



HAL
open science

High order semi-implicit schemes for time dependent partial differential equations

Sebastiano Boscarino, Francis Filbet, Giovanni Russo

► **To cite this version:**

Sebastiano Boscarino, Francis Filbet, Giovanni Russo. High order semi-implicit schemes for time dependent partial differential equations. 2014. hal-00983924v2

HAL Id: hal-00983924

<https://hal.science/hal-00983924v2>

Preprint submitted on 20 Mar 2015 (v2), last revised 8 Jan 2016 (v5)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

HIGH ORDER SEMI-IMPLICIT SCHEMES FOR TIME DEPENDENT PARTIAL DIFFERENTIAL EQUATIONS

SEBASTIANO BOSCARINO, FRANCIS FILBET, AND GIOVANNI RUSSO

ABSTRACT. In this paper we consider a new formulation of implicit-explicit (IMEX) Runge-Kutta (R-K) methods for the numerical discretization of time dependent partial differential equations. The approach is based on identifying the (linear) dependence on the unknown of the system which generates the stiffness. Only the stiff dependence is treated implicitly, then making the whole method much simpler than fully implicit ones. This approach generalizes classical IMEX methods based on additive and partitioned R-K, and allows a novel application of semi-implicit schemes. We adopt several semi-implicit R-K methods up to order three. We illustrate the effectiveness of the new approach with many applications to reaction-diffusion, convection diffusion and nonlinear diffusion system of equations. Finally, we conclude by a stability analysis of the schemes for linear problems.

CONTENTS

1. Introduction	1
2. Numerical methods for ODEs	5
2.1. From Partitioned to semi-implicit Runge-Kutta methods.	5
2.2. Classification of IMEX Runge-Kutta schemes.	9
2.3. Order conditions and numerical schemes	9
3. Applications	12
3.1. Test 1 - Reaction-diffusion problem	13
3.2. Test 2 - Nonlinear convection-diffusion equation	14
3.3. Test 3 - Nonlinear Fokker-Planck equations for fermions and bosons	15
3.4. Test 4 - Hele-Shaw flow	16
3.5. Test 5 - Surface diffusion flow	19
4. Linear Stability analysis	20
5. Acknowledgements	25
References	25

1. INTRODUCTION

A well-known approach in the numerical solution of evolutionary problems in partial differential equations is the method of lines. In this approach a partial differential equation is first discretized in space by finite difference or finite element techniques and converted into a system of ordinary differential equations (ODEs). In some cases the right hand side can be written as the sum of two terms, a stiff one and a non stiff one:

$$\left\{ \begin{array}{l} \frac{du}{dt}(t) = F(t, u(t)) + \frac{1}{\varepsilon} G(t, u(t)), \quad \forall t \geq t_0, \\ u(t_0) = u_0, \end{array} \right.$$

2010 *Mathematics Subject Classification.* Primary: 82C40, Secondary: 65N08, 65N35 .

Key words and phrases. IMEX Schemes, Stiff Problems, Time dependant Partial Differential Equations.

where ε is a small parameter, which generates some stiffness in the system. We call such stiff problem of *additive type* and hereafter denoted by (\mathcal{A}) .

The development of numerical schemes for systems of stiff ODEs of the form (\mathcal{A}) attracted a lot of attention in the last decades. Systems of such form often arise from the discretization of partial differential equations, such as convection-diffusion equations and hyperbolic systems with relaxation. In previous works we considered the latter case which in recent years has been a very active field of research, due to its great impact on applied sciences. In fact, relaxation is important in many physical situations, for example it arises in discrete kinetic theory of rarefied gases, hydrodynamical models for semiconductors, linear and non-linear waves, viscoelasticity, traffic flows, shallow water [15, 16, 20, 31, 32, 33, 28].

Hopefully, when a problem with easily separable stiff and non-stiff components is considered, a combination of implicit and explicit Runge-Kutta methods can be used. The implicit method is used to treat the stiff component $G(t, u(t))/\varepsilon$ in a stable fashion while the non-stiff component $F(t, u(t))$ of the system is treated using the explicit scheme. These combined implicit/explicit (IMEX) schemes are already used for several problems, including convection-diffusion-reaction systems, hyperbolic systems with relaxation, collisional kinetic equations, and so on.

However it is not always easy to separate stiff and non-stiff components, and therefore the use of standard IMEX schemes is not straightforward. In such cases one usually relies on fully implicit schemes or in some linearized version of them, such as Rosenbrock schemes, [26]. The latter are general purpose semi-implicit schemes, that do not make use of the particular structure of the system.

In many cases of interest, it is possible to adopt different semi-implicit schemes, which exploit the structure of the system, resulting in a very effective tool, being a good compromise among accuracy, stability and robustness. For instance in [7], the authors consider nonlinear hyperbolic systems containing fully nonlinear and stiff relaxation terms in the limit of arbitrary late times. The dynamics is asymptotically governed by effective systems which are of parabolic type and may contain degenerate and/or fully nonlinear diffusion terms. Fully nonlinear relaxation terms can arise, for instance, in presence of *strong friction*, see for example in [3] and references therein. Furthermore, a general class of models of the same type were introduced by Kawashima and LeFloch (see for example [7]). For such problems in [7], the authors introduced a semi-implicit formulation based on implicit-explicit (IMEX) Runge-Kutta methods. Similarly in [35], the author introduced a semi-implicit method for computing the two models of motion by mean curvature and motion by surface diffusion which is stable for large time steps. In all such models a semi-implicit method is more effective than a fully implicit one.

In many cases the stiffness is associated to some variables. For example, if a system can be written in the *partitioned* form, hereafter denoted by (\mathcal{P}) ,

$$\boxed{\text{PS1bis}} \quad (\mathcal{P}) \quad \begin{cases} \frac{dy}{dt}(t) = \mathcal{F}(t, y(t), z(t)), \\ \varepsilon \frac{dz}{dt}(t) = \mathcal{G}(t, y(t), z(t)), \end{cases}$$

then the stiffness is associated to variable z , and the corresponding equation will be treated implicitly, while the equation for y is treated explicitly. In other cases it is more convenient to associate the stiffness to a part of the right hand side, for example if a system has the *additive* form (\mathcal{A}) , in this case the term $F(t, u(t))$ is treated explicitly while $G(t, u(t))/\varepsilon$ is treated implicitly. It can be shown that the same system can be written in either form, however sometimes one of the two forms is more convenient.

Directly motivated by the above cases, we consider a more general class of problems of the form

$$\boxed{\text{equation1}} \quad (\mathcal{G}) \quad \begin{cases} \frac{du}{dt}(t) = \mathcal{H}(t, u(t), u(t)/\varepsilon), \quad \forall t \geq t_0, \\ u(t_0) = u_0, \end{cases}$$

where the function $\mathcal{H}: \mathbb{R} \times \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}^m$ is sufficiently differentiable and the right hand side has a stiff dependence only on the last argument, emphasized by the appearance on the small parameter ε in the denominator. We denote this class of problems as *generalised partitioned form* and hereafter denoted by (\mathcal{G}) .

Remark 1.1. *Note that the parameter ε in (\mathcal{G}) does not necessarily appear explicitly, but it just indicates some stiffness in the term. Sometime the stiffness is hidden, and is not explicitly expressed in terms of a small parameter ε . For example, in the case of diffusion terms $\varepsilon = \mathcal{O}(\Delta x^2)$, see for example, [7, 8, 2, 11].*

Now, let us first show the following hierarchical dependence

prop:1 **Proposition 1.1.** *Consider a system in partitioned form (\mathcal{P}) . Then it can be written either in the additive form (\mathcal{A}) or in the generalized partitioned form (\mathcal{G}) .*

Proof. This hierarchical dependence can be illustrated by the following Figure 1.

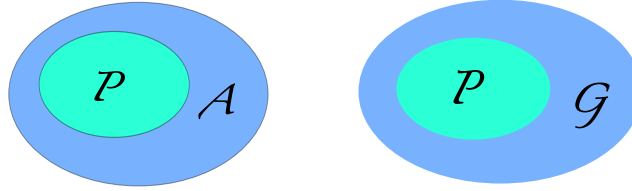


FIGURE 1. Illustration of the hierarchical dependence of (\mathcal{P}) , with respect to (\mathcal{A}) and (\mathcal{G}) approach. Sets1

First we consider the system (\mathcal{P}) and set $u = (y, z)^T$, $F = (\mathcal{F}, 0)$ and $G = (0, \mathcal{G})$, then u is solution to (\mathcal{A}) .

On the other hand, we again consider the system (\mathcal{P}) and set $u = (u_1, u_2)$ with $u_1 = y$ and $u_2 = \varepsilon z$, then we have from (\mathcal{P})

$$\begin{cases} \frac{du_1}{dt}(t) = \mathcal{F}(t, u_1(t), u_2(t)/\varepsilon), \\ \frac{du_2}{dt}(t) = \mathcal{G}(t, u_1(t), u_2(t)/\varepsilon), \end{cases}$$

that is, system (\mathcal{G}) with $u = (u_1, u_2)$ and

$$\begin{pmatrix} \mathcal{F}(t, u_1(t), u_2(t)/\varepsilon) \\ \mathcal{G}(t, u_1(t), u_2(t)/\varepsilon) \end{pmatrix} = \mathcal{H}(t, u(t), u(t)/\varepsilon).$$

□

We note that in this two cases no duplications of unknowns is needed. Unfortunately there is no such hierarchical dependence between (\mathcal{A}) and (\mathcal{G}) systems. Note however that if we allow to double the number of unknowns, then we have the following dependence.

prop:2 **Proposition 1.2.** *We have the following assertions*

- (1) *Consider a system in the additive form (\mathcal{A}) . Then by doubling the number of unknowns, it can be written in the partitioned form (\mathcal{P}) .*
- (2) *Consider a system in the generalized partitioned form (\mathcal{G}) . Then by doubling the number of unknowns, it can be written in the partitioned form (\mathcal{P}) .*

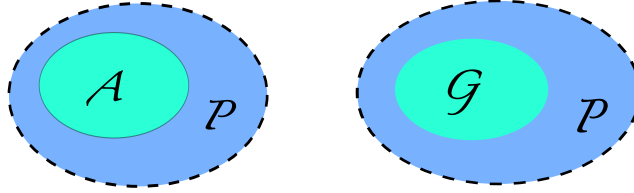


FIGURE 2. Illustration of hierarchical dependence between partitioned approach (\mathcal{P}), additive approach (\mathcal{A}) and *generalised partitioned form* (\mathcal{G}) approach where the dashed lines represent the set of systems with doubled number of unknowns.

Sets2

Proof. This hierarchical dependence can be illustrated by the following Figure 2.

We start with (1) and consider a solution u to (\mathcal{A}). We define the couple (y, z) as the solution to

$$\begin{cases} \frac{dy}{dt}(t) = F(t, y(t) + z(t)), \\ \varepsilon \frac{dz}{dt}(t) = G(t, y(t) + z(t)), \end{cases}$$

hence u is given by $u = y + z$ and the couple (y, z) is written as a solution to (\mathcal{P}) with

$$\begin{cases} \frac{dy}{dt}(t) = \mathcal{F}(t, y(t), z(t)), \\ \varepsilon \frac{dz}{dt}(t) = \mathcal{G}(t, y(t), z(t)), \end{cases}$$

where $\mathcal{F}(t, y(t), z(t)) = F(y(t) + z(t))$ and $\mathcal{G}(t, y(t), z(t)) = G(y(t) + z(t))$.

Then to prove (2), we consider a solution to (\mathcal{G}) and set $v = u/\varepsilon$. Thus, we have

$$\boxed{\text{PS1ter}} \quad (1) \quad \begin{cases} \frac{du}{dt}(t) = \mathcal{H}(t, u(t), v(t)), \\ \varepsilon \frac{dv}{dt}(t) = \mathcal{H}(t, u(t), v(t)), \end{cases}$$

it is a particular case of system (\mathcal{P}). Note that these two last cases require duplication of unknowns. \square

Such inclusions are very important, since the analysis of the numerical schemes applied to one family of schemes is automatically valid also for schemes applied to the other two families. As far as we know, it is not possible to write an \mathcal{A} system in the form \mathcal{G} or viceversa, without doubling the number of unknowns.

Thus, the formal equivalence among the various systems allows us to adopt techniques well know for additive or partitioned systems to more general cases.

A remark is in order at this point. A vast literature exists on the formal analysis of systems (\mathcal{P}), which are denoted as *singular perturbation systems*, [26, 24, 9, 10, 14, 16]. However, system (1) is only formally a particular case of (\mathcal{P}), and the analysis developed for the former cannot be directly applied to this case, since now the two functions \mathcal{F} and \mathcal{G} are the same. Furthermore, a detailed asymptotic analysis of the schemes presented here goes beyond the the scope of the paper, and will be considered in a future work.

The main goal of the paper is to focus on the treatment of systems of the form \mathcal{G} , *i.e.* (\mathcal{G}), by using some IMEX schemes already presented in the literature. Furthermore, in order to simply tie expression of the formulas, we drop the dependence of the parameter ε in the second argument as mentioned before, keeping in mind that the dependence on the second argument is stiff. Then in this

paper we consider the general class of problems of the form

$$\text{uation1BIS} \quad (2) \quad \begin{cases} \frac{du}{dt}(t) = \mathcal{H}(t, u(t), u(t)), & \forall t \geq t_0, \\ u(t_0) = u_0, \end{cases}$$

In particular, we show several examples of systems of the form \mathcal{G} that can be efficiently solved with the new approach.

The aim of this paper is to propose a new class of semi-implicit schemes based on IMEX Runge-Kutta methods which are strongly inspired by partitioned Runge-Kutta methods [25] and very much related to the additive Runge-Kutta methods of Zhong [38].

In the next section, we describe the general framework to construct this new class of semi-implicit Runge-Kutta schemes based on partitioned methods. Several schemes are proposed with different stability properties and order of accuracy. We next compare the numerical solutions with exact ones available in the literature for reaction-diffusion problem and nonlinear convection-diffusion equation. After this validation step, we perform several numerical computations to show the robustness of our approach (nonlinear Fokker-Planck equation, Hele-Shaw flow and surface diffusion flow). The last section is devoted to the analysis of stability properties of our schemes.

2. NUMERICAL METHODS FOR ODES

sec:2

In this section we review the concept of partitioned Runge-Kutta methods and derive a new class of semi-implicit R-K schemes, and we propose several schemes up to third order of accuracy, based on IMEX Runge-Kutta schemes already existing in the literature.

2.1. From Partitioned to semi-implicit Runge-Kutta methods. In the literature some interesting numerical methods do not belong to the classical class of implicit or explicit Runge-Kutta methods. They are called *partitioned* Runge-Kutta methods, [25, 26]. In order to present these methods we consider differential equations in the partitioned form,

$$\text{parsystem} \quad (3) \quad \begin{cases} \frac{dy}{dt}(t) = \mathcal{F}(t, y(t), z(t)), \\ \frac{dz}{dt}(t) = \mathcal{G}(t, y(t), z(t)), \end{cases}$$

where $y(t)$ and $z(t)$ may be vectors of different dimensions and $y(t_0) = y_0$, $z(t_0) = z_0$ are the initial conditions.

The idea of the partitioned Runge-Kutta methods is to apply two different Runge-Kutta methods, *i.e.*

$$\text{hyp:0} \quad (4) \quad \begin{array}{c|c} \hat{c} & \hat{A} \\ \hline & \hat{b}^T \end{array} \quad \begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

where we treat the first variable y with the first method, \hat{A} , $\hat{b}^T = (\hat{b}_1, \dots, \hat{b}_s)$, $\hat{c} = (\hat{c}_1, \dots, \hat{c}_s)$ and the second variable z with the second method, A , $b^T = (b_1, \dots, b_s)$, $c = (c_1, \dots, c_s)$ under the usual assumption

$$\text{hyp:1} \quad (5) \quad \sum_j \hat{a}_{i,j} = \hat{c}_i, \quad \text{and} \quad \sum_j a_{i,j} = c_i, \quad \text{for} \quad 1 \leq i \leq s.$$

In other words, if we consider a numerical approximation (y^n, z^n) of (3) at time t^n , a partitioned Runge-Kutta method for the solution of (3) is given by

$$\boxed{\text{PRKm1}} \quad (6) \quad \begin{cases} k_i = \mathcal{F} \left(t^n + \hat{c}_i \Delta t, y^n + \Delta t \sum_{j=1}^s \hat{a}_{ij} k_j, z^n + \Delta t \sum_{j=1}^s a_{ij} \ell_j \right), & 1 \leq i \leq s, \\ \ell_i = \mathcal{G} \left(t^n + c_i \Delta t, y^n + \Delta t \sum_{j=1}^s \hat{a}_{ij} k_j, z^n + \Delta t \sum_{j=1}^s a_{ij} \ell_j \right), & 1 \leq i \leq s \end{cases}$$

and the numerical solution at the next time step is given by

$$\boxed{\text{PRKm2}} \quad (7) \quad \begin{cases} y^{n+1} = y^n + \Delta t \sum_{i=1}^s \hat{b}_i k_i, \\ z^{n+1} = z^n + \Delta t \sum_{i=1}^s b_i \ell_i. \end{cases}$$

Now to derive a general semi-implicit Runge-Kutta scheme, we only observe that we can rewrite system (2) as

$$\begin{cases} \frac{dy}{dt}(t) = \mathcal{H}(t, y(t), z(t)), \\ \frac{dz}{dt}(t) = \mathcal{H}(t, y(t), z(t)), \end{cases}$$

with initial conditions $y(t_0) = y_0, z(t_0) = y_0$. In this way the system is a particular case of partitioned system in which $\mathcal{F} = \mathcal{G}$ but with an additional computational cost since we double the number of variables. Applying the partitioned Runge-Kutta method (6)-(7) we have

$$\boxed{\text{SIRKm1}} \quad (8) \quad \begin{cases} k_i = \mathcal{H}(t^n + \hat{c}_i \Delta t, Y_i, Z_i), & 1 \leq i \leq s, \\ \ell_i = \mathcal{H}(t^n + c_i \Delta t, Y_i, Z_i), & 1 \leq i \leq s, \end{cases}$$

with

$$\begin{cases} Y_i = z^n + \Delta t \sum_{j=1}^s \hat{a}_{i,j} k_j, & 1 \leq i \leq s, \\ Z_i = z^n + \Delta t \sum_{j=1}^s a_{ij} \ell_j, & 1 \leq i \leq s, \end{cases}$$

and the numerical solutions at the next time step are

$$\begin{cases} y^{n+1} = y^n + \Delta t \sum_{i=1}^s \hat{b}_i k_i, \\ z^{n+1} = z^n + \Delta t \sum_{i=1}^s b_i \ell_i. \end{cases}$$

At this stage let us adress some comments on several issues : number of evaluations, storage, order of accuracy and embedded methods.

Remark 2.1 (Concerning the number of evaluations of \mathcal{H}). *In general, k_i and ℓ_i given by (8) for all $1 \leq i \leq s$ are different. However, there are two cases in which $k_i = \ell_i$, $i = 1, \dots, s$. The first one is when the system is autonomous, i.e. \mathcal{H} does not explicitly depend on time, and the second one is when $\hat{c}_i = c_i$, $i = 1, \dots, s$. In these two cases only one evaluation of \mathcal{H} is needed in (8), and only one set of stage fluxes is computed:*

$$k_i = \mathcal{H} \left(y^n + \Delta t \sum_{i=1}^s \hat{a}_{i,j} k_j, y^n + \Delta t \sum_{i=1}^s a_{i,j} k_j \right), \quad 1 \leq i \leq s.$$

Remark 2.2 (Concerning the storage issues and order of accuracy). *If $\ell_i = k_i$ for $i = 1, \dots, s$, under the additional assumption $\hat{b}_i = b_i$ for $i = 1, \dots, s$ then also the numerical solutions are the same, i.e. $z^{n+1} = y^{n+1}$ and no duplication of variables is needed. In fact, by keeping track of the Runge-Kutta fluxes k_i rather than of the stage values Y_i and Z_i , one avoids the duplication of the number of variables.*

This is indeed the case of s -stage partitioned schemes of classical order p , with $\hat{c} = c$, and with $s = p$, since the quadrature formulas (c, \hat{b}) and (c, b) are necessarily interpolatory and therefore they are the same. Most of the schemes adopted in the paper lie in this category.

Note, however, that even in the general case, i.e. if $b_i \neq \hat{b}_i$, $i = 1, \dots, s$ or $c_i \neq \hat{c}_i$, $i = 1, \dots, s$ and the system is not autonomous, for a method which is consistent to order p , one has:

$$y^{n+1} = y(t^{n+1}) + \mathcal{O}(\Delta t^{p+1}), \quad z^{n+1} = z(t^{n+1}) + \mathcal{O}(\Delta t^{p+1}),$$

where we adopt the classical order of the scheme, i.e. we assume $\varepsilon = 1$.

Considering that $z(t^{n+1}) = y(t^{n+1})$, then one has $z^{n+1} = y^{n+1} + \mathcal{O}(\Delta t^{p+1})$ which means that if we neglect the difference between z^{n+1} and y^{n+1} and choose, for example, to advance y^{n+1} and to set, at the beginning of a new time step, $z^{n+1} = y^{n+1}$, then one obtains another scheme still of order p , with no duplication of variables.

Hereafter we assume that we follow the evolution of y^n , and we set $z^n := y^n$, at the beginning of each time step. We expect that in general it is more accurate to follow the non stiff variable y , but there may be exception, and for some scheme it may be more convenient to follow the evolution of the stiff variable z and to set $y^n := z^n$ at the beginning of the time step.

Since the equations contain stiff terms, classical order analysis may not be sufficient to justify the practical equivalence of the two different choices (i.e. advancing y^n and setting $z^{n+1} = y^{n+1}$ or advancing z^n and setting $z^{n+1} = y^{n+1}$) and a more detailed analysis is needed. In the first example presented in the paper, we compare numerically the two approaches (Test 1 reaction-diffusion problem) whereas all other systems are autonomous, so that if $\hat{b} = b$ then $y^{n+1} = z^{n+1}$. Let us notice that there are only two cases in which $\hat{b} \neq b$. For such cases we explore numerically the two choices, and compare the results, to establish which one is better.

Remark 2.3 (Embedded methods). *From the above remarks, let us consider an IMEX scheme with $\hat{c} = c$, so that $k = \ell$. Then we can select a different vector of weights b for the z variable that provides a lower order approximation of the solution, and construct in this way an embedded method, which can be used for an automatic time step control [25], although we shall not implement any time step control in the present paper.*

From now on we shall adopt IMEX R-K schemes. These are additive R-K schemes that combine explicit and implicit R-K methods. Usually they are applied to additive systems that involve terms with different stiffness properties. An example of additive system is given by (\mathcal{A}) , where we apply an implicit method to the stiff part $G(t, u(t))/\varepsilon$ and an explicit one to $F(t, u(t))$. It is usual to consider diagonally implicit R-K (DIRK) schemes for the implicit part. In addition to be simpler to implement, this will ensure that the terms involving function F are explicit. The coefficients of the method are usually represented in a double Butcher tableau as (4).

Now we are ready to propose semi-implicit Runge-Kutta methods in order to solve problem (\mathcal{G}) when the dependence from the second variable is stiff. We will treat the first variable explicitly, and the second one implicitly. A semi-implicit Runge-Kutta method is implemented as follows. First we set $z^n = y^n$ and compute the stage fluxes for $i = 1, \dots, s$, we set $Y_1 = \tilde{Z}_1 = y^n$ and

$$\boxed{\text{PS1}} \quad (9) \quad \left\{ \begin{array}{l} Y_i = y^n + \Delta t \sum_{j=1}^{i-1} \hat{a}_{ij} k_j, \quad 2 \leq i \leq s, \\ \tilde{Z}_i = y^n + \Delta t \sum_{j=1}^{i-1} a_{ij} \ell_j, \quad 2 \leq i \leq s \\ \ell_i = \mathcal{H} \left(t^n + c_i \Delta t, Y_i, \tilde{Z}_i + \Delta t a_{ii} \ell_i \right), \quad 1 \leq i \leq s, \\ k_i = \mathcal{H} \left(t^n + \hat{c}_i \Delta t, Y_i, \tilde{Z}_i + \Delta t a_{ii} \ell_i \right), \quad 1 \leq i \leq s, \end{array} \right.$$

and, finally update the numerical solution by

$$\boxed{\text{eq1-1}} \quad (10) \quad y^{n+1} = y^n + \Delta t \sum_{i=1}^s \hat{b}_i k_i,$$

or by

$$\boxed{\text{eq1-2}} \quad (11) \quad z^{n+1} = y^n + \Delta t \sum_{i=1}^s b_i \ell_i.$$

At the end of the step the difference $z^{n+1} - y^{n+1}$ can be used as local time step control indicator. After that, we set $n \leftarrow n + 1$, $z^n \leftarrow y^n$, assuming that we advance in time with the scheme for the variable y .

In most cases the system is autonomous, and duplication of variables is not necessary. In this case (9) reduces to

$$\boxed{\text{PS1bisbis}} \quad (12) \quad \left\{ \begin{array}{l} Y_i = y^n + \Delta t \sum_{j=1}^{i-1} \hat{a}_{ij} k_j, \quad 2 \leq i \leq s, \\ \tilde{Z}_i = y^n + \Delta t \sum_{j=1}^{i-1} a_{ij} k_j, \quad 2 \leq i \leq s \\ k_i = \mathcal{H} \left(Y_i, \tilde{Z}_i + \Delta t a_{ii} k_i \right), \quad 1 \leq i \leq s. \end{array} \right.$$

and update the numerical solution

$$\boxed{\text{PS2}} \quad (13) \quad y^{n+1} = y^n + \Delta t \sum_{i=1}^s b_i k_i.$$

Remark 2.4. We note that this new approach includes Zhong's method [38]. The theory developed in [38] for additive semi-implicit Runge-Kutta methods can be extended in a straightforward manner to the semi-implicit Runge-Kutta methods. In fact, by setting $\mathcal{H}(y, y) = \mathcal{F}(y) + \mathcal{G}(y)$ we obtain for the

numerical method

$$\begin{aligned} k_i &= \mathcal{H} \left(y^n + \sum_{j=1}^{j-1} \hat{a}_{ij} k_j, y^n + \sum_{j=1}^{j-1} a_{ij} k_j + a_{ii} k_i \right), \\ &= \mathcal{F} \left(y^n + \sum_{j=1}^{j-1} \hat{a}_{ij} k_j \right) + \mathcal{G} \left(y^n + \sum_{j=1}^{j-1} a_{ij} k_j + a_{ii} k_i \right), \end{aligned}$$

ZhongIS (14)

for $i = 1, \dots, s$ and for the numerical solution

ZhongNS (15)

$$y^{n+1} = y^n + \sum_{i=1}^s b_i k_i,$$

which are exactly those proposed by Zhong [38].

In the following we propose different types of semi-implicit Runge-Kutta methods and verify that the order conditions are the same as the ones satisfied by the explicit and implicit Runge-Kutta schemes.

Sec2bis

2.2. Classification of IMEX Runge-Kutta schemes. Next we give a classification of such schemes and recall the order conditions to obtain second and third order accuracy in time. Finally we list several second and third order IMEX R-K schemes presented in the literature, [31, 32, 33] that we will use for our semi-implicit framework (9)-(13).

IMEX Runge-Kutta schemes presented in the literature can be classified in three different types characterized by the structure of the matrix $A = (a_{ij})_{i,j=1}^s$ of the implicit scheme. Following [9], we will rely on the following notions [1, 16, 33].

Def:A

Definition 2.1. An IMEX Runge-Kutta method is said to be **of type A** [33] if the matrix $A \in \mathbb{R}^{s \times s}$ is invertible. It is said to be **of type CK** [16] if the matrix $A \in \mathbb{R}^{s \times s}$ can be written in the form

$$A = \begin{pmatrix} 0 & 0 \\ a & \mathcal{A} \end{pmatrix},$$

in which the matrix $\mathcal{A} \in \mathbb{R}^{(s-1) \times (s-1)}$ invertible. Finally, it is said to be **of type ARS** [1] if it is a special case of the type CK with the vector $a = 0$.

Schemes of type CK are very attractive since they allow some simplifying assumptions, that make order conditions easier to treat, therefore permitting the construction of higher order IMEX Runge-Kutta schemes. On the other hand, schemes of type A are more amenable to a theoretical analysis, since the matrix A of the implicit scheme is invertible.

Sec2

2.3. Order conditions and numerical schemes. Runge-Kutta methods (9)-(13) are a special case of the semi-implicit ones (6)-(7). Thus, the order conditions for (9) and (13) are a direct consequence of the classical order conditions computed for partitioned Runge-Kutta methods, [25, 27]. Here we recall some known results for IMEX R-K schemes presented in [16, 31, 33]. In particular the order conditions up to order 3 can be simplified if we set $\hat{b}_i = b_i$ for $i = 1, \dots, s$. Then using the previous notation for the explicit and implicit part and assuming (4) and (5), we have for a method of order 2

order:2

$$(16) \quad \sum_i b_i = 1, \quad \sum_i b_i c_i = 1/2, \quad \sum_i b_i \hat{c}_i = 1/2.$$

and for a method of order 3

order3:imp

$$(17) \quad \sum_i b_i c_i^2 = 1/3, \quad \sum_{i,j} b_i a_{ij} c_j = 1/6,$$

order3:exp

$$(18) \quad \sum_i b_i \hat{c}_i^2 = 1/3, \quad \sum_{i,j} b_i \hat{a}_{ij} \hat{c}_j = 1/6,$$

$$(19) \quad \sum_i b_i \hat{c}_i c_i = 1/3, \quad \sum_{i,j} b_i a_{ij} \hat{c}_j = 1/6, \quad \sum_{i,j} b_i \hat{a}_{ij} c_j = 1/6.$$

The general conditions in case $\hat{b} \neq b$ can be found in [33].

We recall some well know definitions that we shall adopt in the paper.

Definition 2.2. *An implicit R-K method is called stiffly accurate if $b^T = e_s^T A$ with $e_s^T = (0, \dots, 0, 1)$, i.e. methods for which the numerical solution is identical to the last internal stage.*

Remark 2.5. *This property is important for the L-stability of the implicit part of the method, i.e. an A-stable implicit method is also L-stable, (see [26] for details).*

Now, we first consider second order schemes with two stages that satisfy the set of order conditions (16). For practical reasons we consider *singly diagonally implicit Runge-Kutta* (SDIRK) schemes for the implicit part, i.e. $a_{ii} = \gamma$, for $i = 1, \dots, s$. The Butcher tableau takes then the following form

$$(20) \quad \begin{array}{c|cc} 0 & 0 & 0 \\ \hat{c} & \hat{c} & 0 \\ \hline & b_1 & b_2 \end{array} \quad \begin{array}{c|cc} \gamma & \gamma & 0 \\ c & c - \gamma & \gamma \\ \hline & b_1 & b_2 \end{array}$$

with the following coefficients:

$$(21) \quad b_1 = 1 - b_2, \quad \hat{c} = 1/(2b_2), \quad c = (1/2 - \gamma(1 - b_2))/b_2,$$

where $b_2 \neq 0$ and $\gamma > 0$ free parameters.

One drawback of the present approach (9)-(13) for non autonomous ODE may come from the fact that it would require twice evaluations of the right hand side \mathcal{H} for the computation of $(k_i)_{1 \leq i \leq s}$ and $(\ell_i)_{1 \leq i \leq s}$. However for second order schemes with two stages, it is easy to verify that the evaluation of $(\ell_i)_{1 \leq i \leq 2}$ is enough and the choice $k_1 = \ell_1$, (which is equivalent to set $\hat{c}_1 = c_1 \neq 0$) does not modify the order of the scheme even if condition (5) is not satisfied. In fact, for low orders, condition (5) is not necessary, see [30] for details.

We list below the second order schemes that we shall use in the paper.

2.3.1. The second order semi-implicit Runge-Kutta scheme - H-SDIRK2(2,2,2). A first example of scheme satisfying the second order conditions given in (16) is $b_2 = \gamma = 1/2$, which yields the following table

$$(22) \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & 1/2 & 1/2 \end{array} \quad \begin{array}{c|cc} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}$$

This scheme is the combination of Heun method (explicit part) and an A-stable second order singly diagonal implicit Runge-Kutta SDIRK method (implicit part), hence we call it H-SDIRK2(2,2,2).

2.3.2. The second order semi-implicit Runge-Kutta scheme of type CK - H-CN(2,2,2). A variant of the previous scheme satisfying the second order conditions (16) is given by the following table

$$(23) \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & 1/2 & 1/2 \end{array} \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}$$

This scheme is the combination of Heun method (explicit part) and a trapezoidal rule Crank Nicolson (implicit part) which is A-stable second order implicit, hence we call it H-CN(2,2,2). It may be a natural choice when dealing with convection-diffusion equation, since the Heun method is an SSP explicit RK [23], and the trapezoidal rule (also known as Crank-Nicolson) is an A-stable method, widely used for diffusion problems.

2.3.3. The stiffly accurate semi-implicit Runge-Kutta scheme - LSDIRK2(2,2,2). Another choice for the coefficients (21) is $b_2 = \gamma$, $c = 1$, where γ is chosen as the smallest root of the polynomial $\gamma^2 - 2\gamma + 1/2 = 0$, *i.e.* $\gamma = 1 - 1/\sqrt{2}$ and $\hat{c} = 1/(2\gamma)$. This gives

$$\text{scheme2} \quad (24) \quad \begin{array}{c|cc} 0 & 0 & 0 \\ \hat{c} & \hat{c} & 0 \\ \hline & 1-\gamma & \gamma \end{array} \quad \begin{array}{c|cc} \gamma & \gamma & 0 \\ 1 & 1-\gamma & \gamma \\ \hline & 1-\gamma & \gamma \end{array}$$

This scheme is the combination of Runge-Kutta method (explicit part) and an L -stable second order SDIRK method in the implicit part. Thus, we call it LSDIRK2(2,2,2).

2.3.4. The IMEX-SSP2(2,2,2) L-stable scheme - H-LDIRK p (2,2,2). We choose $b_2 = 1/2$, $\hat{c} = 1$, *i.e.* the corresponding Butcher tableau is given by

$$\text{scheme3} \quad (25) \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & 1/2 & 1/2 \end{array} \quad \begin{array}{c|cc} \gamma & \gamma & 0 \\ 1-\gamma & 1-2\gamma & \gamma \\ \hline & 1/2 & 1/2 \end{array}$$

This scheme, introduced in [31], is based on Heun method coupled with an L -stable SDIRK Runge-Kutta. We call it H-LDIRK p (2,2,2). The implicit part has order $p = 2$ with $\gamma = 1 - 1/\sqrt{2}$ and order $p = 3$ with $\gamma = (3 + 3\sqrt{6})/3$. The version for $p = 2$ also has the *strongly stability preserving* (SSP) property, see [31]. The choice of γ for $p = 3$ guarantees that the implicit part is a third-order DIRK scheme with the best dampening properties [25].

2.3.5. The stiffly accurate IMEX-SSP2(3,3,2) L-stable scheme - SSP-LDIRK2(3,3,2). The last second order scheme is obtained by combining a three-stage, second-order SSP scheme with an L -stable, second-order DIRK scheme, and is denoted by SSP-LDIRK2(3,3,2). This scheme is given by

$$\text{scheme4} \quad (26) \quad \begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 1 & 1/2 & 1/2 & 0 \\ \hline & 1/3 & 1/3 & 1/3 \end{array} \quad \begin{array}{c|ccc} 1/4 & 1/4 & 0 & 0 \\ 1/4 & 0 & 1/4 & 0 \\ 1 & 1/3 & 1/3 & 1/3 \\ \hline & 1/3 & 1/3 & 1/3 \end{array}$$

2.3.6. The IMEX-SSP3(4,3,3) L-stable scheme - SSP-LDIRK3(4,3,3). As third semi-implicit Runge-Kutta methods of the type (9)-(13), that satisfies the set of order conditions (17)-(18)-(19), a possible choice is given by the IMEX-SSP3(4,3,3) L-stable scheme with $\hat{b}_i = b_i$ for $i = 1, \dots, s$, *i.e.*

$$\text{scheme5} \quad (27) \quad \begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1/2 & 0 & 1/4 & 1/4 & 0 \\ \hline & 0 & 1/6 & 1/6 & 2/3 \end{array} \quad \begin{array}{c|cccc} \alpha & \alpha & 0 & 0 & 0 \\ 0 & -\alpha & \alpha & 0 & 0 \\ 1 & 0 & 1-\alpha & \alpha & 0 \\ 1/2 & \beta & \eta & 1/2 - \beta - \eta - \alpha & \alpha \\ \hline & 0 & 1/6 & 1/6 & 2/3 \end{array}$$

with $\alpha = 0.24169426078821$, $\beta = \alpha/4$ and $\eta = 0.12915286960590$. We call this scheme SSP-LDIRK3(4,3,3). For this particular choice, let us observe that the number of evaluations of the right hand side \mathcal{H} is still reasonable since the coefficients $c_i = \hat{c}_i$ for $2 \leq i \leq 4$ and only c_1 differs from \hat{c}_1 .

We remark that schemes H-LDIRK2(2,2,2), SSP-LDIRK2(3,3,2), and SSP-LDIRK3(4,3,3) were introduced in the context of hyperbolic systems with stiff relaxation in [31].

Remark 2.6. *Even though IMEX R-K methods can be considered as additive R-K methods ([16, 1, 13]), we note that the coefficients in the Zhong form (14), (15) are not the standard coefficients of an IMEX*

R-K method. It is possible to show, by inspection, that equations (14), (15) can be written as an IMEX R-K method. Let

$$\hat{A} = \begin{pmatrix} 0 & 0 & & \\ \hat{a}_{21} & 0 & & \\ \vdots & \vdots & \ddots & \\ a_{s1} & \cdots & \cdots & a_{ss-1} \end{pmatrix}, \quad A = \begin{pmatrix} a_{11} & 0 & & \\ a_{21} & a_{22} & & \\ \vdots & \vdots & \ddots & \\ a_{s1} & \cdots & \cdots & a_{ss} \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_s \end{pmatrix}$$

be the coefficients of the scheme (14), (15), with s stages. Then we write explicitly

$$y^{n+1} = y^n + \sum_{i=1}^s \left(\mathcal{F}(\hat{Y}_i) + \mathcal{G}(Y_i) \right)$$

with

$$\begin{cases} \hat{Y}_1 = y^n, \\ Y_1 = y^n + h a_{11} \left(\mathcal{F}(\hat{Y}_1) + \mathcal{G}(Y_1) \right), \\ \hat{Y}_2 = y^n + h \hat{a}_{21} \left(\mathcal{F}(\hat{Y}_1) + \mathcal{G}(Y_1) \right), \\ Y_2 = y^n + h a_{21} \left(\mathcal{F}(\hat{Y}_1) + \mathcal{G}(Y_1) \right) + h a_{22} \left(\mathcal{F}(\hat{Y}_2) + \mathcal{G}(Y_2) \right), \\ \dots \\ \hat{Y}_s = y^n + h \hat{a}_{s1} \left(\mathcal{F}(\hat{Y}_1) + \mathcal{G}(Y_1) \right) + \dots + h \hat{a}_{s,s-1} \left(\mathcal{F}(\hat{Y}_{s-1}) + \mathcal{G}(Y_{s-1}) \right), \\ Y_s = y^n + h a_{s1} \left(\mathcal{F}(\hat{Y}_1) + \mathcal{G}(Y_1) \right) + \dots + h a_{ss} \left(\mathcal{F}(\hat{Y}_{ss}) + \mathcal{G}(Y_{ss}) \right). \end{cases}$$

and the Butcher tableau is

$$\begin{array}{c|cccccc} 0 & & & & & \\ a_{11} & 0 & & & & \\ \hat{a}_{21} & 0 & 0 & & & \\ a_{21} & 0 & a_{22} & 0 & & \\ \vdots & \vdots & \vdots & \vdots & \ddots & \\ a_{s1} & 0 & a_{s2} & 0 & \cdots & a_{ss} & 0 \\ \hline b_1 & 0 & b_2 & 0 & \cdots & b_s & 0 \end{array} \quad \begin{array}{c|cccccc} 0 & & & & & \\ 0 & a_{11} & & & & \\ 0 & \hat{a}_{21} & 0 & & & \\ 0 & a_{21} & 0 & a_{22} & & \\ \vdots & \vdots & \vdots & \vdots & \ddots & \\ 0 & a_{s1} & 0 & a_{s2} & \cdots & a_{ss} \\ \hline 0 & b_1 & 0 & b_2 & \cdots & b_s \end{array}$$

Using the Kronecker product, this can be written in a more compact form as

$$\left| \frac{(\hat{A} \otimes \hat{B}_1) + (A \otimes \hat{B}_2)}{b \otimes (1, 0)} \right| \quad \left| \frac{(\hat{A} \otimes B_1) + (A \otimes B_2)}{b \otimes (0, 1)} \right|$$

with

$$\hat{B}_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad \hat{B}_2 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$$

and

$$B_1 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

Then from the order conditions (16), (17), (18), (19), we can derive the corresponding conditions for schemes of the form (14)-(15), (for details see [38]).

3. APPLICATIONS

sec:3

In this section we present several numerical tests for nonlinear PDEs for reaction-diffusion systems and nonlinear convection-diffusion equation for which we verify the order of accuracy and stability issues with respect to the CFL condition. Then, we treat a nonlinear Fokker-Planck equation to investigate the long time behavior of the numerical solution obtained from (9)-(13). Finally we complete this section with numerical tests on Hele-Shaw flow and surface diffusion flow.

We monitor L_1 and L_∞ norms of the error, defined as:

$$\begin{cases} \varepsilon_\infty = \max_{0 \leq n \leq N_T} \max_{i,j} \|\omega_{i,j}^n - \omega(t^n, x_i, y_j)\|, \\ \varepsilon_1 = \max_{0 \leq n \leq N_T} \sum_{i,j} \Delta x \Delta y \|\omega_{i,j}^n - \omega(t^n, x_i, y_j)\|. \end{cases}$$

For space discretization we will apply basic fourth order discretization with central finite difference for first derivative

$$\nabla_h \omega_i = \frac{-\omega_{i+2} + 8\omega_{i+1} - 8\omega_{i-1} + \omega_{i-2}}{12h}$$

where h is the space step, and for the second derivative is discretized using a fourth order central finite difference scheme as well

$$\nabla_h^2 \omega_i = \frac{-\omega_{i+2} + 16\omega_{i+1} - 30\omega_i + 16\omega_{i-1} - \omega_{i-2}}{12h^2}.$$

3.1. Test 1 - Reaction-diffusion problem. We first consider a very simple reaction-diffusion system with nonlinear source for which there are explicit solutions.

To demonstrate the optimal accuracy of the semi-implicit method in various norms, we consider the reaction-diffusion system problem [37] together with periodic boundary conditions: $\omega = (\omega_1, \omega_2) : \mathbb{R}^+ \times (0, 2\pi)^2 \mapsto \mathbb{R}^2$

test:bruss

$$(28) \quad \begin{cases} \frac{\partial \omega_1}{\partial t} = \Delta \omega_1 - \alpha_1(t) \omega_1^2 + \frac{9}{2} \omega_1 + \omega_2 + f(t), & t \geq 0, \quad (x, y) \in (0, 2\pi)^2, \\ \frac{\partial \omega_2}{\partial t} = \Delta \omega_2 + \frac{7}{2} \omega_2, & t \geq 0, \quad (x, y) \in (0, 2\pi)^2, \end{cases}$$

with $\alpha(t) = 2e^{t/2}$ and $f(t) = -2e^{-t/2}$. The initial conditions are extracted from the exact solutions

$$\begin{cases} \omega_1(t, x, y) = \exp(-0.5t) (1 + \cos(x)), \\ \omega_2(t, x, y) = \exp(-0.5t) \cos(2x). \end{cases}$$

To apply our semi-implicit scheme (9)-(13) we rewrite this PDE in the form (\mathcal{G}) with $u = (u_1, u_2)$ the component treated explicitly, $v = (v_1, v_2)$ the component treated implicitly and

$$\mathcal{H}(t, u, v) = \begin{pmatrix} \Delta v_1 - \alpha(t) u_1 v_1 + \frac{9u_1}{2} + v_2 + f(t) \\ \Delta v_2 + \frac{7v_2}{2} \end{pmatrix}.$$

Since the Δ operator induces some stiffness it is treated implicitly whereas reaction terms are treated according to the sign of the reaction term and are linearized in order to avoid the numerical solution of a fully nonlinear problem. Concerning the spatial discretization, we simply apply a fourth order central finite difference method to the Δ operator. A fourth order accurate scheme for spatial derivatives is applied in order to bring out the order of accuracy of the second and third order time discretization.

To estimate the order of accuracy of the schemes we compute a numerical approximation and refine the time step Δt according to the space step $\Delta x = \Delta y$ in such a way the CFL condition associated to the diffusion operator is violated, that is, we apply an hyperbolic CFL condition where we refine the time step and the space step simultaneously

$$\lambda = \frac{2\Delta t}{\Delta x},$$

with $\lambda = 1$.

Obviously, for a fully explicit scheme like the Runge-Kutta method, this condition would lead to some instabilities of the numerical solution since a parabolic CFL is necessary.

The semi-implicit schemes are expected to be stable even for large time step when the parabolic CFL condition is not satisfied.

Furthermore, the PDE system (28) is non autonomous since the source term depends on time, hence the two solutions given by the explicit coefficient $(\hat{b}_i, \hat{c}_i)_{1 \leq i \leq s}$ in (10) and by the implicit coefficient $(b_i, c_i)_{1 \leq i \leq s}$ in (11) are different and we want to explore numerically the two choices.

Absolute error in L^∞ norms at time $T = 2$ are shown in Figure 3 for all the schemes presented in the previous section. As expected the order of accuracy is satisfied for all second and third order schemes. The results obtained with the third order SSP-DIRK3(4,3,3) scheme (27) are much more accurate than the others obtained from second order schemes. Moreover, let us also emphasize that comparing the amplitude of the L^∞ error norm obtained with the second order schemes, the SSP-LDIRK2(3,3,2) scheme (26) with three stages is much more accurate than the others. Among the second order schemes with two stages the H-CN scheme (23) is the most accurate for this test.

Finally, we only observe a slight difference between the two solutions y^{n+1} from (10) and z^{n+1} from (11), even if it seems that for this numerical test, the amplitude of the error is smaller when we use the solution given by the implicit stencil (11).

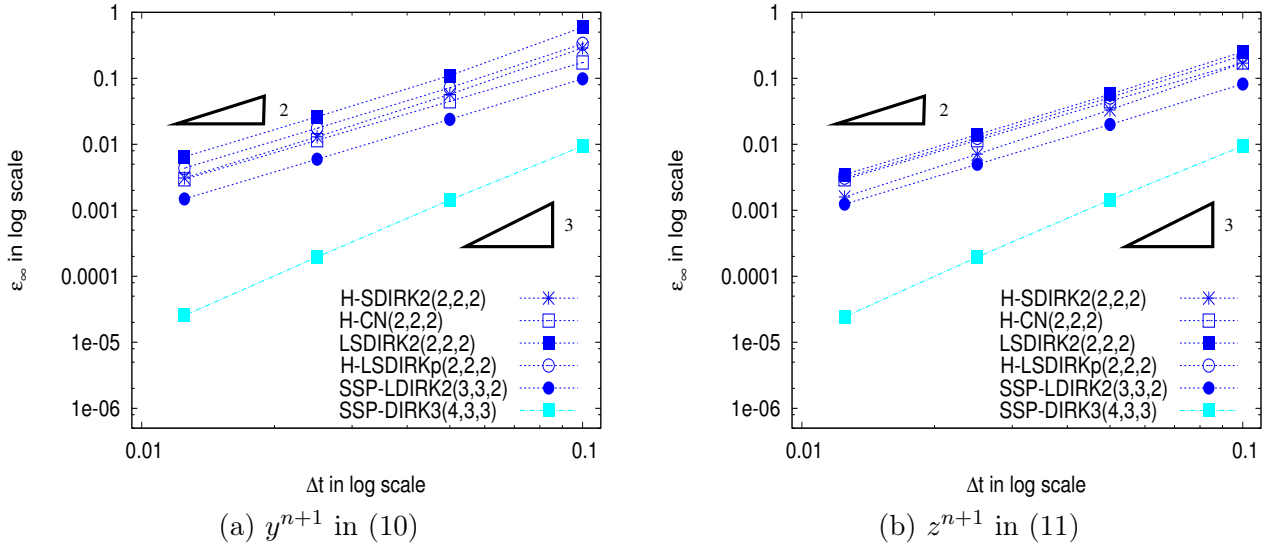


FIGURE 3. Test 1 - Reaction-diffusion problem: L^∞ error norm when the last step is performed using the coefficients obtained from the (a) explicit scheme (10) and (b) and the implicit scheme (11) for the IMEX schemes described in the previous section.

error: bruo

3.2. Test 2 - Nonlinear convection-diffusion equation. We consider the following nonlinear convection diffusion equation on the whole space \mathbb{R}^2 and apply a fourth order central finite difference scheme for the first and second spatial derivatives

$$\begin{cases} \frac{\partial \omega}{\partial t} + [V + \mu \nabla \log(\omega)] \cdot \nabla \omega - \mu \Delta \omega = 0, & (t, x) \in \mathbb{R}^+ \times \mathbb{R}^2, \\ \omega_0(t=0) = e^{-\|x\|^2/2}, \end{cases}$$

where $V = {}^t(1, 1)$, $\mu = 0.5$. The exact solution is given by

$$\omega(t, x) = \frac{1}{\sqrt{4\mu t + 1}} e^{-\frac{\|x - Vt\|^2}{8\mu t + 2}}, \quad t \geq 0, \quad x \in \mathbb{R}^2.$$

After the space discretization, we apply our semi-implicit scheme (9)-(13) by writing the system of ODEs in the form (\mathcal{G}) with u the component treated explicitly, v the component treated implicitly and

$$\mathcal{H}(t, u, v) = -(V + \mu \nabla \log(u)) \cdot \nabla v + \mu \Delta v.$$

We treat both the convection and diffusion implicitly but we only deal with a linear system at each time step. The computational domain in space is $(-10, 10)^2$ and the final time is $T = 0.5$. As in the previous case, the space step is chosen sufficiently small to neglect the influence of the space discretization and the time step Δt is taken proportional to Δx such that $\Delta t = \lambda \Delta x$, with $\lambda = 1$. Therefore, the classical CFL condition for convection diffusion problem $\Delta t = O(\Delta x^2)$ is not verified.

In Figure 4 we present the numerical error both for L^1 and L^∞ norms for the semi-implicit schemes described in the previous section and still verify the correct order of accuracy. Once more, the third order SSP-DIRK3(4,3,3) scheme (27) with four stages gives much more accurate results than second order schemes with several order of magnitude.

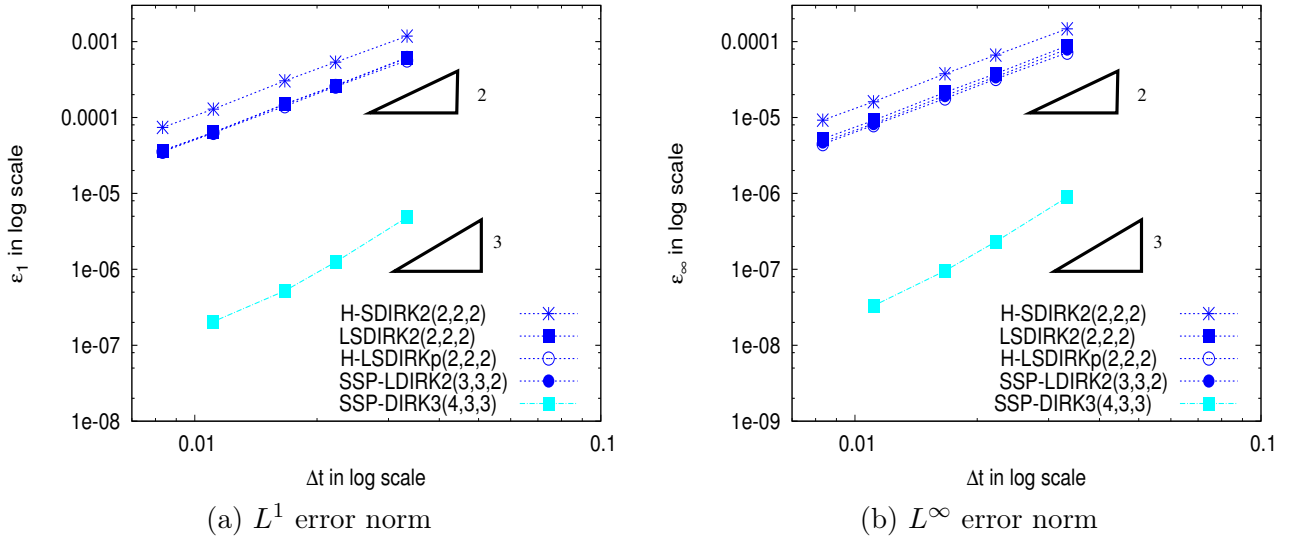


FIGURE 4. Test 2 - Nonlinear convection-diffusion problem: (a) L^1 error norm and (b) L^∞ error norm for the second order schemes (22), (23), (24), (25) and (26) and the third order SSP-DIRK3(4,3,3) L-stable scheme (27).

error:cd0

3.3. Test 3 - Nonlinear Fokker-Planck equations for fermions and bosons. In [18, 17], a nonlinear Fokker-Planck type equation modelling the relaxation of fermion and boson gases is studied. This equation has a linear diffusion and a nonlinear convection term:

$$(29) \quad \begin{cases} \frac{\partial \omega}{\partial t} = \operatorname{div}(x(1+k\omega)\omega + \nabla \omega), & x \in \mathbb{R}^d, \quad t > 0, \\ \omega(x, 0) = \omega_0(x), \end{cases}$$

with $k = 1$ in the boson case and $k = -1$ in the fermion case. For this equation, the explicit solution is not known except steady states, but there are several works devoted to the long time behavior based on the knowledge of the qualitative behavior of the entropy functional. The long time behavior of this model has been rigorously investigated quite recently in [17] via an entropy-dissipation approach. More precisely, the stationary solution of (29) is given by the Fermi-Dirac ($k = -1$) and Bose-Einstein

sonfermion

($k = 1$) distributions:

sonfermion

$$(30) \quad \omega^{eq}(x) = \frac{1}{\beta e^{\frac{|x|^2}{2}} - k},$$

where $\beta \geq 0$ is such that ω^{eq} has the same mass as the initial data ω_0 . For this equation, there exists an entropy functional given by

$$\mathcal{E}(\omega) := \int_{\mathbb{R}^d} \left(\frac{|x|^2}{2} \omega + \omega \log(\omega) - k(1 + k\omega) \log(1 + k\omega) \right) dx,$$

such that

$$\frac{d}{dt} \mathcal{E}(\omega) = -\mathcal{I}(t),$$

where the entropy dissipation $\mathcal{I}(t)$ is defined by

$$\mathcal{I}(t) := \int_{\mathbb{R}^d} \omega(1 + k\omega) \left| \nabla \left(\frac{|x|^2}{2} + \log \left(\frac{\omega}{1 + k\omega} \right) \right) \right|^2 dx.$$

Then decay rates towards equilibrium are given in [18, 17] for fermion case in any dimension and for 1D boson case by relating the entropy and its dissipation. Here we want to approximate this nonlinear equation and study the long time behavior of the numerical solution [6].

To apply our semi-implicit scheme we rewrite this PDE in the form (\mathcal{G}) with u the component treated explicitly, v the component treated implicitly and

$$\mathcal{H}(t, u, v) = \operatorname{div}(x(1 + ku)v + \nabla v) = \operatorname{div}(x(1 + ku)v) + \Delta v$$

and we apply a fourth order spatial discretization for the convective and diffusive components.

We consider the nonlinear Fokker-Planck equation (29) for fermions ($k = -1$) in 2D. The initial condition is chosen as

$$\omega_0(x) = \frac{1}{2\pi} |x|^2 \exp\left(-\frac{|x|^2}{2}\right), \quad x \in \mathbb{R}^2,$$

and the computational domain is $(-10, 10)^2$ with the space step $\Delta x = 0.1$.

Evolution of the discrete relative entropy $\mathcal{E}_\Delta(t^n)$, its dissipation $\mathcal{I}_\Delta(t^n)$ and $\|\omega^n - \omega^{eq}\|_{L^1}$ is presented in Figure 5. This is obtained at the top by second order schemes, *i.e.* classical second order explicit Runge-Kutta scheme and H-LDIRKp(2,2,2) (25), and at the bottom by third order schemes, *i.e.* classical third order explicit Runge-Kutta scheme and SSP-DIRK3(4,3,3) (27).

We observe exponential decay rate of these quantities, which is in agreement with the result proved by J. A. Carrillo, Ph. Laurençot and J. Rosado in [17] and the numerical results proposed in [6]. Classical Runge-Kutta schemes are subject to a parabolic condition whereas semi-implicit schemes can be used with a large time step without affecting the accuracy even for large time asymptotics.

3.4. Test 4 - Hele-Shaw flow. In this section we consider a fourth order nonlinear degenerate diffusion equation in one space dimension called the Hele-Shaw cell [4, 34]

eq:1.3

$$(31) \quad \frac{\partial \omega}{\partial t} + \frac{\partial}{\partial x} \left(\omega \frac{\partial^3 \omega}{\partial x^3} \right) = 0, \quad x \in \mathbb{R}, \quad t \geq 0,$$

with $\omega(x, t = 0) = \omega_0(x) \geq 0$.

One of the remarkable features of equation (31) is that its nonlinearity guarantees the nonnegativity preserving property of the solution [5] and the conservation of mass

$$\int_{\mathbb{R}} \omega(t, x) dx = \int_{\mathbb{R}} \omega_0(x) dx.$$

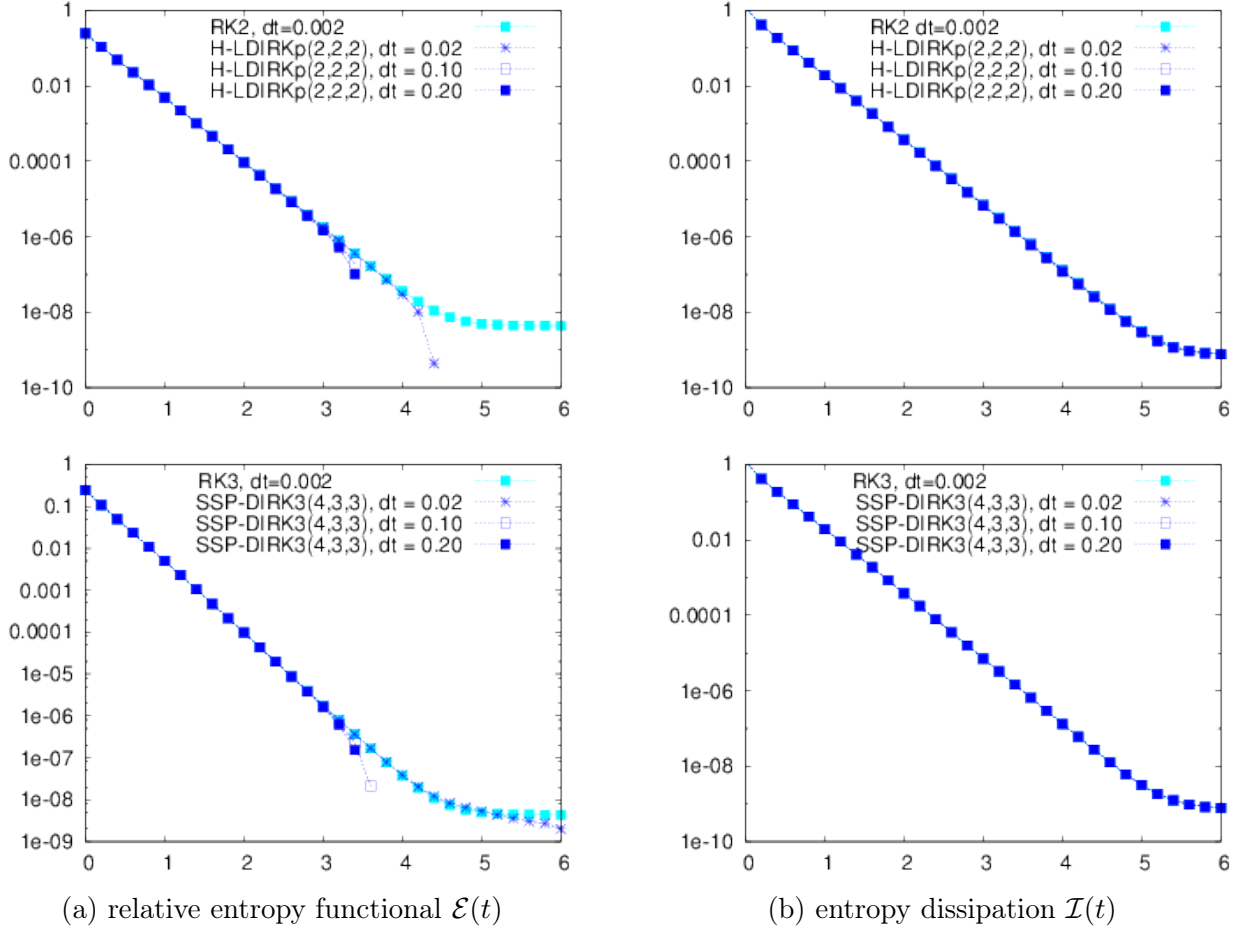


FIGURE 5. Test 3 - Fokker-Planck equation. (a) Evolution of the relative entropy $\mathcal{E}_\Delta(t^n)$ and (b) the dissipation $\mathcal{I}_\Delta(t^n)$ for the second order explicit Runge-Kutta scheme and for the H-LDIRKp(2,2,2) scheme (25) (top) and for the third order explicit third order Runge-Kutta scheme and SSP-DIRK3(4,3,3) scheme (27) (bottom).

fig:00

Moreover there is dissipation of surface-tension energy, that is,

$$\frac{d}{dt} \int_{\mathbb{R}} \left| \frac{\partial \omega}{\partial x} \right|^2 dx = - \int_{\mathbb{R}} \omega \left| \frac{\partial^3 \omega}{\partial x^3} \right|^2 dx,$$

and dissipation of an entropy which highlights similarities with the Boltzmann equation

$$\frac{d}{dt} \int_{\mathbb{R}} \omega \log(\omega) dx = - \int_{\mathbb{R}} \left| \frac{\partial^2 \omega}{\partial x^2} \right|^2 dx.$$

On the one hand, we compare the numerical results obtained with our numerical approximation with the similarity property of monotonicity in time of solution

$$\omega(t, x) = \frac{1}{120(t + \tau)^{1/5}} \left[r^2 - \frac{x^2}{(t + \tau)^{2/5}} \right]_+^2,$$

where $[\cdot]_+$ denotes the positive part. We have chosen $r = 2$, $\tau = 4^{-5}$ and $x \in (-2, 2)$. This solution is only $\omega \in C^1(\mathbb{R} \times \mathbb{R})$ but the second derivative in space is discontinuous, therefore we cannot expect high order accuracy. Exact and numerical solutions at various times are reported in Fig. 7.

On the other hand, we consider the same problem with a given source term

$$f(\tau, x) = \frac{1}{8\tau^4} \exp\left(-\frac{x^2}{4\tau}\right) \left(2x^2\tau^2 + (x^4 + 6\tau^2 - 9x^2\tau) \exp\left(-\frac{x^2}{4\tau}\right)\right),$$

with $\tau = t + 1$ such that the exact solution is smooth and given by $\omega_{exact}(t, x) = \exp(-x^2/4(t+1))$.

For the time discretization we apply the scheme (27) by writing the system of ODEs in the form (\mathcal{G}) with u the component treated explicitly and the v component treated implicitly:

$$\mathcal{H}(t, u, v) = -\frac{\partial}{\partial x} \left(u \frac{\partial^3 v}{\partial x^3} \right) + f(t+1, x).$$

Concerning the space discretization, we apply a second order centred finite difference scheme for the space discretization

$$\mathcal{H}_\Delta(t, u_i, v_i) = -\frac{\mathcal{F}_{i+1/2} - \mathcal{F}_{i-1/2}}{\Delta x} + f(t+1, x_i),$$

with

$$\mathcal{F}_{i+1/2} = u_{i+1/2} \frac{v_{i+2} - 3v_i + 3v_{i-1} - v_{i-2}}{\Delta x^3},$$

with $u_{i+1/2} = (u_i + u_{i+1})/2$. The time step is chosen as previously such that Δt is proportional to the space step Δx . In this way the stability condition associated to an explicit time discretisation for this problem, *i.e.* $\Delta t \leq C\Delta x^4$, is strongly violated.

The numerical error in L^1 and L^∞ for both test cases are reported in Fig. 6 at the final time $t = 0.35$. We observe a rate of convergence about 1.6 for both L^1 and L^∞ norms for the non smooth solution and second order accuracy for the smooth solution.

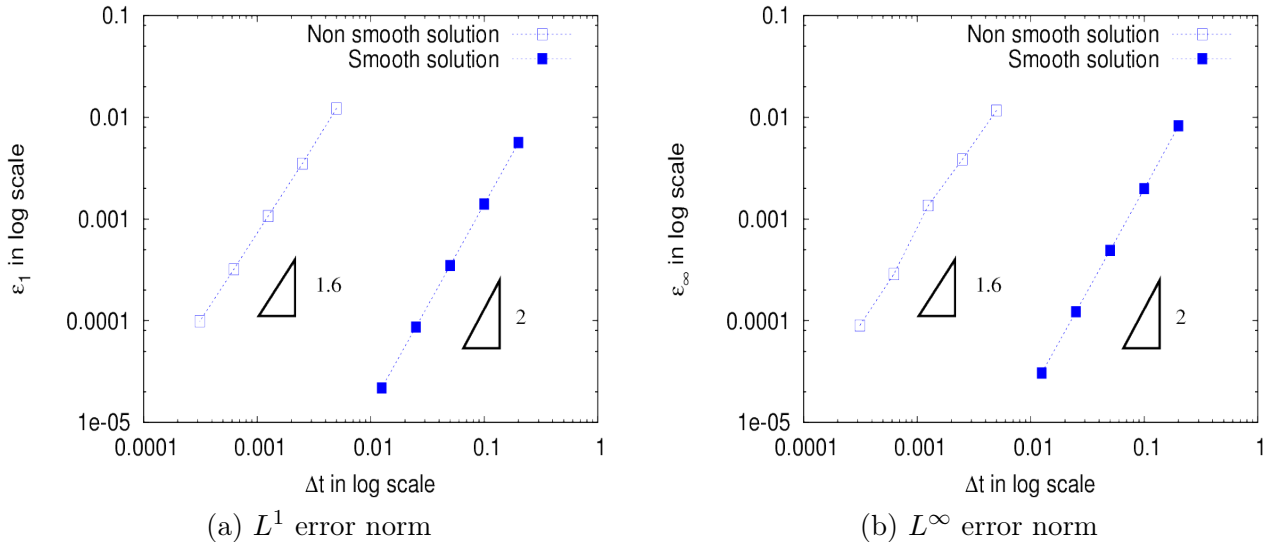


FIGURE 6. Test 4 - Hele-Shaw flow : (a) L^1 error norm and (b) L^∞ error norm for the SSP-LDIRK2(3,3,2) L-stable scheme (25).

fig:t4-0

Of course for these large time steps, the numerical scheme does not preserve positivity, but only some small spurious oscillations occur for short times and then they are damped after several time iteration thanks to the diffusion process (see Fig. 7).

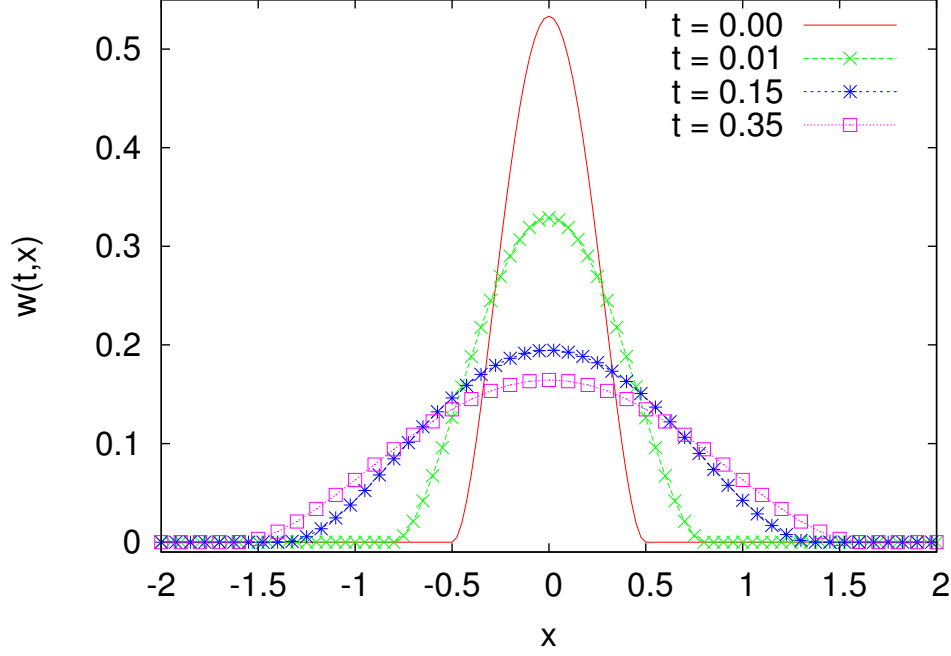


FIGURE 7. Test 4 - Hele-Shaw flow : time evolution of the numerical solution for the SSP-LDIRK3(4,3,3) L-stable scheme (27) for $t = 0, 0.01, 0.15$ and 0.35 .

fig:t4-1

3.5. **Test 5 - Surface diffusion flow.** In this section, we consider the surface diffusion of graphs [21]

$$\frac{\partial \omega}{\partial t} + \operatorname{div} S(\omega) = 0, \quad x \in \mathbb{R}^2, \quad t \geq 0,$$

where the nonlinear differential operator S is given by

$$S(\omega) := \left(Q(\omega) \left(I - \frac{\nabla \omega \otimes \nabla \omega}{Q^2(\omega)} \right) \nabla N(\omega) \right),$$

where Q is the area element

$$Q(\omega) = \sqrt{1 + |\nabla \omega|^2}$$

and N is the mean curvature of the domain boundary Γ

$$N(\omega) := \left(\frac{\nabla \omega}{Q(\omega)} \right).$$

The surface diffusion equation models the diffusion of mass within the bounding surface of a solid body, where $V = \Delta_{\Gamma} N(\omega)$ is the normal velocity of the evolving surface Γ ,

$$V = -\frac{1}{Q(u)} \frac{\partial u}{\partial t},$$

and Δ_{Γ} denotes the Laplace-Beltrami operator [21].

There are many applications of these models, such as body shape dynamics, surface construction, computer data processing or image processing. This equation is a highly nonlinear fourth-order PDE. The higher order differential operators and additional nonlinearities for these kind of problems are difficult to analyze and to simulate numerically due to the stiffness of order Δx^4 , where Δx is the space step [36]. We will apply our stable high order accurate methods based on semi-implicit time

discretizations. Moreover, we will compare our time discretization with the one proposed by P. Smereka in [35] or in [22], where the operator S is split in two parts

$$S(\omega) = \underbrace{S(\omega) - \beta \Delta^2 \omega}_{\text{less stiff part}} + \underbrace{\beta \Delta^2 \omega}_{\text{stiff, dissipative part}},$$

where β is a free parameter to be determined and in [35] it is chosen as $\beta = 2$. The first part is then treated explicitly whereas the stiff and dissipative part is treated implicitly. This splitting technique is very effective to stabilize numerical schemes but it may affect the numerical accuracy.

With our approach there is no need to add and subtract terms, because the system is automatically stabilized by the proper choice of the variable that will be implicitly treated.

The solution of the surface diffusion of graphs verifies

$$\frac{1}{2} \frac{d}{dt} \int_{\Omega} \omega^2 dx + \int_{\omega} N^2(\omega) dx = 0,$$

giving L^2 stability.

We consider numerical solutions of the two-dimensional surface diffusion of graphs equation with the initial condition

$$\omega_0(x) = \frac{1}{2\pi T} \exp\left(-\frac{|x|^2}{2T}\right).$$

The computational domain is $(-10, 10)^2$ and we use a second order central finite difference scheme together with the second order H-LDIRKp(2,2,2) scheme (25) with

$$\mathcal{H}(u, v) := \left(Q(u) \left(I - \frac{\nabla u \otimes \nabla u}{Q^2(u)} \right) \nabla \mathcal{N}(u, v) \right),$$

and \mathcal{N}

$$\mathcal{N}(u, v) := \left(\frac{\nabla v}{Q(u)} \right).$$

We present in Figure 8 the time evolution of the L^2 norm of the numerical solution and its dissipation :

$$\frac{d}{dt} \mathcal{E}(\omega) = -\mathcal{I}(t),$$

where the functional $\mathcal{E}(\omega)$ and the dissipation $\mathcal{I}(t)$ are defined by

$$\mathcal{E}(\omega) = \int_{\Omega} \omega^2(t, x) dx, \quad \mathcal{I}(t) = \int_{\Omega} N^2(\omega(t, x)) dx.$$

The results show that our second order numerical scheme (25) is stable and accurate for large time steps whereas the one based on the splitting technique given in [35] is stable but less accurate for large time step $\Delta t = 0.1$. These numerical simulations illustrate the efficiency of our approach based on semi-implicit numerical schemes.

4. LINEAR STABILITY ANALYSIS

sec:4

In this section we study linear stability properties of semi-implicit Runge-Kutta methods (12)-(13). In order to derive them, we consider an approach similar to the one given in [31]. Here we limit ourselves to a linear stability analysis for the complex scalar equation

TestP

$$(32) \quad \frac{dy}{dt} = \lambda y + \mu y, \quad y(t_0) = y_0,$$

where λ and $\mu \in \mathbb{C}$. We are aware of the limitation of this analysis with respect to the case of classical RK methods, [25]. For example, stability results for the scalar equation cannot be exported to linear systems of the form

$$\frac{dy}{dt} = \mathbf{A}y + \mathbf{B}y,$$

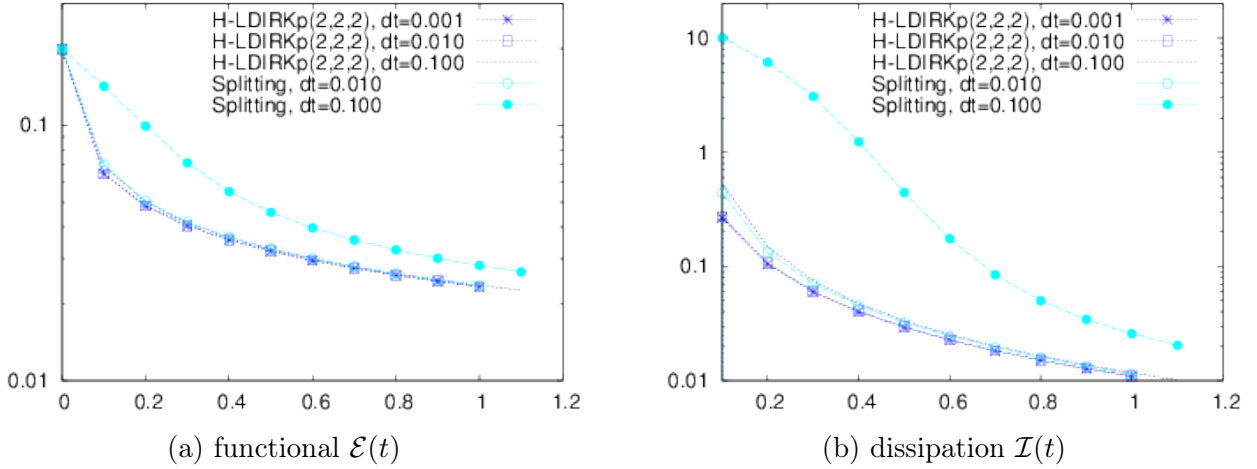


FIGURE 8. Test 5 - Surface diffusion flow. (a) Evolution of the L^2 norm and (b) the dissipation $\mathcal{I}_\Delta(t^n)$ for second order H-LDIRK $_p(2,2,2)$ L-stable scheme (25) and the one proposed in [35] based on a splitting technique in log scale.

fig:can

with \mathbf{A} and \mathbf{B} being real square matrices, unless the two matrices share a basis of eigenvectors. In spite of such limitations, we believe that linear stability gives some information useful to compare the various schemes. The stability properties of semi-implicit schemes (12)-(13) and partitioned Runge-Kutta schemes (6)-(7) are given by an absolute stability function $R(z_1, z_2)$, formalized by the following

Theorem 4.1. *Consider a semi-implicit Runge-Kutta scheme of the form (12)-(13) with $\hat{b}_i = b_i$, $i = 1, \dots, s$, applied to (32). Then, the solution can be written as*

$$y^{n+1} = R(z_1, z_2) y^n,$$

with $z_1 = \lambda \Delta t$ and $z_2 = \mu \Delta t$, and

$$(33) \quad R_{\mathcal{A}}(z_1, z_2) = 1 + (z_1 + z_2) b^T (I - z_1 \hat{A} - z_2 A)^{-1} e,$$

where I denotes the $s \times s$ identity matrix and $e = (1, \dots, 1)^T \in \mathbb{R}^s$.

Furthermore, for the additive equation (32), the stability function of the semi-implicit Runge-Kutta scheme (12)-(13) corresponds to the one of the IMEX scheme (14)-(15) and the one of the partitioned Runge-Kutta method (6)-(7) applied to (\mathcal{P}) , i.e.

$$|R_{\mathcal{A}}(z_1, z_2)| = \rho(R_{\mathcal{P}}(Z)),$$

where $\rho(R_{\mathcal{P}}(Z))$ is the spectral radius of the matrix stability function $R_{\mathcal{P}}(Z)$, $Z = \text{diag}(z_1, z_2)$, for the partitioned method (6)-(7).

Proof. We consider the scheme (12)-(13) applied to (32) and we denote $K := (K_1, \dots, K_s)^T$. Then, the stage values and the vector K can be written as

$$\begin{cases} Y = y^n e + \Delta t \hat{A} K, \\ Z = y^n e + \Delta t A K, \\ K = \lambda Y + \mu Z. \end{cases}$$

Solving for K one gets

$$K = (\lambda + \mu) y^n (I - z_1 \hat{A} - z_2 A)^{-1} e.$$

Assuming $\hat{b}_i = b_i$ for all i , the numerical solution is therefore

$$\begin{aligned} y^{n+1} &= y^n + \Delta t b^T K \\ &= \left(1 + (z_1 + z_2) b^T (I - z_1 \hat{A} - z_2 A)^{-1} e\right) y^n =: R(z_1, z_2) y^n. \end{aligned}$$

Now, we notice that an IMEX Runge-Kutta scheme with $b_i = \tilde{b}_i$ for the problem (32) gives the same the stability function

$$\text{stabIMEX} \quad (34) \quad R(z_1, z_2) = 1 + (z_1 + z_2) b^T (I - z_1 \tilde{A} - z_2 A)^{-1} e.$$

On the other hand, when we apply a semi-implicit additive Runge-Kutta scheme (14)-(15) to (32), it yields

$$\text{ZMbis} \quad (35) \quad \begin{cases} k_i = z_1 \left(y_0 + \sum_{j=1}^{i-1} \tilde{a}_{ij} k_j \right) + z_2 \left(y_0 + \sum_{j=1}^{i-1} a_{ij} k_j + z_2 a_{ii} k_i \right), \\ y_1 = y_0 + \sum_{i=1}^s b_i k_i, \end{cases}$$

which can be written in vectorial form as

$$K = z_1 (e + \tilde{A}K) + z_2 (e + AK),$$

with $K = (k_1, \dots, k_s)^T$. Then, we get that $K = (I - z_1 \tilde{A} - z_2 A)^{-1} (z_1 + z_2) e$ and substituting it in the numerical solution y_1 , we get similarly (34).

In a similar fashion, let us prove that an IMEX scheme applied to a system written in a partitioned form (\mathcal{P}) and in an additive one (\mathcal{A}) have the same stability function.

First of all, we note that the stability function corresponding to a system (\mathcal{P}) approximated by a semi-implicit method of the form (14)-(15) can be written as

$$\text{RZhong} \quad (36) \quad R_{\mathcal{A}}(z_1, z_2) = \frac{\det(I - z_1 \tilde{A} - z_2 A + (z_1 + z_2) b^T e)}{\det(I - z_2 A)}.$$

Now we rewrite the system (32) in partitioned form

$$\text{ParTest} \quad (37) \quad \begin{cases} \frac{du}{dt} = \lambda(u + v), \\ \frac{dv}{dt} = \mu(u + v), \end{cases}$$

where $y = u + v$. Applying to (37) a partitioned Runge-Kutta method (6)-(7) combining an explicit (\tilde{A}, \tilde{b}) and an implicit RK method (A, b) , we read

$$\begin{cases} U_i = u_0 + h \sum_{j=1}^{i-1} \tilde{a}_{ij} k_j, & V_i = v_0 + h \sum_{j=1}^i a_{ij} \ell_j, \\ k_i = \lambda(U_i + V_i), & \ell_i = \mu(U_i + V_i), \\ u_1 = u_0 + h \sum_{i=1}^s \tilde{b}_i k_i, & v_1 = v_0 + h \sum_{i=1}^s b_i \ell_i. \end{cases}$$

By algebraic manipulations we can write the matrix stability function

$$R_{\mathcal{P}}(Z) = I_2 + Z B A \mathcal{E},$$

with $Z = h\Lambda$ where

$$I_2 = \begin{pmatrix} I_s & 0 \\ 0 & I_s \end{pmatrix}, \quad \mathcal{E} = \begin{pmatrix} e_s & 0 \\ 0 & e_s \end{pmatrix},$$

with I_s identical matrix of $s \times s$ dimension and $e_s = \underbrace{(1, \dots, 1)}_s^T$,

$$\Lambda = \begin{pmatrix} \lambda & 0 \\ 0 & \mu \end{pmatrix}, \quad B = \begin{pmatrix} \tilde{b}^T & \tilde{b}^T \\ b^T & b^T \end{pmatrix}$$

and

$$\mathcal{A} = \begin{pmatrix} I_s - z_1 \tilde{A} & -z_2 \tilde{A} \\ -z_2 A & I_s - z_2 A \end{pmatrix}$$

where $\tilde{A} = (\tilde{a}_{ij})$ with $\tilde{a}_{ij} = 0$ for $j \geq i$, and $A = (a_{ij})$ with $a_{ij} = 0$ for $j > i$, are $s \times s$ matrices. Now Looking for the eigenvalues and computing the spectral radius of $R_{\mathcal{P}}(Z)$ we find two values where one is equal to 1 and the other one is

$$R_{\mathcal{P}}(z_1, z_2) = \frac{\det(I - z_1 \tilde{A} - z_2 A + z_1 \tilde{b}^T e + z_2 b^T e)}{\det(I - z_2 A)}.$$

Under the conditions $b_i = \tilde{b}_i$ for all i , $R_{\mathcal{P}}(z_1, z_2)$ coincides exactly with (36). \square

The A -stability region of a semi-implicit RK scheme in the complex plane is defined as

$$S_A = \{(z_1, z_2) \in \mathbb{C}^2 : |R(z_1, z_2)| \leq 1\}.$$

In this paper first we consider the definition of stability region given in [29], namely

$$(38) \quad S_1 = \{z_1 \in \mathbb{C} : \sup_{z_2 \in \mathbb{C}^-} |R(z_1, z_2)| \leq 1\},$$

which has the following motivations: we look for a simpler stability region which is a subset of \mathbb{C} rather than \mathbb{C}^2 . Since the implicit part of the method is A -stable, it is reasonable to look for the region $z_1 \in \mathbb{C}$ such that the scheme remains A -stable, *i.e.* $|R(z_1, z_2)| \leq 1$ for all $z_2 \in \mathbb{C}^-$.

Alternatively, one can insist on using the full stability region of the explicit method S_E , with boundary Γ_E , and look for the region $z_2 \in \mathbb{C}$ such that the scheme is stable for all $z_1 \in S_E$, giving rise to the definition

$$(39) \quad S_2 = \{z_2 \in \mathbb{C} : \sup_{z_1 \in S_E} |R(z_1, z_2)| \leq 1\}.$$

In this way we look for a simpler stability region which is a subset of \mathbb{C} rather than \mathbb{C}^2 .

Both definitions can be simplified by applying maximum principle of analytic functions [29]. For a given $z_2 \in \mathbb{C}$, the $\sup R(z_1, z_2)$ is obtained for some $z_1^* \in \Gamma_E$, therefore the definition of S_2 reduces to

$$(40) \quad S_2 = \{z_2 \in \mathbb{C} : \max_{z_1 \in \Gamma_E} |R(z_1, z_2)| \leq 1\}.$$

This observation simplifies the problem considerably.

Some of the schemes proposed in this paper are L -stable in the implicit part which means that $R(z_1, z_2) \rightarrow 0$ as $z_2 \rightarrow \infty$, and this implies that the coefficients of the semi-implicit scheme (35) satisfy the following condition (see [38] for details)

$$(41) \quad 1 + \sum_{i=1}^s b_i \beta_i = 0, \quad \text{where} \quad \beta_i = -\frac{1}{a_{ii}} \left(1 + \sum_{j=1}^{i-1} a_{ij} \beta_j \right), \quad i = 1, \dots, s.$$

Note that the coefficients of the schemes H-LSDIRK $p(2,2,2)$, LSDIRK2(2,2,2) and SSP-SDIRK2(3,3,2) satisfy (41), while those of schemes H-CN(2,2,2) and H-SDIRK2(2,2,2), SSP-LDIRK3(4,3,3) do not.

In the next figures we show the S_1 (blue) and S_2 (green) stability regions of the schemes adopted in the paper. Some of the S_1 regions appeared in [11], and we reproduce them here for a more immediate comparison with the corresponding region S_2 .

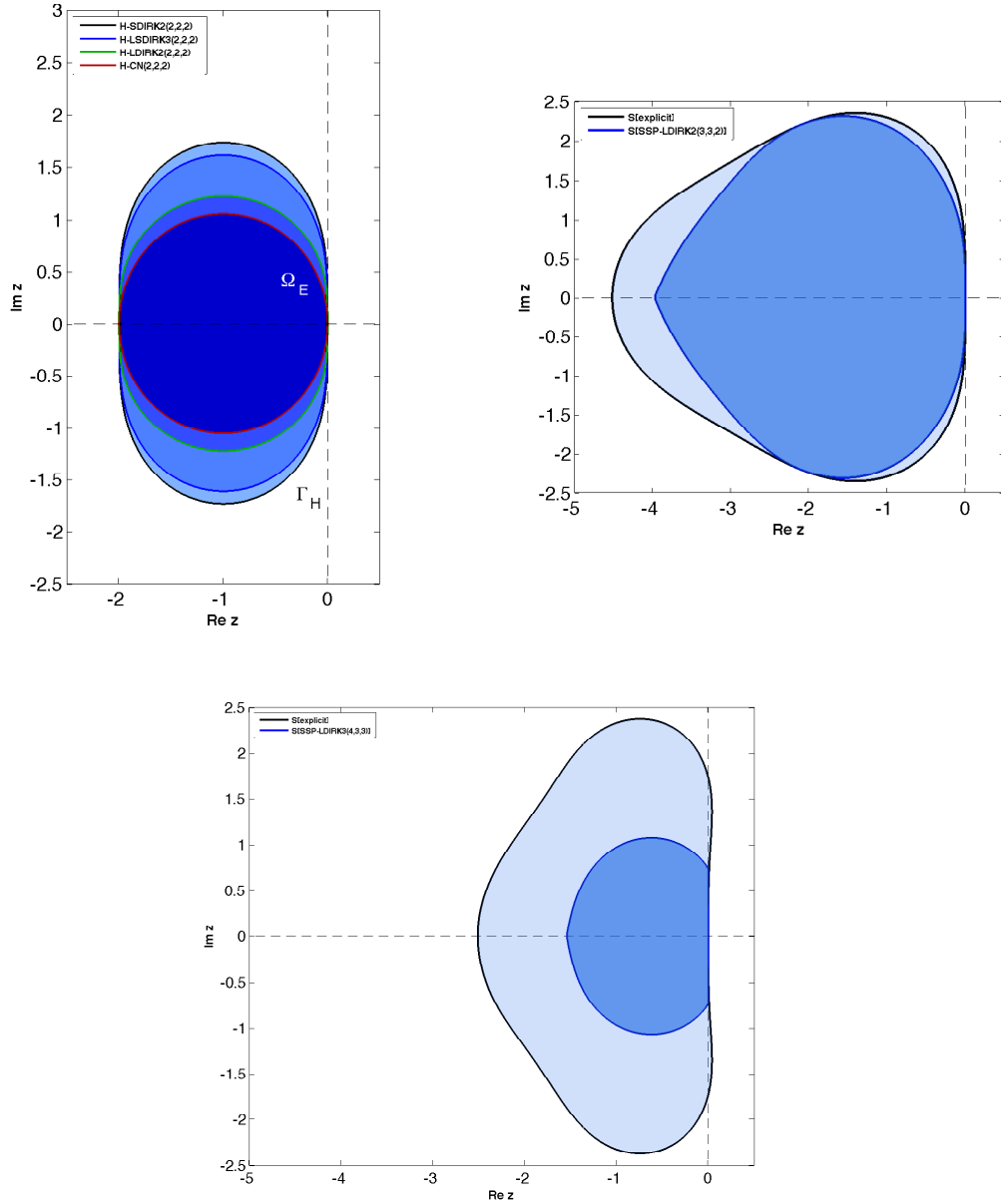


FIGURE 9. Stability regions S_1 of the IMEX schemes. Left panel: (2,2,2) schemes H-SDIRK2, H-LSDIRK3, H-LSDIRK2 and H-CN; right panel: scheme SSP-LDIRK2(3,3,2), and the corresponding explicit RK method when $z_2 = 0$. Below scheme SSP-LDIRK3(4,3,3), and the corresponding explicit RK method when $z_2 = 0$.

figStab

In the panels of Figure 9 we observe that in most cases the stability regions S_1 of the semi-implicit scheme is smaller than the corresponding stability region S_E of the explicit part. An exception is given by scheme H-SDIRK2(2,2,2), for which $S_1 = S_E$. The stability region of LSDIRK2(2,2,2) scheme coincides with the one of scheme H-LDIRK2(2,2,2).

On the contrary if we consider definition (39), i.e. if we impose that the scheme is stable for all $z_1 \in \mathbb{C}$, then z_2 has to be restricted to the set S_2 defined in (40) and this implies that the stability region S_2 (dark green $|R(z_1, z_2)| \leq 1$) of the scheme in general reduces with respect to the stability

regions of implicit methods

$$S_I = \{z_2 \in \mathbb{C} : |R(0, z_2)| \leq 1\}.$$

(light green $|R(0, z_2)| \leq 1$) as shown in the Figures 10.

An exception is given once again by scheme H-SDIRK2(2,2,2), for which $S_2 = S_I = \mathbb{C}^-$ (we omit to plot such region).

We do not plot the stability region of schemes H-CN(2,2,2) and SSP-LDIRK3(4,3,3) because they are reduced to the origin, and the stability region of scheme LSDIRK2(2,2,2) because it coincides with the one of scheme H-LDIRK2(2,2,2).

5. CONCLUSIONS

In this paper we show that classical IMEX schemes can be adopted in a much wider context than the one they were originally introduced for. Indeed, in several contexts, the stiffness of the problem is essentially linear, and therefore IMEX schemes can be successfully adopted by treating the linear stiff part implicitly. Note that this approach is not equivalent to a linearization of the problem, and in several cases it is much simpler to apply than linearization. The overall accuracy of the scheme is automatically guaranteed by the standard order conditions for IMEX schemes. Many practical examples showing how to apply IMEX schemes in this new context are presented. We believe that this new approach can be successfully applied in many other contexts well beyond the ones presented in the paper.

6. ACKNOWLEDGEMENTS

Francis Filbet is partially supported by the European Research Council ERC Starting Grant 2009, project 239983-*NuSiKiMo* and the french ANR project STAB. Giovanni Russo and Sebastiano Boscarino have been partially supported by Italian PRIN 2009 project "Innovative numerical methods for hyperbolic problems with application to fluid dynamics, kinetic theory, and computational biology", prot. n.2009588FHJ.

REFERENCES

- ARS [1] U. Ascher, S. Ruuth, and R.J. Spiteri, *Implicit-explicit Runge-Kutta methods for time dependent partial differential equations*, Appl. Numer. Math. **25** (1997), 151–167.
- ABS [2] A. Araújo, S. Barbeiro, P. Serranho *Stability of finite difference schemes for nonlinear complex reaction-diffusion processes*, IMA Journal of Numerical Analysis, (2014), 1-22
- BLT [3] C. Berthon, P.G. LeFloch, and R. Turpault, *Late-time relaxation limits of nonlinear hyperbolic systems. A general framework*, Math. of Comput. (2012).
- ref3 [4] A.L. Bertozzi, *The mathematics of moving contact lines in thin liquid films*, Notices of the AMS **45**, 689-697, (1998).
- ref4 [5] A.L. Bertozzi, M. Pugh, *The lubrication approximation for thin viscous films: regularity and long-time behavior of weak solutions*, Comm. Pure Appl. Math. **XLIX**, 85–123, (1996).
- ChatardFilbet [6] M. Bessemoulin-Chatard and F. Filbet, *A finite volume scheme for nonlinear degenerate parabolic equations*, SIAM J. Scientific Computing, vol. **34**, no. 5 (2012), pp. 559–583,
- BLR [7] S. Boscarino, P.G. LeFloch, and G. Russo, *High-order asymptotic-preserving methods for fully non linear relaxation problems*, Submit to SISC.
- BRhonom [8] S. Boscarino, G Russo *High-Order Asymptotic-Preserving Methods for Nonlinear Relaxation from Hyperbolic Systems to Convection-Diffusion Equations* submitted to Proceedings, High Order Nonlinear Numerical Methods for Evolutionary PDEs (HONOM 2013) , March 18-22, 2013, Bordeaux.
- Bosc-07 [9] S. Boscarino, *Error analysis of IMEX Runge-Kutta methods derived from differential algebraic systems*, SIAM J. Numer. Anal. **45** (2007), 1600–1621.
- Bosc-09 [10] S. Boscarino, *On an accurate third order implicit-explicit Runge-Kutta method for stiff problems*, Appl. Num. Math. **59** (2009), 1515–1528.
- BBMRV [11] S. Boscarino, R. Bürger, Pep Mulet, G. Russo and L. M. Villada *Linearly Implicit IMEX Runge-Kutta Methods for a class of degenerate convection-diffusion problems*, SIAM J. Sci. Comput., accepted.
- BPR [12] S. Boscarino, L. Pareschi, and G. Russo, *Implicit-explicit Runge-Kutta schemes for hyperbolic systems and kinetic equations in the diffusion limit*, SIAM J. Sc. Comp.

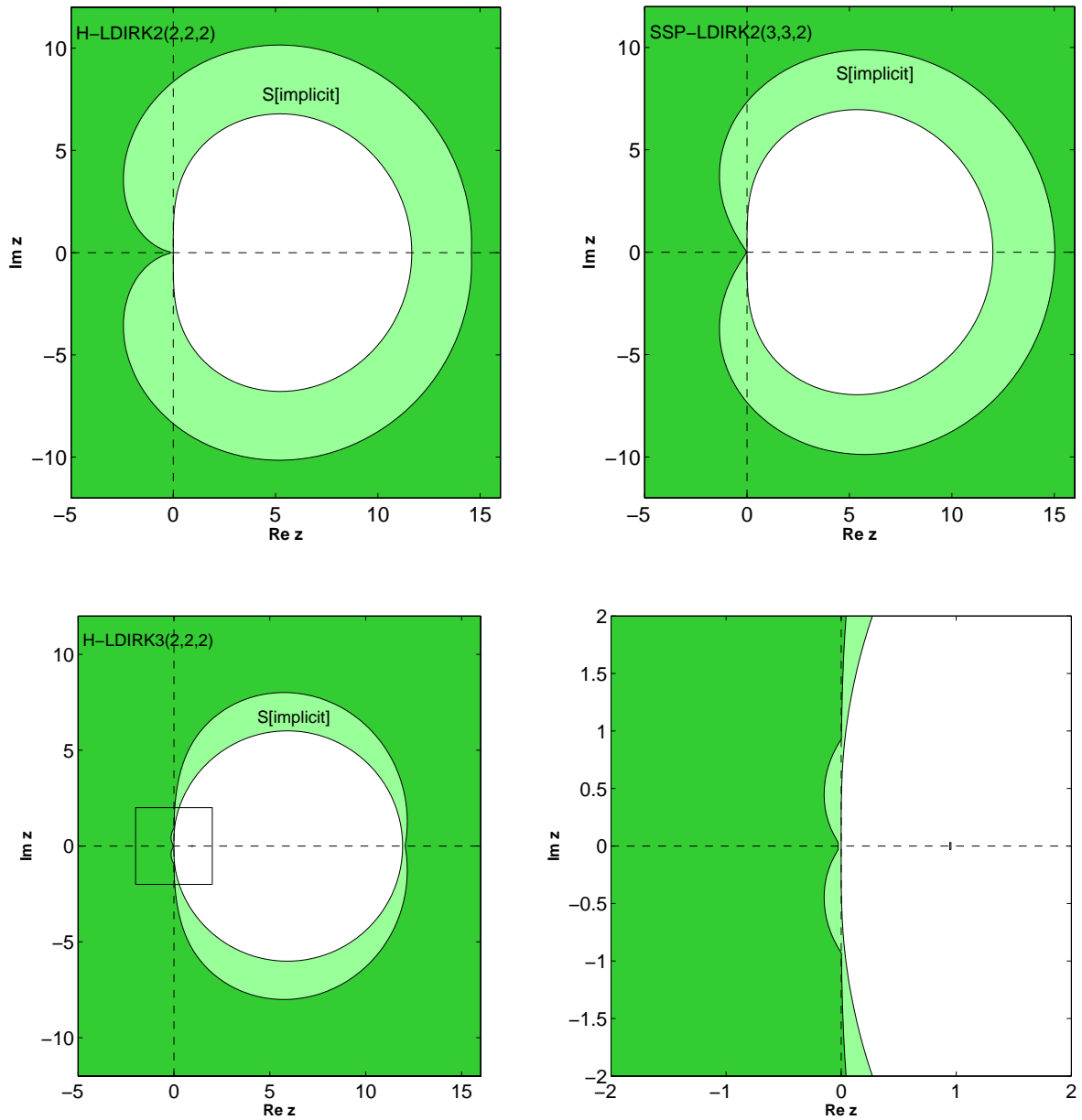


FIGURE 10. Stability regions S_2 of the second order IMEX schemes, H-LDIRK2(2,2,2), SSP-LDIRK2(3,3,2), and H-LDIRK3(2,2,2), in the same figures the corresponding stability regions of the implicit RK methods S_I . The last panel is a zoom of the small square region near the origin.

figStab2

- Bosc-10** [13] S. Boscarino and G. Russo, *On a class of uniformly accurate IMEX Runge-Kutta schemes and application to hyperbolic systems with relaxation*, SIAM J. Sci. Comput. **31** (2009), 1926–1945.
- BR** [14] S. Boscarino and G. Russo, *Flux-explicit IMEX Runge-Kutta schemes for hyperbolic to parabolic relaxation problems*, SIAM J. Num. Anal.
- CaJiRu** [15] R. E. Caflisch, S. Jin, G. Russo, *Uniformly Accurate Schemes for Hyperbolic Systems with Relaxation*, SIAM J. Numer. Anal. **34**, No 1, pp. 246–281 (2001).
- CK** [16] M.H. Carpenter and C.A. Kennedy, *Additive Runge-Kutta schemes for convection-diffusion-reaction equations*, Appl. Numer. Math. **44** (2003), 139–181.

- rrillo2009 [17] J.A. Carrillo, Ph. Laurençot and J. Rosado, *Fermi-Dirac-Fokker-Planck equation: Well-posedness & long-time asymptotics*, Journal of Differential Equations, **247**, pp. 2209–2234, (2009),
- rrillo2008 [18] J.A. Carrillo, J. Rosado and F. Salvarani, *1D nonlinear Fokker-Planck equations for fermions and bosons*, Applied Mathematics Letters, **21**, pp. 148–154, (2008)
- PN [19] F. Cavalli, G. Naldi, G. Puppo, and M. Semplice, *High order relaxation schemes for nonlinear diffusion problems*, SIAM J. Numer. Anal. **45** (2007), 2098–2119.
- CLL [20] C. Q. Chen, C. D. Levermore and T. P. Liu *Hyperbolic conservation laws with relaxation terms and entropy*, Comm. Pure Appl. Math., **47**, pp. 787–830, (1994).
- ref15 [21] K. Deckelnick, G. Dziuk and C.M. Elliott, *Computation of geometric partial differential equations and mean curvature flow*, Acta Numer. **14**, 139–232 (2005)
- FJ [22] F. Filbet and S. Jin, *A class of asymptotic-preserving schemes for kinetic equations and related problems with stiff sources* J. Comput. Phys. **229** (2010), pp. 7625–7648.
- GST [23] S. Gottlieb, C.W. Shu, and E. Tadmor, *Strong stability-preserving high-order time discretization methods*, SIAM Rev., **43** (2001), pp. 89–112.
- irer_error [24] Hairer, E. and Lubich, C. and Roche, M., *Error of Runge-Kutta methods for stiff problems studied via differential algebraic equations*, BIT Numerical Mathematics, V. 28, n. 3, pages 678–700, 1988.
- HNL [25] E. Hairer, S.P. Norsett, and G. Wanner, *Solving Ordinary Differential Equation. I. Non-stiff problems*, Springer Series in Comput. Mathematics, Vol. 8, Springer Verlag, (2nd edition), 1993.
- HW [26] E. Hairer and G. Wanner, *Solving Ordinary Differential Equation. II. Stiff and differential algebraic problems*, Springer Series in Comput. Mathematics, Vol. 14, Springer Verlag, (2nd edition), 1996.
- HLW [27] E. Hairer C. Lubich and G. Wanner, *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*, Springer Series in Comput. Mathematics, vol. **31**, Springer Verlag, (2nd edition), 2006.
- Jin [28] S. Jin, *Runge-Kutta Methods for Hyperbolic Conservation Laws with Stiff Relaxation Terms* J. Comp. Phys. **122**, pp. 51–67, (1995).
- LRR [29] S. F. Liotta, V. Romano and G. Russo *Central schemes for balance laws of relaxation type*, SIAM J. Num. Anal., Vol. 38, N. 4, (2000) pp. 1337-356
- 075 [30] J. Oliver (1975), *A curiosity of low-order explicit Runge-Kutta methods*, Math. Comp., Vol.29, p.1032-1036.
- PaRu00 [31] L. Pareschi, G. Russo *Implicit-Explicit Runge-Kutta schemes for stiff systems of differential equations*. Recent trends in numerical analysis, pp. 269–288, Adv. Theory Comput. Math. **3**, Nova Sci. Publ. Huntington, NY, (2001).
- PaRu0 [32] L Pareschi, G. Russo, *High order asymptotically strong stability preserving methods for hyperbolic systems with stiff relaxation*. Hyperbolic problems: theory, numerics, applications, pp. 241–251, Springer, Berlin, (2003).
- PaRu [33] L. Pareschi and G. Russo, *Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxations*, J. Sci. Comput. (2005), 129–155.
- PRT [34] L. Pareschi, G. Russo, G. Toscani, *A kinetic approximation to Hele-Shaw flow*, C. R. Math., Acad. Sci. Paris, Ser. I **338**, No. 2, pp. 178-182, (2004).
- Sme [35] P. Smereka, *Semi-implicit level set methods for curvature and surface diffusion motion* Journal of Scientific Computing, Volume: 19 Issue: 1-3, pp. 439-456
- huWillmore [36] Y. Xu and C.W. Shu, *Local Discontinuous Galerkin Method for Surface Diffusion and Willmore Flow of Graphs*, **40**, pp. 375–390 (2009).
- ZW [37] K. Zhang, J.C.F. Wong, R. Zhang, *Second-order implicit-explicit scheme for the Gray-Scott model*, J. Comput. Appl. Math. **213** (2008) pp. 559-581.
- Zhong [38] X. Zong, *Additive Semi-Implicit Runge-Kutta methods for computing high-speed non equilibrium reactive flows*, J. Comput. Phys. **128** (1996), 19–31.

SEBASTIANO BOSCARINO, DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, UNIVERSITY OF CATANIA,
VIA A.DORIA 6, 95125 CATANIA, ITALY
E-mail address: boscarino@dmf.unict.it

FRANCIS FILBET, UNIVERSITÉ DE LYON, CNRS UMR 5208, UNIVERSITÉ LYON 1, INSTITUT CAMILLE JORDAN,
43 BLVD. DU 11 NOVEMBRE 1918, F-69622 VILLEURBANNE CEDEX, FRANCE.
E-mail address: filbet@math.univ-lyon1.fr

GIOVANNI RUSSO, DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, UNIVERSITY OF CATANIA, VIA
A.DORIA 6, 95125 CATANIA, ITALY
E-mail address: russo@dmf.unict.it