

Second order fully semi-Lagrangian discretizations of advection–diffusion–reaction systems

Luca Bonaventura · Elisa Calzola ·
Elisabetta Carlini · Roberto Ferretti

Received: date / Accepted: date

Abstract We propose a second order, fully semi-Lagrangian method for the numerical solution of systems of advection–diffusion–reaction equations, which is based on a semi-Lagrangian approach to approximate in time both the advective and the diffusive terms. The proposed method allows to use large time steps, while avoiding the solution of large linear systems, which would be required by implicit time discretization techniques. Standard interpolation procedures are used for the space discretization on structured and unstructured meshes. A novel extrapolation technique is proposed to enforce second-order accurate Dirichlet boundary conditions. We include a theoretical analysis of the scheme, along with numerical experiments which demonstrate the effectiveness of the proposed approach and its superior efficiency with respect to more conventional explicit and implicit time discretizations.

Keywords Advection–diffusion–reaction systems · Semi-Lagrangian methods · Implicit methods · Second-order methods

Mathematics Subject Classification (2010) 35L10 · 65M06 · 65M25 · 65M12

L. Bonaventura
MOX – Modelling and Scientific Computing,
Dipartimento di Matematica, Politecnico di Milano
Via Bonardi 9, 20133 Milano, Italy
E-mail: luca.bonaventura@polimi.it

E. Calzola, E. Carlini
Dipartimento di Matematica
“Sapienza” Università di Roma
P.le Aldo Moro 5, 00185, Roma, Italy
E-mail: calzola@mat.uniroma1.it, carlini@mat.uniroma1.it

R. Ferretti (corresponding author)
Dipartimento di Matematica e Fisica
Università degli Studi Roma Tre
L.go S. Leonardo Murialdo 1, 00146, Roma, Italy
E-mail: ferretti@mat.uniroma3.it

1 Introduction

Systems of advection–diffusion–reaction (ADR) equations model the chemical or biochemical processes involving several species transported by a fluid. These systems are responsible for most of the computational cost of typical environmental fluid dynamics models, such as those applied in climate, water and air quality and oceanic biogeochemistry modeling for long term simulations [6], [16], [17]. Also in applications to medium range weather forecasting, which consider shorter time ranges, the number of interacting transported species can be quite large. This implies that a very large number of ADR equations have to be solved simultaneously, in order to achieve a complete description of the relevant physical processes. As a consequence, even minor efficiency gains in the solution of this very classical problem are of paramount practical importance. This explains why numerical methods that allow the use of large time steps are favoured for these applications, see e.g. the discussion in [44]. The standard ways to enhance efficiency for the solution of advection step are either the use of implicit schemes or the application of semi-Lagrangian (SL) techniques, [18], [39]. These are then coupled to implicit methods for the diffusion and reaction step. As discussed in [16], [17], SL methods have the advantage that all the computational work that makes them computationally more expensive per time step than standard Eulerian techniques is indeed independent of the number of tracers, which allows to achieve easily a superior efficiency level in the limit of a large number of tracers.

In the recent papers [4], [5], a fully SL approach to both the advection and diffusion step was pursued, which combines the standard SL treatment of advection with SL-like schemes for diffusion. In particular, it was shown in [5] that, even for a single advection–diffusion equation, the fully SL approach can be more efficient than standard implicit techniques. SL methods for parabolic, second order problems have also been presented, among others, in [10], [20], [30], [31], [32], [15], [1], [34]. A complete review of the earlier literature on this topic can be found in [18], [27]. We remark that, among the proposals in the literature, the formulation first introduced in [4] is unique since it allows to treat straightforwardly parabolic problems in divergence form, such as are usually encountered in computational fluid dynamics applications.

Since the technique under consideration stems from the Feynman–Kac stochastic representation formula, it could also be possible to mix SL schemes with a MonteCarlo approach, as proposed, for example, in [8]. However, while this latter strategy might be more scalable on massively parallel architectures, on more conventional platforms it suffers from a slow convergence with respect to the number of sample trajectories. By exploiting the concept of *weak convergence* of schemes for Stochastic Differential Equations, the deterministic approach pursued here usually results in a lower computational complexity.

In the present work, we present a number of improvements to the fully SL approach of [4], [5]. In particular, we show how second order accuracy in time can be achieved. An improved treatment of Dirichlet boundary conditions is also discussed and analysed. The resulting approach yields an efficient combination, which is validated on a number of classical benchmarks, on both structured and unstructured meshes. Numerical results show that the method yields good quantitative agreement with reference numerical solutions, while being superior in efficiency to

standard implicit methods and to approaches in which the SL method is only used for the advection term.

The outline of the paper is the following. In Section 2, the considered model problems are introduced. Section 3 describes the SL advection–diffusion solver. A stability and convergence analysis of the method is outlined in Section 4. The possible approaches to the treatment of boundary conditions are discussed in Section 5. A numerical validation of the proposed approach on both structured and unstructured meshes is presented in Section 6, while some conclusions and perspectives for future developments are outlined in Section 7.

2 The model problem

We consider as a model problem the advection–diffusion–reaction equation with Dirichlet boundary conditions

$$\begin{cases} c_t + u \cdot \nabla c - \nu \Delta c = f(c) & (x, t) \in \Omega \times (0, T], \\ c(x, t) = b(x, t) & (x, t) \in \partial\Omega \times (0, T], \\ c(x, 0) = c_0(x) & x \in \Omega. \end{cases} \quad (1)$$

Here, T denotes the final time, $\Omega \subset \mathbb{R}^2$ is an open bounded domain, $u : \Omega \times [0, T] \rightarrow \mathbb{R}^2$ is a velocity field and $b : \partial\Omega \times [0, T] \rightarrow \mathbb{R}$ denotes the boundary value of the species c . The unknown $c : \Omega \times [0, T] \rightarrow \mathbb{R}$ can be interpreted as the concentration of a chemical species that is transported through the domain Ω by the advection and diffusion processes, while undergoing locally a nonlinear evolution determined by the source term $f(c)$, which will be assumed to be globally Lipschitz continuous, with Lipschitz constant L_f .

In the simpler case of homogeneous boundary conditions and time independent advection field and diffusion coefficient, equation (1) can be written as

$$c_t = \mathcal{L}c + f(c), \quad (2)$$

where \mathcal{L} denotes a linear differential operator. We denote by \mathcal{E}_t the evolution operator determining the solution of the associated homogeneous equation

$$\tilde{c}_t = \mathcal{L}\tilde{c} \quad (3)$$

with the same initial datum $\tilde{c}(x, 0) = c_0(x)$ and boundary conditions as in (1), so that $\tilde{c}(t) = \mathcal{E}_t[c_0]$. By formal application of the variation of constants formula, the solution of (2) can then be represented as

$$c(x, t) = \mathcal{E}_t[c_0](x) + \int_0^t \mathcal{E}_{t-s}[f \circ c](x) \, ds. \quad (4)$$

If discrete time levels $t_n, n = 0, \dots, N$ are introduced, so that $t_n = n\Delta t$ and $\Delta t = T/N$, the same representation formula on the interval $[t_n, t_{n+1}]$ reads

$$c(x, t_{n+1}) = \mathcal{E}_{\Delta t}[c(\cdot, t_n)](x) + \int_{t_n}^{t_{n+1}} \mathcal{E}_{t_{n+1}-s}[f \circ c](x) \, ds. \quad (5)$$

The construction of the scheme relies on the application to (5) of the Feynman–Kac formula to represent the solution of (3) (see, e.g., [20]), so that

$$\mathcal{E}_{\Delta t}[c(\cdot, t_n)](x) = \mathbb{E}\{c(X(t_n), t_n)\} \quad (6)$$

where \mathbb{E} denotes the probabilistic expectation w.r.t. the Wiener measure, and $X(t)$ is the solution of the stochastic differential equation (SDE):

$$\begin{cases} dX = -u(X(s), s)ds + \sqrt{2\nu}dW(s), \\ X(t_{n+1}) = x, \end{cases} \quad (7)$$

for $s \in [t_n, t_{n+1}]$, with $W(s)$ denoting a standard 2-dimensional Wiener process. Note that, for $\nu = 0$, (7) reduces to a deterministic ODE and the evolution operator (6) can be approximated accordingly by the well-known method of characteristics for transport problems.

While the proposed numerical method will be presented in this simpler case, the target for more realistic applications are systems of coupled advection–diffusion–reaction equations of the form

$$\begin{cases} (c_k)_t + u \cdot \nabla c_k - \nabla \cdot (D \nabla c_k) = f_k(c_1, \dots, c_S) & (x, t) \in \Omega \times (0, T], \\ c_k(x, t) = b_k(x, t) & (x, t) \in \partial\Omega \times (0, T], \\ c_k(x, 0) = c_{0,k}(x) & x \in \Omega, k = 1, \dots, S. \end{cases} \quad (8)$$

Here, D denotes a symmetric and positive semi-definite diffusivity tensor, possibly dependent on space and time. As remarked in Section 1, systems of this kind, with a possibly large number of species S , are responsible for the largest share of the computational cost of typical environmental fluid dynamics models, so that even minor increases in the efficiency of the discretization for this very classical problem are of great practical relevance.

Notice that, while the use of a representation formula like (4) may recall the procedure that it is followed to introduce exponential integrators (EI) (see e.g. the review in [25]), there are substantial differences between SL and EI methods. For example, EI are based on the approximation of a representation formula for the solutions of a spatially discretized problem, while SL methods employ a space-time representation formula. Furthermore, SL methods approximate the evolution operator by a local approach based on trajectory computation, while standard EI entail a global step for the computation of the matrix exponential, which is computationally quite demanding, see e.g. the discussion in [24].

3 Fully semi-Lagrangian methods

A numerical method for the solution of equation (1) on the interval $[t_n, t_{n+1}]$ can then be derived heuristically from (5) by discretizing the time integral using the trapezoidal rule, so that one obtains

$$c(x, t_{n+1}) \approx \mathcal{E}_{\Delta t}[c(\cdot, t_n)](x) + \frac{\Delta t}{2} [\mathcal{E}_{\Delta t}[f \circ c](x) + f(c(x, t_{n+1}))]. \quad (9)$$

If the diffusion term is dropped in equation (1), and the evolution operator is approximated by a numerical version of the flow streamline together with an interpolation at the departure point of the streamline, a numerical method based on

(9) can be interpreted as a semi-Lagrangian extension of the trapezoidal rule with global truncation error of second order. Semi-Lagrangian methods based on this formula have been successfully used in a large number of applications (see, among many others, [2],[13],[14],[40],[41],[43]). Due to a possible stiffness of the reaction term, we might rather use a first order, off-centered version of the above formula, defined, for $\theta \in [1/2, 1]$ as

$$c(x, t_{n+1}) \approx \mathcal{E}_{\Delta t}[c(\cdot, t_n)](x) + (1 - \theta)\Delta t \mathcal{E}_{\Delta t}[f \circ c](x) + \theta \Delta t f(c(x, t_{n+1})). \quad (10)$$

In order to discretize (9) (or (10)), we introduce a space mesh $\mathcal{G}_{\Delta x} = \{x_i : x_i \in \Omega\}$, where Δx denotes the mesh resolution. The mesh can be structured as well as unstructured; the only necessary restriction is that it should be possible to define a piecewise polynomial interpolation operator I_p of degree p , constructed on the values of a grid function c defined on $\mathcal{G}_{\Delta x}$ (we refer to [36] for a precise definition of the general setting). We denote by $I_p[c^n](x)$ the value at x of the interpolant I_p computed using the values of the grid function c^n . The vector c^n collects the values c_i^n of the numerical solution of (1) at the space-time mesh nodes (x_i, t_n) .

The discretization of (7), whenever aimed at approximating the expectation in (6), is performed via the so-called *weak schemes* for SDEs (see the classical review [26]). At a generic node $x = x_i$, weak schemes approximate the expectation in (6) as

$$\mathbb{E}\{c(X(t_n), t_n)\} = \sum_k \alpha_k c(z_{k,i}^{n+1}, t_n) + O(\Delta t^r) \quad (11)$$

for a suitable definition of the points $z_{k,i}^{n+1}$ of the weights α_k . For our purposes, we will consider cases in which $r = 1, 2$, and set

$$z_{k,i}^{n+1} = x_i + \delta_{k,i}^{n+1}.$$

In the simplest case, a two-dimensional, first-order weak scheme ($r = 1$) which generalizes the explicit Euler scheme, may be obtained for

$$\delta_{k,i}^{n+1} = -\Delta t u(x_i, t_{n+1}) + \sqrt{4\Delta t \nu} d_k$$

for $k = 1, \dots, 4$, with $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 1/4$, and

$$d_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad d_2 = -\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad d_3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad d_4 = -\begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The discrete set of displacements $\sqrt{4\Delta t \nu} d_k$ and weights α_k ($k = 1, \dots, 4$) is constructed (see [26]) in order to approximate the probability density of the 2-dimensional Gaussian random variable

$$\sqrt{2\nu} \Delta W := \sqrt{2\nu} (W(\Delta t) - W(0))$$

with the discrete density

$$P\left(\sqrt{2\nu} \Delta W = \sqrt{4\Delta t \nu} d_k\right) = \alpha_k, \quad (k = 1, \dots, 4)$$

up to a certain number of moments. More precisely, in this first-order case they coincide up to the third moment (note that odd moments are always zero by symmetry).

Introducing the space interpolation, and replacing (6) with its discretization (11), a first order in time approximation c^n of the solution of (1) can then be defined as

$$\begin{aligned} c_i^{n+1} &= \frac{1}{4} \sum_{k=1}^4 I_p[c^n](z_{k,i}^{n+1}) \\ &+ (1-\theta)\Delta t \frac{1}{4} \sum_{k=1}^4 f(I_p[c^n])(z_{k,i}^{n+1}) + \theta\Delta t f(c_i^{n+1}), \end{aligned} \quad (12)$$

Notice that, for simplicity, we neglect in (12) the treatment of boundary conditions. Possible approaches to handle Dirichlet boundary conditions will be discussed in Section 5.

It is easy to show that (12) has a unique solution for Δt small enough. In fact, (12) is in the form of a set of decoupled fixed point equations for the unknowns c_i^{n+1} ,

$$c_i^{n+1} = F_i(c^n) + \theta\Delta t f(c_i^{n+1}) \quad (13)$$

and the Lipschitz constant of the right-hand side is $\theta\Delta t L_f$. Therefore, the right-hand side is a contraction as soon as $\Delta t < \frac{1}{\theta L_f}$, regardless of the Courant number. Moreover, since

$$\begin{aligned} |f(\gamma)| &\leq |f(\gamma) - f(0)| + |f(0)| \\ &\leq L_f |\gamma| + |f(0)| \end{aligned}$$

we can obtain from (13)

$$|F_i(c^n) + \theta\Delta t f(\gamma)| \leq |F_i(c^n)| + \theta\Delta t (L_f |\gamma| + |f(0)|)$$

(note that the right-hand side is increasing with $|\gamma|$). In order to obtain an invariant set of the form $|\gamma| \leq R$, we should therefore satisfy the condition

$$|F_i(c^n)| + \theta\Delta t (L_f R + |f(0)|) \leq R,$$

which gives, provided $\theta\Delta t L_f < 1$,

$$R \geq \frac{|F_i(c^n)| + \theta\Delta t |f(0)|}{1 - \theta\Delta t L_f}.$$

Under this condition, the interval $[-R, R]$ is invariant, both assumptions of the Banach fixed point theorem are satisfied, and (12) has a unique solution $c_i^{n+1} \in [-R, R]$.

The method (12) will be denoted in what follows by SL1. This method inherits the same stability and convergence properties of the parent methods, as it will be discussed in Section 4. Notice that this approach can be extended to spatially varying diffusion coefficients and that, while only first order in time, its effective accuracy can be substantially superior to that of more standard techniques, if higher degree interpolation operators are used, as shown in [4].

In order to derive a method of second order in time, we follow the main steps of [20],[33]. Applying the implicit weak method of order 2 defined in [26] for the

approximation of the stochastic streamlines (7), we define the points $z_{k,i}^{n+1}$ as the solutions of the nonlinear equations

$$z_{k,i}^{n+1} = x_i - \frac{\Delta t}{2} \left(u(x_i, t_{n+1}) + u(z_{k,i}^{n+1}, t_n) \right) + \sqrt{6\Delta t} \nu e_k. \quad (14)$$

Here, the symbols e_k denote the vectors:

$$\begin{aligned} e_1 &= \begin{pmatrix} 0 \\ 0 \end{pmatrix}, & e_2 &= \begin{pmatrix} 0 \\ 1 \end{pmatrix}, & e_3 &= \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \\ e_4 &= \begin{pmatrix} 1 \\ 0 \end{pmatrix}, & e_5 &= \begin{pmatrix} -1 \\ 0 \end{pmatrix}, & e_6 &= \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \\ e_7 &= \begin{pmatrix} 1 \\ -1 \end{pmatrix}, & e_8 &= \begin{pmatrix} -1 \\ 1 \end{pmatrix}, & e_9 &= \begin{pmatrix} -1 \\ -1 \end{pmatrix}. \end{aligned}$$

Accordingly, the weights α_k are given by

$$\alpha_1 = 4/9, \quad \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = 1/9, \quad \alpha_6 = \alpha_7 = \alpha_8 = \alpha_9 = 1/36.$$

In this case (see [26]) the increase in the order of approximation requires that moments of the probability density of $\sqrt{2\nu}\Delta W$ are reproduced by the discrete density up to the fifth moment. This motivates the introduction of further displacements and weights.

It is to be remarked that also method (14) can be rewritten in terms of the displacements $\delta_{k,i}^{n+1} = z_{k,i}^{n+1} - x_i$ as

$$\delta_{k,i}^{n+1} = -\frac{\Delta t}{2} \left(u(x_i, t_{n+1}) + u(x_i + \delta_{k,i}, t_n) \right) + \sqrt{6\Delta t} \nu e_k, \quad (15)$$

thus yielding an implicit method that is a natural extension to stochastic differential equations of that introduced in [37] and commonly used in meteorological applications for the computation of streamlines in SL methods.

A second order in time SL (SL2) scheme can then be defined by a Crank-Nicolson approach ((10) with $\theta = 0.5$) as

$$c_i^{n+1} = \sum_{k=1}^9 \alpha_k \left(I_p[c^n](z_{k,i}^{n+1}) + \frac{\Delta t}{2} f(I_p[c^n](z_{k,i}^{n+1})) \right) + \frac{\Delta t}{2} f(c_i^{n+1}). \quad (16)$$

Solvability of (16) with respect to c_i^{n+1} can be proved with the same arguments used for (12).

Notice that, with respect to the simpler first-order in time variant (12), nine interpolations at the foot of the streamlines must be computed, which clearly makes this approach substantially more expensive. In applications to systems of the form (8), the computational cost of scheme (16) can be marginally reduced by setting

$$\tilde{c}_i^n = \sum_{k=1}^9 \alpha_k I_p[c^n](z_{k,i}^{n+1})$$

and defining

$$c_i^{n+1} = \tilde{c}_i^n + \frac{\Delta t}{2} f(\tilde{c}_i^n) + \frac{\Delta t}{2} f(c_i^{n+1}), \quad (17)$$

so as to reduce the number of the evaluations of a possibly costly nonlinear term. Furthermore, when the coupling of the diffusion and advection term is weak, it should be possible to decouple again the approximation of a single deterministic streamline from that of the diffusive displacements, which could be added at the end of each approximate streamline without increasing too much the error. In particular, in [4],[5] explicit Euler or Heun methods were employed to compute these approximations, coupled to a substepping approach along the lines of [12],[38]. More specifically, given a positive integer m , a time substep was defined as $\Delta\tau = \Delta t/m$ and, for $q = 0, \dots, m-1$, the Euler substepping

$$\begin{cases} \hat{y}_i^{(q+1)} = \hat{y}_i^{(q)} - \Delta\tau u(\hat{y}_i^{(q)}, t_n), \\ \hat{y}_i^{(0)} = x_i \end{cases} \quad (18)$$

was computed, so that a $z_{k,i}^{n+1}$ in (12) was modified with $z_{k,i}^{n+1} = \hat{y}_i^m + \sqrt{4\Delta t\nu}d_k$.

A decoupled substepping variant of (14) might in turn be obtained by computing, for $q = 0, \dots, m-1$,

$$\begin{cases} \hat{y}_i^{(q+1)} = \hat{y}_i^{(q)} - \frac{\Delta\tau}{2} \left[u(\hat{y}_i^{(q)}, t_{n+1} - q\Delta\tau) + u(\hat{y}_i^{(q+1)}, t_{n+1} - (q+1)\Delta\tau) \right], \\ \hat{y}_i^0 = x_i, \end{cases} \quad (19)$$

and setting $z_{k,i}^{n+1} = \hat{y}_i^{(m)} + \sqrt{6\Delta t\nu}e_k$ in (16). We will denote this decoupled variant with substepping by SL2s.

Notice that, in realistic problems, a major shortcoming of scheme (16) is the fact that the Crank–Nicolson method, while A-stable, is not L-stable, see e.g. [28]. Therefore, no damping is introduced by the method for very large values of the time step and spurious oscillations may arise, see also the discussion in [7]. In order to reduce the computational cost and to address the L-stability issue, different variants of the scheme (16) could also be introduced and compared, along the lines proposed in [42] for the pure advection case. However, this development goes beyond the scope of this paper and will not be pursued here.

Finally, even though achieving full second order consistency is quite complicated in the variable diffusion coefficient case, the previously introduced schemes can be nonetheless extended at least in the simpler configurations as suggested in [4] for the first order case, even though full second order accuracy is not guaranteed any more.

4 Convergence analysis

We present in this section a convergence analysis for scheme (16). For simplicity, we assume a one-dimensional problem defined on $\mathbb{R} \times [0, T]$, with a time-independent drift term u :

$$\begin{cases} c_t + u(x)c_x - \nu c_{xx} = f(c) & (x, t) \in \mathbb{R} \times (0, T], \\ c(x, 0) = c_0(x) & x \in \mathbb{R}. \end{cases} \quad (20)$$

The multidimensional case, as well as the time dependence of u , require only small technical adaptations. On the other hand, the convergence analysis on bounded domains is still an open problem for high-order SL schemes, therefore we will not

address this problem here.

First, for $i \in \mathbb{Z}$ and $n = 0, \dots, N-1$, we rewrite scheme (16) with the shorthand notation

$$c_i^{n+1} = S_{\Delta t, \Delta x} \left(c^{n+1}, c^n, x_i \right), \quad (21)$$

where $x_i = i\Delta x$, and

$$\begin{aligned} S_{\Delta t, \Delta x} \left(c^{n+1}, c^n, x_i \right) &= \alpha_+ \left[I_p[c^n](z_+(x_i)) + \frac{\Delta t}{2} f(I_p[c^n](z_+(x_i))) \right] \\ &\quad \alpha_- \left[I_p[c^n](z_-(x_i)) + \frac{\Delta t}{2} f(I_p[c^n](z_-(x_i))) \right] \\ &\quad \alpha_0 \left[I_p[c^n](z_0(x_i)) + \frac{\Delta t}{2} f(I_p[c^n](z_0(x_i))) \right] + \frac{\Delta t}{2} f \left(c_i^{n+1} \right). \end{aligned}$$

In one space dimension, the three discrete characteristics are defined by the equations

$$\begin{aligned} z_+(x) &= x - \frac{\Delta t}{2} [u(x) + u(z_+(x))] + \sqrt{6\Delta t\nu}, \\ z_-(x) &= x - \frac{\Delta t}{2} [u(x) + u(z_-(x))] - \sqrt{6\Delta t\nu}, \\ z_0(x) &= x - \frac{\Delta t}{2} [u(x) + u(z_0(x))], \end{aligned}$$

with corresponding weights $\alpha_+ = \alpha_- = 1/6$ and $\alpha_0 = 2/3$. In what follows, we will use the symbol K to denote various positive constants, which do not depend on $\Delta t, x, t$. We will also assume that:

- (H0) there exists a unique classical solution of (20);
- (H1) $f(x) \in C^4(\mathbb{R})$ with $|f^{(p)}(x)| \leq K$ for $p \leq 4$;
- (H2) $u(x) \in C^2(\mathbb{R})$ with $|u^{(p)}(x)| \leq K$ for $p \leq 2$;
- (H3) for any $v(x) \in C^{p+1}(\mathbb{R})$ with bounded derivatives, $I_p[v]$ is a piecewise polynomial interpolation operator such that for any $x \in \mathbb{R}$

$$|I_p[v](x) - v(x)| \leq K\Delta x^p.$$

4.1 Consistency

First, we derive a consistency result via a Taylor expansion. The same kind of result can be obtained by probabilistic arguments, see [31].

Proposition 1 *Assume (H1)–(H3), and let $c(x, t)$ be a smooth solution with bounded derivatives of (20). Then, for each $(i, n) \in \mathbb{Z} \times \{0, \dots, N-1\}$ the consistency error of the scheme (16), defined as*

$$\mathcal{T}_{\Delta t, \Delta x}(x_i, t_n) = \frac{1}{\Delta t} (c(x_i, t_{n+1}) - S_{\Delta t, \Delta x}(c(t_n), c(t_{n+1}), x_i))$$

where $c(t_n) = (c(x_i, t_n))_i$, is such that

$$\mathcal{T}_{\Delta t, \Delta x}(x, t) = \mathcal{O} \left(\Delta t^2 + \frac{\Delta x^p}{\Delta t} \right).$$

Proof In what follows, we will omit the argument of functions computed at (x, t) . Consider a smooth solution c of (20). Since assumption **(H1)** holds, by differentiating in time and space (20) we get that c is also solution of

$$c_{tt} + u(x) c_{xt} - \nu c_{xxt} = f'(c) c_t, \quad (22)$$

$$c_{tx} + u'(x) c_x + u(x) c_{xx} - \nu c_{xxx} = f'(c) c_x, \quad (23)$$

and hence, by differentiating again in space (23), of

$$\begin{aligned} c_{txx} + u''(x) c_x + u'(x) c_{xx} + u'(x) c_{xx} + u(x) c_{xxx} - \nu c_{xxxx} \\ = f''(c) (c_x)^2 + f'(c) c_{xx}. \end{aligned} \quad (24)$$

Using (23) and (24) in (22), we get :

$$\begin{aligned} c_{tt} &= uu'c_c + u^2 c_{xx} - \nu \nu c_{xxx} - \nu u'' c_x - 2\nu u' c_{xx} - \nu \nu c_{xxx} + \nu^2 c_{xxxx} \\ &= (uu' - \nu u') c_x + (u^2 - 2\nu u') c_{xx} - 2\nu \nu c_{xxx} + \nu^2 c_{xxxx} \\ &\quad + f'(c) (c_t - u c_x + \nu c_{xx}) + \nu f''(c) (c_x)^2. \end{aligned} \quad (25)$$

Define now $U_{\pm}(x) = u(x) + u(z_{\pm}(x))$. By a Taylor expansion of $c(z_{\pm}(x), t)$ in space around (x, t) , we obtain

$$\begin{aligned} c(z_{\pm}(x), t) &= c + \left(\pm \sqrt{6\Delta t \nu} - \frac{\Delta t}{2} U_{\pm} \right) c_x + \frac{1}{2} \left(\pm \sqrt{6\Delta t \nu} - \frac{\Delta t}{2} U_{\pm} \right)^2 c_{xx} \\ &\quad + \frac{1}{6} \left(\pm \sqrt{6\Delta t \nu} - \frac{\Delta t}{2} U_{\pm} \right)^3 c_{xxx} \\ &\quad + \frac{1}{24} \left(\pm \sqrt{6\Delta t \nu} - \frac{\Delta t}{2} U_{\pm} \right)^4 c_{xxxx} \\ &\quad + \frac{1}{120} \left(\pm \sqrt{6\Delta t \nu} - \frac{\Delta t}{2} U_{\pm} \right)^5 c_{xxxxx} + \mathcal{O}(\Delta t^3) \end{aligned} \quad (26)$$

and, defining $U_0(x) = u(x) + u(z_0(x))$,

$$c(z_0(x), t) = c - \frac{\Delta t}{2} U_0 c_x + \frac{1}{2} \left(-\frac{\Delta t}{2} U_0 \right)^2 c_{xx} + \mathcal{O}(\Delta t^3). \quad (27)$$

Using (25),(26),(27) and the Taylor expansion

$$c(x, t + \Delta t) = c + \Delta t c_t + \frac{\Delta t^2}{2} c_{tt} + \mathcal{O}(\Delta t^3),$$

we obtain

$$\begin{aligned} c(x, t + \Delta t) - \sum_k \alpha_k c(z_k(x), t) &= \Delta t (c_t + u(x) c_x - \nu c_{xx}) \\ &\quad + \frac{\Delta t^2}{2} (f'(c) (c_t - u c_x + \nu c_{xx}) + \nu f''(c) (c_x)^2) \\ &\quad + \mathcal{O}(\Delta t^3) \end{aligned} \quad (28)$$

(note that, here and in what follows, k takes values in the set $\{+, -, 0\}$). Consider now the nonlinear reaction term. By assumption **(H3)**, we have that

$$\begin{aligned} f(c(z, t)) &= f(c) + f'(c)(c(z, t) - c) + \frac{1}{2}f''(c)(c(z, t) - c)^2 \\ &\quad + \frac{1}{6}f'''(c)(c(z, t) - c)^3 + \mathcal{O}\left((c(z, t) - c)^4\right) \end{aligned} \quad (29)$$

Using (28) in (29), and taking into account that $c(z_{\pm}, t) = c \pm \sqrt{6\Delta t\nu}c_x + \mathcal{O}(\Delta t)$ and $c(z_0(x), t) = c + \mathcal{O}(\Delta t)$, we obtain

$$\begin{aligned} \sum_k (\alpha_k f(c(z_k(x), t))) &= f(c) + f'(c)(c(x, t + \Delta t) - \Delta t(c_t + uc_x - \nu c_{xx}) - c) \\ &\quad + f''(c)(\Delta t\nu c_x^2) + \mathcal{O}(\Delta t^2). \end{aligned} \quad (30)$$

By (30) and (28), we get the consistency error for the semi-discretization,

$$\begin{aligned} c(x, t + \Delta t) - \sum_k \alpha_k \left(c(z_k(x), t) + \frac{\Delta t}{2} f(c(z_k(x), t)) \right) &- \frac{\Delta t}{2} f(c(x, t + \Delta t)) \\ &= \Delta t(c_t + u(x)c_x - \nu c_{xx} - f(c)) + \mathcal{O}(\Delta t^3). \end{aligned} \quad (31)$$

Introducing the interpolation error and using assumptions **(H2)** and **(H1)**, we finally prove the consistency error for the fully discrete scheme. \square

4.2 Stability

To prove stability, it is convenient to recast (21) in matrix form as

$$c^{n+1} - \frac{\Delta t}{2} f(c^{n+1}) = \sum_k \alpha_k \left[B_k c^n + \frac{\Delta t}{2} f(B_k c^n) \right], \quad (32)$$

where $f(c)$ denotes the vector obtained by applying f elementwise to the components of the vector c , while the matrices B_k (which represent the operation of interpolating c^n at the points $z_k(x_i)$) have elements $b_{k,ij}$ defined by

$$b_{k,ij} = \psi_j(z_k(x_i)), \quad (33)$$

for a suitable basis of cardinal functions $\{\psi_j\}$. The following proposition implies stability for the linear part of the scheme with respect to the 2-norm.

Proposition 2 *Assume **(H2)**, and let the matrix B have elements defined by (33), with (ψ_j) basis functions for odd degree symmetric Lagrange or splines interpolation. Then, for each k , there exists a constant $K_B > 0$ independent on $\Delta x, \Delta t$ such that*

$$\|B_k\|_2 \leq 1 + K_B \Delta t. \quad (34)$$

Proof Following [21], [22], we sketch the arguments to prove (34) for the cases of symmetric Lagrange and splines interpolation. In these cases, the method can be interpreted as Lagrange–Galerkin schemes with area-weighting. First, we make explicit the dependence of the points z_k on x and Δt . We recall that $z_k(x) = x + \delta_k(x)$, with δ_k solving the equation:

$$\delta_k(x) = -\frac{\Delta t}{2} (u(x) + u(x + \delta_k(x))) + \sqrt{6\Delta t}v e_k.$$

Expanding the term $u(x + \delta_k(x))$, we obtain therefore

$$\delta_k(x) = -\frac{\Delta t}{2} \left(u(x) + u(x) + \delta_k u'(x) + \mathcal{O}(\delta_k^2) \right) + \sqrt{6\Delta t}v e_k,$$

and hence,

$$\delta_k(x) \left(1 + \frac{\Delta t}{2} u'(x) \right) = -\Delta t u(x) + \sqrt{6\Delta t}v e_k + \mathcal{O}(\Delta t \delta_k^2)$$

(note that, here and in what follows, assumption **(H2)** ensures that all the remainder terms of the form $\mathcal{O}(\cdot)$ are smooth and uniformly bounded wrt x). Dividing now by $1 + \Delta t u'/2$, and using the fact that $\delta_k = \mathcal{O}(\sqrt{\Delta t})$, we get, for $\Delta t \rightarrow 0$,

$$z_k(x) = x - \Delta t u(x) + \sqrt{6\Delta t}v e_k + \mathcal{O}(\Delta t^{3/2}). \quad (35)$$

Due to the term $\sqrt{6\Delta t}v e_k$, the form of (35) does not coincide with that used in [21] for the points $z_k(x)$. However, for a generic couple of points $x_1, x_2 \in \mathbb{R}$, when considering differences $z_k(x_1) - z_k(x_2)$ this additional term is cancelled, so that

$$\begin{aligned} z_k(x_1) - z_k(x_2) &= (x_1 - x_2) - \Delta t (u(x_1) - u(x_2)) + \mathcal{O}(|x_1 - x_2| \Delta t^{3/2}) \\ &= (x_1 - x_2) - \Delta t u'(\xi)(x_1 - x_2) + \mathcal{O}(|x_1 - x_2| \Delta t^{3/2}), \end{aligned}$$

for a suitable point $\xi \in [\min(x_1, x_2), \max(x_1, x_2)]$ (note that the remainder term may actually be written in the form $\mathcal{O}(|x_1 - x_2| \Delta t^{3/2})$, since it comes from the difference of two remainders of the same form, vanishes for $x_1 = x_2$ and has a smooth dependence on x). As a consequence, the form (35) still satisfies the relevant properties used in the proof of (34). In particular, using the triangle inequality in the form of a difference, we get

$$|z_k(x_1) - z_k(x_2)| \geq |x_1 - x_2| - \Delta t \|u'\|_\infty |x_1 - x_2| + \mathcal{O}(|x_1 - x_2| \Delta t^{3/2}).$$

Therefore, the condition [21, Lemma 3]

$$|z_k(x_1) - z_k(x_2)| \geq \frac{1}{2} |x_1 - x_2|$$

is satisfied as soon as

$$\Delta t \|u'\|_\infty + \mathcal{O}(\Delta t^{3/2}) < \frac{1}{2}.$$

On the other hand, we have

$$\begin{aligned} |z_k(x_1) - (x_1 - x_2 + z_k(x_2))| &\leq \Delta t \|u'\|_\infty |x_1 - x_2| + \mathcal{O}(|x_1 - x_2| \Delta t^{3/2}) \\ &\leq \Delta t \left(\|u'\|_\infty + \mathcal{O}(\Delta t^{1/2}) \right) |x_1 - x_2|, \end{aligned}$$

which implies, for Δt small enough, the condition [21, Theorem 4]

$$|z_k(x_1) - (x_1 - x_2 + z_k(x_2))| \leq K_X |x_1 - x_2| \Delta t,$$

for a suitable positive constant K_X . Then, a careful replica of the arguments used in [21] provides the estimate (34). \square

For a formal definition of the basis functions ψ_j in the case of symmetric Lagrange and spline interpolation, we refer the reader to [21], [22]. While these two cases allow for a complete theory, at least in one space dimension, in the numerical tests with unstructured grids we will also use \mathbb{P}_2 interpolants, for which a first attempt of stability analysis is presented in [23].

4.3 Convergence

We now present a convergence result in the discrete 2-norm.

Theorem 3 *Assume (H0)–(H3), and, in addition, that (34) is satisfied. Let $c(x, t)$ be the classical solution of (20), and c^n be the solution of (21). Then, for any n such that $t_n \in [0, T]$ and for $(\Delta t, \Delta x) \rightarrow 0$,*

$$\|c(t_n) - c^n\|_2 \leq K_T \left(\Delta t^2 + \frac{\Delta x^p}{\Delta t} \right),$$

where K_T is positive constant depending on the final time T .

Proof While a mere convergence proof could be carried out with weaker regularity assumptions, we will focus here on the error estimate above, which requires the regularity assumptions (H0)–(H3). Define the vectors γ^n and ϵ^n , so that $\gamma_i^n = c(x_i, t_n)$, and $\epsilon^n = \gamma^n - c^n$. Then, by Proposition 1, we get

$$\gamma^{n+1} - \frac{\Delta t}{2} f(\gamma^{n+1}) = \sum_k \alpha_k \left[B_k^n \gamma^n + \frac{\Delta t}{2} f(B_k^n \gamma^n) \right] + \mathcal{O}(\Delta t^3 + \Delta x^p), \quad (36)$$

where the matrices B_k^n (which now represent the interpolation of c^n at the points $z_{k,i}^{n+1}$) have elements $b_{k,i,j}^n$ defined by

$$b_{k,i,j}^n = \psi_j(z_{k,i}^{n+1}).$$

Subtracting (21) from (36), using the Lipschitz continuity of f and the triangle inequality, we obtain from the left-hand side:

$$\left\| \gamma^{n+1} - \frac{\Delta t}{2} f(\gamma^{n+1}) - c^{n+1} + \frac{\Delta t}{2} f(c^{n+1}) \right\|_2 \geq \left(1 - \frac{L_f \Delta t}{2} \right) \|\epsilon^{n+1}\|_2.$$

Taking into account that $\sum_k \alpha_k = 1$, along with the bound (34), we also have from the right-hand side:

$$\begin{aligned} & \left\| \gamma^{n+1} - \frac{\Delta t}{2} f(\gamma^{n+1}) - c^{n+1} + \frac{\Delta t}{2} f(c^{n+1}) \right\|_2 \\ & \leq \left(1 + \frac{L_f \Delta t}{2} \right) (1 + K_B \Delta t) \|\epsilon^n\|_2 + \mathcal{O}(\Delta t^3 + \Delta x^p). \end{aligned}$$

Therefore, it turns out that

$$\left(1 - \frac{L_f \Delta t}{2}\right) \|\epsilon^{n+1}\|_2 \leq \left(1 + \frac{L_f \Delta t}{2}\right) (1 + K_B \Delta t) \|\epsilon^n\|_2 + \mathcal{O}(\Delta t^3 + \Delta x^p). \quad (37)$$

Now, for Δt small enough to have $1 - L_f \Delta t/2 > \underline{C} > 0$, we have that there exists a constant $K_T > 0$ such that

$$\frac{1 + \frac{L_f \Delta t}{2}}{1 - \frac{L_f \Delta t}{2}} (1 + K_B \Delta t) \leq 1 + K_T \Delta t,$$

and hence, using this bound in (37),

$$\|\epsilon^{n+1}\|_2 \leq (1 + K_T \Delta t) \|\epsilon^n\|_2 + \mathcal{O}(\Delta t^3 + \Delta x^p), \quad (38)$$

which, by standard arguments, implies that, for any n such that $t_n \in [0, T]$,

$$\|\epsilon^n\|_2 \leq K_T \left(\Delta t^2 + \frac{\Delta x^p}{\Delta t} \right).$$

□

5 Boundary conditions

The treatment of Dirichlet boundary conditions (BCs) for this class of semi-Lagrangian methods has been considered in [32], where two methods are proposed. One approach has first order of consistency, but it does not seem possible to generalize it to multiple dimensions. The second approach has order of consistency 1/2. More recently, in [5], an easier treatment has been proposed for the scheme SL1 with time-independent Dirichlet boundary condition, again with order of consistency 1/2. This approach has been extended in [3] to unstructured meshes.

We propose here a new approach to obtain second order consistency for the scheme SL2 with Dirichlet boundary conditions. This technique is based on the idea of using extrapolation to reconstruct the solution at feet of characteristics falling outside Ω , much in the spirit of the so-called *ghost-point* techniques, see e.g. [29]. We stress the fact that, while the emphasis in our presentation is on the treatment of BCs for the SL approximation of diffusive problem, the same technique and analysis also hold for the approximation of the pure advection problem, for which accurate BCs for SL methods are by no means easy to derive.

5.1 Construction of the extrapolation grid

In addition to the standard mesh $\mathcal{G}_{\Delta x} = \{x_i, x_i \in \overline{\Omega}\}$, on which the numerical solution is computed, we consider a second mesh $\mathcal{G}_h = \{\xi_i, \xi_i \in \overline{\Omega}\}$, used only for extrapolation, formed by a single layer of elements having their external side along the boundary of Ω . This second mesh is constructed with a size parameter $h \sim \sqrt{\Delta t}$, and the degrees of freedom are chosen in order to allow a second-order interpolation. We point out that, as we will soon prove, stability reasons force the

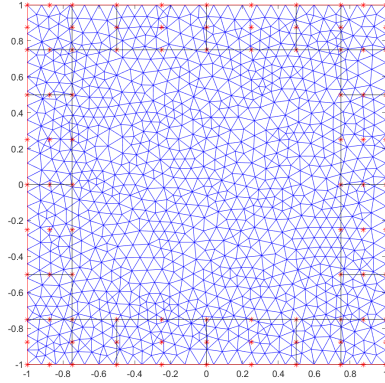


Fig. 1: Unstructured computational mesh (blue triangular elements) together with boundary mesh \mathcal{G}_h (black rectangular elements and red asterisks nodes)

parameter h to be at least of the same order of magnitude of the maximum distance of outgoing characteristics from Ω . This prevents in general from performing extrapolation via the same mesh used for interpolating at interior points.

In Fig. 1 we show, as an example, a square domain $\Omega = (-1, 1) \times (-1, 1)$, for which the standard mesh $\mathcal{G}_{\Delta x}$ is formed by the blue triangular elements and the mesh \mathcal{G}_h is formed by the black rectangular elements. Note that, in Fig. 1, the elements used for the extrapolation overlap at the corners, but this does not preclude the construction of a stable extrapolation. The asterisks in red denote the nodes of \mathcal{G}_h , according to the standard \mathbb{Q}_2 element. The values of the numerical solution on the nodes ξ_i are obtained by interpolation at internal nodes, and by the Dirichlet boundary condition if the nodes lie along the boundary $\partial\Omega$.

We then denote by $\mathcal{T}_{\Delta x}$ a given triangulation, with $\mathcal{G}_{\Delta x}$ the set of the vertices of the elements $K \in \mathcal{T}_{\Delta x}$ and define the polygonal domain $\bar{\Omega}_{\Delta x} := \cup_{K \in \mathcal{T}_{\Delta x}} K \subset \Omega$. If, for some i and k , $z_{k,i}^{n+1} \notin \bar{\Omega}_{\Delta x}$, then its projection $P(z_{k,i}^{n+1})$ onto $\bar{\Omega}_{\Delta x}$ is computed, defined as the point in $\bar{\Omega}_{\Delta x}$ at minimum distance from $z_{k,i}^{n+1}$. The value of the numerical solution $c^n(z_{k,i}^{n+1})$ is then approximated by a quadratic extrapolation operator Ψ_2 . This operator is constructed via the \mathbb{Q}_2 interpolant associated to the element of \mathcal{G}_h to which the projection $P(z_{k,i}^{n+1})$ belongs:

$$c^n(z_{k,i}^{n+1}) \simeq \Psi_2[\hat{c}^n](z_{k,i}^{n+1}),$$

where \hat{c}^n corresponds to

$$\hat{c}^n(\xi_i) = \begin{cases} I_2[c^n](\xi_i) & \text{if } \xi_i \in \Omega, \\ b(\xi_i, t_n) & \text{if } \xi_i \in \partial\Omega. \end{cases}$$

In the case of non-convex domain, the projection may not be unique and we consider as $P(z_{k,i}^{n+1})$ the closest projection point to the starting grid node x_i with respect the euclidean distance. The method can be extended to more general domains, by considering triangular elements for \mathcal{G}_h . In what follows, we provide a

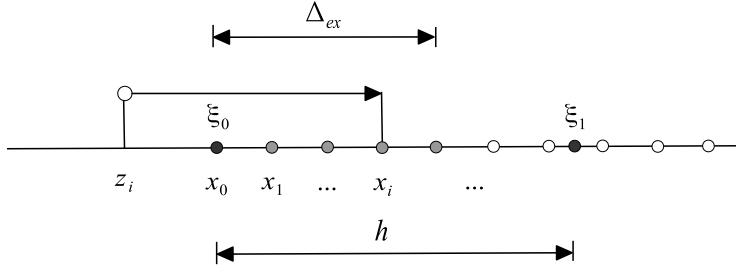


Fig. 2: Boundary extrapolation: basic setting

simplified analysis for this technique only for the one-dimensional problem, while we present a numerical validation for more complex situations in Section 6.5.

5.2 Theoretical analysis

In order to carry out a first theoretical analysis for the extrapolated boundary conditions, we set the problem in one space dimension, use a constant space step, and assume that the computational domain is given by the positive half-line, as in Fig. 2. We consider a scheme in the form

$$\begin{cases} c_i^{n+1} = I[c^n](z_{k,i}^{n+1}) & x_i \notin [x_0, x_0 + \Delta_{ex}], \\ c_i^{n+1} = \Psi[\hat{c}^n](z_{k,i}^{n+1}) & x_i \in (x_0, x_0 + \Delta_{ex}), \\ c_0^{n+1} = b^{n+1}, \end{cases} \quad (39)$$

with $|b^n| \leq M_b$, and examine in turn stability and consistency in the treatment of BCs. The form (39) is intended to represent a single term, for a given k , in (32).

Stability We start for simplicity by using both a first-order interpolation $I[c] = I_1[c]$ at internal points, and a first-order extrapolation $\Psi[c] = \Psi_1[c]$ at the boundary, the latter being performed between the boundary node $x_0 = \xi_0$ and an additional node $\xi_1 = \xi_0 + h$, which needs not coincide with any grid node. We also denote by Δ_{ex} the measure of the interval on which nodes has their respective feet of characteristics falling outside of the computational domain (and therefore use extrapolated values), so that

$$\Delta_{ex} = \max \left\{ \Delta_i = x_i - x_0 : x_i \in \Omega, z_{k,i}^{n+1} \notin \Omega \right\}.$$

In practice, we will soon show that, for the sake of stability, h must be chosen as a function of Δ_{ex} . In Fig. 2, we have marked in black the nodes used for extrapolation, in grey the nodes which require extrapolation, and in white all other nodes.

Consider a generic node $x_i \in (x_0, x_0 + \Delta_{ex})$, the corresponding value c_i^n of the numerical solution, and the associated foot of characteristic $z_{k,i}^{n+1}$. Define

$$\eta = \frac{z_{k,i}^{n+1} - x_0}{h}$$

(note that $\eta < 0$ if and only if $x_i \in (x_0, x_0 + \Delta_{ex})$). Then, using a first-order extrapolation, we have:

$$c_i^{n+1} = \Psi_1[c^n](z_{k,i}^{n+1}) = \eta I_1[c^n](\xi_1) + (1 - \eta)b^n,$$

so that, taking absolute values in the above expression and using the nonexpansivity of I_1 and the boundedness of b^n ,

$$\left| c_i^{n+1} \right| \leq |\eta| \|c^n\|_\infty + |1 - \eta| M_b.$$

Since for all the nodes outside the interval $[x_0, x_0 + \Delta_{ex}]$ the ∞ -norm does not increase, we get

$$\left\| c^{n+1} \right\|_\infty \leq \max(\|c^n\|_\infty, |\eta| \|c^n\|_\infty + |1 - \eta| M_b),$$

which leads to a uniform bound for $\|c^n\|_\infty$ as soon as $|\eta| < 1$, that is, for

$$h > \max_i |z_{k,i}^{n+1} - x_i|,$$

where the maximum is taken among the nodes in the interval $[x_0, x_0 + \Delta_{ex}]$. Note that, in the case of pure advection, we would obtain $h = O(\Delta t)$, whereas, in presence of a diffusion, $h = O(\Delta t^{1/2})$. In both cases, it is natural to choose h of the same order of magnitude of Δ_{ex} .

At a closer look, it turns out that the value $|\eta|$, which affects the stability of the extrapolated values of the solution, is nothing but the absolute value of the Lagrange basis function associated to the node ξ_1 . To treat a more general case, we can assume that the extrapolation is of degree N_{ex} , and uses x_0 and N_{ex} more nodes at constant step h ; in addition, we do not require the interpolation I to be L^∞ -nonexpansive, so that possibly $\|B_k\|_\infty > 1$.

Then, we can prove the following result:

Theorem 4 *Consider the scheme (39), and let the extrapolation $\Psi = \Psi_{N_{ex}}$ be performed with $N_{ex} + 1$ evenly spaced nodes ξ_k with step h and with $\xi_0 = x_0$. Assume moreover that the corresponding values of the numerical solution are computed via a possibly high-order interpolation $I[c^n]$. Then, there exists a constant C , depending only on N_{ex} and I , such that, for any i for which $z_{k,i}^{n+1} \in (x_0 - Ch, x_0]$, the sequence c_i^n remains bounded.*

Proof Denote by $L_m(x)$ the Lagrange basis function associated to the extrapolation node ξ_m . Using the I -interpolated values of the numerical solution at the nodes ξ_m , we obtain for the extrapolated values of c_i^{n+1} (second row of (39)):

$$\begin{aligned} c_i^{n+1} &= \Psi[\hat{c}^n](z_{k,i}^{n+1}) = \sum_{m=0}^{N_{ex}} I[\hat{c}^n](\xi_m) L_m(z_{k,i}^{n+1}) \\ &= b^n L_0(z_{k,i}^{n+1}) + \sum_{m=1}^{N_{ex}} I[\hat{c}^n](\xi_m) L_m(z_{k,i}^{n+1}). \end{aligned}$$

Following now the same ideas applied above for the first-order case, and taking into account the possible expansivity of I , we have

$$\|c^{n+1}\|_\infty \leq M_b |L_0(z_i)| + \|B_k\|_\infty \|c^n\|_\infty \sum_{m=1}^{N_{ex}} |L_m(z_{k,i}^{n+1})|,$$

and therefore, in order to have stability of the extrapolated values, we should require that

$$\|B_k\|_\infty \sum_{m=1}^{N_{ex}} |L_m(z_{k,i}^{n+1})| < 1. \quad (40)$$

On the other hand, replacing $z_{k,i}^{n+1}$ with the variable x and using the fact that the left-hand side of (40) is continuous, since $L_m(\xi_0) = 0$ for all $m \neq 0$,

$$\sum_{m=1}^{N_{ex}} |L_m(\xi_0)| = 0.$$

Then, it follows that

$$\|B_k\|_\infty \sum_{m=1}^{N_{ex}} |L_m(x)| < 1 \quad (41)$$

in a suitable left neighbourhood of ξ_0 . By similarity arguments, this neighbourhood can be written in the form $(x_0 - Ch, x_0]$ for some constant C depending only on the degree N_{ex} and on $\|B_k\|_\infty$ (that is, on the interpolation I). \square

As a consequence of the previous theorem, the step h should be chosen to satisfy the condition

$$h > \frac{1}{C} \max_i |z_{k,i}^{n+1} - x_i|, \quad (42)$$

where i is indexing all the nodes in $(x_0, x_0 + \Delta_{ex}]$. Note that, for first-order interpolation and extrapolation, we have already obtained $C = 1$. Mixing for example a second-order extrapolation with a first-order interpolation, an easy computation based on (41) would provide $C = 1/3$. In the numerical tests, we will use a combination of second-order extrapolation and second-order interpolation, for which it turns out that $C \approx 0.275$.

Consistency In evaluating the accuracy of this technique, we should split the error in two components – one associated to internal nodes, which has already been analysed in the previous section, and one related to the treatment of BCs, which comes into play only in the interval $[x_0, x_0 + \Delta_{ex}]$. A similar analysis for the time-discrete case has been carried out in [32, Theorem 4.1] with probabilistic arguments, and we will not repeat it here. For our purposes, the central argument of this analysis is that, representing the numerical scheme as a Markov chain, the expected number of steps spent by the chain in the interval $[x_0, x_0 + \Delta_{ex}]$ is bounded from above, and therefore the error introduced by the treatment of BCs does not accumulate. In our case, this means obtaining a consistency error bounded by the maximum between the internal truncation error proved in Prop.

1 and the extrapolation error (in which the latter should also include the error in reconstructing the values ξ_i). Then, the form of the truncation error becomes:

$$\begin{aligned}\mathcal{T}_{\Delta t, \Delta x}(x, t) &= \mathcal{O}\left(h^{N_{ex}+1} + \Delta x^p + \Delta t^2 + \frac{\Delta x^p}{\Delta t}\right) \\ &= \mathcal{O}\left(h^{N_{ex}+1} + \Delta t^2 + \frac{\Delta x^p}{\Delta t}\right),\end{aligned}\quad (43)$$

where, in the last row, we have kept only the asymptotically relevant terms. Thus, while the relationship between h and Δt is set according to the stability constraint (42), the degree N_{ex} should be chosen to preserve the consistency rate of the scheme. The choice of N_{ex} provides a specific value for the constant C and ultimately, using (42), for h . We obtain then two different situations:

- *Purely hyperbolic problems* ($\nu = 0$). In this case, according to (42), we have $h \sim \Delta t$, and therefore

$$\mathcal{T}_{\Delta t, \Delta x}(x, t) = \mathcal{O}\left(\Delta t^{\min(N_{ex}+1, 2)} + \frac{\Delta x^p}{\Delta t}\right).$$

In order to preserve second-order consistency wrt Δt , it suffices to enforce BCs with a linear extrapolation.

- *Parabolic problems*. Here, $h \sim \Delta t^{1/2}$ and hence

$$\mathcal{T}_{\Delta t, \Delta x}(x, t) = \mathcal{O}\left(\Delta t^{\min((N_{ex}+1)/2, 2)} + \frac{\Delta x^p}{\Delta t}\right).$$

In order to have a second-order scheme, we should therefore apply an extrapolation of degree three. Surprisingly, we will show in the numerical tests that an extrapolation of second degree suffices to retain second-order accuracy. We delay to a future work a deeper analysis of this effect, as well as of other accuracy issues.

Remark 1 In the numerical tests, we will eventually use a structured grid with centered cubic Lagrange interpolation, which requires a second frame of nodes around the cell in which interpolation is performed. Although, in this situation, interpolation in cells neighbouring the boundary would in principle be performed in the “unstable” region of the interpolation stencil, we have not detected any relevant instability in the numerical tests. A complete analysis of this case is out of the scope of this paper, but we note nevertheless that the idea that errors generated at the boundary do not accumulate, used for obtaining the consistency estimate (43), also applies to this case, and might provide a qualitative explanation for the stable behaviour of the scheme.

6 Numerical results

A number of numerical experiments have been carried out, in order to assess the accuracy of the proposed methods on both structured and unstructured meshes. We start with a simple heat equation, and we increase the level of complexity on the next problems considering an advection- diffusion equation, a reaction- diffusion

equation, an advection-diffusion-reaction system and finally an advection-diffusion equation on a non-convex domain.

In Subsections 6.1 and 6.2, we approximate problems whose analytic solution is known and this allows to compute the errors and to perform a numerical convergence analysis. In Subsections 6.3 and 6.4, we compare the numerical solutions with approximate solutions, obtained with higher order method. We define the errors, in the infinity and l^2 discrete relative norms, as

$$E_\infty = \max_{x_i \in \mathcal{G}_{\Delta x}} |c(x_i, t_N) - c_i^N| / \max_{x_i \in \mathcal{G}_{\Delta x}} |c(x_i, t_N)|,$$

$$E_2 = \left(\sum_{x_i \in \mathcal{G}_{\Delta x}} |c(x_i, t_N) - c_i^N|^2 / \sum_{x_i \in \mathcal{G}_{\Delta x}} |c(x_i, t_N)|^2 \right)^{\frac{1}{2}},$$

and we denote by p_∞ and p_2 the corresponding convergence rates.

In the unstructured case, we have constructed a triangular mesh by the Matlab2019 function `initmesh`, with a maximum mesh edge of Δx , and used a \mathbb{P}_2 space reconstruction. In the structured Cartesian case, the bicubic polynomial interpolation implemented in the Matlab2019 command `interp2`, has been used. Since the goal is to evaluate the accuracy of time discretization, both choices avoid to hide the time discretization error with the error introduced by a lower order space reconstruction.

6.1 Pure diffusion

In a first, basic test, we consider equation (1) in the pure diffusion case, i.e., with zero advection and reaction terms, on the square domain $\Omega = (-2, 2) \times (-2, 2)$, with $T = 1$ and $\nu = 0.05$. Based on the test case proposed in [35], we assume a Gaussian initial datum centered in $(0, 0)$, with $\sigma = 0.1$, so that the exact solution in an infinite plane would be

$$c(x, y, t) = \frac{1}{1 + 2\nu t / \sigma^2} \exp \left\{ -\frac{x^2 + y^2}{2(\sigma^2 + 2\nu t)} \right\}.$$

For this test case, we only consider structured meshes with constant steps $\Delta x = 4/N$.

Following [4], we consider different time step values Δt , which correspond to different values of the parabolic stability parameter $\mu = \Delta t \nu / \Delta x^2$. We compare method SL1 (12) and method SL2, (16), and collect the results in Table 1. Notice that, for method SL1, the value $\theta = 0.52$. This corresponds to a typical procedure in practical applications to realistic problems, see e.g. [2], [43], in which a value of θ slightly above $1/2$ is used to minimize the amount of numerical dissipation introduced by the time discretization. It can be observed that the expected convergence rates are recovered. Furthermore, it is apparent that scheme SL2 yields a substantial accuracy improvement, without an excessive increase in computational cost. Indeed, the SL2 runs require between 30% and 60% more CPU time, depending on the resolution, while leading to corresponding error reductions between 140% and 730%. As a comparison, a standard second order discretization in space coupled to an explicit second order method in time yields at the finest resolution an error 5 times larger than that of method SL2 at approximately the same computational cost.

Resolution			Relative error		Convergence rates	
Δx	Δt	μ	E_2	E_∞	p_2	p_∞
0.08	0.1	0.84	$3.34 \cdot 10^{-2}$	$5.10 \cdot 10^{-2}$	-	-
0.04	0.05	1.6	$1.33 \cdot 10^{-2}$	$2.05 \cdot 10^{-2}$	1.33	1.00
0.02	0.025	3.2	$6.57 \cdot 10^{-3}$	$1.03 \cdot 10^{-2}$	1.02	0.99

Resolution			Relative error		Convergence rates	
Δx	Δt	μ	E_2	E_∞	p_2	p_∞
0.08	0.1	0.84	$2.66 \cdot 10^{-3}$	$4.76 \cdot 10^{-3}$	-	-
0.04	0.5	1.6	$4.89 \cdot 10^{-4}$	$8.24 \cdot 10^{-4}$	2.44	2.53
0.02	0.025	3.2	$8.89 \cdot 10^{-5}$	$1.48 \cdot 10^{-4}$	2.46	2.48

Table 1: Errors for the pure diffusion test, first order method SL1 (upper) and second order method SL2 (lower) on a structured mesh.

6.2 Solid body rotation

Next, we consider the advection–diffusion equation (1) with coefficients $u = (-\omega y, \omega x)$, $\omega = 2\pi$, $\nu = 0.05$ and $f = 0$ on the square domain $\Omega = (-2, 2) \times (-2, 2)$ and $T = 1$. Following [35], we assume a Gaussian initial datum centered at $(x_0, y_0) = (1, 0)$ with $\sigma = 0.05$, so that the exact solution in an infinite plane would be

$$c(x, y, t) = \frac{1}{1 + 2\nu t/\sigma^2} \exp \left\{ -\frac{(x - x(t))^2 + (y - y(t))^2}{2(\sigma^2 + 2\nu t)} \right\}, \quad (44)$$

where $x(t) = x_0 \cos \omega t - y_0 \sin \omega t$, $y(t) = x_0 \sin \omega t - y_0 \cos \omega t$. We first consider structured meshes with constant steps $\Delta x = 4/N$. We consider again values of Δt corresponding to different parabolic stability parameters μ , as well as to different Courant number $\lambda = \Delta t \max |u|/\Delta x$.

In the structured case, we compare method SL1, (12), again with $\theta = 0.52$, and Euler substepping as in (18), the decoupled variant SL2s of method (16) with Heun substepping, and method SL2 (16) with the fully coupling (15). The results are reported in Table 2, in which convergence rates are computed with respect to the values in the first row. Furthermore, the convergence rate estimation for the values in the last row takes into account that the time step has been reduced by a factor 4. It can be observed that the expected convergence rates with respect to the time discretization error are recovered, in the constant Δx , constant C or constant μ convergence studies. It can also be observed that the decoupled variant SL2s, in spite of the loss of second order convergence, does indeed improve the results with respect to the SL1 method and is competitive with the full second order method SL2. As a comparison, a standard centered finite difference, second order discretization in space coupled to an explicit second order method in time yields at the finest resolution an error analogous to that of method SL2 but requires approximately three times its CPU time.

In the unstructured case, the quadratic polynomial interpolation naturally associated to \mathbb{P}_2 finite elements was employed and only the SL2s and SL2 methods were considered. The triangular mesh used was chosen with maximum triangle size Δx approximately equal to the corresponding structured meshes. The results are reported in Table 3. While the behaviour of the SL2 scheme is entirely analogous to that of the structured mesh case, the SL2s method shows in this case little error reduction when the spatial resolution is kept fixed.

Resolution				Relative error		Convergence rates	
Δx	Δt	λ	μ	E_2	E_∞	p_2	p_∞
0.04	0.05	16	1.62	0.15	0.16	-	-
0.04	0.025	8	0.82	$7.71 \cdot 10^{-2}$	$8.13 \cdot 10^{-2}$	0.96	0.98
0.02	0.025	16	3.2	$7.71 \cdot 10^{-2}$	$8.13 \cdot 10^{-2}$	0.96	0.98
0.02	0.0125	8	1.6	$3.92 \cdot 10^{-2}$	$4.13 \cdot 10^{-2}$	0.97	0.97

Resolution				Relative error		Convergence rates	
Δx	Δt	λ	μ	E_2	E_∞	p_2	p_∞
0.04	0.05	16	1.62	$7.65 \cdot 10^{-2}$	$7.95 \cdot 10^{-2}$	-	-
0.04	0.025	8	0.82	$3.89 \cdot 10^{-2}$	$4.02 \cdot 10^{-2}$	0.98	0.98
0.02	0.025	16	3.2	$3.89 \cdot 10^{-2}$	$4.02 \cdot 10^{-2}$	0.98	0.98
0.02	0.0125	8	1.6	$1.96 \cdot 10^{-2}$	$2.02 \cdot 10^{-2}$	0.98	0.99

Resolution				Relative error		Convergence rates	
Δx	Δt	λ	μ	E_2	E_∞	p_2	p_∞
0.04	0.05	16	1.62	0.11	0.11	-	-
0.04	0.025	8	0.82	$2.88 \cdot 10^{-2}$	$2.66 \cdot 10^{-2}$	1.93	2.05
0.02	0.025	16	3.2	$2.89 \cdot 10^{-2}$	$2.67 \cdot 10^{-2}$	1.93	2.04
0.02	0.0125	8	1.6	$7.35 \cdot 10^{-3}$	$6.64 \cdot 10^{-3}$	1.95	2.03

Table 2: Errors for the solid body rotation test, methods SL1 (upper), SL2s (middle) and SL2 (lower) on a structured mesh.

Resolution				Relative error		Convergence rates	
Δx	Δt	λ	μ	E_2	E_∞	p_2	p_∞
0.04	0.05	16	1.62	$2.39 \cdot 10^{-2}$	$2.25 \cdot 10^{-2}$	-	-
0.04	0.025	8	0.82	$2.72 \cdot 10^{-2}$	$2.84 \cdot 10^{-2}$	0.19	0.34
0.02	0.025	16	3.2	$7.20 \cdot 10^{-3}$	$6.32 \cdot 10^{-3}$	1.73	1.83
0.02	0.0125	8	1.6	$2.48 \cdot 10^{-3}$	$2.59 \cdot 10^{-3}$	3.46	3.45

Resolution				Relative error		Convergence rates	
Δx	Δt	λ	μ	E_2	E_∞	p_2	p_∞
0.04	0.05	16	1.62	0.129	0.139	-	-
0.04	0.025	8	0.82	$4.02 \cdot 10^{-2}$	$4.42 \cdot 10^{-2}$	1.68	1.65
0.02	0.025	16	3.2	$2.88 \cdot 10^{-2}$	$2.56 \cdot 10^{-2}$	2.16	2.44
0.02	0.0125	8	1.6	$7.70 \cdot 10^{-3}$	$8.08 \cdot 10^{-3}$	2.38	2.45

Table 3: Errors for the solid body rotation test, methods SL2s (upper) and SL2 (lower) on an unstructured mesh.

6.3 Reaction–diffusion equations

Following [19], we consider the Allen–Cahn equation

$$c_t = \nu \Delta c - c^3 + c$$

on the domain $\Omega = (0, 1) \times (0, 1)$, with periodic boundary conditions and for $t \in [0, 2]$. As in [19], we take the initial datum $c_0(x, y) = \sin(2\pi x) \sin(2\pi y)$ and a reference solution is computed by a pseudo-spectral Fourier discretization in space, see e.g. [11], and a fourth order Runge–Kutta scheme in time with a very large number of time steps. The results are reported in Table 4, for the values $\nu = 0.01$ and $\nu = 0.05$ of the diffusion parameter, respectively. In this case, only the SL2

Resolution			Relative error		Convergence rates	
Δx	Δt	μ	E_2	E_∞	p_2	p_∞
0.04	0.1	0.62	$1.10 \cdot 10^{-3}$	$1.31 \cdot 10^{-3}$	-	-
0.02	0.05	1.25	$2.72 \cdot 10^{-4}$	$2.98 \cdot 10^{-4}$	2.02	2.14
0.01	0.025	2.5	$6.53 \cdot 10^{-5}$	$7.06 \cdot 10^{-5}$	2.06	2.08

Resolution			Relative error		Convergence rates	
Δx	Δt	μ	E_2	E_∞	p_2	p_∞
0.04	0.1	0.62	$2.82 \cdot 10^{-2}$	$4.01 \cdot 10^{-2}$	-	-
0.02	0.05	1.25	$7.13 \cdot 10^{-3}$	$8.47 \cdot 10^{-3}$	1.98	2.24
0.01	0.025	2.5	$1.97 \cdot 10^{-3}$	$2.20 \cdot 10^{-3}$	1.86	1.94

Table 4: Error for the Allen–Cahn test with $\nu = 0.01$ (upper) and $\nu = 0.05$ (lower), second order method SL2 on an unstructured mesh.

scheme on unstructured meshes was considered and the reference solution was interpolated onto the unstructured mesh nodes using a higher order interpolation procedure. Both tests show a quadratic order of convergence.

6.4 Advection–diffusion–reaction systems

We consider in this case a set of four coupled advection–diffusion–reaction equations of the form (8)

$$\frac{\partial c_k}{\partial t} + u \cdot \nabla c_k - \nu \Delta c_k = f_k(c_1, \dots, c_4) \quad k = 1, \dots, 4 \quad (45)$$

on the square domain $\Omega = (-5, 5) \times (-5, 5)$ and on the time interval $t \in [0, 5]$. The advection field is given by coefficients $u = (-\omega y, \omega x)$, $\omega = 2\pi/10$, while the diffusion coefficient is set as $\nu = 0.01$. The reaction terms are given by

$$\begin{aligned} f_1 &= (c_1 - c_1 c_2) - (c_1 - c_3)/5 \\ f_2 &= -2(c_2 - c_1 c_2) - (c_2 - c_4)/5 \\ f_3 &= 2(c_3 - c_3 c_4) \\ f_4 &= -4(c_4 - c_3 c_4), \end{aligned}$$

which represent two coupled Lotka–Volterra prey–predator systems. As initial datum for c_1, c_3 , the function

$$c_0(x, y) = \begin{cases} \cos(2\pi[(x + 2.5)^2 + y^2]) & \text{for } (x + 2.5)^2 + y^2 \leq \frac{1}{4} \\ 0 & \text{for } (x + 2.5)^2 + y^2 > \frac{1}{4} \end{cases}$$

was considered, while the initial datum for c_2, c_4 , was taken to be equal to $3c_0$. In this test, only a structured mesh was considered with constant step $\Delta x = 1/20$. A reference solution is computed by a pseudo-spectral Fourier discretization in space and a fourth order Runge–Kutta scheme in time, using a very large number of time steps. The reference solution is reported for two sample components in Figure 3, while the absolute error distributions obtained for the same components with the second order method SL2 (16) using cubic interpolation, using a timestep

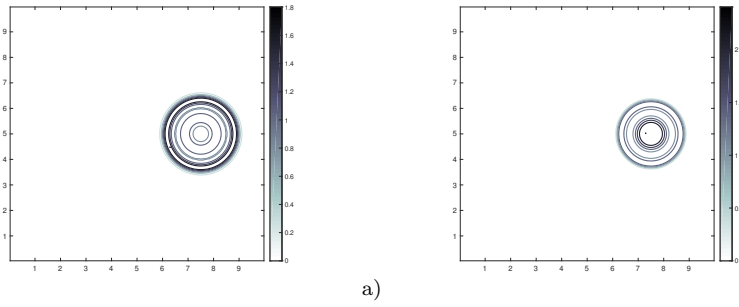


Fig. 3: Reference solutions for problem (45), a) component c_3 , b) component c_4 at time $T = 5$.

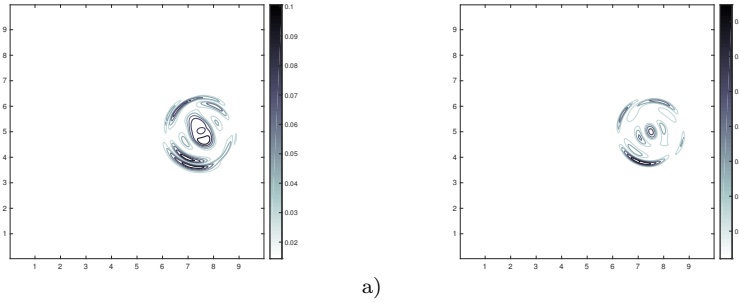


Fig. 4: Absolute errors of second order SL2 method for problem (45), a) component c_3 , b) component c_4 at time $T = 5$.

corresponding to $\lambda \approx 7$ and $\mu \approx 1/2$, are shown in Figure 4. As a reference, the errors for a second order finite difference approximation of (45) using a second order Runge–Kutta scheme in time with a time step 20 times smaller are shown in Figure 5, while the errors obtained using a fourth order finite difference approximation for the advection term in (45) with a third order Runge–Kutta scheme in time are displayed in Figure 6, again computed with a time step 20 times smaller than that used for the SL2 method. It can be seen that the SL2 method allows to achieve errors of the same order of magnitude as those of the third order Runge–Kutta in time, while allowing for a much larger time step without solving large algebraic systems.

6.5 Advection–diffusion equation, nonhomogeneous boundary conditions

In this last set of numerical experiments, we consider nonhomogeneous, possibly time-dependent Dirichlet boundary conditions in four cases: pure diffusion, constant advection–diffusion, solid body rotation with diffusion and advection–diffusion on a nonconvex domain. In all these tests, we have used the SL2 scheme on an unstructured mesh. In the first three cases, we consider $\Omega = (-1, 1) \times (-1, 1)$, final time $T = 1$ and an initial condition in the form of a Gaussian centered at

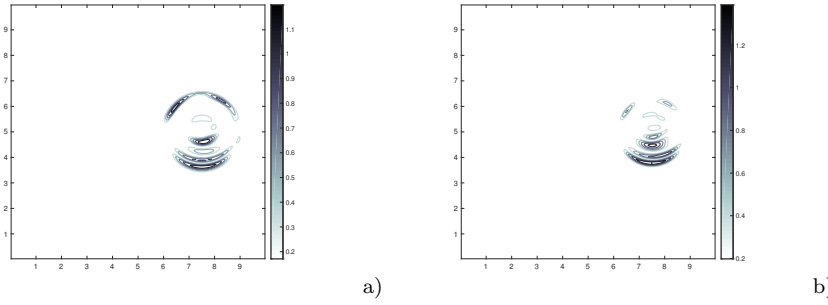


Fig. 5: Absolute errors of second order finite difference method for problem (45), a) component c_3 , b) component c_4 at time $T = 5$.

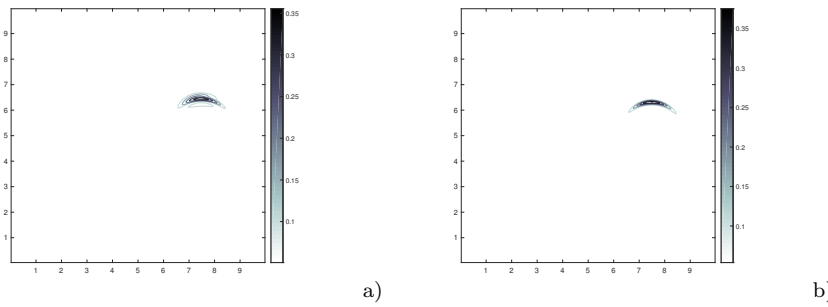


Fig. 6: Absolute errors of fourth order finite difference method for problem (45), a) component c_3 , b) component c_4 at time $T = 5$.

Resolution				Relative error		Convergence rates	
Δx	Δt	μ	h	E_2	E_∞	p_2	p_∞
0.04	0.05	1.56	0.5	$4.70 \cdot 10^{-3}$	$1.39 \cdot 10^{-2}$	-	-
0.04	0.025	0.78	0.5	$3.18 \cdot 10^{-3}$	$1.06 \cdot 10^{-2}$	-	-
0.02	0.025	3.12	0.33	$3.71 \cdot 10^{-4}$	$1.01 \cdot 10^{-3}$	3.66	3.78
0.02	0.0125	1.56	0.33	$4.35 \cdot 10^{-4}$	$9.57 \cdot 10^{-4}$	2.87	3.47

Table 5: Errors and convergence rates for the pure diffusion problem with nonhomogeneous Dirichlet conditions, SL2 method, unstructured mesh

$(x_0, y_0) = (0.5, 0)$, with $\sigma = 0.1$. In Fig. 1, we show the space meshes $\mathcal{G}_{\Delta x}$ and \mathcal{G}_h corresponding to the steps $\Delta x = 0.04$, $h = 0.5$, which were used to compute the results in the first two rows of Tables 5-7. In order to have a reference solution to compare with, we compute the exact solution on the whole of \mathbb{R}^2 and enforce its values at the boundary as boundary conditions, so that $b(x, y, t) = c(x, y, t)$ for $(x, y) \in \partial\Omega$, $t \in [0, T]$. For all the three cases, we have set $\nu = 0.05$ and $T = 1$. In the second and third test, the advection field has been chosen as $u = (1, 0)$, and $u = (-2\pi y, 2\pi x)$, respectively. Tables 5-7 report the numerical errors obtained by the SL2 scheme in these tests, showing in all cases at least a quadratic convergence.

Resolution					Relative error		Convergence rates	
Δx	Δt	λ	μ	h	E_2	E_∞	p_2	p_∞
0.04	0.05	1.25	1.56	0.5	$7.35 \cdot 10^{-3}$	$1.18 \cdot 10^{-2}$	-	-
0.04	0.025	0.625	0.78	0.5	$8.35 \cdot 10^{-3}$	$1.32 \cdot 10^{-2}$	-	-
0.02	0.025	1.25	3.12	0.33	$3.76 \cdot 10^{-4}$	$7.59 \cdot 10^{-4}$	4.29	3.96
0.02	0.0125	0.625	1.56	0.33	$2.64 \cdot 10^{-4}$	$5.58 \cdot 10^{-4}$	4.98	4.56

Table 6: Errors and convergence rates for the advection–diffusion problem with nonhomogeneous Dirichlet conditions, SL2 method, unstructured mesh.

Resolution					Relative error		Convergence rates	
Δx	Δt	λ	μ	h	E_2	E_∞	p_2	p_∞
0.04	0.05	7.85	1.56	0.5	$5.62 \cdot 10^{-2}$	$6.09 \cdot 10^{-2}$	-	-
0.04	0.025	3.92	0.78	0.5	$1.49 \cdot 10^{-2}$	$1.60 \cdot 10^{-2}$	-	-
0.02	0.025	7.85	3.12	0.33	$1.49 \cdot 10^{-2}$	$8.98 \cdot 10^{-2}$	1.91	-
0.02	0.0125	3.92	1.56	0.33	$3.43 \cdot 10^{-3}$	$3.61 \cdot 10^{-3}$	2.12	2.15

Table 7: Errors and convergence rates for the solid body rotation problem with nonhomogeneous Dirichlet conditions, SL2 method, unstructured mesh.

We finally consider the advection–diffusion equation with $\nu = 0.001$, on the domain $\Omega = ([0, 1] \times [0, 0.4]) \setminus B_{r_0}(x_0, y_0)$, where $B_{r_0}(x_0, y_0)$ denotes a circle with radius $r_0 = 0.05$ centered in $(x_0, y_0) = (0.1, 0.2)$. The initial datum is $c_0(x, y) = 0$ and the boundary condition

$$b(x, y, t) = \begin{cases} y(0.4 - y) \frac{4}{0.4^2} & (x, y) \in \{0\} \times [0, 0.4], t \in [0, T] \\ 1 & (x, y) \in \partial B_r(x_0), t \in [0, T] \\ 0 & \text{otherwise.} \end{cases}$$

The velocity field $u(x, y)$ is given by

$$u(x, y) = \left(u_0 + \frac{u_0 r_0^3}{2r^3} - \frac{3u_0 r_0^3 (x - x_0)^2}{2r^5}, -\frac{3r_0^3 u_0 (x - x_0)(y - y_0)}{2r^5} \right),$$

where we set $u_0 = 0.2$ and $r^2 = (x - x_0)^2 + (y - y_0)^2$. In Fig. 7, we show the domain Ω , discretized using a Delaunay mesh $\mathcal{G}_{\Delta x}$ with $\Delta x = 0.1$, refined around the circular hole. In Fig. 8, we show the numerical solution computed with SL2 with time step $\Delta t = 0.005$ for time $t = 0.5, 1, 2, 3$. The nonhomogeneous boundary condition are computed by extrapolation with an extra grid \mathcal{G}_h with $h = 1.5\sqrt{\Delta t}$. In this case, the additional mesh \mathcal{G}_h has been built around the circular hole, as well as along the external rectangular boundary. Note that, even though the wide stencil of the scheme might cause problems with discontinuous initial/boundary data (see the discussion in [20]), the boundary condition is smoothly propagated in the interior of the domain.

7 Conclusions

A family of fully semi-Lagrangian approaches for the discretization of advection–diffusion–reaction systems has been proposed, which extend the methods outlined

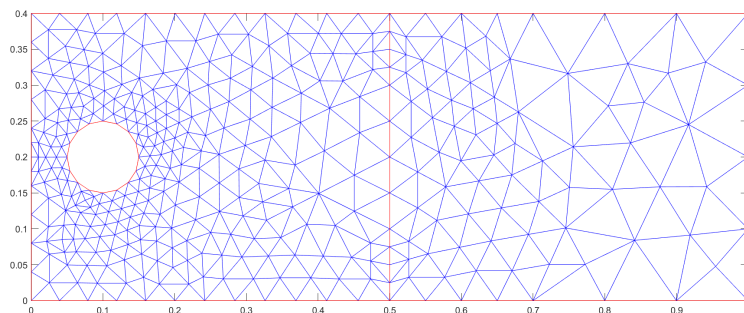


Fig. 7: Unstructured mesh for the non-convex problem

in [4], [5] to full second order accuracy. A numerical treatment of Dirichlet boundary condition, also with second order accuracy, has been proposed. The stability and convergence of the basic second order method have been analyzed. The proposed methods have been validated on a number of classical benchmarks, on both structured and unstructured meshes. Numerical results show that these methods yield good quantitative agreement with reference solutions, while being superior in efficiency to standard implicit methods and to approaches in which the SL method is only used for the advection term. In future developments, the proposed method will be extended to higher order discretizations along the lines of [42] and will be applied to the development of second order fully semi-Lagrangian methods for the Navier-Stokes equations along the lines of [3], [5]. Efficiency improvement for the unstructured implementation of the scheme is currently being studied in [9].

Acknowledgements

This work has been partly supported by PRIN 2017 project *Innovative numerical methods for evolutionary partial differential equations and applications*, and by INDAM–GNCS 2019 project *Approssimazione numerica di problemi di natura iperbolica ed applicazioni*.

We would also like to thank Stefano Micheletti for providing us MATLAB implementations of some finite element methods, and the two anonymous reviewers for their constructive and helpful comments.

Data availability statement

Data and codes produced with this work will be made available upon reasonable request.

References

1. O. Bokanowski and G. Simarmata. Semi-Lagrangian discontinuous Galerkin schemes for some first-and second-order partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 50:1699–1730, 2016.

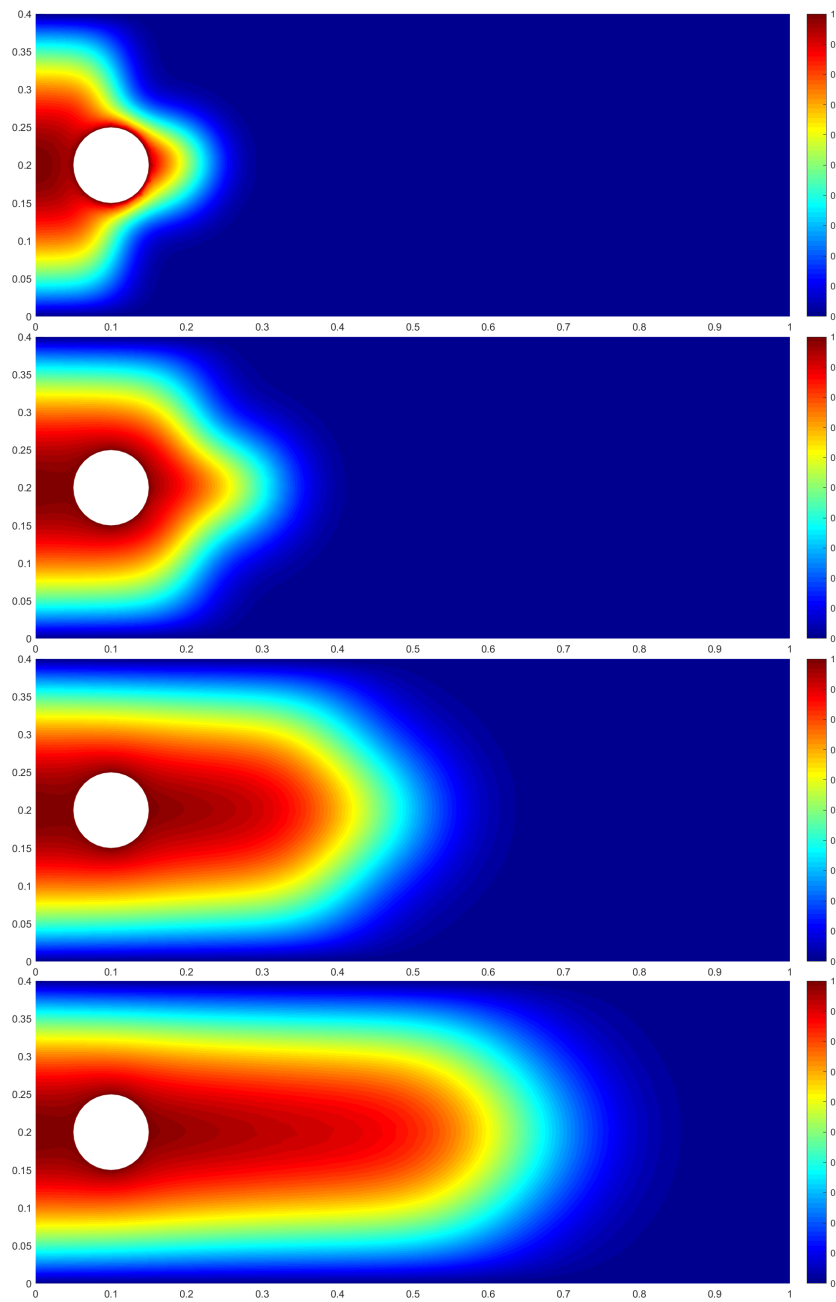


Fig. 8: Numerical solution at time $t = 0.5, 1, 2, 3$

2. L. Bonaventura. A semi-implicit, semi-Lagrangian scheme using the height coordinate for a nonhydrostatic and fully elastic model of atmospheric flows. *Journal of Computational Physics*, 158:186–213, 2000.
3. L. Bonaventura, E. Calzola, E. Carlini, and R. Ferretti. A fully semi-Lagrangian method for the Navier-Stokes equations in primitive variables. In *Proceedings of the 2017 Finite Elements in Flows Conference, Rome*. Springer Verlag, 2020.
4. L. Bonaventura and R. Ferretti. Semi-Lagrangian methods for parabolic problems in divergence form. *SIAM Journal of Scientific Computing*, 36:A2458 – A2477, 2014.
5. L. Bonaventura, R. Ferretti, and L. Rocchi. A fully semi-Lagrangian discretization for the 2D Navier-Stokes equations in the vorticity–streamfunction formulation. *Applied Mathematics and Computation*, 323:132–144, 2018.
6. L. Bonaventura, R. Redler, and R. Budich. *Earth System Modelling 2: Algorithms, Code Infrastructure and Optimisation*. Springer Verlag, New York, 2012.
7. L. Bonaventura and A. Della Rocca. Unconditionally strong stability preserving extensions of the TR-BDF2 method. *Journal of Scientific Computing*, 70:859–895, 2017.
8. C.-E. Bréhier and E. Faou. Analysis of the Monte-Carlo error in a hybrid semi-Lagrangian scheme. *Applied Mathematics Research eXpress*, 2015(2):167–203, 2015.
9. S. Cacace, L. Della Cioppa, and R. Ferretti. Efficient implementation of characteristic-based schemes on unstructured triangular grids. *in preparation*.
10. M. Camilli and M. Falcone. An approximation scheme for the optimal control of diffusion processes. *Mathematical Modelling and Numerical Analysis*, 29:97–122, 1995.
11. C. Canuto, M.Y. Hussaini, A. Quarteron, and T.A. Zang. *Spectral methods: Fundamentals in single domains*. Springer Science & Business Media, 2006.
12. V. Casulli. Semi-implicit finite difference methods for the two dimensional shallow water equations. *Journal of Computational Physics*, 86:56–74, 1990.
13. V. Casulli and E. Cattani. Stability, accuracy and efficiency of a semi-implicit method for three-dimensional shallow water flow. *Computers and Mathematics with Applications*, 27(4):99–112, 1994.
14. J. Côté and A. Staniforth. A two time level semi-Lagrangian semi-implicit scheme for spectral models. *Monthly Weather Review*, 116:2003–2012, 1988.
15. K. Debrabant and E. Jakobsen. Semi-Lagrangian schemes for linear and fully non-linear diffusion equations. *Mathematics of Computation*, 82:1433–1462, 2013.
16. C. Erath, P.H. Lauritzen, J.H. Garcia, and H.M. Tufo. Integrating a scalable and efficient semi-Lagrangian multi-tracer transport scheme in HOMME. *Procedia Computer Science*, 9:994–1003, 2012.
17. C. Erath, M.A. Taylor, and R.D. Nair. Two conservative multi-tracer efficient semi-Lagrangian schemes for multiple processor systems integrated in a spectral element (climate) dynamical core. *Communications in Applied and Industrial Mathematics*, 7:74–98, 2016.
18. M. Falcone and R. Ferretti. *Semi-Lagrangian Approximation Schemes for Linear and Hamilton–Jacobi Equations*. SIAM, 2013.
19. X. Feng, H. Song, T. Tang, and J. Yang. Nonlinear stability of the implicit-explicit methods for the Allen-Cahn equation. *Inverse Problems and Imaging*, 7:679–695, 2013.
20. R. Ferretti. A technique for high-order treatment of diffusion terms in semi-Lagrangian schemes. *Communications in Computational Physics*, 8:445–470, 2010.
21. R. Ferretti. On the relationship between semi-Lagrangian and Lagrange–Galerkin schemes. *Numerische Mathematik*, 124:31–56, 2013.
22. R. Ferretti and M. Mehrenberger. Stability of semi-Lagrangian schemes of arbitrary odd degree under constant and variable advection speed. *Mathematics of Computation*, to appear.
23. R. Ferretti and G. Perrone. On the stability of semi-Lagrangian advection schemes under finite element interpolations. In *Applied And Industrial Mathematics In Italy II*, pages 339–350. World Scientific, 2007.
24. F. Garcia, L. Bonaventura, M. Net, and J. Sánchez. Exponential versus IMEX high-order time integrators for thermal convection in rotating spherical shells. *Journal of Computational Physics*, 264:41–54, 2014.
25. M. Hochbruck and A. Ostermann. Exponential integrators. *Acta Numerica*, pages 209–286, 2010.
26. P.E. Kloeden and E. Platen. *Numerical solution of stochastic differential equations*, volume 23 of *Applications of Mathematics (New York)*. Springer-Verlag, Berlin, 1992.

27. H. Kushner and P.G. Dupuis. *Numerical methods for stochastic control problems in continuous time*, volume 24. Springer Science & Business Media, 2013.
28. J.D. Lambert. *Numerical methods for ordinary differential systems*. Wiley, 1991.
29. R. J. Leveque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, Cambridge, UK, 2002.
30. G.N. Milstein. The probability approach to numerical solution of nonlinear parabolic equations. *Numerical Methods for Partial Differential Equations*, 18:490–522, 2002.
31. G.N. Milstein and M.V. Tretyakov. Numerical algorithms for semilinear parabolic equations with small parameter based on approximation of stochastic equations. *Mathematics of Computation*, 69:237–567, 2000.
32. G.N. Milstein and M.V. Tretyakov. Numerical solution of the Dirichlet problem for nonlinear parabolic equations by a probabilistic approach. *IMA Journal of Numerical Analysis*, 21:887–917, 2001.
33. G.N. Milstein and M.V. Tretyakov. *Stochastic numerics for mathematical physics*. Springer Science & Business Media, 2013.
34. A. Picarelli and C. Reisinger. Probabilistic error analysis for some approximation schemes to optimal control problems. *Systems & Control Letters*, 137:104619, 2020.
35. J. Pudykiewicz and A. Staniforth. Some properties and comparative performance of the semi-Lagrangian method of Robert in the solution of the advection diffusion equation. *Atmosphere-Ocean*, 22:283–304, 1984.
36. A. Quarteroni and A. Valli. *Numerical approximation of partial differential equations*. Springer Verlag, 1994.
37. A. Robert. A semi-Lagrangian and semi-implicit numerical integration scheme for the primitive meteorological equations. *Journal of the Meteorological Society of Japan*, 60:319–325, 1982.
38. G. Rosatti, D. Cesari, and L. Bonaventura. Semi-implicit, semi-Lagrangian environmental modelling on Cartesian grids with cut cells. *Journal of Computational Physics*, 204:353–377, 2005.
39. A. Staniforth and J. Coté. Semi-Lagrangian integration schemes for atmospheric models—a review. *Monthly Weather Review*, 119:2206–2223, 1991.
40. C. Temperton, M. Hortal, and A. Simmons. A two-time-level semi-Lagrangian global spectral model. *Quarterly Journal of the Royal Meteorological Society*, 127:111–127, 2001.
41. C. Temperton and A. Staniforth. An efficient two-time-level semi-Lagrangian semi-implicit integration scheme. *Quarterly Journal of the Royal Meteorological Society*, 113:1025–1039, 1987.
42. G. Tumolo and L. Bonaventura. A semi-implicit, semi-Lagrangian, DG framework for adaptive numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 141:2582–2601, 2015.
43. G. Tumolo, L. Bonaventura, and M. Restelli. A semi-implicit, semi-Lagrangian, p -adaptive discontinuous Galerkin method for the shallow water equations. *Journal of Computational Physics*, 232:46–67, 2013.
44. N.P. Wedi, P. Bauer, M. Diamantakis, M. Hamrud, S. Malardel, K. Mogensen, G. Mozdzyński, and P.K. Smolarkiewicz. The modelling infrastructure of the Integrated Forecasting System: Recent advances and future challenges. Technical Report 760, ECMWF, 2015.