

THE STRUCTURAL METHOD FOR ORDINARY DIFFERENTIAL EQUATIONS: TOWARD EXTREME-ORDERS OF ACCURACY

ARTICLE

 **S. Clain**

Centre of Mathematics, Coimbra University, Largo D. Dinis, 3000-143 Coimbra, Portugal
clain@mat.uc.pt

 **M.T. Malheiro**

Centre of Mathematics, Campus de Gualtar, 4710-057 Braga, Portugal
mtm@math.uminho.pt

 **G.J. Machado**

Centre of Mathematics, Campus de Gualtar, 4710-057 Braga, Portugal
gjm@math.uminho.pt

 **R. Costa**

Institute for Polymers and Composites, Campus de Azurém, University of Minho,
4080-058 Guimarães, Portugal
rcosta@dep.uminho.pt

July 7, 2026

ABSTRACT

We design and analyse a new numerical method to solve ODE system based on the structural method. We compute approximations of solutions together with its derivatives up to order K by solving an entire block corresponding to R time steps. We build the physical relations that connect the function and derivative approximations at each time step by using the ODE and its derivatives and develop the structural equations that establish linear relations between the function and its derivative over the whole block of R times steps. The non-linear system is solved and provide very accurate approximations with nice spectral resolution properties.

Keywords structural method, very high-order, extreme-order, Compact scheme, EDO, stability, multi-derivatives

1 Introduction

First-order ordinary differential equations (ODEs), or systems of the form $\phi'(t) = f(\phi(t), t)$, with $t \mapsto \phi(t) \in \mathbb{R}^I$, arise in most non-stationary models to characterise the evolution of processes over time. The design of accurate and stable numerical schemes for solving these ODEs is a fundamental challenge for improving computational efficiency in critical settings, such as stiff problems or chaotic differential equations.

In the early 1970s, researchers began using the second derivative as an additional variable to improve solution accuracy and reduce the number of substeps in numerical schemes. This approach is rooted in the methodology of compact schemes, which exploit more local information to minimise stencil size [1, 2]. Early studies on multi-derivative methods include works by Urabe [3], Shintani [4, 5], Kastlunger, Hairer and Wanner [6, 7, 8], and Cash [9] within the Runge–Kutta (RK) framework. In the multistep setting, early contributors include Makinson [10], Reimer [11], Liniger and Willoughby [12], Enright [13, 14], Genin [15], Makela et al. [16], Jeltsch [17, 18, 19, 20, 21], Brown [22], Gupta [23], and Cash [24, 25]. Lambert’s book [26] discusses both approaches. References [27, 28], together with

Butcher's textbook [29], highlight Obrechhoff's contribution [30], with particular emphasis on the two-derivative case, as further explored in [31, 32, 33].

Following a productive decade in multi-derivative ODE integration, research activity declined between the 1980s and the 2010s, with only a limited number of publications. For RK methods, the main difficulty lies in the complex and highly nonlinear order conditions [27, 34, 28, 35], which complicate the construction of fourth- or higher-order schemes with more than two or three substeps. Several papers address this issue by imposing additional conditions to reduce complexity. These include explicit schemes [36, 37], methods with few substeps [38], and two-derivative cases [39, 31, 40, 32].

Alternative approaches have also been proposed to achieve very high-order approximations, in which the algebraic order conditions are replaced by a few simple linear constraints. The collocation method [41, 42] provides approximations of arbitrary order and has been identified as a particular case of a class of implicit RK methods [43]. Given the collocation points in a time interval, the algebraic order conditions become linear conditions, as expressed by relations (2.7) and (2.8) in [43]. The method is not compact in the sense considered here, since it involves implicit linear combinations of the function and its derivatives at the collocation points. Nevertheless, we would like to mention the recent works of [44] and [45], where the authors explore a novel approach by introducing second-order derivatives to reduce the stencil size.

The spectral deferred correction method is another class of high-order schemes based on successive approximations of the solution over a time interval $[t_n, t_{n+1}]$, as shown in [46, 47]; see [48] for an extensive review of the SDC method. This method also avoids the introduction of complicated nonlinear algebraic conditions [49]. A p -subdivision $t_{n,m}$, $m = 0, \dots, p$, together with local approximations $\phi_{n,m}$, is introduced to design an iterative scheme that provides a sequence $\phi_n^{[k]}$ of approximations of order $k + 1$. The upgrade operator $\phi_n^{[k]} \rightarrow \phi_n^{[k+1]}$, defined on the interval $[t_n, t_{n+1}]$, makes it possible to increase the order of accuracy using very simple numerical tools, such as first-order Euler schemes and quadrature formulae for integration. Very recently, a multi-derivative version was proposed by introducing the second-order derivative [50].

The structural method is a general technique that provides very high-order approximations by introducing derivatives up to degree K at each node t_n . Moreover, it clearly separates the physical equations, which represent the underlying mathematical model, from the structural equations, which are independent of the problem under consideration and depend only on the grid/mesh structure. The combination of these two sets of equations then yields a global (non-)linear system that provides highly accurate approximations of the function and its derivatives [51].

This method can be regarded as a generalisation of compact schemes in the sense of Lele [2], applied to the ODE setting, where implicit relations between the function and its derivatives are introduced over a prescribed stencil. In our case, the stencil is referred to as an R -block and consists of R successive time steps. As an example, the combined compact difference scheme for elliptic equations in space proposed in [52, 53, 54] can be reformulated as a structural method for ODEs in time; see [55]. It is worth noting that the same structural equations that yield a sixth-order accurate scheme in time for ODEs coincide with the Chu and Fan relations within the compact finite difference framework [52].

The structural method proposed in the present work corresponds to a particular case in which, by taking advantage of all the physical relations, as explained in Sections 2 and 3, the scheme is reinterpreted as a multi-derivative RK (MDRK) scheme. The key point is that MDRK schemes are difficult to construct because of the large number of nonlinear algebraic order conditions involved. In summary, handling the order conditions for K -derivative methods of order p is challenging. For instance, reference [56] lists 33 strongly nonlinear order conditions for a two-derivative RK method with s substeps to provide a sixth-order scheme ($p = 6$; see also [6, 57, 58]). These conditions involve $2s(s + 1)$ coefficients, arising from combinations of first and second derivatives at the different substeps. For example, incorporating third- or higher-order derivatives with s substeps in order to achieve very high accuracy appears to be intractable, potentially involving thousands of nonlinear relations to be satisfied [6, 27, 59, 60, 61].

The main advantage of structural schemes is that there is no need to verify nonlinear algebraic order conditions in order to achieve the prescribed order. Instead, the coefficients of the method are obtained from vectors belonging to the kernel of a matrix, leading to a highly efficient computation of the structural equations. Recently, the MDRK literature has focused on techniques to alleviate this major computational drawback. Explicit formulae for three-derivative and higher-derivative RK schemes have been proposed in [62, 63], while the development of new multistep schemes has been studied in [60, 64], with some improvements in accuracy reported in [65, 66, 67]. Applications include time discretisation for non-stationary partial differential equations (PDEs) [68, 69, 70, 71] and the construction of strong stability preserving (SSP) schemes of RK type [72, 73, 74, 75, 76] or multistage type [77, 78, 79] for conservation laws. A recent review [56] discusses these developments. However, all these methods address only second- or

third-derivative schemes with a small number of substeps, owing to the large number of nonlinear algebraic order conditions involved, whereas the structural method efficiently produces K -derivative schemes over an R -block at any order p by solving a simple linear problem.

This paper introduces a generalisation of the structural method for ODE systems using K derivatives, addressing key challenges in existing RK methods. The main advantages are the following:

- The principal contributions are as follows: The method corresponds to a K -derivative implicit RK method when all possible physical equations are used.
- The intermediate steps are not auxiliary variables, but genuine approximations that enjoy the optimal order of accuracy of the method.
- Unconditional stability is proved together with a formal assessment of the consistency and *a priori* errors.
- The coefficients of the structural equations are computed from the kernel of a matrix, avoiding an intractable non-linear system to solve.
- The method covers all possible meta-parameters K (number of derivatives), R (number of intermediate steps), and p (order of the method) using the same algorithm.
- Excellent spectral resolution is obtained when K increases. In particular, only two or three grid points are necessary to perfectly recover a full revolution; hence, the method is highly effective at capturing high-frequency relative to the grid characteristic size.

The document is divided into two parts. The first is devoted to the design of the method, its implementation, comparisons with existing methods, and numerical simulations. The second part addresses the theoretical aspects of convergence, stability, and spectral resolution, together with a detailed study and justification of our implementation for solving the nonlinear global system.



Part I: Method design and benchmarks

The first part focuses on the general construction of the structural equations and on the coupling between the structural and physical equations. In Section 2, we detail the technical aspects of the method, leading to different types of ODE solvers, while comparisons with classical high-order methods for ODEs are provided in Section 3. Section 4 is devoted to the presentation of a simple algorithm for solving the nonlinear systems, whereas Sections 5 and 6 present numerical evidence concerning the convergence and efficiency of the structural method for ODEs. Annexes A and B include analytical formulae for the implicit structural equations on a uniform grid, together with explicit structural equations for post-processing. We note that a numerical study for the specific case of Hamiltonian systems was carried out in [80]; however, our goal here is to generalize the approach to a generic ODE system.

2 General design of the structural method

Consider the generic first-order scalar ODE problem

$$\phi'(t) = f(\phi(t), t), \quad t \in [0, T], \quad (1)$$

together with the initial condition $\phi(0) = \phi_0$, with $T > 0$ the final time and $f \equiv f(z, t)$ a regular function in $\mathbb{R} \times [0, T]$.

Let $N \in \mathbb{N}$, and define a uniform grid $t_n = n\Delta t$ for $n \in \llbracket 0, N \rrbracket$, such that $T = N\Delta t$. Let $\phi^{(k)} \equiv \phi^{(k)}(t)$ denote the k th derivative of the solution, with $\phi^{(0)}(t) \equiv \phi(t)$ as in Eq. (1). The approximations $\phi_n^{(k)}$ represent $\phi^{(k)}(t_n)$ at each time grid point.

The structural method divides the problem into two distinct subsystems. The first subsystem contains the physical equations, which are derived directly from the generic ODE and its time derivatives. In contrast, the second subsystem comprises structural equations, defined by linear relations that depend only on the time grid structure. Notably, these structural equations do not involve the specific ODE being solved, emphasising their generality.

2.1 Physical equations

The Eq. (1), which relates the first derivative $\phi^{(1)}(t)$ with function $\phi^{(0)}(t)$ and time t , reads

$$\phi^{(1)}(t) = f\left(\phi^{(0)}(t), t\right) \equiv \mathbf{df}^{(0)}[\phi^{(0)}, t]. \quad (2)$$

The time derivative of Eq. (2) provides an additional physical equation, that reads

$$\phi^{(2)}(t) = \partial_z f\left(\phi^{(0)}(t), t\right) \phi^{(1)}(t) + \partial_t f\left(\phi^{(0)}(t), t\right) \equiv \mathbf{df}^{(1)}[\phi^{(0)}, \phi^{(1)}, t] \quad (3)$$

where $\partial_z f$ and $\partial_t f$ denote the derivatives with respect to the first and second arguments, respectively. Once again, the time derivative of Eq. (3) provides a relation between the function $\phi(t)$ and all its derivatives up to the third order, that is,

$$\begin{aligned} \phi^{(3)}(t) = \partial_z f\left(\phi^{(0)}(t), t\right) \phi^{(2)}(t) + \partial_{zz} f\left(\phi^{(0)}(t), t\right) \left[\phi^{(1)}(t)\right]^2 \\ + 2\partial_{zt} f\left(\phi^{(0)}(t), t\right) \phi^{(1)}(t) + \partial_{tt} f\left(\phi^{(0)}(t), t\right) \equiv \mathbf{df}^{(2)}[\phi^{(0)}, \phi^{(1)}, \phi^{(2)}, t]. \end{aligned} \quad (4)$$

By systematically applying the chain rule $K - 1$ times, we obtain a sequence of K physical relations. These can be represented in a generic form as

$$\phi^{(k+1)}(t) = \frac{d}{dt} \mathbf{df}^{(k-1)}[\phi^{(0)}, \phi^{(1)}, \dots, \phi^{(k-1)}, t] \equiv \mathbf{df}^{(k)}[\phi^{(0)}, \phi^{(1)}, \dots, \phi^{(k)}, t], \quad k \in \llbracket 0, K-1 \rrbracket.$$

Remark. Note that, by construction, the application

$$(z_0, z_1, \dots, z_k, t) \rightarrow \mathbf{df}^{(k)}[z_0, z_1, \dots, z_k, t]$$

is a polynomial function with respect to $z_1, \dots, z_k$ with non-linear coefficients solely depending on z_0 and t corresponding to the partial derivatives in z and time. ■

We can now introduce the notion of physical equation, which just consists of substituting the exact values with the approximations $\phi_n^{(k)}(t)$ in the previous equations. Therefore, the first physical equation, associated with time t_n and denoted as $\text{PE}_1(n)$, derives from the relation (2) and reads

$$\phi_n^{(1)} = f(\phi_n^{(0)}, t_n) \equiv \mathbf{df}^{(0)}[\phi_n^{(0)}, t_n].$$

Similarly, the second physical equation $\text{PE}_2(n)$ is deduced from relation (3), that is

$$\phi_n^{(2)} = \partial_z f(\phi_n^{(0)}, t_n) \phi_n^{(1)} + \partial_t f(\phi_n^{(0)}, t_n) \equiv \mathbf{df}^{(1)}[\phi_n^{(0)}, \phi_n^{(1)}, t_n],$$

and the third physical equation $\text{PE}_3(n)$ from relation (4), that is

$$\begin{aligned} \phi_n^{(3)} = \partial_z f\left(\phi_n^{(0)}, t_n\right) \phi_n^{(2)} + \partial_{zz} f\left(\phi_n^{(0)}, t_n\right) \left[\phi_n^{(1)}\right]^2 + 2\partial_{zt} f\left(\phi_n^{(0)}, t_n\right) \phi_n^{(1)} + \partial_{tt} f\left(\phi_n^{(0)}, t_n\right) \\ \equiv \mathbf{df}^{(2)}[\phi_n^{(0)}, \phi_n^{(1)}, \phi_n^{(2)}, t_n]. \end{aligned}$$

The physical equations associated with any derivative of order four or more are obtained in a similar way.

Remark. Each physical equation $\text{PE}_k(n)$ only involves numerical approximations of the solution and its derivatives at the same time grid point, t_n . However, to connect the values of the solution and its derivatives at different nearby time points, we introduce a second set of equations. These are called structural equations and will be defined next. ■

2.2 Structural Equations

Given K as the maximum derivative order, we construct K physical equations, denoted PE_1 through PE_K . Each equation involves the numerical values $\phi_n^{(k)}$, $k \in \llbracket 0, K \rrbracket$, at each time t_n . Once these values at t_n are available, the next task is to compute the solution and its derivatives over a group of R consecutive time steps, called a block. This means assembling a $(K + 1) \times R$ matrix containing the values of the solution and its derivatives at each time point within this block, that is

$$\Phi_n = \left(\phi_{n+r}^{(k)} \right)_{k,r}, \quad k \in \llbracket 0, K \rrbracket, \quad r \in \llbracket 1, R \rrbracket,$$

as depicted in Fig. 1.

By evaluating the K physical equations at each of the R time points t_{n+r} , $r \in \llbracket 1, R \rrbracket$, we obtain $R \times K$ relations. To uniquely determine all variables, we need R additional equations, which are drawn from the structure of the uniform time grid used in the discretisation.

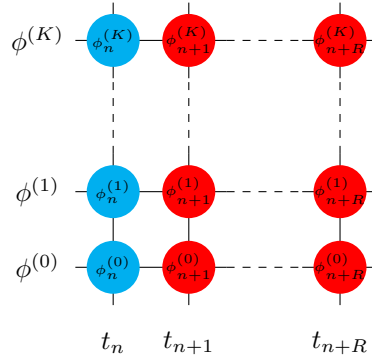


Figure 1: Grid of points for a structural scheme. The given values at time t_n are represented in blue, while the unknown values that constitute the block are marked in red.

2.2.1 General design

Structural equations are linear relations involving the approximations $\phi_{n+r}^{(k)}$, where k denotes the derivative order for $k \in \llbracket 0, K \rrbracket$ and r the time step for $r \in \llbracket 0, R \rrbracket$. More specifically, structural equation takes the form

$$\sum_{k=0}^K \sum_{r=0}^R a_{k,r} \Delta t^k \phi_{n+r}^{(k)} = 0,$$

where $\mathbf{a} = (a_{k,r})_{k,r}$, $k \in \llbracket 0, K \rrbracket$, $r \in \llbracket 0, R \rrbracket$, is the $(K+1) \times (R+1)$ matrix of coefficients, and $\Delta t^k \equiv (\Delta t)^k$ denotes powers of the time increment.

In order to determine the coefficients $a_{k,r}$ of the structural equation, we introduce the functional

$$E(\mathbf{a}; \pi) = \sum_{k=0}^K \sum_{r=0}^R a_{k,r} \Delta t^k \pi^{(k)}(t_{n+r})$$

for any function $\pi = \pi(t)$, regular enough.

We represent the matrix $\mathbf{a}$ as a vector by mapping pairs (k, r) to a single index $(k, r) \rightarrow \ell = 1 + k(R+1) + r$, from $\llbracket 0, K \rrbracket \times \llbracket 0, R \rrbracket$ onto $\llbracket 1, M \rrbracket$, $M = (K+1)(R+1)$. For simplicity, we still denote this vector as $\mathbf{a}$. Using this mapping, the above functional can be rewritten as

$$E(\mathbf{a}; \pi) = \sum_{k=0}^K \sum_{r=0}^R a_\ell \Delta t^k \pi^{(k)}(t_{n+r}), \quad \ell = \ell(k, r).$$

To construct high-order accurate schemes, we require the functional to hold exactly for all polynomials up to a certain degree. For this purpose, consider the following polynomial functions

$$\pi_m(t; t_n) = \left(\frac{t - t_n}{\Delta t} \right)^{m-1}, \quad m \in \llbracket 1, M \rrbracket.$$

Setting $E(\mathbf{a}; \pi_m) = 0$ for all $m \in \llbracket 1, M \rrbracket$, a $M \times M$ linear system $\mathcal{M}\mathbf{a} = 0$ is obtained with entries

$$\mathcal{M}[m, \ell] = \pi_m^{(k)}(t_{n+r}; t_n) = C_k^m r^{m-1-k}, \quad \ell = \ell(k, r),$$

with

$$C_k^m = \begin{cases} \frac{(m-1)!}{(m-1-k)!} & \text{if } m > k, \\ 0 & \text{otherwise,} \end{cases}$$

for $k \in \llbracket 0, K \rrbracket$ and $r \in \llbracket 0, R \rrbracket$. Note that the matrix $\mathcal{M}$ does not depend on the mesh parameter Δt or on the initial time t_n . This independence results from the use of a uniform mesh and the definition of the functions $\pi_m(\cdot; t_n)$.

Remark. Illustrating for $K = 2$ and generic R , the matrix $\mathcal{M}$ is given as

$$\mathcal{M} = \begin{pmatrix} 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & R^1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & R^2 & 0 & 2 & \cdots & 2 \times R^1 & 2 & 0 & \cdots & 2 \\ 0 & 1 & \cdots & R^3 & 0 & 3 & \cdots & 3 \times R^2 & 0 & 3 \times 2 & \cdots & 3 \times 2 \times R^1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \cdots & R^{M-1} & 0 & (M-1) & \cdots & (M-1) \times R^{M-2} & 0 & (M-1)(M-2) & \cdots & (M-1)(M-2)R^{M-3} \end{pmatrix}.$$

We now analyse the properties of the linear system $\mathcal{M}\mathbf{a} = 0$.

Definition 2.1. A set of functions $(\nu_m(t))_{m=1,\dots,M}$ is resolvent of E if, for any $m \in \llbracket 1, M \rrbracket$, we have that, if $E(\mathbf{a}; \nu_m) = 0$, then $\mathbf{a} = 0$.

Proposition 2.1. The set of M polynomial functions of degree less than or equal to $M-1$

$$\tau_{k,r}(t) := \tau_{\ell(k,r)}(t) = \left(\frac{t - t_{n+r}}{\Delta t} \right)^k \prod_{\substack{s \in \llbracket 0, R \rrbracket \\ s \neq r}} \left(\frac{t - t_{n+s}}{\Delta t} \right)^{K+1}, \quad k \in \llbracket 0, K \rrbracket, r \in \llbracket 0, R \rrbracket,$$

is resolvent.

Proof. First, observe that the degree of all these polynomial functions is strictly less than M . Second, these polynomials are linearly independent and therefore form a basis for $\mathbb{P}_{M-1}$. Note that for any $k' \leq K$ and $r' \neq r$, we have $\tau_{k,r}^{(k')}(t_{n+r'}) = 0$, then

$$\begin{aligned} E(\mathbf{a}; \tau_{k,r}) &= \sum_{k'=0}^K \sum_{r'=0}^R a_{k',r'} \Delta t^{k'} \tau_{k,r}^{(k')}(t_{n+r'}) = \sum_{k'=0}^K a_{k',r} \Delta t^{k'} \tau_{k,r}^{(k')}(t_{n+r}) \\ &= \sum_{k'=k}^K a_{k',r} \Delta t^{k'} \tau_{k,r}^{(k')}(t_{n+r}) = \sum_{k'=k}^K a_{k',r} \Delta t^{k'-k} b_{k'-k,r}, \end{aligned}$$

where for $k' \geq k$ we have

$$b_{k'-k,r} = \left[\prod_{s \neq r} \left(\frac{t - t_{n+s}}{\Delta t} \right)^{K+1} \right]_{t=t_{n+r}}^{(k'-k)}.$$

Notice that for $k = k'$, we obtain

$$b_{0,r} = \left[\prod_{s \neq r} \left(\frac{t - t_{n+s}}{\Delta t} \right)^{K+1} \right]_{t=t_{n+r}} = \prod_{s \neq r} (r - s)^{K+1} \neq 0.$$

The rest of the proof is achieved by induction. Let $r \in \llbracket 0, R \rrbracket$. For $k = K$, we just have

$$0 = E(\mathbf{a}; \tau_{K,r}) = a_{K,r} b_{0,r};$$

hence $a_{K,r} = 0$. Now, take $k = K-1$, then we have

$$0 = E(\mathbf{a}; \tau_{K-1,r}) = a_{K,r} \Delta t b_{1,r} + a_{K-1,r} b_{0,r} = a_{K-1,r} b_{0,r}.$$

Therefore $a_{K-1,r} = 0$. Similarly, we successively prove that $a_{K-2,r} = 0$, $a_{K-3,r} = 0$, $\dots$, $a_{0,r} = 0$. In conclusion, all the coefficients of $\mathbf{a}$ are null, so the set of polynomial functions is resolvent. $\square$

Corollary 2.2. The matrix $\mathcal{M}$ is non-singular.

Proof. The sets of polynomial functions $\tau_{k,r}$ and π_m both define a basis for $\mathbb{P}_{M-1}$. Since the set $\tau_{k,r}$ is resolvent, the set of functions π_m is also resolvent. Thus, the matrix $\mathcal{M}$ is non-singular. $\square$

Definition 2.2. Let S be the number of structural equations we intend to produce. We denote by $\mathcal{K}^S \subset \mathbb{R}^M$ the kernel of the rectangular $(M-S) \times M$ matrix $\mathcal{M}[1 : M-S, :]$, with the last S rows eliminated.

Since $\mathcal{M}$ is non-singular, the matrix $\mathcal{M}[1 : M-S, :]$ enjoys the maximum rank property, therefore the dimension of $\mathcal{K}^S$ is S and contains a basis of S linear independent vectors of $\mathbb{R}^M$ we shall denote $\mathbf{a}^s$, $s \in \llbracket 1, S \rrbracket$.

Definition 2.3. Given K the number of derivatives, R the size of the block, and S the total number of structural equations, we denote by $SE_{KR}^S(s)$, $s \in \llbracket 1, S \rrbracket$, the structural equation number s corresponding to the vector $\mathbf{a}^s \in \mathcal{K}^S$.

Definition 2.4. We denote by $\mathbf{A}$ the $S \times M$ matrix composed of the line vectors $(\mathbf{a}^s)^t$, $s \in \llbracket 1, S \rrbracket$.

2.2.2 Choice of the structural equations

The first observation is that, although the kernel $\mathcal{K}^S$ is unique as a vectorial subspace, we do not have the uniqueness of the basis, that is, the uniqueness of the set of S structural equations. We list several methods to produce the S basis vectors.

I. Analytical method. Analytical expressions of the coefficients of the system $E(\mathbf{a}; \pi_m) = 0$, $m \in \llbracket 1, M - S \rrbracket$, can be derived by symbolic computation assuming a uniform time subdivision. Such relations avoid the computation of matrix $\mathcal{M}$ and are mandatory for stability analysis. In A we report several sets of structural equations $\text{SE}_{\text{KR}}^S(s)$ for useful values of K and R .

Remark. The analytical expressions of vectors $(\mathbf{a}^s)_{s=1}^S$ given in A do not necessarily correspond to an orthogonal basis of $\mathcal{K}^S$, that is, we do not necessarily have $\langle \mathbf{a}^s, \mathbf{a}^m \rangle = \delta_{sm}$. ■

II. Numerical method. For larger values of K and R or when dealing with a non-uniform grid, we compute an orthogonal basis of the kernel $\mathcal{K}^S$ using the complete SVD decomposition

$$\mathcal{M}[1 : M - S, 1 : M] = U \Sigma V^t$$

with U a rectangular $(M - S) \times M$ matrix of orthogonal line vectors, Σ the diagonal $M \times M$ matrix, and V^t the square orthogonal $M \times M$ matrix. Since the S last diagonal entries of matrix Σ are null, we deduce that the S last column vectors of matrix V define an orthogonal basis of $\mathcal{K}^S$.

III. Hierarchical basis. We want the first vector $\mathbf{a}^1$ to provide the largest order of consistency error. The second vector $\mathbf{a}^2$ reduces the consistency order of one and so forth. To provide such a property, we propose an algorithm to set a numerical hierarchical basis of the kernel $\mathcal{K}^S$.

1. Firstly compute the sole vector $\mathbf{a}^1$ of the kernel $\mathcal{K}^1$.
2. Then, determine a basis of the kernel $\mathcal{K}^2$. Noting that $\mathbf{a}^1$ automatically belongs to $\mathcal{K}^2$, we determine a normalised vector $\mathbf{a}^2 \in \mathcal{K}^2$ orthogonal to $\mathcal{K}^1$ using an orthogonalisation procedure. Moreover, the consistency error associated with $\mathbf{a}^2$ is one order lower than $\mathbf{a}^1$ as the number of constraints $E(\mathbf{a}, \pi_m) = 0$ has been reduced.
3. For a given $S' < S$, assume that the set of vectors $(\mathbf{a}^s)_{s=1}^{s=S'} \in \mathcal{K}^{S'}$ is orthogonal. Then, determine the kernel $\mathcal{K}^{S'+1}$ and extract a new orthogonal vector $\mathbf{a}^{S'+1} \in \mathcal{K}^{S'+1}$ orthogonal to $\mathcal{K}^{S'}$. By construction, $\mathbf{a}^{S'+1}$ provides a reduced consistency of one order with respect to $\mathbf{a}^{S'}$.
4. The construction ends when $S' = S$ is reached and $\{\mathbf{a}^1, \dots, \mathbf{a}^S\}$ is a hierarchical basis for $\mathcal{K}^S$.

2.2.3 Equivalent set of structural equations

As we pointed out in the previous section, one can determine different representations/bases of the kernel $\mathcal{K}^S$, that is, different sets of structural equations. This question is important because of the impact of the basis choice. Indeed, we can consider two different bases $(\mathbf{a}^s)_{s=1}^S$ and $(\tilde{\mathbf{a}}^s)_{s=1}^S$ of $\mathcal{K}^S$ from which we build two different sets of structural equations. Consequently, there exists a change of basis given by a non-singular $S \times S$ matrix $\Xi = [\xi_{ij}]_{S \times S}$ such as $\tilde{\mathbf{A}} = \Xi \mathbf{A}$, where

$$\tilde{\mathbf{A}} = [(\tilde{\mathbf{a}}^s)^t] \quad \text{and} \quad \mathbf{A} = [(\mathbf{a}^s)^t] \quad (5)$$

are two $S \times M$ matrices. Conversely, any non-singular matrix Ξ transforms a basis of $\mathcal{K}^S$ into another basis of $\mathcal{K}^S$.

Remark. By construction, any set of structural equations provides the same global consistency error order $M - S$ since the equations are exact for any π_m , $m \in \llbracket 1, M - S \rrbracket$. ■

2.3 Design of the schemes

To build a complete numerical scheme to solve a first-order ODE, the set of physical and structural equations to be used must be determined. Several (meta)parameters control the scheme's complexity and accuracy, particularly the number of physical equations per grid point (K) and the number of grid points per block (R).

2.3.1 Notations

Given K and R , the data are arranged into two vectors.

- The approximations regarding the same derivation order k are gathered into vectors

$$\Phi_n^{(k)} = \left(\phi_{n+1}^{(k)}, \dots, \phi_{n+R}^{(k)} \right)^t \in \mathbb{R}^R$$

while $T_n = (t_{n+1}, \dots, t_{n+R})^t$ stands for the time-step vector.

- The approximations for the same time step t_{n+r} are gathered into vectors

$$\Psi_{n+r} = \left(\phi_{n+r}^{(0)}, \dots, \phi_{n+r}^{(K)} \right)^t \in \mathbb{R}^{K+1}.$$

The generic formulation then is, given the initial vector Ψ_n at time t_n , we seek the matrix $\Phi_n = \left(\phi_{n+r}^{(k)} \right)_{k,r} = [\Psi_{n+1}, \dots, \Psi_{n+R}] \in \mathbb{R}^{(K+1) \times R}$, solution of the following system:

- $K \times R$ physical equations $\text{PE}_k(r)$, $k \in \llbracket 1, K \rrbracket$, $r \in \llbracket 1, R \rrbracket$, and
- R structural equations $\text{SE}_{KR}^R(s)$, $s \in \llbracket 1, R \rrbracket$.

Remark. Note that, since we want to consider the maximum number of physical equations, we need R structural equations. ■

Definition 2.5. Functions $\mathbf{df}^{(k-1)}[z_0, z_1, \dots, z_{k-1}, t]$, defined for real value entries, are extended to their vectorial form

$$\Phi_n^{(k)} = \mathbf{df}^{(k-1)}[\Phi_n^{(0)}, \Phi_n^{(1)}, \dots, \Phi_n^{(k-1)}, T_n]$$

by applying the function component-wise, that is

$$\phi_{n+r}^{(k)} = \mathbf{df}^{(k-1)}[\phi_{n+r}^{(0)}, \phi_{n+r}^{(1)}, \dots, \phi_{n+r}^{(k-1)}, t_{n+r}], \quad r \in \llbracket 1, R \rrbracket.$$

2.3.2 Structural schemes

Structural scheme $\text{Skm}[1, R]$. The structural scheme $\text{Skm}[1, R]$ involves $2R$ unknowns per block, namely $\phi_{n+r}^{(0)}$ and $\phi_{n+r}^{(1)}$, $r \in \llbracket 1, R \rrbracket$, given $\phi_n^{(0)}$ and $\phi_n^{(1)}$. Therefore, it requires R physical equations and R structural equations. The scheme then reads

$$\begin{cases} \left\{ \begin{array}{l} \phi_{n+r}^{(1)} = \mathbf{df}^{(0)}[\phi_{n+r}^{(0)}, t_{n+r}], \quad r \in \llbracket 1, R \rrbracket, \\ \sum_{k=0}^1 \sum_{r=0}^R a_{k,r}^s \Delta t^k \phi_{n+r}^{(k)} = 0, \quad s \in \llbracket 1, R \rrbracket. \end{array} \right. \end{cases}$$

The analytical expressions of the coefficients of the structural equations are given in A. We rewrite the system in a more compact way

$$\begin{cases} \Phi_n^{(1)} = \mathbf{df}^{(0)}[\Phi_n^{(0)}, T_n], \\ \mathbf{A}^{(0)} \Phi_n^{(0)} + \Delta t \mathbf{A}^{(1)} \Phi_n^{(1)} + \hat{\mathbf{A}}_n \Psi_n = 0, \end{cases}$$

where $\mathbf{df}^{(0)}[\Phi_n^{(0)}, T_n]$ applies the first physical equation all over the block, while the structural equations provide a linear system with

$$\mathbf{A}^{(0)} = \left(a_{0,r}^s \right)_{r,s=1}^R \in \mathbb{R}^{R \times R}, \quad \mathbf{A}^{(1)} = \left(a_{1,r}^s \right)_{r,s=1}^R \in \mathbb{R}^{R \times R},$$

and

$$\hat{\mathbf{A}}_n = \left(a_{k,0}^s \Delta t^k \right)_{\substack{s=1,\dots,R \\ k=0,1}} \in \mathbb{R}^{R \times 2}, \quad \Psi_n = \left(\phi_n^{(0)}, \phi_n^{(1)} \right)^t.$$

Structural scheme $\text{Skm}[2, R]$. From the first and second physical equations, the $\text{Skm}[2, R]$ scheme reads

$$\begin{cases} \left\{ \begin{array}{l} \phi_{n+r}^{(1)} = \mathbf{df}^{(0)}[\phi_{n+r}^{(0)}, t_{n+r}], \quad r \in \llbracket 1, R \rrbracket, \\ \phi_{n+r}^{(2)} = \mathbf{df}^{(1)}[\phi_{n+r}^{(0)}, \phi_{n+r}^{(1)}, t_{n+r}], \quad r \in \llbracket 1, R \rrbracket, \\ \sum_{k=0}^2 \sum_{r=0}^R a_{k,r}^s \Delta t^k \phi_{n+r}^{(k)} = 0, \quad s \in \llbracket 1, R \rrbracket. \end{array} \right. \end{cases}$$

We rewrite the equations in a more compact way

$$\begin{cases} \Phi_n^{(1)} = \mathbf{df}^{(0)}[\Phi_n^{(0)}, T_n], \\ \Phi_n^{(2)} = \mathbf{df}^{(1)}[\Phi_n^{(0)}, \Phi_n^{(1)}, T_n], \\ \mathbf{A}^{(0)}\Phi_n^{(0)} + \Delta t \mathbf{A}^{(1)}\Phi_n^{(1)} + \Delta t^2 \mathbf{A}^{(2)}\Phi_n^{(2)} + \hat{\mathbf{A}}_n \Psi_n = 0. \end{cases}$$

Remark. The structural method is easily extended to higher derivatives. For example, scheme $Skm[R, 3]$, using the compact notation, reads

$$\begin{cases} \Phi_n^{(1)} = \mathbf{df}^{(0)}[\Phi_n^{(0)}, T_n], \\ \Phi_n^{(2)} = \mathbf{df}^{(1)}[\Phi_n^{(0)}, \Phi_n^{(1)}, T_n], \\ \Phi_n^{(3)} = \mathbf{df}^{(2)}[\Phi_n^{(0)}, \Phi_n^{(1)}, \Phi_n^{(2)}, T_n], \\ \mathbf{A}^{(0)}\Phi_n^{(0)} + \Delta t \mathbf{A}^{(1)}\Phi_n^{(1)} + \Delta t^2 \mathbf{A}^{(2)}\Phi_n^{(2)} + \Delta t^3 \mathbf{A}^{(3)}\Phi_n^{(3)} + \hat{\mathbf{A}}_n \Psi_n. \end{cases}$$

Note that the matrices $\mathbf{A}^{(0)}, \mathbf{A}^{(1)}, \dots, \mathbf{A}^{(K)}$ and $\hat{\mathbf{A}}$ have to be recomputed with the kernel method for each scheme. ■

3 Relation with other methods and novelty

This section was initially motivated by a private communication from Professor Robert McLachlan concerning the interpretation of the structural scheme as an MDRK scheme when the maximum number of physical equations is used. We further extend the comparison to other traditional, well-established methods. Indeed, the R structural equations, combined with the physical equations, correspond to particular cases of Multi-Derivative Runge–Kutta methods. We prove this equivalence for $K = 1$ corresponding to the classical RK scheme, and for $K = 2$ corresponding to the two-derivative RK scheme.).

3.1 Structural method with $K = 1$

3.1.1 An implicit RK reformulation

Let R be given, we just use the ODE as physical equations at point t_{n+r} , $r \in \llbracket 1, R \rrbracket$, and the structural equations read

$$\sum_{r=0}^R a_{0,r}^s \phi_{n+r}^{(0)} + \Delta t \sum_{r=0}^R a_{1,r}^s \phi_{n+r}^{(1)} = 0, \quad s \in \llbracket 1, R \rrbracket,$$

that are rewritten in the form

$$\sum_{r=1}^R a_{0,r}^s \phi_{n+r}^{(0)} + \Delta t \sum_{r=0}^R a_{1,r}^s \phi_{n+r}^{(1)} = -a_{0,0}^s \phi_n^{(0)}, \quad s \in \llbracket 1, R \rrbracket.$$

Noting that the constant function satisfies the structural equation, we deduce that $-a_{0,0}^s = \sum_{r=1}^R a_{0,r}^s$; hence,

$$\sum_{r=1}^R a_{0,r}^s \phi_{n+r}^{(0)} + \Delta t \sum_{r=0}^R a_{1,r}^s \phi_{n+r}^{(1)} = \phi_n^{(0)} \sum_{r=1}^R a_{0,r}^s, \quad s \in \llbracket 1, R \rrbracket.$$

Let $\mathbf{A}_0 = (a_{0,r}^s)_{r,s}$ be the $R \times R$ matrix for the function approximations, $\mathbf{A}_1 = (a_{1,r}^s)_{s,r}$ the $R \times (R+1)$ matrix for the derivative approximations, and $\mathbf{e}$ the vector of entries equal to 1. Then, we have

$$\mathbf{A}_0 \Phi^{(0)} + \Delta t \mathbf{A}_1 \Phi^{(1)} = \mathbf{A}_0 \mathbf{e} \phi_n^{(0)}.$$

Assuming that $\mathbf{A}_0$ is non-singular, then

$$\Phi^{(0)} + \Delta t \mathbf{A}_0^{-1} \mathbf{A}_1 \Phi^{(1)} = \mathbf{e} \phi_n^{(0)}$$

and the system becomes

$$\Phi^{(0)} = \mathbf{e} \phi_n^{(0)} - \Delta t \mathbf{A}_0^{-1} \mathbf{A}_1 \Phi^{(1)},$$

which corresponds to the $(R + 1)$ -stage RK scheme, given as

$$\begin{aligned}\phi_n^{(0)} &= \phi_n^{(0)} + \Delta t \sum_{r=0}^R 0 f(\phi_{n+r}^{(0)}, t_{n+r}), \\ \phi_{n+s}^{(0)} &= \phi_n^{(0)} + \Delta t \sum_{r=0}^R \alpha_{s,r} f(\phi_{n+r}^{(0)}, t_{n+r}), \quad s \in \llbracket 1, R \rrbracket,\end{aligned}$$

where the coefficients $\alpha_{s,r}$ are the entries of the $R \times (R + 1)$ matrix $\mathbf{A}_0^{-1} \mathbf{A}_1$ and intermediate uniform sub-steps [81] over the large time-step $\Delta T = R\Delta t$. Using the RK formalism, we have $c_s = s/R$, $s \in \llbracket 0, R \rrbracket$, and

$$\begin{aligned}Y_s &= \phi_n^{(0)} + \Delta T \sum_{r=0}^R a_{s,r} f(Y_r, t_n + c_r \Delta T), \\ \phi_{n+R}^{(0)} &= \phi_n^{(0)} + \Delta T \sum_{r=0}^R b_r f(Y_r, t_n + c_r \Delta T),\end{aligned}$$

with $a_{s,r} = \alpha_{s,r}/R$ and $b_r = \alpha_{R,r}/R$, $s \in \llbracket 1, R \rrbracket$, $r \in \llbracket 0, R \rrbracket$, and $a_{0r} = 0$.

Remark. An important property of our construction is that the intermediate steps yield $(R + 1)$ th-order accurate approximations. Consequently, the intermediate approximations are no longer considered auxiliary calculus but are effective approximations. ■

Example 1. For $R = 2$, the two structural equations read

$$\begin{aligned}0 &= \frac{\phi_{n+2}^{(0)} - \phi_n^{(0)}}{2} - \Delta t \frac{\phi_{n+2}^{(1)} + 4\phi_{n+1}^{(1)} + \phi_n^{(1)}}{6}, \\ 0 &= \phi_{n+2}^{(0)} - 2\phi_{n+1}^{(0)} + \phi_n^{(0)} - \Delta t \frac{\phi_{n+2}^{(1)} - \phi_n^{(1)}}{2},\end{aligned}$$

which can be rewritten with $\Delta T = 2\Delta t$ as

$$\begin{aligned}\phi_{n+2}^{(0)} &= \phi_n^{(0)} + \Delta T \frac{\phi_n^{(1)} + 4\phi_{n+1}^{(1)} + \phi_{n+2}^{(1)}}{6}, \\ \phi_{n+1}^{(0)} &= \phi_n^{(0)} + \Delta T \frac{5\phi_n^{(1)} + 8\phi_{n+1}^{(1)} - \phi_{n+2}^{(1)}}{24},\end{aligned}$$

which corresponds to the fourth-order Lobatto IIIA method, given by the Butcher tableau

0	0	0	0
$\frac{1}{2}$	$\frac{5}{24}$	$\frac{8}{24}$	$-\frac{1}{24}$
1	$\frac{1}{6}$	$\frac{4}{6}$	$\frac{1}{6}$
	$\frac{1}{6}$	$\frac{4}{6}$	$\frac{1}{6}$

3.1.2 Collocation method

One advantage of the structural scheme approach is that we provide the Butcher tableau without using the Algebraic Order Conditions [29], Sec. 23. Indeed, the number of conditions increases with the order, and the relations turn more non-linear by involving a larger number of multiplications and summations [81] for $p = 5, 6$. The RK scheme derived from the structural equations does not require any complex analytic calculation of the coefficients, since they are determined by the kernel basis. Consequently, the structural method easily provides an RK scheme of arbitrary order.

The collocation method [41, 42, 43] enjoys the same property and enables to easily design a subclass of high-order implicit RK method with simple linear conditions (the collocation point being given [43]), to provide the expected order. In short, the method consists in seeking a polynomial function π of degree m such that

$$\begin{aligned}\pi(t_n) &= \phi_n \\ \pi'(t_n + c_i \Delta t) &= f(\pi(t_n + c_i \Delta t), t_n + c_i \Delta t), i = 1, \dots, m,\end{aligned}$$

with c_i the collocation parameters. We then take $\phi_{n+1} = \pi(t_n + \Delta t)$ as an approximation at time t_{n+1} . Comparing with the structural method, we do not introduce intermediate point between t_n and t_{n+1} . Nevertheless, we can artificially built similar conditions of a collocation method by taking $\Delta T = R\Delta t$ as a super time step and $t_{n+r} = t_n + c_r \Delta R$ with $c_r = \frac{r}{R}$ as collocation coefficients uniformly distributed. The point is that the structural method uses the information $\phi_n^1 = f(\phi_n^0, t_n)$ and $\phi_{n+1}^1 = f(\phi_{n+1}^0, t_{n+1})$ in the approximation whereas the collocation method does not. Consequently, the equivalent Butcher tableau of the structural method will be different of the one associated to the collocation method.

Remark. We have presented the $Skm[1, R]$ with a uniform grid for the sake of simplicity to provide a $(R + 1)$ th-order approximations, but it is perfectly possible to consider a block of R times with a non-uniform distribution (Gauss Lobatto collocation points for instance). Questions then arise about possible improvement of accuracy and the stability of the method.

At last, the structural method enjoys the A-stability property whereas the stability of the Collocation method (A- or L-stability) depends on the choice of the coefficients c_i (Legendre, Gauss, Radeau, Lobatto) [43].

The multistep collocation method extension [82, 83] used additional log-data calculated from the past to improve the accuracy. The structural method only considers the approximations after the last time step t_n , therefore the two methods clearly differ.

3.2 Structural equations with $K = 2$

3.2.1 A Two-Derivative RK (TDRK) reformulation

We proceed with the second derivative ($K = 2$) structural method, for which the structural equations read

$$\sum_{r=0}^R a_{0,r}^s \phi_{n+r}^{(0)} + \Delta t \sum_{r=0}^R a_{1,r}^s \phi_{n+r}^{(1)} + \Delta t^2 \sum_{r=0}^R a_{2,r}^s \phi_{n+r}^{(2)} = 0, \quad s \in \llbracket 1, R \rrbracket,$$

that are rewritten in the form

$$\sum_{r=1}^R a_{0,r}^s \phi_{n+r}^{(0)} + \Delta t \sum_{r=0}^R a_{1,r}^s \phi_{n+r}^{(1)} + \Delta t^2 \sum_{r=0}^R a_{2,r}^s \phi_{n+r}^{(2)} = -a_{0,0}^s \phi_n^{(0)}, \quad s \in \llbracket 1, R \rrbracket.$$

Once again, the constant function satisfies the structural equation, then

$$\sum_{r=1}^R a_{0,r}^s \phi_{n+r}^{(0)} + \Delta t \sum_{r=0}^R a_{1,r}^s \phi_{n+r}^{(1)} + \Delta t^2 \sum_{r=0}^R a_{2,r}^s \phi_{n+r}^{(2)} = \phi_n^{(0)} \sum_{r=1}^R a_{0,r}^s, \quad s \in \llbracket 1, R \rrbracket.$$

Assuming that $\mathbf{A}_0 = (a_{0,r}^s)_{r,s}$ is a non-singular $R \times R$ matrix and defining the $R \times (R + 1)$ matrices $\mathbf{A}_1 = (a_{1,r}^s)_{s,r}$ and $\mathbf{A}_2 = (a_{2,r}^s)_{s,r}$, the structural equations are reformulated in the compact form as

$$\Phi^{(0)} = \mathbf{e} \phi_n^{(0)} - \Delta t \mathbf{A}_0^{-1} \mathbf{A}_1 \Phi^{(1)} - \Delta t^2 \mathbf{A}_0^{-1} \mathbf{A}_2 \Phi^{(2)}.$$

Denote $\alpha_{s,r}^1$ and $\alpha_{s,r}^2$ the entries of the $R \times (R + 1)$ matrices $-\mathbf{A}_0^{-1} \mathbf{A}_1$ and $-\mathbf{A}_0^{-1} \mathbf{A}_2$, respectively. The scheme is written in the form

$$\begin{aligned} \phi_{n+s}^{(0)} &= \phi_n^{(0)} + \Delta t \sum_{r=0}^R \alpha_{s,r}^1 \phi_{n+r}^{(1)} + \Delta t^2 \sum_{r=0}^R \alpha_{s,r}^2 \phi_{n+r}^{(2)}, \\ \phi_{n+s}^{(1)} &= f(\phi_{n+s}^{(0)}, t_{n+s}), \\ \phi_{n+s}^{(2)} &= \partial_z f(\phi_{n+s}^{(0)}, t_{n+s}) \phi_{n+s}^{(1)} + \partial_t f(\phi_{n+s}^{(0)}, t_{n+s}), \end{aligned}$$

with $s \in \llbracket 1, R \rrbracket$. Reshaping the scheme in the RK framework yields

$$\begin{aligned} Y_s^0 &= \phi_n^{(0)} + \Delta T \sum_{r=0}^R a_{s,r}^1 Y_r^1 + \Delta T^2 \sum_{r=0}^R a_{s,r}^2 Y_r^2, \\ Y_s^1 &= f(Y_s^0, t_n + c_s \Delta T), \\ Y_s^2 &= \partial_z f(Y_s^0, t_n + c_s \Delta T) Y_s^1 + \partial_t f(Y_s^0, t_n + c_s \Delta T), \\ \phi_{n+R}^{(0)} &= \phi_n^{(0)} + \Delta T \sum_{r=0}^R b_r^1 Y_r^1 + \Delta T^2 \sum_{r=0}^R b_r^2 Y_r^2, \end{aligned}$$

with $a_{s,r}^o = \alpha_{s,r}^o/R$ and $b_r^o = \alpha_{R,r}^o/R$, $s \in \llbracket 1, R \rrbracket$, $r \in \llbracket 0, R \rrbracket$, and $a_{0,r}^o = 0$, $o = 1, 2$, which corresponds to the TDRK scheme with $R + 1$ stages [33].

Example 2. For $R = 2$, the two structural equations read

$$\begin{aligned} 0 &= +24\phi_n^{(0)} + 9\Delta t\phi_n^{(1)} + \Delta t^2\phi_n^{(2)} - 48\phi_{n+1}^{(0)} + 0\Delta t\phi_{n+1}^{(1)} - 8\Delta t^2\phi_{n+1}^{(2)} \\ &\quad + 24\phi_{n+2}^{(0)} - 9\Delta t\phi_{n+2}^{(1)} + \Delta t^2\phi_{n+2}^{(2)}, \\ 0 &= -15\phi_n^{(0)} - 7\Delta t\phi_n^{(1)} - \Delta t^2\phi_n^{(2)} + 0\phi_{n+1}^{(0)} - 16\Delta t\phi_{n+1}^{(1)} + 0\Delta t^2\phi_{n+1}^{(2)} \\ &\quad + 15\phi_{n+2}^{(0)} - 7\Delta t\phi_{n+2}^{(1)} + \Delta t^2\phi_{n+2}^{(2)} = 0, \end{aligned}$$

which can be rewritten with $\Delta T = 2\Delta t$ as

$$\begin{aligned} \phi_{n+2}^{(0)} &= \phi_n^{(0)} + \Delta T \frac{7\phi_n^{(1)} + 16\phi_{n+1}^{(1)} + 7\phi_{n+2}^{(1)}}{30} + \Delta T^2 \frac{\phi_n^{(2)} - \phi_{n+2}^{(2)}}{60}, \\ \phi_{n+1}^{(0)} &= \phi_n^{(0)} + \Delta T \frac{303\phi_n^{(1)} + 284\phi_{n+1}^{(1)} + 33\phi_{n+2}^{(1)}}{1440} + \Delta T^2 \frac{39\phi_n^{(2)} - 120\phi_{n+1}^{(2)} - 3\phi_{n+2}^{(2)}}{2800}, \end{aligned}$$

given by the extended Butcher tableau

0	0	0	0	0	0	0
$\frac{1}{2}$	$\frac{303}{1440}$	$\frac{284}{1440}$	$\frac{33}{1440}$	$\frac{39}{2880}$	$-\frac{120}{2880}$	$-\frac{3}{2880}$
1	$\frac{7}{30}$	$\frac{16}{30}$	$\frac{7}{30}$	$\frac{1}{60}$	0	$-\frac{1}{60}$
	$\frac{7}{30}$	$\frac{16}{30}$	$\frac{7}{30}$	$\frac{1}{60}$	0	$-\frac{1}{60}$

3.2.2 A non-TDRK scheme

The structural equations provide an MDRK only when the maximum number of physical equations (K) is used. For example, taking $K = 2$ and $R = 1$, the following scheme is composed of one physical equation and two structural equations

$$\begin{aligned} 0 &= 12(\phi_{n+1}^{(0)} - \phi_n^{(0)}) - \Delta t 6(\phi_{n+1}^{(1)} + \phi_n^{(1)}) + \Delta t^2 (\phi_{n+1}^{(2)} - \phi_n^{(2)}), \\ 0 &= 6(\phi_{n+1}^{(0)} - \phi_n^{(0)}) - \Delta t (4\phi_{n+1}^{(1)} + 2D_n) + \Delta t^2 \phi_{n+1}^{(2)}, \\ \phi_{n+1}^{(1)} &= f(\phi_{n+1}^{(0)}, t_{n+1}), \end{aligned}$$

has no equivalent formulation in the TDRK framework. The second physical equation has been replaced by a second structural equation; therefore, the scheme does not require the analytical expression of the derivative of $f(z, t)$.

The above scheme is third-order accurate for the function and derivative, whereas the approximation of the second derivative is second-order accurate. Moreover, stability is no longer unconditional. A restrictive condition on the time step Δt is required to guarantee the convergence of the approximation, as shown in reference [55]. In general, configurations involving fewer physical equations and more structural equations are possible. Nevertheless, a loss of the unconditional stability property, together with lower accuracy, leads us to disregard such schemes in the present work.

3.2.3 Multi-derivative collocation method

Recently, a two-derivative collocation method [44] has been proposed. In fact, the authors also introduce the additional term ϕ_{n-1} , which gives the method a two-step character. The main idea consists of introducing a Hermite interpolation polynomial π such that

$$\begin{aligned} \pi(t_n) &= \phi_n \\ \pi'(t_n + c_i \Delta t) &= \mathbf{df}^{(0)}[\pi(t_n + c_i \Delta t), t_n + c_i \Delta t], i = 1, \dots, m, \\ \pi''(t_n + c_i \Delta t) &= \mathbf{df}^{(0)}[\pi(t_n + c_i \Delta t), \pi'(t_n + c_i \Delta t), t_n + c_i \Delta t], i = 1, \dots, m, \end{aligned}$$

where c_i are the collocation parameters. Once again, we take $\phi_{n+1} = \pi(t_n + \Delta t)$ as an approximation at time t_{n+1} , and the authors prove A-stability. To the best of our knowledge, and owing to the very limited literature on this technique, we do not know whether the method is equivalent to a two-derivative RK scheme.

3.3 Obrechhoff-type method

Obrechhoff methods [30] compute approximations of the function and its derivative at time t_{n+1} using information from the function and its derivatives at time t_n . An extension was proposed in [3, 9], using two time steps and two derivatives; see also the Enright schemes [13, 14]. The two-derivative multistep Obrechhoff-type method was studied in [84, 39, 40]. More generally, multistep multi-derivative methods compute the next time step using backlog data from the previous R nodes [85, 86, 87, 88]; see also [89, 90, 91] for hybrid versions designed to enhance the stability of the method.

Using our notations, the generalized Obrechhoff method reads

$$\sum_{k=0}^K \alpha_k \phi_{n+1}^{(k)} = \sum_{r=0}^R \sum_{k=0}^K \beta_{kr} \phi_{n-r}^{(k)},$$

with

$$\phi_{n+1}^{(k)} = \mathbf{df}^{(k-1)}[\Phi_{n+1}^{(0)}, \dots, \Phi_{n+1}^{(k-1)}, t_{n+1}], \quad k = 1, \dots, K.$$

The scheme is implicit with respect to the current time step $n + 1$, but explicit in terms of the previous time steps. The improvement in order is achieved by exploiting the information computed at the R earlier times. This class of schemes differs from the structural method proposed in the present work, since the resulting system is fully implicit with respect to all R time steps, except for the time t_n . We would like to emphasise that the structural equations technique can also be applied to determine the coefficients of Obrechhoff methods. Indeed, considering the functional

$$E(\alpha\beta; \pi) = \sum_{k=0}^K \alpha_k \phi_{n+1}^{(k)} - \sum_{r=0}^R \sum_{k=0}^K \beta_{r,k} (\Delta t)^k \phi_{n-r}^{(k)},$$

the conditions $E(\alpha\beta; \pi_\ell) = 0$ with $\pi_\ell(t) = \left(\frac{t - t_n}{\Delta t}\right)^\ell$, $\ell = 0, \dots, p$ provide the $p + 1$ linear system

$$A_1 \alpha - \sum_{r=0}^R B_r \beta_r = 0, \quad \text{with } \beta_r = (\beta_{r,k})_{k=0}^{K=K}$$

with A_1 a $p \times (K + 1)$ matrix while B_r are $p \times (K + 1)$ matrices given by the polynomial derivative at t_{n-r} . If $p < (R + 1)(K + 1)$ and the system has maximal rank, then any vector in the kernel provides an Obrechhoff relation. Since there are only $K + 1$ unknowns, namely the solution and its derivatives at time t_{n+1} , and K physical equations at time t_{n+1} , a single vector from the kernel is sufficient to close the system.

We note that our approach avoids the need to determine explicit relations. One only has to construct the matrices and extract a vector from the kernel. Moreover, the vector can be chosen so as to ensure the best stability properties.

4 Numerical solver

An efficient iterative algorithm is proposed in this section for solving the(non-)linear system of equations arising from the structural schemes $\text{Skm}[1, R]$, $\text{Skm}[2, R]$, and $\text{Skm}[3, R]$.

4.1 Algorithm for the structural schemes

Structural scheme $\text{Skm}[1, R]$. The compact matrix form of the structural scheme $\text{Skm}[1, R]$ reads

$$\begin{cases} \Phi_n^{(1)} = \mathbf{df}^{(0)}[\Phi_n^{(0)}, T_n], \\ \mathbf{A}^{(0)} \Phi_n^{(0)} + \Delta t \mathbf{A}^{(1)} \Phi_n^{(1)} + \hat{\mathbf{A}}_n \Psi_n = 0, \end{cases}$$

for which a fixed-point iterative sequence of $2 \times R$ matrices is introduced, given as

$$\Phi_n[\ell] = \left(\phi_{n+r}^{(k)}[\ell] \right)_{k,r}, \quad k = 0, 1, \quad r \in \llbracket 1, R \rrbracket,$$

where ℓ is the iteration number. The algorithm is as follows.

- **Iterative procedure:** Assume that approximations of the function and derivative in vector $\Phi_n[\ell]$ at iteration ℓ are known. The iterative process then proceeds in two main stages:

- ① Compute approximations of the derivative in vector $\Phi_n^{(1)}[\ell + 1]$ using the first physical equation $\text{PE}_1(r)$ at time steps t_{n+r} , $r \in \llbracket 1, R \rrbracket$, that is

$$\Phi_n^{(1)}[\ell + 1] = \text{df}^{(0)}[\Phi_n^{(0)}[\ell], T_n].$$

- ② Compute approximations of the function in vector $\Phi_n^{(0)}[\ell + 1]$ by solving a $R \times R$ linear system in the form

$$\mathbf{A}^{(0)}\Phi_n^{(0)}[\ell + 1] = -\Delta t \mathbf{A}^{(1)}\Phi_n^{(1)}[\ell + 1] - \widehat{\mathbf{A}}_n \Psi_n$$

Note that matrix $\mathbf{A}^{(0)}$ must be non-singular to determine vector $\Phi_n^{(0)}[\ell + 1]$.

The iterative procedure stops when the residual between successive approximations is below a specified tolerance.

- **Initialisation procedure:** At the beginning of the iterative procedure, vector Ψ_n is known but an approximation to vectors $\Psi_{n+r}[0]$, $r \in \llbracket 1, R \rrbracket$, is desirable. To this end, a first-order accurate Taylor expansion can be employed, yielding

$$\phi_{n+r}^{(0)}[0] = \phi_{n+r-1}^{(0)}[0] + \Delta t \phi_{n+r-1}^{(1)}[0], \quad \phi_{n+r}^{(1)}[0] = \text{df}^{(1)}[\phi_{n+r}^{(0)}[0], t_{n+r}], \quad r \in \llbracket 1, R \rrbracket,$$

where $\phi_n^{(0)}[0] = \phi_n^{(0)}$ and $\phi_n^{(1)}[0] = \phi_n^{(1)}$.

Remark. A good initialisation is essential for reducing the number of iterations and ensuring that the initial point lies within the convergence basin of the fixed-point iteration. ■

Structural scheme $\text{Skm}[2, R]$. The procedure for the structural scheme $\text{Skm}[2, R]$ is analogous, introducing a fixed-point iterative sequence of $3 \times R$ matrices, given as

$$\Phi_n[\ell] = \left(\phi_{n+r}^{(k)}[\ell] \right)_{k,r}, \quad k = 0, 1, 2, \quad r \in \llbracket 1, R \rrbracket.$$

The algorithm is as follows.

- **Iterative procedure:** Assume that approximations of the function and derivatives in vector $\Phi_n[\ell]$ at iteration ℓ are known. The iterative process then proceeds in two main stages:

- ① Compute approximations of the derivatives in vectors $\Phi_n^{(1)}[\ell + 1]$ and $\Phi_n^{(2)}[\ell + 1]$ using the first and second physical equations, respectively, at time steps t_{n+r} , $r \in \llbracket 1, R \rrbracket$, that is

$$\begin{aligned} \Phi_n^{(1)}[\ell + 1] &= \text{df}^{(0)}[\Phi_n^{(0)}[\ell], T_n], \\ \Phi_n^{(2)}[\ell + 1] &= \text{df}^{(1)}[\Phi_n^{(0)}[\ell], \Phi_n^{(1)}[\ell + 1], T_n]. \end{aligned}$$

- ② Compute approximations of the function in vector $\Phi_n^{(0)}[\ell + 1]$ by solving a $R \times R$ linear system in the form

$$\mathbf{A}^{(0)}\Phi_n^{(0)}[\ell + 1] = -\Delta t \mathbf{A}^{(1)}\Phi_n^{(1)}[\ell + 1] - \Delta t^2 \mathbf{A}^{(2)}\Phi_n^{(2)}[\ell + 1] - \widehat{\mathbf{A}}_n \Psi_n$$

assuming that matrix $\mathbf{A}^{(0)}$ is non-singular.

- **Initialisation procedure:** An approximation to vectors $\Psi_{n+r}[0]$, $r \in \llbracket 1, R \rrbracket$, is prescribed from a second-order accurate Taylor expansion, given as

$$\begin{aligned} \phi_{n+r}^{(0)}[0] &= \phi_{n+r-1}^{(0)}[0] + \Delta t \phi_{n+r-1}^{(1)}[0] + \frac{\Delta t^2}{2} \phi_{n+r-1}^{(2)}[0], \\ \phi_{n+r}^{(1)}[0] &= \text{df}^{(0)}[\phi_{n+r}^{(0)}[0], t_{n+r}], \\ \phi_{n+r}^{(2)}[0] &= \text{df}^{(1)}[\phi_{n+r}^{(0)}[0], \phi_{n+r}^{(1)}[0], t_{n+r}], \end{aligned}$$

where $\phi_n^{(0)}[0] = \phi_n^{(0)}$, $\phi_n^{(1)}[0] = \phi_n^{(1)}$, $\phi_n^{(2)}[0] = \phi_n^{(2)}$.

Structural scheme $\text{Skm}[3, R]$. A fixed-point iterative sequence of $4 \times R$ matrices is introduced for the structural scheme $\text{Skm}[3, R]$, given as

$$\Phi_n[\ell] = \left(\phi_{n+r}^{(k)}[\ell] \right)_{k,r}, \quad k = 0, \dots, 3, \quad r \in \llbracket 1, R \rrbracket.$$

The algorithm is as follows.

- **Iterative procedure:** Assume that approximations of the function and derivatives in vector $\Phi_n[\ell]$ at iteration ℓ are known. The iterative process then proceeds in two main stages:

- ① Compute approximations of the derivatives in vectors $\Phi_n^{(1)}[\ell+1]$, $\Phi_n^{(2)}[\ell+1]$, and $\Phi_n^{(3)}[\ell+1]$ using the first, second, and third physical equations, respectively, at time steps t_{n+r} , $r \in \llbracket 1, R \rrbracket$, that is

$$\begin{aligned}\Phi_n^{(1)}[\ell+1] &= \mathbf{df}^{(0)}[\Phi_n^{(0)}[\ell], T_n], \\ \Phi_n^{(2)}[\ell+1] &= \mathbf{df}^{(1)}[\Phi_n^{(0)}[\ell], \Phi_n^{(1)}[\ell+1], T_n], \\ \Phi_n^{(3)}[\ell+1] &= \mathbf{df}^{(2)}[\Phi_n^{(0)}[\ell], \Phi_n^{(1)}[\ell+1], \Phi_n^{(2)}[\ell+1], T_n].\end{aligned}$$

- ② Compute approximations of the function in vector $\Phi_n^{(0)}[\ell+1]$ by solving a $R \times R$ linear system in the form

$$\mathbf{A}^{(0)}\Phi_n^{(0)}[\ell+1] = -\Delta t \mathbf{A}^{(1)}\Phi_n^{(1)}[\ell+1] - \Delta t^2 \mathbf{A}^{(2)}\Phi_n^{(2)}[\ell+1] - \Delta t^3 \mathbf{A}^{(3)}\Phi_n^{(3)}[\ell+1] - \hat{\mathbf{A}}_n \Psi_n.$$

- **Initialisation procedure:** An approximation to vectors $\Psi_{n+r}[0]$, $r \in \llbracket 1, R \rrbracket$, is prescribed from a third-order accurate Taylor expansion, given as

$$\begin{aligned}\phi_{n+r}^{(0)}[0] &= \phi_{n+r-1}^{(0)}[0] + \Delta t \phi_{n+r-1}^{(1)}[0] + \frac{\Delta t^2}{2} \phi_{n+r-1}^{(2)}[0] + \frac{\Delta t^3}{6} \phi_{n+r-1}^{(3)}[0], \\ \phi_{n+r}^{(1)}[0] &= \mathbf{df}^{(0)}[\phi_{n+r}^{(0)}[0], t_{n+r}], \\ \phi_{n+r}^{(2)}[0] &= \mathbf{df}^{(1)}[\phi_{n+r}^{(0)}[0], \phi_{n+r}^{(1)}[0], t_{n+r}], \\ \phi_{n+r}^{(3)}[0] &= \mathbf{df}^{(2)}[\phi_{n+r}^{(0)}[0], \phi_{n+r}^{(1)}[0], \phi_{n+r}^{(2)}[0], t_{n+r}],\end{aligned}$$

where $\phi_n^{(k)}[0] = \phi_n^{(k)}$, $k \in \llbracket 0, 3 \rrbracket$.

4.2 Complexity

The $\text{Skm}[K, R]$ method computes successive R -point blocks of approximations, with a total number of points $N = LR$, where L denotes the number of blocks. For each block, an iterative procedure is required to obtain satisfactory approximations. Each step of the iterative procedure is decomposed into two parts: the derivatives' update

$$\Phi_n^{(0)}[\ell] \rightarrow \Phi_n^{(1)}[\ell+1] = \mathbf{df}^{(0)}[\Phi_n^{(0)}[\ell]] \rightarrow \cdots \rightarrow \Phi_n^{(K)}[\ell+1] = \mathbf{df}^{(K-1)}[\Phi_n^{(0)}[\ell], \dots, \Phi_n^{(K-1)}[\ell]],$$

and the solution update

$$\Phi_n^{(0)}[\ell+1] = -\sum_{k=1}^K (\Delta t)^k \mathbf{B}^{(k)} \Phi_n^{(k)}[\ell+1] - \hat{\mathbf{B}}_n \Psi_n,$$

with $\mathbf{B}^{(k)} = [\mathbf{A}^{(0)}]^{-1} \mathbf{A}^{(k)}$ and $\hat{\mathbf{B}}_n \Psi_n = [\mathbf{A}^{(0)}]^{-1} \hat{\mathbf{A}}_n \Psi_n$. Of course, the $R \times R$ matrices $\mathbf{B}^{(k)}$ are precomputed once and thus the system only involves matrix vector product.

To assess the complexity of the method, two major difficulties arise. First, the cost of evaluating $\Phi_n^{(1)}[\ell+1]$, corresponding to the first derivative, may differ substantially from the cost of evaluating $\Phi_n^{(K)}[\ell+1]$, corresponding to the highest derivative. Second, the number of iterations depends on the contraction properties of the fixed-point iteration and on the user-prescribed tolerance used to stop the loop.

- The computation of higher derivatives involves processing more information. Nevertheless, the corresponding routines are highly vectorised, since most arithmetic operations are not sequential. Consequently, we assume that the running time of the calls used to compute each derivative is approximately $R\tau_d$, where τ_d denotes the computational time associated with each point t_{n+r} .
- The linear combination that provides $\Phi_n^{(0)}[\ell+1]$ is also highly parallelisable, or vectorisable, since it only involves independent matrix-vector products and summations. Consequently, we assume that the running-time complexity required to update the vector is $RK\tau_m$, where τ_m denotes the time required to compute a row-column product.
- The total time required for one cycle is therefore $RK\tau_m + R(K-1)\tau_d$. For simple problems, τ_m and τ_d are of the same order, leading to a complexity of $O(2RK\tau_d)$. When the nonlinear contribution is more complex, for instance when it involves trigonometric or exponential functions, we have $\tau_d \gg \tau_m$, and the complexity becomes $O(RK\tau_d)$.

5 Post-processing

Previously, we focused on fully implicit structural equations, where all derivatives up to order K are coupled with the physical equations. However, using explicit structural equations may help compute higher derivatives beyond those obtained directly from the structural scheme. For example, the scheme $\text{Skm}[3,3]$ yields approximations up to the third derivative, whereas post-processing could approximate fourth- or fifth-order derivatives. In general, the structural scheme $\text{Skm}[K,R]$ produces $\phi_N^{(k)} \approx \phi^{(k)}(t_N)$ for $k \in \llbracket 0, K \rrbracket$ at the final time, with convergence order $d = K(R+1)$. Building on these approximations, we present a method to approximate higher derivatives $\phi_N^{(p)}$ for $p > K$.

One basic strategy is to derive additional physical equations, PE_{K+1} to PE_p . However, this approach often becomes impractical, particularly if the partial derivatives are complex or the physical equation uses tabulated functions. As a practical alternative, our goal is to provide an approximation $\phi_N^{(p)} \approx \phi^{(p)}(t_N)$ with the best precision using an explicit structural equation (referred to as the post-processing structural equation) that uses the approximations $\phi_n^{(k)}$ computed from the structural scheme. The explicit relation takes the form

$$\phi_N^{(p)} = \sum_{i=0}^{I_p} \sum_{k=0}^K a_{ik}^p \Delta t^k \phi_{N-i}^{(k)},$$

where I_p is the number of backward time steps required to exactly approximate any polynomial up to degree d . To determine the $(I_p + 1) \times (K + 1)$ matrix $\mathbf{a}^p = (a_{ik}^p)_{\substack{i=0,\dots,I_p \\ k=0,\dots,K}}$, we introduce the functional

$$E_p(\mathbf{a}^p; \pi) = \sum_{i=0}^{I_p} \sum_{k=0}^K a_{ik}^p \Delta t^k \pi^{(k)}(t_{N-i}) - \pi^{(p)}(t_N),$$

for any function $\pi = \pi(t)$, regular enough.

We reshape the matrix $\mathbf{a}^p$ into a single-index entry vector $\mathbf{a}^p$ (we use the same notation for simplicity) with one-to-one mapping $(i, k) \rightarrow \ell = \ell(i, k) = k(I_p + 1) + i + 1$ from $\llbracket 0, K \rrbracket \times \llbracket 0, I_p \rrbracket$ onto $\llbracket 1, M_p \rrbracket$, with $M_p = (K + 1)(I_p + 1)$ the number of entries from $\mathbf{a}^p$, and the functional can be rewritten as

$$E_p(\mathbf{a}^p; \pi) = \sum_{i=0}^{I_p} \sum_{k=0}^K a_\ell^p \Delta t^k \pi^{(k)}(t_{N-i}) - \pi^{(p)}(t_N), \quad \ell = \ell(i, k).$$

To ensure accuracy, we require that the functional exactly satisfies $E(\mathbf{a}; \pi_m) = 0$ for polynomial functions up to a chosen degree. For this, we consider the following set of polynomial functions

$$\pi_m(t; t_N) = \left(\frac{t - t_N}{\Delta t} \right)^{m-1}, \quad m \in \llbracket 1, \overline{M}_p \rrbracket, \quad \overline{M}_p = p + d.$$

Note that the number of coefficients is M_p and we need to impose the condition $\overline{M}_p \leq M_p$ such that the linear system has at least one non-trivial solution. We then deduce a sufficient condition for the number of backward time steps I_p as

$$p + K(R + 1) - K - 1 \leq I_p(K + 1) \implies I_p \geq \frac{KR + p - 1}{K + 1}.$$

Remark. For $K = 1$ and $R = 2$, we have $I_p \geq (1 + p)/2$, while $K = 3$ and $R = 4$ requires $I_p \geq (11 + p)/4$. ■

To compute the coefficients in $\mathbf{a}^p$, we solve an under-determined linear system $\mathcal{M}^p \mathbf{a}^p = \mathbf{b}^p$. Here, $\mathcal{M}^p$ is a $\overline{M}_p \times M_p$ matrix. The general solution consists of a particular solution together with any vector in the kernel of $\mathcal{M}^p$. For practical computation, the least-squares method is effective, since $\mathcal{M}^p$ has maximal rank.

The entries of matrix $\mathcal{M}^p$ given by the k -th derivative of the polynomial functions $\pi_m(t; t_N)$, $m \in \llbracket 0, \overline{M}_p \rrbracket$, read

$$\pi_m^{(k)}(t; t_N) = \frac{(m-1) \cdots (m-k)}{\Delta t^k} \left(\frac{t - t_N}{\Delta t} \right)^{m-1-k}, \quad k < m,$$

and

$$\pi_m^{(k)}(t; t_N) = 0, \quad k \geq m.$$

After calculations, for $k < m$ and $i \in \llbracket 0, I_p \rrbracket$, yields

$$\pi_m^{(k)}(t_{N-i}; t_N) = \frac{(m-1) \cdots (m-k)}{\Delta t^k} (-i)^{m-1-k}.$$

Then, the entries of matrix $\mathcal{M}^p$ are given as

$$\mathcal{M}^p[m, \ell] = C_\ell^m (-i)^{m-1-\ell}, \quad C_\ell^m = \begin{cases} \frac{m!}{(m-\ell)!} & \text{if } m > \ell, \\ 0 & \text{otherwise,} \end{cases} \quad \ell = \ell(i, k),$$

for $m \in \llbracket 1, \overline{M}_p \rrbracket$, $\ell \in \llbracket 1, M_p \rrbracket$. Several cases of post-processing analytical formulas are given in B.

Remark. For larger values of K and p , or for non-uniform time grids, the matrix $\mathcal{M}^p$ and right-hand side vector $\mathbf{b}^p$ are still computed numerically. This involves enforcing the condition $E_p(\mathbf{a}^p; \pi) = 0$ for $m \in \llbracket 0, \overline{M}_p \rrbracket$. In such cases, applying the least-squares method provides a solution and yields the post-processing structural equation.

We propose a simple benchmark to assess approximation errors, using the function $\phi(t) = \exp(t)$ as an example. We compute $\phi_N^{(p)}$ with the explicit structural equations, where $\phi_{N-i}^{(k)}$ are substituted with the exact values. The errors at the final time $\phi^{(p)}(T) - \phi_N^{(p)}$, $T = 1$, together with the convergence orders are provided in Tables 1, 2, and 3. Several examples of p -th derivative approximation using post-processing structural equations are assessed with different parameter values K and R .

Table 1: Consistency errors with $K = 1$.

Ip	N	$\phi^{(2)}$		$\phi^{(3)}$		$\phi^{(4)}$	
		err	ord	err	ord	err	ord
1	60	6.25E-05	—	2.25E-02	—	2.72E+00	—
	120	1.57E-05	2.0	1.13E-02	1.0	2.72E+00	0.0
	240	3.93E-06	2.0	5.66E-03	1.0	2.72E+00	0.0
2	60	2.30E-09	—	1.24E-06	—	3.23E-04	—
	120	1.45E-10	4.0	1.56E-07	3.0	8.13E-05	2.0
	240	9.07E-12	4.0	1.96E-08	3.0	2.04E-05	2.0
3	60	1.02E-13	—	6.72E-11	—	2.36E-08	—
	120	1.61E-15	6.0	2.12E-12	5.0	1.49E-09	4.0
	240	2.53E-17	6.0	6.67E-14	5.0	9.36E-11	4.0
4	60	4.98E-18	—	3.74E-15	—	1.56E-12	—
	120	1.98E-20	8.0	2.97E-17	7.0	2.48E-14	6.0
	240	7.78E-23	8.0	2.33E-19	7.0	3.90E-16	6.0
5	60	2.60E-22	—	2.14E-19	—	1.01E-16	—
	120	2.59E-25	10.0	4.26E-22	9.0	4.01E-19	8.0
	240	2.55E-28	10.0	8.40E-25	9.0	1.58E-21	8.0

Table 2: Consistency errors with $K = 2$.

Ip	N	$\phi^{(3)}$		$\phi^{(4)}$	
		err	ord	err	ord
1	60	1.04E-07	—	7.50E-05	—
	120	1.31E-08	3.0	1.88E-05	2.0
	240	1.64E-09	3.0	4.71E-06	2.0
2	60	7.59E-15	—	8.20E-12	—
	120	1.20E-16	6.0	2.58E-13	5.0
	240	1.87E-18	6.0	8.10E-15	5.0
3	60	7.13E-22	—	9.42E-19	—
	120	1.41E-24	9.0	3.72E-21	8.0
	240	2.77E-27	9.0	1.46E-23	8.0
4	60	7.68E-29	—	1.15E-25	—
	120	1.90E-32	12.0	5.71E-29	11.0
	240	4.68E-36	12.0	2.81E-32	11.0
5	60	9.00E-36	—	1.48E-32	—
	120	2.80E-40	15.0	9.21E-37	14.0
	240	8.63E-45	15.0	5.68E-41	14.0

Table 3: Consistency errors with $K = 3$.

Ip	N	$\phi^{(4)}$	
		err	ord
1	60	1.24E-10	—
	120	7.77E-12	4.0
	240	4.87E-13	4.0
2	60	1.28E-20	—
	120	5.03E-23	8.0
	240	1.97E-25	8.0
3	60	1.81E-30	—
	120	4.48E-34	12.0
	240	1.10E-37	12.0
4	60	3.06E-40	—
	120	4.74E-45	16.0
	240	7.28E-50	16.0
5	60	5.73E-50	—
	120	5.57E-56	20.0
	240	5.37E-62	20.0

The structural scheme $\text{Skm}[K, R]$ has order $d = K(R + 1)$, so $\phi_N^{(K)}$ approximates $\phi^{(K)}(t_N)$ to convergence order $O(\Delta t^d)$. However, approximating higher derivatives results in a loss of convergence order. Specifically, $\phi_N^{(K+1)}$ has order $d - 1$, while $\phi_N^{(p)}$ for $p < d + K$ has order $d - (p - K)$.

Remark. Convergence cannot be achieved for $\phi_N^{(p)}$ if $d - (p - K) \leq 0$. Therefore, the explicit structural equations require $p < K(R + 1) + K = K(R + 2)$, such that convergence is controlled by the two parameters K and R along with the mandatory number of backward time steps $I_p \geq I_{p, \min} = \lceil (KR + p - 1)/(K + 1) \rceil$. ■

6 Benchmarks

A comprehensive benchmark is presented to assess the relevant properties of the structural schemes, namely, numerical accuracy, linear stability, spectral resolution, computational efficiency, and the ability to handle stiff problems. To assess the numerical accuracy of the structural schemes, the approximation error at time $t_N = N\Delta t = T$ is defined as

$$e_N = \left| \phi_N^{(k)} - \phi^{(k)}(T) \right|,$$

where $\phi_N^{(k)}$ is the numerical approximation of $\phi^{(k)}(t_N)$. Then, the convergence order for the approximation error of two numerical solutions $\phi_{N_1}^{(k)}$ and $\phi_{N_2}^{(k)}$ computed on two time grids of sizes N_1 and N_2 , respectively, with $N_1 \neq N_2$, is given as

$$O(N_1, N_2) = \frac{|\log(e_{N_1}/e_{N_2})|}{|\log(N_1/N_2)|}.$$

Remark. We would like to mention that the structural method has also been tested on Hamiltonian problems covering a broad range of situations, including the pendulum, the Euler system, the three-body problem, and the motion of a particle in a variable electromagnetic field [80].

Remark. A library dedicated to the numerical solution of ODEs using the structural method is available through an open-access GitHub repository [92]. The codes were developed in Julia and C++, using the qd library to handle quad- and octuple-precision arithmetic.

6.1 Linear scalar ODE

The first sanity check benchmark concerns the linear ODE $\phi'(t) = -\lambda\phi(t)$ in $(0, 1]$, with the initial condition $\phi(0) = 1$, whose exact solution is $\phi(t) = \exp(-\lambda t)$, to assess several relevant issues that arise in the calculation of the approximations:

1. Verify whether the optimal accuracy, as predicted from the previous analysis, is obtained for this simple example. In particular, when dealing with an asymptotic convergence $C\Delta t^m$, the magnitude of the multiplicative constant C for different structural schemes of the same convergence order is analysed. In that regard, the convergence orders of the structural schemes $\text{Skm}[K, R]$ for the approximation of the solution and derivatives $\phi^{(k)}(t)$, $k \leq K$, at the final time, are reported in black, whereas an approximation of the derivatives $\phi^{(k)}(t)$ with $k > K$, obtained from post-processing structural equations, are reported in blue. Several possible combinations of the number of backward time steps I_p are analysed.
2. Assess the impact of the tolerance parameter ε for the fixed-point iterative process on the accuracy of the structural schemes. Indeed, if ε is not sufficiently small, some accuracy loss is expected since the iterative process stops before reaching the optimal approximation. On the contrary, if ε is too small, there is an unnecessary computational effort from the additional fixed-point iterations performed after the optimal accuracy has already been reached.
3. Assess the influence of the number of variables and the convergence order through the parameters K and R on the memory load. In particular, if the structural schemes are used for non-stationary PDE problems or large differential systems, the computations might become mostly memory-bound.
4. Confirm whether intermediate time grid point approximations in the last block achieve the same optimal convergence order as the final time point, for which the approximation errors $|\phi_{N-r}^{(k)} - \phi^{(k)}(T - r\Delta t)|$ for $r \in \llbracket 1, R - 1 \rrbracket$, are computed.

6.1.1 Convergence order

Considering $\lambda = 1$, we check the convergence order of the structural schemes $\text{Skm}[1, R]$ with $R \in \llbracket 1, 5 \rrbracket$. The errors and convergence orders are reported in Table 4, where the second, third, and fourth derivatives were obtained from the post-processing structural equations using $I_p = I_{p,\min}$. Super-convergence is observed for $R = 2, 4$ (where the third and fifth orders were expected, respectively), although the post-calculated derivatives present the nominal convergence order, which decreases as the derivative order increases, as expected.

Remark. Post-processing has been performed with higher values for I_p , namely $I_p = I_{p,\min} + 1$, $I_p = I_{p,\min} + 2$, and $I_p = I_{p,\min} + 3$, which are not reported since comparable convergence orders to the case with $I_p = I_{p,\min}$ were obtained. Nevertheless, it is observed that the approximation errors increase with I_p due to larger stencils. Thus, the optimal choice remains $I_p = I_{p,\min}$. ■

Table 4: Errors and convergence orders for the linear scalar ODE benchmark with $K = 1$, $I_p = I_{p,\min}$, and $\varepsilon = 10^{-20}$.

R	N	ϕ		$\phi^{(1)}$		$\phi^{(2)}$		$\phi^{(3)}$		$\phi^{(4)}$	
		err	ord	err	ord	err	ord	err	ord	err	ord
1	60	8.52E-06	—	8.52E-06	—	3.08E-03	—	2.33E+00	—	3.36E+02	—
	120	2.13E-06	2.0	2.13E-06	2.0	1.54E-03	1.0	2.32E+00	0.0	6.67E+02	—
	240	5.32E-07	2.0	5.32E-07	2.0	7.67E-04	1.0	2.31E+00	0.0	1.33E+03	—
2	60	6.31E-10	—	6.31E-10	—	3.46E-05	—	6.96E-03	—	3.47E+00	—
	120	3.94E-11	4.0	3.94E-11	4.0	8.59E-06	2.0	3.50E-03	1.0	3.47E+00	—
	240	2.46E-12	4.0	2.46E-12	4.0	2.14E-06	2.0	1.76E-03	1.0	3.47E+00	—
3	60	3.55E-10	—	3.55E-10	—	4.36E-07	—	9.56E-05	—	9.33E-03	—
	120	2.22E-11	4.0	2.22E-11	4.0	5.39E-08	3.0	2.37E-05	2.0	4.63E-03	1.0
	240	1.39E-12	4.0	1.39E-12	4.0	6.69E-09	3.0	5.88E-06	2.0	2.31E-03	1.0
4	60	1.00E-13	—	1.00E-13	—	5.87E-09	—	1.46E-06	—	1.83E-04	—
	120	1.56E-15	6.0	1.56E-15	6.0	3.61E-10	4.0	1.80E-07	3.0	4.53E-05	2.0
	240	2.44E-17	6.0	2.44E-17	6.0	2.24E-11	4.0	2.23E-08	3.0	1.12E-05	2.0
5	60	3.59E-14	—	3.59E-14	—	8.22E-11	—	2.24E-08	—	3.31E-06	—
	120	5.60E-16	6.0	5.60E-16	6.0	2.52E-12	5.0	1.38E-09	4.0	4.06E-07	3.0
	240	8.75E-18	6.0	8.75E-18	6.0	7.78E-14	5.0	8.52E-11	4.0	5.03E-08	3.0

6.1.2 Sensitivity to the tolerance ε

The solution of a block of size R is obtained from an iterative procedure described in Section 4 and controlled by a tolerance ε . We denote by $\bar{\kappa}$ the average number of iterations to solve an R -block. Therefore, for a structural scheme $\text{Skm}[\kappa, R]$, the physical equations are evaluated $K R \bar{\kappa}$ times to solve a block, thus $K N \bar{\kappa}$ is the total number of evaluations for N steps. We then denote by $\bar{\tau} = K \bar{\kappa}$ the average number of evaluations per time step.

To assess the computational effort, the values of $\bar{\kappa}$ and $\bar{\tau}$ for two values of ε are reported in Tables 5, 6, and 7. The first table concerns structural schemes with convergence orders of 2, 4, and 6, where ε is chosen based on the approximation errors. The second table concerns the convergence orders of 8, 10, and 12, for which smaller values of ε are used. The last table concerns the highest convergence orders with a substantial reduction in the values of ε . As expected, the number of iterations increases when ε decreases. Quantifying the additional iterations is not easy, as there is significant variation between cases, ranging from half to twice the number of iterations, even when ε varies by 10 or 15 orders of magnitude. In conclusion, the computational effort is not substantially affected by the strong reduction in the tolerance criterion, and for the remainder of the section, we use, by default, $\varepsilon = 10^{-20}$ for $K = 1$, $\varepsilon = 10^{-40}$ for $K = 2$, and $\varepsilon = 10^{-60}$ for $K = 3$.

Table 5: Average number of fixed-point iterations for the linear scalar ODE benchmark with $N = 60$ for structural schemes with convergence orders of 2, 4, and 6.

K	1						2		3
R (order)	1 (2)	2 (4)	3 (4)	4 (6)	5 (6)		1 (4)	2 (6)	1 (6)
$\bar{\kappa}$ with $\varepsilon=1\text{E-}20$	9.00	5.00	3.33	2.50	2.20		8.00	4.50	7.00
$\bar{\tau}$ with $\varepsilon=1\text{E-}20$	9.00	5.00	3.33	2.50	2.20		16.00	4.50	21.00
$\bar{\kappa}$ with $\varepsilon=1\text{E-}30$	14.00	7.50	5.00	4.00	3.20		13.00	7.00	12.00
$\bar{\tau}$ with $\varepsilon=1\text{E-}30$	14.00	7.50	5.00	4.00	3.20		26.00	14.00	36.00

Table 6: Average number of fixed-point iterations for the linear scalar ODE benchmark with $N = 60$ for structural schemes with convergence orders of 8, 10, and 12.

K	2			3		4	
R (order)	3 (8)	4 (10)	5 (12)	2 (10)	3 (12)	1 (10)	2 (12)
$\bar{\kappa}$ with $\varepsilon=1\text{E-}25$	4.17	3.47	3.00	4.97	3.33	8.00	4.50
$\bar{\tau}$ with $\varepsilon=1\text{E-}25$	8.34	6.94	6.00	14.91	9.99	32.00	18.00
$\bar{\kappa}$ with $\varepsilon=1\text{E-}40$	7.00	5.75	5.00	8.50	6.00	15.00	8.45
$\bar{\tau}$ with $\varepsilon=1\text{E-}40$	14.00	11.50	10.00	25.50	18.00	60.00	33.80

Table 7: Average number of fixed-point iterations for the linear scalar ODE benchmark with $N = 60$ for structural schemes with convergence orders of 16, 18, 20, and 24.

K	3		4		
R (order)	4 (16)	5 (18)	3 (16)	4 (20)	5 (24)
$\bar{\kappa}$ with $\varepsilon=1\text{E-}40$	4.93	4.40	6.12	5.35	5.40
$\bar{\tau}$ with $\varepsilon=1\text{E-}40$	14.79	13.20	24.48	21.40	21.60
$\bar{\kappa}$ with $\varepsilon=1\text{E-}60$	7.75	6.83	10.00	8.75	8.98
$\bar{\tau}$ with $\varepsilon=1\text{E-}60$	23.25	20.49	40.00	35.00	35.92

6.1.3 Memory load

The number of variables to solve in a block of the structural scheme $\text{Skm}[K, R]$ is $M = (R + 1)(K + 1)$, while the convergence order is $O = (R + 1)K$, from which we deduce the simple relation $M = (K + 1)/K O$ between the memory storage and the convergence order. Hence, for a vector system of dimension L , for instance, a finite difference semi-discretisation in space with L nodes, the number of variables to compute is $M = L(R + 1)(K + 1)$. As an example, a sixth-order accurate method in time with $K = 2$ and $R = 1$ requires $M = 6L$ variables.

6.1.4 Convergence order of the intermediate steps

An important property of the structural scheme is to guarantee the same convergence order at all time steps, including the intermediate time grid points within a block, that is, at t_{n+1} up to t_{n+R-1} . We check this property for the simple ODE $\phi'(t) = -\phi(t)$ for two representative situations. First, we consider the structural scheme $\text{Skm}[2, 2]$ and check that the variables at the middle time step t_{n+1} have the same convergence order as at t_{n+2} . To this end, the approximation errors $\phi_{N-1}^{(k)} - \phi^{(k)}(T - \Delta t)$ and $\phi_N^{(k)} - \phi^{(k)}(T)$ for $k \in \llbracket 0, 2 \rrbracket$, and the corresponding convergence orders, are reported in Table 8. The results confirm that the approximations at the intermediate time grid points have the same convergence order as those at the final time.

Table 8: Errors and convergence orders for the linear scalar ODE benchmark at the last R time steps (last block) with $K = 2$ and $R = 2$.

t	N	ϕ		$\phi^{(1)}$		$\phi^{(2)}$	
		err	ord	err	ord	err	ord
$T - 1\Delta t$	60	8.34E-16	---	8.34E-16	---	8.34E-16	---
	120	1.30E-17	6.0	1.30E-17	6.0	1.30E-17	6.0
	240	2.04E-19	6.0	2.04E-19	6.0	2.04E-19	6.0
$T - 0\Delta t$	60	8.34E-16	---	8.34E-16	---	8.34E-16	---
	120	1.30E-17	6.0	1.30E-17	6.0	1.30E-17	6.0
	240	2.04E-19	6.0	2.04E-19	6.0	2.04E-19	6.0

We repeat the experience with a more challenging situation, considering a 24th-order of convergence with the structural scheme $\text{Skm}[4,5]$ and $\varepsilon = 10^{-80}$, and check the approximation errors and convergence orders for all intermediate time steps, that is, $\phi_{N-r}^{(k)} - \phi^{(k)}(T - r\Delta t)$, $r \in \llbracket 0, R - 1 \rrbracket$, which are reported in Table 9. As observed, all intermediate time steps present the same optimal convergence order. Thus, we conclude that $\phi_{N-r}^{(k)}$ are not auxiliary variables but genuine approximations that can be used in complex models involving the coupling of differential equations, non-linearity, or boundary conditions if the system derives from a semi-discretisation in space.

Table 9: Errors and convergence orders for the linear scalar ODE benchmark at the last R time steps (last block) with $K = 4$ and $R = 5$.

t	N	ϕ		$\phi^{(1)}$		$\phi^{(2)}$		$\phi^{(3)}$		$\phi^{(4)}$	
		err	ord	err	ord	err	ord	err	ord	err	ord
$T - 4\Delta t$	60	1.47E-63	---	1.47E-63	---	1.47E-63	---	1.47E-63	---	1.47E-63	---
	120	8.67E-71	24.0	8.67E-71	24.0	8.67E-71	24.0	8.67E-71	24.0	8.67E-71	24.0
	240	5.14E-78	24.0	5.14E-78	24.0	5.14E-78	24.0	5.14E-78	24.0	5.14E-78	24.0
$T - 3\Delta t$	60	1.45E-63	---	1.45E-63	---	1.45E-63	---	1.45E-63	---	1.45E-63	---
	120	8.60E-71	24.0	8.60E-71	24.0	8.60E-71	24.0	8.60E-71	24.0	8.60E-71	24.0
	240	5.12E-78	24.0	5.12E-78	24.0	5.12E-78	24.0	5.12E-78	24.0	5.12E-78	24.0
$T - 2\Delta t$	60	1.42E-63	---	1.42E-63	---	1.42E-63	---	1.42E-63	---	1.42E-63	---
	120	8.53E-71	24.0	8.53E-71	24.0	8.53E-71	24.0	8.53E-71	24.0	8.53E-71	24.0
	240	5.10E-78	24.0	5.10E-78	24.0	5.10E-78	24.0	5.10E-78	24.0	5.10E-78	24.0
$T - 1\Delta t$	60	1.40E-63	---	1.40E-63	---	1.40E-63	---	1.40E-63	---	1.40E-63	---
	120	8.46E-71	24.0	8.46E-71	24.0	8.46E-71	24.0	8.46E-71	24.0	8.46E-71	24.0
	240	5.08E-78	24.0	5.08E-78	24.0	5.08E-78	24.0	5.08E-78	24.0	5.08E-78	24.0
$T - 0\Delta t$	60	1.44E-63	---	1.44E-63	---	1.44E-63	---	1.44E-63	---	1.44E-63	---
	120	8.57E-71	24.0	8.57E-71	24.0	8.57E-71	24.0	8.57E-71	24.0	8.57E-71	24.0
	240	5.11E-78	24.0	5.11E-78	24.0	5.11E-78	24.0	5.11E-78	24.0	5.11E-78	24.0

6.2 Non-linear scalar ODE

Non-linear scalar ODEs with stiff and sensitive solutions are addressed in this section to assess the ability of the structural schemes to provide accurate numerical approximations. Non-linearity may also lead to blow-up in finite time, and the performance of the scheme depends on its ability to capture the solution very close to the final time.

6.2.1 High sensitivity to the initial condition

Consider the non-linear scalar ODE $\phi'(t) = \phi(t)(1 - \phi(t))$ in $(-10, 0]$ with the initial condition $\phi(-10) = 1/(1 + \exp(10))$. The solution is the sigmoid function $\phi(t) = \sigma(t) = 1/(1 + \exp(-t))$, which presents two horizontal asymptotes at $-\infty$ and $+\infty$. The main issue is that small perturbations around the initial condition, for instance, a low-accuracy scheme, can lead to large deviations in the numerical approximation at $t = 0$. To assess the ability of the structural schemes to handle such a stiff initial condition, we run the benchmark for different values of K and R and report in Table 10 the errors and convergence orders.

Table 10: Errors and convergence orders for the non-linear scalar ODE benchmark with high sensitivity to the initial condition.

R	N	K=1						K=2						K=3		
		ϕ		$\phi^{(2)}$		$\phi^{(3)}$		ϕ		$\phi^{(3)}$		ϕ		ϕ		
		err	ord	err	ord	err	ord	err	ord	err	ord	err	ord	err	ord	err
1	120	1.01E-03	—	Ip=1		Ip=2		8.38E-08	—	Ip=2		2.50E-12	—			
	240	2.53E-04	2.0	4.69E-03	—	7.76E-01	—	5.23E-09	4.0	2.55E-04	—	3.90E-14	6.0			
	480	6.33E-05	2.0	2.48E-03	0.9	7.80E-01	—	3.27E-10	4.0	6.34E-05	2.0	6.10E-16	6.0			
	960	1.58E-05	2.0	1.27E-03	1.0	7.81E-01	—	2.04E-11	4.0	1.58E-05	2.0	9.52E-18	6.0			
				6.43E-04	1.0	7.81E-01	—			3.96E-06	2.0					
2	120	1.68E-06	—	Ip=2		Ip=3		2.67E-11	—	Ip=2		2.18E-20	—			
	240	1.05E-07	4.0	3.68E-05	—	9.45E-04	—	4.16E-13	6.0	5.57E-07	—	2.13E-23	10.0			
	480	6.54E-09	4.0	4.57E-06	3.0	2.01E-04	2.2	6.50E-15	6.0	3.54E-08	4.0	2.08E-26	10.0			
	960	4.09E-10	4.0	5.68E-07	3.0	4.80E-05	2.1	1.02E-16	6.0	2.22E-09	4.0	2.04E-29	10.0			
				7.08E-08	3.0	1.19E-05	2.0			1.39E-10	4.0					
3	120	7.59E-07	—	Ip=2		Ip=3		5.71E-15	—	Ip=3		3.94E-23	—			
	240	4.72E-08	4.0	3.56E-05	—	1.56E-03	—	2.23E-17	8.0	4.30E-09	—	9.59E-27	12.0			
	480	2.95E-09	4.0	4.52E-06	3.0	3.96E-04	2.0	8.69E-20	8.0	7.11E-11	5.9	2.34E-30	12.0			
	960	1.84E-10	4.0	5.66E-07	3.0	9.93E-05	2.0	3.39E-22	8.0	1.13E-12	6.0	5.71E-34	12.0			
				7.07E-08	3.0	2.49E-05	2.0			1.77E-14	6.0					
4	120	4.45E-09	—	Ip=3		Ip=4		3.76E-18	—	Ip=4		5.08E-29	—			
	240	6.93E-11	6.0	1.37E-06	—	6.57E-05	—	3.65E-21	10.0	5.22E-11	—	7.72E-34	16.0			
	480	1.08E-12	6.0	4.41E-08	5.0	4.20E-06	4.0	3.56E-24	10.0	2.37E-13	7.8	1.18E-38	16.0			
	960	1.69E-14	6.0	1.39E-09	5.0	2.64E-07	4.0	3.47E-27	10.0	9.61E-16	7.9	1.80E-43	16.0			
				4.34E-11	5.0	1.65E-08	4.0			3.79E-18	8.0					
5	120	1.17E-09	—	Ip=3		Ip=4		9.40E-21	—	Ip=4		5.54E-31	—			
	240	1.80E-11	6.0	6.24E-07	—	3.44E-05	—	2.27E-24	12.0	8.34E-13	—	5.70E-38	23.2			
	480	2.80E-13	6.0	2.15E-08	4.9	2.36E-06	3.9	5.52E-28	12.0	1.17E-15	9.5	2.17E-43	18.0			
	960	4.37E-15	6.0	6.89E-10	5.0	1.51E-07	4.0	1.35E-31	12.0	1.23E-18	9.9	8.28E-49	18.0			
				2.17E-11	5.0	9.50E-09	4.0			1.23E-21	10.0					

Following the same colour scheme as before, the errors and convergence orders in blue correspond to derivative approximations obtained from post-processing structural equations. In black, only the errors and convergence orders for the function are provided, since the derivatives associated with the physical equations have the same convergence order by construction of the structural schemes. In that regard, the errors exhibit the expected convergence orders, and the structural schemes with higher convergence orders prevent the numerical approximation from degrading at the final time. Experiments with a steeper solution $\sigma(\lambda t)$ with $\lambda = 10, 20$, which are not provided for compactness, further support the ability of the structural schemes to handle low-conditioning problems.

6.2.2 Finite time solution with blow-up

Another class of challenging problems concerns solutions that end at a finite time T with a blow-up, that is, $\lim_{t \rightarrow T} \phi(t) = +\infty$. As an example, let us consider the ODE $\phi' = \phi^2$ in $(0, 1)$ with the initial condition $\phi(0) = 1$, whose solution is $\phi(t) = 1/(1 - t)$. We compute the numerical solution up to $t = 0.95$, whose exact value is $\phi(0.95) = 20$. Table 11 reports the errors and convergence orders for different values of K and R , choosing $I_p = I_{p, \min}$ to complete the table with the higher derivatives, determined from post-processing structural equations.

Table 11: Errors and convergence orders for the non-linear scalar ODE benchmark with a blow-up solution at $t = 0.95$.

R	N	K=1						K=2						K=3		
		ϕ		$\phi^{(2)}$		$\phi^{(3)}$		ϕ		$\phi^{(3)}$		ϕ		ϕ		
		err	ord	err	ord	err	ord	err	ord	err	ord	err	ord	err	ord	
1	120	2.45E-01	—	Ip=1		Ip=2		—		Ip=2		—		—		
	240	6.00E-02	2.0	2.56E+03	—	4.04E+06	—	7.00E-04	—	5.65E+04	—	3.12E-06	—	—		
	480	1.49E-02	2.0	1.56E+03	0.7	4.78E+06	—	4.37E-05	4.0	1.62E+04	1.8	4.91E-08	6.0	—		
	960	3.72E-03	2.0	8.58E+02	0.9	5.31E+06	—	2.73E-06	4.0	4.26E+03	1.9	7.69E-10	6.0	—		
	960	3.72E-03	2.0	4.51E+02	0.9	5.63E+06	—	1.70E-07	4.0	1.08E+03	2.0	1.20E-11	6.0	—		
2	120	2.75E-03	—	Ip=2		Ip=3		—		Ip=2		—		—		
	240	1.74E-04	4.0	9.07E+02	—	3.64E+05	—	3.15E-05	—	2.09E+03	—	2.60E-09	—	—		
	480	1.09E-05	4.0	2.95E+02	1.6	2.76E+05	0.4	5.16E-07	5.9	2.08E+02	3.3	3.05E-12	9.7	—		
	960	6.82E-07	4.0	8.54E+01	1.8	1.62E+05	0.8	8.17E-09	6.0	1.67E+01	3.6	3.14E-15	9.9	—		
	960	6.82E-07	4.0	2.31E+01	1.9	8.55E+04	0.9	1.28E-10	6.0	1.20E+00	3.8	3.11E-18	10.0	—		
3	120	6.09E-03	—	Ip=2		Ip=3		—		Ip=3		—		—		
	240	3.89E-04	4.0	3.27E+02	—	1.81E+05	—	2.87E-06	—	2.62E+02	—	7.19E-10	—	—		
	480	2.45E-05	4.0	6.71E+01	2.3	6.78E+04	1.4	1.40E-08	7.7	9.50E+00	4.8	2.90E-13	11.3	—		
	960	1.53E-06	4.0	1.10E+01	2.6	2.14E+04	1.7	5.84E-11	7.9	2.38E-01	5.3	8.37E-17	11.8	—		
	960	1.53E-06	4.0	1.60E+00	2.8	6.05E+03	1.8	2.32E-13	8.0	4.79E-03	5.6	2.14E-20	11.9	—		
4	120	5.12E-04	—	Ip=3		Ip=4		—		Ip=4		—		—		
	240	9.68E-06	5.7	1.51E+02	—	8.97E+04	—	4.00E-07	—	4.57E+01	—	8.18E-12	—	—		
	480	1.61E-07	5.9	1.90E+01	3.0	2.14E+04	2.1	6.70E-10	9.2	6.50E-01	6.1	4.15E-16	14.3	—		
	960	2.55E-09	6.0	1.79E+00	3.4	3.89E+03	2.5	7.92E-13	9.7	5.34E-03	6.9	1.03E-20	15.3	—		
	960	2.55E-09	6.0	1.40E-01	3.7	5.98E+02	2.7	8.17E-16	9.9	3.12E-05	7.4	1.82E-25	15.8	—		
5	120	9.94E-04	—	Ip=3		Ip=4		—		Ip=4		—		—		
	240	2.00E-05	5.6	7.11E+01	—	4.75E+04	—	7.48E-08	—	1.00E+01	—	3.23E-12	—	—		
	480	3.41E-07	5.9	6.04E+00	3.6	7.47E+03	2.7	4.87E-11	10.6	5.99E-02	7.4	7.45E-17	15.4	—		
	960	5.46E-09	6.0	3.38E-01	4.2	8.06E+02	3.2	1.77E-14	11.4	1.69E-04	8.5	6.41E-22	16.8	—		
	960	5.46E-09	6.0	1.45E-02	4.5	6.81E+01	3.6	4.90E-18	11.8	2.96E-07	9.2	3.25E-27	17.6	—		

The convergence orders are perfectly in line with the expected results for both the solution and derivative approximations. Experiments considering the numerical approximations at $t = 0.995$, whose exact solution is $\phi(0.995) = 200$, and which are not provided for compactness, provide the same convergence orders.

6.3 Linear vectorial ODE

Differential system involving several variables coupled via ODEs is a fundamental topic for several reasons: the splitting of a second- or higher-order differential scalar equation into a first-order differential system, Hamiltonian problems, which lead to a set of first-order differential equations combining the primal and dual variables, and semi-discretisation in space of non-stationary PDEs, which provides large (non-)linear systems to numerically solve. We explore the basic case of the linear wave propagation as a representative example, where we split the wave equation into the first-order differential system

$$\begin{cases} \phi'(t) = \alpha\psi(t), \\ \psi'(t) = -\alpha\phi(t), \end{cases}$$

in $(0, 1]$ with the initial conditions $\phi(0) = 1$ and $\psi(0) = 0$. The solutions are simply $\phi(t) = \cos(\alpha t)$ and $\psi(t) = -\sin(\alpha t)$, where α is the angular frequency.

Table 12: Errors and convergence orders for ϕ the linear vectorial ODE benchmark.

R	N	K=1		K=2		K=3	
		err	ord	err	ord	err	ord
1	60	2.03E-03	—	4.14E-07	—	3.57E-11	—
	120	5.12E-04	2.0	2.59E-08	4.0	5.58E-13	6.0
	240	1.28E-04	2.0	1.62E-09	4.0	8.73E-15	6.0
2	60	6.60E-06	—	3.80E-10	—	2.01E-17	—
	120	4.14E-07	4.0	5.95E-12	6.0	1.96E-20	10.0
	240	2.59E-08	4.0	9.31E-14	6.0	1.92E-23	10.0
3	60	3.68E-06	—	4.14E-13	—	3.36E-21	—
	120	2.32E-07	4.0	1.62E-15	8.0	8.25E-25	12.0
	240	1.45E-08	4.0	6.36E-18	8.0	2.02E-28	12.0
4	60	4.53E-08	—	5.28E-16	—	1.09E-26	—
	120	7.13E-10	6.0	5.22E-19	10.0	1.66E-31	16.0
	240	1.12E-11	6.0	5.11E-22	10.0	2.54E-36	16.0
5	60	1.58E-08	—	7.78E-19	—	1.28E-30	—
	120	2.54E-10	6.0	1.94E-22	12.0	4.97E-36	18.0
	240	3.99E-12	6.0	4.76E-26	12.0	1.91E-41	18.0

We consider $\alpha = 2.1\pi$ to avoid the situation where the solution at the final time does not correspond to an extremum, which might produce super-convergence effects. The errors and convergence orders for different values of K and R are reported in Table 12. As observed, the expected convergence orders are obtained, following the same behaviour as in the scalar case, and the coupling between $\phi(t)$ and $\psi(t)$ does not cause any additional difficulty. Experiments considering larger values for α , which are not provided for compactness, reproduced the same convergence orders and stability.

Another point concerns the invariant conservation. Indeed, the quantity $m(t) = \phi^2(t) + \psi^2(t)$ has to be preserved throughout the simulation. Let us denote as m_{n+r} the approximation at time t_{n+r} and, for any structural scheme $\text{Skm}[K, R]$, for which the invariant error is given as

$$e_m = \max_{i=0, \dots, N/R} |m_{iR+r} - 1|, \quad r \in \llbracket 1, R \rrbracket.$$

In other words, $r = 1, \dots, R-1$ are the intermediate time steps, whereas $r = R$ is the final time step within each block.

Table 13: Invariant errors for the linear vectorial ODE benchmark with $K = 1, 2, 3$ and $R = 4$.

K	N	r=1	r=2	r=3	r=4
1	60	6.51E-08	3.82E-08	6.51E-08	1.47E-61
	120	1.03E-09	6.09E-10	1.03E-09	9.45E-62
	240	1.62E-11	9.57E-12	1.62E-11	2.96E-62
2	60	1.73E-17	1.82E-17	1.73E-17	1.35E-60
	120	4.25E-21	4.48E-21	4.25E-21	3.30E-61
	240	1.04E-24	1.09E-24	1.04E-24	7.67E-61
3	60	1.34E-26	1.25E-26	1.34E-26	9.47E-61
	120	2.07E-31	1.94E-31	2.07E-31	3.80E-61
	240	3.17E-36	2.97E-36	3.17E-36	1.77E-61

The invariant errors for $K = 1, 2, 3$ and $R = 4$ are reported in Table 13 using octuple-precision in the Julia framework. As observed, the invariant preservation is only achieved with full precision at the last step of each block ($R = 4$, in blue), whereas the invariant errors of the intermediate steps are of the same convergence order as that of the solution and derivatives. Indeed, for $R = 4$, the invariant errors observed correspond to the machine truncation for octuple-precision numbers. Similar experiments have been carried out with different structural schemes, not provided for compactness, which show the same behaviour.

6.4 The Van der Pol equation

The van der Pol equation enables the modelling of oscillatory non-linear systems in many scientific fields and is a standard benchmark for assessing numerical schemes [93, 94]. Indeed, the solution exhibits several regimes (almost constant in some intervals, with rapid changes) that are challenging from a computational perspective. We consider

the original Van der Pol equation $x''(t) - \mu(1 - x^2(t))x'(t) + x(t) = 0$ in $(0, T]$, where x is the position, with the initial conditions $x(0) = 0$ and $x'(0) = 0$, which is rewritten as a first-order differential system, given as

$$\begin{cases} x'(t) = y(t), \\ y'(t) = \mu(1 - x^2(t))y(t) - x(t), \end{cases}$$

where the parameter μ controls the stiffness of the solution.

6.4.1 Convergence assesemnt

We consider two situations: a smooth transition with $\mu = 1$ and a sharp transition with $\mu = 50$. Following the reference [95], we choose the initial conditions corresponding to the maximum of the solution and evaluate the numerical approximations after a complete cycle, for which the position, velocity, and period with 76 digits are retrieved from the same reference, and read

- For $\mu = 1$:

$x(0)=2.008619860\ 8748431365\ 0964018836\ 2640366192\ 0261377207\ 1446112770\ 3247615558\ 726585,$
 $y(0)=0,$
 $T=6.663286859\ 3231301896\ 9968203048\ 2328706812\ 6463168838\ 7665511486\ 3182080925\ 864845.$

- For $\mu = 50$:

$x(0)=2.002955933\ 7576484028\ 2610971894\ 5871764979\ 9147201460\ 6949163926\ 8919228280\ 941255,$
 $y(0)=0,$
 $T=82.50833389\ 3230782186\ 8333686974\ 9987459374\ 0590858618\ 6277867538\ 5441021240\ 035171.$

The errors and convergence orders for the position are reported in Table 14 with $\mu = 1$, while Figure 2 illustrates the position and the velocity for a complete cycle for $K = 1$, $R = 1$, and $N = 480$. The velocity errors are not reported because they exhibit the same behaviour as the position errors. We first observe that the expected convergence orders are achieved, further supporting the excellent efficiency of the structural schemes. Table 14 also provides the average number of times the physical equations are evaluated per time step, $\bar{\tau}$, required to achieve the convergence of the fixed-point.

Table 14: Position errors and convergence order for the van der Pol equations benchmark with $\mu = 1$.

R	N	K=1			K=2			K=3		
		$\bar{\tau}$	err	ord	$\bar{\tau}$	err	ord	$\bar{\tau}$	err	ord
1	480	20.85	2.88E-05	—	39.94	3.10E-09	—	88.74	8.37E-14	—
	600	19.79	1.85E-05	2.0	37.80	1.27E-09	4.0	84.06	2.20E-14	6.0
	720	19.04	1.28E-05	2.0	36.20	6.12E-10	4.0	80.25	7.35E-15	6.0
2	480	10.90	3.26E-08	—	22.00	8.93E-13	—	45.36	2.35E-20	—
	600	10.33	1.34E-08	4.0	20.76	2.34E-13	6.0	42.72	2.52E-21	10.0
	720	9.89	6.44E-09	4.0	19.84	7.84E-14	6.0	41.07	4.07E-22	10.0
3	480	7.67	2.79E-08	—	16.10	6.39E-16	—	31.80	1.60E-23	—
	600	7.25	1.14E-08	4.0	15.08	1.07E-16	8.0	30.03	1.10E-24	12.0
	720	6.95	5.51E-09	4.0	14.34	2.49E-17	8.0	28.62	1.23E-25	12.0
4	480	5.90	5.16E-11	—	13.30	1.66E-18	—	35.32	1.84E-28	—
	600	5.59	1.36E-11	6.0	12.44	1.78E-19	10.0	33.20	4.91E-30	16.2
	720	5.33	4.54E-12	6.0	11.76	2.87E-20	10.0	31.52	2.58E-31	16.2
5	480	4.87	3.84E-11	—	11.98	3.76E-21	—	24.21	9.41E-31	—
	600	4.59	1.01E-11	6.0	11.04	2.59E-22	12.0	22.44	1.68E-32	18.0
	720	4.38	3.37E-12	6.0	10.38	2.91E-23	12.0	21.24	6.30E-34	18.0

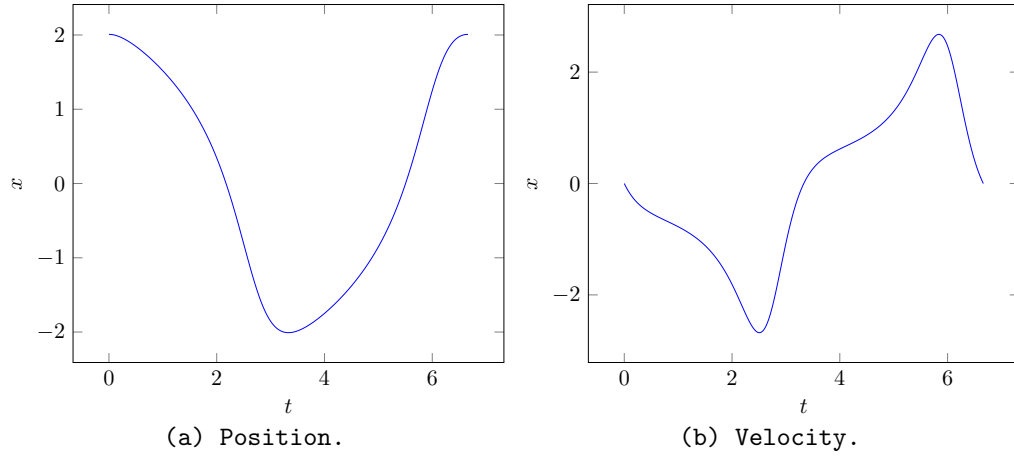


Figure 2: Approximate solution for the van der Pol equations benchmark with $\mu = 1$ for $K = 1$, $R = 1$, and $N = 480$

Table 15 reports the errors and convergence orders for the position with $\mu = 50$ (again, velocity is not reported). Due to the sharp transition, we used a larger number of time grid points to catch the velocity singularity, as shown in Figure 3. We observe a drastic reduction in the parameter $\bar{\tau}$ compared to the previous case with $\mu = 1$, since the smaller Δt provides better predictors and allows fixed-point convergence with fewer iterations per time step.

Table 15: Position errors and convergence orders for the van der Pol equations benchmark with $\mu = 50$.

R	N	K=1			K=2			K=3		
		$\bar{\tau}$	err	ord	$\bar{\tau}$	err	ord	$\bar{\tau}$	err	ord
1	840,000	4.47	1.57E-06	—	21.48	2.16E-12	—	50.73	3.10E-18	—
	960,000	4.17	1.21E-06	2.0	20.94	1.26E-12	4.0	49.20	1.39E-18	6.0
	1,080,000	4.09	9.55E-07	2.0	20.20	7.89E-13	4.0	48.21	6.87E-19	6.0
2	840,000	2.51	3.44E-11	—	11.62	3.31E-17	—	25.95	2.36E-27	—
	960,000	2.51	2.02E-11	4.0	11.32	1.49E-17	6.0	25.32	6.20E-28	10.0
	1,080,000	2.51	1.26E-11	4.0	10.90	7.33E-18	6.0	24.66	1.91E-28	10.0
3	840,000	1.68	1.94E-11	—	8.32	1.06E-21	—	18.03	2.64E-32	—
	960,000	1.68	1.14E-11	4.0	8.06	3.64E-22	8.0	17.49	5.31E-33	12.0
	1,080,000	1.67	7.10E-12	4.0	7.82	1.42E-22	8.0	17.04	1.29E-33	12.0
4	840,000	1.26	3.99E-15	—	6.68	6.27E-26	—	14.64	9.02E-40	—
	960,000	1.26	1.79E-15	6.0	6.48	1.65E-26	10.0	14.16	1.07E-40	16.0
	1,080,000	1.26	8.83E-16	6.0	6.28	5.08E-27	10.0	13.80	1.62E-41	16.0
5	840,000	1.20	1.42E-15	—	5.78	6.21E-30	—	12.84	1.61E-44	—
	960,000	1.03	6.38E-16	6.0	5.58	1.25E-30	12.0	12.39	1.46E-45	18.0
	1,080,000	1.01	3.15E-16	6.0	5.38	3.04E-31	12.0	12.06	1.75E-46	18.0

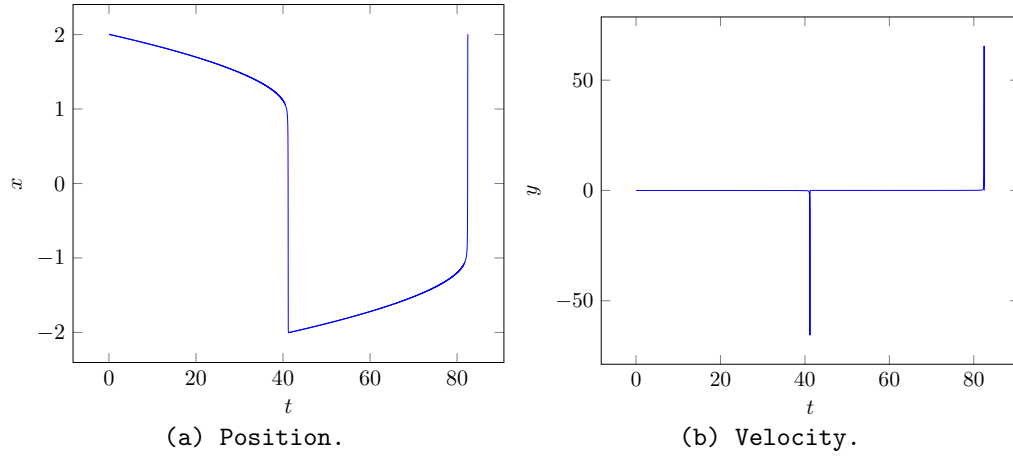


Figure 3: Approximate solution for the van der Pol equations benchmark with $\mu = 50$ for $K = 1$, $R = 1$, and $N = 840000$

6.4.2 Variable time step

The present benchmark aims to demonstrate that a substantial reduction in computational cost can be achieved through better control of the time step according to the smoothness of the solution. The van der Pol equation with large μ is an interesting example, since the solution is almost linear over large portions of the time interval, while strong variations occur over small intervals. We use a grid with two different time steps, referred to as a two-step grid: the smooth part is given by the set $\Omega_s = [0, 41] \cup [42, 79]$, while the peak regions correspond to the domain $[41, 42] \cup [79, 80]$. **Provide the exact values of the intervals.** We use Δt_s , corresponding to a subdivision with N_s points in the smooth region, whereas Δt_p is controlled by N_p in the peak regions. On the other hand, we also consider a uniform time discretisation with time step Δt and N points.

Table 16: Comparison of time (in ms) versus error for $K = 1$, $K = 2$, and $K = 3$.

R	N	K=1		K=2		K=3	
	N_s/N_p	time	err	time	err	time	err
1	840k	878	4.6e-04	843	2.4e-06	1075	4.9e-09
	960k	956	3.5e-04	939	1.6e-06	1274	2.9e-09
	1,080k	1216	2.8e-04	1079	1.1e-06	1372	1.8e-09
	35k/7,5k	73	1.8e-03	85	2.0e-05	112	8.2e-08
	40k/10k	83	1.0e-03	98	8.4e-06	128	2.6e-08
	45k/12,5k	91	6.8e-04	113	4.3e-06	153	1.1e-08
2	840k	3947	5.3e-09	2571	4.9e-15	2089	1.9e-23
	960k	4300	3.1e-09	2825	2.2e-15	2271	2.2e-23
	1,080k	3974	1.9e-09	3065	1.1e-15	2478	2.4e-23
	35k/7,5k	537	9.0e-08	555	4.6e-13	538	6.7e-20
	40k/10k	571	2.9e-08	607	1.1e-13	606	1.8e-20
	45k/12,5k	623	1.2e-08	678	4.0e-14	599	5.5e-21
4	84k	3659	5.9e-13	3648	1.1e-23	2993	8.9e-23
	960k	3878	2.6e-13	3986	4.0e-24	3372	1.0e-22
	1,080k	4200	1.3e-13	4301	2.5e-24	3412	1.1e-22
	35k/7,5k	690	4.0e-11	856	1.7e-17	826	7.9e-24
	40k/10k	712	6.5e-12	922	4.5e-18	884	9.1e-24
	45k/12,5k	767	1.5e-12	1066	1.4e-18	969	1.0e-23

We have selected several choices of R and N_s/N_p that provide similar accuracy errors in order to draw some conclusions. For a given error level, we are able to reduce the running time by a factor between 6 and 8. In the case $R = 4$ for the two-step grid, the method reaches the limits of quad-precision arithmetic, and the errors are therefore bounded by the floating-point representation.

These preliminary results demonstrate the potential for substantial gains by dynamically adjusting the time step ac-

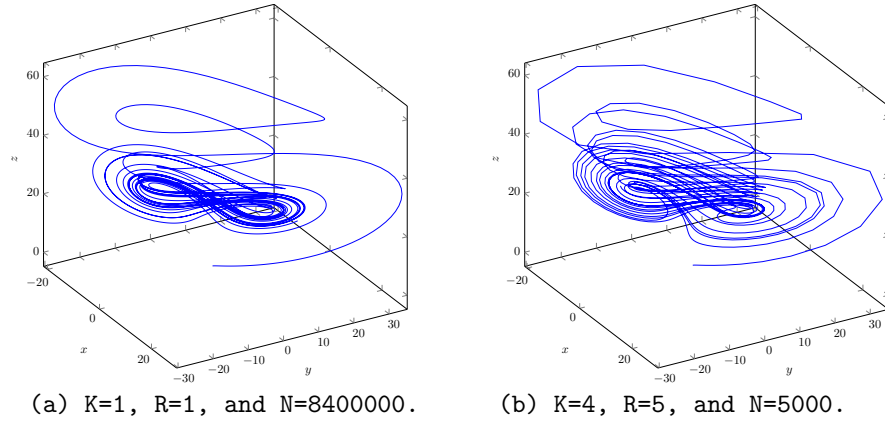


Figure 4: Approximate solution to the Chen system.

cording to a smoothness parameter, such as the first and second derivatives, and by appropriately tuning Δt so as to preserve a prescribed accuracy while reducing the computational cost.

6.5 The Chen system

Chaotic differential equations are unstable in the sense that small perturbations of the initial conditions lead to entirely alternative solutions. Therefore, achieving consistency and stability in numerical computation is challenging. Indeed, chaotic differential equations are quite sensitive to the discretisation errors, which are amplified, leading to wrong approximations.

6.5.1 General view

An important dynamical system is the Chen chaotic problem, first introduced by Chen and Ueta [96]. It is considered as a traditional benchmark for assessing a numerical method's ability to reproduce the correct solution up to a given time since small variations lead to important deviations at the final time. We adopt the original formulation, given as

$$\begin{cases} x'(t) = a(y(t) - x(t)), \\ y'(t) = (c - a)x(t) - x(t)z(t) + cy(t), \\ z'(t) = x(t)y(t) - bz(t), \end{cases}$$

with a , b , and c the parameters of the system. In the present simulation, we consider the usual values of $a = 35$, $b = 3$, and $c = 28$ and $T = 15$, which provide a chaotic behaviour [97, 98].

6.5.2 Numerical simulations

In Figure 4, a three-dimensional view of the solution is illustrated using a second-order accurate approximation with $N = 8400000$ and a 24th-order accurate approximation with $N = 5000$. The solution has two attractors, clearly identified by two planes and two centres, which are captured in both approximations.

The same experience is reproduced with a structural scheme of an intermediate convergence order of 8 with $K = 2$ and $R = 3$, and $N = 60000$, whose approximate solution for the three coordinates as a function of time is illustrated in Figure 5.

6.5.3 Sensitivity

The main issue in computing an approximate solution to a chaotic system is the high sensitivity of the structural schemes to the parameters K and R , which determine the accuracy of the numerical approximations. Moreover, the iterative fixed-point procedure depends on the tolerance ε , which determines the residual of the final converged numerical solution. We study the impact of these parameters on the chaotic Chen system.

Since there is no exact solution to evaluate the approximate solution accuracy and stability, a reference solution is taken from performing a simulation with the structural scheme $\text{Skm}[4,8]$ with convergence of order of 36 using an octuple-precision, a very restrict tolerance of $\varepsilon = 10^{-60}$ for the fixed-point residual, and a very fine time grid with $N = 800000$ points to guarantee an extreme accuracy of the approximate solution. On the other hand, we perform the

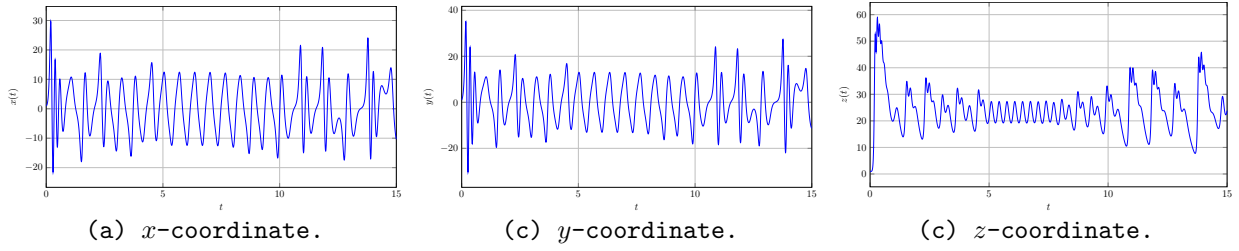


Figure 5: The three coordinate curves as a function of time for the Chen system benchmark with $K = 2$, $R = 3$, and $N = 60000$.

simulation with three different configurations: the structural scheme $\text{Skm}[1,1]$ with $N = 8400000$, $\text{Skm}[1,2]$ with $N = 4200000$, and $\text{Skm}[4,5]$ with $N = 5000$.

The three coordinates of the reference and approximate solutions are reported in Table 17. The correct digits corresponding to the reference solution are reported in blue, whereas the digits that do not match the reference solution are reported in black. The structural scheme $\text{Skm}[1,1]$, with second-order convergence, does not provide an acceptable approximate solution to the chaotic system, even with a large number of points, demonstrating the importance of using higher-order accurate schemes. The structural scheme $\text{Skm}[1,2]$, with fourth-order of convergence, provides a rough approximation and is ineffective from a practical viewpoint. On the other side, the structural scheme $\text{Skm}[4,5]$, with 24th-order of accuracy, manages to compute a reasonable approximation, even with a rather low number of time grid points. Indeed, very high-order accurate structural schemes deliver excellent approximations with compact stencils.

Table 17: Reference and approximate solutions for the Chen system benchmark.

scheme	position coordinates
Reference	$x = -0.1033913519761038851277830669958320E+02$ $y = -0.1110031188034702371431870391381926E+02$ $z = +0.2384877914088903906611669394788490E+02$
$\text{Skm}[4,8]$, $N=800000$	
$\text{Skm}[1,1]$, $N=8400000$	$x = +0.1303137147172269558345093394817314E+02$ $y = +0.1633314173284638979195080632679370E+02$ $z = +0.1990546738498253871112868554565699E+02$
$\text{Skm}[1,2]$, $N=4200000$	$x = -0.1034310938749620698298379213201446E+02$ $y = -0.1109718401089064385254542611775984E+02$ $z = +0.2386294859230536471454733356240557E+02$
$\text{Skm}[4,5]$, $N=5000$	$x = -0.1033913519761047665823283445056424E+02$ $y = -0.1110031188034695540840356160225629E+02$ $z = +0.2384877914088935167622294039641605E+02$

The impact of the tolerance ε on the accuracy of the approximate solution of the Chen system is also assessed. The error of the x -coordinate, in relation to the reference solution, is reported in Table 18 as a function of the tolerance ε with the structural schemes $\text{Skm}[3,R]$ with $R = 1, \dots, 5$. Due to the intricate nature of the chaotic system, $\varepsilon = 10^{-8}$ is far from providing an acceptable approximate solution. For $\varepsilon = 10^{-15}$, a rough approximation to the reference solution is obtained, but the expected convergence order of the structural scheme is not achieved. On the other side, $\varepsilon = 10^{-25}$ enables recovery only of the sixth-order of convergence. In fact, we need to set $\varepsilon = 10^{-30}$ to achieve the tenth-order of convergence, and, finally, $\varepsilon = 10^{-50}$ to get all the expected convergence orders. In conclusion, for this challenging benchmark, the tolerance value is a critical parameter, together with quad- or octuple-precision, to achieve extreme convergence orders.

Table 18: x -coordinate errors and convergence orders for the Chen system benchmark with $K = 3$ and $R = 1, \dots, 5$.

R	N	$\varepsilon=1\text{E-}8$		$\varepsilon=1\text{E-}15$		$\varepsilon=1\text{E-}25$		$\varepsilon=1\text{E-}30$		$\varepsilon=1\text{E-}50$	
		err	ord	err	ord	err	ord	err	ord	err	ord
1	60,000	1.33E+01	—	2.12E-02	—	1.33E-02	—	1.33E-02	—	1.33E-02	—
	120,000	1.27E+01	0.1	7.46E-03	1.5	2.11E-04	6.0	2.11E-04	6.0	2.11E-04	6.0
	180,000	1.41E+01	—	6.97E-03	0.2	1.85E-05	6.0	1.85E-05	6.0	1.85E-05	6.0
2	60,000	3.23E-01	—	4.02E-03	—	1.16E-11	—	1.18E-11	—	1.18E-11	—
	120,000	2.09E+01	—	4.25E-03	—	1.40E-13	6.4	1.15E-14	10.0	1.15E-14	10.0
	180,000	5.63E-02	14.6	1.91E-03	2.0	2.49E-13	—	1.99E-16	10.0	2.00E-16	10.0
3	60,000	1.78E+01	—	3.30E-03	—	1.28E-13	—	1.16E-16	—	1.15E-16	—
	120,000	2.66E+01	—	3.45E-03	—	2.88E-13	—	5.95E-19	7.6	2.81E-20	12.0
	180,000	2.62E+01	0.0	2.33E-03	1.0	2.75E-14	5.8	2.44E-19	2.2	2.16E-22	12.0
4	60,000	1.40E+01	—	1.23E-04	—	2.10E-13	—	1.92E-18	—	2.50E-24	—
	120,000	1.57E+00	3.2	2.88E-03	—	2.15E-13	—	1.12E-18	0.8	3.82E-29	16.0
	180,000	3.09E+00	—	2.48E-03	0.4	1.03E-13	1.8	4.26E-19	2.4	5.82E-32	16.0
5	60,000	3.76E+00	—	3.48E-03	—	1.39E-13	—	1.45E-18	—	3.43E-29	—
	120,000	1.63E+01	—	2.53E-03	0.5	8.36E-15	4.1	5.57E-19	1.4	1.31E-34	18.0
	180,000	4.32E+00	3.3	4.03E-03	—	1.80E-13	—	7.46E-19	—	8.34E-38	18.2

6.6 Numerical efficiency

Running time *versus* accuracy characterises the computational effort to achieve a given solution quality. In that regard, higher-order accurate methods are highly effective at reducing the running time for a given solution quality, since higher-order convergence allows the use of coarser grids. Indeed, for smooth solutions with low variation at the final time, the computational savings are significant. On the other hand, when dealing with solutions involving abrupt time changes, a physical constraint on the time step may arise. For instance, if the solution exhibits a high-frequency phenomenon, a satisfactory spectral resolution can only be achieved with a sufficiently small time step. To assess the numerical efficiency of the structural schemes, the Chen system is analysed at the final time $T = 15$ as a representative case, where we found that $N = 5000$ is the minimum number of time grid points to guarantee an acceptable spectral resolution.

6.6.1 Running time versus accuracy for the $\text{Skm}[K, R]$ schemes

First-derivative structural schemes $\text{Skm}[1, R]$ with convergence order of $R + 1$ involve only the function and the first derivative, and no additional physical equation is required, which is convenient when dealing with problems that do not support additional derivations. Note that one iteration of the fixed-point procedure evaluates R times the physical equation and involves R unknowns. On the other hand, high-order accurate structural schemes, such as $\text{Skm}[2, R]$ and $\text{Skm}[3, R]$, require additional physical equations and unknowns. Moreover, in these cases, one iteration of the fixed-point procedure evaluates KR times the physical equations and involves RK unknowns, hence, a larger system needs to be solved.

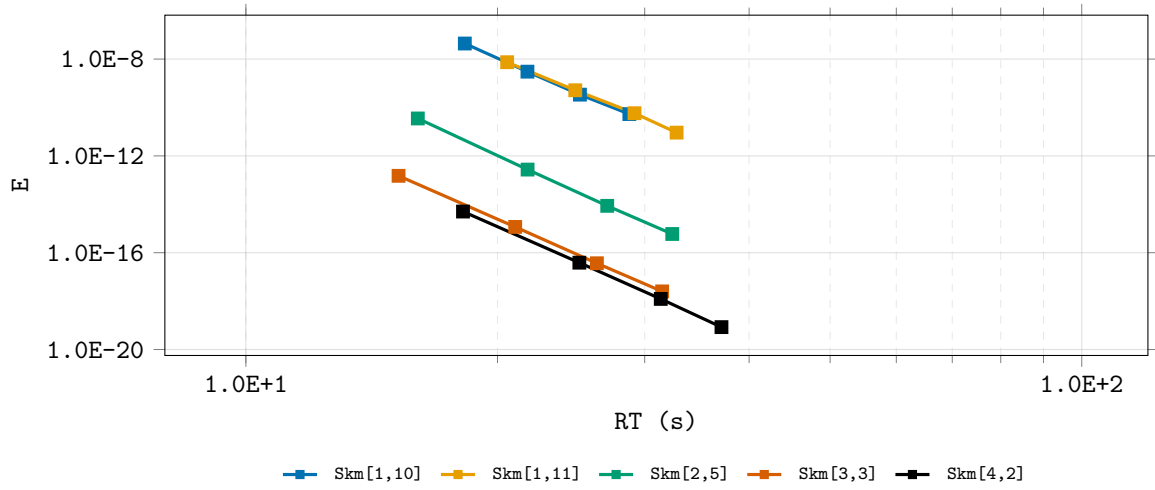
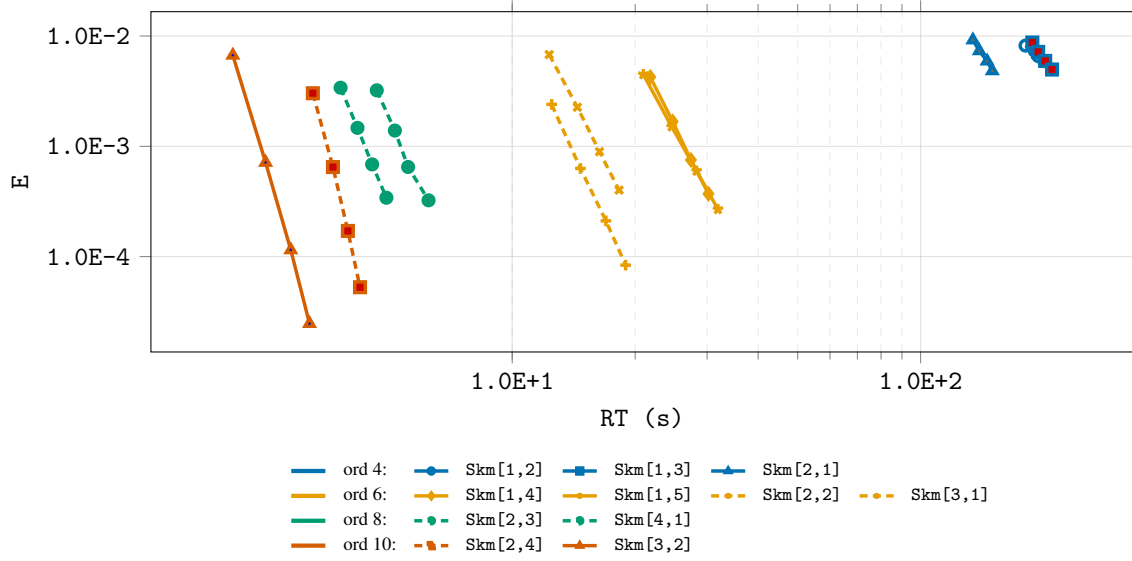


Figure 6: Running time (RT) versus error (E) for various structural schemes.

Figure 6(a) illustrates the accuracy (E , on the vertical axis) as a function of the running time (RT , on the horizontal axis) for various structural schemes with convergence orders of 4, 6, 8, and 10. We clearly distinguish three clusters for the fourth-order accurate schemes (right), the sixth-order accurate schemes (middle), and the eighth-/tenth-order accurate schemes (left). For a target accuracy of $E = 10^{-2}$, it takes around 200s for the fourth-order accurate structural schemes, 20s for the sixth-order accurate structural schemes, and only 2s for the tenth-order accurate structural schemes.

Figure 6(b) illustrates the differences between the possible combinations of parameters K and R that achieve a 12th-order of convergence. The lines are parallel, confirming that all the structural schemes have 12th-order convergence. On the other hand, increasing K rather than R to achieve the same convergence order is more computationally efficient, since it results in a drastic reduction in the multiplicative constant associated with the asymptotic regime. We stress that it would be difficult to take advantage of such a property if the analytical high-order derivatives of the physical equations turn out to be difficult or even impossible.

Assuming that the approximate solution has reached the asymptotic regime of the relation running time versus accuracy, we provide some extrapolations of the running time in Table 19 to reach a given accuracy for several structural

schemes. For example, to achieve an error above $E = 10^{-8}$, the structural scheme $\text{Skm}[1,2]$ with fourth-order of convergence requires about 3000 more time than the structural scheme $\text{Skm}[3,2]$. Such a ratio corresponds to a substantial difference between, for instance, 250s and one week of computation.

Table 19: Extrapolation time for various structural schemes.

ord	scheme	E=10-3	E=10-4	E=10-8
4	Skm [1,2]	357.11	749.16	1,4509.68
	Skm [1,3]	287.83	454.52	2826.47
	Skm [2,1]	197.40	294.63	1462.14
6	Skm [1,4]	26.44	35.99	123.63
	Skm [1,5]	26.18	36.95	146.56
	Skm [2,2]	16.11	22.19	79.80
	Skm [3,1]	13.92	18.56	58.73
8	Skm [2,3]	5.36	7.16	22.67
	Skm [4,1]	4.36	5.64	15.84
10	Skm [2,4]	3.51	4.09	7.50
	Skm [3,2]	2.41	2.88	5.89

6.7 Comparison with the Collocation method

Among the many existing numerical techniques, we restrict our comparison to the collocation method for several reasons. First, both the structural method and the collocation method can be interpreted as particular cases of RK methods and can be constructed to achieve arbitrary order of accuracy. Second, we do not consider multistep methods, which we regard as too different from the structural method, since our approach does not rely on backlog data. Finally, techniques such as deferred correction methods are based on a very different paradigm, increasing the order through successive improvements.

6.7.1 Algorithm

To provide a fair comparison between the structural and collocation methods, we use the same fixed-point technique as that proposed in Section 4. We denote by $A = (a_{ij})_{i,j=1}^s$, $b = (b_i)_{i=1}^s$, and $c = (c_i)_{i=1}^s$ the elements of the Butcher tableau, and by Z_n an approximation of the solution at time t_n .

We construct a sequence $(Z_{n+1}^{[k]})$, $k = 0, 1, \dots$, according to the following three stages:

1. Initialization $Z_{n,i}^{[0]} = Z_n + c_i \Delta t D_n$, $D_{n,i}^{[0]} = f(t_n + c_i \Delta t, Z_{n,i}^{[0]})$.
2. Loop $Z_{n,i}^{[k+1]} = Z_n + \Delta t \sum_{j=1}^s a_{ij} D_{n,j}^{[k]}$, $D_{n,i}^{[k+1]} = f(t_n + c_i \Delta t, Z_{n,i}^{[k+1]})$.
3. Test $Z_n^{[k+1]} = Z_n + \Delta t \sum_{i=1}^s b_i D_{n,i}^{[k+1]}$, stop if $|Z_n^{[k+1]} - Z_n^{[k]}| \leq \varepsilon$ else goto 2.
4. Termination $Z_{n+1} = Z_n^{[k+1]}$.

where $Z_{n,i}^{[k]}$, $D_{n,i}^{[k]}$ are, respectively, the values and derivatives at the intermediate stage $t_{n,i} = t_n + c_i \Delta t$. We have implemented the fourth-order CL4 up to the 12th-order method CL12 and report the Butcher tableau in D.

6.7.2 Running-time versus accuracy

We use the Chen dynamical system as a benchmark since small errors will quickly results into large deviations at the final time. All the code were implemented in C++ using the qd library to enable the quad- and octa-precision. Indeed, the approximations of a chaotic system require very high-precision. The qd library is quite efficient since a quad-precision run lasts 3 times the double precision run and the octa-precision run lasts 14 times the quad-precision run. For orders 4, 6 and 8, quad-precision is adequate but higher orders 10, 12 or more require the full octa-precision. We plot in Figure 7 left panel the accuracy in function of the running time for the 4, 6 and 8 order using the quad-precision, while we report the higher orders 10, 12 and 16 using the full octa-precision. Structural method of type $K = 1$ (only one derivative) provides very similar results than the collocation method. By introducing the compact

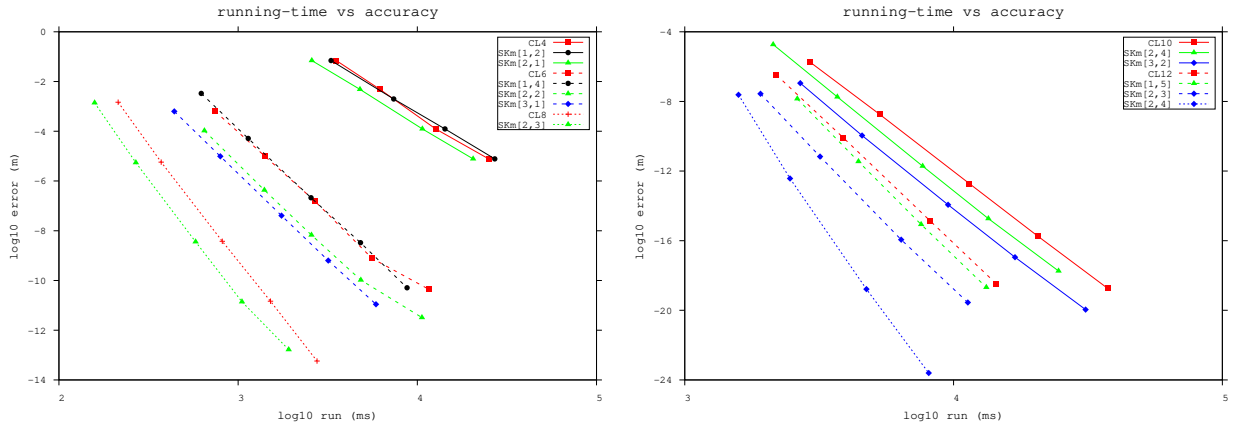


Figure 7: running time versus accuracy. Left panel: Order 4,6,8 using the quad-precision. Right panel: Order 10,12,14 using the octa-precision.

version using the second or the third derivative as additional unknowns, we strongly improve the running-time for the same accuracy regarding to the equivalent collocation methods.

Remark. Compact schemes usually provide better spectral resolution, i.e. the ability to preserve high frequency signal than the non-compact method. We expect that the $Skm[2,R]$ and $Skm[3,R]$ enjoy such a property in comparison to the collocation schemes. ■



Part II: Numerical analysis, accuracy, stability, spectral resolution

The second part is dedicated to the numerical analysis of the structural method, namely the convergence, the linear stability. We also introduce a concept of spectral resolution adapted to the new method to quantify the ability to obtain sub-grid scale information. We first prove the convergence of the iterative method for the non-linear problem in section 7 while the consistency errors and convergence orders are established in Section 8, where we determine the convergence order p as a function of K and R . Section 9 is dedicated to stability, where our method is shown to be unconditionally stable across a wide range of configurations. Appendix C gives the technical detail about the stability of the schemes we propose in the section. We address the spectral resolution issue in Section 10 and focus on the ability of our schemes to provide very high resolution, enabling us to recover sub-scale information, that is, variations between successive time steps.

7 Convergence of the numerical solver

The main result of this section establishes that, under a CFL-like condition linked to the Lipschitz constants of the physical equations and the coefficients of the structural equations, the proposed iterative procedure for numerical approximation converges.

Lemma 7.1. Let $B(M_0) = [-M_0, M_0]$ be the closed ball of radius $M_0 > 0$ centred at the origin and assume that the function $(z, t) \rightarrow f(z, t)$ is a $C^{K+1}(\mathbb{R} \times [0, T])$ function. Then, for any $k \in \llbracket 1, K \rrbracket$, the functions $(z_0, \dots, z_{k-1}, t) \rightarrow z_k = \mathbf{d}f^{(k-1)}[z_0, \dots, z_{k-1}, t]$ are bounded by a constant M_k . Moreover, we have a L_k -Lipschitz property uniform in time $t \in [0, T]$ on the compact set $B(M_0) \times \dots \times B(M_{k-1})$ given by

$$|z_k - z'_k| \leq L_k \left(\sum_{\kappa=0}^{k-1} |z_\kappa - z'_\kappa| \right). \quad (6)$$

Proof. We prove the property by induction. Since (z_0, t) belongs to the compact set $B(M_0) \times [0, T]$, the continuity of f means that $z_1 = \mathbf{d}f^{(0)}[t, z_0]$ is bounded and belongs to a compact set, which we include inside a closed ball $B(M_1)$ centered at the origin for M_1 large enough. On the other hand, for two values $z_0, z'_0 \in B(M_0)$, one has

$$\mathbf{d}f^{(0)}[z_0, t] - \mathbf{d}f^{(0)}[z'_0, t] = f(z_0, t) - f(z'_0, t) = \partial_z f(\theta z_0 + (1 - \theta)z'_0, t)(z_0 - z'_0)$$

with $\theta \in [0, 1]$. Note that the intermediate point belongs to the ball $B(M_0)$. Denoting

$$L_1 = \max_{\substack{\zeta \in B(M_0) \\ t \in [0, T]}} \partial_z f(\zeta, t),$$

we obtain the Lipschitz inequality, uniform in time

$$|z_1 - z'_1| = |\mathbf{df}^{(0)}[z_0, t] - \mathbf{df}^{(0)}[z'_0, t]| \leq L_1 |z_0 - z'_0|.$$

Assuming that the properties hold for $k < K$, we now prove them for $k + 1$. First, we recall that $z_{k+1} = \mathbf{df}^{(k)}[z_0, z_1, \dots, z_k, t]$ is made up of the sum of polynomial functions with respect to variables $z_1, \dots, z_k$ multiplied with partial derivatives of $f(z, t)$ in z_0 and t of order lower than k . Since the partial derivatives depend only on z_0 and t , they are uniformly bounded in $B(M_0) \times [0, T]$, while the polynomials in $z_1, \dots, z_k$ are also bounded from the induction assumption. We then see that z_{k+1} is bounded and belongs to a closed ball $B(M_{k+1})$ with M_{k+1} large enough. On the other hand, the Lipschitz property derives from the fact that $\mathbf{df}^{(k)}[z_0, z_1, \dots, z_k, t]$ is polynomial in $z_1, \dots, z_k$ with bounded values, while the z -derivative of the partial derivatives of order k remains bounded in $B(M_0) \times [0, T]$. Therefore, we have a Lipschitz operator uniform in time, and the property holds with a Lipschitz constant L_{k+1} . $\square$

From Lemma 7.1, we deduce a Lipschitz property regarding the sole variable z_0 .

Lemma 7.2. *Under the assumptions of Lemma 7.1, let $z_0 \in B(M_0)$ and $t \in [0, T]$, and define by induction*

$$z_k = \mathbf{df}^{(k-1)}[z_0, \dots, z_{k-1}, t], \quad k \in \llbracket 1, K \rrbracket.$$

Then, for any $k \in \llbracket 1, K \rrbracket$ and $z_0, z'_0 \in B(M_0)$, we have the inequality

$$|z_k - z'_k| \leq \widehat{L}_k |z_0 - z'_0|, \quad (7)$$

with $\widehat{L}_k$ independent of z_0, z'_0 , and time $t \in [0, T]$.

Proof. For $k = 1$ we take $\widehat{L}_1 = L_1$. For $k = 2$, we note

$$|z_2 - z'_2| \leq L_2(|z_0 - z'_0| + |z_1 - z'_1|) \leq L_2(1 + \widehat{L}_1)|z_0 - z'_0|,$$

and the inequality holds with $\widehat{L}_2 = L_2(1 + \widehat{L}_1)$.

The general case is proven by induction. From inequality (6) we have

$$|z_k - z'_k| \leq L_k \left(|z_0 - z'_0| + \sum_{\kappa=1}^{k-1} |z_\kappa - z'_\kappa| \right) \leq L_k \left(|z_0 - z'_0| + \sum_{\kappa=1}^{k-1} \widehat{L}_\kappa |z_0 - z'_0| \right) \leq \widehat{L}_k |z_0 - z'_0|$$

with

$$\widehat{L}_k = L_k \left(1 + \sum_{\kappa=1}^{k-1} \widehat{L}_\kappa \right),$$

and inequality (7) holds. $\square$

Corollary 7.3. *Let $\|\cdot\|_\infty$ be the maximum norm and*

$$\Phi_n^{(k)} = \mathbf{df}^{(k-1)}[\Phi_n^{(0)}, \dots, \Phi_n^{(k-1)}, T_n], \quad \Phi_n'^{(k)} = \mathbf{df}^{(k-1)}[\Phi_n'^{(0)}, \dots, \Phi_n'^{(k-1)}, T_n], \quad k \in \llbracket 1, K \rrbracket.$$

Then, under the assumptions of Lemma 7.1, we have for any $k \in \llbracket 1, K \rrbracket$ the estimate

$$\|\Phi_n^{(k)} - \Phi_n'^{(k)}\|_\infty \leq \widehat{L}_k \|\Phi_n^{(0)} - \Phi_n'^{(0)}\|_\infty. \quad (8)$$

Proof. There exists r_0 such that $\|\Phi_n^{(k)} - \Phi_n'^{(k)}\|_\infty = |\phi_{n+r_0}^{(k)} - \phi_{n+r_0}'^{(k)}|$. Then, applying inequality (7), we deduce

$$\|\Phi_n^{(k)} - \Phi_n'^{(k)}\|_\infty = |\phi_{n+r_0}^{(k)} - \phi_{n+r_0}'^{(k)}| \leq \widehat{L}_k |\phi_{n+r_0}^{(0)} - \phi_{n+r_0}'^{(0)}| \leq \widehat{L}_k \|\Phi_n^{(0)} - \Phi_n'^{(0)}\|_\infty. \quad \square$$

Remark. *Since we are dealing with finite-dimensional spaces, all norms are equivalent, and the inequality (8) holds for any norm with an adapted coefficient $\widehat{L}_k$ depending on R .* $\blacksquare$

7.1 Convergence of the iterative procedure for structural scheme $\text{Skm}[1, \mathbb{R}]$

Following Lemma 7.1, we assume that the approximations in vectors $\Phi_n^{(0)}[\ell]$, $\ell \geq 0$ remain bounded in the ball $B(M_0)$. Hence, the vectors $\Phi_n^{(1)}[\ell]$ are bounded and the Lipschitz inequality (8) holds between two successive approximations, that is,

$$\|\Phi_n^{(1)}[\ell + 1] - \Phi_n^{(1)}[\ell]\| \leq \widehat{L}_1 \|\Phi_n^{(0)}[\ell + 1] - \Phi_n^{(0)}[\ell]\|.$$

Proposition 7.4. *Let Δt be small enough such that*

$$\Delta t \widehat{L}_1 \|B^{(1)}\| < 1, \quad B^{(1)} = \left(A^{(0)}\right)^{-1} A^{(1)}. \quad (9)$$

Then, the fixed-point is a contraction, and the successive numerical approximations converge to the solution of the non-linear system derived from the structural scheme $\text{Skm}[1, \mathbb{R}]$.

Proof. We note that

$$A^{(0)}(\Phi_n^{(0)}[\ell + 1] - \Phi_n^{(0)}[\ell]) = -\Delta t A^{(1)}(\Phi_n^{(1)}[\ell + 1] - \Phi_n^{(1)}[\ell]),$$

and thus, applying the Lipschitz condition (8), yields

$$\begin{aligned} \|\Phi_n^{(0)}[\ell + 1] - \Phi_n^{(0)}[\ell]\| &\leq \Delta t \|B^{(1)}\| \|\Phi_n^{(1)}[\ell + 1] - \Phi_n^{(1)}[\ell]\| \\ &\leq \Delta t \|B^{(1)}\| \|\mathbf{df}^{(0)}[\Phi_n^{(0)}[\ell], T_n] - \mathbf{df}^{(0)}[\Phi_n^{(0)}[\ell - 1], T_n]\| \\ &\leq \Delta t \|B^{(1)}\| \widehat{L}_1 \|\Phi_n^{(0)}[\ell] - \Phi_n^{(0)}[\ell - 1]\|, \end{aligned}$$

which provides the amplification coefficient $\chi_1 = \Delta t \widehat{L}_1 \|B^{(1)}\|$. Under the CFL-like condition (9), we get a geometric series with coefficient $\chi_1 < 1$, hence the convergence of the fixed-point is guaranteed. $\square$

7.2 Convergence of the iterative procedure for structural scheme $\text{Skm}[2, \mathbb{R}]$

Once again, we use Lemma 7.1. We assume that the approximations in vectors $\Phi_n^{(0)}[\ell]$, $\ell \geq 0$ remain bounded in the ball $B(M_0)$. Hence, vectors $\Phi_n^{(1)}[\ell]$ and $\Phi_n^{(2)}[\ell]$ are bounded and the Lipschitz inequalities (8) hold between two successive approximations, that is

$$\begin{aligned} \|\Phi_n^{(1)}[\ell + 1] - \Phi_n^{(1)}[\ell]\| &\leq \widehat{L}_1 \|\Phi_n^{(0)}[\ell + 1] - \Phi_n^{(0)}[\ell]\|, \\ \|\Phi_n^{(2)}[\ell + 1] - \Phi_n^{(2)}[\ell]\| &\leq \widehat{L}_2 \|\Phi_n^{(0)}[\ell + 1] - \Phi_n^{(0)}[\ell]\|. \end{aligned}$$

Proposition 7.5. *Let Δt be small enough such that*

$$\Delta t \widehat{L}_1 \|B^{(1)}\| + \Delta t^2 \widehat{L}_2 \|B^{(2)}\| < 1, \quad B^{(1)} = \left(A^{(0)}\right)^{-1} A^{(1)}, \quad B^{(2)} = \left(A^{(0)}\right)^{-1} A^{(2)}. \quad (10)$$

Then, the fixed-point is a contraction, and the successive numerical approximations converge to the solution of the non-linear system derived from the structural scheme $\text{Skm}[2, \mathbb{R}]$.

Proof. The proof is very similar to the case of the structural scheme $\text{Skm}[1, \mathbb{R}]$. We note that

$$\begin{aligned} A^{(0)}(\Phi_n^{(0)}[\ell + 1] - \Phi_n^{(0)}[\ell]) &= -\Delta t A^{(1)}(\Phi_n^{(1)}[\ell + 1] - \Phi_n^{(1)}[\ell]) \\ &\quad - \Delta t A^{(2)}(\Phi_n^{(2)}[\ell + 1] - \Phi_n^{(2)}[\ell]), \end{aligned}$$

and thus, applying the Lipschitz condition (8), yields

$$\begin{aligned} \|\Phi_n^{(0)}[\ell + 1] - \Phi_n^{(0)}[\ell]\| &\leq \Delta t \|B^{(1)}\| \|\Phi_n^{(1)}[\ell + 1] - \Phi_n^{(1)}[\ell]\| + \\ &\quad \Delta t^2 \|B^{(2)}\| \|\Phi_n^{(2)}[\ell + 1] - \Phi_n^{(2)}[\ell]\| \\ &\leq \Delta t \|B^{(1)}\| \widehat{L}_1 \|\Phi_n^{(0)}[\ell] - \Phi_n^{(0)}[\ell - 1]\| + \\ &\quad \Delta t^2 \|B^{(2)}\| \widehat{L}_2 \|\Phi_n^{(0)}[\ell] - \Phi_n^{(0)}[\ell - 1]\|, \end{aligned}$$

which provides the amplification coefficient

$$\chi_2 = \Delta t \widehat{L}_1 \left\| \mathbf{B}^{(1)} \right\| + \Delta t^2 \widehat{L}_2 \left\| \mathbf{B}^{(2)} \right\|,$$

and the convergence of the fixed-point is guaranteed under the CFL-like condition (10). $\square$

Remark. Notice that the amplification coefficient is composed of a linear term with respect to Δt , corresponding to $k = 1$, and a quadratic term with respect to Δt^2 , corresponding to $k = 2$. Therefore, the restriction is mainly attributed to the first derivative. In particular, if $\Delta t \widehat{L}_2 \leq \widehat{L}_1$ and $\Delta t \widehat{L}_1 \leq 1/2$, we obtain the contraction of the fixed-point. $\blacksquare$

Remark. The analysis of the convergence of successive iterations for $K > 2$ follows the same ideas as for the structural schemes $\text{Skm}[1, R]$ and $\text{Skm}[2, R]$, and derives from the uniform boundness of the approximations and the Lipschitz property for $f(z, t)$ and its derivatives under a CFL-like condition. A generalisation of the condition for the structural scheme $\text{Skm}[K, R]$ then reads

$$\chi_K = \sum_{k=1}^K \Delta t^k \widehat{L}_k \left\| \mathbf{B}^{(k)} \right\| \leq 1, \quad \mathbf{B}^k = \left(\mathbf{A}^{(0)} \right)^{-1} \mathbf{A}^{(k)}$$

assuming that matrix $\mathbf{A}^{(0)}$ is non-singular. $\blacksquare$

Remark. Note that the CFL-like conditions come only from the stability of the fixed-point algorithm. As we shall see in the sequel, the non-linear schemes are unconditionally stable, but the proposed iterative algorithms for computing the approximate solution may require conditions to guarantee convergence. $\blacksquare$

8 Consistency error and convergence order

The approximations $\phi_{n+r}^{(k)} \approx \phi^{(k)}(t_{n+r})$ are the solution of a non-linear system coupling structural and physical equations. For the consistency error and convergence order analysis, let us denote

$$\begin{aligned} \Phi_n^{(k)} &= \left(\phi_{n+1}^{(k)}, \dots, \phi_{n+R}^{(k)} \right)^t \in \mathbb{R}^R, \quad \overline{\Phi}_n^{(k)} = \left(\overline{\phi}_{n+1}^{(k)}, \dots, \overline{\phi}_{n+R}^{(k)} \right)^t \in \mathbb{R}^R, \\ \Psi_n^K &= \left(\phi_n^{(0)}, \dots, \phi_n^{(K)} \right)^t \in \mathbb{R}^{K+1}, \quad \overline{\Psi}_n^K = \left(\overline{\phi}_n^{(0)}, \dots, \overline{\phi}_n^{(K)} \right)^t \in \mathbb{R}^{K+1}, \\ \mathbf{A}^{(k)} &= \left(a_{k,r}^s \right)_{r,s=1}^R \in \mathbb{R}^{R \times R}, \quad \widehat{\mathbf{A}}_n^K = \left(a_{k,0}^s \Delta t^k \right)_{k \in \llbracket 0, K \rrbracket}^{s \in \llbracket 1, R \rrbracket} \in \mathbb{R}^{R \times (K+1)}, \end{aligned}$$

with $\overline{\phi}_{n+r}^{(k)} = \phi^{(k)}(t_{n+r})$, $k \in \llbracket 0, K \rrbracket$, $r \in \llbracket 1, R \rrbracket$, the exact solution.

Lemma 8.1. Adopting a column vector expression of the matrix $\widehat{\mathbf{A}}_n^K = [\widehat{\mathbf{a}}_0, \Delta t \widehat{\mathbf{a}}_1, \dots, \Delta t^K \widehat{\mathbf{a}}_K]$ with $\widehat{\mathbf{a}}_k \in \mathbb{R}^R$, $k \in \llbracket 0, K \rrbracket$, we have the property

$$\left(\mathbf{A}^{(0)} \right)^{-1} \widehat{\mathbf{a}}_0 = -\mathbf{e} \in \mathbb{R}^R, \quad (11)$$

where $\mathbf{e} = [1, 1, \dots, 1]^t$.

Proof. The structural equations are exact for the constant polynomial function $\pi_1(t; t_n) = 1$. Therefore, taking $\Phi_n^{(0)} = \mathbf{e}^t \in \mathbb{R}^R$, then $\Phi_n^{(k)} = [0, 0, \dots, 0]^t \in \mathbb{R}^R$, $k \in \llbracket 1, K \rrbracket$, and $\Psi_n^K = [1, 0, \dots, 0]^t \in \mathbb{R}^{K+1}$. We deduce the compatibility relation

$$\mathbf{A}^{(0)} \Phi_n^{(0)} + \widehat{\mathbf{A}}_n^K \Psi_n^K = 0 \implies \mathbf{A}^{(0)} \mathbf{e} + \widehat{\mathbf{a}}_0 = 0.$$

Assuming that the matrix $\mathbf{A}^{(0)}$ is nonsingular, we obtain the relation (11). $\square$

8.1 Consistency error of the structural equations

For the structural scheme $\text{Skm}[K, R]$, the R structural equations in compact form read

$$\sum_{k=0}^K \Delta t^k \mathbf{A}^{(k)} \Phi_n^{(k)} + \widehat{\mathbf{A}}_n^K \Psi_n^K = 0$$

that is

$$0 = \sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s \Delta t^k \phi_{n+r}^{(k)}. \quad (12)$$

These equations are satisfied for any polynomial of degree $K(R+1)$. On the one hand, using the Taylor expansion, for $h > 0$, we have

$$\phi(t_n + h) = \pi_\phi(h; t_n) + \frac{h^{K(R+1)+1}}{(K(R+1)+1)!} \phi^{(K(R+1)+1)}(t_n + \theta_0 h), \theta_0 \in [0, 1],$$

where $\pi_\phi(h; t_n)$ is the Taylor polynomial of degree $K(R+1)$. Moreover, the Taylor expansion for the k -th derivative with respect to the variable h is

$$\phi^{(k)}(t_n + h) = \pi_\phi^{(k)}(h; t_n) + \frac{h^{K(R+1)+1-k}}{(K(R+1)+1-k)!} \phi^{(K(R+1)+1)}(t_n + \theta_k h), \theta_k \in [0, 1].$$

Consistency errors correspond to the error when the approximate solution is substituted with the exact solution in the structural equations.

Proposition 8.2. Let ε_n^s be the consistency error for the structural equation $SE_{KR}^R(s)$, $s \in \llbracket 1, R \rrbracket$, given by

$$\varepsilon_n^s = \sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s \Delta t^k \phi^{(k)}(t_n + r \Delta t) = \sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s \Delta t^k \bar{\phi}_{n+r}^{(k)}. \quad (13)$$

Then

$$|\varepsilon_n^s| \leq C_{K,R}^s \Delta t^{K(R+1)+1} \quad (14)$$

with $C_{K,R}^s$ a real number depending on $\max_{[0,T]} |\phi^{(K(R+1)+1)}|$ but independent of Δt .

Proof. Using that the relation holds exactly for the Taylor polynomial $\pi(h; t_n)$ as a function of h , we deduce, with $h = \Delta t, 2\Delta t, \dots, R\Delta t$, that

$$\begin{aligned} \varepsilon_n^s &= \sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s \Delta t^k \frac{(r \Delta t)^{K(R+1)+1-k}}{(K(R+1)+1-k)!} \phi^{(K(R+1)+1)}(t_n + \theta_{r,k} r \Delta t) \\ &= \Delta t^{K(R+1)+1} \sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s \frac{r^{K(R+1)+1-k}}{(K(R+1)+1-k)!} \phi^{(K(R+1)+1)}(t_n + \theta_{r,k} r \Delta t). \end{aligned}$$

Then

$$|\varepsilon_n^s| \leq \Delta t^{K(R+1)+1} L_{K,R} \sum_{k=0}^K \sum_{r=0}^R |a_{k,r}^s| \frac{r^{K(R+1)+1-k}}{(K(R+1)+1-k)!},$$

with

$$L_{K,R} = \max_{t \in [0,T]} |\phi^{(K(R+1)+1)}(t)|.$$

Setting

$$C_{K,R}^s = L_{K,R} \sum_{k=0}^K \sum_{r=0}^R |a_{k,r}^s| \frac{r^{K(R+1)+1-k}}{(K(R+1)+1-k)!},$$

provides the relation (14). □

If $\mathcal{E}_n = (\varepsilon_n^s)_{s \in \llbracket 1, R \rrbracket}$ stands for the vector that gathers the consistency errors, then we have

$$\|\mathcal{E}_n\|_\infty = C_{K,R} \Delta t^{K(R+1)+1}$$

with $C_{K,R} = \max_{s \in \llbracket 1, R \rrbracket} C_{K,R}^s$.

8.2 Convergence order of the structural scheme $\text{Skm}[K, R]$

We adapt the Grönwall lemma [99] for the structural scheme $\text{Skm}[K, R]$.

Lemma 8.3. *Let $(e_n)_{n=0}^N$ be a sequence of real numbers with $N = J \times R$ (J blocks of size R) and assume that, for $\Delta t \leq \overline{\Delta t}$ with $\overline{\Delta t}$ to be fixed further, there exist non-negative real numbers $b^0, \dots, b^K$, independent of Δt , such that for $j \in \llbracket 0, J-1 \rrbracket$*

$$|e_{(j+1)R}| \leq |e_{jR}| \left(1 + \sum_{k=1}^K \Delta t^k b^k \right) + b^0 \Delta t^{P+1},$$

for a certain $P \in \mathbb{N}$. Then for any $j \in \llbracket 1, J \rrbracket$, we have the estimate

$$|e_{jR}| \leq \tilde{C} \Delta t^P + \hat{C} |e_0|, \quad (15)$$

where $\tilde{C}$ and $\hat{C}$ are constants independent of Δt .

Proof. Given $\overline{\Delta t}$, which shall be fixed later, we define

$$C = \sum_{k=1}^K \overline{\Delta t}^{k-1} b^k.$$

From the assumption and noting that $\Delta t / \overline{\Delta t} \leq 1$, we obtain the inequality

$$|e_{(j+1)R}| \leq |e_{jR}| (1 + C \Delta t) + b^0 \Delta t^{P+1}$$

for any $j \in \llbracket 0, J-1 \rrbracket$. By induction, it yields

$$\begin{aligned} |e_{(j+1)R}| &\leq (1 + C \Delta t)^j |e_0| + b^0 \Delta t^{P+1} \sum_{\sigma=0}^j (1 + C \Delta t)^\sigma \\ &\leq \exp(Cj \Delta t) |e_0| + b^0 \Delta t^P \sum_{\sigma=0}^j \Delta t \exp(C\sigma \Delta t) \\ &\leq \exp(CT) |e_0| + b^0 \Delta t^P \int_0^T \exp(C\xi) d\xi \\ &\leq \exp(CT) |e_0| + b^0 \Delta t^P \frac{\exp(CT) - 1}{C}. \end{aligned}$$

Therefore, we deduce the inequality (15) with

$$\tilde{C} = b^0 \frac{\exp(CT) - 1}{C}, \quad \hat{C} = \exp(CT).$$

□

We define the *a priori* error (or convergence error) as the quantity $e_n^{(k)} = \overline{\phi}_n^{(k)} - \phi_n^{(k)}$ for $n \in \llbracket 0, N \rrbracket$ and $k \in \llbracket 0, K \rrbracket$, and $\mathbf{E}_n^{(k)} = [e_{n+1}^{(k)}, \dots, e_{n+R}^{(k)}]^\top$.

Proposition 8.4. *There exists $\overline{\Delta t}$ such that, for $\Delta t \leq \overline{\Delta t}$, we have the estimate*

$$|e_{jR}^{(0)}| \leq \overline{C} \Delta t^{K(R+1)}, \quad (16)$$

for any $j \in \llbracket 1, J \rrbracket$, $jR \leq N$, where $\overline{C}$ is a real number independent of Δt .

Proof. The difference between relations (13) and (12) give

$$\varepsilon_n^s = \sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s \Delta t^k (\overline{\phi}_{n+r}^{(k)} - \phi_{n+r}^{(k)}) = \sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s \Delta t^k e_{n+r}^{(k)},$$

which can be rewritten in the compact form as

$$\mathcal{E}_n = \sum_{k=0}^K \Delta t^k \mathbf{A}^{(k)} \mathbf{E}_n^{(k)} + \sum_{k=0}^K \hat{\mathbf{a}}_k \Delta t^k e_n^{(k)}.$$

Using Lemma 8.1, we rewrite the consistency error in the form

$$\mathbf{E}_n^{(0)} = e_n^{(0)} \mathbf{e} - \sum_{k=1}^K \Delta t^k \mathbf{B}^k \mathbf{E}_n^{(k)} - \sum_{k=1}^K \Delta t^k \mathbf{b}^k e_n^{(k)} + \left(\mathbf{A}^{(0)} \right)^{-1} \mathcal{E}_n$$

with $\mathbf{B}^k = \left(\mathbf{A}^{(0)} \right)^{-1} \mathbf{A}^{(k)}$ and $\mathbf{b}^k = \left(\mathbf{A}^{(0)} \right)^{-1} \hat{\mathbf{a}}_k$.

Using the L^∞ -norm $\| \cdot \|_\infty$ and the inductive matrix norm (also denoted $\| \cdot \|_\infty$), we have

$$\begin{aligned} \|\mathbf{E}_n^{(0)}\|_\infty &\leq |e_n^{(0)}| + \sum_{k=1}^K \Delta t^k \|\mathbf{B}^k\|_\infty \|\mathbf{E}_n^{(k)}\|_\infty \\ &\quad + \sum_{k=1}^K \Delta t^k \|\mathbf{b}^k\|_\infty |e_n^{(k)}| + \left\| \left(\mathbf{A}^{(0)} \right)^{-1} \right\|_\infty \|\mathcal{E}_n\|_\infty. \end{aligned}$$

From inequalities (8), we then deduce

$$\|\mathbf{E}_n^{(k)}\|_\infty \leq \hat{L}_k \|\mathbf{E}_n^{(0)}\|_\infty \text{ and } |e_n^{(k)}| \leq \hat{L}_k |e_n^{(0)}|.$$

Using the consistency error estimate, we obtain

$$C_f \Delta t \|\mathbf{E}_n^{(0)}\|_\infty \leq C_b \Delta t |e_n^{(0)}| + \left\| \left(\mathbf{A}^{(0)} \right)^{-1} \right\|_\infty C_{K,R} \Delta t^{K(R+1)+1}, \quad (17)$$

where

$$C_f \Delta t = 1 - \sum_{k=1}^K \Delta t^k \hat{L}_k \|\mathbf{B}^k\|_\infty, \quad C_b \Delta t = 1 + \sum_{k=1}^K \Delta t^k \hat{L}_k \|\mathbf{b}^k\|_\infty.$$

Due to the inequality $|e_{n+R}^{(0)}| \leq \|\mathbf{E}_n^{(0)}\|_\infty$, we deduce

$$|e_{n+R}^{(0)}| \leq \frac{C_b \Delta t}{C_f \Delta t} |e_n^{(0)}| + \left\| \left(\mathbf{A}^{(0)} \right)^{-1} \right\|_\infty \frac{C_{K,R}}{C_f \Delta t} \Delta t^{K(R+1)+1}.$$

We now define $\bar{\Delta}t > 0$ such that $C_f \bar{\Delta}t = 1/2$. Consequently, for any $\Delta t < \bar{\Delta}t$, we have $C_f \Delta t \geq 1/2$, and, for $\Delta t \leq \bar{\Delta}t$, we obtain the inequality

$$\frac{C_b \Delta t}{C_f \Delta t} = 1 + \frac{C_b \Delta t - C_f \Delta t}{C_f \Delta t} \leq 1 + 2 \sum_{k=1}^K \Delta t^k \hat{L}_k \left(\|\mathbf{B}^k\|_\infty + \|\mathbf{b}^k\|_\infty \right)$$

We finally conclude that, for $\Delta t \leq \bar{\Delta}t$, we have

$$|e_{n+R}^{(0)}| \leq |e_n^{(0)}| \left(1 + \sum_{k=1}^K \Delta t^k b^k \right) + b^0 \Delta t^{K(R+1)+1},$$

with $b^k = 2\hat{L}_k \left(\|\mathbf{B}^k\|_\infty + \|\mathbf{b}^k\|_\infty \right)$ and $b^0 = 2C_{K,R}$, where the non-negative real numbers $b^0, \dots, b^K$ do not depend on Δt .

Since we treat the differential equation using blocks of size R , we have $n = jR$ and from Lemma 8.3, the relation (15) holds with $P = K(R+1)$, that is

$$|e_{jR}^{(0)}| \leq \tilde{C} \Delta t^{K(R+1)} + \hat{C} |e_0^{(0)}|.$$

If we use the exact solution at time $t = 0$ for the initial condition, that is $e_0^{(0)} = 0$, we deduce that $\bar{C} = \tilde{C}$ and the error is of order $\Delta t^{K(R+1)}$. $\square$

Theorem 8.5. Let $n = jR$, $j \in \llbracket 1, J \rrbracket$. Under the assumption of Proposition 8.4, there exists a real value C independent of Δt , r , j , and k such that, if $\Delta t \leq \overline{\Delta t}$, then

$$|e_{n+r}^{(k)}| \leq C \Delta t^{K(R+1)}. \quad (18)$$

The scheme is of order $K(R+1)$ at any point and for any derivative.

Proof. From relation (17) and $|e_{n+r}^{(0)}| \leq \|E_n^{(0)}\|_\infty$, we deduce

$$|e_{n+r}^{(0)}| \leq |e_n^{(0)}| \left(1 + \sum_{k=1}^K \Delta t^k b^k \right) + b^0 \Delta t^{K(R+1)+1},$$

and, with the estimate (16), we deduce the zero-derivative convergence order

$$|e_{n+r}^{(0)}| \leq \tilde{C} \Delta t^{K(R+1)} \left(1 + \sum_{k=1}^K \Delta t^k b^k \right) + b^0 \Delta t^{K(R+1)+1} \leq C^0 \Delta t^{K(R+1)},$$

where C^0 does not depend on Δt . On the other hand, using the Lipschitz property of Lemma 7.2, we get $|e_{n+r}^{(k)}| \leq \hat{L}_k |e_{n+r}^{(0)}| \leq C^k \Delta t^{K(R+1)}$, with $C^k = \hat{L}_k C^0$. Let $C = \max C^k$ over $k \in \llbracket 0, K \rrbracket$, and we deduce the convergence errors (18). $\square$

9 Linear stability

The linear stability of the structural schemes is provided by the von Neumann stability analysis. We compute the growth factor of the Fourier modes and check the magnitude of the block approximations. The analysis is based on the solution of the linear ODE $\phi^{(1)} = \lambda \phi^{(0)}$ with $\lambda \in \mathbb{C}$, that is, $f(z, t) = \lambda z$.

9.1 Linear stability condition

The first physical equation $\text{PE}_1(n+r)$ reads $\phi_{n+r}^{(1)} = \lambda \phi_{n+r}^{(0)}$, $r \in \llbracket 0, T \rrbracket$, and the k -th physical equation $\text{PE}_k(n+r)$ is given by $\phi_{n+r}^{(k)} = \lambda^k \phi_{n+r}^{(0)}$. Substituting $\phi_{n+r}^{(k)}$ with $\lambda^k \phi_{n+r}^{(0)}$, $r \in \llbracket 0, R \rrbracket$, into the R structural equations, provides the system

$$\sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s (\lambda \Delta t)^k \phi_{n+r}^{(0)} = 0, \quad s \in \llbracket 1, R \rrbracket. \quad (19)$$

The relation (19) is rewritten with the variable $z = \lambda \Delta t$, yielding

$$\sum_{r=1}^R \left(\sum_{k=0}^K a_{k,r}^s z^k \right) \phi_{n+r}^{(0)} = - \left(\sum_{k=0}^K a_{k,0}^s z^k \right) \phi_n^{(0)}.$$

Denoting

$$g_{s,r}(z) = \sum_{k=0}^K a_{k,r}^s z^k, \quad \mathbf{G}(z) = \left(g_{s,r}(z) \right)_{s,r=1}^R, \quad \mathbf{b}(z) = \left(g_{s,0}(z) \right)_{s=1}^R,$$

the relation (19) can be expressed in the following compact matrix form

$$\mathbf{G}(z) \Phi_n^{(0)} = -\mathbf{b}(z) \phi_n^{(0)} \implies \Phi_n^{(0)} = \chi(z) \phi_n^{(0)}, \quad (20)$$

where $\chi(z) = -\mathbf{G}^{-1}(z) \mathbf{b}(z) \in \mathbb{R}^R$ is the vector whose entries are $\chi_r(z)$, $r \in \llbracket 1, R \rrbracket$, representing the amplification function. For stability, each entry must satisfy $|\chi_r(z)| \leq 1$.

Because the computation of the next block only uses the vector Φ_{n+R} , our analysis focuses on the amplification function $\chi_R(z)$ as a function of the parameter z , derived from system (20).

We introduce an adaptation of the A-stability for the block structural method.

Definition 9.1. The scheme is A-stable if $|\chi_R(z)| \leq 1$.

Since there are multiple ways to choose the set of R structural equations, we establish that the stability condition does not depend on the particular basis chosen for the kernel.

Proposition 9.1. *Let $\mathbf{A} = [a^s]_{R \times M}$ and $\tilde{\mathbf{A}} = [\tilde{a}^s]_{R \times M}$ be two basis of the Kernel $\mathcal{K}^R$ given in Eq. (5). Then*

$$\mathbf{G}(z)\Phi_n^{(0)} = -\mathbf{b}(z)\phi_n^{(0)} \quad \text{and} \quad \tilde{\mathbf{G}}(z)\tilde{\Phi}_n^{(0)} = -\tilde{\mathbf{b}}(z)\tilde{\phi}_n^{(0)},$$

where, $\tilde{\mathbf{G}}(z)$ and $\tilde{\mathbf{b}}(z)$ refer to the matrix and vector defined as in Eq. (20), but associated with the alternative basis $\tilde{a}^s$. The amplification vector $\chi(z)$ therefore remains the same regardless of the basis chosen, showing that stability is determined only by properties of the kernel subspace.

Proof. Let $\tilde{\mathbf{A}}$ and $\mathbf{A}$ be two different kernel bases, as in Eq. (5). We have $\tilde{\mathbf{A}} = \Xi \mathbf{A}$ where Ξ is a nonsingular $R \times R$ matrix that changes the basis, that is, $\tilde{\mathbf{G}}(z) = \Xi \mathbf{G}(z)$ and $\tilde{\mathbf{b}}(z) = \Xi \mathbf{b}(z)$. From Eq. (20), we have $\Phi_n^{(0)} = \chi(z)\phi_n^{(0)}$ and, using the alternative basis, $\tilde{\mathbf{G}}(z)\tilde{\Phi}_n^{(0)} = -\tilde{\mathbf{b}}(z)\tilde{\phi}_n^{(0)}$, hence

$$\tilde{\chi}(z) = -\tilde{\mathbf{G}}^{-1}(z)\tilde{\mathbf{b}}(z) = -\left(\Xi \mathbf{G}(z)\right)^{-1} \Xi \mathbf{b}(z) = -\mathbf{G}^{-1}(z)(\Xi^{-1}\Xi)\mathbf{b}(z) = \chi(z).$$

As a result, all amplification functions χ_r , $r \in \llbracket 1, R \rrbracket$, are independent of the kernel basis chosen. Consequently, the stability condition $|\chi_R(z)| \leq 1$ for the approximation at time t_{n+R} is unaffected by the choice of basis. $\square$

To assess stability, we propose a new expression for the amplification function.

Proposition 9.2. *The amplification coefficient at time t_{n+R} is given by*

$$\chi_R(z) = \frac{\det \left[\mathbf{G}(z)(\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t) - \mathbf{b}(z) \mathbf{e}_R^t \right]}{\det [\mathbf{G}(z)]}, \quad (21)$$

where $\mathbf{I}_d$ is the identity matrix of size $R \times R$ and $\mathbf{e}_R$ is the last canonic column vector. In particular, $\chi_R(z)$ is a rational function.

Proof. For simplicity, we suppress the explicit z dependence and apply Cramer's rule to the system (20) to determine the last unknown. Consider the upper triangular matrix $\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t + \Phi_n^{(0)} \mathbf{e}_R^t$, which is the identity matrix with its R -th column replaced by the vector $\Phi_n^{(0)}$. Multiplying matrix $\mathbf{G}$ on the left side, yields

$$\mathbf{G} \left(\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t + \Phi_n^{(0)} \mathbf{e}_R^t \right) = \mathbf{G} \left(\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t \right) - \mathbf{b} \phi_n^{(0)} \mathbf{e}_R^t.$$

On the one hand, $\det [\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t + \Phi_n^{(0)} \mathbf{e}_R^t] = \phi_{n+R}^{(0)}$, thus

$$\det \left[\mathbf{G} \left(\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t + \Phi_n^{(0)} \mathbf{e}_R^t \right) \right] = \det [\mathbf{G}] \phi_{n+R}^{(0)},$$

which, after some algebraic manipulations, provides the expression

$$\det \left[\mathbf{G} \left(\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t + \Phi_n^{(0)} \mathbf{e}_R^t \right) \right] = \det \left[\mathbf{G} \left(\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t \right) - \mathbf{b} \phi_n^{(0)} \mathbf{e}_R^t \right] = \det \left[\mathbf{G} \left(\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t \right) - \mathbf{b} \mathbf{e}_R^t \right] \phi_n^{(0)},$$

and finally

$$\phi_{n+R}^{(0)}(z) = \frac{\det \left[\mathbf{G}(z) \left(\mathbf{I}_d - \mathbf{e}_R \mathbf{e}_R^t \right) - \mathbf{b}(z) \mathbf{e}_R^t \right]}{\det [\mathbf{G}(z)]} \phi_n^{(0)},$$

and the relation (21) holds. Since the entries of the matrix $\mathbf{G}$ and vector $\mathbf{b}$ are polynomials of z of degree K with real coefficients, the amplification function χ_R is therefore a rational function with real coefficients. $\square$

9.2 Criteria for the unconditional stability

Next, we introduce several tools for analysing the stability of the structural scheme $\text{Skm}[K, R]$. This approach assumes that the transfer function has the form $\chi_R(z) = p(z)/p(-z)$, where $p(z)$ is a polynomial. With the Routh-Hurwitz criterion [100], we thus demonstrate that if all roots of $p(z)$ have negative real parts, the scheme is A-stable.

Lemma 9.3. *Assume that the transfer function associated with the structural scheme $\text{Skm}[K, R]$ is given by $\chi_R(z) = p(z)/p(-z)$, where $p(z)$ is a polynomial in $\mathbb{C}$ in which all roots have a negative real part. Then, the structural scheme $\text{Skm}[K, R]$ is A-stable.*

Proof. If all roots of $p(z)$ are in the left-half of the complex plane, then all roots of $p(-z)$ lie in the right-half. Thus, $\chi_R(z)$ is analytic in the left-half plane. According to the Maximum Modulus Theorem, the maximum value of $|\chi_R|$ occurs at the boundary of the semicircular region bounded by the imaginary axis from $-\rho$ to ρ and the semicircle $z = \rho \exp(i\theta)$, $\theta \in [\pi/2, 3\pi/2]$. Since $\chi_R(z) = p(z)/p(-z)$ then, for any $\theta \in [\pi/2, 3\pi/2]$, $\lim_{\rho \rightarrow +\infty} |\chi_R(z)| = 1$. On the other hand, for imaginary numbers $z = iy$, for all $y \in \mathbb{R}$, and taking advantage of the fact that the polynomial coefficients are real numbers, we have

$$|\chi_R(iy)| = \left| \frac{p(iy)}{p(-iy)} \right| = \left| \frac{p(iy)}{\overline{p(iy)}} \right| = 1.$$

We then conclude that $|\chi_R(z)| \leq 1$ for $\operatorname{Re} z \leq 0$, that is, the structural scheme $\operatorname{Skm}[K, R]$ is A-stable. $\square$

In summary, Lemma (9.3) stipulates that unconditional stability requires: (1) $\chi_R(z) = p(z)/p(-z)$, and (2) all zeros of $p(z)$ are in the left-half of the complex plane. To check the second condition, we demonstrate that $p(z)$ is Hurwitz stable [101], which is verifiable via the Routh-Hurwitz criterion [101, 102] as detailed below.

Definition 9.2. Let

$$p(z) = a_I z^I + a_{I-1} z^{I-1} + \cdots + a_1 z + a_0 \quad (22)$$

be a polynomial with real coefficients. The $I \times I$ matrix

$$H = \begin{bmatrix} a_{I-1} & a_{I-3} & a_{I-5} & \cdots & \cdots & \cdots & 0 & 0 & 0 \\ a_I & a_{I-2} & a_{I-4} & \cdots & \cdots & \cdots & 0 & 0 & 0 \\ 0 & a_{I-1} & a_{I-3} & \cdots & \cdots & \cdots & 0 & 0 & 0 \\ 0 & a_I & a_{I-2} & \cdots & \cdots & \cdots & 0 & 0 & 0 \\ 0 & 0 & a_{I-1} & \cdots & \cdots & \cdots & a_0 & 0 & 0 \\ 0 & 0 & a_I & \cdots & \cdots & \cdots & a_1 & 0 & 0 \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & a_2 & a_0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & a_3 & a_1 & 0 \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & a_4 & a_2 & a_0 \end{bmatrix}$$

is termed the Hurwitz matrix associated with $p(z)$.

Definition 9.3. The polynomial $p(z)$ is called the Hurwitz polynomial if $\operatorname{Re} z_i < 0$ for all $i \in \llbracket 1, I \rrbracket$, where z_i are the complex roots of $p(z)$.

The Routh-Hurwitz criterion is given in the following theorem (see [100]).

Theorem 9.4. The polynomial $p(z)$ given in Eq. (22) is a Hurwitz polynomial if and only if the principal minors of the Hurwitz matrix H are strictly positive.

From Lemma 9.3 and the definition of a Hurwitz polynomial, we deduce the following theorem.

Theorem 9.5. Let $\chi_R(z) = p(z)/p(-z)$ be the transfer function at time t_{n+R} associated with the structural scheme $\operatorname{Skm}[K, R]$. If $p(z)$ is a Hurwitz polynomial in $\mathbb{C}$, then the structural scheme $\operatorname{Skm}[K, R]$ is A-stable.

There is no generic result showing that all structural schemes $\operatorname{Skm}[K, R]$ are unconditionally stable; we must check the A-stability of each scheme individually. To this end, the stability of several structural schemes $\operatorname{Skm}[K, R]$ has been studied by computation of the $\chi_R(z)$ function given in Tables 30, 31, and 32 in C. The expressions $\chi_R(z)$ of all these schemes have the form $\chi_R(z) = p(z)/p(-z)$, where $p(z)$ is a polynomial of real positive coefficients. Verification of the Routh-Hurwitz criterion is also given in C.

We summarise the results in the following corollary and provide, in the proof, the methodology for checking A-stability using Hurwitz polynomials.

Corollary 9.6. The structural schemes $\operatorname{Skm}[K, R]$, $K, R = 1, 2, 3$ are A-stable.

Proof. The methodology for providing evidence of A-stability is illustrated with the following cases.

- **Structural scheme $\operatorname{Skm}[1, 3]$.** The Hurwitz polynomial $p(z) = 3z^3 + 11z^2 + 18z + 12$ is a third-degree polynomial with all positive coefficients (see C), while the associated Hurwitz matrix reads

$$H_{13} = \begin{bmatrix} 11 & 12 & 0 \\ 3 & 18 & 0 \\ 0 & 11 & 12 \end{bmatrix}.$$

Calculations show that all principal minors m_i are strictly positive and, from Theorem 9.5, we conclude that the structural scheme $\operatorname{Skm}[1, 3]$ is A-stable.

- **Structural scheme** $\text{Skm}[2, 2]$. The Hurwitz polynomial $p(z) = z^4 + 9z^3 + 39z^2 + 90z + 90$ is a fourth-degree polynomial with all positive coefficients (see C), while the associated Hurwitz matrix reads

$$H_{22} = \begin{bmatrix} 9 & 90 & 0 & 0 \\ 1 & 39 & 90 & 0 \\ 0 & 9 & 90 & 0 \\ 0 & 1 & 39 & 90 \end{bmatrix}.$$

Calculations show that all principal minors m_i are strictly positive and, from Theorem 9.5, we conclude that the structural scheme $\text{Skm}[2, 2]$ is A-stable. □

Remark. Since increasing the values of K and R yields Hurwitz polynomials with cumbersome expressions, the above methodology is illustrated only for the structural schemes $\text{Skm}[1, 3]$ and $\text{Skm}[2, 2]$. ■

9.3 About the stability of other classes of structural method

The $\text{Skm}[K, R]$ schemes is a particular case of structural schemes. Indeed, Given a block of size R and considering a scheme based on K derivatives, we have $R(K + 1)$ unknowns that requires the same number of equations. The schemes we propose use K physical equations per node, thus RK physical equations, we complete with R structural equations. But other combinations are available. For instance we just use $K - 1$ physical equations per nodes and then we need $2R$ structural equations to close the system.

More generally, any combination between a set of physical equations and a set of structural equation provides a potential structural scheme but some combination are not consistent (not enough physical equations) and others are not very efficient.

Our conjecture is that a scheme that does not employ all the possible physical equations suffers of a reduction of the order and unconditional stability is lost. Several experiments (not presented here) show that we obtain a conditional stability condition that restrict the size of the time step. Such "weaker scheme" may be interesting if one does not want (or cannot) to compute all the relations $\text{df}^{(k)}[\cdot, \cdot]$, $k = 0, \dots, K - 1$.

10 Spectral resolution

Fourier analysis is the standard tool for assessing the spectral resolution of a compact method. The pioneer paper of Lele [2] provides an explicit relation for the first and second derivative approximations with respect to the local wave number, $\omega = 2\pi\kappa\Delta t$. More precisely, the analysis is based on two linear relations (two structural equations, indeed) that connect zero- and first-derivative approximations on one side and zero- and second-derivative approximations on the other, as follows.

- The first relation for the zero and first derivative approximations is in the form

$$\sum_{r=1}^R a_r^0 \phi_{n+r}^{(0)} + \sum_{r=1}^R a_r^1 \phi_{n+r}^{(1)} = 0.$$

By introducing the local modified wave number $\omega^{(1)}$ for the first derivative, we obtain an algebraic expression in terms of ω . In this context, we assume the modified wavenumber remains constant at each discrete time t_{n+r} .

- The second relation for the zero and second derivative approximations is in the form

$$\sum_{r=1}^R b_r^0 \phi_{n+r}^{(0)} + \sum_{r=1}^R b_r^2 \phi_{n+r}^{(2)} = 0,$$

that provides an explicit algebraic relation of the modified wave number $\omega^{(2)}$ as a function of ω , independent of the discrete time t_{n+r} .

- The extension to the k -th derivative is based on an implicit coupling of $\phi_r^{(k)}$ with $\phi_r^{(0)}$ [103] by introducing the modified wave number $\omega^{(k)}$. Once again, spectral analysis using von Neumann analysis is available, since we connect the function solely to the k -th derivative.

The Compact Combined Scheme (CCS) [52, 53, 54] uses the same approach but with a slight modification. More specifically, it combines the zero, first, and second derivative approximations at the same time but with two linearly independent equations, leading to a small 2×2 system, which provides the modified wave numbers $\omega^{(1)}$ and $\omega^{(2)}$ for the first and second derivatives as a function of ω , respectively.

10.1 A new notion of spectral resolution

Such an approach presents challenges for the structural equations. First, we want to assess the impact of the structural scheme over the entire block, particularly at the last time step t_{n+R} relative to the initial time step t_n . Second, when $K \neq R$, there is no square, non-singular system to provide a direct solution for $\omega^{(1)}(\theta)$ up to $\omega^{(K)}(\theta)$. Finally, standard Fourier analysis assumes stationarity, so the phase shifts of the first and second derivatives are invariant with respect to t_n , which is not the case when dealing with an initial condition problem where we prescribe all the derivatives at $t = 0$.

We propose a new approach for assessing spectral resolution tailored to the structural scheme $\text{Skm}[K, R]$, specifically addressing the coupling between structural and physical equations. For this, we replace the fundamental solution $\exp(i2\pi\kappa t)$ with the equation $\phi' = i2\pi\kappa\phi$. Then, we connect all the derivatives to the solution thanks to the physical equations as

$$\phi^{(k)} = (i2\pi\kappa)^k \phi^{(0)}, \quad k \in \llbracket 1, K \rrbracket.$$

As a result, the R structural equations yield an R -dimensional linear system as previously defined in Eq. (20), for the specific case $z = \lambda\Delta t$ with $\lambda = i2\pi\kappa$. To analyse diffusion and dispersion in the structural schemes $\text{Skm}[K, R]$, we set the initial block condition at t_n to the exact solution $\phi_n^{(0)} = \exp(i2\pi\kappa t_n)$. The resulting transfer function, $\chi_r(i\omega)$, describes the diffusion and dispersion at time t_{n+r} . The exact solution at t_{n+r} is $\phi^{(0)}(t_{n+r}) = \exp(ir\omega) \exp(i2\pi\kappa t_n)$, so the deviation from the exact solution is given by the ratio between the approximation and the exact value at t_{n+r} :

$$\frac{\phi_{n+r}^{(0)}}{\phi^{(0)}(t_{n+r})} = \frac{\chi_r(i\omega)\phi_n^{(0)}}{\exp(ir\omega) \exp(i2\pi\kappa t_n)} = \chi_r(i\omega) \exp(-ir\omega).$$

Definition 10.1. *The diffusion of the scheme is given by the modulus of $\zeta_r(\omega) = \chi_r(i\omega) \exp(-ir\omega)$ (amplification factor), while the dispersion corresponds to its argument (phase deviation).*

Remark. *It is important to note that the function ζ_r also depends on the parameters R and K , thus we employ the notation $\zeta_r(\omega; K, R)$ when necessary (as in Section 10.4).* ■

Noting that the solution at time t_{n+R} will be used for the next block resolution, we focus our study on the case $r = R$. Of course, all intermediate steps can be analysed in a similar way.

Proposition 10.1. *If $\chi_R(z)$ is a rational function of the form $p(z)/p(-z)$ then we have $|\chi_R(i\omega)| = 1$ from which $|\zeta_R(\omega)| = 1$ for all $\omega > 0$. The schemes do not have diffusion at time t_{n+R} .*

Remark. *For $K, R = 1, 2, 3$, we have explicitly determined the amplification factor $\chi_R(z)$, case by case, as the quotient $p(z)/p(-z)$. We make the conjecture that this is always the case for any K and R .* ■

10.2 Phase error analysis

Thanks to the explicit expressions of the coefficients of the structural equations, we get an analytical expression of the transfer functions $\chi_R(z)$ as rational functions for a set of situations given in Tables 30, 31, and 32 in C. We then produce the polynomial expression as a function of ω , which we report in Table 20 and use to compute the $\zeta_R(\omega)$ function.

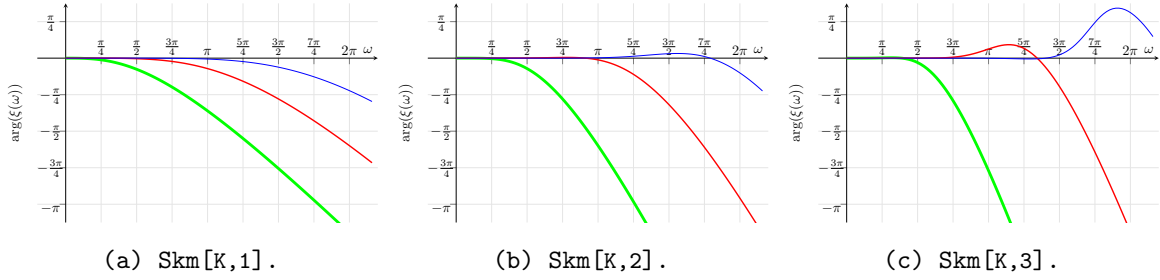
The phase deviation (in radians) as a function of the local wavelength ω is plotted in Figure 8 for $R = 1, 2, 3$ with $K = 1, 2, 3$. As expected, spectral resolution improves drastically with increasing K , whereas the impact of R is less obvious. We observe that large deviations occur for $\omega \geq \pi$ when $R = 3$. Particular attention is paid to the case with $R = 2$ and $K = 3$, where the phase deviation remains very small until $\omega = 2\pi$. Indeed, only three points approximate a complete revolution with a tiny phase deviation, below $\pi/20$.

10.3 Spectral resolution analysis with respect to Δt and K

Given N and ω , the impact of duplicating the number of points on the phase deviation is assessed. Indeed, taking $2N$ points provides the time parameter $\Delta t/2$ and, consequently, the local wavelength $\omega/2$, but the number of blocks is

Table 20: Spectral deviation – polynomial expressions with respect to the local wavelength ω as function of K and R .

scheme	$p(i\omega) = p(i\omega; K, R)$
Skm [1, 1]	$2 + i\omega$
Skm [1, 2]	$-\omega^2 + 3i\omega + 3$
Skm [1, 3]	$-3i\omega^3 - 11\omega^2 + 18i\omega + 12$
Skm [2, 1]	$-\omega^2 + 6i\omega + 12$
Skm [2, 2]	$\omega^4 - 9i\omega^3 - 39\omega^2 + 90i\omega + 90$
Skm [2, 3]	$-3\omega^6 + 33i\omega^5 + 193\omega^4 - 720i\omega^3 - 1740\omega^2 + 2520i\omega + 1680$
Skm [3, 1]	$-i\omega^3 - 12\omega^2 + 60i\omega + 120$
Skm [3, 2]	$-\omega^6 + 18i\omega^5 + 165\omega^4 - 945i\omega^3 - 3465\omega^2 + 7560i\omega + 7560$
Skm [3, 3]	$9i\omega^9 + 198\omega^8 - 2355i\omega^7 - 19075\omega^6 + 112770i\omega^5$ $+ 494760\omega^4 - 1587600i\omega^3 - 3553200\omega^2 + 4989600i\omega + 3326400$

Figure 8: Dispersion curves $\arg(\xi_R)$ as function of ω with $K = 1$ (—), $K = 2$ (—), and $K = 3$ (—).

uplicated. Hence, the reference phase deviation $\zeta_R(\omega)$, corresponding to N points, must be compared with $\zeta_R^2(\omega/2)$, which involves $2N$ points. More generally, for ℓN points, the phase deviation $\zeta_R(\omega)$ must be compared with $\zeta_R^\ell(\omega/\ell)$, since ℓ blocks are necessary to reach the same final time.

The phase deviations $\zeta_R(\omega)$, $\zeta_R^2(\omega/2)$, and $\zeta_R^3(\omega/3)$ using the reference time step $\Delta t = T/N$ as a function of K (columns) and R (rows) are plotted in Figure 9. First, as expected, a smaller time step strongly improves the spectral resolution by providing more time grid points. The trade-off between the argument reduction and the power increase of the phase deviation $\zeta_R^\ell(\omega/\ell)$ favors a larger ℓ . Second, for fixed ℓ , the phase error is drastically reduced with larger K . For example, with $\ell = 1$, the phase error is almost contained in $\omega = \pi/4$ ($K = 1$), $\omega = \pi/2$ ($K = 2$), and $\omega = \pi$ ($K = 3$). This last case indicates that we recovered subscale information, namely the presence of an extremum within the cell.

10.4 Spectral resolution analysis with respect to K and R

Let $T = 1$ and $\kappa \in \mathbb{N}$ be given. We fix the value of N such that $J = N/R \in \mathbb{N}$, with $R = 1, \dots, 4$. The aim is to compare the deviation of the numerical approximation in the final time T for different choices of K and R , while the time step $\Delta t = T/N$ is constant, independently of the two parameters. For $R = 1$, the final deviation is $\text{dev}(\omega; K, 1) = (\zeta_R(\omega; K, 1))^N$ since N blocks of size $R = 1$ must be solved. For blocks of size R , the final deviation at the final time is given by $\text{dev}(\omega; K, R) = (\zeta_R(\omega; K, R))^J$. Consequently, numerical simulations can be performed to assess the phase deviation relative to the analytic expression.

The deviation errors computed analytically and numerically for different values of κ are reported in Tables 21 and 22. We observe a perfect match between theory and simulation. For $\kappa = 1$, the solution is a full revolution discretised with 36 points. The deviation decreases as K and R increase, in accordance with the scheme's convergence order. We note that doubling the frequency to $\kappa = 2$, that is, 18 points per full revolution, strongly impacts accuracy, especially for the highest-order-accurate schemes. For example, one order of magnitude is lost for $K = 1$ and $R = 1$, while seven orders of magnitude are lost for $K = 4$ and $R = 4$.

To assess the spectral resolution in more challenging situations, we increase κ to 4, 8, and 12. The last case means that we have 9 time grid points to compute two revolutions. The analytical errors of the phase deviation for these cases are reported in Table 23. Clearly, when $K = 1$, we do not resolve the sub-scaled variations with $\kappa = 8, 12$, and the solution is no longer admissible. The case $K = 2$ provides a better approximation of the solution but requires a larger

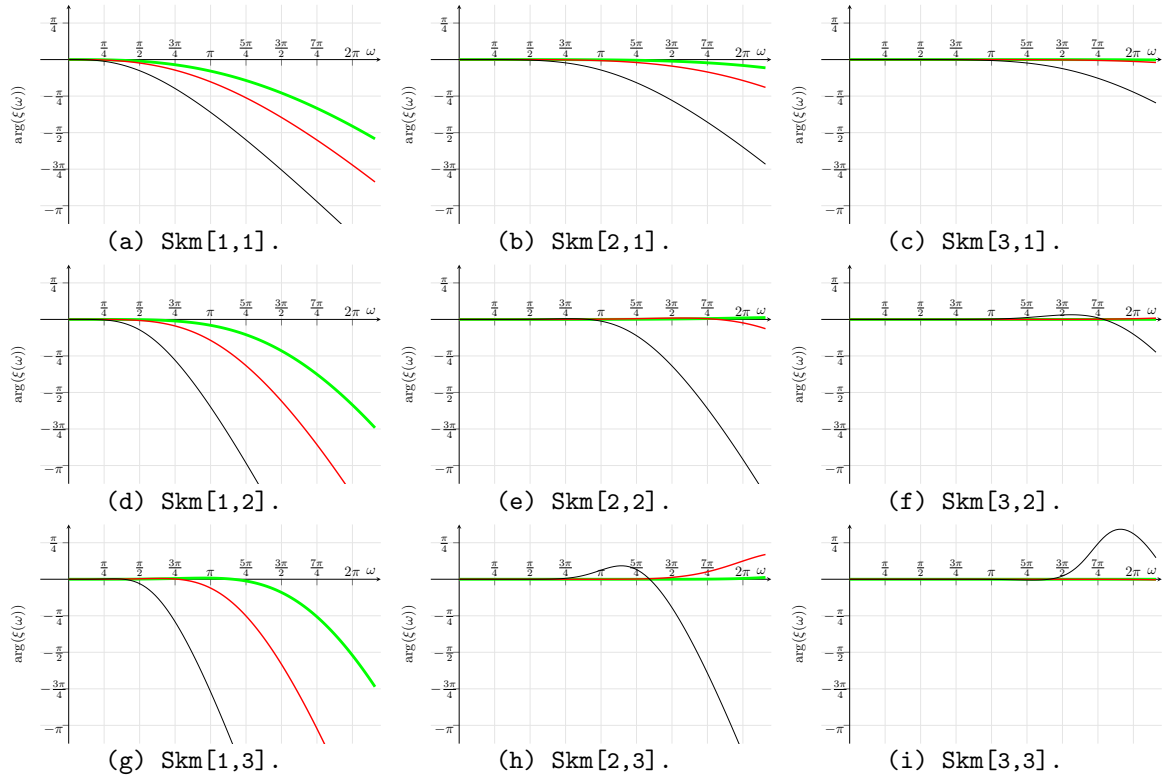


Figure 9: Dispersion curves $\arg(\zeta_R(\omega))$ for Δt (—), $\arg(\zeta_R^2(\omega/2))$ for $\Delta t/2$ (—), and $\arg(\zeta_R^3(\omega/3))$ for $\Delta t/3$ (—) as function of ω for schemes $\text{Skm}[K, R]$, $K, R = 1, 2, 3$.

Table 21: Comparison between the analytical error calculated with $\arg(\zeta_R(\omega; K, R))$ and the numerical error obtained with the structural schemes $\text{Skm}[K, R]$ for $N = 36$, $T = 1$, and $\kappa = 1$ ($\omega = 2\pi/36$).

K	err	R=1	R=2	R=3	R=4
1	analytic	-1.59E-02	-1.29E-04	7.07E-05	2.20E-06
	numeric	-1.59E-02	-1.29E-04	7.07E-05	2.20E-06
2	analytic	-8.08E-06	1.87E-08	-5.08E-11	1.61E-13
	numeric	-8.08E-06	1.87E-08	-5.08E-11	1.61E-13
3	analytic	-1.76E-09	6.27E-15	2.61E-18	-5.38E-23
	numeric	-1.76E-09	6.27E-15	2.61E-18	-5.38E-23
4	analytic	-2.13E-13	-8.88E-20	-5.71E-26	-5.46E-32
	numeric	-2.13E-13	-8.88E-20	-5.71E-26	-5.46E-32

R . From $K = 3$ and above, we managed to solve the sub-scaled information (at least the one associated with the highest frequency $\kappa = 12$).

An interesting point to highlight is the trade-off between the total number of unknowns and accuracy. With K derivatives, we handle $N(K + 1)$ unknowns, and the number is invariant with respect to the parameter R . For example, with $\kappa = 2$, comparable errors are obtained with the structural schemes $\text{Skm}[2, 4]$ and $\text{Skm}[4, 2]$, but with $5N$ unknowns for the first case and $3N$ unknowns for the latter. For $\kappa = 2$, the structural schemes $\text{Skm}[1, 4]$ and $\text{Skm}[2, 1]$ have an accuracy comparable with $3N$ and $2N$ unknowns, respectively. Nevertheless, when increasing the frequency, there may be significant limitations on retrieving sub-scaled information if K is too small, regardless of R . For instance, with $\kappa = 12$, the structural schemes $\text{Skm}[2, 4]$ and $\text{Skm}[3, 1]$ have the same phase deviation error, but a larger K (that is, increasing local data) is preferred to provide better insights on the sub-scaled information.

Table 22: Comparison between the analytical error calculated with $\arg(\zeta_R(\omega; K, R))$ and the numerical error obtained with the structural schemes $\text{Skm}[K, R]$ for $N = 36$, $T = 1$, and $\kappa = 2$, ($\omega = 2\pi/18$).

K	err	R=1	R=2	R=3	R=4
1	analytic	-1.25E-01	-4.03E-03	2.05E-03	2.64E-04
	numeric	-1.25E-01	-4.03E-03	2.05E-03	2.64E-04
2	analytic	-2.57E-04	2.34E-06	-2.46E-08	2.94E-10
	numeric	-2.57E-04	2.34E-06	-2.46E-08	2.94E-10
3	analytic	-2.24E-07	1.27E-11	2.00E-14	-6.72E-18
	numeric	-2.24E-07	1.27E-11	2.00E-14	-6.72E-18
4	analytic	-1.09E-10	-7.14E-16	-7.06E-21	-1.01E-25
	numeric	-1.09E-10	-7.14E-16	-7.06E-21	-1.01E-25

Table 23: Analytical error calculated as $\arg(\zeta_R(\omega; K, R))$ for the structural schemes $\text{Skm}[K, R]$ for $N = 36$, $T = 1$ and three values of κ .

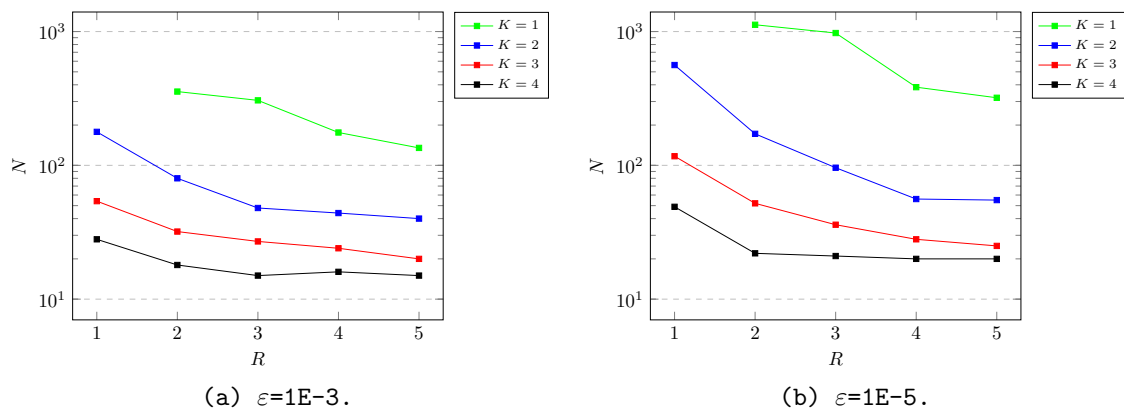
κ	K	R=1	R=2	R=3	R=4
4	1	-9.52E-01	-1.17E-01	4.23E-02	2.54E-02
	2	-8.05E-03	2.74E-04	-9.92E-06	3.38E-07
	3	-2.83E-05	2.51E-08	1.21E-10	-7.21E-13
	4	-5.50E-08	-5.44E-12	-7.19E-16	-1.07E-19
8	1	-1.00E-01	-2.56E-00	-3.75E-01	7.52E-01
	2	-2.35E-01	2.41E-02	-1.14E-03	-4.72E-04
	3	-3.42E-03	4.39E-05	-5.19E-08	-3.53E-08
	4	-2.70E-05	-3.22E-08	-1.12E-11	2.78E-13
12	1	1.66E-00	1.54E-00	-2.61E-01	-2.89E-00
	2	-1.52E-00	1.89E-01	6.68E-02	-5.55E-02
	3	-5.30E-02	2.90E-03	-1.71E-04	7.06E-06
	4	-9.65E-04	-3.10E-06	5.90E-08	1.87E-09

10.5 Phase convergence

To assess phase convergence, we solve the linear equation $\phi'(t) = -i2\kappa\pi\phi(t)$ in $(0, 1]$, and assess the deviation at the final time T , given as

$$\zeta(T) = \frac{\phi_N}{\phi(T)} = \phi_N \exp(i2\kappa\pi T),$$

which represents the transfer function at the end of the simulation. $|\zeta(T)|$ is the amplitude variation while $\arg \zeta(T)$ is the phase shift of the numerical approximation at the final time. Note that if $\kappa \in \mathbb{Z}$ and $T \in \mathbb{Z}$, we simply have $|\zeta(T)| = |\phi_N|$ and $\arg \zeta(T) = \arg \phi_N$.

Figure 10: Smallest values for N to provide a phase error above the parameter ε .

The numerical test we propose aims to determine the smallest value of N for which the scheme provides an error below a prescribed accuracy ε as a function of the two parameters K and R . To this end, we take $\kappa = 10$, and by successive

approximations, we determine the smallest N . The value of N as a function of R for the four curves ($K = 1, \dots, 4$) with tolerances of $\varepsilon = 10^{-3}$ and $\varepsilon = 10^{-5}$ are plotted in Figure 10. Except for the case $K = 1$, a constant value N is quickly reached depending on R . The parameter Δt is mainly controlled by the spectral resolution, that is, the minimum number of points to resolve a full revolution.

11 Conclusions

We proposed and analysed a novel class of numerical schemes for ODEs based on the structural method. These schemes achieve extreme convergence orders and excellent spectral resolution while using compact stencils. We have established their stability and convergence. Several numerical benchmarks demonstrate that the method successfully balances running time and accuracy, resulting in superior numerical efficiency. The structural schemes analysed use the maximum number of physical equations. This leads to a novel multi-derivative Runge-Kutta formulation. Standard techniques often struggle to design and implement such methods, mainly because of the numerous non-linear order conditions. In contrast, our structural schemes determine coefficients directly from the kernel basis. This approach drastically reduces the design complexity and enables arbitrary convergence orders. An additional advantage is the accuracy of the intermediate time steps. Traditional Runge-Kutta methods treat sub-steps as auxiliary calculations. However, in the structural schemes, the values at t_{n+r} , $r \in \llbracket 1, R \rrbracket$, achieve the same convergence order and can be used as genuine approximations. Finally, the structural method naturally connects with compact schemes commonly used in space discretisation. This opens the potential to combine structural equations in both time and space.

Acknowledgements

The authors thank Professor Robert McLachlan for his important note on the connection between the structural method and the multi-derivative Runge-Kutta schemes.

S. Clain, M.T. Malheiro, and G.J. Machado acknowledge the financial support of the Portuguese Foundation for Science and Technology (FCT) through a national funding for projects IC&DT with the reference 2023.16854.ICDT.

S. Clain acknowledge financial support by the Centre for Mathematics of the University of Coimbra (CMUC, <https://doi.org/10.54499/UID/00324/2025>) under the Portuguese Foundation for Science and Technology (FCT), Grants UID/00324/2025 and UID/PRR/00324/2025.

The research of M.T. Malheiro and G.J. Machado was financially supported by the Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) under the scope of the project UID/00013/2025 <https://doi.org/10.54499/UID/00013/2025>, Center of Mathematics of the University of Minho.

The research of R. Costa was partially financed by Portuguese Funds through FCT (Fundação para a Ciência e a Tecnologia) under the scope of the projects UIDB/05256/2025 and UIDP/05256/2025 <https://doi.org/10.54499/UID/00013/2025>, Institute for Polymers and Composites of the University of Minho.

R. Costa also acknowledges the financial support of Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) through the Contract-Program signed between the Fundação para a Ciência e a Tecnologia and the University of Minho, under the scope of the project 2023.09481.CEECIND/CP2841/CT0008, Individual Scientific Employment Stimulus Call – 6th Edition (CEEC IND6ed).

Statement on the use of Artificial Intelligence Tools

The document was reviewed and refined using Large Language Models to improve grammar, correct typographical errors, and enhance clarity. Any text or content generated by Artificial Intelligence is explicitly identified within the document and is utilized only to a limited extent. The authors retain full responsibility for the overall content and ideas presented.

A Analytical structural equations

The structural schemes $\text{Skm}[K, R]$ are based on the structural equations $\text{SE}_{KR}^S(s)$, $s \in \llbracket 1, S \rrbracket$, characterised by the basis $(\mathbf{a}^s)_{s=1}^S$ of the kernel $\mathcal{K}^S \subset \mathbb{R}^M$, with $M = (K + 1)(R + 1)$ the size of the vectors. By construction, the structural

equations are exact up to a polynomial degree $d = M - S$ and read

$$\sum_{k=0}^K \sum_{r=0}^R a_{k,r}^s \Delta t^k \phi_{n+r}^{(k)} = 0, \quad s \in \llbracket 1, K \rrbracket.$$

In practice, we use $S = R$ to design the structural method, and the structural schemes $\text{Skm}[K, R]$ provide R linearly independent equations.

Tables 24, 25, and 26 present the analytical expressions of the coefficients $a_{k,r}^s$ for $R = 1, 2, 3$, each with $K = 1, 2, 3$. Additionally, the kernel method can be used to compute these coefficients numerically for any choice of K, R , and S .

Table 24: Structural equations coefficients for $R = 1$.

K	s	$a_{0,0}^s$	$a_{1,0}^s$	$a_{2,0}^s$	$a_{3,0}^s$	$a_{0,1}^s$	$a_{1,1}^s$	$a_{2,1}^s$	$a_{3,1}^s$
1	1	-2	-1			2	-1		
2	1	-12	-6	-1		12	-6	1	
3	1	120	60	12	1	-120	60	-12	1

Table 25: Structural equations coefficients for $R = 2$.

K	s	$a_{0,0}^s$	$a_{1,0}^s$	$a_{2,0}^s$	$a_{3,0}^s$	$a_{0,1}^s$	$a_{1,1}^s$	$a_{2,1}^s$	$a_{3,1}^s$	$a_{0,2}^s$	$a_{1,2}^s$	$a_{2,2}^s$	$a_{3,2}^s$
1	1	-3	-1			0	-4			3	-1		
	2	2	1			-4	0			2	-1		
2	1	24	9	1		-48	0	-8		24	-9	1	
	2	-15	-7	-1		0	-16	0		15	-7	-1	
3	1	315	123	18	1	0	384	0	16	-315	123	-18	1
	2	-192	-87	-15	-1	384	0	48	0	-192	87	-15	-1

Table 26: Structural equations coefficients for $R = 3$.

K	s	$a_{0,0}^s$	$a_{1,0}^s$	$a_{2,0}^s$	$a_{3,0}^s$	$a_{0,1}^s$	$a_{1,1}^s$	$a_{2,1}^s$	$a_{3,1}^s$	$a_{0,2}^s$	$a_{1,2}^s$	$a_{2,2}^s$	$a_{3,2}^s$	$a_{0,3}^s$	$a_{1,3}^s$	$a_{2,3}^s$	$a_{3,3}^s$
1	1	11	3			27	27			-27	27			-11	3		
	2	19	6			0	27			-27	0			8	-3		
	3	-13	-6			27	0			-27	0			13	-6		
2	1	-103	-33	-3		729	243	81		-729	243	-81		103	-33	3	
	2	173	60	6		-486	-81	-81		243	-162	0		70	-27	3	
	3	-113	-48	-6		-81	-162	0		81	-162	0		113	-48	6	
3	1	4711	1599	198	9	45927	28431	4374	729	-45927	28431	-4374	729	-4711	1599	-198	9
	2	-7823	-2802	-369	-18	-17496	-19683	-2187	-729	28431	-8748	2187	0	-3112	1203	-171	9
	3	5021	2064	315	18	-19683	-4374	-2187	0	19683	-4374	2187	0	-5021	2064	-315	18

B Post-processing structural equations

Assuming that we have performed the computation with a structural scheme $\text{Skm}[K, R]$, post-processing structural equations enable computing higher derivatives of order p with $p > K$ at the final time step t_N using the approximate solution and derivatives from the backward time steps. We then propose an approximation for $\phi_N^{(p)}$ of convergence order O using a relation in the form

$$\phi_N^{(p)} = \sum_{i=0}^{I_p} \sum_{k=0}^K a_{ik}^p \Delta t^k \phi_{N-i}^{(k)},$$

where the convergence order of the method and the coefficients are given in the following tables.

B.1 Case $K = 1$

We assume that the approximations $\phi_{-i}^{(k)}$ are available at any point $T - i\Delta t$, $i \in \llbracket 0, I_d \rrbracket$, and $k = 0, 1$. Table 27 provides the formula and the corresponding convergence order for the approximation $\phi_N^{(p)}$ for $p = 2, 3, 4$.

Table 27: Coefficients of the post-processing structural equations with $K = 1$ for $p = 2, 3, 4$.

p	Ip	0	$\phi_N^{(p)}$
2	1	2	$\phi_N^{(2)} = 6\phi_{N-1}^{(0)} - 6\phi_N^{(0)} + \Delta t(2\phi_{N-1}^{(1)} + 4\phi_N^{(1)})$
2	2	4	$\phi_N^{(2)} = \frac{7}{2}\phi_{N-2}^{(0)} + 8\phi_{N-1}^{(0)} - \frac{23}{2}\phi_N^{(0)} + \Delta t(\phi_{N-2}^{(1)} + 8\phi_{N-1}^{(1)} + 6\phi_N^{(1)})$
2	3	6	$\phi_N^{(2)} = \frac{8}{3}\phi_{N-3}^{(0)} + \frac{27}{2}\phi_{N-2}^{(0)} - \frac{97}{6}\phi_N^{(0)} + \Delta t(\frac{2}{3}\phi_{N-3}^{(1)} + 9\phi_{N-2}^{(1)} + 18\phi_{N-1}^{(1)} + \frac{22}{3}\phi_N^{(1)})$
3	2	3	$\phi_N^{(3)} = \frac{51}{2}\phi_{N-2}^{(0)} + 24\phi_{N-1}^{(0)} - \frac{99}{2}\phi_N^{(0)} + \Delta t(\frac{15}{2}\phi_{N-2}^{(1)} + 48\phi_{N-1}^{(1)} + \frac{39}{2}\phi_N^{(1)})$
3	3	5	$\phi_N^{(3)} = \frac{238}{9}\phi_{N-3}^{(0)} + \frac{243}{2}\phi_{N-2}^{(0)} - 54\phi_{N-1}^{(0)} - \frac{1691}{18}\phi_N^{(0)} + \Delta t(\frac{20}{3}\phi_{N-3}^{(1)} + \frac{171}{2}\phi_{N-2}^{(1)} + 144\phi_{N-1}^{(1)} + \frac{193}{6}\phi_N^{(1)})$
3	4	7	$\phi_N^{(3)} = \frac{1241}{48}\phi_{N-4}^{(0)} + \frac{2176}{9}\phi_{N-3}^{(0)} + 171\phi_{N-2}^{(0)} - \frac{896}{3}\phi_{N-1}^{(0)} - \frac{20155}{144}\phi_N^{(0)} + \Delta t(\frac{47}{8}\phi_{N-4}^{(1)} + \frac{368}{3}\phi_{N-3}^{(1)} + 396\phi_{N-2}^{(1)} + 304\phi_{N-1}^{(1)} + \frac{1045}{24}\phi_N^{(1)})$
4	2	2	$\phi_N^{(4)} = 78\phi_{N-2}^{(0)} + 24\phi_{N-1}^{(0)} - 102\phi_N^{(0)} + \Delta t(24\phi_{N-2}^{(1)} + 120\phi_{N-1}^{(1)} + 36\phi_N^{(1)})$
4	3	4	$\phi_N^{(4)} = \frac{400}{3}\phi_{N-3}^{(0)} + 540\phi_{N-2}^{(0)} - 360\phi_{N-1}^{(0)} - \frac{940}{3}\phi_N^{(0)} + \Delta t(34\phi_{N-3}^{(1)} + 408\phi_{N-2}^{(1)} + 582\phi_{N-1}^{(1)} + 96\phi_N^{(1)})$
4	4	6	$\phi_N^{(4)} = \frac{11879}{72}\phi_{N-4}^{(0)} + \frac{13328}{9}\phi_{N-3}^{(0)} + \frac{1659}{2}\phi_{N-2}^{(0)} - \frac{16912}{9}\phi_{N-1}^{(0)} - \frac{42931}{72}\phi_N^{(0)} + \Delta t(\frac{113}{3}\phi_{N-4}^{(1)} + \frac{2296}{3}\phi_{N-3}^{(1)} + 2343\phi_{N-2}^{(1)} + \frac{4712}{3}\phi_{N-1}^{(1)} + \frac{995}{6}\phi_N^{(1)})$

B.2 Case $K = 2$

We proceed with $K = 2$ and provide in Table 28 the post-processing structural equations that provide an approximation of $\phi_N^{(p)}$ for $p = 3, 4$.

Table 28: Coefficients of the post-processing structural equations with $K = 2$ for $p = 3, 4$.

p	Ip	0	$\phi_N^{(p)}$
3	2	6	$\phi_N^{(3)} = 87\phi_{N-2}^{(0)} - 384\phi_{N-1}^{(0)} + 297\phi_N^{(0)}$ $+ 2\Delta t(15\phi_{N-2}^{(1)} - 48\phi_{N-1}^{(1)} - 72\phi_N^{(1)})$ $+ \Delta t^2(3\phi_{N-2}^{(2)} - 48\phi_{N-1}^{(2)} + 27\phi_N^{(2)})$
3	3	9	$\phi_N^{(3)} = -\frac{344}{9}\phi_{N-3}^{(0)} + \frac{891}{2}\phi_{N-2}^{(0)} - 648\phi_{N-1}^{(0)} + \frac{4333}{18}\phi_N^{(0)}$ $+ \Delta t(-\frac{35}{3}\phi_{N-3}^{(1)} + 162\phi_{N-2}^{(1)} + 81\phi_{N-1}^{(1)} - 103\phi_N^{(1)})$ $+ \Delta t^2(-\phi_{N-3}^{(2)} + \frac{81}{2}\phi_{N-2}^{(2)} - 81\phi_{N-1}^{(2)} + \frac{33}{2}\phi_N^{(2)})$
3	4	12	$\phi_N^{(3)} = \frac{559}{16}\phi_{N-4}^{(0)} - \frac{8768}{9}\phi_{N-3}^{(0)} + 2592\phi_{N-2}^{(0)} - 1984\phi_{N-1}^{(0)} + \frac{47705}{144}\phi_N^{(0)}$ $+ \Delta t(\frac{39}{4}\phi_{N-4}^{(1)} - \frac{1088}{3}\phi_{N-3}^{(1)} + 324\phi_{N-2}^{(1)} + 576\phi_{N-1}^{(1)} - 130\phi_N^{(1)})$ $+ \Delta t^2(\frac{3}{4}\phi_{N-4}^{(2)} - 64\phi_{N-3}^{(2)} + 324\phi_{N-2}^{(2)} - 192\phi_{N-1}^{(2)} + \frac{75}{4}\phi_N^{(2)})$
4	2	5	$\phi_N^{(4)} = \frac{1359}{2}\phi_{N-2}^{(0)} - 2304\phi_{N-1}^{(0)} + \frac{3249}{2}\phi_N^{(0)}$ $+ \Delta t(237\phi_{N-2}^{(1)} - 480\phi_{N-1}^{(1)} - 702\phi_N^{(1)})$ $+ \Delta t^2(24\phi_{N-2}^{(2)} - 336\phi_{N-1}^{(2)} + 99\phi_N^{(2)})$
4	3	8	$\phi_N^{(4)} = -\frac{7060}{9}\phi_{N-3}^{(0)} + \frac{17415}{2}\phi_{N-2}^{(0)} - 11340\phi_{N-1}^{(0)} + \frac{61505}{18}\phi_N^{(0)}$ $+ \Delta t(-\frac{2162}{9}\phi_{N-3}^{(1)} + 3159\phi_{N-2}^{(1)} + 2106\phi_{N-1}^{(1)} - \frac{11728}{9}\phi_N^{(1)})$ $+ \Delta t^2(-\frac{62}{3}\phi_{N-3}^{(2)} + 810\phi_{N-2}^{(2)} - 1458\phi_{N-1}^{(2)} + 157\phi_N^{(2)})$
4	4	11	$\phi_N^{(4)} = \frac{26751}{32}\phi_{N-4}^{(0)} - \frac{205888}{9}\phi_{N-3}^{(0)} + 58968\phi_{N-2}^{(0)} - 42432\phi_{N-1}^{(0)} + \frac{1585289}{288}\phi_N^{(0)}$ $+ \Delta t(\frac{1869}{8}\phi_{N-4}^{(1)} - 76736\phi_{N-3}^{(1)} + 6804\phi_{N-2}^{(1)} + 13632\phi_{N-1}^{(1)} - \frac{69295}{36}\phi_N^{(1)})$ $+ \Delta t^2(18\phi_{N-4}^{(2)} - \frac{4544}{3}\phi_{N-3}^{(2)} + 7452\phi_{N-2}^{(2)} - 4032\phi_{N-1}^{(2)} + \frac{835}{4}\phi_N^{(2)})$

B.3 Case $K = 3$

Finally, for $K = 3$ we provide in Table 28 the post-processing structural equations that provide an approximation of $\phi_N^{(4)}$.

C Scheme stability data

In Section 9, we established the stability condition of the structural schemes $\text{Skm}[K, R]$ with Theorem 9.5 if the transfer function reads $\chi_R(z) = p(z)/p(-z)$ and $p(z)$ is a Hurwitz polynomial in $\mathbb{C}$. In this appendix, we first provide the expressions for $\chi_R(z)$, in Tables 30, 31, and 32 for $K, R = 1, 2, 3$ obtained with the Symbolic Toolbox of MATLAB. We then check that the polynomials satisfy the Hurwitz property.

C.1 Case $K = 1$

- $\text{Skm}[1, 1]$: unconditional second-order accurate scheme with $p(z) = z + 2$ and its only zero ($z = -2$) is in the left half-plane.
- $\text{Skm}[1, 2]$: unconditional fourth-order accurate scheme with $p(z) = z^2 + 3z + 3$ and

$$H_{12} = \begin{bmatrix} 3 & 0 \\ 1 & 3 \end{bmatrix}.$$

Table 29: Coefficients of the post-processing structural equations with $K = 3$ for $p = 4$.

p	Ip	0	$\phi_N^{(p)}$
4	1	4	$\phi_N^{(4)} = 840\phi_{N-1}^{(0)} - 840\phi_N^{(0)}$ $+ \Delta t(360\phi_{N-1}^{(1)} + 480\phi_N^{(1)})$ $+ \Delta t^2(60\phi_{N-1}^{(2)} - 120\phi_N^{(2)})$ $+ \Delta t^3(4\phi_{N-1}^{(3)} + 16\phi_N^{(3)})$
4	2	8	$\phi_N^{(4)} = 1545\phi_{N-2}^{(0)} + 1920\phi_{N-1}^{(0)} - \frac{5385}{2}\phi_N^{(0)}$ $+ \Delta t(285\phi_{N-2}^{(1)} + 1920\phi_{N-1}^{(1)} + 1260\phi_N^{(1)})$ $+ \Delta t^2(39\phi_{N-2}^{(2)} + 192\phi_{N-1}^{(2)} - 246\phi_N^{(2)})$ $+ \Delta t^3(2\phi_{N-2}^{(3)} + 64\phi_{N-1}^{(3)} + 24\phi_N^{(3)})$
4	3	12	$\phi_N^{(4)} = \frac{21160}{27}\phi_{N-3}^{(0)} + \frac{27945}{2}\phi_{N-2}^{(0)} - 9720\phi_{N-1}^{(0)} - \frac{271955}{54}\phi_N^{(0)}$ $+ \Delta t(\frac{772}{3}\phi_{N-3}^{(1)} + 7533\phi_{N-2}^{(1)} + 10692\phi_{N-1}^{(1)} + \frac{18844}{9}\phi_N^{(1)})$ $+ \Delta t^2(\frac{92}{3}\phi_{N-3}^{(2)} + 1215\phi_{N-2}^{(2)} - 972\phi_{N-1}^{(2)} - \frac{1066}{3}\phi_N^{(2)})$ $+ \Delta t^3(\frac{4}{3}\phi_{N-3}^{(3)} + 162\phi_{N-2}^{(3)} + 324\phi_{N-1}^{(3)} + \frac{88}{3}\phi_N^{(3)})$

Table 30: Transfer functions of the structural schemes Skm[1, R].

scheme	R	$\chi_R(z)$
Skm[1, 1]	1	$\frac{2+z}{2-z}$
Skm[1, 2]	2	$\frac{z^2+3z+3}{z^2-3z+3}$
Skm[1, 3]	3	$\frac{3z^3+11z^2+18z+12}{-3z^3+11z^2-18z+12}$

Its determinant is positive.

- Skm[1, 3]: unconditional fourth-order accurate scheme with $p(z) = 3z^3 + 11z^2 + 18z + 12$ and

$$H_{13} = \begin{bmatrix} 11 & 12 & 0 \\ 3 & 18 & 0 \\ 0 & 11 & 12 \end{bmatrix}.$$

All its principal minors m_i for $i \in \llbracket 1, 3 \rrbracket$ of $\det H_{13}$ are strictly positive.

C.2 Case $K = 2$

Table 31: Transfer functions of the structural schemes Skm[2, R].

scheme	R	$\chi_R(z)$
Skm[2, 1]	1	$\frac{z^2+6z+12}{z^2-6z+12}$
Skm[2, 2]	2	$\frac{z^4+9z^3+39z^2+90z+90}{z^4-9z^3+39z^2-90z+90}$
Skm[2, 3]	3	$\frac{3z^6+33z^5+193z^4+720z^3+1740z^2+2520z+1680}{3z^6-33z^5+193z^4-720z^3+1740z^2-2520z+1680}$

- Skm[2, 1]: unconditional fourth-order accurate scheme with $p(z) = z^2 + 6z + 12$ and

$$H_{21} = \begin{bmatrix} 6 & 0 \\ 1 & 12 \end{bmatrix}.$$

Its determinant is positive.

- Skm[2,2]: unconditional sixth-order accurate scheme with $p(z) = z^4 + 9z^3 + 39z^2 + 90z + 90$ and

$$H_{22} = \begin{bmatrix} 9 & 90 & 0 & 0 \\ 1 & 39 & 90 & 0 \\ 0 & 9 & 90 & 0 \\ 0 & 1 & 39 & 90 \end{bmatrix}.$$

All principal minors m_i for $i \in \llbracket 1, 4 \rrbracket$ of $\det H_{22}$ are strictly positive.

- Skm[2,3]: unconditional eighth-order accurate scheme with $p(z) = 3z^6 + 33z^5 + 193z^4 + 720z^3 + 1740z^2 + 2520z + 1680$ and

$$H_{23} = \begin{bmatrix} 33 & 720 & 2520 & 0 & 0 & 0 \\ 3 & 193 & 1740 & 1680 & 0 & 0 \\ 0 & 33 & 720 & 2520 & 0 & 0 \\ 0 & 3 & 193 & 1740 & 1680 & 0 \\ 0 & 0 & 33 & 720 & 2520 & 0 \\ 0 & 0 & 3 & 193 & 1740 & 1680 \end{bmatrix}.$$

All principal minors m_i for $i \in \llbracket 1, 6 \rrbracket$ of $\det H_{23}$ are strictly positive.

C.3 Case $K = 3$

Table 32: Transfer functions of the structural schemes Skm[3,R].

scheme	R	$\chi_R(z)$
Skm[3,1]	1	$\frac{z^3 + 12z^2 + 60z + 120}{-z^3 + 12z^2 - 60z + 120}$
Skm[3,2]	2	$\frac{z^6 + 18z^5 + 165z^4 + 945z^3 + 3465z^2 + 7560z + 7560}{z^6 - 18z^5 + 165z^4 - 945z^3 + 3465z^2 - 7560z + 7560}$
Skm[3,3]	3	$\frac{9z^9 + 198z^8 + 2355z^7 + 19075z^6 + 112770z^5 + 494760z^4 + 1587600z^3 + 3553200z^2 + 4989600z + 3326400}{-9z^9 + 198z^8 - 2355z^7 + 19075z^6 - 112770z^5 + 494760z^4 - 1587600z^3 + 3553200z^2 - 4989600z + 3326400}$

- Skm[3,1]: unconditional sixth-order accurate scheme with $p(z) = z^3 + 12z^2 + 60z + 120$ and

$$H_{31} = \begin{bmatrix} 12 & 120 & 0 \\ 1 & 60 & 0 \\ 0 & 12 & 120 \end{bmatrix}.$$

All principal minors m_i for $i \in \llbracket 1, 3 \rrbracket$ of $\det H_{31}$ are strictly positive.

- Skm[3,2]: unconditional tenth-order accurate scheme with $p(z) = z^6 + 18z^5 + 165z^4 + 945z^3 + 3465z^2 + 7560z + 7560$ and

$$H_{32} = \begin{bmatrix} 18 & 945 & 7560 & 0 & 0 & 0 \\ 1 & 165 & 3465 & 7560 & 0 & 0 \\ 0 & 18 & 945 & 7560 & 0 & 0 \\ 0 & 1 & 165 & 3465 & 7560 & 0 \\ 0 & 0 & 18 & 945 & 7560 & 0 \\ 0 & 0 & 1 & 165 & 3465 & 7560 \end{bmatrix}.$$

All principal minors m_i , for $i \in \llbracket 1, 6 \rrbracket$ of $\det H_{32}$ are strictly positive.

- Skm[3,3]: unconditional twelfth-order accurate scheme with $p(z) = 9z^9 + 198z^8 + 2355z^7 + 19075z^6 + 112770z^5 + 494760z^4 + 1587600z^3 + 3553200z^2 + 4989600z + 3326400$ and

$$H_{33} = \begin{bmatrix} 198 & 19075 & 494760 & 3553200 & 3326400 & 0 & 0 & 0 & 0 \\ 9 & 2355 & 112770 & 15876000 & 4989600 & 0 & 0 & 0 & 0 \\ 0 & 198 & 19075 & 494760 & 3553200 & 3326400 & 0 & 0 & 0 \\ 0 & 9 & 2355 & 112770 & 15876000 & 4989600 & 0 & 0 & 0 \\ 0 & 0 & 198 & 19075 & 494760 & 3553200 & 3326400 & 0 & 0 \\ 0 & 0 & 9 & 2355 & 112770 & 15876000 & 4989600 & 0 & 0 \\ 0 & 0 & 0 & 198 & 19075 & 494760 & 3553200 & 3326400 & 0 \\ 0 & 0 & 0 & 9 & 2355 & 112770 & 15876000 & 4989600 & 0 \\ 0 & 0 & 0 & 0 & 198 & 19075 & 494760 & 3553200 & 3326400 \end{bmatrix}.$$

All principal minors m_i for $i \in \llbracket 1, 9 \rrbracket$ of $\det H_{33}$ are strictly positive.

Hence, all schemes are A-stable.

D Butcher Tableau for the collocation method

D.1 The 4th-Order Method (2-Stage Gauss-Legendre)

The 2-stage 4th-order Butcher tableau.

$$\begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \quad (23)$$

D.2 The 6th-Order Method (3-Stage Gauss-Legendre)

The 3-stage 6th-order Butcher tableau

$$\begin{array}{c|ccc} \frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\ \frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\ \frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array} \quad (24)$$

D.3 8th-Order Gauss-Legendre Butcher Tableau

The 4-stage, 8th-order Butcher Tableau with 32 digits of precision.

Stage Nodes (c_i)		Weights (b_i)	
$c_1 = 0.06943184420297371238854284555813$		$b_1 = 0.17392742256872692868653197461040$	
$c_2 = 0.33000947820757186759865892120042$		$b_2 = 0.32607257743127307131346802538960$	
$c_3 = 0.66999052179242813240134107879958$		$b_3 = 0.32607257743127307131346802538960$	
$c_4 = 0.93056815579702628761145715444187$		$b_4 = 0.17392742256872692868653197461040$	
Stage Matrix Entries (a_{ij})			
Row 1		Row 3	
$a_{11} = 0.08696371128436346434326598730520$		$a_{31} = 0.16755826838249626870218730585505$	
$a_{12} = -0.02619985959089069670008594246830$		$a_{32} = 0.35285195727314370598990176843438$	
$a_{13} = 0.01509374026369527376785640232490$		$a_{33} = 0.16303628871563653565673401269480$	
$a_{14} = -0.00642574775419432902249360160367$		$a_{34} = -0.01345599258210340356560943180470$	
Row 2		Row 4	
$a_{21} = 0.18738361514757530663415174092415$		$a_{41} = 0.18035317032292125770902557621407$	
$a_{22} = 0.16303628871563653565673401269480$		$a_{42} = 0.31097883716757779754561162306470$	
$a_{23} = -0.02677937984187063467657150117387$		$a_{43} = 0.35227243702216376801355403061452$	
$a_{24} = 0.00636915418623065998434466875535$		$a_{44} = 0.08696371128436346434326598730520$	

10th-Order Gauss-Legendre Butcher Tableau

The 5-stage, 10th-order Butcher Tableau with 32 digits of precision.

Stage Nodes (c_i)		Weights (b_i)	
$c_1 = 0.04691007703066800360118656085031$		$b_1 = 0.11846344252809454375713202035994$	
$c_2 = 0.23076534494715845448184278821844$		$b_2 = 0.23931433529974125961272126693111$	
$c_3 = 0.50000000000000000000000000000000$		$b_3 = 0.28444444444444444444444444444444$	
$c_4 = 0.76923465505284154551815721178156$		$b_4 = 0.23931433529974125961272126693111$	
$c_5 = 0.95308992296933199639881343914969$		$b_5 = 0.11846344252809454375713202035994$	
Stage Matrix Entries (a_{ij})			

Row 1	Row 4
$a_{11} = 0.05923172126404727187856601017997$	$a_{41} = 0.12030981742486557590637196787362$
$a_{12} = -0.01783063595561081525091763781744$	$a_{42} = 0.23191456794265233266718866723923$
$a_{13} = 0.00913988019488390772702758169116$	$a_{43} = 0.30276330363184353298642084661853$
$a_{14} = -0.00494576326190820521316694605917$	$a_{44} = 0.11965716764987062980636063346555$
$a_{15} = 0.00131487478925584445967755285579$	$a_{45} = -0.00621020159639052585250444372982$
Row 2	Row 5
$a_{21} = 0.12467364402636901844577884784116$	$a_{51} = 0.11714856773883869929745446750415$
$a_{22} = 0.11965716764987062980636063346555$	$a_{52} = 0.24426009856164946482588821299028$
$a_{23} = -0.01831885918739908856658934526647$	$a_{53} = 0.27530456424956053671741686275328$
$a_{24} = 0.00739976735508892694553259969188$	$a_{54} = 0.25714497125535207486363890474855$
$a_{25} = -0.00184637489677103214923994751368$	$a_{55} = 0.05923172126404727187856601017997$
Row 3	
$a_{31} = 0.12151608752230752538964344402283$	
$a_{32} = 0.26127548981650392306774641973053$	
$a_{33} = 0.14222222222222222222222222222222$	
$a_{34} = -0.02195982511624518268046768666164$	
$a_{35} = 0.00294602555521151199341883713653$	

D.3.1 12th-Order Gauss-Legendre Butcher Tableau

The 6-stage, 12th-order Butcher Tableau with 32 digits of precision

Stage Nodes (c_i)	Weights (b_i)
$c_1 = 0.03376524289842398612197577545465$	$b_1 = 0.08566224618958517252277579178949$
$c_2 = 0.16939530676686774311894236047318$	$b_2 = 0.18038078652406930335017409240366$
$c_3 = 0.38069040695840154571932338148970$	$b_3 = 0.23395696728634552412705011580685$
$c_4 = 0.61930959304159845428067661851030$	$b_4 = 0.23395696728634552412705011580685$
$c_5 = 0.83060469323313225688105763952682$	$b_5 = 0.18038078652406930335017409240366$
$c_6 = 0.96623475710157601387802422454535$	$b_6 = 0.08566224618958517252277579178949$
Stage Matrix Entries (a_{ij})	
Row 1	Row 4
$a_{11} = 0.04283112309479258626138789589475$	$a_{41} = 0.08625666550248192070640445809090$
$a_{12} = -0.01255762837335661011195655554625$	$a_{42} = 0.17596461234875152182442169320172$
$a_{13} = 0.00557476646535555986877903822818$	$a_{43} = 0.25265567757907443762884976996730$
$a_{14} = -0.00310860551065116744839818815124$	$a_{44} = 0.11697848364317276206352505790343$
$a_{15} = 0.00127116790938676239162985331562$	$a_{45} = -0.01121852115999424888493132714526$
$a_{16} = -0.00024558068710314483946626828641$	$a_{46} = 0.00180479109015978161730628292887$
Row 2	Row 5
$a_{21} = 0.08865487711929944062145889369324$	$a_{51} = 0.08537126964880648192196395492033$
$a_{22} = 0.09019039326203465167508704620183$	$a_{52} = 0.18226690839722077906398316721729$
$a_{23} = -0.01235650742136195759187383216892$	$a_{53} = 0.22856527814707713060232532463428$
$a_{24} = 0.00539168913926839352726749069150$	$a_{54} = 0.24631347470770748171638210344583$
$a_{25} = -0.00188612187315147571380907481363$	$a_{55} = 0.09019039326203465167508704620183$
$a_{26} = 0.00029097654077869060081183686916$	$a_{56} = -0.00223594892742918868840224021234$
Row 3	Row 6
$a_{31} = 0.08728956977264879555234249112444$	$a_{61} = 0.08541666550248192070640445809090$
$a_{32} = 0.19159930768407428399587442111893$	$a_{62} = 0.17910961861468254095854394571928$
$a_{33} = 0.11697848364317276206352505790343$	$a_{63} = 0.22954079937695876173542026249123$
$a_{34} = -0.01869871029272891350179965416045$	$a_{64} = 0.23706557280011504936496464871261$
$a_{35} = 0.00441617417531778152575239920194$	$a_{65} = 0.19293841489742591346213064794991$
$a_{36} = -0.00059441802408316391637133369859$	$a_{66} = 0.04283112309479258626138789589475$

References

- [1] M. Pettigrew, On compact finite difference schemes with applications to moving boundary problems, phd thesis (1989).
- [2] S. Lele, Compact finite difference schemes with spectral-like resolution, *J. Comput. Phys.* 103 (1992) 16–42. doi:10.1016/0021-9991(92)90324-R.
- [3] M. Urabe, An implicit one-step method of high-order accuracy for the numerical integration of ordinary differential equations, *Numer. Math.* 15 (1970) 151–164. doi:10.1007/BF02165379.
- [4] H. Shintani, On one-step methods utilizing the second derivative, *Hiroshima Math. J.* 1 (1971) 349–372. doi:10.32917/hmj/1206137979.
- [5] H. Shintani, On explicit one-step methods utilizing the second derivative, *Hiroshima Math. J.* 2 (1972) 353–368. doi:10.32917/hmj/1206137626.
- [6] K. Kastlunger, G. Wanner, Runge kutta processes with multiple nodes, *Computing* 9 (1972) 9–24. doi:10.1007/BF02236372.
- [7] K. Kastlunger, G. Wanner, On turan type implicit runge-kutta methods, *Computing* 9 (1972) 317–325. doi:10.1007/BF02241605.
- [8] E. Hairer, G. Wanner, Multistep-multistage-multiderivative methods for ordinary differential equations, *Computing* 11 (1973) 287–303. doi:10.1007/BF02252917.
- [9] J. Cash, High order methods for the numerical integration of ordinary differential equations, *Numer. Math.* 36 (1978) 385–409. doi:10.1007/BF01398507.
- [10] G. Makinson, Stable high order implicit methods for the numerical solution of systems of differential equations, *Comput. J.* 11 (1968) 305–310. doi:10.1093/comjnl/11.3.305.
- [11] M. Reimer, Finite difference forms containing derivatives of higher order, *SIAM J. Numer. Anal.* 5 (1968) 725–738. doi:10.1137/0705056.
- [12] W. Liniger, R. Willoughby, Efficient integration methods for stiff systems of ordinary differential equations, *SIAM J. Numer. Anal.* 7 (1970) 47–66. doi:10.1137/0707002.
- [13] W. Enright, Second derivative multistep methods for stiff ordinary differential equations, *SIAM J. Numer. Anal.* 11 (2) (1974) 321–331. doi:10.1137/0711029.
- [14] W. Enright, *Optimal Second Derivative Methods for Stiff Systems*, Springer US, Boston, MA, 1974, pp. 95–109. doi:10.1007/978-1-4684-2100-2_8.
- [15] Y. Genin, An algebraic approach to a-stable linear multistep-multiderivative integration formulas, *BIT* 14 (1974) 382–406. doi:10.1007/BF01932535.
- [16] M. Makela, O. Nevanlinna, A. H. Sipila, Exponentially fitted multi-step methods by generalized hermite birkhoff interpolation, *BIT* 14 (1974) 437–451. doi:10.1007/BF01932540.
- [17] R. Jeltsch, A necessary condition for a-stability of multistep multiderivative methods, *Math. Comp.* 30 (1976) 739–746. doi:10.1007/BF01940779.
- [18] R. Jeltsch, Note on a-stability of multistep multiderivative methods, *BIT* 16 (1976) 74–78. doi:10.1007/BF01940779.
- [19] R. Jeltsch, Stiff stability of multistep multiderivatives methods, *SIAM J. Numer. Anal.* 14 (1977) 760–772. doi:10.1137/0714052.
- [20] R. Jeltsch, Multistep methods using higher derivatives and damping at infinity, *Math. Comp.* 31 (1977) 124–138. doi:10.2307/2005784.
- [21] R. Jeltsch, A0-stability and stiff stability of brown’s multistep multiderivative methods, *Numer. Math.* 32 (1979) 167–181. doi:10.1007/BF01404873.
- [22] R. Brown, Some characteristics of implicit multistep multi-derivative integration formulas, *SIAM J. Numer. Anal.* 14 (1977) 982–993. doi:10.1137/0714066.
- [23] G. Gupta, Implementing second-derivative multistep methods using the nordsieck polynomial representation, *Math. Comput.* 32 (1978) 13–18. doi:10.2307/2006255.
- [24] J. Cash, Second derivative extended backward differentiation formulas for the numerical integration of stiff systems, *SIAM J. Numer. Anal.* 18 (1981) 21–36. doi:10.1137/0718003.

- [25] J. Cash, On the exponential fitting of composite multiderivative linear multistep methods, *SIAM J. Numer. Anal.* 18 (1981) 808–821. doi:10.1137/0718055.
- [26] J. Lambert, *Computational methods in ordinary differential equations*, Math. Sci. Eng., Wiley, London, 1973.
- [27] E. Gekeler, R. Widmann, On the order conditions of runge-kutta methods with higher derivatives, *Numer. Math.* 50 (1986) 183–203.
- [28] D. Goeken, O. Johnson, Runge–kutta with higher order derivative approximations, *Appl. Numer. Math.* 34 (2000) 207–218. doi:10.1016/S0168-9274(99)00128-2.
- [29] J. Butcher, *Numerical Methods for Ordinary Differential Equations*, Wiley, 2008. doi:10.1002/9780470753767.
- [30] N. Obrechhoff, Neue quadraturformeln, *Abh. Preuss. Akad. Wiss. Math. Nat. Kl.* 4 (1940) 1–20.
- [31] J. Butcher, G. Hojjati, Second derivative methods with rk stability, *Numer. Algorithms* 40 (2005) 415–429. doi:10.1007/s11075-005-0413-1.
- [32] R. Chan, A. Tsai, On explicit two-derivative runge-kutta methods, *Numer. Algor.* 53 (2010) 171–194. doi:10.1007/s11075-009-9349-1.
- [33] A. Tsai, R. Chan, S. Wang, Two-derivative runge–kutta methods for pdes using a novel discretization approach, *Numer. Algor.* 65 (2014) 687–703. doi:10.1007/s11075-014-9823-2.
- [34] E. Gekeler, On implicit runge-kutta methods with higher derivatives, *BIT* 28 (1988) 809–816.
- [35] T. Chan, R. Chan, A simplified approach to the order conditions of integration methods, *Computing* 77 (2006) 237–252. doi:10.1007/s00607-005-0151-1.
- [36] H. Ono, T. Yoshida, Two-stage explicit runge-kutta type methods using derivatives, *Japan J. Indust. Appl. Math.* 21 (2004) 361–374.
- [37] M. Akanbi, S. Okunuga, A. Sofoluwe, A family of 2-stage multiderivative explicit runge–kutta methods for non-stiff ordinary differential equations, *National Mathematical Centre* 6 (2005) 87–100.
- [38] J. Butcher, S. Tracogna, Order conditions for two-step runge-kutta methods, *Appl. Numer. Math.* 24 (1997) 351–364.
- [39] G. Ismail, I. Ibrahim, New efficient second derivative multistep methods for stiff systems, *Appl. Math. Model.* 23 (1999) 279–288. doi:10.1016/S0307-904X(98)10086-0.
- [40] G. Hojjati, M. R. Ardabili, S. Hosseini, New second derivative multistep methods for stiff systems, *Appl. Math. Model.* 30 (2006) 466–476.
- [41] A. Guillou, F. L. Soulé, La résolution numérique des problèmes différentiels aux conditions par des méthodes de collocation, *RAIRO Anal. Numér. Ser. Rouge* 3 (1969) 17–44.
- [42] K. Wright, Some relationships between implicit runge-kutta, collocation and lanczos methods, and their stability properties, *BIT Numerical Mathematics* 10 (1970) 217–227.
URL <https://api.semanticscholar.org/CorpusID:120474267>
- [43] D. Conte, R. D’Ambrosio, B. Paternoster, Advances on collocation based numerical methods for ordinary differential equations and volterra integral equations, in: T. E. Simos (Ed.), *Recent Advances in Computational and Applied Mathematics*, Springer Netherlands, Dordrecht, 2011, pp. 41–66.
- [44] S. Fazeli, G. Hojjati, Second derivative two-step collocation methods for ordinary differential equations, *Applied Numerical Mathematics* 156 (2020) 514–527. doi:<https://doi.org/10.1016/j.apnum.2020.05.019>.
URL <https://www.sciencedirect.com/science/article/pii/S016892742030163X>
- [45] K. Lee, I. Hashim, M. Mohd Aris, N. Senu, A novel two-derivative multistep collocation method with fitting-techniques with application to duffing problem, *Mathematics and Computers in Simulation* 239 (2026) 420–441. doi:<https://doi.org/10.1016/j.matcom.2025.05.024>.
URL <https://www.sciencedirect.com/science/article/pii/S0378475425002174>
- [46] A. Dutt, L. Greengard, V. Rokhlin, Spectral deferred correction methods for ordinary differential equations, *BIT Numerical Mathematics* 40 (2000) 241–266. doi:10.1023/A:1022338906936.
- [47] M. L. Minion, Semi-implicit spectral deferred correction methods for ordinary differential equations, *Communications in Mathematical Sciences* 1 (3) (2003) 471 – 500.
- [48] B. W. R. J. Spiteri, Deferred correction methods for ordinary differential equations, *J Sci Comput* 83:60 (2020). doi:10.1007/s10915-020-01235-8.

- [49] A. Christlieb, B. Ong, J.-M. Qiu, Comments on high-order integrators embedded within integral deferred correction methods, *Communications in Applied Mathematics and Computational Science* 4 (1) (2009) 27 – 56. doi:10.2140/camcos.2009.4.27. URL <https://doi.org/10.2140/camcos.2009.4.27>
- [50] J. Zeifang, J. Schütz, D. C. Seal, Stability of implicit multiderivative deferred correction methods, *Bit Numer Math* 62 (2022) 1487–1503. doi:10.1007/s10543-022-00919-x.
- [51] S. Clain, R. Pereira, P. Pereira, D. Lopes, Structural schemes for one dimension stationary equations, *Appl. Math. Comput.* 457 (2023). doi:10.1016/j.amc.2023.128207.
- [52] P. Chu, C. Fan, A three-point combined compact difference scheme, *J. Comput. Phys.* 140 (1998) 370–399.
- [53] P. Chu, C. Fan, A three-point sixth-order nonuniform combined compact difference scheme, *J. Comput. Phys.* 148 (1999) 663–674.
- [54] P. Chu, C. Fan, A three-point sixth-order staggered combined compact difference scheme, *Math. Comput. Model.* 32 (2000) 323–340.
- [55] S. Clain, G. Machado, M. Malheiro, Compact schemes in time with applications to partial differential equations, *Comput. Math. Appl.* 140 (2023). doi:10.1016/j.camwa.2023.03.011.
- [56] S. Gottlieb, Z. Grant, A review of high order strong stability preserving two-derivative explicit, implicit, and imex methods, *Appl. Math. Model.* 149 (2026) 116270. doi:10.1016/j.apm.2025.116270.
- [57] E. Hairer, A. Murua, J. Sanz-Serna, The non-existence of symplectic multi-derivative runge-kutta methods, *BIT* 34 (1994) 80–87. doi:10.1007/BF01935017.
- [58] T. Nguyen-Ba, V. Božić, E. Kengne, R. Vaillancourt, Nine-stage multi-derivative runge-kutta method of order 12, *Publications de l’institut mathématique, Nouvelle série* 86 (2009) 75–96. doi:10.2298/PIM0900075N.
- [59] T. Feagin, High-order explicit runge-kutta methods using m -symmetry, *Neural, Parallel & Scientific Computations* 30 (2012) 437.
- [60] A. M. amd J. Farzi, A. Abdi, Order conditions for second derivative general linear methods, *J. Comput. Appl. Math.* 387 (2021) 112488. doi:10.1016/j.cam.2019.112488.
- [61] X. Qin, Z. Jiang, J. Yu, L. Huang, C. Yan, Strong stability-preserving three-derivative runge-kutta methods, *Computational and Applied Mathematics* 42:171 (2023). doi:10.1007/s40314-023-02285-y.
- [62] A. Ezzeddine, G. Hojjati, Third derivative multistep methods for stiff systems, *Int. J. Nonlinear Sci.* 14 (2012) 443–450.
- [63] A. S. Wusu, M. A. Akanbi, S. A. Okunuga, A three-stage multiderivative explicit runge-kutta method, *Am. J. Comput. Math.* 3 (2013) 121–126. doi:10.4236/ajcm.2013.32020.
- [64] T. Majidi, A. Abdi, G. Hojjati, Generalized second derivative linear multistep methods for ordinary differential equations, *Numer. Algor.* 91 (2022) 227–250. doi:10.1007/s11075-022-01260-8.
- [65] T. Nguyen-Ba, V. Božić, E. Kengne, R. Vaillancourt, Nine-stage multi-derivative runge-kutta method of order 12, *Publications de l’institut mathématique Nouvelle série* 86 (2009) 75–96. doi:10.2298/PIM0900075N.
- [66] A. Abdi, G. Hojjati, Maximal order for second derivative general linear methods with runge-kutta stability, *Appl. Numer. Math.* 61 (2011) 1046–1058. doi:10.1016/j.apnum.2011.06.004.
- [67] R. I. Okuonghae1, M. N. O. Ikhile, $l(\alpha)$ -stable multi-derivative glm, *J. Algorithms Comput. Technol.* 9 (2015) 339–376. doi:10.1260/1748-3018.9.4.
- [68] D. C. Seal, Y. Güçlü, A. J. Christlieb, High-order multiderivative time integrators for hyperbolic conservation laws, *J. Sci. Comput.* 60 (2014) 101–140. doi:10.1007/s10915-013-9787-8.
- [69] A. Y. J. Tsai, R. P. K. Chan, S. Wang, Two-derivative runge-kutta methods for pdes using a novel discretization approach, *Numer. Algor.* 65 (2014) 687–703. doi:10.1007/s11075-014-9823-2.
- [70] J. Schütz, D. C. Seal, A. Jaust, Implicit multiderivative collocation solvers for linear partial differential equations with discontinuous galerkin spatial discretizations, *J. Sci. Comput.* 73 (2017) 1145—1163. doi:10.1007/s10915-017-0485-9.
- [71] J. Jaglan, A. Singh, V. Maurya, V. S. Yadav, M. K. Rajpoot, Strong stability preserving multiderivative time marching methods for stiff reaction-diffusion systems, *Math. Comput. Simul.* 225 (2024) 267–282. doi:10.1016/j.matcom.2024.05.020.
- [72] A. Christlieb, S. Gottlieb, Z. Grant, D. Seal, Explicit strong stability preserving multistage two-derivative time-stepping schemes, *J. Sci. Comput.* 68 (2016) 914–942. doi:10.1007/s10915-016-0164-2.

- [73] A. Moradi, A. Abdi, J. Farzi, Strong stability preserving second derivative general linear methods with runge–kutta stability, *J. Sci. Comput.* 85 (2020) 1–39. doi:10.1007/s10915-020-01306-w.
- [74] S. Gottlieb, Z. Grant, J. Hu, R. Shu, High order strong stability preserving multiderivative implicit and imex runge-kutta methods with asymptotic preserving properties, *SIAM J. Numer. Anal.* 69 (2022) 423–449. doi:10.1137/21M1403175.
- [75] A. Moradi, A. Abdi, G. Hojjati, Strong stability preserving implicit and implicit–explicit second derivative general linear methods with rk stability, *Comput. Appl. Math.* 41 (2022) 1–23. doi:/10.1007/s40314-022-01839-w.
- [76] X. Qin, Z. Jiang, J. Yu, L. Huang, C. Yan, Strong stability-preserving three-derivative runge–kutta methods, *Comput. Appl. Math.* 42 (2023) 171. doi:10.1007/s40314-023-02285-y.
- [77] A. Moradi, J. Farzi, A. Abdi, Strong stability preserving second derivative general linear methods, *Journal of Scientific Computing* 81 (2019) 392–435. doi:10.1007/s10915-019-01021-1.
- [78] Z. Grant, S. Gottlieb, D. Seal, A strong stability preserving analysis for explicit multistage two-derivative time-stepping schemes based on taylor series conditions, *Commun. Appl. Math. Comput.* 1 (2019) 21–59. doi:10.1007/s42967-019-0001-3.
- [79] A. Moradi, A. A. G. Hojjati, Rk-stable second derivative multistage methods with strong stability preserving based on taylor series conditions, *Computational and Applied Mathematics* 42 (2023) 193. doi:10.1007/s40314-023-02336-4.
- [80] S. Clain, E. Franck, V. Michel-Dansac, Structural schemes for hamiltonian systems (2025). arXiv:2501.13515.
URL <https://arxiv.org/abs/2501.13515>
- [81] H. Mizerová, K. Tvrdá, A new way of deriving implicit runge-kutta methods based on repeated integrals (2024). arXiv:2404.16665.
URL <https://arxiv.org/abs/2404.16665>
- [82] I. Lie, S. Norsett, Superconvergence for multistep collocation, *Math. Comp.* 52 (1989) 65–79. doi:10.1090/S0025-5718-1989-0971403-5.
- [83] I. Lie, The stability function for multistep collocation methods, *Numer. Math.* 57 (779-787) 1990. doi:10.1007/BF01386443.
- [84] S. Kowali, G. K. Gupta, Polynomial formulation of second derivative multistep methods, *Mathematics of Computation* 38 (1982) 447–458. doi:10.1090/S0025-5718-1982-0645662-6.
- [85] W. Gragg, H. Steeter, Generalized multistep predictor-corrector methods, *J. ACM* 11 (1964) 188–209. doi:10.1145/321217.321223.
- [86] A. S. Rai, U. Ananthakrishnaiah, Obrechhoff methods having additional parameters for general second-order differential equations, *Journal of Computational and Applied Mathematics* 79 (1997) 167–182. doi:10.1016/S0377-0427(96)00132-X.
- [87] B. Neta, T. Fukushima, Obrechhoff versus super-implicit methods for the solution of first- and second-order initial value problems, *Computers and Mathematics with Applications* 45 (2003) 383–390. doi:10.1016/S0898-1221(03)80024-X.
- [88] A. Shokri, the multistep multiderivative methods for the numerical solution of first order initial value problems, *TWMS J. Pure Appl. Math.* 7 (2016) 88–97.
- [89] C. Gear, Hybrid methods for initial value problems in ordinary differential equations, *SIAM J. Numer. Anal.* 2 (1964) 69–86. doi:10.1137/0702006.
- [90] A. Shokri, A. A. Shokri, The new class of implicit l-stable hybrid obrechhoff method for the numerical solution of first order initial value problems, *Computer Physics Communications* 184 (2013) 529–531. doi:j.cpc.2012.09.035.
- [91] P. Agarwall, I. H. Ibrahim, A new type of hybrid multistep multiderivative formula for solving stiff ivps, *Advances in Difference Equations* 286 (2019). doi:10.1186/s13662-019-2215-0.
- [92] S. Clain, Structural method library and examples, https://github.com/clain67/Structural_schemes.git (2026).
- [93] D. Greenspan, Numerical approximation of periodic solutions of van der pol’s equation, *J. Math. Anal. Appl.* 39 (1972) 574–579. doi:10.1016/0022-247X(72)90181-3.

- [94] Y. Xu, O. P. Agrawal, Models and numerical schemes for generalized van der pol equations, *Commun. Nonlinear Sci. Numer. Simulat.* 18 (2013) 3575–3589. doi:10.1016/j.cnsns.2013.04.022.
- [95] P. Amore, Computing the solutions of the van der pol equation to arbitrary precision, *Phys. D: Nonlinear Phenom.* 435 (2022) 133279. doi:10.1016/j.physd.2022.133279.
- [96] G. Chen, T. Ueta, Yet another chaotic attractor, *Int. J. Bifurc. Chaos* 9 (1999) 1465–1466. doi:10.1142/S0218127499001024.
- [97] T. Ueta, G. Chen, Bifurcation analysis of chen’s equation, *Internat. J. Bifur. Chaos* 8 (2000) 1917–1931. doi:10.1142/S0218127400001183.
- [98] J. Lu, T. Zhou, G. Chen, S. Zhang, Local bifurcation of the chen system, *Int. J. Bifurc. Chaos* 12 (2002) 2257–2270. doi:10.1142/S0218127402005819.
- [99] Introduction to Ordinary Differential Equations, Springer New York, New York, NY, 2006, pp. 1–144. doi:10.1007/0-387-35794-7_1.
URL https://doi.org/10.1007/0-387-35794-7_1
- [100] A. Poznyak, Classical and analytical Mechanics. Theory, Applied Examples, and Practice, Elsevier, 2021. doi:10.1016/C2020-0-01741-X.
- [101] A. Hurwitz, Ueber die bedingungen unter welchen eine gleichung nur wurzeln mit negativen reellen theilen besitzt, *Math. Ann.* 46 (1895) 273–284. doi:10.1007/BF01446812.
- [102] E. Routh, A treatise on the stability of a given state of motion: Particularly steady motion (1877).
- [103] L. Caban, A. Tyliczszak, High-order compact difference schemes on wide computational stencils with a spectral-like accuracy, *Comput. Math. Appl.* 108 (2022) 123–140. doi:10.1016/j.camwa.2022.01.006.