Multi-Step Hermite–Birkhoff Predictor-Corrector Schemes

A. Thenery Manikantan and J. Schütz

UHasselt Computational Mathematics Preprint Nr. UP-24-02

March 11th, 2024

Multi-Step Hermite–Birkhoff Predictor-Corrector Schemes

Arjun Thenery Manikantan[a,*], Jochen Schütz[a]

[a]*Faculty of Sciences & Data Science Institute, Hasselt University, Agoralaan Gebouw D, Diepenbeek, 3590, Belgium*

Abstract

In this study, we introduce a multi-step multi-derivative predictor-corrector time integration scheme analogous to the schemes in Schütz et al. (J Sci Comput 90(54):1–33, 2022), incorporating a multi-step quadrature rule. We conduct stability analysis up to order eight and optimize the schemes to achieve $A(\alpha)$-stability for large α. Numerical experiments are performed on ordinary differential equations exhibiting diverse stiffness conditions, as well as on partial differential equations showcasing non-linearity and higher-order terms. Results demonstrate the convergence and flexibility of the proposed schemes across diverse situations.

Keywords: Multi-derivative, Multi-step, Time integration, Predictor-corrector, IVPs
2020 MSC: 65L04, 65L05, 65L20, 65M22

1. Introduction

In this work, we consider the numerical solutions of initial value problems of form

$$\begin{aligned} y'(t) &= \phi(y), \quad t \in (0, T_{end}) \\ y(0) &= y_0. \end{aligned} \tag{1}$$

There are plenty of schemes in literature to find the numerical solution of the above system (1), which can be broadly categorized into explicit and implicit schemes. Implicit schemes are chosen over explicit schemes for better stability and less severe time-step restriction properties, especially when it comes to solving stiff problems. Implicit schemes, while advantageous in certain aspects, come with drawbacks. These include the need to solve nonlinear equations, potentially leading to added errors in the solution. Moreover, implicit schemes often demand longer execution times and encounter additional bottlenecks when solving larger systems.

In classical implicit time integration methods found in literature, the numerical scheme mostly relies on the first temporal derivative (y') of the problem. For one-derivative schemes, the orders of consistency can only be increased by incorporating additional stages or steps to the numerical scheme. At the same time, the addition of intermediate stages ends up with an arduous task of solving involved order conditions. In order to achieve higher-order schemes with fewer stages, one can include higher-order derivatives of the problem (1) to the scheme, which results in multi-derivative time stepping methods [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. In this paper, we focus on two-derivative schemes, which necessitate the computation of the second-order derivative of the problem (1)

$$y''(t) = \phi'(y)y'(t) = \phi'(y)\phi(y) =: \dot{\phi}(y).$$

In [11], the authors have introduced a fourth-order two-derivative asymptotic preserving IMEX time stepping scheme for solving stiff ODEs, which was structured in a predictor-corrector fashion. Later in

*Corresponding author
Email addresses: arjun.thenerymanikantan@uhasselt.be (Arjun Thenery Manikantan), jochen.schuetz@uhasselt.be (Jochen Schütz)

[12], the schemes were extended to higher-orders with parallel-in-time property, termed as **H**ermite-**B**irkhoff **P**redictor-**C**orrector (HBPC(q,$k_{\max}$)) schemes, where q is the maximum achievable order and $k_{\max}$ is the number of correction steps used. The shemes in [12] were optimized in [13] to have $A(\alpha)$ stability with α close to 90^0. The optimized HBPC schemes were applied to the discontinuous Galerkin method in [14, 15]. Explicitness-preserving IMEX-HBPC(q,$k_{\max}$) schemes were presented in [16] recently.

In order to achieve higher-orders, authors have used a higher-order quadrature rule in [12], which utilizes the intermediate stage values, see [12, Eq. (8)]. Generally, methods involving more intermediate implicit stages require longer execution time. Schemes with extended computational time pose a drawback when implemented for expansive systems, such as the Euler equations, Navier-Stokes equations, and more. One strategy for addressing this situation could involve investigating higher-order schemes that demand fewer intermediate implicit stages. Hence, an evident direction is to explore schemes involving multiple steps. Several multi-step multi-derivative schemes have been documented in the literature, including Brown's schemes [17, 18], second derivative BDF schemes [19], strong stability preserving schemes [20, 10], schemes for chemical stiff equations [21], general linear methods [22, 23] and more.

To integrate the concept of a multi-step method for HBPC(q,$k_{\max}$) schemes, we will explore a higher-order quadrature rule that leverages the previously computed solutions rather than relying solely on intermediate stages. This results in ***m-Step Hermite-Birkhoff Predictor-Corrector*** schemes abbreviated as mS-HBPC(q,$k_{\max}$). The parameter m represents the number of steps used from the previous time instances, q is the maximum achievable order of convergence, and $k_{\max}$ is the maximum number of correction steps. The 1S-HBPC(4,$k_{\max}$) is none other than the serial HBPC(4,$k_{\max}$) scheme in [12]. In this paper we analyze the stability properties and convergence of the mS-HBPC(q,$k_{\max}$) schemes for orders up to eight (up to three-step schemes).

The paper is structured as follows: In Sec. 2, the algorithm for the mS-HBPC(q,$k_{\max}$) schemes are presented, followed by the study of their stability properties in Sec. 3. The optimization of the mS-HBPC(q,$k_{\max}$) schemes for $A(\alpha)$-stability are discussed in further subsections of Sec. 3. Numerical results of the schemes on various test-cases that includes ordinary and partial differential equations, are shown in Sec. 4. Finally, the paper is concluded and an outlook is given in Sec. 5.

2. Numerical Scheme

We consider a fixed time-step for the scheme throughout the paper. For a given number of total time-steps N, we have

$$\Delta t := \frac{T_{end}}{N}.$$

The approximate solution at time instance

$$t^n := n\Delta t, \quad 0 \le n \le N,$$

is denoted as $y^n \approx y(t^n)$. The mS-HBPC(q,$k_{\max}$) scheme and the HBPC(q,$k_{\max}$) schemes from [12] utilize an approach similar to the spectrally deferred correction (SDC) method [24, 25, 26]. These schemes are designed to initiate with a predicted solution, represented by $y^{[0],n}$ at t^n. Then the predicted solution is subsequently refined through a sequence of correction steps. The k^{th} corrected solution at t^n is denoted as $y^{[k],n}$. At time t^{n+1}, the solution is updated with the highest corrected solution $y^{[k_{\max}],n}$. The detailed algorithm is defined below:

Algorithm 1 (mS-HBPC(q,$k_{\max}$))**.** *Given the solutions $y^n, y^{n-1}, y^{n-2}, \ldots, y^{n+1-m}$, the updated solution at t^{n+1} for $n \ge m-1$, is computed using the following prediction and correction steps. The predicted solution is computed using an implicit second-order Taylor scheme.*

1. ***Predict.*** *Solve the following expression for $y^{[0],n}$:*

$$y^{[0],n} := y^n + \Delta t \phi^{[0],n} - \frac{\Delta t^2}{2} \dot{\phi}^{[0],n}. \tag{2}$$

2. ***Correct.*** *Solve the following expression for* $y^{[k+1],n}$*, for each* $0 \leq k < k_{\max}$*:*

$$y^{[k+1],n} := y^n + \Delta t \theta_1 \left(\phi^{[k+1],n} - \phi^{[k],n}\right) - \frac{\Delta t^2}{2}\theta_2 \left(\dot{\phi}^{[k+1],n} - \dot{\phi}^{[k],n}\right) + \mathcal{I}\left(\phi^{n+1-m}, \ldots, \phi^{n-1}, \phi^n, \phi^{[k],n}\right), \tag{3}$$

where the q^{th} *order quadrature rule (see Tab. 1) is given by*

$$\mathcal{I}\left(\psi_1, \psi_2 \ldots, \psi_m, \psi_{m+1}\right) := \Delta t \sum_{i=1}^{m+1} b_i^{(1)} \psi_i + \Delta t^2 \sum_{i=1}^{m+1} b_i^{(2)} \dot{\psi}_i.$$

Note that θ_1 *and* θ_2 *are constants; more details are given in Remark. 4.*

3. ***Update.*** *Setting the solution at the final corrected step as the updated solution:*

$$y^{n+1} := y^{[k_{\max}],n}.$$

Remark 1. *In order to start up the procedure, the mS-HBPC(q,*$k_{\max}$*) scheme requires solutions at the first* m *time instances,*

$$y^\ell := y(t^\ell), \quad 0 \leq \ell \leq m-1,$$

which are either given or found using an appropriate solver. For the numerical results in this paper, we use the explicit solutions of the test problems, if available, for the start-up procedure. In cases where explicit solutions are inaccessible, we employ the MATLAB *solver* `ode15s` *[27] to compute a highly accurate solution.*

Remark 2. *The quadrature rule for the mS-HBPC(q,*$k_{\max}$*) scheme is derived by fitting a Hermite–Birkhoff polynomial interpolant through the uniformly spaced time instances* $t^{n+1}, t^n, t^{n-1}, \ldots, t^{n+1-m}$ *and integrating the result from* t^n *to* t^{n+1}*. This constructions results into a quadrature rule,*

$$\int_{t^n}^{t^{n+1}} \psi(y(t))dt = \mathcal{I}\left(\psi(y(t^{n+1-m}), \ldots, \psi(y(t^{n-1}), \psi(y(t^n), \psi(y(t^{n+1}))\right) + \mathcal{O}(\Delta t^{q+1}).$$

The $m+1$ *points gives the Hermite–Birkhoff polynomial of order* $2m+1$*, which gives an integral approximation of order* $(2m+1)+2 =: q+1$*. Hence, we have*

$$q = 2(m+1).$$

The quadrature rules for the schemes up to order eight are given in Tab. 1.

Remark 3. *Similar to the results from [11, 12], in every iteration, for each correction step* $k \leq k_{\max}$*, the mS-HBPC(q,*$k_{\max}$*) provides an approximation to* $y(t^{n+1})$ *with an order of accuracy of* $\min\{q, 2+k\}$*, i.e.*

$$y(t^{n+1}) = y^{[k],n} + \mathcal{O}(\Delta t^{\min\{q,2+k\}}).$$

Convergence results are shown numerically in Sec. 4.1.

Remark 4. *The parameters* $(\theta_1, \theta_2) \in \mathbb{R}^2$ *utilized in the correction step (3) are the tuning parameters aimed at optimizing the algorithm to ensure better stability properties. Sec. 3 of the paper delves into the analysis of stability and the optimization of parameters* (θ_1, θ_2)*.*

Scheme	Quadrature rule							
	$\boldsymbol{b}^{(1)}$				$\boldsymbol{b}^{(2)}$			
1S-HBPC(4,$k_{\max}$)[11]	$\frac{1}{2}$	$\frac{1}{2}$			$\frac{1}{12}$	$-\frac{1}{12}$		
2S-HBPC(6,$k_{\max}$)	$\frac{11}{240}$	$\frac{128}{240}$	$\frac{101}{240}$		$\frac{3}{240}$	$\frac{40}{240}$	$-\frac{13}{240}$	
3S-HBPC(8,$k_{\max}$)	$\frac{1985}{90720}$	$\frac{12015}{90720}$	$\frac{42255}{90720}$	$\frac{34465}{90720}$	$\frac{489}{90720}$	$\frac{7263}{90720}$	$\frac{22977}{90720}$	$-\frac{3849}{90720}$

Table 1: The quadrature rules for the schemes up to order eight. The methodology for constructing these quadrature rules is outlined in Remark. 2.

3. Stability Analysis

In this section, the linear stability of the mS-HBPC(q,$k_{\max}$) method is analyzed using Dahlquist's equation $y' = \lambda y$, where $\lambda \in \mathbb{C}$. Define $z := \Delta t\lambda$ and the following functions:

$$\mathcal{R}_m^{[0]}(z) := \frac{2}{2-2z+z^2} \quad \text{and} \quad \mathcal{R}_i^{[0]}(z) := 0,\ \forall i < m,$$
$$\mathcal{P}_m(z) := 1 + b_m^{(1)} z + b_m^{(2)} z^2 \quad \text{and} \quad \mathcal{P}_i(z) := b_i^{(1)} z + b_i^{(2)} z^2,\ \forall i < m,$$
$$\mathcal{S}(z) := \{b_{m+1}^{(1)} - \theta_1\} z + \{b_{m+1}^{(2)} + \frac{\theta_2}{2}\} z^2 \quad \text{and} \quad \mathcal{T}(z) := 1 - \theta_1 z + \frac{\theta_2}{2} z^2.$$

For a k^{th} correction step, define the functions $\mathcal{R}_\ell^{[k]}$ for $1 \le \ell \le m$ such that

$$y^{[k]} =: \sum_{i=0}^{m-1} \mathcal{R}_{m-i}^{[k]}(z)\ y^{n-i}.$$

Apply the mS-HBPC(q,$k_{\max}$) scheme as defined in Alg. 1 for the Dahlquist's equation $y' = \lambda y$. The predicted solution can be written as

$$y^{[0]} = \sum_{i=0}^{m-1} \mathcal{R}_{m-i}^{[0]}(z)\ y^{n-i} = \mathcal{R}_m^{[0]}(z) y^n.$$

Then for the k^{th} correction step, we have

$$y^{[k]} = \frac{\mathcal{S}(z) y^{[k-1]} + \sum_{i=0}^{m-1} \mathcal{P}_{m-i}(z)\ y^{n-i}}{\mathcal{T}(z)} = \frac{\mathcal{S}(z) \sum_{i=0}^{m-1} \mathcal{R}_{m-i}^{[k-1]}(z)\ y^{n-i} + \sum_{i=0}^{m-1} \mathcal{P}_{m-i}(z)\ y^{n-i}}{\mathcal{T}(z)}$$

$$= \sum_{i=0}^{m-1} \frac{\mathcal{S}(z)\mathcal{R}_{m-i}^{[k-1]}(z) + \mathcal{P}_{m-i}(z)}{\mathcal{T}(z)}\ y^{n-i} =: \sum_{i=0}^{m-1} \mathcal{R}_{m-i}^{[k]}(z)\ y^{n-i}.$$

Hence the functions $\mathcal{R}_\ell^{[k]}$ are recursively given by

$$\mathcal{R}_\ell^{[k]}(z) = \frac{\mathcal{S}(z)\mathcal{R}_\ell^{[k-1]}(z) + \mathcal{P}_\ell(z)}{\mathcal{T}(z)} \quad \text{for } 1 \le \ell \le m.$$

Therefore the updated solution at t^{n+1} can be obtained as

$$y^{n+1} = y^{[k_{\max}]} = \sum_{i=0}^{m-1} \mathcal{R}_{m-i}^{[k_{\max}]}(z)\ y^{n-i}$$

$$=: \sum_{i=0}^{m-1} \mathcal{R}_{m-i}(z)\ y^{n-i}. \tag{4}$$

Assume a solution of type $y^s := r^s$ to the difference equation Eq. (4) and substituting it in Eq. (4) leads to the following polynomial of degree m,

$$r^m - \mathcal{R}_m(z)\, r^{m-1} - \mathcal{R}_{m-1}(z)\, r^{m-2} \cdots - \mathcal{R}_2(z)\, r - \mathcal{R}_1(z) = 0. \tag{5}$$

If $r_i(z)$ are roots of the polynomial (5), then the stability of the scheme mandates each root $r_i(z)$ to have an absolute value less than or equal to one.

Theorem 1. *The multi-step HBPC schemes mS-HBPC(q,$k_{\max}$) are zero-stable for any values of (θ_1, θ_2).*

Proof. Substituting $\lambda = 0$ into the defined functions yields,

$$\begin{aligned} \mathcal{R}_m^{[0]}(z) = 1 \quad &\text{and} \quad \mathcal{R}_i^{[0]}(z) = 0, \ \forall i < m, \\ \mathcal{P}_m(z) = 1 \quad &\text{and} \quad \mathcal{P}_i(z) = 0, \ \forall i < m, \\ \mathcal{S}(z) = 0 \quad &\text{and} \quad \mathcal{T}(z) = 1, \end{aligned}$$

which are independent of (θ_1, θ_2) values. Therefore, we get

$$\mathcal{R}_\ell^{[k]}(z) = \begin{cases} 1, & \ell = m, \\ 0, & \ell \neq m. \end{cases}$$

for every k. Then the polynomial (5) is

$$r^m - r^{m-1} = 0,$$

with roots $r_i \in \{0, 1\}$, and hence their magnitude $|r_i| \leq 1$. Being the multiplicity of the root $r_m = 1$ equal to one completes the proof. □

Define the function

$$\mathcal{R}_{max}(z) := \max_i \{|r_i(z)|\}. \tag{6}$$

Then the stability region for the numerical scheme is given by

$$\text{Stability Region} = \{z \in \mathbb{C} \ | \mathcal{R}_{max}(z) \leq 1\}, \tag{7}$$

under the assumption that the polynomial (5) has only single roots. Since the scheme 1S-HBPC(4,$k_{\max}$) is the same as scheme HBPC(4,$k_{\max}$) in [11], it has been shown already in [13] that the scheme is $A(\alpha)$-stable, and A-stable with optimized coefficients $(\theta_1, \theta_2) = \left(\frac{1}{2}, \frac{1}{6}\right)$. The plots in Fig. 1 show the stability regions of the 2S-HBPC(6,4) (left) and 3S-HBPC(8,6) (right) schemes with values θ_1 and θ_2 equal to one. The shaded

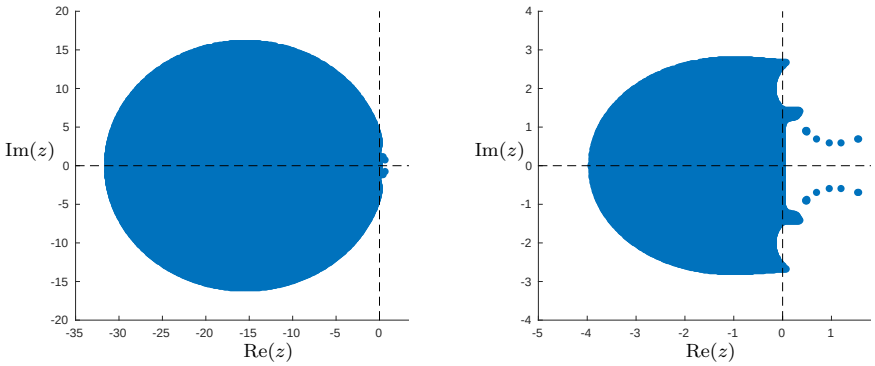


Figure 1: Stability regions of 2S-HBPC(6,4) (left) and 3S-HBPC(8,6) (right) schemes with $(\theta_1, \theta_2) = (1, 1)$.

regions in Fig. 1 show that the stability regions are bounded. To harness the advantage of an unconstrained

time-step condition offered by an implicit scheme, it is crucial to possess a stability region that extends infinitely, contrasting with the finite boundaries illustrated in Fig. 1. Hence, the tuning parameters θ_1 and θ_2 are to be brought into service for expanding the stability region. In the following section, the tuning parameters are optimized to obtain an $A(\alpha)$-stable mS-HBPC(q,$k_{\max}$) scheme. In this paper, we specifically limit the optimization process to schemes 2S-HBPC(6,4) and 3S-HBPC(8,6).

3.1. Constraints on tuning parameters for $A(\alpha)$-stability

Theorem 2. *There exist positive (θ_1, θ_2) values such that the multi-step HBPC schemes 2S-HBPC(6,4) and 3S-HBPC(8,6) are $A(\alpha)$-stable.*

Proof. The objective is to determine conditions on tuning parameters by examining the absolute values of the roots of the difference equation (4) as $-\infty$ approaches. The complete proof of the theorem follows from Lemma. 1 and Lemma. 2. □

Remark 5. *We use the Symbolic Math Toolbox in* MATLAB *[28] for solving/simplifying/deriving certain equations/expressions in the paper due to their complexity. In particular,*

- *to find the roots, and their limits of the polynomials (8) and (9), in the proofs of Lemma. 1 and Lemma. 2, respectively;*
- *to find the error constant $\mathcal{C}(\theta_1, \theta_2)$ defined in Sec. 3.2.1 and Sec. 3.2.2.*

Remark 6 (Methodology for evaluating stability angle)**.** *We utilize the algorithm outlined in [13, Sec. 3] to compute the stability angles for $A(\alpha)$-stable mS-HBPC(q,$k_{\max}$) schemes.*

Lemma 1. *The 2S-HBPC(6,4) scheme is $A(\alpha)$-stable for any $\theta_1 > 0$ and $\theta_2 \geq 1.25868$. The minimum value of θ_2 provided is rounded to five decimal places.*

Proof. Begin with the polynomial equation in (5) for the 2S-HBPC(6,4) scheme

$$r^2 - \mathcal{R}_2(z)\, r - \mathcal{R}_1(z) = 0. \tag{8}$$

Its roots are given by

$$r_1(z) = \frac{\mathcal{R}_2(z)}{2} + \sqrt{\frac{\mathcal{R}_2^2(z)}{4} + \mathcal{R}_1(z)},$$
$$r_2(z) = \frac{\mathcal{R}_2(z)}{2} - \sqrt{\frac{\mathcal{R}_2^2(z)}{4} + \mathcal{R}_1(z)}.$$

The limits of the roots $r_1(z)$ and $r_2(z)$ as z tends to $-\infty$ result into functions that depends on θ_2 only. It is because θ_2 occurs in the algorithm Alg. 1 with terms involving Δt^2, whereas θ_1 pair up with Δt terms. Therefore, we can write

$$\lim_{z\to-\infty} r_1(z) =: g_1(\theta_2),$$
$$\lim_{z\to-\infty} r_2(z) =: g_2(\theta_2).$$

As we need to expand the stability region, conditions on θ_2 can be found by analyzing the absolute values of $g_1(\theta_2)$ and $g_2(\theta_2)$.

In Fig. 2a it is shown that $|g_2(\theta_2)| < 1$ for all the given θ_2 values whereas $|g_1(\theta_2)| < 1$ only for values of $\theta_2 \gtrsim 1.25868$. Hence we have $\forall \theta_2 \geq 1.25868$

$$\lim_{z\to-\infty} \mathcal{R}_{max}(z) < 1.$$

The stability angles for various (θ_1, θ_2) values are given in Tab. 2 for A(α)-stable 2S-HBPC(6,4). As illustrated in Fig. 3 and Tab. 2, it is evident that the 2S-HBPC(6,4) scheme exhibits $A(\alpha)$-stability for θ_2 greater than or equal to 1.25868. The stability angle corresponding to the pair $(\theta_1, \theta_2) = (1, 1.25868)$ is 83.64°. □

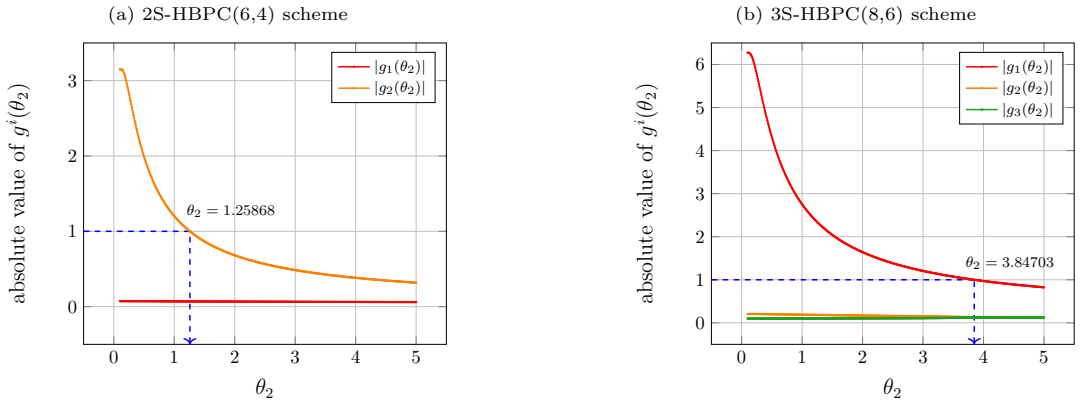


Figure 2: Absolute values of the limits of the roots are plotted against different θ_2 values. The value of θ_2 for which $\max_i |g_i(\theta_2)| \leq 1$ is marked on the plot. This value of θ_2 ensures A(α)-stability for the scheme.

Lemma 2. *The 3S-HBPC(8,6) scheme is $A(\alpha)$-stable for any $\theta_1 > 0$ and $\theta_2 \geq 3.84703$. The minimum value of θ_2 provided is rounded to five decimal places.*

Proof. Begin with the polynomial equation in (5) for the 3S-HBPC(8,6) scheme

$$r^3 - \mathcal{R}_3(z)\, r^2 - \mathcal{R}_2(z)\, r - \mathcal{R}_1(z) = 0. \tag{9}$$

Here, we have a third degree polynomial and its roots can be found explicitly using Cardano's formula for cubic polynomials. The roots $r_1(z)$, $r_2(z)$ and $r_3(z)$ are not provided here due to their lengthy formulations. Similar to 2S-HBPC(6,4), the limits of the roots $r_1(z)$, $r_2(z)$ and $r_3(z)$ as z tends to $-\infty$ also result into functions that depends on θ_2 alone,

$$\lim_{z\to-\infty} r_k(z) =: g_k(\theta_2), \quad 1 \leq k \leq 3.$$

The explicit limits of the roots are omitted here due to the extended nature of the expressions.

In Fig. 2b, it is shown that $|g^2(\theta_2)| < 1$ and $|g^3(\theta_2)| < 1$ for all the given θ_2 values whereas $|g^1(\theta_2)| < 1$ only for values of $\theta_2 \gtrsim 3.84703$. Hence we have $\forall \theta_2 \geq 3.84703$

$$\lim_{z\to-\infty} \mathcal{R}_{max}(z) < 1.$$

The stability angles for various (θ_1, θ_2) values are given in Tab. 2 for A(α)-stable 3S-HBPC(8,6). From Fig. 4 and Tab. 2, it is evident that the 3S-HBPC(8,6) scheme exhibits $A(\alpha)$-stability for values of θ_2 greater than or equal to 3.84703. The stability angle corresponding to the pair $(\theta_1, \theta_2) = (1, 1.25868)$ is 78.93°. □

The detailed analysis on (θ_1, θ_2) values for 2S-HBPC(6,4) and 3S-HBPC(8,6) schemes are given in Sec. 3.2.

3.2. Optimization of the tuning parameters

When analyzing the stability angles in Tab. 2 for various (θ_1, θ_2) values with the minimum requirement on the θ_2 value for $A(\alpha)$-stability, it is observed that there is an increase in the stability angle as we increase θ_1 keeping θ_2 fixed, whereas a decrease in the stability angle as we increase θ_2 keeping θ_1 fixed. In this section, we seek optimized tuning parameters that can minimize the one-step error when using a linear test problem.

Consider the test problem

$$y' = y, \quad y(0) = 1,$$

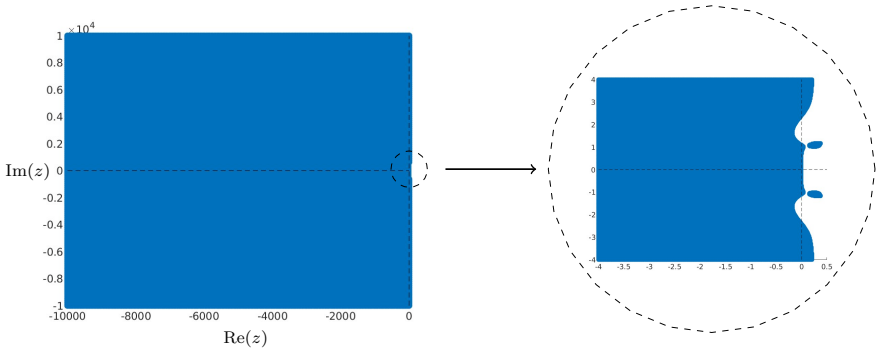


Figure 3: Stability region of 2S-HBPC(6,4) scheme for $(\theta_1, \theta_2) = (1, 1.25868)$ with zoomed image on the right.

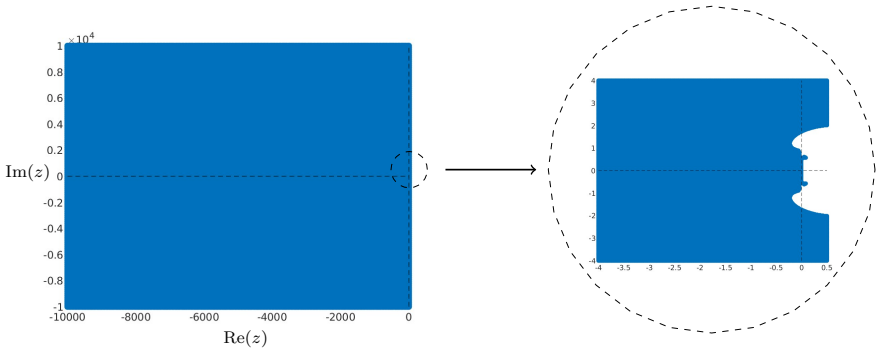


Figure 4: Stability region of 3S-HBPC(8,6) scheme for $(\theta_1, \theta_2) = (1, 3.84703)$ with zoomed image on the right.

which admits an explicit solution $y(t) = e^t$. The one-step error $\mathcal{C}(\theta_1, \theta_2, \Delta t)$ for an mS-HBPC(q,$k_{\max}$) scheme is defined as

$$\mathcal{C}(\theta_1, \theta_2, \Delta t) := \left| \frac{e^{m\Delta t} - \left(\sum_{i=0}^{m-1} \mathcal{R}_{i+1}(\Delta t) e^{i\Delta t} \right)}{\Delta t^{q+1}} \right|. \tag{10}$$

3.2.1. 2S-HBPC(6,4) scheme

Begin with the error constant $\mathcal{C}(\theta_1, \theta_2)$, which is obtained by considering $\lim_{\Delta t \to 0} \mathcal{C}(\theta_1, \theta_2, \Delta t)$. It is given by

$$\mathcal{C}(\theta_1, \theta_2) = \left| \frac{1}{6}\theta_1^4 - \frac{101}{360}\theta_1^3 + \frac{10201}{57600}\theta_1^2 - \frac{1030301}{20736000}\theta_1 + \frac{743168407}{139345920000} \right|. \tag{11}$$

It can be noted that the error constant is independent of θ_2. Therefore, it is plotted against various θ_1 values in Fig. 5a.

It can be seen from Fig. 5a that the error constant reaches its minimum value at $\theta_1 \approx 0.42083$. Hence we consider only values of θ_1 starting from 0.42083 to obtain different stability angles.

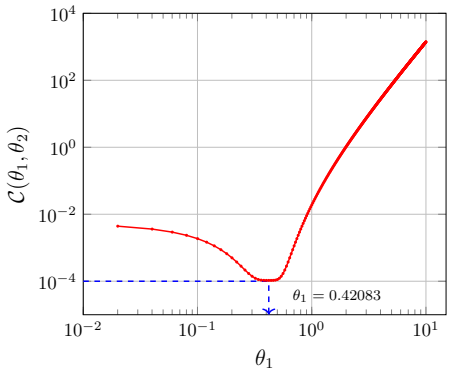
Plot in Fig. 6 shows the one-step error $\mathcal{C}(\theta_1, \theta_2, \Delta t)$ versus timestep graph for various (θ_1, θ_2) pairs corresponding to stability angles approximately equal to 77.85° (left) and 85° (right) respectively. It can be observed from Fig. 6 that the one-step error $\mathcal{C}(\theta_1, \theta_2, \Delta t)$ corresponding to a given stability angle gives a considerably minimal value for lowest (θ_1, θ_2) values. Hence, for an $A(\alpha)$-stable 2S-HBPC(6,4) scheme we fix θ_2 at the minimum value of 1.25868 (rounded to five decimal places). To enhance the optimization of the scheme, Fig. 8a illustrates the stability angles (α) across various θ_1 values. It can be seen from that Fig. 8a that the stability angle (α) increases with increased θ_1 values.

$\theta_1 \downarrow\ \theta_2 \rightarrow$	1.25868	1.5	2	2.5	3	3.5	4	4.5	5
0.5	78.9	78.2	77.3	76.8	76.5	76.3	76.2	76.2	76.2
0.6	80.1	79.3	78.3	77.7	77.3	77.1	77.0	76.9	76.9
0.7	81.2	80.4	79.2	78.6	78.2	77.9	77.8	77.7	77.6
0.8	82.1	81.3	80.1	79.4	78.9	78.7	78.5	78.3	78.3
0.9	82.9	82.1	80.9	80.2	79.7	79.3	79.1	79.0	78.9
1.0	83.6	82.8	81.6	80.8	80.3	80.0	79.8	79.6	79.5
1.1	84.3	83.5	82.3	81.5	81.0	80.6	80.3	80.2	80.0
1.2	84.8	84.0	82.9	82.1	81.5	81.2	80.9	80.7	80.6
1.3	85.2	84.5	83.4	82.6	82.1	81.7	81.4	81.2	81.1
1.4	85.6	85.0	83.9	83.1	82.6	82.2	81.9	81.7	81.5
1.5	85.9	85.3	84.3	83.6	83.0	82.6	82.3	82.1	82.0
1.6	86.2	85.7	84.7	84.0	83.4	83.0	82.8	82.5	82.4
1.7	86.5	85.9	85.0	84.3	83.8	83.4	83.1	82.9	82.8
1.8	86.6	86.2	85.4	84.7	84.2	83.8	83.5	83.3	83.1
1.9	86.8	86.4	85.6	85.0	84.5	84.1	83.9	83.6	83.5
2.0	87.0	86.6	85.9	85.3	84.8	84.5	84.2	83.9	83.8
2.1	87.1	86.8	86.1	85.6	85.1	84.8	84.5	84.2	84.1
2.2	87.2	86.9	86.3	85.8	85.4	85.0	84.7	84.5	84.4
2.3	87.3	87.0	86.5	86.0	85.6	85.3	85.0	84.8	84.6
2.4	87.4	87.2	86.7	86.2	85.8	85.5	85.2	85.0	84.9
2.5	87.5	87.3	86.8	86.4	86.0	85.7	85.5	85.3	85.1
2.6	87.6	87.4	86.9	86.6	86.2	85.9	85.7	85.5	85.3
2.7	87.6	87.5	87.1	86.7	86.4	86.1	85.9	85.7	85.5
2.8	87.7	87.5	87.2	86.8	86.5	86.3	86.1	85.9	85.7
2.9	87.8	87.6	87.3	87.0	86.7	86.4	86.2	86.1	85.9
3.0	87.8	87.7	87.4	87.1	86.8	86.6	86.4	86.2	86.1

$\theta_1 \downarrow\ \theta_2 \rightarrow$	3.84703	4.5	5	5.5	6	6.5	7	7.5	8
0.5	76.2	76.0	75.9	75.8	75.7	75.6	75.5	75.5	75.4
0.6	76.8	76.6	76.4	76.3	76.2	76.1	76.0	75.9	75.9
0.7	77.4	77.1	76.9	76.7	76.6	76.5	76.4	76.4	76.3
0.8	77.9	77.6	77.4	77.2	77.1	77.0	76.9	76.8	76.7
0.9	78.4	78.1	77.9	77.7	77.5	77.4	77.3	77.2	77.1
1.0	78.9	78.5	78.3	78.1	78.0	77.8	77.7	77.6	77.5
1.1	79.4	79.0	78.7	78.5	78.4	78.2	78.1	78.0	77.9
1.2	79.8	79.4	79.2	78.9	78.8	78.6	78.5	78.4	78.3
1.3	80.3	79.8	79.6	79.3	79.2	79.0	78.9	78.8	78.7
1.4	80.7	80.2	79.9	79.7	79.5	79.4	79.2	79.1	79.0
1.5	81.1	80.6	80.3	80.1	79.9	79.7	79.6	79.5	79.4
1.6	81.4	81.0	80.7	80.4	80.2	80.1	79.9	79.8	79.7
1.7	81.8	81.3	81.0	80.8	80.6	80.4	80.2	80.1	80.0
1.8	82.1	81.6	81.3	81.1	80.9	80.7	80.6	80.4	80.3
1.9	82.4	81.9	81.6	81.4	81.2	81.0	80.9	80.7	80.6
2.0	82.7	82.2	81.9	81.7	81.5	81.3	81.1	81.0	80.9
2.1	83.0	82.5	82.2	82.0	81.8	81.6	81.4	81.3	81.2
2.2	83.3	82.8	82.5	82.3	82.0	81.9	81.7	81.6	81.5
2.3	83.5	83.1	82.8	82.5	82.3	82.1	82.0	81.8	81.7
2.4	83.8	83.3	83.0	82.8	82.5	82.4	82.2	82.1	82.0
2.5	83.9	83.4	83.1	82.9	82.7	82.5	82.3	82.2	82.1
2.5	84.0	83.5	83.2	83.0	82.8	82.6	82.5	82.3	82.2
2.6	84.2	83.8	83.5	83.2	83.0	82.8	82.7	82.6	82.4
2.7	84.4	84.0	83.7	83.4	83.2	83.1	82.9	82.8	82.7
2.8	84.6	84.2	83.9	83.7	83.5	83.3	83.1	83.0	82.9
2.9	84.8	84.4	84.1	83.9	83.7	83.5	83.3	83.2	83.1
3.0	84.9	84.5	84.3	84.0	83.9	83.7	83.5	83.4	83.3

Table 2: Stability angles (in degrees) for various (θ_1, θ_2) values for $A(\alpha)$-stable 2S-HBPC(6,4) (left) and 3S-HBPC(8,6) (right) schemes, respectively. The values have been rounded to one decimal place.

(a) 2S-HBPC(6,4) scheme



(b) 3S-HBPC(8,6) scheme

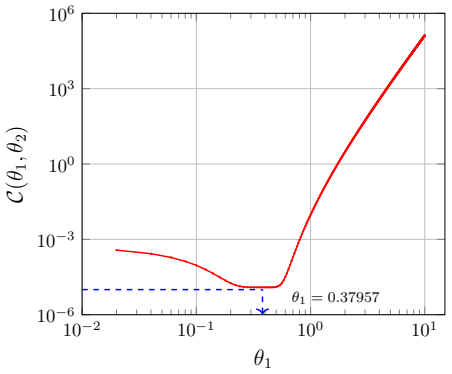


Figure 5: Error constant for sixth and eighth-order schemes plotted for various θ_1 values. The value of θ_1, for which $\mathcal{C}(\theta_1, \theta_2)$ is minimum, is marked on the plot.

3.2.2. 3S-HBPC(8,6) scheme

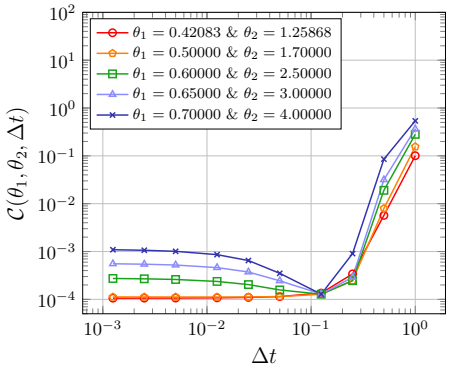
Similar to the sixth-order scheme, the error constant $\mathcal{C}(\theta_1, \theta_2)$ for 3S-HBPC(8,6) scheme is given by

$$\mathcal{C}(\theta_1, \theta_2) = \Bigg| \frac{1}{6}\theta_1^6 - \frac{6893}{18144}\theta_1^5 + \frac{237567245}{658409472}\theta_1^4 - \frac{1637551019785}{8959636094976}\theta_1^3 + \frac{11287639179378005}{216751516409659392}\theta_1^2 - \frac{15561139372690517693}{1966369756868430004224}\theta_1 + \frac{2747516956745302596230737}{535167193029311909949603840 0}\Bigg|. \quad (12)$$

It can be observed from Fig. 5b that the error constant reaches its minimum value at $\theta_1 \approx 0.37957$. Hence we consider only values of θ_1 starting from 0.37957 to obtain different stability angles.

Plots in Fig. 7 shows the one-step error $\mathcal{C}(\theta_1, \theta_2, \Delta t)$ versus timestep graph for various (θ_1, θ_2) pairs corresponds to stability angles approximately equal to 75.45° (left) and 80° (right) respectively. Likewise as in the sixth-order scheme, it can be noted from Fig. 7 that the one-step error $\mathcal{C}(\theta_1, \theta_2, \Delta t)$ corresponding to a given stability angle gives a considerably minimal value for lowest (θ_1, θ_2) values. Therefore, to achieve an $A(\alpha)$-stable 2S-HBPC(6,4) scheme we fix θ_2 at the minimum value of 3.84703 (rounded to five decimal

(a) Stability angle (α)$\approx$ 77.85°



(b) Stability angle (α)$\approx$ 85°

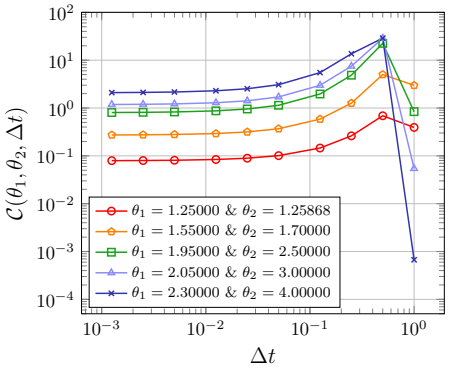
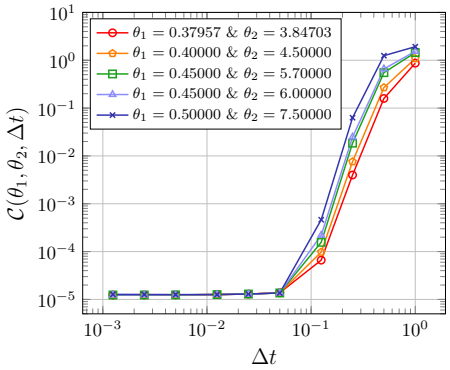


Figure 6: The one-step error $\mathcal{C}(\theta_1, \theta_2, \Delta t)$ for 2S-HBPC(6,4) scheme is plotted against different timesteps Δt.

(a) Stability angle (α)$\approx$ 75.45°



(b) Stability angle (α)$\approx$ 80°

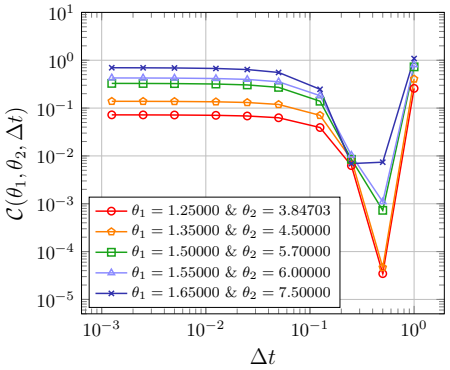


Figure 7: The one-step error $\mathcal{C}(\theta_1, \theta_2, \Delta t)$ for 3S-HBPC(8,6) scheme is plotted against different timesteps Δt.

places). To enhance the optimization of the scheme, Fig. 8b illustrates the stability angles (α) across various θ_1 values. An increment in the stability angle (α) with increased θ_1 values can be observed from Fig. 8b.

3.3. Optimized tuning parameters for numerical results

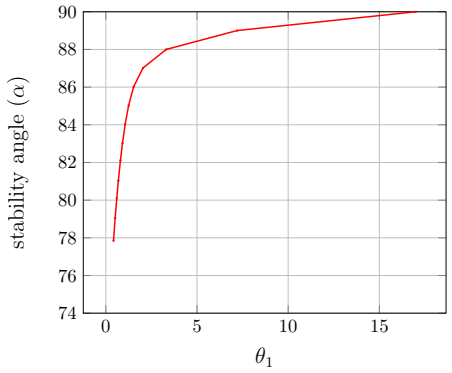
We found constraints on the tuning parameters in the preceding sections that minimize one-step errors (10) and the error constant (11)(12) using a linear test problem. Consequently, the values obtained from these procedures are employed for the numerical results in the forthcoming section. Some of these values are presented in Tab. 3.

4. Numerical Results

For the convergence analysis, we calculate the Euclidean error denoted as $\|y - y_h\|_2$, comparing the approximate solution y_h using the explicit solution $y(t)$. Otherwise, a highly accurate solution is found numerically for the error comparison. The error is calculated at a given final time T_{end}.

Definition 1. *The **unoptimized** mS-HBPC(q,$k_{\max}$) schemes used in the numerical results refers to schemes with tuning parameters* $(\theta_1, \theta_2) = (1, 1)$.

(a) 2S-HBPC(6,4) scheme with $\theta_2 = 1.25868$.



(b) 3S-HBPC(8,6) scheme with $\theta_2 = 3.84703$.

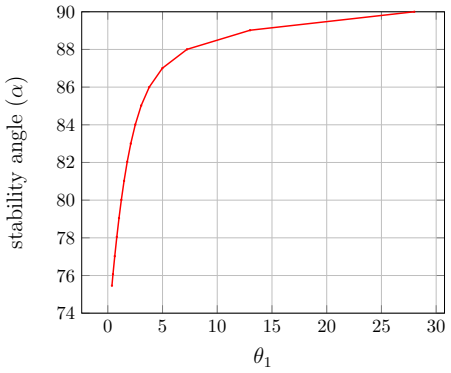


Figure 8: Stability angles α for the $A(\alpha)$-stable sixth and eighth-order schemes plotted for varying θ_1 values.

2S-HBPC(6,4) ($\theta_2 = 1.25868$)								
Approx. stability angle (α)	77.85°	80.00°	85.00°	86.00°	87.00°	88.00°	89.00°	89.99°
θ_1	0.42083	0.60000	1.25000	1.52500	2.03750	3.31250	7.2000	17.0000
Error constant	$1.05{\cdot}10^{-4}$	$2.77{\cdot}10^{-4}$	$7.88{\cdot}10^{-2}$	$2.47{\cdot}10^{-1}$	$1.13 \cdot 10^{0}$	$1.16 \cdot 10^{1}$	$3.52 \cdot 10^{2}$	$1.25 \cdot 10^{4}$
3S-HBPC(8,6) ($\theta_2 = 3.84703$)								
Approx. stability angle (α)	75.45°	80.00°	85.00°	86.00°	87.00°	88.00°	89.00°	89.99°
θ_1	0.37957	1.23750	3.05000	3.78750	4.98750	7.23750	13.0000	28.0000
Error constant	$1.23{\cdot}10^{-5}$	$6.63{\cdot}10^{-2}$	$6.03 \cdot 10^{1}$	$2.60 \cdot 10^{2}$	$1.59 \cdot 10^{3}$	$1.73 \cdot 10^{4}$	$6.73 \cdot 10^{5}$	$7.39 \cdot 10^{7}$

Table 3: Stability optimized tuning parameters for $A(\alpha)$-stable 2S-HBPC(6,4) and 3S-HBPC(8,6) schemes.

4.1. Convergence

We consider the non-stiff ODE

$$y'(t) = -y^{-\frac{5}{2}}, \quad y_0 = 1, \tag{13}$$

for showing the convergence of the mS-HBPC(q,$k_{\max}$) schemes. The explicit solution of the problem (13) is

$$y(t) = \left(y_0^{\frac{7}{2}} - \frac{7}{2}t\right)^{\frac{2}{7}}.$$

The error is calculated at $T_{end} = 0.25$.

In Fig. 9, the convergence results are shown for 2S-HBPC(6,4) (top) and 3S-HBPC(8,6) (bottom) for various correction steps k. The results are compared for unoptimized and $A(\alpha)$-stable mS-HBPC(q,$k_{\max}$) schemes. It can be seen from Fig. 9 that the unoptimized (first column) and $A(85^o)$-stable (third column), 2S-HBPC(6,4) (top) and 3S-HBPC(8,6) (bottom) schemes, clearly follows the trend of order increment by one after each correction step. This observation provides evidence in favor of the theoretical order of convergence of $\min\{q, 2+k\}$ for a k-th correction step. However, for $A(77.85°)$-stable 2S-HBPC(6,4) (second column, top) and $A(75.45°)$-stable 3S-HBPC(8,6) (second column, bottom) schemes, there is an order increment by two until the second and third correction steps, respectively. This exception arises from the dominance of $\mathcal{O}(\Delta t^2)$ terms over $\mathcal{O}(\Delta t)$ terms in the correction steps for parameters $\theta_1 \ll \theta_2$. Eventually, the schemes, 2S-HBPC(6,4) and 3S-HBPC(8,6) achieve their desired order of convergences six ($k_{\max} = 4$) and eight ($k_{\max} = 6$), respectively.

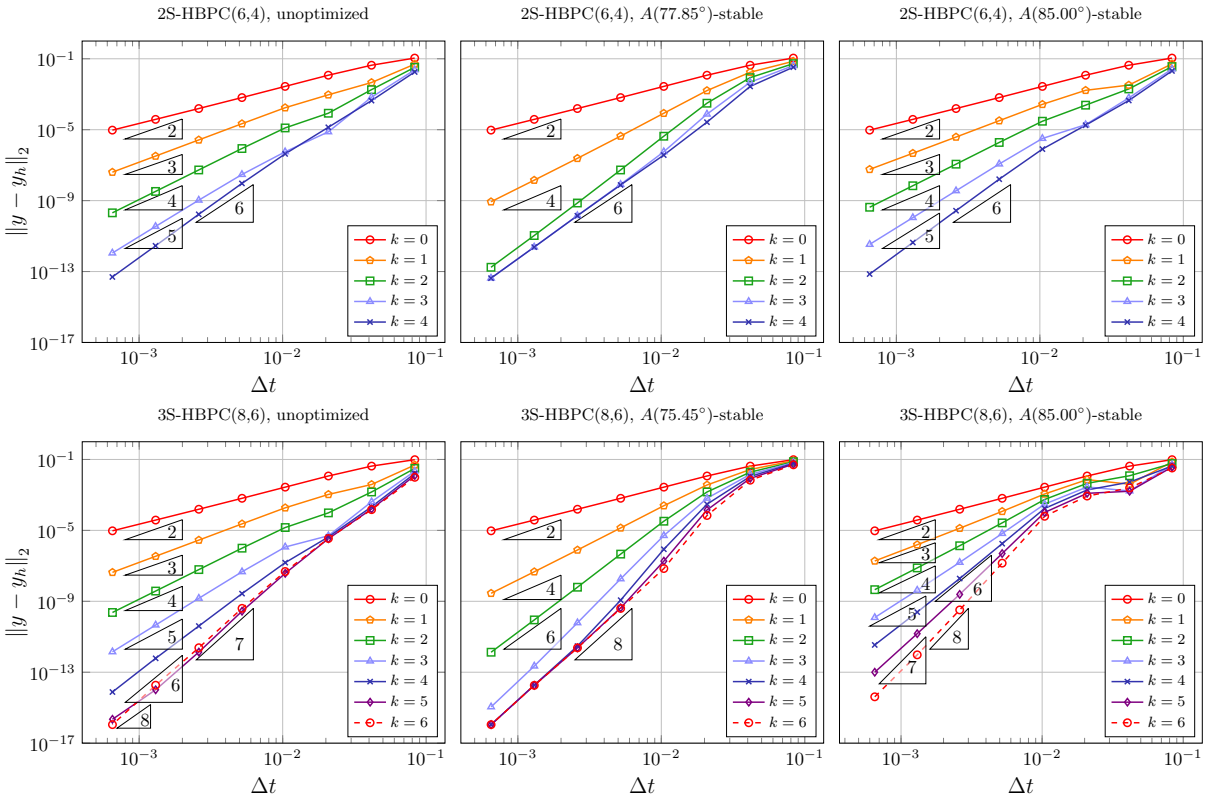


Figure 9: Convergence plots of the 2S-HBPC(6,4) (top) and 3S-HBPC(8,6) (bottom) schemes for a simple ODE (13). The error is calculated at $T_{end} = 0.25$. Consult Tab. 3 for (θ_1, θ_2) values corresponding to the given stability angles.

4.2. Pareschi–Russo Problem

Here, we investigate the performance of mS-HBPC(q,$k_{\max}$) schemes on stiff ODEs. The model problem considered is a system of IVP given by

$$y_1'(t) = -y_2, \quad y_2'(t) = y_1 + \frac{\sin(y_1) - y_2}{\varepsilon}, \quad y_0 = \left(\frac{\pi}{2}, 1\right), \tag{14}$$

which was introduced in [29]. The error is calculated at $T_{end} = 5$, using a highly accurate solution found numerically. To investigate the performance of the scheme across non-stiff and stiff equations, we vary the stiffness parameter ε across different values, specifically $\varepsilon = 1$, $\varepsilon = 10^{-2}$, and $\varepsilon = 10^{-3}$.

In Fig. 10, the convergence plots for 2S-HBPC(6,4) (top) and 3S-HBPC(8,6) (bottom) with different stability angles are shown for various stiffness parameters. It can be seen that the unoptimized scheme becomes unstable as the problem becomes more stiff. For the non-stiff problem ($\varepsilon = 1$), all the schemes exhibits their expected convergence order except the schemes $A(77.85°)$-stable 2S-HBPC(6,4)and $A(75.45°)$-stable 3S-HBPC(8,6) that show some irregularities. This is a consequence of the additional decrease in their error constants, which are specifically connected to their (θ_1, θ_2) values. When stiffness parameter $\varepsilon = 10^{-2}$, the convergence of $A(\alpha)$-stable schemes with low stability angles becomes irregular when relatively large timesteps are employed to solve the problem. As the timesteps decrease, they eventually attain their desired convergence order. For extremely small stiffness parameters ($\varepsilon = 10^{-3}$), the inconsistencies in convergence order are noticeable over a wider range of time steps, particularly in schemes with lower stability angles. For all specified stiffness parameters, it is notable that the $A(\alpha)$-stable scheme exhibits smoother convergence and achieves the intended convergence order as the stability angles increase. However, this improvement comes at the cost of slight shifts in the error.

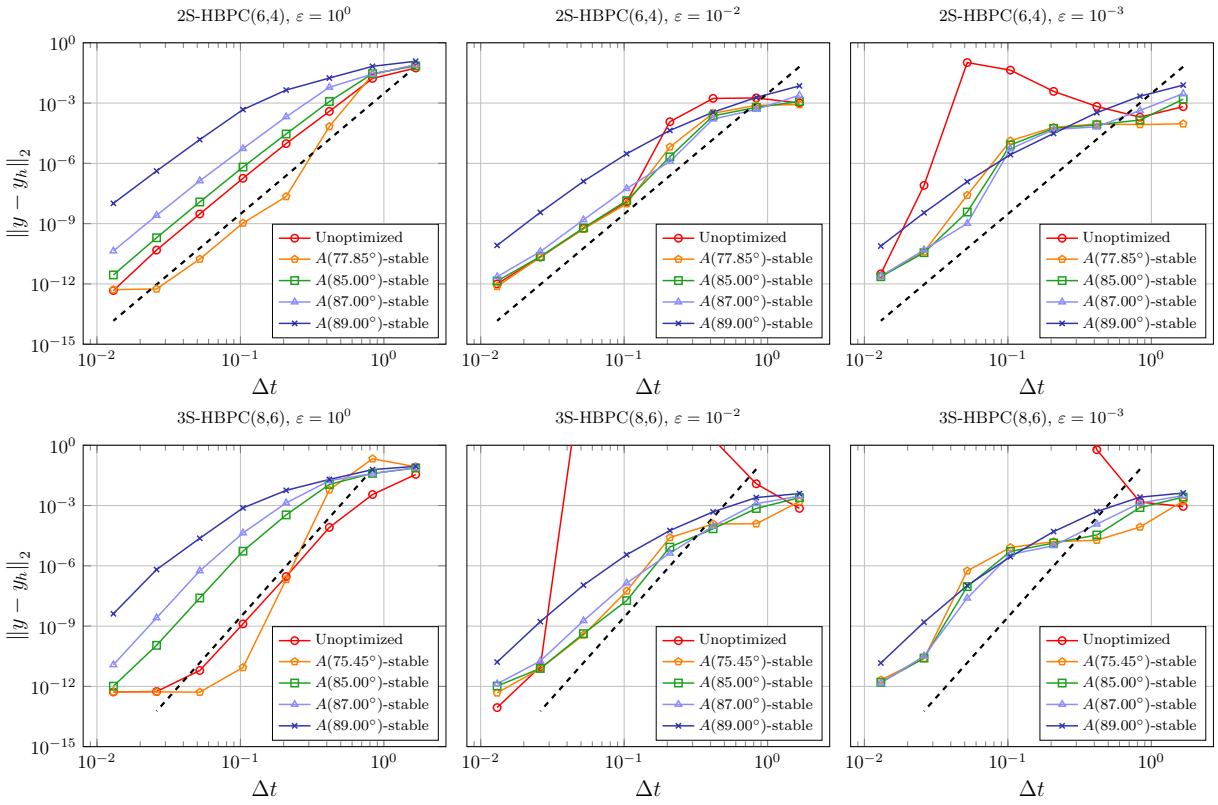


Figure 10: Error of Pareschi–Russo problem (14) with 2S-HBPC(6,4) (top) and 3S-HBPC(8,6) (bottom) schemes at $T_{end} = 5$. The dashed black thick line represents convergence orders six (top) and eight (bottom), respectively. Consult Tab. 3 for (θ_1, θ_2) values corresponding to the given stability angles.

4.3. Van-der-Pol Equation

To analyze the convergence behavior of mS-HBPC(q,$k_{\max}$) schemes on highly stiff problems, we consider Van-der-Pol equation. It is given by

$$y_1'(t) = y_2, \quad y_2'(t) = \frac{(1 - y_1^2)y_2 - y_1}{\varepsilon}, \quad y_0 = \left(2, -\frac{2}{3} + \frac{10}{81}\varepsilon\right). \tag{15}$$

The error is calculated at $T_{end} = 0.5$, using a refined solution found numerically. The stiffness parameters are varied across different values, from $\varepsilon = 10^{-1}$ to $\varepsilon = 10^{-5}$, for the numerical investigations.

In Fig. 11, the convergence plots for 2S-HBPC(6,4) (top) and 3S-HBPC(8,6) (bottom) with different stability angles are shown for various stiffness parameters. Similar to the Pareschi–Russo Problem, the unoptimized scheme becomes unstable as the problem becomes more stiff. For the equation (15) with a stiffness parameter $\varepsilon = 10^{-1}$, both sixth and eighth-order mS-HBPC(q,$k_{\max}$) schemes converge as expected, maintaining their designated order, although with some error shifts observed for $A(\alpha)$-stable schemes with larger stability angles. When $\varepsilon = 10^{-2}$, the sixth and eighth-order schemes demonstrate their expected convergence rates for lower stability angles. However, there is a slight reduction in order as the stability angle increases. In cases where the stiffness parameters are even smaller, such as $\varepsilon = 10^{-3}$, 10^{-4}, and 10^{-5}, the sixth and eighth-order $A(\alpha)$-stable schemes exhibit the desired convergence order for larger timesteps. However, as the timestep decreases, there's a noticeable reduction in order. Although the schemes with larger stability angles showcase significant error shifts in the plot for larger timesteps, the errors diminish in accordance with their respective convergence orders.

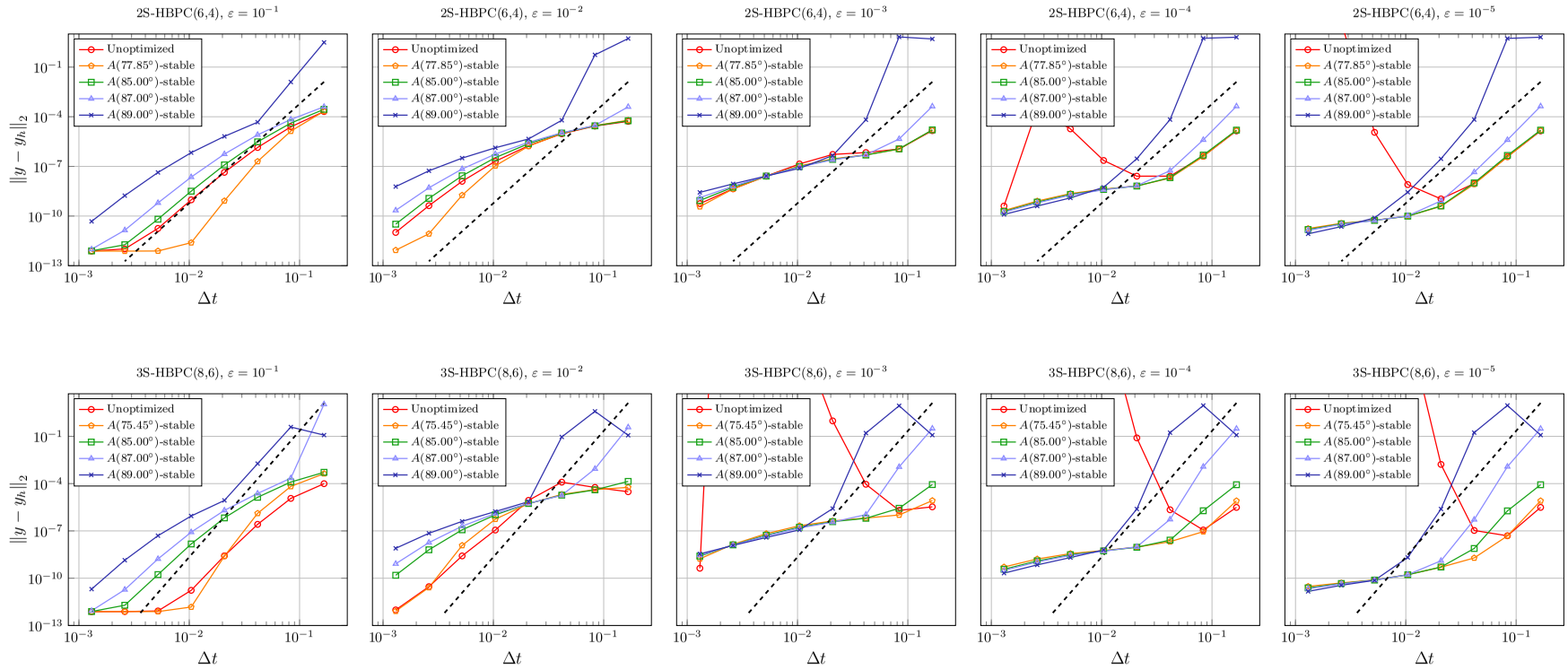


Figure 11: Error of Van-der-Pol equation (15) with 2S-HBPC(6,4) (top) and 3S-HBPC(8,6) (bottom) schemes at $T_{end} = 0.5$. The dashed black thick line represents convergence orders six (top) and eight (bottom), respectively. Consult Tab. 3 for (θ_1, θ_2) values corresponding to the given stability angles.

4.4. Viscous Burgers' equation

Here, we consider the viscous Burgers' equation to test the $A(\alpha)$-stable mS-HBPC(q,$k_{\max}$) schemes on partial differential equations. It is given by

$$\partial_t u + \partial_x \left(\frac{u^2}{2} \right) = \partial_{xx}^2 u, \quad u_0(x) = \frac{1 - \cos(2x)}{2}, \tag{16}$$

where $x \in [0, 2\pi]$ with periodic boundary conditions. The equation (16) has an explicit exact solution computed using Hopf-Cole transformation [30]. The spatial domain is discretized into a grid consisting of 200 elements with a eighth-order finite difference setting. The error is calculated at $T_{end} = 0.5$.

In Fig. 12, the convergence plots for $A(\alpha)$-stable 2S-HBPC(6,4) (left) and 3S-HBPC(8,6) (right) are shown for different stability angles. The sixth-order $A(\alpha)$-stable schemes exhibit desired order of convergence for all the given stability angles. The eight-order $A(\alpha)$-stable schemes achieves desired order of convergence for lower stability angles, whereas an order reduction is observed for larger stability angles. This phenomenon is more likely to result from larger values of θ_1 for 3S-HBPC(8,6) compared to 2S-HBPC(6,4), for the same stability angles.

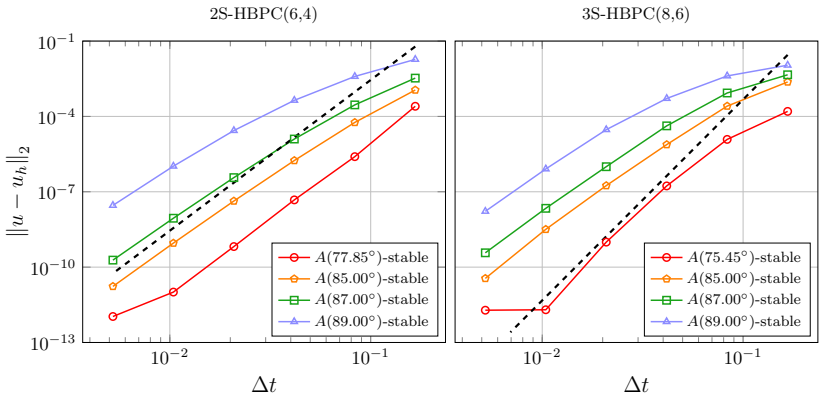


Figure 12: Convergence results of viscous Burgers' equation (16) with 2S-HBPC(6,4) (left) and 3S-HBPC(8,6) (right) schemes at $T_{end} = 0.5$. The dashed black thick line represents convergence orders six (left) and eight (right), respectively. Consult Tab. 3 for (θ_1, θ_2) values corresponding to the given stability angles.

5. Conclusion and Outlook

In this work, we have presented a class of multi-derivative predictor-corrector time-stepping schemes analogous to the schemes presented in [12], but with an underlying quadrature rule that is constructed using a multi-step fashion. However, the initial schemes are found not to be $A(\alpha)$-stable. Therefore, the schemes have been analyzed and optimized for better stability properties.

It has been found that the 2S-HBPC(6,4) and 3S-HBPC(8,6) schemes are $A(\alpha)$-stable for any values of θ_2 greater than or equal to 1.25868 and 3.84703, respectively. These threshold values have been fixed for θ_2 as they give the minimum one-step error for a fixed stability angle. Similarly, lower limit of $\theta_1 = 0.42083$ for the 2S-HBPC(6,4) scheme and $\theta_2 = 0.37957$ for the 3S-HBPC(8,6) scheme have been proposed by analyzing the error constant $\mathcal{C}(\theta_1, \theta_2)$. Therefore, schemes can be designed to have higher stability angles for an increased value of θ_1. In Tab. 3, a selection of θ_1 values corresponding to specific stability angles (α) have been presented.

Convergence of the 2S-HBPC(6,4) and 3S-HBPC(8,6) schemes have been shown numerically. For non-stiff ordinary differential equations, the $A(\alpha)$-stable schemes converge with their expected rates, particularly demonstrating minimal error at lower stability angles. The $A(\alpha)$-stable schemes gradually converge towards their expected order of convergence in stiff problems. In contrast, a reduction in the order of convergence has

been observed in very stiff problems. However, the unoptimized schemes were unstable for stiff problems. For higher-order non-linear partial differential equations, 2S-HBPC(6,4) scheme has exhibited the desired convergence order for all the given stability angles, whereas 3S-HBPC(8,6) scheme has exhibited desired order of convergence for lower stability angles.

As a future work, we are interested in combining the scheme with discontinuous Galerkin (DG) spatial discretization and studying the performance of the schemes over Navier-Stokes equations. An efficient scheme is paramount when the vast run time requirement for larger systems such as Navier-Stokes equations is concerned. Therefore, another possible direction for future investigation would be to utilize the parallelizability of the scheme across the correction steps and, hence, to implement a parallel-in-time mS-HBPC(q,$k_{\max}$) scheme.

Acknowledgments

Arjun Thenery Manikantan was funded by the "Bijzonder Onderzoeksfonds" (BOF) from UHasselt - project no. BOF21KP12. We acknowledge the VSC (Flemish Supercomputer Center) for providing computing resources. The VSC is funded by the Research Foundation - Flanders (FWO) and the Flemish Government.

References

[1] P. Turán, On the theory of the mechanical quadrature, Acta Universitatis Szegediensis Acta Scientiarum Mathematicarum 12 (1950) 30–37.
[2] A. H. Stroud, D. D. Stancu, Quadrature formulas with multiple Gaussian nodes, SIAM Journal on Numerical Analysis 2 (1965) 129–143.
[3] R. Chan, A. Tsai, On explicit two-derivative Runge-Kutta methods, Numerical Algorithms 53 (2010) 171–194.
[4] E. Hairer, G. Wanner, Multistep-multistage-multiderivative methods for ordinary differential equations, Computing (Arch. Elektron. Rechnen) 11 (1973) 287–303.
[5] J. Chouchoulis, J. Schütz, J. Zeifang, Jacobian-free explicit multiderivative Runge–Kutta methods for hyperbolic conservation laws, Journal of Scientific Computing 90 (2022) 96.
[6] S. Gottlieb, Z. J. Grant, J. Hu, R. Shu, High order strong stability preserving multiderivative implicit and IMEX Runge–Kutta methods with asymptotic preserving properties, SIAM Journal on Numerical Analysis 60 (2022) 423–449.
[7] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Review 43 (2001) 89–112.
[8] A. J. Christlieb, S. Gottlieb, Z. J. Grant, D. C. Seal, Explicit strong stability preserving multistage two-derivative time-stepping schemes, Journal of Scientific Computing 68 (2016) 914–942.
[9] Z. Grant, S. Gottlieb, D. C. Seal, A strong stability preserving analysis for explicit multistage two-derivative time-stepping schemes based on Taylor series conditions, Communications on Applied Mathematics and Computation 1 (2019) 21–59.
[10] A. Moradi, J. Farzi, A. Abdi, Strong stability preserving second derivative general linear methods, Journal of Scientific Computing 81 (2019) 392–435.
[11] J. Schütz, D. Seal, An asymptotic preserving semi-implicit multiderivative solver, Applied Numerical Mathematics 160 (2021) 84–101.
[12] J. Schütz, D. C. Seal, J. Zeifang, Parallel-in-time high-order multiderivative IMEX solvers, Journal of Scientific Computing 90 (2022) 1–33.
[13] J. Zeifang, J. Schütz, D. Seal, Stability of implicit multiderivative deferred correction methods, BIT Numerical Mathematics (2022).
[14] J. Zeifang, J. Schütz, Implicit two-derivative deferred correction time discretization for the discontinuous Galerkin method, Journal of Computational Physics 464 (2022) 111353.
[15] J. Zeifang, A. Thenery Manikantan, J. Schütz, Time parallelism and Newton-adaptivity of the two-derivative deferred correction discontinuous Galerkin method, Applied Mathematics and Computation 457 (2023) 128198.
[16] E. Theodosiou, J. Schütz, D. Seal, An explicitness-preserving imex-split multiderivative method, Computers & Mathematics with Applications 158 (2024) 139–149. URL: https://www.sciencedirect.com/science/article/pii/S089812212400021X. doi:https://doi.org/10.1016/j.camwa.2023.12.040.
[17] R. L. Brown, Multi-derivative numerical methods for the solution of stiff ordinary differential equations, Technical Report UIUCDCS-R-74-672, Department of Computer Science, University of Illinois, 1974.
[18] R. Jeltsch, A0-stability and stiff stability of Brown's multistep multiderivative methods, Numerische Mathematik 32 (1979) 167–181. URL: https://doi.org/10.1007/BF01404873. doi:10.1007/BF01404873.
[19] E. Hairer, G. Wanner, Solving ordinary differential equations II, Springer Series in Computational Mathematics, 1991.
[20] G. Califano, G. Izzo, Z. Jackiewicz, Strong stability preserving general linear methods with Runge–Kutta stability, Journal of Scientific Computing 76 (2018) 943–968. URL: https://doi.org/10.1007/s10915-018-0646-5. doi:10.1007/s10915-018-0646-5.

[21] M. M. Khalsaraei, A. Shokri, M. Molayi, The new class of multistep multiderivative hybrid methods for the numerical solution of chemical stiff systems of first order IVPs, Journal of Mathematical Chemistry 58 (2020) 1987–2012. URL: https://doi.org/10.1007/s10910-020-01160-z. doi:10.1007/s10910-020-01160-z.
[22] J. C. Butcher, General linear methods, Acta Numerica 15 (2006) 157–256.
[23] H. Zhang, A. Sandu, S. Blaise, Partitioned and implicit–explicit general linear methods for ordinary differential equations, Journal of Scientific Computing 61 (2014) 119–144.
[24] S. Boscarino, J. Qiu, G. Russo, Implicit-explicit integral deferred correction methods for stiff problems, SIAM Journal on Scientific Computing 40 (2018) A787–A816.
[25] A. Dutt, L. Greengard, V. Rokhlin, Spectral deferred correction methods for ordinary differential equations, BIT. Numerical Mathematics 40 (2000) 241–266.
[26] M. Minion, Semi-implicit spectral deferred correction methods for ordinary differential equations, Communications in Mathematical Sciences 1 (2003) 471–500.
[27] MathWorks, ode15s documentation, Online Documentation, 2022. URL: https://www.mathworks.com/help/matlab/ref/ode15s.html.
[28] MathWorks, Symbolic Math Toolbox, Natick, Massachusetts, United States, 2020. URL: https://www.mathworks.com/help/symbolic/, URL https://www.mathworks.com/help/symbolic/.
[29] L. Pareschi, G. Russo, Implicit-explicit Runge-Kutta schemes for stiff systems of differential equations, Recent Trends in Numerical Analysis 3 (2000) 269–289.
[30] E. Hopf, The partial differential equation $u_t + uu_x = \mu u_{xx}$, Communications on Pure and Applied Mathematics 3 (1950) 201–230.

UHasselt Computational Mathematics Preprint Series

www.uhasselt.be/cmat