Analysis for Implicit and Implicit-Explicit ADER and DeC Methods for Ordinary Differential Equations, Advection-Diffusion and Advection-Dispersion Equations

Philipp Öffner[*], Louis Petri[†], Davide Torlo[‡]

Abstract

In this manuscript, we present the development of implicit and implicit-explicit ADER and DeC methodologies within the DeC framework using the two-operators formulation, with a focus on their stability analysis both as solvers for ordinary differential equations (ODEs) and within the context of linear partial differential equations (PDEs). To analyze their stability, we reinterpret these methods as Runge-Kutta schemes and uncover significant variations in stability behavior, ranging from A-stable to bounded stability regions, depending on the chosen order, method, and quadrature nodes. This differentiation contrasts with their explicit counterparts. When applied to advection-diffusion and advection-dispersion equations employing finite difference spatial discretization, the von Neumann stability analysis demonstrates stability under CFL-like conditions. Particularly noteworthy is the stability maintenance observed for the advection-diffusion equation, even under spatial-independent constraints. Furthermore, we establish precise boundaries for relevant coefficients and provide suggestions regarding the suitability of specific schemes for different problem.

1 Introduction

Many systems of time-dependent differential equations can be separated into multiple parts that differ in their stiffness. For such systems, using implicit-explicit (IMEX) time-marching methods [36] is of paramount importance to guarantee stability and accuracy in many applications.

At the same time, high-order time-marching methods are sought for their efficiency and to match with the spatial discretization order in time-dependent partial differential equations (PDEs). Explicit high-order ADER and deferred correction (DeC) methods, due to their automatic construction, emerge as suitable alternatives to the traditional Runge-Kutta (RK) methods and have been extensively explored in various studies. The explicit DeC method, introduced by Dutt et al. [11] and then reinterpreted by Abgrall [1], is an explicit, arbitrarily high-order method for ODEs. Further extensions of DeC, including implicit, semi-implicit and modified Patankar versions, are available in the literature [7, 28, 32, 2, 19, 40]. The ADER method was originally developed for hyperbolic systems exploiting the Cauchy-Kovalevskaya theorem [49, 39, 47], then reinterpreted as a space-time discontinuous Galerkin (DG) method, which is solved through a fixed-point iteration procedure [9, 13, 10, 5, 25, 15, 16].

In this research, we present an detailed investigation of both implicit and IMEX versions of ADER and DeC, investigating their efficacy as solvers for ordinary differential equations (ODEs) and in the context of linear advection-diffusion or advection-dispersion PDEs. Building upon prior work [16, 28, 3, 10], which has explored IMEX descriptions of DeC and ADER, we expand our analysis to encompass the most used methods, employing varying quadrature points and levels of accuracy. Additionally, we explore their IMEX

[*] Institute of Mathematics, Johannes Gutenberg University, Mainz, and Institute of Numerical Analysis, TU Clausthal, Clausthal-Zellerfeld, Germany, mail@philippoeffner.de

[†] Institute of Mathematics, Johannes Gutenberg University, Mainz, Germany, lpetri01@uni-mainz.de

[‡] Dipartimento di Matematica Guido Castelnuovo, Università di Roma La Sapienza, Roma, Italy, davide.torlo@uniroma1.it

stability following the approach of previous studies [17, 21, 28], uncovering notable discrepancies among them, ranging from bounded stability regions to A-stable ones. This diverges significantly from the behavior observed in explicit versions [16].

Extending our investigation to the PDE case and inspired by [42], we conduct a von Neumann analysis for the presented IMEX time discretizations, paired with finite difference spatial discretizations of corresponding accuracy levels. We find that the stability regions are bounded by CFL-type conditions as well as simple conditions on Δt for the advection–diffusion case.

The analysis of advection-dispersion presents less definitive outcomes, with only a few cases indicating conditions solely influenced by spatial discretization. However, these findings align with the stability regions observed in the ODE case.

The structure of the paper is as follows. In Sections 2 and 3, we introduce the implicit and IMEX DeC and ADER methods, respectively, and incorporate them into the RK framework. Additionally, we present theoretical stability results for the pure implicit ADER method. In Section 4, we establish the high order accuracy of both the implicit and IMEX ADER and DeC methods. Following this, in Section 5, we delineate their stability regions. Next, in Sections 6 and 7 we extend the stability analysis to the PDE scenario by applying our IMEX methods to advection-diffusion and advection-dispersion equations. In Section 8, we finally present several numerical examples aimed at validating the stability and convergence analyses, while in Section 9 we summarize the conclusions drawn from our deep analysis.

2 Deferred Correction

In this section, we present the DeC in its explicit, implicit, and IMEX versions using the notation introduced in [16]. Consider the system of ODEs

$$\boldsymbol{\alpha}'(t) - F(\boldsymbol{\alpha}) = 0, \tag{1}$$

with $\boldsymbol{\alpha} : [0,T] \to \mathbb{R}^I$. Given a time interval $[t_n, t_{n+1}]$ with length Δt, we subdivide it into M subintervals $\{[t_m^{m-1}, t_n^m]\}_{m=1}^M$, where $t_n^0 = t_n$ and $t_n^M = t_{n+1}$. DeC methods are one step methods, hence we want to obtain $\boldsymbol{\alpha}_{n+1} \approx \boldsymbol{\alpha}(t_{n+1})$ from $\boldsymbol{\alpha}_n \approx \boldsymbol{\alpha}(t_n)$. We mimic for every subinterval $[t_n^0, t_n^m]$ the Picard–Lindelöf theorem. We drop the dependency on the timestep n for subtimesteps t_n^m and associated substates $\boldsymbol{\alpha}_n^m$, such that

$$t_n = t_n^0 = t^0,\ t_n^1 = t^1,\ t_n^2 = t^2, \dots,\ t_n^M = t^M = t_{n+1}.$$

Then, we introduce the $\mathcal{L}^2$ operator, which represents an implicit high order discretization of ODE obtained integrating (1) in each subinterval $[t^0, t^m]$ and applying a quadrature formula,

$$\mathcal{L}^2(\boldsymbol{\alpha}^0, \dots, \boldsymbol{\alpha}^M) := \begin{cases} \boldsymbol{\alpha}^M - \boldsymbol{\alpha}_n - \Delta t \sum_{r=0}^M \theta_r^M F(\boldsymbol{\alpha}^r) \\ \vdots \\ \boldsymbol{\alpha}^0 - \boldsymbol{\alpha}_n - \Delta t \sum_{r=0}^M \theta_r^0 F(\boldsymbol{\alpha}^r) \end{cases}. \tag{2}$$

We obtain this operator by applying to $\big(F(\boldsymbol{\alpha}^0), \dots, F(\boldsymbol{\alpha}^M)\big)$ an interpolation polynomial of degree M. The term $\theta_r^m := \frac{1}{\Delta t}\int_{t_n}^{t_n^m} \phi_r(s) ds$ denotes the weights, which can be obtained with Lagrange polynomials $\{\phi_r\}_{r=0}^M$ in the subnodes $\{t^m\}_{m=0}^M$, where $\phi_r(t^m) = \delta_{r,m}$ and $\sum_{r=0}^M \phi_r(s) \equiv 1$ for any $s \in [0,1]$. We notice that $\mathcal{L}^2 = 0$ with $\boldsymbol{\alpha}_{n+1} = \sum_{r=0}^M \phi_r(t_{n+1})\boldsymbol{\alpha}^r$ is a collocation method and, when using Gauss–Lobatto nodes, it coincides with Lobatto IIIA schemes [15].

2.1 Explicit DeC

To obtain an explicit versions of the DeC, we introduce $\mathcal{L}^1$, an explicit first order approximation of the $\mathcal{L}^2$ operator following [1, 29]:

$$\mathcal{L}^1(\boldsymbol{\alpha}^0,\dots,\boldsymbol{\alpha}^M) := \begin{cases} \boldsymbol{\alpha}^M - \boldsymbol{\alpha}_n - \Delta t \beta^M F(\boldsymbol{\alpha}_n) \\ \vdots \\ \boldsymbol{\alpha}^0 - \boldsymbol{\alpha}_n - \Delta t \beta^0 F(\boldsymbol{\alpha}_n) \end{cases}, \tag{3}$$

with coefficients $\beta^m := \frac{t_n^m - t_n}{\Delta t}$. To simplify the notation, we introduce the vector of states for the variable $\boldsymbol{\alpha}$ at all subtimesteps

$$\underline{\boldsymbol{\alpha}} := (\boldsymbol{\alpha}^0,\dots,\boldsymbol{\alpha}^M) \in \mathbb{R}^{M\times I}, \text{ such that} \tag{4}$$

$$\mathcal{L}^1(\underline{\boldsymbol{\alpha}}) := \mathcal{L}^1(\boldsymbol{\alpha}^0,\dots,\boldsymbol{\alpha}^M) \text{ and } \mathcal{L}^2(\underline{\boldsymbol{\alpha}}) := \mathcal{L}^2(\boldsymbol{\alpha}^0,\dots,\boldsymbol{\alpha}^M). \tag{5}$$

Now, the DeC algorithm uses a combination of the $\mathcal{L}^1$ and $\mathcal{L}^2$ operators to recursively approximate $\underline{\boldsymbol{\alpha}}^*$, the high order accurate numerical solution of the $\mathcal{L}^2(\underline{\boldsymbol{\alpha}}^*) = 0$ scheme. We denote by Y the order of accuracy of the solution $\underline{\boldsymbol{\alpha}}^*$. This is contingent upon the nodes selected [24], where we have $Y = M+1$ for equispaced nodes and $Y = 2M$ for Gauss-Lobatto nodes. In detailing the process, for each variable, we need to reference both the m-th subnode and the k-th iteration of the DeC algorithm, denoted by $\boldsymbol{\alpha}^{m,(k)} \in \mathbb{R}^I$. Finally, the DeC method can be written as

DeC Algorithm

$$\boldsymbol{\alpha}^{m,(0)} := \boldsymbol{\alpha}_n, \quad m = 1,\dots,M, \tag{6}$$

$$\mathcal{L}^1(\underline{\boldsymbol{\alpha}}^{(k)}) = \mathcal{L}^1(\underline{\boldsymbol{\alpha}}^{(k-1)}) - \mathcal{L}^2(\underline{\boldsymbol{\alpha}}^{(k-1)}), \quad \text{for } k = 1,\dots,K.$$

The DeC method attains an order of accuracy $\min(K,Y)$, enhancing its accuracy by one with each iteration [1]. Therefore, selecting $K = Y$ is optimal. It's important to note that only the explicit operator $\mathcal{L}^1$ needs to be solved, while $\mathcal{L}^2$ is solely evaluated using the previously computed predictions $\underline{\boldsymbol{\alpha}}^{(k-1)}$.

Remark 2.1 (Variations of DeC)**.** *The presented DeC algorithm is just one of a whole family of DeC methods. For example, instead of considering integrations on* $[t^0, t^m]$ *in the* m*'th equation, we could switch to the smaller intervals* $[t^{m-1}, t^m]$ *like in [28, 11, 24], changing the* m*'th line of the operators to*

$$\mathcal{L}^{1,m}(\boldsymbol{\alpha}^0,\dots,\boldsymbol{\alpha}^M) = \boldsymbol{\alpha}^m - \boldsymbol{\alpha}^{m-1} - \Delta t \gamma^m F(\boldsymbol{\alpha}^{m-1}),$$

$$\mathcal{L}^{2,m}(\boldsymbol{\alpha}^0,\dots,\boldsymbol{\alpha}^M) = \boldsymbol{\alpha}^m - \boldsymbol{\alpha}^{m-1} - \Delta t \sum_{r=0}^{M} \delta_r^m F(\boldsymbol{\alpha}^r),$$

with $\gamma^m := \frac{t_n^m - t_n^{m-1}}{\Delta t}$ *and* $\delta_r^m = \frac{1}{\Delta t}\int_{t_n^{m-1}}^{t_n^m} \phi_r(s)ds$*. The iteration process (6) with these* $\mathcal{L}^1$ *and* $\mathcal{L}^2$ *operators does not change. Due to the smaller steps, this method is also referred to as the sDeC algorithm. One downside of the sDeC algorithm is the necessity of* $\boldsymbol{\alpha}^{m-1,(k)}$ *to calculate* $\boldsymbol{\alpha}^{m,(k)}$*, which does not allow to use parallel computation on the different subnodes, which is possible for the DeC with larger subintervals presented above.*

In addition, we note that instead of using explicit Euler steps inside the $\mathcal{L}^1$ *operators, other explicit RK methods can be applied inside the* $\mathcal{L}^1$ *operator [44, 7]. Here, additional problems may rise. For a detailed overview of the different variations of DeC, we refer to the nice overview article [33] and the references therein.*

2.2 Implicit and IMEX DeC

In this section, we will construct the implicit and IMEX DeC methods using the presented framework. Consider the ODE (1), where $F(\boldsymbol{\alpha})$ is a stiff term. We proceed by taking an implicit version of the $\mathcal{L}^1$

operator, which corresponds to implicit Euler steps at each subinterval, for brevity, we will just describe the m'th equation for $m = 0, \dots, M$

$$\mathcal{L}^{1,m}(\boldsymbol{\alpha}^0, \dots, \boldsymbol{\alpha}^M) := \boldsymbol{\alpha}^m - \boldsymbol{\alpha}^0 - \beta^m \Delta t F(\boldsymbol{\alpha}^m) \tag{7}$$

and assembling the implicit DeC method as in (6).

If we have a problem, whose right-hand side can be separated into the sum of a stiff term $S(\boldsymbol{\alpha})$ and a non-stiff term $G(\boldsymbol{\alpha})$, i.e.

$$\partial_t \boldsymbol{\alpha} = S(\boldsymbol{\alpha}) + G(\boldsymbol{\alpha}), \tag{8}$$

we create an implicit-explicit DeC (IMEX DeC) method, by adding up the implicit and explicit treatments of the $\mathcal{L}^1$ operator, obtaining for $m = 0, \dots, M$

$$\mathcal{L}^{1,m}(\boldsymbol{\alpha}^0, \dots, \boldsymbol{\alpha}^M) := \boldsymbol{\alpha}^m - \boldsymbol{\alpha}^0 - \beta^m \Delta t S(\boldsymbol{\alpha}^m) - \beta^m \Delta t G(\boldsymbol{\alpha}^0), \tag{9}$$

which we can substitute into the known correction procedure (6). In case of nonlinear stiff terms S, a further linearization of S could be introduced in the definition of $\mathcal{L}^1$ [16].

2.3 Implicit and IMEX DeC as RK

In this section, we rewrite the DeC methods presented above into RK methods to study their stability, as done in [24]. The DeC has the advantage that one does not need to specify the coefficients for every order of accuracy as usually necessary in classical RK methods, see for example [3, Remark 4.3]. This is done automatically, through the quadrature weights Θ and β, which fully determine the coefficients of the corresponding Butcher Tableau. Let us consider our version of DeC and rewrite it in a Runge–Kutta method with Z stages defined by its Butcher tableau

$$\begin{cases} \boldsymbol{u}^{(s)} = \boldsymbol{\alpha}^n + \Delta t \sum_{i=1}^{Z} A_i^s G(\boldsymbol{u}^{(i)}), \quad s = 1, \dots, Z, \\ \boldsymbol{\alpha}^{n+1} = \boldsymbol{\alpha}^n + \Delta t \sum_{i=1}^{Z} b_i G(\boldsymbol{u}^{(i)}), \end{cases} \qquad \begin{array}{c|c} c & \underline{\underline{A}} \\ \hline & b \end{array}. \tag{10}$$

The process of rewriting DeC as Runge-Kutta schemes is elaborated in detail in [24, 2]. Here, we emphasize the final form for comparison with the implicit and IMEX versions[1]. We introduce $\underline{\beta} = \{\beta^i\}_{i=0}^M$ as the vector of the $\mathcal{L}^1$ operator coefficients, and we define the matrix $\underline{\underline{\theta}} = \{\theta_r^m\}_{m=0,\dots,M; r=0,\dots,M}$. Finally, we introduce the vector $\underline{\theta}^M = (\theta_0^M, \dots, \theta_M^M)$ for the final update. The Butcher tableau for an arbitrary DeC approach is given by

$$\begin{array}{c|ccccccc}
0 & & & & & & & \\
\underline{\beta} & \underline{\beta} & & & & & & \\
\underline{\beta} & \underline{0} & \underline{\underline{\theta}} & & & & & \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{\theta}} & & & & \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{\theta}} & & & \\
\vdots & \vdots & \vdots & \vdots & \ddots & \ddots & & \\
\underline{\beta} & \underline{0} & \underline{\underline{0}} & \dots & \dots & \underline{\underline{0}} & \underline{\underline{\theta}} & \\
\hline
 & 0 & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{\theta}^M
\end{array}. \tag{11}$$

Let us notice that the first iteration is derived by a simplification that can be obtained noticing that $\sum_{r=0}^M \theta_r^m = \beta^m$, thanks to the properties

$$\sum_{r=0}^{M} \phi_r(s) = 1, \quad \boldsymbol{\alpha}^{m,(0)} = \boldsymbol{\alpha}^0, \quad m = 0, \dots, M. \tag{12}$$

Moreover, since we have chosen $t_n^0 = t_n$, there are several trivial stages where the corresponding lines of A is composed only of 0s. These lines can be simplified as in [16, 24].

[1] Note that the re-interpretation of DeC as a RK scheme is not new and it was applied in various contexts, e.g. [4].

Remark 2.2 (Number of stages)**.** *To obtain order p, we require $Y \overset{!}{=} K \overset{!}{=} p$, which means $M = p - 1$ subtimesteps for equispaced nodes and $M = \lfloor \frac{p}{2} \rfloor$ for Gauss–Lobatto nodes, and $K = p$ corrections. In total, considering all nontrivial stages and excluding the final iteration where only the last sub-time node is pertinent, we arrive at $Z = M(K-1) + 1$ stages, which equates to $Z = (p-1)^2 + 1$ stages for equispaced nodes and $Z = \lfloor \frac{p}{2} \rfloor (p-1) + 1$ for Gauss-Lobatto nodes [15].*

For the implicit case, the DeC iteration process for every $m = 0, \dots, M$ and every iteration $k = 1, \dots, K$ reads

$$\begin{aligned} &\mathcal{L}^{1,m}(\underline{\boldsymbol{\alpha}}^{(k)}) = \mathcal{L}^{1,m}(\underline{\boldsymbol{\alpha}}^{(k-1)}) - \mathcal{L}^{2,m}(\underline{\boldsymbol{\alpha}}^{(k-1)}) \\ \Leftrightarrow \quad &\boldsymbol{\alpha}^{m,(k)} = \boldsymbol{\alpha}_n + \Delta t \left[\left(\sum_{r=0,\, r\neq m}^{M} \theta_r^m F(\boldsymbol{\alpha}^{r,(k-1)}) \right) + (\theta_m^m - \beta^m) F(\boldsymbol{\alpha}^{m,(k-1)}) + \beta^m F(\boldsymbol{\alpha}^{m,(k)}) \right]. \end{aligned} \tag{13}$$

From (13), we derive the blocks of the RK matrix $\underline{\underline{A}}$, keeping in mind that for the first iteration $k = 1$ many terms simplify due to (12). We obtain

$$\boldsymbol{\alpha}^{m,(1)} = \boldsymbol{\alpha}^0 + \Delta t \beta^m F(\boldsymbol{\alpha}^{m,(1)}).$$

The final update is given by $\boldsymbol{\alpha}^{n+1} := \boldsymbol{\alpha}^{M,(K)}$. This leads to the following RK Butcher tableau

$$\begin{array}{c|cccccccc}
0 & 0 & & & & & & & \\
\underline{\beta} & \underline{0} & \underline{\underline{B}} & & & & & & \\
\underline{\beta} & \underline{0} & \underline{\underline{\theta}} - \underline{\underline{B}} & \underline{\underline{B}} & & & & & \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{\theta}} - \underline{\underline{B}} & \underline{\underline{B}} & & & & \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{\theta}} - \underline{\underline{B}} & \underline{\underline{B}} & & & \\
\vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & & \\
\underline{\beta} & \underline{0} & \underline{\underline{0}} & \dots & \dots & \underline{\underline{0}} & \underline{\underline{\theta}} - \underline{\underline{B}} & \underline{\underline{B}} & \\
1 & 0 & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{\theta}^M - \underline{B}^M & \beta^M \\
\hline
 & 0 & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{\theta}^M - \underline{B}^M & \beta^M
\end{array}, \tag{14}$$

with $B_{mr} = \delta_{mr}\beta^m$ for $m, r = 0, \dots, M$ and δ_{mr} the Kronecker delta. We notice that the ImDeC method is diagonally implicit and that the last stage coincide with the final update which makes the method stiffly accurate [53, Proposition 3.8].

Next, we describe the IMEX DeC in the notation of an IMEX RK scheme, i.e.,

$$\begin{cases} \boldsymbol{u}^{(s)} = \boldsymbol{\alpha}^n + \Delta t \sum_{i=1}^{s} A_i^s S(\boldsymbol{u}^{(i)}) + \Delta t \sum_{i=1}^{s-1} \hat{A}_i^s G(\boldsymbol{u}^{(i)}), \quad s = 1, \dots, Z, \\ \boldsymbol{\alpha}^{n+1} = \boldsymbol{\alpha}^n + \Delta t \sum_{i=1}^{Z} b_i S(\boldsymbol{u}^{(i)}) + \Delta t \sum_{i=1}^{Z} \hat{b}_i G(\boldsymbol{u}^{(i)}), \end{cases} \tag{15}$$

where the matrices $\underline{\underline{A}}, \underline{\underline{\hat{A}}}$ and vectors $b, \hat{b}$ are provided in two Butcher tableaux

$$\begin{array}{c|c} c & \underline{\underline{A}} \\ \hline & b \end{array}, \qquad \begin{array}{c|c} c & \underline{\underline{\hat{A}}} \\ \hline & \hat{b} \end{array} \tag{16}$$

and directly correspond to the simpler implicit and explicit cases presented, with an extra stage in the explicit case to match the implicit description. Therefore, we obtain the following Butcher tableaux

$$\begin{array}{c|cccccccc}
0 & 0 \\
\underline{\beta} & \underline{0} & \underline{\underline{B}} \\
\underline{\beta} & \underline{0} & \underline{\underline{\theta}} - \underline{\underline{B}} & \underline{\underline{B}} \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{\theta}} - \underline{\underline{B}} & \underline{\underline{B}} \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{\theta}} - \underline{\underline{B}} & \underline{\underline{B}} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \\
\underline{\beta} & \underline{0} & \underline{\underline{0}} & \dots & \dots & \underline{\underline{0}} & \underline{\underline{\theta}} - \underline{\underline{B}} & \underline{\underline{B}} \\
1 & 0 & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{\theta}^M - \underline{B}^M & \beta^M \\
\hline
 & 0 & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{\theta}^M - \underline{B}^M & \beta^M
\end{array}, \qquad
\begin{array}{c|cccccccc}
0 & 0 \\
\underline{\beta} & \underline{\beta} \\
\underline{\beta} & \underline{0} & \underline{\underline{\theta}} \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{\theta}} \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{\theta}} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \ddots \\
\underline{\beta} & \underline{0} & \underline{\underline{0}} & \dots & \dots & \underline{\underline{0}} & \underline{\underline{\theta}} \\
1 & 0 & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{\theta}_r^M & 0 \\
\hline
 & 0 & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{\theta}_r^M & 0
\end{array}. \tag{17}$$

Remark 2.3 (Runge-Kutta for sDeC)**.** *Using the same properties and techniques, we can construct Butcher tableaux respectively for the sDeC, ImsDeC and IMEX sDeC. Their construction in the explicit case is performed in [24] and we proceed similarly for the implicit and IMEX cases.*

Also in the implicit and IMEX cases, several trivial stages can be simplified in the Butcher tableaux [24].

3 ADER

In this section, we present the modern ADER introduced as space-time DG solver for hyperbolic PDEs in [9] and adapted for ODEs in [16, 15]. Starting with an I-dimensional system of ODEs (1), we consider the interval $T^n := [t^n, t^{n+1}]$ where we represent $\boldsymbol{\alpha}(t)$ as a linear combination of $(M+1)$ basis functions

$$\boldsymbol{\alpha}(t) = \sum_{m=0}^{M} \phi_m(t)\boldsymbol{\alpha}^m = \underline{\phi}(t)^T \underline{\boldsymbol{\alpha}}, \tag{18}$$

where $\underline{\phi} = [\phi_0, \dots, \phi_M]^T : T^n \to \mathbb{R}^{M+1}$ is the vector of Lagrangian basis functions in some given nodes $\{t_m\}_{m=0}^M \subset T^n$, e.g. equispaced or Gauss-Lobatto nodes, and $\underline{\boldsymbol{\alpha}} = [\boldsymbol{\alpha}^0, \dots, \boldsymbol{\alpha}^M]^T \in \mathbb{R}^{(M+1)\times I}$ is the vector of coefficients of the basis functions respectively.
As described in [16], ADER is derived multiplying (1) with a smooth test function and integrating over T^n to obtain the weak formulation. We insert the reconstruction (18) in the weak formulation, we use as test functions the basis functions, we apply integration by parts in time and using the quadrature $\{(w_q, x_q)\}_{q=0}^Q$ in $[t_n, t_{n+1}]$ results in a system

$$\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{\alpha}} = \underline{r}(\underline{\boldsymbol{\alpha}}) \iff \mathcal{L}^2(\underline{\boldsymbol{\alpha}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{\alpha}} - \underline{r}(\underline{\boldsymbol{\alpha}}) \overset{!}{=} 0. \tag{19}$$

Thereby, the mass-matrix $\underline{\underline{\mathrm{M}}} \in \mathbb{R}^{(M+1)\times(M+1)}$ and the (nonlinear) right-hand side functional $\underline{r}(\underline{\boldsymbol{\alpha}}) : \mathbb{R}^{(M+1)\times I} \to \mathbb{R}^{(M+1)\times I}$ are given by

$$\underline{\underline{\mathrm{M}}}_{m,l} := \phi_m(t_{n+1})\phi_l(t_{n+1}) - \sum_{q=0}^{Q} \partial_t \phi_m(x_q)\phi_l(x_q) w_q \tag{20}$$

$$\underline{r}(\underline{\boldsymbol{\alpha}})_m := \phi_m(t_n)\boldsymbol{\alpha}_n + \Delta t \underbrace{\sum_{q=0}^{Q} w_q \phi_m(x_q) \underline{\phi}(x_q)^T}_{=:\underline{\underline{R}}_z^m} F(\underline{\boldsymbol{\alpha}}), \quad m = 0, \dots, M+1. \tag{21}$$

The right hand side can also be written in matricial form as $\underline{r}(\underline{\boldsymbol{\alpha}}) = \underline{\phi}(t^n)\boldsymbol{\alpha}(t^n) + \Delta t \underline{\underline{R}}\, F(\underline{\boldsymbol{\alpha}})$.

3.1 Explicit ADER

To obtain an explicit approximation of the system (19) for the unknown $\underline{\boldsymbol{\alpha}}$, ADER resorts to a fixed-point problem, whose solution will give us a high order Y accurate solution in t. In particular, the order of accuracy will be $Y = M+1$ for equispaced nodes and quadrature, order $Y = 2M$ for Gauss–Lobatto nodes and quadrature and, order $Y = 2M+1$ for Gauss–Legendre nodes and quadrature [15]. We will use the following iterative procedure and then we reconstruct $\boldsymbol{\alpha}_{n+1} \approx \boldsymbol{\alpha}(t^{n+1})$ as

ADER Algorithm

$$\begin{aligned}
&\underline{\boldsymbol{\alpha}}^{(0)} := [\boldsymbol{\alpha}_n, \dots, \boldsymbol{\alpha}_n]^T,\\
&\underline{\boldsymbol{\alpha}}^{(k)} = \underline{\underline{\mathrm{M}}}^{-1}\underline{r}(\underline{\boldsymbol{\alpha}}^{(k-1)}), \quad k = 1, \dots, K,\\
&\boldsymbol{\alpha}_{n+1} = \boldsymbol{\alpha}_n + \Delta t \sum_{m=0}^{M} \int_{t_n}^{t_{n+1}} \phi_m(t) F(\boldsymbol{\alpha}^{(K),m}) dt = \boldsymbol{\alpha}_n + \Delta t \underline{b}^T F(\underline{\boldsymbol{\alpha}}^{(K)}),
\end{aligned} \tag{22}$$

with $b_i := \int_{t^n}^{t^{n+1}} \phi_i(t) dt$.

Remark 3.1 (ADER as DeC and order of accuracy of (19))**.** *As shown in [16, 15], ADER can be written into the DeC formalism by defining*

$$\mathcal{L}^1(\underline{\boldsymbol{\alpha}}) := \underline{\underline{M}}\,\underline{\boldsymbol{\alpha}} - \underline{r}(\underline{1}\boldsymbol{\alpha}_n), \tag{23}$$

so that the DeC iterations (6) coincide with the fixed point iterations of (22), i.e.,

$$\underline{\underline{M}}\,\underline{\boldsymbol{\alpha}}^{(k)} - \underline{r}(\underline{\boldsymbol{\alpha}}^{(k-1)}) = 0. \tag{24}$$

If we set the starting values as $\boldsymbol{\alpha}^{(0),m} = \boldsymbol{\alpha}(t^n)$ for every m, like it is done in DeC, we obtain exactly the fixed-point iteration (22) and therefore an equivalent definition of the ADER method. This also tells us that the order of accuracy of (22) with respect to the solution of $\mathcal{L}^2(\underline{\boldsymbol{\alpha}}^*) = 0$ is K.

3.2 Implicit and IMEX ADER

We start describing the implicit version of ADER (ImADER), considering (1) with $F(\boldsymbol{\alpha})$ stiff, by modifying the iterative process to

$$\underline{\boldsymbol{\alpha}}^{(k)} = \underline{\underline{\mathrm{M}}}^{-1}\underline{r}(\underline{\boldsymbol{\alpha}}^{(k)}) \tag{25}$$

or, as explained in Remark 3.1, with the ADER-DeC notation:

$$\mathcal{L}^1(\underline{\boldsymbol{\alpha}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{\alpha}} - \underline{\phi}(0)\boldsymbol{\alpha}_n - \Delta t \underline{\underline{R}}\, F(\underline{\boldsymbol{\alpha}}). \tag{26}$$

We soon realize that performing multiple corrections does not give us any advantages as (25) does not depend on the previous iteration. Hence, the construction of ImADER does not seem purposeful, but it will become useful for the IMEX case, as presented in [10] for PDE with stiff source terms or in [16] for ODEs.
We consider the separated ODE system (8). To construct the IMEX ADER $\mathcal{L}^1$ operator, we combine the implicit (26) and explicit (23) treatments of the ADER iteration for the stiff and non-stiff terms, and get

$$\mathcal{L}^1(\underline{\boldsymbol{\alpha}}) := \underline{\underline{\mathrm{M}}}\underline{\boldsymbol{\alpha}} - \underline{\phi}(0)\boldsymbol{\alpha}_n - \Delta t \underline{\underline{R}}\, S(\underline{\boldsymbol{\alpha}}) - \Delta t \underline{\underline{R}}\, G(\underline{1}\boldsymbol{\alpha}_n). \tag{27}$$

This leads to the iterative process (22) with the iteration given by

$$\begin{aligned}
&\underline{\underline{\mathrm{M}}}\underline{\boldsymbol{\alpha}}^{(k)} - \underline{\phi}(0)\boldsymbol{\alpha}_n - \Delta t \underline{\underline{R}}\, S(\underline{\boldsymbol{\alpha}}^{(k)}) - \Delta t \underline{\underline{R}}\, G(\underline{\boldsymbol{\alpha}}^{(k-1)}) = 0\\
\iff \quad &\underline{\boldsymbol{\alpha}}^{(k)} = \underline{1}\boldsymbol{\alpha}_n - \Delta t \underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\, S(\underline{\boldsymbol{\alpha}}^{(k)}) - \Delta t \underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\, G(\underline{\boldsymbol{\alpha}}^{(k-1)}),
\end{aligned} \tag{28}$$

which is in fact just an additive combination of the implicit and explicit parts. We apply the following proposition demonstrated in [15, Proposition 2.4] to write the IMEX ADER algorithms.

Proposition 1 (ADER right-hand side)**.** *Given the definition of* $\underline{\underline{M}}$ *in* (20) *and defining with* $\underline{1} = [1, \dots, 1]^T \in \mathbb{R}^{M+1}$*, we have that*

$$\underline{\underline{\mathrm{M}}}^{-1} \underline{\phi}(t_n) = \underline{1}. \tag{29}$$

If we take this under consideration in the whole ADER process, it leads to the

IMEX ADER Algorithm

$$\begin{aligned}
&\underline{\boldsymbol{\alpha}}^{(0)} := [\boldsymbol{\alpha}_n, \dots, \boldsymbol{\alpha}_n]^T,\\
&\underline{\boldsymbol{\alpha}}^{(k)} = \underline{1}\boldsymbol{\alpha}_n - \Delta t \underline{\underline{\mathrm{M}}}^{-1} \underline{\underline{R}}\, S(\underline{\boldsymbol{\alpha}}^{(k)}) - \Delta t \underline{\underline{\mathrm{M}}}^{-1} \underline{\underline{R}}\, G(\underline{\boldsymbol{\alpha}}^{(k-1)}), \quad k = 1, \dots, K, \qquad (30)\\
&\boldsymbol{\alpha}_{n+1} = \boldsymbol{\alpha}_n + \Delta t \sum_{m=0}^{M} \int_{t_n}^{t_{n+1}} \left(S(\boldsymbol{\alpha}^{(K),m}) + G(\boldsymbol{\alpha}^{(K),m}) \right) dt = \boldsymbol{\alpha}_n + \Delta t \underline{b}^T \left(S(\underline{\boldsymbol{\alpha}}^{(K)}) + G(\underline{\boldsymbol{\alpha}}^{(K)}) \right).
\end{aligned}$$

We remark that the iteration process contains a (nonlinear) system of equations in $\underline{\boldsymbol{\alpha}}^{(k)}$ of dimension $(M+1) \times I$ for every (k). If the stiff term is linear, the system becomes linear, otherwise, it is possible to linearize the stiff term in the definition of $\mathcal{L}^1$ [16].

3.3 IMEX ADER as RK

In order to put the ADER method into a RK form, in (22) we have to multiply the right-hand side by the inverse of the mass matrix. This will show the explicit dependence on $\boldsymbol{\alpha}_n$ for every iteration and subtimestep. Proposition 1 allows us to rewrite the explicit iteration process (22) as

$$\underline{\boldsymbol{\alpha}}^{(k)} = \underline{\underline{\mathrm{M}}}^{-1} \underline{r}(\underline{\boldsymbol{\alpha}}^{(k-1)}) = \underline{\underline{\mathrm{M}}}^{-1} \underline{\phi}(t_n) \boldsymbol{\alpha}_n + \Delta t \underline{\underline{\mathrm{M}}}^{-1} \underline{\underline{R}}\, F(\underline{\boldsymbol{\alpha}}^{(k-1)}) = \underline{1}\boldsymbol{\alpha}_n + \Delta t \underline{\underline{\mathrm{M}}}^{-1} \underline{\underline{R}}\, F(\underline{\boldsymbol{\alpha}}^{(k-1)}). \tag{31}$$

Let us define the matrix $\underline{\underline{Q}} := \underline{\underline{\mathrm{M}}}^{-1} \underline{\underline{R}}$ and the vector $\underline{P}$ such that $P_{m=0}^M = \sum_l Q_{ml}$. This last equation is directly used in the first iteration of the ADER process where all coefficients are initialized as $\boldsymbol{\alpha}_n$:

$$\underline{\boldsymbol{\alpha}}^{(1)} = \underline{1}\boldsymbol{\alpha}_n + \Delta t \underline{\underline{Q}}\, F(\underline{\boldsymbol{\alpha}}^{(0)}) = \underline{1}\boldsymbol{\alpha}_n + \Delta t \underline{\underline{Q}}\, F(\underline{1}\boldsymbol{\alpha}_n) = \underline{1}\boldsymbol{\alpha}_n + \Delta t \underline{P} F(\boldsymbol{\alpha}_n),$$

representing the non-zero entries in the first column of the Butcher matrix $\underline{\underline{A}}$. The further $(K-1)$ iterations just use the previous steps as in (31), which give the entries of $\underline{\underline{Q}}$. To achieve order p, we choose $p = K = Y$, i.e., $M = p-1$ for equispaced, $M = \lceil \frac{p}{2} \rceil$ for Gauss–Lobatto and $M = \lfloor \frac{p}{2} \rfloor$ for Gauss–Legendre, for a total amount of $Z = K \times (M+1) + 1$ stages, which for example for equispaced nodes is equal to $Z = p^2 + 1$ [15] (some stages can be avoided when the row is identically 0). We can write the ADER, ImADER and IMEX ADER method of order p as RK methods in a blockdiagonal matrix structure. Then, the Butcher tableaux of the IMEX ADER, which is composed of the explicit and implicit ones, is given by

$$\begin{array}{c|cccccccc}
0 & 0 & & & & & & \\
\underline{P} & \underline{0} & \underline{\underline{Q}} & & & & & \\
\underline{P} & \underline{0} & \underline{\underline{0}} & \underline{\underline{Q}} & & & & \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{Q}} & & & \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{Q}} & & \\
\vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \\
\underline{P} & \underline{0} & \underline{\underline{0}} & \dots & \dots & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{Q}} \\
\hline
 & \underline{0}^T & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{b}^T
\end{array}, \qquad
\begin{array}{c|ccccccc}
0 & 0 & & & & & \\
\underline{P} & \underline{P} & & & & & \\
\underline{P} & \underline{0} & \underline{\underline{Q}} & & & & \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{Q}} & & & \\
\vdots & \underline{0} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{Q}} & & \\
\vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \\
\underline{P} & \underline{0} & \underline{\underline{0}} & \dots & \dots & \underline{\underline{0}} & \underline{\underline{Q}} \\
\hline
 & \underline{0}^T & \underline{0}^T & \dots & & \dots & \underline{0}^T & \underline{b}^T
\end{array}. \tag{32}$$

Note that the implicit matrix $\underline{\underline{A}}$ is particularly sparse and that it is block diagonal. Nevertheless, the method is not diagonally implicit, and also the final update does not have the same coefficients of the last stage.

3.4 A-Stability of ImADER methods

First of all, let us notice that for all ImADER, the only determining iteration is the last one as $\underline{\boldsymbol{\alpha}}^{(k)} = \underline{1}\boldsymbol{\alpha}_n - \Delta t \underline{\underline{Q}}\, S(\underline{\boldsymbol{\alpha}}^{(k)})$ for all k does not depend on previous iterations in the purely implicit case. Hence, the stability function reduces for all ImADER to the $\mathcal{L}^2 = 0$ methods denoted as ADER-IWF-RK in [15]. It is given by the following Butcher tableau

$$\begin{array}{c|c} \underline{P} & \underline{\underline{Q}} \\ \hline & b^T \end{array}. \tag{33}$$

For the Gauss–Lobatto nodes, it has been proven in [15, Theorem A.3] that the ADER-IWF-RK method coincide with the Lobatto IIIC method, hence, its stability function is the Padé$(M-1, M+1)$ approximation, with $M+1$ the number of stages, and for [53, Theorem 4.12] it is A-stable. This means that all ImADER with Gauss–Lobatto nodes are A-stable.
For Gauss–Legendre nodes, we need some theorems that can be found in [53] to prove the same results. First a classical result of [41, 38] on the stability function of a RK method.

Theorem 3.2 ([53, Proposition 3.2]). *The stability function of a RK scheme satisfies*

$$R(z) = \frac{\det(\underline{\underline{Id}} - z\underline{\underline{A}} + z\underline{1}b^T)}{\det(\underline{\underline{Id}} - z\underline{\underline{A}})} = \frac{N(z)}{Q(z)}. \tag{34}$$

Then, we introduce the Padé approximations and the following result.

Theorem 3.3 ([53, Theorem 3.11]). *The (k, j)-Padé approximation to e^z is given by*

$$R_{kj}(z) = \frac{N_{kj}(z)}{Q_{kj}(z)} \tag{35}$$

is the unique rational approximation to e^z of order $j+k$, such that the degrees of the numerator and denominator are k and j, respectively.

Finally, we need the A-stability result for the Padé approximations.

Theorem 3.4 ([53, Theorem 4.12]). *A (k, j)-Padé approximation $R_{kj}(z)$ to e^z is A-stable if and only if $k \leq j \leq k+2$. All zeros and all poles are simple.*

With these theorems, we can proceed studying the A-stability of the implicit ADER methods for Gauss–Legendre nodes. We prove that the stability function of the ADER-IWF-RK with $M+1$ Gauss–Legendre nodes is the Padé$(M, M+1)$. Prior to proceeding, we require an additional outcome concerning the matrices present in the numerator of the stability function.

Proposition 3.5 (Zero determinant of $\underline{\underline{A}} - \underline{1}b^T$). *For the ADER-IWF-RK, we have* $\det(\underline{\underline{A}} - \underline{1}b^T) = 0$.

Proof. Without loss of generality, we consider the interval $[t_n, t_{n+1}] = [0, 1]$ for simplicity. To prove the result, let us recall the definition of the matrices. $\underline{\underline{A}} = \underline{\underline{Q}} = \underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}$ and

$$b_i = w_i = \int_0^1 \phi_i(t)dt, \qquad M_{ij} = \phi_i(1)\phi_j(1) - \int_0^1 \phi_i'(t)\phi_j(t)dt, \qquad R_{ij} = \int_0^1 \phi_i(t)\phi_j(t)dt = \delta_{ij}w_j.$$

Proving that $\det(\underline{\underline{A}} - \underline{1}b^T) = \det(\underline{\underline{M}}^{-1}\underline{\underline{R}} - \underline{1}b^T) = 0$ is equivalent to show that $\det(\underline{\underline{R}} - \underline{\underline{M}}\,\underline{1}b^T) = 0$.
First, we study the matrix $\underline{\underline{M}}\,\underline{1}b^T$. It can be rewritten as follows:

$$(\underline{\underline{M}}\,\underline{1}b^T)_{ij} = \sum_k M_{ik}1_k b_j = \sum_k \left(\phi_i(1)\phi_k(1) - \int_0^1 \phi_i'(t)\phi_k(t)dt\right) w_j = \left(\phi_i(1) - \int_0^1 \phi_i'(t)dt\right) w_j = \phi_i(0)w_j.$$

Then, we define $\underline{\underline{E}} := \underline{\underline{R}} - \underline{\underline{M}}\,\underline{1}b^T$ and we show that the sum of its rows is identically 0, hence, its rows are linearly dependent. It is

$$\sum_i E_{ij} = \sum_i \left(R_{ij} - \phi_i(0)w_j\right) = \sum_i \left(\delta_{ij}w_j - \phi_i(0)w_j\right) = \sum_i \left(\delta_{ij} - \phi_i(0)\right)w_j = (1-1)w_j = 0, \quad \forall j. \tag{36}$$

This proves the statement. □

Theorem 3.6 (Stability function of ADER-IWF-RK Gauss–Legendre)**.** *The stability function of ADER-IWF-RK Gauss–Legendre is the Padé$(M, M+1)$ approximation.*

Proof. In [15, Theorem 3.9], it has been shown that ADER-IWF-RK Gauss–Legendre is of order $2M+1$. Now, since the determinant of $\underline{\underline{A}} - \mathbb{1}b^T$ is zero, see Proposition 3.5, there exists a unitary matrix $\underline{\underline{W}} \in \mathbb{R}^{(M+1)\times(M+1)}$ such that $\underline{\underline{W}}\,\underline{\underline{W}}^T = \underline{\underline{Id}}$ and

$$\underline{\underline{W}}\,(\underline{\underline{A}} - \underline{1}b^T)\underline{\underline{W}}^T = \begin{pmatrix} 0 & 0 & \dots & 0 \\ * & * & \dots & * \\ \vdots & * & \dots & * \\ * & * & \dots & * \end{pmatrix}. \tag{37}$$

Therefore, the numerator of (34) can be expressed as follows:

$$\det\left(\underline{\underline{Id}} - z\underline{\underline{A}} + z\underline{1}b^T\right) = \det\left(\underline{\underline{Id}} + z\underline{\underline{W}}\left(\underline{\underline{A}} - \underline{1}b^T\right)\underline{\underline{W}}^T\right) = \det\left(\begin{pmatrix} 1 & 0^T \\ \underline{*} & \underline{\underline{Id}} - z\underline{\underline{G}} \end{pmatrix}\right) = \det\left(\underline{\underline{Id}} - z\underline{\underline{G}}\right) \tag{38}$$

with $\underline{\underline{G}} \in \mathbb{R}^{M\times M}$. Te degree of $N(z)$ is smaller or equal to M. Since the order of the ADER-IWF-RK Gauss–Legendre is $2M+1$, it must be that the degree of $N(z)$ is M and the degree of $Q(z)$ is $M+1$. Theorem 3.3 establishes that the unique approximation of e^z with an order of $2M+1$, a numerator of degree M, and a denominator of degree $M+1$ is the Padé$(M, M+1)$, implying it aligns with the ADER-IWF-RK Gauss–Legendre method. □

Corollary 3.7 (A-stability of ImADER Gauss–Legendre)**.** *ADER-IWF-RK Gauss–Legendre and all ImADER Gauss–Legendre are A-stable.*

Deriving similar results for equispaced ADER methods poses a challenge. Numerically, starting from the fifth order onward, we notice instabilities along the imaginary axis and unstable regions within $\mathbb{C}^-$, with widths approaching machine precision. This observation suggests the possibility of the stability region intersecting that axis.

4 Convergence Analysis

As seen above, the order of accuracy of the DeC procedure (6) is $\min\{K, Y\}$ where K is the number of iteration and Y the order of the $\mathcal{L}^2$ operator. The proof of this statement, as stated in [1, 16, 15], requires some hypotheses that must be checked for the implicit and IMEX cases.

Proposition 4.1 (DeC iterative method)**.** *Let $\mathcal{L}^1$ and $\mathcal{L}^2$ be two operators defined on $\mathbb{R}^{(M+1)\times I}$, which depend on the discretization scale $\Delta = \Delta t$, such that*

C1. *$\mathcal{L}^1$ is coercive with respect to a norm, i.e., $\exists\, \gamma_1 > 0$ independent of Δ, such that for any $\underline{\boldsymbol{\alpha}}, \underline{\mathbf{d}}$*

$$\gamma_1||\underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}}|| \leq ||\mathcal{L}^1(\underline{\boldsymbol{\alpha}}) - \mathcal{L}^1(\underline{\mathbf{d}})||,$$

C2. *$\mathcal{L}^1 - \mathcal{L}^2$ is Lipschitz with constant $\gamma_2 > 0$ uniformly with respect to Δ, i.e., for any $\underline{\boldsymbol{\alpha}}, \underline{\mathbf{d}}$*

$$||(\mathcal{L}^1(\underline{\boldsymbol{\alpha}}) - \mathcal{L}^2(\underline{\boldsymbol{\alpha}})) - (\mathcal{L}^1(\underline{\mathbf{d}}) - \mathcal{L}^2(\underline{\mathbf{d}}))|| \leq \gamma_2\Delta||\underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}}||,$$

C3. *there exists a unique* $\underline{\boldsymbol{\alpha}}^*$ *such that* $\mathcal{L}^2(\underline{\boldsymbol{\alpha}}^*) = 0$.

Then, if $\eta := \frac{\gamma_2}{\gamma_1}\Delta < 1$*, the DeC is converging to* $\underline{\boldsymbol{\alpha}}^*$ *and after* k *iterations the error* $||\underline{\boldsymbol{\alpha}}^{(k)} - \underline{\boldsymbol{\alpha}}^*||$ *is smaller than* $\eta^k ||\underline{\boldsymbol{\alpha}}^{(0)} - \underline{\boldsymbol{\alpha}}^*||$.

Proofs of this proposition and of the hypotheses of the proposition for operators $\mathcal{L}^1$ and $\mathcal{L}^2$ for explicit DeC and ADER can be found in [1, 3, 32]. The condition for η comes from the fixed–point theorem and it is needed to converge. As foreshadowed in Proposition 4.1, we need to prove the conditions **C1.**, **C2.** and **C3.** also in the implicit and IMEX cases, as for the explicit cases this was shown in [16]. Here, we want to extend this proof to the IMEX $\mathcal{L}^1$ operators. The proof of **C3.** is as in the explicit case, because just the $\mathcal{L}^1$ operator changes in the implicit and IMEX cases. The arguments to prove **C2.** are also exactly the same as in the explicit case, because it all boils down to the Lipschitz continuity of the right hand side of the ODE. In the general scenario, we find ourselves unable to prove coercivity. Therefore, we opt to linearize the stiff term in $\mathcal{L}^1$, as outlined in [16]. This linearization still provides a first-order approximation to the implicit terms. We substitute $S(\underline{\boldsymbol{\alpha}})$ with $S'(\underline{1}\boldsymbol{\alpha}_n)\underline{\boldsymbol{\alpha}}$, where S' is the Jacobian of S. Note that this simplification is exact for linear systems where $S(\underline{\boldsymbol{\alpha}}) = \underline{\underline{S}}'(\underline{1}\boldsymbol{\alpha}_n)\underline{\boldsymbol{\alpha}}$. Moreover, this formulation can also be used to incorporate the nonlinear solver inside the DeC iteration method, without the need of further nonlinear solvers [16].

Proposition 2 (IMEX DeC: **C1.**)**.** *Assume that we apply a first order approximation of the IMEX DeC method linearizing the implicit terms, i.e.,* $\mathcal{L}^1$ *is defined as*

$$\tilde{\mathcal{L}}^1(\underline{\boldsymbol{\alpha}}) := \underline{\boldsymbol{\alpha}} - \underline{1}\boldsymbol{\alpha}_n - \Delta t \underline{\beta} S'(\underline{1}\boldsymbol{\alpha}_n)(\underline{\boldsymbol{\alpha}} - \underline{1}\boldsymbol{\alpha}_n) - \Delta t \underline{\beta}\left(S(\underline{1}\boldsymbol{\alpha}_n) + F(\underline{1}\boldsymbol{\alpha}_n)\right). \tag{39}$$

Let

$$\Delta t < \frac{1}{2\tilde{\beta}L}, \tag{40}$$

where $\tilde{\beta} := \max_{1\leq m\leq M}\{\beta^i\} \leq 1$ *and* L *is the Lipschitz constant of* S*. Then, given any* $\underline{\boldsymbol{\alpha}}, \underline{\mathbf{d}} \in \mathbb{R}^{(M+1)\times I}$*, there exists a positive* C_0*, such that*

$$C_0||\underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}}|| \leq ||\tilde{\mathcal{L}}^1(\underline{\boldsymbol{\alpha}}) - \tilde{\mathcal{L}}^1(\underline{\mathbf{d}})||$$

is fulfilled for the $\tilde{\mathcal{L}}^1$ *IMEX DeC operator.*

Proof. First, we recall some basic properties for eigenvalues, which we will use in the proof.

i) Let λ be an eigenvalue of $\underline{\underline{A}} \in \mathbb{C}^{n\times n}$. Then, it holds $|\lambda| \leq \left\|\underline{\underline{A}}\right\|$.

ii) Let $\underline{\underline{A}} \in \mathbb{C}^{n\times n}$, $\underline{\underline{Id}}$ the n-dimensional identity matrix and $\gamma, \delta \in \mathbb{C}$ and $\lambda \in \mathbb{C}$ an eigenvalue of $\underline{\underline{A}}$. Then, $\gamma\lambda + \delta$ is an eigenvalue of $\gamma\underline{\underline{A}} + \delta\underline{\underline{Id}}$.

We know that $\|S'(\boldsymbol{\alpha})\|$ is bounded by L, the Lipschitz continuity constant of S, and, by property i), all the absolute values of the eigenvalues $\lambda_{S'}(\boldsymbol{\alpha})$ of $S'(\boldsymbol{\alpha})$ are bounded by L for every $\boldsymbol{\alpha} \in \mathbb{R}^I$. Now, using property ii) and the restriction (40), we have that for every eigenvalue λ_{β_m} of $\Delta t \beta_m S'(\boldsymbol{\alpha})$ it holds $|\lambda_{\beta_m}(\boldsymbol{\alpha})| < \frac{1}{2}$. Using property ii), we can estimate for each $m = 0, \dots, M$ the absolute value of the eigenvalues of

$$\underline{\underline{Z_m}} := \underline{\underline{Id}} - \Delta t \beta^m S'(\boldsymbol{\alpha}),$$

which are therefore all larger than $\frac{1}{2}$ for every $\boldsymbol{\alpha} \in \mathbb{R}^I$, $1 \leq m \leq M$, leading to the invertibility of $\underline{\underline{Z_m}}$. We consider the block-diagonal matrix $\underline{\underline{Z}}$ with $\underline{\underline{Z_m}}$ on each block-entry, that correspond to the system matrix of $\tilde{\mathcal{L}}^1$. The eigenvalues of $\underline{\underline{Z}}^{-1}$ are the reciprocal of the eigenvalues of $\underline{\underline{Z}}$, hence, all smaller than 2, and so $\|\underline{\underline{Z}}^{-1}\| \leq 2$. Note that for any $\underline{\boldsymbol{\alpha}}, \underline{\mathbf{d}} \in \mathbb{R}^{(M+1)\times I}$, it holds

$$\|\underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}}\| = \|\underline{\underline{Z}}^{-1}\underline{\underline{Z}}(\underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}})\| \leq \|\underline{\underline{Z}}^{-1}\|\|\underline{\underline{Z}}(\underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}})\| \leq 2\|\underline{\underline{Z}}(\underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}})\|. \tag{41}$$

By considering the $\tilde{\mathcal{L}}^1$ operator of the ImDeC, we expand

$$\begin{aligned}\tilde{\mathcal{L}}^1(\underline{\boldsymbol{\alpha}})-\tilde{\mathcal{L}}^1(\underline{\mathbf{d}})&=\underline{\boldsymbol{\alpha}}-\underline{\boldsymbol{\alpha}}^0-\Delta t\underline{\beta}S'(\underline{\boldsymbol{\alpha}}^0)\underline{\boldsymbol{\alpha}}-\Delta t\underline{\beta}F(\underline{\boldsymbol{\alpha}}^0)-\underline{\mathbf{d}}+\underline{\mathbf{d}}^0+\Delta t\underline{\beta}S'(\underline{\boldsymbol{\alpha}}^0)\underline{\mathbf{d}}+\Delta t\underline{\beta}F(\underline{\boldsymbol{\alpha}}^0)\\&=\left(\underline{\underline{Id}}-\Delta t\underline{\beta}S'(\underline{\boldsymbol{\alpha}}^0)\right)(\underline{\boldsymbol{\alpha}}-\underline{\mathbf{d}})=Z\left(\underline{\boldsymbol{\alpha}}-\underline{\mathbf{d}}\right).\end{aligned}$$

Finally, we conclude that

$$\left\|\tilde{\mathcal{L}}^1(\underline{\boldsymbol{\alpha}})-\tilde{\mathcal{L}}^1(\underline{\mathbf{d}})\right\|=\left\|\underline{\underline{Z}}\left(\underline{\boldsymbol{\alpha}}-\underline{\mathbf{d}}\right)\right\|\geq\frac{1}{2}\left\|\underline{\boldsymbol{\alpha}}-\underline{\mathbf{d}}\right\|, \tag{42}$$

thanks to (41). □

Proposition 3 (IMEX ADER: **C1.**)**.** *Assume we apply a first order linear approximation of the IMEX ADER method, i.e., we change the $\mathcal{L}^1$ operator to*

$$\tilde{\mathcal{L}}^1(\underline{\boldsymbol{\alpha}}):=\underline{\boldsymbol{\alpha}}-\underline{\boldsymbol{\alpha}}^0-\Delta t\underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\,S'(\underline{\boldsymbol{\alpha}}^0)(\underline{\boldsymbol{\alpha}}-\underline{\boldsymbol{\alpha}}^0)-\Delta t\underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\,(S(\underline{\boldsymbol{\alpha}}^0)+F(\underline{\boldsymbol{\alpha}}^0)). \tag{43}$$

Let

$$\Delta t<\frac{1}{2CL}, \tag{44}$$

where $C=\left\|\underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\right\|=\mathcal{O}(1)$ and L is the Lipschitz constant of S. Then, given any $\underline{\boldsymbol{\alpha}},\underline{\mathbf{d}}\in\mathbb{R}^{(M+1)\times I}$, there exists a positive C_0, such that

$$C_0||\underline{\boldsymbol{\alpha}}-\underline{\mathbf{d}}||\leq||\tilde{\mathcal{L}}^1(\underline{\boldsymbol{\alpha}})-\tilde{\mathcal{L}}^1(\underline{\mathbf{d}})||$$

is fulfilled for the $\tilde{\mathcal{L}}^1$ IMEX ADER operator.

Proof. Also for the $\tilde{\mathcal{L}}^1$ operator of the ImADER, the proof is similar to the ImDeC one. We know that $\|S'(\boldsymbol{\alpha})\|\leq L$ holds for every $\boldsymbol{\alpha}$ and $C=\left\|\underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\right\|>0$, because $\underline{\underline{M}}$ and $\underline{\underline{R}}$ are constant. Therefore, we can deduce that the eigenvalues $\lambda_{M^{-1}R}(\underline{\boldsymbol{\alpha}})$ of $\underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\,S'(\underline{\boldsymbol{\alpha}})$ can be estimated by

$$|\lambda_{M^{-1}R}(\underline{\boldsymbol{\alpha}})|\leq\left\|(\underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}})S'(\underline{\boldsymbol{\alpha}})\right\|\leq\left\|\underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\right\|\left\|S'(\underline{\boldsymbol{\alpha}})\right\|\leq C\cdot L$$

for every $\boldsymbol{\alpha}$. This, combined with condition (44), leads us again to the property that all the absolute values of the eigenvalues of

$$\underline{\underline{Id}}-\Delta t\underline{\underline{\mathrm{M}}}^{-1}\underline{\underline{R}}\,S'(\underline{\boldsymbol{\alpha}})$$

are bigger than $\frac{1}{2}$. With the same arguments as in the proof of Proposition 2, we conclude that $\tilde{\mathcal{L}}^1$ is coercive. □

In many applications, the hypothesis on the time-step as assumed in Propositions 2 and 3 are too restrictive, especially when considering stiff equations. So, we present a variation of this proof, which does not require these time-step restrictions but uses, instead, another assumption on S that is typical for damping/diffusion operators.

Proposition 4 (IMEX DeC/ADER: Variation of **C1.** for diffusion terms)**.** *Consider again the first order approximations $\tilde{\mathcal{L}}^1$ as in (39) for the IMEX DeC and in (43) for IMEX ADER. Assume additionally that the Jacobian S' is symmetric negative definite. Then, given any $\underline{\boldsymbol{\alpha}},\underline{\mathbf{d}}\in\mathbb{R}^{(M+1)\times I}$, there exists a positive $C_0\geq 1$ independent of Δt, such that*

$$C_0||\underline{\boldsymbol{\alpha}}-\underline{\mathbf{d}}||\leq||\tilde{\mathcal{L}}^1(\underline{\boldsymbol{\alpha}})-\tilde{\mathcal{L}}^1(\underline{\mathbf{d}})||$$

is fulfilled for both $\tilde{\mathcal{L}}^1$ operators.

Proof. We start from the IMEX DeC. Using again property ii) from the proof of Proposition 2, we can deduce immediately that $\underline{\underline{Z}}:=\underline{\underline{Id}}-\Delta t\beta^m S'$ is symmetric positive definite and any eigenvalue λ_Z is larger than

$1 + \Delta t \beta^m |\lambda_{S'}| > 1$. Hence, the eigenvalues of $\underline{\underline{Z}}^{-1}$ are all smaller than one. Therefore, as in Proposition 2, we have that

$$\left\| \tilde{\mathcal{L}}^1(\underline{\boldsymbol{\alpha}}) - \tilde{\mathcal{L}}^1(\underline{\mathbf{d}}) \right\| = \left\| \underline{\underline{Z}}\,(\underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}}) \right\| \geq \frac{1}{\left\| \underline{\underline{Z}}^{-1} \right\|} \left\| \underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}} \right\| > \left\| \underline{\boldsymbol{\alpha}} - \underline{\mathbf{d}} \right\|. \tag{45}$$

This result holds independently of the size of Δt. For IMEX ADER the matrix in consideration is

$$\underline{\underline{Z}} := \underline{\underline{Id}} - \Delta t \underline{\underline{M}}^{-1} \underline{\underline{R}}\, S'(\boldsymbol{\alpha}_n),$$

where, implicitly, there is a Kronecker product between $\underline{\underline{M}}^{-1}\underline{\underline{R}} \in \mathbb{R}^{(M+1)\times(M+1)}$ and $S'(\boldsymbol{\alpha}_n) \in \mathbb{R}^{I \times I}$. We know that $-S'(\boldsymbol{\alpha}_n)$ is positive definite, if also $\underline{\underline{M}}^{-1}\underline{\underline{R}}$ is positive define. Then, we have that their Kronecker product is positive definite [50, Section 1]. In [15], it has been shown that $\underline{\underline{M}}$ is invertible and it is equivalent to a Hilbert matrix with an extra column of zeros on the left and an extra row of ones on the top. Since, Hilbert matrices are positive definite, also $\underline{\underline{M}}$ is positive definite and its inverse is positive definite as well. Now, $\underline{\boldsymbol{\alpha}}^T \underline{\underline{R}}\, \underline{\boldsymbol{\alpha}} = \int_0^1 \boldsymbol{\alpha}(t)^2 \geq 0$ when computed exactly, and for Gauss–Lobatto nodes, where the quadrature is not exact [15], it is $\underline{\boldsymbol{\alpha}}^T \underline{\underline{R}}\, \underline{\boldsymbol{\alpha}} = \sum_{m=0}^{M} w_m (\boldsymbol{\alpha}^m)^2 \geq 0$ as $w_m > 0$. Hence, $\underline{\underline{R}}$ is positive definite and all eigenvalues of $\underline{\underline{Z}}$ are $\lambda_Z > 1$. We can then proceed as in (45) to show that $\tilde{\mathcal{L}}^1$ is coercive with $C_0 \geq 1$. □

5 Numerical Stability Analysis

In this section, we study the stability of the presented method for the linear Dahlquist equation $u' = -\lambda u$ or $u' = -\lambda_I u - \lambda_E u$ for IMEX methods. All methods can be rewritten as $u_{n+1} = R(z) u_n$ or $u_{n+1} = R(z_I, z_E)$ being R stability functions and $z, z_I, z_E \in \mathbb{C}$. We will study the stability regions $\{|R| \leq 1\} \subset \mathbb{C}$ for the implicit ADER/DeC, while for the IMEX schemes we consider different approaches. The explicit cases the stability functions for DeC and ADER have already been investigated in [16] and reported, while we show in the repository [31] the sDeC for completeness. We will use Gauss-Lobatto (GLB) nodes as quadrature nodes, while we compare equispaced nodes also in the repository [31] and we highlight here only the main differences. We will numerically compute the stability regions obtained from the stability functions of ADER/DeC that are defined through their Butcher tableaux (14), (17), (32), see [14]. In detail, we compute on 200×200 grid points with an offset of $+0.01$ from the origin for both axes to avoid singularities. The plot bounds are dependent on the type of scheme and their stability regions. We decreased the offset in Figures 3 to a fraction of 10^{-2} of the largest real value displayed when zooming on small areas.

To distinguish the different orders, we apply different colors and line styles to the outer and inner bounds according to the legend that are plot next to each stability region plot.

5.1 Implicit schemes

In the following, we plot the contour lines of the bounds of the stability regions of various implicit methods. We start from ImDeC and ImADER schemes in Figure 1. Clearly, all these stability regions are unbounded, but they are not all A-stable, as we will see soon. Moreover, we can observe a great variability changing the scheme or the nodes, in opposition to the explicit case [16]. In most of the cases, ImADER have larger stability regions than ImDeC.

The stability regions for the implicit sDeC (ImsDeC) are also shown in Figure 1 and, surprisingly, do not behave like the other methods. Up to a certain order (i.e. sDeC8 with Gauss-Lobatto), the stability regions are unbounded and *seem* A-stable, but for higher orders, we lose this property, obtaining large, but finite, stability regions. This behavior is not uniform and, at certain orders, the stability region will be unbounded again, as for example shown in Figure 2 for very high order ImsDeC. In detail, the methods with bounded stability region up to order 20 are the ImsDeC with Gauss–Lobatto nodes with orders 9, 10, 11, 12, 13, 14 and 15 and with equispaced nodes with orders 12, 13, 16, 17, 18, 19 and 20. For the sDeC, we can conclude that the choice of an implicit version does not guarantee an unbounded stability region. Nevertheless, even these implicit sDeC methods have larger stability regions than their explicit counterparts and therefore may

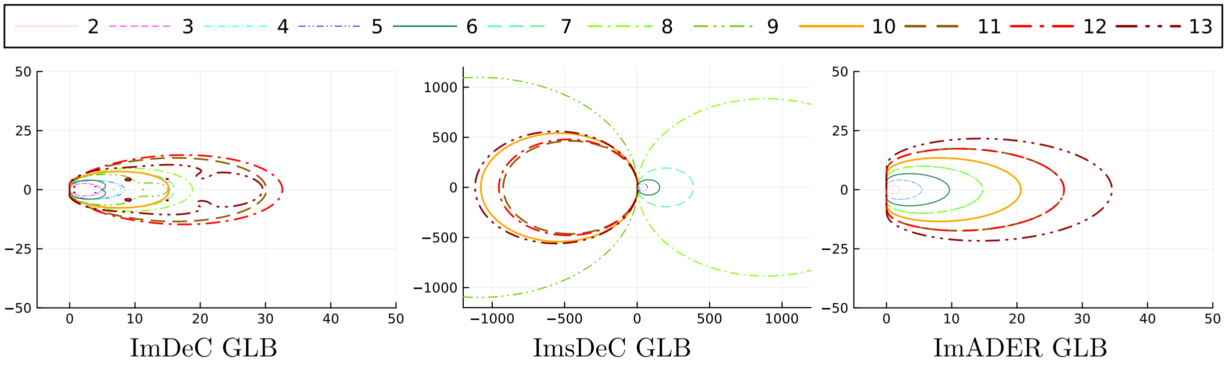


Figure 1: Implicit DeC (left), sDeC (center) and ADER (right) with equispaced (top) and Gauss-Lobatto (bottom) nodes for orders 2 to 13

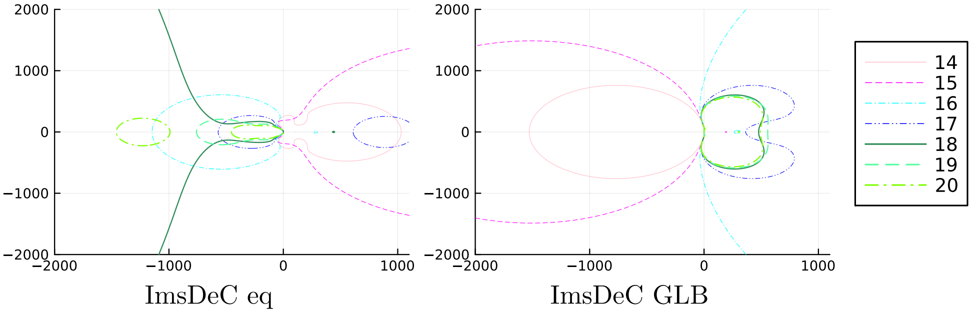
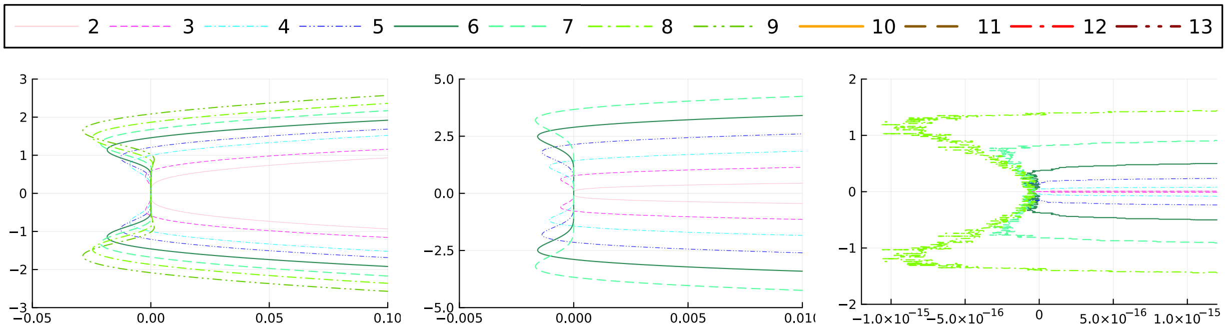


Figure 2: Implicit sDeC for orders 14 to 20

be applicable to mildly stiff problems. We notice again that this odd loss of stability in the left half plane could not be found in the ImDeC and ImADER methods. We checked it numerically up to order 50.
Taking a closer look at the implicit methods, we additionally detect some minor instability regions on the negative half-plane, see Figure 3. It turns out that these instabilities appear for all ImDeC and ImsDeC methods of orders larger than 2 and both types of nodes. We display the sDeC only with GLB nodes, as the equispaced nodes version is very similar to that. For a very small scale, the same can be seen for the ImADER methods with equispaced nodes of orders at least larger than 4, displayed on the bottom right of Figure 3. Notice that the sizes of the unstable regions are close to machine precision, which results in non-smooth boundaries and it is unclear if the ImADER with equispaced nodes are not A-stable or the visualization of the unstable area is given by machine precision errors.
As proved in Section 3.4, also numerically we observe that the ImADER methods with Gauss-Lobatto nodes are A-stable for all orders.
Summarizing, we can categorize our methods in 3 different classes:

- The A-stable schemes: all ImADER GLB and all second order implicit methods;
- The *almost A-stable* schemes, when the stability region is unbounded and it *almost* includes the whole left half–plane: high order ImDeC, some ImsDeC and ImADER equispaced;
- The bounded stability schemes: some ImsDeC.

Remark that for *almost A-stable* methods these minor instabilities do not influence the behavior of the scheme on many stiff problems. Nevertheless, when the eigenvalues λ of the system are (almost) purely imaginary (typical for high order advection operators), they might encounter instabilities for some discretizations.



ImDeC GLB from order 2 to 9 ImsDeC GLB from orders 2 to 7 ImADER eq from orders 3 to 8

Figure 3: Zoomed stability region of various implicit schemes

5.2 IMEX schemes

To study the stability of IMEX schemes, we will use the RK stability function

$$R(z_I, z_E) = 1 + \left(z_I \underline{b}^T + z_E \hat{\underline{b}}^T \right) \underline{u} = 1 + \left(z_I \underline{b}^T + z_E \hat{\underline{b}}^T \right) \left(\underline{\underline{Id}} - z_I \underline{\underline{A}} - z_E \underline{\underline{\hat{A}}} \right)^{-1} \underline{1} \tag{46}$$

that uses the matrices defined by the Butcher tableau of an IMEX RK (16). Remark that the standard approach of A-stability cannot be used anymore. Indeed, the region of absolute stability

$$S = \left\{ (z_I, z_E) \in \mathbb{C}^2 \;:\; |R(z_I, z_E)| \leq 1 \right\}$$

lays in a larger space, with respect to classical RK schemes, therefore, its study, computation and visualization are challenging. Hence, we need to rely on some simplifications. In [28], Minion simplifies the Dahlquist equation by imposing

$$\lambda_I \in \mathbb{R}, \quad \lambda_E = i\lambda_E', \; \lambda_E' \in \mathbb{R}.$$

This procedure neglects respectively the imaginary or real part of the coefficients in the Dahlquist equation to display a two-dimensional region. This idea is lead by classical PDE discrete operators, where typically the diffusion is symmetric negative definite, while the advection is mainly with imaginary eigenvalues. A second approach where for each λ_E the A-stability is required for the implicit part of the scheme was originally studied in [54, 6] and formalized in [21]. Another approach studies, instead, the stability for each λ_I requiring at least the stability region of the explicit Euler method to the explicit part [17]. We collect these definitions of stability region in the following.

Definition 5.1 (Stability regions)**.** *Consider the modified test equation with stability function* (46)*. Then, we define multiple approaches for IMEX stability regions by*

- $S := \left\{ (z_I, z_E) \in \mathbb{C}^2 \;:\; |R(z_I, z_E)| \leq 1 \right\}$ *(Region of absolute stability),*
- $\mathcal{D}_M := \left\{ (z_I, z_E) \in \mathbb{R}^2 \;:\; |R(z_I, iz_E)| \leq 1 \right\}$ *(Minion's stability region) [28],*
- $\mathcal{D}_0 := \{ z_E \in \mathbb{C} \;:\; |R(z_I, z_E)| \leq 1$ *for any* $z_I \in \mathbb{C}^- \}$ *[21],*
- $\mathcal{D}_1 := \{ z_I \in \mathbb{C} \;:\; |R(z_I, z_E)| \leq 1$ *for any* $z_E \in \mathcal{S}_0 \}$ *[17],*

where $\mathcal{S}_0 = \{ z_E \in \mathbb{C} \;:\; |1 + z_E| \leq 1 \}$ *is the stability region of the explicit Euler method.*

$\mathcal{D}_0$ is a very strict condition of IMEX stability, in particular for the considered high order schemes. Theoretically, the terms A-stability and A(α)-stability may be applied for all 3 of these subsets of $\mathbb{C}$ analogously to the classical cases, so we will make use of this terminology too.

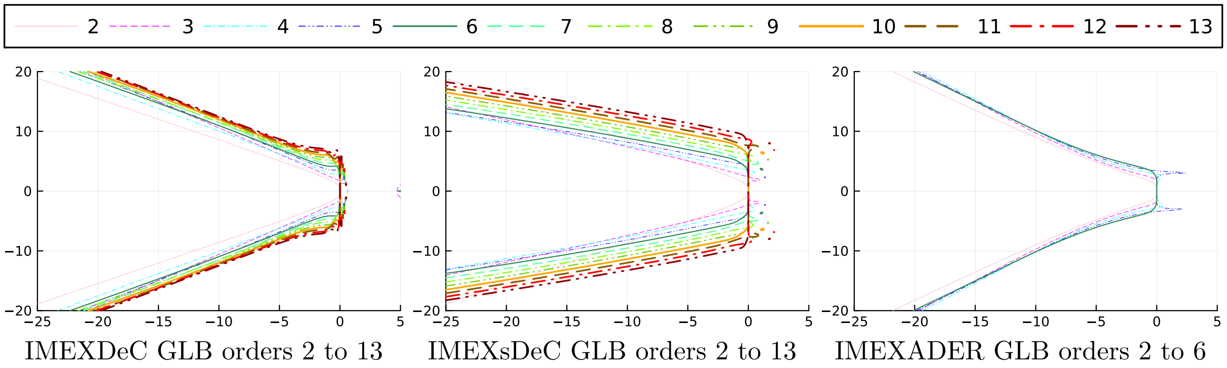


Figure 4: Minion's stability region for IMEX DeC (left), sDeC (center) and ADER (right) with equispaced (top) and GLB (bottom) nodes

$\mathcal{D}_M$ stability region

We recall that the plots of the stability regions have very different meaning according to the chosen approach. Starting from Minion's approach [28], we evaluate the IMEX stability function (46) numerically to calculate the respective stability regions.

For the IMEX DeC, we can observe in Figure 4 (left) that the choice of nodes change the regions on some details but the qualitative behavior is the same. We can also conclude on an $A(\alpha)-$stability for approximately $\alpha = 35°$.

Going on to the IMEX ADER, we can see in Figure 4 (right) a similar behavior, even if the stability regions differ in small details, we observe $A(\alpha)-$stability for at least $\alpha = 35°$.

Finally, for the IMEX sDeC method in Figure 4 (center) we see a slightly different behavior, still resulting in an $A(\alpha)-$stability, but for significantly smaller angles, approximately $\alpha = 18°$. Nevertheless, the result on the bottom center in Figure 4 with GLB nodes coincides with the one in [28], as expected. It is also noticeable that the IMEX sDeC stability region of order 2 is $A(\alpha)$-stable with larger α as it coincides with the IMEX DeC2.

$\mathcal{D}_0$ stability region

Now, we want to evaluate $\mathcal{D}_0$ stability for our IMEX methods. We want to emphasize that the requirements here are stricter than in Minion's approach. Indeed, for $\mathcal{D}_0$ we require the method to be at least fully A-stable for the implicit part and we look at the stability of the explicit part. The IMEX DeC and IMEX sDeC have $\mathcal{D}_0 = \emptyset$ and this is probably related to the fact that their implicit counterpart is not A-stable. For the IMEX ADER, only few orders have non-empty $\mathcal{D}_0$ stability region. In Figure 5, we show the few stability regions, which eventually vanish when increasing the order of accuracy.

$\mathcal{D}_1$ stability region

We plot the $\mathcal{D}_1$ stability regions for DeC and sDeC methods in Figures 6, where we require the explicit part to cover the stability region of the explicit Euler method and we look at the stability of the implicit part. Contrary to the $\mathcal{D}_0$ cases, we observe non-empty, limited regions of stability for every order for the IMEX DeC methods. Moreover, there is no regularity in their shape and their size grow significantly as the order of accuracy increases. Notice that the plots do not show the full stability regions of higher orders, for example for orders 6, 7, and 8 with equispaced nodes, but they are anyway bounded regions.

For the IMEX sDeC, see Figure 6 (center), we observe some remarkable differences. In the case of equispaced nodes, even orders just show the small bounded stability regions in the negative half-plane nearby the origin, odd orders smaller than 6 show large stability regions, while they are unstable starting from order 7. Also

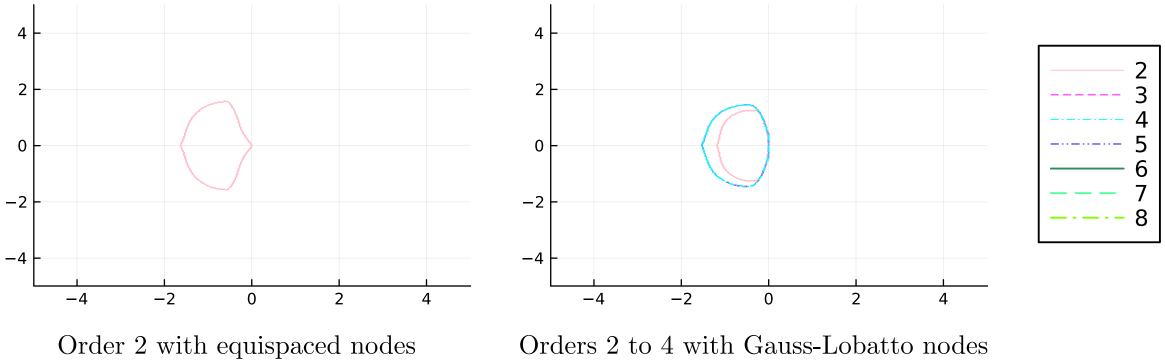


Figure 5: $\mathcal{D}_0$ Stability regions for IMEX ADER. The smaller stability region displays order 2, while the larger one displays order 3 and 4 (right)

in the GLB case, we do not observe much regularity. We notice that the largest stability region is obtained for order 5, while, for higher orders, the stability region almost fit in the plot.
In Figure 6 (right), we show the results for the IMEX ADER methods. We note that most of the methods fulfill nearly A-stability by almost covering the negative half-plane. We highlight that for the equispaced case we do not show the full outreach of the stability regions. While in most of the cases, for the almost A-stable cases, these contour lines represent the inner bounds of unlimited stability regions, in the cases of order 5 and 8 we just have large, limited stability regions, as we also could observe for example in the $\mathcal{D}_1$ IMEX DeC case for equispaced nodes. Therefore, it seems like we can not guarantee this almost A-stability for the IMEX ADER, but just for some of the orders of accuracy.
Therefore, we can conclude, that some of the IMEX ADER stability regions cover the areas in the complex plane, that we assumed for a stable method in the context of $\mathcal{D}_0$ and $\mathcal{D}_1$, while the DeC and sDeC methods have their limitations with $\mathcal{D}_0 = \emptyset$ in every case and bounded stability regions for most of the cases in the scope of $\mathcal{D}_1$. Nevertheless, we need to keep in mind that the set conditions are very strict, so, the methods might still be applicable to some stiff equations. These results reflect what we have seen for the respective implicit methods.

6 PDE: analysis of advection-diffusion

In this section, we want to extend our stability analysis to the one-dimensional advection-diffusion equation

$$u_t(x,t) + au_x(x,t) = du_{xx}(x,t), \quad a \geq 0,\ d \geq 0, \qquad x \in \Omega \subset \mathbb{R}, \tag{47}$$

where a is the coefficient of the advection term and d the coefficient of the diffusion term, using the von Neumann stability analysis. After the space discretization, we discretize the advection part with an explicit time-integration scheme and the diffusion with an implicit one. The linear stability of the DeC method in PDE contexts was studied for explicit methods for advection equations with FEM spatial discretizations and various stabilization techniques in [26, 27], while in the IMEX context for FEM methods applied to kinetic models in [48]. For the ADER method, a von Neumann stability analysis was applied to the original formulation [46, 8], but not on the modern version that we are studying. We close this gap with our investigation in the following.

6.1 Finite Difference discretization

We apply spatial discretizations to the spatial derivative operators, namely ∂_x and ∂_{xx}. We consider a uniformed grid $\Omega_{\Delta x} = \{x_j : x_j = x_0 + j\Delta x,\ j \in \{0, \dots, J\}\}$ with periodic boundary conditions and we denote the approximation of $u(x_j) = u(x_j, t)$ by w_j.

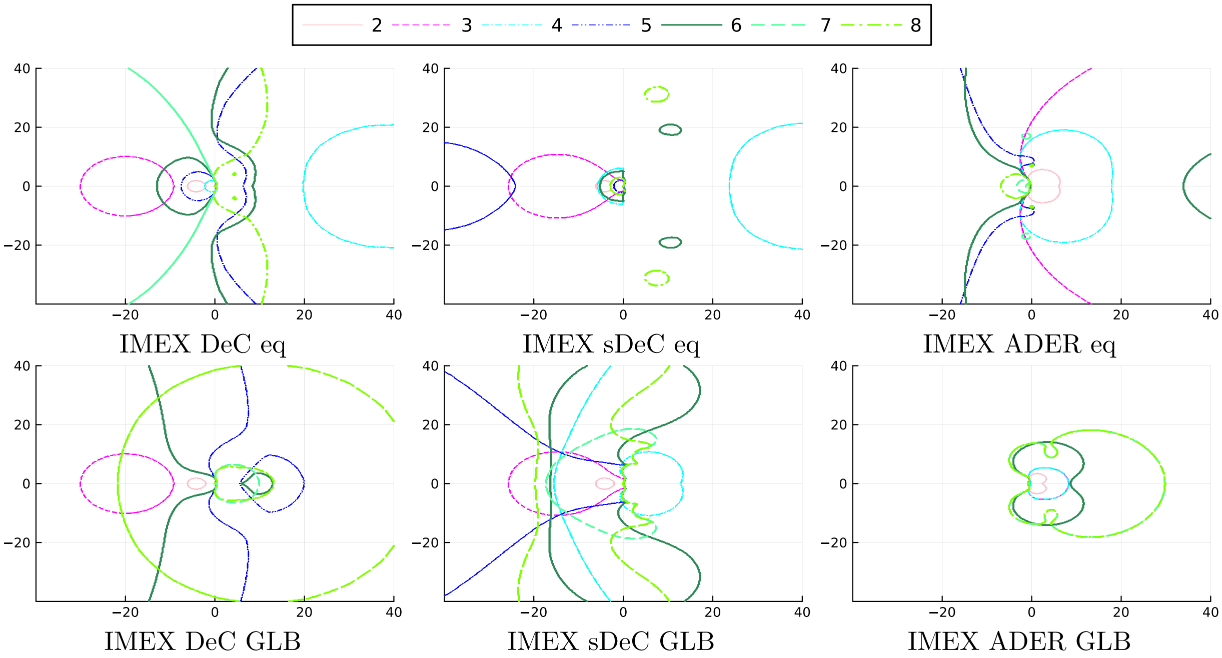


Figure 6: $\mathcal{D}_1$ Stability Region for IMEX DeC (left), sDeC (center) and ADER (right) with equispaced (top) and GLB (bottom) nodes: orders 2 to 8

To discretize the advection term in (47), i.e., the first spatial derivative $\partial_x u(x)$, we make usage of the stable finite difference stencils introduced in [18]. Assume we discretize $\partial_x u$ at x_j by an $[r, s]$-discretization

$$\partial_{\Delta x}^{[r,s]}(u(x_j)) = \frac{1}{\Delta x} \sum_{k=-r}^{s} \alpha_k w_{j+k}, \tag{48}$$

with $r,\ s$ such that $\alpha_{j-r}, \alpha_{j+s} \neq 0$. The maximum order we can achieve with an $[r, s]$-discretization is $q = r + s$ and this discretization actually reaches order q and is unique by setting the coefficients in (48) as

$$\alpha_0 = \begin{cases} \sum\limits_{k=r+1}^{s} \frac{1}{k}, & s \geq r+1, \\ 0, & s = r, \\ \sum\limits_{k=s+1}^{r} \frac{1}{k}, & r \geq s+1, \end{cases} \qquad \alpha_k = \frac{(-1)^{k+1}}{k} \cdot \frac{r!s!}{(r+k)!(s-k)!}, \quad -r \leq k \leq s,\ k \neq 0. \tag{49}$$

It is also proven in [18] that these so-called *optimal-order* schemes of order q are stable if and only if $s \leq r \leq s + 2$ for $a > 0$. We involve these stable *optimal-order* schemes into our analysis. We introduce upwinding in the choice of the stencils, in particular, we will consider $[r, r + 1]$ stencils for odd optimal-order scheme and $[r, r + 2]$ stencils for an even optimal-order scheme for the advection part.

For the diffusion term, we will just use a central finite difference discretization of the second spatial derivative $\partial_{xx} u(x)$ given in Table 1.

6.2 von Neumann analysis

To analyze the stability of the described methods, we make use of the von Neumann stability analysis for linear partial differential equations [20]. Briefly summarized, we investigate the behavior inside the numerical scheme of the Fourier modes

$$w_j^n = v^n e^{ikx_j}, \tag{50}$$

Table 1: Central finite difference discretizations of ∂_{xx} applied onto w centered in j [12]

order	finite difference for $\partial^2_{\Delta x}(u(x_j))$
2	$\frac{1}{\Delta x^2}\left(w_{j-1} - 2w_j + w_{j+1}\right)$
4	$\frac{1}{\Delta x^2}\left(-\frac{1}{12}w_{j-2} + \frac{4}{3}w_{j-1} - \frac{5}{2}w_j + \frac{4}{3}w_{j+1} - \frac{1}{12}w_{j+2}\right)$
6	$\frac{1}{\Delta x^2}\left(\frac{1}{90}w_{j-3} - \frac{3}{20}w_{j-2} + \frac{3}{2}w_j - \frac{49}{18}w_j + \frac{3}{2}w_j - \frac{3}{20}w_{j+2} + \frac{1}{90}w_{j+3}\right)$
8	$\frac{1}{\Delta x^2}\left(-\frac{1}{56}w_{j-4} + \frac{1}{420}w_{j-3} - \frac{1}{5}w_{j-2} + \frac{8}{5}w_{j-1} - \frac{205}{72}w_j + \frac{8}{5}w_{j+1} - \frac{1}{5}w_{j+2} + \frac{1}{420}w_{j+3} - \frac{1}{56}w_{j+4}\right)$

where w_j^n is the discretization of $u(x_j, t_n)$ and k is the wavenumber and we focus on the representation coefficient v^n. Indeed, e^{ikx} are eigenfunctions of the differential operator ∂_x and therefore for any linear differential operator. If we use (50) in our discretized system, we obtain a system of the form

$$v^{n+1} = G(k, \Delta x, \Delta t, a, d)v^n. \tag{51}$$

with the amplification factor $G \in \mathbb{C}$ independent on the mesh point x_j. Stability means in our context that $|v^{n+1}| = |G(k, \Delta x, \Delta t, a, d)v^n| \leq |v^n|$ holds. In practice, we check that $|G(k, \Delta x, \Delta t, a, d)| \leq 1$. This implies stability for the related method and due to the Lax-Richtmeyer theorem [20] convergence can be ensured. Note that for consistency, we included the parameters a and d into the dependency of G to cover all advection-diffusion equations.

Typically, in order to estimate the stability of the advection-diffusion equation, an analytical study of G in all the parameters should be performed. This is not feasible when considering high order schemes as ADER and DeC. Hence, we will evaluate the amplification factor numerically, similarly to what we did with the stability functions in the ODE case.

Before running all the simulations, we need to understand what are the free variables of the function G. First of all, the wavenumbers should be bounded $k \in \{-n_0 - 1, \dots, n_0 + 1\} \subset \mathbb{Z}$ and the maximum wavenumber $n_0 + 1$ is strongly related with the discretization scale Δx. Indeed, by Nyquist–Shannon sampling theorem, only functions with frequency less than $\frac{|x_J - x_0|}{2\Delta x}$ can be represented on our discretization. Hence, we will choose $n_0 = 10^3$ to take in consideration fine grids.

Then, we have further variables $a, d, \Delta x, \Delta t$ that are actually coupled together: in the advection term $(a\Delta t/\Delta x)$ and in the diffusion term $(d\Delta t/\Delta x^2)$. We keep this in mind when studying the behavior of G, as we can recast few methods to the same coefficients.

6.2.1 Displaying stability

It is stated and numerically shown in [42, 51, 52] that several schemes as the local discontinuous Galerkin scheme [51, 52] and other finite difference schemes combined with an IMEX RK method are stable if the time step is upper bounded by some τ_0. This τ_0 is proportional to $\frac{d}{a^2}$, i.e., if $\Delta t \leq \tau_0 = E_0 \cdot \frac{d}{a^2}$ for some $E_0 > 0$. Considering the before mentioned parameters $\Delta x, \Delta t, a, d$, we introduce two new coefficients

$$C = \frac{a\Delta t}{\Delta x}, \quad D = \frac{d\Delta t}{(\Delta x)^2}. \tag{52}$$

Moreover, using the coefficients C and D reveals an equivalent condition to [42], if we assume that the quotient

$$E := \frac{C^2}{D} = \frac{\Delta t^2 a^2}{\Delta x^2}\frac{\Delta x^2}{d\Delta t} = \frac{a^2}{d}\Delta t$$

is bounded by some constant E_0, indeed,

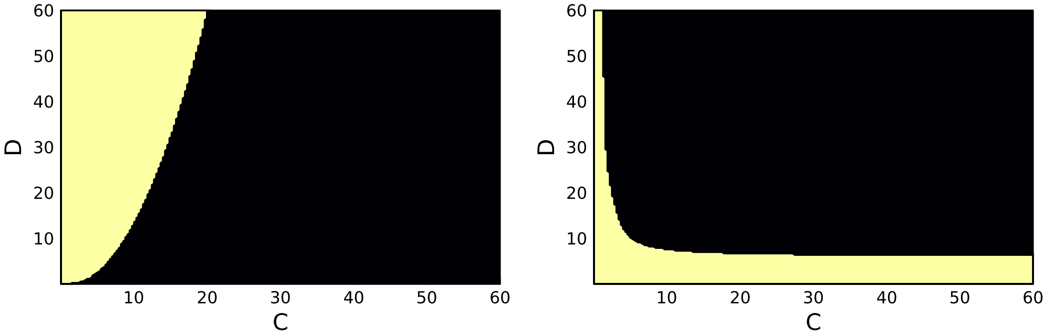
$$E = \frac{a^2}{d}\Delta t \leq E_0 \quad \iff \quad \Delta t \leq E_0 \cdot \frac{d}{a^2} = \tau_0.$$

Therefore, for a given method solving the advection-diffusion equation, we can rewrite the amplification factor in (51) as

$$g(k, C, E) = G(k, \Delta x, \Delta t, a, d). \tag{53}$$

Definition 6.1 (Scheme notation)**.** *To shorten the notation, we denote the considered method for the advection-diffusion equation by* $[\text{TMM}, \text{NODES}, N, A_n, D_m]$*, where*

- TMM *stands for the respective IMEX time-marching method, among* DeC*,* ADER*,* sDeC*,*
- NODES *stands for the used quadrature nodes for the* TMM*, among* EQ *or* GLB*,*
- N *stands for the order of the considered time-marching method* TMM*,*
- A_n *denotes the optimal first derivative stencil of order* n *defined in* (49) *used for the advection term,*
- D_m *denotes the central second derivative stencil of order* m *in Table* 1 *used for the diffusion term.*



Coefficients C and D

Coefficients C and E

Figure 7: Stability areas (yellow) for the $[DeC, eq, 3, A_1, D_2]$.

Example 6.2 (Stability of IMEX DeC3 with A_1 and D_2 operators)**.** *To give an example of how to study the stability region, we show in Figure* 7 *the stability areas* $\{|g(k, C, E)| \leq 1, \forall k \in [-n_0 - 1, n_0 + 1]\}$ *for the* $[\text{DeC}, \text{EQ}, 3, A_1, D_2]$*. On the left, we plot the stability area as a function of* C *and* D*, while on the right as a function of* C *and* E*. The black area is associated to the unstable area, while the yellow displays the stable region. We recognize two sufficient conditions to obtain stability:*

- *the well known CFL-condition, i.e., if* C *is lower than some constant* C_0 *only dependent on the method, then the method is stable.*
- *the new numerically obtained condition, designated as the* E_0*-condition: If* E *is lower than some constant* E_0 *dependent on the method, then the method is stable.*

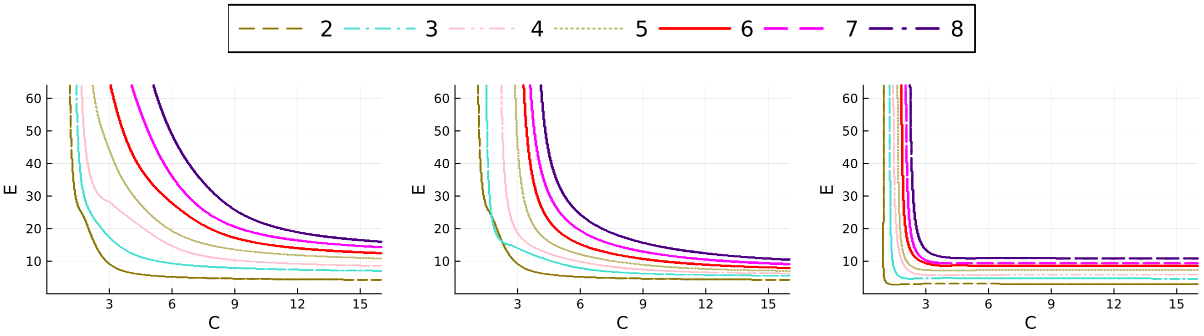
The two parameters C *and* E *include all the remaining ones and are enough to characterize the whole scheme.*

As we can see in Figure 7, the unstable area of this specific method seems to be bounded by the linear constraints $C \geq C_0$ and $E \geq E_0$. We will observe numerically that these unstable regions are indeed similarly bounded in most of our methods.

Definition 6.3 (Stability parameters C_0 and E_0)**.** *Given the amplification factor* $g(k, C, E)$ *of a discretization of the advection-diffusion equation, we define the two stability parameters* C_0 *and* E_0 *by*

- $C_0 := \max_{C \in \mathcal{S}} C$ *with* $\mathcal{S} = \{C : |g(k, C, E)| \leq 1, \forall E > 0, \forall k \in [-n_0 - 1, n_0 + 1]\}$*,*
- $E_0 := \max_{E \in \mathcal{R}} E$ *with* $\mathcal{R} = \{E : |g(k, C, E)| \leq 1, \forall C > 0, \forall k \in [-n_0 - 1, n_0 + 1]\}$.

Therefore, the strategy we want to follow is to look at the areas of stability by evaluating the amplification factor $\max_k |g(k, C, E)|$ like in Figure 7 and numerically calculating the parameters C_0 and E_0 for our methods, when possible.



$[\textsc{DeC}, \mathrm{GLB}, k, A_1, D_2]$ $[\textsc{sDeC}, \mathrm{GLB}, k, A_1, D_2]$ $[\mathrm{ADER}, \mathrm{GLB}, k, A_1, D_2]$

Figure 8: Stability areas for $[\mathrm{TMM}, \mathrm{GLB}, k, A_1, D_2]$ for TMM as IMEX DeC (left), IMEX sDeC (center) and IMEX ADER (right) with eq (top) and GLB (bottom) nodes: TMM order k from 2 to 8

Note, that the condition $E < E_0$ does not depend on Δx and avoids CFL restrictions. We should always keep in mind that our numerical evaluations can just cover finite ranges of C and E. Hence, we checked that the displayed limits for E and C are actually bounds also for larger domains of C and E. Moreover, we also observed that the considered results do not vary much for large values of n_0, hence, we set $n_0 = 10^3$. Due time-efficiency, we will use this value for every evaluation in the von Neumann stability analysis context for the rest of this work. Further, all plots in this section will be evaluated and displayed at 400 × 400 grid points.

6.2.2 Numerical analysis

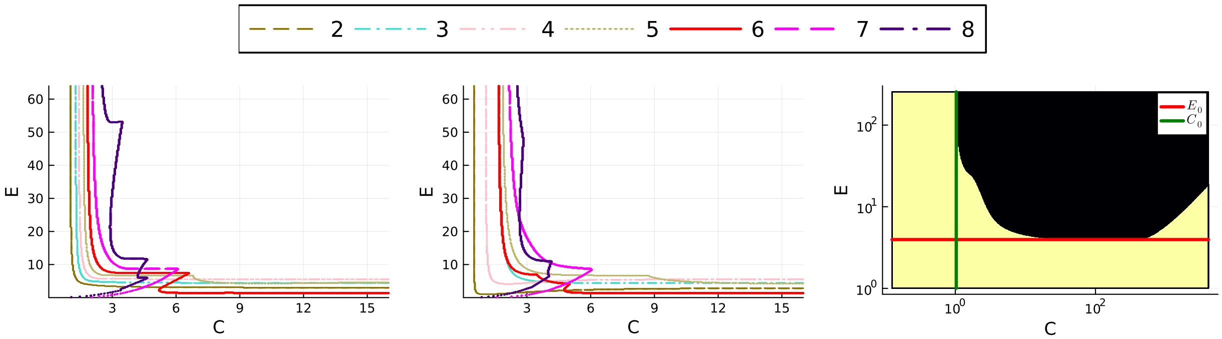
In the following plots, we will study various configurations of methods and the variation of the stability region changing the order of the methods. In particular, we will focus on changing the order of the time discretization only and varying the order of all operators. In the repository [31], we include other variations of the order of only the advection or the diffusion operator that we discard here for brevity. Moreover, we will focus on Gauss–Lobatto nodes for the time discretization only mentioning the peculiarity of the equispaced ones that can be found, again, in the repository [31].

We start by varying only the time scheme order. In Figure 8, the stability areas of several methods are shown, i.e., $[\mathrm{TMM}, \mathrm{NODES}, k, A_1, D_2]$ varying the time scheme and the nodes. As in Figure 7, the plotted lines separates the stable region in the lower left part from the unstable region in the upper right side of the C-E plane. We see a similar behavior for mostly all methods. Increasing the order of the time marching method results in a larger stability region and in larger values for E_0 and C_0.

In the equispaced case we do not observe major differences for the DeC and sDeC methods, while, for the ADER cases, the usage of equispaced nodes results in an irregular reduction to $C_0 = 0$ for some orders (7 and 8), meaning that we can not ensure stability as we did in the other cases for high order methods, see Figure 9 (left). This is probably due to the numerical cancellation of Newton-Cotes quadratures with more than 8 points that occur for orders larger than 6, used in the considered IMEX ADER method.

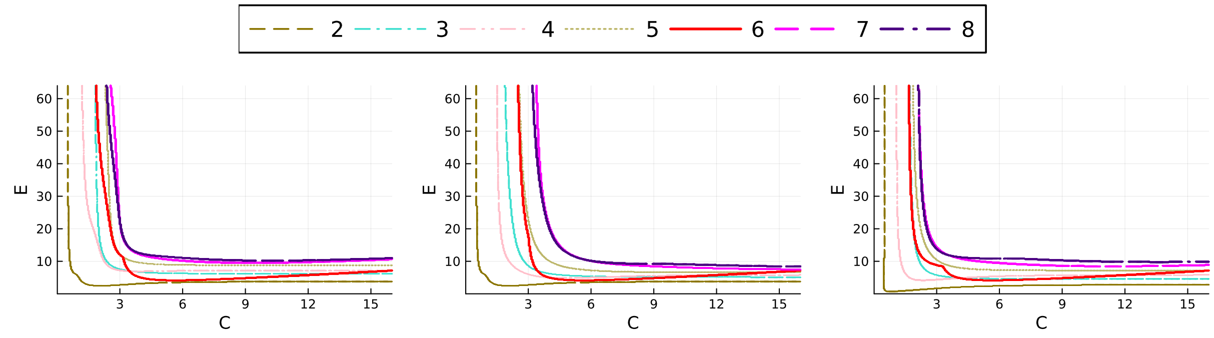
In Figure 10, we study the space–time discretizations matching the orders of the time scheme with the order of the spatial terms, i.e. $[\mathrm{TMM}, \mathrm{GLB}, k, A_k, D_{2\lceil k/2 \rceil}]$, where TMM is one of the 3 considered methods and $k \in \{2, \dots, 8\}$. Here, we can observe for all cases that the higher order terms results in slightly larger stability areas and also in bigger C_0 and E_0, which leads to the behavior we have already seen in Figure 8 varying only the order of the time scheme. Again, for high order IMEX ADER with equispaced nodes we lose the E_0 bound from below as shown in Figure 9.

We can conclude that increasing the order of the time-marching method results in higher values for both C_0 and E_0. However, the considered higher order finite differences for the spatial discretizations do not grant significant improvements. A special mention to high order IMEX ADER with equispaced nodes is necessary also here. The border E_0 vanishes, indeed, for order > 7, see Figure 9 (center). Moreover, also varying the



$[\text{ADER}, \text{EQ}, k, A_1, D_2]$ $[\text{ADER}, \text{EQ}, k, A_k, D_{2\lceil k/2\rceil}]$ $[\text{sDeC}, \text{GLB}, 2, A_1, D_2]$

Figure 9: Stability areas of ADER with equispaced nodes (left and center): varying the order of the time method (left) and varying the order of all discretizations (center). Bounds by C_0 and E_0 of the stability region (in yellow) for an sDeC method (right).



$[\text{DeC}, \text{GLB}, k, A_k, D_{2\lceil k/2\rceil}]$ $[\text{sDeC}, \text{GLB}, k, A_k, D_{2\lceil k/2\rceil}]$ $[\text{ADER}, \text{GLB}, k, A_k, D_{2\lceil k/2\rceil}]$

Figure 10: Stability areas varying the order of all partial methods.

Table 2: Approximated border values C_0 (up to 2 decimals) and E_0 (up to 1 decimal) for Gauss–Lobatto methods with operators with optimal order k

k	$[\text{DeC}, \text{GLB}, k, A_k, D_{2\lceil k/2\rceil}]$		$[\text{sDeC}, \text{GLB}, k, A_k, D_{2\lceil k/2\rceil}]$		$[\text{ADER}, \text{GLB}, k, A_k, D_{2\lceil k/2\rceil}]$	
	C_0	E_0	C_0	E_0	C_0	E_0
2	0.50	2.5	0.50	2.5	0.50	0.7
3	1.63	6.1	1.69	5.1	1.63	4.5
4	1.04	6.9	1.43	4.9	1.04	4.2
5	1.74	8.8	2.31	6.6	1.74	7.2
6	1.60	4.1	2.33	4.2	1.60	4.1
7	1.94	9.5	3.12	7.5	1.94	8.5
8	2.00	10.2	2.85	5.9	2.00	9.8

spatial discretization we obtain the same results. We remark that there are plenty of more options to test, which we do not include in this analysis, some of which can be found in the repository [31]. Nevertheless, we presented all remarkable combinations of methods we tested and a summary of their results.
As an example, we plot the stability region of $[\text{sDeC}, \text{GLB}, 2, A_1, D_2]$ in Figure 9. We observe an asymptotic behavior both for $E \to \infty$ against a line $C = C_0$ and also some sort of border line $E < E_0$ for large values of C, which ensures stability for an arbitrary C, if $E \leq E_0$. These are the desired values for C_0 and E_0. In Table 2, we study the operators with order k matching for the time and spatial discretization for time schemes defined by GLB nodes. We display in that table the maximal values C_0 and E_0. We clearly see that they increase as the order increases. The only value that is not uniform among the methods is E_0 for ADER GLB of order 2, for which we have a restrictive bound. We also want to highlight, that C_0 matches inbetween the DeC and ADER methods for the same orders. This is probably due to their coinciding explicit stability regions for ODEs, as pointed out in [16].

7 PDE: analysis of advection-dispersion

In this section, we extend the analysis and results to observe the behavior of IMEX ADER and DeC methods onto the advection-dispersion equation

$$u_t(x,t) + a u_x(x,t) + \beta u_{xxx}(x,t) = 0, \quad a \geq 0, \ \beta \geq 0. \tag{54}$$

7.1 FD discretization

First, we introduce at this point the considered spatial discretizations for the advection-dispersion equation. Thereby, we consider the same discretization for the advection term, as introduced in Section 6.1.
For the dispersion term, we will consider the upwind scheme used in [42] to test stability for the advection-dispersion equation (54). It is of order 3 and given by

$$\partial^3_{\Delta x}(u(x_j)) = \frac{1}{4\Delta x^3}\left(-w_{j-2} - w_{j-1} + 10 w_j - 14 w_{j+1} + 7 w_{j+2} - w_{j+3}\right). \tag{55}$$

For higher orders, we have used the optimal $2r+1$ order formula on stencils of the type $[-r, r+1]$ with the tool provided in [45].
We have also tested the methods with a central finite difference formula of order 2, always leading to less stable methods, hence, we will not include them in the following discussion.

7.2 von Neumann analysis

As previously done for the advection–diffusion problem, we will perform the von Neumann analysis by looking at the coefficients of the finite difference schemes, i.e.,

$$C = a\frac{\Delta t}{\Delta x}, \qquad P = \beta\frac{\Delta t}{\Delta x^3}.$$

The procedure is analogous to the advection–diffusion one, with C, P instead of C, E.

7.2.1 Displaying stability

To denote the considered methods, we use again the notation introduced for the advection-diffusion equation

$$[\text{TMM}, \text{NODES}, N, A_n, B_m],$$

where B_m refers to the upwind m-th order stencils of type $[-r, r+1]$. We proceed evaluating the amplification factor

$$G(k, \Delta x, \Delta t, a, \beta) = g(k, C, P)$$

to observe the stability region as a function of C and P. In opposition to the advection–diffusion case, in [42] only a CFL condition is found, even if, numerically, they observe larger stability regions with a little of dispersion. We want to give a more comprehensive study of this behavior for different schemes and, as before, we look for meaningful coefficients that bounded by some constants give the stability. To find such coefficients, we proceed with an example.

Example 7.1. *In Figure 11, we display the stability areas for the* $[\text{DeC}, \text{GLB}, 2, A_1, B_3]$ *and* $[\text{DeC}, \text{GLB}, 3, A_1, B_3]$ *on the* (C, P) *plane (left). In the IMEXDeC2 case, we note that for low* P *a CFL constraint* $C \leq 1$ *guarantees stability, while for large* P *we see a linear constraint of the type* $P \gtrsim E_0 C$*. In the IMEXDeC3 case, there is a further unstable region close to the* $C = 0$ *axis. This extra unstable region is due to the fact that IMDeC3 is not A-stable and, hence, for low values of* C *not enough numerical dissipation is brought to the system.*
Anyway, the linear constraint on the large P *motivates the following definition of*

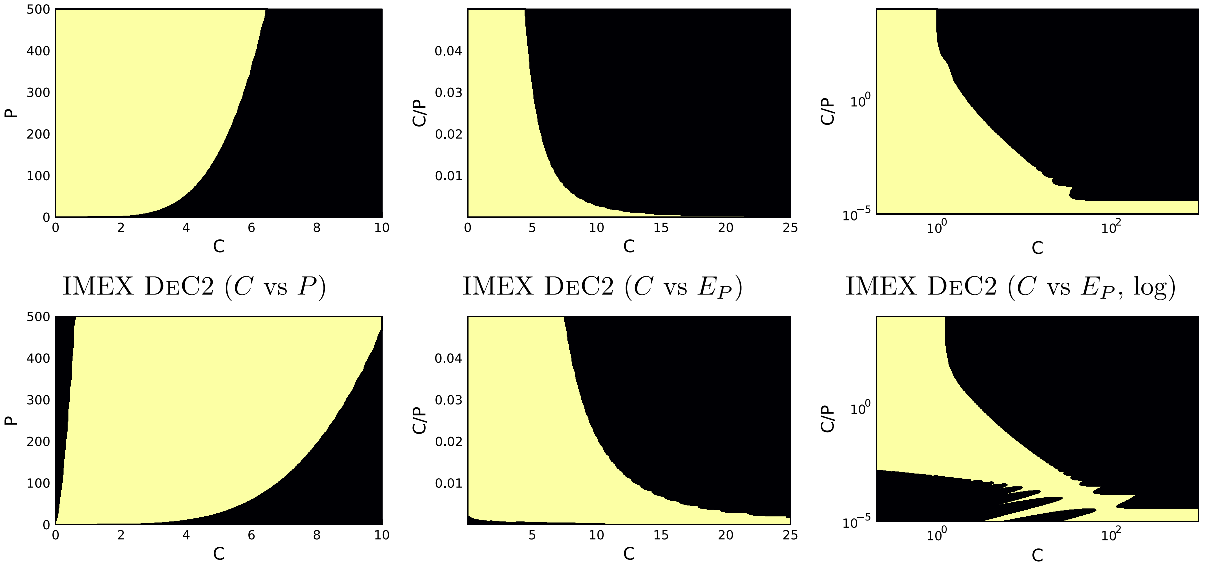
$$E_P := \frac{C}{P} = \frac{\Delta t a}{\Delta x}\frac{\Delta x^3}{\beta \Delta t} = \frac{a\Delta x^2}{\beta}.$$

Now, looking at the right plot for IMEXDeC2 in Figure 11, we observe that either $C \leq 1$ *or* $E_P \leq E_0 \approx 10^{-4}$ *guarantee stability. This is a peculiar result as* $E_P = \frac{a\Delta x^2}{\beta}$ *does not depend on the time discretization. The same does not hold for IMEXDeC3, where this area is stable only for large values of* C*, which leads to ridiculously small* Δx *and large* Δt*.*

These exemplary stability regions hold for most of the considered cases, i.e. all methods of order 2 do not have the instability areas for small C and large P, as well as the IMEX ADER methods with equispaced nodes until order 4 and all IMEX ADER methods with Gauss-Lobatto nodes. Remark that these are exactly the methods which seem to be A-stable in their implicit ODE application as discussed in section 5. All remaining methods possess this unfavorable stability region.

7.2.2 Results for IMEX DeC, sDeC and ADER

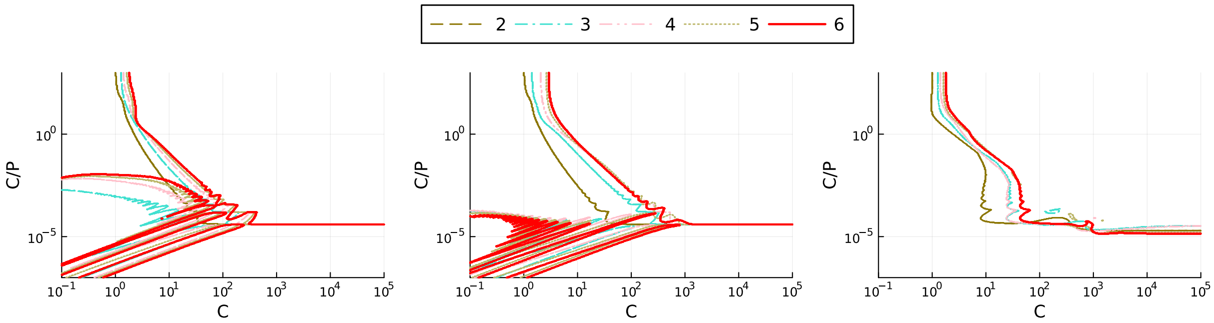
In this section, we present the analysis results as displayed in Example 7.1, varying numerical methods. We proceed now studying the stability regions increasing the order of the time scheme only, keeping fixed the advection and dispersion operators (A_1 and B_3), later on we also increase the accuracy of the advection and dispersion operators. In Figure 12, we can observe the stability regions changing the time scheme order from 2 to 6 for GLB nodes. For DeC methods of order larger than 2, we cannot not provide bounds that



IMEX DeC2 (C vs P) IMEX DeC2 (C vs E_P) IMEX DeC2 (C vs E_P, log)

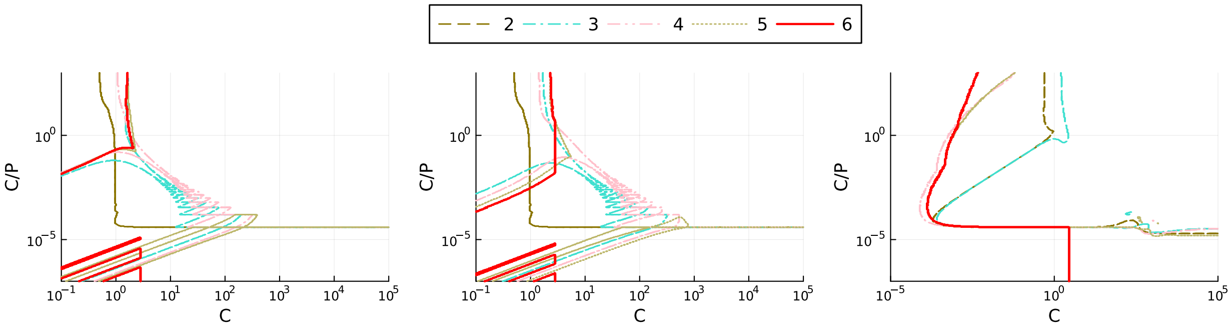
IMEX DeC3 (C vs P) IMEX DeC3 (C vs E_P) IMEX DeC3 (C vs E_P, log)

Figure 11: Stability areas for [DeC, EQ, 2, A_1, B_3] and [DeC, EQ, 3, A_1, B_3] with third order upwind dispersion operator



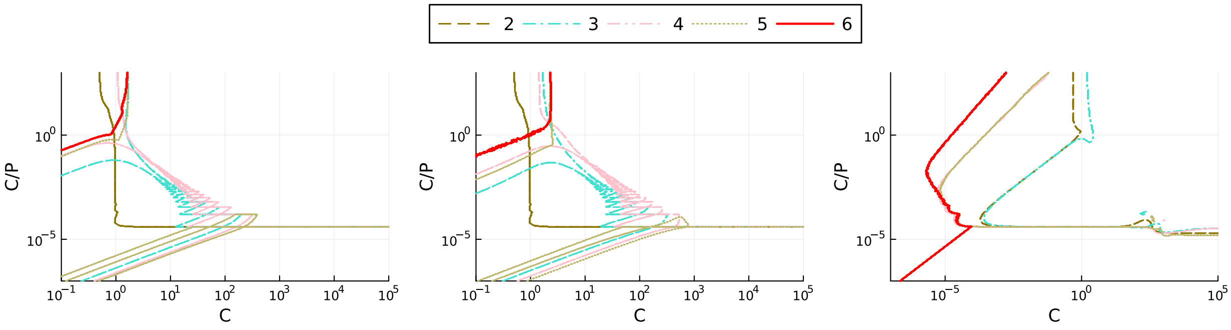
[DeC, GLB, k, A_1, B_3] [sDeC, GLB, k, A_1, B_3] [ADER, GLB, k, A_1, B_3]

Figure 12: Stability areas for orders 2 to 6 with GLB nodes, the upwind scheme of (55) for the dispersion and an first order backward scheme for the advection term



$[\text{DeC}, \text{GLB}, k, A_k, B_3]$ $[\text{sDeC}, \text{GLB}, k, A_k, B_3]$ $[\text{ADER}, \text{GLB}, k, A_k, B_3]$

Figure 13: Stability areas varying orders 2 to 6 of the advection scheme and time scheme.



$[\text{DeC}, \text{GLB}, k, A_k, B_{\lceil k/2 \rceil}]$ $[\text{sDeC}, \text{GLB}, k, A_k, B_{\lceil k/2 \rceil}]$ $[\text{ADER}, \text{GLB}, k, A_k, B_{\lceil k/2 \rceil}]$

Figure 14: Stability areas for orders 2 to 6 with GLB (top) and equi (bottom) nodes, dispersion with stencil $[-\lceil (k+2)/2 \rceil + 1, \lceil (k+2)/2 \rceil]$, advection of order k as in (48) and time scheme of order k. Notice the difference in scales for C between ADER and DeC.

guarantee the stability for small C and E_P. Anyway, away from this area, we observe stable regions for both $C \leq C_0$ with C_0 values similar to the ones of the advection–diffusion section, see Table 2, and for $E_P \leq E_{P,0}$ with $E_{P,0} \approx 4 \cdot 10^{-5}$ independently on the DeC method used. In general, sDeC guarantees more stability in the region with $C \in [1, 10]$ and $C/P \in [10^{-4}, 10^{-2}]$. The differences between equispaced and GLB are not so relevant.

On the other hand, IMEXADER with GLB nodes is very stable and there are clear bounds $C \leq C_0 \leq 3$ and $E_P \leq E_{P,0} \leq 10^{-4}$ that guarantee stability. Moreover, there is a large stability area for large $C \leq 10$ and not so small E_P. On the contrary, IMEXADER with equispaced points (figure available in the repository [31]) for order more than 4 is much more unstable and only the are with both $C \leq C_0$ and $E_P \geq E_{P,0}$, which quite restrictive.

Again, the behavior of all these schemes reflects the A-stability property of the corresponding implicit methods.

In Figure 13, we check the stability regions varying the advection and time order of accuracy for DeC, sDeC and ADER GLB methods. We find a loss of stability by increasing the order. We already see a slight reduction of our border C_0 for order 2 and 3. Going onto orders 4, 5 and 6, we obtain way larger unstable regions, in particular in the low E_P region that was stable in the previous test. For IMEXADER, we observe a strong reduction of the stability region in the low C high E_P region (observe the different scale) as well as in low E_P values for order 6.

In Figure 14, we increase all together the order of all operators. In particular, for a given order k for the dispersion operator we use the optimal stencil with support $[-\lceil (k+2)/2 \rceil + 1, \lceil (k+2)/2 \rceil]$, similar to the upwinding of (55). To compute the coefficients of the dispersion operator, we have used the tool [45]. We are working with the dispersion stencil of order 3, 5 and 7. The Fourier symbol of the stencils of order 5 and 7 take values very close to the imaginary axis also for quite large imaginary values. This means that schemes that are not A-stable will poorly perform on such higher orders. On the other hand, the dispersion operator of order 3 is only tangent to the imaginary axis, but it quickly has real values away from zero for large imaginary values. This will influence the stability regions.

We immediately see that the stability regions shrink and for high order DeC (greater than 5), we lose the stability region $E_P \leq E_{P,0}$. For ADER methods again the region with $C \leq C_0$ and moderate E_P shrinks quite a lot and for order 6 the stability region $E_P \leq E_{P,0}$ essentially disappears. For the equispaced case, as for the time only case, from order 5 on there is no stable region for low E_P [31].

We conclude that the observed IMEX methods combined with the finite difference stencils for the spatial discretization do not possess a spatial-independent condition on the time step (as for the diffusion case). Still, in most of the methods a classical stability region for $C \leq C_0$ and $E_P \geq E_{P,1}$, i.e., $P \geq C/E_{P,1}$, is observable, while a time independent stability region for $E_P \leq E_{P,0}$ is present only in few low order cases and it is really linked to the used spatial discretization.

8 Numerical tests

8.1 ODE tests

In this section, we will apply the introduced explicit, implicit and IMEX methods on ordinary differential equations to compare them with the theoretical results obtained before. Remark that our implementation takes usage of the linearized versions as in (43). This is exact for linear systems, but has to be kept in mind while considering nonlinear ODEs.

8.1.1 Validation of the ImsDeC stability region

As seen in Figures 1 and 2, the ImsDeC has an unexpected behavior by being the only considered pure implicit method here, which is not *almost A-stable* for high orders. We want to validate this statement applying the method on some ODEs.

As example we take the ImsDeC11 with Gauss-Lobatto nodes, because it has the smallest stability region out of all ImsDeC methods. We observe that the limited, stable region has its left border approximately at $z = 900 + 0i$.

Solving at first the linear, scalar ODE

$$\partial_t y(t) = -10^3 y(t), y(0) = 1, \tag{56}$$

with different step sizes $h_1 = 1.0, h_2 = 0.5, h_3 = 0.02, h_4 = 0.01$, we expect the ImsDeC method for h_1 to be unstable and for the remaining 3 h_i to be stable, because $z_1 = \lambda \cdot h_1 = -10^3 \cdot 1 = -10^3 \notin S$, while $|z_i| = |\lambda h_i| < 900$ and belong to S for $i = 2, 3, 4$, where S denotes the stability region of the ImsDeC11 with Gauss-Lobatto nodes. The numerical results are shown in figure 15. We observe that for h_1, the method diverges from the decaying exact solution $y(t) = e^{-10^3 t}$, which is in agreement with $z_1 \notin S$. In the second case for h_2, we see the solution curve mimicking the decaying behavior, which validate the fact that $z_2 \in S$. As a comparison, we also plot the ImDeC11 with Gauss-Lobatto nodes which is known as almost A-stable and shows appropriate results. As seen in figure 15, of course for both z_3 and z_4, we obtain qualitative good solutions. We also display the sDeC11 in these figures and observe that their stability properties are still much worse than these of the ImsDeC11, as its stability region ends around $z \approx 6$, which make $h_c = \frac{6}{10^3}$ the threshold for a stable discretization.

We can summarize that the numerically calculated stability regions are validated through this experiment.

Moving to a nonlinear example, we test the ImsDeC11 with Gauss-Lobatto nodes for the nonlinear ODE

$$\partial_t y(t) = -10^6 |y(t)| \cdot y(t) + 1, \quad y(0) = \frac{1}{\sqrt{10^6}} \tag{57}$$

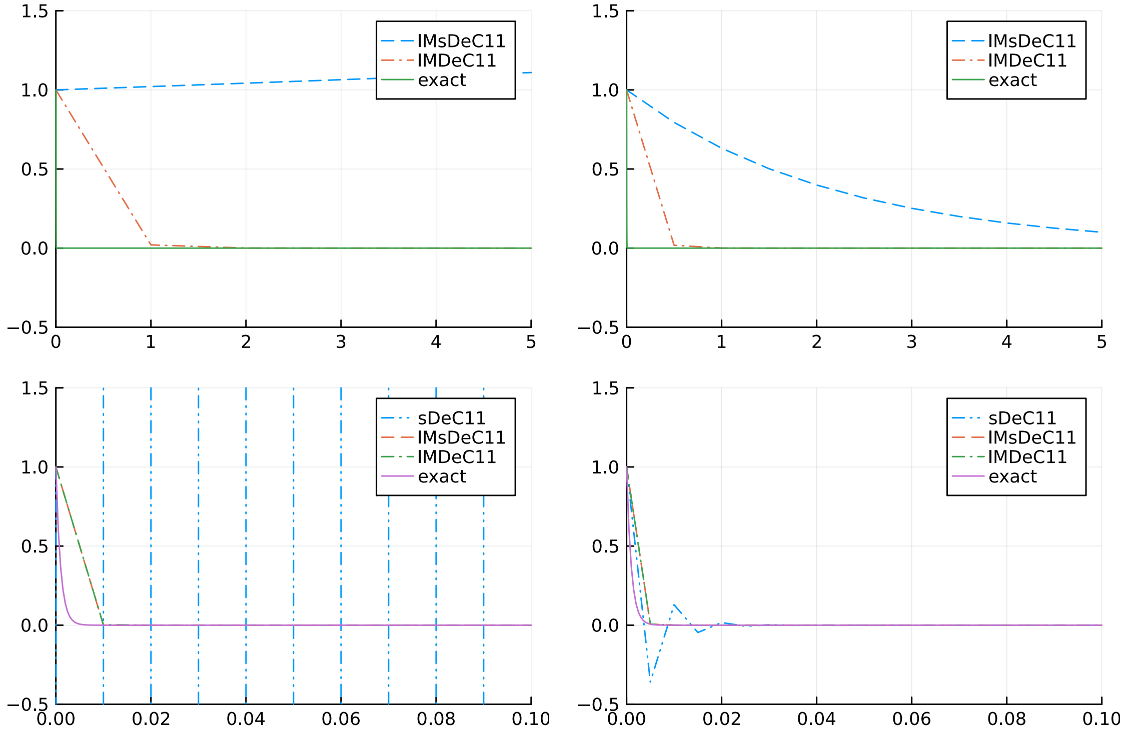


Figure 15: Solving (56) using ImsDeC11 with $h = 1$ (top left), $h = 0.5$ (top right), $h = 0.01$ (bottom left) and $h = 0.005$ (bottom right).

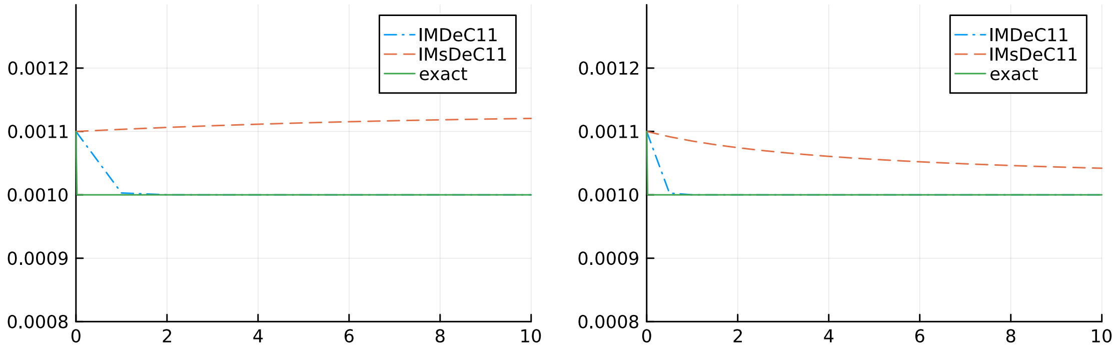


Figure 16: Solving equation (57) using ImsDeC11 with $h = 1.0$ (left) and $h = 0.5$ (right).

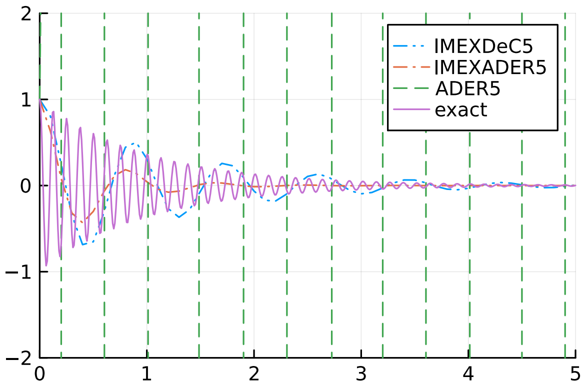


Figure 17: Solving equation (59) with different methods using equispaced nodes with $h = 0.1$.

and compare it again with the analogue ImDeC. We observe in Figure 16 on the left that the ImDeC handles the stiff equation well, while the ImsDeC again has an unstable behavior for the step $h_1 = 1$. This coincides with the stability region because for our nonlinear equation, an equivalent to λ could be roughly obtained by linearizing the equation, so that $\lambda := -10^6|y(t)| \approx -10^6 \cdot \frac{1}{\sqrt{10^6}} = -10^3$ and therefore $z_1 \approx -10^3 \notin S$. Nevertheless, after refining to $h_2 = 0.5$ we have $z_2 \approx -500$ and we expect again stability for the ImsDeC, which can be seen on the right of Figure 16.
Summarizing we can observe that the non-A-stable ImsDeC performs worse than the ImDeC for stiff applications, but may still be applicable to mildly stiff problems providing better results than explicit sDeC.

8.1.2 A stiff linear example for the IMEX methods

We consider the second order ODE

$$\partial_{tt}y(t) = -2\partial_t y(t) - 2501y(t), \quad y(0) = 1,\ \partial_y(0) = 0, \tag{58}$$

which can be rewritten as a system of ODEs

$$\begin{aligned} \partial_t u_1(t) &= -2u_1(t) - 2501u_2(t), \qquad && u_1(0) = 1, \\ \partial_t u_2(t) &= u_1(t), && u_2(0) = 0. \end{aligned} \tag{59}$$

The exact solution is given by

$$\underline{u}(t) = \begin{pmatrix} y(t) \\ \partial_t y(t) \end{pmatrix} = \begin{pmatrix} \frac{1}{50}e^{-t}\left(\sin(50t) + 50\cos(50t)\right) \\ -\frac{2501}{50}e^{-t}\sin(50t) \end{pmatrix},$$

i.e., it shows a rapid oscillation and a slow transient part. Therefore, we separate the right-hand side of (59) by

$$F_I(\underline{u}(t)) = \begin{pmatrix} -2501u_2(t) \\ u_1(t) \end{pmatrix}, \quad F_E(\underline{u}(t)) = \begin{pmatrix} -2u_1(t) \\ 0 \end{pmatrix}$$

to treat F_I implicitly and F_E explicitly.
The numerical solutions for IMEX ADER, IMEX DeC and to comparison ADER are shown in Figure 17, all methods of order 5 and use equispaced nodes. As before, we observe stability for the IMEX methods while the explicit ADER does not converge towards the exact solution. Moreover, it is remarkable that the IMEX methods are able to catch the slow transient part even when the discretization scale does not allow to represent the fast oscillatory behavior.

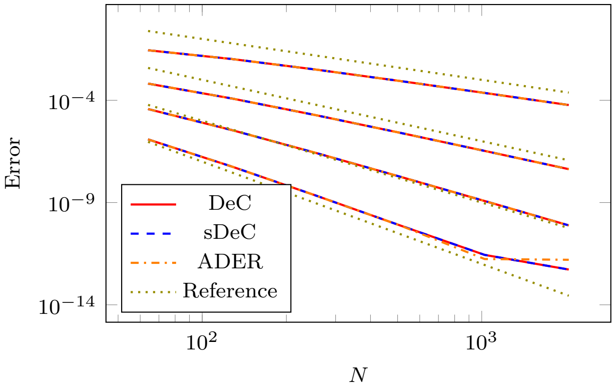


Figure 18: Error convergence on an advection diffusion problem for DeC, sDeC, ADER and dotted reference error line N^{-K}, for orders from $K = 2$ to $K = 5$ (from top to bottom).

8.2 PDE tests

Finally, we assess the accuracy of the proposed schemes on the advection–diffusion equation (47). We consider the domain $\Omega = [0, 2\pi]$, $u_0(x) = \sin(x)$, $a = 1$ and $C = \frac{a\Delta t}{\Delta x} = 0.4$. We fix for all simulations $E = 0.5$, leading to a variable $d = \frac{a^2\Delta t}{E}$. We test ADER, DeC and sDeC with GLB nodes for variable orders from 2 to 5 and meshes with sizes from $N = 2^5$ to $N = 2^{11}$. The spatial operators are generated using [37]. For all methods we obtain stable simulations. In Figure 18, we observe how all methods converge to the exact solution $u(t, x) = e^{-dt} \sin(x - at)$ with the expected order of accuracy.

9 Conclusions

In our study, we have analyzed the implicit and implicit-explicit ADER and DeC methods in terms of their stability properties. To this end, we initially reformulated them as RK methods and investigated them based on the selected order, method, and quadrature nodes. Unlike our prior work [16], which focused on explicit versions, we observed significant variations in stability behavior, ranging from A-stable to bounded stability regions. In general, the implicit (and implicit-explicit) ADER methodology demonstrated greater stability compared to the DeC framework.

After the ODE case, we further extended our analysis to the PDE case, focusing on advection-diffusion and advection-dispersion equations inspired by previous works [42, 51]. For space discretization, we utilized up-to-date finite difference stencils and derived CFL-like stability conditions through von Neumann stability analysis. Notably, we expanded upon the investigation of [42] by introducing two new auxiliary coefficients for the advection-diffusion equation. These coefficients yielded equivalent conditions to those in [42], but they do not depend on the spatial discretization. We established precise boundaries for relevant coefficients for advection-diffusion and advection-dispersion and offered recommendations regarding the suitability of specific schemes.

In the future, further potential research directions include exploring different space discretization methods and focusing on continuous and discontinuous Galerkin formulations [35, 34]. Additionally, investigating stability in the context of nonlinear problems would be desirable, particularly focusing on the entropy production of such schemes as suggested by previous works [23, 22, 30], or using the add-and-subtract version of the implicit ADER and DeC to study the stability in the same style of [43].

Acknowledgements

P.Ö. is supported by the DFG within SPP 2410, project 525866748 (OE 661/5-1) and under the personal grant 520756621 (OE 661/4-1). P.Ö. gratefully acknowledges the support of the Gutenberg Research College, JGU Mainz. D. T. is funded by a SISSA Mathematical Fellowship. L. P. is supported by the DFG within SPP 2410, project number 526031774 and the Daimler und Benz Stiftung (Daimler and Benz foundation), project number 32-10/22.
D. T. and P. Ö. wish to express their gratitude to prof. Giovanni Russo, who brought to their attention various seminal works on the stability of IMEX schemes. Engaging in discussions with him has deepened our comprehension in this domain.

References

[1] R. Abgrall. High order schemes for hyperbolic problems using globally continuous approximation and avoiding mass matrices. *Journal of Scientific Computing*, 73(2):461–494, Dec 2017.

[2] R. Abgrall, É. Le Mélédo, P. Öffner, and D. Torlo. Relaxation deferred correction methods and their applications to residual distribution schemes. *SMAI J. Comput. Math.*, 8:125–160, 2022.

[3] R. Abgrall and D. Torlo. High order asymptotic preserving deferred correction implicit-explicit schemes for kinetic models. *SIAM Journal on Scientific Computing*, 42(3):B816–B845, Jan. 2020.

[4] S. Boscarino, J.-M. Qiu, and G. Russo. Implicit-explicit integral deferred correction methods for stiff problems. *SIAM J. Sci. Comput.*, 40(2):a787–a816, 2018.

[5] W. Boscheri and M. Dumbser. A direct Arbitrary-Lagrangian–Eulerian ADER-WENO finite volume scheme on unstructured tetrahedral meshes for conservative and non-conservative hyperbolic systems in 3D. *Journal of Computational Physics*, 275:484–523, 2014.

[6] R. E. Caflisch, S. Jin, and G. Russo. Uniformly accurate schemes for hyperbolic systems with relaxation. *SIAM Journal on Numerical Analysis*, 34(1):246–281, 1997.

[7] A. Christlieb, B. Ong, and J.-M. Qiu. Integral deferred correction methods constructed with high order Runge-Kutta integrators. *Math. Comput.*, 79(270):761–783, 2010.

[8] R. Dematté, V. Titarev, G. Montecinos, and E. Toro. ADER methods for hyperbolic equations with a time-reconstruction solver for the generalized Riemann problem: the scalar case. *Communications on Applied Mathematics and Computation*, 2:369–402, 2020.

[9] M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz. A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes. *Journal of Computational Physics*, 227(18):8209–8253, Sept. 2008.

[10] M. Dumbser, C. Enaux, and E. F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *J. Comput. Phys.*, 227(8):3971–4001, Apr. 2008.

[11] A. Dutt, L. Greengard, and V. Rokhlin. Spectral Deferred Correction Methods for Ordinary Differential Equations. *BIT Numerical Mathematics*, 40(2):241–266, 2000.

[12] B. Fornberg. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of Computation*, 51(184):699–706, 1988.

[13] E. Gaburro, P. Öffner, M. Ricchiuto, and D. Torlo. High order entropy preserving ADER-DG schemes. *Appl. Math. Comput.*, 440:21, 2023.

[14] E. Hairer, S. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer-Verlag, Berlin, 1987.

[15] M. Han Veiga, L. Micalizzi, and D. Torlo. On improving the efficiency of ADER methods. *Applied Mathematics and Computation*, 466:128426, 2024.

[16] M. Han Veiga, P. Öffner, and D. Torlo. DeC and ADER: similarities, differences and a unified framework. *Journal of Scientific Computing*, 87(1), Feb. 2021.

[17] W. Hundsdorfer and J. Verwer. *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*. Springer Berlin Heidelberg, 2003.

[18] A. Iserles. Order stars and a saturation theorem for first-order hyperbolics. *IMA Journal of Numerical Analysis*, 2(1):49–61, 1982.

[19] A. T. Layton and M. L. Minion. Conservative multi-implicit spectral deferred correction methods for reacting gas dynamics. *Journal of Computational Physics*, 194(2):697–715, 2004.

[20] R. J. LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM, 2007.

[21] S. F. Liotta, V. Romano, and G. Russo. Central schemes for balance laws of relaxation type. *SIAM Journal on Numerical Analysis*, 38(4):1337–1356, 2000.

[22] C. Lozano. Entropy production by explicit Runge-Kutta schemes. *J. Sci. Comput.*, 76(1):521–564, 2018.

[23] C. Lozano. Entropy production by implicit Runge-Kutta schemes. *J. Sci. Comput.*, 79(3):1832–1853, 2019.

[24] L. Micalizzi and D. Torlo. A new efficient explicit deferred correction framework: Analysis and applications to hyperbolic PDEs and adaptivity. *Communications on Applied Mathematics and Computation*, 2023.

[25] L. Micalizzi, D. Torlo, and W. Boscheri. Efficient iterative arbitrary high-order methods: an adaptive bridge between low and high order. *Communications on Applied Mathematics and Computation*, pages 1–38, 2023.

[26] S. Michel, D. Torlo, M. Ricchiuto, and R. Abgrall. Spectral analysis of continuous FEM for hyperbolic PDEs: influence of approximation, stabilization, and time-stepping. *Journal of Scientific Computing*, 89:1–41, 2021.

[27] S. Michel, D. Torlo, M. Ricchiuto, and R. Abgrall. Spectral analysis of high order continuous FEM for hyperbolic PDEs on triangular meshes: influence of approximation, stabilization, and time-stepping. *Journal of Scientific Computing*, 94(3):49, 2023.

[28] M. L. Minion. Semi-implicit spectral deferred correction methods for ordinary differential equations. *Commun. Math. Sci.*, 1(3):471–500, 09 2003.

[29] P. Öffner. *Approximation and stability properties of numerical methods for hyperbolic conservation laws*. Springer Nature, 2023.

[30] P. Öffner, J. Glaubitz, and H. Ranocha. Analysis of artificial dissipation of explicit and implicit time-integration methods. *Int. J. Numer. Anal. Model.*, 17(3):332–349, 2020.

[31] P. Öffner, L. Petri, and D. Torlo. IMEX_DeC_ADER github repository. https://github.com/accdavlo/IMEX_DeC_ADER.git, 2024.

[32] P. Öffner and D. Torlo. Arbitrary high-order, conservative and positivity preserving Patankar-type deferred correction schemes. *Applied Numerical Mathematics*, 153:15–34, July 2020.

[33] B. W. Ong and R. J. Spiteri. Deferred correction methods for ordinary differential equations. *J. Sci. Comput.*, 83(3):29, 2020.

[34] S. Ortleb. L^2-stability analysis of IMEX-(σ, μ) DG schemes for linear advection-diffusion equations. *Appl. Numer. Math.*, 147:43–65, 2020.

[35] S. Ortleb. On the stability of IMEX Upwind gSBP schemes for 1D linear advection-diffusion equations. *Communications on Applied Mathematics and Computation*, pages 1–30, 2023.

[36] L. Pareschi and G. Russo. Implicit-explicit Runge-Kutta schemes for stiff systems of differential equations. *Recent trends in numerical analysis*, 3:269–289, 2000.

[37] H. Ranocha. SummationByPartsOperators.jl: A Julia library of provably stable semidiscretization techniques with mimetic properties. *Journal of Open Source Software*, 6(64):3454, 08 2021.

[38] R. Scherer. A necessary condition for B-stability. *BIT Numerical Mathematics*, 19:111–115, 1979.

[39] T. Schwartzkopff, C. D. Munz, and E. F. Toro. ADER: A high-order approach for linear hyperbolic systems in 2d. *J. Sci. Comput.*, 17(1–4):231–240, dec 2002.

[40] R. Speck, D. Ruprecht, M. Emmett, M. Minion, M. Bolten, and R. Krause. A multi-level spectral deferred correction method. *BIT Numerical Mathematics*, 55(3):843–867, 2015.

[41] H. J. Stetter et al. *Analysis of discretization methods for ordinary differential equations*, volume 23. Springer, 1973.

[42] M. Tan, J. Cheng, and C.-W. Shu. Stability of high order finite difference schemes with implicit-explicit time-marching for convection-diffusion and convection-dispersion equations. *Int. J. Numer. Anal. Model.*, 18(3):362–383, 2021.

[43] M. Tan, J. Cheng, and C.-W. Shu. Stability of high order finite difference and local discontinuous Galerkin schemes with explicit-implicit-null time-marching for high order dissipative and dispersive equations. *Journal of Computational Physics*, 464:111314, 2022.

[44] T. Tang, H. Xie, and X. Yin. High-order convergence of spectral deferred correction methods on general quadrature nodes. *J. Sci. Comput.*, 56(1):1–13, 2013.

[45] C. R. Taylor. Finite difference coefficients calculator. https://web.media.mit.edu/~crtaylor/calculator.html, 2016.

[46] V. Titarev and E. F. Toro. Analysis of ADER and ADER-WAF schemes. *IMA journal of numerical analysis*, 27(3):616–630, 2007.

[47] V. A. Titarev and E. F. Toro. ADER: Arbitrary high order Godunov approach. *Journal of Scientific Computing*, 17(1-4):609–618, 2002.

[48] D. Torlo. *Hyperbolic problems: high order methods and model order reduction*. PhD thesis, Universität Zürich, 2020.

[49] E. F. Toro, R. C. Millington, and L. A. M. Nejad. *Towards Very High Order Godunov Schemes*, page 907–940. Springer US, 2001.

[50] C. F. Van Loan and N. Pitsianis. *Approximation with Kronecker products*. Springer, 1993.

[51] H. Wang, C.-W. Shu, and Q. Zhang. Stability and error estimates of local discontinuous Galerkin methods with implicit-explicit time-marching for advection-diffusion problems. *SIAM Journal on Numerical Analysis*, 53(1):206–227, Jan. 2015.

[52] H. Wang, C.-W. Shu, and Q. Zhang. Stability analysis and error estimates of local discontinuous Galerkin methods with implicit–explicit time-marching for nonlinear convection–diffusion problems. *Applied Mathematics and Computation*, 272:237–258, Jan. 2016.

[53] G. Wanner and E. Hairer. *Solving ordinary differential equations II: Stiff and Differential-Algebraic Problems*, volume 375. Springer Berlin Heidelberg, Berlin, 1996.

[54] X. Zhong. Additive semi-implicit Runge–Kutta methods for computing high-speed nonequilibrium reactive flows. *Journal of Computational Physics*, 128(1):19–31, 1996.