EPJ manuscript No.
(will be inserted by the editor)

Exponential integrators for time–fractional partial differential equations

Roberto Garrappa[a]

Dipartimento di Matematica, Università degli Studi di Bari, Via E.Orabona 4, 70125 Bari, Italy

Abstract. Time–fractional partial differential equations can be numerically solved by first discretizing with respect to the spatial derivatives and hence applying a time–step integrator. An exponential integrator for fractional differential equations is proposed to overcome the stability issues due to the stiffness in the resulting semi–discrete system. Convergence properties and the main implementation issues are studied. The advantages of the proposed method are illustrated by means of some test problems.

1 Introduction

Partial differential equations (PDEs) are one of the main tools in modeling many fundamental real–life systems.

For a long time, integer–order derivatives have been the leading instrument for describing dynamical systems. However, more recent studies have shown that models based only on derivatives of integer order are not completely satisfactory to depict the complex and sophisticated nature of several phenomena and therefore, on the basis of physical and mathematical considerations, differential operators of fractional (i.e., non integer) order have been introduced (see, for instance, [23,31,38]). Nowadays PDEs with fractional derivatives are very popular in a wide range of research areas, from biology to chemistry, economics, mechanics, seismology and so on.

One of the main reasons leading to the introduction of fractional–order derivatives in equations governing real–life systems is to adequately describe and simulate the behavior of non–ideal materials showing anomalous phenomena and/or dynamics with long–memory features; some classical examples can be observed by studying transfer processes in non–heterogeneous media or non–local interactions in continuum mechanics (e.g., see [29,30]).

Recently in [12] it has been discussed a nonlinear fractional–time derivative model of temperature and pressure waves propagating in porous rocks. It is widely accepted that in order to obtain a realistic simulation of wave propagation in fluid–saturated porous rocks it is necessary to take into account the variability of the main coefficients of the porous media due to the obstruction of pores as the result of transport of solid fine particles. As proposed by Caputo [2], an efficient way to study and model this variability is by means of a memory formalism in which classical time derivatives

[a] e-mail: roberto.garrappa@uniba.it

are replaced by derivatives of fractional order. Following this approach, the analysis carried out in [12] has led to a nonlinear time–fractional equation in the form

$$\frac{\partial^\alpha}{\partial t^\alpha}u + \varepsilon u\frac{\partial}{\partial x}u = \nu\frac{\partial^2}{\partial x^2}u, \tag{1}$$

where ε and ν are constant coefficients and $\alpha \in (0,1)$ is the fractional order usually determined on the basis of experimental data.

Throughout the paper, t and x will denote time and spatial variables respectively and $u = u(t,x)$ the unknown solution in a domain $\mathcal{D} = [0,T] \times \Omega$, $\Omega \subseteq \mathbb{R}^q$, $q \in \{1,2,3\}$. With d^α/dt^α we denote the fractional derivative operator with respect to the origin and defined, for $\alpha > 0$, according to the Caputo's definition [5]

$$\frac{d^\alpha}{dt^\alpha}y(t) = \frac{1}{\Gamma(m-\alpha)}\int_0^t \big(t-s\big)^{m-\alpha-1} y^{(m)}(s)ds,$$

where $\Gamma(\cdot)$ is the Euler gamma function, $m = \lceil \alpha \rceil$ the smallest integer such that $m > \alpha$ and $y^{(m)}$ denotes the classical derivative of integer order m. Some alternative definitions for fractional–order derivatives have been also introduced; the Caputo's definition is however the most convenient for several practical applications since it allows to couple the problem with standard initial conditions of Cauchy type

$$\frac{d^k}{dt^k}u(0,x) = u_{0,k}(x), \quad k = 0,\dots,m-1,$$

which are the most common in practical real–life applications (as usual, some other Dirichlet, Neumann or mixed boundary conditions are involved to guarantee the uniqueness of the solution).

Equation (1) can be regarded as a special case of the time–fractional advection–diffusion–reaction (ADR) equation

$$\frac{\partial^\alpha}{\partial t^\alpha}u + \varepsilon\frac{\partial}{\partial x}u = \nu\frac{\partial^2}{\partial x^2}u + \rho f(u) \tag{2}$$

which will be used as the reference equation throughout this paper.

Examples of fractional systems with a nonlinear reaction term $f(u)$ of *Bonhoeffer-van der Pol* and *Brusselator* type have been investigated, for instance, in [9]. A very special case is the time–fractional diffusion equation [41,43], with $\varepsilon = \rho = 0$, generalizing the standard heat equation.

The numerical solution of time fractional partial differential equations (TFPDEs) is more involved than the integer–order case. Not only physical and mathematical aspects have to be taken into account to obtain reasonable results but also algorithmic issues must be carefully investigated in order to perform the computation in an effective way. The lack of smoothness in the true solution and the long–persistent memory of fractional order systems make indeed the computation very challenging.

This work concerns with the numerical treatment of TFPDEs and we discuss an approach, called exponential integrators, which has been successfully employed in the numerical simulation of integer–order PDEs but, so far, scarcely investigated in the context of fractional PDEs.

This paper is organized as follows: Section 2 deals with the discretization of spatial derivatives in TFPDEs and some major issues in the solution of the semi–discrete systems are discussed. In Section 3 we device an exponential integrator for fractional–order problems and we study the convergence properties; some details concerning the practical implementation are hence discussed in Section 4. Finally, in Section 5 some numerical experiments are presented in order to illustrate reliability and advantages of the proposed method.

2 Semi–discretization of time–fractional PDEs

A well–established strategy in the numerical treatment of PDEs is to discretize the spatial derivatives and leave the time variable continuous in order to transform the PDE into a system of ordinary differential equations (ODEs) approximating the original problem. The spatial discretization can be pursued in several ways, for instance by finite differences, finite elements or spectral methods. For simplicity of presentation, in this paper we confine our attention on semi–discrete methods based on finite differences but most of the results can be readily extended to further approaches.

To this purpose, for one dimensional problems we consider an equispaced mesh–grid $x_j = j\Delta x$ on a spatial domain $\Omega = [0,1]$, where $\Delta x = 1/(N_x+1)$. A straightforward way to discretize the spatial derivatives is to replace them by central differences

$$\begin{aligned}\frac{d}{dx}u(x_j,t) &\approx \frac{u(x_{j+1},t)-u(x_{j-1}t)}{2\Delta x} + \mathcal{O}\big(\Delta x^2\big),\\ \frac{d^2}{dx^2}u(x_j,t) &\approx \frac{u(x_{j-1},t)-2u(x_j,t)+u(x_{j+1},t)}{\Delta x^2} + \mathcal{O}\big(\Delta x^2\big),\end{aligned}$$

where the second–order accuracy is achieved under the assumption of sufficiently smoothness of u_{xxx} and u_{xxxx}. In several situations, other schemes are preferable for stability reasons; we do not pursue this topic further here and we refer to some classical textbooks on the numerical treatment of PDEs (e.g., [22] or [24]).

By focusing, for simplicity, on $0 < \alpha < 1$, the application of the above finite difference schemes in TFPDEs (2) leads to semi–discrete systems of fractional differential equations (FDEs) in the form

$$\frac{\partial^\alpha}{\partial t^\alpha}U(t) + AU(t) = F(U(t)), \quad U(0) = U_0, \tag{3}$$

where A is an $N_x \times N_x$ matrix, $F : \mathbb{R}^{N_x} \to \mathbb{R}^{N_x}$ is a vector–valued function and $U(t) = \big(U_1(t), U_2(t), \ldots, U_{N_x}(t)\big)^T$ with $U_j(t) = u(x_j,t)$, $j = 1,\ldots,N_x$. The initial value U_0 is directly derived from the initial condition and the source term $F(U(t))$ collects not only information related to (possibly) non–linear terms but also discretized boundary conditions.

There are many well developed time–discretization schemes to solve the semilinear system (3); we just refer, for instance, to [6,11,13,14,25,28].

Numerical schemes can be roughly divided into explicit and implicit methods; the choice of the most appropriate strictly depends on the properties of the problem.

One of the more distinctive features of systems coming from spatial discretization of PDEs is their size. Regardless of the approach used to discretize the space variables, the dimension of the resulting system can indeed be very large, up to a dimension in the range $10^4 \sim 10^6$ for 3D modeling.

For problems of large size the use of explicit integrators is often precluded. Indeed, as the size of the problem increases (this means finer and more accurate mesh partitions for spatial discretization) the spectrum of A usually includes eigenvalues with a very large real part, thus producing stiff problems which are hardly solved by means of explicit methods.

This shortcoming, which is well–understood in the numerical simulation of differential systems of integer order, is sometimes overlooked within fractional problems. To better illustrate it, let us focus on the simple generalization of the heat equation, i.e. problem (2) with $\varepsilon = \rho = 0$, and hence $F(U) = 0$ after assuming homogeneous

boundary conditions. The resulting semi–discrete system can be integrated, for instance, by means of the explicit Grünwald–Letnikov scheme [5,38]

$$\sum_{k=0}^{n} \omega_{n-k}^{(\alpha)} \big(U_k - U_0\big) = h^{\alpha} A U_{n-1} \tag{4}$$

with $\omega_n^{(\alpha)}$ the coefficients in the power series expansion of $(1-\xi)^{\alpha}$, $h > 0$ the time step–size and U_k the approximation for $U(kh)$. Method (4) belongs to the more general class of fractional Adams–Bashforth methods discussed in [13] and developed on the basis of the work in [28]. By means of the linear stability analysis carried out in [10] it is easy to see that, in order to obtain stable solutions, it is required that $-2^{\alpha} < -h^{\alpha}\lambda < 0$, with $\lambda = 4\nu/\Delta x^2$ the dominant eigenvalue of A. Thus, the counterpart for this TFPDE of the Courant-Friedrichs-Lewy (CFL) condition is

$$\frac{h^{\alpha}}{\Delta x^2} < \frac{2^{\alpha}}{4\nu}.$$

Whenever α is smaller then 1, the above CFL condition is stronger with respect to integer–order PDEs and it can turn out in the requirement of a very small step–size h to ensure stability.

With fractional–order systems, which are characterized by the presence of a persistent memory, the use of a small-step size entails a large amount of computation to cover the whole interval of integration, thus making the process of integration (especially over long time intervals) very slow and demanding.

Unlike explicit methods, implicit schemes do not suffer from these serious limitations and present better stability properties; thus they are usually preferred for the numerical treatment of stiff problems. However, the presence of a nonlinearity in (3) is a source of computational complexity since the solution, at each integration step, of a non linear system (by means of some Newton–like iterations) can be exceedingly demanding for problems of large dimension.

3 Exponential integrators for fractional problems

An alternative approach, named as exponential integrators (EIs), aims to solve efficiently semilinear stiff systems by applying explicit integrators just to the non-linear term, which is usually non–stiff, and solving exactly the linear stiff term by means of matrix functions [21].

Although EIs have been introduced since early 1960s (see [32] for an historical review), for several years they have been considered of little practical interest because the difficulty in evaluating functions with matrix arguments of, possibly, large dimension. In recent years the spread of efficient methods to perform this task [19] has renewed the interest in EIs.

Anyway, the application of EIs to time–fractional problems is a subject which so far has not been properly investigated.

A straightforward way to derive EIs for the semilinear system (3) is to express the true solution by means of the variation–of–constant formula

$$U(t) = e_{\alpha,1}(t;A)U_0 + \int_0^t e_{\alpha,\alpha}(t-s;A)F(U(s))ds, \tag{5}$$

where the function $e_{\alpha,\beta}(t;\lambda)$ denotes the generalization of the inverse of the Laplace transform $s^{\alpha-\beta}/(s^{\alpha}+\lambda)$ to the matrix argument A (see, for instance, [16] or [38]).

It is well–known (e.g., see [31,38]) that $e_{\alpha,\beta}(t;\lambda)$ can be written in terms of the Mittag–Leffler (ML) function

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + \beta)} \tag{6}$$

according to

$$e_{\alpha,\beta}(t;\lambda) = t^{\beta-1} E_{\alpha,\beta}(-t^\alpha \lambda). \tag{7}$$

The following result on the integration of the $e_{\alpha,\beta}$ function will be used later on.

Lemma 1 *Let $a < b \le t$ and $\alpha, \beta > 0$. Then for any $z \in \mathbb{C}$*

$$\int_a^b e_{\alpha,\beta}(t-s;z)ds = e_{\alpha,\beta+1}(t-a;z) - e_{\alpha,\beta+1}(t-b;z).$$

Proof First split the integral in two, say $\int_a^b = \int_a^t - \int_b^t$; equation (1.100) in [38] is hence applied twice to conclude the proof after elementary changes of variable. □

There are many equivalent ways to generalize a scalar function to matrix arguments; we refer to the introductory part of [19] for a general reference to this subject. We just recall here that any matrix $A \in \mathbb{R}^{m\times m}$ can be expressed in its Jordan canonical form $A = ZJZ^{-1}$, where J is the diagonal block matrix $J = diag(J_1, J_2, \ldots, J_p)$ and

$$J_k = \begin{bmatrix} \lambda_k & 1 & & \\ & \lambda_k & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_k \end{bmatrix} \in \mathbb{C}^{m_k \times m_k}, \quad k = 1, \ldots, p,$$

are the Jordan blocks of size $m_k \times m_k$, with $m_1 + m_2 + \cdots + m_p = m$; thus, a function $f(x)$ can be evaluated at the matrix argument A as

$$f(A) = Z f(J) Z^{-1} = Z\, diag\big(f(J_1), f(J_2), \ldots, f(J_p)\big) Z^{-1}, \tag{8}$$

where

$$f(J_k) = \begin{bmatrix} f(\lambda_k) & f'(\lambda_k) & \ldots & \frac{f^{(m_k-1)}(\lambda_k)}{(m_k-1)!} \\ & f(\lambda_k) & \ddots & \vdots \\ & & \ddots & f'(\lambda_k) \\ & & & f(\lambda_k) \end{bmatrix}. \tag{9}$$

Note that λ_k's are the eigenvalues of A and the above approach just requires that the function and some derivatives are defined on the spectrum of A.

Whenever A is diagonalizable (a situation which is very common with matrix arising from semi–discretization of PDEs), all the Jordan blocks have size $m_k = 1$ and J is diagonal; thus we can write $A = ZDZ^{-1}$, with $D = diag\big(\lambda_1, \lambda_2, \ldots, \lambda_m\big)$, and

$$f(A) = Z f(D) Z^{-1} = Z \begin{pmatrix} f(\lambda_1) & & & \\ & f(\lambda_2) & & \\ & & \ddots & \\ & & & f(\lambda_m) \end{pmatrix} Z^{-1}, \tag{10}$$

where now the columns of Z are the eigenvectors of A.

To exploit (5) for numerical computation, given on the interval $[0,T]$ a mesh–grid $t_n = nh$, $n = 0,1,\ldots$, which for simplicity we assume equispaced with step–size h, we first write (5) in the piece–wise manner

$$U(t_n) = e_{\alpha,1}(t_n;A)U_0 + \sum_{j=0}^{n-1}\int_{t_j}^{t_{j+1}} e_{\alpha,\alpha}(t_n - s;A)F(U(s))ds; \tag{11}$$

hence in each subinterval $[t_j, t_{j+1}]$ we approximate the vector field $F(U(s))$ by the constant value $F(U_j)$, where U_j stands for the approximation of $U(t_j)$. The application of Lemma 1 into (11) leads to

$$U_n = e_{\alpha,1}(t_n;A)U_0 + \sum_{j=0}^{n-1} W_{n,j}F(U_j), \tag{12}$$

where the weights $W_{n,j}$ are matrix functions defined as

$$W_{n,j} = e_{\alpha,\alpha+1}(t_n - t_j;A) - e_{\alpha,\alpha+1}(t_n - t_{j+1};A).$$

In order to study the main properties of (12) it is useful to first introduce the following result on the matrix weights $W_{n,j}$.

Proposition 1 *Let $0 < \alpha < 1$ and A a diagonalizable matrix with real and positive eigenvalues. There exist a positive constant M depending only on α and A such that for any $n = 1,2,\ldots$*

$$\big\|W_{n,j}\| \le Mh^\alpha (n-j)^{\alpha-1}, \quad j = 0,1,\ldots,n-1.$$

Proof First observe that $E_{\alpha,\beta}(z)$ is a positive and monotonically decreasing function for $z < 0$ and $\beta \ge \alpha$ [40]. Thus, for any real $\lambda > 0$ it is immediate to verify that

$$e_{\alpha,\beta}(t;\lambda) \le t^{\beta-1}E_{\alpha,\beta}(0) = \frac{t^{\beta-1}}{\Gamma(\beta)}. \tag{13}$$

Thanks to (8), it is

$$\|W_{n,j}\| \le \mathcal{K}(Z)\cdot \max_{k=1,p}\big|e_{\alpha,\alpha+1}(t_n - t_j;\lambda_k) - e_{\alpha,\alpha+1}(t_n - t_{j+1};\lambda_k)\big|$$

with $\mathcal{K}(Z) = \|Z\|\cdot\|Z^{-1}\| \ge 1$ the condition number of Z and λ_k, $k = 1,\ldots,p$, the eigenvalues of A.

When $j = n-1$, under the assumption that λ_k are real and positive, the proof is a direct consequence of (13). For $j = 0,\ldots,n-2$, by applying classical Taylor expansions in the definition (6) of the ML function, we may assert the existence of $\xi_j \in [t_j, t_{j+1}]$ such that

$$e_{\alpha,\alpha+1}(t_n - t_j;\lambda) - e_{\alpha,\alpha+1}(t_n - t_{j+1};\lambda) = he_{\alpha,\alpha}(t_n - \xi_j;\lambda)$$

and, thanks again to (13), it is

$$e_{\alpha,\alpha}(t_n - \xi_j;\lambda) \le \frac{1}{\Gamma(\alpha)}(t_n - \xi_j)^{\alpha-1} \le \frac{1}{\Gamma(\alpha)}(t_n - t_{j+1})^{\alpha-1}.$$

It is therefore immediate to show that $\|W_{n,j}\| \le \mathcal{K}(Z)h^\alpha(n-j-1)^{\alpha-1}/\Gamma(\alpha)$ and the proof follows after observing, by means of standard arguments, that $(n-j-1)^{\alpha-1} \le 2^{1-\alpha}(n-j)^{\alpha-1}$. □

Because the presence of the factor $\mathcal{K}(Z)$, one could expect a very large value of M when the spatial mesh width Δx is very small; however, A is frequently normal and hence Z unitary, thus having minimum values for $\mathcal{K}(Z)$.

The convergence of the exponential integrator (12) can be proved by the same kind of arguments used to prove the convergence of standard discretization methods for classical weakly singular Volterra integral equations [8].

Theorem 1 *Let $0 < \alpha < 1$, A diagonalizable with real and positive eigenvalues and F Lipschitz continuous. The error $E_n = \|U(t_n) - U_n\|$ of the exponential integrator (12) satisfies*

$$E_n \leq C_1 h + C_2 t_n^{\alpha-1} h^{1+\alpha}$$

for some positive constants C_1 and C_2.

Proof Let us write the true solution of (5) as

$$U(t_n) = e_{\alpha,1}(t_n; A)U_0 + \sum_{j=0}^{n-1} W_{n,j} F(U(t_j)) + T_n, \tag{14}$$

where T_n is the quadrature error

$$T_n = \sum_{j=0}^{n-1} \int_{t_j}^{t_{j+1}} e_{\alpha,\alpha}(t_n - s; A)\Big(F(U(s)) - F(U(t_j))\Big) ds.$$

Since F is Lipschitz continuous, by subtracting (12) form (14), we observe that

$$E_n \leq \|T_n\| + \sum_{j=0}^{n-1} \|W_{n,j}\| \|F(U(t_j)) - F(U_j)\| \leq \|T_n\| + L \sum_{j=0}^{n-1} \|W_{n,j}\| E_j$$

for some constant $L > 0$; hence, thanks to Proposition 1, it is

$$E_n \leq \|T_n\| + L M_1 h^\alpha \sum_{j=0}^{n-1} (n-j)^{\alpha-1} E_j.$$

To study the term $\|T_n\|$ use again the Lipschitz continuity of F

$$\|T_n\| \leq L \cdot \mathcal{K}(Z) \sum_{j=0}^{n-1} \int_{t_j}^{t_{j+1}} \|e_{\alpha,\alpha}(t_n - s; D)\| \cdot \|U(s) - U(t_j)\| ds,$$

with $A = ZDZ^{-1}$ and D the diagonal matrix of the eigenvalues of A. After denoting $M_2 = L \cdot \mathcal{K}(Z)/\Gamma(\alpha)$, the application of (13) leads to

$$\|T_n\| \leq M_2 \sum_{j=0}^{n-1} \int_{t_j}^{t_{j+1}} (t_n - s)^{\alpha-1} \|U(s) - U(t_j)\| ds.$$

The behavior of the true solution $U(t)$ has been investigated by Lubich [27] who proved that $U(t)$ expands in mixed powers of integer and fractional order according to $U(t) = C_{0,0} + C_{0,1} t^\alpha + C_{0,2} t^{2\alpha} + C_{1,0} t + C_{1,1} t^{1+\alpha} + \ldots$. Thus, in the vicinity of the origin, say in an interval $[0, t_p]$, $\|U(s) - U(t_j)\|$ can be bounded, for $s \in (t_j, t_{j+1}]$, by a constant times h^α whereas, as t goes away from the origin, a bound proportional

to h is indeed involved. Hence, by following a reasoning similar to that in [8, Section 3.] we obtain

$$\|T_n\| \leq M_3 h^{1+\alpha} t_n^{\alpha-1} + M_4 h,$$

with M_3 and M_4 some positive constants. The proof thus follows by applying Theorem 2.1. in [8]. □

Theorem 1 states first order convergence for the EI (12), in accordance with convergence of the classical rectangular product integration rule for FDEs [7]. Indeed, the advantage of EIs with respect to standard approaches does not actually lie in accuracy rather in numerical stability: since the linear (and usually stiff) term of (3) is solved exactly by the ML function, strong restrictions on the step–size are no more necessary and the explicit scheme (12) is able to provide suitable results with a reasonably large step–size, as we will show in a subsequent section.

4 Implementation issues

The evaluation of weights W_n's in the EI (12) involves the computation of ML functions with matrix arguments. It is well–known that this is not a trivial task, especially when the size of the matrix is large.

In the last years great efforts have been devoted to device efficient methods for evaluating matrix functions [19], with special emphasis on exponential and related functions [3,4,20,26,33,35,39]. Only recently the attention has moved towards the extension of some of these techniques to ML functions [17,18,15,34,36,37].

To avoid unnecessary too technical details (which are outside the scope of this paper) we propose a very simple (but effective in several applications) implementation of (12). Given A in its Jordan canonical form $A = ZJZ^{-1}$, after putting $\widetilde{U}_j = Z^{-1}U_j$ and $\widetilde{F}_j = Z^{-1}F_j$ we can write (12) equivalently as $U_n = Z\widetilde{U}_n$ with

$$\widetilde{U}_n = e_{\alpha,1}(nh; J)\widetilde{U}_0 + e_{\alpha,\alpha+1}(nh; J)\widetilde{F}_0 + \sum_{j=1}^{n-1} e_{\alpha,\alpha+1}((n-j)h; J)\nabla\widetilde{F}_j, \tag{15}$$

where $\nabla\widetilde{F}_j = \widetilde{F}(U_j) - \widetilde{F}(U_{j-1})$ are standard finite differences and the evaluation of the ML function with the argument J is performed by means of (9). Note that the above formulation is not expensive and just adds at each step the solution of a single linear system and a matrix–vector product, as summarized in the Algorithm 1.

Algorithm 1 Practical implementation of the EI (12)

Solve the system $Z\widetilde{U}_0 = U_0$
Solve the system $Z\widetilde{F}_0 = F(U_0)$
for n=1,2,... **do**
  Evaluate $\widetilde{U}_n$ by equation (15)
  $U_n = Z\widetilde{U}_n$
  Solve the system $Z\widetilde{F}_n = F(U_n)$
end for

In most cases the matrix A is diagonalizable and hence, due to (10), just the values of the ML $e_{\alpha,\alpha+1}(t; \lambda)$ on scalar arguments λ have to be computed. Moreover, in several situations an explicit formulation of eigenvalues and eigenvectors of A is

available, thus avoiding the ill–conditioned problem of diagonalizing A. For instance, it is a well–known result that for the 1–dimensional heat equation obtained from (2) with $\varepsilon = \rho = 0$ the eigenvalues of A are

$$\lambda_\ell = -\frac{2\nu}{\Delta x^2}\left(1 - \cos\frac{\ell\pi}{N_x+1}\right), \quad \ell = 1, 2, \ldots, N_x$$

and the corresponding eigenvectors can be expressed as

$$z_\ell = \left(\sin\frac{\ell\pi}{N_x+1}, \sin\frac{2\ell\pi}{N_x+1}, \ldots, \sin\frac{N_x\ell\pi}{N_x+1}\right)^T \in \mathbb{R}^{N_x}.$$

In [42] it has been analyzed the computation of some functions, also with matrix arguments, by inverting numerically their Laplace transform on well suited contours. This technique has been successively proved [18] to apply, with some modifications, also for the generalized ML function (7). According to the analysis in the aforementioned papers, the value of $e_{\alpha,\beta}(t; J)$ can be approximated by means of

$$e^{[K]}_{\alpha,\beta}(t; J) = \frac{\delta}{2\pi i}\sum_{k=-K}^{K} e^{z_k t} z_k^{\alpha-\beta} z'_k \left(z_k^\alpha I + J\right)^{-1} \tag{16}$$

where $z_k = z(\delta k)$, $k = -K, \ldots, K$ are equispaced nodes (with step–size $\delta > 0$) on the parabolic contour $z(u) = \mu(iu + 1)^2$ in the complex plane and $z'_k = z'(\delta k)$. Here the parameters δ, μ and K are chosen, in order to reduce the computational cost to a minimum, by balancing the different components of the error. The analysis performed in [42] suggests to select

$$\delta = \frac{3}{K}, \quad \mu = \frac{\pi N}{12t}$$

to obtain an error proportional to $\mathcal{O}\left(e^{-2\pi K/3}\right)$. Therefore, to ensure the error below a certain tolerance $\tau > 0$, it is sufficient to use a number of nodes given by

$$K = K(\tau) > -\frac{3}{2\pi}\log\tau.$$

(as discussed in [18], these parameters turn out to be optimal for ML functions $e_{\alpha,\beta}$ with low values of β; when $\beta > \alpha + 1$ a more involved analysis is necessary).

Since in (15) we are interested in the evaluation of products of $e_{\alpha,\alpha+1}(n, J)$ times a vector, the main task when using (16) is the solution of a sequence of linear systems with coefficient matrix $z_k^\alpha I + J$. In the case A is diagonalizable, this is obviously a not expensive task which can be performed in a number of floating–point operations proportional to the size of A. In different instances, and for systems of large size, Krylov subspace methods are usually preferable.

When J is real we can exploit the symmetry of the parabolic contour with respect to the real axis to almost halve the computational effort by the equivalent formula

$$e^{[K]}_{\alpha,\beta}(t_n; J) = \frac{\delta}{\pi}{\sum_{k=0}^{K}}' \Im\left(e^{z_{k,n}t_n} z_{k,n}^{\alpha-\beta} z'_{k,n}\left(z_{k,n}^\alpha I + J\right)^{-1}\right),$$

where $\Im$ denotes the imaginary part and the $'$ symbol in the summation means that the first term (i.e, the one for $k = 0$) must be halved (we have introduced the notation $z_{k,n}$ to stress the dependence of the nodes on the time–step $t_n = nh$).

Interestingly, when $0 < \alpha < 1$ just the location of the nodes in the complex plane changes according to the time point t_n but their number K and the step–size δ on

the parabola remain constant throughout the integration process. This observation allows to recast the computation process in order to minimize storage requirements and number of operations. Indeed, after introducing for shortness

$$r_{k,n}^{(\beta)} = \begin{cases} \frac{1}{2} e^{z_{k,n} t_n} z_{k,n}^{\alpha-\beta} z'_{k,n} & k = 0 \\ e^{z_{k,n} t_n} z_{k,n}^{\alpha-\beta} z'_{k,n} & k \neq 0 \end{cases}, \quad E_{k,n}^{(\beta)} = \Im \left(r_{k,n}^{(\beta)} \left(z_{k,n}^{\alpha} I + J \right)^{-1} \right)$$

we can rewrite (15) as

$$\widetilde{U}_n = \frac{\delta}{\pi} \sum_{k=0}^{K} \left(E_{k,n}^{(1)} \widetilde{U}_0 + E_{k,n}^{(\alpha+1)} \widetilde{F}_0 + \sum_{j=1}^{n-1} E_{k,n-j}^{(\alpha+1)} \nabla \widetilde{F}_j \right),$$

thus allowing a fully parallel implementation with respect to index k. We just remark that K is usually not very large (a value of $K = 16$ returns an approximation with an accuracy close to the machine precision); moreover, for diagonalizable matrices, each $E_{k,n}^{(\beta)}$ can be stored as a vector and $e_{\alpha,\beta}^{[K]}(t_n; J)$ is evaluated with a reduced effort.

5 Numerical experiments

In order to verify the potentials of EIs for solving TFPDEs we present in this section some numerical experiments. All the simulations have been performed in Matlab, version 7.9.0.529, on a computer equipped with the Intel Dual Core E5400 processor running at 2.70 GHz.

For some selected test problems we have used the EI (12) and compared the results with those obtained by an implicit first order fractional Adams–Moulton (FAM) method [11] which generalizes to fractional–order problems the classical Adams–Moulton methods for ODEs. It is not advisable to use explicit methods because of the numerical instability which would provide meaningless results.

In all the experiments a tolerance of $\tau = 10^{-12}$ has been used in the evaluation of the ML function as described in Section 4; this is a sufficiently high level of accuracy with respect to the convergence order of the method under investigation.

Time–fractional heat equation

As a first test problem we consider the two–dimensional time–fractional heat equation

$$\frac{\partial^{\alpha}}{\partial t^{\alpha}} u(t,x,y) = \nu \left(\frac{\partial^2}{\partial x^2} u(t,x,y) + \frac{\partial^2}{\partial y^2} u(t,x,y) \right)$$

on the unit square domain $\Omega = [0,1] \times [0,1]$ with initial function $u(0,x,y) = x(1-x)y(1-y)$ and compatible homogeneous boundary conditions on $\partial\Omega$.

For such linear problems the advantages of EIs are expected to be better exploited; indeed, the source term in the resulting semi–discrete system (3) is a constant function $F(U) = V$, being V a vector containing just discretized boundary conditions, and the numerical solution can be evaluated directly in each grid–points t_n by the EI (12) without implementing a step–by–step procedure.

In our simulations we have considered $\alpha = 0.7$, $\nu = 0.1$ and a mesh grid of size $N_x = N_y = 40$ on Ω. The left plot in Figure 1 shows the numerical solution at $t = 1$ obtained by means of the EI (12) with a step–size $h = 2^{-4}$. To verify the goodness of

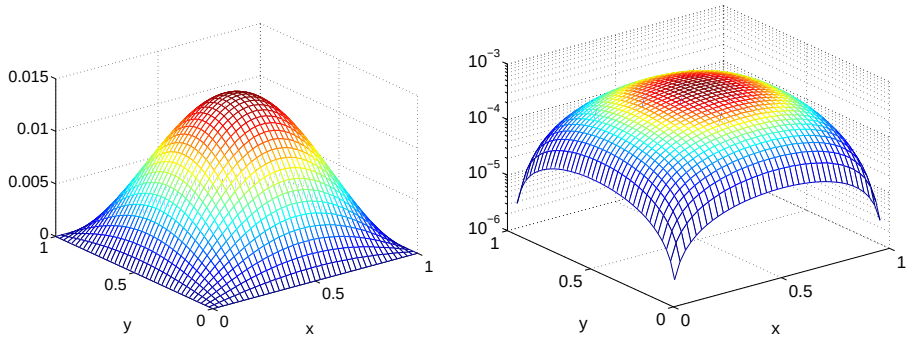

Fig. 1. Fractional heat equation. solution by the EI at $t = 1$ with $N_x = N_y = 40$ and $h = 2^{-4}$ (left) and difference between the solutions obtained by EI and FAM (right)

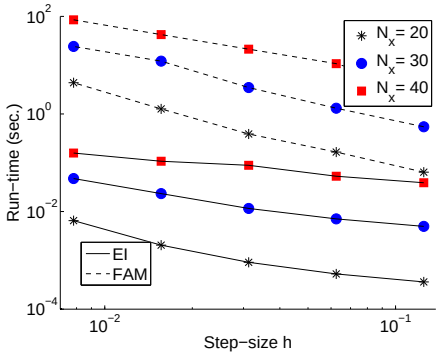

Fig. 2. Fractional heat equation: comparison of execution times between EI (solid lines) and FAM method (dashed lines).

this approximation we plot in the right part of Figure 1 the difference with respect to the FAM method described in the introductory part of this section.

The outcome turns up to be very satisfactory, being the difference between the two solutions under the expected error as stated in Theorem 1.

A more interesting comparison concerns with the execution time in dependence of various step–sizes h used to perform the computation on the same time interval $[0, 1]$. As clearly shown in Figure 2, the exponential approach performs the computation in a CPU time which is several orders of magnitude shorter with respect to the classical FAM method. As the size of the problem becomes larger and larger, it is noticeable that the FAM method involves a computational time that can turn out to be impracticable for most applications.

Fractional advection–diffusion-reaction equation

For our second test we consider the fractional ADR equation (2) with constant coefficients $\varepsilon = -1$, $\nu = 0.1$, $\rho = 1/2$ and the nonlinear reaction term $f(u) = u(u - 1/2)(u - 1)$ (a similar test problem for an integer–order equation has been used in [1]). In our experiment we consider the 1D domain $[0, 1]$ and the value $\alpha = 0.7$ for the fractional order .

The results are presented in Figure 3 where, as for the previous experiment, we show both the numerical solution obtained by the EI (12) and the difference with

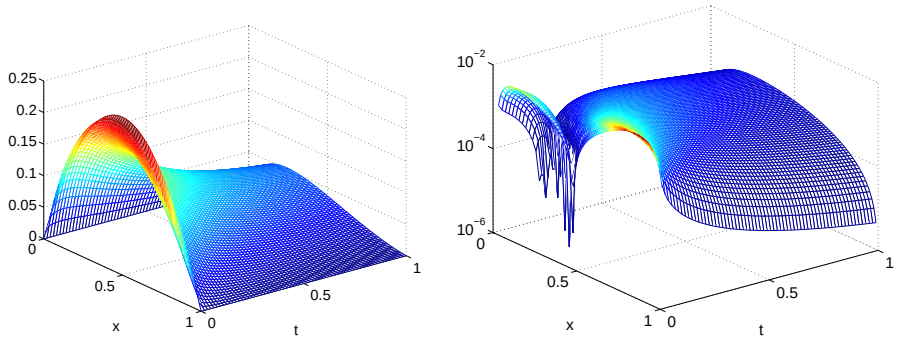

Fig. 3. Fractional ADR equation: solution by the EI at $t = 1$ with $N_x = 80$ and $h = 2^{-6}$ (left) and difference between the solutions obtained by EI and FAM (right)

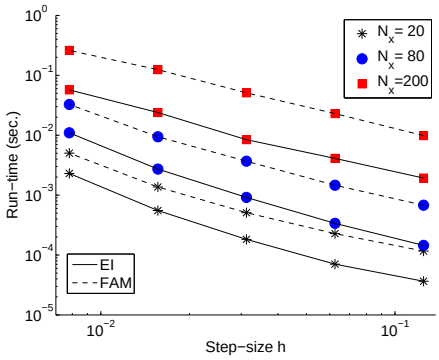

Fig. 4. Fractional ADR equation: comparison of execution times between EI (solid lines) and FAM method (dashed lines).

respect to the classical FAM method; a spatial mesh–grid with $N_x = 80$ has been used together with a step–size $h = 2^{-6}$ for the time integration.

Also in this case the approximation provided by the EI is within the same accuracy of the FAM method. Anyway, although in a not so striking way as for the previous example, we appreciate from Figure 4 that the EI performs the computation, also for this test problem presenting a more involved non–linearity (due to the presence of the reaction term), in a faster way with respect to the classical FAM method.

This work is supported by the project "High-performance robust fractional controllers for applications in industry and mechatronics", which is part of the national project PRIN 2009F4NZJP "Non integer order systems in modeling and control", funded by the Italian Ministry of University and Research (MIUR).

References

1. M. Caliari and A. Ostermann. Implementation of exponential Rosenbrock-type integrators. *Appl. Numer. Math.*, 59(3-4):568–581, 2009.
2. M. Caputo. Models of flux in porous media with memory. *Water Resources Research*, 36(3):693–705, 2000.

3. N. Del Buono and L. Lopez. A survey on methods for computing matrix exponentials in numerical schemes for ODEs. In *Computational science—ICCS 2003. Part II*, volume 2658 of *Lecture Notes in Comput. Sci.*, pages 111–120. Springer, Berlin, 2003.
4. F. Diele, I. Moret, and S. Ragni. Error estimates for polynomial Krylov approximations to matrix functions. *SIAM J. Matrix Anal. Appl.*, 30(4):1546–1565, 2008/09.
5. K. Diethelm. *The analysis of fractional differential equations*, volume 2004 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 2010.
6. K. Diethelm, N. J. Ford, and A. D. Freed. A predictor-corrector approach for the numerical solution of fractional differential equations. *Nonlinear Dynam.*, 29(1-4):3–22, 2002.
7. K. Diethelm, N. J. Ford, and A. D. Freed. Detailed error analysis for a fractional Adams method. *Numer. Algorithms*, 36(1):31–52, 2004.
8. J. Dixon. On the order of the error in discretization methods for weakly singular second kind Volterra integral equations with nonsmooth solutions. *BIT*, 25(4):624–634, 1985.
9. V. Gafiychuk and B. Datsko. Mathematical modeling of different types of instabilities in time fractional reaction-diffusion systems. *Comput. Math. Appl.*, 59(3):1101–1107, 2010.
10. L. Galeone and R. Garrappa. On multistep methods for differential equations of fractional order. *Mediterr. J. Math.*, 3(3-4):565–580, 2006.
11. L. Galeone and R. Garrappa. Fractional Adams-Moulton methods. *Math. Comput. Simulation*, 79(4):1358–1367, 2008.
12. R. Garra. Fractional-calculus model for temperature and pressure waves in fluid-saturated porous rocks. *Phys. Rev. E*, 84:036605, Sep 2011.
13. R. Garrappa. On some explicit Adams multistep methods for fractional differential equations. *J. Comput. Appl. Math.*, 229(2):392–399, 2009.
14. R. Garrappa. Stability-preserving high-order methods for multiterm fractional differential equations. *Int. J. Bifurcat. Chaos*, 22(04):1250073, 2012.
15. R. Garrappa. A family of Adams exponential integrators for fractional linear systems. *Comput. Math. Appl.*, 2013. doi:10.1016/j.camwa.2013.01.022.
16. R. Garrappa and M. Popolizio. On accurate product integration rules for linear fractional differential equations. *J. Comput. Appl. Math.*, 235(5):1085–1097, 2011.
17. R. Garrappa and M. Popolizio. On the use of matrix functions for fractional partial differential equations. *Math. Comput. Simulation*, 81(5):1045–1056, 2011.
18. R. Garrappa and M. Popolizio. Evaluation of generalized Mittag–Leffler functions on the real line. *Adv. Comput. Math.*, pages 1–21, 2012. doi:10.1007/s10444-012-9274-z.
19. N. J. Higham. *Functions of matrices*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008.
20. M. Hochbruck and C. Lubich. On Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 34(5):1911–1925, 1997.
21. M. Hochbruck and A. Ostermann. Exponential integrators. *Acta Numer.*, 19:209–286, 2010.
22. W. Hundsdorfer and J. Verwer. *Numerical solution of time-dependent advection-diffusion-reaction equations*, volume 33 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2003.
23. A. A. Kilbas, H. M. Srivastava, and J.J. Trujillo. *Theory and applications of fractional differential equations*, volume 204 of *North-Holland Mathematics Studies*. Elsevier Science B.V., Amsterdam, 2006.
24. R. J. LeVeque. *Finite difference methods for ordinary and partial differential equations*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2007.
25. C. Li and F. Zeng. The finite difference methods for fractional ordinary differential equations. *Numerical Functional Analysis and Optimization*, 34(2):149–179, 2013.
26. L. Lopez and V. Simoncini. Analysis of projection methods for rational function approximation to the matrix exponential. *SIAM J. Numer. Anal.*, 44(2):613–635, 2006.
27. C. Lubich. Runge-Kutta theory for Volterra and Abel integral equations of the second kind. *Math. Comp.*, 41(163):87–102, 1983.
28. C. Lubich. Discretized fractional calculus. *SIAM J. Math. Anal.*, 17(3):704–719, 1986.

29. F. Mainardi. Fractional relaxation-oscillation and fractional diffusion-wave phenomena. *Chaos Solitons Fractals*, 7(9):1461–1477, 1996.
30. F. Mainardi. Fractional calculus: some basic problems in continuum and statistical mechanics. In *Fractals and fractional calculus in continuum mechanics (Udine, 1996)*, volume 378 of *CISM Courses and Lectures*, pages 291–348. Springer, 1997.
31. F. Mainardi. *Fractional calculus and waves in linear viscoelasticity*. Imperial College Press, London, 2010.
32. B. V. Minchev and W. M. Wright. A review of exponential integrators for first order semi-linear problems. Technical Report 2/2005, Norwegian University of Science and Technology, Trondheim, Norway, 2005.
33. C. Moler and C. Van Loan. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Rev.*, 45(1):3–49, 2003.
34. I. Moret. A note on Krylov methods for fractional evolution problems. *Numer. Func. Anal. Opt.*, 34(5):539–556, 2013.
35. I. Moret and P. Novati. RD-rational approximations of the matrix exponential. *BIT*, 44(3):595–615, 2004.
36. I. Moret and P. Novati. On the convergence of Krylov subspace methods for matrix Mittag-Leffler functions. *SIAM J. Numer. Anal.*, 49(5):2144–2164, 2011.
37. I. Moret and M. Popolizio. The restarted shift-and-invert Krylov method for matrix functions. *Numerical Linear Algebra with Applications*, 2012. doi:10.1002/nla.1862.
38. I. Podlubny. *Fractional differential equations*, volume 198 of *Mathematics in Science and Engineering*. Academic Press Inc., San Diego, CA, 1999.
39. M. Popolizio and V. Simoncini. Acceleration techniques for approximating the matrix exponential operator. *SIAM J. Matrix Anal. Appl.*, 30(2):657–683, 2008.
40. W. R. Schneider. Completely monotone generalized Mittag-Leffler functions. *Exposition. Math.*, 14(1):3–16, 1996.
41. W. R. Schneider and W. Wyss. Fractional diffusion and wave equations. *J. Math. Phys.*, 30(1):134–144, 1989.
42. J. A. C. Weideman and L. N. Trefethen. Parabolic and hyperbolic contours for computing the Bromwich integral. *Math. Comp.*, 76(259):1341–1356, 2007.
43. P. Zhuang and F. Liu. Implicit difference approximation for the time fractional diffusion equation. *J. Appl. Math. Comput.*, 22(3):87–99, 2006.