Limited Memory Block Preconditioners for Fast Solution of Fractional Partial Differential Equations

Daniele Bertaccini & Fabio Durastante

Journal of Scientific Computing

ISSN 0885-7474

J Sci Comput
DOI 10.1007/s10915-018-0729-3

Springer

J Sci Comput
https://doi.org/10.1007/s10915-018-0729-3

Limited Memory Block Preconditioners for Fast Solution of Fractional Partial Differential Equations

Daniele Bertaccini[1,2] · **Fabio Durastante**[3]

Received: 5 October 2017 / Revised: 4 April 2018 / Accepted: 4 May 2018

Abstract An innovative block structured with sparse blocks multi iterative preconditioner for linear multistep formulas used in boundary value form is proposed here to accelerate GMRES, FGMRES and BiCGstab(l). The preconditioner is based on block ω-circulant matrices and a short-memory approximation of the underlying Jacobian matrix of the fractional partial differential equations. Convergence results, numerical tests and comparisons with other techniques confirm the effectiveness of the approach.

Keywords Preconditioners · Fractional calculus · Krylov iterative methods

Mathematics Subject Classification 65F08 · 65M22 · 35R11

1 Introduction and Rationale

In this paper we consider two classes of initial value problems (IVPs) with fractional derivatives *in space*. The latter fractional partial differential equations (FPDE for short in the sequel) are sometimes used to model anomalous dispersion phenomena. In particular, we focus on the *fractional diffusion equation*

The authors are members of the INdAM research group GNCS and this work has been partially supported by the GNCS 2018 Project "Tecniche innovative per problemi di algebra lineare".

✉ Daniele Bertaccini
bertaccini@mat.uniroma2.it

Fabio Durastante
fabio.durastante@di.unipi.it

1 Dipartimento di Matematica, Università di Roma "Tor Vergata", viale della Ricerca Scientifica 1, 00133 Rome, Italy

2 Istituto per le Applicazioni del Calcolo (IAC) "M. Picone", National Research Council (CNR), Rome, Italy

3 Dipartimento di Informatica, Università di Pisa, Largo Bruno Pontecorvo 3, 56127 Pisa, Italy

Published online: 11 May 2018

$$\begin{cases} \frac{\partial}{\partial t} y(x,t) = d_+(x,t)\, {}_{\mathrm{RL}}D^{\alpha}_{x_L,x} y(x,t) + d_-(x,t)\, {}_{\mathrm{RL}}D^{\alpha}_{x,x_R} y(x,t) + g(x,t), \\ x \in (x_L, x_R),\ t \in (t_0, T], \\ y(x_L,t) = y(x_R,t) = 0, \quad 0 \le t \le T, \\ y(x,t_0) = y_0(x), \quad x \in [x_L, x_R], \end{cases} \tag{1}$$

for $\alpha \in (1,2)$, $f(x,t)$ the source (or sink) term and the diffusion coefficients $d_{\pm}(x,t) \ge 0$ with $d_+(x,t) + d_-(x,t) > 0 \,\forall\, x,t \in [x_L, x_R] \times [0,T]$, and more in general on the *fractional advection dispersion equation*

$$\begin{cases} \frac{\partial}{\partial t} y(x,t) = d_+(x,t)\, {}_{\mathrm{RL}}D^{\alpha}_{x_L,x} y(x,t) + d_-(x,t)\, {}_{\mathrm{RL}}D^{\alpha}_{x,x_R} y(x,t) + \\ \qquad + b(x) y_x(x,t) + c(x) y(x,t) + g(x,t), \\ \qquad x \in (x_L, x_R), t \in (t_0, T], \\ y(x_L,t) = y(x_R,t) = 0, \quad 0 \le t \le T, \\ y(x,t_0) = y_0(x), \quad x \in [x_L, x_R], \end{cases} \tag{2}$$

where $b(x) \ge 0 \in \mathcal{C}^1$ and $c(x) \ge 0 \in \mathcal{C}^0$. Similarly, one can take into account the 2D symmetric (Riesz) version of the *fractional diffusion equation*, given by

$$\begin{cases} \frac{\partial u}{\partial t} - K_x \frac{\partial^{2\alpha} u}{\partial |x|^{2\alpha}} - K_y \frac{\partial^{2\beta} u}{\partial |y|^{2\beta}} + \mathbf{b} \cdot \nabla u + cu = g, & (x,y) \in \Omega,\ t \in [0,T] \\ u(x,y,t) = 0, & (x,y) \in \partial\Omega\ t \in [0,T], \\ u(x,y,0) = u_0(x,y), & (x,y) \in \partial\Omega, \end{cases} \tag{3}$$

where $\mathbf{b} \in \mathcal{C}^1(\Omega, \mathbb{R}^2)$, $c \in \mathcal{C}(\Omega)$, $u \in \mathbb{L}^2(\Omega)$, $K_x, K_y \ge 0$ and $K_x + K_y > 0, \alpha, \beta \in (1/2, 1)$. See, e.g., [20,27] and references therein for more details on these fractional partial differential equations.

We use the following definition for Riemann–Liouville fractional derivatives.

Definition 1 Given a function $y(t)$ we define the left-side Riemann–Liouville fractional derivative of order $\alpha > 0$ with $m \in \mathbb{Z}^+$ such that $m - 1 < \alpha \le m$ as

$${}_{\mathrm{RL}}D^{\alpha}_{a,x} y(x) = \frac{1}{\Gamma(m-\alpha)} \left(\frac{d}{dx}\right)^m \int_a^x \frac{y(\xi)\, d\xi}{(x-\xi)^{\alpha-m+1}},$$

where $\Gamma(\cdot)$ is the *Euler gamma function*, and the right-side Riemann–Liouville fractional derivative as

$${}_{\mathrm{RL}}D^{\alpha}_{x,b} y(x) = \frac{1}{\Gamma(m-\alpha)} \left(-\frac{d}{dx}\right)^m \int_x^b \frac{y(\xi)\, d\xi}{(\xi-x)^{\alpha-m+1}}.$$

Given a function $u(x,y)$ and given $1/2 < \mu \le 1$ and $n - 1 < 2\mu \le n$, we define the *symmetric Riesz derivative*

$$\frac{\partial^{2\mu} u(x,y)}{\partial |x|^{2\mu}} = -c_{2\mu} \left({}_{\mathrm{RL}}D^{2\mu}_{a,x} + {}_{\mathrm{RL}}D^{2\mu}_{x,b}\right) u(x,y), \quad c_{2\mu} = \frac{1}{2\cos(\mu\pi)},$$

where

$$
{}_{\mathrm{RL}}D^{2\mu}_{a,x}u(x,y) = \frac{1}{\Gamma(n-2\mu)}\left(\frac{\partial}{\partial x}\right)^n \int_a^x \frac{u(\xi,y)d\xi}{(x-\xi)^{2\mu-n+1}},
$$

$$
{}_{\mathrm{RL}}D^{\mu}_{x,b}u(x,y) = \frac{1}{\Gamma(n-2\mu)}\left(-\frac{\partial}{\partial x}\right)^n \int_x^b \frac{u(\xi,y)d\xi}{(\xi-x)^{2\mu-n+1}}.
$$

The symmetric Riesz derivative with respect to y is defined similarly and is not duplicated here.

To semidiscretize Eqs. (1) and (2), we use the p-shifted Grünwald–Letnikov discretization for the fractional Riemann–Liouville operators [22,23] over the grid $\{x_k = L + k\Delta x\}_{k=0}^m$ and $\Delta x = \frac{R-L}{m}$,

$$
{}_{\mathrm{RL}}D^{\alpha}_{x_L,x}y(x)\big|_{x=x_k} = \frac{1}{\Delta x^\alpha}\sum_{j=0}^{k+p}\omega_j^{(\alpha)}\left[y(x_{k-j+p}) - y(x_L)\right] + \frac{y(x_L)x_k^{-\alpha}}{\Gamma(1-\alpha)} + O(\Delta x^2),
$$

where the coefficients $\omega_j^{(\alpha)}$ are obtained recursively [27] as

$$
\omega_0^{(\alpha)} = 1,\ \omega_j^{(\alpha)} = \left(1 - \frac{\alpha+1}{j}\right)\omega_{j-1}^{(\alpha)},\quad j \geq 1.
$$

The shift parameter p is chosen to optimize the approximation performance as the minimizer of $|\alpha - p/2|$, i.e., $p = 1$ for us because $1 < \alpha \leq 2$. At the same time, the right-sided operator can be obtained as

$$
{}_{\mathrm{RL}}D^{\alpha}_{x,x_R}y(x)\big|_{x=x_k} = \frac{1}{\Delta x^\alpha}\sum_{j=0}^{n-k+p}\omega_j^{(\alpha)}\left[y(x_{k+j-p}) - y(x_R)\right] + \frac{y(x_R)x_k^{-\alpha}}{\Gamma(1-\alpha)} + O(\Delta x^2).
$$

Analogously, we can obtain a discretization of the Eq. (3) by means of the fractional centered discretization in [25]

$$
\left.\frac{\partial^{2\alpha}y(x)}{\partial|x|^{2\alpha}}\right|_{x=x_k} = \frac{\cos(\alpha\pi)}{\Delta x^\alpha}\sum_{j=-k}^{k}\frac{\Gamma(2\alpha+1)(-1)^j}{\Gamma(\alpha-j+1)\Gamma(\alpha+j+1)}y(x_{k-j}) + O(\Delta x^2),
$$

that, as observed in [6], shares an analogous decay property for the coefficients of the p-shifted Grünwald–Letnikov discretization; see, e.g., [17,20] for other feasible finite differences schemes. By means of the above discretization, together with the centered finite difference scheme for $b(x)u_x(x,t)$ and for $\mathbf{b}\cdot\nabla u$, we get a semidiscretization for Eqs. (1), (2) and (3):

$$
\frac{d}{dt}\mathbf{y}(t) = J_m\mathbf{y}(t) + \mathbf{g}(t),\quad t \in (t_0, T],\quad \mathbf{y}(t) = [y^{(1)}(t),\ldots,y^{(m)}(t)]^T. \tag{4}
$$

The initial condition for (4) is

$$
\mathbf{y}(t_0) = [y_0(x_1),\ldots,y_0(x_m)]^T = \mathbf{y}_0,
$$

the Jacobian matrix and forcing term are $J_m \in \mathbb{R}^{m\times m}$, $g(t) \in \mathbb{R}^m$, respectively.

Two properties of J_m are crucial for us:

- the decay in absolute values of the entries along the diagonals [6,28];
- the behavior of its eigenvalues, i.e., the spectral distribution.

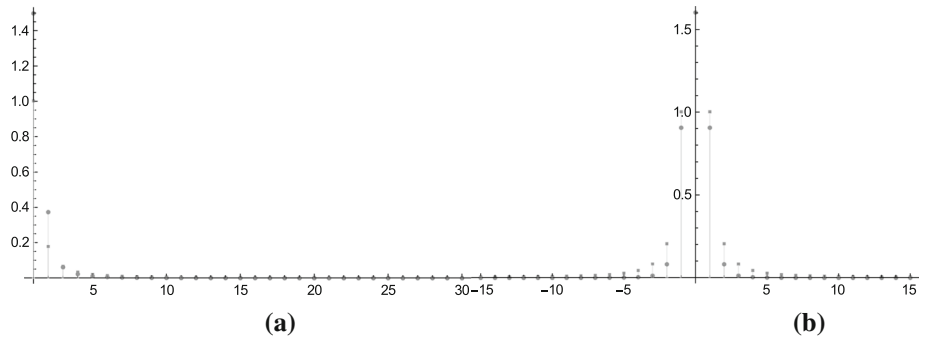

Fig. 1 Bound (5) for both Grünwald–Letnikov (on the left) and fractional centered (on the right) discretizations. The dots represents the values of the coefficients in the discretization, while the squares are the values of the bound. The example is for order of fractional derivative 1.5. **a** Right Grünwald–Letnikov discretization. **b** Fractional centered discretization

The decay of the coefficients represents a structural property of the continuous operators called *short-memory principle* [27], that is inherited by the Grünwald–Letnikov and the fractional centered discretizations: (see Fig. 1)

$$|\omega_j^{(\alpha)}| = O(j^{-\alpha-1}), \quad \text{for } j \to +\infty. \tag{5}$$

The latter property has been recently exploited in several ways. Among them:

- to drop diagonals with entries of small modulus in J_m in order to reduce the computational cost for approximating the matrix exponential or classical time stepping schemes; see, e.g., [28,36,37];
- for the computation and the update of *approximate inverse preconditioners* to solve time-dependent FPDEs [6] or standard incomplete LU factorizations [21].

In Sect. 3 we propose an hybrid preconditioner based on block ω-circulant matrices using the short-memory principle of the Jacobian matrix.

We recall that an $n \times n$ matrix $A_n = (a_{j,k})$ is said to be Toeplitz if $a_{j,k} = \alpha_{j-k}$, $j, k = 1, \ldots, n$, i.e., A_n is constant along its diagonals. An $n \times n$ matrix $\breve{A}_n$ is said to be circulant if it is Toeplitz and its diagonals satisfy $\breve{a}_{n-j} = \breve{a}_{-j}$, $j = 1, \ldots, n-1$. The circulant matrices $\breve{A}_n$ are diagonalized by the Fourier matrix $F = (F_{j,k})$, $F_{j,k} = e^{2\pi \mathbf{i} jk/n}/\sqrt{n}$, $j, k = 0, \ldots, n-1$, $\mathbf{i}$ is the imaginary unit, see, e.g., [7,24] and references therein.

Definition 2 W is an $n \times n$ $\{\omega\}$-circulant matrix if there exists a number ω such that

$$W = \begin{pmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ \omega a_{n-1} & a_0 & \cdots & a_{n-2} \\ \omega a_{n-2} & \omega a_{n-1} & \cdots & a_{n-3} \\ \vdots & \vdots & \cdots & \vdots \\ \omega a_1 & \omega a_2 & \cdots & a_0 \end{pmatrix},$$

Theorem 1 *Let $\omega = \exp(\mathbf{i}\theta)$, $-\pi < \theta \leq \pi$ and W an $n \times n$ $\{\omega\}$-circulant matrix. Then, the following Schur decomposition for W holds true:*

$$W = \Omega^* F^* \Lambda F \Omega, \tag{6}$$

where $\Omega = diag(1, \omega^{-1/n}, \ldots, \omega^{-(n-1)/n})$, Λ is a diagonal matrix containing the eigenvalues of W and F is the Fourier matrix.

Note that circulant matrices are simply the {1}-circulant matrices and thus Theorem 1 gives also their Schur decomposition.

Let us mention a tool that will be useful to determine the distribution of singular values and of the eigenvalues of matrix sequences: the *Generalized Locally Toeplitz* Theory (GLT); see [16].

Proposition 1 [14] *Let us fix a time t_m and assume that the functions $d_+(x) = d_+(x, t_m)$ and $d_-(x) = d_-(x, t_m)$ are both Riemann integrable over $[x_L, x_R]$. Then, the matrix sequence $\{\Delta x^\alpha J_m\}_m$ is a GLT sequence with symbol*

$$\hat{f}(\hat{x}, \theta) = f(x_L + (x_R - x_L)\hat{x}, \theta),$$

where

$$f(x, \theta) = -d_+(x)e^{-\mathbf{i}\theta}(1 - e^{\mathbf{i}\theta})^\alpha - d_-(x)e^{\mathbf{i}\theta}(1 - e^{-\mathbf{i}\theta})^\alpha,$$

$(\hat{x}, \theta) \in [0, 1] \times [-\pi, \pi]$, $(x, \theta) \in [x_L, x_R] \times [-\pi, \pi]$.

Definition 3 Let $f : \mathbb{R}^k \supset D \to \mathbb{C}$ be a measurable function with $k \geq 1$, $\mu(D) < +\infty$. Then, if $\mathcal{C}_0(\mathbb{K})$ is the set of measurable functions with compact support over $\mathbb{K}$, with $\mathbb{K} = \mathbb{C}$ or $\mathbb{R}_0^+$, given a sequence of matrices $\{J_m\}_m$ with eigenvalues $\{\lambda_j\}$ and singular values $\{\sigma_j\}$, then

$J_m \sim_\lambda (f, D)$: $\{J_m\}_m$ is distributed in the sense of the eigenvalues as the pair (f, D) if

$$\lim_{m\to+\infty} \frac{1}{m} \sum_{j=1}^{m} F(\lambda_j) = \frac{1}{\mu(D)} \int_D F(f(t))\, dt, \quad \forall\, F \in \mathcal{C}_0(\mathbb{C}),$$

$J_m \sim_\sigma (f, D)$: $\{J_m\}_m$ is distributed in the sense of the singular values as the pair $f(f, D)$ if

$$\lim_{m\to+\infty} \frac{1}{m} \sum_{j=1}^{m} F(\sigma_j) = \frac{1}{\mu(D)} \int_D F(|f(t)|)\, dt, \quad \forall\, F \in \mathcal{C}_0(\mathbb{R}_0^+),$$

In particular, $\{\Delta x^\alpha J_m\}_m$ is a GLT sequence for f, and thus $J_m \sim_\sigma f$, with f from $[0, 1] \times [-\pi, \pi]$. Moreover, if J_m is Hermitian, this holds also in the sense of the eigenvalues; see Fig. 2. This gives that the eigenvalues of any J_m have negative real part, i.e., $\Re(\lambda_i) < 0$.

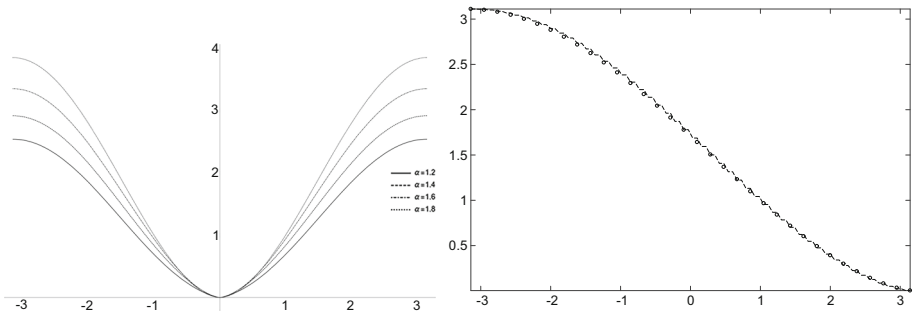

Fig. 2 Proposition 1. Some GLT symbols for different values of α for the Jacobian matrix with coefficients $d_+(x) \equiv 0.5$ and $d_-(x) \equiv 0.6$ and computed singular values (dot) and approximated by the sampling of the GLT symbol (dashed gray line). **a** GLT symbols. **b** Singular values distribution

Note that there is a zero of order α in 0 for the GLT symbol of J_m, see [14, Proposition 6]. Thus, *any* circulant preconditioner that produces a clustering at the unity in the case of constant d_+ and d_- coefficients, is not anymore effective in the general, Hermitian or not, variable coefficient case.

The outline of the paper is the following. In Sect. 2 some notions on linear multistep formulas used in boundary value form that we will use for our FPDEs are briefly recalled. Section 3 includes our proposals to precondition the large, sparse and structured linear systems generated by the discretization of the underlying time-dependent FPDEs along with some convergence and spectral results. Finally, in Sect. 4 some numerical experiments and comparisons with some of the most recent solution strategies are proposed.

2 Linear Multistep Formulas Used in Boundary Value Form

After discretization with respect to space variables, FPDEs (1), (2) and (3), but also many time-dependent PDEs, can be reduced to the solution of the IVP

$$\begin{cases} \mathbf{y}'(t) = J_m \mathbf{y}(t) + \mathbf{g}(t), \ t \in (t_0, T], \\ \mathbf{y}(t_0) = \mathbf{y}_0, \end{cases} \tag{7}$$

where $\mathbf{y}(t), \mathbf{g}(t) : \mathbb{R} \to \mathbb{R}^m$, $\mathbf{y_0} \in \mathbb{R}^m$, and $J_m \in \mathbb{R}^{m \times m}$. We apply to (7) fully implicit methods for differential equations based on linear multistep formulas (*LMF* for short) in boundary value form, see [1,12] and references therein. These methods approximate the solution of the IVP (7) by means of a discrete boundary value problem. Consider the following k-step linear multistep formula over a uniform mesh $t_j = t_0 + jh$, for $j = 0, \ldots, s$, $h = (T - t_0)/s$ to (7):

$$\sum_{i=-\nu}^{k-\nu} \alpha_{i+\nu} \mathbf{y}_{n+i} = h \sum_{i=-\nu}^{k-\nu} \beta_{i+\nu} \mathbf{f}_{n+i}, \quad n = \nu, \ldots, s - k + \nu. \tag{8}$$

Here, $\mathbf{y}_n$ is the discrete approximation to $\mathbf{y}(t_n)$, $\mathbf{f}_n = J_m \mathbf{y}_n + \mathbf{g}_n$ and $\mathbf{g}_n = \mathbf{g}(t_n)$. The method in (8) should be used with ν initial conditions and $k - \nu$ final conditions. That is, we need the values $\mathbf{y}_0, \ldots, \mathbf{y}_{\nu-1}$ and the values $\mathbf{y}_{s-k+\nu+1}, \ldots, \mathbf{y}_s$. An initial value problem like (7) provides only one value, i.e., $\mathbf{y}_0$. In order to get the other initial and final values, we have to provide additional $(k-1)$ equations. The coefficients $\alpha_i^{(j)}$ and $\beta_i^{(j)}$ of these equations can be chosen such that the truncation errors for these initial and final conditions are of the same order as that in (8), see [12] for details. Note that for a boundary value problem we can have other values than $\mathbf{y}_0$. We stress that all the methods considered are consistent, i.e., their characteristic polynomials

$$\rho(z) = z^\nu \sum_{j=-\nu}^{k-\nu} \alpha_{j+\nu} z^j, \qquad \sigma(z) = z^\nu \sum_{j=-\nu}^{k-\nu} \beta_{j+\nu} z^j,$$

are such that

$$\rho(1) = 0, \quad \rho'(1) = \sigma(1).$$

Combining (8) with the above mentioned additional methods gives a discrete boundary value problem, also called *BVM* in [12]. These equations can be restated to give the following linear system of algebraic equations:

$$M\mathbf{y} \equiv (A \otimes I_m - hB \otimes J_m)\mathbf{y} = \mathbf{e}_1 \otimes \mathbf{y}_0 + h(B \otimes I_m)\mathbf{g} \equiv \mathbf{b}, \tag{9}$$

where

$$\mathbf{e}_1 = [1, 0, \ldots, 0]^T \in \mathbb{R}^{s+1},\ \mathbf{y} = [\mathbf{y}_0^T, \ldots, \mathbf{y}_s^T]^T \in \mathbb{R}^{(s+1)m},$$
$$\mathbf{g} = [\mathbf{g}_0^T, \ldots, \mathbf{g}_s^T]^T \in \mathbb{R}^{(s+1)m},\ A, B \in \mathbb{R}^{(s+1)\times(s+1)}.$$

The matrices A and B are obtained from the coefficients of the formula (8) and the auxiliary linear multistep formulas:

$$A = \begin{bmatrix}
1 & \ldots & 0 & & & \\
\alpha_0^{(1)} & \ldots & \nu\alpha_k^{(1)}, & & & \\
\vdots & & \vdots & & & \\
\alpha_0^{(\nu-1)} & \ldots & \alpha_k^{(\nu-1)} & & & \\
\alpha_0 & \ldots & \alpha_k & & & \\
 & \alpha_0 & \ldots & \alpha_k & & \\
 & & \ddots & \ddots & \ddots & \\
 & & & \alpha_0 & \ldots & \alpha_k \\
 & & & \alpha_0^{(s-k+\nu+1)} & \ldots & \alpha_k^{(s-k+\nu+1)}, \\
 & & & \vdots & & \vdots \\
 & & & \alpha_0^{(s)} & \ldots & \alpha_k^{(s)}
\end{bmatrix},$$

and

$$B = \begin{bmatrix}
0 & \ldots & 0 & & & \\
\beta_0^{(1)} & \ldots & \beta_k^{(1)}, & & & \\
\vdots & & \vdots & & & \\
\beta_0^{(\nu-1)} & \ldots & \beta_k^{(\nu-1)} & & & \\
\beta_0 & \ldots & \beta_k & & & \\
 & \beta_0 & \ldots & \beta_k & & \\
 & & \ddots & \ddots & \ddots & \\
 & & & \beta_0 & \ldots & \beta_k \\
 & & & \beta_0^{(s-k+\nu+1)} & \ldots & \beta_k^{(s-k+\nu+1)}, \\
 & & & \vdots & & \vdots \\
 & & & \beta_0^{(s)} & \ldots & \beta_k^{(s)}
\end{bmatrix}.$$

We recall that auxiliary methods cannot have the same coefficients of (8). More details on the matrices A and B, M and their entries can be found in [12]. Properties and information on their eigenvalues can be found in [3] and in [5].

The size of the matrix M can be very large when s or m are large. If a direct method is used to solve the system (9), e.g., for a multidimensional FPDE, the operation count can be high, see also the comparisons on sparse direct/iterative methods for a PDEs in [3]. Therefore, we concentrate on Kyrlov iterative solvers. Note that in general it is not necessary to assemble explicitly the matrix M from Eq. (9), since to apply Krylov iterative solvers we need only to form the matrix vector product $M\mathbf{y}$. Thus, by the properties of Kronecker products, we get:

$$\mathbf{x} = M\mathbf{y} = \mathrm{vec}(I_m Y A^T - h J_m Y B^T) = \mathrm{vec}(Y A^T - h J_m Y B^T),$$

where the operator $\mathrm{vec}(\cdot)$ stacks the columns of a matrix and Y is obtained by simply reshaping $\mathbf{y}$ as an $m \times s$ matrix.

Differently to PDEs discretized by finite differences or using finite elements, in case of FPDEs, also Krylov iterative solvers with the block circulant preconditioners introduced in [3] can be not so effective since J_m is a dense matrix; see Remark 1. The same conclusion can be achieved for all (block or not) preconditioners for the linear systems of other time-step integrators based, e.g., on linear multistep formulas or on Runge–Kutta methods; see [3].

By Proposition 1, the discretizations considered here for fractional differential equations produce Jacobian matrices J_m whose eigenvalues have non positive real parts. Therefore, it is natural to use Generalized BDFs, or GBDFs for short, the generalization of BDF formulas proposed in [12] that are $A_{\nu,k-\nu}$-stable and $L_{\nu,k-\nu}$-stable for all $k \geq 1$, a sort of A-stability and L-stability in a generalized sense, instead of the generalizations of the Adams-Moulton formulas used in [18], that are not.

The GBDF formula for a problem of the form (7) with k steps can be obtained starting from the expression of the classical BDF formulas

$$\sum_{i=0}^{k} \alpha_i \mathbf{y}_{n+i} = h\beta_k \mathbf{f}_{n+k} \tag{10}$$

with the same notation of (8). Note that the second stability polynomial is $\sigma(z) = \beta_k z^k$ for (10). It is well known that the BDF formulas from order 7 on are 0-unstable and for any $k > 2$ they are also not A-stable. On the other hand, if we use the underlying generalization of linear multistep formulas, we can build methods of both maximal order k and potentially better stability properties. In particular, with a second stability polynomial $\sigma(z) = \beta_j z^j$ with $j = \nu$ chosen as

$$\nu = \begin{cases} k+1/2, & k \text{ odd}, \\ k/2 + 1, & k \text{ even}, \end{cases}$$

instead of $j = k$, we obtain formulas that are both $0_{\nu,k-\nu}$-stable and $A_{\nu,k-\nu}$-stable for all $k \geq 1$, i.e., the stability regions of these methods are outside the curve

$$\Gamma_k = \{q \in \mathbb{C} : |\pi(z,q)| \equiv |\rho(z) - qz^{\nu}| = 1, \ \forall z \in \mathbb{C}\}.$$

Thus, by normalizing the coefficients, a GBDF with k steps, ν initial and $k-\nu$ final conditions can be written as

$$\sum_{i=-\nu}^{k-\nu} \alpha_{i+\nu} \mathbf{y}_{i+\nu} = h\mathbf{f}_n, \quad n = \nu, \ldots, s-k+\nu, \tag{11}$$

that is clearly an instance of the general formula in (8); see again [12] for the full derivation. We stress that also $L_{\nu,k-\nu}$-stability matters in this case. Indeed, in the occurrence of rapid decaying transients in the solution, supposing we are not interested in resolving them accurately, the use of L-stable (and thus $L_{\nu,k-\nu}$-stable) methods permit to use sensibly larger time steps without compromising the qualitative behavior of the approximation.

Here we use low order linear multistep formulas (maximum order 3) because the discretization of the fractional differential operator shows a order at most linear in our equispaced mesh. Note also that a higher order formula requires a higher computational effort to solve the related linear systems; see next sections. Moreover, a higher order formula (in both time and space) requires a higher regularity of the solution to express its potentialities, that is not guaranteed to hold for a fractional equation, even when the coefficients of the underlying FPDE are arbitrarily regular.

Theorem 2 [12] *In exact arithmetic, a linear multistep formula in boundary value form with* $(\nu, k-\nu)$*-boundary conditions is convergent if it is consistent and* $0_{\nu,k-\nu}$*-stable.*

From the above arguments, we can state our main convergence result.

Proposition 2 *The GBDF formula* (11) *with* $k = 2$ *applied to Problem* (1) *discretized by the* 1*-shifted Grünwald–Letnikov formulas is convergent whenever* $y \in \mathcal{C}^{\alpha+1}$.

Proof We wish to apply Theorem 2, thus we only need to prove that the resulting method is consistent, since, as we have seen, GBDF formulas are $0_{\nu,k-\nu}$-stable. Let $u(x,t)$ be the true solution of (1). Then, the local truncation error $\tau(x,t)$ is consistent of order two in space and one in time by the same arguments used in [22, Theorem 2.7]. Similar arguments can be used in several spatial dimensions. □

In our opinion, using a discretization in time of order five as in [18] is unnecessarily expensive because the global accuracy cannot increase in general. Indeed, the low order of the discretization in space dominates the global error. We stress also that several other methods for integrating system (7) are available in the literature, consider, e.g., Contour Integral Methods [26], Exponential Quadrature Rules [38], and the classical LMFs [20]; nevertheless, we used the proposed BVM schemes since they are suitable for a possible parallel implementation: we need to solve *only once* the linear system (9) to have an evaluation of the solution of (7) at all the time steps simultaneously.

3 Structured Preconditioners

To solve linear systems (9), let us focus on the application of the following iterative Krylov methods: BiCGSTAB(2) [35], GMRES(20) [31] and FGMRES [29], coupled with block preconditioners that take into account their block structure. In the style of [3,4,9–11], we propose here a preconditioner of the form

$$P = \breve{A} \otimes I - h\breve{B} \otimes \tilde{J}_m,$$

where $\breve{A}$ and $\breve{B}$ are circulant-like approximations of the Toeplitz matrices A and B, respectively, containing the coefficients of the LMF formulas (8) and of the additional LMFs, while $\tilde{J}_m$ is a suitable approximation of the Jacobian matrix detailed below.

By properties of the Kronecker product, the eigenvalues of the preconditioner P are given by

$$\phi_i - h\psi_i\lambda_j, \qquad i = 1,\dots,s, \quad j = 1,\dots,m,$$

where $\{\phi_i\}$ and $\{\psi_i\}$ are the eigenvalues of the circulant-like approximations $\breve{A}$ and $\breve{B}$, respectively, and $\{\lambda_j\}$ are the eigenvalues of the selected approximation of J_m.

In [18] the authors proposed the following block-preconditioner based on the Strang circulant approximation (see [24] for details) for the FPDEs semidiscretized in space with p-shifted Grünwald–Letnikov:

$$P_{\mathfrak{s}} = \mathfrak{s}(A) \otimes I_m - h\mathfrak{s}(B) \otimes J_m, \tag{12}$$

where

$$\mathfrak{s}(A) = \begin{bmatrix} \alpha_\nu & \cdots & \alpha_k & & & & \alpha_0 & \cdots & \alpha_{\nu-1} \\ \vdots & \ddots & & \ddots & & & & \ddots & \vdots \\ \alpha_0 & & \ddots & & \ddots & & & & \alpha_0 \\ & \ddots & & \ddots & & \ddots & & 0 & \\ & & \ddots & & \ddots & & \ddots & & \\ & 0 & & \ddots & & \ddots & & \ddots & \\ \alpha_k & & & & \ddots & & \ddots & & \alpha_k \\ \vdots & \ddots & & & & \ddots & & \ddots & \vdots \\ \alpha_{\nu+1} & \cdots & \alpha_k & & & & \alpha_0 & \cdots & \alpha_\nu \end{bmatrix},$$

and $\mathfrak{s}(B)$ is defined similarly. The preconditioner (12) for a generic LMF in boundary value form was first introduced in 1998 in [2] and in [3] using also other circulant approximations, and later studied also in [13]. In particular, (12) was introduced for LMF in boundary value form to solve a generic differential problem and thus also for an initial value problem generated by semidiscretization in space of the underlying FPDE problem. In this framework, the following preconditioner, based on the modified Strang circulant introduced in [4], can be a better approach:

$$P_{\tilde{\mathfrak{s}}} = \tilde{\mathfrak{s}}(A) \otimes I_m - h\tilde{\mathfrak{s}}(B) \otimes J_m. \tag{13}$$

The above, discussed in [4], is able to recover problems of severe ill-conditioning or also singularity of the block preconditioners based on Strang circulant approximation of a LMF. In particular $\tilde{\mathfrak{s}}(\cdot)$ is obtained simply as a rank-one correction of the natural Strang preconditioner $\mathfrak{s}(\cdot)$, i.e., $\tilde{\mathfrak{s}}(A) = \mathfrak{s}(A) + E$ where E is a rank-one circulant matrix given by

$$E = F^* \mathrm{diag}(\hat{\phi}_0 - \phi_0, 0, \ldots, 0) F,$$

with $\hat{\phi}_0$ that, as suggested in [4], can be $\hat{\phi}_0 = 1/s+1$ or $\hat{\phi}_0 = \Re(\phi_s)$; see [2,4,5] for further details. Surprisingly, none of the above mentioned researches on block circulant preconditioners for LMF in boundary value form have been mentioned in Gu et al. [18].

Differently from PDEs, for FPDEs J_m can be a dense matrix. So, in order to reduce the computational complexity, the following two block-circulant with circulant blocks versions

$$P'_{\mathfrak{s}} = \mathfrak{s}(A) \otimes I_m - h\mathfrak{s}(B) \otimes \mathfrak{s}(J_m), \tag{14}$$

and

$$P'_{\tilde{\mathfrak{s}}} = \tilde{\mathfrak{s}}(A) \otimes I_m - h\tilde{\mathfrak{s}}(B) \otimes \tilde{\mathfrak{s}}(J_m), \tag{15}$$

based on the application of the same circulant preconditioner to the Jacobian matrix were also considered in [18].

The eigenvalues of the circulant approximation $\mathfrak{s}(\cdot)$ and $\tilde{\mathfrak{s}}(\cdot)$ can be read on the main diagonal of the matrix Λ in Theorem 1 with $\omega = 1$ and thus $\Omega = I$.

Remark 1 We do not recommend the choice of the Strang circulant approximation for the Jacobian matrix J_m or for A in (9). As clearly remarked in [2] and in [4], the Strang's approximation for the matrix A in Eq. (9) is singular for every number of step $k \geq 1$, independently from the value of s, by the consistency requirements of the linear multistep formulas. Moreover, the analysis of the spectral distribution of $\{\Delta x^\alpha J_m\}_m$ matrices in Sect. 1, does not recommend the use of the Strang preconditioner for the Jacobian matrix J_m as well.

Here we focus on other preconditioners that do not need the above mentioned patch. In particular, we consider the ω-circulant approximation $\omega(\cdot)$ introduced for LMF in boundary value form to integrate PDEs in [8,11]:

$$P_\omega = \omega(A) \otimes I_m - h\omega(B) \otimes J_m, \tag{16}$$

where

$$\omega(A) = \begin{bmatrix} \alpha_\nu & \cdots & \alpha_k & & & \omega\alpha_0 & \cdots & \omega\alpha_{\nu-1} \\ \vdots & \ddots & & \ddots & & & \ddots & \vdots \\ \alpha_0 & & \ddots & & \ddots & & & \omega\alpha_0 \\ & \ddots & & \ddots & & \ddots & 0 & \\ & & \ddots & & \ddots & & \ddots & \\ & 0 & & \ddots & & \ddots & & \\ \omega\alpha_k & & & \ddots & & \ddots & & \alpha_k \\ \vdots & \ddots & & & \ddots & & \ddots & \vdots \\ \omega\alpha_{\nu+1} & \cdots & \omega\alpha_k & & & \alpha_0 & \cdots & \alpha_\nu \end{bmatrix},$$

and $\omega(B)$ is defined similarly.

Since for FPDEs J_m is a dense matrix, in order to reduce the computational cost of matrix-vector multiplications using preconditioner (16), we firstly propose to use $\omega(J_m)$ instead of J_m in (16), i.e., an ω-circulant approximation also for J_m:

$$P'_\omega = \omega(A) \otimes I_m - h\omega(B) \otimes \omega(J_m).$$

Our second proposal is based on exploiting the short-memory principle. This means using a banded approximation of the Jacobian matrix J_m instead of a circulant or an ω-circulant one (for J_m). We apply the short-memory principle by the function $g_k(J_m)$ that extracts the k lower and upper main diagonals of J_m producing the following *limited memory block ω-circulant preconditioner*:

$$P_{\omega,k} = \omega(A) \otimes I_m - h\omega(B) \otimes g_k(J_m). \tag{17}$$

To further reduce the computational effort needed to apply a $P_{\omega,k}$-circulant preconditioner at each iteration, we can consider, instead of a direct method for sparse systems, the use of nested iterative methods, e.g., the GMRES(m) method. To apply preconditioner $P_{\omega,k}$ from (17) we observe that by Theorem 1 we have

$$P_{\omega,k} = (\Omega^* F^* \otimes I_m)(\Lambda_A \otimes I_m)(F\Omega \otimes I_m) - (\Omega^* F^* \otimes I_m)(h\Lambda_B \otimes g_k(J_m))(F\Omega \otimes I_m)$$

and thus

$$P_{\omega,k}^{-1} = (F\Omega \otimes I_m)^{-1}(\Lambda_A \otimes I_m - h\Lambda_B \otimes g_k(J_m))^{-1}(\Omega^* F^* \otimes I_m)^{-1}.$$

Then, the application $\mathbf{z} = P_{\omega,k}^{-1}\mathbf{v}$ is computed with the following three steps:

1. Use the Fast Fourier Transform (or FFT for short) and solve a diagonal system to compute $\mathbf{v}_1 = (\Omega^* F^* \otimes I_m)^{-1}\mathbf{v}$,
2. Solve $(\Lambda_A \otimes I_m - h\Lambda_B \otimes g_k(J_m))\mathbf{v}_2 = \mathbf{v}_1$,
3. Use the Inverse Fast Fourier Transform (or IFFT for short) and solve a diagonal system to compute $\mathbf{v} = (F\Omega \otimes I_m)^{-1}\mathbf{v}_2$.

In particular, to solve the block diagonal linear system at Step 2, we solve the s auxiliary linear systems of the form

$$T_{j,k} \triangleq \phi_j I - h\psi_j\, g_k(J_m), \quad j = 1, \ldots, s, \quad \Lambda_A = \mathrm{diag}(\phi_j),\ \Lambda_B = \mathrm{diag}(\psi_j),$$

required to apply all block circulant or block $P_{\omega,k}$-circulant preconditioners described above; see, e.g., [3,11] for technical details. In this way we are moving into the framework of preconditioners changing during the iterations and then we need to use *Flexible GMRES* method or its restarted version, see Saad [29,30]. In some cases, to ensure a fast convergence of the outer method (FGMRES), we need to use a preconditioner for the inner (GMRES(m)). To this end, we propose the use of an approximate inverse Toeplitz preconditioner for $T_{j,k}$ based on the ω-circulant preconditioner from [15,19]. Thus, we consider the ω-circulant extension $W_{j,n+k}$ of $T_{j,k}$, obtained as

$$W_{j,n+k} = \begin{bmatrix} \tilde{T}_{j,k} & T_{2,1}^* \\ T_{2,1} & T_{2,2} \end{bmatrix}, \quad T_{2,1} = \begin{bmatrix} \omega t_k & & & 0 & \ldots & 0 & \overline{t_k} & \ldots & \overline{t}_1 \\ \vdots & \ddots & & \vdots & \ddots & \vdots & & \ddots & \vdots \\ \omega t_1 & \ldots & \omega t_k & 0 & \ldots & 0 & & & \overline{t_k} \end{bmatrix},$$

where $\tilde{T}_{j,k}$ is the Toeplitz matrix obtained with the first column and row of $T_{j,k}$ and $\omega = \exp(i\theta)$ with $\theta \in [-\pi, \pi]$. In this way, the diagonal matrix $\Lambda_{j,n+k}$ containing the eigenvalues of $W_{j,n+k}$ is given by

$$\Lambda_{j,n+k} = F_{n+k}\Omega_{n+k} W_{j,n+k} \Omega_{n+k}^* F_{n+k}^*.$$

Once the eigenvalues have been computed, the inverse of the ω-circulant matrix is

$$W_{j,n+k}^{-1} = \begin{bmatrix} P & P_{1,2} \\ P_{1,2} & P_{2,2} \end{bmatrix} = \Omega_{n+k}^* F_{n+k}^* \Lambda_{j,n+k}^{-1} F_{n+k} \Omega_{n+k}. \tag{18}$$

To avoid breakdowns, we can take care of the non-positive entries of $\Lambda_{j,n+k}$ by setting them to a positive value δ in the corresponding positions of $\Lambda_{j,n+k}^{-1}$ also warning the user with a message. But this never happened in our experiments. Then, the preconditioner used for the inner (GMRES(m)) is the matrix P in (18).

We also tried various recycling Krylov subspace approaches to solve the s auxiliary linear systems at Step 2 but the performances were not satisfactory and we do not report the underlying experiments here.

Lemma 1 *Let us consider the approximation $g_k(J_m)$ for J_m. Then, for $\varepsilon > 0$ and $m > 0$ integer, there exists a bandwidth parameter $\tilde{k} = \tilde{k}(\varepsilon, m, \alpha) > 0$ such that $g_k(J_m)^{-1} J_m = I + N$ with $\|N\| \le \varepsilon\ \forall k \ge \tilde{k}$.*

Proof Let us fix $\varepsilon > 0$ and assume $y(x)$ such that $y(x) \le M$ for $x \in \Omega = [x_L, x_R]$. Then, for each $L \in \Omega$, we can write the error as (see [27, Chapter 7.3]),

$$E(x) = |\,{}_{\mathrm{RL}}D_{a,x}^{\alpha} y(x) - {}_{\mathrm{RL}}D_{x-L,x}^{\alpha} y(x)| \le \frac{ML^{-\alpha}}{|\Gamma(1-\alpha)|}.$$

We can find the required values of L by solving

$$|E(x)| \le \varepsilon,\ (x_L + L \le x \le x_R), \quad \Rightarrow \quad L \ge \left(\frac{M}{\varepsilon|\Gamma(1-\alpha)|}\right)^{1/\alpha}.$$

Therefore, by using the same arguments for the fractional derivative of the other side, fixed a discretization step and a value of $\varepsilon > 0$, we can choose a bandwidth k giving the wanted

residual and such that its norm is less the ε. Otherwise, we can look at it from the spectral point of view. From Proposition 1 we get $J_m \sim_{\text{GLT}} f$, where the domain of f is given by $[0, 1] \times [-\pi, \pi]$. Thus, we can consider, at the same way, the spectral distribution $f_{\tilde{k}}$ of $g_{\tilde{k}}(J_m)$. This is obtained by replacing $e^{-\mathbf{i}\theta}(1 - e^{\mathbf{i}\theta})^{\alpha}$ and $e^{\mathbf{i}\theta}(1 - e^{\mathbf{i}\theta})^{-\alpha}$ in f with the first $\tilde{k}$ terms of their real binomial expansion. Therefore, $g_{\tilde{k}}(J_m)^{-1} J_m \sim_{\text{GLT}} f/f_{\tilde{k}}$, and this can be expressed as

$$\frac{f}{f_{\tilde{k}}} = 1 + n_{\tilde{k}},$$

where $n_{\tilde{k}}$ is again the f function in which we have replaced $e^{-\mathbf{i}\theta}(1-e^{\mathbf{i}\theta})^{\alpha}$ and $e^{\mathbf{i}\theta}(1-e^{\mathbf{i}\theta})^{-\alpha}$ with the first $m - \tilde{k}$ term of their real binomial expansion. We conclude by referring to the decay property in (5) and recalling that the coefficients of $n_{\tilde{k}}$ are exactly the $\omega_j^{(\alpha)}$ for $j > \tilde{k}$, thus finding the minimum integer $\tilde{k}$ such that the bound $\|n_{\tilde{k}}\| < \varepsilon$ holds. Therefore, we immediately get also $\|n_k\| < \varepsilon \ \forall k \geq \tilde{k}$. □

Remark 2 Observe that Lemma 1 is quite independent from the discretization adopted, since the tools used to prove it are based mainly on a structural property of the fractional operators, namely the *short-memory principle*, that, as extensively discussed in [6], is inherited by various discretizations of the underlying operators. Therefore, with a little additional effort, also the spectral part of the proof can be extended to other discretizations. A depiction of the results of Lemma 1 is given in Fig. 3. We stress again that this property is lost if we use *any* circulant approximation for the variable coefficient case in both Problems (1) and (2). See the discussion at the end of Sect. 1.

Theorem 3 *Let us consider the limited memory block ω-circulant preconditioner (17) such that $\omega = \exp(i\omega\theta)$, $\theta = \pi$ and $k \geq \tilde{k}$, $\tilde{k}$ as in Lemma 1. Then, the eigenvalues of the preconditioned matrix $P_{\omega,k}^{-1} M$ are equal to $1 \in \mathbb{C}$ except for at most $2mk$ outliers.*

Proof Follows by applying Lemma 1 and Theorem 4 in [11]. □

As a direct consequence, we have the convergence result.

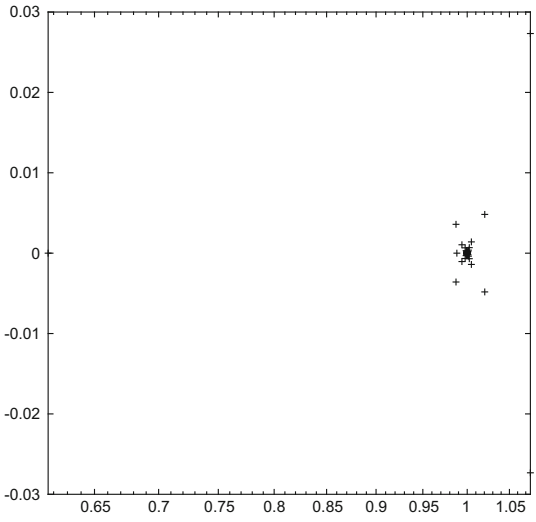

Fig. 3 Lemma 1. Clustering of the eigenvalues for the preconditioner $g_k(J_m)^{-1} J_m$ for J_m as in Problem (1) and $k = \lceil m/5 \rceil$, $m = 2^{10}$

Corollary 1 *If the matrix $P_{\omega,k}^{-1} M$ is diagonalizable, GMRES converges in at most $2mk + 1$ iterations, independently of s, where k is the number of steps of the LMF formula.*

We stress that the above result shows a number of iteration proportional to m. However, in practice, we experience convergence of iterations for GMRES, GMRES(r), $r > 1$, and BiCGstab, preconditioned by the *limited memory block ω-circulant preconditioner*, much less dependent on the mesh than Corollary 1 suggests; see tables in Sect. 4. On the other hand, if we choose, e.g., $k = \lceil m/5 \rceil$ for $P_{\omega,k}$ in (17), then the underlying Krylov iterative solvers converge in a number of iterations more or less constant with the mesh parameters. Unfortunately, by taking $k = \lceil m/5 \rceil$, the iterations are kept (almost) constant but the computational cost increases with m, suggesting that a choice of a constant k, can be a good (but of course somewhat problem-dependent) compromise.

Convergence results similar to Theorem 3 can be derived for other values of θ different from π. However, as observed in [11, Section 2.2] and confirmed by our numerical experiments, the $\{\omega\}$-circulant block preconditioners which give slightly "best" results are those with $\omega = -1$, i.e., $\theta = \pi$, and then are based on skew-circulant matrices.

4 Numerical Experiments

We summarize in Table 1 the preconditioning strategies tested in our experiments.

The results have been obtained on a laptop running Linux with 8 Gb memory and CPU Intel® Core™ i7-4710HQ CPU with clock 2.50 GHz and MATLAB version R2016b.

We use our implementation of FGMRES, based on the algorithms and suggestions in [7,29]. GMRES(20) and BiCGstab are provided by Matlab. BiCGstab(2) is implemented similarly to [7,35]. We report the number of matrix–vector operations performed by the solvers in the tables. Moreover, the main stopping criterium require the relative residuals less than $\varepsilon = 10^{-8}$. Here all the $\{\omega\}$-circulant approximations have $\omega = -1$, i.e., $\theta = \pi$, and then are based on skew-circulant matrices. Motivations for this choice are detailed at the end of the previous section and in [11, Section 2.2].

Experiment 1 As a first test case, let us consider the fractional diffusion Eq. (1) with coefficients

Table 1 Preconditioners tested in the numerical experiments, details in Sect. 3

	Preconditioner for:			Computational cost
	A	B	J_m	
I	None	None	None	–
$P_{\mathfrak{s}}$	Strang	Strang	None	$O(ms\log(s) + sm^2)$
$P_{\tilde{\mathfrak{s}}}$	Modified Strang	Modified Strang	None	$O(ms\log(s) + sm^2)$
$P'_{\mathfrak{s}}$	Strang	Strang	Strang	$O(ms\log(ms))$
P_{ω}	ω-Circulant	ω-Circulant	None	$O(ms\log(s) + sm^2)$
P'_{ω}	ω-Circulant	ω-Circulant	ω-Circulant	$O(ms\log(ms))$
$P_{\omega,k}$	ω-Circulant	ω-Circulant	$g_k(J_m)$	$O(ms\log(s) + sk^2m)$
$P_{\omega,k}^{\text{FGMRES}}$	ω-Circulant	ω-Circulant	$g_k(J_m)$ GMRES	$O(ms\log(s) + s(2k-1)m)$

$$
\begin{aligned}
&x_L = 0,\ x_R = 2,\ t_0 = 0,\ T = 1,\\
g(x,t) = -32e^{-t}&\left\{x^2 + \frac{1}{8}(2-x)^2(8+x^2) - \frac{3}{3-\alpha}\left[x^3 + (2-x)^3\right] + \ldots\right.\\
&\left.\ldots + \frac{3}{(4-\alpha)(3-\alpha)}\left[x^4 + (2-x)^4\right]\right\}\\
&d_+(x,t) = \Gamma(3-\alpha)x^\alpha,\ d_-(x,t) = \Gamma(3-\alpha)(2-x)^\alpha,\\
&u_0(x) = 4x^2(2-x)^2,
\end{aligned}
\tag{19}
$$

where the order of the fractional derivatives is $\alpha = 1.5$ and $\alpha = 1.8$, respectively. For this choice of the coefficients we have the exact solution of the FPDEs that is $u_e(x,t) = 4e^{-t^2}x^2(2-x)^2$ for any value of $\alpha \in (1,2)$. In Table 2 we show the results obtained with various preconditioning strategies. For this case, we use the GBFD formula with two step, that gives a more reasonable behavior of the error when mixed with the first order approximation used for the discretization in space, with GMRES(20) and FGMRES iterative methods. Moreover, in Fig. 4, we give both the spectrum of the unpreconditioned matrix M and of the preconditioned matrix M for all the proposed preconditioners. Consistently with the results in Table 2 and the analysis in Sect. 3, the preconditioner based on the short-memory principle achieves the better clustering among the others. We observe also that, with BiCGstab(2), timings are greater than those obtained with FGMRES, even if the limited memory preconditioners $P_{\omega,\lceil m/10\rceil}$ are always better than their competitors. Therefore, we omitted the numerical results for this case.

Experiment 2 We consider the fractional partial differential equation in two dimension in (3) with the following choice of the coefficients:

$$
\begin{aligned}
&K_x = 2,\ K_y = 1.5,\ c(x,y) = 1 + 0.5\cos(xy),\\
&\mathbf{b} = (\beta + 0.5\sin(4\pi x)\cos(5\pi y), \ldots\\
&\ldots \alpha + 0.7\sin(7\pi y)\cos(4\pi x)),\\
&g(x,y,t) = \sin(5\pi x)\sin(5\pi y)\exp(-t),\\
&u_0(x,y) = xy(x-1)(y-1).
\end{aligned}
\tag{20}
$$

The domain is $\Omega \times [0,T] = [0,1]^2 \times [0,1]$. In Table 3 we give the results for the solution of the semidiscrete problem with the GBDF formula with 2 steps and GMRES(20)/FGMRES(20) iterative methods with the various proposed preconditioners. Similarly to the other experiments, we observe that all the limited memory preconditioners, i.e., based on the short-memory principle, are optimal: the number of iterations to reach a prescribed tolerance is fixed, independent from the dimension. Moreover, similarly to the other experiments, the approach with FGMRES turns out to be the fastest one. In this 2D case, we do not give the results with the circulant approximation of the Jacobian matrix, because it does not give a reasonable spectral approximation for the underlying block-matrix.

5 Conclusions

We presented a strategy for solving the large linear systems generated by discretizing time-dependent fractional partial differential equations (FDEs) integrated in time by using linear multistep formulas used in boundary value form. We use p-shifted Grünwald–Letnikov dis-

Table 2 Experiment 1: coefficients from Eq. (19), fractional order of differentiation $\alpha = 1.8$

GMRES(20)		I		P_5		P_ω		P'_5		P'_ω		$P_{\omega,\lceil m/5 \rceil}$		$P^{\text{FGMRES}}_{\omega,10}$	
m	s	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)
25	32	359	0.498077	28	0.162611	28	**0.026580**	105	0.086074	106	0.064794	30	0.063938	12	0.390941
	64	417	0.065541	28	0.042177	29	0.043950	114	0.155477	108	0.091509	31	**0.035493**	3	0.130120
	128	618	0.230025	29	0.084474	29	0.075497	114	0.229144	115	0.179190	32	**0.048669**	3	0.249523
	256	1004	0.556410	28	**0.129884**	29	0.144670	116	0.350810	116	0.345380	32	0.141315	12	1.929159
	512	1700	1.367553	27	**0.229644**	28	0.254450	119	0.578919	118	0.562342	32	0.165098	12	3.750084
49	32	511	0.061850	28	0.074696	28	0.044302	160	0.147374	158	0.146482	31	**0.024588**	4	0.111798
	64	582	0.213668	28	0.079720	29	0.088193	174	0.228514	179	0.289489	31	**0.043588**	3	0.153173
	128	874	0.474627	29	0.164856	30	0.184416	184	0.380261	189	0.456471	32	**0.080920**	12	1.139513
	256	1361	1.110566	28	0.289902	30	0.355965	192	0.678054	194	0.795719	32	**0.151359**	12	2.129826
	512	2140	2.639608	28	0.579192	29	0.651599	195	1.302407	197	1.433853	32	**0.293215**	12	4.170408
97	32	803	0.300108	28	0.175620	28	0.131053	242	0.345816	253	0.440888	31	**0.057435**	3	0.099084
	64	1004	0.554717	28	0.254270	29	0.278705	275	0.569038	290	0.712328	31	**0.098882**	12	0.719274
	128	1479	1.108901	29	0.518899	30	0.588185	296	0.958323	312	1.192231	32	**0.192708**	12	1.351855
	256	2048	2.361870	28	0.896838	30	1.153711	311	1.818854	324	2.246043	32	**0.374582**	12	2.551376
	512	3066	6.686213	28	1.838700	29	2.046596	320	3.518119	335	4.218966	32	**0.747920**	12	4.802997
193	32	1633	0.928106	28	0.312824	28	0.328000	359	0.907019	371	1.182327	31	**0.174783**	12	0.564502
	64	2216	1.758452	28	0.845711	29	0.839759	437	1.579116	454	2.007617	32	**0.318097**	12	1.026921
	128	2833	3.458109	29	1.701079	30	1.625637	502	3.061141	516	3.689773	32	**0.652213**	12	1.870541
	256	3636	8.624317	28	2.893450	30	3.241947	538	5.941900	558	7.393889	32	**1.311898**	12	3.556693
	512	5192	25.985645	28	5.410698	29	5.354344	567	11.399223	591	14.493837	32	**2.580167**	13	7.276354

Table 2 continued

GMRES(20)		I		$P_{\mathfrak{s}}$		P_{ω}		$P'_{\mathfrak{s}}$		P'_{ω}		$P_{\omega,\lceil m/5\rceil}$		$P^{\text{FGMRES}}_{\omega,10}$	
m	s	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)
385	32	4483	3.821502	28	1.335643	28	1.265713	512	2.577256	517	3.118356	31	**0.565555**	12	0.926664
	64	4046	7.132265	28	2.835499	29	3.091758	655	4.619358	671	5.792371	32	**1.131976**	12	1.678028
	128	6815	21.917207	29	5.495737	30	8.458964	810	9.476015	826	11.498704	32	**2.122971**	12	3.047989
	256	8692	48.467187	28	13.032433	30	13.832191	944	19.470493	963	24.454555	32	**4.353830**	12	5.621248
	512	10,024	114.332704	28	27.187875	29	25.970556	1035	45.186098	1057	53.588831	32	**8.243228**	13	11.728037
769	32	17,445	34.674128	28	10.607206	28	9.517020	722	6.652855	734	9.026140	31	3.160935	12	**1.543622**
	64	16,028	56.537935	28	19.210925	29	20.501788	1010	13.752854	1025	17.565095	32	6.743467	12	**2.712531**
	128	†	†	29	44.872253	30	49.015992	1305	29.568440	1339	36.218418	32	14.076085	13	**5.460369**
	256	†	†	28	81.530878	30	98.281073	1573	70.327472	1621	92.542980	32	28.237991	13	**10.281042**
	512	†	†	28	159.517824	29	181.350848	1817	199.941485	1852	239.047024	32	57.684876	12	**18.724137**
1537	32	†	†	28	72.905661	28	77.110471	978	20.342664	978	25.780316	31	15.895901	12	**2.313077**
	64	†	†	28	150.587943	29	168.761350	1385	42.090657	1396	51.705245	32	33.590205	13	**4.522511**
	128	†	†	29	335.425009	30	369.889844	1902	102.604220	1923	121.577188	32	67.754216	13	**8.845778**
	256	†	†	28	613.583401	30	743.860882	2483	307.027933	2503	362.077657	32	135.465453	13	**17.389843**
	512	†	†	28	1234.166093	29	1377.924956	2968	859.863198	2970	931.043836	32	278.942161	13	**33.900990**

The method used is GMRES(20) for all the preconditioners except for $P^{\text{FGMRES}}_{\omega,10}$, using the approach based on FGMRES as discussed in Sect. 3
We reported in boldface the better timings achieved

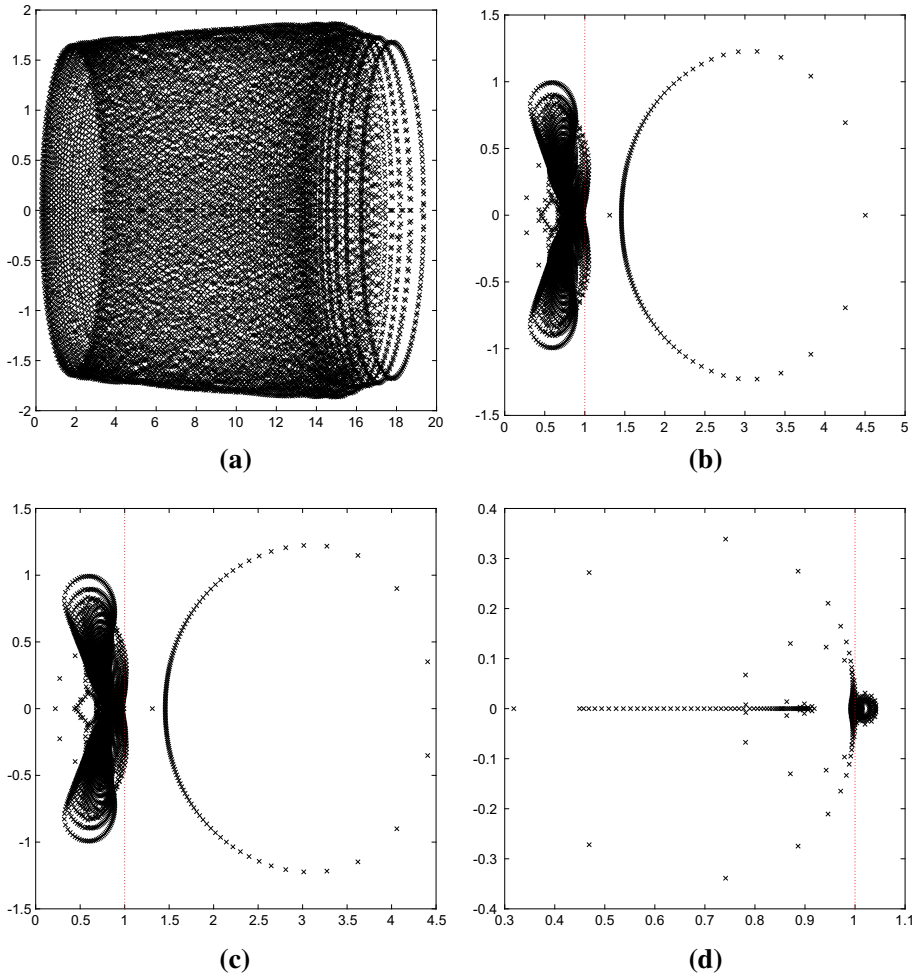

Fig. 4 Experiment 1. Spectra of both the matrix of the system and of the preconditioned matrices with $\alpha = 2$ and 2 step GBDF formula with $m = 97$ and $s = 128$. **a** Eigenvalues of the block matrix M in (9). **b** Eigenvalues of $(P'_s)^{-1}M$. **c** Eigenvalues of $(P'_\omega)^{-1}M$. **d** Eigenvalues of $(P_{\omega,\lceil m/10\rceil})^{-1}M$

cretization in space for the fractional Riemann–Liouville operators and fractional centered discretization for the symmetric Riesz derivative.

We proposed to solve the underlying block structured with dense blocks linear systems by restarted GMRES [31] and FGMRES [29] using our hybrid preconditioners. Several tests have been performed with other methods, either as solver for the whole problem or in conjunction with the preconditioner for the FGMRES method, they are not reported because less competitive. Among them we mention: IDR(s) [33], BiCGStab [34], BiCGStab(2) [32], alternate directions, various versions of gaussian elimination, most popular Krylov iterative solvers with several incomplete factorization preconditioners, etc.

The preconditioners proposed here are based on block ω-circulant matrices and a short-memory approximation of the underlying Jacobian matrix of the FDE.

Table 3 Experiment 2: coefficients from Eq. (20), fractional order of differentiation $2\alpha = 1.1$, $2\beta = 1.8$

GMRES(20)		I		P'_5		$P_{\omega,\lceil m/10\rceil}$		$P^{\text{FGMRES}}_{\omega,10}$		$P^{\text{FGMRES}}_{\omega,5}$	
m	s	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)	M_v	T(s)
25×25	32	266	**0.340921**	114	1.709925	34	2.509950	3	0.687541	3	0.220345
	64	354	**0.737725**	126	2.498359	33	4.031727	12	1.641349	11	1.411272
	128	520	**2.310232**	135	4.906858	33	8.018807	12	2.991839	12	2.874120
	256	838	7.673421	137	9.559409	32	14.279442	13	6.076410	13	**5.953192**
	512	1471	33.862683	138	18.976753	31	26.585789	13	12.048800	13	**11.433036**
49×49	32	512	3.470923	537	51.713398	34	26.305325	12	2.383546	12	**2.216880**
	64	633	8.502515	1214	228.157016	34	51.005210	12	4.383754	12	**3.791248**
	128	861	26.832703	1261	430.258106	33	100.230792	13	9.349177	13	**8.199694**
	256	1269	120.173661	1606	1192.707353	32	192.221346	13	19.039451	13	**17.025160**
	512	1985	443.243425	2391	3661.130769	32	433.940684	13	37.788963	13	**34.709295**
97×97	32	1209	77.714770	1919	1531.313896	34	1208.996561	13	10.958864	13	**9.126362**
	64	1362	220.144413	8051	12,804.937808	34	2535.711822	13	20.057837	13	**17.836464**
	128	1622	577.350800	–	> 4 h	34	5216.717315	13	38.867441	6	**15.510864**
	256	2275	1649.682022	–	> 4 h	32	9198.798614	13	75.631715	13	**69.373086**
	512	3257	4636.613558	–	> 4 h	31	16,705.028289	14	160.233433	14	**147.280593**

We use GMRES(20) for all the proposed preconditioners with the exception of $P^{\text{FGMRES}}_{\omega,\cdot}$, used with FGMRES. See also the discussion in Sect. 3 for more details
We reported in boldface the better timings achieved

Theoretical convergence results for our limited memory block ω-circulant preconditioners are also given, confirming that the preconditioned matrices have a clustered spectrum of eigenvalues.

Numerical tests and comparisons with other techniques proposed in the recent literature confirm the effectiveness of the following two preconditioned multi iterative solvers proposed here:

- FGMRES using a hybrid preconditioner based on block ω-circulant matrices and in which the auxiliary linear systems are solved with GMRES(20). The latter is preconditioned by a sequence of approximate inverse Toeplitz preconditioners computed on the short-memory approximations of the underlying Jacobian matrices, and,
- restarted GMRES with a block ω-circulant preconditioner with approximate inverse Toeplitz preconditioners using a short-memory approximation of the underlying Jacobian matrices and direct solution of the auxiliary linear systems.

We observe also that the preconditioners $P_{\omega,k}$ and $P_{\omega,k}^{\mathrm{FGMRES}}$, i.e., those based on the short-memory principle, are *optimal* in the sense that the number of iterations to reach a prescribed tolerance is almost independent from the mesh used.

Acknowledgements We wish to thank two anonymous referees for their constructive comments which have improved the readability of the paper.

References

1. Axelsson, A., Verwer, J.: Boundary value techniques for initial value problems in ordinary differential equations. Math. Comp. **45**, 153–171 (1985)
2. Bertaccini, D.: P-Circulant Preconditioners and the Systems of the ODE Codes, IMACS Series in Computational and Applied Mathematics, vol. 5, pp. 179–193. IMACS, New Brunswick (1999)
3. Bertaccini, D.: A circulant preconditioner for the systems of LMF-based ODE codes. SIAM J. Sci. Comput. **22**(3), 767–786 (2000)
4. Bertaccini, D.: Reliable preconditioned iterative linear solvers for some numerical integrators. Numer. Linear Algebra Appl. **8**(2), 111–125 (2001)
5. Bertaccini, D.: The spectrum of circulant-like preconditioners for some general linear multistep formulas for linear boundary value problems. SIAM J. Numer. Anal. **40**(5), 1798–1822 (2002)
6. 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). https://doi.org/10.1007/s11075-016-0186-8
7. Bertaccini, D., Durastante, F.: Iterative Methods and Preconditioning for Large and Sparse Linear Systems with Applications, Monographs and Research Notes in Mathematics. CRC Press, Chapman & Hall/CRC, London (2018)
8. Bertaccini, D., Ng, M.K.: Skew–circulant preconditioners for systems of LMF–based ODE codes. In: International Conference on Numerical Analysis and Its Applications, pp. 93–101. Springer (2000)
9. Bertaccini, D., Ng, M.K.: The convergence rate of block preconditioned systems arising from LMF-based ode codes. BIT **41**(3), 433–450 (2001)
10. Bertaccini, D., Ng, M.K.: Band-Toeplitz preconditioned GMRES iterations for time-dependent PDEs. BIT **43**(5), 901–914 (2003)
11. Bertaccini, D., Ng, M.K.: Block $\{\omega\}$-circulant preconditioners for the systems of differential equations. Calcolo **40**(2), 71–90 (2003)
12. Brugnano, L., Trigiante, D.: Solving Differential Equations by Multistep Initial and Boundary Value Methods. Stability and Control: Theory, Methods and Applications. Taylor & Francis, London (1998)
13. Chan, R., Ng, M., Jin, X.Q.: Strang-type preconditioner for systems of LMF-based ODE codes. IMA J. Numer. Anal. **21**(2), 451–62 (2001)
14. Donatelli, M., Mazza, M., Serra-Capizzano, S.: Spectral analysis and structure preserving preconditioners for fractional diffusion equations. J. Comp. Phys. **307**, 262–279 (2016)
15. Fischer, R., Huckle, T.: Using ω-circulant matrices for the preconditioning of toeplitz systems. Selçuk J. Appl. Math **4**, 71–88 (2003)

16. Garoni, C., Serra-Capizzano, S.: Generalized Locally Toeplitz Sequences: Theory and Applications, 1st edn. Springer, New York (2017)
17. Gu, X.M., Huang, T.Z., Ji, C.C., Carpentieri, B., Alikhanov, A.A.: Fast iterative method with a second-order implicit difference scheme for time-space fractional convection–diffusion equation. J. Sci. Comput. **72**(3), 957–985 (2017)
18. Gu, X.M., Huang, T.Z., Zhao, X.L., Li, H.B., Li, L.: Strang-type preconditioners for solving fractional diffusion equations by boundary value methods. J. Comput. Appl. Math. **277**, 73–86 (2015)
19. Hanke, M., Nagy, J.G.: Toeplitz approximate inverse preconditioner for banded Toeplitz matrices. Numer. Algorithms **7**(2), 183–199 (1994). https://doi.org/10.1007/BF02140682
20. Li, C., Zeng, F.: Numerical Methods for Fractional Calculus. Numerical Analysis and Scientific Computing Series. CRC Press, Chapman & Hall/CRC, London (2015)
21. Lin, F.R., Yang, S.W., Jin, X.Q.: Preconditioned iterative methods for fractional diffusion equation. J. Comp. Phys. **256**, 109–117 (2014)
22. Meerschaert, M.M., Tadjeran, C.: Finite difference approximations for fractional advection-dispersion flow equations. J. Comput. Appl. Math. **172**(1), 65–77 (2004)
23. Meerschaert, M.M., Tadjeran, C.: Finite difference approximations for two-sided space-fractional partial differential equations. Appl. Numer. Math. **56**(1), 80–90 (2006)
24. Ng, M.K.: Iterative Methods for Toeplitz Systems. Oxford University Press, Oxford (2004)
25. Ortigueira, M.D.: Riesz potential operators and inverses via fractional centred derivatives. Int. J. Math. Math. Sci. (2006). https://doi.org/10.1155/IJMMS/2006/48391
26. Pang, H.K., Sun, H.W.: Fast numerical contour integral method for fractional diffusion equations. J. Sci. Comput. **66**(1), 41–66 (2016)
27. Podlubny, I.: Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications, vol. 198. Academic Press, New York (1998)
28. Popolizio, M.: A matrix approach for partial differential equations with riesz space fractional derivatives. Eur. Phys. J. Spec. Top. **222**(8), 1975–1985 (2013)
29. Saad, Y.: A flexible inner–outer preconditioned GMRES algorithm. SIAM J. Sci. Comput. **14**(2), 461–469 (1993)
30. Saad, Y.: Iterative Methods for Sparse Linear Systems, 2nd edn. Society for Industrial and Applied Mathematics, Philadelphia (2003)
31. Saad, Y., Schultz, M.H.: Gmres: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **7**(3), 856–869 (1986)
32. Sleijpen, G.L., Fokkema, D.R.: BiCGstab(l) for linear equations involving unsymmetric matrices with complex spectrum. Electron. Trans. Numer. Anal. **1**(11), 2000 (1993)
33. Sonneveld, P., Van Gijzen, M.B.: IDR(s): a family of simple and fast algorithms for solving large non-symmetric systems of linear equations. SIAM J. Sci. Comput. **31**(2), 1035–1062 (2008)
34. Van der Vorst, H.A.: Bi-CGSTAB: a fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **13**(2), 631–644 (1992)
35. Van der Vorst, H.A.: Iterative Krylov Methods for Large Linear Systems, vol. 13. Cambridge University Press, Cambridge (2003)
36. Wang, H., Wang, K.: An $o(n \log^2 n)$ alternating-direction finite difference method for two-dimensional fractional diffusion equations. J. Comp. Phys. **230**(21), 7830–7839 (2011)
37. Wang, H., Wang, K., Sircar, T.: A direct $o(n \log^2 n)$ finite difference method for fractional diffusion equations. J. Comp. Phys. **229**(21), 8095–8104 (2010)
38. Zhang, L., Sun, H.W., Pang, H.K.: Fast numerical solution for fractional diffusion equations by exponential quadrature rule. J. Comp. Phys. **299**, 130–143 (2015)