E-07 | 5.0 | 1.3 |
| 100 | 3.6E-09 | 5.0 | 4.0E-09 | 5.0 | 5.7E-09 | 5.0 | 5.7 |
| 200 | 1.1E-10 | 5.1 | 1.2E-10 | 5.1 | 1.7E-10 | 5.1 | 22.6 |
| 400 | 3.4E-12 | 5.0 | 3.8E-12 | 5.0 | 5.4E-12 | 5.0 | 88.6 |

6.1 The Governing Equations

Now we extend and test the new spatial reconstruction AENO method to a non-linear system of hyperbolic equations, namely the blood flow equations for arteries. The system, in general conservation-law form reads

$$\partial_t \mathbf{Q} + \partial_x \mathbf{F}(\mathbf{Q}) = \mathbf{S}(\mathbf{Q}), \tag{109}$$

with

$$\mathbf{Q} = \begin{bmatrix} A \\ Au \\ A\psi \end{bmatrix}, \quad \mathbf{F}(\mathbf{Q}) = \begin{bmatrix} Au \\ Au^2 + \gamma A^{3/2} \\ Au\psi \end{bmatrix}, \quad \mathbf{S}(\mathbf{Q}) = \begin{bmatrix} 0 \\ -Ru \\ 0 \end{bmatrix}. \tag{110}$$

Table 8 ENO results for IVP3 (106): convergence-rate study for the ADER scheme with the ENO reconstruction for the linear advection with $\lambda = 1, T_{\text{out}} = 4$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 25 | 1.4E-01 | – | 1.2E-01 | – | 2.0E-01 | – | 0.1 |
| 50 | 4.6E-02 | 1.6 | 4.5E-02 | 1.4 | 9.4E-02 | 1.1 | 0.2 |
| 100 | 1.7E-02 | 1.4 | 1.8E-02 | 1.3 | 4.0E-02 | 1.2 | 0.8 |
| 200 | 5.0E-03 | 1.8 | 5.9E-03 | 1.6 | 1.7E-02 | 1.2 | 3.2 |
| 400 | 1.4E-03 | 1.9 | 1.8E-03 | 1.7 | 7.2E-03 | 1.3 | 12.7 |
| ADER-3 | | | | | | | |
| 25 | 4.5E-02 | – | 3.9E-02 | – | 5.6E-02 | – | 0.2 |
| 50 | 8.3E-03 | 2.5 | 7.0E-03 | 2.5 | 8.9E-03 | 2.6 | 0.8 |
| 100 | 1.4E-03 | 2.6 | 1.2E-03 | 2.6 | 1.8E-03 | 2.3 | 3.0 |
| 200 | 2.0E-04 | 2.8 | 1.8E-04 | 2.7 | 3.4E-04 | 2.4 | 10.8 |
| 400 | 4.0E-05 | 2.3 | 4.1E-05 | 2.1 | 9.5E-05 | 1.9 | 42.5 |
| ADER-4 | | | | | | | |
| 25 | 2.5E-02 | – | 2.2E-02 | – | 2.8E-02 | – | 0.5 |
| 50 | 3.6E-03 | 2.8 | 3.4E-03 | 2.7 | 5.4E-03 | 2.4 | 2.0 |
| 100 | 6.1E-04 | 2.6 | 5.8E-04 | 2.6 | 1.2E-03 | 2.1 | 7.7 |
| 200 | 1.4E-04 | 2.1 | 1.6E-04 | 1.8 | 4.9E-04 | 1.3 | 30.8 |
| 400 | 3.4E-05 | 2.0 | 4.5E-05 | 1.9 | 1.4E-04 | 1.8 | 124.3 |
| ADER-5 | | | | | | | |
| 25 | 1.0E-02 | – | 8.4E-03 | – | 1.0E-02 | – | 0.8 |
| 50 | 4.5E-04 | 4.5 | 3.8E-04 | 4.5 | 5.5E-04 | 4.2 | 3.1 |
| 100 | 1.5E-05 | 4.9 | 1.3E-05 | 4.9 | 2.1E-05 | 4.7 | 12.2 |
| 200 | 5.7E-07 | 4.7 | 5.0E-07 | 4.7 | 9.8E-07 | 4.4 | 48.2 |
| 400 | 3.1E-08 | 4.2 | 3.4E-08 | 3.9 | 8.5E-08 | 3.5 | 187.9 |

The first two equations are statements of conservation of mass and momentum, while the third equation represents the advection of a tracer $\psi(x, t)$ transported with the blood velocity $u(x, t)$. Here $A(x, t)$ is the blood vessel cross-sectional area. In the source term R is resistance, which is neglected here. The system is closed with a tube law for arteries, given as

$$p = p_{\text{ext}} + K(x) \left[\frac{\sqrt{A}}{\sqrt{A_0}} - 1 \right], \quad K = \frac{\sqrt{\pi}}{1 - v^2} \frac{E(x)h_0(x)}{\sqrt{A_0}}. \tag{111}$$

Table 9 m-ENO results for IVP3 (106): convergence-rate study for the ADER scheme with the m-ENO reconstruction for the linear advection with $\lambda = 1, T_{\text{out}} = 4$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 25 | 1.4E-01 | – | 1.2E-01 | – | 2.0E-01 | – | 0.1 |
| 50 | 4.6E-02 | 1.6 | 4.5E-02 | 1.4 | 9.4E-02 | 1.1 | 0.2 |
| 100 | 1.7E-02 | 1.4 | 1.8E-02 | 1.3 | 4.0E-02 | 1.2 | 0.8 |
| 200 | 5.0E-03 | 1.8 | 5.9E-03 | 1.6 | 1.7E-02 | 1.2 | 3.2 |
| 400 | 1.4E-03 | 1.9 | 1.8E-03 | 1.7 | 7.2E-03 | 1.3 | 12.7 |
| ADER-3 | | | | | | | |
| 25 | 4.9E-02 | – | 4.0E-02 | – | 5.3E-02 | – | 0.2 |
| 50 | 6.7E-03 | 2.9 | 5.8E-03 | 2.8 | 8.2E-03 | 2.7 | 0.7 |
| 100 | 1.1E-03 | 2.6 | 8.8E-04 | 2.7 | 1.1E-03 | 2.9 | 2.7 |
| 200 | 1.4E-04 | 2.9 | 1.1E-04 | 3.0 | 1.5E-04 | 2.8 | 10.9 |
| 400 | 1.8E-05 | 3.0 | 1.4E-05 | 3.0 | 1.9E-05 | 3.0 | 43.1 |
| ADER-4 | | | | | | | |
| 25 | 2.2E-02 | – | 1.9E-02 | – | 2.7E-02 | – | 0.5 |
| 50 | 2.7E-03 | 3.0 | 2.5E-03 | 2.9 | 4.2E-03 | 2.7 | 2.0 |
| 100 | 3.9E-04 | 2.8 | 3.4E-04 | 2.9 | 8.6E-04 | 2.3 | 8.1 |
| 200 | 2.9E-05 | 3.8 | 2.9E-05 | 3.5 | 7.6E-05 | 3.5 | 32.2 |
| 400 | 2.0E-06 | 3.8 | 2.4E-06 | 3.6 | 9.5E-06 | 3.0 | 129.1 |
| ADER-5 | | | | | | | |
| 25 | 6.4E-03 | – | 5.5E-03 | – | 8.4E-03 | – | 0.8 |
| 50 | 5.0E-04 | 3.7 | 4.9E-04 | 3.5 | 7.7E-04 | 3.4 | 3.1 |
| 100 | 4.9E-05 | 3.4 | 4.2E-05 | 3.6 | 6.7E-05 | 3.5 | 12.5 |
| 200 | 2.2E-06 | 4.5 | 2.0E-06 | 4.4 | 3.9E-06 | 4.1 | 49.3 |
| 400 | 8.9E-08 | 4.6 | 9.0E-08 | 4.5 | 2.2E-07 | 4.2 | 194.1 |

Here A_0 is the equilibrium cross-sectional area, E is the Young modulus of the vessel wall and h_0 is the vessel-wall thickness. These three quantities are parameters of the problem and in general they vary along the length of the vessel. In (110) we have assumed that all parameters of the problem are constant and

$$\gamma = \frac{K}{3\rho\sqrt{A_0}}, \tag{112}$$

where ρ is the constant density of the blood. We remark that for blood flow in veins, a different tube law applies [39].

We assess the method in terms of two problems. First, we consider a Riemann problem with exact solution containing discontinuities. The purpose is to assess the performance of the non-linear reconstruction method at discontinuities, expecting absent or much reduced

Table 10 AENO results for IVP3 (106): convergence-rate study for the ADER schemes with the AENO reconstruction with $\epsilon^2 = 0.5$ for the linear advection for $\lambda = 1, T_{\text{out}} = 4, C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------------------------------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 1.3E-01 | – | 1.1E-01 | – | 1.9E-01 | – | 0.1 |
| 50 | 4.1E-02 | 1.7 | 4.2E-02 | 1.4 | 8.5E-02 | 1.2 | 0.5 |
| 100 | 1.3E-02 | 1.7 | 1.5E-02 | 1.5 | 3.5E-02 | 1.3 | 1.8 |
| 200 | 3.3E-03 | 2.0 | 4.7E-03 | 1.7 | 1.4E-02 | 1.3 | 7.1 |
| 400 | 7.9E-04 | 2.0 | 1.4E-03 | 1.7 | 5.5E-03 | 1.4 | 28.6 |
| ADER-3 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 2.4E-02 | – | 2.0E-02 | – | 2.7E-02 | – | 0.3 |
| 50 | 4.9E-03 | 2.3 | 4.4E-03 | 2.2 | 6.8E-03 | 2.0 | 1.3 |
| 100 | 7.2E-04 | 2.8 | 6.2E-04 | 2.8 | 1.0E-03 | 2.7 | 5.1 |
| 200 | 8.6E-05 | 3.1 | 8.1E-05 | 2.9 | 1.4E-04 | 2.9 | 20.2 |
| 400 | 1.1E-05 | 3.0 | 1.1E-05 | 2.8 | 2.9E-05 | 2.3 | 81.0 |
| ADER-4 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 2.1E-02 | – | 1.8E-02 | – | 2.4E-02 | – | 0.9 |
| 50 | 2.5E-03 | 3.1 | 2.3E-03 | 3.0 | 4.3E-03 | 2.5 | 3.5 |
| 100 | 2.8E-04 | 3.1 | 2.8E-04 | 3.1 | 5.2E-04 | 3.0 | 13.8 |
| 200 | 5.6E-05 | 2.3 | 6.6E-05 | 2.1 | 1.4E-04 | 1.9 | 54.1 |
| 400 | 1.6E-05 | 1.8 | 2.2E-05 | 1.6 | 6.1E-05 | 1.2 | 219.1 |
| ADER-5 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 4.3E-03 | – | 3.8E-03 | – | 5.0E-03 | – | 1.3 |
| 50 | 2.2E-04 | 4.3 | 1.8E-04 | 4.4 | 2.5E-04 | 4.3 | 5.2 |
| 100 | 2.0E-05 | 3.5 | 2.3E-05 | 3.0 | 5.5E-05 | 2.2 | 20.2 |
| 200 | 6.2E-06 | 1.7 | 1.1E-05 | 1.1 | 3.8E-05 | 0.5 | 80.2 |
| 400 | 9.0E-06 | – 0.5 | 1.6E-05 | – 0.6 | 5.3E-05 | – 0.5 | 318.9 |

spurious oscillations. The second problem has a smooth exact solution and is used to evaluate the accuracy, expecting to attain the desired convergence rates.

6.2 Riemann Problem Solution Profiles

Here we assess the methods as applied to a Riemann problem test with the exact solution. For this purpose, we solve the homogeneous equations (109)–(110) with initial conditions as given in Table 13. The solution of this problem is discontinuous and therefore the objective of the exercise is to test the expected ENO character of the non-linear reconstruction methods. More particularly, to test the newly presented AENO method. We expect good resolution of smooth parts of the flow, sharp discontinuities and absence, or much reduced, spurious oscillations behind the shock.

Table 11 m-AENO results for IVP3 (106): convergence-rate study for the ADER schemes with the m-AENO reconstruction with $\epsilon^2 = 0.5$ for the linear advection for $\lambda = 1, T_{\text{out}} = 4, C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------------------------------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 1.3E-01 | – | 1.1E-01 | – | 1.9E-01 | – | 0.1 |
| 50 | 4.1E-02 | 1.7 | 4.2E-02 | 1.4 | 8.5E-02 | 1.2 | 0.5 |
| 100 | 1.3E-02 | 1.7 | 1.5E-02 | 1.5 | 3.5E-02 | 1.3 | 1.8 |
| 200 | 3.3E-03 | 2.0 | 4.7E-03 | 1.7 | 1.4E-02 | 1.3 | 7.1 |
| 400 | 7.9E-04 | 2.0 | 1.4E-03 | 1.7 | 5.5E-03 | 1.4 | 28.6 |
| ADER-3 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 2.4E-02 | – | 2.0E-02 | – | 2.7E-02 | – | 0.3 |
| 50 | 4.9E-03 | 2.3 | 4.4E-03 | 2.2 | 6.8E-03 | 2.0 | 1.3 |
| 100 | 7.2E-04 | 2.8 | 6.2E-04 | 2.8 | 1.0E-03 | 2.7 | 5.1 |
| 200 | 8.6E-05 | 3.1 | 8.1E-05 | 2.9 | 1.4E-04 | 2.9 | 20.2 |
| 400 | 1.1E-05 | 3.0 | 1.1E-05 | 2.8 | 2.9E-05 | 2.3 | 81.0 |
| ADER-4 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 2.1E-02 | – | 1.7E-02 | – | 2.2E-02 | – | 0.9 |
| 50 | 2.0E-03 | 3.4 | 2.0E-03 | 3.1 | 3.2E-03 | 2.8 | 3.6 |
| 100 | 1.5E-04 | 3.8 | 1.8E-04 | 3.5 | 4.8E-04 | 2.7 | 14.4 |
| 200 | 9.3E-06 | 4.0 | 1.5E-05 | 3.6 | 5.5E-05 | 3.1 | 56.6 |
| 400 | 5.7E-07 | 4.0 | 1.1E-06 | 3.7 | 5.2E-06 | 3.4 | 224.6 |
| ADER-5 with $\epsilon^2 = 0.5$ | | | | | | | |
| 25 | 4.1E-03 | – | 3.3E-03 | – | 5.1E-03 | – | 1.3 |
| 50 | 2.3E-04 | 4.2 | 1.9E-04 | 4.1 | 2.6E-04 | 4.3 | 5.3 |
| 100 | 7.2E-06 | 5.0 | 6.2E-06 | 4.9 | 1.1E-05 | 4.6 | 20.7 |
| 200 | 2.2E-07 | 5.1 | 1.9E-07 | 5.1 | 3.9E-07 | 4.8 | 81.3 |
| 400 | 6.9E-09 | 5.0 | 5.8E-09 | 5.0 | 1.6E-08 | 4.6 | 321.8 |

Results are shown in Fig. 20 for ENO, Fig. 21 for m-ENO, Fig. 22 for AENO, Fig. 23 for m-AENO and Fig. 24 for WENO. The results show that the numerical solution approximates well the exact one for every order of accuracy and the discontinuity is handled well. For all orders, no visible spurious oscillations are present, except for WENO, and the approximation of the waves improves as the order of accuracy increases. To plotting accuracy, all methods give virtually identical results. The new method performs as well as the well-established ENO and WENO methods. As already noted, WENO differs slightly from ENO, mENO, AENO and m-AENO, in that small spurious oscillations are seen behind the shock for the 4th and 5th order methods.

Table 12 WENO results for IVP3 (106): convergence-rate study for ADER schemes with the WENO reconstruction for the linear advection with $\lambda = 1, T_{\text{out}} = 4$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 25 | 9.4E-02 | – | 8.8E-02 | – | 1.6E-01 | – | 0.2 |
| 50 | 3.0E-02 | 1.7 | 3.2E-02 | 1.5 | 6.9E-02 | 1.2 | 0.6 |
| 100 | 8.6E-03 | 1.8 | 1.0E-02 | 1.6 | 2.5E-02 | 1.4 | 2.5 |
| 200 | 2.1E-03 | 2.0 | 3.0E-03 | 1.8 | 9.5E-03 | 1.4 | 10.6 |
| 400 | 4.9E-04 | 2.1 | 8.3E-04 | 1.9 | 3.7E-03 | 1.4 | 40.8 |
| ADER-3 | | | | | | | |
| 25 | 5.0E-02 | – | 4.1E-02 | – | 5.5E-02 | – | 0.3 |
| 50 | 7.4E-03 | 2.8 | 6.3E-03 | 2.7 | 8.3E-03 | 2.7 | 1.2 |
| 100 | 1.1E-03 | 2.7 | 8.9E-04 | 2.8 | 1.1E-03 | 2.9 | 4.8 |
| 200 | 1.4E-04 | 3.0 | 1.1E-04 | 3.0 | 1.4E-04 | 3.0 | 19.2 |
| 400 | 1.8E-05 | 3.0 | 1.4E-05 | 3.0 | 1.7E-05 | 3.0 | 76.5 |
| ADER-4 | | | | | | | |
| 25 | 3.9E-02 | – | 3.3E-02 | – | 4.5E-02 | – | 0.9 |
| 50 | 3.9E-03 | 3.3 | 3.2E-03 | 3.3 | 5.1E-03 | 3.2 | 3.7 |
| 100 | 2.9E-04 | 3.7 | 3.0E-04 | 3.4 | 6.9E-04 | 2.9 | 14.7 |
| 200 | 2.0E-05 | 3.9 | 2.8E-05 | 3.4 | 9.2E-05 | 2.9 | 57.1 |
| 400 | 1.4E-06 | 3.8 | 2.7E-06 | 3.4 | 1.2E-05 | 2.9 | 226.8 |
| ADER-5 | | | | | | | |
| 25 | 1.2E-02 | – | 9.6E-03 | – | 1.4E-02 | – | 1.4 |
| 50 | 4.3E-04 | 4.8 | 3.4E-04 | 4.8 | 3.6E-04 | 5.3 | 5.6 |
| 100 | 1.4E-05 | 4.9 | 1.1E-05 | 4.9 | 1.2E-05 | 4.9 | 21.7 |
| 200 | 4.4E-07 | 5.0 | 3.5E-07 | 5.0 | 3.7E-07 | 5.0 | 85.4 |
| 400 | 1.4E-08 | 5.0 | 1.1E-08 | 5.0 | 1.2E-08 | 5.0 | 338.5 |

Table 13 Riemann problem test. Initial condition for the classical Riemann problem for the blood flow equations for arteries. The left column defines the quantities on the left and right of the initial discontinuity. Here $A_0 = 3.1353 \times 10^{-4} \text{ m}^2$

| Variables | Initial value |
|-----------|---------------|
| A_L | $2A_0$ |
| u_L | 0 |
| ψ_L | 1 |
| A_R | A_0 |
| u_R | 0 |
| ψ_R | 0.2 |

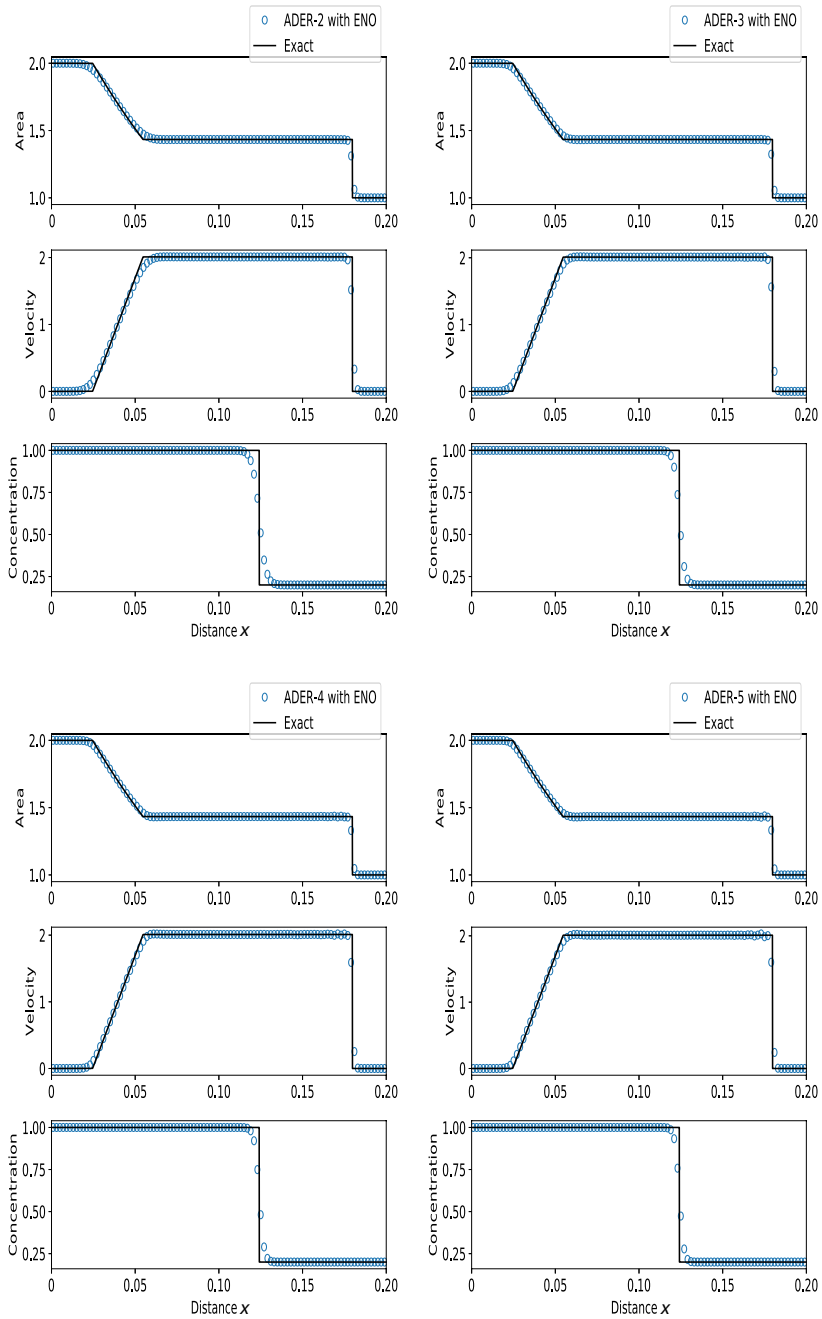


Fig. 20 ENO results for a Riemann problem for blood flow. Numerical solutions from 2nd to 5th order ADER with ENO reconstruction are compared with the exact solution. Computational parameters are: $M = 100$, $T_{out} = 0.012$ and $C_{eff} = 0.9$

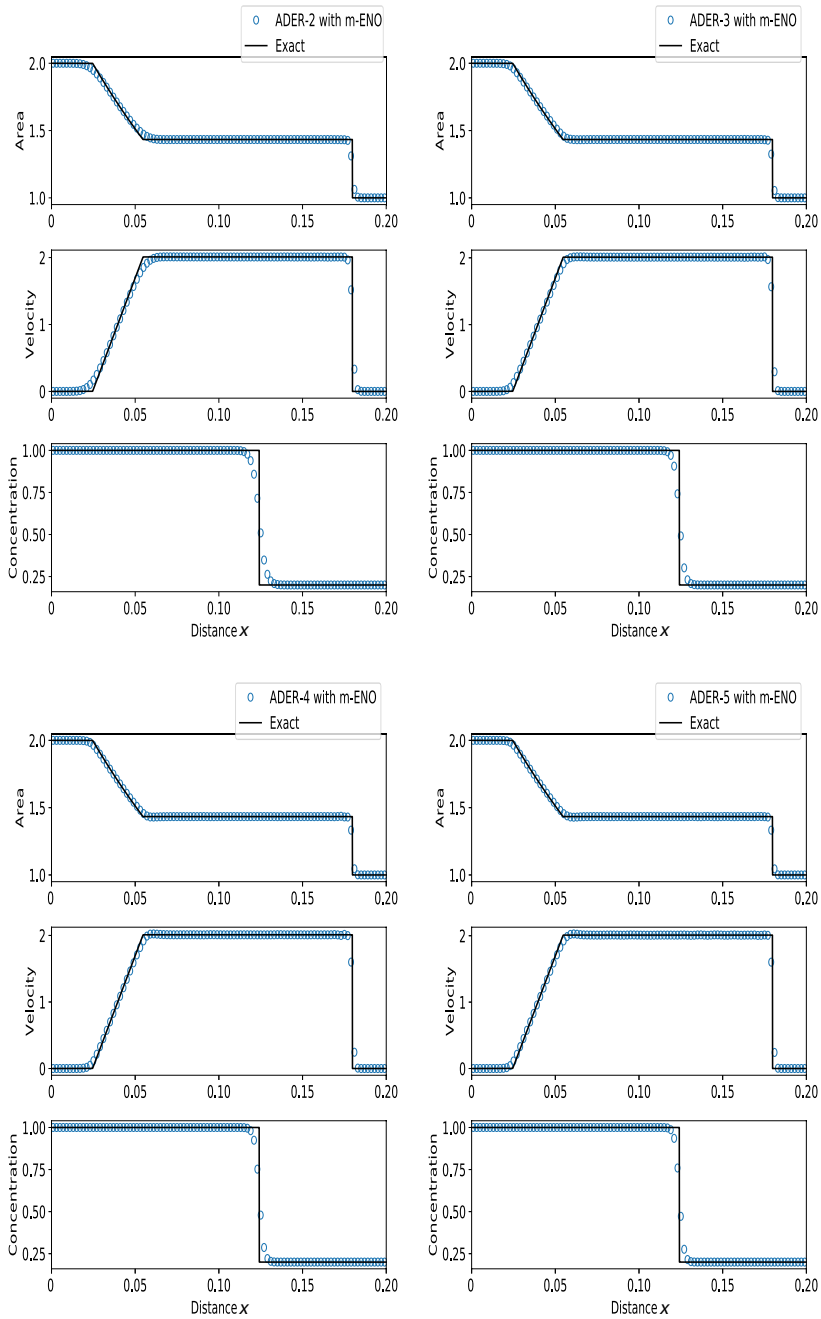


Fig. 21 m-ENO results for a Riemann problem for blood flow. Numerical solutions from 2nd to 5th order ADER with m-ENO reconstruction are compared with the exact solution. Computational parameters are: $M = 100$, $T_{out} = 0.012$ and $C_{cfl} = 0.9$

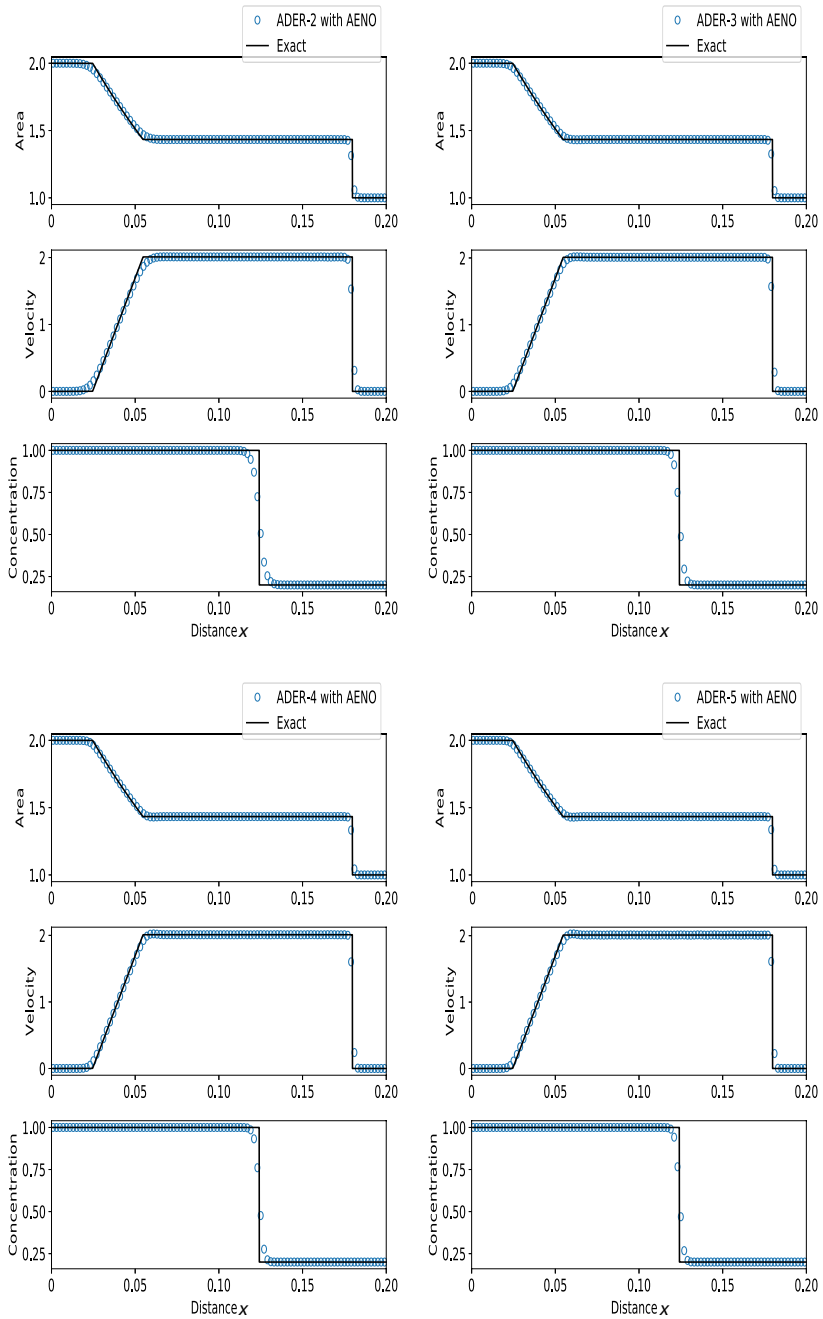


Fig. 22 AENO results for a Riemann problem for blood flow. Numerical solutions from 2nd to 5th order ADER with AENO reconstruction are compared with the exact solution. Computational parameters are: $M = 100$, $T_{out} = 0.012$, $C_{eff} = 0.9$ and $\epsilon = 0.5$

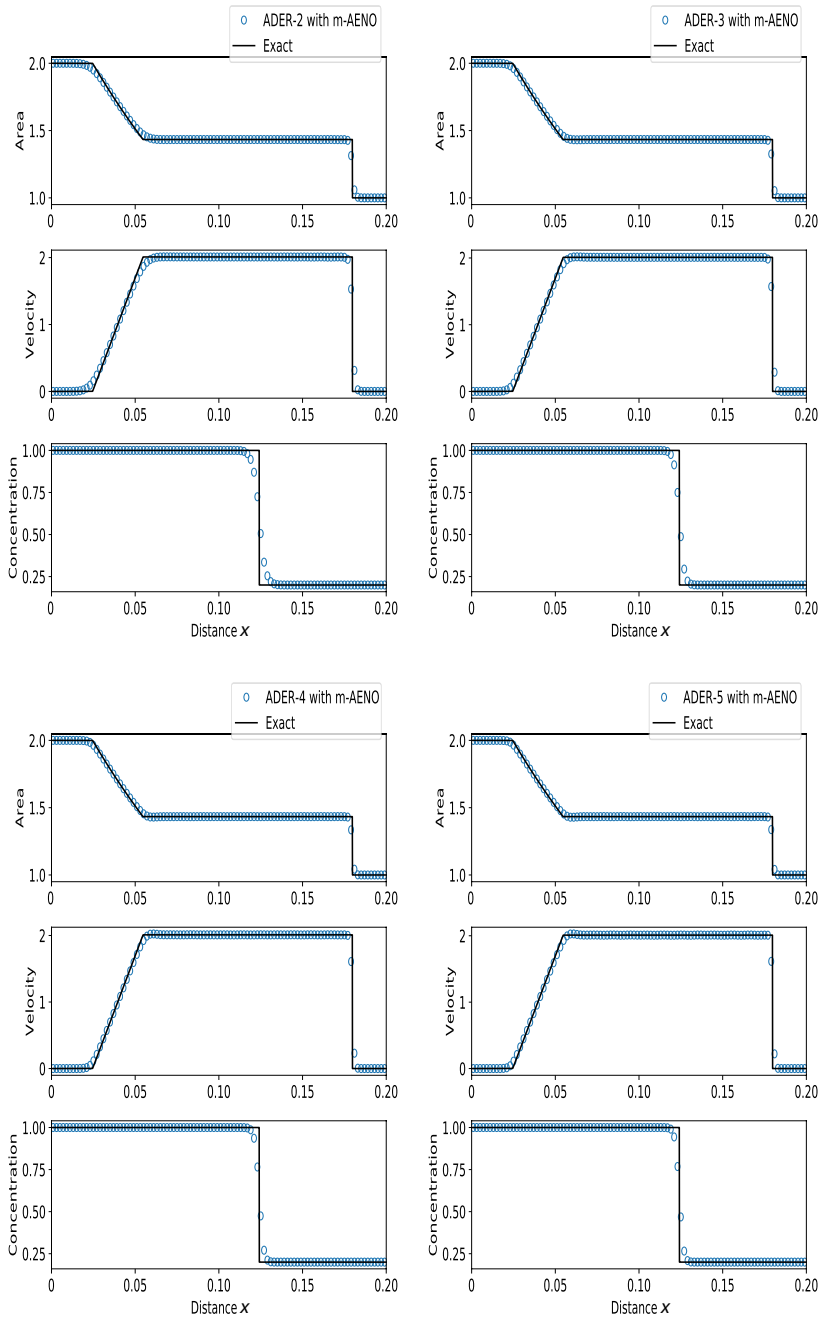


Fig. 23 m-AENO results for a Riemann problem for blood flow. Numerical solutions from 2nd to 5th order ADER with m-AENO reconstruction are compared with the exact solution. Computational parameters are: $M = 100$, $T_{out} = 0.012$, $C_{eff} = 0.9$ and $\epsilon = 0.5$

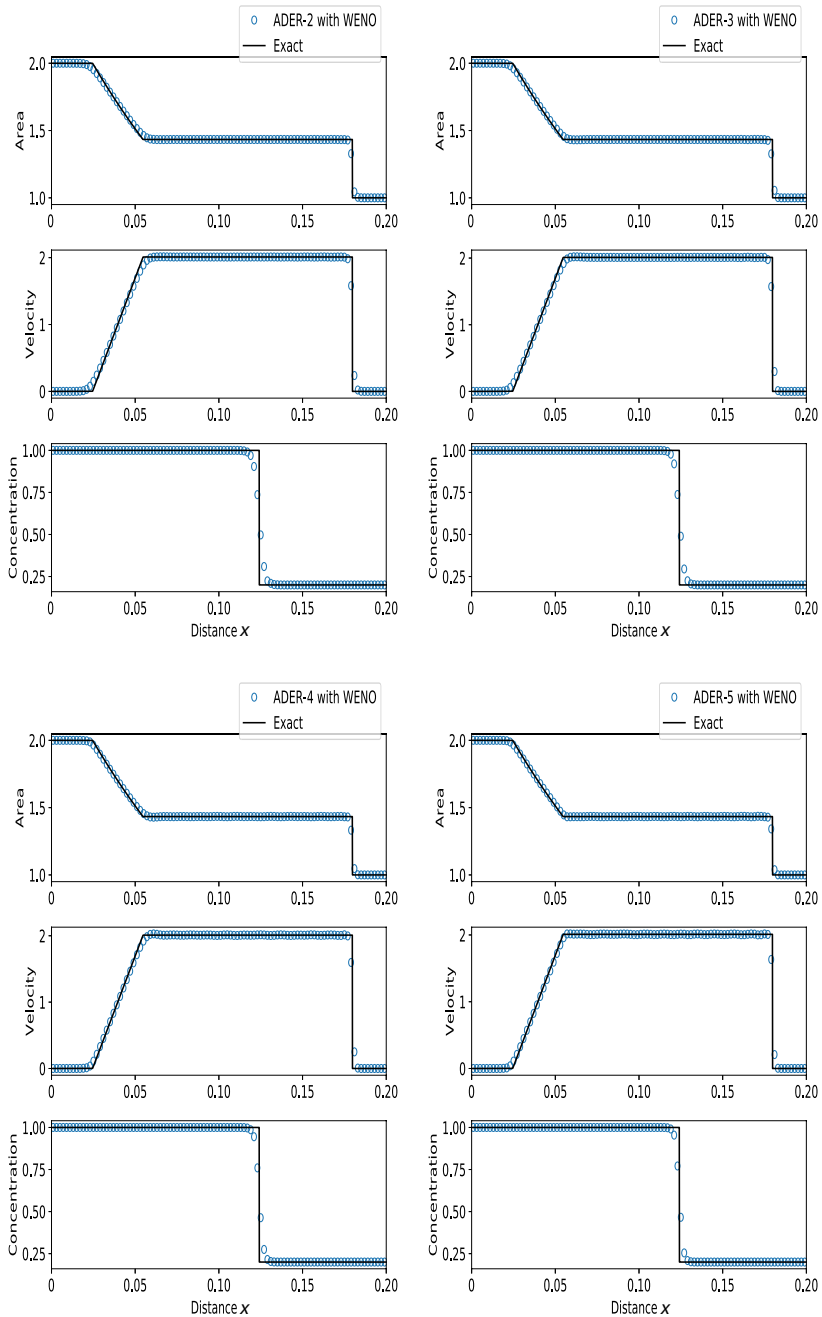


Fig. 24 WENO results for a Riemann problem for blood flow. Numerical solutions from 2nd to 5th order ADER with **WENO** reconstruction are compared with the exact solution. Computational parameters are: $M = 100$, $T_{out} = 0.012$ and $C_{eff} = 0.9$

6.3 Efficiency Assessment on a Riemann Problem Test

A criterion of fundamental importance in assessing the performance of numerical methods is the **cost/benefit** relation. In other words, **accuracy against cost**, or equivalently **error against cost**. This is conventionally achieved by computing the solution to a test problem with the exact solution on a sequence of successively refined meshes M_k , and computing the corresponding errors E_k on a given norm. Associated to each pair (M_k, E_k) there is a computing cost CPU_k . Then, one usually fits a least-square line to the set of points (CPU_k, E_k) on log-log axes. The procedure is performed for each method to be assessed, which produces a graph of several straight lines, one for each method, as shown in Fig. 25.

For the efficiency exercise here, we choose the Riemann problem, with the exact solution, given in Table 14.

Results are shown in Fig. 25 for **third-order schemes** for all reconstruction methods. The top frame shows results for the error in cross-sectional area A , while the bottom frame shows results for the error in velocity u . We have used the sequence of meshes $M = 25, 50, 100, 200, 400, 800, 1600$; see symbols. Then, as is customary in the literature, we have fitted least square straight lines for each scheme. From the results, it is clearly seen that as the mesh is refined, the CPU cost increases and the error decreases. The straight lines corresponding to the schemes diverge as the error decreases. It is seen that for a given cost, the smallest error is obtained from the newly proposed AENO scheme, while the largest error is given by the WENO scheme. The error lines from the remaining schemes lie in between AENO and WENO.

To assess the error/cost relation we fix an error and draw the corresponding horizontal line. For example, in Fig. 25 we first choose a constant error denoted by E_{1600}^{AENO} , this is the error of the AENO scheme for the finest mesh used with $M = 1600$ cells. Then the intercept of this horizontal line with the least square line corresponding to AENO gives the CPU cost. For the cross-sectional area we obtain 12 827 s for AENO and 54 648 s for WENO. This means that AENO is 4.3 times more efficient than WENO.

Obviously, for even smaller errors, this difference in efficiency will increase. By extrapolating to smaller errors, assuming straight lines, we gain a clearer idea of the difference in efficiency of all methods assessed. Extrapolating the straight lines to the chosen small errors of $E = 10^{-8}$ for area A and of $E = 10^{-4}$ for velocity u we obtain estimates for the cost of the schemes; see corresponding arrows in Fig. 25. From the given numbers in Fig. 25 it is seen that the efficiency gain of AENO over WENO is 9.7 for area, while for velocity, bottom plot, this gain is 9.13. In other words, for small errors, the AENO reconstruction method is estimated to be one order of magnitude more efficient than WENO.

6.4 Convergence-Rate Study

To carry out an empirical convergence rate study we consider the method of manufactured solutions. First we prescribe a smooth vector-value function $\tilde{Q}(x, t)$ [22] given as follows:

$$\tilde{Q}(x, t) = \begin{bmatrix} \tilde{A} + \tilde{a}\sin(2\pi x)\cos(2\pi t) \\ -\tilde{a}\cos(2\pi x)\sin(2\pi t) \end{bmatrix}, \tag{113}$$

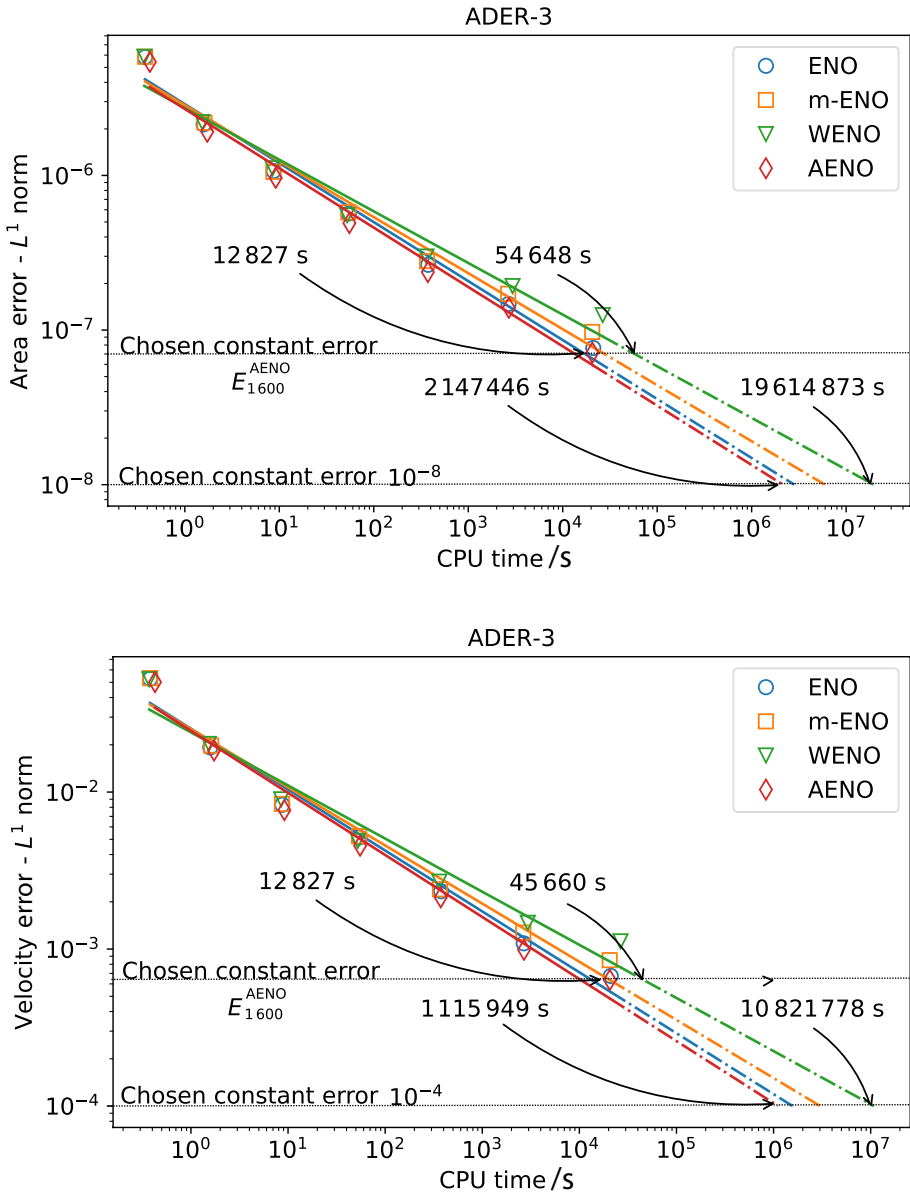


Fig. 25 Efficiency plot for a Riemann problem for blood flow. Numerical solutions for 3rd order ADER with **ENO**, **m-ENO**, **WENO** and **AENO** reconstructions for meshes $M = 25, 50, 100, 200, 400, 800, 1600$; see symbols. For the finest mesh used ($M = 1600$), AENO is seen to be 4.3 times more efficient than WENO, for area A . For extrapolated lines to the small error $E = 10^{-8}$ for area, AENO is seen to be one order of magnitude more efficient than WENO

Table 14 Riemann problem for efficiency test. Initial condition for the classical Riemann problem for the blood flow equations for arteries. Here $A_0 = 3.14 \times 10^{-4} \text{ m}^2$; length of domain = 0.5 m; position of initial discontinuity = 0.25 m

| Variables | Initial value |
|-----------|---------------------------------|
| A_L | $10 \times 10^{-4} \text{ m}^2$ |
| u_L | 0 |
| ψ_L | 1 |
| A_R | $1 \times 10^{-4} \text{ m}^2$ |
| u_R | 0 |
| ψ_R | 0 |

with

$$A_0 = 3.1353 \times 10^{-4} \text{ m}^2, \quad \tilde{A} = A_0, \quad \tilde{a} = A_0 \times 10^{-1}. \tag{114}$$

After inserting (113) into (109) we obtain a new system of balance laws, namely

$$\partial_t \tilde{\mathbf{Q}} + \partial_x \mathbf{F}(\tilde{\mathbf{Q}}) = \mathbf{S}(\tilde{\mathbf{Q}}) + \mathbf{S}_{\text{new}}(x, t). \tag{115}$$

The function $\tilde{\mathbf{Q}}(x, t)$ in (113) is the exact solution of the new balance law (115). We remark that for this test we considered the equations without the passive scalar.

Convergence rates are calculated with respect to the cross-sectional area $A(x, t)$ and flow $q(x, t) = A(x, t) u(x, t)$. Results are shown in Tables 15, 16, 17, 18, 19, 20, 21, 22, 23 and 24. The overall results differ somehow from those for the linear advection equation. Now, essentially, the expected convergence rates are reached for all orders in all three norms, by all three reconstruction methods. In the linear advection case, the AENO method reached the expected rate only in the L_1 norm, being sub-optimal in the L_2 norm; the expected rate in the L_∞ norm was not reached.

Regarding errors, Figs. 26 and 27 show a comparison of L_1 errors for all three methods for all orders and all meshes. Figure 26 shows results for the second (top) and third (bottom) order methods. For the second-order case all three reconstruction schemes give essentially equivalent results. For the third-order schemes, surprisingly, ENO exhibits smaller errors than AENO. Figure 27 shows results for the fourth (top) and fifth

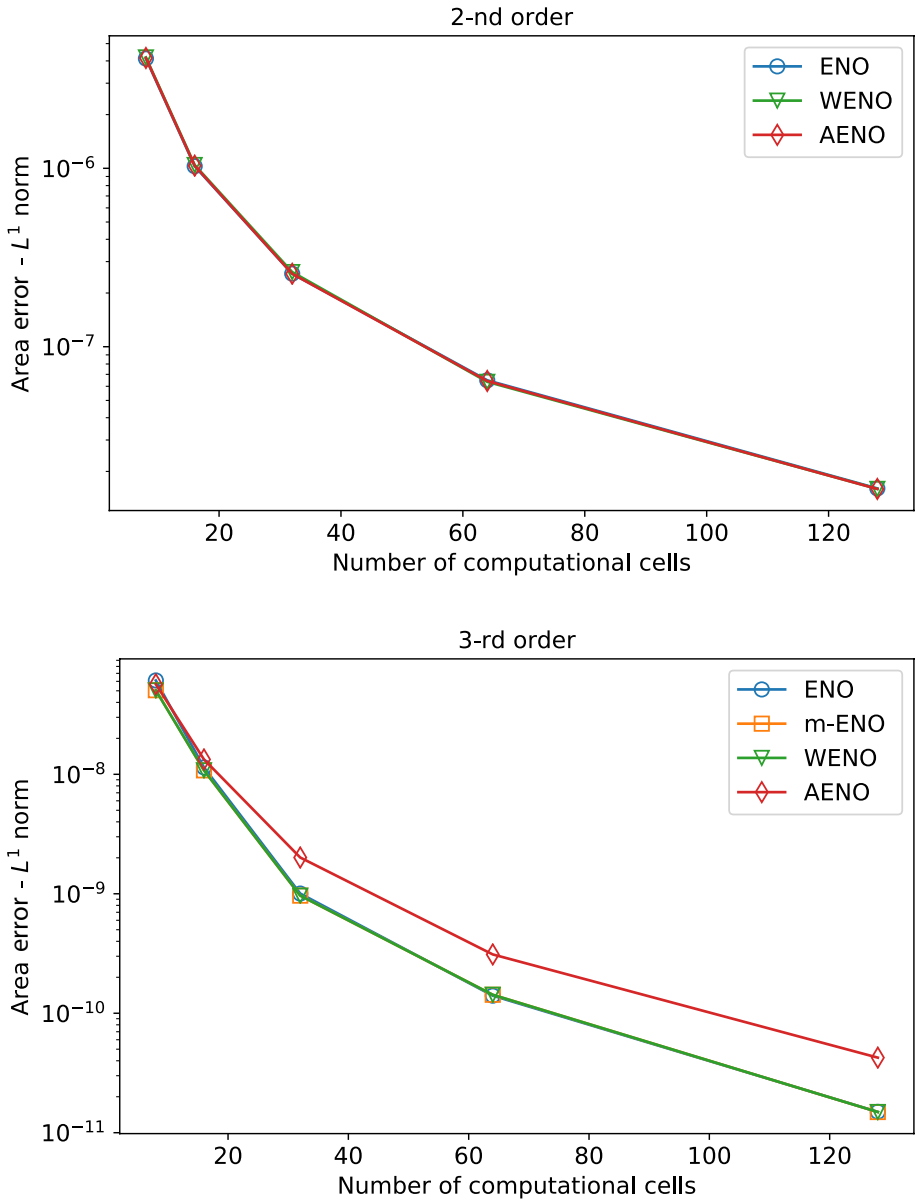


Fig. 26 L_1 -errors for the blood flow equations. Computed errors for cross-sectional area A for the 2nd and 3rd order ADER schemes with ENO, m-ENO, AENO and WENO

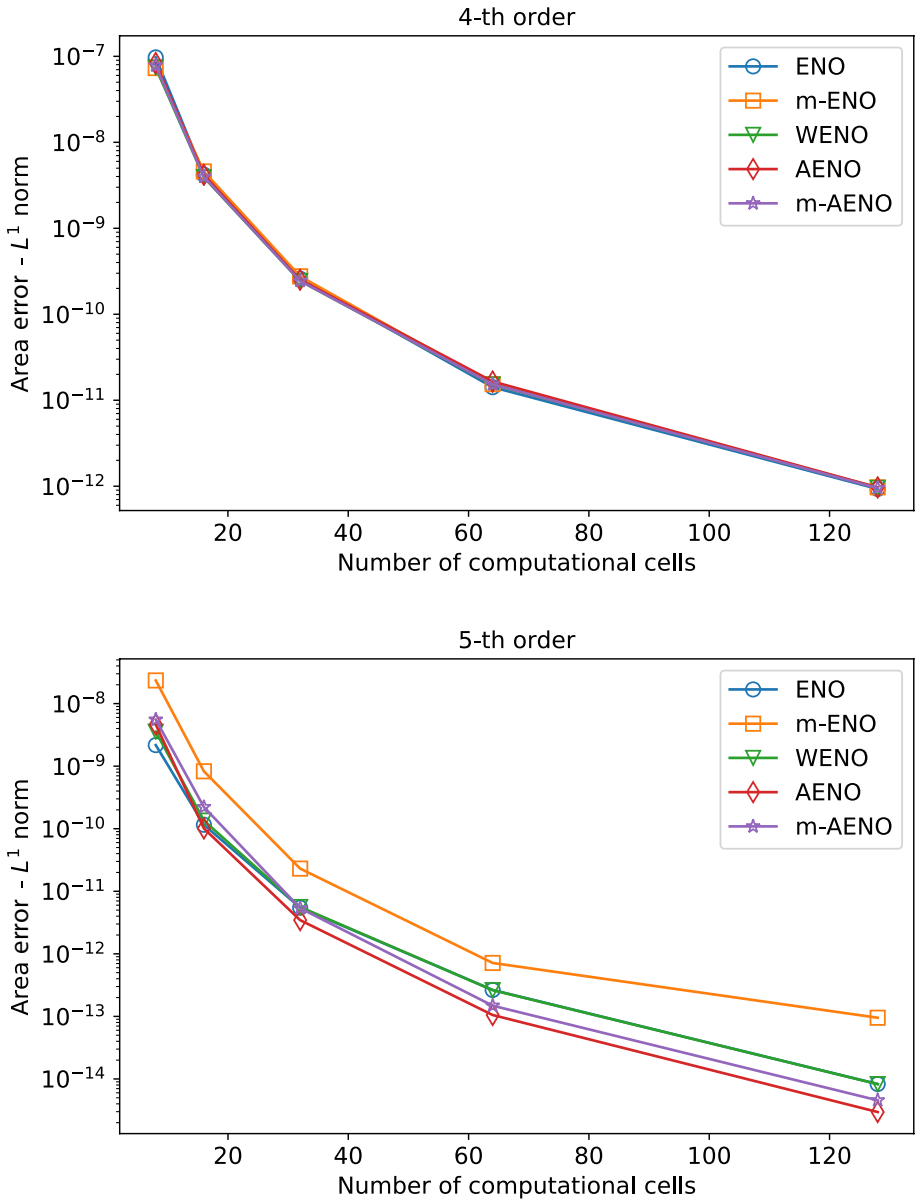


Fig. 27 L_1 -errors for the blood flow equations. Computed errors for cross-sectional area A for the 4th and 5th order ADER schemes with ENO, m-ENO, AENO, m-AENO and WENO

Table 15 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **ENO** reconstruction. Rates are calculated for the cross-sectional area A . Computational parameters are: $T_{\text{out}} = 0.1$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 4.1E-06 | – | 4.5E-06 | – | 6.0E-06 | – | 0 |
| 16 | 1.0E-06 | 2.0 | 1.1E-06 | 2.0 | 1.6E-06 | 1.9 | 0.1 |
| 32 | 2.6E-07 | 2.0 | 2.9E-07 | 2.0 | 4.9E-07 | 1.7 | 0.4 |
| 64 | 6.5E-08 | 2.0 | 7.3E-08 | 2.0 | 1.2E-07 | 2.0 | 2.3 |
| 128 | 1.6E-08 | 2.0 | 1.8E-08 | 2.0 | 3.1E-08 | 2.0 | 14.7 |
| ADER-3 | | | | | | | |
| 8 | 6.1E-08 | – | 8.0E-08 | – | 1.2E-07 | – | 0.1 |
| 16 | 1.1E-08 | 2.4 | 1.4E-08 | 2.6 | 2.4E-08 | 2.3 | 0.3 |
| 32 | 1.0E-09 | 3.5 | 1.1E-09 | 3.6 | 1.6E-09 | 3.9 | 1.1 |
| 64 | 1.4E-10 | 2.8 | 1.6E-10 | 2.8 | 2.7E-10 | 2.6 | 5.0 |
| 128 | 1.5E-11 | 3.2 | 1.7E-11 | 3.3 | 2.5E-11 | 3.4 | 26.3 |
| ADER-4 | | | | | | | |
| 8 | 9.7E-08 | – | 1.1E-07 | – | 1.4E-07 | – | 0.2 |
| 16 | 4.2E-09 | 4.5 | 4.8E-09 | 4.5 | 7.9E-09 | 4.2 | 0.7 |
| 32 | 2.6E-10 | 4.0 | 3.1E-10 | 4.0 | 6.1E-10 | 3.7 | 2.6 |
| 64 | 1.4E-11 | 4.2 | 1.8E-11 | 4.1 | 5.0E-11 | 3.6 | 11.2 |
| 128 | 9.3E-13 | 3.9 | 1.0E-12 | 4.2 | 1.9E-12 | 4.7 | 51.9 |
| ADER-5 | | | | | | | |
| 8 | 2.2E-09 | – | 2.5E-09 | – | 3.7E-09 | – | 0.5 |
| 16 | 1.2E-10 | 4.2 | 1.5E-10 | 4.0 | 2.9E-10 | 3.7 | 1.9 |
| 32 | 5.5E-12 | 4.4 | 6.4E-12 | 4.6 | 1.2E-11 | 4.6 | 7.6 |
| 64 | 2.6E-13 | 4.4 | 2.9E-13 | 4.4 | 4.8E-13 | 4.6 | 31.3 |
| 128 | 8.3E-15 | 5.0 | 9.2E-15 | 5.0 | 1.5E-14 | 5.1 | 129.5 |

Table 16 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **ENO** reconstruction. Rates are calculated for the flow Au . Computational parameters are: $T_{out} = 0.1$ and $C_{cfl} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 5.5E-06 | – | 6.0E-06 | – | 8.1E-06 | – | 0 |
| 16 | 9.5E-07 | 2.5 | 1.1E-06 | 2.4 | 1.9E-06 | 2.1 | 0.1 |
| 32 | 1.9E-07 | 2.3 | 2.2E-07 | 2.4 | 4.0E-07 | 2.3 | 0.4 |
| 64 | 4.3E-08 | 2.1 | 5.2E-08 | 2.1 | 1.1E-07 | 1.9 | 2.3 |
| 128 | 1.1E-08 | 2.0 | 1.2E-08 | 2.1 | 2.6E-08 | 2.1 | 14.7 |
| ADER-3 | | | | | | | |
| 8 | 1.6E-06 | – | 1.9E-06 | – | 2.6E-06 | – | 0.1 |
| 16 | 2.1E-07 | 2.9 | 2.4E-07 | 2.9 | 3.4E-07 | 2.9 | 0.3 |
| 32 | 2.7E-08 | 3.0 | 3.0E-08 | 3.0 | 4.3E-08 | 3.0 | 1.1 |
| 64 | 3.4E-09 | 3.0 | 3.8E-09 | 3.0 | 5.4E-09 | 3.0 | 5.0 |
| 128 | 4.3E-10 | 3.0 | 4.7E-10 | 3.0 | 6.7E-10 | 3.0 | 26.3 |
| ADER-4 | | | | | | | |
| 8 | 5.9E-07 | – | 6.8E-07 | – | 9.7E-07 | – | 0.2 |
| 16 | 2.8E-08 | 4.4 | 3.0E-08 | 4.5 | 4.1E-08 | 4.6 | 0.7 |
| 32 | 1.7E-09 | 4.1 | 2.0E-09 | 4.0 | 3.7E-09 | 3.5 | 2.6 |
| 64 | 9.9E-11 | 4.1 | 1.1E-10 | 4.1 | 2.7E-10 | 3.8 | 11.2 |
| 128 | 5.7E-12 | 4.1 | 6.3E-12 | 4.2 | 1.1E-11 | 4.6 | 51.9 |
| ADER-5 | | | | | | | |
| 8 | 1.7E-07 | – | 1.8E-07 | – | 2.3E-07 | – | 0.5 |
| 16 | 5.3E-09 | 5.0 | 5.7E-09 | 5.0 | 7.9E-09 | 4.8 | 1.9 |
| 32 | 1.6E-10 | 5.0 | 1.8E-10 | 5.0 | 2.6E-10 | 5.0 | 7.6 |
| 64 | 4.9E-12 | 5.0 | 5.4E-12 | 5.0 | 7.6E-12 | 5.1 | 31.3 |
| 128 | 1.5E-13 | 5.0 | 1.7E-13 | 5.0 | 2.4E-13 | 5.0 | 129.5 |

(bottom) order methods. For the fourth-order schemes, all results are virtually coincident. For the fifth-order case, the AENO schemes give the smaller errors.

6.5 Shock/Turbulence Interaction Problem

Here we assess the performance of the methods for a blood-flow analogue of the so-called shock/turbulence interaction problem for gas dynamics proposed in [14]. A variation of

Table 17 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **m-ENO** reconstruction. Rates are calculated for the cross-sectional area A . Computational parameters are: $T_{\text{out}} = 0.1$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 4.1E-06 | – | 4.5E-06 | – | 6.0E-06 | – | 0 |
| 16 | 1.0E-06 | 2.0 | 1.1E-06 | 2.0 | 1.6E-06 | 1.9 | 0.1 |
| 32 | 2.6E-07 | 2.0 | 2.9E-07 | 2.0 | 4.9E-07 | 1.7 | 0.4 |
| 64 | 6.5E-08 | 2.0 | 7.3E-08 | 2.0 | 1.2E-07 | 2.0 | 2.3 |
| 128 | 1.6E-08 | 2.0 | 1.8E-08 | 2.0 | 3.1E-08 | 2.0 | 14.7 |
| ADER-3 | | | | | | | |
| 8 | 5.1E-08 | – | 5.6E-08 | – | 8.1E-08 | – | 0.1 |
| 16 | 1.1E-08 | 2.2 | 1.2E-08 | 2.2 | 1.8E-08 | 2.2 | 0.3 |
| 32 | 9.6E-10 | 3.5 | 1.1E-09 | 3.5 | 1.6E-09 | 3.5 | 1.1 |
| 64 | 1.4E-10 | 2.8 | 1.6E-10 | 2.8 | 2.3E-10 | 2.7 | 5.1 |
| 128 | 1.5E-11 | 3.3 | 1.7E-11 | 3.3 | 2.5E-11 | 3.2 | 28.4 |
| ADER-4 | | | | | | | |
| 8 | 7.3E-08 | – | 9.4E-08 | – | 1.4E-07 | – | 0.3 |
| 16 | 4.6E-09 | 4.0 | 5.5E-09 | 4.1 | 9.9E-09 | 3.9 | 0.9 |
| 32 | 2.8E-10 | 4.1 | 3.6E-10 | 3.9 | 8.7E-10 | 3.5 | 4.2 |
| 64 | 1.5E-11 | 4.2 | 2.0E-11 | 4.2 | 5.9E-11 | 3.9 | 18.4 |
| 128 | 9.7E-13 | 4.0 | 1.1E-12 | 4.2 | 2.4E-12 | 4.6 | 83.1 |
| ADER-5 | | | | | | | |
| 8 | 2.3E-08 | – | 3.1E-08 | – | 6.0E-08 | – | 0.5 |
| 16 | 8.3E-10 | 4.8 | 9.6E-10 | 5.0 | 1.8E-09 | 5.1 | 1.9 |
| 32 | 2.3E-11 | 5.2 | 3.1E-11 | 4.9 | 9.1E-11 | 4.3 | 7.6 |
| 64 | 7.1E-13 | 5.0 | 8.8E-13 | 5.1 | 2.2E-12 | 5.4 | 31.0 |
| 128 | 9.5E-14 | 2.9 | 1.5E-13 | 2.5 | 5.8E-13 | 1.9 | 129.1 |

this test that is more suitable to assess methods for long evolution times was proposed in [42] and [32]. Our test problem here is inspired on the modified version of the problem. We solve the blood flow equations with the initial condition (Figs. 28 and 29)

$$[A, u, \psi] = \begin{cases} [15A_0, 919.7, 1] & \text{if } x \leq 10, \\ [A_0(1 + 0.2 \sin(\frac{80}{50}\pi x)), 0, 1] & \text{if } x > 10 \end{cases} \tag{116}$$

on a domain of length 50 cm, with the reference cross-sectional area chosen as $A_0 = 3.14 \text{ cm}^2$. The solution of the problem consists of a right facing shock wave running

Table 18 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **m-ENO** reconstruction. Rates are calculated for the flow Au . Computational parameters are: $T_{\text{out}} = 0.1$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 5.5E-06 | – | 6.0E-06 | – | 8.1E-06 | – | 0 |
| 16 | 9.5E-07 | 2.5 | 1.1E-06 | 2.4 | 1.9E-06 | 2.1 | 0.1 |
| 32 | 1.9E-07 | 2.3 | 2.2E-07 | 2.4 | 4.0E-07 | 2.3 | 0.4 |
| 64 | 4.3E-08 | 2.1 | 5.2E-08 | 2.1 | 1.1E-07 | 1.9 | 2.3 |
| 128 | 1.1E-08 | 2.0 | 1.2E-08 | 2.1 | 2.6E-08 | 2.1 | 14.7 |
| ADER-3 | | | | | | | |
| 8 | 1.7E-06 | – | 1.8E-06 | – | 2.4E-06 | – | 0.1 |
| 16 | 2.1E-07 | 3.0 | 2.4E-07 | 2.9 | 3.3E-07 | 2.9 | 0.3 |
| 32 | 2.7E-08 | 3.0 | 3.0E-08 | 3.0 | 4.3E-08 | 3.0 | 1.1 |
| 64 | 3.4E-09 | 3.0 | 3.8E-09 | 3.0 | 5.3E-09 | 3.0 | 5.1 |
| 128 | 4.3E-10 | 3.0 | 4.7E-10 | 3.0 | 6.7E-10 | 3.0 | 28.4 |
| ADER-4 | | | | | | | |
| 8 | 6.8E-07 | – | 7.3E-07 | – | 1.0E-06 | – | 0.3 |
| 16 | 2.3E-08 | 4.9 | 2.8E-08 | 4.7 | 4.8E-08 | 4.4 | 0.9 |
| 32 | 1.5E-09 | 3.9 | 1.8E-09 | 4.0 | 4.3E-09 | 3.5 | 4.2 |
| 64 | 9.7E-11 | 4.0 | 1.1E-10 | 4.1 | 2.4E-10 | 4.2 | 18.4 |
| 128 | 5.7E-12 | 4.1 | 6.3E-12 | 4.1 | 1.3E-11 | 4.2 | 83.1 |
| ADER-5 | | | | | | | |
| 8 | 7.5E-08 | – | 9.5E-08 | – | 1.8E-07 | – | 0.5 |
| 16 | 4.4E-09 | 4.1 | 5.8E-09 | 4.0 | 1.2E-08 | 3.9 | 1.9 |
| 32 | 1.1E-10 | 5.3 | 1.3E-10 | 5.4 | 2.8E-10 | 5.4 | 7.6 |
| 64 | 4.7E-12 | 4.5 | 5.2E-12 | 4.7 | 9.6E-12 | 4.9 | 31.0 |
| 128 | 4.9E-13 | 3.3 | 7.7E-13 | 2.8 | 2.9E-12 | 1.7 | 129.1 |

into a smooth, high-frequency fluctuation in cross-sectional area A . As time evolves, the shock runs into the fluctuation, which spreads upstream. The numerical solution is evaluated at output time $T_{\text{out}} = 0.03$ s. Figures 30, 31, 32, 33 and 34 show the results obtained with ADER schemes of orders 2nd to 5th, with different reconstruction methods. This problem does not have an exact solution. We therefore compute a reference solution using the fifth order ADER scheme with WENO reconstruction on a mesh of 5 000 computational cells. In all cases the reference solution is shown in black and the numerical solution on a mesh of 1 000 cells is shown with blue dots. For orders of accuracy 2nd to 5th, Figs. 30, 31, 32, 33 and 34 compare the numerical solution with the reference solution in the portion of domain [25, 50], showing on top the shock wave and below the domain portion [28, 40] displaying the **physical oscillations** upstream of the shock.

Table 19 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **AENO** reconstruction with $\epsilon^2 = 0.5$. Rates are calculated for the cross-sectional area A . Computational parameters are: $T_{\text{out}} = 0.1$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 4.2E-06 | – | 4.5E-06 | – | 6.0E-06 | – | 0 |
| 16 | 1.0E-06 | 2.0 | 1.1E-06 | 2.0 | 1.6E-06 | 1.9 | 0.1 |
| 32 | 2.6E-07 | 2.0 | 2.9E-07 | 2.0 | 4.7E-07 | 1.8 | 0.4 |
| 64 | 6.4E-08 | 2.0 | 7.2E-08 | 2.0 | 1.2E-07 | 2.0 | 2.3 |
| 128 | 1.6E-08 | 2.0 | 1.8E-08 | 2.0 | 2.8E-08 | 2.1 | 15.6 |
| ADER-3 | | | | | | | |
| 8 | 5.7E-08 | – | 6.1E-08 | – | 8.2E-08 | – | 0.1 |
| 16 | 1.3E-08 | 2.1 | 1.4E-08 | 2.1 | 2.1E-08 | 2.0 | 0.3 |
| 32 | 2.0E-09 | 2.7 | 2.2E-09 | 2.7 | 3.0E-09 | 2.8 | 1.2 |
| 64 | 3.1E-10 | 2.7 | 3.3E-10 | 2.7 | 4.4E-10 | 2.8 | 5.4 |
| 128 | 4.2E-11 | 2.9 | 4.5E-11 | 2.9 | 5.8E-11 | 2.9 | 28.2 |
| ADER-4 | | | | | | | |
| 8 | 8.4E-08 | – | 9.7E-08 | – | 1.4E-07 | – | 0.2 |
| 16 | 4.2E-09 | 4.3 | 4.9E-09 | 4.3 | 8.4E-09 | 4.0 | 0.8 |
| 32 | 2.5E-10 | 4.1 | 3.1E-10 | 4.0 | 6.8E-10 | 3.6 | 2.7 |
| 64 | 1.7E-11 | 3.9 | 1.8E-11 | 4.1 | 3.3E-11 | 4.4 | 11.7 |
| 128 | 9.7E-13 | 4.1 | 1.1E-12 | 4.0 | 2.0E-12 | 4.0 | 60.5 |
| ADER-5 | | | | | | | |
| 8 | 4.6E-09 | – | 4.9E-09 | – | 6.4E-09 | – | 0.5 |
| 16 | 1.0E-10 | 5.5 | 1.1E-10 | 5.4 | 1.7E-10 | 5.3 | 2.0 |
| 32 | 3.5E-12 | 4.9 | 3.8E-12 | 4.9 | 6.2E-12 | 4.8 | 8.0 |
| 64 | 1.1E-13 | 5.0 | 1.3E-13 | 4.9 | 2.8E-13 | 4.4 | 31.9 |
| 128 | 3.0E-15 | 5.2 | 3.8E-15 | 5.1 | 8.4E-15 | 5.1 | 133.7 |

As the order of the numerical method increases, the resolution of the physical oscillations of the solution is improved. For second order methods and all the reconstructions techniques, only two oscillations just behind the shock are captured. The WENO reconstruction method shows a small advantage over the other reconstruction schemes, for the second-order case. However, for the fourth and fifth order cases, the WENO numerical solution presents spurious oscillations and over/under shoots. For the third-order case, AENO shows a small advantage over the other methods. For the fourth-order case, AENO is comparable to ENO and m-ENO; both AENO and m-AENO show a small advantage over the remaining methods. For the fifth-order case, all methods are comparable, with the exception of WENO, which shows the worse performance, due to the spurious oscillations and over/under shoots.

Table 20 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **AENO** reconstruction with $\epsilon^2 = 0.5$. Rates are calculated for the flow Au . Computational parameters are: $T_{\text{out}} = 0.1$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 4.9E-06 | – | 5.8E-06 | – | 8.2E-06 | – | 0 |
| 16 | 1.0E-06 | 2.3 | 1.2E-06 | 2.3 | 2.0E-06 | 2.0 | 0.1 |
| 32 | 2.0E-07 | 2.3 | 2.3E-07 | 2.4 | 3.3E-07 | 2.6 | 0.4 |
| 64 | 5.8E-08 | 1.8 | 6.6E-08 | 1.8 | 9.3E-08 | 1.8 | 2.3 |
| 128 | 1.5E-08 | 2.0 | 1.7E-08 | 2.0 | 2.5E-08 | 1.9 | 15.6 |
| ADER-3 | | | | | | | |
| 8 | 2.3E-06 | – | 2.7E-06 | – | 3.7E-06 | – | 0.1 |
| 16 | 3.5E-07 | 2.7 | 4.1E-07 | 2.7 | 5.9E-07 | 2.7 | 0.3 |
| 32 | 4.8E-08 | 2.9 | 5.3E-08 | 2.9 | 7.6E-08 | 3.0 | 1.2 |
| 64 | 6.2E-09 | 3.0 | 6.8E-09 | 3.0 | 9.4E-09 | 3.0 | 5.4 |
| 128 | 7.8E-10 | 3.0 | 8.6E-10 | 3.0 | 1.2E-09 | 3.0 | 28.2 |
| ADER-4 | | | | | | | |
| 8 | 1.8E-07 | – | 2.0E-07 | – | 3.1E-07 | – | 0.2 |
| 16 | 4.5E-09 | 5.3 | 5.1E-09 | 5.3 | 9.1E-09 | 5.1 | 0.8 |
| 32 | 3.1E-10 | 3.9 | 4.1E-10 | 3.6 | 1.1E-09 | 3.1 | 2.7 |
| 64 | 1.4E-11 | 4.5 | 1.9E-11 | 4.4 | 6.3E-11 | 4.1 | 11.7 |
| 128 | 7.5E-13 | 4.2 | 9.7E-13 | 4.3 | 2.5E-12 | 4.7 | 60.5 |
| ADER-5 | | | | | | | |
| 8 | 8.1E-08 | – | 8.1E-08 | – | 9.2E-08 | – | 0.5 |
| 16 | 1.4E-09 | 5.9 | 1.4E-09 | 5.9 | 1.7E-09 | 5.8 | 2.0 |
| 32 | 2.6E-11 | 5.7 | 3.1E-11 | 5.5 | 5.7E-11 | 4.9 | 8.0 |
| 64 | 6.5E-13 | 5.3 | 8.4E-13 | 5.2 | 1.9E-12 | 4.9 | 31.9 |
| 128 | 1.7E-14 | 5.2 | 2.4E-14 | 5.2 | 5.9E-14 | 5.0 | 133.7 |

Table 21 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **m-AENO** reconstruction with $\epsilon^2 = 0.5$. Rates are calculated for the cross-sectional area A . Computational parameters are: $T_{\text{out}} = 0.1$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 4.2E-06 | – | 4.5E-06 | – | 6.0E-06 | – | 0 |
| 16 | 1.0E-06 | 2.0 | 1.1E-06 | 2.0 | 1.6E-06 | 1.9 | 0.1 |
| 32 | 2.6E-07 | 2.0 | 2.9E-07 | 2.0 | 4.7E-07 | 1.8 | 0.4 |
| 64 | 6.4E-08 | 2.0 | 7.2E-08 | 2.0 | 1.2E-07 | 2.0 | 2.3 |
| 128 | 1.6E-08 | 2.0 | 1.8E-08 | 2.0 | 2.8E-08 | 2.1 | 15.6 |
| ADER-3 | | | | | | | |
| 8 | 5.7E-08 | – | 6.1E-08 | – | 8.2E-08 | – | 0.1 |
| 16 | 1.3E-08 | 2.1 | 1.4E-08 | 2.1 | 2.1E-08 | 2.0 | 0.3 |
| 32 | 2.0E-09 | 2.7 | 2.2E-09 | 2.7 | 3.0E-09 | 2.8 | 1.2 |
| 64 | 3.1E-10 | 2.7 | 3.3E-10 | 2.7 | 4.4E-10 | 2.8 | 5.4 |
| 128 | 4.2E-11 | 2.9 | 4.5E-11 | 2.9 | 5.8E-11 | 2.9 | 28.2 |
| ADER-4 | | | | | | | |
| 8 | 7.8E-08 | – | 9.2E-08 | – | 1.3E-07 | – | 0.2 |
| 16 | 4.0E-09 | 4.3 | 4.5E-09 | 4.4 | 7.0E-09 | 4.2 | 0.7 |
| 32 | 2.4E-10 | 4.0 | 2.7E-10 | 4.0 | 4.3E-10 | 4.0 | 3.1 |
| 64 | 1.5E-11 | 4.0 | 1.7E-11 | 4.0 | 3.3E-11 | 3.7 | 12.5 |
| 128 | 9.5E-13 | 4.0 | 1.1E-12 | 4.0 | 1.7E-12 | 4.3 | 62.8 |
| ADER-5 | | | | | | | |
| 8 | 5.5E-09 | – | 6.1E-09 | – | 1.0E-08 | – | 0.5 |
| 16 | 2.2E-10 | 4.6 | 2.5E-10 | 4.6 | 4.4E-10 | 4.6 | 2.0 |
| 32 | 5.4E-12 | 5.4 | 5.7E-12 | 5.4 | 8.4E-12 | 5.7 | 9.5 |
| 64 | 1.5E-13 | 5.2 | 1.7E-13 | 5.1 | 2.8E-13 | 4.9 | 35.1 |
| 128 | 4.6E-15 | 5.0 | 5.1E-15 | 5.0 | 8.3E-15 | 5.1 | 139.5 |

Table 22 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **m-AENO** reconstruction with $\epsilon^2 = 0.5$. Rates are calculated for the flow Au . Computational parameters are: $T_{out} = 0.1$ and $C_{cfl} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 4.9E-06 | – | 5.8E-06 | – | 8.2E-06 | – | 0 |
| 16 | 1.0E-06 | 2.3 | 1.2E-06 | 2.3 | 2.0E-06 | 2.0 | 0.1 |
| 32 | 2.0E-07 | 2.3 | 2.3E-07 | 2.4 | 3.3E-07 | 2.6 | 0.4 |
| 64 | 5.8E-08 | 1.8 | 6.6E-08 | 1.8 | 9.3E-08 | 1.8 | 2.3 |
| 128 | 1.5E-08 | 2.0 | 1.7E-08 | 2.0 | 2.5E-08 | 1.9 | 15.6 |
| ADER-3 | | | | | | | |
| 8 | 2.3E-06 | – | 2.7E-06 | – | 3.7E-06 | – | 0.1 |
| 16 | 3.5E-07 | 2.7 | 4.1E-07 | 2.7 | 5.9E-07 | 2.7 | 0.3 |
| 32 | 4.8E-08 | 2.9 | 5.3E-08 | 2.9 | 7.6E-08 | 3.0 | 1.2 |
| 64 | 6.2E-09 | 3.0 | 6.8E-09 | 3.0 | 9.4E-09 | 3.0 | 5.4 |
| 128 | 7.8E-10 | 3.0 | 8.6E-10 | 3.0 | 1.2E-09 | 3.0 | 28.2 |
| ADER-4 | | | | | | | |
| 8 | 1.7E-07 | – | 2.0E-07 | – | 3.1E-07 | – | 0.2 |
| 16 | 3.2E-09 | 5.7 | 3.8E-09 | 5.7 | 6.3E-09 | 5.6 | 0.7 |
| 32 | 2.3E-10 | 3.8 | 2.8E-10 | 3.8 | 6.2E-10 | 3.3 | 3.1 |
| 64 | 1.0E-11 | 4.5 | 1.4E-11 | 4.3 | 6.2E-11 | 3.3 | 12.5 |
| 128 | 4.1E-13 | 4.6 | 4.8E-13 | 4.9 | 9.5E-13 | 6.0 | 62.8 |
| ADER-5 | | | | | | | |
| 8 | 6.8E-08 | – | 7.6E-08 | – | 9.5E-08 | – | 0.5 |
| 16 | 9.2E-10 | 6.2 | 1.0E-09 | 6.2 | 1.6E-09 | 5.9 | 2.0 |
| 32 | 1.8E-11 | 5.6 | 2.1E-11 | 5.6 | 3.8E-11 | 5.4 | 9.5 |
| 64 | 4.3E-13 | 5.4 | 4.7E-13 | 5.5 | 7.8E-13 | 5.6 | 35.1 |
| 128 | 1.3E-14 | 5.0 | 1.5E-14 | 5.0 | 2.4E-14 | 5.1 | 139.5 |

Table 23 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **WENO** reconstruction. Rates are calculated for the cross-sectional area A . Computational parameters are: $T_{\text{out}} = 0.1$ and $C_{\text{eff}} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 4.2E-06 | – | 4.5E-06 | – | 6.0E-06 | – | 0 |
| 16 | 1.0E-06 | 2.0 | 1.2E-06 | 2.0 | 1.8E-06 | 1.8 | 0.1 |
| 32 | 2.6E-07 | 2.0 | 2.9E-07 | 2.0 | 4.3E-07 | 2.0 | 0.5 |
| 64 | 6.4E-08 | 2.0 | 7.1E-08 | 2.0 | 1.2E-07 | 1.9 | 2.4 |
| 128 | 1.6E-08 | 2.0 | 1.8E-08 | 2.0 | 2.6E-08 | 2.1 | 14.9 |
| ADER-3 | | | | | | | |
| 8 | 5.0E-08 | – | 5.5E-08 | – | 8.0E-08 | – | 0.1 |
| 16 | 1.1E-08 | 2.2 | 1.2E-08 | 2.2 | 1.8E-08 | 2.2 | 0.3 |
| 32 | 9.6E-10 | 3.5 | 1.1E-09 | 3.5 | 1.6E-09 | 3.5 | 1.1 |
| 64 | 1.4E-10 | 2.8 | 1.6E-10 | 2.8 | 2.3E-10 | 2.7 | 5.2 |
| 128 | 1.5E-11 | 3.3 | 1.7E-11 | 3.3 | 2.5E-11 | 3.2 | 26.6 |
| ADER-4 | | | | | | | |
| 8 | 7.4E-08 | – | 8.5E-08 | – | 1.2E-07 | – | 0.2 |
| 16 | 3.9E-09 | 4.2 | 4.3E-09 | 4.3 | 6.3E-09 | 4.3 | 0.7 |
| 32 | 2.4E-10 | 4.0 | 2.7E-10 | 4.0 | 4.0E-10 | 4.0 | 2.8 |
| 64 | 1.5E-11 | 4.0 | 1.7E-11 | 4.0 | 2.6E-11 | 4.0 | 12.0 |
| 128 | 9.6E-13 | 4.0 | 1.1E-12 | 4.0 | 1.6E-12 | 4.0 | 59.0 |
| ADER-5 | | | | | | | |
| 8 | 3.6E-09 | – | 3.9E-09 | – | 5.5E-09 | – | 0.5 |
| 16 | 1.4E-10 | 4.7 | 1.5E-10 | 4.7 | 2.5E-10 | 4.5 | 2.0 |
| 32 | 5.6E-12 | 4.6 | 6.2E-12 | 4.6 | 1.0E-11 | 4.6 | 7.8 |
| 64 | 2.6E-13 | 4.4 | 2.9E-13 | 4.4 | 4.6E-13 | 4.5 | 31.4 |
| 128 | 8.3E-15 | 5.0 | 9.2E-15 | 5.0 | 1.5E-14 | 5.0 | 132.1 |

Table 24 Convergence-rate study for the blood flow equations. ADER schemes from 2nd to 5th order of accuracy with the **WENO** reconstruction. Rates are calculated for the flow Au . Computational parameters are: $T_{out} = 0.1$ and $C_{eff} = 0.9$

| Mesh | L_1 -err | L_1 -ord | L_2 -err | L_2 -ord | L_∞ -err | L_∞ -ord | CPU/s |
|--------|------------|------------|------------|------------|-----------------|-----------------|-------|
| ADER-2 | | | | | | | |
| 8 | 4.3E-06 | – | 5.8E-06 | – | 8.4E-06 | – | 0 |
| 16 | 7.7E-07 | 2.5 | 8.8E-07 | 2.7 | 1.4E-06 | 2.6 | 0.1 |
| 32 | 2.1E-07 | 1.9 | 2.4E-07 | 1.9 | 3.4E-07 | 2.0 | 0.5 |
| 64 | 6.4E-08 | 1.7 | 7.1E-08 | 1.8 | 1.0E-07 | 1.7 | 2.4 |
| 128 | 1.6E-08 | 2.0 | 1.8E-08 | 2.0 | 2.5E-08 | 2.0 | 14.9 |
| ADER-3 | | | | | | | |
| 8 | 1.7E-06 | – | 1.8E-06 | – | 2.4E-06 | – | 0.1 |
| 16 | 2.1E-07 | 3.0 | 2.4E-07 | 2.9 | 3.3E-07 | 2.9 | 0.3 |
| 32 | 2.7E-08 | 3.0 | 3.0E-08 | 3.0 | 4.3E-08 | 3.0 | 1.1 |
| 64 | 3.4E-09 | 3.0 | 3.8E-09 | 3.0 | 5.3E-09 | 3.0 | 5.2 |
| 128 | 4.3E-10 | 3.0 | 4.7E-10 | 3.0 | 6.7E-10 | 3.0 | 26.6 |
| ADER-4 | | | | | | | |
| 8 | 4.5E-07 | – | 5.1E-07 | – | 7.6E-07 | – | 0.2 |
| 16 | 2.5E-08 | 4.2 | 2.8E-08 | 4.2 | 4.0E-08 | 4.2 | 0.7 |
| 32 | 1.6E-09 | 4.0 | 1.7E-09 | 4.0 | 2.5E-09 | 4.0 | 2.8 |
| 64 | 9.6E-11 | 4.0 | 1.1E-10 | 4.0 | 1.5E-10 | 4.0 | 12.0 |
| 128 | 5.9E-12 | 4.0 | 6.6E-12 | 4.0 | 9.3E-12 | 4.0 | 59.0 |
| ADER-5 | | | | | | | |
| 8 | 1.7E-07 | – | 1.8E-07 | – | 2.4E-07 | – | 0.5 |
| 16 | 5.2E-09 | 5.0 | 5.8E-09 | 5.0 | 8.0E-09 | 4.9 | 2.0 |
| 32 | 1.6E-10 | 5.0 | 1.8E-10 | 5.0 | 2.5E-10 | 5.0 | 7.8 |
| 64 | 4.9E-12 | 5.0 | 5.4E-12 | 5.0 | 7.7E-12 | 5.0 | 31.4 |
| 128 | 1.5E-13 | 5.0 | 1.7E-13 | 5.0 | 2.4E-13 | 5.0 | 132.1 |

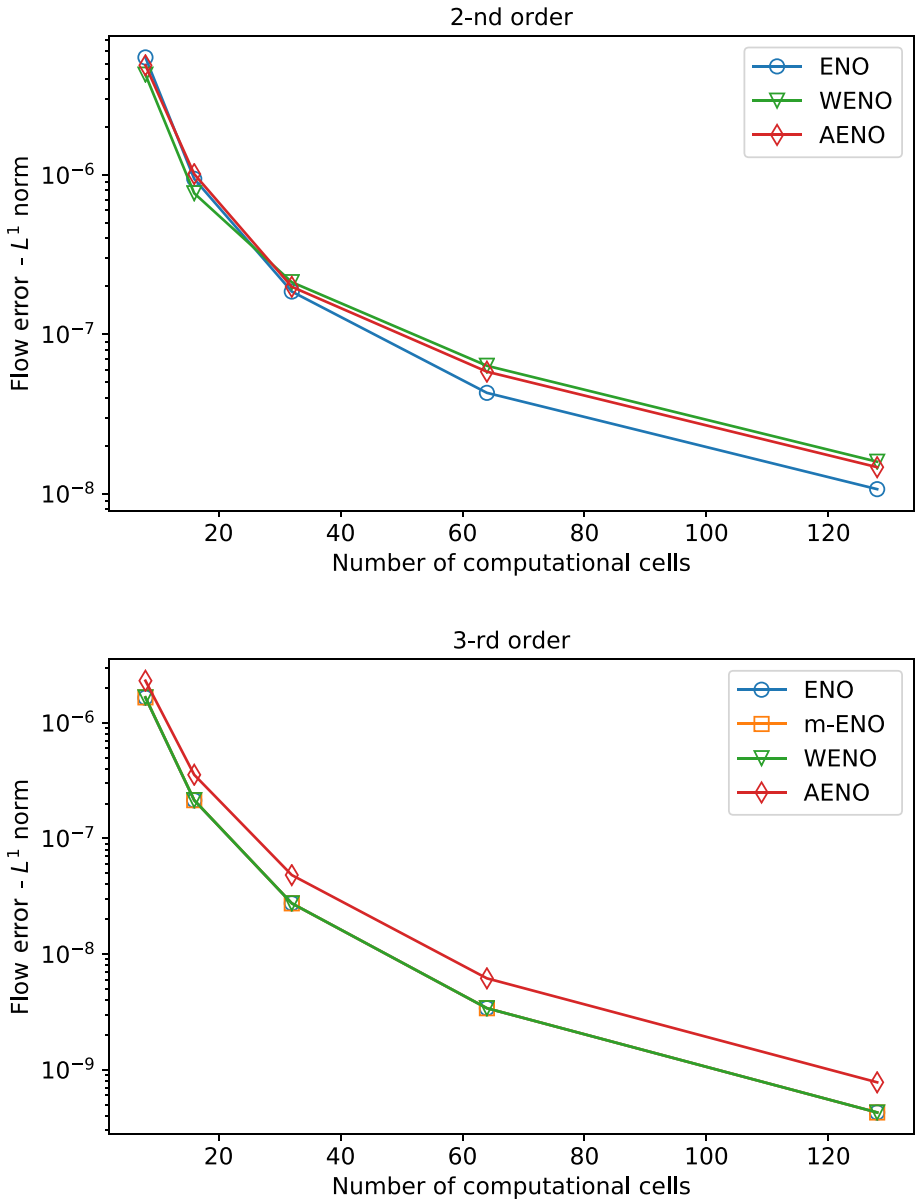


Fig. 28 L_1 -errors for the blood flow equations. Computed errors for flow Au for the 2nd and 3rd order ADER schemes with **ENO**, **m-ENO**, **AENO** and **WENO**

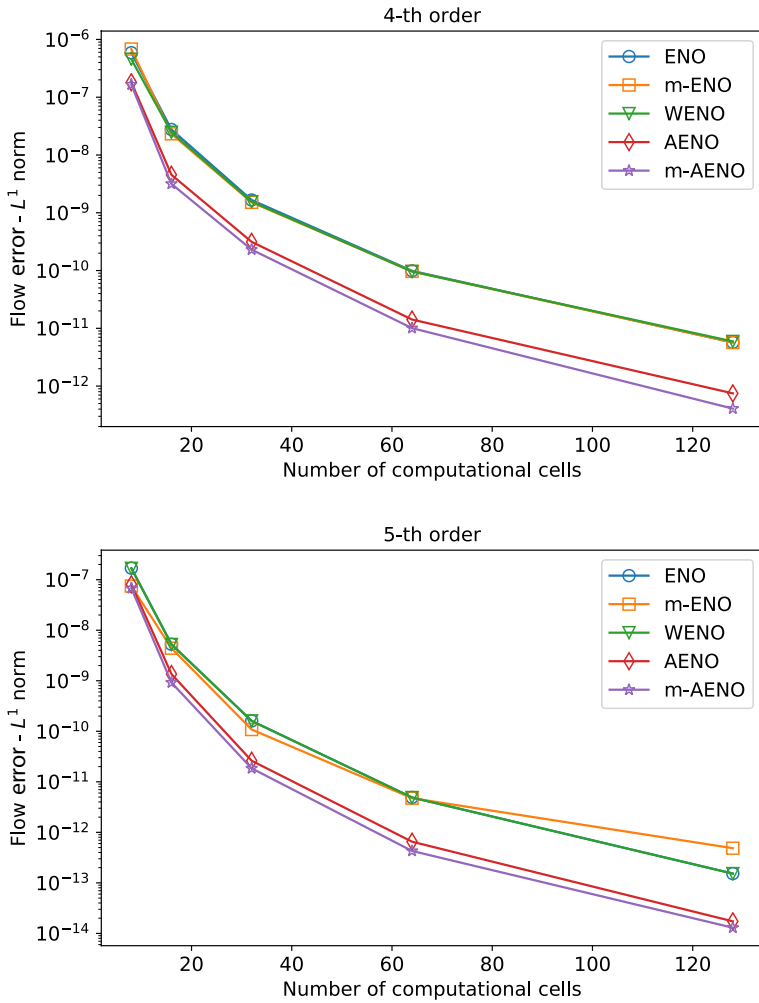


Fig. 29 L_1 -errors for the blood flow equations. Computed errors for flow Au for the 4th and 5th order ADER schemes with ENO, m-ENO, AENO, m-AENO and WENO

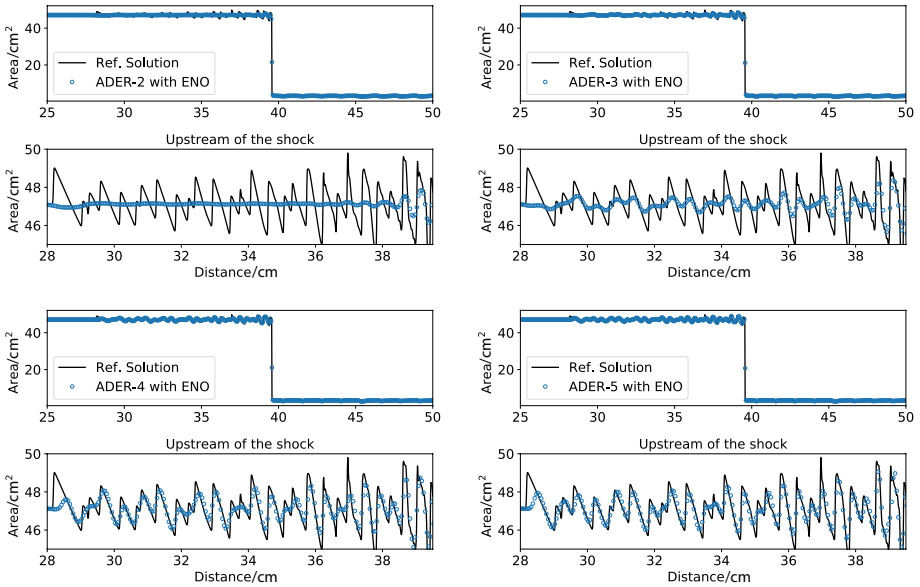


Fig. 30 Shock/turbulence interaction test problem: ADER schemes from 2nd to 5th order with **ENO** reconstruction

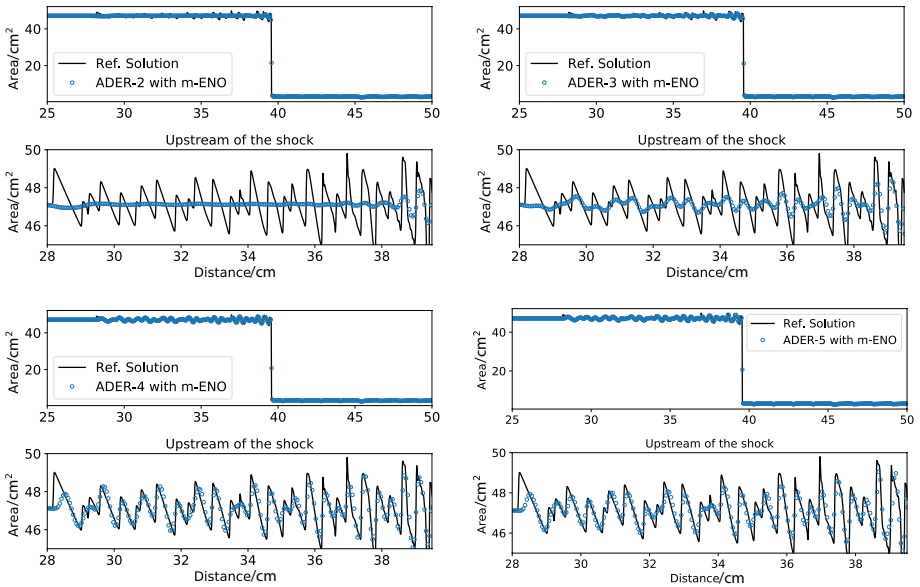


Fig. 31 Shock/turbulence interaction test problem: ADER schemes from 2nd to 5th order with **m-ENO** reconstruction

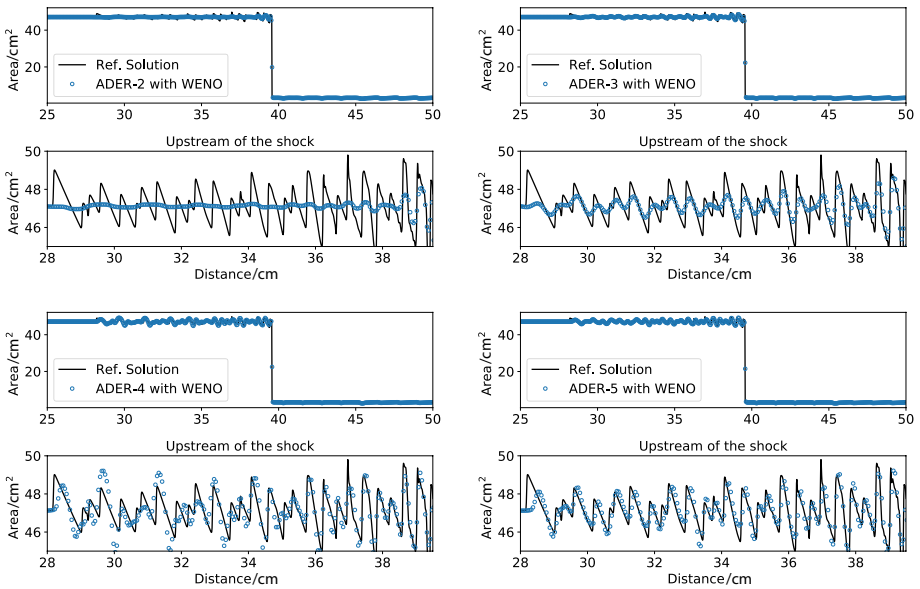


Fig. 32 Shock/turbulence interaction test problem: ADER schemes from 2nd to 5th order with WENO reconstruction

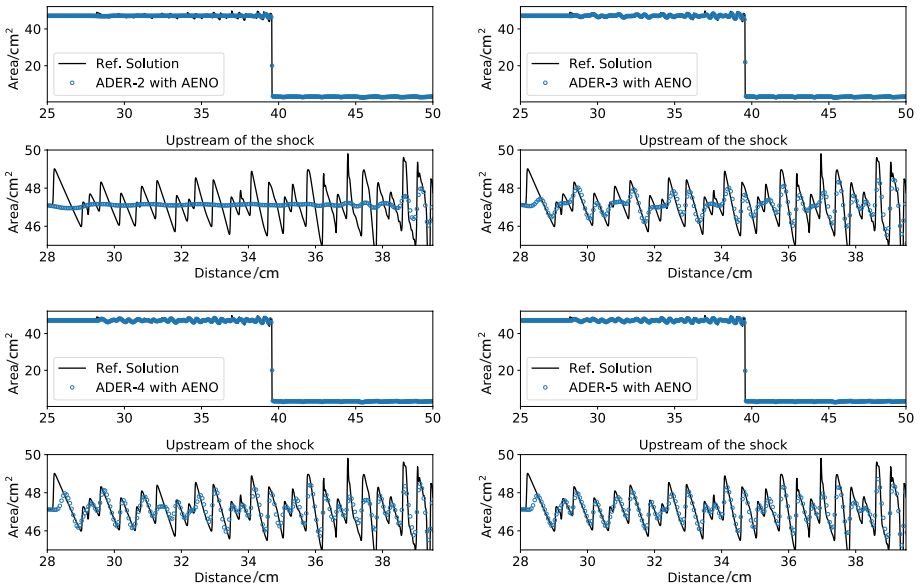


Fig. 33 Shock/turbulence interaction test problem: ADER schemes from 2nd to 5th order with AENO reconstruction

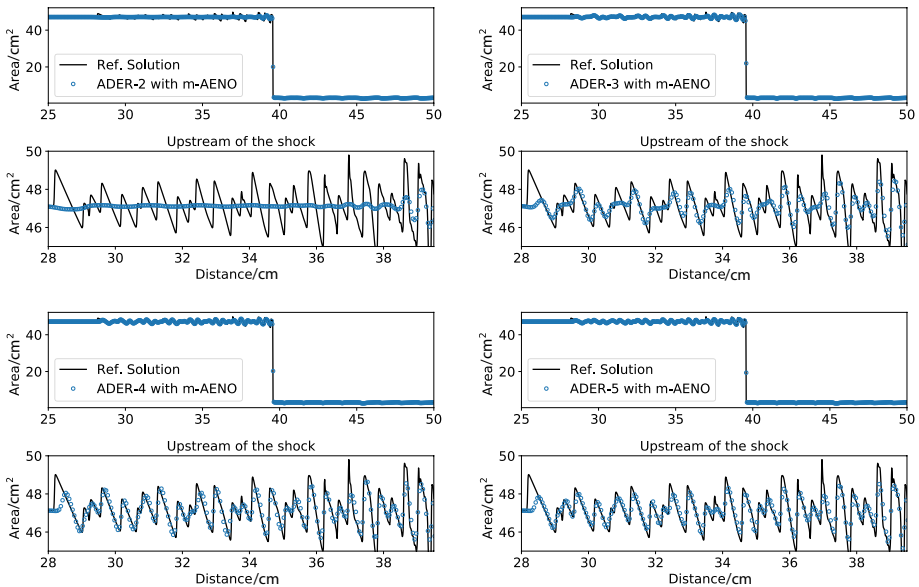


Fig. 34 Shock interaction test problem: ADER schemes from 2nd to 5th order with **m-AENO** reconstruction

7 Conclusions

In this paper, we have presented a novel, non-linear spatial reconstruction scheme called AENO, with two variants. The method is akin to the established methods ENO, m-ENO and WENO. All reconstruction schemes have been implemented in conjunction with the fully discrete ADER scheme, from second to fifth order of accuracy, in both space and time, along with the Toro-Titarev solver for the generalized Riemann problem. The methods have been tested thoroughly for the linear advection equation and for a non-linear hyperbolic system, namely the blood flow equations. The assessment of the methods has been through comparison of profiles with exact solutions, convergence rate studies and efficiency studies. Overall, the results of the new AENO reconstruction methods are comparable to the ENO, m-ENO and WENO. However, AENO shows a distinctive advantage over ENO for long-time evolution problems; this is more obvious for second- and third-order methods, but will also be apparent for high-order methods on coarse meshes. Crucially, AENO turns out to be the most efficient of all schemes tested. Estimates reveal AENO to be up to one order of magnitude more efficient than WENO, for sought small errors. Desirable future developments are implementation and assessment of the AENO schemes in conjunction with semi-discrete methods in one and multiple space dimensions.

Funding Open access funding provided by University of Helsinki including Helsinki University Central Hospital.

Compliance with Ethical Standards

Conflicts of Interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Ben-Artzi, M., Falcovitz, J.: A second order Godunov-type scheme for compressible fluid dynamics. *J. Comput. Phys.* **55**, 1–32 (1984)
2. Castro, C.E., Toro, E.F.: Solvers for the high-order Riemann problem for hyperbolic balance laws. *J. Comput. Phys.* **227**, 2481–2513 (2008)
3. Dematté, R., Titarev, V.A., Motecinos, G.I., Toro, E.F.: ADER methods for hyperbolic equations with a time-reconstruction solver for the generalized Riemann problem. The scalar case. *Commun. Appl. Math. Comput.* **2**, 369–402 (2020)
4. Dumbser, M., Enaux, C., Toro, E.F.: Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *J. Comput. Phys.* **227**(8), 3971–4001 (2008)
5. Dumbser, M., Käser, M.: Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *J. Comput. Phys.* **221**(2), 693–723 (2007)
6. Dumbser, M., Käser, M., Titarev, V.A., Toro, E.F.: Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *J. Comput. Phys.* **226**(8), 204–243 (2007)
7. Dumbser, M., Munz, C.D.: ADER discontinuous Galerkin schemes for aeroacoustics. *Comptes Rendus Mécanique* **333**, 683–687 (2005)
8. Godunov, S.K.: A finite difference method for the computation of discontinuous solutions of the equations of fluid dynamics. *Mat. Sb.* **47**, 357–393 (1959)
9. Götz, C.R., Dumbser, M.: A novel solver for the generalized Riemann problem based on a simplified LeFloch-Raviart expansion and a local space-time discontinuous Galerkin formulation. *J. Sci. Comput.* (2016). <https://doi.org/10.1007/s10915-016-0218-5>
10. Götz, C.R., Iske, A.: Approximate solutions of generalized Riemann problems for nonlinear systems of hyperbolic conservation laws. *Math. Comput.* **85**, 35–62 (2016)
11. Harten, A.: High resolution schemes for hyperbolic conservation laws. *J. Comput. Phys.* **49**, 357–393 (1983)
12. Harten, A., Engquist, B., Osher, S., Chakravarthy, S.R.: Uniformly high order accuracy essentially non-oscillatory schemes, III. *J. Comput. Phys.* **71**, 231–303 (1987)
13. Harten, A., Osher, S.: Uniformly high-order accurate nonoscillatory schemes, I. *SIAM J. Numer. Anal.* **24**(2), 279–309 (1987)
14. Jiang, G.S., Shu, C.-W.: Efficient Implementation of Weighted ENO Schemes. Technical Report ICASE 95-73. NASA Langley Research Center, Hampton (1995)
15. Jiang, G.S., Shu, C.-W.: Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* **126**, 202–228 (1996)
16. Kolgan, V.P.: Application of the principle of minimum derivatives to the construction of difference schemes for computing discontinuous solutions of gas dynamics. *Uch. Zap. TsaGI, Russia* **3**(6), 68–77 (1972) (in Russian)
17. Laney, C.B.: *Computational Gasdynamics*. Cambridge University Press, Cambridge (1998)
18. Le Floch, P.: Entropy weak solutions to nonlinear hyperbolic systems under nonconservative form. *Commun. Part. Differ. Equ.* **13**(6), 669–727 (1988)
19. Le Floch, P., Raviart, P.A.: An asymptotic expansion for the solution of the generalized Riemann problem. Part I: general theory. *Ann. Inst. Henri Poincaré. Analyse non Linéaire* **5**(2), 179–207 (1988)
20. Montecinos, G.I., Castro, C.E., Dumbser, M., Toro, E.F.: Comparison of solvers for the generalized Riemann problem for hyperbolic systems with source terms. *J. Comput. Phys.* **231**, 6472–6494 (2012)
21. Montecinos, G.I., Toro, E.F.: Reformulations for general advection-diffusion-reaction equations and locally implicit ADER schemes. *J. Comput. Phys.* **275**, 415–442 (2014)
22. Müller, Lucas O., Toro, Eleuterio F.: A global multi-scale model for the human circulation with emphasis on the venous system. *Int. J. Numer. Methods Biomed. Eng.* **30**(7), 681–725 (2014)

23. Roe, P.L.: Numerical Algorithms for the Linear Wave Equation. Technical Report 81047. Royal Aircraft Establishment, Bedford (1981)
24. Santaca. High-order ADER schemes for hyperbolic equations. Selected topics. Master Thesis, Department of Mathematics, University of Trento (2017)
25. Schwartzkopff, T., Munz, C.D., Toro, E.F.: ADER: high-order approach for linear hyperbolic systems in 2D. *J. Sci. Comput.* **17**, 231–240 (2002)
26. Shu, C.-W.: Numerical experiments on the accuracy of ENO and modified ENO schemes. *J. Sci. Comput.* **5**, 127–149 (1990)
27. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.* **77**, 439–471 (1988)
28. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. *J. Comput. Phys.* **83**, 32–78 (1989)
29. Sweby, P.K.: High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM J. Numer. Anal.* **21**, 995–1011 (1984)
30. Titarev, V.A., Toro, E.F.: ADER: arbitrary high order Godunov approach. *J. Sci. Comput.* **17**, 609–618 (2002)
31. Titarev, V.A., Toro, E.F.: Finite volume WENO schemes for three-dimensional conservation laws. *J. Comput. Phys.* **201**(1), 238–260 (2004)
32. Titarev, V.A., Toro, E.F.: ENO and WENO schemes based on upwind and centred TVD fluxes. *Comput. Fluids* **3**, 705–720 (2005)
33. Toro, E.F.: Riemann Solvers and Numerical Methods for Fluid Dynamics. Springer, Berlin (1997)
34. Toro, E.F.: Shock-Capturing Methods for Free-Surface Shallow Flows. Wiley, New York (2001)
35. Toro, E.F.: Riemann Solvers and Numerical Methods for Fluid Dynamics, 3rd edn. Springer, Berlin (2009)
36. Toro, E.F.: The ADER path to high-order Godunov methods. In: Continuum Mechanics, Applied Mathematics and Scientific Computing: Godunov’s Legacy—A Liber Amicorum to Professor Godunov, pp. 359–366. Springer, Berlin (2020)
37. Toro, E.F., Billett, S.J.: Centred TVD schemes for hyperbolic conservation laws. *IMA J. Numer. Anal.* **20**, 47–79 (2000)
38. Toro, E.F., Millington, R.C., Nejad, L.A.M.: Towards very high-order Godunov schemes. In: Toro, E.F. (ed.) Godunov Methods: Theory and Applications. Edited Review, pp. 905–937. Kluwer Academic/Plenum Publishers (2001)
39. Toro, E.F., Siviglia, A.: Flow in collapsible tubes with discontinuous mechanical properties: mathematical model and exact solutions. *Commun. Comput. Phys.* **13**(2), 361–385 (2013)
40. Toro, E.F., Montecinos, G.I.: Implicit, semi-analytical solution of the generalised Riemann problem for stiff hyperbolic balance laws. *J. Comput. Phys.* **303**, 146–172 (2015)
41. Toro, E.F., Titarev, V.A.: Solution of the generalised Riemann problem for advection-reaction equations. *Proc. R. Soc. Lond. A* **458**, 271–281 (2002)
42. Toro, E.F., Titarev, V.A.: TVD fluxes for the high-order ADER schemes. *J. Sci. Comput.* **24**, 285–309 (2005)
43. van Leer, B.: Towards the ultimate conservative difference scheme I. The quest for monotonicity. *Lect. Notes Phys.* **18**, 163–168 (1973)
44. van Leer, B.: On the relation between the upwind-differencing schemes of Godunov, Enguist-Osher and Roe. *SIAM J. Sci. Stat. Comput.* **5**(1), 1–20 (1985)
45. Woodward, P., Colella, P.: The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.* **54**, 115–173 (1984)