Numerical solution of the space-time fractional diffusion equation: Alternatives to finite differences

Emmanuel Hanert * **Cécile Piret** **

* *Earth and Life Institute, Université catholique de Louvain, 1348 Louvain-la-Neuve, Belgium (e-mail: emmanuel.hanert@uclouvain.be).*
** *Institute of Mechanics, Materials and Civil Engineering, Université catholique de Louvain, 1348 Louvain-la-Neuve, Belgium (e-mail: cecile.piret@uclouvain.be).*

Abstract: One of the ongoing issues with fractional-order diffusion models is the design of efficient numerical schemes for the space and time discretizations. Until now, most models have relied on a low-order finite difference method to discretize both the fractional-order space and time derivatives. While the finite difference method is simple and straightforward to solve integer-order differential equations, its appeal is reduced for fractional-order differential equations as it leads to systems of linear equation defined by large full matrices. Alternatives to the finite difference method exist but a unified presentation and comparison of these methods is still missing. In this paper, we compare 4 different numerical discretizations of the space-time fractional diffusion model. These consist of the finite difference, finite element, pseudo-spectral and radial basis functions methods. We suggest that non-local methods, like the pseudo-spectral and radial basis functions method, are well-suited to discretize the non-local operators like fractional-order derivatives. These methods naturally take the global behavior of the solution into account and thus do not result in an extra computational cost when moving from an integer-order to a fractional-order diffusion model.

Keywords: Space-time fractional diffusion equation, finite difference, finite element, pseudo-spectral and radial basis functions method, Mittag-Leffler functions

1. INTRODUCTION

One of the key issues with fractional-order diffusion models is the design of efficient numerical schemes for the space and time discretization. Until now, most models have relied on the finite difference (FD) method to discretize both the fractional-order space diffusion term (Meerschaert and Tadjeran, 2004, 2006; Tadjeran et al., 2006) and time derivative (Lin and Xu, 2007; Podlubny et al., 2009). Some numerical schemes using low-order finite elements (FE) have also been proposed (Fix and Roop, 2004; Roop, 2006). Fractional derivatives being non-local operators, they require a large number of operations and a large memory storage capacity when discretized with low-order FD and FE schemes. To reduce the computational burden, truncated numerical schemes based on a "short memory principle" (Podlubny, 1999) and a "logarithmic memory principle" (Ford and Simpson, 2001) have been proposed. Another approach to design an efficient numerical scheme is to discretize the equation with a non-local numerical method, *i.e.* a numerical method that naturally takes the global behavior of the solution into account. Following that approach, Hanert (2010, 2011) has proposed a pseudo-spectral (PS) method based on Chebyshev basis functions in space and Mittag-Leffler basis functions in time to discretize the space-time fractional diffusion equation. A similar approach has been followed by Li and Xu (2009) to discretize the time-fractional diffusion equation with a Jacobi PS method.

In this paper, we compare different numerical discretizations of the space-time fractional diffusion equation on the computational domain $[0, L]$. That equation can be expressed as follows:

$$ {}_0^C D_t^\gamma f(x,t) = K_{\alpha,\gamma} \left[\frac{1+\beta}{2} {}_0D_x^\alpha f(x,t) + \frac{1-\beta}{2} {}_xD_L^\alpha f(x,t) \right] \tag{1} $$

where ${}_0^C D_t^\gamma$ is the time-fractional Caputo derivative of order γ and ${}_0D_x^\alpha$ and ${}_xD_L^\alpha$ are the left and right space-fractional Riemann-Liouville derivatives on $[0, L]$ (see for instance (Podlubny, 1999; Li and Deng, 2007)). The parameter $\beta \in [-1, 1]$ is a skewness parameter representing a preferential direction of jumps that can be observed in heterogeneous systems. When $\beta = 0$, the space derivative reduces to a so-called symmetric Riesz derivative (Podlubny, 1999). The coefficient $K_{\alpha,\gamma}$ is a generalized diffusivity whose dimension is $[K_{\alpha,\gamma}] = \mathrm{m}^\alpha \mathrm{s}^{-\gamma}$. The fractional-order derivatives can be defined as follows:

$$ {}_0^C D_t^\gamma u(t) = \frac{1}{\Gamma(1-\gamma)} \int_0^t \frac{\frac{\partial u(\theta)}{\partial \theta}}{(t-\theta)^\gamma} \, \mathrm{d}\theta, $$

$$ {}_0D_x^\alpha v(x) = \frac{1}{\Gamma(2-\alpha)} \frac{\partial^2}{\partial x^2} \int_0^x \frac{v(y)}{(x-y)^{\alpha-1}} \, \mathrm{d}y, $$

$$ {}_xD_L^\alpha v(x) = \frac{(-1)^2}{\Gamma(2-\alpha)} \frac{\partial^2}{\partial x^2} \int_x^L \frac{v(y)}{(y-x)^{\alpha-1}} \, \mathrm{d}y, $$

where $\Gamma(.)$ is Euler's gamma function. For simplicity, we have assumed that $1 < \alpha \leq 2$ and $0 < \gamma \leq 1$. The Riemann-Liouville derivatives being singular at the domain boundaries, they are often replaced by space-fractional Caputo derivatives of order α:

$$ {}_0^C D_x^\alpha v(x) = \frac{1}{\Gamma(2-\alpha)} \int_0^x \frac{\frac{\partial^2 v(y)}{\partial y^2}}{(x-y)^{\alpha-1}} \, \mathrm{d}y, $$

$$ {}_x^C D_L^\alpha v(x) = \frac{(-1)^2}{\Gamma(2-\alpha)} \int_x^L \frac{\frac{\partial^2 v(y)}{\partial y^2}}{(y-x)^{\alpha-1}} \, \mathrm{d}y. $$

2. DISCRETIZATION OF THE FRACTIONAL-ORDER DIFFUSION TERM

In this section, we present 4 different discretization of the fractional-order diffusion term in Eq. (1) and discuss their respective advantages and disadvantages. The discretizations are presented by order of popularity (but not efficiency or accuracy!). The finite difference is currently the most widely used method to solve the fractional-order diffusion equation. It is then followed by the finite element, pseudo-spectral and radial basis functions methods. To our knowledge, the latter has never been used to solve the fractional-order diffusion equation.

2.1 Finite difference method

A FD discretization of the fractional-order ADE can be obtained by using the so-called Grünwald approximation of the Riemann-Liouville derivatives (Podlubny, 1999; Meerschaert and Tadjeran, 2004). Such a FD scheme can easily be implemented and generalized to higher dimensions. It is only first order accurate in space but Tadjeran et al. (2006) have proposed a method to improve the accuracy to second order. By partitioning the computational domain in $N-1$ segments of length $\Delta x = L/(N-1)$, we can consider the following FD discretization of the fractional diffusion term:

$$ K_{\alpha,\gamma} \left[\frac{1+\beta}{2} {}_0D_x^\alpha f(x,t) + \frac{1-\beta}{2} {}_xD_L^\alpha f(x,t) \right] \approx \mathsf{D}_{ij} f_j(t), $$

where $f_j(t)$ $(1 \leq j \leq N)$ are the time-dependent FD nodal values and D is the matrix resulting from the FD discretization of the diffusion term. The elements of D have the following expression:

$$ \begin{aligned} \mathsf{D}_{ij} &= \frac{1+\beta}{2} w_{i-j+1} && \text{if } j < i-1, \\ &= \frac{1+\beta}{2} w_{i-j+1} + \frac{1-\beta}{2} w_{j-i+1} && \text{if } i-1 \leq j \leq i+1, \\ &= \frac{1-\beta}{2} w_{j-i+1} && \text{if } j > i+1, \end{aligned} $$

where $w_k = \frac{K_{\alpha,\gamma}}{\Delta x^\alpha} g_k$ and

$$ g_k = \frac{\Gamma(k-\alpha)}{\Gamma(-\alpha)\Gamma(k+1)} \; (0 \leq k \leq N) $$

are the normalized Grünwald weights.

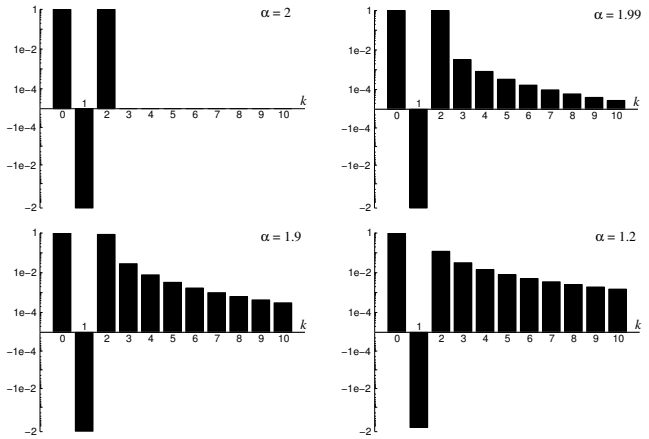

Fig. 1. First Grünwald weights g_k for different values of the exponent coefficient α and $k = 0, 1, \ldots, 10$. When $\alpha = 2$, only weights corresponding to $k = 0, 1$ and 2 are non-zero. As soon as $\alpha < 2$, all the other weights become positive and see their value increase as α decreases. Note that the y-axis is logarithmic and discontinuous between positive and negative values.

As shown in Fig. 1, the FD stencil used to discretize the fractional-order derivative covers the whole domain as soon as $\alpha < 2$. This could obviously be expected as fractional-order derivatives are global operators that have more similarities with integrals than with traditional derivatives. However, since the FD method usually requires a large number of nodes to obtain a precise approximation, the resulting increase in the computational cost can be important. In 1D, the number of operations required to compute the fractional-order derivative at a given node will be about $N/3$ times larger than the number of operations required to compute a second order derivative. Moreover, the use of an implicit time integration scheme seems totally prohibitive as it would require to solve a large full-matrix system of equations.

2.2 Finite element method

With the FE method, the exact solution is approximated by an expansion in terms of piecewise polynomials $\phi_j(x)$:

$$ f(x,t) \approx \sum_{j=1}^{N} f_j(t) \phi_j(x). \tag{2} $$

The basis function $\phi_j(x)$ has a compact support and corresponds to the node j of a grid (or mesh) that partitions the computational domain (see Fig. 2a for an example). The position of node j is denoted x_j. Unlike the FD method, the FE method is based on the exact expression of the fractional derivative. The latter is a linear operator and it can thus be easily applied to the FE expansion. Computing the fractional derivative of the solution then amounts to compute the fractional derivative of all basis functions $\phi_j(x)$. Since these are low-order piecewise polynomials, all the calculations can be done analytically (see Hanert (2010) for details).

With the FE methods, it is common practice to use a Galerkin formulation to derive the discrete equations. That formulation amounts to write the model equation in its weak form and perform integration by parts when

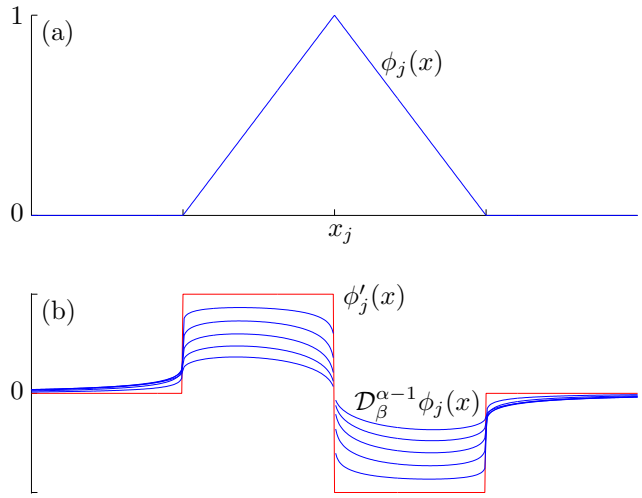

Fig. 2. Example of (a) a linear FE basis function ϕ_j, and (b) the first-order derivative (ϕ'_j, in red) and the fractional-order derivatives ($\mathcal{D}_\beta^{\alpha-1}\phi_j$, in blue) of ϕ_j for $\alpha = 1.5, 1.6, 1.7, 1.8, 1.9$ and $\beta = 0$. Note that although ϕ_j and ϕ'_j are local-support functions, the fractional derivatives are not.

possible. Doing so, one can remove some of the singularities of the Riemann-Liouville derivative. The fractional-order diffusion term is then discretized as follows:

$$K_{\alpha,\gamma}\left[\frac{1+\beta}{2}{}_0D_x^\alpha f(x,t) + \frac{1-\beta}{2}{}_xD_L^\alpha f(x,t)\right] \approx \mathsf{D}_{ij} f_j(t),$$

where the elements of the diffusion matrix D read:

$$\mathsf{D}_{ij} = -K_{\alpha,\gamma}\int_0^L \frac{\mathrm{d}\phi_i}{\mathrm{d}x}(x)\mathcal{D}_\beta^{\alpha-1}\phi_j(x)\ \mathrm{d}x,$$

and

$$\mathcal{D}_\beta^{\alpha-1}\phi_j(x) \equiv \frac{1+\beta}{2}{}_0D_x^{\alpha-1}\phi_j(x) + \frac{1-\beta}{2}{}_xD_L^{\alpha-1}\phi_j(x)$$

is a combination of the left and right Riemann-Liouville derivatives of order $\alpha-1$ of ϕ_j. That function is no more a compactly supported function as it spans the entire computational domain as soon as $\alpha < 2$ (see Fig. 2b). More details about the derivation can be found in the papers by Fix and Roop (2004), Roop (2006) and Hanert (2010).

Compared to the FD method, the FE method has the advantage of allowing a straightforward extension to high-order discretizations of the fractional diffusion equations. Indeed, it is based on the exact expression of the fractional-order derivative and not on an approximation like the Grunwald expansion. Therefore increasing the order of the scheme can simply be achieved by increasing the polynomial order of the basis functions. While the FE method can be more accurate than the FD method, it is not really more efficient since it is based on an expansion of the solution in terms of piecewise basis functions. These basis functions and their integer-order derivatives have a local support and thus lead to sparse discrete operators. However, the fractional order derivative of a FE basis function is not a compactly supported function anymore. The resulting diffusion matrix is therefore also a large full matrix and the computational issues mentioned for the FD method remain.

2.3 Pseudo-spectral method

Like the FE method, the PS method is also based on an expansion of the solution in terms of polynomial basis functions. That expansion has the same form as (2). However, the basis functions $\phi_j(x)$ are now global high-order functions that span over the entire computational domain. For problems with non-periodic boundary conditions, Chebyshev or Legendre polynomials are often used as basis functions (Boyd, 2001). As an illustration, the 8th-order Chebyshev polynomials $\phi_8(x)$ is represented in Fig. 3a. All these basis functions take non-zero values on the boundaries of the domain and their Riemann-Liouville fractional derivative therefore diverges on the boundaries. Such a spurious behaviour can be avoided by using instead a Caputo fractional derivative in the diffusion term. With a Galerkin formulation, the following diffusion matrix is then obtained:

$$\mathsf{D}_{ij} = K_{\alpha,\gamma}\int_0^L \phi_i(x){}^C\mathcal{D}_\beta^\alpha\phi_j(x)w(x)\ \mathrm{d}x.$$

where ${}^C\mathcal{D}_\beta^\alpha$ represents the combination of the left and right Caputo derivatives:

$${}^C\mathcal{D}_\beta^\alpha\phi_j(x) = \frac{1+\beta}{2}{}_0^CD_x^\alpha\phi_j(x) + \frac{1-\beta}{2}{}_x^CD_L^\alpha\phi_j(x)$$

and $w(x)$ is the weight function associated with Chebyshev polynomials. The fractional derivative of $\phi_8(x)$, ${}^C\mathcal{D}_\beta^\alpha\phi_8(x)$, is represented in Fig. 3b for different values of α. It can be seen that the second-order and all the fractional-order derivatives are global functions.

Thanks to the use of high-order basis functions, the PS method can achieve the same accuracy as the FD and FE methods with much less degrees of freedoms (dof's). Since the basis functions span over the entire domain, the diffusion matrix is always a full matrix whatever the order of the diffusion term. Therefore the computational cost of the numerical scheme is not substantially increased when going from a second-order to a fractional-order diffusion equation. In each case, we have to handle a small full diffusion matrix. The main disadvantage of the PS method is that high-order approximations are prone to Gibbs oscillations when the model solution is not smooth. The number of dof's required to obtain a good approximation of a steep function might also be pretty large. For those situations, the FD or FE methods might perform better.

2.4 Radial basis functions method

With the RBF method, the model solution is generally expressed as follows:

$$f(x,t) \approx \sum_{j=1}^N f_j(t)\phi(|x-x_j|),$$

where $\phi(r)$ is a radial function and x_j its center. There are many possible radial functions (see for instance Fasshauer (2007)) but here we will only consider the Gaussian radial functions $\phi(r) = e^{-(\varepsilon r)^2}$, where the shape parameter ε controls the flatness of the function. By introducing a set of basis functions $\{\phi_j(x)\}_{j=1}^N$ such that $\phi_j(x) = \phi(|x-x_j|)$, we can write the RBF expansion in the same form as (2). With the RBF method, a collocation formulation is

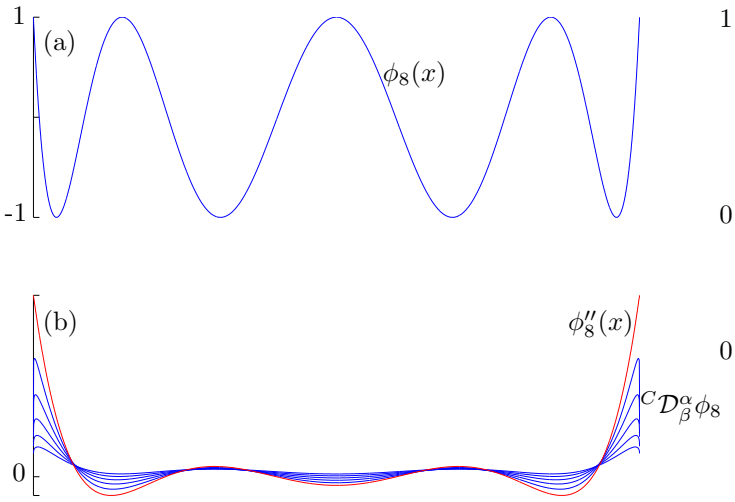

Fig. 3. Example of (a) the 8th-order Chebyshev PS basis function ϕ_8, and (b) the second-order derivative (ϕ_8'', in red) and the fractional-order derivatives (${}^C\mathcal{D}_\beta^\alpha\phi_8$, in blue) of ϕ_8 for $\alpha = 1.5, 1.6, 1.7, 1.8, 1.9$ and $\beta = 0$. Note that ϕ_8, ϕ_8'' and all the fractional derivatives are global functions.

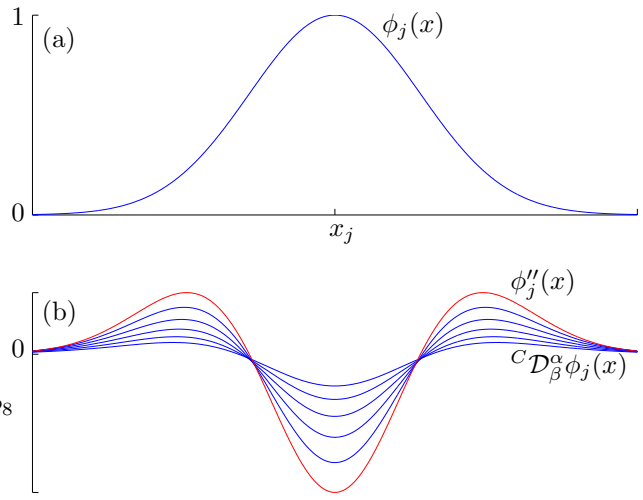

Fig. 4. Example of (a) Gaussian RBF ϕ_j, and (b) the second-order derivative (ϕ_j'', in red) and the fractional-order derivatives (${}^C\mathcal{D}_\beta^\alpha\phi_j$, in blue) of ϕ_j for $\alpha = 1.5, 1.6, 1.7, 1.8, 1.9$ and $\beta = 0$. Note that ϕ_j, ϕ_j'' and all the fractional derivatives are global functions.

generally used to derive the discrete equations. In that case, the diffusion matrix takes the following form:

$$\begin{aligned}\mathsf{D}_{ij} &= K_{\alpha,\gamma}\left[\frac{1+\beta}{2}{}_0^C D_x^\alpha\phi_j(x_i) + \frac{1-\beta}{2}{}_x^C D_L^\alpha\phi_j(x_i)\right],\\ &= {}^C\mathcal{D}_\beta^\alpha\phi_j(x_i),\end{aligned}$$

where we have again replaced the Riemman-Liouville derivative by a Caputo derivative to avoid the singularities on the boundaries. Fig. 4 shows an example of a Gaussian radial function and its fractional derivatives. Details on how to compute the fractional derivative of a radial function can be found in Piret and Hanert (2012).

The RBF method can be seen as a sort of compromise between the FE and the PS methods. On the one hand, the RBF method is based on an expansion into basis functions that have a spatial location like with the FE method. In that sense, these basis functions can be clustered in a specific region to locally increase the accuracy of the method. On the other hand, the radial functions used in the RBF expansion are high-order functions that span the entire domain like with the PS method. The parameter ε determines the degree of locality of the radial function. Increasing (resp. decreasing) the value of ε makes the radial function more local (resp. global). For smooth solutions, it is thus better to select a small value of ε while a better representations of solutions with steep gradients will be achieved by clustering basis functions with a larger value of ε in the region where the solution is steep.

Driscoll and Fornberg (2002) proved that when using the RBF method, we recover PS methods in the limit of $\epsilon \to 0$. Thus the RBF method is a generalization to irregular domains and scattered nodes of PS methods (Fornberg and Piret, 2008). However, although lowering the shape parameter value tends to improve the accuracy, it also has the detrimental effect of increasing the condition number of the collocation matrix. Thus using the direct implementation of the RBF method does not allow us to fully take advantage of the RBF properties. In order to avoid this ill-conditioning issue, Fornberg and Piret (2007) introduced the RBF-QR method, which consists of finding a new basis, which spans exactly the same space as the original radial function translates, but whose terms are strongly linearly independent with each other. The RBF-QR method thus allows us to safely compute and evaluate RBF expansions for any small value of the shape parameter. If one finds a better accuracy when the shape parameter is very small, why should one use the RBF method rather than a PS method? The first reason is that RBFs are suitable for local node refinements that provides local resolution enhancements. Secondly, one can use the flexibility of both node location and shape parameter values to eliminate the Gibbs phenomenon (Fornberg and Zuev, 2007). Thirdly, while a node distributions on different geometries can lead to singular systems with the PS method, the RBF collocation matrix remains unconditionally nonsingular in any dimension and on any geometry under minimal constraints. In addition, even the RBF-QR code is fairly simple to implement and the new basis is usually given in terms of combinations of polynomials and exponentials, for which the fractional derivatives are easy to compute.

3. DISCRETIZATION OF THE FRACTIONAL-ORDER TIME DERIVATIVE

A FD or a FE scheme could be used to discretize the fractional-order time derivative in a similar fashion as for the diffusion term. However that would be rather inefficient as the solution values at all the previous time steps would have to be kept in memory. Instead, we prefer to consider an approach based on the PS method in order to limit as much as possible the computational cost associated with the discrete fractional time derivative. The following PS expansion in time of the model solution is thus considered:

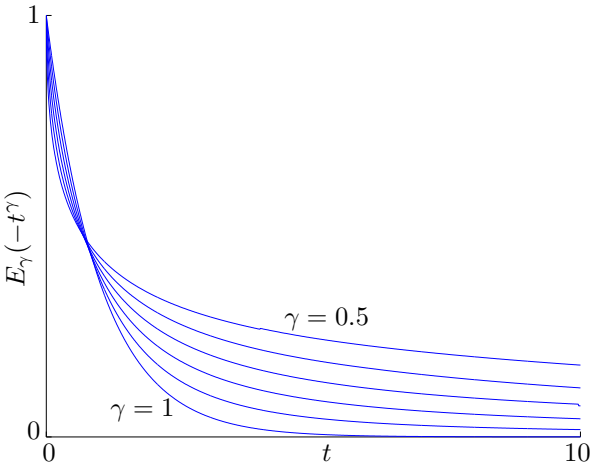

Fig. 5. Behaviour of the Mittag-Leffler function $E_\gamma(-t^\gamma)$ for $\gamma = 0.5, 0.6, 0.7, 0.8, 0.9, 1$. It decays exponentially when $\gamma = 1$ and algebraically (as a power-law) as soon as $\gamma < 1$.

$$f(x,t) \approx \sum_{k=-M}^{M} f_k(x)\psi_k(t),$$

where $\{\psi_k(t)\}_{k=-M}^{M}$ is a given set of basis functions. The coefficients $f_k(x)$ still depend on x since only the time discretization is considered at the moment. Chebyshev polynomials could still be used for the time expansion but other choices other choices could prove more efficient. In the present work, we have considered the PS method with Mittag-Leffler basis functions introduced by Hanert (2011).

The Mittag-Leffler function $E_\gamma(t)$ is defined as follows:

$$E_\gamma(t) = \sum_{n=0}^{\infty} \frac{t^n}{\Gamma(\gamma n + 1)}$$

and can be seen as a generalization of the exponential function since $\Gamma(n+1) = n!$ and thus $E_1(t) = \exp(t)$. When the order γ is not an integer, these functions exhibit power-law asymptotic behaviour (Mainardi and Gorenflo, 2000). Interestingly, Mittag-Leffler functions are eigenfunctions of the Caputo fractional-derivative of order $\gamma \leq 1$ (see for instance Mainardi and Gorenflo (2000)):

$${}_0^C D_t^\gamma E_\gamma(\omega t^\gamma) = \frac{1}{\Gamma(1-\gamma)} \int_0^t \frac{\frac{\mathrm{d}}{\mathrm{d}\tau} E_\gamma(\omega \tau^\gamma)}{(t-\tau)^\gamma}\, \mathrm{d}\tau = \omega E_\gamma(\omega t^\gamma).$$

Fig. 5 shows a sketch of the function $E_\gamma(-t^\gamma)$, which is the solution of the fractional relaxation equation ${}_0^C D_t^\gamma g(t) = -g(t)$ with the initial condition $g(0) = 1$. A long-tail, power-law behaviour is observed as soon as $\gamma < 1$. As the value of γ decreases, the thickness of the tail increases, indicating a slowly-decaying, scale-free memory effect. Mittag-Leffler functions thus generalize the classical exponential relaxation to systems with a non-Markovian dynamics (Metzler and Klafter, 2002).

By considering the following Mittag-Leffler basis functions:

$$\psi_k(t) = E_\gamma\left(ik\left(\frac{2\pi t}{T}\right)^\gamma\right)$$

for $k = -M, \ldots, M$ and using a Galerkin formulation, we can easily compute the time-derivative matrix T resulting from the discretization of ${}_0^C D_t^\gamma f(x,t)$:

$$\begin{aligned}\mathsf{T}_{lk} &= \int_0^T \psi_l(t)\, {}_0^C D_t^\gamma \psi_k(t)\, \mathrm{d}t, \\ &= ik\left(\frac{2\pi}{T}\right)^\gamma \int_0^T \psi_l(t)\psi_k(t)\, \mathrm{d}t, \\ &\equiv ik\left(\frac{2\pi}{T}\right)^\gamma \mathsf{N}_{lk},\end{aligned}$$

where T is the duration of the simulation and N is the mass matrix associated with the basis functions $\psi_k(t)$.

FD	FE	PS	RBF
δ_{ij}	$\int_0^L \phi_i(x)\phi_j(x)\, \mathrm{d}x$	$\int_0^L \phi_i(x)\phi_j(x)w(x)\, \mathrm{d}x$	$\phi_j(x_i)$

Table 1. Expression of the mass matrix elements M_{ij} for the different space discretizations.

4. DISCRETIZATION OF THE FULL SPACE-TIME FRACTIONAL DIFFUSION EQUATION

We now consider the discretization in space and time of Eq. (1). For the time discretization, we use the Mittag-Leffler PS scheme derived in the previous section. For the space discretization, we would like to have the possibility to use either the FD, FE, PS or RBF methods. For the sake if illustration, let us assume that we approximate the solution $f(x,t)$ with a series-expansion in terms of some basis functions $\psi_k(t)$ and $\phi_j(x)$. The former correspond to the Mittag-Leffler basis functions previously defined while the latter can be either FE, PS or RBF basis functions. The discrete solution can then be expressed in terms of a matrix of unknown nodal values F_{jk}:

$$f(x,t) \approx \sum_{j=1}^{N}\sum_{k=-M}^{M} \phi_j(x)\mathsf{F}_{jk}\psi_k(t).$$

By using the Mittag-Leffler PS scheme in time and the relevant space discretization, we obtain a set of discrete equations that can be expressed in matrix form as follows:

$$\mathsf{MFT} = \mathsf{DFN}, \tag{3}$$

where M and D are the mass and diffusion matrices, respectively, corresponding to the selected spatial discretization. The expression of the mass matrices M obtained with the FD, FE, PS and RBF is given in Table 4. At this stage, it is important to note that the space and time discretizations are totally independent in Eq. (3) and that the matrices M and D could also have been obtained by using the FD method in space. Therefore, although a series-expansion method has been used to illustrate the derivation of Eq. (3), the FD method could fit within that formulation as well.

In order to be able to solve Eq. (3), we shall first recast it in a more convenient form. To do so, we make use of the Kronecker product (represented by "$\otimes$") to express Eq. (3) as follows:

$$(\mathsf{T}^t \otimes \mathsf{M} - \mathsf{N}^t \otimes \mathsf{D})\mathrm{vec}(\mathsf{F}) = 0, \tag{4}$$

where $\mathrm{vec}(\mathsf{F})$ is the vector obtained by stacking the columns of F on top of one another (see Appendix for

details) and the superscript t denotes the transpose. Eq. (4) has to be supplemented with the initial and boundary conditions. On the one hand, the initial condition $f(x,0) = f_0(x)$ can be discretized with a Galerkin formulation in space as follows:

$$\left\langle \phi_i \phi_j \right\rangle_x \mathsf{F}_{jk} \psi_k(0) = \left\langle \phi_i f_0(x) \right\rangle_x,$$

where $\langle . \rangle_x$ represents the integral over space. In matrix form, it reads

$$(\Psi(0)^t \otimes \mathsf{M}) \mathrm{vec}(\mathsf{F}) = \left\langle \Phi(x) f_0(x) \right\rangle_x,$$

where $\Psi(0) = (\psi_{-N}(0), \ldots, \psi_N(0))^t$ and $\Phi(x) = (\phi_0(x), \ldots, \phi_M(x))^t$. On the other hand, the left boundary condition $c_d f(0,t) + c_n \frac{\partial f}{\partial x}(0,t) = f_l(t)$, where c_d and $c_n \in [0,1]$ are coefficients that allow us to specify Dirichlet or Neumann boundary conditions, can be discretized as follows

$$\left(c_d \phi_j(0) + c_n \frac{\partial \phi_j}{\partial x}(0) \right) \mathsf{F}_{jk} \left\langle \psi_k \psi_l \right\rangle_t = \left\langle f_l(t) \psi_l \right\rangle_t.$$

In matrix form, it reads

$$\left(\mathsf{N}^t \otimes \left(c_d \Phi^t(0) + c_n \frac{\partial \Phi^t}{\partial x}(0) \right) \right) \mathrm{vec}(\mathsf{F}) = \left\langle f_l(t) \Psi(t) \right\rangle_t.$$

The discretization of the right boundary condition at $x = L$ is similar.

Note that the matrix approach presented here shares similarities with the approach presented by Podlubny et al. (Podlubny et al., 2009). However, the present approach does not require homogeneous initial and boundary conditions. It can also easily accommodate space-dependent, linear reaction terms in a similar fashion as in Podlubny et al. (2009). For non-linear reaction terms, like for instance in the fractional-order Fisher equation, a non-linear solver would be required.

5. NUMERICAL EXAMPLES

5.1 Convergence analysis

The accuracy and efficiency of the 4 spatial discretizations can be assessed by performing a convergence analysis. For simplicity, we consider the one-dimensional benchmark problem introduced by Sousa (2011) which consists in finding $f(x,t)$ such that

$$\frac{\partial f(x,t)}{\partial t} = d(x) {}_0D_x^\alpha f(x,t) + q(x,t) \quad \text{for } x \in [0,1] \text{ and } t > 0, \tag{5}$$

with $d(x) = \frac{\Gamma(5-\alpha)}{24} x^\alpha$, $q(x,t) = -2e^{-t}x^4$, $f(x,0) = x^4$, $f(0,t) = 0$ and $f(1,t) = e^{-t}$. In that case, the exact solution of (5) reads:

$$f(x,t) = e^{-t} x^4.$$

Eq. (5) involves only a left-sided Riemann-Liouville fractional derivative and thus amounts to set $\gamma = 1$ and $\beta = 1$ in Eq. (1). Note that both the solution slope and value vanish on the left boundary and hence the left Riemann-Liouville and Caputo derivatives of $f(x,t)$ are totally equivalent. Eq. (5) has been discretized with the FD, FE, PS and RBF schemes and solved until $t = 1$. A second-order Crank-Nicolson time integration scheme with a time step equal to 10^{-3} has been used. At the end of the simulation, the numerical solutions have been compared with the exact solution.

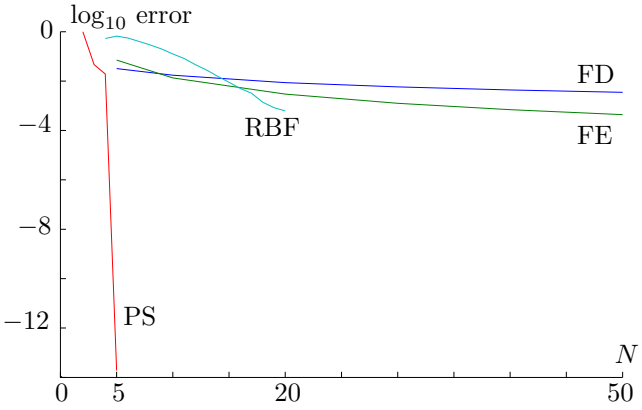

Fig. 6. Convergence analysis of the FD, FE, PS and RBF methods for Sousa (2011)'s test case.

Fig. 6 shows the rate of convergence of the 4 methods. For the FE, PS and RBF schemes, the relative error has been computed in the L_2 norm. A root mean square error has been computed for the FD scheme. As expected, the convergence rate of the FD scheme is linear but it could be increased to quadratic by using the approach proposed by Tadjeran et al. (2006). The FE scheme basis functions being piecewise linear, its convergence rate is quadratic. The PS scheme converges very quickly to the exact solution. This was obviously expected as the exact solution is a 4th-order polynomial in space. A PS expansion with only 5 Chebyshev polynomials is therefore sufficient to represent it exactly. A test case with a more complex solution would be necessary to clearly highlight the exponential convergence rate of the PS method. Regarding the RBF scheme, the convergence rate is on average of order 6 until $N = 20$. For larger values of N, the system matrix becomes ill-conditioned and convergence is lost. This highlights one of the main issues with the direct RBF method. The conditioning of the system matrix gets worse as the number of dof's increases and as the shape parameter ε decreases. The latter controls the flatness of the radial functions and thus the accuracy of the numerical scheme. Ideally one would have to use the RBF-QR method or at least to increase ε as N increases in order to prevent the system matrix from becoming ill-conditioned. These approaches will be considered in a future study.

Since all the schemes result in a full diffusion matrix, the computational cost per dof is similar for the 4 schemes. In this example, the PS scheme is obviously the most efficient but that might not be very representative. However, this example shows how global numerical methods, like the PS and RBF methods, can outperform the FD and FE methods to solve fractional-order integrodifferential equations. While the computational cost per dof is similar for all methods, the former require much less dof's to achieve the same level of accuracy.

5.2 Numerical solution of the space-time fractional diffusion equation

In this section, we present some numerical solutions of Eq. (1) for different values of α and γ and the following initial and boundary conditions

$$f(x,0) = \exp(-10(x-L/2)^2),$$
$$\frac{\partial f}{\partial x}(0,t) = 0,$$
$$f(L,t) = 0,$$

for $L = 10$ and $T = 2$. To account for the change of dimension of the diffusion coefficient $K_{\alpha,\gamma}$ when changing the values of α and γ, we define it as $K_{\alpha,\gamma} = k\mathcal{L}^{\alpha}\mathcal{T}^{-\gamma}$ where $\mathcal{L}$ and $\mathcal{T}$ are characteristic length and time scales, respectively, and k is a dimensionless constant. This amounts to make Eq. (1) dimensionless with respect to those scales and take a dimensionless diffusion coefficient equal to k. For the example presented here, we have selected the following values: $k = \frac{1}{62.5}$, $\mathcal{L} = \frac{L}{8}$, $\mathcal{T} = \frac{T}{8}$.

The time discretization is based on a Mittag-Leffler PS schemes with $M = 7$, *i.e.* the expansion uses 15 dof's in time. However, when $\gamma = 1$, the Mittag-Leffler basis functions reduce to the traditional Fourier basis functions ($\psi_k(t) = \exp(i2\pi kt/T)$). Since the model solution is not periodic in time, such an approximation is not optimal and a 2nd-order Crank-Nicolson FD time discretization is used instead. That scheme is unconditionally stable. Note that when $\gamma = 1$, the time derivative is local and the use of a FD time discretization leads to sparse matrices T and N. The model performances are thus not impaired despite the largest number of dof's. Because of the steepness of the initial solution, we had to use a PS scheme with 191 dof's for the space discretization. For this example, it could have been more advantageous to use a FE or a RBF scheme. However, since the goal here is to provide qualitative results showing the evolution of the solution behavior for different values of α and γ, the choice of the numerical scheme is less important.

Fig. 7 shows the evolution of the solution of Eq. (1) for different values of α and γ. The classical diffusion pattern is recovered when $\alpha = 2$ and $\gamma = 1$. A superdiffusive effect is observed when $\alpha < 2$. In that case, the diffusion operator is non-local and the initially-localized solution is quickly spread over the entire domain. A subdiffusive effect is observed when $\gamma < 1$. In that case, the solution spreads more slowly and eventually almost freezes. Such a behavior highlights the memory effect that is introduced in the model by replacing the first-order time derivative by a fractional-order derivative of degree less than 1. For the general case where $\alpha < 2$ and $\gamma < 1$, the superdiffusive effect in space competes with the subdiffusive effect in time. The resulting behavior of the solution depends on the ratio $\frac{2\gamma}{\alpha}$. For $\frac{2\gamma}{\alpha} > 1$, superdiffusion dominates, while subdiffusion dominates for $\frac{2\gamma}{\alpha} < 1$ (Metzler and Klafter, 2000). Here, $\frac{2\gamma}{\alpha} = \frac{3}{4}$ and the diffusion processes is thus "mostly" subdiffusive as the solution spreads more slowly than predicted by classical models based on the Brownian motion assumption.

6. CONCLUSION

Unlike integer-order derivatives, fractional-order derivatives are non-local operators. As such, they are not well suited to standard numerical methods like the FD and FE methods. These numerical methods are generally of low order and thus require many grid points to obtain an accurate solution. For local differential operators, this

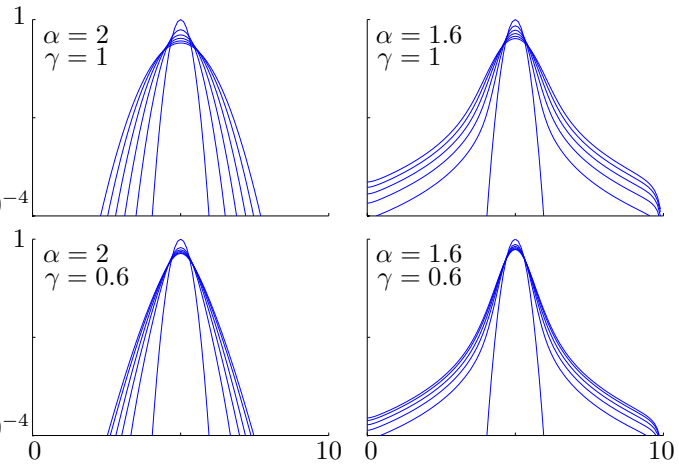

Fig. 7. Diffusion patterns obtained for different values of α and γ at equidistant time instants.

results in large sparse matrices that can usually be handled easily. However, for global differential operators, like fractional-order derivatives, the resulting matrix is full as the global behavior of the function has to be taken into account. The efficiency of the FD or FE methods is thus severely impaired. High-order, global numerical methods like the PS and RBF methods therefore appear to be a better choice as they naturally take the global behavior of the solution into account and use a limited number of degrees of freedom.

Appendix A. KRONECKER PRODUCT

If we consider the matrices $\mathsf{A} \in \mathbb{R}^{m\times n}$ and $\mathsf{B} \in \mathbb{R}^{p\times q}$, then the Kronecker product of A and B is defined as the matrix

$$\mathsf{A}\otimes\mathsf{B} = \begin{bmatrix} a_{11}\mathsf{B} & \dots & a_{1n}\mathsf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathsf{B} & \dots & a_{mn}\mathsf{B} \end{bmatrix} \in \mathbb{R}^{mp\times nq}.$$

The Kronecker product has the useful property that for any 3 matrices A, B and C for which the matrix product is defined, we have:

$$\text{vec}(\mathsf{ABC}) = (\mathsf{C}^T \otimes \mathsf{A})\text{vec}(\mathsf{B}), \qquad \text{(A.1)}$$

where vec(B) is the vector obtained by stacking the columns of B on top of one another (Laub, 2005).

REFERENCES

Boyd, J.P. (2001). *Chebyshev and Fourier Spectral Methods (2nd edition)*. Dover Publications.

Driscoll, T.A. and Fornberg, B. (2002). Interpolation in the limit of increasingly flat radial basis functions. *Computers and Mathematics with Applications*, 43(3-5), 413–422.

Fasshauer, G.E. (2007). *Meshfree Approximation Methods with MATLAB*. Interdisciplinary Mathematical Sciences. World Scientific.

Fix, G.J. and Roop, J.P. (2004). Least square finite-element solution of a fractional order two-point boundary value problem. *Computers and Mathematics with Applications*, 48, 1017–1033.

Ford, N.J. and Simpson, A.C. (2001). The numerical solution of fractional differential equations: Speed versus accuracy. *Numerical Algorithms*, 26(4), 333–346.

Fornberg, B. and Piret, C. (2007). A stable algorithm for flat radial basis functions on a sphere. *SIAM Journal on Scientific Computing*, 30(1), 60–80.

Fornberg, B. and Piret, C. (2008). On choosing a radial basis function and a shape parameter when solving a convective pde on a sphere. *Journal of Computational Physics*, 227(5), 2758–2780.

Fornberg, B. and Zuev, J. (2007). The runge phenomenon and spatially variable shape parameters in rbf interpolation. *Computers and Mathematics with Applications*, 54(3), 379–398.

Hanert, E. (2010). A comparison of three Eulerian numerical methods for fractional-order transport models. *Environmental Fluid Mechanics*, 10, 7–20. Doi:10.1007/s10652-009-9145-4.

Hanert, E. (2011). On the numerical solution of space-time fractional diffusion models. *Computers and Fluids*, 46, 33–39. Doi:10.1016/j.compfluid.2010.08.010.

Laub, A.J. (2005). *Matrix analysis for scientists and engineers*. SIAM.

Li, C. and Deng, W. (2007). Remarks on fractional derivatives. *Applied Mathematics and Computation*, 187, 777–784.

Li, X. and Xu, C. (2009). A space-time spectral method for the time fractional diffusion equation. *SIAM Journal on Numerical Analysis*, 47, 2108–2131.

Lin, Y. and Xu, C. (2007). Finite difference/spectral approximations for the time-fractional diffusion equation. *Journal of Computational Physics*, 225(2), 1533 – 1552.

Mainardi, F. and Gorenflo, R. (2000). On Mittag-Leffler-type functions in fractional evolution processes. *Journal of Computational and Applied Mathematics*, 118, 283–299.

Meerschaert, M.M. and Tadjeran, C. (2004). Finite difference approximations for fractional advection-diffusion flow equations. *Journal of Computational and Applied Mathematics*, 172, 65–77.

Meerschaert, M.M. and Tadjeran, C. (2006). Finite difference approximations for two-sided space-fractional partial differential equations. *Applied Numerical Mathematics*, 56(1), 80–90.

Metzler, R. and Klafter, J. (2000). The random walk's guide to anomalous diffusion: A fractional dynamics approach. *Physics Reports*, 339, 1–77.

Metzler, R. and Klafter, J. (2002). From stretched exponential to inverse power-law: Fractional dynamics, Cole-Cole relaxation processes, and beyond. *Journal of Non-Crystalline Solids*, 305(1-3), 81–87.

Piret, C. and Hanert, E. (2012). Fractional differential operator discretization using the radial basis functions method. *in preparation*.

Podlubny, I. (1999). *Fractional Differential Equations*. Mathematics in Science and Engineering, Volume 198. Academic Press.

Podlubny, I., Chechkin, A., Skovranek, T., Chen, Y., and Jara, B.M.V. (2009). Matrix approach to discrete fractional calculus II: Partial fractional differential equations. *Journal of Computational Physics*, 228, 3137–3153.

Roop, J.P. (2006). Computational aspects of FEM approximation of fractional advection dispersion equations on bounded domains in $\mathbb{R}^2$. *Journal of Computational and Applied Mathematics*, 193, 243–268.

Sousa, E. (2011). Numerical approximations for fractional diffusion equations via splines. *Computers and Mathematics with Applications*, 62(3), 938–944.

Tadjeran, C., Meerschaert, M.M., and Scheffler, H.-P. (2006). A second-order accurate numerical approximation for the fractional diffusion equation. *Journal of Computational Physics*, 213, 205–213.