RECONSTRUCTION BASED A POSTERIORI ESTIMATES FOR SOME FINITE DIFFERENCE SCHEMES

TRISTAN PRYER AND GEORGIOS SIALOUNAS

Abstract. In this work we present a framework for the construction of reliable a posteriori estimates for classes of finite difference schemes approximating systems of hyperbolic conservation laws. We design appropriate reconstructions of the finite difference solution which allows us to utilise the relative entropy stability framework. This, in turn, results in a fully computable a posteriori bound enabling error control.

We showcase our results with thorough numerical testing of certain well used schemes approximating hyperbolic problems including classical Lax-Friedrich and Lax-Wendroff methods as well as more recent ENO and WENO schemes.

1. Introduction

Hyperbolic conservation laws ubiquitously arise in many physical applications. Inviscid compressible flows are well described by Euler's equations which have meteorological applications, for example. A major difficulty in designing numerical schemes for hyperbolic conservation laws is that they can form shocks in finite time. There has been considerable activity in this area based on various numerical techniques, such as finite difference, volume and element approaches [CCL95, KR94, L$^+$02]. The formation and tracking of these discontinuities is a significant challenge.

A substantial body of work has accumulated over the years in applications of FD schemes for hyperbolic problems, resulting in several noteworthy contributions (see [LeV92], [JT97] for overviews). Early examples include Godunov's scheme ([God59]), the Lax-Friedrichs (LxF) scheme ([Lax54]), the two-step Lax-Wendroff scheme (see the recent work of [LVW21]), as well as the works of van Leer (see ([VL73], [VL74], [VL77a], [VL77b] and [VL79])). The works of [NT90], who use the LxF solver in conjunction MUSCL-type interpolants to compensate for the excessive LxF viscosity are also of note. Two classes of FD schemes that are of particular importance in the context of hyperbolic conservation laws are the Essentially Non-Oscillatory schemes (see [HEOC87], [SO88], [SO89]) and the Weighted ENO schemes ([LOC94], [JS96], [JT98]; see [Shu98] and references therein for an overview). ENO and WENO schemes combine high orders of approximation in smooth regions and non-oscillatory behaviour in the vicinity of discontinuities.

A posteriori error estimation aims to provide the user with local computational control over the error incurred in approximating a partial differential equation (PDE) with a given numerical scheme. A posteriori error estimates for hyperbolic problems have received considerable attention, particularly for discontinuous Galerkin (dG) finite element methods [Joh90, JS95, DMO07, GMP15, GP17, DGPR19, AO11, Ver13] and finite volume (FV) methods [CCL94, CG14, BHO18, SCR16, SL18].

By comparison, finite difference (FD) schemes have seen less interest with regard to a posteriori estimates. This is predominantly due to the problem lacking a variational structure, something quite crucial for typical a posteriori techniques to be applied. Indeed, goal-oriented a posteriori estimates have been derived for the Lax-Wendroff scheme by proving the method is equivalent to a finite element scheme [CET14]. Although there are the approaches that work for general numerical schemes approximating scalar conservation laws [CG95] and a posteriori estimates derived for FD schemes with local error estimation based on Richardson extrapolation [BO84, ABF88] . The estimates are used to facilitate mesh adaptivity.

In this work we derive a posteriori bounds for general finite difference schemes approximating systems of nonlinear conservation laws. The main novelty of our work is that it enables the construction of reliable a posteriori estimate for general FD schemes. The result is quite general, in that we assume nothing on the exact solution although the final estimate is conditional, in that it holds only under some conditions on the

Date: December 21, 2022.

numerical solution. The framework has inherent mechanisms to construct robust estimates of high-order, at least in the pre-shock regime, enabling the user to obtain optimal bounds for high order FD schemes using WENO interpolation [LSZ09, JSB+19].

The a posteriori error analysis is carried out using the stability framework of the PDE (see [GMP15]) and it therefore yields a bound which is usable regardless of the chosen numerical discretisation technique. The construction of the estimate for computational purposes is done using reconstruction techniques. Similar techniques have been used for dG methods (see e.g. [Mak07, GMP15, GP17, DGPR19]).

We conduct a range of numerical experiments to benchmark the behaviour of the estimates and to demonstrate the concept of optimal order for smooth solutions. We showcase relevant results for widely used schemes for both linear and nonlinear examples. The reader should note that the estimates are robust in the pre-shock regime but not in the post-shock regime.

We will also demonstrate the estimate's ability to track parasitic waves, spurious numerical oscillations often travelling in the opposite direction than that of the physical solution, a well known numerical artefact in the approximation of this class of problem. They often are the result of initial condition travelling over a area of changing grid resolution. The issue of parasitic waves in non-uniform grids is well-known (cf. [Vic81a],[Vic81b],[Tre82] and more recently [LT11]).

The rest of this work is structured as follows. In §2 we introduce notation, we setup the problem and we present the stability framework for the PDE and the a posteriori bound. We also obtain an a posteriori bound for the advection equation. In §2.2 we demonstrate the construction and motivate the use of the a posteriori bound using an illustrative example. In §3 we present the spatial and temporal discretisations we will be using. In §4 we present WENO schemes which will be used as spatial discretisations in some of the examples we consider. In §5 we expand upon §2.2 by providing the general framework for constructing bounds for FD schemes. In §6 we conduct benchmarking experiments to assess the behaviour of the bounds constructed using the proposed framework. In §7 we present the implementation details required for adaptivity using the framework we derived. In §8 we use the a posteriori bounds as drivers for adaptivity for a linear example (the advection equation) and a nonlinear system (shallow water equations). We conclude this work in §9.

2. Preliminaries and problem setup

In this section we introduce notation and present the mains ideas of subsequent sections through an illustrative example. Let $\Omega \subset \mathbb{R}$. The Lebesgue spaces notation is

$$\mathrm{L}^p(\Omega) = \left\{\phi\colon \int_\Omega |\phi|^p < \infty\right\} \quad \text{for } p \in [1,\infty) \text{ and } \mathrm{L}^\infty(\Omega) = \left\{\phi\colon \operatorname{ess}\sup_{x\in\Omega} |\phi(x)| < \infty\right\} \tag{2.1}$$

which are equipped with corresponding norms

$$\|u\|_{\mathrm{L}^p(\Omega)} = \begin{cases} \left(\int_\Omega |u|^p\right)^{1/p}, \\ \operatorname{ess\,sup}_{x\in\Omega} |u(x)|. \end{cases} \tag{2.2}$$

We also consider the Sobolev spaces

$$\mathrm{W}^{k,p}(\Omega) = \{\phi \in \mathrm{L}^p(\Omega) : \mathrm{D}^\alpha \phi \in \mathrm{L}^p(\Omega) \quad \text{for} \quad |\alpha| \le k\}, \tag{2.3}$$

where α is a multi-index and the derivatives D^α are understood in the weak sense. The Sobolev spaces (2.3) are equipped with norms and seminorms given by

$$\|u\|_{\mathrm{W}^{k,p}(\Omega)} := \left(\sum_{|\alpha|\le k} \|\mathrm{D}^\alpha u\|^p_{\mathrm{L}^p(\Omega)}\right)^{1/p} \quad \text{and} \quad |u|_{\mathrm{W}^{k,p}(\Omega)} := \left\|\mathrm{D}^k u\right\|^p_{\mathrm{L}^p(\Omega)}. \tag{2.4}$$

Finally, let us introduce the following time-dependent Bochner spaces [Eva98]:

$$\mathrm{C}^i(0,T;\mathrm{W}^{k,p}(\Omega)) = \left\|u : [0,T] \to \mathrm{W}^{k,p}(\Omega) : \text{u and time derivatives up to order } i \text{ are in } \mathrm{W}^{k,p}(\Omega)\right\|. \tag{2.5}$$

2.1. **Lemma** (Stability and error control for the linear advection equation)**.** *Let u be an entropy solution of the initial boundary value problem*

$$\begin{aligned} u_t + u_x &= 0 \quad \text{in } \Omega \times (0,T] \\ u(x,0) &= u_0(x) \quad \text{in } \Omega \times \{0\} \end{aligned} \tag{2.6}$$

with periodic boundary conditions and suppose v is an entropy solution of the perturbed problem for some $R \in \mathrm{L}^\infty(0,T;\mathrm{L}^2(\Omega))$

$$\begin{aligned} v_t + v_x &= -R \quad \text{in } \Omega \times (0,T] \\ v(x,0) &= v_0(x) \quad \text{in } \Omega \times \{0\}, \end{aligned} \tag{2.7}$$

also with periodic boundary conditiosn. Then, the error between the two functions, $e := u - v$, satisfies the following bound for all $t \in [0,T]$:

$$\|e(t)\|^2_{\mathrm{L}^2(\Omega)} \leq \omega(t) \left[\|e(0)\|^2_{\mathrm{L}^2(\Omega)} + \int_0^t \|\delta(s) R(s)\|^2_{\mathrm{L}^2(\Omega)} \, \mathrm{d}s \right], \tag{2.8}$$

where

$$\omega(t) = \begin{cases} \exp(t) & \text{for } t \leq 1 \\ t \exp(1) & \text{for } t \geq 1. \end{cases} \tag{2.9}$$

and

$$\delta(s) = \begin{cases} 1 & \text{for } s \leq 1 \\ \sqrt{s} & \text{for } s \geq 1. \end{cases} \tag{2.10}$$

Proof. Subtracting (2.7) from (2.6) we have the following error equation for e

$$\begin{aligned} e_t + e_x &= R \quad \text{in } \Omega \times (0,T] \\ e(x,0) &= (u_0 - v_0)(x) \quad \text{in } \Omega \times \{0\}. \end{aligned} \tag{2.11}$$

Testing (2.11) with e we see

$$\begin{aligned} \int_\Omega Re &= \int_\Omega e_t e + e_x e \\ &= \frac{1}{2} \frac{\mathrm{d}}{\mathrm{d}t} \|e\|^2_{\mathrm{L}^2(\Omega)} + \int_\Omega \frac{1}{2} \left(e^2\right)_x . \end{aligned} \tag{2.12}$$

Since the domain is periodic, it follows that

$$\int_\Omega Re = \frac{1}{2} \frac{\mathrm{d}}{\mathrm{d}t} \|e\|^2_{\mathrm{L}^2(\Omega)} . \tag{2.13}$$

Now, applying Cauchy-Schwarz and Cauchy's inequalities to (2.13) we see,

$$\frac{1}{2} \frac{\mathrm{d}}{\mathrm{d}t} \|e\|^2_{\mathrm{L}^2(\Omega)} = \int_\Omega \delta R \delta^{-1} e \leq \|\delta R\|_{\mathrm{L}^2(\Omega)} \left\|\delta^{-1} e\right\|_{\mathrm{L}^2(\Omega)} \leq \frac{1}{2} \|\delta R\|^2_{\mathrm{L}^2(\Omega)} + \frac{1}{2} \left\|\delta^{-1} e\right\|^2_{\mathrm{L}^2(\Omega)}, \tag{2.14}$$

for any $\delta \in \mathrm{C}^0([0,T], \mathbb{R}^+)$.

Now we can make use of Gronwall's inequality to realise the bound

$$\|e(t)\|^2_{\mathrm{L}^2(\Omega)} \leq \exp\left(\int_0^t \delta(s)^{-2} \, \mathrm{d}s \right) \left[\|e(0)\|^2_{\mathrm{L}^2(\Omega)} + \int_0^t \|\delta(s) R(s)\|^2_{\mathrm{L}^2(\Omega)} \, \mathrm{d}s \right]. \tag{2.15}$$

Choosing

$$\delta(s) = \begin{cases} 1 & s \leq 1 \\ \sqrt{s} & s \geq 1 \end{cases} \tag{2.16}$$

concludes the proof. □

2.2. **Fundamental numerical methods and a posteriori bounds.** Let Ω denote the unit interval with matching endpoints. We partition by choosing $0 = x_0 < \cdots < x_M = 1$. We denote the spatial mesh size $h_j := x_{j+1} - x_j$ for $0 \le j \le M-1$ and we use I_j to denote the sub-interval $[x_j, x_{j+1}]$ of Ω. For the temporal variable we partition $[0,T]$ into sub-intervals with endpoints given by $0 = t^0 < \cdots < t^N = T$. The time-step is defined by $\tau^n := t^{n+1} - t^n$. We denote by U_j^n as an approximation to $u(x_j, t^n)$, the solution of (2.6) given by either one of two different schemes. The two schemes we consider are both posed over a uniform temporal and spatial partition, that is $\tau^n \equiv \tau$ for all n and $h_j \equiv h$ for all j and are both classical schemes in the study of conservation laws. We consider an upwinding forward-time backward-space (FTBS)

$$\begin{aligned} U_j^{n+1} &= U_j^n + \frac{\tau}{h}\big(U_j^n - U_{j-1}^n\big) \text{ for } n = 0, \dots, N-1 \text{ and } j = 0, \dots, M-1 \\ U_j^0 &= u_0(x_j) \text{ for } j = 0, \dots, M \end{aligned} \tag{2.17}$$

and a forward-time central-space (FTCS)

$$\begin{aligned} U_j^{n+1} &= U_j^n + \frac{\tau}{2h}\big(U_{j+1}^n - U_{j-1}^n\big) \text{ for } n = 0, \dots, N-1 \text{ and } j = 0, \dots, M-1. \\ U_j^0 &= u_0(x_j) \text{ for } j = 0, \dots, M. \end{aligned} \tag{2.18}$$

Formally, these methods have truncation error $\mathrm{O}(\tau + h)$ and $\mathrm{O}(\tau + h^2)$ respectively.

In order to make use of the abstract bounds given in §2 we must have an interpretation of the numerical approximation $\{U_j^n\}_j^n$, which is only defined as point values over the space-time domain. The most intuitive post-processing is to apply a bilinear Lagrange interpolant in space-time. With a globally defined reconstruction, $\widehat{U}$ say. As a consequence of Lemma 2.1, we arrive at the a posteriori bound:

$$\left\| (u - \widehat{U})(t) \right\|_{\mathrm{L}^2(\Omega)}^2 \le \omega(t) \left[\left\| (u - \widehat{U})(0) \right\|_{\mathrm{L}^2(\Omega)}^2 + \int_0^t \|\delta(s) R(s)\|_{\mathrm{L}^2(\Omega)}^2 \, \mathrm{d}s \right] =: \omega(t)\, \mathscr{E}(t)^2, \tag{2.19}$$

where

$$R := -\widehat{U}_t - \widehat{U}_x \tag{2.20}$$

is the discrete residual of the reconstruction. Note that given the numerical solution, the right hand side of (2.19) is fully computable. It can even be shown to be fully robust when u_0 is a sufficiently smooth initial condition.

2.3. **Lemma.** *Asymptotic convergence rate for the reconstruction residual Let $\{U_j^n\}_j^n$ be the FTBS approximation of u, the solution of (2.6) with $u_0 \in \mathrm{C}^2(\Omega)$. Suppose $\widehat{U}$ is the piecewise bilinear interpolant of the nodal values of $\{U_j^n\}_j^n$ and let $\omega(t)\,\mathscr{E}(t)^2$ be defined in (2.19), then*

$$\omega(t)\, \mathscr{E}(t)^2 = \mathrm{O}(\tau^2 + h^2). \tag{2.21}$$

Proof. We begin by defining, $\widehat{U}^t$ as

$$\widehat{U}_j^t(t) := U_j^n + \frac{U_j^{n+1} - U_j^n}{\tau}(t - t^n), \quad \text{for} \quad t \in \left[t^n, t^{n+1}\right] \text{ and } j \in [0, M-1], \tag{2.22}$$

which represents interpolant in the temporal direction. That allows us to write $\widehat{U}$ as

$$\widehat{U}(x,t) := \widehat{U}_j^t(t) + \frac{\widehat{U}_{j+1}^t(t) - \widehat{U}_j^t(t)}{h}(x - x_j) \quad \text{for} \quad (x,t) \in [x_j, x_{j+1}] \times \left[t^n, t^{n+1}\right]. \tag{2.23}$$

Since $\widehat{U}$ is bilinear on a space-time slab we can compute R explicitly as

$$-R = \partial_t \widehat{U} + \partial_x \widehat{U} = \partial_t \widehat{U}_j^t + \frac{\partial_t \widehat{U}_{j+1}^t - \partial_t \widehat{U}_j^t}{h}(x - x_j) + \frac{\widehat{U}_{j+1}^t - \widehat{U}_j^t}{h}. \tag{2.24}$$

Now

$$\begin{aligned} \|\omega(s) R(s)\|_{\mathrm{L}^2(\Omega)}^2 &= \sum_j \int_{x_j}^{x_{j+1}} |\omega(s) R(s,x)|^2 \, \mathrm{d}x \\ &= \sum_j h \left| w(s) R(s, x_{j+1/2}) \right|^2, \end{aligned} \tag{2.25}$$

as ωR is linear over each spatial interval. Note that, because ω changes form once $t > 1$, we split the integral into two parts. We also denote by n_1 the index of the time-step where $t > 1$ for the first time. Now

(2.26)
$$\begin{aligned}\int_0^T \|\omega(s)R(s)\|^2_{\mathrm{L}^2(\Omega)}\,\mathrm{d}s &= \sum_n \int_{t^n}^{t^{n+1}} \sum_j h\left|w(s)R(s,x_{j+1/2})\right|^2\,\mathrm{d}s,\\ &\le \sum_{n=0}^{n=n_1}\int_{t^n}^{t^{n+1}}\sum_j h\left|R(s,x_{j+1/2})\right|^2\,\mathrm{d}s + \sum_{n=n_1}^{n=N-1}\int_{t^n}^{t^{n+1}}\sum_j h\left|sR(s,x_{j+1/2})\right|^2\,\mathrm{d}s\\ &\le \tau h\left(\sum_{n=0}^{n=n_1-1}\sum_j\left|R(t^{n+1/2},x_{j+1/2})\right|^2 + \sum_{n=n_1}^{N-1}\sum_j\left|t^{n+1/2}R(t^{n+1/2},x_{j+1/2})\right|^2\right).\end{aligned}$$

Using (2.24) we see

$$-R(t^{n+1/2},x_{j+1/2}) = \frac{U_j^{n+1}-U_j^n}{\tau} + \frac{U_{j+1}^n - U_j^n}{h} + \left(\frac{1}{2\tau}+\frac{1}{2h}\right)\left(U_{j+1}^{n+1}-U_{j+1}^n - \left(U_j^{n+1}-U_j^n\right)\right), \tag{2.27}$$

which simplifies to

$$\begin{aligned}-R(t^{n+1/2},x_{j+1/2}) &= \frac{1}{2\tau}\left(U_{j+1}^{n+1}-U_{j+1}^n+\left(U_j^{n+1}-U_j^n\right)\right)+\frac{1}{2h}\left(U_{j+1}^{n+1}-U_j^{n+1}+\left(U_{j+1}^n-U_j^n\right)\right)\\ &= \frac{1}{2}\left(\frac{U_{j+1}^{n+1}-U_{j+1}^n}{\tau}+\frac{U_{j+1}^n-U_j^n}{h}\right)+\frac{1}{2}\left(\frac{U_j^{n+1}-U_j^n}{\tau}+\frac{U_{j+1}^{n+1}-U_j^{n+1}}{h}\right)\\ &= \frac{1}{2}\left(\frac{U_j^{n+1}-U_j^n}{\tau}+\frac{U_{j+1}^{n+1}-U_j^{n+1}}{h}\right),\end{aligned} \tag{2.28}$$

since the term in the first bracket is the numerical scheme we use. The contribution to the residual comes from the second term. In order to assess this we add the PDE at (t^n, x_j) and discretise the resulting terms. For ease of notation we will denote the exact solution $u(t^n, x_j)$ as u_j^n and define a truncation error at (t, x), $T(t, x)$, as

$$T(t,x) := \frac{u(t+\tau,x)-u(x,t)}{\tau}+\frac{u(t+\tau,x+h)-u(t+\tau,x)}{h} = \frac{1}{2}(\tau u_{tt}(\eta,x)+hu_{xx}(t,\xi)), \tag{2.29}$$

for some $\xi \in (x, x + h)$, $\eta \in (t, t + \tau)$. This is of the same form as the truncation error for the FTBS scheme for this problem (see [MM05, §4]). We use this to write (2.28) as

(2.30)
$$-R(t^{n+1/2},x_{j+1/2}) = \frac{1}{2}\left(\frac{U_j^{n+1}-U_j^n}{\tau}+\frac{U_{j+1}^{n+1}-U_j^{n+1}}{h}\right)-\frac{1}{2}\left(\frac{u_j^{n+1}-u_j^n}{\tau}+\frac{u_{j+1}^{n+1}-u_j^{n+1}}{h}\right)+\frac{1}{2}T_{j+1}^{n+1},$$

where T_{j+1}^{n+1} is the truncation error at the $\left(t^{n+1}, x_{j+1}\right)$. This simplifies to

$$-R(t^{n+1/2},x_{j+1/2}) = \frac{1}{2}\left(\frac{e_j^{n+1}-e_j^n}{\tau}+\frac{e_{j+1}^{n+1}-e_j^{n+1}}{h}\right)+\frac{1}{2}T_{j+1}^{n+1}. \tag{2.31}$$

Therefore,

$$\left|R(t^{n+1/2},x_{j+1/2})\right| \le \left(\frac{\left|e_j^{n+1}\right|+\left|e_j^n\right|}{\tau}+\frac{\left|e_{j+1}^{n+1}\right|+\left|e_j^{n+1}\right|}{h}\right)+\frac{1}{2}\left|T_{j+1}^{n+1}\right|. \tag{2.32}$$

We denote the maximum truncation error in the computation as

$$T := \max_n \max_j \left|T_j^n\right|, \tag{2.33}$$

noting that $T = \mathcal{O}(\tau + h)$ and that (cf. [MM05])

$$\max_j \left|e_j^n\right| \le n\tau T. \tag{2.34}$$

Hence, (2.32) simplifies to

$$\left|R\left(t^{n+1/2}, x_{j+1/2}\right)\right| \leq T + 4(n+1)T \leq D(\tau + h)\,, \tag{2.35}$$

for some constant D. Finally, substituting (2.35) into (2.26) and noting that $\tau := 1/N$ and $h := 1/M$, after some simple algebraic manipulations we obtain

$$\|\omega(s)R(s)\|^2_{\mathrm{L}^2(\Omega)} \leq \tau h\left(\sum_{n=0}^{n_1-1}\sum_j |D(\tau+h)|^2 + \sum_{n=n_1}^{N-1}\sum_j \left|Dt^{n+1/2}(\tau+h)\right|^2\right) = \mathcal{O}\left(\tau^2+h^2\right), \tag{2.36}$$

which concludes the proof. □

2.4. **Remark.** The residual in Lemma 2.3 would result in an optimal bound if FTBS was used, because, in this case, the error would behave as $\mathcal{O}(\tau + h)$. We will demonstrate this with numerical examples.

2.5. **Definition** (EOC and EI)**.** To test the validity and robustness of our estimate we will examine the *estimated order of convergence* (EOC) of the estimate and the *effectivity index* (EI).

Consider two sequences $a_i(t)$ and h_i which converge to zero from above we define the EOC for these to be

$$EOC(a_i(t); h_i) := \frac{\log(a_{i+1}(t)/a_i(t))}{\log(h_{i+1}/h_i)}. \tag{2.37}$$

We define the EI at a time t to be the ratio of the estimator and the error at that time, that is:

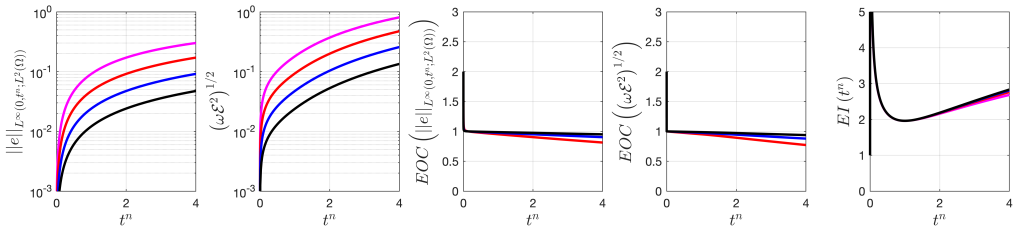
$$EI(t) := \frac{\mathscr{E}(t)}{\left\|u - \widehat{U}\right\|_{\mathrm{L}^\infty(0,t;\mathrm{L}^2(\Omega))}}. \tag{2.38}$$

This allows us to quantify how effective a bound the estimator is over time.

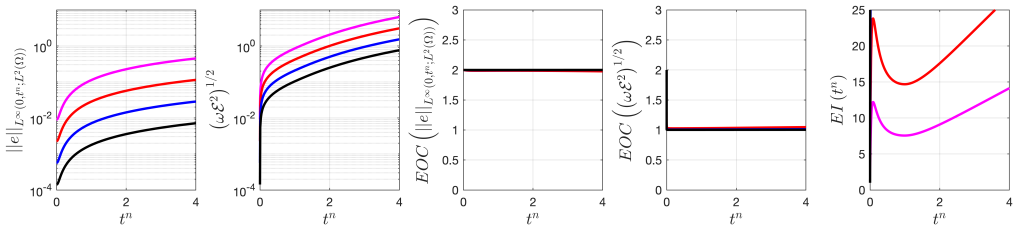
We illustrate the asymptotic behaviour of the a posteriori bound by considering the solution of (2.6) with

$$u_0(x) = \sin(2\pi x)\,. \tag{2.39}$$

We examine the behaviour of the bound in both the FTBS and FTCS schemes. The results are shown in Figure 1. As indicated in Lemma 2.3 the asymptotic convergence rate of the estimate matches that of the error for the FTBS scheme but the Lemma does not naturally extend to the FTCS scheme. The reason for this is that the naive bilinear interpolant lacks the approximability to achieve the optimal convergence rate of the residual.



(a) FTBS (2.17).



(b) FTCS (2.18).

Fig. 1. Errors and asymptotic convergence rates for the bilinear interpolant of FTBS (2.17) and FTCS (2.18) approximations of (2.6) with initial condition (2.39) and periodic boundary conditions. The simulations were conducted over a family of meshes with discretisation parameter $h = 2^{-m}, m = 4, \dots 7$, with a timestep $\tau = \frac{h^2}{10}$. Note that the estimate is optimal for the FTBS scheme and achieves a very favourable effectivity of $EI(4) < 3$, however it is suboptimal for the FTCS scheme.

The suboptimality in the a posteriori bound for the FTCS scheme can be circumvented. We do this by building information from within the finite difference spatial discretisation directly into the post-processor. Indeed, augmenting $\widehat{U}$ such that it is defined as the piecewise linear Lagrange interpolant of the finite difference coefficients $\{U_j^n\}$ in time and the unique piecewise quadratic interpolant satisfying

$$\begin{aligned} \widehat{U}(x_j, t^n) &= U_j^n \\ \widehat{U}(x_{j+1}, t^n) &= U_{j+1}^n \\ \partial_x \widehat{U}(x_j, t^n) &= \frac{U_{j+1}^n - U_{j-1}^n}{2h} \end{aligned} \tag{2.40}$$

we are closer to the spirit of a finite difference method. The argument from Lemma 2.3 can be modified to apply in this case. Indeed, one can show that applying the same argument the interpolant given by (2.40) yields $\sqrt{w\mathscr{E}^2} = \mathrm{O}\big(\tau + h^2\big)$, which is optimal for the FTCS scheme. To illustrate this asymptotic convergence properties we have replicated the same experiment as in Figure 1 for the quadratic reconstruction. This is shown in Figure 2. It can be seen that the additional information provided by appropriately increasing the order of data representation allows us to achieve optimal convergence rates of the a posteriori bound and favourable effectivities.

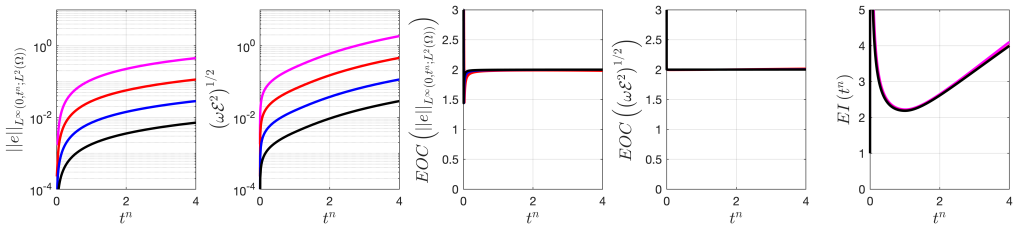


Fig. 2. Errors and asymptotic convergence rates for the linear in time, quadratic in space Hermite interpolant of FTCS (2.18) approximations of (2.6) with initial condition (2.39) and periodic boundary conditions. The simulations were conducted over a family of meshes with discretisation parameter $h = 2^{-m}, m = 4, \dots 7$, with a timestep $\tau = \frac{h^2}{10}$. Note that the estimate is now optimal for the FTCS scheme with favourable effectivity of $EI(4) \sim 4$.

The ideas presented in this section form an intuitive way to obtain the reconstruction. It is possible to generalise this quite naturally to other spatio-temporal discrestisations as we will present in the forthcoming sections.

3. General Systems

In this section we present spatial and temporal discretisations for schemes for more general conservation laws than in §2.2. Since we consider vectorial problems, we will denote by $\boldsymbol{U}_j^n$ the numerical approximation to $\boldsymbol{u}(x_j, t^n)$.

3.1. **Remark** (One-dimensional system of conservation laws)**.** We consider problems of the form

$$\begin{aligned} \boldsymbol{u}_t + \partial_x \boldsymbol{f}(\boldsymbol{u}) &= \boldsymbol{0}, \\ \boldsymbol{u}(x,0) &= \boldsymbol{u}_0(x)\,, \end{aligned} \qquad \text{for } (x,t) \in \Omega \times (0,\infty) \tag{3.1}$$

with $\boldsymbol{u} = (u_1, \dots, u_p)^T$ and $\boldsymbol{f}(\boldsymbol{u}) = (f_1(\boldsymbol{u}), \dots, f_p(\boldsymbol{u}))^T$ and complemented with periodic boundary conditions. In particular,

$$\begin{array}{rccc} \boldsymbol{u}: & \mathbb{R} \times \mathbb{R}^+ & \to & \mathbb{R}^p \\ & (x,t) & \mapsto & \boldsymbol{u}(x,t) \end{array} \tag{3.2}$$

and the flux function $\boldsymbol{f}$

$$\begin{array}{rccc} \boldsymbol{f}: & \mathbb{R}^p & \to & \mathbb{R}^p \\ & \boldsymbol{u}(x,t) & \mapsto & \boldsymbol{f}(\boldsymbol{u}(x,t)) \end{array} \tag{3.3}$$

3.2. **Definition** (Entropy/entropy-flux pair)**.** The pair (η, q) is an entropy/entropy-flux pair associated with the conservation law (3.1) iff η is convex and

$$\mathrm{D}q = \mathrm{D}\eta \mathrm{D}f. \tag{3.4}$$

3.3. **Definition** (Entropy solution)**.** A function $\boldsymbol{u} \in \mathrm{L}^\infty\left(\mathbb{R}^{\mathrm{d}} \times [0,\infty]\right)$ is an entropy solution of (3.1) with an associated entropy/entropy-flux pair (η, q) if

$$\begin{aligned} &\int_0^\infty \int_\Omega \boldsymbol{u} \cdot \partial_t \phi + \boldsymbol{f}(\boldsymbol{u}) \cdot \partial_x \phi \,\mathrm{d}x\mathrm{d}t + \int_\Omega \boldsymbol{u}_0 \cdot \phi(\cdot,0)\,\mathrm{d}x = 0 \quad \forall \phi \in C_0^1(\mathbb{R} \times [0,\infty)) \quad \text{and} \\ &\int_0^\infty \int_\Omega \eta(\boldsymbol{u})\,\partial_t \phi + q(\boldsymbol{u})\,\partial_x \phi \,\mathrm{d}x\mathrm{d}t + \int_\Omega \eta(\boldsymbol{u}_0)\,\phi(\cdot,0)\,\mathrm{d}x \geq 0 \quad \forall \phi \in C_c^1(\mathbb{R} \times [0,\infty)) \end{aligned} \tag{3.5}$$

It can be verified that strong solutions of (3.1) also satisfy the additional conservation law

$$\partial_t \eta(\boldsymbol{u}) + \partial_x q(\boldsymbol{u}) = 0. \tag{3.6}$$

3.4. **Spatial discretisation.** It is well known that numerical schemes for non-linear conservation laws may converge to functions which are not weak-solutions of the original problem (see [LeV92, §12.1]). We address this problem by expressing the method in conservation form. We use a consistent numerical flux function $\boldsymbol{F}$, which takes $p+q+1$ arguments:

$$\begin{array}{rccl} \big(\boldsymbol{U}^n_{j-p+1},\cdots,\boldsymbol{U}^n_{j+q}\big): & \boldsymbol{F}^n_j & \mapsto & \boldsymbol{F}\big(\boldsymbol{U}^n_{j-p},\cdots,\boldsymbol{U}^n_{j+q}\big) \\ & \boldsymbol{F}(\boldsymbol{v},\cdots,\boldsymbol{v}) & = & \boldsymbol{f}(\boldsymbol{v}), \end{array} \tag{3.7}$$

where p and q are simply used to determine the width of the computational stencil. We use $\boldsymbol{F}$ to approximate $\partial_x \boldsymbol{f}$ such that

$$\partial_x \boldsymbol{f}(\boldsymbol{u}) \approx \frac{1}{h}\big(\boldsymbol{F}\big(\boldsymbol{U}^n_{j-p},\cdots,\boldsymbol{U}^n_{j+q}\big) - \boldsymbol{F}\big(\boldsymbol{U}^n_{j-p-1},\cdots,\boldsymbol{U}^n_{j+q-1}\big)\big)\,. \tag{3.8}$$

We can then use a method-of-lines approach in the discretisation of (3.1) by requiring

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{U}_j = \frac{1}{h}\big(\boldsymbol{F}\big(\boldsymbol{U}^n_{j-p},\cdots,\boldsymbol{U}^n_{j+q}\big) - \boldsymbol{F}\big(\boldsymbol{U}^n_{j-p-1},\cdots,\boldsymbol{U}^n_{j+q-1}\big)\big) \quad \forall\, j = 0,\ldots,M. \tag{3.9}$$

For clarity, we provide illustrative examples of $\boldsymbol{F}$ for the Lax-Friedrichs and the Lax-Wendroff scheme.

3.5. **Remark** (Conservation form for the Lax-Friedrichs scheme)**.** The Lax-Friedrichs scheme can be written in conservation form, (3.9), by defining the numerical flux function $\boldsymbol{F}$ as

$$\boldsymbol{F}(\boldsymbol{U}_j,\boldsymbol{U}_{j+1}) := \frac{h}{2\tau}(\boldsymbol{U}_j - \boldsymbol{U}_{j+1}) + \frac{1}{2}(\boldsymbol{f}(\boldsymbol{U}_j) + \boldsymbol{f}(\boldsymbol{U}_{j+1}))\,. \tag{3.10}$$

The Lax-Friedrichs flux is formally $\mathcal{O}(h)$.

3.6. **Remark** (Conservation form for the Lax-Wendroff scheme)**.** The Lax-Wendroff scheme can be written in conservation form by using the Richtmayer two-stage method. We define the numerical flux function $\boldsymbol{F}$ as

$$\boldsymbol{F}(\boldsymbol{U}_j,\boldsymbol{U}_{j+1}) := \boldsymbol{f}\big(\boldsymbol{U}^{n+1/2}_{j+1/2}\big)\,, \tag{3.11}$$

where

$$\begin{aligned} \boldsymbol{U}^{n+1/2}_{j+1/2} &:= \frac{1}{2}\big(\boldsymbol{U}^n_{j+1} + \boldsymbol{U}^n_j\big) - \frac{\tau}{2h}\big(\boldsymbol{f}\big(\boldsymbol{U}^n_{j+1}\big) - \boldsymbol{f}\big(\boldsymbol{U}^n_j\big)\big) \\ \boldsymbol{U}^{n+1/2}_{j-1/2} &:= \frac{1}{2}\big(\boldsymbol{U}^n_j + \boldsymbol{U}^n_{j-1}\big) - \frac{\tau}{2h}\big(\boldsymbol{f}\big(\boldsymbol{U}^n_j\big) - \boldsymbol{f}\big(\boldsymbol{U}^n_{j-1}\big)\big)\,. \end{aligned} \tag{3.12}$$

The conservation form of the scheme is then given by:

$$\boldsymbol{U}^{n+1}_j := \boldsymbol{U}^n_j - \frac{\tau}{h}\Big(\boldsymbol{f}\Big(\boldsymbol{U}^{n+1/2}_{j+1/2}\Big) - \boldsymbol{f}\Big(\boldsymbol{U}^{n+1/2}_{j-1/2}\Big)\Big)\,. \tag{3.13}$$

The Lax-Wendroff scheme is formally $\mathcal{O}\big(\tau^2 + h^2\big)$.

3.7. **Temporal discretisation.** We approximate the temporal variable using both implicit and explicit temporal discretisations. For example, using various 1-stage Runge-Kutta methods given by the Butcher tableau

$$\begin{array}{c|c|c} \theta & \theta \\ \hline & 1 \end{array}. \tag{3.14}$$

We also make use of Strong-Stability Preserving Runge-Kutta (SSPRK) methods (see [GST01]).

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1/2 & 1/4 & 1/4 & 0 \\ \hline & 1/6 & 1/6 & 2/3 \end{array}. \tag{3.15}$$

4. WENO Schemes

Tristan says: Dont be lazy, you do the reorganising! In this section we briefly summarise the details behind Weighted Essentially Non-Oscillatory (WENO) schemes, (cf. [JS96], [Shu98]), which pertain to our implementation. ENO/WENO schemes have been used with great success in several areas and in particular in the discretisation of hyperbolic conservation laws and convection-diffusion equations (see e.g. [Shu20] for an overview).

They have several features which contributed to their widespread success. These include a mechanism of computational stencil construction that can achieve arbitrarily high accuracy in regions where the solution is smooth, essentially non-oscillatory behavior in the vicinity of discontinuities (no artificial, numerical overshoots and undershoots) and proven ability to simulate complex smooth solution structures (see [Shu98] for more details). In the present work we only consider WENO schemes.

There are two, closely related procedures that we use in the context of WENO schemes: reconstruction and approximation. The reconstruction procedure is used to form $\boldsymbol{F}$ in the spatial discritization of (3.8) while the approximation procedure is used to obtain the spatial component of the post-processor from Lemma 2.1 (see §5).

4.1. **Remark.** In order to avoid cofusion, it is emphasized that the ENO/WENO reconstruction procedure, as that is defined in [Shu98], refers to the procedure used to formulate the ENO/WENO numerical scheme. The reconstruction procedure we have developed refers to the procedure used to obtain the post-processor which is subsequently used for a posteriori error computations.

The readers should note that this procedure, as well as the characteristics of the resulting reconstruction, are built using information from the scheme and are therefore inherently linked with it.

4.2. **Definition** (WENO reconstruction problem from [Shu98])**.** Given the cell averages of a function $v(x)$:

$$\bar{v}_j := \frac{1}{h_j}\int_{x_j}^{x_{j+1}} v(\xi)\,\mathrm{d}\xi, \quad j = 0,\dots,M-1, \tag{4.1}$$

find a polynomial $p_j(x)$ of degree at most $k-1$ for each cell I_j such that it is a $k-th$ order accurate approximation to $v(x)$ inside I_j:

$$p_j(x) = v(x) + \mathcal{O}\big(h^k\big), \quad x \in I_j, \quad j = 0,\dots,M-1. \tag{4.2}$$

The procedure for obtaining the polynomial p_j from $\bar{v}_j$ can be found in [Shu98, Procedure 2.2] and in [Shu20, §.2.2]. This procedure is used both for finite volumes and for finite differences with a minor difference. In the context of finite volumes, we use the cell averages $\bar{v}_j$ to obtain a high order approximation to v. We then substitute this in an expression for $\boldsymbol{F}$ to calculate the numerical flux. In contrast, in the context of finite differences, the computational variables are point values rather than cell averages. In this case, the values $\boldsymbol{f}(\boldsymbol{U}_j)$, $j = 1,\dots,M$, are used to obtain a high-order accurate approximation to $\boldsymbol{F}$ and subsequently $\partial_x \boldsymbol{f}$ in (3.7) and (3.8). This is the approach we will be using throughout this work.

We present this procedure below for a uniform scheme for simplicity. The reader should note that in the shallow water numerical experiment in §8, the WENO scheme used is derived for a non-uniform scheme over an adaptive grid. As a result, all geometry-related quantities - such as the sub-stencil polynomials, the smoothenss indicators and the resulting weights - are no longer pre-computable constants. Instead, they have to be re-computed at every time-step. The reader can find the detailed procedure on how to derive the scheme on a non-uniform grid in [Shu98].

4.3. **Remark** (Order of WENO schemes on non-uniform grids)**.** The reader is advised that, as is pointed out in [Shu98], WENO schemes for finite differences which are posed on non-uniform grids cannot be higher than order two.

4.4. **The WENO-3 scheme.** In one spatial dimension, the WENO reconstruction procedure for the third order finite difference WENO scheme is used to approximate $\boldsymbol{f}_x$ on the cell $I_j := [x_j, x_{j+1}]$. The cell I_j is chosen to be the central cell of the computational stencil $S := \{I_{j-1}, I_j, I_{j+1}\}$. The approximation is then obtained as a convex combination of polynomials over two sub-stencils of S, namely

$$\begin{aligned} S_1 &:= \{I_{j-1}, I_j\} \quad \text{and} \\ S_2 &:= \{I_j, I_{j+1}\}. \end{aligned} \tag{4.3}$$

The polynomials $p_1(x)$and $p_2(x)$ are the ENO reconstructions of $f(u)$ on the substencils S_1 and S_2 respectively. The numerical flux at x_j, denoted by F_j, is obtained as the combination

$$F_j := w_1 p_1(x_j) + w_2 p_2(x_j)\,, \tag{4.4}$$

where w_1 and w_2 are the non-linear weights (see (4.17)) corresponding to $S_1$and S_2, which have to satisfy the conditions

$$w_l \geq 0, \quad \sum_{l=1}^{2} w_l = 1. \tag{4.5}$$

Finally, the WENO approximation to the flux derivative is obtained using

$$\partial_x f \simeq \frac{1}{h_{j-1}}(F_j - F_{j-1})\,. \tag{4.6}$$

4.5. **Remark** (Flux-splitting)**.** When WENO schemes are implemented using finite differences one should ensure upwinding and stability (see [Shu98]). This can be achieved in various ways. In this paper, this was done by applying the chosen finite difference method to a *flux-splitting* $f^{\pm}(u)$ of $f(u)$. In particular,

$$f(u) = f^{+}(u) + f^{-}(u)\,. \tag{4.7}$$

where

$$\frac{d}{du} f^{+}(u) \geq 0 \quad \text{and} \quad \frac{d}{du} f^{-}(u) \leq 0. \tag{4.8}$$

A simple flux-splitting is the Lax-Friedrichs splitting, which is given by

$$f^{\pm}(u) := \frac{1}{2}(f(u) \pm \alpha u)\,, \tag{4.9}$$

For 1D scalar conservation laws

$$\alpha := \max_u |f'(u)|\,. \tag{4.10}$$

For hyperbolic systems of conservation laws, f' is a Jacobian (with real eigenvalues). In this case upwinding is slightly more involved. The reader should note that in that case one can either perform a characteristic decomposition (which is the more robust approach) or use flux-splitting on a component-by-component basis. We opt for the latter route as it is sufficient for the purposes of this study. In this case, α is calculated using the eigenvalues, λ_i, of the Jacobian f':

$$\alpha := \max_u \max_i |\lambda_i(u)|\,. \tag{4.11}$$

The reader should also note that when coding WENO schemes, the stencil used for $f^{+}(u)$ is biased to the left, while the stencil for $f^{-}(u)$ is biased one point to the right.

We will demonstrate the process for obtaining f^{+} as an example. The ENO sub-stencil polynomials for f^{+}, for a third order WENO scheme are given by

$$\begin{aligned} p_1(U) &= \frac{1}{2}(-f(U_{j-1}) + 3f(U_j))\,, \\ p_2(U) &= \frac{1}{2}(f(U_j) + f(U_{j+1})) \end{aligned} \tag{4.12}$$

Next, we construct the weights w_1 and w_2. Suppose we wanted to create a reconstruction for a function $v(x)$, which is piecewise smooth in sub-stencils S_1 and S_2. There are constants d_r, $r = 1, 2$ such that

$$v_{j+1} = d_1 v^{(1)}_{j+1} + d_2 v^{(2)}_{j+1} = v(x_{j+1}) + \mathcal{O}\big(h^{2k-1}\big)\,, \tag{4.13}$$

where $v^{(i)}_{j+1}$ is the reconstruction of $v(x)$ in the sub-stencil S_i evaluated at x_{j+1}. More specifically, for f^{+},

$$d_0 = \frac{1}{3}, \quad d_1 = \frac{2}{3}. \tag{4.14}$$

If $v(x)$ is smooth, the nonlinear weights w_i should be very close to d_i. If, instead, $v(x)$ has a discontinuity in some stencil, the w_i from that stencil should be close to zero to avoid spurious oscillatory behaviour. This is accomplished by using smoothness indicators β_i, where

$$\beta_i := \sum_{l=1}^{k-1} \int_{x_j}^{x_{j+1}} h^{2l-1} \left(\frac{\partial^l p_i(x)}{\partial x^l} \right)^2 \mathrm{d}x. \tag{4.15}$$

This is simply a sum of scaled L^2 norms of the derivatives of p_i. The factor h^{2l-1} ensures that β_i scales like an L^2 −norm over polynomials. In the case of the third order WENO scheme over a uniform grid,

$$\beta_1 = (v_{j-1} - v_j)^2, \quad \beta_2 = (v_j - v_{j+1})^2. \tag{4.16}$$

We can now obtain the weights w_i, which are given by

$$w_i := \frac{\alpha_i}{\sum_{s=0}^{k-1} \alpha_s}, \text{ with } \alpha_i = \frac{d_i}{(\epsilon + \beta_i)^2}. \tag{4.17}$$

The constant $\epsilon \ll 1$ is a small constant to ensure the denominator does not vanish. In experiments we use $\epsilon = 10^{-6}$. We repeat this process to obtain f^- noting that in this case the entire computational stencil is biased one position to the right.

4.6. **Remark** (Choice of nonlinear weights)**.** The choice of nonlinear weights is very important. As is demonstrated in [JS96], an appropriate choice of nonlinear weights can upgrade the order of accuracy of (4.4) in smooth regions relative to an ENO scheme with a stencil S_1 or S_2. Furthermore, because these weights are designed to reflect the smoothness of the reconstruction polynomial in the relevant stencil, they are also used to facilitate the non-oscillatory property of the WENO scheme.

The Smoothness Increasing Accuracy Preserving (SIAC) filtering is a comparable concept. This is a post-processing technique which has been used to reduce error oscillations and recover smoothness in the solution and its derivatives in the context of the Discontinuous Galerkin method (see [DGPR19]).

5. General reconstructions

In this section we formally define a reconstruction and describe properties we would like it to possess. Furthermore, we present a methodology to obtain a space-time reconstruction from a computed numerical solution for (3.1). We conclude the section by illustrating the procedure of obtaining the spatio-temporal reconstruction from the numerical solution.

5.1. **Definition** (Reconstruction)**.** The reconstruction, $\widehat{\boldsymbol{U}}$, of the numerical solution, $\boldsymbol{U}$, to (3.1) is a function that satisfies

$$\begin{aligned} \widehat{\boldsymbol{U}}_t + \partial_x \boldsymbol{f}\left(\widehat{\boldsymbol{U}}\right) &=: -\boldsymbol{R} \quad \text{in } \Omega \times (0, T] \\ \widehat{\boldsymbol{U}}(\boldsymbol{x}, 0) &= \boldsymbol{u}_0(\boldsymbol{x}) \quad \text{in } \Omega \times \{0\} \end{aligned} \tag{5.1}$$

and periodic boundary conditions, such that the reconstruction residual, $\boldsymbol{R}$, is well-defined and explicitly computable. Furthermore, $\widehat{\boldsymbol{U}}$ should lead to an optimal a posteriori error estimate. An estimate is optimal when it converges at the same rate as the chosen numerical scheme.

5.2. **Reconstruction Procedure.** We obtain the polynomial reconstruction $\widehat{\boldsymbol{U}}$ by using the nodal values of $\boldsymbol{U}$ as well as the temporal and spatial approximations of the partial derivatives of the equation. The reader should note that we interpolate firstly in time and subsequently in space, because the temporal component of (5.1) is linear while the spatial one may be non-linear.

In the exposition that follows, we will use the superscripts t and s to represent that the function in question is either time or space dependent only. We will also use the supercript ts to denote dependence on both time and space.

5.3. **Definition** (Space of the spatial reconstruction step)**.** Let $\mathbb{P}^q([x_j, x_{j+1}])$ denote the space of polynomials of degree $q \in \mathbb{N}$ over the sub-interval $[x_j, x_{j+1}]$. We define the space of the spatial step of the reconstruction,

$$\mathbb{V}_q^s := \left\{ w : [0, L] \to \mathbb{R} : w|_{[x_j, x_{j+1}]} \in \mathbb{P}^q([x_j, x_{j+1}]) \right\}, \tag{5.2}$$

to be the space of piecewise polynomials of degree q over $[0, L]$. The superscript s indicates the space dependence.

5.4. **Definition** (Space of the temporal reconstruction step)**.** We define the space of the temporal step of the reconstruction as the space of piecewise polynomials of degree r over $[0, T]$ such that

$$\mathbb{V}_r^t\big(0, T; \mathrm{L}^2(\Omega)\big) := \big\{ g : [0, T] \to V : g_{|[t^n, t^{n+1}]} \in \mathbb{P}^q\big(\big[t^n, t^{n+1}\big], \mathrm{L}^2(\Omega)\big)\big\}. \tag{5.3}$$

Here, $\mathbb{P}^q\big(\big[t^n, t^{n+1}\big], V\big)$ is the space of functions which are polynomials of degree q in time and belong to V in space.

5.5. **Definition** (Temporal reconstruction)**.** The temporal reconstruction, $\widehat{\boldsymbol{U}}^t \in \mathbb{V}_r^t\big(0, T; \mathbb{V}_q^s\big)$ of the numerical solution $\boldsymbol{U}$, is the unique function satisfying

$$\begin{aligned} \widehat{\boldsymbol{U}}^t(t^n) &= \boldsymbol{U}_j^n \quad \text{and} \\ \partial_t \widehat{\boldsymbol{U}}^t(t^n) &= -\frac{1}{h}\big(\boldsymbol{F}\big(\boldsymbol{U}_{j-p}^n, \cdots, \boldsymbol{U}_{j+q}^n\big) - \boldsymbol{F}\big(\boldsymbol{U}_{j-p-1}^n, \cdots, \boldsymbol{U}_{j+q-1}^n\big)\big). \end{aligned} \tag{5.4}$$

for $n = 0, \ldots, N$.

5.6. **Remark** (Order of the temporal reconstruction)**.** The procedure presented in Defn. 5.5 limits the temporal component of the reconstruction to third order. A possibility for increasing the order of the temporal reconstruction is by obtaining a WENO interpolant for the temporal component in the same way we will demonstrate for the spatial one in the following section.

Once we obtain the temporal reconstruction we use it to define the full spatio-temporal reconstruction, which is used in all computations involving the post-processor in Lemma 2.1. The procedure used to obtain the spatial component is based on the Weighted Essentially Non-Oscillatory (WENO) interpolation procedure which is derived and presented in detail in [JSB+19].

5.7. **WENO approximation.** In this section we present the procedure for obtaining the spatial component of the reconstruction using the WENO interpolation procedure of [JSB+19]. An advantage of this interpolant is that all its aspects (sub-stencil polynomials, linear and non-linear weights) have been modified for use on non-uniform grids. In addition, it has the other advantages of WENO-interpolants. These include high orders of approximation in regionwhere the solution is smooth and essentially non-oscillatory behavior in the vicinity of dicontinuities.

Consider a function $y(x)$ with a set of point values $\{y_j\}$ at locations $\{x_j\}$, where the grid is not necessarily uniform. We want to construct a third order WENO intepolating polynomial in an interval $[x_j, x_{j+1}]$ by using the 4-point stencil

$$S := \{x_{j-1}, \ldots, x_{j+2}\} \tag{5.5}$$

The interpolant is obtained as a convex combination of polynomials which are constructed on two 3-point sub-stencils, S_1 and S_2 of S, which are given by

$$\begin{aligned} S_1 &:= \{x_{j-1}, x_j, x_{j+1}\}, \\ S_2 &:= \{x_j, x_{j+1}, x_{j+2}\}. \end{aligned} \tag{5.6}$$

The polynomials are Lagrange interpolants over the sub-stencils:

$$\begin{aligned} p_1(x) &:= y_{j-1}\frac{(x - x_j)(x - x_{j+1})}{h_{j-1}(h_{j-1} + h_j)} + y_j\frac{(x - x_{j-1})(x - x_{j+1})}{h_{j-1}h_j} + y_{j+1}\frac{(x - x_{j-1})(x - x_j)}{(h_{j-1} + h_j)\,h_j} \quad \text{and} \\ p_2(x) &:= y_j\frac{(x - x_{j+1})(x - x_{j+2})}{h_j(h_j + h_{j+1})} + y_{j+1}\frac{(x - x_j)(x - x_{j+2})}{h_j h_{j+1}} + y_{j+2}\frac{(x - x_j)(x - x_{j+1})}{(h_j + h_{j+1})\,h_{j+1}} \end{aligned} \tag{5.7}$$

for $x \in [x_j, x_{j+1}]$. A polynomial approximation to $u(x)$, $p(x)$, can be obtained as a convex combination of the $p^{(i)}$. The WENO approach is such that $p(x)$ is a high order approximation in intervals where $u(x)$ is

smooth. $p(x)$ is obtained as a weighted sum of $p^{(1)}$ with the (linear) weights γ_1 and γ_2, each corresponding to a sub-stencil of the large stencil:

$$\begin{aligned} \gamma_1(x) &:= -\frac{x - x_{j+2}}{x_{j+2} - x_{j-1}} \quad \text{and} \\ \gamma_2(x) &:= \frac{x - x_{j-1}}{x_{j+2} - x_{j-1}}. \end{aligned} \tag{5.8}$$

The linear weights are positive and satisfy

$$\sum_i \gamma_i = 1. \tag{5.9}$$

Interested readers can find details on the construction of these weights in ([CFR05]) and [LSZ09]. If the solution is discontinuous inside a sub-stencil, we would like that stencil to have little contribution to ensure the non-oscillatory behaviour of the scheme. This is achieved by using the non-linear weights $\omega_i(x)$, which are obtained from the $\gamma_i(x)$ as follows:

$$\omega_j(x) := \frac{\alpha_j(x)}{\sum_{i=1}^{2} \alpha_i(x)}, \quad \alpha_i(x) := \frac{\gamma_i(x)}{\epsilon + \beta_i}, \tag{5.10}$$

where the β_i are the *smoothness indicators* for the sub-stencil to which they pertain. They are an indication of how non-smooth the solution is in the corresponding sub-stencil. If the solution is smooth in the sub-stencil S_j, then the relevant β_j is small and the relevant ω_j is close to the γ_j in S_j. If instead the solution has a discontinuity in S_j, then the β_j is large, leading to a small ω_j and ensuring the non-oscillatory behaviour.

The β_i which are used in this paper are given in [JSB+19] and are defined as

$$\begin{aligned} \beta_1 &:= (h_j + h_{j+1})^2 \left(\frac{\left| y'_{j+1} - y'_j \right|}{h_j} - \frac{\left| y'_j - y'_{j-1} \right|}{h_{j-1}} \right)^2 \quad \text{and} \\ \beta_2 &:= (h_{j-1} + h_j)^2 \left(\frac{\left| y'_{j+2} - y'_{j+1} \right|}{h_{j+1}} - \frac{\left| y'_{j+1} - y'_j \right|}{h_j} \right)^2 . \end{aligned} \tag{5.11}$$

The calculation of the y'_i is presented in detail in [JSB+19, §3.3.2]. Finally, the WENO approximation to $u(x)$ in the interval $[x_j, x_{j+1}]$ based on the stencil $S = S_1 \cup S_2 = \{x_{j-1}, x_j, x_{j+1}, x_{j+2}\}$ can be obtained as

$$p(x) := \omega_1 p_1(x) + \omega_2 p_2(x)\,. \tag{5.12}$$

Now we can define the spatio-temporal reconstruction in terms of the WENO approximation.

5.8. **Definition** (Spatio-temporal reconstruction)**.** Let $\widehat{\boldsymbol{U}}^t$ be a temporal reconstruction of the numerical solution $\boldsymbol{U}$ of (3.1). The spatio-temporal reconstruction $\widehat{\boldsymbol{U}}^{ts} \in \mathbb{V}_r^t\big(0, T; \mathbb{V}_q^s\big)$ in the interval $[x_j, x_{j+1}]$ is given by the WENO interpolant of $\widehat{\boldsymbol{U}}^t$, which is defined in (5.12).

5.9. **Remark** (Order of the reconstruction)**.** The conditions presented in Defn. 5.5 result in a polynomial which is of third order in the temporal variable. In contrast Defn. 5.8 can be used to obtain spatial reconstructions of arbitrary order in space, simply by using a higher order WENO interpolant. The limiting factor in the oder of the full spatio-temporal reconstruction will be the lowest order between the spatial and temporal steps. In this case, this will be the order of the temporal component (order three).

5.10. **Remark** (Boundary conditions)**.** Note that for periodic boundary conditions, quantities calculated in Defns. 5.5 and 5.8 that pertain to the boundaries - i.e. $j = 0$ and $j = M$ - should be equal. In the case of non-periodic boundary conditions - Neumann boundary conditions in particular - the computational domain is extended using an appropriate number of ghost nodes. For WENO schemes the number of ghost nodes depends on the WENO sub-stencil width. The values for the ghost nodes can be obtained by extrapolation based on the local WENO polynomials.

We illustrate the reconstruction procedure in Fig. 3. Firstly, we use Defn. 5.5 to obtain $\widehat{\boldsymbol{U}}^t(t)$ from $\boldsymbol{U}$, where the latter is produced by the chosen numerical discretisation to (2.6). This is the solid red line in the middle plot. Then, we use Defn. 5.8 to obtain $\widehat{\boldsymbol{U}}^{ts}(x,t)$ from $\widehat{\boldsymbol{U}}^t(t)$. This is the transparent red surface in the third plot.

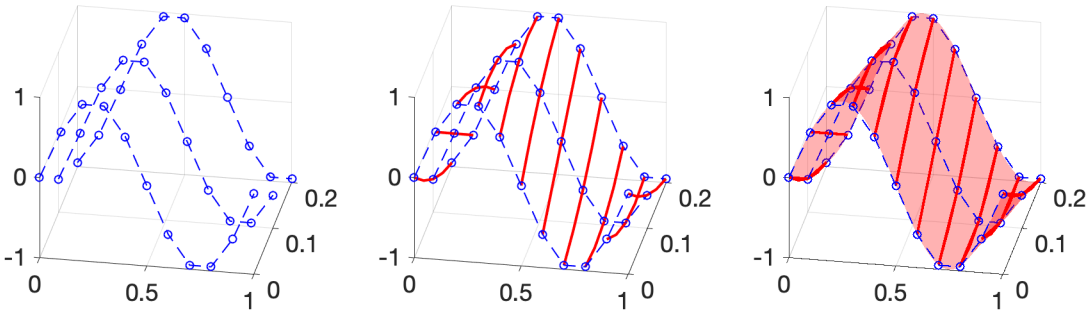


Fig. 3. Example of a reconstruction procedure. (Left) $\boldsymbol{U}^n$ for $n = 0, 1, 2$ produced by a Finite Difference scheme in 1D, with periodic boundary conditions and a sinusoidal initial condition (blue dashed lines). (Middle) Temporal reconstruction step $\widehat{\boldsymbol{U}}^t(t)$, depicted as solid, red lines. (Right) Spatio-temporal reconstruction step $\widehat{\boldsymbol{U}}^{ts}(x,t)$, depicted as a transparent, red surface.

6. Numerical verification

In this section we will study the asymptotic behaviour of the a posteriori bound and compare and contrast it with the behaviour of the error for two nonlinear examples: Burgers' equation and the shallow water equations. In both cases we are using periodic boundary conditions. The tests in this section are a preliminary step before the next section, in which the a posteriori bound is used as a refinement/coarsening criterion in adaptive tests. We will firstly present the bounds we will be testing for the non-linear problems we benchmark in this section, along with the numerical schemes we will use.

6.1. **Remark** (a posteriori bound for nonlinear problems)**.** The test cases we examine in this section are nonlinear so the post-processor from Lemma 2.1 is not applicable. Instead, we will use a different a posteriori estimate which is appropriate for nonlinear systems of hyperbolic conservation laws.

6.2. **Remark** (Reconstruction residual)**.** The reconstruction residual, $\boldsymbol{R}$, is used to compute the smooth post-processor that bounds the error of the problem from above in Thm. 2.1. We obtain $\boldsymbol{R}$ by substituting $\widehat{\boldsymbol{U}}^{ts}$ in (3.1):

$$-\widehat{\boldsymbol{R}}(x,t) := \partial_t \widehat{\boldsymbol{U}}^{ts}(x,t) + \partial_x \boldsymbol{f}\left(\widehat{\boldsymbol{U}}^{ts}(x,t)\right) \tag{6.1}$$

6.3. **Remark.** (Derivative notation) We use the convention that derivatives of a vector-valued function $\boldsymbol{u} = (u_1, \ldots, u_d)^T$, are understood component-wise:

$$\partial_x \boldsymbol{u} := (\partial_x u_1, \ldots, \partial_x u_d)^T, \tag{6.2}$$

where $(\cdots)^T$ denotes a column vector. Derivatives of a field, q, are denoted $\mathrm{D}q := (\partial_{u_1} q(\boldsymbol{u}), \ldots, \partial_{u_d} q(\boldsymbol{u}))$. The matrix of second derivatives is

$$\mathrm{D}^2 q(\boldsymbol{u}) := \begin{bmatrix} \partial_{u_1,u_1} q(\boldsymbol{u}) & \ldots & \partial_{u_1,u_d} q(\boldsymbol{u}) \\ \vdots & \ddots & \vdots \\ \partial_{u_d,u_1} q(\boldsymbol{u}) & \ldots & \partial_{u_d,u_d} q(\boldsymbol{u}) \end{bmatrix}. \tag{6.3}$$

6.4. **Lemma** (a posteriori error control for nonlinear systems of 1D conservation laws from [GMP15])**.** *Let $\boldsymbol{f} \in C^2\big(U, \mathbb{R}^d\big)$ satisfy (3.6) and let $\boldsymbol{u}$ be an entropy solution of (3.1) with periodic boundary conditions. Let $\widehat{\boldsymbol{U}}$ take values in $\mathcal{D}$ (which is a convex, compact subset of the state space). Then for $0 \leq t \leq T$ the error between $\boldsymbol{u}$ an $\widehat{\boldsymbol{U}}$ is given by*

$$\begin{aligned}\left\|\boldsymbol{u}(\cdot,t)-\widehat{\boldsymbol{U}}(\cdot,t)\right\|^2_{\mathrm{L}^2(I)} \leq & C_{\underline{\eta}}^{-1}\left(\|\boldsymbol{R}\|^2_{\mathrm{L}^2(I\times(0,t))}+C_{\overline{\eta}}\left\|\boldsymbol{u}_0-\widehat{\boldsymbol{U}}_0\right\|^2_{\mathrm{L}^2(I)}\right)\\ & \times\exp\left(\int_0^t \frac{C_{\overline{\eta}}C_{\overline{\boldsymbol{f}}}\left\|\partial_x\widehat{\boldsymbol{U}}(\cdot,s)\right\|_{\mathrm{L}^\infty(I)}+C_{\overline{\eta}}^2}{C_{\underline{\eta}}}\mathrm{d}s\right).\end{aligned} \tag{6.4}$$

The constants $C_{\underline{\eta}}$ and $C_{\overline{\eta}}$ represent the minimum and maximum absolute eigenvalues of $\mathrm{D}^2\eta$. Furthermore, $C_{\overline{\boldsymbol{f}}} := \left(\sum_i C_{\overline{\boldsymbol{f}}_i}\right)^{1/2}$, where $C_{\overline{\boldsymbol{f}}_i}$ is an upper bound for the absolute values of the eigenvalues of the ith component of $\boldsymbol{f}$.

6.5. **Corollary** (Stability and error control for Burgers' equation with periodic boundary conditions)**.** *Let the conditions of Lemma 6.4 hold with $f(u) := \frac{1}{2}u^2$, i.e. the scalar Burgers' equation. Suppose the initial value problems for u and $v := \widehat{U}$ are coupled with periodic boundary data. Then, the error between the two functions, $e := u - \widehat{U}$, satisfies the following bound for all $t \in [0,T]$:*

$$\|e(t)\|^2_{\mathrm{L}^2} \leq \exp\left(\int_0^t\left(\left\|\partial_x\widehat{U}(s)\right\|_{\mathrm{L}^\infty}+1\right)\mathrm{d}s\right)\left[\|e(0)\|_{\mathrm{L}^2}+\int_0^t\|R(s)\|^2_{\mathrm{L}^2}\,\mathrm{d}s\right] =: \omega(t)\,\mathscr{E}_b^2(t)\,, \tag{6.5}$$

where

$$-R := \partial_t\widehat{U}+\partial_x\left(\frac{\widehat{U}^2}{2}\right). \tag{6.6}$$

We test the bound (6.5) for three different schemes - Lax-Friedrichs, Lax-Wendroff and SSP3-WENO - with uniform temporal and spatial discretizations, i.e. $\tau^n := \tau\ \forall n$ and $h_j := h\ \forall j$. The reconstruction residual, (6.1), will be obtained by using Defn. 5.5 for the temporal component and Defn. 5.8 for the spatial component.

We have implemented the a posteriori bound with these numerical schemes in order to benchmark its behaviour before applying it to adaptive scenarios, where the bound is used as a driver for adaptivity.

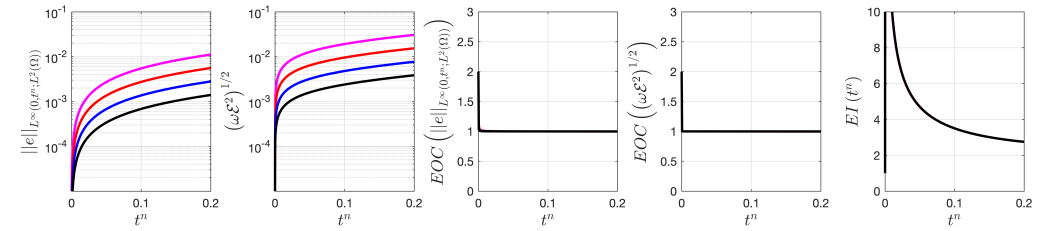
6.6. **Test 1: Scalar Inviscid Burgers' equation.** We will benchmark the performance of the a posteriori bound (6.5) in using the 1D inviscid Burgers' equation with smooth a initial condition and periodic boundary conditions as the test case:

$$\begin{aligned}\partial_t u+\partial_x\left(\frac{u^2}{2}\right)&=0, \qquad && \text{for } (x,t)\in[-\pi,\pi]\times(0,T].\\ u(x,0)&=-\sin(x) && \end{aligned} \tag{6.7}$$

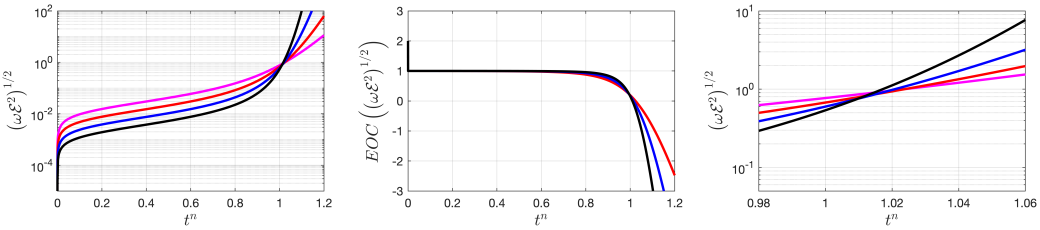
The exact solution in the pre-shock regime can be represented as an infinite sum of Bessel functions (see [GMP15]):

$$u(x,t) = -2\sum_{k=1}^{\infty}\frac{J_k(kt)}{kt}\sin(kx)\,, \tag{6.8}$$

where J_k denotes the kth Bessel function. This is a decaying sequence, so we can approximate the solution by truncating it (at say $k = 100$). The results are shown in Figs. 4a and 4b using the Lax-Friedrichs scheme

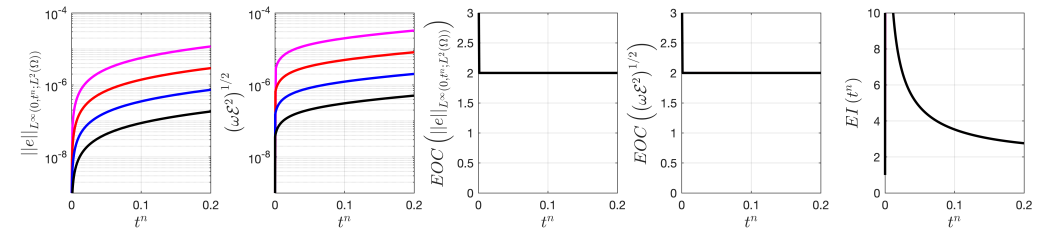


(a) Pre-shock.

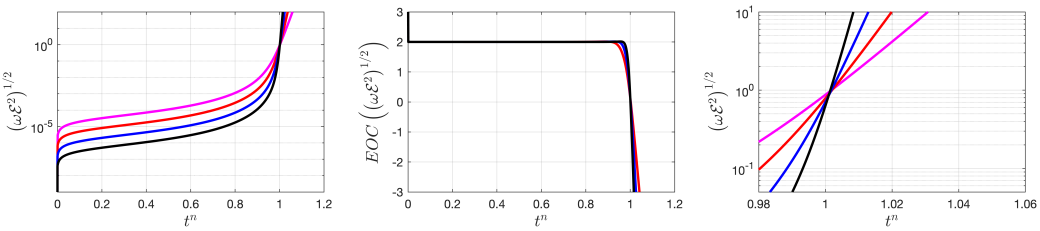


(b) Post-shock.

Fig. 4. Errors and asymptotic convergence rates for abound constructed using a bilinear interpolant for the Lax-Friedrichs scheme, (3.10), for Burgers' equation with sinusoidal initial conditions and periodic boundary conditions given by (6.7). The simulations were conducted over a family of meshes with discretisation parameter $h = 2^{-m}, m = 11, \dots 14$, with a timestep $\tau = \frac{h}{10}$. Note that the estimate is optimal prior to shock formation, and that it blows up once the shock forms, as the exponential factor in (6.5) blows up at $t \approx 1$.

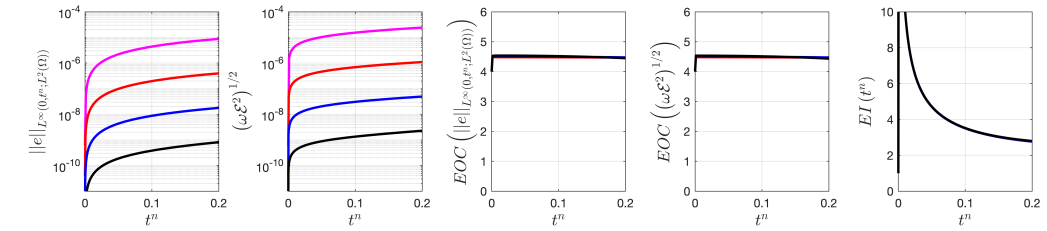


(a) Pre-shock.

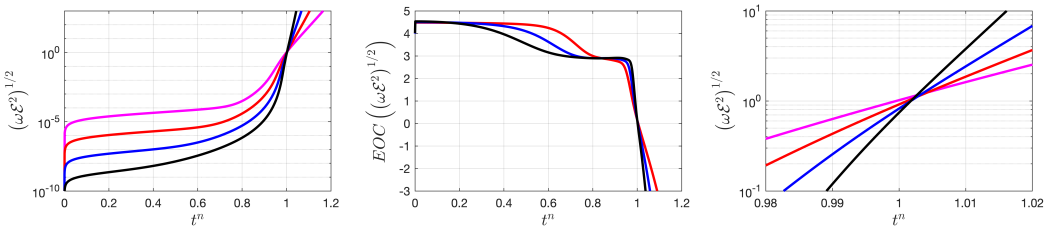


(b) Post-shock.

Fig. 5. Errors and asymptotic convergence rates for the Lax-Wendroff scheme, (3.13), for Burgers' equation with sinusoidal initial conditions and periodic boundary conditions given by (6.7). The simulations were conducted over a family of meshes with discretisation parameter $h = 2^{-m}, m = 9, \dots 12$, with a timestep $\tau = \frac{h}{10}$. The a posteriori bound is constructed using a Hermite interpolant in time and a WENO3 interpolant in space. The estimate is optimal prior to shock-formation. In the post-shock regime the bound blows up, because the exponential factor in (6.5) blows up at $t \approx 1$.



(a) Pre-shock.



(b) Post-shock.

Fig. 6. Errors and asymptotic convergence rates for the SSP3-WENO3 discretisation of Burgers' equation with a sinusoidal initial condition and periodic boundary conditions (6.7). The reconstruction is obtained using a Hermite temporal interpolant and a WENO3 spatial interpolant. The simulations were conducted over a family of meshes with discretisation parameter $h = 2^{-m}$, $m = 9, \dots 12$, with a timestep $\tau = \frac{h}{10}$. Notice that the estimator loses optimality gradually in the interval $0.2 \leq t \leq 0.8$. This is because the solution itself is losing regularity, only this time the scheme and the residual are both of sufficiently high order to capture this. At $t \approx 1$ the shock forms and the exponential factor in (6.5) blows up.

In order to explain the behaviour of the bound (6.5), we decouple the time-accumulation factor from the residual component of the post-processor (see Fig. 7). Notice that at $t \approx 1$ the exponential factor blows up rapidly as v_x blows up. The reason for this is that at $t = 1$ the solution starts forming a shock. This explains the behaviour for $t >= 1$.

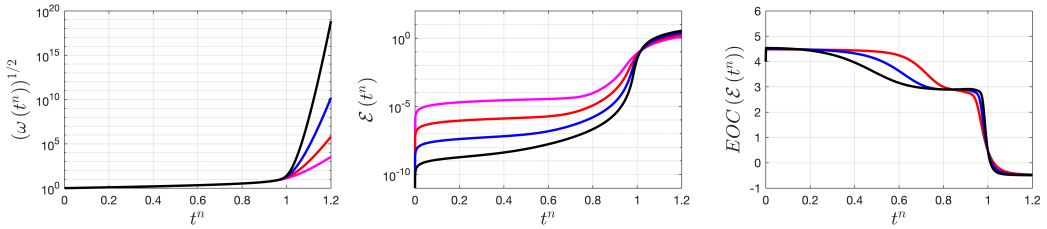


Fig. 7. Decoupling for the post-processor the cubic spatio-temporal interpolant of SSP3-WENO3 approximations of Burgers' equation with a sinusoidal initial condition and periodic boundary conditions (6.7). The simulations were conducted over a family of meshes with discretisation parameter $h = 2^{-m}, m = 9, \dots 12$, with a timestep $\tau = \frac{h}{10}$. The blow-up in the time accumulation term is because of the spatial derivative in the exponential factor, which becomes exceedingly large in the presence of shocks.

6.7. **Test 2: Shallow Water equation.** We benchmark the behaviour of the residual for the shallow water equations.

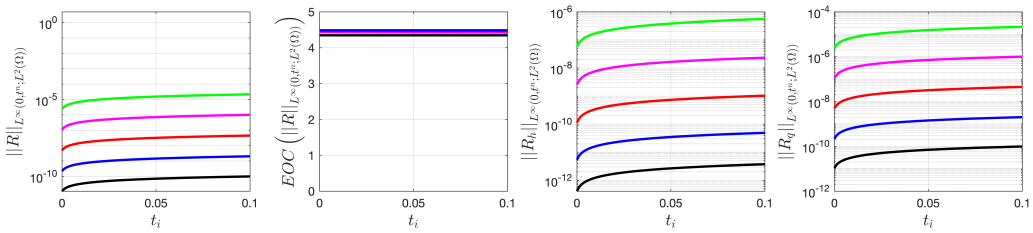


Fig. 8. Bench marking for the shallow-water equations with a sinusoidal initial condition and periodic boundary conditions. We use an SSP3-WENO3 scheme with $h = 2^{-m}, m = 9, \dots 12$, with a timestep $\tau = \frac{h}{10}$. The bound is constructed using a Hermite-WENO-3 interpolant. Observe that the residual maintains a high order of convergence throughout the simulation.

The last test in this section is presented in order to motivate adaptivity in the context of finite difference schemes for conservation laws. We highlight additional challenges that arise in implementing adaptivity, even for scalar examples in one spatial dimension.

6.8. **Test 3: Parasite detection in 1D.** In this test we investigate the resulting behaviour of (2.19) in the presence of parasitic waves and we verify the capability of the bound we construct using Defns 5.5 and 5.8 to detect such waves. Very briefly, parasitic waves are numerical artefacts which are generated whenever the numerical solution encounters some discontinuity in the numerical model. Such discontinuities include local changes in grid spacing (see [Vic81b]), abrupt changes in some aspect of the specific PDE model, such as a discontinuous change in coefficients (see [Tre82]) or even a change in the PDE used in different regions of the domain, e.g. advection in one region and advection-diffusion in another, (see the work of [GP17] for more details on model adaptivity).

We use a Crank-Nicholson in time Central-Space scheme on a piecewise non-uniform grid given by

$$h_j = x_{j+1} - x_j = \begin{cases} 2^{-(9)} & \text{if} \quad x_{j+1} > L/2, \\ 2^{-(10)} & \text{otherwise} \end{cases}. \tag{6.9}$$

6.9. **Remark** (Truncation error of the central difference quotient)**.** The truncation error of the usual central difference quotient,

$$u_x \approx \frac{U^n_{j+1} - U^n_{j-1}}{h_{j+1} + h_j}, \tag{6.10}$$

on a uniform grid is two globally. On a non-uniform grid, it is locally one wherever $[\![h_j]\!] := h_{j+1} - h_j \neq 0$. In order to ensure that the spatial discretisation is second order on a non-uniform grid, we use the modified quotient

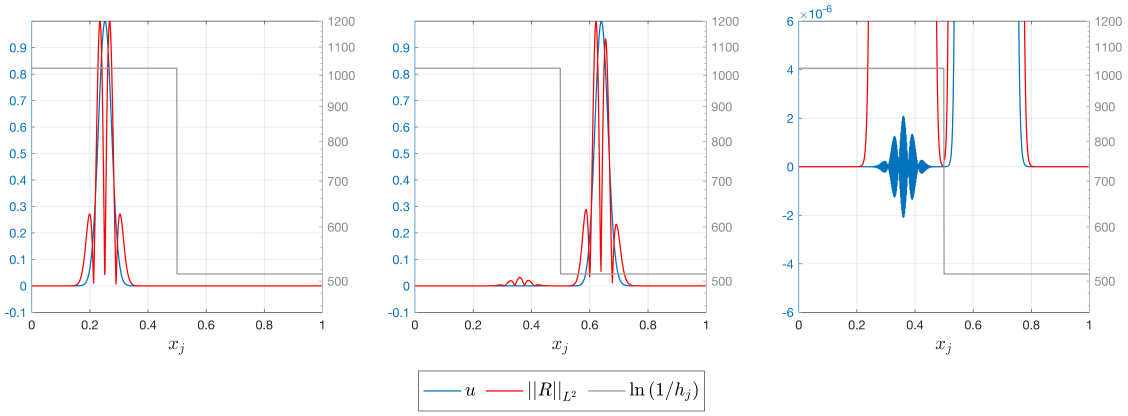
$$u_x \approx \left(\frac{1}{h_{j+1}} - \frac{1}{h_j + h_{j+1}}\right) U^n_{j+1} + \left(\frac{1}{h_j} - \frac{1}{h_{j+1}}\right) U^n_j + \left(\frac{1}{h_j + h_{j+1}} - \frac{1}{h_j}\right) U^n_{j-1}, \tag{6.11}$$

which is order two globally on a non-uniform grid as well and simplifies to (6.10) on a uniform grid.

In order to demonstrate the effect of the parasite, we plot $\|R\|_{\mathrm{L}^2(I_j)}$, for R given by (6.1) for an exponential initial condition,

$$u_0(x) = \exp\left(-100(x - 0.25)^2\right), \tag{6.12}$$

which makes the parasites more visible. The results are shown in Fig. 9.



(a) Before parasite formation. **(b)** After parasite formation. **(c)** Magnified Parasite.

Fig. 9. A parasitic wave which occurs to due an abrupt mesh change. We use a CNCS scheme on a grid given by (6.9) and a temporal-to-spatial step coupling of $\tau = \frac{h}{10}$. We plot the solution and the normalized L^2-norm of the local residual (6.1) before and after parasite formation. The parasitic wave is the highly oscillatory wave in the right-most plot, which is zoomed in. Notice that it is moving in the opposite direction from the solution. Such parasites can arise when implementing adaptivity and can rapidly pollute the computation. Notice that the residual detects and tracks the parasite.

In Fig.9 we see a parasite, which is the highly oscillatory wave-train shown in the bottom left plot. Notice that it travels in the opposite direction compared to the initial condition. Parasitic waves can rapidly pollute the computation. Notice that the bound we have constructed using Defns. 5.5 and 5.8 is able to detect the parasitic wave. The reader should note that implementing adaptivity in the presence of an existing grid discontinuity may exacerbate parasite formation and propagation.

7. Adaptive experiments

In this section we discuss practical implementation details for the solution of (3.1) in an adaptive context. We describe the setup we use to test the suitability of (6.1) as a criterion for mesh adaptivity in 1D and we compare and contrast the results produced from the adaptive simulation with results produced from a uniform grid.

We begin by presenting the adaptive algorithm we use. Firstly, we will explain how we facilitate the comparison of the adaptive numerical simulation with the uniform one and then we present the marking and the refinement/coarsening strategies.

7.1. **Adaptive Algorithm.** Our adaptive algorithm is of SOLVE $\rightarrow$ ESTIMATE $\rightarrow$ MARK $\rightarrow$ REFINE type.

7.2. **Remark** (Maximum number of refinements)**.** For the purposes of this paper we allow a maximum of four refinement levels relative to the initial, uniform triangulation.

There are potentially several ways to compare the performance of a numerical solution on a uniform and an adaptive mesh. In this case we will use what we will refer to as an *equivalent uniform mesh.*

7.3. **Definition** (Equivalent Uniform Mesh)**.** We define a uniform mesh to be equivalent to an adaptive mesh if it has the same cumulative number of degrees of freedom, which we define as

$$\sum_{n=0}^{N} N_{dof}(t^n)\,. \tag{7.1}$$

We find this number by firstly running the simulation on the adaptive mesh and recording the number of dof at each time-step. We then average this over the number of time steps and set the resulting value to be the number of degrees of freedom for the equivalent uniform mesh.

7.4. **Marking.** The criterion for marking cells for refinement/coarsening is based is a maximum strategy. We refine cells wherein the value of the local indicator is larger than some multiple of the maximum value of the local residual and coarsen cells where it is lower. We do nothing in cases where the local value of the indicator falls in between the two values. Lastly, we do not coarsen a cell marked if its sibling is marked for refinement at the same time-step.

This strategy is described in detail in [SS05, §1.5] and it is modified for time-dependent problems for the purposes of these experiments. Briefly, let η_S denote the local residual term $\|R\|_{\mathrm{L}^2(I_j)}$ in a 'cell' $S := I_j = [x_j, x_{j+1}]$ for $S \in S_K$, where S_K is the initial parent triangulation. We set two predefined tolerances γ_r and γ_c for refining and coarsening respectively. We mark a cell for refinement if

$$\eta_S \geq \gamma_r \max_{S'} \eta_{S'},\ S' \in S_k \tag{7.2}$$

and we mark for coarsening coarsen if

$$\eta_S \leq \gamma_c \max_{S'} \eta_{S'},\ S' \in S_k. \tag{7.3}$$

The reader should note that there are several ways of choosing γ_c and γ_r. We chose them empirically as $\gamma_c = .05e-10$ and $\gamma_r = 0.5e-8$. We summarize the marking-refinement/coarsening process in Algorithm 1.

Algorithm 1 Mesh Adaptivity

Require: Maximum number of refinement levels relative to initial triangulation, refinement parameter γ_r and coarsening parameter γ_c.
while $t^n < T$ **do**
  set $S_K^{(0)} = S_k$, the initial grid for the time-step t^n.
  Solve (3.9) on $S_K^{(0)}$ and compute the local indicator η_S for all $S \in S_K^{(0)}$.
  Set $\eta_{\max} := \max_{S' \in \mathcal{S}_k} \eta_{S'}$.
  for $S \in S_K^{(0)}$ **do**
    if $\eta_S > \gamma_r \eta_{\max}$ **then**
      Mark S for refiment
    end if
  end for
  for $S \in S_K^{(0)}$ **do**
    if $\eta_S < \gamma_c \eta_{\max}$ **then**
      if $S \notin S_K$ **and** the siblings of S are not marked for refinement **then**
        Mark S for coarsening
      end if
    end if
  end for
  for $S \in S_K^{(0)}$ **do**
    if S is marked for refinement **then**
      Create two children of S
      Prolong $\boldsymbol{U}$ over S and assign corresponding values to children nodes
      Prolong the grid $\{x_j\}$ assigning corresponding values to children nodes
    elseif S is marked for coarsening
      Restrict $\boldsymbol{U}$ by assigning the relevant values from S and its sibling to their parent
      Restict the grid $\{x_j\}$ by assigning corresponding values from S and its sibling to their parent
      Delete S and its sibling
    end if
  end for
  Recompute both the error and the bound from (2.19) and record them as the values for time step t^n
  do $n := n + 1$
end while
end

8. Adaptive experiments

In this section we describe the numerical experiments we run to test the a posteriori bounds as criteria for adaptivity. We use two different problems: a linear advection problem with a discontinuous initial condition and periodic boundary conditions, and a shallow water dam-break problem with zero-neumann boundary conditions. In this way we benchmark the performance of the indicator in more challenging settings than in previous sections.

8.1. **Remark** (Grid-spacing for the adaptive mesh)**.** We start the simulations after having globally refined the entire grid the maximum allowable number of times. Then, the indicator detects where this is excessive and coarsens locally. This is the reason for the steep decrease in the number of dof in the left plots in Fig. 10 and Fig. 12.

8.2. **Remark** (Time-step for the adaptive mesh)**.** In order to maintain numerical stability, in the absence of an adaptive mechanism for the time-step, we couple the time-step to the smallest spatial step present in the computation, to maintain numerical stability.

8.3. **Linear advection.** We run a benchmarking experiment experiment using the linear advection equation

$$u_t + u_x = 0 \tag{8.1}$$

over a domain $\Omega = [0,1]$ with $T = 0.5$ using periodic boundary conditions and a discontinuous initial condition given by

$$u_0(x) = \begin{cases} 1 & \text{if} \quad |x - .25| \leq 0.125 \\ 0 & \text{if} \quad |x - .25| > 0.125. \end{cases} \tag{8.2}$$

We discretise the problem using a Forward-Time Backward Space scheme given by (2.17) for both the adaptive and the uniform case. The residual in this test case is constructed using a bilinear Lagrange interpolant. Lastly, the grid-spacing for the equivalent uniform mesh (see Rem. 7.3) is found to have 741 degrees of freedom, i.e. $0 = x_0 < \cdots < x_{740} = 1$. The results are shown in Fig. 10.

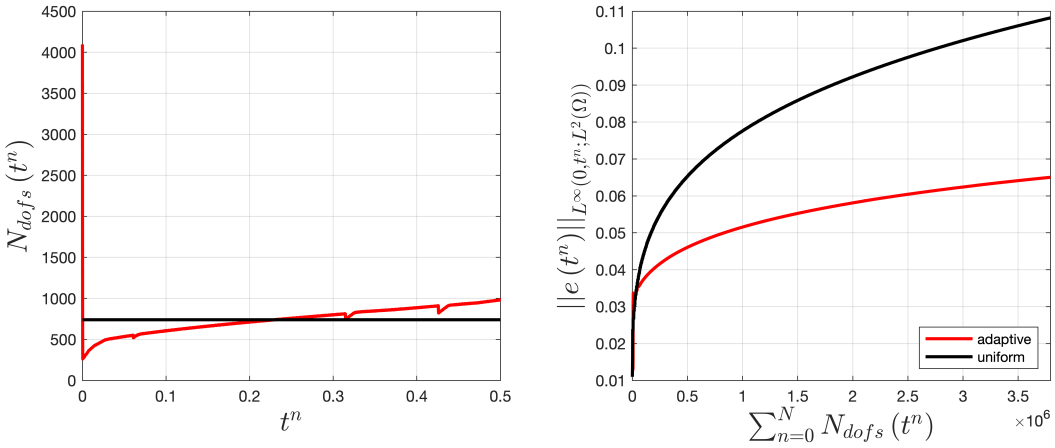


Fig. 10. An adaptive simulation of the linear advection equation with an FTBS discretization, periodic boundary conditions and a step initial condition given by (8.2). The simulation starts at the maximum allowable refinement level (we set this to be 4) with an initial (fine) grid with spacing $h = 2^{-12}$ and a temporal step $\tau = \frac{2^{-10}}{10}$, which remains constant throughout the simulation. The equivalent uniform grid has 741 degrees of freedom. The indicator used in this case is a simple bilinear Lagrange indicator. Notice that the indicator automatically coarsens the grid at the beginning of the simulation, resulting in a steep decrease in the number of decrease of freedom in parts of the domain which are away from the discontinuity. Also notice that the adaptive mesh compares favorably with an equivalent uniform grid as it consistently maintains a lower level of error throughout the major part of the simulation.

8.4. **Shallow water equations.** In this case we conduct a numerical experiment to test the residual term $\boldsymbol{R}$ in (6.4) as a local refinement criterion for the shallow water equations using a dam-break initial condition and free outflow conditions. The model we test is given by

$$\begin{array}{ccccc} \eta_t & + & (\eta v)_x & = & 0 \\ (\eta v)_t & + & \left(\eta v^2 + \frac{1}{2} g \eta^2\right)_x & = & 0, \end{array} \tag{8.3}$$

equipped with the initial conditions

$$\begin{aligned} h(x,0) &= \begin{cases} h_0 & \text{for} \quad x \leq x_0 \\ h_1 & \text{for} \quad x > x_0, \end{cases} \\ v(x,0) &= v_0(x)\,. \end{aligned} \tag{8.4}$$

over a domain $\Omega = [0, 32\pi]$ and $T \approxeq 10$. We use $v_0(x,0) = 0$, $x_0 = 30$, $h_0 = 0.2$ and $h_1 = 0.1$ for the initial condition. The initial fine mesh is given by $h = 32\pi \times 2^{-14}$ and a constant time-step $\tau = \frac{32\pi \times 2^{-12}}{10}$. We show an evolution of this solution in Fig. 11. The equivalent uniform mesh is found to have 1622 degrees of freedom, i.e. $0 = x_0 < \cdots < x_{1621} = 32\pi$. The benchmarking results are shown in Figure 12.

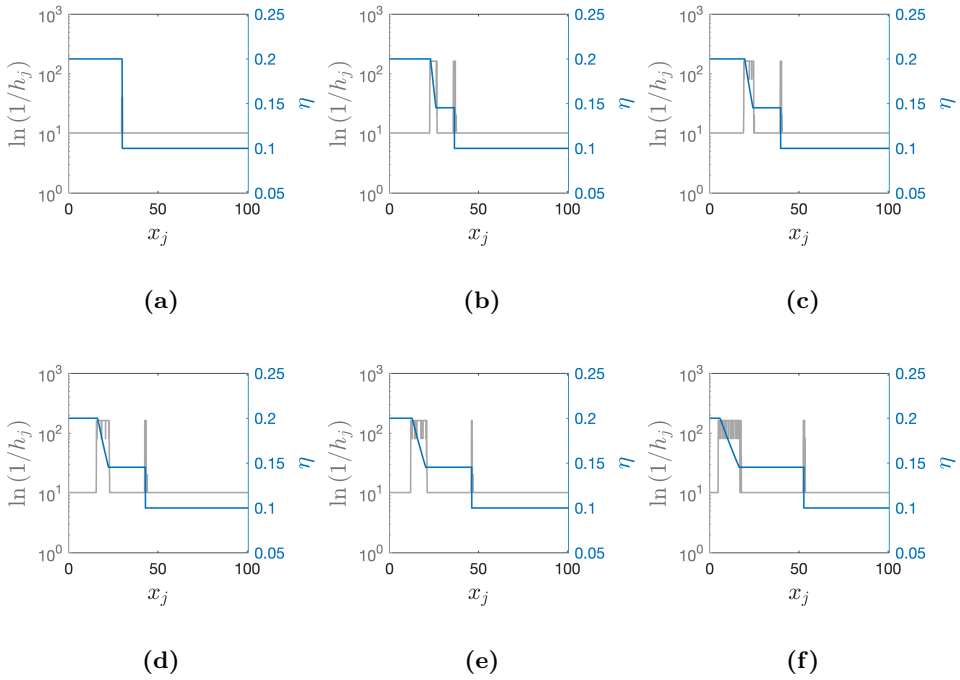


Fig. 11. Evolution of the surface elevation (blue line) for the shallow water dam break problem, using and SSP3-WENO3 spatio temporal discretization. The logarithm of the reciprocal of the local grid-spacing is overlaid with a grey line. We use a Hermite-WENO3 spatio-temporal reconstruction to construct the residual, R (see (6.1)). The simulation starts at the maximum allowable refinement level with a uniform grid with spatial step $h = 32\pi \times 2^{-14}$ and a temporal step $\tau = \frac{32\pi \times 2^{-12}}{10}$ which remains constant throughout the simulation. Notice that the residual accurately detects regions where refinement is required - such as in the vicinity of the shock and the rarefaction) and where coarsening is appropriate.

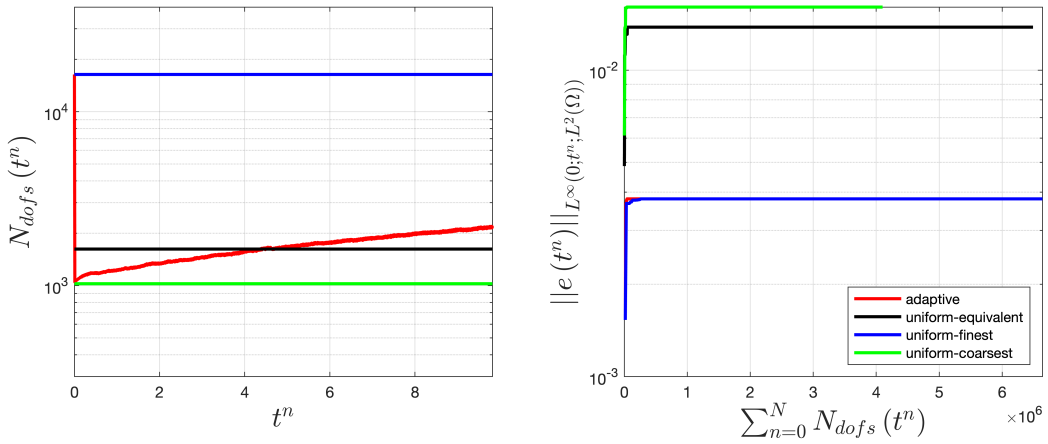


Fig. 12. An adaptive simulation of the shallow water equations for the dam-break problem using an SSP3-WENO3 discretization and free outflow boundary conditions. The simulation starts at the maximum allowable refinement level with a uniform grid with $h = 32\pi \times 2^{-14}$ and a temporal step $\tau = \frac{32\pi \times 2^{-12}}{10}$ which remains constant throughout the simulation. The equivalent uniform grid has 1622 degrees of freedom. The residual is constructed using a bilinear Lagrange indicator. Notice that the indicator automatically coarsens the grid at the beginning of the simulation, resulting in a steep decrease in the number of freedom away from the discontinuity. The behaviour in the adaptive case is almost identical with that of the finest resolution despite the considerable difference in degrees of freedom (the two results are almost entirely coincident on the right plot). This is because, for this problem, the features of refinement interest are concentrated in small areas, allowing for good resolution with relatively few degrees of freedom.

9. Conclusions

The main contribution from this article is a framework for constructing reliable, optimal a posteriori error estimates for classes of Finite Difference schemes, which have not received as much attention in the context of a posteriori estimates as FE and FV. The framework is generally applicable: it does not depend on the specific choice of the underlying FD scheme.

The methodology incorporates both the numerical solution and information from the FD scheme itself, thereby facilitating the construction of high order a aposteriori estimates using reconstruction techniques. This is a desirable property as it enables the user to construct optimal estimates for high order FD schemes. At the moment, this is limited to order three on account of the approach taken for the temporal component. Despite this limitation, the method for exceeding order three has been delineated. We used this with success for the spatial component. This will be incorporated for the temporal component in future work.

We demonstrate that the obtained estimates possess desirable characteristics using a range of numerical tests with both linear and nonlinear, scalar and vectorial examples. In these tests, the estimates show optimal convergence characteristics while the solution is smooth, but break down in the presence of shocks. This is because they contain terms which blow up when shocks appear and as the mesh size goes to zero.

We also show that the residuals constructed using the methodology we propose can be used as local mesh refinement criteria with favourable results relative to equivalent uniform grids. We demonstrate this using a scalar linear problem and a nonlinear system of conservation laws in one dimension.

The behaviour in the post shock regime, pertaining not only to optimality, but indeed to convergence, remains a future challenge.

References

[ABF88] David C Arney, Rupak Biswas, and Joseph E Flaherty. A posteriori error estimation of adaptive finite difference schemes for hyperbolic systems. Technical report, US Army Armanent Research Development and Engineering Center, Watervliet NY, 1988.

[AO11] Mark Ainsworth and J Tinsley Oden. *A posteriori error estimation in finite element analysis*, volume 37. John Wiley & Sons, 2011.

[BHO18] Timothy Barth, Raphaèle Herbin, and Mario Ohlberger. Finite volume methods: foundation and analysis. *Encyclopedia of Computational Mechanics Second Edition*, pages 1–60, 2018.

[BO84] Marsha J Berger and Joseph Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of computational Physics*, 53(3):484–512, 1984.

[CCL94] Bernardo Cockburn, Frédéric Coquel, and Philippe LeFloch. An error estimate for finite volume methods for multidimensional conservation laws. *mathematics of computation*, 63(207):77–103, 1994.

[CCL95] Bernardo Cockburn, Frédéric Coquel, and Philippe G LeFloch. Convergence of the finite volume method for multidimensional conservation laws. *SIAM Journal on Numerical Analysis*, 32(3):687–705, 1995.

[CET14] James B Collins, Don Estep, and Simon Tavener. A posteriori error estimation for the lax–wendroff finite difference scheme. *Journal of Computational and Applied Mathematics*, 263:299–311, 2014.

[CFR05] Elisabetta Carlini, Roberto Ferretti, and Giovanni Russo. A weighted essentially nonoscillatory, large time-step scheme for hamilton–jacobi equations. *SIAM Journal on Scientific Computing*, 27(3):1071–1091, 2005.

[CG95] Bernardo Cockburn and Huiing Gau. A posteriori error estimates for general numerical methods for scalar conservation laws. *Mat. Aplic. Comp*, 14(1):37–47, 1995.

[CG14] Qingshan Chen and Max Gunzburger. Goal-oriented a posteriori error estimation for finite volume methods. *Journal of Computational and Applied Mathematics*, 265:69–82, 2014.

[DGPR19] Andreas Dedner, Jan Giesselmann, Tristan Pryer, and Jennifer K Ryan. Residual estimates for post-processors in elliptic problems. *arXiv preprint arXiv:1906.04658*, 2019.

[DMO07] Andreas Dedner, Charalambos Makridakis, and Mario Ohlberger. Error control for a class of runge–kutta discontinuous galerkin methods for nonlinear conservation laws. *SIAM Journal on Numerical Analysis*, 45(2):514–538, 2007.

[Eva98] Lawrence C. Evans. *Partial differential equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 1998.

[GMP15] Jan Giesselmann, Charalambos Makridakis, and Tristan Pryer. A posteriori analysis of discontinuous galerkin schemes for systems of hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 53(3):1280–1303, 2015.

[God59] Sergei Konstantinovich Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3):271–306, 1959.

[GP17] Jan Giesselmann and Tristan Pryer. A posteriori analysis for dynamic model adaptation in convection-dominated problems. *Mathematical Models and Methods in Applied Sciences*, 27(13):2381–2423, 2017.

[GST01] Sigal Gottlieb, Chi-Wang Shu, and Eitan Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM review*, 43(1):89–112, 2001.

[HEOC87] Ami Harten, Bjorn Engquist, Stanley Osher, and Sukumar R Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, iii. In *Upwind and high-resolution schemes*, pages 218–290. Springer, 1987.

[Joh90] Claes Johnson. Adaptive finite element methods for diffusion and convection problems. *Computer Methods in Applied Mechanics and Engineering*, 82(1-3):301–322, 1990.

[JS95] Claes Johnson and Anders Szepessy. Adaptive finite element methods for conservation laws based on a posteriori error estimates. *Communications on Pure and Applied Mathematics*, 48(3):199–234, 1995.

[JS96] Guang-Shan Jiang and Chi-Wang Shu. Efficient implementation of weighted eno schemes. *Journal of computational physics*, 126(1):202–228, 1996.

[JSB+19] Gioele Janett, Oskar Steiner, Ernest Alsina Ballester, Luca Belluzzi, and Siddhartha Mishra. A novel fourth-order weno interpolation technique-a possible new tool designed for radiative transfer. *Astronomy & Astrophysics*, 624:A104, 2019.

[JT97] B Cockburn C Johnson and C-W Shu E Tadmor. Advanced numerical approximation of nonlinear hyperbolic equations. 1997.

[JT98] Guang-Shan Jiang and Eitan Tadmor. Nonoscillatory central schemes for multidimensional hyperbolic conservation laws. *SIAM Journal on Scientific Computing*, 19(6):1892–1917, 1998.

[KR94] Dietmar Kröner and Mirko Rokyta. Convergence of upwind finite volume schemes for scalar conservation laws in two dimensions. *SIAM journal on numerical analysis*, 31(2):324–343, 1994.

[L+02] Randall J LeVeque et al. *Finite volume methods for hyperbolic problems*, volume 31. Cambridge university press, 2002.

[Lax54] Peter D Lax. Weak solutions of nonlinear hyperbolic equations and their numerical computation. *Communications on pure and applied mathematics*, 7(1):159–193, 1954.

[LeV92] Randall J LeVeque. *Numerical methods for conservation laws*, volume 132. Springer, 1992.

[LOC94] Xu-Dong Liu, Stanley Osher, and Tony Chan. Weighted essentially non-oscillatory schemes. *Journal of computational physics*, 115(1):200–212, 1994.

[LSZ09] Yuan-yuan Liu, Chi-wang Shu, and Meng-ping Zhang. On the positivity of linear weights in weno approximations. *Acta Mathematicae Applicatae Sinica, English Series*, 25(3):503–538, 2009.

[LT11] David Long and John Thuburn. Numerical wave propagation on non-uniform one-dimensional staggered grids. *Journal of Computational Physics*, 230(7):2643–2659, 2011.

[LVW21] Richard Liska, Pavel Váchal, and Burton Wendroff. Lax-wendroff methods on highly non-uniform meshes. dedicated to the memory of blair swartz (1932–2019). *Applied Numerical Mathematics*, 163:167–181, 2021.

[Mak07] Charalambos Makridakis. Space and time reconstructions in a posteriori analysis of evolution problems. In *ESAIM: Proceedings*, volume 21, pages 31–44. EDP Sciences, 2007.

[MM05] Keith W Morton and David Francis Mayers. *Numerical solution of partial differential equations: an introduction*. Cambridge university press, 2005.

[NT90] Haim Nessyahu and Eitan Tadmor. Non-oscillatory central differencing for hyperbolic conservation laws. *Journal of computational physics*, 87(2):408–463, 1990.

[SCR16] Matteo Semplice, Armando Coco, and Giovanni Russo. Adaptive mesh refinement for hyperbolic systems based on third-order compact weno reconstruction. *Journal of Scientific Computing*, 66(2):692–724, 2016.

[Shu98] Chi-Wang Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In *Advanced numerical approximation of nonlinear hyperbolic equations*, pages 325–432. Springer, 1998.

[Shu20] Chi-Wang Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes. *Acta Numerica*, 29:701–762, 2020.

[SL18] Matteo Semplice and Raphaël Loubère. Adaptive-mesh-refinement for hyperbolic systems of conservation laws based on a posteriori stabilized high order polynomial reconstructions. *Journal of Computational Physics*, 354:86–110, 2018.

[SO88] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of computational physics*, 77(2):439–471, 1988.

[SO89] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, ii. In *Upwind and High-Resolution Schemes*, pages 328–374. Springer, 1989.

[SS05] Alfred Schmidt and Kunibert G Siebert. Design of adaptive finite element software. *Lecture Notes in Computational Science and Engineering. Springer-Verlag, Berlin*, 2005.

[Tre82] Lloyd N Trefethen. Group velocity in finite difference schemes. *SIAM review*, 24(2):113–136, 1982.

[Ver13] Rüdiger Verfürth. *A posteriori error estimation techniques for finite element methods*. OUP Oxford, 2013.

[Vic81a] Robert Vichnevetsky. Energy and group velocity in semi discretizations of hyperbolic equations. 1981.

[Vic81b] Robert Vichnevetsky. Propagation through numerical mesh refinement for hyperbolic equations. 1981.

[VL73] Bram Van Leer. Towards the ultimate conservative difference scheme i. the quest of monotonicity. In *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics*, pages 163–168. Springer, 1973.

[VL74] Bram Van Leer. Towards the ultimate conservative difference scheme. ii. monotonicity and conservation combined in a second-order scheme. *Journal of computational physics*, 14(4):361–370, 1974.

[VL77a] Bram Van Leer. Towards the ultimate conservative difference scheme iii. upstream-centered finite-difference schemes for ideal compressible flow. *Journal of Computational Physics*, 23(3):263–275, 1977.

[VL77b] Bram Van Leer. Towards the ultimate conservative difference scheme. iv. a new approach to numerical convection. *Journal of computational physics*, 23(3):276–299, 1977.

[VL79] Bram Van Leer. Towards the ultimate conservative difference scheme. v. a second-order sequel to godunov's method. *Journal of computational Physics*, 32(1):101–136, 1979.

Tristan Pryer Department of Mathematical Sciences, University of Bath, Bath BA2 7AY, UK tmp38@bath.ac.uk.

Georgios Sialounas Department of Mathematics and Statistics, University of Reading, Reading, RG6 6AX, UK gs1511@ic.ac.uk.