Springer INdAM Series 50

Angelamaria Cardone · Marco Donatelli · Fabio Durastante · Roberto Garrappa · Mariarosa Mazza · Marina Popolizio *Editors*

Fractional Differential Equations

Modeling, Discretization, and Numerical Solvers

Springer

Springer INdAM Series

Volume 50

This series will publish textbooks, multi-authors books, thesis and monographs in English language resulting from workshops, conferences, courses, schools, seminars, doctoral thesis, and research activities carried out at INDAM - Istituto Nazionale di Alta Matematica, http://www.altamatematica.it/en. The books in the series will discuss recent results and analyze new trends in mathematics and its applications.
THE SERIES IS INDEXED IN SCOPUS

Angelamaria Cardone • Marco Donatelli • Fabio Durastante • Roberto Garrappa • Mariarosa Mazza • Marina Popolizio
Editors

Fractional Differential Equations

Modeling, Discretization, and Numerical Solvers

Springer

Editors
Angelamaria Cardone
Department of Mathematics
University of Salerno
Fisciano, Salerno, Italy

Marco Donatelli
Department of Science and High Technology
University of Insubria
Como, Italy

Fabio Durastante
Department of Mathematics
University of Pisa
Pisa, Italy

Roberto Garrappa
Department of Mathematics
University of Bari Aldo Moro
Bari, Italy

Mariarosa Mazza
Department of Science and High Technology
University of Insubria
Como, Italy

Marina Popolizio
Department of Electrical and Information Engineering
Polytechnic University of Bari
Bari, Italy

ISSN 2281-518X ISSN 2281-5198 (electronic)
Springer INdAM Series
ISBN 978-981-19-7715-2 ISBN 978-981-19-7716-9 (eBook)
https://doi.org/10.1007/978-981-19-7716-9

Preface

This volume in the Springer INdAM series collects some contributions related to the talks presented during the INdAM Workshop "Fractional Differential Equations: Modeling, Discretization, and Numerical Solvers," held in Rome, Italy, from July 12 to 14, 2021.

Fractional calculus deals with the study and application of integrals and derivatives of non-integer order. These operators, unlike the classic operators of integer order, are non-local operators and are better suited to describe phenomena with memory (with respect to time and/or space).

Although the basic ideas of fractional calculus go back over three centuries, only in recent decades there has been a rapid increase in interest in this field of research, due not only to the increasing use of fractional calculus in applications (in biology, physics, engineering, probability, etc.) but also thanks to the availability of new and more powerful numerical tools that allow for an efficient solution of problems that until a few years ago appeared unsolvable.

The analytical solution of fractional differential equations (FDEs) appears even more difficult than in the integer case. Hence, numerical analysis plays a decisive role since practically every type of application of fractional calculus requires adequate numerical tools.

The aim of the INdAM workshop was, therefore, to bring together the two communities of numerical analysts operating in this field—the one working on methods for the solution of differential problems and the one working on the numerical linear algebra side—to share knowledge and create synergies. At the same time, the workshop intended to realize a direct bridge between researchers working on applications and numerical analysts.

The spirit of the workshop is therefore reflected in this book which collects papers on applications, numerical methods for differential problems of fractional order, and related aspects in numerical linear algebra.

Salerno, Italy — Angelamaria Cardone
Como, Italy — Marco Donatelli
Pisa, Italy — Fabio Durastante
Bari, Italy — Roberto Garrappa
Como, Italy — Mariarosa Mazza
Bari, Italy — Marina Popolizio
May 2022

Organization

Proceedings Chair	Roberto Garrappa Universitá degli Studi di Bari, Italy
Program Chairs	
Roberto Garrappa	Universitá degli Studi di Bari, Italy
Fabio Durastante	Universitá di Pisa, Italy
Scientific Committee	
Angelamaria Cardone	Universitá degli Studi di Salerno, Italy
Marco Donatelli	University of Insubria, Italy
Fabio Durastante	Universitá di Pisa, Italy
Roberto Garrappa	Universitá degli Studi di Bari, Italy
Mariarosa Mazza	Universitá degli Studi dell'Insubria, Italy
Marina Popolizio	Politecnico di Bari , Italy

Contents

About the Editors

Angelamaria Cardone is Associate Professor of Numerical Analysis in the Department of Mathematics, University of Salerno, Italy. Her scientific activity is mainly focused on the numerical solution of Volterra integral equations and of differential equations, also of fractional type, and on the development of related mathematical software.

Marco Donatelli is Associate Professor of Numerical Analysis at the University of Insubria, Italy. His scientific activity is mainly focused on regularization of inverse problems and numerical linear algebra methods, with special attention to iterative methods for large linear systems arising from the discretization of integral and differential equations.

Fabio Durastante is a researcher in numerical analysis in the Department of Mathematics, University of Pisa, Italy. His scientific activity is mainly focused on numerical linear algebra and its application to the solution of partial differential equations of both integer and fractional order, high-performance computing, and computation of matrix-functions.

Roberto Garrappa is Associate Professor of Numerical Analysis in the Department of Mathematics, University of Bari, Italy. His scientific activity is mainly focused on numerical methods for fractional differential equations and for the computation of special functions in fractional calculus.

Mariarosa Mazza is a researcher in numerical analysis at the University of Insubria, Italy. Her scientific activity is mainly focused on numerical linear algebra problems and related applications, with special attention to iterative methods for

discretized partial differential equations, also of fractional type. Other interests include image deblurring and approximation issues.

Marina Popolizio is Associate Professor of Numerical Analysis at the Polytechnic University of Bari, Italy. Her scientific activity is mainly focused on numerical linear algebra and numerical methods for fractional differential equations, with special attention to the computation of matrix functions.

A New Diffusive Representation for Fractional Derivatives, Part I: Construction, Implementation and Numerical Examples

Kai Diethelm

Abstract Diffusive representations of fractional derivatives have proven to be useful tools in the construction of fast and memory efficient numerical methods for solving fractional differential equations. A common challenge in many of the known variants of this approach is that they require the numerical approximation of some integrals over an unbounded integral whose integrand decays rather slowly, which implies that their numerical handling is difficult and costly. We present a novel variant of such a diffusive representation. This form also requires the numerical approximation of an integral over an unbounded domain, but the integrand decays much faster. This property allows to use well established quadrature rules with much better convergence properties.

1 Introduction and Statement of the Problem

1.1 Classical Discretizations in Fractional Calculus

The efficient numerical solution of initial value problems with fractional differential equations like, e.g.,

$$D_a^\alpha y(t) = f(t, y(t)), \qquad y(a) = y_0, \tag{1}$$

is a significant computational challenge due to, among other reasons, the non-locality of fractional differential operators. In our formulation (1), D_a^α denotes the standard Caputo differential operator of order α with starting point $a \in \mathbb{R}$ [6, Chapter 3], and we assume here and throughout some other parts of this chapter that $0 < \alpha < 1$ (although we explicitly point out that the generalization of our

K. Diethelm (✉)
Faculty of Applied Natural Sciences and Humanities, Technical University of Applied Sciences Würzburg-Schweinfurt, Schweinfurt, Germany
e-mail: kai.diethelm@thws.de

A. Cardone et al. (eds.), *Fractional Differential Equations*,
Springer INdAM Series 50, https://doi.org/10.1007/978-981-19-7716-9_1

findings to the case that α is a non-integer number greater than 1 is a relatively straightforward matter).

When dealing with the problem (1), one usually introduces a discretization of the interval $[a, a+T]$, say, on which the solution is sought by defining some grid points $a = t_0 < t_1 < t_2 < \cdots < t_N = a+T$. For each grid point t_j, $j = 1, 2, \ldots, N$, typical numerical methods then introduce an approximation formula for a discretization of $D_a^\alpha y(t_j)$ based on function values of y at the grid points, replace the exact fractional derivative in Eq. (1) by this approximation, discard the approximation error and solve the resulting algebraic equation to obtain an approximation for $y(t_j)$. In their standard forms, classical methods like fractional linear multistep methods [20, 21] or the Adams method [9, 10] require $O(j)$ operations to compute the required approximation at the j-th grid point, thus leading to an $O(N^2)$ complexity for the overall calculation of the approximate solution at all N grid points. Moreover, the construction of the algorithms requires the entire history of the process to be in the active memory at any time, thus leading to an $O(N)$ memory requirement. This may be prohibitive in situations like, e.g., the simulation of the mechanical behaviour of viscoelastic materials via some finite element code where a very large number of such differential equations needs to be solved simultaneously [18].

Numerous modifications of these basic algorithms have been proposed to resolve these issues. Specifically (see, e.g., [12, Section 3]), one may use FFT techniques to evaluate the sums that arise in the formulas [14–16], thus reducing the overall computational complexity to $O(N \log^2 N)$; however, this approach does not improve the memory requirements. Alternatively, nested mesh techniques [11, 13] can be employed; this typically reduces the computational complexity to $O(N \log N)$, and some of these methods are also able to cut down the active memory requirements to $O(\log N)$.

1.2 Diffusive Representations in Discretized Fractional Calculus

From the properties recalled above, it becomes clear that none of the schemes mentioned so far allows to reach the level known for traditional algorithms for first-order initial value problems that, due to their differential operators being local, have an $O(N)$ complexity and an $O(1)$ memory requirement. However, it is possible to achieve these performance features by using methods based on diffusive representations for the fractional derivatives [24]. Typically, such representations take the form

$$D_a^\alpha y(t) = \int_0^\infty \phi(w,t)\mathrm{d}w \tag{2}$$

where, for a fixed value of w, the function $\phi(w,\cdot)$ is characterized as the solution to an initial value problem for a first-order differential equation the formulation of which contains the function y whose fractional derivative is to be computed. In the presently available literature, many different special cases of this representation are known, e.g. the version of Yuan and Agrawal [29] (originally proposed in that paper for $0 < \alpha < 1$ and extended to $1 < \alpha < 2$ in [28] and to general positive non-integer values of α in [5]; see also [25] for further properties of this method) where the associated initial value problem reads

$$\frac{\partial \phi^{\mathrm{YA}}}{\partial t}(w,t) = -w^2\phi^{\mathrm{YA}}(w,t) + (-1)^{\lfloor\alpha\rfloor}\frac{2\sin\pi\alpha}{\pi}w^{2\alpha-2\lceil\alpha\rceil+1}y^{(\lceil\alpha\rceil)}(t),\ \ \phi^{\mathrm{YA}}(w,a) = 0, \tag{3a}$$

such that the function ϕ^{YA} has the form

$$\phi^{\mathrm{YA}}(w,t) = (-1)^{\lfloor\alpha\rfloor}\frac{2\sin\pi\alpha}{\pi}w^{2\alpha-2\lceil\alpha\rceil+1}\int_a^t y^{(\lceil\alpha\rceil)}(\tau)\exp(-(t-\tau)w^2)\mathrm{d}\tau. \tag{3b}$$

An alternative has been proposed by Chatterjee [3] (see also [26]) using the initial value problem

$$\frac{\partial \phi^{\mathrm{C}}}{\partial t}(w,t) = -w^{1/(\alpha-\lceil\alpha\rceil+1)}\phi^{\mathrm{C}}(w,t) + (-1)^{\lfloor\alpha\rfloor}\frac{\sin\pi\alpha}{\pi(\alpha-\lceil\alpha\rceil+1)}y^{(\lceil\alpha\rceil)}(t),\quad \phi^{\mathrm{C}}(w,a) = 0, \tag{4a}$$

such that the function ϕ^{C} has the form

$$\phi^{\mathrm{C}}(w,t) = \frac{(-1)^{\lfloor\alpha\rfloor}\sin\pi\alpha}{\pi(\alpha-\lceil\alpha\rceil+1)}\int_a^t y^{(\lceil\alpha\rceil)}(\tau)\exp\left(-(t-\tau)w^{1/(\alpha-\lceil\alpha\rceil+1)}\right)\mathrm{d}\tau. \tag{4b}$$

In either case (or in the case of the many variants thereof that have been proposed; cf., e.g., [1, 2, 19, 23, 30]), the numerical calculation of $D_a^\alpha y(t_j)$ requires

1. a quadrature formula

$$\sum_{k=1}^{K}\lambda_k\phi(w_k,t_j) \approx \int_0^\infty \phi(w,t_j)\mathrm{d}w = D_a^\alpha y(t_j) \tag{5}$$

   with nodes $w_1, w_2, \ldots, w_K \in [0,\infty)$ and weights $\lambda_1, \lambda_2, \ldots, \lambda_K \in \mathbb{R}$ for numerically evaluating the integral in Eq. (2),
2. a standard numerical solver for the associated differential equation (e.g., a linear multistep method) to approximately compute, for each $k \in \{1, 2, \ldots, K\}$, the values $\phi(w_k, t_j)$ required to evaluate the formula (5).

Evidently, the run time and the memory requirements of the operation in step 1 do not depend on j. Also, one can perform step 2 in an amount of time that is independent of j. Furthermore, if an ℓ-step method is used in step 2, one needs to have (approximate) information about $y(t_{j-1})$, $y(t_{j-2})$, $\ldots$ $y(t_{j-\ell})$, which has to be kept in the active memory—but the amount of storage space required for this purpose is also independent of j.

In summary, approaches of this type require $O(1)$ arithmetic operations per time step, i.e. we have a computational cost of $O(N)$ for all N time steps combined, and the required amount of memory is $O(1)$ as desired. A further pleasant property of these methods is that they impose no restrictions at all on the choice of the grid points t_j, whereas this can not always be achieved with the other approaches. Thus, from a theoretical point of view, algorithms of this form are very attractive. In practice, however, the implied constants in the O-terms may be very large. This is due to the following observation [6, Theorems 3.20(b) and 3.21(b)]:

Proposition 1 *Let $t \in [a, a+T]$ be fixed. Then, for $w \to \infty$, we have*

$$\phi^{\mathrm{YA}}(w,t) = c^{\mathrm{YA}} w^{q_{\mathrm{YA}}}(1+o(1)) \quad \text{with} \quad q_{\mathrm{YA}} = 2\alpha - 2\lceil\alpha\rceil - 1 \in (-3,-1)$$

and

$$\phi^{\mathrm{C}}(w,t) = c^{\mathrm{C}} w^{q_{\mathrm{C}}}(1+o(1)) \quad \text{with} \quad q_{\mathrm{C}} = -\frac{1}{\alpha - \lceil\alpha\rceil + 1} < -1,$$

where c^{C} and c^{YA} are some constants independent of w (which may, however, depend on t, a, α and y).

From Proposition 1, we can see that the integrands in Eq. (2) decay to zero in an algebraic manner as $w \to \infty$. Figure 1 shows the behaviour of the exponents q_{YA} and q_{C} as they depend on α. It can be seen that the exponents are less than -1 for all $\alpha \in (0,1)$. This suffices to assert that the integrals $\int_0^\infty \phi(w,t)\mathrm{d}w$ are convergent. On the other hand, step 1 of the algorithm outlined above requires to numerically approximate this integral, and to this end, classical results from approximation theory [22] imply that such an algebraic decay does not admit a very fast convergence of such numerical methods. Indeed, as the constant q^{C} is slightly larger than q^{YA} for $\alpha \geq 1/2$ and (significantly) smaller for $\alpha < 1/2$, one may state that overall Chatterjee's method has more preferable properties from this point of view (although its properties are still far from good enough). To the best of the author's knowledge, this is a feature shared by very many algorithms based on this type of approach. Therefore, one needs a relatively large number K of quadrature nodes in Eq. (5) to obtain an approximation with an acceptable accuracy (with the approaches known so far, a common choice for K is in the range between 200 and 500, cf. [1, 18]). This number K clearly has a strong influence on the constants implied in the O-term for the computational complexity estimate. The main goal of this chapter thus is to develop a method that is also based on the same fundamental idea but that leads to a function $\phi(w,t)$ which exhibits an exponential decay for

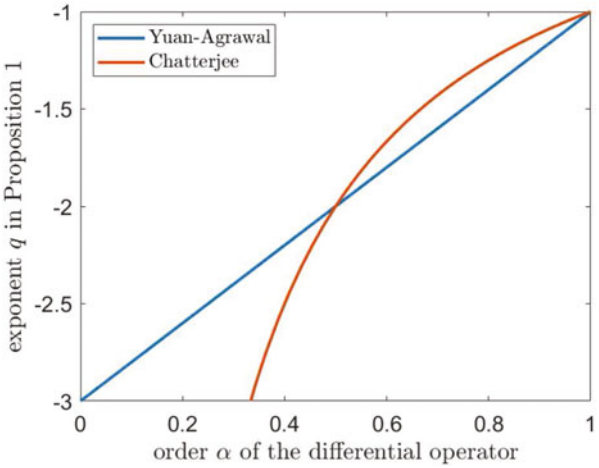


Fig. 1 Behaviour of the exponents q_{YA} (blue) and q_{C} (orange) as introduced in Proposition 1 vs. the order α of the differential operator

large w. This behaviour is much more pleasant from an approximation theoretic point of view because it allows to use well-understood and rapidly convergent classical techniques like Gauss–Laguerre quadrature formulas. The hope behind this idea is that the improved convergence behaviour will admit using quadrature formulas as in Eq. (5) with a significantly smaller number K of nodes, so that the resulting algorithms can produce results with a comparable accuracy as the known methods in a much shorter amount of time (which is still proportional to N but with a significantly smaller implied constant).

Note that our approach yields a discretization of the fractional differential operator, and so—when used to set up an algorithm for solving fractional order initial value problems like (1)—it works with the differential form of the problem, i.e. with Eq. (1) directly. In this respect, the method proposed here follows the same path as those proposed in, e.g., [2, 3, 5, 18, 26, 28–30]. Other authors [1, 19, 23] have alternatively suggested to apply diffusive representations to the equivalent integral representation of the initial value problem [6, Lemma 6.2], which amounts to a substantially different algorithm.

2 The New Diffusive Representation and Its Properties

Our main idea is based on the following result that summarizes the diffusive representation itself, given in Eq. (9) in conjunction with Eq. (7), and its most important properties (listed in parts (d) and (e) of the theorem) that play the fundamental role for the convergence properties, cf. Theorem 2 and its proof.

We emphasize here that our method can not only be used (as indicated above) in the context of solution algorithms for fractional differential equations but that it also immediately yields a fast method for the approximate numerical calculation of the fractional derivative $D_a^\alpha y(t_j)$ of a given function y at a number of points t_j, $j = 1, 2, \ldots, N$.

Theorem 1 *For given values $a \in \mathbb{R}$, $T > 0$ and $\alpha \in \mathbb{R}_+ \setminus \mathbb{N}$ and a given function $y \in C^{\lceil\alpha\rceil}[a, a+T]$, let*

$$q_{\mathrm{D}} = \alpha - \lceil\alpha\rceil + 1 \tag{6}$$

and

$$\phi_{\mathrm{D}}(w,t) = (-1)^{\lfloor\alpha\rfloor} \frac{\sin\alpha\pi}{\pi} \mathrm{e}^{wq_{\mathrm{D}}} \int_a^t y^{(\lceil\alpha\rceil)}(\tau) \exp\left(-(t-\tau)\mathrm{e}^w\right) \mathrm{d}\tau \tag{7}$$

for all $w \in \mathbb{R}$ and $t \in [a, a+T]$. Then, we have the following properties:

(a) The value q_{D} satisfies $0 < q_{\mathrm{D}} < 1$.
(b) For any $w \in \mathbb{R}$, the function $\phi_{\mathrm{D}}(w, \cdot)$ solves the initial value problem

$$\frac{\partial\phi_{\mathrm{D}}}{\partial t}(w,t) = -\mathrm{e}^w \phi_{\mathrm{D}}(w,t) + (-1)^{\lfloor\alpha\rfloor} \frac{\sin\alpha\pi}{\pi} \mathrm{e}^{wq_{\mathrm{D}}} y^{(\lceil\alpha\rceil)}(t), \quad \phi_{\mathrm{D}}(w,a) = 0 \tag{8}$$

for $t \in [a, a+T]$.
(c) For any $t \in [a, a+T]$,

$$D_a^\alpha y(t) = \int_{-\infty}^{\infty} \phi_{\mathrm{D}}(w,t)\mathrm{d}w. \tag{9}$$

(d) For any $t \in [a, a+T]$, we have $\phi_{\mathrm{D}}(\cdot, t) \in C^\infty(\mathbb{R})$.
(e) For any $t \in [a, a+T]$,

$$\phi_{\mathrm{D}}(w,t) = O(\mathrm{e}^{w(q_{\mathrm{D}}-1)}) \quad \text{as } w \to \infty \tag{10}$$

and

$$\phi_{\mathrm{D}}(w,t) = O(\mathrm{e}^{wq_{\mathrm{D}}}) \quad \text{as } w \to -\infty. \tag{11}$$

So, part (b) of Theorem 1 asserts that our function ϕ_{D} solves an initial value problem of the same type as the previously considered functions, cf. (3a) or (4a). Moreover, according to part (c), by integrating this function with respect to w, we obtain the fractional derivative of the given function y, which is in analogy with the corresponding equation (2) for the known approaches mentioned above. Note that there is a marginal difference between Eqs. (2) and (9) in the sense that the

latter involves an integration over the entire real line whereas the former requires to integrate over the positive half line only, but from the point of view of approximation (or quadrature) theory this does not introduce any substantial problems. (The index D in ϕ_{D} and q_{D} can be interpreted to stand for "doubly infinite integration range".) Thus, in these respects, the new model behaves in very much the same way as the known ones. The significant difference between the known approach and the new one is evident from part (e) of the theorem: it asserts (in view of the property of q_{D} shown in part (a)) that the integrand exhibits the desired exponential decay as $w \to \pm\infty$, thus allowing, in combination with the smoothness result of part (d), a much more efficient numerical integration.

Proof Part (a) is an immediate consequence of the definition of q_{D} given in Eq. (6).

For part (b), we first note that the integrand in Eq. (7) is continuous by assumption. Hence, the integral is zero for $t = a$ which implies that the initial condition given in Eq. (8) is correct. Also, a standard differentiation of the integral in the definition (7) with respect to the parameter t yields the differential equation.

To prove (c), we recall from [6, Proof of Theorem 3.18] that

$$D_a^{\alpha} y(t) = (-1)^{\lfloor \alpha \rfloor} \frac{\sin \alpha\pi}{\pi} \int_a^t \int_0^{\infty} \frac{\mathrm{e}^{-z}}{z} \left(\frac{z}{x-\tau} \right)^{q_{\mathrm{D}}} y^{(\lceil \alpha \rceil)}(\tau) \mathrm{d}z \, \mathrm{d}\tau.$$

The substitution $z = (x - \tau)\mathrm{e}^{w}$, combined with an interchange of the two integrations (which is admissible in view of Fubini's theorem), then leads to the desired result.

Statement (d) directly follows from the definition (7) of the function ϕ_{D}. (Here, one may note an interesting consequence of the representation of Eq. (7): the smoothness properties of the function y whose fractional derivative we aim to compute do not play any role at all in the context of the smoothness of $\phi_{\mathrm{D}}(\cdot, t)$.)

Finally, we show that the estimates of (e) are true. To this end, let us first discuss what happens for $w \to +\infty$. Here, we can see that

$$\phi_{\mathrm{D}}(w, t) = (-1)^{\lfloor \alpha \rfloor} \frac{\sin \alpha\pi}{\pi} (I_1 + I_2),$$

where

$$\begin{aligned} |I_1| &= \left| \mathrm{e}^{w q_{\mathrm{D}}} \int_{t - w \exp(-w)}^{t} y^{(\lceil \alpha \rceil)}(\tau) \exp\left(-(t-\tau)\mathrm{e}^{w}\right) \mathrm{d}\tau \right| \\ &\le \| y^{(\lceil \alpha \rceil)} \|_{L_\infty[a, a+T]} \mathrm{e}^{w q_{\mathrm{D}}} \left| \int_{t - w \exp(-w)}^{t} \exp\left(-(t-\tau)\mathrm{e}^{w}\right) \mathrm{d}\tau \right| \\ &\le \| y^{(\lceil \alpha \rceil)} \|_{L_\infty[a, a+T]} \mathrm{e}^{w q_{\mathrm{D}}} \mathrm{e}^{-w} \left[1 - \mathrm{e}^{-w}\right] < \| y^{(\lceil \alpha \rceil)} \|_{L_\infty[a, a+T]} \mathrm{e}^{w(q_{\mathrm{D}} - 1)} \end{aligned}$$

and

$$
\begin{aligned}
|I_2| &= \left| \mathrm{e}^{wq_{\mathrm{D}}} \int_a^{t-w\exp(-w)} y^{(\lceil\alpha\rceil)}(\tau) \exp\left(-(t-\tau)\mathrm{e}^w\right) \mathrm{d}\tau \right| \\
&\le \mathrm{e}^{wq_{\mathrm{D}}} \max_{\tau\in[a,t-w\exp(-w)]} \exp\left(-(t-\tau)\mathrm{e}^w\right) \int_a^{t-w\exp(-w)} |y^{(\lceil\alpha\rceil)}(\tau)| \mathrm{d}\tau \\
&\le \mathrm{e}^{wq_{\mathrm{D}}} \mathrm{e}^{-w} \int_a^{a+T} |y^{(\lceil\alpha\rceil)}(\tau)| \mathrm{d}\tau = \mathrm{e}^{w(q_{\mathrm{D}}-1)} \int_a^{a+T} |y^{(\lceil\alpha\rceil)}(\tau)| \mathrm{d}\tau,
\end{aligned}
$$

which shows the desired result (10) in this case; in particular, the upper bound decays exponentially for $w \to \infty$ because $q_{\mathrm{D}} < 1$. Regarding the behaviour for $w \to -\infty$, we start from the representation (7) and apply a partial integration. This yields, taking into consideration that $t \ge a$, that

$$
\begin{aligned}
|\phi_{\mathrm{D}}(w,t)| &= \frac{|\sin\alpha\pi|}{\pi} \mathrm{e}^{wq_{\mathrm{D}}} \Big| \exp\left(-(x-\tau)\mathrm{e}^w\right) y^{(\lceil\alpha\rceil-1)}(\tau)\Big|_{\tau=a}^{\tau=t} \\
&\qquad\qquad - \mathrm{e}^w \int_a^t \exp\left(-(t-\tau)\mathrm{e}^w\right) y^{(\lceil\alpha\rceil-1)}(\tau)\mathrm{d}\tau \Big| \\
&\le \frac{|\sin\alpha\pi|}{\pi} \mathrm{e}^{wq_{\mathrm{D}}} \left| y^{(\lceil\alpha\rceil-1)}(t) - y^{(\lceil\alpha\rceil-1)}(a) \exp\left(-(t-a)\mathrm{e}^w\right)\right| \\
&\quad + \frac{|\sin\alpha\pi|}{\pi} \mathrm{e}^{wq_{\mathrm{D}}} \|y^{(\lceil\alpha\rceil-1)}\|_{L_\infty[a,a+T]} \left| \mathrm{e}^w \int_a^t \exp\left(-(t-\tau)\mathrm{e}^w\right) \mathrm{d}\tau \right| \\
&\le \frac{|\sin\alpha\pi|}{\pi} \|y^{(\lceil\alpha\rceil-1)}\|_{L_\infty[a,a+T]} \mathrm{e}^{wq_{\mathrm{D}}} \left(2 + 1 - \exp\left(-(t-a)\mathrm{e}^w\right)\right) \\
&\le 3\frac{|\sin\alpha\pi|}{\pi} \|y^{(\lceil\alpha\rceil-1)}\|_{L_\infty[a,a+T]} \mathrm{e}^{wq_{\mathrm{D}}},
\end{aligned}
$$

thus proving the relation (11) and demonstrating, in view of $q_{\mathrm{D}} > 0$, that $\phi_{\mathrm{D}}(w,t)$ decays to zero exponentially as $w \to -\infty$. □

3 The Complete Numerical Method

Based on Theorem 1—in particular, using the properties shown in parts (d) and (e)—we thus proceed as follows to obtain the required approximation of $D_a^\alpha y(t_j)$, $j = 1, 2, \ldots, N$. Splitting up the integral from Eq. (9) into the integrals over the negative and over the positive half line, respectively, and introducing some obvious

substitutions, we notice that

$$\int_{-\infty}^{\infty} \phi_{\mathrm{D}}(w,t)\mathrm{d}w = \frac{1}{q_{\mathrm{D}}} \int_0^{\infty} \mathrm{e}^{-u}\mathrm{e}^{u}\phi_{\mathrm{D}}(-u/q_{\mathrm{D}},t)\mathrm{d}u + \frac{1}{1-q_{\mathrm{D}}} \int_0^{\infty} \mathrm{e}^{-u}\mathrm{e}^{u}\phi_{\mathrm{D}}(u/(1-q_{\mathrm{D}}),t)\mathrm{d}u.$$

Therefore, using

$$\hat{\phi}_{\mathrm{D}}(u,t) := \mathrm{e}^{u}\left(\frac{1}{q_{\mathrm{D}}}\phi_{\mathrm{D}}(-u/q_{\mathrm{D}},t) + \frac{1}{1-q_{\mathrm{D}}}\phi_{\mathrm{D}}(u/(1-q_{\mathrm{D}}),t)\right), \tag{12}$$

we find that

$$D_a^{\alpha} y(t) = \int_{-\infty}^{\infty} \phi_{\mathrm{D}}(w,t)\mathrm{d}w = \int_0^{\infty} \mathrm{e}^{-u}\hat{\phi}_{\mathrm{D}}(u,t)\mathrm{d}u \approx Q_K^{\mathrm{GLa}}[\hat{\phi}_{\mathrm{D}}(\cdot,t)], \tag{13}$$

where

$$Q_K^{\mathrm{GLa}}[f] = \sum_{k=1}^{K} a_k^{\mathrm{GLa}} f(x_k^{\mathrm{GLa}})$$

is the K-point Gauss–Laguerre quadrature formula, i.e. the Gaussian quadrature formula for the weight function e^{-u} on the interval $[0,\infty)$ [4, Sections 3.6 and 3.7]. For the sake of simplicity, we have chosen to omit from our notation for the nodes x_k^{GLa} and the weights a_k^{GLa} of the Gauss–Laguerre quadrature formula the fact that these quantities depend on the total number K of quadrature nodes. We can then show the following qualitative convergence result.

Theorem 2 *Under the assumptions of Theorem 1, we have*

$$\lim_{K\to\infty} Q_K^{\mathrm{GLa}}[\hat{\phi}_{\mathrm{D}}(\cdot,t)] = D_a^{\alpha} y(t)$$

for all $t \in [a, a+T]$.

This chapter is dedicated to the development and the discussion of the approach from a computational and practical perspective. For this purpose, a qualitative result such as the one shown in Theorem 2 suffices. Therefore, we shall not provide a more detailed convergence analysis; in particular, we shall refrain from investigating the rate of convergence. Such an analysis is possible though and will be given in a forthcoming work [8] that concentrates on the analysis of the method's properties from a more fundamental and theoretical point of view.

Proof We have seen in Theorem 1(d) that the function $\phi_{\mathrm{D}}(\cdot,t)$ is differentiable infinitely many times. Together with the decay properties shown in Theorem 1(e),

this allows us to invoke a classical convergence result for Gauss–Laguerre quadrature formulas [4, p. 227] and derive the desired result. □

For a given number K of quadrature points, it is known that the nodes x_k^{GLa}, $k = 1, 2, \ldots, K$, are the zeros of the Laguerre polynomial L_K of order K, and the associated weights are given by

$$a_k^{\mathrm{GLa}} = \frac{x_k^{\mathrm{GLa}}}{[L_{K+1}(x_k)]^2},$$

cf., e.g., [4, p. 223]. (In our definition of the Laguerre polynomials, the normalization is such that $\int_0^\infty \mathrm{e}^{-x}(L_K(x))^2\mathrm{d}x = 1$.) From [27, eqs. (6.31.7), (6.31.11) and (6.31.12)], we know that, at least for $K \geq 3$,

$$\frac{2.89}{2K+1} < x_1^{\mathrm{GLa}} < \frac{3}{2K} \quad \text{and} \quad 2K < x_K^{\mathrm{GLa}} < 4K + 3.$$

We are now in a position to describe the method for the numerical computation of $D_a^\alpha y(t_j)$, $j = 1, 2, \ldots, N$, that we propose. In this algorithm, the symbol ϕ_k is used to denote the approximate value of $\phi_{\mathrm{D}}(x_k^{\mathrm{GLa}}, t_j)$ for the current time step, i.e. for the currently considered value of j. Steps 1 and 2 here are merely preparatory in nature; the core of the algorithm is step 3.

Given the initial point a, the order α, the grid points t_j, $j = 1, 2, \ldots, N$ and the number $K \in \mathbb{N}$ of quadrature nodes,

1. Set $q_{\mathrm{D}} \leftarrow\!\!\shortmid\ \alpha - \lceil\alpha\rceil + 1$.
2. For $k = 1, 2, \ldots, K$:
   a. Compute the Gauss–Laguerre nodes x_k^{GLa} and the associated weights a_k^{GLa}.
   b. Define the auxiliary quantities $w_k \leftarrow\!\!\shortmid\ -x_k^{\mathrm{GLa}}/q_{\mathrm{D}}$ and $\tilde{w}_k \leftarrow\!\!\shortmid\ x_k^{\mathrm{GLa}}/(1 - q_{\mathrm{D}})$.
   c. Set $\phi_k \leftarrow\!\!\shortmid\ 0$ and $\tilde{\phi}_k \leftarrow\!\!\shortmid\ 0$ (to represent the initial condition of the differential equation (8) for $t = t_0 = a$).
3. For $j = 1, 2, \ldots, N$:
   a. Set $h \leftarrow\!\!\shortmid\ t_j - t_{j-1}$.
   b. For $k = 1, 2, \ldots, K$:
      i. Update the value ϕ_k by means of solving the associated differential equation (8) with, e.g., the backward Euler method, viz.

$$\phi_k \leftarrow\!\!\shortmid\ \frac{1}{1 + h\mathrm{e}^{w_k}} \left(\phi_k + h(-1)^{\lfloor\alpha\rfloor} \frac{\sin\alpha\pi}{\pi} \mathrm{e}^{w_k q_{\mathrm{D}}} y^{(\lceil\alpha\rceil)}(t_j) \right) \tag{14a}$$

      (note that the index k used here is not the time index).
      ii. Similarly, update the value $\tilde{\phi}_k$ by

$$\tilde{\phi}_k \leftarrow\!\!\shortmid\ \frac{1}{1 + h\mathrm{e}^{\tilde{w}_k}} \left(\tilde{\phi}_k + h(-1)^{\lfloor\alpha\rfloor} \frac{\sin\alpha\pi}{\pi} \mathrm{e}^{\tilde{w}_k q_{\mathrm{D}}} y^{(\lceil\alpha\rceil)}(t_j) \right). \tag{14b}$$

c. Compute the desired approximate value for $D_a^\alpha y(t_j)$ using the formula

$$D_a^\alpha y(t_j) = \sum_{k=1}^{K} a_k^{\mathrm{GLa}} \exp(x_k^{\mathrm{GLa}}) \left(\frac{1}{q_{\mathrm{D}}} \phi_k + \frac{1}{1-q_{\mathrm{D}}} \tilde{\phi}_k \right).$$

The main goal of this chapter is to develop a diffusive representation that can be numerically handled in a more efficient way than traditional formulas. Therefore, our work concentrates on the aspects related to the integral, i.e. on the properties of the integrand and on the associated numerical quadrature. The solution of the differential equation is not in the focus of our work; we only use some very simple (but nevertheless reasonable) methods here. Our specific choice is based on the observation that the magnitude of the constant factor with which the unknown function $\phi(w, \cdot)$ on the right-hand side of (8) is multiplied is such that an A-stable method should be used [17]. Therefore, as the simplest possible choice among these methods, we have suggested the backward Euler method in our description given above. Alternatively, one could, e.g., use the trapezoidal method which is also A-stable. This would mean that the formulas given in Eqs. (14a) and (14b) would have to be replaced by

$$\phi_k \leftarrow \frac{1}{1+h\mathrm{e}^{w_k}/2} \left(\left(1 - \frac{h}{2}\mathrm{e}^{w_k}\right) \phi_k + \frac{h}{2}(-1)^{\lfloor \alpha \rfloor} \frac{\sin \alpha\pi}{\pi} \mathrm{e}^{w_k q_{\mathrm{D}}} (y^{(\lceil \alpha \rceil)}(t_j) + y^{(\lceil \alpha \rceil)}(t_{j-1})) \right) \tag{15a}$$

and

$$\tilde{\phi}_k \leftarrow \frac{1}{1+h\mathrm{e}^{\tilde{w}_k}/2} \left(\left(1 - \frac{h}{2}\mathrm{e}^{\tilde{w}_k}\right) \tilde{\phi}_k + \frac{h}{2}(-1)^{\lfloor \alpha \rfloor} \frac{\sin \alpha\pi}{\pi} \mathrm{e}^{\tilde{w}_k q_{\mathrm{D}}} (y^{(\lceil \alpha \rceil)}(t_j) + y^{(\lceil \alpha \rceil)}(t_{j-1})) \right), \tag{15b}$$

respectively. In the following section, we shall report the results of our numerical experiments for both variants.

Remark 1 From a formal point of view, Eqs. (14a) and (14b) have exactly the same structure. From a numerical perspective, however, there is a significant difference between them that needs to be taken into account when implementing the algorithm in finite-precision arithmetic: in view of the definitions of the quantities w_k and $\tilde{w}_k$ given in step 2b of the algorithm and the facts that the Gauss–Laguerre nodes x_k^{GLa} are strictly positive for all k and that $q_{\mathrm{D}} \in (0, 1)$, it is clear that $w_k < 0$ for all k, and hence the powers e^{w_k} and $\mathrm{e}^{w_k q_{\mathrm{D}}}$ that occur in Eq. (14a) are always in the interval $(0, 1)$. It may be, if $|w_k|$ is very large, that the calculation of e^{w_k} in IEEE arithmetic results in an underflow, but this number can then safely be replaced by

0 without causing any problems. Therefore, Eq. (14a) can be implemented directly in its given form. On the other hand, using the same arguments, we can see that $\tilde{w}_k > 0$ for all k, and indeed (at least if k is large and/or q_D is close to 1) $\tilde{w}_k$ may be so large that the computation of $e^{\tilde{w}_k}$ results in a fatal overflow. For this reason, in a practical implementation, Eq. (14b) should not be used in its form given above but in the equivalent form

$$\tilde{\phi}_k \leftarrow \frac{e^{-\tilde{w}_k}}{e^{-\tilde{w}_k}+h}\tilde{\phi}_k + h(-1)^{\lfloor\alpha\rfloor}\frac{\sin\alpha\pi}{\pi}\frac{e^{\tilde{w}_k(q_D-1)}}{e^{-\tilde{w}_k}+h}y^{(\lceil\alpha\rceil)}(t_j), \tag{14c}$$

which avoids all potential overflows.

Evidently, an analog comment applies to Eqs. (15a) and (15b).

4 Experimental Results and Conclusion

In [7], we have reported some numerical results illustrating the convergence behaviour of the RISS method proposed by Hinze et al. [18]. Here, now we present similar numerical results obtained with the new algorithm. A comparison with the corresponding data shown in [7] reveals that, in many cases, our new method requires a smaller number of quadrature nodes than the RISS approach (with otherwise identical parameters) to obtain approximations of a similar quality.

A typical result is shown in Fig. 2 where we have numerically computed the Caputo derivative of order 0.4 of the function $y(t) = t^{1.6}$ over the interval $[0, 3]$. The calculations have been performed on an equispaced grid for the interval $[0, 3]$ with various different step sizes (i.e. with different numbers of grid points) and different choices of the number K of quadrature nodes. Both the backward Euler and the trapezoidal scheme have been tried as the ODE solvers. The figure exhibits the maximal absolute error over all grid points.

The findings of this example can be summarized as follows:

- The trapezoidal method clearly leads to a more accurate approximation than the backward Euler method. Obviously, in view of the trapezoidal method's higher convergence order, this behaviour is exactly what would have been expected.
- The number of quadrature points, i.e. our parameter K, only has a very small influence on the overall error. Therefore, one can afford to work with a relatively small value of K, thus significantly reducing the computational cost, without a substantial loss of accuracy.
- A comparison of the results for the trapezoidal method shows that a certain kind of saturation is reached at an error level of $4.5 \cdot 10^{-6}$ for $K = 40$, i.e. we do not achieve a better accuracy even if we continue to decrease the step size for the ODE solver. This is an indication that this level reflects the contribution of the total error caused by the quadrature formula. If a smaller error is required, one therefore needs to use more quadrature nodes. For example, choosing $K = 70$

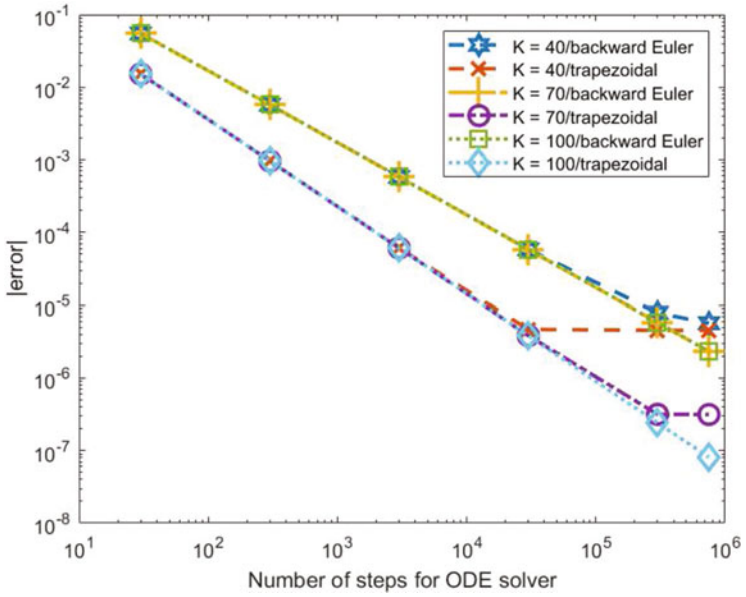


Fig. 2 Maximal errors for the calculation of $D_0^{0.4} y(t)$ with $t \in [0, 3]$ for $y(t) = t^{1.6}$ using different step sizes for the ODE solver and different numbers of quadrature nodes

leads to a saturation level of approximately $3.2 \cdot 10^{-7}$. This indicates that the saturation level might be proportional to $K^{-0.6}$, leading to the conjecture that the exponent of K in this expression could be related to the smoothness properties of the function y (note that the function y' that appears in the formulas which describe our algorithm satisfies a Lipschitz condition of order 0.6).

The fact that this phenomenon is hardly visible if the backward Euler method is used is due to the fact that this ODE solver has a larger error which only just about reaches this range for the chosen step sizes. It would be possible to more clearly observe a similar behaviour if the step sizes were reduced even more.

We have also used a number of other test cases; the behaviour has usually been very similar. Also, the findings of [7] for a significantly different method based on a related fundamental approach point into the same direction. In our future work, we will attempt to provide a thorough analysis of the approximation properties of methods of this type that should confirm the experimental results.

References

1. Baffet, D.: A Gauss-Jacobi kernel compression scheme for fractional differential equations. J. Sci. Comput. **79**, 227–248 (2019). https://doi.org/10.1007/s10915-018-0848-x

2. Birk, C., Song, C.: An improved non-classical method for the solution of fractional differential equations. Comput. Mech. **46**, 721–734 (2010). https://doi.org/10.1007/s00466-010-0510-4
3. Chatterjee, A.: Statistical origins of fractional derivatives in viscoelasticity. J. Sound Vibrations **284** (2005), 1239–1245. https://doi.org/10.1016/j.jsv.2004.09.019
4. Davis, P.J., Rabinowitz, P.: Methods of Numerical Integration, 2nd edn. Academic Press, San Diego (1984)
5. Diethelm, K.: An investigation of some nonclassical methods for the numerical approximation of Caputo-type fractional derivatives. Numer. Algorithms **47**, 361–390 (2008). https://doi.org/10.1007/s11075-008-9193-8
6. Diethelm, K.: The Analysis of Fractional Differential Equations. Springer, Berlin (2010). https://doi.org/10.1007/978-3-642-14574-2
7. Diethelm, K.: Fast solution methods for fractional differential equations in the modeling of viscoelastic materials. In: Proc. 9th International Conference on Systems and Control (ICSC 2021), pp. 455–460. IEEE, Piscataway (2021). https://doi.org/10.1109/ICSC50472.2021.9666636
8. Diethelm, K.: A new diffusive representation for fractional derivatives, part II: convergence analysis of the numerical scheme. Mathematics **10**, 1245 (2022). https://doi.org/10.3390/math10081245
9. Diethelm, K., Ford, N.J., Freed, A.D.: A predictor-corrector approach for the numerical solution of fractional differential equations. Nonlinear Dyn. **29**, 3–22 (2002). https://doi.org/10.1023/A:1016592219341
10. Diethelm, K., Ford, N.J., Freed, A.D.: Detailed error analysis for a fractional adams method. Numer. Algorithms **36**, 31–52 (2004). https://doi.org/10.1023/B:NUMA.0000027736.85078.be
11. Diethelm, K., Freed, A.D.: An efficient algorithm for the evaluation of convolution integrals. Comput. Math. Appl. **51**, 51–72 (2006). https://doi.org/10.1016/j.camwa.2005.07.010
12. Diethelm, K., Kiryakova, V., Luchko, Y., Machado, J.A.T., Tarasov, V.E.: Trends, directions for further research, and some open problems of fractional calculus. Nonlinear Dynam. **107**, 3245–3270 (2022). https://doi.org/10.1007/s11071-021-07158-9
13. Ford, N.J., Simpson, A.C.: The numerical solution of fractional differential equations: speed versus accuracy. Numer. Algorithms **26**, 333–346 (2001). https://doi.org/10.1023/A:1016601312158
14. Garrappa, R.: Numerical solution of fractional differential equations: a survey and a software tutorial. Mathematics **6**, 16 (2018). https://doi.org/10.3390/math6020016
15. Hairer, E., Lubich, C., Schlichte, M.: Fast numerical solution of nonlinear Volterra convolution equations. SIAM J. Sci. Stat. Comput. **6**, 532–541 (1985). https://doi.org/10.1137/0906037
16. Hairer, E., Lubich, C., Schlichte, M.: Fast numerical solution of weakly singular Volterra integral equations. J. Comput. Appl. Math. **23**, 87–98 (1988). https://doi.org/10.1016/0377-0427(88)90332-9
17. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II, 2nd revised edition, corrected second printing. Springer, Berlin (2002). https://doi.org/10.1007/978-3-642-05221-7
18. Hinze, M., Schmidt, A., Leine, R.I.: Numerical solution of fractional-order ordinary differential equations using the reformulated infinite state representation. Fract. Calc. Appl. Anal. **22**, 1321–1350 (2019). https://doi.org/10.1515/fca-2019-0070
19. Li, J.-R.: A fast time stepping method for evaluating fractional integrals. SIAM J. Sci. Comput. **31**, 4696–4714 (2010). https://doi.org/10.1137/080736533
20. Lubich, C.: Fractional linear multistep methods for Abel-Volterra integral equations of the second kind. Math. Comput. **45**, 463–469 (1985). https://doi.org/10.1090/S0025-5718-1985-0804935-7
21. Lubich, C.: Discretized fractional calculus. SIAM J. Math. Anal. **17**, 704–719 (1986). https://doi.org/10.1137/0517050
22. Lubinsky, D.S.: A survey of weighted polynomial approximation with exponential weights. Surv. Approx. Theory **3**, 1–105 (2007)

23. McLean, W.: Exponential sum approximations for $t^{-\beta}$. In: Dick, J., Kuo, F.Y., Woźniakowski, H. (eds.) Contemporary Computational Mathematics, pp. 911–930. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-72456-0_40
24. Montseny, G.: Diffusive representation of pseudo-differential time-operators. ESAIM Proc. **5**, 159–175 (1998)
25. Schmidt, A., Gaul, L.: On a critique of a numerical scheme for the calculation of fractionally damped dynamical systems. Mech. Res. Commun. **33**, 99–107 (2006). https://doi.org/10.1016/j.mechrescom.2005.02.018
26. Singh, S.J., Chatterjee, A.: Galerkin projections and finite elements for fractional order derivatives. Nonlinear Dyn. **45**, 183–206 (2006). https://doi.org/10.1007/s11071-005-9002-z
27. Szegő, G.: Orthogonal Polynomials, 4th edn. Amer. Math. Soc., Providence (1975)
28. Trinks, C., Ruge, P.: Treatment of dynamic systems with fractional derivatives without evaluating memory-integrals. Comput. Mech. **29**, 471–476 (2002). https://doi.org/10.1007/s00466-002-0356-5
29. Yuan, L., Agrawal, O.P.: A numerical scheme for dynamic systems containing fractional derivatives. J. Vibration Acoust. **124**, 321–324 (2002). https://doi.org/10.1115/1.1448322
30. Zhang, W., Capilnasiu, A., Sommer, G., Holzapfel, G.A., Nordsletten, D.: An efficient and accurate method for modeling nonlinear fractional viscoelastic biomaterials. Comput. Methods Appl. Mech. Eng. **362**, 112834 (2020). https://doi.org/10.1016/j.cma.2020.112834

Exact Solutions for the Fractional Nonlinear Boussinesq Equation

Andrea Ceretani, Federico Falcini, and Roberto Garra

Abstract We investigate the existence of exact solutions in closed form to a fractional version of the nonlinear Boussinesq equation for groundwater flow through an unconfined aquifer. We show this fractional equation appears naturally when the classical nonlinear Darcy's law is replaced by a space-fractional one. After a physical discussion on the fractional model, we give several exact solutions in closed form for special choices of initial and boundary data. We provide solutions for steady and unsteady cases, by considering both classical and fractional derivatives in time.

1 Introduction

This chapter is devoted to show that, by simple methods, it is possible to find exact solutions in closed form to a nonlocal generalization of the well-known Boussinesq equation for a horizontal unconfined aquifer,

$$\frac{\partial h}{\partial t}(x,t) = k\frac{\partial}{\partial x}\left(h(x,t)\frac{\partial h}{\partial x}(x,t)\right) + \frac{\Lambda(h(x,t),t)}{n} \qquad x > 0,\ t > 0, \tag{1}$$

A. Ceretani
Department of Mathematics of the Faculty of Exact and Natural Sciences, University of Buenos Aires, and Mathematics Research Institute "Luis A. Santaló" (IMAS), CONICET, Buenos Aires, Argentina
e-mail: aceretani@dm.uba.ar

F. Falcini · R. Garra (✉)
Institute of Marine Sciences, National Research Council (CNR), Rome, Italy
e-mail: federico.falcini@cnr.it; roberto.garra@uniroma1.it

A. Cardone et al. (eds.), *Fractional Differential Equations*,
Springer INdAM Series 50, https://doi.org/10.1007/978-981-19-7716-9_2

where k and n are given positive parameters and Λ is a known function. Here, we consider the fully nonlinear fractional equation

$$\frac{\partial^\gamma h}{\partial t^\gamma}(x,t) = k\frac{\partial}{\partial x}\left(h(x,t)\frac{\partial^\nu h}{\partial x^\nu}(x,t)\right) + \frac{\Lambda(h(x,t),t)}{n} \qquad x > 0,\ t > 0, \tag{2}$$

with a sink term Λ of the form

$$\Lambda(h,t) = -n\phi h, \tag{3}$$

where ϕ is a non-negative parameter. The operator $\partial^\nu/\partial x^\nu$ is the partial fractional derivative with respect to the space variable x, of order $\nu \in (0,1)$, associated with the left-sided Caputo fractional derivative with zero starting point. The operator $\partial^\gamma/\partial t^\gamma$ is defined analogously if $\gamma \in (0,1)$, whereas it is given by the usual derivative if $\gamma = 1$. We recall that the Caputo fractional derivative of order $\alpha \in (0,1)$ is defined by

$$\frac{d^\alpha f}{dz^\alpha}(z) = \frac{1}{\Gamma(1-\alpha)}\int_0^z (z-z')^{-\alpha}\frac{df}{dz'}(z')\,dz' \qquad z \geq 0, \tag{4}$$

for absolutely continuous functions f in $[0,\infty)$, where $\Gamma(\cdot)$ is the gamma function. Notice that we have a space-time-fractional problem if $\gamma \in (0,1)$ and a space-fractional one if $\gamma = 1$. In addition, observe that the local version ($\gamma = \nu = 1$) of Eq. (2) is the usual Boussinesq equation for a horizontal unconfined aquifer.

Our work is motivated by modeling groundwater flows through anisotropic or non-homogeneous soils in hillslope areas. It is widely accepted that the Boussinesq equation is a suitable model to describe groundwater flows through isotropic and homogeneous soils; see, for example, [3, 4, 16] and the references therein. However, in the absence of isotropy or homogeneity and, in particular, under long-range effects in groundwater flows, the Boussinesq equation can no longer capture the nonlocal effects of pressure gradients; see, for example, [18–20]. In the last decades, fractional calculus tools have shown remarkable potential to accurate modeling phenomena in anisotropic, non-homogeneous media, see, for example, [5, 23]. We show in Sect. 2 that the space-fractional equation (2) arises naturally when the local, nonlinear Darcy's law is replaced by a fractional one in the usual derivation of the Boussinesq equation.

As a first step, we are interested in deriving exact solutions in closed form to our model. This is a hard task because we have to deal with a nonlinear fractional equation. Exact solutions to the Boussinesq equation were given in several works for special choices of initial and boundary data; see for example [4, 12, 18]. Following this approach, we consider here a *toy model* for which solutions can be obtained in closed form. More precisely, we consider the initial and boundary conditions

$$h(x,0) = x^\sigma \qquad x \geq 0, \qquad\qquad h(0,t) = 0 \quad t \geq 0, \tag{5}$$

where $\sigma > 0$.

Frequently, dealing with more realistic fractional models requires to develop methods to find approximate solutions. We refer to the recent monograph [10] and the references therein for a comprehensive survey on this topic. The exact solutions provided here may serve to test analytical or numerical methods to approximate solutions to fully nonlinear fractional equations. The complexity of fractional models has further motivated the study on generalized solutions to them. We refer, for example, to the recent article [22] where the authors investigate the existence and properties of viscosity solutions for a fractional problem obtained from a fractional linear Darcy-type law. To the best of our knowledge, the problem considered here (i.e. the one derived from a fractional *nonlinear* Darcy-type law) was not yet investigated in the literature.

The outline of the article is as follows. In Sect. 2, we provide a simple derivation of Eq. (2), based on the continuity equation and a fractional Darcy-type law. Exact solutions for the steady and unsteady space-fractional problems are given in Sects. 3 and 4, respectively. The time-fractional case is analyzed in Sect. 5, where exact solutions are provided also for problems with time-dependent coefficients. The chapter closes with a brief discussion in Sect. 6.

2 Physical Motivation

The goal of this section is to show that Eq. (2) is a special case of a fractional version of the classical nonlinear Boussinesq equation, which plays a central role to understand groundwater flows through unconfined aquifers.

For isotropic and homogeneous soils, the depth of water h is governed by the continuity equation

$$n\frac{\partial h}{\partial t}(x,t) = -\frac{\partial q}{\partial x}(x,t) + \Lambda(h(x,t),t), \tag{6}$$

where the flow rate q is assumed to be given by Darcy's law

$$q(x,t) = -h(x,t)K_s\sin(\theta) - h(x,t)\frac{\partial h}{\partial x}(x,t)K_s\cos(\theta), \tag{7}$$

see, for example, [1]. Here, x is the coordinate parallel to the bed slope and θ is the angle of the bed slope to the horizontal. The parameter n is the soil porosity, K_s is the saturated hydraulic conductivity, and $\Lambda(h,t)$ is a source or a sink term.

Replacing Eq. (7) into Eq. (6), we directly obtain the nonlinear Boussinesq equation

$$\frac{\partial h}{\partial t}(x,t) = k\frac{\partial}{\partial x}\left(h(x,t)\frac{\partial h}{\partial x}(x,t)\right) + v_s\frac{\partial h}{\partial x}(x,t) + \frac{\Lambda(h(x,t),t)}{n}, \tag{8}$$

where we have written $v_s = (K_s/n)\sin(\theta)$ and $k = (K_s/n)\cos(\theta)$ for shortness.

In the absence of homogeneity or isotropy in the porous medium, a model based on the local Darcy's law (7) may no longer capture the effects of nonlocal pressure differences. The role of them on Darcy's law was firstly investigated by Sen and Ramos in [18], for a steady problem. There, the porous medium is conceived as a network of short-, medium-, and long-range interstitial channels with impermeable walls, where the flow is driven by both local and nonlocal pressure gradients. The flow rate q is then given by

$$q(x) = -\int_{-\infty}^{+\infty} f(x, x')(p(x') - p(x))dx', \tag{9}$$

where p is the pressure field and f is a nonlocal flow conductivity. The later takes into account the effects of the channel length and thickness between the locations x' and x, and it is directly related to the medium properties. Both the classical Darcy's law (7) and some nonlocal versions of it can be obtained as special cases of (9); see [18] for the details. In particular, by considering a power-law nonlocal conductivity f, one finds a fractional Darcy-type law given in terms of the Riemann–Liouville fractional derivative operator [18]. Encouraged by this pioneering work, models based on fractional Darcy-type laws were investigated in several papers, see, for example, [7] and the references therein. In the same vein of them, we consider here that the flow rate q is given by the nonlinear fractional Darcy-type law

$$q(x,t) = -h(x,t)K_s \sin(\theta) - h(x,t)\frac{\partial^\nu h}{\partial x^\nu}(x,t)K_s \cos(\theta), \tag{10}$$

where $\nu \in (0, 1)$. Replacing (10) into the continuity equation (6), we obtain the nonlinear space-fractional Boussinesq equation

$$\frac{\partial h}{\partial t}(x,t) = k\frac{\partial}{\partial x}\left(h\frac{\partial^\nu h}{\partial x^\nu}(x,t)\right) + v_s\frac{\partial h}{\partial x}(x,t) + \Lambda(h(x,t),t). \tag{11}$$

To keep the problem as simple as possible while retaining the nonlinearity, we consider a horizontal aquifer (then $\theta = 0$ and so v_s) and a sink term of the form $\Lambda(h,t) = -n\phi h$, where ϕ is a transfer coefficient that controls the magnitude of the seepage loss of water into the bedrock; see [1]. Thus, Eq. (11) reduces to (2) with $\gamma = 1$.

3 The Steady Solution

We first consider the steady case, which is given by the boundary value problem

$$k\frac{d}{dx}\left(h(x)\frac{d^\nu h}{dx^\nu}(x)\right) = \phi h(x) \qquad x > 0, \qquad h(0) = 0, \tag{12}$$

for a nonlinear fractional ordinary differential equation. This problem admits a solution in the power-law form $h(x) = cx^{\beta}$ provided that c and β are properly chosen, as we show below. Notice that any function h in this form satisfies the boundary condition $h(0) = 0$.

Exploiting that

$$\frac{d^{\nu}}{dx^{\nu}}(x^{\beta}) = \frac{\Gamma(\beta+1)}{\Gamma(\beta+1-\nu)}x^{\beta-\nu}, \tag{13}$$

we obtain that $h(x) = cx^{\beta}$ satisfies the equation if and only if

$$kc(2\beta-\nu)\frac{\Gamma(\beta+1)}{\Gamma(\beta+1-\nu)}x^{2\beta-\nu-1} = \phi x^{\beta}. \tag{14}$$

If $\phi = 0$, Eq. (14) is fulfilled only if $c = 0$. If $\phi > 0$, an elementary computation shows that (14) holds true only if $\beta = \nu + 1$ and $c = \phi/k\Gamma(\nu+3)$. Thus, we find the exact solution

$$h(x) = \frac{\phi x^{\nu+1}}{k\Gamma(\nu+3)} \qquad x \geq 0. \tag{15}$$

Remark 1 Following the same argument, we find the exact solution

$$h_{\ell}(x) = \frac{\phi x^2}{6k} \qquad x \geq 0, \tag{16}$$

for the associated nonlinear local problem

$$k\frac{d}{dx}\left(h(x)\frac{dh}{dx}(x)\right) = \phi h(x) \qquad x > 0, \qquad h(0) = 0. \tag{17}$$

4 The Unsteady Space-Fractional Case

We now address the unsteady space-fractional problem given by

$$\frac{\partial h}{\partial t}(x,t) = k\frac{\partial}{\partial x}\left(h(x,t)\frac{\partial^{\nu} h}{\partial x^{\nu}}(x,t)\right) - \phi h(x,t) \qquad x > 0,\ t > 0, \tag{18}$$

subject to the initial and boundary conditions (5).

Motivated by the steady solution found in Sect. 3, we assume that the exponent σ in the initial data $h(x,0) = x^{\sigma}$ is given by $\sigma = \nu + 1$. We show below that this

problem admits a similarity-type solution in the form

$$h(x,t) = x^{\nu+1} f(t), \tag{19}$$

for a suitable selection of the function f.

Exploiting formula (13) again, we observe that $h(x,t) = x^{\nu+1} f(t)$ solves Eq. (18) if and only if f solves the following boundary value problem for a nonlinear ordinary differential equation:

$$\frac{df}{dt}(t) = k\,\Gamma(\nu+3) f(t)^2 - \phi f(t) \qquad t > 0, \qquad f(0) = 1. \tag{20}$$

If $\phi = 0$, a simple computation shows that

$$f(t) = \frac{1}{1 - k\Gamma(\nu+3)t}$$

solves (20) for $t \geq 0$ with $t \neq 1/k\Gamma(\nu+3)$. If $\phi > 0$, we find that

$$f(t) = \frac{\phi}{e^{\phi t}(\phi - k\Gamma(\nu+3)) + k\Gamma(\nu+3)}$$

solves (20) for $t \geq 0$ if $\phi \geq k\Gamma(\nu+3)$ or for $t \geq 0$ with $t \neq \frac{1}{\phi} \ln\left(\frac{k\Gamma(\nu+3)}{k\Gamma(\nu+3)-\phi}\right)$ if $0 < \phi < k\Gamma(\nu+3)$. Physically realistic solutions for $\phi = 0$ must be non-negative. This is achieved only if $0 \leq t \leq \frac{1}{k\Gamma(\nu+3)}$. If $\phi > 0$, physically realistic solutions must be, in addition, non-increasing with respect to time at every $x > 0$. This occurs only if $\phi \geq k\Gamma(\nu+3)$. Thus, we obtain the exact solution

$$h(x,t) = \begin{cases} \dfrac{x^{\nu+1}}{1 - k\,\Gamma(\nu+3)t} & x \geq 0,\ 0 \leq t < \frac{1}{k\Gamma(\nu+3)},\ \text{if} \quad \phi = 0, \\ \dfrac{\phi\, x^{\nu+1}}{e^{\phi t}(\phi - k\Gamma(\nu+3)) + k\Gamma(\nu+3)} & x \geq 0,\ t \geq 0,\ \text{if} \quad \phi \geq k\Gamma(\nu+3). \end{cases}$$

Remark 2 As for the steady case, similar steps allow us to obtain the exact solution

$$h_\ell(x,t) = \begin{cases} \dfrac{x^2}{1 - 6kt} & x \geq 0,\ 0 \leq t < \frac{1}{6k} \quad \text{if } \phi = 0, \\ \dfrac{\phi\, x^2}{e^{\phi t}(\phi - 6k) + 6k} & x \geq 0,\ t \geq 0 \qquad \text{if } \phi \geq 6k, \end{cases}$$

to the associated local problem with initial data $h(x,0) = x^2$.

Minor changes on the above arguments allow to find exact solutions in the form

$$h(x,t) = x^{\nu+1} f_1(t) + f_2(t), \tag{21}$$

where f_1 and f_2 satisfy the system of nonlinear ordinary differential equations

$$\frac{df_1}{dt}(t) = k\Gamma(\nu+3)f_1(t)^2 - \phi f_1(t),$$
$$\frac{dg}{dt}(t) = f_1(t)f_2(t)\Gamma(\nu+2) - \phi f_2(t),$$

with the initial conditions $f_1(0) = 1$ and $f_2(0) = 0$.

The strategy developed so far to obtain exact solutions in the form (19) or (21) can be framed into the invariant subspace method (see, for example, [6] for the general theory and [8] for applications to nonlinear fractional differential equations). We briefly recall the main idea of this method: consider the equation

$$\frac{\partial h}{\partial t}(x,t) = F[h](x,t), \tag{22}$$

where $F[h]$ is a nonlinear operator. A space of functions W is said to be *invariant* under the operator F if $F[h] \in W$ for every $h \in W$. It is straightforward that if W admits a finite basis $\{w_1, \ldots, w_n\}$, then W is invariant under F if and only if there exist n scalar functions $\Phi_1, \ldots, \Phi_n$ defined on $\mathbb{R}^n$ that satisfy

$$F[\sum_{k=1}^{n} f_k w_k] = \sum_{k=1}^{n} \Phi_k(f_1, \ldots, f_n)w_k,$$

for every $(f_1, f_2, \ldots, f_n) \in \mathbb{R}^n$. Thus, if W is as before, then Eq. (22) admits a solution in the form

$$h(x,t) = \sum_{k=1}^{n} f_k(t)w_k(x),$$

if and only if $f_1, \ldots, f_n$ satisfy

$$\frac{df_1}{dt}(t) = \Phi_1(f_1(t), \ldots, f_n(t)),$$
$$\frac{df_2}{dt}(t) = \Phi_2(f_1(t), \ldots, f_n(t)),$$
$$\vdots$$
$$\frac{df_n}{dt}(t) = \Phi_n(f_1(t), \ldots, f_n(t)).$$

In this way, we reduce the original problem (22) to a system of ordinary differential equations that can be solved exactly in many cases. The useful role of this method

to find exact solutions for nonlinear fractional partial differential equations has been shown in recent publications, and we refer, for example, to [15, 17].

Let us now go back to our problem. The operator F associated with Eq. (18) is given by

$$F[h] = k\frac{\partial}{\partial x}\left(h\frac{\partial^{\nu} h}{\partial x^{\nu}}\right) - \phi h. \tag{23}$$

Let w_1 and w_2 be the functions given by

$$w_1(x) = x^{\nu+1} \qquad \text{and} \qquad w_2(x) = 1. \tag{24}$$

A straightforward computation shows that the spaces W_1 and W_2 spanned by $\{w_1\}$ and $\{w_1, w_2\}$, respectively, are invariant under the operator F. The solutions in the form (19) are those related to W_1, and the solutions in the form (21) are those related to W_2.

Exploiting the invariant subspace method, one can look for exact solutions to Eq. (18) by first finding a finite-dimensional invariant subspace for the operator F given in (23). To illustrate, it is also possible to find exact solutions from the invariant subspaces $\tilde{W}_1$ and $\tilde{W}_2$ spanned by $\{\tilde{w}_1\}$ and $\{\tilde{w}_1, w_2\}$, respectively, where

$$\tilde{w}_1(x) = x^{\nu/2}, \tag{25}$$

provided that the exponent in the initial data $h(x, 0) = x^{\sigma}$ in (5) is $\sigma = \nu/2$. For example, the solution related to $\tilde{W}_1$ is given by

$$h(x, t) = x^{\nu/2} e^{-\phi t} \qquad x \geq 0,\ t \geq 0. \tag{26}$$

We omit the details to avoid repetition.

5 The Time-Fractional Case

We finally consider the space–time-fractional problem given by Eq. (2) with $\gamma \in (0, 1)$ and the sink term given by (3), with the initial and boundary conditions (5).

The invariant subspace method described at the end of Sect. 4 extends naturally to equations in the form

$$\frac{\partial^{\gamma} h}{\partial t^{\gamma}} = F[h]. \tag{27}$$

Let W_1, W_2, $\tilde{W}_1$, $\tilde{W}_2$ be the subspaces spanned by $\{w_1\}$, $\{w_1, w_2\}$, $\{\tilde{w}_1\}$, $\{\tilde{w}_1, w_2\}$, where w_1, w_2, and $\tilde{w}_1$ are given in (24) and (25) (see Sect. 4). We already know that they are invariant spaces under the operator F given by (23).

This motivates to look for solutions that belong to some of them. For example, by considering the invariant subspace W_1, we find the exact solution

$$h(x,t) = x^{\nu/2} E_\gamma(-\phi\, t^\gamma) \qquad x \geq 0,\ t \geq 0, \tag{28}$$

for the case when the exponent in the initial data is $\sigma = \nu/2$. Here,

$$E_\gamma(-\phi\, t^\gamma) = \sum_{i=0}^{\infty} \frac{(-\phi\, t^\gamma)^i}{\Gamma(\gamma i + 1)} \tag{29}$$

is the one-parameter Mittag–Leffler function (see [9]).

Remark 3 The same arguments enable to find exact solutions when the transfer coefficient ϕ depends on time according to $\phi(t) = \lambda\, t^\beta$, where $\lambda > 0$ and $\beta \in \mathbb{R}$. Here we recall that the so-called Kilbas–Saigo function (see [9] and [2] for details about this special function)

$$E_{a,1+\frac{b}{a},\frac{b}{a}}\left(-\lambda t^{a+b}\right) = 1 + \sum_{i=1}^{\infty} (-\lambda)^i t^{i(a+b)} \prod_{j=0}^{i-1} \frac{\Gamma\left(a\left(j + j\frac{b}{a} + \frac{b}{a}\right) + 1\right)}{\Gamma\left(a\left(j + j\frac{b}{a} + \frac{b}{a} + 1\right) + 1\right)} \tag{30}$$

solves the fractional Cauchy problem:

$$\begin{cases} \frac{d^a y}{dt}(t) = -\lambda t^b y(t), \quad t \geq 0,\ a \in (0,1],\ -a < b \leq 1 - a, \\ y(0) = 1. \end{cases} \tag{31}$$

Therefore, in this case, we obtain the solution

$$h(x,t) = x^{\nu/2} E_{\gamma,1+\frac{\beta}{\gamma},\frac{\beta}{\gamma}}\left(-\lambda\, t^{\beta+\gamma}\right) \qquad x \geq 0,\ t \geq 0, \tag{32}$$

under the assumptions $\beta \in (-\gamma, 1-\gamma]$ and $\sigma = \nu/2$.

6 Conclusions

We have explored the existence of exact solutions in closed form to a fully nonlinear fractional equation involving Caputo partial derivatives. This equation is closely related to the Boussinesq equation for unconfined horizontal aquifers, and it was obtained by a fractional Darcy-type law for the flow rate. Supplemented with appropriated initial and boundary conditions, the fractional equation considered here may serve as a model for groundwater flows through anisotropic or non-homogeneous porous medium. As a first step to validate the later, we addressed

here a toy model that allowed us to find exact solutions in closed form. This model was obtained by a specific selection of initial and boundary conditions. We provided exact solutions to steady and unsteady space-fractional problems, with and without memory effects. These solutions may be used to test analytical or numerical methods to approximate solutions to more realistic, steady, and unsteady models, based on the fractional equation proposed here.

Nonlinearity, nonlocality, and memory effects may be directly related to unknown, long-range factors that vary across landscape types. This represents a core general hydrologic descriptions at both hillslope and watershed scales, since nonlinearity, nonlocality, and memory may affect the groundwater flow at different time and spatial scales. As for [1], our general solutions also have an explicit dependence on the along-channel variable x, a feature that allows for analytic investigations of surface topography and water table profile. Finally, our set of solutions reinforce previous finding on fractional studies on the Boussinesq equation for groundwater flow [19, 20], presenting new solutions related to different initial and boundary conditions and thus providing new insights into how water flows in fractal media, also for transient problems.

Acknowledgments This work was partially supported by EMODnet (European Marine Observation and Data Network Physics) and by CNR, in the frame of the Italian Flag Project Ritmare and the Italian Marine Strategy Framework Directive Programs.

References

1. Bartlett, M.S., Porporato, A.: A class of exact solutions of the Boussinesq equation for horizontal and sloping aquifers. Water Resour. Res. **54**(2), 767–778 (2018)
2. Boudabsa, L., Simon, T.: Some properties of the Kilbas-Saigo function. Mathematics **9**(3), 217 (2021)
3. Brutsaert, W., Nieber, J.L.: Regionalized drought flow hydrographs from a mature glaciated plateau. Water Resour. Res. **13**(3), 637–643 (1977)
4. Daly, E., Porporato, A.: A note on groundwater flow along a hillslope. Water Resour. Res. **40**, W01601 (2004)
5. Falcini, F., Garra, R., Voller, V.: Modeling anomalous heat diffusion: comparing fractional derivative and non-linear diffusivity treatments. Int. J. Thermal Sci. **137**, 584–588 (2019)
6. Galaktionov, V., Svirshchevskii, S.: Exact Solutions and Invariant Subspaces of Nonlinear Partial Differential Equations in Mechanics and Physics. Applied Mathematics and Nonlinear Science Series. Chapman and Hall/CRC (2007)
7. Garra, R., Salusti, E.: Application of the nonlocal Darcy law to the propagation of nonlinear thermoelastic waves in fluid saturated porous media. Phys. D: Nonlinear Phenom. **250**, 52–57 (2013)
8. Gazizov, R., Kasatkin, A.: Construction of exact solutions for fractional order differential equations by the invariant subspace method. Comput. Math. Appl. **66**(5), 576–584 (2013)
9. Gorenflo, R., Kilbas, A., Mainardi, F., Rogosin, S.: Mittag–Leffler Functions, Related Topics and Applications, 2d edn. Springer, Berlin (2014)
10. Karniadakis, G. (ed.): Handbook of Fractional Calculus with Applications: Numerical Methods, volume 3 of De Gruyter Reference. De Gruyter, Berlin (2019)

11. Kilbas, A., Srivastava, H., Trujillo, J.: Theory and Applications of Fractional Differential Equations, vol. 204. Elsevier, Amsterdam (2006)
12. Parlange, J.-Y., Hogarth, W., Govindaraju, R., Parlange, M., Lockington, D.: On an exact analytical solution of the Boussinesq equation. Transp. Porous Media **39**(3), 339–345 (2000)
13. Peletier, L.: Asymptotic behavior of solutions of the porous media equation. SIAM J. Appl. Math. **21**(4), 542–551 (1971)
14. Polyanin, A., Zaitsev, V.: Handbook of Nonlinear Partial Differential Equations. CRC Press, Boca Raton (2004)
15. Prakash, P.: New exact solutions of generalized convection-reaction-diffusion equation. Eur. Phys. J. Plus **134**(6), 1–11 (2019)
16. Rupp, D.E., Selker, J.S.: On the use of the Boussinesq equation for interpreting recession hydrographs from sloping aquifers. Water Resour. Res. **42**, W12421 (2006)
17. Sahadevan, R., Bakkyaraj, T.: Invariant subspace method and exact solutions of certain nonlinear time fractional partial differential equations. Fract. Calc. Appl. Anal. **18**(1), 146–162 (2015)
18. Sen, M., Ramos, E.: A spatially non-local model for flow in porous media. Transp. Porous Media **92**(1), 29–39 (2012)
19. Su, N.: The fractional Boussinesq equation of groundwater flow and its applications. J. Hydrol. **547**, 403–412 (2017)
20. Su, N.: Fractional Calculus for Hydrology, Soil Science and Geomechanics: An Introduction to Applications. CRC Press, Boca Raton (2020)
21. Vázquez, J.L.: The Porous Medium Equation: Mathematical Theory. Oxford University Press, Oxford (2007)
22. Tokinaga, N., Rybka, P.: On viscosity solutions of space-fractional diffusion equations of Caputo type. SIAM J. Math. Anal. **52**(1), 653–681 (2020)
23. Voller, V.R.: On a fractional derivative form of the Green–Ampt infiltration model. Adv. Water Resour. **34**(2), 257–262 (2011)
24. Zhuang, P., Liu, F., Turner, I., Gu, Y.: Finite volume and finite element methods for solving a one-dimensional space-fractional Boussinesq equation. Appl. Math. Model. **38**(15–16), 3860–3870 (2014)

A Numerical Procedure for Fractional-Time-Space Differential Equations with the Spectral Fractional Laplacian

Fabio Vito Difonzo and Roberto Garrappa

Abstract The aim of this chapter is to device a computationally effective procedure for numerically solving fractional-time-space differential equations with the spectral fractional Laplacian. A truncated spectral representation of the solution in terms of the eigenfunctions of the usual integer-order Laplacian is considered. Time-dependent coefficients in this representation, which are solutions to some linear fractional differential equations, are evaluated by means of a generalized exponential time-differencing method, with some advantages in terms of accuracy and computational effectiveness. Rigorous a priori error estimates are derived, and they are verified by means of some numerical experiments.

1 Introduction

In the last decades, fractional calculus, namely the study of integrals, derivatives, and differential equations of non-integer order, has been attracting an increasing attention due to the large extent of fields (biology, engineer, finance, physics, and so on) in which its application has proved to be useful to improve the description of complex systems or anomalous phenomena.

One of the main reasons for introducing fractional calculus in modeling is indeed related to the need of using adequate operators to describe the non-locality observed in anomalous phenomena and complex systems. Very often one observes a dependence of the system state from its whole past history (non-locality in time), as well as its dependence at one point on the whole domain (non-locality in space). It is worth mentioning that fractional calculus is just one of the approaches available to incorporate non-locality or memory effects in modeling differential equations (we could cite at this purpose peridynamics [35] and the several works for dealing

F. V. Difonzo (✉) · R. Garrappa
Dipartimento di Matematica, Università degli Studi di Bari, Bari, Italy

Member of the INdAM Research Group GNCS, Roma, Italy
e-mail: fabio.difonzo@uniba.it; roberto.garrappa@uniba.it

A. Cardone et al. (eds.), *Fractional Differential Equations*,
Springer INdAM Series 50, https://doi.org/10.1007/978-981-19-7716-9_3

with peridynamics from a numerical perspective, e.g., [28, 29, 39, 40]). Anyway, since fractional calculus appears as one of the most powerful and used tools in mathematics, developing new and efficient methods for the numerical solution of fractional differential equations (FDEs) is a research area still deserving a particular attention.

We are interested in solving, on a bounded Lipschitzian open domain $\Omega \subset \mathbb{R}^d$, a fractional-time-space differential equation (FTSDE) in the form

$$\partial_t^\alpha u(\mathbf{x}, t) + \nu\big(-\Delta\big)^s u(\mathbf{x}, t) = f(\mathbf{x}, t), \quad \mathbf{x} \in \Omega, \quad t \in (0, T], \tag{1}$$

where $\nu > 0$ is a constant diffusion coefficient and $0 < \alpha < 1$, $0 < s < 1$ are, respectively, the non-integer (i.e., fractional) differentiation orders of the time and the spatial derivative. Here, for an absolutely continuous function g on $[0, T]$, the operator ∂_t^α represents the Caputo fractional derivative of order $0 < \alpha < 1$ [13] with respect to time and initialized at $t = 0$

$$\partial_t^\alpha g(t) = {}_0^{\mathrm{C}}D_t^\alpha g(t) := \frac{1}{\Gamma(1-\alpha)} \int_0^t (t-\tau)^{-\alpha} g'(\tau) \mathrm{d}\tau, \quad t \in (0, T],$$

with $\Gamma(\cdot)$ the Euler-gamma function (the order of the time-fractional derivative is restricted to $0 < \alpha < 1$ just for ease of presentation, but the extension to super-diffusive problems in which $1 < \alpha < 2$ is always possible). $\big(-\Delta\big)^s$ is the spectral fractional Laplacian, whose definition will be discussed later on. In view of the use of the Caputo time-fractional derivative, the FTSDE (1) is coupled with an initial condition

$$u(\mathbf{x}, 0) = u_0(\mathbf{x}), \quad \mathbf{x} \in \Omega.$$

Finally, homogeneous boundary conditions of Dirichlet type

$$u(\mathbf{x}, t) = 0, \quad (\mathbf{x}, t) \in \partial\Omega \times (0, T],$$

or of Neumann type

$$\frac{\partial}{\partial n} u(\mathbf{x}, t) = 0, \quad (\mathbf{x}, t) \in \partial\Omega \times (0, T],$$

will be imposed to guarantee uniqueness of the solution to the FTSDE (1).

Data $f(\mathbf{x}, t)$ and $u_0(\mathbf{x})$ are assumed to satisfy some not restrictive regularity assumptions, such as $f(\mathbf{x}, t) \in L^2(\Omega \times [0, T])$ and $u_0(\mathbf{x}) \in L^2(\Omega)$.

As it is well known, regularity of data in fractional-order problems does not ensure regularity of the solution [38], and indeed, solutions of fractional-order problems lack smoothness, especially with respect to time. For this reason, classical methods based on polynomial approximations, such as product-integration (PI)

rules, usually turn out to be not really competitive due to their low convergence order, as well as for stability and/or computational issues.

In [19, 20], it has been proposed an alternative approach in order to generalize exponential integrators to FDEs. In particular, for linear systems of FDEs with a forcing term, it is possible to exploit an integral formulation in terms of the Mittag–Leffler function and devise PI rules in which one approximates by polynomials the forcing term only (which reasonably possesses some regularity) and not the whole vector field of the equation that, instead, usually lacks regularity at $t = 0$. Rules of this kind are recognized as *generalized exponential time-differencing* (GETD) methods and present some advantages in terms of convergence order, stability, and computational efficiency [16].

In this chapter, we explore the application of GETD methods to the FTSDE (1). The approach we propose exploits some features of the spectral fractional Laplacian that allow to decompose the original FTSDE into a sequence of FDEs. After truncation of this sequence, each FDE is hence solved by a trapezoidal GETD method, whose main characteristics are the preservation of the full order 2 of convergence—typical of trapezoidal rule—and the possibility of saving computation by directly evaluating the solution at any time t without a computationally expensive step-by-step procedure.

The main contribution proposed in this chapter is hence the derivation of some accurate error estimates in order to provide a tool for tuning the method and identify the number of terms in the sequence of FDEs necessary to achieve a given accuracy. At the same time, the error estimates are verified by means of some numerical examples to show the reliability of the obtained results.

This chapter is organized in the following way. In Sect. 2, we introduce the spectral fractional Laplacian adopted to define $(-\Delta)^s$, and we review some of its main properties. In Sect. 3, we present the decomposition of the FTSDE (1) in terms of a sequence of FDEs, and in Sect. 4, we provide a description of GETD methods for solving FDEs. Section 5 is devoted to the error analysis and to find some bounds for the two components of the error (truncation and discretization errors) that are hence verified by means of some numerical experiments in Sect. 6. Some concluding remarks are finally discussed in Sect. 7.

2 The Spectral Fractional Laplacian: A Brief Introduction

In the recent past, several approaches to define fractional counterparts of the classical integer-order Laplacian have been investigated (e.g., see [4, 12, 26, 27, 37]). The various approaches are usually not equivalent when posed on bounded domains, and different definitions involve different theoretical properties and different numerical difficulties.

It is worth mentioning that fractional Laplacian is a very active research field (see, for instance, [7, 8, 10, 11, 33, 36]), and nowadays, the interest is moving toward possible definitions of the variable-order fractional Laplacian as well [9] (a topic that is however not discussed here).

In this chapter, we focus on the spectral definition of the fractional Laplacian that extends, in a quite straightforward way, spectral features of the classical integer-order Laplacian operator. To provide a comprehensive introduction to the spectral fractional Laplacian, we first review some background material.

In the following, for real functions $f, g \in L^2(\Omega)$, $\langle f, g\rangle$ will denote the usual inner product $\langle f, g\rangle = \int_\Omega f(\mathbf{x})g(\mathbf{x})\mathrm{d}\mathbf{x}$.

2.1 Eigendecomposition of the Laplacian

Let us assume that a set of orthonormal eigenfunctions $\{\varphi_k\}_{k\in\mathbb{N}}$, with corresponding eigenvalues $\{\lambda_k\}_{k\in\mathbb{N}}$, of the negative Laplacian $-\Delta$ on Ω coupled with homogeneous Dirichlet or Neumann boundary conditions is known, namely:

1. $-\Delta\varphi_k(\mathbf{x}) = \lambda_k\varphi_k(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega.$
2. $\varphi_k(\mathbf{x}) = 0$ (Dirichlet case) or $\frac{\partial}{\partial n}\varphi_k(\mathbf{x}) = 0$ (Neumann case), $\quad \forall \mathbf{x} \in \partial\Omega.$
3. $\langle \varphi_k, \varphi_j\rangle = \delta_{k,j}$, with $\delta_{k,j}$ being the Kronecker symbol.

Eigenfunctions and eigenvalues of the Laplacian possess some useful properties. Since $-\Delta$, coupled with homogeneous boundary conditions, is a positive definite operator, namely $\langle -\Delta u, u\rangle \geq 0$, its eigenvalues λ_k are all positive, and moreover, they form a divergent sequence, i.e.,

$$\lim_{k\to\infty} \lambda_k = +\infty.$$

Depending on the boundary condition, the functions $\{\varphi_k\}_{k\in\mathbb{N}}$ belong to one of the following spaces:

$$H_0^2(\Omega) = \left\{u : \Omega \to \mathbb{R} \,|\, u, \nabla u \in L^2(\Omega),\ u\big|_{\partial\Omega} = 0\right\} \quad \text{Dirichlet's case,}$$
$$H_n^2(\Omega) = \left\{u : \Omega \to \mathbb{R} \,|\, u, \nabla u \in L^2(\Omega),\ \tfrac{\partial}{\partial n}u\big|_{\partial\Omega} = 0\right\} \text{ Neumann's case,}$$

and they form a complete set in $L^2(\Omega)$. Thus, whenever $u \in L^2(\Omega)$, one can consider the Fourier expansion

$$u(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{u}_k\varphi_k(\mathbf{x}), \quad \hat{u}_k = \langle u, \varphi_k\rangle, \tag{2}$$

and entirely describe the action of the Laplacian by means of eigenvalues and eigenfunctions as

$$-\Delta u(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{u}_k \lambda_k \varphi_k(\mathbf{x}).$$

For $f \in L^2(\Omega)$, the Poisson equation

$$-\Delta u(\mathbf{x}) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{3}$$

Subject to homogeneous Dirichlet or Neumann boundary conditions on $\partial\Omega$, has solution $u(\mathbf{x}) \in L^2(\Omega)$ in view of the Lax–Milgram Lemma. Therefore, given the Fourier expansion $f(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{f}_k \varphi_k(\mathbf{x})$, Eq. (3) can be equivalently reformulated as

$$\sum_{k=0}^{\infty} \hat{u}_k \lambda_k \varphi_k(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{f}_k \varphi_k(\mathbf{x}) \quad \Longleftrightarrow \quad \sum_{k=0}^{\infty} \big(\hat{u}_k \lambda_k - \hat{f}_k\big) \varphi_k(\mathbf{x}) = 0,$$

and the solution $u(\mathbf{x})$ can be explicitly determined by its expansion (2) whose coefficients are $\hat{u}_k = \hat{f}_k / \lambda_k$, $k = 0, 1, \ldots$.

For a more in-depth treatment of properties of the spectral decomposition of the Laplacian, we refer to any classical textbook on the subject (e.g., [24]).

2.2 Spectral Fractional Laplacian

Let us now consider a set of orthonormal eigenfunctions and eigenvalues $\{\varphi_k, \lambda_k\}_{k \in \mathbb{N}}$ of $-\Delta$ in the domain $\Omega \subset \mathbb{R}^d$, coupled with homogeneous Dirichlet or Neumann boundary conditions on $\partial\Omega$.

Definition 1 Given $u \in L^2(\Omega)$, and its series representation (2), the *spectral fractional Laplacian* $\big(-\Delta\big)^s u(\mathbf{x})$ of non-integer order $0 < s < 1$ is defined as [4, 5, 27]

$$\big(-\Delta\big)^s u(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{u}_k \lambda_k^s \varphi_k(\mathbf{x}). \tag{4}$$

Clearly, (4) recovers the identity when $s = 0$ and the standard Laplacian when $s = 1$. Moreover, $\big(-\Delta\big)^s : \mathbb{H}^s(\Omega) \to \mathbb{H}^{-s}(\Omega)$, where for $s \in [-1, 1]$ it is

$$\mathbb{H}^s(\Omega) = \Big\{ u(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{u}_k \varphi_k(\mathbf{x}) \ : \ \sum_{k=0}^{\infty} \lambda_k^s \hat{u}_k^2 < \infty \Big\}.$$

Although (4) differs from other available definitions of the fractional Laplacian in bounded domains, it however turns out to be equivalent [5, Lemma 2.2] to the semigroup formula for the fractional Laplacian [37]

$$\big(-\Delta\big)^s u(\mathbf{x}) = \frac{1}{\Gamma(-s)} \int_0^\infty \frac{1}{t^{1+s}} \big(\mathrm{e}^{t\Delta} u(\mathbf{x}) - u(\mathbf{x})\big) \mathrm{d}t,$$

where $\mathrm{e}^{t\Delta} u(\mathbf{x})$ denotes the solution of the heat equation

$$\begin{cases} v_t(\mathbf{x}, t) = -\Delta v(\mathbf{x}, t), & (\mathbf{x}, t) \in \Omega \times (0, +\infty), \\ v(\mathbf{x}, 0) = u(\mathbf{x}), & \mathbf{x} \in \Omega, \end{cases}$$

subject to proper boundary conditions.

By following the same reasoning as in the previous subsection, the solution to the Poisson equation

$$\big(-\Delta\big)^s u(\mathbf{x}) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{5}$$

with homogeneous conditions on $\partial\Omega$ and $f \in L^2(\Omega)$, can be expressed as

$$u(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{u}_k \varphi_k(\mathbf{x}), \quad \hat{u}_k = \frac{\langle f, \varphi_k \rangle}{\lambda_k^s}.$$

Therefore, whenever $f \in \mathbb{H}^r(\Omega)$, for any $r \in \mathbb{R}$, it is $u \in \mathbb{H}^{r+2s}(\Omega)$. More detailed results about regularity of the solution of (5) are available, for instance, in [5, 23, 27, 31, 32]. However, for the purposes of this chapter, a spatial regularity of L^2 type will be sufficient.

Obviously, for a function $u(\mathbf{x}, t)$ depending on time as well, coefficients $\hat{u}_k$ will be time-dependent, and hence,

$$\big(-\Delta\big)^s u(\mathbf{x}, t) = \sum_{k=0}^{\infty} \hat{u}_k(t) \lambda_k^s \varphi_k(\mathbf{x}). \tag{6}$$

Remark 1 Definition 1 applies only with homogeneous boundary conditions. The extension to non-homogeneous conditions is however possible [1] but involves further technical difficulties. Since the focus of this chapter is mainly related on studying the effectiveness of a class of time integrators when applied to (1), to keep the treatment at a simpler level, we confine the analysis just to vanishing boundary conditions.

3 Fractional-Time-Space Differential Equation

To solve (1), we assume $f, u_0 \in L^2(\Omega)$, and since $\{\varphi_k\}_{k\in\mathbb{N}}$ forms a complete set, we can consider the following representations of the initial function $u_0(\mathbf{x})$ and of the source term $f(\mathbf{x}, t)$

$$u_0(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{u}_{0,k}\varphi_k(\mathbf{x}), \qquad f(\mathbf{x}, t) = \sum_{k=0}^{\infty} \hat{f}_k(t)\varphi_k(\mathbf{x}),$$

and the solution $u(\mathbf{x}, t)$ of the FTSDE (1) can be written as

$$u(\mathbf{x}, t) = \sum_{k=0}^{\infty} \hat{u}_k(t)\varphi_k(\mathbf{x}),$$

where obviously it is $\hat{u}_{0,k} = \hat{u}_k(0)$. After applying the definition (6) of the fractional Laplacian, and replacing the above representation of the solution, FTSDE (1) can be reformulated as

$${}_0^{\mathrm{C}}D_t^{\alpha} \sum_{k=0}^{\infty} \hat{u}_k(t)\varphi_k(\mathbf{x}) + \nu \sum_{k=1}^{\infty} \hat{u}_k(t)\lambda_k^s\varphi_k(\mathbf{x}) = \sum_{k=0}^{\infty} \hat{f}_k(t)\varphi_k(\mathbf{x}),$$

and thus, in view of the linearity of ${}_0^{\mathrm{C}}D_t^{\alpha}$, we can write

$$\sum_{k=0}^{\infty} \left[{}_0^{\mathrm{C}}D_t^{\alpha}\hat{u}_k(t) + \nu\lambda_k^s\hat{u}_k(t) - \hat{f}_k(t) \right] \varphi_k(\mathbf{x}) = 0$$

thanks to which the original FTSDE (1) is decomposed into the sequence of the initial value problems for linear FDEs

$$\begin{cases} {}_0^{\mathrm{C}}D_t^{\alpha}\hat{u}_k(t) = -\nu\lambda_k^s\hat{u}_k(t) + \hat{f}_k(t), \quad k \in \mathbb{N}, \\ \hat{u}_k(0) = \hat{u}_{0,k}. \end{cases} \tag{7}$$

The idea we propose in this chapter is to fix a number $K \in \mathbb{N}$ of terms and consider a truncated series expansion $u_K(\mathbf{x}, t)$ of the solution $u(\mathbf{x}, t)$, namely

$$u_K(\mathbf{x}, t) = \sum_{k=0}^{K} \hat{u}_k(t)\varphi_k(\mathbf{x}), \tag{8}$$

so to approximate each solution $\hat{u}_k(t)$ of (7) by an efficient method for FDEs.

An accurate error analysis will be carried out in order to ensure that, for an assigned tolerance $\varepsilon > 0$, it is possible to predict the number K of terms necessary to achieve the assigned accuracy, namely such that $\|u(\mathbf{x}, t) - u_K(\mathbf{x}, t)\|_{L^2(\Omega)} \leq \varepsilon$ for any $t \in [0, T]$.

4 Generalized Exponential Time-Differencing Methods

Each FDE (7) by which the original TSFDE (1) is decomposed can be solved by any standard method for FDEs, as for instance product-integration (PI) rules. These rules are based on the integral representation of (7)

$$\hat{u}_k(t) = \hat{u}_{0,k} + \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} \big(- \nu \lambda_k^s \hat{u}_k(\tau) + \hat{f}_k(\tau) \big) \mathrm{d}\tau$$

and on the approximation of the vector field $-\nu\lambda_k^s \hat{u}_k(t) + \hat{f}_k(t)$ by some piecewise interpolant polynomial.

Although PI rules are widely employed for FDEs, they however suffer from some limitations. Exact solutions of FDEs indeed usually expand in integer and fractional powers as $t \to 0$ [30], and hence, they lack smoothness at the origin where derivatives are in general unbounded [38]. This lack of regularity, which is inherited by the vector field, does not allow PI rules to achieve high accuracy since, as it is well known, polynomials approximate in a quite poor way functions with unbounded derivatives. In practice, without constructing data in a unreasonably artificial way, even obtaining order of convergence equal to 2 is not possible [14, 15].

For this reason, it may be more convenient to exploit an alternative, but equivalent, representation of the exact solution $\hat{u}_k(t)$ based on the following variation-of-constant formula [20]

$$\hat{u}_k(t) = e_{\alpha,1}\big(t; \nu\lambda_k^s\big)\hat{u}_{0,k} + \int_0^t e_{\alpha,\alpha}\big(t-\tau; \nu\lambda_k^s\big)\hat{f}_k(\tau)\mathrm{d}\tau, \tag{9}$$

where, for real $t > 0$ and any (possibly complex) argument λ, $e_{\alpha,\beta}(t;\lambda)$ denotes the following generalization of the two-parameter Mittag–Leffler (ML) function

$$e_{\alpha,\beta}(t;\lambda) := t^{\beta-1} E_{\alpha,\beta}(-t^\alpha \lambda), \qquad E_{\alpha,\beta}(z) = \sum_{j=0}^{\infty} \frac{z^j}{\Gamma(\alpha j + \beta)}, \quad \alpha > 0,\ \beta \in \mathbb{R}. \tag{10}$$

The ML function plays in fractional calculus the same fundamental role played by the exponential in integer-order calculus (and, indeed, the exponential is the special instance of the ML function when $\alpha = \beta = 1$). We do not pursue here a detailed investigation of the ML function (for which we refer the reader to the recent monograph [22]). We just list the main properties concerning differentiation and integration and which will be used later on:

$$\frac{\mathrm{d}}{\mathrm{d}t} e_{\alpha,\beta}(t;\lambda) = e_{\alpha,\beta-1}(t;\lambda)$$

and

$$\frac{1}{\Gamma(\gamma)}\int_0^t e_{\alpha,\beta}\big(t-\tau;\lambda\big)\tau^{\gamma-1}\mathrm{d}\tau = e_{\alpha,\beta+\gamma}\big(t;\lambda\big), \quad \gamma > 0. \tag{11}$$

In particular, a special case of (11), which will be of interest throughout this chapter, is given by the relationship

$$\int_0^t e_{\alpha,\alpha}\big(\tau;\lambda\big)\mathrm{d}\tau = e_{\alpha,\alpha+1}\big(t;\lambda\big). \tag{12}$$

Moreover, we will make use of the following properties of the ML function.

Proposition 1 *Let* $0 < \alpha \leq 1$ *and real* $\lambda > 0$. *Then for any* $t \geq 0$, *it is*

$$e_{\alpha,1}(t;\lambda) \geq 0 \quad \textit{with} \quad \max_{t\geq 0} e_{\alpha,1}(t;\lambda) = 1 \tag{13}$$

and

$$e_{\alpha,\alpha+1}(t;\lambda) \geq 0 \quad \textit{with} \quad \max_{t\geq 0} e_{\alpha,\alpha+1}(t;\lambda) = \lim_{t\to\infty} e_{\alpha,\alpha+1}(t;\lambda) = \frac{1}{\lambda}. \tag{14}$$

Proof The function $e_{\alpha,1}(t;\lambda) = E_{\alpha,1}(-t^\alpha\lambda)$ is completely monotonic (CM) for $0 < \alpha \leq 1$, i.e.,

$$(-1)^k \frac{\mathrm{d}^k}{\mathrm{d}t^k} E_{\alpha,1}(-t^\alpha\lambda) \geq 0, \quad k = 0, 1, \ldots.$$

since it is the composition of the two CM functions $E_\alpha(-t)$ and t^α. Hence, $E_{\alpha,1}(-t^\alpha\lambda)$ is a non-negative and decreasing function, and (13) immediately follows since $e_{\alpha,1}(0;\lambda) = 1$. By a simple manipulation of the series representation of the ML (10), it is immediate to see that

$$\begin{aligned} e_{\alpha,\alpha+1}(t;\lambda) &= t^\alpha \sum_{k=0}^{\infty} \frac{t^{\alpha k}(-\lambda)^k}{\Gamma(\alpha k+\alpha+1)} = \sum_{k=0}^{\infty} \frac{t^{\alpha(k+1)}(-\lambda)^k}{\Gamma(\alpha(k+1)+1)} \\ &= -\frac{1}{\lambda}\sum_{k=1}^{\infty} \frac{t^{\alpha k}(-\lambda)^k}{\Gamma(\alpha k+1)} = \frac{1}{\lambda} - \frac{1}{\lambda} E_{\alpha,1}(-t^\alpha\lambda), \end{aligned}$$

and, again, the proof follows from the CM property of $E_{\alpha,1}(-t^\alpha\lambda)$. □

The advantage of using (9) lies in the possibility of deriving numerical schemes in which only the source term $\hat{f}_k(t)$, instead of the whole vector field $-\nu\lambda_k^s\hat{u}_k(\tau) + \hat{f}_k(\tau)$, is approximated by means of polynomials. In applications, indeed, it is reasonable to assume that source terms $\hat{f}_k(t)$ possess some regularity, and hence,

they can be accurately approximated by polynomials. Moreover, in some cases, the convolution integral of the ML function in (9) can be analytically determined: this is the case, for instance, in which $\hat{f}_k(t)$ is a polynomial or even a combination of real powers for which (11) applies. In some other cases, this convolution integral can be approximated, with high accuracy, by numerical methods for the inversion of the Laplace transform.

Once considered a grid $t_n = nh$ with a constant step size $h > 0$ and, after rewriting (9) in a piecewise way

$$\hat{u}_k(t_n) = e_{\alpha,1}\big(t_n; \nu\lambda_k^s\big)\hat{u}_{0,k} + \sum_{j=0}^{n-1} \int_{t_j}^{t_{j+1}} e_{\alpha,\alpha}\big(t_n - \tau; \nu\lambda_k^s\big)\hat{f}_k(\tau)\mathrm{d}\tau, \tag{15}$$

we approximate $\hat{f}_k(t)$ in each interval $[t_j, t_{j+1}]$ by means of an interpolant polynomial. This approach, which has been studied in [19, 20, 34], is known as *generalized exponential time-differencing* (GETD) method and generalizes to fractional-order problems some families of exponential integrators introduced for integer-order problems (see, for instance, [25]).

According to the degree of the polynomial interpolants, different GETD methods can be devised. In this chapter, we are interested in using a second-order polynomial interpolant

$$p_{k,j}(t) = \frac{(t - t_j)}{h}\hat{f}_k(t_{j+1}) - \frac{(t - t_{j+1})}{h}\hat{f}_k(t_j), \quad t \in [t_j, t_{j+1}],$$

which leads to the following approximation of (9)

$$\hat{u}_k^h(t_n) = e_{\alpha,1}\big(t_n; \nu\lambda_k^s\big)\hat{u}_{0,k} + h^\alpha\Big[w_n(h^\alpha\nu\lambda_k^s)\hat{f}_k(0) + \sum_{j=1}^{n}\omega_{n-j}(h^\alpha\nu\lambda_k^s)\hat{f}_k(t_j)\Big], \tag{16}$$

where coefficients $w_n(z)$ and $\omega_n(z)$ are, respectively, given by

$$w_n(z) = e_{\alpha,\alpha+2}\big(n-1; z\big) - e_{\alpha,\alpha+2}\big(n; z\big) + e_{\alpha,\alpha+1}\big(n-1; z\big)$$

and

$$\omega_n(z) = \begin{cases} e_{\alpha,\alpha+2}\big(1; z\big), & n = 0, \\ e_{\alpha,\alpha+2}\big(n-1; z\big) - 2e_{\alpha,\alpha+2}\big(n; z\big) + e_{\alpha,\alpha+2}\big(n+1; z\big), & n \geq 1. \end{cases}$$

The evaluation of these coefficients does not represent a particular issue, since methods for the efficient and accurate computation of the ML function are nowadays available [18, 21].

The advantages of using the trapezoidal GETD method (16) to solve (9), instead of a classical trapezoidal PI rule, is twofold.

First, it is possible to approximate the solution $\hat{u}_k^h(t_n)$ of (9) directly at any time t_n without the need of evaluating, in a step-by-step way, approximations at the previous time steps. In this way, a consistent amount of computation can be saved if just the solution at t_n is requested, since a number of floating-point operations proportional to n, instead of n^2, are requested. At the same time, storage needs are also reduced since it is not necessary to store all the previous approximations $\hat{u}_k^h(t_j)$, $j = 0, 1, \ldots, n-1$ of the solution to compute $\hat{u}_k^h(t_n)$.

Moreover, full order 2 of convergence is achieved under a reasonable smoothness assumption for $\hat{f}_k(t)$, thus avoiding the limitations of the classical trapezoidal PI rules that instead, when applied to (7), converge with order $1 + \alpha$ [14, 15]. The same limitation holds for classical PI rules based on polynomials of higher degree as well, while higher order can be obtained with GETD schemes. In particular, for the trapezoidal GETD method (16), the following result holds [17].

Theorem 1 *Let $0 < \alpha < 1$ and $\hat{f}_k(t) \in C^2[0, T]$. Then for any $t_n > 0$, it is*

$$|\hat{u}_k(t_n) - \hat{u}_k^h(t_n)| = \frac{h^2}{12} e_{\alpha,\alpha+1}\left(t_n; \nu\lambda_k^s\right) |\hat{f}_k^{(2)}(\eta_n)| + O\left(h^{2+\alpha}\right), \quad h \to 0,$$

where $\eta_n \in (0, t_n)$.

5 Error Analysis

In order to perform an error analysis of the proposed approach, and hence select the number K of terms in the truncated series expansion (8) and the step size h for the GETD scheme (16) to achieve a given accuracy, we have to consider two different sources of errors.

Indeed, since each term $\hat{u}_k(t)$ in the truncated series expansion (8) is approximated by the solution $\hat{u}_k^h(t)$ provided by the GETD method, then to actually compute

$$u_K^h(\mathbf{x}, t) := \sum_{k=0}^{K} \hat{u}_k^h(t)\varphi_k(\mathbf{x}),$$

the overall error is given by

$$\begin{aligned} |u(\mathbf{x}, t) - u_K^h(\mathbf{x}, t)| &\le |u(\mathbf{x}, t) - u_K(\mathbf{x}, t)| + |u_K(\mathbf{x}, t) - u_K^h(\mathbf{x}, t)| \\ &= \mathrm{Err}_K^{\mathrm{T}}(\mathbf{x}, t) + \mathrm{Err}_{K,h}^{\mathrm{D}}(\mathbf{x}, t), \end{aligned} \tag{17}$$

where we identify a *truncation error*

$$\mathrm{Err}_K^{\mathrm{T}}(\mathbf{x},t) = |u(\mathbf{x},t) - u_K(\mathbf{x},t)|,$$

which is solely due to the truncation (8) of the infinite series describing the solution $u(\mathbf{x},t)$ and the *discretization error*

$$\mathrm{Err}_{K,h}^{\mathrm{D}}(\mathbf{x},t) = |u_K(\mathbf{x},t) - u_K^h(\mathbf{x},t)|,$$

which instead is related to the convergence properties of the GETD scheme used to approximate each of the FDEs (7).

We therefore study here both kinds of errors, and we provide some bounds for both $\mathrm{Err}_K^{\mathrm{T}}$ and $\mathrm{Err}_K^{\mathrm{D}}$.

Theorem 2 (Bound for the Truncation Error) *Let $0 < \alpha < 1$, $0 < s < 1$, $f \in L^2(\Omega \times [0,T])$. Then for any $K \geq 0$ and $t \in [0,T]$, it is*

$$\|u(\cdot,t) - u_K(\cdot,t)\|_{L^2(\Omega)}^2 \leq \sum_{k=K+1}^{\infty} \left(|\hat{u}_{0,k}| e_{\alpha,1}\left(t; \nu\lambda_k^s\right) + \|\hat{f}_k\|_{L^2([0,t])} e_{\alpha,\alpha+1}\left(t; \nu\lambda_k^s\right) \right)^2.$$

Proof Let $K \geq 0$ and $t \in [0,T]$ be given. We start by observing that for $\mathbf{x} \in \Omega$, in view of (9), it is

$$u(\mathbf{x},t) - u_K(\mathbf{x},t) = \sum_{k=K+1}^{\infty} e_{\alpha,1}\left(t; \nu\lambda_k^s\right) \hat{u}_{0,k} \varphi_k(\mathbf{x}) + \sum_{k=K+1}^{\infty} \left(\hat{f}_k * e_{\alpha,\alpha}(\cdot; \nu\lambda_k^s) \right)(t) \varphi_k(\mathbf{x}),$$

where for convenience we write

$$\left(\hat{f}_k * e_{\alpha,\alpha}(\cdot; z) \right)(t) = \int_0^t e_{\alpha,\alpha}(\tau; z)\, \hat{f}_k(t-\tau)\, \mathrm{d}\tau.$$

Since $\{\varphi_k\}$ is a complete orthonormal set, then Parseval's identity holds, and therefore,

$$\begin{aligned} \|u(\cdot,t) - u_K(\cdot,t)\|_{L^2(\Omega)}^2 &= \\ &= \sum_{k=K+1}^{\infty} \left| e_{\alpha,1}\left(t; \nu\lambda_k^s\right) \hat{u}_{0,k} + \left(\hat{f}_k * e_{\alpha,\alpha}\left(\cdot; \nu\lambda_k^s\right) \right)(t) \right|^2 \\ &\leq \sum_{k=K+1}^{\infty} \left(|e_{\alpha,1}\left(t; \nu\lambda_k^s\right)| \cdot |\hat{u}_{0,k}| + \left| \left(\hat{f}_k * e_{\alpha,\alpha}\left(\cdot; \nu\lambda_k^s\right) \right)(t) \right| \right)^2. \end{aligned} \tag{18}$$

Let us now consider the term $\left|\left(\hat{f}_k * e_{\alpha,\alpha}\left(\cdot;\nu\lambda_k^s\right)\right)(t)\right|^2$, and, by recalling the Young's inequality (e.g., see [2, Theorem 3]), if $f \in L^p(\mathbb{R}^n)$, $g \in L^q(\mathbb{R}^n)$, $1 \le p, q, r \le \infty$, with $\frac{1}{r} = \frac{1}{p} + \frac{1}{q} - 1$, it is

$$\|f * g\|_r \le (A_p \cdot A_q \cdot A_{r'})^n \|f\|_p \cdot \|g\|_q,$$

where $A_m := \left(m^{\frac{1}{m}}/m'^{\frac{1}{m'}}\right)^{\frac{1}{2}}$ and the prime denotes the dual exponent.

As a consequence of the assumptions, it holds that $\hat{f}_k \in L^2([0,t])$ for all k and $e_{\alpha,\alpha} \in L^\infty([0,t]) \cap L^p([0,t])$ for all $p \le 1$. It thus follows, with $p = r = 2$ and $q = 1$, which in turn provide $A_1 A_2^2 = 1$ after some algebra, that

$$\left|\left(\hat{f}_k * e_{\alpha,\alpha}\left(\cdot;\nu\lambda_k^s\right)\right)(t)\right|^2 \le \|\hat{f}_k\|^2_{L^2([0,t])} \cdot \|e_{\alpha,\alpha}\left(\cdot;\nu\lambda_k^s\right)\|^2_{L^1([0,t])}. \tag{19}$$

Since $e_{\alpha,\alpha}(t;\nu\lambda_k^s) \ge 0$ for all $t \ge 0$, we can exploit (12) to observe that

$$\|e_{\alpha,\alpha}(\cdot;\nu\lambda_k^s)\|_{L^1([0,t])} = \int_0^t |e_{\alpha,\alpha}(\tau;\nu\lambda_k^s)|\,\mathrm{d}\tau = e_{\alpha,\alpha+1}(t;\nu\lambda_k^s),$$

from which the proof follows. □

A more pessimistic bound can be obtained by means of the following result. We think that this result is useful mainly to highlight the way by which the truncation error is related to the number K of terms in the series expansion, but for a more accurate error estimation, the previous result must be kept into account.

Corollary 1 *Let $0 < \alpha < 1$, $0 < s < 1$, $f \in L^2(\Omega \times [0,T])$. Then for any $K \ge 0$ and $t \in [0,T]$, it is*

$$\|u(\cdot,t) - u_K(\cdot,t)\|^2_{L^2(\Omega)} \le \sum_{k=K+1}^{\infty} \left(|\hat{u}_{0,k}| + \frac{1}{\nu\lambda_k^s}\|\hat{f}_k\|_{L^2([0,t])}\right)^2.$$

Proof From Proposition 1, we know that

$$|e_{\alpha,1}\left(t;\nu\lambda_k^s\right)| \cdot |\hat{u}_{0,k}| \le |\hat{u}_{0,k}|,$$

$$\|\hat{f}_k\|_{L^2([0,t])} \cdot e_{\alpha,\alpha+1}\left(t;\nu\lambda_k^s\right) \le \frac{1}{\nu\lambda_k^s}\|\hat{f}_k\|_{L^2([0,t])}.$$

Thus, the claim immediately follows from Theorem 2. □

As far as it concerns the discretization error, we can see the following result.

Theorem 3 (Bound for the Discretization Error) *Let $0 < \alpha < 1$, $0 < s < 1$, $f \in L^2(\Omega \times [0,T])$. Then for any $K \ge 0$, $t_n \in [0,T]$, and sufficiently small $h > 0$,*

it is

$$\|u(\cdot,t_n)-u_K(\cdot,t_n)\|_{L^2(\Omega)} \le \frac{h^2}{12}\sqrt{\sum_{k=0}^{K}\left(M_{k,n}e_{\alpha,\alpha+1}(t_n;\,\nu\lambda_k^s)\right)^2},$$

where $M_{k,n}=\max_{t\in[0,t_n]}\left|\hat{f}_k^{(2)}(t)\right|$.

Proof The proof easily follows from applying Theorem 1 to the truncation error. □

6 Numerical Experiments

In this section, we present some numerical experiments with the aim of verifying the theoretical findings concerning the error estimates. In particular, we will check how reliable are the bounds for the truncation error and for the discretization error, respectively, given by Theorem 2 and 3, in order to provide the estimate of the overall error (17). We will not use the bound from Corollary 1 since it does not strictly depend on the time t on which the solution is evaluated, but it depends on the whole interval $[0,t]$, and therefore, it is in general too pessimistic.

All tests are carried out in Matlab ver. 9.10.0.1649659 (R2021a) on a computer equipped with an Intel(R) i7-9700 CPU running at 3.00GHz (with 16.0 GB of RAM) under the 64 bit Windows 11 Pro operative system.

6.1 Homogeneous Dirichlet Boundary Conditions in a 1D Domain

In the first example, we consider a one-dimensional domain $[a,b]$ in which the FTSDE (1) is subject to homogeneous boundary conditions of Dirichlet type. The analytical formulation of the eigenpairs (λ_k,φ_k) of the Laplacian is available according to

$$\lambda_k=\left(\frac{(k+1)\pi}{b-a}\right)^2,\quad \varphi_k(x)=\sqrt{\frac{2}{b-a}}\sin\left(\frac{(k+1)\pi(x-a)}{b-a}\right),\quad k=0,1,\dots.$$

We consider the initial condition and the source term for the FTSDE (1), respectively, given by

$$u_0(x)=(x-a)(b-x),\quad f(x,t)=(1-\mathrm{e}^{-\sigma t})u_0(x),\quad \sigma>0, \tag{20}$$

for which, by elementary derivations, it is possible to derive the corresponding series expansion with respect to the orthonormal basis $\{\varphi_k\}_{k\in\mathbb{N}}$

$$u_0(x) = \sum_{k=0}^{\infty} \hat{u}_{0,k}\varphi_k(x), \quad \hat{u}_{0,k} = \begin{cases} 0, & k \text{ odd}, \\ 4\sqrt{\frac{2}{b-a}}\left(\frac{b-a}{(k+1)\pi}\right)^3, & k \text{ even}, \end{cases}$$

and

$$f(x,t) = \sum_{k=0}^{\infty} \hat{f}_k(t)\varphi_k(x), \quad \hat{f}_k(t) = (1-\mathrm{e}^{-\sigma t})\hat{u}_{0,k}.$$

By exploiting the series representation of the exponential e^{-ct} and using (11), one can see that the exact solution of each FDE (7), obtained by means of the variation-of-constant formula (9), is given by

$$\hat{u}_k(t) = e_{\alpha,1}\left(t;\nu\lambda_k^s\right)\hat{u}_{0,k} - \sum_{j=1}^{\infty}(-1)^j\sigma^j e_{\alpha,\alpha+j+1}\left(t;\nu\lambda_k^s\right)\hat{u}_{0,k},$$

thus leading to an analytical representation of the exact solution $u(x,t)$ to (1) in terms of series of the ML function

$$u(x,t) = \sum_{\substack{k=0 \\ k \text{ even}}}^{\infty}\left[e_{\alpha,1}\left(t;\nu\lambda_k^s\right) - \sum_{j=1}^{\infty}(-1)^j\sigma^j e_{\alpha,\alpha+j+1}\left(t;\nu\lambda_k^s\right)\right]\hat{u}_{0,k}\varphi_k(x).$$

The above solution can be approximated, even with very high accuracy (although in a quite expensive way), after truncating the two summations at a sufficiently large number of terms and computing the ML functions by the code devised in [18] that allows to perform the computation with an accuracy close to the precision machine. This approximation will be used as reference solution.

The solution plot in the interval $[a,b] = [0,2]$ as $t \in [0,4]$ is shown in the left plot of Fig. 1 for $\alpha = 0.8$, $s = 0.6$, $\nu = 0.2$, and $\sigma = 0.8$.

In the right plot of the same Fig. 1, it is instead presented a comparison between the error actually obtained at $T = 4$ after integrating each of the FDEs (7) by the GETD method (16) and the error estimate (17) as the number K of the considered series terms varies. The bounds obtained by Theorems 2 and 3 are used for the truncation and discretization errors (just the terms from $K+1$ to $2K$ are used to determine these bounds). For the term $\|\hat{f}_k\|_{L^2[0,t]}$ that appears in both bounds for the truncation and discretization errors, one can easily compute

$$\|\hat{f}_k\|^2_{L^2[0,t]} = \left(t + \frac{2}{\sigma}\mathrm{e}^{-\sigma t} - \frac{1}{2c}\mathrm{e}^{-2\sigma t} - \frac{3}{2\sigma}\right)|\hat{u}_{0,k}|.$$

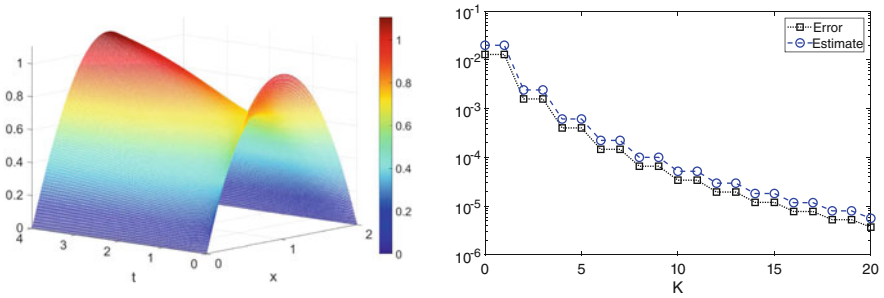


Fig. 1 Solution of problem (20) for $\alpha = 0.8$, $s = 0.6$, $\nu = 0.2$, and $\sigma = 0.8$ (left plot) and comparison between actual error and estimate for $h = 2^{-12}$

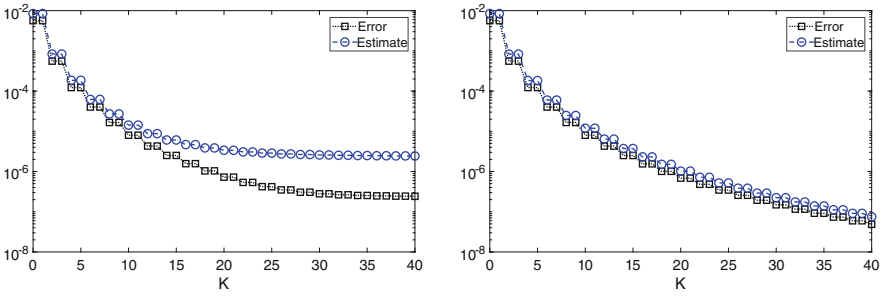


Fig. 2 Comparison between actual error and estimate for problem (20) with $\alpha = 0.7$, $s = 0.8$, $\nu = 0.2$, and $\sigma = 0.8$ and $h = 2^{-7}$ (left plot) and $h = 2^{-12}$ (right plot)

The same experiment is repeated, after changing the fractional differentiation orders to $\alpha = 0.7$ and $s = 0.8$, and the comparison between the actual errors and the estimates are presented in Fig. 2: here, in the left plot, computation has been performed with a step size $h = 2^{-7}$ and in the right plot with $h = 2^{-12}$.

In the left plot, it is possible to note the effect of the discretization error, which is proportional to h^2 and provides a limit for the truncation error, preventing it to decrease below its value. In the right plot, having used a smaller step size $h = 2^{-12}$, the truncation error produces less limitations in the decreasing of the overall error. Although the estimate of the truncation error is more pessimistic (maybe due to pessimistic bound $M_{k,n}$ in Theorem 3), results in Fig. 2 however well highlight the need of taking into account both truncation and discretization errors in the error analysis.

To provide a rough idea of the involved computation, in Fig. 3, we present a comparison between accuracy (errors in L_2 norm) and computational time (expressed in CPU seconds) for the same increasing number K of series terms of the previous test. Clearly, execution times are affected not only by the number of the series terms (which increases from right to left in both plots) but also by the step size used in the GETD ($h = 2^{-7}$ in the left plot and $h = 2^{-12}$ in the right plot).

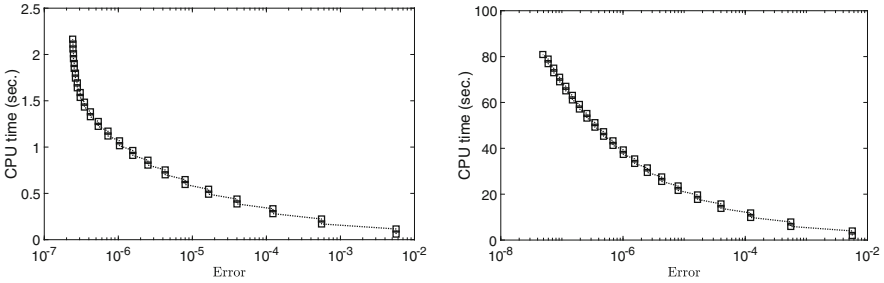


Fig. 3 Comparison between actual error and estimate for problem (20) with $\alpha = 0.7$, $s = 0.8$, $\nu = 0.2$ and $\sigma = 0.8$ and $h = 2^{-7}$ (left plot) and $h = 2^{-12}$ (right plot)

Obviously, these outcomes do not provide a complete view of the computational cost of methods exploiting the spectral definition of the fractional Laplacian, since in this special case eigenvalues and eigenfunctions are known analytically, while on general domains they must be numerically approximated, an usually expensive task. They however highlight the need of performing an accurate error analysis in order to better calibrate all parameters, in particular the step size of the GETD method, to avoid not strictly necessary computations.

The left plot of Fig. 3 well clarifies this point. Once the maximum accuracy obtainable from GETD has been achieved, namely $O(h^2)$, increasing the number K of series terms does not lead to a real improvement of the overall accuracy but, instead, involves a major, and useless, computational cost. Considering a large number of series terms makes sense only when the accuracy of the underlying GETD method is improved (see the right plot), but a consistently larger amount of computation must be taken into account in this case.

6.2 *Homogeneous Neumann Boundary Conditions in a 1D Domain*

We consider now a one-dimensional domain $[a, b]$ in which the FTSDE (1) is subject to homogeneous boundary conditions of Neumann type and for which the corresponding eigenpairs (λ_k, φ_k) are

$$\lambda_k = \left(\frac{k\pi}{b-a}\right)^2, \quad \varphi_k(x) = \begin{cases} \sqrt{\frac{1}{b-a}} & k = 0 \\ \sqrt{\frac{2}{b-a}} \cos\left(\frac{k\pi(x-a)}{b-a}\right) & k = 1, 2, \ldots \end{cases}$$

As initial condition and source term, on the interval $[a, b] = [0, 2]$, we consider the functions

$$u_0(x) = x^4 - 4x^3 + 4x^2, \quad f(x,t) = \cos(\sigma t)\left(x^4 - 4x^3 + 4x^2\right), \quad \sigma > 0 \qquad (21)$$

for which it is not difficult to evaluate

$$u_0(x) = \sum_{k=0}^{\infty} \hat{u}_{0,k}\varphi_k(x), \quad \hat{u}_{0,k} = \begin{cases} \frac{8\sqrt{2}}{15}, & k = 0, \\ 0, & k \text{ odd}, \\ -3\left(\frac{4}{k\pi}\right)^4, & k \geq 2 \text{ even}, \end{cases}$$

and

$$f(x,t) = \sum_{k=0}^{\infty} \hat{f}_k(t)\varphi_k(x), \quad \hat{f}_k(t) = \cos(\sigma t)\hat{u}_{0,k}.$$

The exact solution $u(\mathbf{x}, t)$ is obtained by exploiting the series representation of $\cos(\sigma t)$ together with Eq. (11)

$$u(\mathbf{x}, t) = \sum_{\substack{k=0 \\ k \text{ even}}}^{\infty} \left[e_{\alpha,1}\left(t; \nu\lambda_k^s\right) - \sum_{j=0}^{\infty} (-1)^j \sigma^{2j} e_{\alpha,\alpha+2j+1}\left(t; \nu\lambda_k^s\right) \right] \hat{u}_{0,k}\varphi_k(x),$$

and also in this case, an approximation obtained after truncation of the infinite series will be used as a reference solution. A plot of this is presented in the left plot of Fig. 4 for $\alpha = 0.8$, $s = 0.6$, $\nu = 0.2$, and $\sigma = 2.0$.

The right plot of Fig. 4 instead presents the comparison between the actual error at $T = 4$ and the corresponding error estimate provided by Eq. (17) and using the bounds derived in Theorems 2 and 3. Also in this case, the error estimates provide accurate predictions of the actual errors, thus offering a useful tool to predict the number K of terms necessary to achieve a given accuracy.

Again the experiment is repeated by changing the fractional orders in $\alpha = 0.7$ and $s = 0.8$, and the results are presented in Fig. 5 too $h = 2^{-7}$ (left plot) and $h = 2^{-12}$ (right plot).

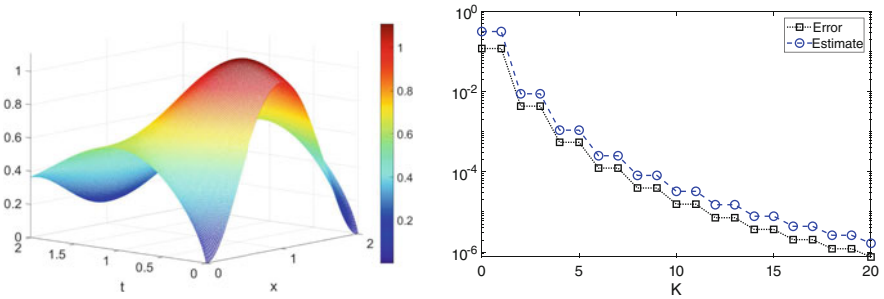


Fig. 4 Solution of problem (21) for $\alpha = 0.8$, $s = 0.6$, $\nu = 0.2$, and $\sigma = 2.0$ (left plot) and comparison between actual error and estimate for $h = 2^{-12}$

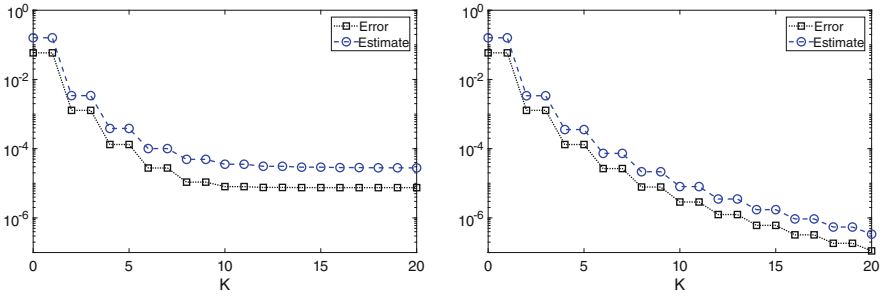


Fig. 5 Comparison between actual error and estimate for problem (21) with $\alpha = 0.7$, $s = 0.8$, $\nu = 0.2$ and $\sigma = 2.0$, and $h = 2^{-7}$ (left plot) and $h = 2^{-12}$ (right plot)

It is clear also in this case the action of the truncation error in limiting the decay of the error with respect to the increase of the number K of terms used in the computation.

6.3 Homogeneous Dirichlet Boundary Conditions in a 2D Domain

We now consider a two-dimensional rectangular domain $[a, b] \times [c, d]$ and assume that the FTSDE (1) is subject to homogeneous Dirichlet boundary conditions. In this case, the eigenpairs (λ_k, φ_k) are given by

$$\lambda_{j,k} = \left(\frac{(j+1)\pi}{b-a}\right)^2 + \left(\frac{(k+1)\pi}{d-c}\right)^2,$$

$$\varphi_{j,k}(x, y) = \sqrt{\frac{2}{b-a}}\sqrt{\frac{2}{d-c}} \sin\left(\frac{(j+1)\pi(x-a)}{b-a}\right) \sin\left(\frac{(k+1)\pi(y-c)}{d-c}\right)$$

for $j, k = 0, 1, \ldots$.

As initial condition and source term, we consider the functions

$$u_0(x) = (x-a)(b-x)(y-c)(d-y), \quad f(x,t) = \left(1 - \mathrm{e}^{-\sigma t}\right) u_0(x), \quad \sigma > 0. \tag{22}$$

Note that now the truncated solution is

$$u^h_{J,K}(\mathbf{x}, t) = \sum_{j=0}^{J} \sum_{k=0}^{K} \hat{u}^h_{j,k}(t) \varphi_{j,k}(\mathbf{x}),$$

and in the plots, we report errors and estimates when $J = K$.

The solution in the square domain $[a, b] \times [c, d] = [0, 1] \times [0, 1]$ for $\alpha = 0.8$, $s = 0.6$, $\nu = 0.2$, and $\sigma = 0.5$ is presented in the left plot of Fig. 6 together with the comparison between actual error and estimates obtained thanks to Theorems 2 and 3. Also in the 2D case the error estimates seem to provide an efficient prediction of the actual error.

In Fig. 7, the same experiment is repeated after changing the fractional order in $\alpha = 0.7$ and $s = 0.8$ and using two different step sizes $h = 2^{-7}$ (left plot) and $h = 2^{-12}$ (right plot) in the GETD method.

Similarly to the 1D case, we observe that also in the 2D the contribution of the truncation and discretization errors is in line with the theoretical findings.

Also in the 2D case, we present, in Fig. 8, a comparison between errors and computational times. Obviously, the 2D problem leads to a remarkable increase of computation, but, as in the 1D case, we observe the importance of a detailed error analysis in order to establish the proper combination of terms in the series expansion and step size in the GETD procedure in order to achieve a given accuracy with a reasonable computational task.

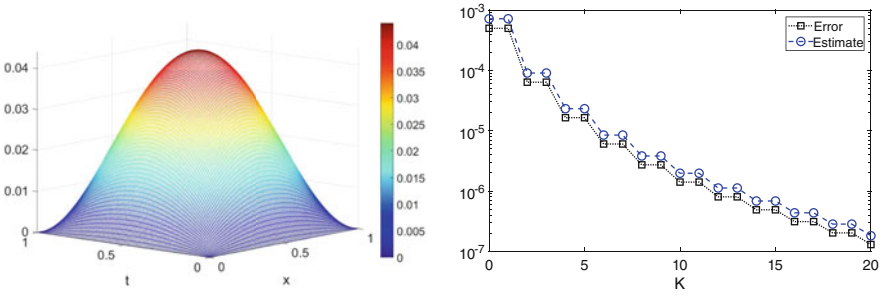


Fig. 6 Solution at $T = 4$ of problem (22) for $\alpha = 0.8$, $s = 0.6$, $\nu = 0.2$, and $\sigma = 0.5$ (left plot) and comparison between actual error and estimate for $h = 2^{-12}$

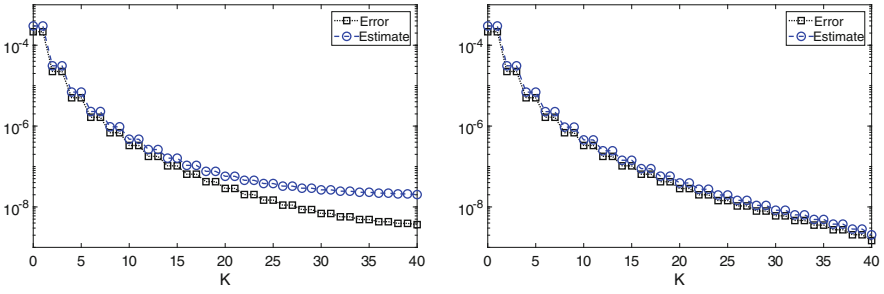


Fig. 7 Comparison between actual error and estimate for problem (22) with $\alpha = 0.7$, $s = 0.8$, $\nu = 0.2$ and $\sigma = 0.5$ and $h = 2^{-7}$ (left plot) and $h = 2^{-12}$ (right plot)

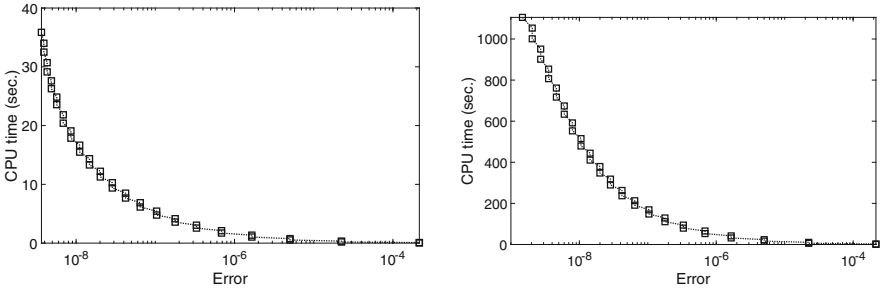


Fig. 8 Comparison between actual error and estimate for problem (20) with $\alpha = 0.7$, $s = 0.8$, $\nu = 0.2$ and $\sigma = 0.8$, and $h = 2^{-7}$ (left plot) and $h = 2^{-12}$ (right plot)

7 Concluding Remarks

In this chapter, it has been studied a procedure for solving fractional-time-space differential equations with the spectral fractional Laplacian on bounded domains. The spectral representation of the solution in terms of eigenfunctions of the usual Laplacian (truncated to a finite number of terms) has been exploited, with the time-dependent coefficients evaluated as a solution of some linear FDEs.

Since the proposed approach demands for the solution of a possible large number of FDEs, a GETD method has been used. This class of methods allows to directly evaluate the solution at any time without the need for implementing a considerably more expensive step-by-step computation. At the same time, it is possible to avoid the limitations in convergence order of classical methods for FDEs (although, for simplicity, we confined this investigation to a second-order GETD method, higher-order methods can be actually used).

Accurate error estimates have been obtained, thus to allow a fine-tuning of the numerical procedure in order to achieve some prescribed accuracy; the obtained error estimates have been verified by means of some numerical experiments.

Although the numerical simulations have been performed on very simple domains, for which an analytical representation of eigenvalues and eigenfunctions of the Laplacian is known, they have however provided a clear indication about the reliability of the proposed approach, and they have shown the good agreement between the actual error and error estimates. The next step, for future investigations, will be the application in combination with efficient procedures for the numerical approximation of eigenpairs on general domains (e.g., see [3, 6]).

This must be considered as a preliminary work about the numerical solution of time-fractional differential equations with the spectral Laplacian combining spectral representation of the solutions with GETD methods; the obtained results seem however to encourage the further continuation of research activities in this field.

Acknowledgments FVD has been supported by *REFIN* Project, grant number 812E4967. The work of RG has been partially supported by the INdAM-GNCS group and partially by the MIUR under the PRIN2017 project n. 2017E844SL.

References

1. Antil, H., Pfefferer, J., Rogovs, S.: Fractional operators with inhomogeneous boundary conditions: analysis, control, and discretization. Commun. Math. Sci. **16**(5), 1395–1426 (2018)
2. Beckner, W.: Inequalities in Fourier analysis. Ann. Math. **102**(1), 159–182 (1975)
3. Boffi, D.: Finite element approximation of eigenvalue problems. Acta Numer. **19**, 1–120 (2010)
4. Bonito, A., Borthagaray, J.P., Nochetto, R.H., Otárola, E., Salgado, A.J.: Numerical methods for fractional diffusion. Comput. Vis. Sci. **19**(5–6), 19–46 (2018)
5. Caffarelli, L.A., Stinga, P.R.: Fractional elliptic equations, Caccioppoli estimates and regularity. Ann. Inst. H. Poincaré Anal. Non Linéaire **33**(3), 767–807 (2016)
6. Carlson, M., Zheng, X., Sundar, H., Karniadakis, G.E., Kirby, R.M.: An open-source parallel code for computing the spectral fractional Laplacian on 3D complex geometry domains. Comput. Phys. Commun. **261**, 107695 (2021)
7. Chen, W., Li, Y., Ma, P.: The Fractional Laplacian. World Scientific Publishing, Hackensack (2020)
8. Cusimano, N., del Teso, F., Gerardo-Giorda, L., Pagnini, G.: Discretizations of the spectral fractional Laplacian on general domains with Dirichlet, Neumann, and Robin boundary conditions. SIAM J. Numer. Anal. **56**(3), 1243–1272 (2018)
9. Darve, E., D'Elia, M., Garrappa, R., Giusti, A., Rubio, N.L.: On the fractional Laplacian of variable order. Fract. Calc. Appl. Anal. **25**(1), 15–28 (2022)
10. D'Elia, M., Gulian, M., Olson, H., Karniadakis, G.E.: Towards a unified theory of fractional and nonlocal vector calculus. Fract. Calc. Appl. Anal. **24**(5), 1301–1355 (2021)
11. D'Elia, M., Gunzburger, M.: The fractional Laplacian operator on bounded domains as a special case of the nonlocal diffusion operator. Comput. Math. Appl. **66**(7), 1245–1260 (2013)
12. del Teso, F., Gómez-Castro, D., Vázquez, J.L.: Three representations of the fractional p-Laplacian: semigroup, extension and Balakrishnan formulas. Fract. Calc. Appl. Anal. **24**(4), 966–1002 (2021)
13. Diethelm, K.: The Analysis of Fractional Differential Equations. Lecture Notes in Mathematics, vol. 2004. Springer, Berlin (2010)
14. Diethelm, K., Ford, N.J., Freed, A.D.: Detailed error analysis for a fractional Adams method. Numer. Algorithms **36**(1), 31–52 (2004)
15. Dixon, J.: 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)
16. Garrappa, R.: Exponential integrators for time-fractional partial differential equations. Eur. Phys. J. Spec. Top. **222**(8), 1915–1927 (2013)
17. Garrappa, R.: A family of Adams exponential integrators for fractional linear systems. Comput. Math. Appl. **66**(5), 717–727 (2013)
18. Garrappa, R.: Numerical evaluation of two and three parameter Mittag-Leffler functions. SIAM J. Numer. Anal. **53**(3), 1350–1369 (2015)
19. Garrappa, R., Popolizio, M.: Generalized exponential time differencing methods for fractional order problems. Comput. Math. Appl. **62**(3), 876–890 (2011)
20. Garrappa, R., Popolizio, M.: On accurate product integration rules for linear fractional differential equations. J. Comput. Appl. Math. **235**(5), 1085–1097 (2011)
21. Garrappa, R., Popolizio, M.: Evaluation of generalized Mittag–Leffler functions on the real line. Adv. Comput. Math. **39**(1), 205–225 (2013)

22. Gorenflo, R., Kilbas, A.A., Mainardi, F., Rogosin, S.V.: Mittag–Leffler Functions, Related Topics and Applications. Springer Monographs in Mathematics. Springer, Berlin, Heidelberg (2020)
23. Grubb, G.: Regularity of spectral fractional Dirichlet and Neumann problems. Math. Nachr. **289**(7), 831–844 (2016)
24. Henrot, A.: Extremum Problems for Eigenvalues of Elliptic Operators. Frontiers in Mathematics. Birkhäuser, Basel (2006)
25. Hochbruck, M., Ostermann, A.: Exponential integrators. Acta Numer. **19**, 209–286 (2010)
26. Kwaśnicki, M.: Ten equivalent definitions of the fractional Laplace operator. Fract. Calc. Appl. Anal. **20**(1), 7–51 (2017)
27. Lischke, A., Pang, G., Gulian, M., Song, F., Glusa, C., Zheng, X., Mao, Z., Cai, W., Meerschaert, M.M., Ainsworth, M., Karniadakis, G.E.: What is the fractional Laplacian? A comparative review with new results. J. Comput. Phys. **404**, 109009, 62 (2020)
28. Lopez, L., Pellegrino, S.: A space-time discretization of a nonlinear peridynamic model on a 2D lamina. Comput. Math. Appl. **116**, 161–175 (2022). https://doi.org/10.1016/j.camwa.2021.07.004
29. Lopez, L., Pellegrino, S.F.: A spectral method with volume penalization for a nonlinear peridynamic model. Int. J. Numer. Methods Eng. **122**(3), 707–725 (2021)
30. Lubich, C.: Runge-Kutta theory for Volterra and Abel integral equations of the second kind. Math. Comp. **41**(163), 87–102 (1983)
31. Nochetto, R.H., Otárola, E., Salgado, A.J.: A PDE approach to space-time fractional parabolic problems. SIAM J. Numer. Anal. **54**(2), 848–873 (2016)
32. Otárola, E., Salgado, A.J.: Regularity of solutions to space-time fractional wave equations: a PDE approach. Fract. Calc. Appl. Anal. **21**(5), 1262–1293 (2018)
33. Pozrikidis, C.: The Fractional Laplacian. CRC Press, Boca Raton (2016)
34. Sarumi, I.O., Furati, K.M., Khaliq, A.Q.M., Mustapha, K.: Generalized exponential time differencing schemes for stiff fractional systems with nonsmooth source term. J. Sci. Comput. **86**(2), 23 (2021)
35. Silling, S.A.: Reformulation of elasticity theory for discontinuities and long-range forces. J. Mech. Phys. Solids **48**(1), 175–209 (2000)
36. Song, F., Xu, C., Karniadakis, G.E.: Computing fractional Laplacians on complex-geometry domains: algorithms and simulations. SIAM J. Sci. Comput. **39**(4), A1320–A1344 (2017)
37. Stinga, P.R.: User's guide to the fractional Laplacian and the method of semigroups. In: Handbook of Fractional Calculus with Applications, vol. 2, pp. 235–265. De Gruyter, Berlin (2019)
38. Stynes, M.: Too much regularity may force too much uniqueness. Fract. Calc. Appl. Anal. **19**(6), 1554–1562 (2016)
39. You, H., Yu, Y., Silling, S., D'Elia, M.: A data-driven peridynamic continuum model for upscaling molecular dynamics. Comput. Methods Appl. Mech. Eng. **389**, 114400 (2022)
40. Zhou, K., Du, Q.: Mathematical and numerical analysis of linear peridynamic models with nonlocal boundary conditions. SIAM J. Numer. Anal. **48**(5), 1759–1780 (2010)

Spectral Analysis of Matrices in B-Spline Galerkin Methods for Riesz Fractional Equations

Marco Donatelli, Carla Manni, Mariarosa Mazza, and Hendrik Speleers

Abstract Polynomial B-spline collocation discretizations for Riesz fractional diffusion equations over uniform meshes have recently appeared in the literature and a spectral study of the related coefficient matrices has been performed. Here we focus on the coefficient matrices obtained by the Galerkin approach. For an arbitrary polynomial degree p we show that, as for collocation, the resulting coefficient matrices possess a Toeplitz-like structure. The derivation of their spectral distribution is simpler compared to collocation, due to the symmetry of the coefficient matrices in this case and by leveraging the generalized locally Toeplitz theory. We see that, like for second-order differential problems, also in the fractional context the spectral distribution in the Galerkin formulation with B-splines of degree p is the same as in the collocation formulation with B-splines of degree $2p + 1$. As a consequence, the Galerkin matrices are poorly conditioned in both low and high frequencies similar to the collocation ones. Finally, we numerically observe that the approximation order of the Galerkin approach for smooth solutions does not depend on the fractional derivative order as for collocation and that it coincides with $p + 1$ as for non-fractional diffusion problems.

1 Introduction

More and more frequently, fractional differential operators appear in science and engineering problems. When used with respect to time, they allow for the modeling of long-time heavy-tail decay; see, e.g., [32] for applications of the time-fractional paradigm to tumor growth models. When used with respect to space, instead, fractional derivatives correspond to diffusion non-locality and are applied, e.g., in

M. Donatelli · M. Mazza (✉)
Department of Science and High Technology, University of Insubria, Como, Italy
e-mail: marco.donatelli@uninsubria.it; mariarosa.mazza@uninsubria.it

C. Manni · H. Speleers
Department of Mathematics, University of Rome Tor Vergata, Rome, Italy
e-mail: manni@mat.uniroma2.it; speleers@mat.uniroma2.it

A. Cardone et al. (eds.), *Fractional Differential Equations*,
Springer INdAM Series 50, https://doi.org/10.1007/978-981-19-7716-9_4

electrophysiology [1] to account for the heterogeneity present in a high number of scales of a biological excitable medium.

Since fractional differential operators are non-local, extending numerical methods designed for integer-order differential equations to equations that involve fractional derivatives is not a trivial task. Recently, the numerical study of fractional differential equations is rapidly increasing, and indeed many discretization alternatives can be found in the literature: finite differences [23, 31], finite elements [9, 17], and spectral methods [20, 35]. Although of relatively easy implementation, finite differences have limited accuracy, while finite elements and spectral methods can ensure higher accuracy; see [14, 15]. Other alternatives, which go in the direction of high-order approximations for fractional problems, are B-spline collocation methods; see [22, 27, 28, 34] for polynomial spline collocation methods and [26] for an application of fractional B-splines [33].

What brings together all previously mentioned methods is that the discretization matrices related to uniform grids inherit a Toeplitz-like structure from the space-invariant property of the underlying operators. Such aspect allows for a spectral analysis of the matrices through specialized tools and can be leveraged for the design of fast iterative solvers. Examples of this structure-based approach in case of low-order discretization methods for fractional problems are given in [7, 8, 16, 25].

A spectral study of high-order methods for fractional problems has recently appeared in the context of B-spline discretizations in [22]. Therein, we chose polynomial B-splines (which contrarily to their fractional counterpart have compact support and naturally fulfill boundary and/or initial conditions), and we spectrally studied the collocation discretization matrices obtained from the following fractional diffusion boundary value problem with absorbing boundary conditions:

$$\begin{cases} \frac{\mathrm{d}^\alpha u(x)}{\mathrm{d}|x|^\alpha} = s(x), & x \in \Omega, \\ u(x) = 0, & x \in \mathbb{R}\backslash\Omega, \end{cases} \tag{1}$$

where $\Omega = (0, 1)$, $\alpha \in (1, 2)$, and

$$\frac{\mathrm{d}^\alpha u(x)}{\mathrm{d}|x|^\alpha} := \frac{1}{2\cos(\frac{\pi\alpha}{2})} \left({}^{RL}_{0}D^\alpha_x u(x) + {}^{RL}_{x}D^\alpha_1 u(x) \right)$$

is the so-called Riesz fractional operator, while ${}^{RL}_{0}D^\alpha_x u(x)$ and ${}^{RL}_{x}D^\alpha_1 u(x)$ are the left-handed and right-handed Riemann-Liouville fractional derivatives of u (see Sect. 2.1 for their definition). The differences in terms of structure and conditioning of the resulting matrices when Caputo derivatives (see again Sect. 2.1) replace Riemann–Liouville ones were investigated in [21].

In the context of non-fractional diffusion problems, the Galerkin approach is often preferred over collocation because it performs better in terms of convergence, and its behavior is completely understood from the theoretical point of view.

Moreover, the Galerkin matrices for self-adjoint operators are symmetric, while the collocation ones are not. Here, we focus on the Galerkin discretization of problem (1) using B-splines, and we are mainly interested in the analysis of the properties of the resulting coefficient matrices.

Since these matrices are symmetric, they fit in the Generalized Locally Toeplitz (GLT) theory and this simplifies the derivation of their spectral distribution compared to the collocation approach. It turns out that, like for second-order differential problems, also in the fractional context the spectral distribution in the Galerkin formulation with B-splines of degree p is the same as in the collocation formulation with B-splines of degree $2p + 1$; see [5, Remark 4.4]. As a consequence, the Galerkin matrices are poorly conditioned in both low and high frequencies similar to the collocation ones, with a mitigated conditioning in the low frequencies and a deterioration in the high frequencies when compared to second-order problems [5]. Moreover, the Galerkin matrix properties remain substantially unchanged when replacing Riemann–Liouville derivatives with Caputo ones, which is in contrast to collocation where Riemann–Liouville and Caputo formulations were proved to differ by a rank-one correction that causes worse conditioning in high frequencies of the latter for fractional orders close to 1 [21].

Finally, we provide a numerical study of the approximation behavior of the B-spline Galerkin approach for an arbitrary degree p. We see that the approximation order of the Galerkin approach for smooth solutions does not depend on the fractional derivative order as for collocation and that it coincides with $p + 1$ as for non-fractional diffusion problems (see, e.g., [3]).

The outline of the chapter is the following. In Sect. 2, we introduce some preliminary tools on fractional derivatives, Toeplitz-like matrices, and polynomial B-splines. In Sect. 3, we formally write the B-spline Galerkin discretization matrices of (1) highlighting their structure, and in Sect. 4, we compute their spectral distribution. We end with some numerical results in Sect. 5 and some concluding remarks in Sect. 6.

2 Preliminaries

In this section, we first focus on the Riemann–Liouville and Caputo definitions of fractional derivatives (Sect. 2.1). Then, we summarize the essentials of Toeplitz and GLT sequences (Sect. 2.2), and finally, we introduce both B-splines and cardinal B-splines and some of their properties (Sect. 2.3).

2.1 Fractional Derivatives

A common definition of fractional derivatives is given by the Riemann–Liouville formula. For a given function with absolutely continuous first derivative on $[a, b]$,

the left-handed and right-handed Riemann–Liouville fractional derivatives of order α are defined by

$$
\begin{aligned}
{}^{RL}_{a}D^{\alpha}_{x}u(x) &:= \frac{1}{\Gamma(m-\alpha)}\frac{\mathrm{d}^m}{\mathrm{d}x^m}\int_a^x (x-y)^{m-\alpha-1}u(y)\,\mathrm{d}y,\\
{}^{RL}_{x}D^{\alpha}_{b}u(x) &:= \frac{(-1)^m}{\Gamma(m-\alpha)}\frac{\mathrm{d}^m}{\mathrm{d}x^m}\int_x^b (y-x)^{m-\alpha-1}u(y)\,\mathrm{d}y,
\end{aligned}
$$

with m the integer such that $m-1 \le \alpha < m$ and $\Gamma(\cdot)$ the Euler gamma function.

In case of functions with absolutely integrable m-th derivative on $[a,b]$, another common definition of fractional derivatives was proposed by Caputo:

$$
\begin{aligned}
{}^{C}_{a}D^{\alpha}_{x}u(x) &:= \frac{1}{\Gamma(m-\alpha)}\int_a^x (x-y)^{m-\alpha-1}u^{(m)}(y)\,\mathrm{d}y,\\
{}^{C}_{x}D^{\alpha}_{b}u(x) &:= \frac{(-1)^m}{\Gamma(m-\alpha)}\int_x^b (y-x)^{m-\alpha-1}u^{(m)}(y)\,\mathrm{d}y,
\end{aligned}
$$

with $u^{(m)}$ the m-th derivative of u.

The Riemann–Liouville derivatives relate to the Caputo ones as follows:

$$
\begin{aligned}
{}^{RL}_{a}D^{\alpha}_{x}u(x) &= {}^{C}_{a}D^{\alpha}_{x}u(x) + \sum_{k=0}^{m-1}\frac{(x-a)^{k-\alpha}}{\Gamma(k-\alpha+1)}u^{(k)}(a^+),\\
{}^{RL}_{x}D^{\alpha}_{b}u(x) &= {}^{C}_{x}D^{\alpha}_{b}u(x) + \sum_{k=0}^{m-1}(-1)^k\frac{(b-x)^{k-\alpha}}{\Gamma(k-\alpha+1)}u^{(k)}(b^-)\,,
\end{aligned}
\tag{2}
$$

and so the two coincide if u satisfies homogeneous conditions, i.e., $u^{(k)}(a^+) = u^{(k)}(b^-) = 0$ for $k = 0,\ldots,m-1$.

In case of the whole real axis (and more in general of unbounded domains), the Riemann–Liouville and Caputo definitions coincide and can be expressed in the form

$$
\begin{aligned}
{}_{-\infty}D^{\alpha}_{x}u(x) &:= \frac{1}{\Gamma(m-\alpha)}\frac{\mathrm{d}^{\mathrm{m}}}{\mathrm{d}x^m}\int_{-\infty}^x (x-y)^{m-\alpha-1}u(y)\,\mathrm{d}y,\\
{}_{x}D^{\alpha}_{+\infty}u(x) &:= \frac{(-1)^m}{\Gamma(m-\alpha)}\frac{\mathrm{d}^m}{\mathrm{d}x^m}\int_x^{+\infty} (y-x)^{m-\alpha-1}u(y)\,\mathrm{d}y.
\end{aligned}
\tag{3}
$$

Throughout the chapter, whenever we write ${}_{-\infty}D^{\alpha}_{x}\,u(\xi)$ or ${}^{RL}_{a}D^{\alpha}_{x}\,u(\xi)$ for a fixed ξ, we mean

$$
{}_{-\infty}D^{\alpha}_{x}\,u(x) \;\text{ or }\; {}^{RL}_{a}D^{\alpha}_{x}\,u(x), \;\text{ where } \; x=\xi,
$$

respectively. Analogous meaning will be given to ${}_xD^{\alpha}_{+\infty}\,u(\xi)$ and ${}^{RL}_{x}D^{\alpha}_{b}\,u(\xi)$.

2.2 Spectral Tools

We begin with the formal definition of spectral distribution in the sense of the eigenvalues and singular values for a general matrix sequence.

Definition 1 Let $f : G \to \mathbb{C}$ be a measurable function, defined on a measurable set $G \subset \mathbb{R}^k$ with $k \geq 1$ and Lebesgue measure $0 < m_k(G) < \infty$. Let $C_0(\mathbb{K})$ be the set of continuous functions with compact support over $\mathbb{K} \in \{\mathbb{C}, \mathbb{R}^+_0\}$, and let $\{A_n\}_n$ be a sequence of matrices of size n with eigenvalues $\lambda_j(A_n)$, $j = 1, \ldots, n$, and singular values $\sigma_j(A_n)$, $j = 1, \ldots, n$.

- $\{A_n\}_n$ is *distributed as the pair* (f, G) *in the sense of the eigenvalues,* in symbols

$$\{A_n\}_n \sim_\lambda (f, G),$$

  if the following limit relation holds for all $F \in C_0(\mathbb{C})$:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{j=1}^{n} F(\lambda_j(A_n)) = \frac{1}{m_k(G)} \int_G F(f(t))\, \mathrm{d}t. \tag{4}$$

  In this case, we say that f is the (spectral) symbol of the matrix sequence $\{A_n\}_n$.
- $\{A_n\}_n$ is *distributed as the pair* (f, G) *in the sense of the singular values,* in symbols

$$\{A_n\}_n \sim_\sigma (f, G),$$

  if the following limit relation holds for all $F \in C_0(\mathbb{R}^+_0)$:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{j=1}^{n} F(\sigma_j(A_n)) = \frac{1}{m_k(G)} \int_G F(|f(t)|)\, \mathrm{d}t. \tag{5}$$

  In this case, we say that f is the singular value symbol of the matrix sequence $\{A_n\}_n$.

Throughout the chapter, when it is not of crucial importance to know which is the domain of f, we replace the notation $\{A_n\}_n \sim_\lambda (f, G)$ with $\{A_n\}_n \sim_\lambda f$.

Remark 1 When f is continuous, an informal interpretation of the limit relation (4) (respectively, (5)) is that when the matrix size is sufficiently large, the eigenvalues (respectively, singular values) of A_n can be approximated by a sampling of f

(respectively, $|f|$) on a uniform grid of the domain G, possibly up to few outliers whose number is $o(n)$.

In the case where the singular value symbol is the zero function, we introduce the following definition.

Definition 2 We say that $\{A_n\}_n$ is a zero-distributed matrix sequence if $\{A_n\}_n \sim_\sigma (0, G)$.

The following result provides an important characterization of zero-distributed sequences (see [11]).

Proposition 1 *Let $\{A_n\}_n$ be a matrix sequence with A_n of size n. Then, $\{A_n\}_n \sim_\sigma 0$ if and only if there exist two matrix sequences $\{R_n\}_n$ and $\{E_n\}_n$ such that $A_n = R_n + E_n$, and*

$$\lim_{n\to\infty} \frac{\operatorname{rank}(R_n)}{n} = 0, \qquad \lim_{n\to\infty} \|E_n\| = 0,$$

with $\|\cdot\|$ the spectral norm.

We now recall the definition of Toeplitz sequences generated by univariate functions in $L^1([-\pi,\pi])$.

Definition 3 Let $f \in L^1([-\pi,\pi])$, and let f_k be its Fourier coefficients,

$$f_k := \frac{1}{2\pi}\int_{-\pi}^{\pi} f(\theta)\mathrm{e}^{-\mathrm{i}(k\theta)}\,\mathrm{d}\theta, \quad k \in \mathbb{Z}.$$

The n-th (unilevel) Toeplitz matrix associated with f is the $n \times n$ matrix defined by

$$T_n(f) := \begin{bmatrix} f_0 & f_{-1} & \cdots & \cdots & f_{-(n-1)} \\ f_1 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & f_{-1} \\ f_{n-1} & \cdots & \cdots & f_1 & f_0 \end{bmatrix} \in \mathbb{C}^{n\times n}.$$

The matrix sequence $\{T_n(f)\}_n$ is called the Toeplitz sequence generated by f.

For Hermitian Toeplitz sequences, the following theorem proved by Szegő, Tyrtyshnikov et al. holds (see, e.g., [13]).

Theorem 1 *Let $f \in L^1([-\pi,\pi])$ be a real-valued function. Then,*

$$\{T_n(f)\}_n \sim_\lambda (f, [-\pi,\pi]).$$

Both zero-distributed matrix sequences and Toeplitz sequences belong to a larger class of matrix sequences known as *Generalized Locally Toeplitz (GLT) class*. In short, the GLT class is an algebra virtually containing any sequence of matrices coming from "reasonable" approximations on uniform grids by local discretization methods for differential problems (finite differences, finite elements, isogeometric analysis, etc.). Without going into the details of the GLT algebra, we list some interesting properties of GLT sequences in the following (see [11]).

Throughout the chapter, we use the notation

$$\{A_n\}_n \sim_{\text{GLT}} \kappa$$

to say that $\{A_n\}_n$ is a GLT sequence with symbol $\kappa : [0, 1] \times [-\pi, \pi] \to \mathbb{C}$.

GLT1 Let $\{A_n\}_n \sim_{\text{GLT}} \kappa$ with $\kappa : G \to \mathbb{C}$ and $G = [0, 1] \times [-\pi, \pi]$, and then $\{A_n\}_n \sim_\sigma (\kappa, G)$ according to Definition 1 with $k = 2$. If the matrices A_n are Hermitian, then it holds also $\{A_n\}_n \sim_\lambda (\kappa, G)$.

GLT2 The set of GLT sequences forms a $*$-algebra, i.e., it is closed under linear combinations, products, and conjugation. Moreover, in case the symbol is non-zero a.e., it is closed also under inversion. In formulae, let $\{A_n\}_n \sim_{\text{GLT}} \kappa_1$ and $\{B_n\}_n \sim_{\text{GLT}} \kappa_2$, then

- $\{\alpha A_n + \beta B_n\}_n \sim_{\text{GLT}} \alpha\kappa_1 + \beta\kappa_2, \quad \alpha, \beta \in \mathbb{C}$
- $\{A_n B_n\}_n \sim_{\text{GLT}} \kappa_1\kappa_2$
- $\{A_n^*\}_n \sim_{\text{GLT}} \bar{\kappa}_1$
- $\{A_n^{-1}\}_n \sim_{\text{GLT}} \kappa_1^{-1}$ provided that κ_1 is non-zero a.e.

GLT3 Every Toeplitz sequence $\{T_n(f)\}_n$ generated by a function $f \in L^1([-\pi, \pi])$ is $\{T_n(f)\}_n \sim_{\text{GLT}} f$, with the specifications reported in item **GLT1**.

GLT4 Let $G = [0, 1] \times [-\pi, \pi]$, then

$$\{A_n\}_n \sim_\sigma (0, G) \iff \{A_n\}_n \sim_{\text{GLT}} 0.$$

In particular, thanks to Proposition 1, any sequence in which the rank divided by the size tends to zero as the matrix size tends to infinity (rank-correction) and any sequence with infinitesimal spectral norm (norm-correction) has symbol 0.

2.3 B-Splines and Cardinal B-Splines

In this subsection, we recall the definition of both B-splines and cardinal B-splines together with their main properties and with a focus on how they behave under fractional derivation.

B-Splines For $p \geq 0$ and $n \geq 1$, consider the uniform knot sequence

$$\xi_1 = \cdots = \xi_{p+1} := 0 < \xi_{p+2} < \cdots < \xi_{p+n} < 1 =: \xi_{p+n+1} = \cdots = \xi_{2p+n+1},$$

where

$$\xi_{i+p+1} := \frac{i}{n}, \quad i = 0, \ldots, n.$$

This knot sequence allows us to define $n + p$ B-splines of degree p.

Definition 4 The B-splines of degree p over a uniform mesh of $[0, 1]$, consisting of n intervals, are denoted by

$$N_i^p : [0, 1] \to \mathbb{R}, \quad i = 1, \ldots, n + p,$$

and defined recursively as follows: for $1 \leq i \leq n + 2p$,

$$N_i^0(x) := \begin{cases} 1, & x \in [\xi_i, \xi_{i+1}), \\ 0, & \text{otherwise;} \end{cases}$$

for $1 \leq k \leq p$ and $1 \leq i \leq n + 2p - k$,

$$N_i^k(x) := \frac{x - \xi_i}{\xi_{i+k} - \xi_i} N_i^{k-1}(x) + \frac{\xi_{i+k+1} - x}{\xi_{i+k+1} - \xi_{i+1}} N_{i+1}^{k-1}(x),$$

where a fraction with zero denominator is assumed to be zero.

It is well known that the B-splines N_i^p, $i = 1, \ldots, n + p$, are linearly independent and that they have local support and C^{p-1} smoothness on $(0, 1)$. Finally, they possess simple interpolation properties at the boundary and form a non-negative partition of unity (see, e.g., [4, 18] for these and other B-spline properties). Moreover, concerning fractional derivatives of a B-spline, by recalling (2), the following property holds:

$$\begin{aligned} {}^{RL}_{0}D_x^\alpha N_{i+1}^p &= {}^{C}_{0}D_x^\alpha N_{i+1}^p, \\ {}^{RL}_{x}D_1^\alpha N_{i+1}^p &= {}^{C}_{x}D_1^\alpha N_{i+1}^p, \quad i = m, \ldots, n + p - m - 1. \end{aligned} \tag{6}$$

Cardinal B-Splines The central B-splines N_i^p, $i = p + 1, \ldots, n$, are uniformly shifted and scaled versions of a single shape function, the so-called cardinal B-spline $\phi_p : \mathbb{R} \to \mathbb{R}$,

$$\phi_0(t) := \begin{cases} 1, & t \in [0, 1), \\ 0, & \text{otherwise,} \end{cases}$$

and

$$\phi_p(t) := \frac{t}{p}\phi_{p-1}(t) + \frac{p+1-t}{p}\phi_{p-1}(t-1), \quad p \geq 1.$$

More precisely, we have

$$N_i^p(x) = \phi_p(nx - i + p + 1), \quad i = p+1, \ldots, n,$$

and

$$\left(N_i^p(x)\right)' = n\phi_p'(nx - i + p + 1), \quad i = p+1, \ldots, n. \tag{7}$$

Similar to the B-splines, the cardinal B-spline has local support (on $[0, p+1]$) and C^{p-1} smoothness (over $\mathbb{R}$). Among other properties, we mention:

- Differentiation: its derivative can be easily expressed as

$$\phi_p'(t) = \phi_{p-1}(t) - \phi_{p-1}(t-1), \quad p \geq 1.$$

- Inner-product: given integers $0 \leq r_1 \leq p_1$ and $0 \leq r_2 \leq p_2$, we have

$$\begin{aligned}\int_{\mathbb{R}} \phi_{p_1}^{(r_1)}(t)\phi_{p_2}^{(r_2)}(t+\tau)\,\mathrm{d}t &= (-1)^{r_1}\phi_{p_1+p_2+1}^{(r_1+r_2)}(p_1+1+\tau) \\ &= (-1)^{r_2}\phi_{p_1+p_2+1}^{(r_1+r_2)}(p_2+1-\tau).\end{aligned}$$

For more properties, we refer the reader to [18] and the references therein. Regarding fractional derivatives of the cardinal B-spline, the following explicit formula for the left-handed Caputo derivative holds (see [27]):

$${}_0^C D_t^\alpha \phi_p(t) = \frac{1}{\Gamma(p-\alpha+1)} \sum_{j=0}^{p+1} (-1)^j \binom{p+1}{j} (t-j)_+^{p-\alpha}, \quad 0 < \alpha < p,$$

where $(\cdot)_+^q$ is the truncated power function of degree q. Moreover, as shown in [22], for $0 \leq \alpha_1 < p_1$ and $0 \leq \alpha_2 < p_2$, we have

$$\begin{aligned}\int_{\mathbb{R}} {}_{-\infty}D_x^{\alpha_1}\phi_{p_1}(x)\,{}_xD_{+\infty}^{\alpha_2}\phi_{p_2}(x+k)\,\mathrm{d}x &= {}_{-\infty}D_x^{\alpha_1+\alpha_2}\phi_{p_1+p_2+1}(p_2+1-k), \\ \int_{\mathbb{R}} {}_xD_{+\infty}^{\alpha_1}\phi_{p_1}(x)\,{}_{-\infty}D_x^{\alpha_2}\phi_{p_2}(x+k)\,\mathrm{d}x &= {}_xD_{+\infty}^{\alpha_1+\alpha_2}\phi_{p_1+p_2+1}(p_2+1-k),\end{aligned} \tag{8}$$

where the fractional derivatives on half-axes are defined as in (3).

3 B-Spline Galerkin Discretization of the Fractional Riesz Operator

Let us denote by $\mathcal{W}$, with $N := \dim(\mathcal{W})$, a finite dimensional vector space of sufficiently smooth functions defined on the closure of Ω and vanishing on its boundary. Once we switch to the weak form of (1), we can apply the Galerkin approach by looking for a solution $u_{\mathcal{W}}$ in $\mathcal{W}$ such that

$$\mathrm{a}(u_{\mathcal{W}}, v) = \mathrm{f}(v), \quad v \in \mathcal{W}, \tag{9}$$

where

$$\begin{aligned}
\mathrm{a}(u, v) &:= \frac{1}{2\cos(\frac{\pi\alpha}{2})}\left(\int_0^1 v(x)\, {}^{RL}_{0}D^{\alpha}_{x}u(x)\,\mathrm{d}x + \int_0^1 v(x)\, {}^{RL}_{x}D^{\alpha}_{1}u(x)\,\mathrm{d}x\right)\\
&= \frac{1}{2\cos(\frac{\pi\alpha}{2})}\left(\int_0^1 v(x)\, {}^{RL}_{0}D^{\frac{\alpha}{2}}_{x}\, {}^{RL}_{0}D^{\frac{\alpha}{2}}_{x}u(x)\,\mathrm{d}x + \int_0^1 v(x)\, {}^{RL}_{x}D^{\frac{\alpha}{2}}_{1}\, {}^{RL}_{x}D^{\frac{\alpha}{2}}_{1}u(x)\,\mathrm{d}x\right)\\
&= \frac{1}{2\cos(\frac{\pi\alpha}{2})}\left(\int_0^1 {}^{RL}_{x}D^{\frac{\alpha}{2}}_{1}v(x)\, {}^{RL}_{0}D^{\frac{\alpha}{2}}_{x}u(x)\,\mathrm{d}x + \int_0^1 {}^{RL}_{0}D^{\frac{\alpha}{2}}_{x}v(x)\, {}^{RL}_{x}D^{\frac{\alpha}{2}}_{1}u(x)\,\mathrm{d}x\right),
\end{aligned} \tag{10}$$

and

$$\mathrm{f}(v) := \int_0^1 s(x)v(x)\,\mathrm{d}x.$$

The second equality in (10) is a consequence of the identities

$$\begin{aligned}
{}^{RL}_{0}D^{\nu}_{x}({}^{RL}_{0}D^{\mu}_{x}u(x)) &= {}^{RL}_{0}D^{\mu}_{x}({}^{RL}_{0}D^{\nu}_{x}u(x)) = {}^{RL}_{0}D^{\nu+\mu}_{x}u(x), \text{ when } u^{(j)}(0) = 0,\\
{}^{RL}_{x}D^{\nu}_{1}({}^{RL}_{x}D^{\mu}_{1}u(x)) &= {}^{RL}_{x}D^{\mu}_{1}({}^{RL}_{x}D^{\nu}_{1}u(x)) = {}^{RL}_{x}D^{\nu+\mu}_{1}u(x), \text{ when } u^{(j)}(1) = 0,
\end{aligned}$$

which hold for $j = 0, 1, \ldots, r-1$ and where $n-1 \le \nu < n$, $m-1 \le \mu < m$ with $n, m \in \mathbb{N}$, and $r = \max\{n, m\}$ (see [29, formula (2.127)]). The third equality in (10) is obtained by applying integration by parts (see [30, formula (2.64)]).

If $\{\varphi_j : j = 1, \ldots, N\}$ is a basis of $\mathcal{W}$, we can write $u_{\mathcal{W}}(x) = \sum_{j=1}^{N} u_j\varphi_j(x)$ and the computation of $u_{\mathcal{W}}$ reduces to solving the following symmetric linear system:

$$A_{\mathrm{gal}}\mathbf{u} = \mathbf{b}_{\mathrm{gal}},$$

with

$$A_{\mathrm{gal}} := [\mathrm{a}(\varphi_j, \varphi_i)]_{i,j=1}^{N}, \quad \mathbf{b}_{\mathrm{gal}} := [\mathrm{f}(\varphi_i)]_{i=1}^{N}.$$

In this chapter, we choose $\mathcal{W}$ as the space of splines of degree p that vanish at the boundary. More precisely, we take

$$\mathcal{W} = \mathbb{S}_n^p := \operatorname{span}\{N_j^p : j = 2, \dots, n+p-1\}.$$

Thus, Eq. (9) translates in a linear system whose coefficient matrix has the form

$$A_n^{p,\alpha} := \frac{1}{2\cos(\frac{\pi\alpha}{2})}(A_n^L + A_n^R),$$

with

$$A_n^L := \left[\int_0^1 {}^{RL}_{x}D_1^{\frac{\alpha}{2}} N_{i+1}^p(x)\, {}^{RL}_{0}D_x^{\frac{\alpha}{2}} N_{j+1}^p(x)\,\mathrm{d}x\right]_{i,j=1}^{n+p-2},$$

$$A_n^R := \left[\int_0^1 {}^{RL}_{0}D_x^{\frac{\alpha}{2}} N_{i+1}^p(x)\, {}^{RL}_{x}D_1^{\frac{\alpha}{2}} N_{j+1}^p(x)\,\mathrm{d}x\right]_{i,j=1}^{n+p-2}.$$

Remark 2 Since $\frac{\alpha}{2} \in (0,1)$ and due to (6), for each $i = 1, \dots, n+p-2$, it holds

$${}^{RL}_{0}D_x^{\frac{\alpha}{2}} N_{i+1}^p(x) = {}^{C}_{0}D_x^{\frac{\alpha}{2}} N_{i+1}^p(x) = \frac{1}{\Gamma(1-\frac{\alpha}{2})}\int_0^x (x-y)^{-\frac{\alpha}{2}} (N_{i+1}^p(y))'\,\mathrm{d}y,$$

$${}^{RL}_{x}D_1^{\frac{\alpha}{2}} N_{i+1}^p(x) = {}^{C}_{x}D_1^{\frac{\alpha}{2}} N_{i+1}^p(x) = \frac{-1}{\Gamma(1-\frac{\alpha}{2})}\int_x^1 (y-x)^{-\frac{\alpha}{2}} (N_{i+1}^p(y))'\,\mathrm{d}y.$$

Therefore, A_n^L and A_n^R remain unchanged if we replace Riemann–Liouville derivatives with Caputo ones. Because of this, from now on, we just omit the superscripts "RL" or "C."

Thanks to (7), the local support property, and Remark 2, for $i = p, \dots, n-1$, we have

$${}_{0}D_x^{\frac{\alpha}{2}} N_{i+1}^p(x) = n^{\frac{\alpha}{2}}\, {}_{0}D_{nx}^{\frac{\alpha}{2}} \phi_p(nx-i+p) = n^{\frac{\alpha}{2}}\, {}_{-\infty}D_{nx}^{\frac{\alpha}{2}} \phi_p(nx-i+p),$$

$${}_{x}D_1^{\frac{\alpha}{2}} N_{i+1}^p(x) = n^{\frac{\alpha}{2}}\, {}_{nx}D_n^{\frac{\alpha}{2}} \phi_p(nx-i+p) = n^{\frac{\alpha}{2}}\, {}_{nx}D_{+\infty}^{\frac{\alpha}{2}} \phi_p(nx-i+p).$$

Noticing that, for $i = p, \dots, n-1$,

$${}_{-\infty}D_{nx}^{\frac{\alpha}{2}} \phi_p(nx-i+p) = 0, \quad \text{for } x \le 0,$$

$${}_{nx}D_{+\infty}^{\frac{\alpha}{2}} \phi_p(nx-i+p) = 0, \quad \text{for } x \ge 1,$$

we get that, for $i, j = p, \dots n-1$, the entries of A_n^L are given by

$$\begin{aligned}(A_n^L)_{i,j} &= n^\alpha \int_0^1 {}_{nx}D_n^{\frac{\alpha}{2}}\, \phi_p(nx-i+p)\, {}_0D_{nx}^{\frac{\alpha}{2}}\, \phi_p(nx-j+p)\,\mathrm{d}x \\ &= n^\alpha \int_0^1 {}_{nx}D_{+\infty}^{\frac{\alpha}{2}}\, \phi_p(nx-i+p)\, {}_{-\infty}D_{nx}^{\frac{\alpha}{2}}\, \phi_p(nx-j+p)\,\mathrm{d}x \\ &= n^\alpha \int_{\mathbb{R}} {}_{nx}D_{+\infty}^{\frac{\alpha}{2}}\, \phi_p(nx-i+p)\, {}_{-\infty}D_{nx}^{\frac{\alpha}{2}}\, \phi_p(nx-j+p)\,\mathrm{d}x.\end{aligned}$$

By using the change of variable $t = nx - j + p$ and thanks to (8), we finally obtain

$$\begin{aligned}(A_n^L)_{i,j} &= n^{\alpha-1} \int_{\mathbb{R}} {}_{nx}D_{+\infty}^{\frac{\alpha}{2}}\, \phi_p(t+j-i)\, {}_{-\infty}D_{nx}^{\frac{\alpha}{2}}\, \phi_p(t)\,\mathrm{d}t \\ &= n^{\alpha-1}\, {}_{-\infty}D_{nx}^{\alpha}\, \phi_{2p+1}(p+1-(j-i)) = n^{\alpha-1}\, {}_0D_{nx}^{\alpha}\, \phi_{2p+1}(p+1+i-j).\end{aligned}$$

Similarly, for $i, j = p, \dots n-1$, we obtain

$$(A_n^R)_{i,j} = n^{\alpha-1}\, {}_{nx}D_n^{\alpha}\, \phi_{2p+1}(p+1+i-j).$$

Therefore, both A_n^L and A_n^R show a Toeplitz plus rank-correction structure and can be written as follows:

$$A_n^L = n^{\alpha-1}(T_n^L + R_n^L), \quad A_n^R = n^{\alpha-1}(T_n^R + R_n^R),$$

with

$$\begin{aligned}T_n^L &:= \left[{}_0D_{nx}^{\alpha}\, \phi_{2p+1}(p+1+i-j)\right]_{i,j=1}^{n+p-2}, \\ T_n^R &:= \left[{}_{nx}D_n^{\alpha}\, \phi_{2p+1}(p+1+i-j)\right]_{i,j=1}^{n+p-2},\end{aligned}$$

and the rank of both R_n^L and R_n^R depends only on p. The coefficient matrix $A_n^{p,\alpha}$ inherits the Toeplitz plus rank-correction structure of A_n^L and A_n^R and becomes

$$A_n^{p,\alpha} = \frac{1}{2\cos(\frac{\pi\alpha}{2})}(A_n^L + A_n^R) = n^{\alpha-1}(T_n^{p,\alpha} + R_n^{p,\alpha}),$$

with

$$T_n^{p,\alpha} := \frac{1}{2\cos(\frac{\pi\alpha}{2})}(T_n^L + T_n^R), \quad R_n^{p,\alpha} := \frac{1}{2\cos(\frac{\pi\alpha}{2})}(R_n^L + R_n^R).$$

In Sect. 4, we will show that the symbol of $\{n^{1-\alpha}A_n^{p,\alpha}\}_n$ coincides with the one of $\{T_n^{p,\alpha}\}_n$.

4 Spectral Symbol of $\{n^{1-\alpha}A_n^{p,\alpha}\}_n$ and Its Properties

This section is devoted to the computation and the study of the symbol of the coefficient matrix sequence $\{n^{1-\alpha}A_n^{p,\alpha}\}_n$. As we have already anticipated, it turns out that the symbol of $\{n^{1-\alpha}A_n^{p,\alpha}\}_n$ coincides with the symbol of the Toeplitz part $\{T_n^{p,\alpha}\}_n$. In order to give evidence of this, we start with a proposition that provides an explicit expression for the symbol of $\{T_n^{p,\alpha}\}_n$.

Proposition 2 *We have* $T_n^{p,\alpha} = T_{n+p-2}(f^{2p+1,\alpha})$ *with*

$$f^{2p+1,\alpha}(\theta) = \sum_{l\in\mathbb{Z}} |\theta + 2l\pi|^{\alpha} \left(\frac{\sin(\theta/2 + l\pi)}{\theta/2 + l\pi}\right)^{2p+2}$$
$$= (2 - 2\cos(\theta))^{p+1} \sum_{l\in\mathbb{Z}} \frac{1}{|\theta + 2l\pi|^{2p+2-\alpha}},$$

and

$$\{T_n^{p,\alpha}\}_n = \{T_{n+p-2}(f^{2p+1,\alpha})\} \sim_\lambda (f^{2p+1,\alpha}, [-\pi, \pi]). \tag{11}$$

Proof The equality $T_n^{p,\alpha} = T_{n+p-2}(f^{2p+1,\alpha})$ and the expression of $f^{2p+1,\alpha}$ can be easily obtained from [22, Theorem 4] and from elementary computations. Equation (11) immediately follows by observing that $f^{2p+1,\alpha}$ is a real-valued function and by applying Theorem 1. □

We are now in the position to discuss the spectral distribution of $\{n^{1-\alpha}A_n^{p,\alpha}\}_n$.

Theorem 2 *Given* $\{n^{1-\alpha}A_n^{p,\alpha}\}_n$, *it holds*

$$\{n^{1-\alpha}A_n^{p,\alpha}\}_n \sim_\lambda (f^{2p+1,\alpha}, [-\pi, \pi]). \tag{12}$$

Proof Due to **GLT3**, we have $\{T_{n+p-2}(f^{2p+1,\alpha})\}_n \sim_{\text{GLT}} f^{2p+1,\alpha}$. On the other hand, by **GLT4**, we have $\{R_n^{p,\alpha}\}_n \sim_{\text{GLT}} 0$. This is because the rank of $R_n^{p,\alpha}$ depends only on p, so

$$\lim_{n\to\infty} \frac{\text{rank}(R_n^{p,\alpha})}{n + p - 2} = 0.$$

By using **GLT2**, we can then conclude that $\{n^{1-\alpha}A_n^{p,\alpha} = T_{n+p-2}(f^{2p+1,\alpha}) + R_n^{p,\alpha}\}_n$ is a GLT sequence with symbol $f^{2p+1,\alpha}$. As a consequence of **GLT1**, this implies $\{n^{1-\alpha}A_n^{p,\alpha}\} \sim_\sigma f^{2p+1,\alpha}$ and (12), recalling that $n^{1-\alpha}A_n^{p,\alpha}$ is symmetric. □

Remark 3 Theorem 2 is the analog of [22, Theorem 6], where the Galerkin approach replaces collocation. We stress that, due to the symmetry of the coefficient matrices $A_n^{p,\alpha}$, in this case we do not need to bound their trace and spectral norms as

done in [22], but we can more easily retrieve their spectral distribution by leveraging the GLT theory.

From Theorem 2, we see that, like for second-order differential problems (see [5, 10]), also in the fractional context the spectral distribution in the Galerkin formulation with B-splines of degree p is the same as in the collocation formulation with B-splines of degree $2p+1$. Therefore, as discussed in [22], $f^{2p+1,\alpha}$ is equipped with the following properties:

- It only vanishes at 0 with order α.
- It presents an exponential decay to zero at π for increasing p that becomes faster as α approaches 1.

In other words, we expect that the Galerkin matrices are poorly conditioned in both low and high frequencies similar to the collocation ones.

5 Numerical Results

In this section, we first check the approximation properties of the considered B-spline Galerkin discretization of problem (1), and then we numerically verify that relations (11) and (12) hold.

In all the numerical experiments, the fractional derivatives that define the entries of the coefficient matrix $A_n^{p,\alpha}$ have been computed using the Gauss–Jacobi-type quadrature rules introduced in [24]. More specifically, in order to compute the Riemann–Liouville derivative of order $\alpha/2$ of a piecewise polynomial, we integrate its first derivative with an exact Gauss–Jacobi quadrature. The external integral that defines the entries of $A_n^{p,\alpha}$ has instead been approximated by means of a Gauss–Legendre quadrature rule with $n + p - 2$ points.

In Tables 1, 2, and 3, we fix the source function s such that the true solution of (1) is given by the following functions:

- $u(x) = x^3(1-x)^3$
- $u(x) = \sin(\pi x^2)$
- $u(x) = x^{2+\alpha}(1-x)^{2+\alpha}$

respectively. Note that the first two are smooth functions allover [0, 1], while the last one is non-smooth at the boundary. Then, by doubling n repeatedly, we show the corresponding infinity-norm errors and convergence orders for various p and α. The infinity-norm of the error is computed by taking the maximum value of the error sampled in 1024 points uniformly distributed over [0, 1].

From the results in Tables 1, 2, and 3, we notice that, contrarily to what happens in case of collocation where we observed a dependency of the approximation order on α, here the approximation order depends only on p and seems to substantially coincide with $p+1$, for smooth solutions. This is in agreement with the approximation results known for standard non-fractional diffusion problems (see,

Table 1 Errors and predicted convergence orders of the proposed B-spline Galerkin method for problem (1) when $u(x) = x^3(1-x)^3$

		$p=2$		$p=3$		$p=4$		$p=5$	
α	n	Error	Order	Error	Order	Error	Order	Error	Order
1.2	4	5.7797e-04		1.9472e-04		5.0008e-05		7.3202e-06	
	8	7.5241e-05	2.94	1.2086e-05	4.01	1.9868e-06	4.65	1.0151e-07	6.17
	16	1.1591e-05	2.70	1.0295e-06	3.55	7.2372e-08	4.78	1.5752e-09	6.01
	32	1.8240e-06	2.67	7.8471e-08	3.71	2.4589e-09	4.88	2.6991e-11	5.87
	64	2.6949e-07	2.76	5.7119e-09	3.78	8.0346e-11	4.94	4.8414e-11	–
1.5	4	5.9094e-04		1.8226e-04		5.1836e-05		7.4826e-06	
	8	6.7314e-05	3.13	1.1941e-05	3.93	2.0003e-06	4.70	1.0210e-07	6.20
	16	1.1397e-05	2.56	1.0543e-06	3.50	7.3343e-08	4.77	1.5817e-09	6.01
	32	1.5348e-06	2.89	7.8517e-08	3.75	2.4850e-09	4.88	2.5439e-11	5.96
	64	2.0041e-07	2.94	5.7740e-09	3.77	8.1240e-11	4.93	2.0949e-11	–
1.8	4	6.2731e-04		1.6842e-04		5.3106e-05		7.5828e-06	
	8	6.3165e-05	3.31	1.1590e-05	3.86	2.0119e-06	4.72	1.0621e-07	6.16
	16	1.0809e-05	2.55	1.0337e-06	3.49	7.5156e-08	4.74	1.6588e-09	6.00
	32	1.5860e-06	2.77	7.7071e-08	3.75	2.5341e-09	4.89	2.5804e-11	6.01
	64	3.4303e-07	2.21	5.6814e-09	3.76	8.3106e-11	4.93	1.5379e-11	–

Table 2 Errors and predicted convergence orders of the proposed B-spline Galerkin method for problem (1) when $u(x) = \sin(\pi x^2)$

		$p=2$		$p=3$		$p=4$		$p=5$	
α	n	Error	Order	Error	Order	Error	Order	Error	Order
1.2	4	2.2474e-02		7.0143e-03		1.4527e-03		4.9926e-04	
	8	3.4578e-03	2.70	4.9966e-04	3.81	6.8647e-05	4.40	1.0161e-05	5.62
	16	5.3896e-04	2.68	3.7767e-05	3.73	1.6374e-06	5.39	2.1199e-07	5.58
	32	8.9627e-05	2.59	2.3724e-06	3.99	4.7374e-08	5.11	3.9447e-09	5.75
	64	1.3738e-05	2.71	1.4821e-07	4.00	1.6634e-09	4.83	6.1795e-09	–
1.5	4	2.2375e-02		6.5750e-03		1.4754e-03		5.1427e-04	
	8	3.5255e-03	2.67	4.8540e-04	3.76	7.0616e-05	4.38	1.0334e-05	5.64
	16	5.2884e-04	2.74	3.6469e-05	3.73	1.5973e-06	5.47	2.1366e-07	5.60
	32	7.3591e-05	2.85	2.3946e-06	3.93	4.6519e-08	5.10	4.0898e-09	5.71
	64	1.2152e-05	2.60	1.4952e-07	4.00	1.5106e-09	4.94	3.0398e-09	–
1.8	4	2.1961e-02		5.8414e-03		1.4917e-03		5.3969e-04	
	8	3.3207e-03	2.73	4.7328e-04	3.63	7.1611e-05	4.38	1.0510e-05	5.68
	16	5.0736e-04	2.71	3.6663e-05	3.69	1.5918e-06	5.49	2.2548e-07	5.54
	32	8.5424e-05	2.57	2.3404e-06	3.97	4.6582e-08	5.09	4.3315e-09	5.70
	64	1.5385e-05	2.47	1.4783e-07	3.98	1.5362e-09	4.92	1.8977e-09	–

Table 3 Errors and predicted convergence orders of the proposed B-spline Galerkin method for problem (1) when $u(x) = x^{2+\alpha}(1-x)^{2+\alpha}$

		$p=2$		$p=3$		$p=4$		$p=5$	
α	n	Error	Order	Error	Order	Error	Order	Error	Order
1.2	4	5.1722e-04		1.5830e-04		5.9343e-05		9.2923e-06	
	8	5.6428e-05	3.20	9.8594e-06	4.01	1.9711e-06	4.91	3.0282e-07	4.94
	16	7.7450e-06	2.87	8.8745e-07	3.47	1.0187e-07	4.27	2.5896e-08	3.55
	32	1.5663e-06	2.31	5.5348e-08	4.00	9.2200e-09	3.47	2.6047e-09	3.31
	64	2.2362e-07	2.81	4.3291e-09	3.68	8.7876e-10	3.39	2.7733e-10	3.23
1.5	4	4.7299e-04		1.0036e-04		6.8778e-05		1.0329e-05	
	8	3.8077e-05	3.63	8.8327e-06	3.51	1.5796e-06	5.44	2.3165e-07	5.48
	16	4.7110e-06	3.01	6.2947e-07	3.81	8.0972e-08	4.29	1.4660e-08	3.98
	32	9.5382e-07	2.30	3.7404e-08	4.07	5.4991e-09	3.88	1.2180e-09	3.59
	64	1.4732e-07	2.69	3.5081e-09	3.41	4.0949e-10	3.75	1.0493e-10	3.54
1.8	4	3.8102e-04		6.2802e-05		6.9806e-05		9.1928e-06	
	8	3.2503e-05	3.55	7.9412e-06	2.98	1.1149e-06	5.97	2.8789e-07	5.00
	16	4.0344e-06	3.01	3.5961e-07	4.46	4.8537e-08	4.52	9.0202e-09	5.00
	32	7.9805e-07	2.34	3.1392e-08	3.52	3.1359e-09	3.95	2.6381e-10	5.10
	64	1.9118e-07	2.06	2.4189e-09	3.70	1.6002e-10	4.29	1.5901e-11	4.05

Table 4 Number of large outliers of $n^{1-\alpha}A_n^{p,\alpha}$ for different p, n, and α

p	n	$\alpha=1.2$	$\alpha=1.5$	$\alpha=1.8$	$\alpha=1.99$
3	63	0	2	2	2
4	62	0	2	2	2
5	61	2	2	2	4
6	60	2	2	4	4
7	59	2	4	4	4

e.g., [3]). We also see a stagnation in the convergence for $p = 5$; this is due to numerical issues in the difficult computation of the entries of $A_n^{p,\alpha}$ and the right-hand side for large p. In this perspective, a deeper investigation of the effect of the used quadrature can be of interest.

We continue this section by checking relations (11) and (12). In order to do this, for fixed n and p, we define the following uniform grid on $[0, \pi]$:

$$\Gamma := \left\{ \theta_k := \frac{k\pi}{n+p-2} : k = 1, \ldots, n+p-2 \right\}.$$

Then, we compare the (accordingly reordered) sampling of $f^{2p+1,\alpha}$ on Γ with the eigenvalues of both $T_n^{p,\alpha}$ and $n^{1-\alpha}A_n^{p,\alpha}$. In Fig. 1, we fix $p = 3$, $n = 63$ and vary $\alpha \in \{1.2, 1.5, 1.8\}$. For both $T_n^{p,\alpha}$ and $n^{1-\alpha}A_n^{p,\alpha}$, we observe a very good matching, which numerically validates Proposition 2 and Theorem 2. As a further confirmation, we obtained similar results also for $p = 4$, $n = 62$, and $\alpha \in \{1.2, 1.5, 1.8\}$; see Fig. 2. Note that, in accordance with Remark 1, in case of $n^{1-\alpha}A_n^{p,\alpha}$ and $\alpha = 1.8$, there are few large outliers, which do not behave like

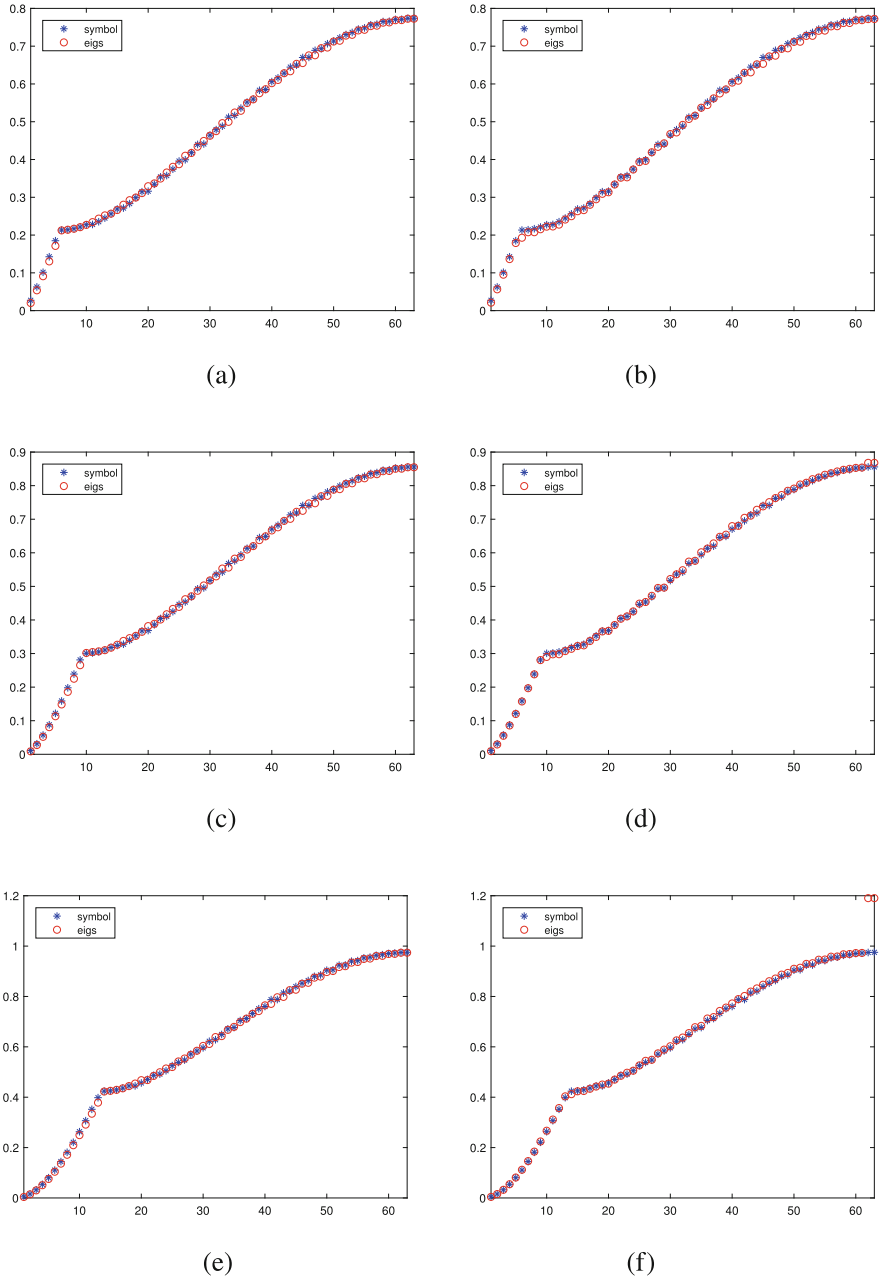


Fig. 1 Comparison of the eigenvalues of $T_n^{p,\alpha}$ and $n^{1-\alpha}A_n^{p,\alpha}$ (red circled) with a uniform sampling of $f^{2p+1,\alpha}$ on Γ, ordered in ascending way (blue asterisk), for $n = 63$, $p = 3$, and $\alpha = 1.2$ (top row), $\alpha = 1.5$ (middle row), and $\alpha = 1.8$ (bottom row). (**a**) $T_n^{p,1.2}$. (**b**) $n^{-0.2}A_n^{p,1.2}$. (**c**) $T_n^{p,1.5}$. (**d**) $n^{-0.5}A_n^{p,1.5}$. (**e**) $T_n^{p,1.8}$. (**f**) $n^{-0.8}A_n^{p,1.8}$

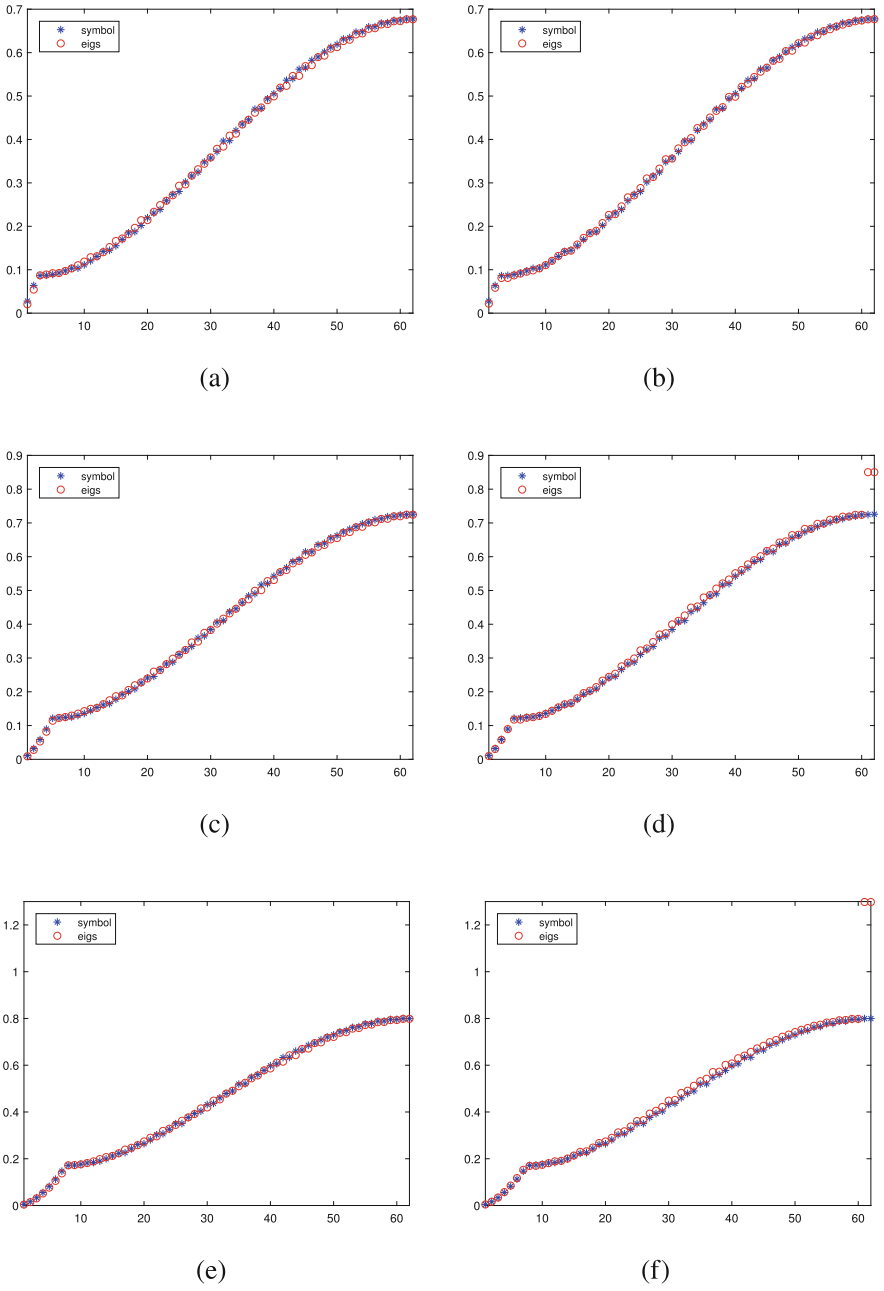


Fig. 2 Comparison of the eigenvalues of $T_n^{p,\alpha}$ and $n^{1-\alpha}A_n^{p,\alpha}$ (red circled) with a uniform sampling of $f^{2p+1,\alpha}$ on Γ, ordered in ascending way (blue asterisk), for $n = 62$, $p = 4$, and $\alpha = 1.2$ (top row), $\alpha = 1.5$ (middle row), and $\alpha = 1.8$ (bottom row). (**a**) $T_n^{p,1.2}$. (**b**) $n^{-0.2}A_n^{p,1.2}$. (**c**) $T_n^{p,1.5}$. (**d**) $n^{-0.5}A_n^{p,1.5}$. (**e**) $T_n^{p,1.8}$. (**f**) $n^{-0.8}A_n^{p,1.8}$

the symbol. The magnitude of these outliers seems to be less and less large as α approaches 1. Such a result is somehow in contrast with what we experienced in collocation, where the outlier magnitude was independent of α (see [22]).

Table 4 shows the precise number of large outliers obtained by counting how many eigenvalues of $n^{1-\alpha} A_n^{p,\alpha}$ are larger than $\max_{[0,\pi]} f^{2p+1,\alpha} + \epsilon$, with $\epsilon = n^{-1}$. We clearly see that the number of outliers depends on p and that, when α approaches 2, it increases in a similar way as it was observed for B-spline discretizations of integer-order differential equations (see, e.g., [12]).

6 Conclusions

We focused on the coefficient matrices obtained by B-spline Galerkin discretizations of Riesz fractional diffusion equations over uniform meshes. For an arbitrary polynomial degree p, we showed that, as for collocation, the resulting coefficient matrices possess a Toeplitz-like structure and that their spectral distribution is the same as for the collocation formulation with B-splines of degree $2p + 1$. As a consequence, the Galerkin matrices were proved to be poorly conditioned in both low and high frequencies, just like the collocation ones. Finally, we numerically observed that the approximation order of the Galerkin approach for smooth solutions does not depend on the fractional derivative order as for collocation and that it coincides with $p + 1$ as for non-fractional diffusion problems.

As future work, the use of reduced spline spaces could be explored in the fractional context since they lead to Galerkin discretizations with superior spectral properties for non-fractional diffusion problems [19]. Moreover, since it is known that the solution of (1) can exhibit singularities near the boundaries [2], locally non-uniform knot sequences could be considered as well to improve the overall accuracy of the method in case of non-smooth functions. Finally, following the results in [6–8], all the information provided by the symbol may be leveraged for the design of effective preconditioners and fast multigrid/multi-iterative solvers whose convergence speed is independent of the fineness parameters and the approximation parameters as well as the fractional derivative order.

Acknowledgments All the authors are members of the INDAM research group GNCS. The second and the last authors are partially supported by the Beyond Borders Program of the University of Rome Tor Vergata through the project ASTRID (CUP E84I19002250005) and by the MIUR Excellence Department Project awarded to the Department of Mathematics, University of Rome Tor Vergata (CUP E83C18000100006).

References

1. Bueno-Orovio, A., Kay, D., Grau, V., Rodriguez, B., Burrage, K.: Fractional diffusion models of cardiac electrical propagation: role of structural heterogeneity in dispersion of repolarization. J. R. Soc. Interface **11**, 20140352 (2014)
2. Cai, M., Li, C.: Regularity of the solution to Riesz-type fractional differential equation. Integr. Transf. Spec. Funct. **30**, 711–742 (2019)
3. Cottrell, J.A., Hughes, T.J.R., Bazilevs, Y.: Isogeometric Analysis: Toward Integration of CAD and FEA. Wiley, Chichester (2009)
4. de Boor, C.: A Practical Guide to Splines. Springer, New York (2001)
5. Donatelli, M., Garoni, C., Manni, C., Serra-Capizzano, S., Speleers, H.: Spectral analysis and spectral symbol of matrices in isogeometric collocation methods. Math. Comput. **85**, 1639–1680 (2016)
6. Donatelli, M., Garoni, C., Manni, C., Serra-Capizzano, S., Speleers, H.: Symbol-based multigrid methods for Galerkin B-spline isogeometric analysis. SIAM J. Numer. Anal. **55**, 31–62 (2017)
7. Donatelli, M., Mazza, M., Serra-Capizzano, S.: Spectral analysis and structure preserving preconditioners for fractional diffusion equations. J. Comput. Phys. **307**, 262–279 (2016)
8. Donatelli, M., Mazza, M., Serra-Capizzano, S.: Spectral analysis and multigrid methods for finite volume approximations of space-fractional diffusion equations. SIAM J. Sci. Comput. **40**, A4007–A4039 (2018)
9. Ervin, V.J., Roop, J.P.: Variational formulation for the stationary fractional advection dispersion equation. Numer. Methods Partial Differ. Equ. **22**, 558–576 (2006)
10. Garoni, C., Manni, C., Pelosi, F., Serra-Capizzano, S., Speleers, H.: On the spectrum of stiffness matrices arising from isogeometric analysis. Numer. Math. **127**, 751–799 (2014)
11. Garoni, C., Serra-Capizzano, S.: Generalized Locally Toeplitz Sequences: Theory and Applications, vol. I. Springer, Cham (2017)
12. Garoni, C., Speleers, H., Ekström, S.-E., Reali, A., Serra-Capizzano, S., Hughes, T.J.R.: Symbol-based analysis of finite element and isogeometric B-spline discretizations of eigenvalue problems: exposition and review. Arch. Comput. Methods Eng. **26**, 1639–1690 (2019)
13. Grenander, U., Szegő, G.: Toeplitz Forms and Their Applications, 2nd edn. Chelsea, New York (1984)
14. Hao, Z., Zhang, Z.: Optimal regularity and error estimates of a spectral Galerkin method for fractional advection-diffusion-reaction equations. SIAM J. Numer. Anal. **58**, 211–233 (2020)
15. Jiang, Y., Ma, J.: High-order finite element methods for time-fractional partial differential equations. J. Comput. Appl. Math. **235**, 3285–3290 (2011)
16. Lei, S.L., Sun, H.W.: A circulant preconditioner for fractional diffusion equations. J. Comput. Phys. **242**, 715–725 (2013)
17. Lin, Z., Wang, D.: A finite element formulation preserving symmetric and banded diffusion stiffness matrix characteristics for fractional differential equations. Comput. Mech. **62**, 185–211 (2018)
18. Lyche, T., Manni, C., Speleers, H.: Foundations of spline theory: B-splines, spline approximation, and hierarchical refinement. In: Lyche T., et al. (eds.) Splines and PDEs: From Approximation Theory to Numerical Linear Algebra. Lect. Notes Math., vol. 2219, pp. 1–76. Springer, Berlin (2018)
19. Manni, C., Sande, E., Speleers, H.: Application of optimal spline subspaces for the removal of spurious outliers in isogeometric discretizations. Comput. Methods Appl. Mech. Eng. **389**, 114260 (2022)
20. Mao, Z., Karniakadis, G.E.: A spectral method (of exponential convergence) for singular solutions of the diffusion equation with general two-sided fractional derivative. SIAM J. Numer. Anal. **56**, 24–49 (2018)
21. Mazza, M.: B-spline collocation discretizations of Caputo and Riemann-Liouville derivatives: a matrix comparison. Fract. Calc. Appl. Anal. **24**, 1670–1698 (2021)

22. Mazza, M., Donatelli, M., Manni, C., Speleers, H.: On the matrices in B-spline collocation methods for Riesz fractional equations and their spectral properties. Numer. Linear Algebra Appl. **30**, e2462 (2023)
23. Meerschaert, M.M., Tadjeran, C.: Finite difference approximations for fractional advection-dispersion flow equations. J. Comput. Appl. Math. **172**, 65–77 (2004)
24. Pan, G., Chen, W., Sze, K.Y.: Gauss-Jacobi-type quadrature rules for fractional directional integrals. Comput. Math. Appl. **66**, 597–607 (2013)
25. Pan, J., Ng, M.K., Wang, H.: Fast preconditioned iterative methods for finite volume discretization of steady-state space-fractional diffusion equations. Numer. Algorithms **74**, 153–173 (2017)
26. Pezza, L., Pitolli, F.: A multiscale collocation method for fractional differential problems. Math. Comput. Simul. **147**, 210–219 (2018)
27. Pitolli, F.: Optimal B-spline bases for numerical solution of fractional differential problems. Axioms **7**, 46 (2018)
28. Pitolli, F.: On the numerical solution of fractional boundary value problems by a spline quasi-interpolant operator. Axioms **9**, 61 (2020)
29. Podlubny, I.: Fractional Differential Equations. Academic Press, New York (1998)
30. Samko, S.G., Kilbas, A.A., Marichev, O.I.: Fractional Integrals and Derivatives, vol. 1. Yverdon-les-Bains, Switzerland: Gordon and Breach Science Publishers, Yverdon (1993)
31. Tian, W., Zhou, H., Deng, W.: A class of second order difference approximations for solving space-fractional diffusion equations. Math. Comput. **84**, 1703–1727 (2015)
32. Tuğba, A.Y., Arshad, S., Baleanu, D.: Optimal chemotherapy and immunotherapy schedules for a cancer-obesity model with Caputo time fractional derivative. Math. Methods Appl. Sci. **41**, 9390–9407 (2018)
33. Unser, M., Blu, T.: Fractional splines and wavelets. SIAM Rev. **42**, 43–67 (2000)
34. Xu, K., Darve, K.: Isogeometric collocation method for the fractional Laplacian in the 2D bounded domain. Comput. Methods Appl. Mech. Eng. **364**, 112936 (2020)
35. Zeng, F., Mao, Z., Karniadakis, G.E.: A generalized spectral collocation method with tunable accuracy for fractional differential equations with end-point singularities. SIAM J. Sci. Comput. **39**, A360–A383 (2017)

Do the Mittag–Leffler Functions Preserve the Properties of Their Matrix Arguments?

Marina Popolizio

Abstract The matrix Mittag–Leffler (ML) functions are receiving great attention at the moment. As a matter of fact, in many applications, the matrix argument has special features and/or structure and is important to know if the application of ML functions preserves them. We collect results here that can help researchers working with this topic and who usually struggle to find them because they are scattered around or not explicitly derived for the special case of ML functions. Furthermore, the treatment is also suitable for non-experts in linear algebra, giving adequate references to the appropriate literature. In particular, nonnegativity and circulant structure are addressed, which may be of great interest in the analysis of systems of fractional differential equations or in the context of graph theory.

1 Introduction

The matrix Mittag–Leffler (ML) functions nowadays enjoy considerable interest thanks to the important role they deserve in many applications. We cite, for example, the Fractional Calculus, that is, the branch of mathematical analysis which studies integrals and derivatives of arbitrary order [20, 28, 33], with interesting applications to nonlocal models [2, 4, 26]. In this context, the matrix ML functions are used to solve systems of fractional differential equations, to analyze their properties, and to assess the observability and controllability of fractional linear systems, just to quote some instances [8, 11, 15, 16, 18, 31]. Recently, the matrix ML functions have been used also in the context of network science to define new measures that, thanks to the behavior of the underlying ML functions, interpolate between resolvent-based and exponential-based measures [1, 12, 13]. The matrix ML functions have been recently used also to solve time and space generalized diffusion equations on graphs

M. Popolizio (✉)
Dipartimento di Ingegneria Elettrica e dell'Informazione, Politecnico di Bari, Bari, Italy

Member of the INdAM Research Group GNCS, Roma, Italy
e-mail: marina.popolizio@poliba.it

A. Cardone et al. (eds.), *Fractional Differential Equations*,
Springer INdAM Series 50, https://doi.org/10.1007/978-981-19-7716-9_5

[7]. In these cases, as in many other circumstances, it is important to establish if the application of the matrix ML functions preserves features and/or structures characterizing the matrix argument.

The ML function is defined as

$$E_{\alpha,\beta}(z) = \sum_{j=0}^{\infty} \frac{z^j}{\Gamma(\alpha j + \beta)}, \quad z \in \mathbb{C}, \quad \alpha, \beta \in \mathbb{C}, \Re(\alpha) > 0, \tag{1}$$

where

$$\Gamma(z) = \int_0^{\infty} t^{z-1} \mathrm{e}^{-t} \mathrm{d}t$$

is the Euler gamma function. From the definition (1), it is immediate to recognize that for $\alpha = \beta = 1$ the ML function reduces to the exponential.

Even at the scalar level, ML functions do not always satisfy the properties that researchers expect. For example, this is the case of the semigroup property; indeed, in general for $s, t \geq 0$ and $a \in \mathbb{R}$

$$E_{\alpha,\beta}\left(a(s+t)^{\alpha\beta}\right) \neq E_{\alpha,\beta}\left(as^{\alpha\beta}\right) E_{\alpha,\beta}\left(at^{\alpha\beta}\right)$$

(see, for example, [10, 30]), unless the special cases $\alpha = \beta = 1$, and $a = 0, \beta = 1$, or $\beta = 2$. Unfortunately, it has been considered true several times in the literature, and this has often created confusion and led to mathematically inconsistent conclusions. We therefore believe that, given the increasing use of matrix ML functions even by non-specialists, collecting some results could be useful for easy reference.

The ML function is clearly entire, so its definition simply generalizes to matrix arguments as

$$E_{\alpha,\beta}(A) = \sum_{j=0}^{\infty} \frac{A^j}{\Gamma(\alpha j + \beta)}, \quad A \in \mathbb{C}^{n \times n}, \quad \alpha, \beta \in \mathbb{C}, \Re(\alpha) > 0. \tag{2}$$

Many issues arise when the numerical approximation of the quantity above is required. The definition (2), or even (1) for the scalar case, may be simply applied together with a suitable truncation. This approach, however, works only when the argument is "small," since differently the convergence may be extremely slow, thus requiring a large number of terms for its numerical approximation, with the related problem of the fast growth of the Gamma function. Other approaches are then necessary and we just refer to [17, 18, 32] for a comprehensive discussion on them

and for the algorithm they propose that effectively combines high accuracy and low computational cost.[1]

This chapter is organized as follows: in Sect. 2, we list some of the features that are not preserved by the ML functions. Section 3 discusses the case of nonnegative, positive and nonnegative definite matrix arguments. We show that the ML functions preserve these features for any choice of the parameters such that $\alpha, \beta > 0$. To guarantee the nonnegativity of $E_{\alpha,\beta}(A)$ when A is a Metzler matrix, more restrictions on the parameters are needed, as proven in Proposition 6, by resorting to a new result on the strictly completely monotonicity of the ML functions. Sections 4 and 5 deal with the preservation of the centrosymmetric and circulant structures, respectively. In Sect. 6, quasi-Toeplitz matrices are addressed whose structure is preserved by the ML functions, unlike Toeplitz matrices discussed in Sect. 2. In Sect. 7, we deal with the case of parameter-dependent matrices, as in the case of temporal networks. Some conclusions are collected in Sect. 8.

2 What Is Not Preserved

In this section, we analyze some characteristics which are not preserved by the matrix ML functions. We report in the following basic concepts of matrix analysis, and we refer to [19, 22, 25] for further insights.

Toeplitz Matrices

Definition 1 A real matrix A is a *Toeplitz* matrix if each descending diagonal of A from left to right is constant, that is to say,

$$A_{i,j} = a_{i-j},$$

so

$$A = \begin{bmatrix} a_0 & a_{-1} & a_{-2} & \cdots & \cdots & a_{-n+1} \\ a_1 & a_0 & a_{-1} & \ddots & & \vdots \\ a_2 & a_1 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & a_{-1} & a_{-2} \\ \vdots & & \ddots & a_1 & a_0 & a_{-1} \\ a_{n-1} & \cdots & \cdots & a_2 & a_1 & a_0 \end{bmatrix}. \tag{3}$$

[1] www.mathworks.com/matlabcentral/fileexchange/66272-mittag-leffler-function-with-matrix-arguments.

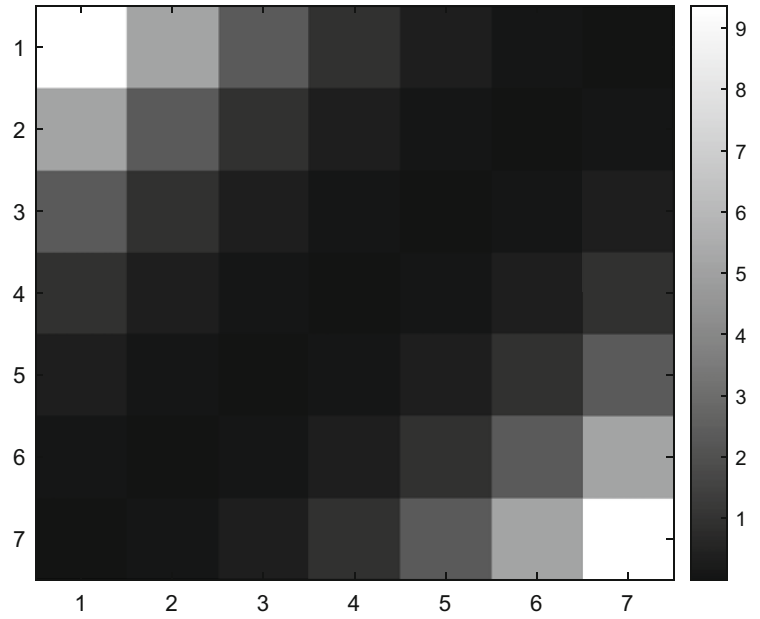


Fig. 1 Absolute values of the differences along the diagonals of $E_{\alpha,\beta}(T)$ with the symmetric Toeplitz matrix $T = \text{tridiag}(0.5, 1, 0.5)$ of dimension 8×8, $\alpha = 0.5$ and $\beta = 1$

This structure is not preserved by the action of ML functions; indeed, in general, the power of Toeplitz matrices is not a Toeplitz matrix, and thus the matrix (2) is not in general a Toeplitz matrix, even if A is so.

However, as seen for the exponential function [5], the ML function of a Toeplitz matrix is still Toeplitz except for few nonzero entries in the upper left-hand corner and in the lower right-hand corner.

We show this property by means of a simple example carried out in Matlab ver. 9.10.0.1602886 (R2021a): we consider an $n \times n$ tridiagonal matrix T with row entries $(0.5, 1, 0.5)$, we compute the ML function $F = E_{\alpha,\beta}(T)$ by means of the `ml_matrix` code [18], and we consider the error measured as `abs(F(1:n-1,1:n-1)-F(2:n,2:n))`. For Toeplitz matrices, this difference matrix is identically zero, while we observe, from Fig. 1 referring to $n = 8, \alpha = 0.5$ and $\beta = 1$, that entries in the upper left-hand corner and in the lower right-hand corner are significantly larger than zero.

In Sect. 6, we specifically address this topic by considering quasi-Toeplitz matrices.

Stochastic Matrices

Definition 2 A square matrix A is (row) *stochastic* if all its entries are nonnegative and the entries of each row sum to 1.

A column stochastic matrix is a real square matrix of nonnegative entries, with each column summing to 1.

A doubly stochastic matrix is a square matrix of nonnegative entries such that the sum of the entries in every row and every column is 1.

In common applications, each entry of these matrices represents a probability and they are largely used, for example, in probability theory, statistics, and mathematical finance.

Proposition 1 ([22]) *Any row stochastic matrix has* 1 *as eigenvalue with corresponding eigenvector* $e = (1, \ldots, 1)^T$.

Any column stochastic matrix has 1 *as eigenvalue with corresponding left eigenvector* e^T.

The stochasticity is not preserved by ML functions. Indeed, given a row stochastic matrix A, as stated in the characterization above, it has the eigenvector e with corresponding eigenvalue 1. It follows, from the general theory of matrix functions [22], that $E_{\alpha,\beta}(A)$ will have e as eigenvector with corresponding eigenvalue $E_{\alpha,\beta}(1)$ which is different from 1 for any choice of α and β. Analogously, for column stochastic matrices, in which case e^T is the left eigenvector corresponding to the eigenvalue 1 for A but not for $E_{\alpha,\beta}(A)$.

Monotonicity

To define the monotonicity of a matrix function, we need to introduce a specific ordering. For Hermitian matrices X and Y, we use the notation $X > Y$ if the matrix $X - Y$ is positive definite. With this notation, a matrix function $f : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$ is monotone if $f(A) > f(B)$ whenever $A > B$.

The ML functions cannot be monotone according to the definition above, as we may infer from the fact that for $r > 1$ the power function A^r is not monotone either [22].

However, if $0 < \alpha \leq 1$ and $\beta \geq \alpha$, the scalar ML functions are completely monotonic (c.m.) in $(-\infty, 0)$ [28], that is to say,

$$(-1)^k E^{(k)}_{\alpha,\beta}(x) \geq 0, \quad \text{for } k = 0, 1, 2, \ldots$$

for any $x \in (-\infty, 0)$ with $E^{(k)}_{\alpha,\beta}$ denoting the k-th order derivative of the ML function. We exploit this characteristic in Sect. 3 to extend a result by Varga [34], which connects c.m. functions and Metzler matrices.

M-matrices

M-matrices, as defined below, are important in many applications like finite difference methods for partial differential equations, Markov chains, etc.

Definition 3 A real matrix A is a non-singular *M-matrix* if $A = sI - B$, where B has all nonnegative entries, I denotes the identity matrix, and s is greater than the spectral radius of B, i.e., the largest absolute value of its eigenvalues.

M-matrices are characterized by negative off-diagonal entries. We take as example the M-matrix arising from the finite difference discretization of second-order derivatives, with swapped sign,

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}.$$

We show that not even the exponential function preserves this property. Indeed, by using the Matlab function `expm`, which is nowadays one of most effective tool to numerically compute the matrix exponential, we get

$$\texttt{expm(A)} = \begin{bmatrix} 11.7419 & -10.1104 & 4.3528 \\ -10.1104 & 16.0947 & -10.1104 \\ 4.3528 & -10.1104 & 11.7419 \end{bmatrix};$$

this is clearly not an M-matrix for the presence of positive extra diagonal entries. We conclude that ML functions do not preserve M-matrices in view of the fact that the exponential is one instance of ML functions.

In particular, we derive that ML functions do not preserve the Stieltjes matrices, since they represent a subset of M-matrices, as we deduce from their definition:

Definition 4 ([22]) A *Stieltjes* matrix is a symmetric positive definite matrix $A \in \mathbb{R}^{n\times n}$ such that $A_{ij} \leq 0$ for $i \neq j$.

Matrix Groups

Higham and coauthors [23] gave a complete description of matrix functions which preserve matrix groups. In particular, they show that these functions have to commute with the inverse function to preserve, for example, orthogonality, perplecticity and symplecticity, or with the conjugate inverse function in order to preserve unitarity and conjugate symplecticity. The ML functions do not commute as required, so they do not preserve these groups.

3 Nonnegativity Preservation

We introduce some useful definitions to discuss whether the ML functions preserve nonnegativity and/or essential nonnegativity of their matrix argument.

Definition 5 ([22]) A matrix $A \in \mathbb{R}^{n\times n}$ is

- *Nonnegative* if it is entrywise nonnegative. In this case, we use the notation $A \geq 0$.
- *Positive* if it is entrywise positive. In this case, we use the notation $A > 0$.
- A *Metzler* matrix, or essentially nonnegative, if $A_{ij} \geq 0$ for $i \neq j$.

Proposition 2 *For any value of $\alpha, \beta \geq 0$, the ML functions preserve the class of*

- *Nonnegative matrices*
- *Nonnegative symmetric nonnegative definite matrices, i.e., matrices having only nonnegative eigenvalues*

Proof The first claim easily follows since the coefficients defining the ML function (1) are all nonnegative for the assumptions on α and β [29, 34].

For the second claim, we recall that a characterization due to Micchelli and Willoughby [29] states that a continuous function preserves nonnegative symmetric matrices if and only if its divided differences, or equivalently its derivatives, are nonnegative. In our case, a term-by-term derivation of (1) leads to derivatives expressed as

$$E_{\alpha,\beta}^{(k)}(z) \equiv \frac{d^k}{dz^k} E_{\alpha,\beta}(z) = \sum_{j=k}^{\infty} \frac{(j)_k}{\Gamma(\alpha j + \beta)} z^{j-k}, \quad k \in \mathbb{N}, \tag{4}$$

with $(x)_k$ denoting the falling factorial

$$(x)_k = x(x-1)\cdots(x-k+1),$$

and so they are nonnegative on $\mathbb{R}_+$ for the hypothesis on α and β. Moreover, this hypothesis also guarantees that the eigenvalues of $E_{\alpha,\beta}(A)$ are all nonnegative, so it results nonnegative definite too. □

Proposition 3 *The ML functions with $\alpha, \beta \geq 0$ map positive matrices into nonnegative matrices.*

Proof The continuity of the ML functions and the fact that the set of strictly positive matrices is dense in the set of all nonnegative matrices lead to the result [3]. □

Remark 1 We observe that $1/\Gamma(x) > 0$ (respectively, ≥ 0) when $-2k < x < -2k+1$ (respectively, $-2k \leq x \leq -2k+1$) for $k \in \mathbb{N}$. Thus, the above propositions extend to some negative values of β and sufficiently large values of α (namely, when $\beta \in (-2k, -2k+1)$ and $\alpha > -\beta$). We think however that these cases are of less importance in common applications.

Metzler matrices are of particular importance in view of their occurrence in significant applications such as Markov chains, stability of linear dynamical systems, and the numerical solution of partial differential equations [22, 27]. They differ from nonnegative matrices as their diagonal entries are free in the sign and this weaker hypothesis makes the application of Proposition 2 impossible. Varga [34] found interesting connections between strictly c.m. functions and Metzler matrices. Here, we prove that the ML functions are strictly c.m. to extend these results.

Proposition 4 *When $0 < \alpha \leq \beta < 1$, the ML functions are strictly c.m. in $(-\infty, 0)$, i.e., $(-1)^j E_{\alpha,\beta}^{(j)}(x) > 0$ for any $x < 0$ and for any $j = 0, 1, \ldots$.*

Proof It is well known that ML functions are c.m. in $(-\infty, 0)$ when $0 < \alpha \leq 1$ and $\beta \geq \alpha$ [28] as they are Laplace transform of a nonnegative absolutely continuous function $S_{\alpha,\beta}$. If we restrict to $0 < \alpha \leq \beta < 1$, the function $S_{\alpha,\beta}$ may be represented as

$$S_{\alpha,\beta}(u) = \sum_{n=0}^{\infty} (-1)^n \frac{u^n}{n!\Gamma(\beta - \alpha - \alpha n)}, \quad u \in \mathbb{R}^+.$$

Dubourdieu found that Laplace transforms of nonnegative absolutely continuous functions which do not identically reduce to a constant are strictly c.m. [9]; this allows us to infer that the ML function is strictly c.m. since the function $S_{\alpha,\beta}$ does not identically reduce to a constant. □

Remark 2 For the special case $\alpha = \beta = 1$, we recover the exponential which is strictly c.m. in $(-\infty, 0)$ [34].

When ML functions are c.m., they transform Metzler matrices into nonnegative matrices, as we will demonstrate in Proposition 6. For clarity, we report here a result due to Varga [34], which states the same property for the exponential function and represents our starting point.

Proposition 5 ([34, Theorem 4]) *Let B be a Metzler matrix; then, for any $t \geq 0$, $\exp(tB) \geq 0$. Moreover, for any $t > 0$, $\exp(tB) > 0$ if and only if B is irreducible, i.e., it is not similar via a permutation to a block upper triangular matrix.*

Proposition 6 *Let A be a Metzler matrix and $0 < \alpha \leq \beta < 1$.*
Then, for any $t \geq 0$, $E_{\alpha,\beta}(tA) \geq 0$.
Moreover, for any $t > 0$, $E_{\alpha,\beta}(tA) > 0$ if and only if A is irreducible.

Proof For the given parameters, the ML function is strictly c.m., and thus, as shown in Proposition 4, the function $G(x) = E_{\alpha,\beta}(-x)$ is c.m. on $(0, \infty)$ and is entire. Then, by [34, Theorem 1], $G(yI - C) \geq 0$ for any nonnegative matrix C and for any positive value of y. On the other side, a Metzler matrix can be easily transformed into a nonnegative matrix by adding sufficiently large positive values to its diagonal, namely, for any positive s sufficiently large, the matrix $C \equiv tA + sI$ is nonnegative for any $t \geq 0$. So, $0 \leq G(sI - C) = G(-tA) = E_{\alpha,\beta}(tA)$. The proof of the second part of the claim essentially follows that of Proposition 5 given in [34]. □

4 Centrosymmetric Matrices

Centrosymmetric matrices are important in applications since, for example, they represent the transition matrices of certain types of Markov processes [35].

Definition 6 ([22]) A matrix A is *centrosymmetric* if $JAJ = A$, where J is the "exchange" matrix, that is, J is the identity matrix I with the columns in reverse order.

Proposition 7 *The ML function preserves the centrosymmetric structure of its matrix argument.*

Proof We repeatedly use the property $JJ = I$ in the expression for $E_{\alpha,\beta}(A)$ (2) to get the result. □

To give a graphical representation of this property, we consider the centrosymmetric 10×10 matrix:

$$C = \begin{bmatrix} 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & -1 \\ 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & -1 & -2 \\ 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & -1 & -2 & -3 \\ 6 & 5 & 4 & 3 & 2 & 1 & 0 & -1 & -2 & -3 & -4 \\ 5 & 4 & 3 & 2 & 1 & 0 & -1 & -2 & -3 & -4 & -5 \\ 6 & 5 & 0 & -5 & 0 & -1 & 0 & -5 & & 5 & 6 \\ -5 & -4 & -3 & -2 & -1 & 0 & 1 & 2 & 3 & 4 & 5 \\ -4 & -3 & -2 & -1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 \\ -3 & -2 & -1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ -2 & -1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ -1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{bmatrix}, \tag{5}$$

we scale it by a factor of -10 and we compute $E_{\alpha,\beta}(C)$ for $\alpha = 1.5$ and $\beta = 1$. Figure 2 shows the centrosymmetric structure of both C and $E_{\alpha,\beta}(C)$.

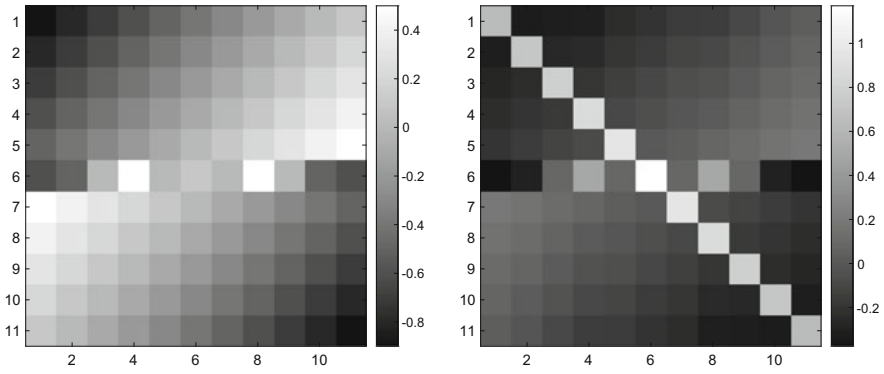


Fig. 2 Centrosymmetric matrix C in (5) (left plot) and $E_{\alpha,\beta}(C)$ (right plot) for $\alpha = 1.5$ and $\beta = 1$

5 Circulant Matrices

Circulant matrices are encountered in applications involving the discrete Fourier transform or the study of cyclic codes for error correction. Moreover, they are particularly important in the analysis of circulant graphs or networks, since their adjacency matrices have this form. Circulant networks are used, for example, in the design of computer and telecommunication networks and distributed computation [14].

Definition 7 ([19]) Define the downshift permutation S_n as

$$S_n = \begin{bmatrix} 0 & 0 & & & 0 & 1 \\ 1 & 0 & 0 & \ddots & 0 & \\ 0 & 1 & 0 & \ddots & & \\ & \ddots & \ddots & \ddots & 0 & \\ 0 & \dots & 0 & 1 & 0 & \end{bmatrix}.$$

Given an n-vector $a = [a_1, a_2, \dots, a_n]^T$, an $n \times n$ *circulant* matrix is defined as

$$A = [a, S_n a, S_n^2 a, \dots, S_n^{n-1} a],$$

so each column is a "downshifted" version of its predecessor.

Equivalently, a circulant matrix may be viewed as a special kind of Toeplitz matrix where each row is obtained from the previous one by cyclically moving the entries one place to the right.

An example of circulant matrix of order $n = 5$ is

$$A = \begin{bmatrix} a_1 & a_5 & a_4 & a_3 & a_2 \\ a_2 & a_1 & a_5 & a_4 & a_3 \\ a_3 & a_2 & a_1 & a_5 & a_4 \\ a_4 & a_3 & a_2 & a_1 & a_5 \\ a_5 & a_4 & a_3 & a_2 & a_1 \end{bmatrix}.$$

Proposition 8 ([6, 22]) *A circulant matrix A is characterized by the fact that it is diagonalized by the discrete Fourier transform, that is,*

$$FAF^{-1} = D$$

with D diagonal and F denoting the Fourier matrix whose (r, s) entry is defined as

$$F_{r,s} = \frac{1}{\sqrt{n}} \exp\left(-\frac{2\pi \mathrm{i}}{n}(r-1)(s-1) \right)$$

if n is the dimension of A, i denotes the imaginary unit, and $r, s = 1, \dots, n$.

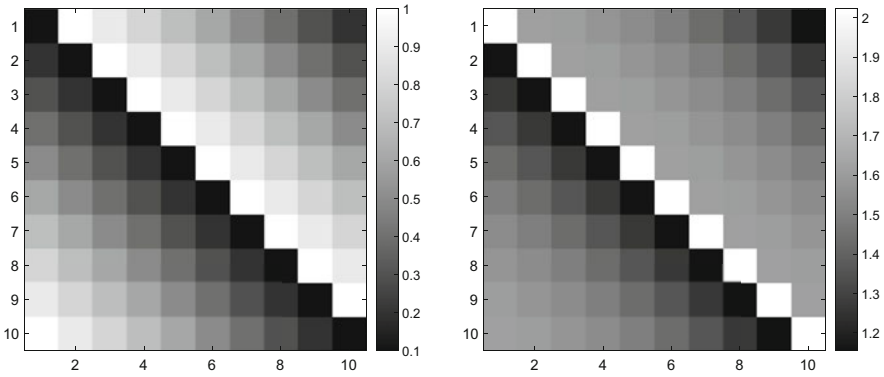


Fig. 3 Circulant matrix A of dimension 10×10 defined by means of the vector $a = [1, 2, 3, \ldots, 10]$ scaled by a factor 10 (left plot) and $E_{\alpha,\beta}(A)$ (right plot) for $\alpha = 1.5$, $\beta = 1$

It is then interesting to show that ML functions preserve this structure, as stated in the following proposition.

Proposition 9 *The ML function preserves the circulant structure of its matrix argument.*

Proof From the characterization above of circulant matrices, we deduce that also $E_{\alpha,\beta}(A)$ is diagonalized by the discrete Fourier transform, namely $E_{\alpha,\beta}(A) = F^{-1}E_{\alpha,\beta}(D)F$, thus leading to the claim. □

In Fig. 3, we give a graphical representation of the property above: we consider a 10×10 circulant matrix A defined by means of the vector $a = [1, 2, 3, \ldots, 10]$ scaled by a factor 10, and we compute $E_{\alpha,\beta}(A)$ for $\alpha = 1.5$ and $\beta = 1$. In the left plot, the circulant structure of A is evident, while the right plot confirms the circulant structure of $E_{\alpha,\beta}(A)$.

In view of the fact that ML functions preserve the circulant structure, we deduce a useful property of the ML functions, which descends from the following characterization due to Bharali and Holtz [3] that we report here for completeness.

Proposition 10 ([3, Theorem 10]) *For an entire function f to preserve nonnegativity of a circulant matrix of order n with first row $a_0, \ldots, a_{n-1}$, it is necessary and sufficient that for $l = 0, \ldots, n-1$*

$$\sum_{k=0}^{n-1} \omega^{-lk} f\left(\sum_{j=0}^{n-1} \omega^{jk} a_j\right) \geq 0$$

whenever $a_j \geq 0$ for $j = 0, \ldots, n-1$ and $\omega = \mathrm{e}^{2\pi \mathrm{i}/n}$.

Proposition 11 *For any $n \in \mathbb{N}$, whenever $a_j \geq 0$ for $j = 0, \ldots, n-1$, it results that*

$$\sum_{k=0}^{n-1} \omega^{-lk} E_{\alpha,\beta}\left(\sum_{j=0}^{n-1} \omega^{jk} a_j\right) \geq 0$$

for $\omega = \mathrm{e}^{2\pi \mathrm{i}/n}$ and for $l = 0, \ldots, n-1$.

Proof Since ML functions preserve nonnegativity of circulant matrices of any order $n \in \mathbb{N}$, the characterization given in Proposition 10 holds, thus concluding the proof. □

6 Quasi-Toeplitz Matrices

By starting from (3), we may extend the definition of a Toeplitz matrix to a semi-infinite matrix if

$$T_{i,j} = a_{j-i}, \ \ i, j \in \mathbb{Z}^+.$$

Many applications related to semi-infinite domains involve matrices of this form. For instance, when describing queuing models or random walks in the quarter plane or the discretization of boundary value problems by means of finite differences.

Let us introduce some definitions useful for the following [5]:

$$\mathbb{T} \equiv \left\{z \in \mathbb{C} : |z| = 1\right\},$$

$$\mathcal{W} \equiv \left\{a(z) = \sum_{i\in\mathbb{Z}} a_i z^i : \mathbb{T} \to \mathbb{C}, \sum_{i\in\mathbb{Z}} |a_i| < \infty\right\} \text{ with } \|a\|_{\mathcal{W}} \equiv \sum_{i\in\mathbb{Z}} |a_i|$$

$$\mathcal{W}_1 \equiv \left\{a(z) \in \mathcal{W} : a'(z) \in \mathcal{W}\right\} \text{ where } a'(z) = \sum_{i\in\mathbb{Z}} i a_i z^{i-1}$$

$$\mathcal{F} \equiv \left\{F = (f_{i,j})_{i,j\in\mathbb{Z}^+} : \sum_{i,j\in\mathbb{Z}^+} |f_{i,j}| < \infty\right\} \text{ with } \|F\|_{\mathcal{F}} \equiv \sum_{i,j\in\mathbb{Z}^+} |f_{i,j}|.$$

Definition 8 A semi-infinite matrix A is said to be *quasi-Toeplitz* (QT) if it can be written in the form

$$A = T(a) + C_a,$$

where $a(z) \in \mathcal{W}_1$ is called *symbol* and $C_a \in \mathcal{F}$.

We refer to $T(a)$ as the Toeplitz part of A and to C_a as the correction.
The class $\mathcal{QT}$ of QT-matrices is a Banach space equipped with the norm

$$\|A\|_{\mathcal{QT}} := \|a\|_{\mathcal{W}} + \|a'\|_{\mathcal{W}} + \|C_a\|_{\mathcal{F}}$$

with $a'(z)$ denoting the first derivative of $a(z)$.
If $a(z)$ is analytic in an open annulus enclosing $\mathbb{T}$, then the matrix A is said to be *analytically quasi-Toeplitz* (AQT).

Functions of QT-matrices have been studied in depth by Bini and coauthors [5], and we follow their analysis for the specific case of ML functions.

Proposition 12 *Let $A = T(a) + C_a$ be a QT-matrix. Then, there exists a QT-matrix Y such that*

$$\lim_{k\to\infty} \left\| Y - \sum_{j=0}^{k} \frac{A^j}{\Gamma(\alpha j + \beta)} \right\|_{\mathcal{QT}} = 0,$$

and Y is a QT-matrix with symbol $s \equiv E_{\alpha,\beta}(a)$, i.e.,

$$Y = T(s) + C_s, \quad C_s \in \mathcal{F}.$$

Moreover, if A is an AQT-matrix, then $E_{\alpha,\beta}(A)$ is an AQT-matrix too.

Proof The claim follows from [5, Theorem 3.1] since the ML functions are entire. □

Numerical evidence of this result is given in Fig. 1, which shows that the correction term C_s in Proposition 12 for the ML function of a QT-matrix has just few nonzero entries in the upper left-hand corner and in the lower right-hand corner.

The following result gives a bound on the norm of the correction part of $E_{\alpha,\beta}(A)$, by applying to the ML function the general result by Bini and coauthors [5, Theorem 3.3].

Proposition 13 *Under the hypotheses of Proposition 12, if $A = T(a) + C_a$ and $E_{\alpha,\beta}(A) = T(s) + C_s$ with $s \equiv E_{\alpha,\beta}(a)$, then*

$$\|C_s\|_{\mathcal{F}} \le \frac{1}{\|C_a\|_{\mathcal{F}}} \left((\|a'\|_{\mathcal{W}}^2 + \|C_a\|_{\mathcal{F}}) \frac{g(\|a\|_{\mathcal{W}} + \|C_a\|_{\mathcal{F}}) - g(\|a\|_{\mathcal{W}})}{\|C_a\|_{\mathcal{F}}} - \|a'\|_{\mathcal{W}}^2 g'(\|a\|_{\mathcal{W}}) \right),$$

where $g(z) = \sum_{i=0}^{\infty} \frac{z^i}{|\Gamma(\alpha i+\beta)|}$.

7 The ML Function with Time-Dependent Matrix Arguments

Many applications nowadays, rather than focusing on the network itself, on the importance of single vertices or edges for example, focus on dynamic systems that develop on the network. This is the case, just to cite some instances, of the flow of information via e-mail messages, mobile telephone calls, and social media or the spreading dynamics of some biological viruses [21, 24]. In these situations, one separates the underlying static network and the dynamical system on the network, thus dealing with *temporal networks*. In these models, interactions can appear and disappear over time, so the adjacency matrix varies with the time and its entries can be defined as

$$A_{i,j}(t) = \begin{cases} 1, & \text{if } i \text{ and } j \text{ are connected at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

Also, in this case, as in the static one, the matrix $E_{\alpha,\beta}(A(t))$ may offer a good measure that interpolates between the resolvent and the exponential behavior [1].

In this situation, the interest is preserving features that the matrix argument $A(t)$ has for any $t \geq 0$. Plainly, in these cases, all the results described above may be applied.

8 Conclusions

This chapter collects a series of results about the features that the matrix ML functions preserve. In many applications, outcomes of this type are useful to determine whether the models under investigation have specific properties or not. The use of ML functions is in fact spreading in many different research areas, such as fractional calculus and network theory. Here, for example, the nonnegativity of the ML functions of nonnegative matrix arguments is important to assess the features of differential systems or to properly define network measures. In these cases, the conditions $\alpha, \beta > 0$ guarantee the nonnegativity also of the matrix ML function. Analogously, in graph theory, circulant or centrosymmetric structures commonly appear and we have proved that these structures are always preserved by ML functions.

Some of the presented results are well known, and some others are new; this is the case, for example, of the nonnegativity of $E_{\alpha,\beta}(A)$ when A is a Metzler matrix, which is grounded on a new result on the strictly complete monotonicity of the ML functions, under suitable hypotheses on the parameters α and β. A list of features that are not preserved is also given, which may be equally useful.

Acknowledgments The author wishes to thank the anonymous referees for the accurate review of the manuscript and for the suggestions which helped to improve the presentation.

The work of M. Popolizio is partially supported by PRIN2017-MIUR project.

References

1. Arrigo, F., Durastante, F.: Mittag–Leffler functions and their applications in network science. SIAM J. Matrix Anal. Appl. **42**(4), 1581–1601 (2021)
2. Benzi, M., Bertaccini, D., Durastante, F., Simunec, I.: Non-local network dynamics via fractional graph Laplacians. J. Complex Netw. **8**(3), cnaa017 (2020)
3. Bharali, G., Holtz, O.: Functions preserving nonnegativity of matrices. SIAM J. Matrix Anal. Appl. **30**(1), 84–101 (2008)
4. Bianchi, D., Donatelli, M., Durastante, F., Mazza, M.: Compatibility, embedding and regularization of non-local random walks on graphs. J. Math. Anal. Appl. **511**(1), 126020 (2022)
5. Bini, D.A., Massei, S., Meini, B.: On functions of quasi-Toeplitz matrices. Sb. Math. **208**(11), 1628 (2017)
6. Davis, P.J.: Circulant Matrices. John Wiley&Sons, New York (1979)
7. Diaz-Diaz, F., Estrada, E.: Time and space generalized diffusion equation on graph/networks. Chaos, Solitons Fractals **156**, 111791 (2022)
8. Diethelm, K.: The analysis of fractional differential equations. Lecture Notes in Mathematics, vol. 2004. Springer, Berlin (2010)
9. Dubourdieu, M.J.: Sur un théorème de M. S. Bernstein relatif à la transformation de Laplace-Stieltjes. Comp. Math. **7**, 96–111 (1940)
10. Elagan, S.: On the invalidity of semigroup property for the Mittag–Leffler function with two parameters. J. Egypt. Math. Soc. **24**(2), 200–203 (2016)
11. Esmaeili, S., Garrappa, R.: A pseudo-spectral scheme for the approximate solution of a time-fractional diffusion equation. Int. J. Comput. Math. **92**(5), 980–994 (2015)
12. Estrada, E.: Generalized walks-based centrality measures for complex biological networks. J. Theor. Biol. **263**(4), 556–565 (2010)
13. Estrada, E.: Networks virology. Focus on Covid-19. Modeling and Simulation in Science, Engineering and Technology (2021)
14. Estrada, E., Knight, P.: A First Course in Network Theory. OUP, Oxford (2015)
15. Garrappa, R.: Exponential integrators for time–fractional partial differential equations. Eur. Phys. J. Special Topics **222**(8), 1915–1927 (2013)
16. Garrappa, R.: A family of Adams exponential integrators for fractional linear systems. Comput. Math. Appl. **66**(5), 717–727 (2013)
17. Garrappa, R., Politi, T., Popolizio, M.: Numerical approximation of the Mittag–Leffler function for large sparse low rank matrices. In preparation (2022)
18. Garrappa, R., Popolizio, M.: Computing the matrix Mittag–Leffler function with applications to fractional calculus. J. Sci. Comput. **77**(1), 129–153 (2018)
19. Golub, G., Van Loan, C.F.: Matrix Computations, 3rd edn. The Johns Hopkins University Press, Baltimore (1996)
20. Gorenflo, R., Kilbas, A.A., Mainardi, F., Rogosin, S.: Mittag–Leffler Functions. Theory and Applications. Springer Monographs in Mathematics. Springer, Berlin (2014)
21. Grindrod, P., Higham, D.J.: A dynamical systems view of network centrality. Proc. R. Soc. A: Math. Phys. Eng. Sci. **470**(2165), 20130835 (2014)
22. Higham, N.J.: Functions of matrices. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2008)
23. Higham, N.J., Mackey, D.S., Mackey, N., Tisseur, F.: Functions preserving matrix groups and iterations for the matrix square root. SIAM J. Matrix Anal. Appl. **26**(3), 849–877 (2005)
24. Holme, P., Saramoki, J.: Temporal networks. Phys. Rep. **519**(3), 97–125 (2012)

25. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press, Cambridge (1990)
26. Lopez, L., Pellegrino, S.F.: A spectral method with volume penalization for a nonlinear peridynamic model. Int. J. Numer. Methods Eng. **122**(3), 707–725 (2021)
27. Luenberger, D.G.: Introduction to Dynamic Systems: Theory, Models, and Applications. Wiley, London (1979)
28. Mainardi, F.: Fractional Calculus and Waves in Linear Viscoelasticity. Imperial College Press, London (2010)
29. Micchelli, C.A., Willoughby, R.A.: On functions which preserve the class of Stieltjes matrices. Linear Algebra Appl. **23**, 141–156 (1979)
30. Peng, J., Li, K.: A note on property of the Mittag–Leffler function. J. Math. Anal. Appl. **370**(2), 635–638 (2010)
31. Popolizio, M.: Numerical solution of multiterm fractional differential equations using the matrix Mittag–Leffler functions. Mathematics **6**(1), 7 (2018)
32. Popolizio, M.: On the matrix Mittag–Leffler function: theoretical properties and numerical computation. Mathematics **7**(12), 1140 (2019)
33. Samko, S.G., Kilbas, A.A., Marichev, O.I.: Fractional Integrals and Derivatives. Gordon and Breach Science Publishers, Yverdon (1993)
34. Varga, R.S.: Nonnegatively posed problems and completely monotonic functions. Linear Algebra Appl. **1**(3), 329–347 (1968)
35. Weaver, J.R.: Centrosymmetric (cross-symmetric) matrices, their basic properties, eigenvalues, and eigenvectors. Am. Math. Monthly **92**(10), 711–717 (1985)

On the Solutions of the Fractional Generalized Gierer–Meinhardt Model

Alessandra Jannelli and Maria Paola Speciale

Abstract In this chapter, we consider a time-fractional generalized Gierer–Meinhardt model described by a system of two coupled nonlinear time-fractional reaction–diffusion equations. Solutions are computed by applying a procedure that combines the Lie symmetry analysis with classical numerical methods. Lie symmetries reduce the target system into time-fractional coupled ordinary differential equations. The numerical solutions are determined by introducing the Caputo definition fractional derivative and by using an implicit classical numerical method. Numerical results are presented to validate the effectiveness of the proposed approach and to show, by a comparison with the integer-order case, that the fractional-order model can be considered a reliable generalization of the classical model.

1 Introduction

In this chapter, we consider a system of two coupled nonlinear time-fractional reaction–diffusion (TF–RD) equations

$$\begin{cases} \partial_t^\alpha u(t,x) = k_1 \partial_{xx} u(t,x) + f(t,x,u,v), \\ \partial_t^\alpha v(t,x) = k_2 \partial_{xx} v(t,x) + g(t,x,u,v), \end{cases} \quad 0 < \alpha < 1, \tag{1}$$

subject to suitable initial and boundary conditions. In (1), ∂_t^α is the Riemann–Liouville fractional derivative operator [1–4]

$$\partial_t^\alpha w(t,x) = \frac{1}{\Gamma(1-\alpha)} \frac{\partial}{\partial t} \int_0^t \frac{w(s,x)}{(t-s)^\alpha} ds.$$

A. Jannelli (✉) · M. P. Speciale (✉)
Department of Mathematical and Computer Sciences, Physical Sciences and Earth Sciences, University of Messina, Messina, Italy
e-mail: ajannelli@unime.it; mpspeciale@unime.it

A. Cardone et al. (eds.), *Fractional Differential Equations*,
Springer INdAM Series 50, https://doi.org/10.1007/978-981-19-7716-9_6

$u(t,x)$ and $v(t,x)$ are the field variables at time t and position x. $k_1 > 0$ and $k_2 > 0$ the diffusion coefficients of $u(t,x)$ and $v(t,x)$, respectively. The functions $f = f(t,x,u,v)$ and $g = g(t,x,u,v)$ represent the nonlinear interaction terms.

The system (1) has been proved to be a strong tool in the modeling of many physical and chemical phenomena [5–14]. Many mathematical models have been proposed in different fields of the applied sciences, and their analytical and numerical solutions have been extensively investigated. In 1972, Gierer and Meinhardt [15] proposed a prototypical depletion-type chemical model, which is described as follows:

$$\begin{cases} \partial_t u(t,x) = k_1 \partial_{xx} u(t,x) + c_1 u + d_1 \rho_1(x) \dfrac{u^p}{v^q} + \sigma_1, \\ \partial_t v(t,x) = k_2 \partial_{xx} v(t,x) + c_2 v + d_2 \rho_2(x) \dfrac{u^r}{v^s} + \sigma_2, \end{cases} \tag{2}$$

where the involved parameters are assigned constants, except for the distributions ρ_1 and ρ_2, which instead are assigned functions of the space variable x. It describes the interaction between two substances, the activator $u(t,x)$ and the inhibitor $v(t,x)$, and it is commonly used to explain the underlying complex mechanism for pattern formation in nature, describing the interaction of two sources in processes such as biological and chemical ones.

In this chapter, we propose a time-fractional model, generalization of the above classical Gierer–Meinhardt model (2), given by the fractional system (1) assuming the nonlinear interaction terms as follows:

$$\begin{aligned} f(t,x,u,v) &= c_1 u + d_1 \rho_1(x) \frac{u^p}{v^q} + \sigma_1(t,x) \\ g(t,x,u,v) &= c_2 v + d_2 \rho_2(x) \frac{u^r}{v^s} + \sigma_2(t,x) \end{aligned} \tag{3}$$

according to the source terms involved in the Gierer and Meinhardt model (2), but with $\sigma_1(t,x)$ and $\sigma_2(t,x)$ arbitrary functions of time and space. We study the mathematical model in which the fractional order involves the time derivative. Unlike other works (see, for instance, [16–20]) in which models describing space-fractional reaction–diffusion equations with anomalous diffusion process are investigated, in this work, we focus on a mathematical model in which the fractional order involves the time derivative. In particular, we analyze the model when $p = r = 2$ and $s = q = -1$, that when $\alpha = 1$ describes a depletion process where the autocatalysis is counteracted by the depletion of a substrate of the concentration $v(x,t)$ required for activation $u(t,x)$. This model was used to describe pigmentation patterns in seashells and the ontogeny of ribbing on ammonoid shells [21, 22]. The stability, the Hopf bifurcation and the Turing instability by the technique of stability theory, normal form theory, and center manifold reduction were carried out in [23, 24].

The main aim of this study is to solve the proposed fractional generalized mathematical model by applying a procedure that combines the Lie symmetry analysis with the numerical methods. It was applied to a wide class of FDES: space-fractional

advection–diffusion–reaction equations with linear [25] and nonlinear sources terms [26], a system of time-fractional advection–diffusion–reaction equations (1) with arbitrary nonlinear source terms [27] and two-dimensional time-fractional reaction–diffusion equations [28, 29].

The authors choose to approach the study of the target model initially by considering the Riemann–Liouville derivative since it is widely used in the context of the group method. The fractional Lie symmetries are determined by using a package of symbolic computing: FracSym on MAPLE platform, an algorithmic that automates the method, which is an extension of classical Lie symmetries approach to FPDEs [30, 31], of finding symmetries for FPDEs with Riemann–Liouville fractional derivative [32, 33]. By using the Lie symmetries, the target system is mapped into a system of time-fractional ordinary differential equations, and by solving the reduced FODEs, exact and numerical solutions are found. The numerical solutions are computed by introducing the Caputo definition fractional derivative and by using an implicit classical numerical method. The Caputo definition of the fractional derivative allows defining an initial value problem whose the initial conditions are given in terms of the field variable and of its integer-order derivatives, in agreement with the clear physical meaning of most of the processes arising in the real world.

We want to remark that, in this context, we are not interested in studying the stability of the model, or in the effects of diffusion on the stability of equilibrium points, or to explore the dynamical behaviors induced by diffusion or the bifurcated limit cycle from the bifurcation, topics widely studied in the specialized literature. Our aim is to find analytical and numerical solutions of the generalized mathematical model with fractional derivatives (1)–(3), in order to show the effectiveness of the proposed model and of the procedure that reveals to be a good tool for solving a wide class of fractional partial differential equations (FPDEs).

In Sect. 2, we introduce the Lie transformation that reduces the target system into time-fractional ordinary differential equations. In Sect. 3, we search for analytical solutions for the fractional generalized depletion model. In Sect. 4, by introducing the Caputo definition of the fractional derivative and by using an implicit classical trapezoidal method, the numerical solutions are found. In Sect. 5, we report concluding remarks.

2 Lie Transformation and FODEs

Analytical and numerical solutions of the mathematical model are found by applying a procedure that combines the Lie symmetry analysis with the numerical methods: by using the extension of Lie symmetries approach to FPDEs [30, 31], we find the Lie fractional symmetries admitted by the model and, then, Lie transformations that map the system of coupled nonlinear TF-RD equations (1) into a system of fractional ordinary differential equations (FODEs). By solving the reduced FODEs, we obtain solutions of the original model.

In this section, we present the Lie transformation that maps (1), with the reactions (3), into a system of two coupled FODEs. The Lie transformations of the system (1) with arbitrary sources terms were recently determined in [27] and assume the following form, according to the choice of source terms given by (3)

$$T = t, \qquad U = u(t,x)e^{-a_1 x}, \qquad V = v(t,x)e^{-a_2 x}\,, \tag{4}$$

where a_1 and a_2 are arbitrary constants and

$$\begin{aligned} \rho_1(x) &= e^{(a_1(1-p)+a_2 q)x}\rho_1' \qquad & \sigma_1(T,x) &= e^{a_1 x}h_1(T) \\ \rho_2(x) &= e^{(a_2(s+1)-a_1 r)x}\rho_2' & \sigma_2(T,x) &= e^{a_2 x}h_2(T), \end{aligned} \tag{5}$$

with ρ_1' and ρ_2' arbitrary constants and $h_1 = h_1(T)$ and $h_2 = h_2(T)$ arbitrary functions of their argument. The above transformations lead the source terms to assume the following form:

$$\begin{aligned} f(t,x,u,v) &= c_1 u + d_1 e^{(a_1(1-p)+a_2 q)x}\frac{u^p}{v^q} + h_1(t)e^{a_1 x}, \\ g(t,x,u,v) &= c_2 v + d_2 e^{(-a_1 p+a_2(q+1))x}\frac{u^p}{v^q} + h_2(t)e^{a_2 x}. \end{aligned} \tag{6}$$

Note that the functional forms of the distributions, ρ_1 and ρ_2, consistent with ones of the classical model of Gierer–Meinhardt [15], and the additional terms, σ_1 and σ_2 depending also on variables t and x, lead us to define the target model as a generalization of the model studied in [15].

When the transformation (4) and the above forms of ρ_1, ρ_2, σ_1, and σ_2 are inserted into the system (1) with sources terms given by (3), it is reduced into the following system of fractional nonlinear ordinary differential equations:

$$D_T^\alpha U - (c_1 + k_1 a_1^2)U - \frac{U^p}{V^q}d_1 - h_1 = 0 \tag{7}$$

$$D_T^\alpha V - (c_2 + k_2 a_2^2)V - \frac{U^r}{V^s}d_2 - h_2 = 0 \tag{8}$$

setting $\rho_1' = \rho_2' = 1$. The choice of the arbitrary functions h_1 and h_2 characterizes the solutions of the equations (7) and (8) and, then, the class of solutions given by (4).

In order to solve the system (7) and (8), we assume $p = r$ and $s = q$. Multiplying (7) by d_2 and (8) by $-d_1$ and, then, adding the equations, we get the following fractional ordinary differential equation (FODE):

$$D_T^\alpha(d_2 U - d_1 V) - (c_1 + k_1 a_1^2)(d_2 U - d_1 V) - (d_2 h_1 - d_1 h_2) = 0, \tag{9}$$

where we set $c_2 + k_2 a_2^2 = c_1 + k_1 a_1^2$. Under non-vanishing initial condition

$$\lim_{T \to 0} D_T^{\alpha-1}(d_2 U(T) - d_1 V(T)) = b^0, \tag{10}$$

the FODE (9) admits the following analytical solution:

$$\begin{aligned} U(T) = &\frac{d_1}{d_2} V + \frac{b^0}{d_2} T^{\alpha-1} E_{\alpha,\alpha}(\lambda T^\alpha) \\ &+ \int_0^T (T-S)^{\alpha-1} E_{\alpha,\alpha}(\lambda (T-S)^\alpha)(h_1 - \frac{d_1}{d_2} h_2) dS, \end{aligned} \tag{11}$$

where $\lambda = c_1 + k_1 a_1^2$. Substituting $U(T)$ in (8), we get the following FODE:

$$\begin{aligned} &D_T^\alpha V - \lambda V - d_2^{1-p} \\ &\frac{\left(d_1 V + b^0 T^{\alpha-1} E_{\alpha,\alpha}(\lambda T^\alpha) + \int_0^T (T-S)^{\alpha-1} E_{\alpha,\alpha}(\lambda (T-S)^\alpha))(h_1 d_2 - h_2 d_1) dS\right)^p}{V^q} \\ &- h_2 = 0, \end{aligned} \tag{12}$$

subject to the non-vanishing initial condition

$$\lim_{T \to 0} D_T^{\alpha-1} V(T) = V^0. \tag{13}$$

By solving the initial values problem (12)–(13), we find the solution $V(T)$ and, as consequence, by (11) the solution $U(T)$ and, finally, by the inverse transformations (4) the solutions $u(t, x)$ and $v(t, x)$ of the proposed system of FPDEs (1) with source terms given by (6) and with the initial conditions

$$\lim_{t \to 0} \partial_t^{\alpha-1} u(t, x) = \frac{b^0 - d_1 V^0}{d_2} e^{a_1 x}, \qquad \lim_{t \to 0} \partial_t^{\alpha-1} v(t, x) = V^0 e^{a_2 x}. \tag{14}$$

The target model describes a generalized depletion process when $p = 2$ and $q = -1$, and this will be the object of the analysis in the next section.

3 Analytical Solutions of the Generalized Depletion Model

In order to find solutions of the generalized depletion model, we choose the functions $h_1(T)$ and $h_2(T)$ in (12) by setting the parameters $b_0 = 0$ and $\lambda = 0$

so that Eq. (12) reads

$$D_T^{\alpha} V - \frac{1}{d_2}\left(d_1 V + \frac{1}{\Gamma(\alpha)}\int_0^T (T-S)^{\alpha-1}(h_1 d_2 - h_2 d_1) dS\right)^2 V - h_2 = 0. \quad (15)$$

We set

$$h_1(T) = \frac{d_1}{d_2} h_2(T) - \frac{h_0(T)}{d_2},$$

with $h_0(T)$ arbitrary function of T, and we obtain

$$D_T^{\alpha} V - \frac{1}{d_2}\left(d_1 V - \frac{1}{\Gamma(\alpha)}\int_0^T (T-S)^{\alpha-1} h_0(S) dS\right)^2 V - h_2 = 0. \quad (16)$$

The involved arbitrary function $h_2(T)$ is assigned in such a way that the FODE (16) admits analytical solutions for $\alpha = 1$.

1. We set

$$h_2(T) = Be^{-AT} \left((1-AT) - \frac{t}{d_2}\left(d_1 BTe^{-AT} - \frac{1}{\Gamma(\alpha)}\int_0^T (T-S)^{\alpha-1} h_0(S) dS\right)^2\right) \quad (17)$$

with A and B arbitrary constants. With the above choice of the function $h_2(T)$ and when $\alpha = 1$, we get the following exact solution;

$$V(T) = BTe^{-AT},$$

and substituting it in (11) we obtain

$$U(T) = \frac{1}{d_2}\left(d_1 BTe^{-AT} - \int_0^T h_0(S) dS\right).$$

By means of the inverse transformations (4), we get the exact solution of the generalized depletion Gierer–Meinhardt model

$$u(t,x) = e^{a_1 x}\frac{1}{d_2}\left(d_1 Bte^{-At} - \int_0^t h_0(s) ds\right),$$
$$v(t,x) = e^{a_2 x} Bte^{-At}.$$

2. We set

$$h_2(T) = \frac{A}{\cosh^2(T)} + \frac{1}{d_2}(B + A\tanh(T))$$
$$\left(d_1(B + A\tanh(T)) - \frac{1}{\Gamma(\alpha)}\int_0^T (T-S)^{\alpha-1}h_0(S)dS\right)^2. \quad (18)$$

By the above choice of function h_2 and when $\alpha = 1$, we get the following exact solution:

$$V(T) = B + A\tanh(T),$$

and substituting it in (11), we obtain

$$U(T) = \frac{1}{d_2}\left(d_1(B + A\tanh(T)) - \int_0^T h_0(S)dS\right).$$

By means of the inverse transformations (4), we get the exact solution of the generalized depletion Gierer–Meinhardt model

$$u(t,x) = e^{a_1 x}\frac{1}{d_2}\left(d_1(B + A\tanh(t)) - \int_0^t h_0(s)ds\right),$$
$$v(t,x) = e^{a_2 x}(B + A\tanh(t)).$$

In the next section, starting from the expressions of $h_1(T)$ and $h_2(T)$ reported in the above examples, and setting $h_0(T)$, we get numerical solutions of the fractional generalized depletion model.

4 Numerical Method and Solutions

In this section, we find the numerical solutions of the system of FPDEs (1) computed by solving the FODE (16), obtained in the above section by a suitable choice of the involved parameters and functions and where the function $h_2(T)$ is given by (17) or (18). Computed the numerical solution $V(T)$, we obtain the numerical solution $U(T)$ by (11) and then the solutions $u(t,x)$ and $v(t,x)$ of the target model (1) by the inverse transformations by (4).

We introduce the α-order Caputo fractional derivative of the solution $V(T)$

$${}^C D_T^\alpha V(T) = \frac{1}{\Gamma(1-\alpha)}\int_0^T (T-S)^{-\alpha}V'(S)dS$$

and its connection with the α-order Riemann–Liouville fractional derivative [1]

$$ {}^{C}D_T^{\alpha} V(T) = D_T^{\alpha} \left(V(T) - V(0) \right), $$

with $V(0)$ initial condition. In terms of the Caputo derivative, the following fractional initial value problem (FIVP) is obtained:

$$ \begin{aligned} &{}^{C}D_T^{\alpha} V(T) = F(T, V), \quad 0 < \alpha < 1 \\ &V(0) = V^0, \end{aligned} \tag{19} $$

where

$$ F(T, V) = -\frac{V^0}{\Gamma(1-\alpha) T^{\alpha}} + \frac{1}{d_2} \left(d_1 V - \frac{1}{\Gamma(\alpha)} \int_0^T (T - S)^{\alpha - 1} h_0(S) dS \right)^2 V + h_2 $$

and

$$ h_0(T) = \frac{h}{(1 + T^2)^{1.5}} \tag{20} $$

with h arbitrary constant. We remark that the Caputo definition of the fractional derivative allows defining an FIVP whose the initial conditions are given in terms of the field variable and of its integer-order derivatives, in agreement with the clear physical meaning of most of the processes arising in the real world.

In order to numerically solve the FIVP (19), we propose the classical implicit trapezoidal method (PI_2 Im), an efficient numerical method used for its good stability and accuracy properties [4, 34], numerical results by multi-steps methods are presented in [18]. We build a computational uniform mesh of $N + 1$ grid points denoted by T^n, with $T^n = T^0 + n\Delta T$ and integration step sizes ΔT and N positive integer. We denote by U^n the numerical approximation provided by the numerical method of the exact solution $U(T^n)$ at the mesh points T^n, for $n = 0, \cdots, N$. The numerical method reads

$$ V^{n+1} = V^0 + \frac{1}{\Gamma(\alpha)} \left(\beta_0 F(T^0, V^0) + \sum_{k=1}^{n+1} \beta_k F(T^k, V^k) \right), \tag{21} $$

where the coefficient values β_k, for $k = 0, \cdots, n + 1$, are computed as follows:

$$ \beta_0 = \frac{1}{\alpha(\alpha + 1)} \frac{(T^{n+1})^{\alpha} ((T^1 - T^0)(\alpha + 1) - T^{n+1}) + (T^{n+1} - T^1)^{\alpha + 1}}{T^1 - T^0}, $$

$$ \beta_k = \frac{1}{\alpha(\alpha + 1)} \times \qquad k = 1, \cdots, n, $$

$$\times \left(\frac{(T^{n+1} - T^{k-1})^{\alpha+1} - (T^{n+1} - T^{k})^{\alpha+1}}{T^{k} - T^{k-1}} \right.$$
$$\left. - \frac{(T^{n+1} - T^{k})^{\alpha+1} - (T^{n+1} - T^{k+1})^{\alpha+1}}{T^{k+1} - T^{k}} \right),$$

$$\beta_{n+1} = \frac{1}{\alpha(\alpha+1)} (T^{n+1} - T^{n})^{\alpha} .$$

The convergence order of the scheme is $O((\Delta T)^{min(1+\alpha,2)})$. Note that the convergence order of the trapezoidal method usually is $1+\alpha$ when $0 < \alpha < 1$. Only when $\alpha > 1$ or when the solution is sufficiently smooth, we obtain the reached order 2 [34]. In general, the numerical method (21) leads to obtain a nonlinear equation at each step for whose resolution a root-finding solver is needed. The classical Newton method is proposed.

In the following, we report two test problems characterized by the function $h_2(T)$, given by (17) or (18). Different simulation parameters, involved functions, and α parameter values are considered as the input features of the model under study. The exact solutions for $\alpha = 1$, reported in Sect. 3, are considered as reference ones for testing qualitative behavior of the numerical solutions of the models for values of the α parameter increasing toward 1. All numerical simulations are performed on Intel Core i7 by using Matlab 2020 software.

Example 1 In this example, we consider the FIVP (19) with $V^0 = 0$ and

$$h_2(T) = Be^{-AT} \left((1 - AT) - \frac{T}{d_2} \left(d_1 BTe^{-AT} - \frac{1}{\Gamma(\alpha)} \int_0^T \frac{h(T-S)^{\alpha-1}}{(1+S^2)^{1.5}} dS \right)^2 \right).$$

For $\alpha = 1$, the analytical solution $V(T)$ is given by

$$V(T) = BTe^{-AT}, \tag{22}$$

and then, we are able to compute the analytical solution $U(T)$

$$U(T) = \frac{1}{d_2} \left(d_1 BTe^{-AT} - \frac{hT}{\sqrt{1+T^2}} \right). \tag{23}$$

We choose a computational domain $[0, T_{max}]$ with $T_{max} = 10$ and $N = 100$ grid points and set the parameters values as follows: $d_1 = 0.00016$, $d_2 = -0.00048$, $A = 2$, $B = 0.02$, and $h = 0.0001$. We remark that, in this example, the parameters values are chosen according to the values by Gierer and Meinhardt reported in [15], values ensuring that the model is a well-posed one, with the aim to validate our numerical results obtained by the proposed procedure.

In Fig. 1, we report the numerical results obtained for different values of the α parameter: in the left frame, the numerical solution V^n obtained by solving the FIVP

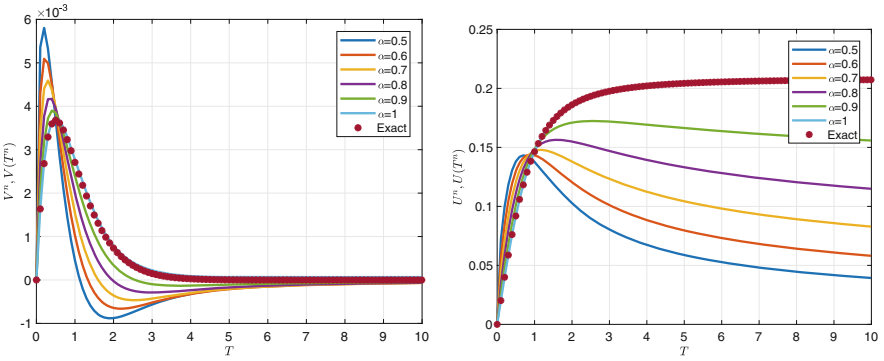


Fig. 1 Numerical solutions for different values of α. Left frame: numerical solution V^n of the FIVP (19). Right frame: numerical solution U^n computed by (11)

(19) by using the PI_2 Im numerical method, and in the right frame, the numerical solution U^n computed by means of (11) by the suitable substitution of the involved functions and parameters

$$U^n = \frac{1}{d_2}\left(d_1 V^n - T\ {}_3F_2([0.5, 1, 1.5]; [(1+\alpha)/2, 1+\alpha/2]\,; -(T^n)^2)\right)$$

and given in terms of the generalized hypergeometric function ${}_pF_q(a; b; z)$ of order p, q defined as follows:

$$\begin{aligned} {}_pF_q(a; b; z) &= {}_pF_q([a_1, a_2, \ldots, a_p]; [b_1, b_2, \ldots, b_q]; z) \\ &= \sum_{n=0}^{\infty} \left(\frac{(a_1)_n, (a_2)_n, \ldots, (a_p)_n}{(b_1)_n, (b_2)_n, \ldots, (b_q)_n}\right)\left(\frac{z^2}{n!}\right) \end{aligned}$$

with $a = [a_1, a_2, \ldots, a_p]$ and $b = [b_1, b_2, \ldots, b_q]$ vectors of lengths p and q, respectively. $(a)_k$ and $(b)_k$ represent the Pochhammer symbols defined in the following way:

$$(w)_m = \frac{\Gamma(w+m)}{\Gamma(w)}.$$

The red dot points represent the exact solutions $V(T)$, given by (22), and $U(T)$, given by (23), of the model with $\alpha = 1$. They are reported in order to test the qualitative behavior of the numerical solutions as the α parameter increases toward 1. The solutions reveal very different behavior: both start from the value 0, and both increase at the beginning of the integration process. After, as time evolves, the solution $U(T)$ increases and converges to a constant state. The solution $V(T)$ decreases and converges to zero. Note that, as the value of α increases, for $T < 1$ both solutions decrease and, then, for $T > 1$ increase. The qualitative behavior of

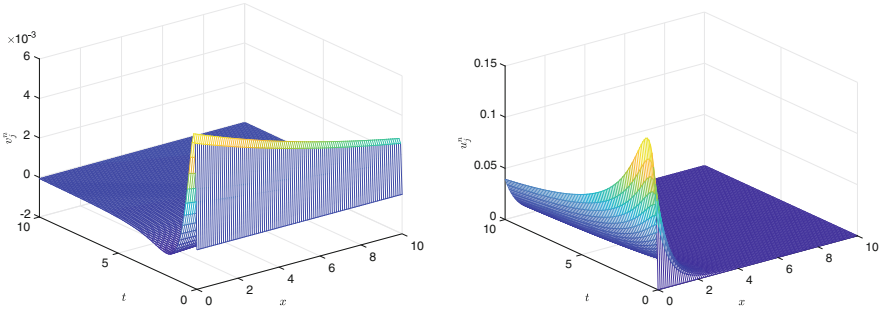


Fig. 2 Numerical solutions of the system of FPDE (1) for $\alpha = 0.5$. Left frame: numerical solution v_j^n. Right frame: numerical solution u_j^n

the solutions is in agreement with one of the classical depletion models. This means that the proposed mathematical model can represent a good generalization of the classical model.

In Fig. 2, we report the numerical solutions u_j^n and v_j^n, approximations of the exact solutions obtained by the inverse transformations (4)

$$u(t,x) = U(t)e^{a_1 x}, \qquad v(t,x) = V(t)e^{a_2 x},$$

solutions of the model (1) integrated with the initial and boundary conditions obtained by (14) with $V^0 = 0$ and $b^0 = 0$ and computed for $\alpha = 0.5$. In this contest, the source terms assume the following form:

$$f(t,x,u,v) = c_1 u + d_1 e^{-(a_1+a_2)x} uv^2 + h_1 e^{a_1 x},$$
$$g(t,x,u,v) = c_2 v + d_2 e^{-2a_1 x} uv^2 + h_2 e^{a_2 x}.$$

The parameters are chosen in the following way: $a_1 = -\sqrt{-c_1/k_1}$, $a_2 = -\sqrt{-c_2/k_2}$ with $k_1 = 10^{-3}$, $k_2 = 0.45$, and $c_1 = c_2 = -0.0025$. The computational domain is given by $[0, T_{max}] \times [0, X_{max}]$ with $T_{max} = 10$, $X_{max} = 10$, and $N = J = 100$ grid points.

Example 2 In this example, we consider the FIVP (19) with $V^0 = 0$ and

$$h_2(T) = \frac{A}{\cosh^2(T)} + \frac{1}{d_2}(B + A\tanh(T)) \left(d_1(B + A\tanh(T)) - \frac{1}{\Gamma(\alpha)} \int_0^T \frac{h(T-S)^{\alpha-1}}{(1+S^2)^{1.5}} dS \right)^2.$$

For $\alpha = 1$, the analytical solution of the FIVP is given by

$$V(T) = B + A\tanh(T), \tag{24}$$

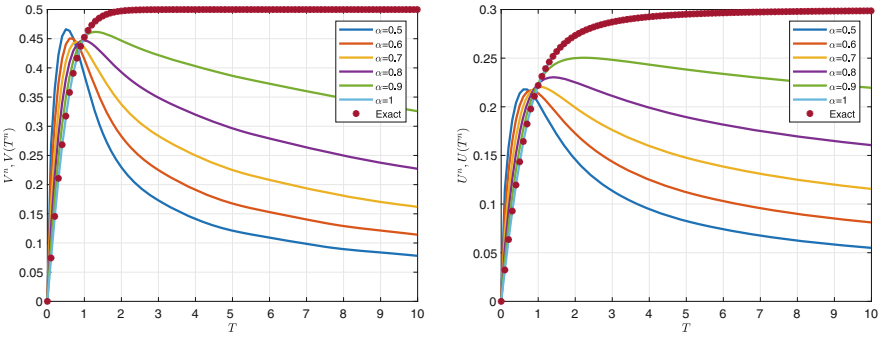


Fig. 3 Numerical solutions for different values of α. Left frame: Numerical solution V^n of the FIVP (19). Right frame: Numerical solution U^n computed by (11)

and then, we are able to compute the analytical solution $U(T)$

$$U(T) = \frac{1}{d_2}\left(B + A\tanh(T) - \frac{hT}{\sqrt{1+T^2}}\right). \tag{25}$$

We set the parameters values as follows: $d_1 = 0.1$, $d_2 = 1$, $A = -0.5$, $B = 0$, and $h = 0.25$. We choose a computational domain $[0, T_{max}]$ with $T_{max} = 10$ and $N = 100$ grid points. In Fig. 3, we report the numerical results obtained for different values of the α parameter: in the left frame, the numerical solution V^n obtained by solving the FIVP (19) by using the PI_2 Im numerical method; in the right frame, the numerical solution U^n computed by means of (11) in terms of the generalized hypergeometric function. The red dot points represent the exact solution $V(T)$, given by (24), and the exact solution $U(T)$, given by (25), of the model with $\alpha = 1$. In this case, the solutions reveal very similar behavior: both start from the value 0, and both increase at the beginning of the integration process. After, as time evolves, both decrease and converge to a constant state. Note that, as the value of α increases, for $T < 1$ both solutions decrease and, then, for $T > 1$ increase. The qualitative behavior of the solutions is in agreement with the choice of the distributions which are both positive functions of x; in fact, the parameters d_1 and d_2 agree in the sign.

In Fig. 4, we report the numerical solutions u_j^n and v_j^n, approximations of the exact solutions obtained by the inverse transformations (4). The source terms are assigned as before. The parameters are chosen in the following way: $a_1 = -\sqrt{-c_1/k_1}$, $a_2 = -\sqrt{-c_2/k_2}$ with $k_1 = 1$, $k_2 = 10$, and $c_1 = c_2 = -0.2$. The computational domain is given by $[0, T_{max}] \times [0, X_{max}]$ with $T_{max} = 10$, $X_{max} = 10$ and $N = J = 100$ grid points.

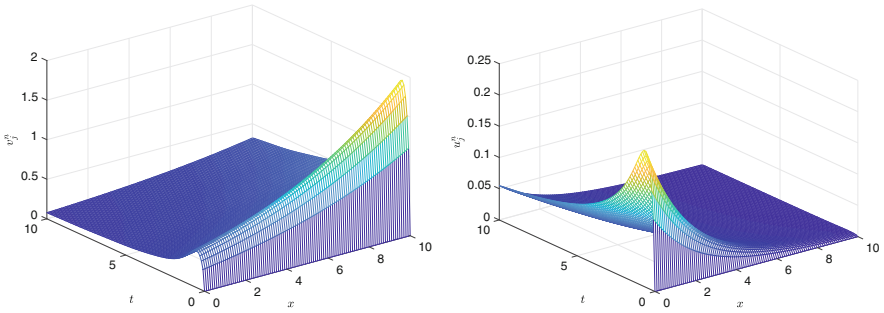


Fig. 4 Numerical solutions of the system of FPDE (1) for $\alpha = 0.5$. Left frame: numerical solution v_j^n. Right frame: numerical solution u_j^n

5 Concluding Remarks

In this study, analytical and numerical solutions for the generalized Gierer–Meinhardt fractional model are presented. We propose a mathematical model governed by a system of two time-fractional diffusion–reaction equations describing the interaction between two chemical substances, commonly used to explain the underlying complex mechanism for pattern formation in nature. The numerical results, obtained by applying the procedure that combines the Lie symmetry analysis with the trapezoidal numerical method, reveal the effectiveness of the proposed model. Moreover, it is important to note that some analytical solutions of the generalized Gierer–Meinhardt model of integer order are found for a suitable choice of the involved parameters and functions. The topic of the next study is the spatial-fractional reaction–diffusion equations with an anomalous diffusion process that occurs in spatially inhomogeneous media.

Acknowledgments A. J. is a member of GNCS-INdAM Research Group. M.P. S. is a member of GNFM-INdAM Research Group.

References

1. Podlubny, I.: Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, Some Methods of Their Solution and Some of Their Applications. Academic Press, San Diego (1999)
2. Kilbas, A.A., Srivastava, H.M., Trujillo, J.J.: Theory and Applications of Fractional Differential Equations. North-Holland Mathematics Studies. Elsevier (2006)
3. Samko, S., Kilbas, A.A., Marichev, O.: Fractional Integrals and Derivatives. Taylor and Francis (1993)
4. Diethelm, K.: The Analysis of Fractional Differential Equations. Springer, Berlin (2004)
5. Henry, B.I., Wearne, S.L.: Fractional reaction-diffusion. Physica A, **276**, 448–455 (2000)

6. Seki, K., Wojcik, M., Tachiya, M.: Fractional reaction-diffusion equation. J. Chem. Phys. **119**, 2165 (2003)
7. Gafiychuk, V., Datsko, B.: Pattern formation in a fractional reaction-diffusion system. Phys. A Stat. Mech. Appl. **365**, 300–306 (2006)
8. Gafiychuk, V.V., Datsko, B.Y.: Spatio-temporal pattern formation in fractional reaction-diffusion systems with indices of different order. Phys. Rev. E **77**(6), 066210 (2008)
9. Gafiychuk, V., Datsko, B.: Different types of instabilities and complex dynamics in reaction-diffusion systems with fractional derivatives. J. Comput. Nonlin. Dyn. **7**, 031001 (2012)
10. Datsko, B., Gafiychuk, V.: Complex nonlinear dynamics in subdiffusive activator-inhibitor systems. Commun. Nonlinear Sci. Numer. Simul. **17**, 1673–1680 (2012)
11. Nec, Y.: Explicitly solvable eigenvalue problem and bifurcation delay in sub-diffusive Gierer-Meinhardt model. Eur. J. Appl. Math. **27**(5), 699–725 (2016)
12. Nec, Y.: Spike solutions in Gierer#x2013; Meinhardt model with a time dependent anomaly exponent. Commun. Nonlinear Sci. Numer. Simul. **54**, 267–285 (2018)
13. Jannelli, A.: Numerical solutions of fractional differential equations arising in engineering sciences. Mathematics **8**, 215 (2020)
14. Jannelli, A.: Adaptive numerical solutions of time-fractional advection-diffusion-reaction equations. Commun. Nonlinear. Sci. Numer. Simul. **105**, 106073 (2022)
15. Gierer, A., Meinhardt, H.: A theory of biological pattern formation. Kybernetik **12**, 30–39 (1972); Biological Cybernetics **12**(1), 30–39 (1972). https://doi.org/10.1007/BF00289234
16. Henry, B.I., Wearne, S.L.: Existence of turing instabilities in a two-species fractional reaction-diffusion system. SIAM J. Appl. Math. **62**(3), 870–887 (2002)
17. Henry, B.I., Langlands, T.A.M., Wearne, S.L.: Turing pattern formation in fractional activator-inhibitor systems. Phys. Rev. E Stat. Nonlinear Soft Matter Phys. **72**(2), 026101 (2005)
18. Guo, L., Zeng, F., Turner, I., Burrage, K., Karniadakis, G.E.M.: Efficient multistep methods for tempered fractional calculus: Algorithms and simulations. SIAM J. Sci. Comput. **41**(4), 2510–2535 (2019)
19. Yu, W., Rongpei, Z., Zhen, Zijian, H.: Turing pattern in the fractional Gierer-Meinhardt model. Chin. Phys. B **28**(5), 050503 (2019)
20. Wei, J., Yang, W.: Multi-bump Ground states of the fractional Gierer-Meinhardt system on the real line. J. Dyn. Differ. Equ. **31**, 385–417 (2019)
21. Meinhardt, H., Klingler, M.: A model for pattern formation on the shells of molluscs. J. Theor. Biol. **126**, 63–89 (1987)
22. Buceta, J., Lindenberg, K.: Switching-induced Turing instability. Phys. Rev. E **66**, 046202 (2002)
23. Wu, R., Shao, Y., Zhou, Y., Chen, L.: Turing and Hopf bifurcation of Gierer-Meinhardt activator-substrate model. Electron. J. Differ. Equ. **2017**(173), 1–19 (2017)
24. Chen, L., Wu, R., Xu, Y.: Dynamics of a depletion-type Gierer-Meinhardt model with Langmuir-Hinshelwood reaction scheme. Discrete Contin. Dyn. Syst. B (2020). https://doi.org/10.3934/dcdsb.2021132
25. Jannelli, A., Ruggieri, M., Speciale, M.P.: Numerical solutions of space fractional advection-diffusion equation with source term. AIP Confer. Proc. **2116**, 280007 (2019)
26. Jannelli, A., Ruggieri, M., Speciale, M.P.: Numerical solutions of space fractional advection–diffusion equation, with nonlinear source term. Appl. Num. Math. **155**, 93–102 (2020)
27. Jannelli, A., Speciale, M.P.: On the numerical solutions of coupled nonlinear time-fractional reaction–diffusion equations. AIMS Math. **6**(8), 9109–9125 (2021)
28. Jannelli, A., Speciale, M.P.: Exact and numerical solutions of two-dimensional time–fractional diffusion-reaction equation through the lie symmetries. Nonlinear Dynamics **105**, 2375–2385 (2021)
29. Jannelli, A., Speciale, M.P.: Comparison between Solutions of two-dimensional time-fractional diffusion-reaction equation through the lie symmetries. Atti della Accademia Peloritana dei Pericolanti **99**, A4 (2021)
30. Gazizov, R.K., Kasatkin, A.A., Lukashchuk, S.Y.: Continuous transformation groups of fractional differential equations, Vestn. USATU **9**, 125–35 (2007)

31. Gazizov, R.K., Kasatkin, A.A., Lukashchuk, S.Y.: Group-invariant solutions of fractional differential equations. Nonlinear Sci. Complex. 51–59 (2011)
32. Vu, K.T., Jefferson, G.F., Carminati, J.: Finding generalized symmetries of differential equations using the MAPLE package DESOLVII. Comput. Phys. Commun. **183**, 1044–1054 (2012)
33. Jefferson, G.F., Carminati, J.: ASP: Automated symbolic computation of approximate symmetries of differential equations. Comput. Phys. Commun. **184**, 1045–1063 (2013)
34. Garrappa, R.: Trapezoidal methods for fractional differential equations: Theoretical and computational aspects. Math. Comput. Simul. **110**, 96–112 (2015)

A Convolution-Based Method for an Integro-Differential Equation in Mechanics

Sabrina Francesca Pellegrino

Abstract Peridynamics is a second order in time partial integro-differential equation introduced to study elastodynamic problems in the nonlocal framework. In this chapter, we focus on the nonlinear problem in both a two-dimensional and a one-dimensional spatial domain. We consider a spectral method based on the Fourier polynomials for the space discretization and make both an analytical and a numerical comparison between the Störmer–Verlet and the Newmark-β methods for time marching. A volume penalization technique is also proposed to overcome the limitation of periodic boundary condition related to the implementation of the Fourier expansion. We prove the convergence and the stability of the fully discrete linear problem. We perform some simulations to validate our results.

1 Introduction

Studying fracture problems is a difficult issue as classical theory of continuum mechanics is not able to model cracks and defects since partial derivatives do not exist on discontinuities. A strategy to treat such issue is to move in the framework of nonlocal theory where transmission conditions at crack boundaries are implicitly included in the equations (see for instance [3, 6, 8, 12, 23]). In particular, in [26], Silling introduces a nonlocal theory, called peridynamics, to treat discontinuous problems without using partial derivatives. The peridynamic equation consists in a second order in time partial integro-differential equation, where the unknown is the displacement field. The model describes the interactions between material particles, which distance each other less than a critical value $\delta > 0$ called *horizon* (see [9, 15, 28]).

S. F. Pellegrino (✉)
Dipartimento di Ingegneria Elettrica e dell'Informazione, Politecnico di Bari, Bari, Italy
e-mail: sabrinafrancesca.pellegrino@poliba.it

A. Cardone et al. (eds.), *Fractional Differential Equations*,
Springer INdAM Series 50, https://doi.org/10.1007/978-981-19-7716-9_7

Let $[0, T]$ be the time interval and $\Omega \subset \mathbb{R}^2$ be the spatial domain; then the peridynamic equation is given by

$$\rho \partial_{tt}^2 u(x,t) = \int_{\Omega \cap B_\delta(x)} f(x' - x, u(x',t) - u(x,t)) dx', \tag{1}$$

with initial conditions

$$u(x,0) = u_0(x), \quad \partial_t u(x,0) = v_0(x), \qquad x \in \Omega, \tag{2}$$

where u is the displacement field, ρ is the mass density, b collects all the external forces acting on the material body, and $B_\delta(x) = \{x' \in \mathbb{R}^2 : \|x - x'\| \leq \delta\}$. The integrand f is a response function that describes the interaction force among two material particles and is called *pairwise force function*.

The horizon δ can be thought as a measure of the nonlocality of the problem; indeed, peridynamic theory converges to the classical one when δ approaches zero (see [10]).

In this chapter, we study the nonlinear problem in two spatial dimensions where the pairwise force function is of convolution type in separable form

$$f(\xi, \eta) = C(\xi) w(\eta), \tag{3}$$

with

$$w(\eta) = \eta^r, \quad r \text{ odd}, \quad r \geq 1, \tag{4}$$

where the variables ξ and η represent, respectively, the relative position of two points and their relative displacement:

$$\xi = x' - x, \qquad \eta = u(x',t) - u(x,t).$$

Moreover, in (3), the function C is a non-negative even function, i.e., $C(-\xi) = C(\xi)$, called **micromodulus function** and w is an odd function satisfying the Lipschitz property:

$$|w(\eta') - w(\eta)| \leq \ell(\xi) |\eta' - \eta|,$$

for some non-negative function $\ell \in L^1(B_\delta(0)) \cap L^\infty(B_\delta(0))$ such that for all $\xi \in \mathbb{R}^2$, with $|\xi| \leq \delta$ and for every η, η'.

We define the nonlinear peridynamic operator of (1) as

$$\mathcal{L}(u(x,t)) = \int_{\Omega \cap B_\delta(x)} C(x' - x) \left(u(x',t) - u(x,t)\right)^r dx', \quad x \in \Omega, \quad 0 \leq t \leq T. \tag{5}$$

Hence, the peridynamic model becomes

$$\begin{cases} \rho u_{tt}(x,t) = \mathcal{L}(u(x,t)) + b(x,t), & x \in \Omega, \quad t \in [0,T], \\ u(x,0) = u_0(x), \quad v(x,0) = v_0(x), & x \in \Omega, \end{cases} \tag{6}$$

where $v(x,t) = u_t(x,t)$.

The choice of w in the form given by (4) is due to the fact that this kind of nonlinearity is correlated to the definition of a fractional derivative (see [5, 13, 19, 20]). This aspect and the definition of the pairwise force function f allow us to exploit the properties of the convolution product, so that, as we will see below, the implementation of the Fourier spectral method would transform the convolution product in the physical space into a multiplication in the frequency space. As a consequence, we can obtain a reduction of the total computational cost. Additionally, in this setting, the well-posedness holds (see [5, 11]). We can also notice that when $r = 1$, we find the linear case analyzed in [7, 24].

The spatial discretization of (6) is usually based on the implementation of quadrature formula, finite element methods, and meshfree methods (see [1, 2, 7, 24, 25, 27]). Even spectral techniques based on the Fourier trigonometric polynomials are suitable in peridynamic setting, as they exploit the convolution properties of the peridynamic operator $\mathcal{L}$, so they are able to reduce the computational cost by using the fast Fourier transform (see [7, 16–20]). However, the Fourier spectral method requires the assumption of periodic boundary condition. This restriction can be removed in different ways: for instance, by substituting the Fourier polynomials by the Chebyshev ones (see [21, 22]), or otherwise, by associating the spectral method to a volume penalization technique (see [16, 19, 20]).

On the other hand, both explicit and implicit methods can be applied to integrate in time the model. Here, we focus on the Störmer–Verlet method and the Newmark-β method. Both the methods are of the second order: the first one is an explicit scheme often used in the wave propagation context, while the second one depends on a parameter β that takes into account the acceleration of the system, is implicit and unconditionally stable in time for $\beta \in [1/4, 1/2]$.

In this chapter, we summarize an approach for the spatial discretization of the model based on the Fourier spectral method proposed in [19, 20]. We work on spatial domain of the form $\Omega = [a,b] \times [a,b]$ and consider a Fourier spectral method with a volume penalization technique for the spatial discretization, and we make a comparison between the Störmer–Verlet and the Newmark-β scheme. An extension to more general two-dimensional domains is possible following the results presented in [4, 14].

This chapter is organized as follows. In Sect. 2, we discretize the peridynamic model (6) by means of the Fourier polynomials, and we prove the convergence of the semi-discrete scheme. Section 3 describes the volume penalization technique. In Sect. 4, we present the Störmer–Verlet and the Newmark-β methods, and we show the convergence and the stability of the fully discrete scheme based on the

Newmark-β method for the linear peridynamic problem. Section 5 is devoted to the simulations, and finally, Sect. 6 concludes the paper.

2 Fourier Semi-Discretization of the Problem

The spectral discretization of the problem (6) is based on the use of the Fourier trigonometric polynomials and requires the assumption of periodic solution.

Let $\delta > 0$ be the horizon and $\Delta x > 0$ be the space step in both directions. We discretize the spatial domain $\Omega = [a, b] \times [a, b]$ by the collocation points $x_n = \left(x_{n_1},\ x_{n_2}\right) \in \Omega$, with $n = (n_1,\ n_2)$, such that

$$x_{n_1} = a + n_1 \Delta x, \quad x_{n_2} = a + n_2 \Delta x, \quad \text{for} \quad n_1,\ n_2 \in \{0, \ldots, N\},$$

where $N = \left\lfloor \frac{b-a}{\Delta x} \right\rfloor$.

The definition of periodic convolution product allows us to rewrite the peridynamic operator (5) as

$$\begin{aligned} \mathcal{L}\left(u(x,t)\right) =& (C *_\Omega u^r)(x,t) + \sum_{\ell=1}^{r-1} \binom{r}{\ell} (-1)^\ell u^\ell(x,t) \left(C *_\Omega u^{r-\ell}\right)(x,t) \\ & - \gamma u^r(x,t) + b(x,t), \qquad x \in \Omega, \quad t \in [0,T], \end{aligned} \tag{7}$$

where $\gamma = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} C(x)\, dx$.

We approximate $u(x,t)$ by the Fourier trigonometric polynomials $u^N(x,t)$

$$u^N(x,t) = u^N(x_1, x_2, t) = \sum_{k_1 \le N} \sum_{k_2 \le N} \tilde{u}(k_1, k_2, t) e^{\Im(k_1 x_1 + k_2 x_2)}, \tag{8}$$

where $x = (x_1, x_2) \in \Omega$, $t \in [0,T]$, $\Im$ denotes the imaginary unit $\Im = \sqrt{-1}$, and $\tilde{u}(k_1, k_2, t)$ for every $k = (k_1, k_2)$ represents the discrete 2D Fourier transform

$$\tilde{u}(k_1, k_2, t) = \frac{1}{(N+1)^2 c_{k_1} c_{k_2}} \sum_{n_1=0}^{N} \sum_{n_2=0}^{N} u^N(x_{n_1}, x_{n_2}, t) e^{-\Im(k_1 x_{n_1} + k_2 x_{n_2})}, \quad t \in [0,T], \tag{9}$$

with

$$c_{k_i} = \begin{cases} 2, & \text{if } k_i = \pm N, \\ 1, & \text{otherwise}, \end{cases} \qquad i = 1,\ 2.$$

We substitute $u(x,t)$ in (6) by $u^N(x,t)$ defined in (8). Then, since in the expression of the peridynamic operator $\mathcal{L}$ in (7) a convolution product appears, we can use the convolution theorem to approximate the model (6) at x_n by the **Fourier collocation method**:

$$\begin{aligned}\rho\partial_{tt}^2 u_n^N(t) &= \left(C_n *_\Omega (u_n^N)^r\right)(t) + \sum_{\ell=1}^{r-1}\binom{r}{\ell}(-1)^\ell (u_n^N)^\ell(t)\left(C_n *_\Omega (u_n^N)^{r-\ell}\right)(t) \\ &\quad - \gamma (u_n^N)^r(t) + b_n^N(t), \\ &= \left(\mathcal{F}_N^{-1}\left(\mathcal{F}_N(C)\mathcal{F}_N\left((u_n^N)^r\right)(\Delta x)^2\right)\right) \\ &\quad + \left(\sum_{\ell=1}^{r-1}\binom{r}{\ell}(-1)^\ell \mathcal{F}_N^{-1}\left(\mathcal{F}_N\left((u_n^N)^\ell\right) *_\Omega \left(\mathcal{F}_N(C)\mathcal{F}_N\left((u_n^N)^{r-\ell}\right)(\Delta x)^2\right)\right)\right) \\ &\quad - \gamma (u_n^N)^r + b_n^N,\end{aligned} \tag{10}$$

where $\mathcal{F}_N$ and $\mathcal{F}_N^{-1}$ denote the 2D discrete Fourier transform and the 2D inverse discrete Fourier transform, respectively. Additionally, $u_n^N(t) = u^N(x_n,t)$, $C_n(t) = C(x_n,t)$, for $n=(n_1,n_2)$, with $n_1,\ n_2 \in \{0,\dots,N\}$, $t\in[0,T]$, and so on, and

$$u_0(x_n) = \sum_{|k|\le N} \tilde{u}_{0,k} e^{\Im k\cdot x_n}.$$

We denote by $\mathcal{L}_N$ the discrete counterpart of the peridynamic operator $\mathcal{L}$:

$$\begin{aligned}\mathcal{L}_N(u_n^N)(t) &= \left(\mathcal{F}_N^{-1}\left(\mathcal{F}_N(C)\mathcal{F}_N\left((u_n^N)^r\right)(\Delta x)^2\right)\right) \\ &\quad + \left(\sum_{\ell=1}^{r-1}\binom{r}{\ell}(-1)^\ell \mathcal{F}_N^{-1}\left(\mathcal{F}_N\left((u_n^N)^\ell\right) *_\Omega \left(\mathcal{F}_N(C)\mathcal{F}_N\left((u_n^N)^{r-\ell}\right)(\Delta x)^2\right)\right)\right) \\ &\quad - \gamma (u_n^N)^r,\end{aligned} \tag{11}$$

for $n=(n_1,n_2)$, with $n_1,n_2\in\{0,\dots,N\}$.

Then, the spectral semi-discrete method (10) becomes

$$\begin{cases}\rho\partial_{tt}^2 u_n^N(t) = \mathcal{L}_N(u_n^N)(t) + b_n^N(t), & t\in[0,T], \\ u_n^N(0) = u_{n,0}, \quad v_n^N(0) = v_{n,0}, & \\ n=(n_1,n_2),\ \text{with } n_1,\ n_2\in\{0,\dots,N\}, & \end{cases} \tag{12}$$

where $u_{n,0} = u_0(x_n)$ and $v_{n,0} = v_0(x_n)$.

Remark 1 Using the same argument as before, one can find the spectral semi-discretization in the one-dimensional case:

$$\begin{aligned}\frac{d^2}{dt^2}u_n^N = &\left(\mathcal{F}_N^{-1}\left(\mathcal{F}_N(C)\mathcal{F}_N\left((u_n^N)^r\right)\Delta x\right)\right) \\ &+\left(\sum_{\ell=1}^{r-1}\binom{r}{\ell}(-1)^\ell \mathcal{F}_N^{-1}\left(\mathcal{F}_N\left((u_n^N)^\ell\right) *_\Omega \left(\mathcal{F}_N(C)\mathcal{F}_N\left((u_n^N)^{r-\ell}\right)\Delta x\right)\right)\right) \\ &-\gamma(u_n^N)^r + b_n^N,\end{aligned} \tag{13}$$

with initial condition

$$u_n^N(0) = u_0(x_n) = \sum_{k=-N}^{N} \tilde{u}_{0,k} e^{\Im k x_n}, \qquad v_n^N(0) = v_0(x_n) = \sum_{k=-N}^{N} \tilde{v}_{0,k} e^{\Im k x_n}$$

for $n \in \{0, \dots, N\}$. Here, with abuse of notation, $\mathcal{F}_N$ and $\mathcal{F}_N^{-1}$ represent the 1D discrete Fourier transform and the 1D inverse discrete Fourier transform, respectively.

In order to reduce the computational cost associated to the evaluation of the discrete peridynamic operator $\mathcal{L}_N$, we use the fast Fourier transform (FFT) to compute the discrete Fourier transform $\mathcal{F}_N$.

Now, we show the convergence of the semi-discrete scheme (12) proved in [19]. While to apply the method to more general problems with non-periodic boundary conditions, volume penalization techniques can be exploited as shown in Sect. 3.

Theorem 1 *Let $u(x,t) \in X_s = \mathcal{C}^1\left(H_p^s(\Omega); [0,T]\right)$, $s \geq 1$, be the solution of the problem (6) with periodic boundary conditions and initial condition u_0, $v \in H_p^s(\Omega)$. Let $u^N(x,t)$ be the solution of the semi-discrete scheme (12). Assume that $C \in L^\infty(\Omega)$; then, for every $T > 0$, there exists a constant $M = M(T)$, independent of N, such that*

$$\left\| u - u^N \right\|_{X_1} \leq M(T)\,(\Delta x)^{s-1}\,\|u\|_{X_s}\,. \tag{14}$$

The proof consists in considering a reformulation of the problem in the frequency space thanks to the introduction of the projection operator P_N defined as follows:

$$P_N u(x) = \sum_{k_1=-N}^{N}\sum_{k_2=-N}^{N} \tilde{u}_k e^{\Im(k_1 x_1 + k_2 x_2)}.$$

3 Volume Penalization Technique

Spectral methods based on Fourier polynomials require the assumption of periodic solution. A way to overcome such limitation is using a volume penalization technique. It consists in extending the computational domain Ω to a fictitious domain V by adding a constrained domain Γ such that

$$V = \Omega \cup \Gamma.$$

We ask that the solution is periodic on the extended domain V, and we penalize the solution on Γ thanks to the introduction of a penalization factor.

More precisely, let $\varepsilon > 0$ be the penalization factor and u_ε be the periodic solution on V, satisfying the local Dirichlet boundary condition

$$u_\varepsilon(\cdot, t)\big|_{\partial\Omega} = u_{bc},$$

for a given value u_{bc}; then the points on the constrained domain Γ satisfy

$$u(\cdot, t)\big|_\Gamma = u_\Gamma(x, t) = 2u_{bc} - u(2x^* - x, t), \qquad x^* \in \partial\Omega, \quad x \in \Gamma,$$

where x^* is such that its distance from $2x^* - x$ is the shortest for every $x \in \Gamma$.

The Fourier spectral method with volume penalization technique is the following:

$$\begin{aligned} \rho \frac{d^2}{dt^2} u^N_{\varepsilon,n} =& \mathcal{F}_N^{-1}\left(\mathcal{F}_N(C)\mathcal{F}_N\left((u^N_{\varepsilon,n})^r\right)(\Delta x)^2\right) \\ &+ \sum_{\ell=1}^{r-1} \binom{r}{\ell} (-1)^\ell \mathcal{F}_N^{-1}\left(\mathcal{F}_N\left((u^N_{\varepsilon,n})^\ell *_\Omega \left(\mathcal{F}_N(C)\mathcal{F}_N\left(u^N_{\varepsilon,n}\right)^{r-\ell}\right)(\Delta x)^2\right)\right) \\ &- \gamma\left(u^N_{\varepsilon,n}\right)^r + b^N_n - \frac{\chi_n}{\varepsilon}\left(u^N_{\varepsilon,n} - (u_\Gamma)^N_{\varepsilon,n}\right), \end{aligned} \tag{15}$$

for

$$\chi_n = \begin{cases} 1, & \text{if } x_n \in \Gamma, \\ 0, & \text{if } x_n \in \Omega, \end{cases}$$

and $u^N_{\varepsilon,n} = u^N_\varepsilon(x_n, t)$, $n \in \{-N, \ldots, N\}$.

4 The Fully Discrete Problem

In this section, we present the time discretization of the semi-discrete system (12) by means of the Störmer–Verlet method and Newmark-β methods.

Let $\Delta t > 0$ be the time step. We fix a partition of the time interval $[0, T]$ by means of $t_s = s\Delta t$, for $s = 0, \ldots, S_T$, where $S_T = \left\lfloor \frac{T}{\Delta t} \right\rfloor$. We define (u_N^s, v_N^s) as the numerical approximation of $(u^N(\cdot, t_s), v^N(\cdot, t_s))$. For the sake of simplicity, we assume the absence of external forces, namely $b \equiv 0$.

4.1 Störmer–Verlet Scheme

Störmer–Verlet method is a symplectic, second-order explicit scheme:

$$\begin{cases} u_N^{s+1} = u_N^s + \Delta t \left(v_N^s + \frac{\Delta t}{2\rho} \mathcal{L}_N(u_N^s) \right), \\ v_N^{s+1} = v_N^s + \frac{\Delta t}{2\rho} \left(\mathcal{L}_N(u_N^s) + \mathcal{L}_N(u_N^{s+1}) \right), \\ u_N^0 = u_{n,0}, \quad v_N^0 = v_{n,0}, \end{cases} \tag{16}$$

for $s = 0, \ldots, S_T$. A similar scheme is used for the method with volume penalization defined in (15). In [7], the authors make a von Neumann analysis to study the stability of such scheme.

4.2 Newmark-β Method

This is a symplectic second-order scheme implicit for $\beta \in (0, 1/2]$:

$$\begin{cases} u_N^{s+1} = u_N^s + \Delta t v_N^s + (\Delta t)^2 \left(\left(\frac{1}{2} - \beta \right) \mathcal{L}_N(u_N^s) + \beta \mathcal{L}_N(u_N^{s+1}) \right), \\ v_N^{s+1} = v_N^s + \frac{\Delta t}{2} \left(\mathcal{L}_N(u_N^s) + \mathcal{L}_N(u_N^{s+1}) \right), \\ u_N^0 = u_{n,0}, \quad v_N^0 = v_{n,0}. \end{cases} \tag{17}$$

We notice that the method coincides with the Störmer–Verlet scheme when $\beta = 0$. Additionally, it is unconditionally stable for $\beta \in [1/4, 1/2]$.

In [19], the authors prove the convergence of the sequence defined by the Newmark-β method in (17) to the exact solution of the problem (12) in the case in which the integral operator $\mathcal{L}$ and its discrete approximation $\mathcal{L}_N$ are linear.

Theorem 2 *Let $1/4 \leq \beta \leq 1/2$. If $u \in \mathcal{C}^3\left([0, T], H_p^2(V)\right)$ is the solution of the problem (6) with initial condition u_0, $v_0 \in H_p^2(V)$, and if $\{u_N^s\}_{s=0}^{S_T}$ is the sequence generated by the method (17). Then*

$$\left\| u(t_s) - u_N^s \right\|_{H_p^2(V)} \leq M (\Delta x)^2 \left(\|u_0\|_{H_p^2(V)} + \|u_t\|_{L^1(0, t_s, H_p^2(V))} \right) + M (\Delta t)^2, \tag{18}$$

where $M > 0$ is a constant depending on the regularity of u.

In the linear case, in [19], a stability result for the Newmark-β method is also achieved.

Theorem 3 *Let* $1/4 \leq \beta \leq 1/2$. *If* $\{u_N^s\}_{s=0}^{S_T} = \{u_N(t_s)\}_{s=0}^{S_T}$ *is the sequence generated by the method* (17)*, then there exist two positive constants* M_0 *and* M_1 *such that*

$$\|u_N(t_s)\|_{H_p^2(V)} \leq M_1 + M_0 t_s, \qquad t_s \in [0, T]. \tag{19}$$

5 Numerical Simulations

In this section, we present several simulations to confirm our results both in the two-dimensional and in the one-dimensional case.

5.1 Simulations on a 2D Lamina

Let a thin lamina in the spatial domain $[0, 1] \times [0, 1]$ be discretized by a bi-dimensional grid having the same space step $\Delta x = 0.01$ on both directions. We assume that the lamina is subjected to the uniform initial displacement $u_0(x_1, x_2) = -0.5x_1 - 0.5x_2$, and we fix $\delta = 0.2$ as horizon.

As micromodulus function, we take $C(x_1, x_2) = \exp(-x_1^2 - x_2^2)$, and we choose $w(\eta) = \eta^r$, with $r = 3$. Moreover, we assume that $b(x, t) \equiv 0$, and the mass density of the body is $\rho = 1$. Additionally, we take $\beta = 1/4$ for the implementation of the Newmark-β method.

We compute the numerical solution by using the Fourier spectral method with volume penalization defined in (15).

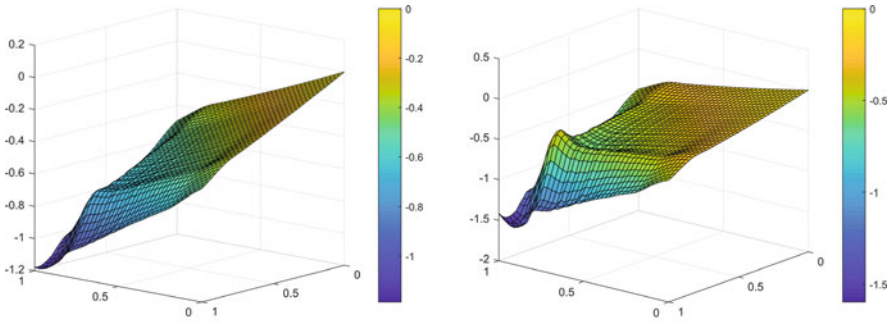
Figure 1 shows the dynamic of the solution in the spatial domain as time evolves. To evaluate the convergence of the fully discrete scheme, we introduce the relative error in the discrete $L^2(\Omega)$ norm at time t:

$$E_{L^2}^t = \frac{\sum_{n=0}^{N} |u_N(x_n, t) - u^*(x_n, t)|^2}{\sum_{n=0}^{N} |u_N(x_n, t)|^2},$$

where u^* is the reference solution for the problem.

Table 1 depicts the error $E_{L^2}^t$ between the exact solution and the numerical one with respect to the space step Δx for a fixed value of time.

Using the same setting, we make a comparison with the Störmer–Verlet method and the Newmark-β method in terms of L^2 error with respect to the time step and in terms of CPU execution time. Table 2 shows that the Newmark-β method provides the same accuracy of the Störmer–Verlet scheme using a larger time step, while



$t = 7.5.$ $t = 10.$

Fig. 1 With reference to Sect. 5.1, the solution at time $t = 5$ and $t = 7.5$. The parameters for the simulation are $\delta = 0.2$, $\Delta x = 10^{-2}$, and $\Delta t = 10^{-4}$

Table 1 With reference to Sect. 5.1, the relative L^2 error corresponding to the spectral method with volume penalization at time $t = 5$ as a function of the space step, for $\Delta t = 10^{-4}$

Δx	$E^t_{L^2}$	Convergence rate
0.2	7.8142×10^{-1}	–
0.1	1.2049×10^{-1}	2.6972
0.05	2.5370×10^{-2}	2.4725
0.025	6.1826×10^{-3}	2.3193
0.01	8.2514×10^{-4}	2.2570

Table 2 With reference to Sect. 5.1, the relative L^2-error at time $t_s = 5$ as function of the time step, for $\Delta x = 0.01$

	$E^{t_s}_{L^2}$	
Δt	Newmark-β	Störmer-Verlet
0.1	4.6812×10^{-6}	5.1511×10^{-4}
0.05	2.9276×10^{-7}	3.4650×10^{-5}
0.01	4.6629×10^{-9}	3.4634×10^{-5}
0.005	2.8706×10^{-10}	3.0868×10^{-6}
0.001	2.6360×10^{-12}	2.3476×10^{-7}

Table 3 With reference to Sect. 5.1, the execution time of the Newmark-β method and the Störmer-Verlet method as function of the time step, for $\Delta x = 0.01$

	CPU time [s]	
Δt	Newmark-β	Störmer-Verlet
0.5	9.8208×10^{0}	2.2801×10^{0}
0.1	3.0492×10^{2}	5.7025×10^{1}
0.05	7.8184×10^{2}	2.1405×10^{2}
0.01	2.1945×10^{4}	6.5321×10^{3}
0.005	9.8574×10^{4}	3.6108×10^{4}

Table 3 shows the CPU execution time for both methods. We can state that the Störmer–Verlet method is more efficient, even if Newmark-β method seems very competitive as it gains at least two orders of accuracy in terms of error.

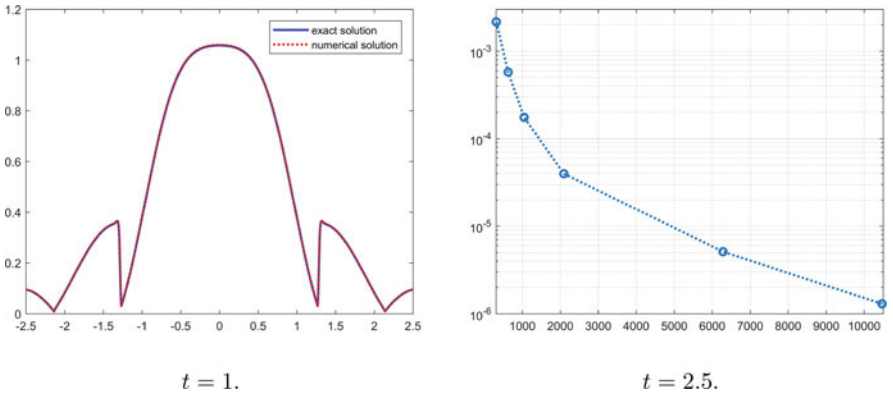


Fig. 2 With reference to Sect. 5.2. Left panel: Comparison between the exact solution with the numerical solution obtained by the spectral method with volume penalization at time $t = 1$. The parameters for the computed solution are $\delta = 0.5$, $\varepsilon = 0.5$, $N = 6284$, $\Delta x = 10^{-3}$ and $\Delta t = 10^{-4}$. Right panel: the relative L^2 error $E^t_{L^2}$ by varying N, using the semilogy scale

5.2 Simulations on a 1D Bar

Let $[-2.5, 2.5]$ be the spatial domain under consideration, and we discretize it by a uniform meshgrid with $N = 6284$ collocation points. We choose $u_0(x) = \exp(-x^2)$ as initial condition, and we fix $\delta = 0.5$ as horizon.

As micromodulus function, we take $C(x) = \exp(-x^2)$, and we choose $w(\eta) = \eta^r$, with $r = 3$. Moreover, we assume that $b(x, t) \equiv 0$, and the mass density of the body is $\rho = 1$. Again, we take $\beta = 1/4$ for the implementation of the Newmark-β method.

In the left panel of Fig. 2, we plot a comparison between the exact solution and the numerical one computed by means of the spectral method with volume penalization at times $t = 1$ and $t = 2.5$. We observe a good agreement with the reference solution for this non-periodic problem. In the right, we show the decreasing behavior of the relative L^2 error as a function of the total number of collocation points N for $t = 1$.

In Fig. 3, we also perform a convergence analysis of the semi-discretization problem with respect to the penalization factor ε showing that the penalization term in (15) decreases as a function of $1/\varepsilon$. For the simulation, we use the same parameters and setting as before.

6 Conclusions

In this chapter, we propose a discretization of the peridynamic model in both one- and two-dimensional domains based on the spectral Fourier polynomials with

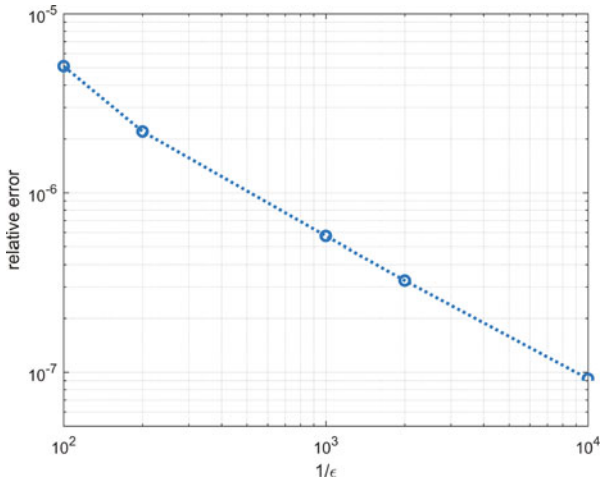


Fig. 3 With reference to Sect. 5.2, a convergence analysis with respect to the penalization factor ε

a volume penalization technique for the spatial domain and the implementation of the Störmer–Verlet method and Newmark-β methods for the time marching. The volume penalization is required in order to avoid the assumption of periodic solution. We show a convergence result for the semi-discrete problem and for the fully discrete linear problem. A stability result for the Newmark-β method is also provided.

Simulations show that spectral techniques are very accurate in the peridynamic framework and that a good accuracy can be reached by using the Newmark-β method.

Acknowledgments This paper has been supported by GNCS of Istituto Nazionale di Alta Matematica, by PRIN 2017 "Discontinuous dynamical systems: theory, numerics and applications" and by Regione Puglia, "Programma POR Puglia 2014/2020-Asse X-Azione 10.4 Research for Innovation-REFIN - (D1AB726C)."

References

1. Alebrahim, R.: Peridynamic modeling of Lamb wave propagation in bimaterial plates. Composite Structures **214**, 12–22 (2019)
2. Alebrahim, R., Packo, P., Zaccariotto, M., Galvanetto, U.: Wave propagation improvement in two-dimensional bond-based peridynamics model. In: Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science (2021)
3. Berardi, M., Difonzo, F., Lopez, L.: A mixed MoL–TMoL for the numerical solution of the 2D Richards' equation in layered soils. Comput. Math. Appl. **79**(7), 1990–2001 (2020)

4. Bueno-Orovio, A., Pèrez-García, V.M., Fenton, F.H.: Spectral methods for partial differential equations in irregular domains: The spectral smoothed boundary method. SIAM J. Sci. Comput. **28**(3), 886–900 (2006)
5. Coclite, G.M., Dipierro, S., Maddalena, F., Valdinoci, E.: Wellposedness of a nonlinear peridynamic model. Nonlinearity **32**(1), 1–21 (2018)
6. Coclite, G.M., Paparella, F., Pellegrino, S.F.: On a salt fingers model. Nonlinear Analysis **176**, 100–116 (2018)
7. Coclite, G.M., Fanizzi, A., Lopez, L., Maddalena, F., Pellegrino, S.F.: Numerical methods for the nonlocal wave equation of the peridynamics. Appl. Numer. Math. **155**, 119–139 (2020)
8. Dal Santo, E., Donadello, C., Pellegrino, S.F., Rosini, M.D.: Representation of capacity drop at a road merge via point constraints in a first order traffic model. ESAIM: M2AN **53**(1), 1–34 (2019)
9. D'Elia, M., Du, Q., Gunzburger, M.: Recent Progress in Mathematical and Computational Aspects of Peridynamics. Springer International Publishing (2017)
10. Emmrich, E., Weckner, O.: Numerical simulation of the dynamics of a nonlocal, inhomogeneous, infinite bar. J. Comput. Appl. Mech. **6**(2), 311–319 (2005)
11. Erbay, H.A., Erkip, A., Muslu, G.M.: The Cauchy problem for a one-dimensional nonlinear elastic peridynamic model. J. Differ. Equ. **252**(8), 4392–4409 (2012)
12. Eringen, A.C., Edelen, D.G.B.: On nonlocal elasticity. Int. J. Eng. Sci. **10**(3), 233–248 (1972)
13. Garrappa, R.: Numerical evaluation of two and three parameter Mittag-Leffler Functions. SIAM J. Numer. Anal. **53**(3), 1350–1369 (2015)
14. Guimarães, O., Piqueira, J.R.: Novel approach to spectral methods for irregular domains. Comput. Math. Appl. **80**, 1–12 (2020)
15. Hafezi, M.H., Alebrahim, R., Kundu, T.: Crack propagation modeling using peridynamic theory. In: Health Monitoring of Structural and Biological Systems, vol. 9805, pp. 209–216. SPIE (2016)
16. Jafarzadeh, S., Larios, A., Bobaru, F.: Efficient solutions for nonlocal diffusion problems via boundary-adapted spectral methods. J. Peridynam. Nonlocal Model. **2**, 85–110 (2020)
17. Jafarzadeh, S., Mousavi, F., Larios, A., Bobaru, F.: A general and fast convolution-based method for peridynamics: applications to elasticity and brittle fracture. Preprint (2021). arXiv:2105.06055v1
18. Jafarzadeh, S., Wang, L., Larios, A., Bobaru, F.: A fast convolution-based method for peridynamic transient diffusion in arbitrary domains. Comput. Methods Appl. Mech. Eng. **375**, 113633 (2021)
19. Lopez, L., Pellegrino, S.F.: A space-time discretization of a nonlinear peridynamic model on a 2D lamina. Comput. Math. Appl. **116**, 161–175 (2022). https://doi.org/10.1016/j.camwa.2021.07.004
20. Lopez, L., Pellegrino, S.F.: A spectral method with volume penalization for a nonlinear peridynamic model. Int. J. Numer. Methods Eng. **122**(3), 707–725 (2021)
21. Lopez, L., Pellegrino, S.F.: Computation of eigenvalues for nonlocal models by spectral methods. J. Peridynam. Nonlocal Model. (2021, in press). https://doi.org/10.1007/s42102-021-00069-8
22. Lopez, L., Pellegrino, S.F.: A non-periodic Chebyshev spectral method avoiding penalization techniques for a class of nonlinear peridynamic models. Int. J. Numer. Methods Eng. **123**(20), 4859–4876 (2022). https://doi.org/10.1002/nme.7058
23. Pellegrino, S.F.: On the implementation of a finite volumes scheme with monotone transmission conditions for scalar conservation laws on a star-shaped network. Appl. Numer. Math. **155**, 181–191 (2020)
24. Pellegrino, S.F.: Simulations on the peridynamic equation in continuum mechanics. In: 13th Chaotic Modeling and Simulation International Conference, vol. 46, pp. 613–626 (2021). https://doi.org/10.1007/978-3-030-70795-8_46
25. Shojaei, A., Mudric, T., Zaccariotto, M., Galvanetto, U.: A coupled meshless finite point/Peridynamic method for 2D dynamic fracture analysis. Int. J. Mech. Sci. **119**, 419–431 (2016)

26. Silling, S.A.: Reformulation of elasticity theory for discontinuities and long-range forces. J. Mech. Phys. Solids **48**(17–18), 175–209 (2000)
27. Silling, S., Askari, E.: A meshfree based on the peridynamic model of solid mechanics. Comput. Struct. **83**(17–18), 1526–1535 (2005)
28. Silling, S.A., Bobaru, F.: Peridynamic modeling of membranes and fibers. Int. J. Non-Linear Mech. **40**(2), 395–409 (2005). Special Issue in Honour of C.O. Horgan

A MATLAB Code for Fractional Differential Equations Based on Two-Step Spline Collocation Methods

Angelamaria Cardone, Dajana Conte, and Beatrice Paternoster

Abstract We illustrate a MATLAB implementation of two-step spline collocation methods for the numerical solution of fractional differential equations, introduced by Cardone, Conte and Paternoster in (Discrete Dyn. Syst. Ser. B 23(7) 2709–2725 (2018)). The computational tasks include the evaluation of fractional integrals, a suitable starting procedure, and the efficient computation of the coefficients of certain polynomials. The whole algorithm is described in detail. Some numerical experiments show the performances of the proposed algorithm.

1 Introduction

A considerable and increasing number of problems are modeled by fractional differential equation (FDE), e.g., the behavior of viscoelastic materials [40], anomalous diffusion [22, 29], signal processing [51], and financial processes [49]. However, the analytical solution of FDEs is usually not available and numerical methods must be employed. The literature on numerical methods for FDEs and FPDEs is rich and covers a wide range of approaches, for example, Adomain decomposition methods [20], methods based on block-pulse functions [39], methods based on the Grünwald–Letnikov formula [41, 50], wavelet-based methods [35], methods that use the short-memory principle [3, 24], collocation methods [12, 38, 43–45, 55], and spectral methods [7, 54].

Applied sciences need accurate, reliable, and easy-to-use mathematical software to compute the solution of fractional differential problems. At present, neither computing environments, such as MATLAB, nor numerical libraries, such as NAG, offer professional routines to solve FDEs. A limited number of codes have been proposed by some scientists, for reference, see [13, 31, 46, 52, 53]. A recent overview on the available software is given in [31].

A. Cardone (✉) · D. Conte · B. Paternoster
Department of Mathematics, University of Salerno, Fisciano, Salerno, Italy
e-mail: ancardone@unisa.it; djconte@unisa.it; beapat@unisa.it

A. Cardone et al. (eds.), *Fractional Differential Equations*,
Springer INdAM Series 50, https://doi.org/10.1007/978-981-19-7716-9_8

In the design of a high-quality mathematical software to solve FDEs, a challenging issue regards the computational cost. On one hand, many numerical methods for FDEs have a low order of convergence, usually not exceeding 2; thus high accuracy may be reached at the price of considering fine meshes, with an increase of the computation time. On the other hand, due to the hereditary nature of the fractional differential operator, a considerable amount of computation is required to discretize the history term of the solution, usually $O(N^2)$ flops, for a mesh of length N. The most famous procedure to considerably reduce this cost relies on the FFT technique, which can be applied when the numerical method can be expressed as a discrete convolution product, and yields to $O(N(\log_2 N)^2)$ flops. This procedure was introduced in [33, 34] and later on applied in several papers, see, e.g., [9, 10, 18, 27, 30, 31, 36, 48].

In our research, we focus on the efficient implementation of spline collocation methods introduced and analyzed in [5, 11, 12, 14, 16, 43–45]. These methods offer many advantages: they have high order of convergence; they are continuous methods, i.e., they provide the approximate solution at each point of the integration interval; they have good stability properties. These good properties are well known not only in the context of FDEs, but also for the solution of ordinary differential equations and Volterra integral equations [4, 6, 15, 19, 21, 23, 28, 42]. In [13], we developed a MATLAB algorithm based on one-step spline collocation methods introduced in [44].

In this chapter, we treat the efficient implementation of two-step spline collocation methods introduced in [12] (see also [14, 16, 17]). These methods have almost the same computational cost of the one-step collocation methods but double the order of convergence. Being multivalue methods, a starting procedure is needed, which provides a continuous approximation of the solution in first integration interval and preserves the order of convergence of the overall method. We consider as starting method the one-step spline collocation method [44]. The application of the two-step collocation method requires the evaluation of several fractional integrals: this can be done without need for quadrature approximation, once the coefficients of some Lagrange polynomials are known. This task has been addressed, at a reduced computational cost, by MATLAB inbuilt routines. A compact matrix formulation of the method has been provided, to get maximum efficiency of MATLAB with array computation.

This chapter is organized as follows. We describe the two-step collocation methods in Sect. 2. The computation of fractional integrals, appearing in the method, is illustrated in Sect. 3, while a suitable starting procedure is proposed in Sect. 4. In Sect. 5, we resume main convergence results and discuss the proper setting of the method parameters. Section 6 contains a compact matrix formulation of the method. The MATLAB codes are given in Sect. 7; the input and output arguments of the algorithm are defined in Sect. 8; an example of usage of the program is provided in Sect. 9. Finally, we show some numerical results in Sect. 10 to verify convergence results and for comparison with one-step spline collocation methods and with a fractional BDF method.

2 The Two-Step Spline Collocation Method

We consider the following initial value problem:

$$\begin{cases} D^{\alpha} y(t) = f(t, y(t)), \ 0 \le t \le b, \\ y^{(i)}(0) = \gamma_i, \qquad\qquad i = 0, \ldots, n-1, \end{cases} \tag{1}$$

where $n - 1 < \alpha < n$, $n \in \mathbb{N}$, $\gamma_i \in \mathbb{R}$. The real-valued function f is continuous in $[0, b] \times \mathbb{R}$. The fractional derivative is defined in the Caputo sense [25, 37, 47]:

$$D^{\alpha} y(t) = \frac{1}{\Gamma(n-\alpha)} \int_0^t \frac{y^{(n)}(s)}{(t-s)^{\alpha+1-n}} ds.$$

Some sufficient conditions for the existence and uniqueness of solution of (1) may be found in [25].

The preliminary step to apply spline collocation methods transforms the IVP (1) into the functional equation

$$z = f(t, J^{\alpha} z + Q), \tag{2}$$

where $z = D^{\alpha} y$,

$$(J^{\alpha} z)(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} z(s)\, ds, \quad t > 0, \tag{3}$$

$$Q(t) = \sum_{i=0}^{\lceil \alpha \rceil - 1} \frac{\gamma_i}{i!} t^i, \tag{4}$$

and $\lceil \alpha \rceil$ is the smallest integer not less than α. Once the equation (2) is solved, the solution of (1) is given by

$$y = J^{\alpha} z + Q. \tag{5}$$

We introduce some quantities:

- The *graded mesh* $I_N = \{0 = t_0 < t_1 < \cdots < t_N = b\}$, with grading exponent $r \in \mathbb{R}$, $r \ge 1$:

$$t_j = b \left(\frac{j}{N} \right)^r, \qquad j = 0, \ldots, N. \tag{6}$$

If $r = 1$, the mesh is uniform. The best choice of the grading exponent will be discussed in Sect. 5. Moreover, we define $\sigma_j = [t_{j-1}, t_j]$, $h_j = t_j - t_{j-1}$.

- The set of *collocation abscissae*:

$$0 \leq \eta_1 < \ldots < \eta_m \leq 1, \text{ with } (\eta_1, \eta_m) \neq (0, 1). \tag{7}$$

- The *collocation points*:

$$t_{jk} = t_{j-1} + h_j \eta_k, \quad j = 1, \ldots, N, \ k = 1, \ldots, m. \tag{8}$$

The two-step spline collocation method computes an approximate solution of problem (2), as a function of the space

$$S_{2m-1}^{(-1)}(I_N) = \left\{ v : v|_{\sigma_j} \in \pi_{2m-1}, \ j = 1, \ldots, N \right\}, \tag{9}$$

where π_{2m-1} is the space of algebraic polynomials of degree not exceeding $2m-1$. This function $v(t)$ is computed as follows:

$$v(t) = v_1(t) + \sum_{\lambda=2}^{N} \left(\sum_{k=1}^{m} z_{\lambda k} L_{\lambda,m+k}(t) + \sum_{k=1}^{m} z_{\lambda-1,k} L_{\lambda k}(t) \right), \quad t \in [0, b], \tag{10}$$

with $z_{\lambda k} = v(t_{\lambda k})$ and

$$L_{\lambda k}(t) = \begin{cases} k - \text{th Lagrange fund. pol. wrt to } \{t_{\lambda-1,i}, t_{\lambda,i}\}_{i=1}^{m}, \ t \in [t_{\lambda-1}, t_{\lambda}] \\ 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{otherwise.} \end{cases} \tag{11}$$

The coefficients $z_{\lambda k}$ are computed by imposing that $v(t)$ fulfills equation (2) at the collocation points, i.e.,

$$z_{jk} = f\left(t_{jk}, (J^{\alpha} v)(t_{jk}) + Q(t_{jk})\right), \quad k = 1, \ldots, m, \tag{12}$$

$j = 2, \ldots, N$. By (10) and (11), we obtain this equivalent formulation of system (12):

$$\begin{aligned} z_{jk} = f \Bigg(& t_{jk}, (J^{\alpha} v_1)(t_{jk}) + \sum_{\mu=1}^{m} z_{j\mu} (J^{\alpha} L_{j,m+\mu})(t_{jk}) + \\ & \sum_{\lambda=2}^{j-1} \sum_{\mu=1}^{m} z_{\lambda\mu} (J^{\alpha} L_{\lambda,m+\mu})(t_{jk}) + \sum_{\mu=1}^{m} z_{j-1,\mu} (J^{\alpha} L_{j\mu})(t_{jk}) + \\ & \sum_{\lambda=2}^{j-1} \sum_{\mu=1}^{m} z_{\lambda-1,\mu} (J^{\alpha} L_{\lambda\mu})(t_{jk}) + Q(t_{jk}) \Bigg), \quad k = 1, \ldots, m, \end{aligned} \tag{13}$$

$j = 2, \ldots, N$.

Moreover, we have

$$z_{1k} = v_1(t_{1k}), \quad k = 1, \ldots, m. \tag{14}$$

The function $v_1(t)$ vanishes outside the interval $[t_0, t_1]$ and must be defined by a suitable starting procedure, to preserve the accuracy of the method. This point will be discussed in Sect. 4.

Finally, the numerical solution $y_N(t)$ of (1) is computed by (compare (5))

$$y_N = J^\alpha v + Q,$$

which is equivalent to

$$\begin{aligned} y_N(t) = J^\alpha(v_1)(t) + \sum_{\mu=1}^{m} z_{j\mu}(J^\alpha L_{j,m+\mu})(t) + \sum_{\lambda=2}^{j-1}\sum_{\mu=1}^{m} z_{\lambda\mu}(J^\alpha L_{\lambda,m+\mu})(t) \\ + \sum_{\mu=1}^{m} z_{j-1,\mu}(J^\alpha L_{j\mu})(t) + \sum_{\lambda=2}^{j-1}\sum_{\mu=1}^{m} z_{\lambda-1,\mu}(J^\alpha L_{\lambda\mu})(t) + Q(t), \end{aligned} \tag{15}$$

for $t \in [t_{j-1}, t_j]$, with $j = 2, \ldots, N$, while

$$y_N(t) = J^\alpha(v_1)(t) + Q(t), \quad t \in [t_0, t_1]. \tag{16}$$

3 Computation of Fractional Integrals

The two-step collocation method requires the computation of fractional integrals in (13) and in (15). Since the integrand functions are polynomials, these integrals can be exactly evaluated, without need for quadrature approximation. In this section, we resume and complete the procedure illustrated in [12], adding necessary details to implement the method.

We will use the incomplete beta function, i.e.,

$$\beta(x, a, b) = \int_0^x s^{a-1}(1-s)^{1-b} ds,$$

and the following formulas [1, 25, 32]:

$$(J^\alpha t^\nu)(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} s^\nu \, ds = t^{\nu+\alpha} \frac{\Gamma(1+\nu)}{\Gamma(1+\nu+\alpha)}, \tag{17}$$

$$\int_0^1 (t-s)^{\alpha-1} s^\nu ds = t^{\nu+\alpha} \beta\left(\frac{1}{t}, 1+\nu, \alpha\right). \tag{18}$$

Let us consider the polynomial

$$\varphi_{j\mu}(\tau) := L_\mu(t_{j-1} + \tau h_j), \quad \tau \in [0,1],\ \mu = 1, \ldots, 2m.$$

In [12], it has been proven that $\varphi_{j\mu}(\tau)$, $\mu = 1, \ldots, 2m$, are the fundamental Lagrange polynomial corresponding to the nodes

$$\left\{(\eta_1 - 1)\frac{h_{j-1}}{h_j}, \ldots, (\eta_m - 1)\frac{h_{j-1}}{h_j}, \eta_1, \ldots, \eta_m\right\}. \tag{19}$$

It will be useful to express this polynomial in the canonical form

$$\varphi_{j\mu}(\tau) = \sum_{\nu=0}^{2m-1} a_\nu^{(j\mu)} \tau^\nu, \quad \mu = 1, \ldots, 2m. \tag{20}$$

Remark 1 We adopt a simple procedure to derive the coefficients of the Lagrange polynomials (20). To simplify the description, let us name the nodes (19) as $\{x_1, \ldots, x_{2m}\}$, and then the μ-th Lagrange fundamental polynomial corresponding to these nodes is

$$\prod_{\substack{i=1 \\ i\neq\mu}}^{m} \frac{\tau - x_i}{x_\mu - x_i} = \sum_{\nu=0}^{2m-1} a_\nu^{(\mu)} \tau^\nu, \qquad \mu = 1, \ldots, 2m.$$

It is well known that the coefficients of the product of polynomials $p(x) = q(x)r(x)$ can be obtained by convolution of the vectors of coefficients $q(x)$ and $r(x)$. Therefore, a straightforward technique to compute the coefficients $a_\nu^{(\mu)}$ is based on the iterative convolution products of vectors $\left[\frac{1}{x_\mu - x_i}, \frac{-x_i}{x_\mu - x_i}\right]$, $i = 1, \ldots, 2m$, $i \neq \mu$.

The fractional integrals in (13) and in (15) can be computed as follows (cfr. (17), (18) and (20)). Let $t \in [t_{j-1}, t_j]$, with $t = t_{j-1} + \sigma h_j$; then,

$$(J^\alpha L_{j\mu})(t) = h_j^\alpha \sum_{\nu=0}^{2m-1} a_\nu^{(j\mu)} \sigma^{\nu+\alpha} \frac{\Gamma(1+\nu)}{\Gamma(1+\nu+\alpha)}, \quad \mu = 1, \ldots, 2m, \tag{21}$$

and

$$(J^\alpha L_{\lambda\mu})(t) = \frac{h_\lambda^\alpha}{\Gamma(\alpha)} \sum_{\nu=0}^{2m-1} a_\nu^{(\lambda\mu)} \left(\frac{t_{j-1} + \sigma h_j - t_{\lambda-1}}{h_\lambda}\right)^{\nu+\alpha} \beta\left(\frac{h_\lambda}{t_{j-1} + \sigma h_j - t_{\lambda-1}}, 1+\nu, \alpha\right), \tag{22}$$

$\lambda = 2, \ldots, j-1,\ \mu = 1, \ldots, 2m.$

Therefore, if $\sigma = \eta_k$, we obtain the values of fractional integrals appearing in (13). While, if $\sigma \in [0, 1]$, we get the values of fractional integrals of formula (15).

4 Starting Procedure

The starting procedure of the two-step spline collocation method (10)–(12) must provide a continuous approximation v_1 of the solution of (2) in the interval $[t_0, t_1]$. With this aim, we apply in $[t_0, t_1]$ the one-step collocation method proposed in [44], with $2m$ collocation abscissae, to preserve the accuracy of the overall method. In particular, we assume that:

- $v_1 \in S^{(-1)}_{2m-1}(I_N)$.
- $v_1(t) = 0$, for $t > t_1$.
- $0 \le \tilde{\eta}_1 < \cdots < \tilde{\eta}_{2m} \le 1$ and $\tilde{t}_k = \tilde{\eta}_k h_1$.
- $\tilde{z}_k = v_1(\tilde{t}_k)$, $k = 1, \ldots, 2m$.

Then

$$v_1(t) = \begin{cases} \displaystyle\sum_{\lambda=1}^{2m} \tilde{z}_\lambda L_\lambda(t), \ t \in [t_0, t_1], \\ 0, \qquad\qquad \text{elsewhere}, \end{cases} \tag{23}$$

with

$$L_\lambda(t = \sigma h_1) = \prod_{\substack{i=1 \\ i \neq \lambda}}^{2m} \frac{\sigma - \tilde{\eta}_i}{\tilde{\eta}_\lambda - \tilde{\eta}_i} = \sum_{\nu=0}^{2m-1} \tilde{a}_\nu^{(\lambda)} \sigma^\nu, \quad \sigma \in [0, 1], \tag{24}$$

where coefficients $\tilde{a}_\nu^{(\lambda)}$ can be computed along the lines of Remark 1 of Sect. 3.

The coefficients $\tilde{z}_1, \ldots, \tilde{z}_{2m}$ satisfy the collocation conditions:

$$\tilde{z}_k = f(\tilde{t}_k, (J^\alpha v_1)(\tilde{t}_k) + Q(\tilde{t}_k)), \quad k = 1, \ldots, 2m. \tag{25}$$

Now we can derive the formula for $z_{1k} = v_1(t_{1k})$, $k = 1, \ldots, m$ (cfr. (14)). Formulas (23) and (24) yield:

$$z_{1k} = \sum_{\lambda=1}^{2m} \tilde{z}_\lambda \sum_{\nu=0}^{2m-1} \tilde{a}_\nu^{(\lambda)} \eta_k^\nu, \quad k = 1, \ldots, m. \tag{26}$$

Obviously, the approximate solution y_N in the interval $[t_0, t_1]$, which is formally given by (16), can be computed by the considered one-step collocation method.

Further details on the computation of v_1 and y_N in the interval $[t_0, t_1]$ are given in [11, 13, 44].

Now we evaluate the fractional integrals of v_1 appearing in (13) and (15). From (23), we have

$$(J^\alpha v_1)(t) = \frac{1}{\Gamma(\alpha)} \sum_{\lambda=1}^{2m} \tilde{z}_\lambda \int_0^{t_1} (t-\tau)^{\alpha-1} L_\lambda(\tau) d\tau, \quad t \in [t_1, b]. \tag{27}$$

Then, (24) and (18) lead to

$$\begin{aligned} \int_0^{t_1} (t-\tau)^{\alpha-1} L_\lambda(\tau) &= h_1 \int_0^1 (t-h_1\sigma)^{\alpha-1} L_\lambda(h_1\sigma) d\sigma \\ &= h_1 \sum_{\nu=1}^{2m} \tilde{a}_\nu^{(k)} \int_0^1 (t-h_1\sigma)^{\alpha-1} \sigma^\nu d\sigma \\ &= h_1^\alpha \sum_{\nu=1}^{2m} \tilde{a}_\nu^{(\lambda)} \left(\frac{t}{h_1}\right)^{\nu+\alpha} \beta\left(\frac{h_1}{t}, 1+\nu, \alpha\right). \end{aligned}$$

Finally, (27) can be recast as

$$(J^\alpha v_1)(t) = \frac{h_1^\alpha}{\Gamma(\alpha)} \sum_{\lambda=1}^{2m} \tilde{z}_\lambda \sum_{\nu=1}^{2m} \tilde{a}_\nu^{(\lambda)} \left(\frac{t}{h_1}\right)^{\nu+\alpha} \beta\left(\frac{h_1}{t}, 1+\nu, \alpha\right), \quad t \in [t_1, b]. \tag{28}$$

5 Convergence and Optimal Parameters Setting

Here we resume the convergence analysis, and according to the theoretical error estimate, we will we describe the proper setting of the method parameters, i.e.:

- The number m and the location of the collocation abscissae $\eta_1, \ldots, \eta_m$ (7)
- The value of the grading exponent r of the graded mesh (6)

With this aim, we introduce the functional space $C^{q,\nu}(0, b]$, with $q \in \mathbb{N}$ and $\nu \in (-\infty, 1)$ [44]. A function $y : [0, b] \to \mathbb{R}$ belongs to $C^{q,\nu}(0, b]$, if $y \in C^q(0, b]$ and

$$|y^{(i)}(t)| \le c \begin{cases} 1, & \text{if } i < 1-\nu, \\ 1 + |\log t|, & \text{if } i = 1-\nu, \\ t^{1-\nu-i}, & \text{if } i > 1-\nu, \end{cases} \quad t \in (0, b], \quad i = 1, \ldots, q. \tag{29}$$

The solution of problem (1) is in the space $C^q(0, b]$ if the conditions of the following theorem are fulfilled.

Theorem 1 ([44]) *Assume that: the real-valued function $f \in C^q([0, b] \times \mathbb{R})$; there exists $\nu \in [1 - \alpha, 1)$ such that*

$$\left| \frac{\partial^{i+j}}{\partial t^i \partial y^j} f(t, y) \right| \leq \phi(|y|) \begin{cases} 1, & \text{if } i < 1 - \nu \\ 1 + |\log t|, & \text{if } i = 1 - \nu \\ t^{1-\nu-i}, & \text{if } i > 1 - \nu \end{cases}, \quad (t, y) \in (0, b] \times \mathbb{R}, \tag{30}$$

$\forall i, j \in \mathbb{N}$ *with* $i + j \leq q$. *For* $\alpha \in (0, 1)$, *assume also that*

$$\left| \frac{\partial^{i+j}}{\partial t^i \partial y^j} [f(t, y_1) - f(t, y_2)] \right| \leq \phi(\max\{|y_1|, |y_2|\}) |y_1 - y_2| \begin{cases} 1 & \text{if } i = 0 \\ t^{1-\nu-i} & \text{if } i > 0 \end{cases}, \tag{31}$$

$(t, y_\ell) \in (0, b] \times \mathbb{R}$, $\ell = 1, 2$, *where the real-valued function ϕ is monotonically increasing in $[0, \infty)$. Assume that there exists a solution $y \in C[0, b]$ of the problem (1), with $D^\alpha y \in C[0, b]$. Then $y \in C^{q,\nu}(0, b]$ and $D^\alpha y \in C^{q,\nu}(0, b]$.*

In the following theorem, the error will be bounded by E_N, with

$$E_N(p, \nu, r) = \begin{cases} N^{-r(1-\nu)} & \text{if } 1 \leq r \leq \frac{p}{1-\nu} \\ N^{-p}(1 + \log N) & \text{if } r = \frac{p}{1-\nu} = 1 \\ N^{-p} & \text{if } r = \frac{p}{1-\nu} > 1 \text{ or } r > \frac{p}{1-\nu}. \end{cases} \tag{32}$$

Theorem 2 ([12]) *Let the problem (1) have a solution $y \in C[0, T]$ such that $D^\alpha y \in C[0, T]$, and let $f : [0, T] \times \mathbb{R} \to \mathbb{R}$ be a continuous function such that its derivatives $\frac{\partial}{\partial t} f(t, y)$ and $\frac{\partial^2}{\partial t^2} f(t, y)$ are continuous in $(0, T] \times \mathbb{R}$ and*

$$\left| \frac{\partial^j}{\partial y^j} f(t, y) \right| \leq \psi(|y|), \quad (t, x) \in (0, T] \times \mathbb{R}, \ j = 0, 1, 2.$$

$\psi : [0, \infty) \to \mathbb{R}$ is a monotonically increasing function. Moreover, assume that the collocation points (8) with grid points (6) and arbitrary parameters $\eta_1, \ldots, \eta_m$ satisfying (7) are used. Then there exist $N_0 \in \mathbb{N}$ and $\delta_0 > 0$ such that, for all $N \geq N_0$, the two-step collocation method possesses a unique solution $v \in S^{(-1)}_{2m-1}(I_N)$ in the ball $\|u - z\|_\infty \leq \delta_0$, where $z = D^\alpha y \in C[0, T]$. If, in addition, the assumptions of Theorem 1 with $q := 2m$ and $\nu \in [1 - \alpha, 1)$ are fulfilled, then for all $N \geq N_0$ the following error estimate holds:

$$\|y_N - y\|_\infty \leq C E_N(2m, \nu, r), \tag{33}$$

with y_N given by the formula (15). Here C is a constant not depending on N, $r \in [1, \infty)$ is the grading exponent in (6), and E_N is defined by (32).

We observe that the previous theorem holds if the starting procedure is the one we proposed in Sect. 2 or another one that satisfies conditions mentioned in [12, Remark 2].

From the proof of Theorem 2, given in [12], we can easily derive the following corollary.

Corollary 1 *The error constant of (33) is given by*

$$C = (1 + 2\Lambda_m)\frac{b}{\alpha\Gamma(\alpha)} \cdot c_f \cdot c_y, \tag{34}$$

where Λ_m is the Lebesgue constant corresponding to the nodes $\{(\eta_1 - 1)\frac{h_1}{h_2}, \ldots, (\eta_m - 1)\frac{h_1}{h_2}, \eta_1, \ldots, \eta_m\}$; c_f depends on f; c_y is the constant appearing in inequality (29), corresponding to $D^\alpha y \in C^{q,\nu}(0, b]$.

Thus, constant C is an increasing function with respect to m. In addition, we observe that $\frac{1}{\alpha\Gamma(\alpha)} \in (0, 1.2)$ and rapidly decreases with α.

Theorem 2 yields that, if $y(t) \in C^{2m,\nu}(0, b]$ and

$$r \geq \frac{2m}{1 - \nu}, \tag{35}$$

then the fastest error decrease is reached, i.e.,

$$\|y - y_N\|_\infty \leq CN^{-2m},$$

where C is given by (34). This result holds independently from the location of collocation abscissae $\eta_1, \ldots, \eta_m$.

Finally, we conclude that the method is more accurate as m increases, as usual for collocation methods; the collocation abscissae can be arbitrarily chosen; the optimal choice of r is given by (35), thus depending on the problem we are solving. In Sect. 8, we will show, on one test example, how to verify the conditions of Theorem 1, necessary to derive the value of ν in formula (35).

6 Matrix Formulation of the Method

In this section, we collect the results obtained so far, to get a compact formulation of the method.

6.1 Matrix Formulation of Nonlinear System (13)

We define matrices $\mathbf{A}^{[j]} \in \mathbb{R}^{m\times 2m}$ and $\mathbf{E}^{[\lambda,j]} \in \mathbb{R}^{m\times 2m}$, for $j = 2,\ldots,N$ and $\lambda = 2,\ldots,j-1$. Their elements are proportional to the fractional integrals computed in (21) and in (22), with $\sigma = \eta_k$.

$$A_{k\mu}^{[j]} = \sum_{\nu=0}^{2m-1} a_\nu^{(j,\mu)} \eta_k^{\nu+\alpha} \frac{\Gamma(1+\nu)}{\Gamma(1+\nu+\alpha)},$$

$$E_{k\mu}^{[\lambda,j]} = \frac{1}{\Gamma(\alpha)} \sum_{\nu=0}^{2m-1} a_\nu^{(\lambda,\mu)} \left(\frac{t_{j-1}+\eta_k h_j - t_{\lambda-1}}{h_\lambda}\right)^{\nu+\alpha} \beta\left(\frac{h_\lambda}{t_{j-1}+\eta_k h_j - t_{\lambda-1}}; 1+\nu, \alpha\right),$$

$k = 1,\ldots,m$, $\mu = 1,\ldots,2m$. The coefficients $a_\nu^{(\mu,\lambda)}$ have been defined in (20) and can be computed as illustrated in Sect. 3, Remark 1.

Second, let us consider the following m-square matrices:

- $\bar{\mathbf{A}}^{[j]}$ and $\bar{\mathbf{E}}^{[\lambda,j]}$, which consist of the first m columns of $\mathbf{A}^{[j]}$ and $\mathbf{E}^{[\lambda,j]}$, respectively
- $\tilde{\mathbf{A}}^{[j]}$ and $\tilde{\mathbf{E}}^{[\lambda,j]}$, which consist of the last m columns of $\mathbf{A}^{[j]}$ and $\mathbf{E}^{[\lambda,j]}$, respectively

With these settings, the nonlinear system (13) can be recast as

$$\mathbf{z}_j = f\left(\mathbf{t}_j, h_j^\alpha \tilde{\mathbf{A}}^{[j]}\mathbf{z}_j + \sum_{\lambda=1}^{j-1} h_\lambda^\alpha \mathbf{B}^{[\lambda,j]}\mathbf{z}_\lambda + h_1^\alpha \mathbf{r}_j + \mathbf{q}_j\right), \tag{36}$$

for $j = 2,\ldots,N$, where:

- $\mathbf{z}_j = [z_{j1},\ldots,z_{jm}]^T$.
- $\mathbf{t}_j = [t_{j1},\ldots,t_{jm}]^T$.
- $\mathbf{q}_j = [Q(t_{j1}),\ldots,Q(t_{jm})]^T$.
- $f(\mathbf{t}_j,\mathbf{u}) = [f(t_{j1},u_1),\ldots,f(t_{jm},u_m)]^T$ for any m-dimensional vector $\mathbf{u}$.
- Set $d_j = h_j/h_{j-1}$, then

$$\mathbf{B}^{[1,2]} = d_2^\alpha \bar{\mathbf{A}}^{[2]},$$

while, for $j = 3,\ldots,N$:

$$\mathbf{B}^{[\lambda,j]} := \begin{cases} d_2^\alpha \bar{\mathbf{E}}^{[2,j]}, & \lambda = 1, \\ \tilde{\mathbf{E}}^{[\lambda,j]} + d_{\lambda+1}^\alpha \bar{\mathbf{E}}^{[\lambda+1,j]}, & \lambda = 2,\ldots,j-2, \\ d_j^\alpha \bar{\mathbf{A}}^{[j]} + \tilde{\mathbf{E}}^{[j-1,j]}, & \lambda = j-1. \end{cases}$$

- $\left(\mathbf{r}_j\right)_k = \dfrac{1}{\Gamma(\alpha)} \displaystyle\sum_{\lambda=1}^{2m} \tilde{z}_\lambda \sum_{\nu=0}^{2m-1} \tilde{a}_\nu^{(\lambda)} \left(\frac{t_{jk}}{h_1}\right)^{\nu+\alpha} \beta\left(\frac{h_1}{t_{jk}}, 1+\nu, \alpha\right), k = 1, \ldots, m.$

Last, from (26), it results

$$\mathbf{z}_1 = \mathbf{H}\,\tilde{\mathbf{A}}\mathbf{lagr}^T \tilde{\mathbf{z}},$$

where:

- $H_{k,\nu+1} = \eta_k^\nu, k = 1 \ldots, m, \nu = 0 \ldots, 2m-1.$
- $\tilde{A}lagr_{\lambda,\nu+1} = \tilde{a}_\nu^{(\lambda)}, \lambda = 1, \ldots, 2m, \nu = 0 \ldots, 2m-1.$
- $\tilde{\mathbf{z}} = [\tilde{z}_1, \ldots, \tilde{z}_{2m}]^T.$

We recall that $\tilde{a}_\nu^{(\lambda)}$ and $\tilde{z}_k$ have been defined in Sect. 4.

The computational kernel of this method consists of the formulation and the solution of the nonlinear system (36) of dimension m, at each step $j = 2, \ldots, N$. The memory term present in the nonlinear system (36) (i.e., the summation with index λ) gives a significant contribution to the computational effort; as a matter of fact, its cost amounts to $O(m^2N^2)$ flops. The FFT technique [33, 34] cannot be applied, since the memory term of the two-step spline collocation method can be recast as a discrete convolution product only for a uniform mesh. Instead, the considered method has a high order of convergence for a graded mesh (6) with a grading exponent r chosen according to (35), as illustrated in Sect. 5.

6.2 Vector Formulation of the Numerical Solution y_N

Let us introduce the $2m$-dimensional arrays, for $j = 2, \ldots, N$, $\lambda = 1, \ldots, j-1$:

$$\mathbf{b}_\mu^{[j]} = \sum_{\nu=0}^{2m-1} a_\nu^{(j,\mu)} \sigma^{\nu+\alpha} \frac{\Gamma(1+\nu)}{\Gamma(1+\nu+\alpha)},$$

$$\mathbf{g}_\mu^{[\lambda,j]} = \frac{1}{\Gamma(\alpha)} \sum_{\nu=0}^{2m-1} a_\nu^{(\lambda,\mu)} \left(\frac{t_{j-1} + \sigma h_j - t_{\lambda-1}}{h_\lambda}\right)^{\nu+\alpha} \beta\left(\frac{h_\lambda}{t_{j-1} + \sigma h_j - t_{\lambda-1}}; 1+\nu, \alpha\right),$$

$\mu = 1, \ldots, 2m$. Second, let us consider the following m-dimensional arrays:

- $\bar{\mathbf{b}}^{[j]}$ and $\bar{\mathbf{g}}^{[\lambda,j]}$, which consist of the first m elements of $\mathbf{b}^{[j]}$ and $\mathbf{g}^{[\lambda,j]}$, respectively
- $\tilde{\mathbf{b}}^{[j]}$ and $\tilde{\mathbf{g}}^{[\lambda,j]}$, which consist of the last m elements of $\mathbf{b}^{[j]}$ and $\mathbf{g}^{[\lambda,j]}$, respectively

Now we define the array $\mathbf{w}^{[\lambda,j]}$, of dimension m, with

$$\mathbf{w}^{[1,2]} = d_2^\alpha \bar{\mathbf{b}}^{[2]},$$

while, for $j = 3, \ldots, N$:

$$\mathbf{w}^{[\lambda,j]} = \begin{cases} d_2^{\alpha}\tilde{\mathbf{g}}^{[2,j]}, & \lambda = 1, \\ \tilde{\mathbf{g}}^{[\lambda,j]} + d_{\lambda+1}^{\alpha}\tilde{\mathbf{g}}^{[\lambda+1,j]}, & \lambda = 2, \ldots, j-2, \\ d_j^{\alpha}\tilde{\mathbf{b}}^{[j]} + \tilde{\mathbf{g}}^{[j-1,j]} & \lambda = j-1. \end{cases}$$

In the interval $[t_0, t_1]$, the solution is provided by the starting procedure illustrated in Sect. 4. The details of computation are object of [13].

In the interval $[t_1, b]$, the solution is given by (15), where the fractional integrals have been computed in (21), (22), and (28). Hence, assumed that $t = t_{j-1} + \sigma h_j$, with $\sigma \in [0, 1]$:

$$y_N(t) = h_j^{\alpha}\tilde{\mathbf{b}}_{\mu}^{[j]} \cdot \mathbf{z}_j + \sum_{\lambda=1}^{j-1} h_{\lambda}^{\alpha}\mathbf{w}^{[\lambda,j]} \cdot \mathbf{z}_{\lambda} + h_1^{\alpha} r(t) + Q(t), \quad t \in [t_{j-1}, t_j], \tag{37}$$

$j = 2, \ldots, N$. In (37), function $r(t)$ is defined as

$$r(t) = \frac{1}{\Gamma(\alpha)} \sum_{\lambda=1}^{2m} \tilde{z}_{\lambda} \sum_{\nu=1}^{2m} \tilde{a}_{\nu}^{(\lambda)} \left(\frac{t}{h_1}\right)^{\nu+\alpha} \beta\left(\frac{h_1}{t}, 1+\nu, \alpha\right), \quad t \in [t_1, b].$$

For the memory term present in formula (37), we can draw similar observation as at the end of Sect. 6.1. The computational cost of this part is $O(mj)$ flops.

Remark 2 We notice that, if $\sigma = \eta_k$, then $g_{\mu}^{[\lambda,j]} = E_{k,\mu}^{[\lambda,j]}$ and $b_{\mu}^{[j]} = A_{k,\mu}^{[j]}$. Moreover, if $t = t_{jk}$, then $r(t_{jk}) = (\mathbf{r}_j)_k$.

7 The MATLAB Algorithm

In this section, we describe the organization of the whole MATLAB algorithm, report all the programs, and list in Table 1 the MATLAB inbuilt functions used in the programs.

The main program of the algorithm is `tsfcoll.m` (Fig. 1). The first part of the program computes the starting solution in the interval $[0 = t_0, t_1]$. In the second part, the solution is computed in the rest of the time interval.

As explained in Sect. 4, the starting solution is computed by one-step collocation method, which is implemented in the code `fcoll.m` [13]. Therefore, for the `tsfcoll.m` to run, it is necessary to include all the routines called by the program `fcoll.m`, listed in [13]. In particular, from the same code for one-step collocation methods, the routine `matrix_Lagrange.m` is recalled to compute the coefficients of the Lagrange polynomial (24).

The solution in the rest of the interval $[t_1, b]$ is computed in a for loop.

Table 1 Auxiliary MATLAB functions adopted in the algorithm in alphabetical order

Function	Task
betainc	The incomplete beta function
ceil	Function $\lceil \cdot \rceil$
conv	Convolution of two vectors
diff	Differences between adjacent elements of array
gamma	The gamma function
factorial	The factorial
fliplr	Flips array left to right
fsolve	Solves a nonlinear system

```
function [t,y]=tsfcoll(f,b,gam,alpha,eta,r,N)

t=b*((0:N)'/N).^r;
h=diff(t);
m=length(eta);
etatilde=linspace(0,1,2*m)';
% starting solution in [t(1),t(2)]
[Alagr_tilde]=matrix_Lagrange(etatilde);
[~,ytilde,ztilde]=fcoll(f,t(2),gam,alpha,etatilde,r,1);
H=[fliplr(vander(eta)),diag(eta.^m)*fliplr(vander(eta))];
Z=zeros(m,N); Z(:,1)=H*Alagr_tilde'*ztilde;
y=zeros(N+1,1); y(1)=gam(1); y(2)=ytilde(2);

Alagr=zeros(2*m,2*m,N);
options=optimset('TolFun',1e-14,'TolX',1e-14,'Display','off');
for j=2:N
    tj=t(j)+eta*h(j);
    Alagr(:,:,j)=tsmatrix_Lagrange(eta,h,j);
    A=tsmatrix_A(alpha,eta,Alagr(:,:,j));
    E=tsmatrix_E(alpha,eta,Alagr,t,h,j);
    Bj=tslag(Z,ztilde,A,E,Alagr,Alagr_tilde,t,h,eta,alpha,j);
    Qj=Q(tj,alpha,gam);
    iniz=Z(:,j-1);
    Z(:,j)=fsolve(@tssystem_F,iniz,options,f,tj,A,Bj,Qj,h,j,alpha);

    b=tsmatrix_A(alpha,1,Alagr(:,:,j));
    g=tsmatrix_E(alpha,1,Alagr,t,h,j);
    Wj=tslag(Z,ztilde,b,g,Alagr,Alagr_tilde,t,h,1,alpha,j);
    Qjp1=Q(t(j+1),alpha,gam);
    y(j+1)=h(j)^(alpha)*b(:,m+1:2*m)*Z(:,j)+Wj+Qjp1;
end
end
```

Fig. 1 Main program `tsfcoll.m`

At each step j of the loop, the nonlinear system (36) is formulated in the function `tssystem_F.m` and then solved by MATLAB inbuilt function `fsolve`, with accuracy 10^{-14}. To construct the nonlinear system, the coefficients $a_\nu^{(j,\mu)}$ are computed in the routine `tsmatrix_Lagrange.m`, which is based on the procedure

```
function [Alagr]=tsmatrix_Lagrange(eta,h,j)

eta_h=[(eta'-1)*(h(j-1)/h(j)),eta']';
m_h=length(eta_h);
mu=1;
a=[1/(eta_h(mu)-eta_h(2)),-eta_h(2)/(eta_h(mu)-eta_h(2))];
for i=3:length(eta_h)
    b=[1/(eta_h(mu)-eta_h(i)),-eta_h(i)/(eta_h(mu)-eta_h(i))];
    a=conv(a,b);
end
Alagr(mu,:)=a;
for mu=2:m_h
    a=[1/(eta_h(mu)-eta_h(1)),-eta_h(1)/(eta_h(mu)-eta_h(1))];
    for i=2:mu-1
        b=[1/(eta_h(mu)-eta_h(i)),-eta_h(i)/(eta_h(mu)-eta_h(i))];
        a=conv(a,b);
    end
    for i=mu+1:m_h
        b=[1/(eta_h(mu)-eta_h(i)),-eta_h(i)/(eta_h(mu)-eta_h(i))];
        a=conv(a,b);
    end
    Alagr(mu,:)=a;
end
Alagr=fliplr(Alagr);
end
```

Fig. 2 Function `tsmatrix_Lagrange.m`

```
function A=tsmatrix_A(alpha,eta,Alagr)

m=size(Alagr,1)/2;
interv=0:2*m-1;
w=gamma(interv+1)./gamma(interv+1+alpha);
A=((eta.^((alpha+interv))).*w)*Alagr';
end
```

Fig. 3 Function `tsmatrix_A.m`

illustrated in Sect. 3, Remark 1 (Fig. 2). Moreover, matrices $\mathbf{A}^{[j]}$ and $\mathbf{E}^{[\lambda,j]}$ are constructed by routines `tsmatrix_A.m` and `tsmatrix_E.m`, respectively; the sum over the past computed solution is computed in `tslag.m`; the function $Q(t)$ is evaluated by routine `Q.m` (Figs. 3, 4, 5, 6). The nonlinear system (36) is returned by function `tssystem_F.m` (Fig. 7).

In the second part of each step j, the approximate solution y_N at the mesh points of the interval $[t_{j-1}, t_j]$ is computed, by formula (37) with $\sigma = 1$. This task is carried out by employing again functions `tsmatrix_A.m`, `tsmatrix_E.m`, `tslag.m`, thanks to observations made in Remark 2.

```
function E=tsmatrix_E(alpha,eta,Alagr,t,h,j)

s=length(eta);
m=size(Alagr,1)/2;
W=zeros(s,2*m);
E=zeros(s,2*m,j-1);
interv=0:2*m-1;
for lambda=2:j-1
    d=(t(j)-t(lambda)+eta*h(j))/h(lambda);
    for k=1:s
        W(k,:)= betainc(1/d(k),1+interv,alpha);
    end
    w=gamma(1+interv)./gamma(1+interv+alpha);
    E(:,:,lambda)=(((d.^(alpha+interv)).*w).*W)*Alagr(:,:,lambda)';
end
end
```

Fig. 4 Function `tsmatrix_E.m`

```
function lagterm=tslag(Z,ztilde,A,E,Alagr,Alagr_tilde,t,h,eta,alpha,j)

s=length(eta); m=size(Alagr,1)/2; N=length(t)-1;
hr=zeros(N,1);
for i=2:N
    hr(i)=h(i)/h(i-1);
end
hr=hr.^alpha;
lagterm=zeros(s,1);
if j==2
    B=hr(2)*A(:,1:m);
    lagterm=lagterm+h(1)^alpha*B*Z(:,1);
else
    B=hr(2)*E(:,1:m,2);
    lagterm=lagterm+h(1)^alpha*B*Z(:,1);
    for lambda=2:j-2
        B=E(:,m+1:2*m,lambda)+hr(lambda+1)*E(:,1:m,lambda+1);
        lagterm=lagterm+h(lambda)^alpha*B*Z(:,lambda);
    end
    B=hr(j)*A(:,1:m)+E(:,m+1:2*m,j-1);
    lagterm=lagterm+h(j-1)^alpha*B*Z(:,j-1);
end
interv=0:2*m-1;
d=(t(j)+eta*h(j))/h(1);
W=zeros(s,2*m);
for k=1:s
    W(k,:)= betainc(1/d(k),1+interv,alpha);
end
Etilde=((d.^(alpha+interv)).*W)*Alagr_tilde';
Etilde=Etilde/gamma(alpha);
lagterm=lagterm+h(1)^alpha*Etilde*ztilde;
end
```

Fig. 5 Function `tslag.m`

```
function z=Q(t,alpha,gam)
interv=0:ceil(alpha)-1;
z=sum(((t.^interv)./factorial(interv)).*(gam(interv+1)'),2);
end
```

Fig. 6 Function `Q.m`

```
function F=tssystem_F(x,f,tj,A,Bj,Qj,h,j,alpha)
m=size(A,1);
F=x-feval(f,tj,h(j)^(alpha)*A(:,m+1:2*m)*x+Bj+Qj);
end
```

Fig. 7 Function `tssystem_F.m`

We stress that all the routines have been organized to exploit MATLAB efficiency with array computation (see routines `tsmatrix_A.m`, `tsmatrix_E.m`, `tslag.m`, `Q.m`).

8 Input and Output Parameters

The mandatory input arguments of the code ftscoll.m are: `f,b,gam,alpha, eta,r,N`, while the output arguments are `t,y`. In this section, we describe each of them.

8.1 Input Parameters

1. **f**—function handle or string containing name of m-file.
   f must return the value of the function $f(t, y)$ at a given point (t, y).

```
[result] = f(t,y)
```

Input Parameters

- **t**—double scalar
  The current value of the independent variable t
- **y**—double scalar
  The current value of the independent variable y

Output Parameters

- **result**—double scalar
  The value of $f(t, y)$.

2. **b**—double scalar
   b, the end point of the integration interval $[0, b]$.
3. **gam**—double array
   The vector of the initial values $[\gamma_0, \ldots, \gamma_{n-1}]^T$ of the IVP (1).
   Constraint: The length n of the array **gam** should be equal to $\lceil \alpha \rceil$.
4. **alpha**—double scalar
   α, the fractional index.
5. **eta(m)**—double array
   eta is equal to the vector $[\eta_1, \ldots, \eta_m]^T$ of the collocation parameters.
   Constraint: $0 \leq$ **eta(1)** $\leq \cdots \leq$**eta(m)**≤ 1, (**eta(1)**,**eta(m)**)$\neq (0, 1)$.
6. **r**—double scalar
   r, the grading exponent. $r >= 1$.
7. **N**—double scalar
   N, the number of mesh points.

8.2 Output Parameters

1. **t(N+1)**—double array
   t is the graded mesh $[t_0, \ldots, t_N]^T$, defined in (6).
2. **y(N+1)**—double array
   $y(j)$ is the approximate value of the solution $y(t_{j+1})$.

9 Example of Usage

Here we illustrate on a test example how to compute the parameter ν that appears in the formula (35) and how to use our software to solve a FDE.

Problem 1 ([44])

$$D^{1/2}y(t) = y^2(t) + \frac{1}{\Gamma(1.5)}t^{0.5} - t^2, \ t \in [0, 1] \quad y(0) = 0, \tag{38}$$

with analytical solution $y(t) = t$. Hypotheses of Theorem 1 are fulfilled for any $q \in \mathbb{N}$ and $\nu = 0.5$. The verification is straightforward. As a matter of fact,

$$|f(t, y)| \leq \text{const} \cdot |y|^2, \quad \forall (t, y) \in (0, 1] \times \mathbb{R}$$

$$\left| \frac{\partial}{\partial t} f(t, y) \right| = \left| \frac{1}{2\Gamma(1.5)} t^{-0.5} - t \right| \leq \text{const} \cdot t^{-0.5}, \quad \forall (t, y) \in (0, 1] \times \mathbb{R},$$

and in general

$$\left| \frac{\partial^i}{\partial t^i} f(t, y) \right| \leq \text{const} \cdot t^{1-0.5-i}, \ \forall (t, y) \in (0, 1] \times \mathbb{R}, \quad i = 1, 2, \ldots.$$

Moreover,

$$\left|\frac{\partial^j}{\partial y^j} f(t,y)\right| \le 2|y|, \ \forall (t,y) \in (0,1] \times \mathbb{R}, \quad j = 1,2,\ldots$$
$$\frac{\partial^{i+j}}{\partial t^i \partial y^j} f(t,y) = 0, \quad \text{if } i \neq 0, \text{ and } j \neq 0.$$

Thus, condition (30) holds with $\phi(|y|) = \text{const} \cdot |y|^2$. Second, since $\alpha \in (0,1)$, we must verify condition (31) of Theorem 1, too. It follows from the following relations:

$$|f(t,y_1) - f(t,y_2)| = |y_1^2 - y_2^2| \le 2\max\{|y_1|,|y_2|\}|y_1 - y_2|,$$
$$\left|\frac{\partial}{\partial y} f(t,y_1) - \frac{\partial}{\partial y} f(t,y_2)\right| = 2|y_1 - y_2|,$$
$$\frac{\partial^j}{\partial y^j} f(t,y) = 0, \quad j = 2,3,\ldots$$
$$\frac{\partial}{\partial t} f(t,y_1) - \frac{\partial}{\partial t} f(t,y_2) = 0.$$

We show in Fig. 8 an example of script to set input arguments for Problem 1 (38) and report in Fig. 9 the corresponding output.

```
f=@(t,y) y.^2+1/gamma(1.5)*t.^0.5-t.^2;
b=1;
gam=[0];
alpha=1/2;
eta=[1/2,1]';
r=8;
N=b/2^-5;
[t,y]=tsfcoll(f,b,gam,alpha,eta,r,N);
err=abs(y(end)-b)
```

Fig. 8 Script `test_example.m`

```
>>test_example
err =

   8.2959e-06
```

Fig. 9 Output of `test_example.m`

10 Numerical Experiments

In this section, we will show on some test examples the accuracy of our code, to confirm theoretical expectations on the error. Moreover, we make comparison with the code proposed on [13], which implements the one-step spline collocation method and with the code `flmm2.m` [30], which implements a fractional BDF method.

We consider four test problems from the literature.

Problem 2 ([26])

$$\begin{aligned} D^{\alpha} y(t) &= \frac{40320}{\Gamma(9-\alpha)} t^{8-\alpha} - 3\frac{\Gamma(5+\frac{\alpha}{2})}{\Gamma(5-\frac{\alpha}{2})} t^{4-\frac{\alpha}{2}} + \left(\frac{3}{2} t^{\frac{\alpha}{2}} - t^4\right)^3 \\ &\quad + \frac{9}{4}\Gamma(\alpha+1) - (y(t))^{\frac{3}{2}}, \qquad t \in [0,1], \\ y(0) &= 0, \end{aligned}$$

with $\alpha = 1/2$. The analytical solution is $y(t) = t^8 - 3t^{4+\alpha/2} + \frac{9}{4}t^{\alpha}$. Hypotheses of Theorem 1 are fulfilled for any $q \in \mathbb{N}$ and $\nu = 0.5$.

Problem 3 ([8])

$$D^{\alpha} y(t) = \lambda y + \rho y(1-y^2) + g(t), \quad t \in (0,8], \qquad y(0) = 2,$$

with $\alpha = 0.3$, $\lambda = -3$, and $\rho = 0.8$. Function $g(t)$ is such that

$$y(t) = y_0 + \sum_{k=1}^{6} t^{\sigma_k}, \qquad \sigma_k = k\alpha,\ k = 1, \ldots, 5,\ \sigma_6 = 2+\alpha.$$

Hypotheses of Theorem 1 are fulfilled for any $q \in \mathbb{N}$ and $\nu = 1 - \alpha = 0.7$.

Problem 4 ([2])

$$\begin{aligned} &D^{\frac{5}{2}} y(t) + y(t) + y^2(t) = (1 + \mathrm{erf}(\sqrt{t}) e^t + e^{2t}, \quad t \in (0,1], \\ &y(0) = y'(0) = y''(0) = 1. \end{aligned}$$

The exact solution is $y(t) = \exp(t)$. Hypotheses of Theorem 1 are fulfilled for any $q \in \mathbb{N}$ and $\nu = 0.5$.

Problem 5 ([2])

$$\begin{aligned} &D^{\alpha} y(t) + (1+t^2) y^2(t) = \frac{t^{(1-\alpha)}}{(1-\alpha)\Gamma(1-\alpha)} + (1+t^2)(1+t)^2, \quad t \in (0,1], \\ &y(0) = 1, \end{aligned}$$

$\alpha = 0.4$. The exact solution is $y(t) = 1 + t$. Hypotheses of Theorem 1 are fulfilled for any $q \in \mathbb{N}$ and $\nu = 1 - \alpha$.

In the experiments, we set:

- $\eta = \left[\frac{1}{3}, \frac{2}{3}\right]$ when $m = 2$.
- $\eta = \left[\frac{1}{2}, \frac{3}{4}, 1\right]$ when $m = 3$.

The grading exponent is chosen as $r = 2m/(1 - \nu)$ (cfr. (35)).

In Fig. 10, we plotted the absolute error (err) at the end point versus the number of mesh points N, for two-step collocation method (TSC) with $m = 2$ and $m = 3$. The theoretical order of convergence $p = 2m$ is confirmed; moreover, the effective order for the method with $m = 2$ is equal to 4.5 (instead of 4) on the problems 3 and 5.

We computed the estimated error constant as $C = \text{err} \cdot N^{2m}$ and show these values for various values of N in Fig. 11. As expected from (34), the error constant increases with m. Similar results can be obtained for the other test problems.

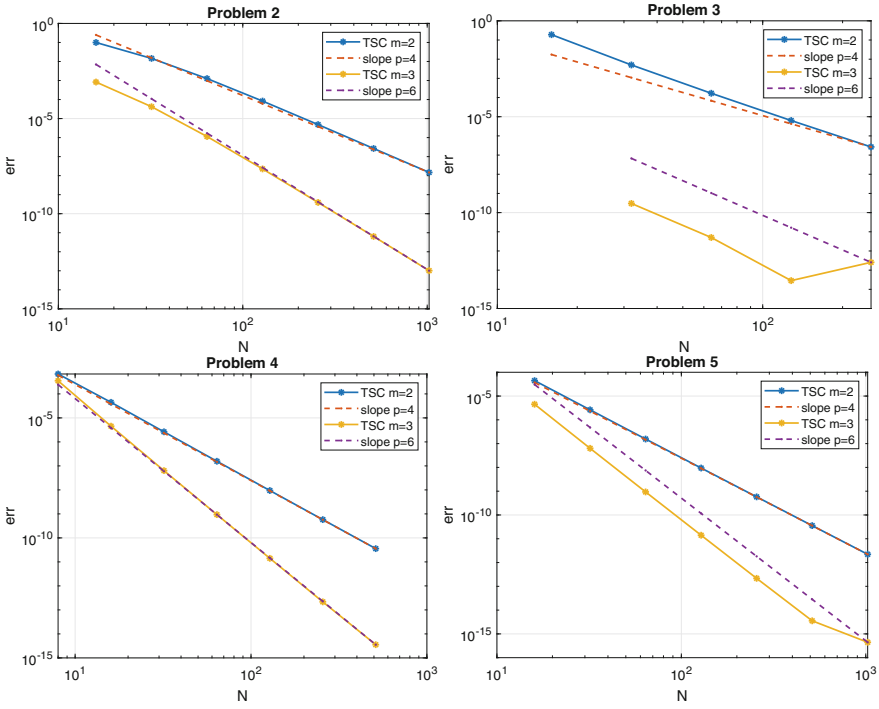


Fig. 10 Error for the two-step spline collocation method with m collocation points, compared with slope $p = 2m$. Logarithmic scale on x- and y-axis

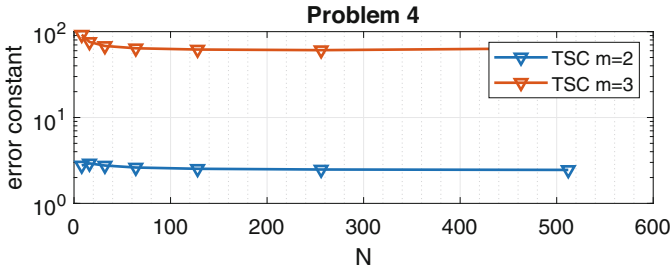


Fig. 11 Estimated error constant versus N for the two-step spline collocation method, on Problem 4. Logarithmic scale on the y-axis

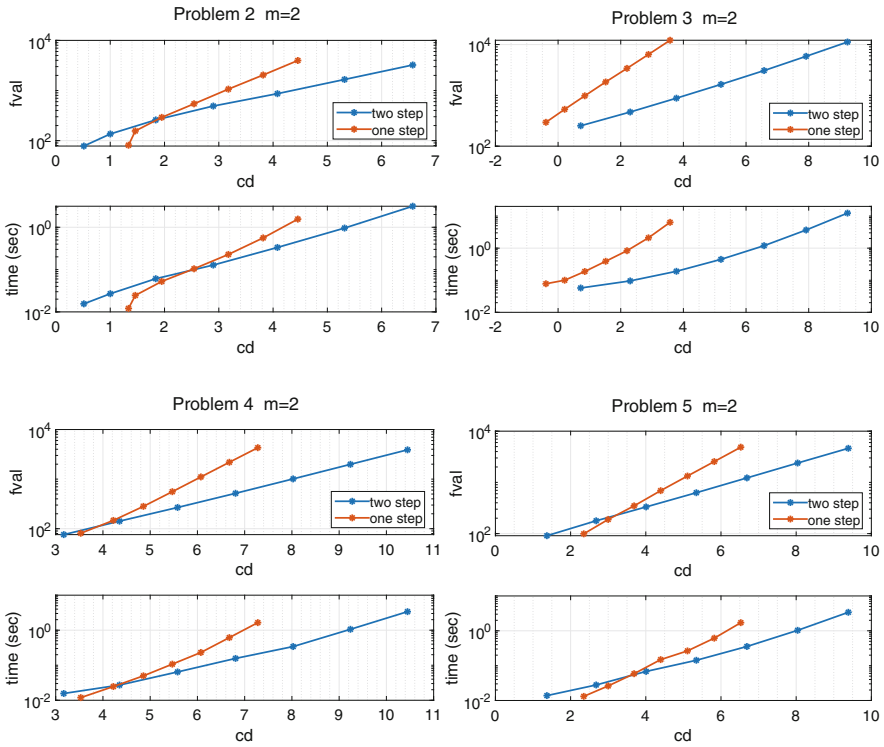


Fig. 12 Work precision diagrams and execution time versus N for the two-step spline collocation method with $m = 2$. Logarithmic scale on x- and y-axis

In Figs. 12 and 13, we compared the two-step collocation method implemented as showed in this chapter with the code `fcoll.m` [13], which implements the one-step spline collocation method proposed in [44]. For the one-step collocation methods, we considered the same collocation parameters, and we set $r = m/(1-\nu)$ (compare [13, 44]). In the top figures, we plotted the number of function evaluation (fval)

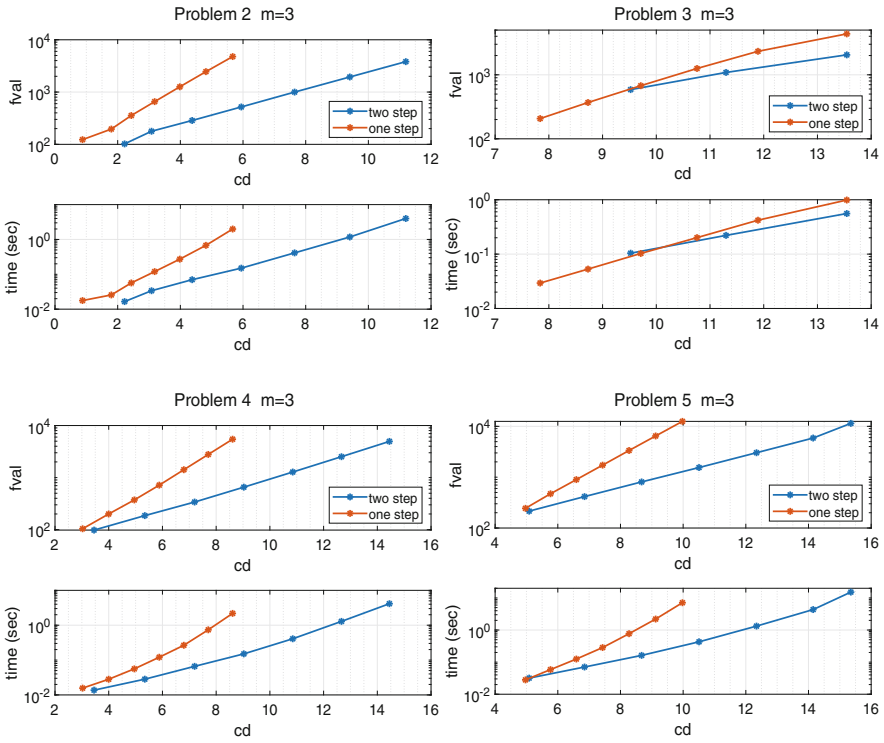


Fig. 13 Work precision diagrams and execution time versus N for the two-step spline collocation method with $m = 3$. Logarithmic scale on x- and y-axis

versus the number of correct digits, while at the bottom, we plotted the execution time (in seconds) versus the number of correct digits (cd). The two-step collocation methods provide better results in all test problems.

Finally, we compared our code with flmm2.m [30], available on MATLAB website. We set the input parameters as to apply a fractional BDF method of order 2. Figure 14 shows that our code is competitive especially when high degree of accuracy is required.

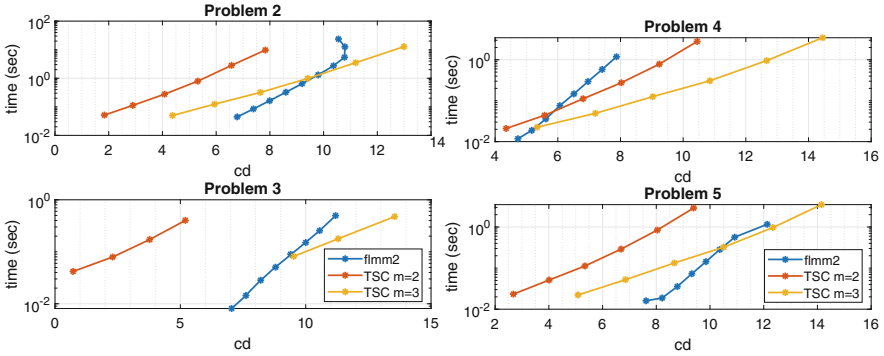


Fig. 14 Work precision diagrams and execution time versus N for the two-step spline collocation methods and for the code flmm2.m. Logarithmic scale on x- and y-axis

Acknowledgments The authors are members of the GNCS group. This work was supported by GNCS-INDAM project and by the Italian Ministry of University and Research (MUR) through the PRIN 2017 project (No. 2017JYCLSF) "Structure preserving approximation of evolutionary problems," and the PRIN 2020 project (No. 2020JLWP23) "Integrated Mathematical Approaches to Socio–Epidemiological Dynamics" (CUP: E15F21005420006).

References

1. Abramowitz, M., Stegun, I.A.: Handbook of mathematical functions with formulas, graphs, and mathematical tables. National Bureau of Standards Applied Mathematics Series, No. 55. U. S. Government Printing Office, Washington, D.C. (1964).
2. Babolian, E., Vahidi, A.R., Shoja, A.: An efficient method for nonlinear fractional differential equations: combination of the Adomian decomposition method and spectral method. Indian J. Pure Appl. Math. **45**(6), 1017–1028 (2014)
3. Bertaccini, D., Durastante, F.: Solving mixed classical and fractional partial differential equations using short-memory principle and approximate inverses. Numer. Algorithms **74**(4), 1061–1082 (2017)
4. Bialecki, B.: Convergence analysis of orthogonal spline collocation for elliptic boundary value problems. SIAM J. Numer. Anal. **35**(2), 617–631 (1998)
5. Blank, L.: Numerical treatment of differential equations of fractional order. Tech. rep., University of Manchester, Department of Mathematics (1996). Numerical Analysis Report
6. Brunner, H.: Collocation methods for Volterra integral and related functional differential equations. Cambridge Monographs on Applied and Computational Mathematics, vol. 15. Cambridge University Press, Cambridge (2004)
7. Burrage, K., Cardone, A., D'Ambrosio, R., Paternoster, B.: Numerical solution of time fractional diffusion systems. Appl. Numer. Math. **116**, 82–94 (2017)
8. Cao, W., Zeng, F., Zhang, Z., Karniadakis, G.E.: Implicit-explicit difference schemes for nonlinear fractional differential equations with nonsmooth solutions. SIAM J. Sci. Comput. **38**(5), A3070–A3093 (2016)
9. Capobianco, G., Cardone, A.: A parallel algorithm for large systems of Volterra integral equations of abel type. J. Comput. Appl. Math. **220**(1-2), 749–758 (2008)

10. Capobianco, G., Conte, D., Del Prete, I., Russo, E.: Fast Runge-Kutta methods for nonlinear convolution systems of Volterra integral equations. BIT **47**(2), 259–275 (2007)
11. Cardone, A., Conte, D.: Stability analysis of spline collocation methods for fractional differential equations. Math. Comput. Simul. **178**, 501–514 (2020)
12. Cardone, A., Conte, D., Paternoster, B.: Two-step collocation methods for fractional differential equations. Discrete Contin. Dyn. Syst. Ser. B **23**(7), 2709–2725 (2018)
13. Cardone, A., Conte, D., Paternoster, B.: A MATLAB implementation of spline collocation methods for fractional differential equations. Lect. Notes Comput. Sci. **12949 LNCS**, 387–401 (2021)
14. Cardone, A., Conte, D., Paternoster, B.: Numerical treatment of fractional differential models. In: M. Abdel Wahab (ed.) Proceedings of the 8th International Conference on Fracture, Fatigue and Wear, pp. 289–302. Springer Singapore, Singapore (2021)
15. Cardone, A., Conte, D., D'Ambrosio, R., Paternoster, B.: Multivalue collocation methods for ordinary and fractional differential equations. Mathematics **10**(2), 185 (2022)
16. Cardone, A., Conte, D., Paternoster, B.: On spline collocation methods for fractional differential equations. AIP Conf. Proc To appear
17. Cardone, A., Conte, D., Paternoster, B.: Stability analysis of two-step spline collocation methods for fractional differential equations. Commun. Nonlinear Sci. Numer. Simul. **115**, 106726 (2022)
18. Conte, D., Prete, I.D.: Fast collocation methods for Volterra integral equations of convolution type. J. Comput. Appl. Math. **196**(2), 652–663 (2006)
19. Conte, D., D'Ambrosio, R., D'Arienzo, M., Paternoster, B.: Multivalue mixed collocation methods. Appl. Math. Comput. **409**, 126346 (2021)
20. Daftardar-Gejji, V., Jafari, H.: Adomian decomposition: a tool for solving a system of fractional differential equations. J. Math. Anal. Appl. **301**(2), 508–518 (2005)
21. D'Ambrosio, R., Paternoster, B.: Multivalue collocation methods free from order reduction. J. Comput. Appl. Math. **387**, Paper No. 112515, 11 (2021)
22. Datsko, B.: Mathematical modeling of complex spatio-temporal dynamics in autocatalytic reaction-diffusion systems with anomalous diffusion. Comput. Math. Methods **3**(3), Paper No. e1112, 15 (2021)
23. de Boor, C., Swartz, B.: Collocation at Gaussian points. SIAM J. Numer. Anal. **10**, 582–606 (1973)
24. Deng, W.: Short memory principle and a predictor-corrector approach for fractional differential equations. J. Comput. Appl. Math. **206**(1), 174–188 (2007)
25. Diethelm, K.: The analysis of fractional differential equations. Lecture Notes in Mathematics, vol. 2004. Springer, Berlin (2010). An application-oriented exposition using differential operators of Caputo type
26. Diethelm, K., Ford, N.J., Freed, A.D.: Detailed error analysis for a fractional Adams method. Numer. Algorithms **36**(1), 31–52 (2004)
27. Dölz, J., Egger, H., Shashkov, V.: A fast and oblivious matrix compression algorithm for Volterra integral operators. Adv. Comput. Math. **47**(6), Paper No. 81, 24 (2021)
28. Fairweather, G., Meade, D.: A survey of spline collocation methods for the numerical solution of differential equations. In: Mathematics for Large Scale Computing. Lecture Notes in Pure and Appl. Math., vol. 120, pp. 297–341. Dekker, New York (1989)
29. Feng, L., Turner, I., Perré, P., Burrage, K.: An investigation of nonlinear time-fractional anomalous diffusion models for simulating transport processes in heterogeneous binary media. Commun. Nonlinear Sci. Numer. Simul. **92**, Paper No. 105454, 22 (2021)
30. Garrappa, R.: Trapezoidal methods for fractional differential equations: theoretical and computational aspects. Math. Comput. Simul. **110**, 96–112 (2015)
31. Garrappa, R.: Numerical solution of fractional differential equations: A survey and a software tutorial. Mathematics **6**(2), 16 (2018)
32. Garrappa, R., Kaslik, E., Popolizio, M.: Evaluation of fractional integrals and derivatives of elementary functions: Overview and tutorial. Mathematics **7**(5), 407 (2019)

33. Hairer, E., Lubich, C., Schlichte, M.: Fast numerical solution of nonlinear Volterra convolution equations. SIAM J. Sci. Stat. Comput. **6**(3), 532–541 (1985)
34. Hairer, E., Lubich, C., Schlichte, M.: Fast numerical solution of weakly singular Volterra integral equations. J. Comput. Appl. Math. **23**(1), 87–98 (1988)
35. Jafari, H., Yousefi, S.A., Firoozjaee, M.A., Momani, S., Khalique, C.M.: Application of Legendre wavelets for solving fractional differential equations. Comput. Math. Appl. **62**(3), 1038–1045 (2011)
36. Jia, J., Wang, H., Zheng, X.: A fast collocation approximation to a two-sided variable-order space-fractional diffusion equation and its analysis. J. Comput. Appl. Math. **388**, Paper No. 113234, 14 (2021)
37. Kilbas, A.A., Srivastava, H.M., Trujillo, J.J.: Theory and applications of fractional differential equations. North-Holland Mathematics Studies, vol. 204. Elsevier Science B.V., Amsterdam (2006)
38. Li, X.: Numerical solution of fractional differential equations using cubic B-spline wavelet collocation method. Commun. Nonlinear Sci. Numer. Simul. **17**(10), 3934–3946 (2012)
39. Li, Y., Sun, N.: Numerical solution of fractional differential equations using the generalized block pulse operational matrix. Comput. Math. Appl. **62**(3), 1046–1054 (2011)
40. Mainardi, F.: Fractional Calculus and Waves in Linear Viscoelasticity. Imperial College Press, London (2010). An Introduction to Mathematical Models
41. Moghaderi, H., Dehghan, M., Donatelli, M., Mazza, M.: Spectral analysis and multigrid preconditioners for two-dimensional space-fractional diffusion equations. J. Comput. Phys. **350**, 992–1011 (2017)
42. Paternoster, B.: A phase-fitted collocation-based Runge-Kutta-Nyström method. Appl. Numer. Math. **35**(4), 339–355 (2000)
43. Pedas, A., Tamme, E.: Spline collocation methods for linear multi-term fractional differential equations. J. Comput. Appl. Math. **236**(2), 167–176 (2011)
44. Pedas, A., Tamme, E.: Numerical solution of nonlinear fractional differential equations by spline collocation methods. J. Comput. Appl. Math. **255**, 216–230 (2014)
45. Pedas, A., Tamme, E.: Spline collocation for nonlinear fractional boundary value problems. Appl. Math. Comput. **244**, 502–513 (2014)
46. Petrás, I.: Fractional derivatives, fractional integrals, and fractional differential equations in Matlab. In: A.H. Assi (ed.) Engineering Education and Research Using MATLAB, chap. 10. IntechOpen, Rijeka (2011)
47. Podlubny, I.: Fractional differential equations. Mathematics in Science and Engineering, vol. 198. Academic Press, Inc., San Diego, CA (1999). An introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications
48. Schädle, A., López-Fernández, M., Lubich, C.: Fast and oblivious convolution quadrature. SIAM J. Sci. Comput. **28**(2), 421–438 (2006). All Open Access, Green Open Access
49. Scalas, E., Gorenflo, R., Mainardi, F.: Fractional calculus and continuous-time finance. Phys. A **284**(1-4), 376–384 (2000)
50. Scherer, R., Kalla, S.L., Tang, Y., Huang, J.: The Grünwald-Letnikov method for fractional differential equations. Comput. Math. Appl. **62**(3), 902–917 (2011)
51. Sheng, H., Chen, Y., Qiu, T.: Fractional Processes and Fractional-Order Signal Processing. Signals and Communication Technology. Springer London, London (2012). Techniques and applications, With a foreword by Richard L. Magin
52. Sowa, M., Kawala-Janik, A., Bauer, W.: Fractional differential equation solvers in Octave/Matlab. In: 2018 23rd International Conference on Methods & Models in Automation & Robotics (MMAR), pp. 628–633. IEEE (2018)
53. Wei, S., Chen, W.: A Matlab toolbox for fractional relaxation-oscillation equations. Preprint (2013). arXiv:1302.3384
54. Zayernouri, M., Karniadakis, G.E.: Exponentially accurate spectral and spectral element methods for fractional ODEs. J. Comput. Phys. **257**(part A), 460–480 (2014)
55. Zayernouri, M., Karniadakis, G.E.: Fractional spectral collocation method. SIAM J. Sci. Comput. **36**(1), A40–A62 (2014)