Jacobian-Free Explicit Multiderivative Runge–Kutta Methods for Hyperbolic Conservation Laws

CHOUCHOULIS, Jeremy; SCHUETZ, Jochen & ZEIFANG, Jonas (2022)
Jacobian-Free Explicit Multiderivative Runge–Kutta Methods for Hyperbolic Conservation Laws. In: JOURNAL OF SCIENTIFIC COMPUTING, 90 (3) (Art N° 96).

DOI: 10.1007/s10915-021-01753-z
Handle: http://hdl.handle.net/1942/36712

Journal of Scientific Computing manuscript No.
(will be inserted by the editor)

Jacobian-free explicit multiderivative Runge-Kutta methods for hyperbolic conservation laws

Jeremy Chouchoulis · Jochen Schütz · Jonas Zeifang

Received: date / Accepted: date

Abstract Based on the recent development of Jacobian-free Lax-Wendroff (LW) approaches for solving hyperbolic conservation laws [Zorio, Baeza and Mulet, Journal of Scientific Computing 71:246-273, 2017], [Carrillo and Parés, Journal of Scientific Computing 80:1832-1866, 2019], a novel collection of explicit Jacobian-free multistage multiderivative solvers for hyperbolic conservation laws is presented in this work. In contrast to Taylor time-integration methods, multiderivative Runge-Kutta (MDRK) techniques achieve higher-order of consistency not only through the excessive addition of higher temporal derivatives, but also through the addition of Runge-Kutta-type stages. This adds more flexibility to the time integration in such a way that more stable and more efficient schemes could be identified. The novel method permits the practical application of MDRK schemes. In their original form, they are difficult to utilize as higher-order flux derivatives have to be computed analytically. Here we overcome this by adopting a Jacobian-free approximation of those derivatives. In this paper, we analyze the novel method with respect to order of consistency and stability. We show that the linear CFL number varies significantly with the number of derivatives used. Results are verified numerically on several representative testcases.

Keywords Hyperbolic conservation laws · Multiderivative Runge-Kutta · Lax-Wendroff · Finite differences

Mathematics Subject Classification 65M06 · 65M08 · 65M12 · 35L65

1 Introduction

In this work, we present a novel discretization method for the numerical approximation of one-dimensional hyperbolic conservation laws on domain $\Omega \subset \mathbb{R}$,

$$w_t + f(w)_x = 0\,, \qquad \text{on} \quad (x,t) \in \Omega \times (0, T_{\text{end}}]\,, \tag{1}$$

Jeremy Chouchoulis, Jochen Schuetz, Jonas Zeifang
E-mail: firstname.lastname@uhasselt.be

Hasselt University, Faculty of Sciences & Data Science Institute, Agoralaan Gebouw D, BE-3590 Diepenbeek, Belgium

$$w(x,0) = w_0(x)\,.$$

Our primary interest is on temporal integration. In recent years, there has been quite some progress on the further development of the *multiderivative* paradigm for temporal integration, see, e.g., [5, 6, 11, 27, 30] and the references therein. Assume that one is given a scalar ODE, e.g.,

$$y'(t) = \varPhi(y) \tag{2}$$

for some flux function $\varPhi$. Multiderivative schemes make use of not only $\varPhi$, but also of the quantities $y''(t) \equiv \varPhi'(y)\varPhi$, $y'''(t) \equiv \dots$ and so on. Using this approach, one can derive stable, high-order and storage-efficient schemes very easily [28]. This can be extended to partial differential equations (PDEs) with a time-component, such as Eq. (1), depending on the method either directly through the method of lines-discretization [27] or through a Lax-Wendroff procedure, see, e.g., [3, 4, 16, 18, 22, 36]. The Lax-Wendroff method expresses temporal derivatives of the unknown function w in terms of the fluxes through the Cauchy-Kowalevskaya procedure. As an example, we consider – for simplicity given that f is scalar – the second time-derivative of w. Due to Eq. (1), there holds

$$w_{tt} = -(f(w)_x)_t = -(f(w)_t)_x, \tag{3}$$

and

$$f(w)_t = f'(w)w_t = -f'(w)f(w)_x;$$

hence

$$w_{tt} = \left(f'(w)f(w)_x\right)_x = 2f'(w)f''(w)w_x^2 + f'(w)^2 w_{xx}. \tag{4}$$

Already at this stage, one can see that this approach is very tedious as it necessitates highly complex symbolic calculations.

Still, the potential LW-methods bear is very well recognized among researchers. Over the last two decades, plenty of authors have put effort into developing high-order variants of the LW-method for nonlinear systems. Particularly the ADER (Arbitrary order using DERivatives) methods, see, e.g., [8–10, 29, 33, 34] and the references therein, gained a lot of interest. Also, higher-order extensions of the LW-method using WENO and discontinuous Galerkin (DG) reconstructions were investigated [13, 15, 19, 22–24].

Our essential intent of this paper is to make explicit multistage multiderivative solvers more accessible as a means to solve PDEs. Although such solvers have been theoretically studied since the early 1940's (see [30] for an extensive review), the schemes have not been put much to practice, which is most likely due to the necessary cumbersome calculation of flux derivatives. In [3], Carrillo and Parés have, based on the earlier work [36], developed the *compact approximate Taylor* (CAT) method to circumvent having to symbolically compute flux derivatives. By means of an automatic procedure, the higher-order temporal derivatives of w, such as in Eq. (4), are approximated. Their work is based on Taylor methods, i.e., time integration is given by

$$w(x,t^{n+1}) = w(x,t^n) + \sum_{k=1}^{\mathtt{r}} \frac{\varDelta t^k}{k!}\partial_t^k w(x,t^n) + \mathcal{O}(\varDelta t^{\mathtt{r}+1})\,.$$

In this work, we extend their approach to more general multiderivative integration methods, more precisely, to multiderivative Runge-Kutta (MDRK) methods.

The paper is structured in the following manner: In Sect. 2 multiderivative Runge-Kutta (MDRK) time integrators for ODEs are introduced, given that they form the central mechanism of this work. Thereafter, in Sect. 3 we shortly revisit the Jacobian-free approach of the CAT method and introduce the explicit Jacobian-free MDRK solver for hyperbolic conservation laws, termed MDRKCAT. After describing the numerical scheme, in Sect. 4 we prove consistency, and in Sect. 5 analyze linear stability. Via several numerical cases we verify and expand on the theoretical results in Sect. 6. At last, we draw our conclusions and discuss future perspectives in Sect. 7.

2 Explicit multiderivative Runge-Kutta solvers

We start by considering the system of ODEs defined by Eq. (2) in which Φ is a function of the solution variable $y \in \mathbb{R}^{\mathtt{m}}$. In order to apply a time-marching scheme, we discretize the temporal domain with a fixed timestep Δt by iterating N amount of steps such that $\Delta t = T_{\text{end}}/N$. Consequently, we define the time levels by

$$t^n := n\Delta t \qquad 0 \leq n \leq N.$$

Remark 1 Note that, although the fully space-time-discrete algorithm (Alg. 1) *seems* to have a multistep flavour, this is ultimately not the case. It is therefore of no necessity to consider a uniform timestep, which is also demonstrated numerically in Sect. 6.

The central class of time integrators in this work are *explicit* multiderivative Runge-Kutta (MDRK) methods. These form a natural generalization of classical explicit Runge-Kutta methods by adding extra temporal derivatives of $\Phi(w)$. The additional time derivatives can be recursively calculated via the chain rule, there holds

$$\frac{\mathrm{d}^k}{\mathrm{d}t^k}\Phi\left(y\right) = \frac{\mathrm{d}^{k-1}}{\mathrm{d}t^{k-1}}\left(\Phi'(y)\dot{y}\right) = \frac{\mathrm{d}^{k-1}}{\mathrm{d}t^{k-1}}\left(\Phi'(y)\Phi(y)\right).$$

For a more detailed description, we refer to [30]. To present our ideas, let us formally define the MDRK scheme as follows:

Definition 1 ([30, Def. 2]) An explicit q-th order accurate $\mathtt{r}$-derivative Runge-Kutta scheme using $\mathtt{s}$ stages ($\mathtt{r}$DRKq-$\mathtt{s}$) is any method which can be formalized as

$$y^{n,l} := y^n + \sum_{k=1}^{\mathtt{r}} \Delta t^k \sum_{\nu=1}^{l-1} a_{l\nu}^{(k)} \frac{\mathrm{d}^{k-1}}{\mathrm{d}t^{k-1}}\Phi\left(y^{n,\nu}\right) \qquad l = 1, \dots, \mathtt{s},$$

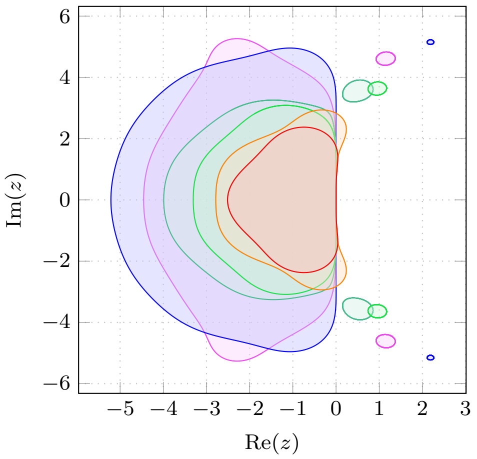
where $y^{n,l}$ is a stage approximation at time $t^{n,l} := t^n + c_l \Delta t$. The update is given by

$$y^{n+1} := y^n + \sum_{k=1}^{\mathtt{r}} \Delta t^k \sum_{l=1}^{\mathtt{s}} b_l^{(k)} \frac{\mathrm{d}^{k-1}}{\mathrm{d}t^{k-1}}\Phi(y^{n,l}).$$

The given coefficients $a_{l\nu}^{(k)}$ and $b_l^{(k)}$ determine the scheme; they are typically summarized in an extended Butcher tableau.

Remark 2 Note that standard Taylor methods can be cast in the framework of Def. 1 through setting $\mathtt{s} = 1$, $a_{11}^{(k)} = 0$ and $b_1^{(k)} = 1/k!$ $(k = 1, \dots, \mathtt{r})$. The multiderivative Runge-Kutta schemes used in this work can be found through their extended Butcher tableaux in Appendix A.

Remark 3 The stability regions of the used Runge-Kutta methods are visualized in Fig. 1, see [5, 14, 21] for more details. Note that except for 3DRK5-2 and 4DRK6-2, all schemes contain parts of the imaginary axis.



Scheme	$R(z)$
— 2DRK3-2	$\sum_{k=0}^{3} \frac{z^k}{k!}$
— 2DRK4-2	$\sum_{k=0}^{4} \frac{z^k}{k!}$
— 2DRK5-3	$\sum_{k=0}^{5} \frac{z^k}{k!} + \frac{z^6}{600}$
— 3DRK5-2	$\sum_{k=0}^{5} \frac{z^k}{k!} + \frac{z^6}{900}$
— 3DRK7-3	$\sum_{k=0}^{7} \frac{z^k}{k!} + c_8 z^8 + c_9 z^9$
— 4DRK6-2	$\sum_{k=0}^{6} \frac{z^k}{k!} + \frac{z^7}{6480} + \frac{z^8}{77760}$

Fig. 1: Regions of absolute stability $\mathcal{R} := \{z \in \mathbb{C} \mid |R(z)| \leq 1\}$ and stability functions $R(z)$ for the two-derivative schemes 2DRK3-2, 2DRK4-2, 2DRK5-3 [5], the three-derivative schemes 3DRK5-2 and 3DRK7-3 [21] and the four-derivative scheme 4DRK6-2 (see Appendix A). Except for 3DRK5-2 and 4DRK6-2, all schemes contain parts of the imaginary axis. Note that in the stability function of 3DRK7-3, we have defined $c_8 := \frac{1}{23520} - \frac{\sqrt{2}}{70560}$ and $c_9 := \frac{11}{1481760} - \frac{\sqrt{2}}{246960}$.

3 Multiderivative Runge-Kutta solvers for hyperbolic conservation laws

Discretizing the spatial part of the hyperbolic conservation law (1) necessitates a discretization of the domain Ω. Hence, consider

$$\{x_1, \dots, x_M\}$$

to be a uniform partition of Ω into M cells of size Δx. A natural extension of Def. 1 applied to Eq. (1) can then be expressed as

$$w_i^{n,l} := w_i^n - \sum_{k=1}^{\mathtt{r}} \Delta t^k \sum_{\nu=1}^{l-1} a_{l\nu}^{(k)} D_x D_t^{k-1} f(w_i^{n,\nu}) \,, \tag{6a}$$

$$w_i^{n+1} := w_i^n - \sum_{k=1}^{\mathtt{r}} \Delta t^k \sum_{l=1}^{\mathtt{s}} b_l^{(k)} D_x D_t^{k-1} f(w_i^{n,l}), \tag{6b}$$

for $l = 1, \ldots, \mathtt{s}$; with D_x and D_t being suitable approximations to ∂_x and ∂_t to be explained in the sequel. Contrary to the complete Cauchy-Kovalevskaya procedure as outlined for the second derivative in (4), only *one* time derivative of the solution is transformed into a spatial derivative (see (3)).

The core focus of this paper is to avoid the explicit use of Jacobians of the flux function f. Jacobians of f arise due to the usage of higher temporal derivatives, see, e.g., Eq. (4). In this, we follow the compact approximate Taylor (CAT) approach outlined in [3]. Since the CAT method heavily relies on discrete differentiation, first a small part is devoted to introducing the fundamental notation. Thereafter the method is described and applied to Eqs. (6).

3.1 Discrete differentiation

In this short section, we fix the notation on using finite differencing [1, 25, 26]. Considering central differences, the $(2p+1)$-point Lagrangian polynomials are given by

$$L_{p,j}(\omega) := \prod_{\substack{r=-p \\ r \neq j}}^{p} \frac{\omega - r}{j - r} \qquad j = -p, \ldots, p\,. \tag{7}$$

It is well-known that these polynomials can be used to interpolate $\varphi : \mathbb{R} \to \mathbb{R}$ in the points $x_{i-p}, \ldots, x_{i+p}$ through

$$\mathcal{P}_i \varphi(x) := \sum_{j=-p}^{p} L_{p,j}\left(\frac{x - x_i}{\Delta x}\right) \varphi(x_{i+j}). \tag{8}$$

Similarly, from the $2p$-point Lagrangian polynomials

$$\ell_{p,j}(\omega) := \prod_{\substack{r=-p+1 \\ r \neq j}}^{p} \frac{\omega - r}{j - r} \qquad j = -p+1, \ldots, p\,, \tag{9}$$

(note that the index of the product begins at $r = -p+1$) we obtain the unique polynomial of degree $2p-1$ interpolating φ in the points $x_{i-p+1}, \ldots, x_{i+p}$ through

$$\mathcal{Q}_i \varphi(x) := \sum_{j=-p+1}^{p} \ell_{p,j}\left(\frac{x - x_i}{\Delta x}\right) \varphi(x_{i+j}). \tag{10}$$

In the sequel, we use a similar notation as in [3]:

Definition 2 ([3]) For the k-th derivative $0 \leq k \leq 2p$, we define the following quantities:

$$\begin{aligned} \delta_{p,j}^{k} &:= L_{p,j}^{(k)}(0), & j &= -p, \ldots, p, \\ \gamma_{p,j}^{k,m} &:= \ell_{p,j}^{(k)}(m), & j &= -p+1, \ldots, p, \quad m = -p+1, \ldots, p. \end{aligned}$$

Note that the k-th derivatives $L_{p,j}^{(k)}(0)$ and $\ell_{p,j}^{(k)}(m)$ are derived analytically from Eq. (7) and Eq. (9), respectively.

Approximate derivatives can thus be derived from

$$(\mathcal{P}_i\varphi)^{(k)}(x_i) = \frac{1}{\Delta x^k} \sum_{j=-p}^{p} \delta^k_{p,j}\varphi(x_{i+j})\,. \tag{11a}$$

$$(\mathcal{Q}_i\varphi)^{(k)}(x_{i+m}) = \frac{1}{\Delta x^k} \sum_{j=-p+1}^{p} \gamma^{k,m}_{p,j}\varphi(x_{i+j})\,, \tag{11b}$$

with $m = -p+1, \dots, p$. Since we are working in a discrete context, we define the linear operator counterparts of Eq. (11a) and Eq. (11b) as

$$\begin{aligned} P^{(k)} &: \mathbb{R}^{2p+1} \to \mathbb{R}, \qquad \mathbf{v} \mapsto \frac{1}{\Delta x^k} \sum_{j=-p}^{p} \delta^k_{p,j} v_j\,, \\ Q^{(k)}_m &: \mathbb{R}^{2p} \to \mathbb{R}, \qquad \mathbf{w} \mapsto \frac{1}{\Delta x^k} \sum_{j=-p+1}^{p} \gamma^{k,m}_{p,j} w_j\,. \end{aligned}$$

Remark 4 The spatial index i is neglected for the linear operators as its direct dependency on the node x_i is lost, cf. Eq. (8) and Eq. (10). Notice also that $Q^{(k)}_m$ takes vectors in $\mathbb{R}^{2p}$ as input, whereas $P^{(k)}$ takes vectors in $\mathbb{R}^{2p+1}$ as input.

A non-centered $2p$-point finite difference method to approximate $\partial^k_t w(x_i, t^{n+m})$ for $m = -p+1, \dots, p$ can therefore be written as

$$Q^{(k)}_m \mathbf{w}^{\langle n \rangle}_i = \frac{1}{\Delta t^k} \sum_{r=-p+1}^{p} \gamma^{k,m}_{p,j} w^{n+r}_i\,,$$

with vector notation

$$\mathbf{w}^{\langle n \rangle}_i := \begin{pmatrix} w^{n-p+1}_i \\ \vdots \\ w^{n+p}_i \end{pmatrix}\,. \tag{12}$$

The angled brackets represent the local stencil function

$$\langle \cdot \rangle : \mathbb{Z} \to \mathbb{Z}^{2p} : n \mapsto \big(n-p+1,\, \dots,\, n+p\big)^T \tag{13}$$

throughout this paper, and will be considered for both the spatial index i as the temporal index n. Note that the position of the angled bracket (top or bottom) determines whether derivation is w.r.t. time (top) or space (bottom).

To put the scheme into conservation form, in [36] auxiliary centered coefficients have been introduced. Here, the operators $P^{(k)}$ for $k \geq 1$ are written as differences of new 'half-way point' interpolation operators.

Definition 3 ([36]) Define $\lambda^{k-1}_{p,j}$ via the relations

$$\delta^k_{p,p} =: \lambda^{k-1}_{p,p}\,, \tag{14a}$$

$$\delta^k_{p,j} =: \lambda^{k-1}_{p,j} - \lambda^{k-1}_{p,j+1}\,, \quad j = -p+1, \dots, p-1\,, \tag{14b}$$

$$\delta^k_{p,-p} =: -\lambda^{k-1}_{p,-p+1}\,. \tag{14c}$$

Remark 5 The relations given in Eq. (14) make up an overdetermined system, yet provide a unique solution $\lambda^{k-1}_{p,j}$ obtained from Eq. (14a) and Eq. (14b), see [36, Theorem 2].

Notice the shift between k and $k-1$. This is justified because we enforce a first order derivative relation for the approximation $(\mathcal{P}_i\varphi)^{(k)}(x_i)$ by splitting the operator as

$$(\mathcal{P}_i\varphi)^{(k)}(x_i) = \frac{\left(\Lambda^{(k-1)}\varphi\right)(x_{i+1/2}) - \left(\Lambda^{(k-1)}\varphi\right)(x_{i-1/2})}{\mathit{\Delta} x}\,, \tag{15}$$

in which $\Lambda^{(k-1)}$ is an operator mapping to $\mathbb{P}_{2p-1}$ so that

$$\left(\Lambda^{(k-1)}\varphi\right)(x_{i+1/2}) := \frac{1}{\mathit{\Delta} x^{k-1}} \sum_{j=-p+1}^{p} \lambda^{k-1}_{p,j}\,\varphi(x_{i+j})\,.$$

The linear operator alternative is defined by

$$\mathit{\Lambda}^{(k-1)} \colon \mathbb{R}^{2p} \longrightarrow \mathbb{R}, \qquad \mathbf{v} \mapsto \frac{1}{\mathit{\Delta} x^{k-1}} \sum_{j=-p+1}^{p} \lambda^{k-1}_{p,j} v_j\,. \tag{16}$$

An overview of all the defined interpolation operators is given in Tbl. 1.

Table 1: A summary of the defined interpolation operators scaled and shifted to fit the uniform mesh locally at x_i.

Functional	Linear	
$(\mathcal{P}_i\varphi)^{(k)}$	$P^{(k)}\mathbf{v}$	k-th derivative of the Lagrangian interpolation polynomial in the nodes $x_{i-p},\ldots,x_{i+p}$
$(\mathcal{Q}_i\varphi)^{(k)}$	$Q^{(k)}_m\mathbf{v}$	k-th derivative of the Lagrangian interpolation polynomial in the nodes $x_{i-p+1},\ldots,x_{i+p}$
$\left(\Lambda^{(k-1)}\varphi\right)(x_{i+1/2})$	$\mathit{\Lambda}^{(k-1)}\mathbf{v}$	$(k-1)$-th derivative of a half-way interpolation at $x_{i+1/2}$ using the nodes $x_{i-p+1},\ldots,x_{i+p}$ with the difference coefficients defined by Eq. (14)

3.2 A Jacobian-free MDRK scheme

With all the building blocks at our disposal, we can now describe how the final class of methods, that we call MDRKCAT, is assembled. Starting from Eq. (6), we define the conservative updates of the solution via

$$w_i^{n,l} := w_i^n - \frac{\mathit{\Delta} t}{\mathit{\Delta} x}\left(F^{n,l}_{i+1/2} - F^{n,l}_{i-1/2}\right) \qquad l = 1,\ldots,\mathtt{s}\,, \tag{17a}$$

$$w_i^{n+1} := w_i^n - \frac{\mathit{\Delta} t}{\mathit{\Delta} x}\left(F^{n}_{i+1/2} - F^{n}_{i-1/2}\right), \tag{17b}$$

in which the numerical fluxes are given by,

$$F^{n,l}_{i+1/2} = \sum_{k=1}^{\mathtt{r}} \mathit{\Delta} t^{k-1} \sum_{\nu=1}^{l-1} a^{(k)}_{l\nu}\, \mathit{\Lambda}^{(0)} (\widetilde{\mathbf{f}}^{\nu})^{(k-1)}_{i,\langle 0\rangle} \qquad l = 1,\ldots,\mathtt{s}\,, \tag{18a}$$

$$F^n_{i+1/2} = \sum_{k=1}^{\mathtt{r}} \Delta t^{k-1} \sum_{l=1}^{\mathtt{s}} b^{(k)}_l \Lambda^{(0)} (\widetilde{\mathbf{f}}^{l})^{(k-1)}_{i,\langle 0\rangle} \,. \tag{18b}$$

For the calculation of $\widetilde{\mathbf{f}}^{(k-1)}_{i,\langle 0\rangle}$ the compact approximate Taylor (CAT) procedure [3] is used and the flux derivatives can be calculated according to Eq. (16) by

$$\Lambda^{(0)}\widetilde{\mathbf{f}}^{(k-1)}_{i,\langle 0\rangle} := \sum_{j=-p+1}^{p} \lambda^0_{p,j} \widetilde{f}^{(k-1)}_{i,j} \,.$$

$\widetilde{f}^{(k-1)}_{i,j} \approx \partial_t^{k-1} f(w)(x_{i+j}, t^n)$ indicates the local approximations for the time-derivatives of the flux and are given by

$$\widetilde{f}^{(k-1)}_{i,j} := Q^{(k-1)}_0 \left(\mathfrak{f}_{\boldsymbol{T}}\right)^{k-1,\langle n\rangle}_{i,j} \qquad j = -p+1, \ldots, p \,.$$

They rely on the approximate flux values $(\mathfrak{f}_T)^{k-1,n+r}_{i,j} \approx f(w(x_{i+j}, t^{n+r}))$. In other words, we take the $(k-1)$-st discrete temporal derivative in x_{i+j} using approximate fluxes

$$(\mathfrak{f}_T)^{k-1,n+r}_{i,j} := f\left(w^n_{i+j} + \sum_{m=1}^{k-1} \frac{(r\Delta t)^m}{m!} \widetilde{w}^{(m)}_{i,j}\right),$$

for $j, r = -p+1, \ldots, p$. The only thing that is left to define are the quantities $\widetilde{w}^{(m)}_{i,j} \approx \frac{\partial^m}{\partial t^m} w(x_{i+j}, t^n)$. Their approximation makes heavy use of the Cauchy-Kovalevskaya identity $\partial_t^m w = -\partial_x \partial_t^{m-1} f(w)$, they are hence approximated by

$$\widetilde{w}^{(m)}_{i,j} := -Q^{(1)}_j \widetilde{\mathbf{f}}^{(m-1)}_{i,\langle 0\rangle} \qquad j = -p+1, \ldots, p.$$

Via the described steps, the vectors $\widetilde{\mathbf{f}}^{(k-1)}_{i,\langle 0\rangle}$ are recursively obtained, see also [3].

Definition 4 For a more precise terminology, we define the specific $\mathtt{r}$-derivative, q-th order, $\mathtt{s}$-stage MDRKCAT method as $\mathtt{r}$DRKCATq-$\mathtt{s}$.

A summary of the $\mathtt{r}$DRKCATq-$\mathtt{s}$ procedure to obtain the stage values is given in Alg. 1. Note that the flux at the left half-way point is obtained either from a shift of the index, i.e. $F^n_{i-1/2} = F^n_{i-1+1/2}$ or is given by the boundary condition.

Algorithm 1 Stages of $\mathtt{r}$DRKCATq-$\mathtt{s}$, an $\mathtt{r}$-derivative, q-th order, $\mathtt{s}$-stage MDRKCAT method

```
Stage solution ($l = 2, \dots, \mathtt{s}$):
for $j = -p+1$ to $p$ do
    $(\widetilde{f}^{l-1})^{(0)}_{i,j} = f(w^{n,l-1}_{i+j})$
end
$F^{n,l}_{i+1/2} = \sum_{\nu=1}^{l-1} a^{(1)}_{l\nu} \Lambda^{(0)} (\widetilde{\mathbf{f}}^{\nu})^{(0)}_{i,\langle 0 \rangle}$
for $k = 2$ to $\mathtt{r}$ do
    Get $(\widetilde{f}^{l-1})^{(k-1)}_{i,j}$ via CAT procedure.
    $F^{n,l}_{i+1/2} \mathrel{+}= \Delta t^{k-1} \sum_{\nu=1}^{l-1} a^{(k)}_{l\nu} \Lambda^{(0)} (\widetilde{\mathbf{f}}^{\nu})^{(k-1)}_{i,\langle 0 \rangle}$
end
$w^{n,l}_i = w^n_i - \frac{\Delta t}{\Delta x} (F^{n,l}_{i+1/2} - F^{n,l}_{i-1/2})$

CAT procedure [3] ($k = 2, \dots, \mathtt{r}$):
for $j = -p+1$ to $p$ do
    $\widetilde{w}^{(k-1)}_{i,j} = -Q^{(1)}_j \widetilde{\mathbf{f}}^{(k-2)}_{i,\langle 0 \rangle}$
        $= -\frac{1}{\Delta x} \sum_{r=-p+1}^{p} \gamma^{1,j}_{p,r} \widetilde{f}^{(k-2)}_{i,r}$
    for $r = -p+1$ to $p$ do
        $(\mathfrak{f}_T)^{k-1,n+r}_{i,j} =$
            $f\left( w^n_{i+j} + \sum_{m=1}^{k-1} \frac{(r\Delta t)^m}{m!} \widetilde{w}^{(m)}_{i,j} \right)$
    end
    $\widetilde{f}^{(k-1)}_{i,j} = Q^{(k-1)}_0 (\mathfrak{f}_{\boldsymbol{T}})^{k-1,\langle n \rangle}_{i,j}$
        $= \frac{1}{\Delta t^{k-1}} \sum_{r=-p+1}^{p} \gamma^{k-1,0}_{p,r} (\mathfrak{f}_T)^{k-1,n+r}_{i,j}$
end
```

4 Consistency analysis

In this section, we show that the $\mathtt{r}$DRKCATq-$\mathtt{s}$ methods are consistent. The order of consistency is, as to be expected, the minimum of the underlying Runge-Kutta order (q) and the order of the interpolation ($2p$). Let us make the following two important assumptions:

Assumption 1 *We assume both f and w to be smooth functions in C^∞. Furthermore, we assume that Δt and Δx are asymptotically comparable in size, i.e.,*

$$\mathcal{O}(\Delta t) = \mathcal{O}(\Delta x).$$

Throughout this section, we use the following notation to reduce the number of function arguments:

$$\partial_t^k f(w)_{i+j} \equiv \partial_t^k f(w(x_{i+j}, t^n)).$$

Whenever possible, similar notation is used for other functions. The time index n is only mentioned when necessary. We immediately state the main result and thereafter, in a successive form, deduce the necessary lemmas upon which its proof relies.

Theorem 1 *The consistency order of an explicit $\mathtt{r}$DRKCATq-$\mathtt{s}$ method is given by $\min(2p, q)$. Here, q is the consistency order of the underlying MDRK method, while the stencil to update $w(x_i, t^n)$ is given by $\{i-p, i-p+1, \dots, i+p\}$.*

Proof The proof relies on Lemmas that will be proven in the sequel. In La. 1, it is shown that the numerical flux difference gives the correct flux up to an order of $2p$. We can hence substitute the exact solution $w(x,t)$ into Eq. (17b), which immediately gives the requested result due to the fact that the Runge-Kutta update is an integration scheme of order $q+1$:

$$w(x_i, t^{n+1}) - w(x_i, t^n) + \frac{\Delta t}{\Delta x} \left(F^n_{i+1/2} - F^n_{i-1/2} \right)$$

$$
\begin{aligned}
&= w(x_i, t^{n+1}) - w(x_i, t^n) + \sum_{k=1}^{\mathtt{r}} \Delta t^k \sum_{l=1}^{\mathtt{s}} b_l^{(k)} \partial_x \partial_t^{k-1} f(w)_i^{n,l} + \mathcal{O}(\Delta x^{2p+1}) \\
&= \mathcal{O}(\Delta t^{q+1}) + \mathcal{O}(\Delta x^{2p+1}) \,.
\end{aligned}
$$

□

Remark 6 Since the convergence order is $\min(2p, q)$, the optimal choice w.r.t. computational efficiency is to set $p = \lceil q/2 \rceil$. Hence, "$\mathtt{r}$DRKCATq-$\mathtt{s}$" does not contain the variable p.

Lemma 1 *The update numerical flux* (18b) *satisfies*

$$
\frac{F_{i+1/2}^n - F_{i-1/2}^n}{\Delta x} = \sum_{k=1}^{\mathtt{r}} \Delta t^{k-1} \sum_{l=1}^{\mathtt{s}} b_l^{(k)} \partial_x \partial_t^{k-1} f(w)_i^{n,l} + \mathcal{O}(\Delta x^{2p}) \,.
$$

An analogous result holds for the stage flux (18a).

Proof From La. 3 and La. 4, we obtain that for $k > 1$ and $l = 1, \ldots, \mathtt{s}$ there holds

$$
\begin{aligned}
(\widetilde{f}^l)_{i,j}^{(k-1)} = \partial_t^{k-1} f(w)_{i+j}^{n,l} &+ \eta_p^{(k-1)} \big(R_{f,t}^{(k-1)}\big)_{i+j}^{n,l} \cdot \Delta t^{2p-k+1} \\
&+ \xi_{p,j}^{(k-1)} \big(R_{f,x}^{(k-1)}\big)_i^{n,l} \cdot \Delta x^{2p-k+1} + \mathcal{O}(\Delta x^{2p-k+2}) \,,
\end{aligned}
$$

with $\eta_p^{(k-1)}$ and $\xi_{p,j}^{(k-1)}$ real-valued coefficients; and $R_{f,x}^{(k-1)}$, $R_{f,t}^{(k-1)}$ smooth functions of space and time. The above formula is put into use by substituting it into the numerical flux (18b):

$$
\begin{aligned}
&\frac{\Lambda^{(0)} (\widetilde{\mathbf{f}}^l)_{i,\langle 0 \rangle}^{(k-1)} - \Lambda^{(0)} (\widetilde{\mathbf{f}}^l)_{i-1,\langle 0 \rangle}^{(k-1)}}{\Delta x} \\
&\quad = \frac{\Big(\Lambda^{(0)} \partial_t^{k-1} f(w)\Big)_{i+1/2}^{n,l} - \Big(\Lambda^{(0)} \partial_t^{k-1} f(w)\Big)_{i-1/2}^{n,l}}{\Delta x} \\
&\qquad + \Delta t^{2p-k+1} \cdot \eta_p^{(k-1)} \frac{\big(\Lambda^{(0)} R_{f,t}^{(k-1)}\big)_{i+1/2}^{n,l} - \big(\Lambda^{(0)} R_{f,t}^{(k-1)}\big)_{i-1/2}^{n,l}}{\Delta x} \\
&\qquad + \Delta x^{2p-k+1} \cdot \frac{\big(R_{f,x}^{(k-1)}\big)_i^{n,l} - \big(R_{f,x}^{(k-1)}\big)_{i-1}^{n,l}}{\Delta x} \underbrace{\sum_{j=-p+1}^{p} \lambda_{p,j}^0 \xi_{p,j}^{(k-1)}}_{\mathcal{O}(1)} \\
&\qquad + \frac{1}{\Delta x} \mathcal{O}(\Delta x^{2p-k+2}) \\
&\quad \overset{(15)}{=} \Big(\mathcal{P}_i \partial_t^{k-1} f(w)\Big)^{(1)} (x_i, t^{n,l}) \\
&\qquad + \Delta t^{2p-k+1} \cdot \eta_p^{(k-1)} \big(\mathcal{P}_i R_{f,t}^{(k-1)}\big)^{(1)} (x_i, t^{n,l}) + \mathcal{O}(\Delta x^{2p-k+1}) \\
&\quad = \partial_x \partial_t^{k-1} f(w)_i^{n,l} + \mathcal{O}(\Delta x^{2p-k+1}) \,.
\end{aligned}
$$

Please note that the term $\Big(\big(R_{f,x}^{(k-1)}\big)_i^{n,l} - \big(R_{f,x}^{(k-1)}\big)_{i-1}^{n,l}\Big) / \Delta x$ can be interpreted as a finite difference approximation of the derivative of the smooth function $R_{f,x}^{k-1}$ and therefore remains bounded, i.e., is $\mathcal{O}(1)$.

□

Before we regard the consistency analysis of the CAT steps in La. 3 and La. 4, the follwing identity on the difference coefficients is described.

Lemma 2 *Consider a local stencil index $j = -p+1, \ldots, p$. Then there holds for $k = 1, \ldots, 2p-1$:*

$$\sum_{r=-p+1}^{p} \gamma_{p,r}^{k,j} (r-j)^s = k!\delta_{s,k} \,, \quad s = 0, \ldots, 2p-1 \,.$$

The symbol $\delta_{s,k}$ here represents the Kronecker-delta function.

Proof We consider a mesh centered around $x_i = 0$ with spatial size $\Delta x = 1$. The operator $\mathcal{Q}_i$ exactly interpolates the polynomial function $\varphi(x) := (x-j)^s$ for $s = 0, \ldots, 2p-1$ such that Eq. (10) becomes

$$(x-j)^s = \mathcal{Q}_i \varphi(x) = \sum_{r=-p+1}^{p} \ell_{p,r}(x)\varphi(r) = \sum_{r=-p+1}^{p} \ell_{p,r}(x)(r-j)^s \,.$$

Deriving the above relation k times in x and thereafter evaluating in $x = j$ gives the result.

□

Now we can provide a consistency proof for the CAT procedure in Alg. 1. To this purpose we establish the following notation for the exact Taylor approximation in time of order $k \in \mathbb{N}$,

$$T_{i,j}^{k}(\tau) := \sum_{m=0}^{k} \frac{\tau^m}{m!} \partial_t^m w(x_{i+j}, t^n) \,. \tag{19}$$

It is the k-th order approximation of $w(x_{i+j}, t^n + \tau)$. The proof itself is built in a similar fashion as [3, Theorem 2] and [36, Proposition 1].

Lemma 3 *For $k = 1$ the steps of the CAT algorithm (Alg. 1, right side) satisfy*

$$\begin{aligned}
\widetilde{w}_{i,j}^{(1)} &= \partial_t w_{i+j} + \xi_{p,j}^{(1)} \big(R_w^{(1)}\big)_i \cdot \Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p}) \,, \\
\widetilde{f}_{i,j}^{(1)} &= \partial_t f(w)_{i+j} + \eta_p^{(1)} \big(R_{f,t}^{(1)}\big)_{i+j} \cdot \Delta t^{2p-1} + \xi_{p,j}^{(1)} \big(R_{f,x}^{(1)}\big)_i \cdot \Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p}) \,,
\end{aligned}$$

with real-valued coefficients $\xi_{p,j}^{(1)}$, $\eta_p^{(1)}$; and smooth functions $\big(R_w^{(1)}\big)_i$, $\big(R_{f,t}^{(1)}\big)_{i+j}$, $\big(R_{f,x}^{(1)}\big)_i$. (Please note again that $(\cdot)_i$ stands for function-evaluation at $x = x_i$.)

Proof A straightforward computation on $\widetilde{w}_{i,j}^{(1)}$, using the Taylor expansion of $f(w)$, reveals that

$$\begin{aligned}
\widetilde{w}_{i,j}^{(1)} &= -Q_j^{(1)} \widetilde{\mathbf{f}}_{i,\langle 0\rangle}^{(0)} = -\frac{1}{\Delta x} \sum_{r=-p+1}^{p} \gamma_{p,r}^{1,j} \widetilde{f}_{i,r}^{(0)} \\
&= -\frac{1}{\Delta x} \sum_{r=-p+1}^{p} \gamma_{p,r}^{1,j} f(w(x_{i+r}, t^n))
\end{aligned}$$

$$
\begin{aligned}
&= -\frac{1}{\Delta x}\sum_{r=-p+1}^{p}\gamma_{p,r}^{1,j}\left(\sum_{s=0}^{2p}\frac{((r-j)\Delta x)^s}{s!}\partial_x^s f(w)_{i+j} + \mathcal{O}(\Delta x^{2p+1})\right)\\
&= -\frac{1}{\Delta x}\sum_{s=0}^{2p-1}\frac{\Delta x^s}{s!}\partial_x^s f(w)_{i+j}\left(\sum_{r=-p+1}^{p}\gamma_{p,r}^{1,j}(r-j)^s\right)\\
&\quad -\left[\frac{1}{(2p)!}\left(\sum_{r=-p+1}^{p}\gamma_{p,r}^{1,j}(r-j)^{2p}\right)\partial_x^{2p} f(w)_i + \mathcal{O}(\Delta x)\right]\Delta x^{2p-1}\\
&\quad + \mathcal{O}(\Delta x^{2p})\\
&\overset{\text{La. 2}}{=} -\partial_x f(w)_{i+j} + \xi_{p,j}^{(1)}\big(R_w^{(1)}\big)_i\cdot\Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p})\\
&= \partial_t w_{i+j} + \xi_{p,j}^{(1)}\big(R_w^{(1)}\big)_i\cdot\Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p}),
\end{aligned}
$$

in which we define $\xi_{p,j}^{(1)}$ and $\big(R_w^{(1)}\big)_i$ by

$$
\xi_{p,j}^{(1)} := \frac{-1}{(2p)!}\sum_{r=-p+1}^{p}\gamma_{p,r}^{1,j}(r-j)^{2p}, \quad \big(R_w^{(1)}\big)_i := \partial_x^{2p} f(w)_i.
$$

The succeeding step of the method evaluates the flux f in an approximate Taylor series. We find,

$$
\begin{aligned}
(\mathfrak{f}_T)_{i,j}^{1,n+r} &= f\left(w(x_{i+j},t^n) + (r\Delta t)\widetilde{w}_{i,j}^{(1)}\right)\\
&\overset{(19)}{=} f\left(T_{i,j}^1(r\Delta t) + (r\Delta t)\xi_{p,j}^{(1)}\big(R_w^{(1)}\big)_i\cdot\Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p+1})\right)\\
&= (f\circ T_{i,j}^1)(r\Delta t)\\
&\qquad + (f'\circ T_{i,j}^1)(r\Delta t)\cdot\left[(r\Delta t)\xi_{p,j}^{(1)}\big(R_w^{(1)}\big)_i\right]\cdot\Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p+1})\\
&= (f\circ T_{i,j}^1)(r\Delta t)\\
&\qquad + (f'\circ T_{i,j}^1)(0)\cdot\left[(r\Delta t)\xi_{p,j}^{(1)}\big(R_w^{(1)}\big)_i\right]\cdot\Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p+1})\\
&= (f\circ T_{i,j}^1)(r\Delta t) + f'(w)_i\left[(r\Delta t)\xi_{p,j}^{(1)}\big(R_w^{(1)}\big)_i\right]\cdot\Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p+1}).
\end{aligned}
$$

Consequently, we can find for the temporal interpolation of the fluxes:

$$
\begin{aligned}
\widetilde{f}_{i,j}^{(1)} &= Q_0^{(1)}\,(\mathfrak{f}_{\boldsymbol{T}})_{i,j}^{1,\langle n\rangle} = \frac{1}{\Delta t}\sum_{r=-p+1}^{p}\gamma_{p,r}^{1,0}\,(\mathfrak{f}_T)_{i,j}^{1,n+r}\\
&= \frac{1}{\Delta t}\sum_{r=-p+1}^{p}\gamma_{p,r}^{1,0}\left[\sum_{s=0}^{2p}\left(\frac{(r\Delta t)^s}{s!}\frac{\mathrm{d}^s(f\circ T_{i,j}^1)}{\mathrm{d}\tau^s}(0)\right) + \mathcal{O}(\Delta x^{2p+1})\right]\\
&\quad + f'(w)_i\underbrace{\left(\sum_{r=-p+1}^{p}\gamma_{p,r}^{1,0}r\right)}_{=1,\text{ La. 2}}\xi_{p,j}^{(1)}\big(R_w^{(1)}\big)_i\cdot\Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p})
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{\Delta t} \sum_{s=0}^{2p-1} \frac{\Delta t^s}{s!} \underbrace{\left(\sum_{r=-p+1}^{p} \gamma_{p,r}^{1,0} r^s \right)}_{=\delta_{1,s},\ \text{La. } 2} \frac{\mathrm{d}^s (f \circ T_{i,j}^1)}{\mathrm{d}\tau^s}(0) \\
&\quad + \underbrace{\frac{1}{(2p)!} \left(\sum_{r=-p+1}^{p} \gamma_{p,r}^{1,0} r^{2p} \right)}_{=:\eta_p^{(1)}} \underbrace{\frac{\mathrm{d}^{2p} (f \circ T_{i,j}^1)}{\mathrm{d}\tau^{2p}}(0)}_{=:\left(R_{f,t}^{(1)}\right)_{i+j}} \cdot \Delta t^{2p-1} \\
&\quad + \xi_{p,j}^{(1)} \underbrace{f'(w)_i \big(R_w^{(1)}\big)_i}_{=:\left(R_{f,x}^{(1)}\right)_i} \cdot \Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p}) \\
&= \frac{\mathrm{d}(f \circ T_{i,j}^1)}{\mathrm{d}\tau}(0) + \eta_p^{(1)} \big(R_{f,t}^{(1)}\big)_{i+j} \cdot \Delta t^{2p-1} + \xi_{p,j}^{(1)} \big(R_{f,x}^{(1)}\big)_i \cdot \Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p}) \\
&= \partial_t f(w)_{i+j} + \eta_p^{(1)} \big(R_{f,t}^{(1)}\big)_{i+j} \cdot \Delta t^{2p-1} + \xi_{p,j}^{(1)} \big(R_{f,x}^{(1)}\big)_i \cdot \Delta x^{2p-1} + \mathcal{O}(\Delta x^{2p}).
\end{aligned}
$$

□

Via induction one can generalize this result.

Lemma 4 *For $k = 2, \ldots 2p-1$ the steps of the CAT algorithm (Alg. 1, right side) satisfy*

$$
\begin{aligned}
\widetilde{w}_{i,j}^{(k)} &= \partial_t^k w_{i+j} + \xi_{p,j}^{(k)} \big(R_w^{(k)}\big)_i \cdot \Delta x^{2p-k} + \mathcal{O}(\Delta x^{2p-k+1}), \\
\widetilde{f}_{i,j}^{(k)} &= \partial_t^k f(w)_{i+j} + \eta_p^{(k)} \big(R_{f,t}^{(k)}\big)_{i+j} \cdot \Delta t^{2p-k} \\
&\qquad + \xi_{p,j}^{(k)} \big(R_{f,x}^{(k)}\big)_i \cdot \Delta x^{2p-k} + \mathcal{O}(\Delta x^{2p-k+1}),
\end{aligned}
$$

with

$$
\eta_p^{(k)} := \frac{1}{(2p)!} \left(\sum_{r=-p+1}^{p} \gamma_{p,r}^{k,0} r^{2p} \right), \qquad \xi_{p,j}^{(k)} := -\sum_{r=-p+1}^{p} \gamma_{p,r}^{1,j} \xi_{p,r}^{(k-1)},
$$

$$
\begin{aligned}
&\big(R_w^{(k)}\big)_i := \big(f'(w)_i\big)^{k-1} \partial_x^{2p} f(w)_i, \quad \big(R_{f,t}^{(k)}\big)_{i+j} := \frac{\mathrm{d}^{2p}(f \circ T_{i,j}^k)}{\mathrm{d}\tau^{2p}}(0), \\
&\big(R_{f,x}^{(k)}\big)_i := f'(w)_i \big(R_w^{(k)}\big)_i .
\end{aligned}
$$

5 von Neumann stability of rDRKCATq-s methods

The original CAT procedure was developed with the intention to create a scheme which linearly reduces back to high-order Lax-Wendroff methods. As a consequence, the scheme is CFL-1 stable for linear equations [3, Theorem 1]. In this section, we discuss the stability properties of the rDRKCATq-s scheme presented in this work.

Theorem 2 (MDRK-LW scheme) *Explicit* $\mathtt{r}$*DRKCATq-*$\mathtt{s}$ *methods for the linear advection flux* $f(w) = \alpha w$ *reduce to the numerical scheme*

$$w_j^{n,l} = w_j^n + \sum_{k=1}^{\mathtt{r}} (-1)^k \alpha^k \Delta t^k \left(\sum_{\nu=1}^{l-1} a_{l\nu}^{(k)} P^{(k)} \mathbf{w}_{\langle j \rangle}^{n,\nu} \right) \qquad l = 1, \dots, \mathtt{s}\,, \tag{20a}$$

$$w_j^{n+1} = w_j^n + \sum_{k=1}^{\mathtt{r}} (-1)^k \alpha^k \Delta t^k \left(\sum_{l=1}^{\mathtt{s}} b_l^{(k)} P^{(k)} \mathbf{w}_{\langle j \rangle}^{n,l} \right)\,, \tag{20b}$$

with $P^{(k)} \mathbf{w}_{\langle j \rangle}^{n,l}$ *the centered difference approximation of the k-th spatial derivative.*

The proof is similar to [3, Theorem 1] and is hence left out. In this form it is possible to perform a von Neumann stability analysis (see for example [25, 32]). That is, we fill in the Fourier mode $w_j^n = \mathcal{W}(t^n) e^{\mathrm{i}\kappa x_j}$ with wave number $\kappa \in \mathbb{Z}$ and search for the the amplification factors via the relations

$$\mathcal{W}(t^{n,l}) := g^{n,l}(\kappa) \mathcal{W}(t^n) \quad \text{and} \quad \mathcal{W}(t^{n+1}) := g(\kappa) \mathcal{W}(t^n)\,.$$

Doing so gives an additional recurrence relation.

Proposition 1 *The amplification factors obtained from a von Neumann analysis on the MDRK-LW scheme Eq.* (20) *are defined by the recurrence relations*

$$g^{n,l}(\kappa) = 1 + \sum_{k=1}^{\mathtt{r}} (-1)^k \sigma^k\, \mathtt{P}^{(k)}(\kappa) \left(\sum_{\nu=1}^{l-1} a_{l\nu}^{(k)} g^{n,\nu}(\kappa) \right) \qquad l = 1, \dots, \mathtt{s}\,, \tag{21a}$$

$$g(\kappa) = 1 + \sum_{k=1}^{\mathtt{r}} (-1)^k \sigma^k\, \mathtt{P}^{(k)}(\kappa) \left(\sum_{l=1}^{\mathtt{s}} b_l^{(k)} g^{n,l}(\kappa) \right)\,, \tag{21b}$$

with wave number $\kappa \in \mathbb{Z}$, $\sigma := \frac{\alpha \Delta t}{\Delta x}$ *the corresponding CFL number and*

$$\mathtt{P}^{(k)}(\kappa) := \sum_{r=-p}^{p} \delta_{p,r}^k e^{\mathrm{i}r\kappa\Delta x}\,. \tag{22}$$

The term $\mathtt{P}^{(k)}(\kappa)$ can be interpreted as the k-th derivative of the centered $(2p+1)$-point Lagrangian interpolation Eq. (11a) using Fourier basis $\left\{e^{\mathrm{i}l\chi} \mid l \in \mathbb{Z}\right\}$ with the grid frequency $\chi = \kappa \Delta x$.

In Tbl. 2 we display the amplification factors $g(\kappa)$ for the considered MDRK schemes summarized earlier in Fig. 1. Of main interest w.r.t. to these amplification factors is to obtain a critical CFL value $\sigma^* \in \mathbb{R}^+$ such that

$$|g(\kappa)| \leq 1 \quad \Leftrightarrow \quad \frac{\alpha \Delta t}{\Delta x} \leq \sigma^*\,.$$

We have used the Symbolic Math Toolbox in `MATLAB` [20] along with a bisection method on the CFL variable σ in Eq. (21b) to numerically obtain a σ^*:

- In our approach we take $\Delta x = 1$; only the frequency of $g(\kappa)$ is influenced by Δx, there is no change in absolute value. Assume we would compare mesh sizes Δx and $\Delta \tilde{x}$, then

$$\mathtt{P}^{(k)}(\kappa) = \sum_{r=-p}^{p} \delta_{p,r}^{k} e^{\mathrm{i} r \kappa \Delta x} = \sum_{r=-p}^{p} \delta_{p,r}^{k} e^{\mathrm{i} r \left(\kappa \frac{\Delta x}{\Delta \tilde{x}}\right) \Delta \tilde{x}} = \mathtt{P}^{(k)}(\widetilde{\kappa}) \,,$$

  where $\widetilde{\kappa} := \kappa \frac{\Delta x}{\Delta \tilde{x}}$. Thus behavior of $g(\kappa)$ is the same up to a recalibration of the frequency space.
- Via the toolbox, the value $\max |g(\kappa)|$ is calculated on a uniform 1000-cell mesh of $\kappa \in [-\pi, \pi]$. This domain suffices for this purpose, since for $\Delta x = 1$ all terms $\mathtt{P}^{(k)}(\kappa)$ in Eq. (22) are 2π-periodic.

Table 2: Amplification factors $g(\kappa)$ of the considered MDRK schemes (see Appendix A) and the corresponding CFL values σ^* up to four decimals obtained from $|g(\kappa)| \leq 1$.

Scheme	$g(\kappa)$	CFL
2DRKCAT3-2	$1 - \sigma\ \mathtt{P}^{(1)} + \frac{1}{2}\sigma^2 \left(\frac{2}{3}\mathtt{P}^{(1)}\mathtt{P}^{(1)} + \frac{1}{3}\mathtt{P}^{(2)}\right) - \frac{1}{6}\sigma^3\ \mathtt{P}^{(1)}\mathtt{P}^{(2)}$	1.2954
2DRKCAT4-2	$1 - \sigma\ \mathtt{P}^{(1)} + \frac{1}{2}\sigma^2\ \mathtt{P}^{(2)} - \frac{1}{6}\sigma^3\ \mathtt{P}^{(1)}\mathtt{P}^{(2)} + \frac{1}{24}\sigma^4\ \mathtt{P}^{(2)}\mathtt{P}^{(2)}$	1.4718
2DRKCAT5-3	$1 - \sigma\ \mathtt{P}^{(1)} + \frac{1}{2}\sigma^2\ \mathtt{P}^{(2)} - \frac{1}{6}\sigma^3\ \mathtt{P}^{(1)}\mathtt{P}^{(2)} + \frac{1}{24}\sigma^4\ \mathtt{P}^{(2)}\mathtt{P}^{(2)}$ $-\frac{1}{120}\sigma^5\ \mathtt{P}^{(1)}\mathtt{P}^{(2)}\mathtt{P}^{(2)} + \frac{1}{600}\sigma^6\ \mathtt{P}^{(2)}\mathtt{P}^{(2)}\mathtt{P}^{(2)}$	1.0619
3DRKCAT5-2	$1 - \sigma\ \mathtt{P}^{(1)} + \frac{1}{2}\sigma^2\ \mathtt{P}^{(2)} - \frac{1}{6}\sigma^3\ \mathtt{P}^{(3)} + \frac{1}{24}\sigma^4\ \mathtt{P}^{(1)}\mathtt{P}^{(3)}$ $-\frac{1}{120}\sigma^5\ \mathtt{P}^{(2)}\mathtt{P}^{(3)} + \frac{1}{900}\sigma^6\ \mathtt{P}^{(3)}\mathtt{P}^{(3)}$	0.4275
3DRKCAT7-3	$1 - \sigma\ \mathtt{P}^{(1)} + \frac{1}{2}\sigma^2\ \mathtt{P}^{(2)} - \frac{1}{6}\sigma^3\ \mathtt{P}^{(3)} + \frac{1}{24}\sigma^4\ \mathtt{P}^{(1)}\mathtt{P}^{(3)}$ $-\frac{1}{120}\sigma^5\ \mathtt{P}^{(2)}\mathtt{P}^{(3)} + \frac{1}{720}\sigma^6\ \mathtt{P}^{(3)}\mathtt{P}^{(3)} - \frac{1}{5040}\sigma^7\ \mathtt{P}^{(1)}\mathtt{P}^{(2)}\mathtt{P}^{(3)}$ $+\left(\frac{1}{23520} - \frac{\sqrt{2}}{70560}\right)\sigma^8\ \mathtt{P}^{(2)}\mathtt{P}^{(3)}\mathtt{P}^{(3)} - \left(\frac{11}{1481760} - \frac{\sqrt{2}}{246960}\right)\sigma^9\ \mathtt{P}^{(3)}\mathtt{P}^{(3)}\mathtt{P}^{(3)}$	0.2300
4DRKCAT6-2	$1 - \sigma\ \mathtt{P}^{(1)} + \frac{1}{2}\sigma^2\ \mathtt{P}^{(2)} - \frac{1}{6}\sigma^3\ \mathtt{P}^{(3)} + \frac{1}{24}\sigma^4\ \mathtt{P}^{(4)}$ $-\frac{1}{120}\sigma^5\ \mathtt{P}^{(1)}\mathtt{P}^{(4)} + \frac{1}{720}\sigma^6\ \mathtt{P}^{(2)}\mathtt{P}^{(4)} - \frac{1}{6480}\sigma^7\ \mathtt{P}^{(3)}\mathtt{P}^{(4)} + \frac{1}{77760}\sigma^8\ \mathtt{P}^{(4)}\mathtt{P}^{(4)}$	0.8563

The critical CFL values σ^* up to four decimals are shown in Tbl. 2. Notice that the two-derivative schemes improve the linear stability, whereas the other schemes reduce the stability compared to the original CAT method. To put this observation into perspective, let us point out that the CAT algorithm uses $2p$ derivatives, and thus is based on high-order Lax-Wendroff methods with $\mathtt{r}$ even. If, only for the sake of discussion, we take *uneven* $\mathtt{r}$, we get another picture. For $\mathtt{r} = 1$, we obtain the forward-time central-space scheme, which is infamous for being unconditionally unstable [17] and for odd $\mathtt{r} > 1$, we obtain CFL numbers smaller than one. Hence, we can make some important observations:

- Most importantly we can conclude that a method of lines (MOL) viewpoint is inadequate. Solely regarding the stability regions $\mathcal{R}$ in Fig. 1 would give the idea that the 3DRKCAT7-3 scheme provides the best stability. This is clearly not the case since the even-derivative schemes are shown to be better in terms of stability.

- Choosing even order derivatives gives better results than choosing an odd number of derivatives, i.e. the two- and four-derivative schemes show better stability than the three-derivative schemes. This is in very good agreement with the observations on the original CAT method.
- In contrast to Taylor-methods, where only the number of derivatives can be prescribed, MDRK methods have more free parameters, such as the number of derivatives *and* the number of stages. We observe that stages and derivatives highly influence the stability properties of the MDRKCAT method. This allows the identification of well-suited MDRK schemes and gives more flexibility compared to original Taylor-methods.

6 Numerical results

In this section, we show numerical results validating our analytical findings. By means of several continuous test cases ranging from scalar PDEs to the system of Euler equations, we show that the expected orders of convergence are obtained. For brevity, we do not include flux limiting techniques to this work; hence, we avoid setups where shock formation occurs. Note that flux limiting can be incorporated in a straightforward way as in [3].

The measure for the accuracy in this section is the scaled l_1-error at time $t^N \equiv T_{\text{end}}$

$$\|\mathbf{w}(T_{\text{end}}) - \mathbf{w}^N\|_1 := \Delta x \sum_{i=1}^{M} |w(x_i, T_{\text{end}}) - w_i^N| \,,$$

$\mathbf{w} : t \mapsto \mathbf{w}(t)$ being a function of time returning a vector of exact solution values (or a reference solution) in the nodes $x_1, \ldots, x_M$, and $\mathbf{w}^N$ being the vector of approximations w_i^N at t^N. For systems, the sum of the l_1-errors corresponding to the separate solution variables is considered. All displayed convergence plots begin at $M = 8$ cells and double the amount of cells with each iteration. In order to enforce the adopted CFL value σ, the local eigenvalues $\lambda^n_{\text{eig},i}$ of the Jacobian w.r.t. w_i^n are computed. By means of the relation

$$\Delta t^n := \frac{\sigma \Delta x}{\max_i |\lambda^n_{\text{eig},i}|}$$

the timestep is then computed. As the maximum eigenvalue in the computational domain varies over time, a non-constant timestep is prescribed. This highlights the ability of the novel method to use varying timestep sizes, see Rem. 1.

6.1 Burgers equation

First, we consider Burgers equation

$$\partial_t w + \partial_x \left(\frac{w^2}{2} \right) = 0,$$

with the cosine-wave initial condition with periodic boundary conditions

$$w(x, 0) = \frac{1}{4} \cos(\pi x) \qquad \text{on} \quad x \in [0, 2] \,, \tag{23}$$

to certify the accuracy $\min(2p, q)$ obtained in Thm. 1. Since the cosine-wave Eq. (23) has both positive and negative values, the characteristic lines must cross and a shock is formed at some point. The breaking time of the wave [17, 35] is at $t^* = \frac{4}{\pi}$. Hence we set the final time to $T_{\text{end}} = 0.8$, well before shock formation.

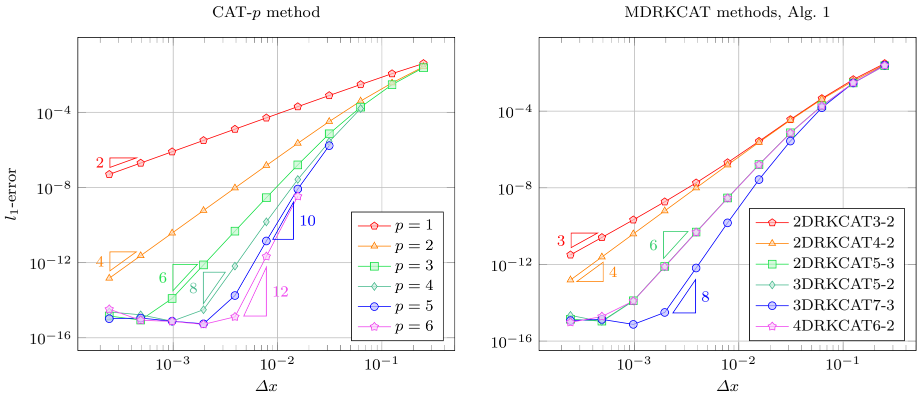


Fig. 2: Convergence order of the CAT and explicit MDRKCAT methods applied to Burgers equation on the cosine wave $w_0(x) = \frac{1}{4}\cos(\pi x)$ up to $T_{\text{end}} = 0.8$ with CFL $\sigma = 0.5$. The 2DRKCAT5-3, 3DRKCAT5-2 and 4DRKCAT6-2 schemes behave in a very alike manner. For large Δt and large p, we have observed stability problems for the CAT method. Those divergent results of the CAT method have been omitted in the convergence plot.

Using the characteristic lines solution, l_1-errors have been calculated for the CAT-p methods ($p = 1, \ldots, 6$) and the MDRKCAT methods (Appendix A). The results with CFL $\sigma = 0.5$ are visualized in Fig. 2. All expected convergence orders are obtained; order $2p$ for the CAT-p methods and at least order q of the MDRK schemes. The schemes 2DRKCAT5-3, 3DRKCAT5-2 and 3DRKCAT7-3 behave better than expected. This is caused by the fact that the spatial order of accuracy is higher than the temporal one; and spatial errors dominate the overall behavior at least for 'large' Δt. We have observed similar behavior also for other schemes where $2p > q$. Given that CAT methods have been designed as a natural generalization of Lax-Wendroff methods with an even-order accuracy, odd-order MDRK schemes take advantage here. We expect this behavior to become more apparent when computing with finer machine precision. Convergence plots such as in Fig. 2 will then manifest as a stretched out S-curve.

Further, we notice that for higher values of p and larger values Δt, the CAT methods tend to be less stable. Unstable results have been left out in Fig. 2 for $p = 4, 5$ and 6. Even though CAT methods have been shown to be linearly stable under a CFL-1 condition [3], rapid divergence is observed well before shocks are formed for larger values of p at regions where the derivative of the solution $w'(x)$ is large in absolute value. The MDRKCAT methods used in this work do not suffer this fate.

Next, we perform a similar study having the initial solution,

$$w(x,0) = \frac{1}{4} e^{\cos(\pi x)+\sin(\pi x)} \qquad \text{on} \quad x \in [0,2]\,,$$

with periodic boundary conditions. Having a steeper peak than the cosine (23), the breaking time will be earlier. We find $t^* = \frac{4}{\pi e}$, and choose $T_{\text{end}} = 0.3$ accordingly.

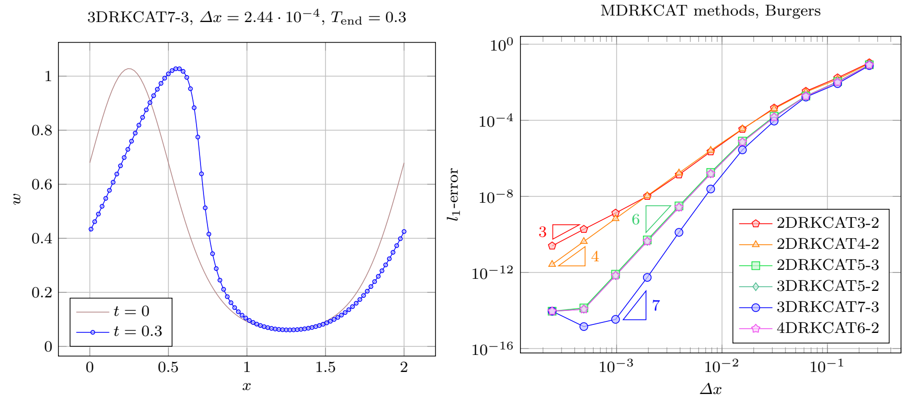


Fig. 3: Burgers equation applied to $w_0(x) = \frac{1}{4} e^{\cos(\pi x)+\sin(\pi x)}$ up to $T_{\text{end}} = 0.3$ with CFL $\sigma = 0.5$. Left: 3DRKCAT7-3 solution using $M = 8192$ cells, less nodal points are shown for better visual distinctness. Right: convergence order of the explicit MDRKCAT methods.

In Fig. 3 the final solution at $T_{\text{end}} = 0.3$ is visualized and accuracy is studied in the convergence plots, solely focused on the MDRKCAT method. The behavior is very similar to the previous case, except that the 3DRKCAT7-3 scheme is driven back faster to order 7.

6.2 Buckley-Leverett equation

Next, we consider the Buckley-Leverett flux [17],

$$f(w) = \frac{4w^2}{4w^2 + (1-w)^2}\,.$$

This flux is non-convex and introduces more nonlinearities compared to the Burgers flux. We consider the initial condition

$$w(x,0) = 1 - \frac{3}{4}\cos^2\left(\frac{\pi}{2}x\right) \qquad \text{on} \quad x \in [-1,1]\,.$$

The typical Buckley-Leverett profile consists of a shock wave followed directly by a rarefaction wave. We set $T_{\text{end}} = 0.1$ to remain continuous and be able to calculate the exact solution via its characteristics. The solution and convergence plots are visualized in Fig. 4 with CFL $\sigma = 0.5$. We notice that the numerical solution tends toward the expected Buckley-Leverett profile. All schemes converge with the expected accuracy in a similar way as for Burgers equation.

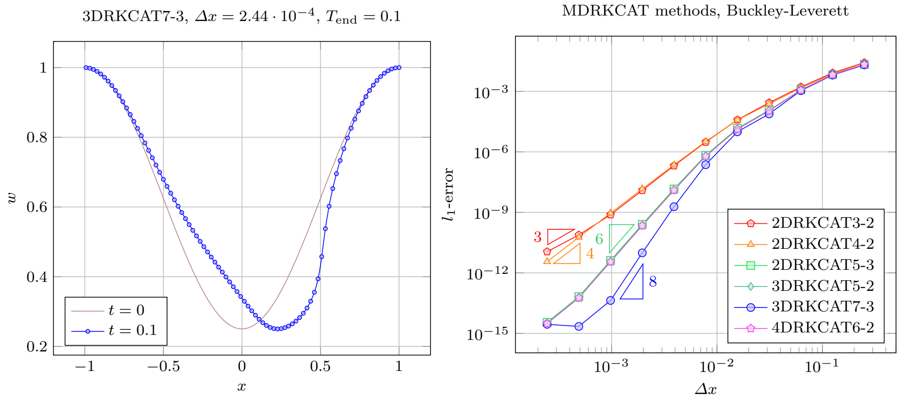


Fig. 4: Buckley-Leverett equation applied to $w_0(x) = 1 - \frac{3}{4}\cos^2\left(\frac{\pi}{2}x\right)$ up to $T_{\text{end}} = 0.1$ with CFL $\sigma = 0.5$. Left: 3DRKCAT7-3 solution using $M = 8192$ cells, less nodal points are shown for better visual distinctness. Right: convergence order of the explicit MDRKCAT methods.

6.3 One-dimensional Euler equations

Finally, we consider the Euler equations of gas dynamics

$$\partial_t w + \partial_x f(w) = 0\,,$$

in which

$$w = \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}\,, \quad f(w) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(E+p) \end{pmatrix}\,,$$

with ρ the density, u the velocity, E the energy and p the pressure. The system is closed via the equation of state for an ideal gas

$$p = (\gamma - 1)\left(E - \frac{1}{2}\rho u^2\right)\,,$$

with γ being the ratio of specific heats, assumed to be 1.4 [17, 35].

First we initialize the primitive variables (ρ, u, p) such that the Euler equations describe the linear advection of a density profile. To this end, we take

$$\rho(x, 0) = 1 + 0.3\sin(\pi x) \qquad \text{on} \quad x \in [0, 4]\,,$$

and set both $u(x, 0)$ and $p(x, 0)$ to be one. Periodic boundary conditions are used and T_{end} is set to 0.8. In Fig. 5 the corresponding convergence plots are displayed with CFL $\sigma = 0.5$. Immediately starting from the coarsest meshes the expected convergence orders are obtained.

Secondly, we consider the initial condition

$$w(x, 0) = \frac{1}{4}\begin{pmatrix} 3 \\ 1 \\ 3 \end{pmatrix} + \frac{\sin(\pi x)}{2}\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \qquad \text{on} \quad x \in [0, 2]\,,$$

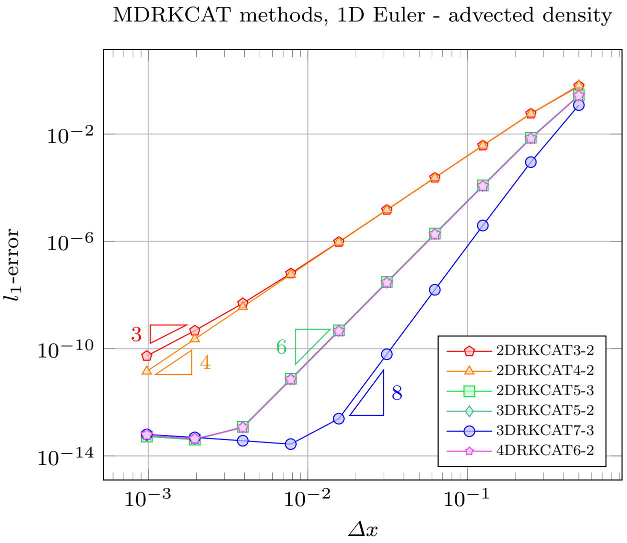


Fig. 5: Convergence order of the explicit MDRKCAT methods applied to $\rho_0(x) = 1 + 0.3\sin(\pi x)$, $u_0(x) = p_0(x) = 1$ on $x \in [0, 4]$ up to $T_{\text{end}} = 0.8$ with $\sigma = 0.5$.

with periodic boundary conditions and $T_{\text{end}} = 0.2$ for w to remain continuous. In Fig. 6 the solution is visualized for the 3DRKCAT7-3 scheme with $p = 4$, CFL $\sigma = 0.5$ on $M = 4096$ cells.

In order to inspect accuracy, a reference solution has been computed via a discontinuous Galerkin (DG) method. We have used third-order polynomials in space, and a third-order strong-stability-preserving Runge-Kutta method in time [12]. The reference computation was executed on 10240 cells with a CFL number of $\sigma = 0.15$.

Convergence plots have been generated in Fig. 7 using CFL $\sigma = 0.15$. All expected orders were obtained. For smaller Δx, the l_1-error converges towards approximately $2 \cdot 10^{-11}$, which is the accuracy of the reference DG solution.

Akin to the earlier cases, a CFL value of $\sigma = 0.5$ was attempted for the construction of the convergence plots. However not all simulations were stable, more specifically the 3DRKCAT7-3 scheme using $p = 4$ diverged for $M = 8, 16, 32$ and 64 cells.

In order to better grasp the efficiency of the MDRKCAT methods, in the same Fig. 7 convergence plots have been generated by means of a DG code that uses polynomial basis functions in $\mathbb{P}_3$ and $\mathbb{P}_4$ for each cell respectively. A fourth order SSP-RK scheme [31, order 4, p.21] has been used as time integrator. The same amount of cells $M = 8, \ldots, 1024$ has been used as for the MDRKCAT runs, the CFL $\sigma = 0.1$ with a maximum eigenvalue of 1.5 so that $\Delta t = \Delta x \frac{\sigma}{1.5}$. The l_1-errors, computed at cell-midpoints, have been generated relative to the earlier mentioned order three SSP-DG reference solution.

Overall, the MDRKCAT methods compare well with the DG solutions. A large discrepancy can be noticed in the manner at which the expected convergence order

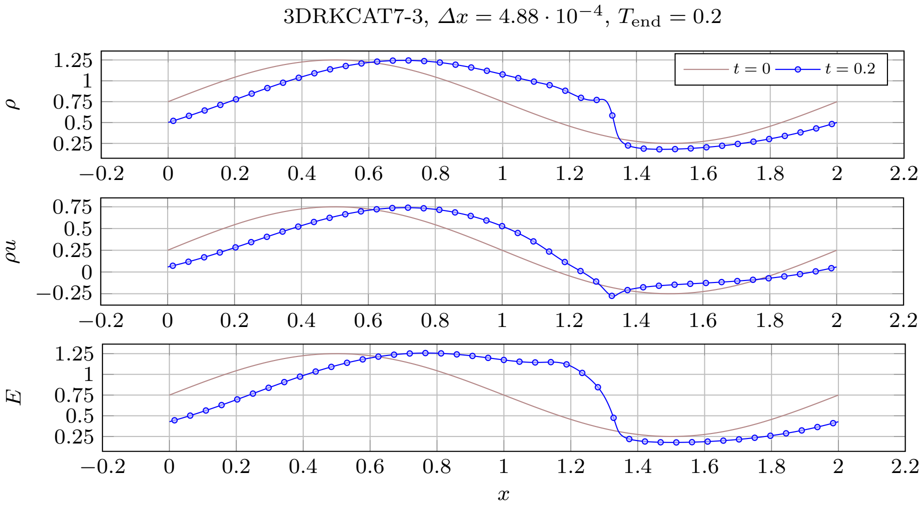


Fig. 6: MDRKCAT 1D Euler solution of $\rho_0(x) = 0.75 + 0.5\sin(\pi x)$, $(\rho u)_0(x) = 0.25 + 0.5\sin(\pi x)$ and $E_0(x) = 0.75 + 0.5\sin(\pi x)$ up to $T_{\text{end}} = 0.2$ with $\sigma = 0.5$. The 3DRKCAT7-3 scheme has been used on $M = 4096$ cells; less nodal points are shown for better visual distinctness.

is achieved; the MDRKCAT methods gradually head toward order $\min(2p, q)$ with each refinement, whereas the DG solvers achieve convergence already going from 32 to 64 cells. This is to be expected: By definition the MDRKCAT methods only approximate the time derivatives of the flux $\partial_t^{k-1} f(w)$. Hence the achieved accuracy is intertwined with the mesh resolution of the problem at hand. For a lower amount of cells M the numerical flux $F_{i+1/2}^n$ at the faces can thus not be an accurate representation, whereas the DG solvers calculate the fluxes at the halfway points $i + 1/2$ on the basis of the exact flux $f(w)$. As soon as enough cells M are used to finely represent the initial data, full advantage can be taken of the CAT method.

Moreover, the difference between the methods should be brought into perspective by studying the amount of effective *spatial* degrees of freedom (DOF) and the effective spatial size that influences the order of accuracy. DG methods make use of numerical integration points on each cell for the integration of the solution variable multiplied with the chosen basis functions [7]. This illustrates why the DG solvers more quickly capture the expected convergence order and why a direct comparison of the DG schemes and the MDRKCAT is difficult in Fig. 7. The actual amount of spatial DOF used by `r`DRKCATq-`s` schemes is $(2p+1)M$; each node uses its own local stencil in the calculations. However, as explained in [3], the local stencils are merely a manner to assure that the CAT methods linearly reduce back to Lax-Wendroff schemes. The same accuracy is achieved by the approximate Taylor methods in [36] of which the CAT procedure is established. Summing up, we can conclude that the novel `r`DRKCATq-`s` schemes compare well with a state-of-the-art DG solver in terms of accuracy.

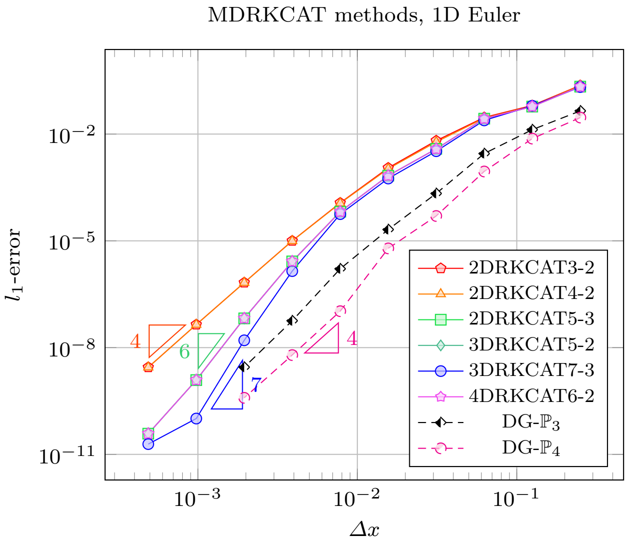


Fig. 7: Convergence order of the explicit MDRKCAT methods applied to $\rho_0(x) = 0.75 + 0.5\sin(\pi x)$, $(\rho u)_0(x) = 0.25 + 0.5\sin(\pi x)$ and $E_0(x) = 0.75 + 0.5\sin(\pi x)$ up to $T_{\text{end}} = 0.2$ with $\sigma = 0.15$. Very similar behavior can be seen between the 2DRKCAT3-2 and the 2DRKCAT4-2 schemes. The same can be said for the 5th order MDRKCAT schemes and 4DRKCAT6-2. For comparison, l_1-errors of a DG code with basis functions in $\mathbb{P}_3$ and $\mathbb{P}_4$ and CFL $\sigma = 0.1$ are visualized.

7 Conclusion and outlook

In this paper we have formulated a family of Jacobian-free multistage multiderivative solvers for hyperbolic conservation laws, so-called MDRKCAT methods. Following the Compact Approximate Taylor (CAT) method in [3], instead of computing the exact flux derivative expressions, local approximations for the time derivatives of the fluxes are obtained recursively. There are many advantages by virtue of this procedure: no costly symbolic computations are needed; and we hope that many multiderivative Runge-Kutta (MDRK) schemes can now actually be of practical use.

Both theoretically and numerically it is proven that the desired convergence order $\min(2p, q)$ is achieved, $2p$ being the spatial order and q the temporal order. Universally among the different test cases the spatial accuracy is seen to be dominant. A comparison with SSP-DG methods for the Euler equations shows that MDRKCAT methods compare well with state-of-the-art schemes in terms of accuracy.

A von Neumann analysis revealed that the stability of the MDRKCAT methods depends heavily on the number of stages and the underlying high-order Lax-Wendroff method. The latter one solely utilizes centered differences for the spatial discretization. Consequently, odd-derivative Runge-Kutta schemes seem less adequate in conjuction with the CAT algorithm.

In the future, there are two main routes to follow: extend and apply the scheme to more challenging settings and to further examine the stability properties of the novel scheme. Concerning more challenging settings the investigation of multidimensional hyperbolic conservation laws with (possibly) unstructured meshes and parabolic PDEs with viscous effects are attractive. In order to accomplish such extensions it might be interesting to combine MDRKCAT methods with DG techniques [27]. Presumably, also implicit MDRK schemes need to be considered to take care of the diffusive effects. A possible starting point could be the implicit variant of the approximate Taylor methods, which have been recently developed for ODEs in [2]. Concerning the stability properties of the scheme one could think of exploring more types of MDRK schemes, possibly with SSP properties [6, 11]. Moreover, at the same time, it will be possible to identify more efficient schemes.

Declarations

Funding J. Zeifang was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through the project no. 457811052. The HPC-resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government.

Conflicts of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Availability of data and material The datasets generated and/or analyzed during the current study are available from the corresponding author on reasonable request `jeremy.chouchoulis@uhasselt.be`.

Code availability The code can be downloaded from the personal webpage of Jochen Schütz at `www.uhasselt.be/cmat` or directly from `http://www.uhasselt.be/Documents/CMAT/Code/MDRKCAT-CMAT.zip`.

A Butcher tableaux

In this section, we show the multiderivative Runge-Kutta methods used in this work through their Butcher tableaux. We use three two-derivative methods taken from [5], see Tbl. 3–5; two three-derivative methods taken from [21], see Tbl. 6–7; and one four-derivative method, constructed for this paper, see Tbl. 8. This last scheme has been derived from the idea that it should be of form

$$y^{n,l} = y^n + \sum_{k=1}^{\mathtt{r}-1} \frac{(c_l \Delta t)^k}{k!} \Phi^{(k-1)}\left(y^n\right) + \Delta t^{\mathtt{r}} \sum_{\nu=1}^{l-1} a_{l\nu}^{(\mathtt{r})} \Phi^{(\mathtt{r}-1)}\left(y^{n,\nu}\right) ,$$

for $l = 1, \ldots, \mathtt{s}$, with update

$$y^{n+1} = y^n + \sum_{k=1}^{\mathtt{r}-1} \frac{\Delta t^k}{k!} \Phi^{(k-1)}\left(y^n\right) + \Delta t^{\mathtt{r}} \sum_{l=1}^{\mathtt{s}} b_l^{(\mathtt{r})} \Phi^{(\mathtt{r}-1)}(y^{n,l}) .$$

These forms have also been used in [5] and [21].

Table 3: 2DRK3-2 - Third order two-derivative Runge-Kutta scheme using two stages [5].

0	0	0	0	0
1	1	0	1/2	0
	2/3	1/3	1/6	0

Table 4: 2DRK4-2 - Fourth order two-derivative Runge-Kutta scheme using two stages [5].

0	0	0	0	0
1/2	1/2	0	1/8	0
	1	0	1/6	1/3

Table 5: 2DRK5-3 - Fifth order two-derivative Runge-Kutta scheme using three stages [5].

0	0	0	0	0	0	0
2/5	2/5	0	0	2/25	0	0
1	1	0	0	−1/4	3/4	0
	1	0	0	1/8	25/72	1/36

Table 6: 3DRK5-2 - Fifth order three-derivative Runge-Kutta scheme using two stages [21].

0	0	0	0	0	0	0
2/5	2/5	0	2/25	0	4/375	0
	1	0	1/2	0	1/16	5/48

Table 7: 3DRK7-3 - Seventh order three-derivative Runge-Kutta scheme using three stages [21]. The coefficients are given by $c_2 = \frac{3-\sqrt{2}}{7}$, $c_3 = \frac{3+\sqrt{2}}{7}$, $a_{32}^{(3)} = \frac{122+71\sqrt{2}}{7203}$, $b_1^{(3)} = \frac{1}{30}$, $b_2^{(3)} = \frac{1}{15} + \frac{13\sqrt{2}}{480}$, $b_3^{(3)} = \frac{1}{15} - \frac{13\sqrt{2}}{480}$.

0	0	0	0	0	0	0	0	0	0
c_2	c_2	0	0	$c_2^2/2$	0	0	$c_2^3/6$	0	0
c_3	c_3	0	0	$c_3^2/2$	0	0	$c_2^3/6 - a_{32}^{(3)}$	$a_{32}^{(3)}$	0
	1	0	0	1/2	0	0	$b_1^{(3)}$	$b_2^{(3)}$	$b_3^{(3)}$

Table 8: 4DRK6-2 - Sixth order four-derivative Runge-Kutta scheme using two stages.

0	0	0	0	0	0	0	0	0
1/3	1/3	0	1/18	0	1/162	0	1/1944	0
	1	0	1/2	0	1/6	0	1/60	1/40

References

1. M. Abramowitz and I. A. Stegun, editors. *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables.* Dover Publications, Inc., New York, 1964.
2. A. Baeza, R. Bürger, M. d. C. Martí, P. Mulet, and D. Zorío. On approximate implicit Taylor methods for ordinary differential equations. *Computational and Applied Mathematics*, 39(4):304, 2020.

3. H. Carrillo and C. Parés. Compact approximate Taylor methods for systems of conservation laws. *Journal of Scientific Computing*, 80(3):1832–1866, 2019.
4. H. Carrillo, C. Parés, and D. Zorío. Lax-Wendroff approximate Taylor methods with fast and optimized weighted essentially non-oscillatory reconstructions. *Journal of Scientific Computing*, 86(1), 2021.
5. R. Chan and A. Tsai. On explicit two-derivative Runge-Kutta methods. *Numerical Algorithms*, 53:171–194, 2010.
6. A. J. Christlieb, S. Gottlieb, Z. J. Grant, and D. C. Seal. Explicit strong stability preserving multistage two-derivative time-stepping schemes. *Journal of Scientific Computing*, 68:914–942, 2016.
7. B. Cockburn, G. E. Karniadakis, and C.-W. Shu. The development of discontinuous Galerkin methods. In B. Cockburn, G. E. Karniadakis, and C.-W. Shu, editors, *Discontinuous Galerkin Methods: Theory, Computation and Applications*, volume 11 of *Lecture Notes in Computational Science and Engineering*, pages 3–50. Springer, 2000.
8. M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz. A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes. *Journal of Computational Physics*, 227(18):8209–8253, 2008.
9. M. Dumbser, C. Enaux, and E. F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *Journal of Computational Physics*, 227(8):3971–4001, 2008.
10. M. Dumbser, F. Fambri, M. Tavelli, M. Bader, and T. Weinzierl. Efficient implementation of ADER discontinuous Galerkin schemes for a scalable hyperbolic PDE engine. *Axioms*, 7(3), 2018.
11. S. Gottlieb, Z. J. Grant, J. Hu, and R. Shu. High order unconditionally strong stability preserving multi-derivative implicit and IMEX Runge–Kutta methods with asymptotic preserving properties. *arXiv preprint arXiv:2102.11939*, 2021.
12. S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Review*, 43(1):89–112, 2001.
13. W. Guo, J.-M. Qiu, and J. Qiu. A new Lax–Wendroff discontinuous Galerkin method with superconvergence. *Journal of Scientific Computing*, 65(1):299–326, 2015.
14. E. Hairer and G. Wanner. *Solving ordinary differential equations II.* Springer Series in Computational Mathematics, 1991.
15. A. Jaust, J. Schütz, and D. C. Seal. Implicit multistage two-derivative discontinuous Galerkin schemes for viscous conservation laws. *Journal of Scientific Computing*, 69:866–891, 2016.
16. P. Lax and B. Wendroff. Systems of conservation laws. *Communications on Pure and Applied Mathematics*, 13(2):217–237, 1960.
17. R. J. LeVeque. *Numerical Methods for Conservation Laws.* Birkhäuser Basel, 1990.
18. J. Li and Z. Du. A two-stage fourth order time-accurate discretization for Lax–Wendroff type flow solvers I. Hyperbolic conservation laws. *SIAM Journal on Scientific Computing*, 38(5):A3046–A3069, 2016.
19. C. Lu and J. Qiu. Simulations of shallow water equations with finite difference Lax-Wendroff weighted essentially non-oscillatory schemes. *Journal of Scientific Computing*, 47(3):281–302, Jun 2011.
20. MathWorks. *Symbolic Math Toolbox.* Natick, Massachusetts, United States, 2020. URL https://www.mathworks.com/help/symbolic/.
21. M. Ökten Turacı and T. Öziş. Derivation of three-derivative Runge-Kutta methods. *Numerical Algorithms*, 74(1):247–265, 2017.
22. J. Qiu. Development and comparison of numerical fluxes for LWDG methods. *Numerical Mathematics: Theory, Methods and Applications*, 1(4):435–459, 2008.
23. J. Qiu, M. Dumbser, and C.-W. Shu. The discontinuous Galerkin method with Lax–Wendroff type time discretizations. *Computer Methods in Applied Mechanics and Engineering*, 194(42-44):4528–4543, 2005.
24. J. Qiu and C.-W. Shu. Finite difference WENO schemes with Lax-Wendroff-type time discretizations. *SIAM Journal of Scientific Computing*, 24(6):2185–2198, 2003.
25. A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics.* Springer, 2007.
26. A. Ralston and P. Rabinowitz. *A First Course in Numerical Analysis.* Dover books on mathematics. Dover Publications, 2001.
27. J. Schütz, D. C. Seal, and A. Jaust. Implicit multiderivative collocation solvers for linear partial differential equations with discontinuous Galerkin spatial discretizations. *Journal*

of Scientific Computing, 73:1145–1163, 2017.

28. J. Schütz, D. C. Seal, and J. Zeifang. Parallel-in-time high-order multiderivative IMEX methods. *Accepted for publication in Journal of Scientific Computing,* *http://www.uhasselt.be/Documents/CMAT/Preprints/2021/UP2101.pdf*, 2021.
29. T. Schwartzkopff, M. Dumbser, and C.-D. Munz. ADER: A high-order approach for linear hyperbolic systems in 2d. *Journal of Scientific Computing*, 17(1-4):231–240, 2002.
30. D. C. Seal, Y. Güçlü, and A. Christlieb. High-order multiderivative time integrators for hyperbolic conservation laws. *Journal of Scientific Computing*, 60:101–140, 2014.
31. R. Spiteri and S. Ruuth. A new class of optimal high-order strong-stability-preserving time discretization methods. *SIAM Journal on Numerical Analysis*, 40, 2002.
32. J. C. Strikwerda. *Finite Difference Schemes and Partial Differential Equations, Second Edition.* Society for Industrial and Applied Mathematics, 2004.
33. V. A. Titarev and E. F. Toro. ADER: Arbitrary high order Godunov approach. *Journal of Scientific Computing*, 17(1):609–618, 2002.
34. V. A. Titarev and E. F. Toro. ADER schemes for three-dimensional non-linear hyperbolic systems. *Journal of Computational Physics*, 204(2):715–736, 2005.
35. G. Whitham. *Linear and Nonlinear Waves.* Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley, 2011.
36. D. Zorío, A. Baeza, and P. Mulet. An approximate Lax-Wendroff-type procedure for high order accurate schemes for hyperbolic conservation laws. *Journal of Scientific Computing*, 71:246–273, 2017.