Boundary Value Problems
a SpringerOpen Journal

RESEARCH Open Access

A fast multiscale Galerkin algorithm for solving boundary value problem of the fractional Bagley–Torvik equation

Jian Chen[1]*

*Correspondence:
chenjian704@163.com
[1] School of Mathematics and Big Data, Foshan University, Foshan, China

Abstract

In this paper, a fast multiscale Galerkin algorithm is developed for solving the boundary value problem of the fractional Bagley–Torvik equation. For this purpose, we employ multiscale orthogonal functions having vanishing moments as the basis of the trial space, and we propose a truncation strategy for the coefficient matrix of the corresponding discrete system which leads to a fast algorithm. We show the algorithm has nearly linear computational complexity (up to a logarithmic factor). Numerical experiments are presented to illustrate the efficiency, accuracy and convergence of the proposed algorithm. Also, comparisons with some other existing methods are made to confirm the reliability of the algorithm

Keywords: Fast multiscale algorithm; Matrix truncation; Caputo fractional derivative; Fractional Bagley–Torvik equation

1 Introduction

Over the last few decades, fractional calculus has attracted significant interest of many researchers. Both fractional integral operators and derivative operators are non-local operators, which makes these operators very suitable for describing long term memory, asymptotic scaling and hereditary properties of various phenomena. Therefore, the fractional equations appear frequently in many fields such as viscoelasticity, fluid mechanics, biochemistry, signal processing, digital control theory, bioengineering, finance and thermoelastic materials; see [1–9] and the references therein. Motivated by increasing applications of fractional equations, the analytical and numerical methods [10–21] for the solution of the fractional equations became a subject of intensive investigations.

In this paper, we consider the following Bagley–Torvik equation:

$$u''(t) + \theta D^{\alpha} u(t) + \sigma u(t) = f(t), \quad 0 \le t \le 1, \tag{1}$$

subject to boundary conditions:

$$u(0) = u(1) = 0,$$

where θ, σ and $0 < \alpha < 1$ are constants, and $f(t)$ is a continuous function defined on the interval $[0, 1]$. Here the notation D^α represents the Caputo fractional derivative defined by

$$D^\alpha u(t) := \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{u'(s)}{(t-s)^\alpha} \,\mathrm{d}s, \quad 0 < \alpha < 1,$$

where $\Gamma(\cdot)$ is the gamma function defined by

$$\Gamma(s) := \int_0^\infty e^{-x} x^{s-1} \,\mathrm{d}x.$$

The Bagley–Torvik equation was first introduced by Bagley and Torvik and treated as a mathematical model of the motion of a thin rigid plate immersed in a Newtonian fluid [22]. It was then widely used to simulate the dynamic responses of viscoelastically damped structures [23]. Many numerical approaches have been developed for solving the Bagley–Torvik equation. These methods include the modified Galerkin method [10], collocation methods [12, 13], the wavelet method [14, 15], the finite difference method [16], the spline method [17–19] and the hybridizable discontinuous Galerkin method [20], and the operational Tau method [21]. However, due to the non-local property of the fractional differential operators, the numerical methods for fractional equation including Bagley–Torvik equation lead to dense coefficient matrices. When the dimension of coefficient matrix is large, the computational cost for generating the matrix and then solving the corresponding linear system is huge.

The main purpose of this paper is to develop a fast multiscale Galerkin algorithm for solving the Bagley–Torvik equation. We introduce a matrix truncation strategy by choosing multiscale orthogonal basis functions having vanishing moments, which forms a basis for the fast algorithm for solving Eq. (1). Specifically, the multiscale orthonormal basis constructed in [24] is employed to discretize the Bagley–Torvik equation, which leads to a linear system with a numerically sparse coefficient matrix. That is, the absolute values of most of the entries of the matrix are very small. We then set the entries with small value to zero by a matrix truncation strategy, which yields a sparse matrix. We show that the number of nonzero entries of the truncated matrix is linear (up to a logarithmic factor) with respect to the dimensions of the matrix. This method cannot only generate the coefficient matrix rapidly, but also make it easy to solve the resulting linear system with sparse coefficient matrix.

This paper is organized as follows. In Sect. 2, the multiscale orthonormal bases in Sobolev space $H_0^1(0, 1)$ is introduced. In Sect. 3, the fast multiscale Galerkin algorithm with a practical matrix truncation strategy is proposed to solve the Bagley–Torvik equation boundary value problem. Some numerical examples are presented in Sect. 4, and the conclusion is included in the last section.

2 Multiscale orthonormal bases

The multiscale orthonomal bases for Sobolev space on the unit interval $I = [0, 1]$ were constructed by Chen, Wu and Xu in Ref. [24] for solving differential equations. For the convenience of the reader, we briefly introduce the construction and properties of this multiscale orthonomal bases, which are required necessarily for designing the fast algorithm for the Bagley–Torvik equation.

We start with some useful notations. Let $\mathbb{N} := \{1, 2, \dots\}$, $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$ and $\mathbb{Z}_n := \{0, 1, 2, \dots, n-1\}$, for $n \in \mathbb{N}$. We denote by $(\cdot, \cdot)$ the inner product on the space $L^2(I)$ with the L^2 norm $\|\cdot\|_2$. Let $H_0^1(I)$ denote the Sobolev space of elements u vanishing at the endpoints that $u(0) = u(1) = 0$. The inner product and norm of $H_0^1(I)$ are defined by

$$\langle u, v\rangle_1 := \left(u', v'\right) = \int_0^1 u'(t)v'(t)\,\mathrm{d}t, \quad u, v \in H_0^1(I),$$

and

$$\|u\|_1 := \sqrt{\langle u, u\rangle_1}, \quad u \in H_0^1(I),$$

respectively. Let Δ_n be the uniform mesh which divides the interval I into 2^n pieces, and let $\mathbb{X}_n$ be the piecewise polynomial space of order k associated with the partition Δ_n. It is concluded that the sequence of $\mathbb{X}_n$ is nested, i.e.,

$$\mathbb{X}_n \subset \mathbb{X}_{n+1}, \quad n \in \mathbb{N}_0,$$

which yields the decomposition

$$\mathbb{X}_n = \mathbb{X}_{n-1} \oplus^\perp \mathbb{W}_n,$$

where $\mathbb{W}_n$ is the orthogonal complement of $\mathbb{X}_{n-1}$ in $\mathbb{X}_n$ in the sense of the inner product $\langle\cdot, \cdot\rangle_1$. Repeating this decomposition, we have the multiscale space decomposition as follows:

$$\mathbb{X}_n = \mathbb{W}_0 \oplus^\perp \mathbb{W}_1 \oplus^\perp \cdots \oplus^\perp \mathbb{W}_n,$$

where $\mathbb{W}_0 := \mathbb{X}_0$. Denoting $x(n) := \dim \mathbb{X}_n$ and $w(n) := \dim \mathbb{W}_n$, by the definition of $\mathbb{X}_n$ we have

$$x(n) = (k-1)2^n - 1, \qquad w(n) = x(n) - x(n-1) = (k-1)2^{n-1}.$$

The following lemma shows that when the bases of $\mathbb{X}_0$ and $\mathbb{W}_1$ are constructed, the basis of the space $\mathbb{X}_n$ can be recursively generated, and the sequence of $\mathbb{X}_n$ is ultimately dense in the Sobolev space $H_0^1(I)$.

Lemma 2.1 (See [24]) *Let w_{ij}, $j \in \mathbb{Z}_{w(i)}$, be an orthonormal basis of the space $\mathbb{W}_i$, $i \geq 1$. Then the orthonormal basis of the space $\mathbb{W}_{i+1}$ can be recursively constructed by*

$$\left.\begin{aligned} w_{i+1,j}(t) &= \frac{\sqrt{2}}{2} w_{ij}(2t), \\ w_{i+1,w(i)+j}(t) &= \frac{\sqrt{2}}{2} w_{ij}(2t-1), \end{aligned}\right. \qquad t \in [0,1], j \in \mathbb{Z}_{w(i)}, i \geq 1.$$

Furthermore,

$$H_0^1(I) = \overline{\mathbb{W}_0 \oplus^\perp \mathbb{W}_1 \oplus^\perp \cdots \oplus^\perp \mathbb{W}_n \oplus^\perp \cdots}.$$

We define $J_n := \{(i,j), j \in \mathbb{Z}_{w(i)}, i \in \mathbb{Z}_{n+1}\}$ and assume that $\mathbb{W}_i$ has an orthonormal basis $\{w_{ij} : j \in \mathbb{Z}_{w(i)}\}$, then $\{w_{ij} : (i,j) \in J_n\}$ forms an orthonormal basis of $\mathbb{X}_n$. The basis has several important properties, we describe parts of them that are relevant to our algorithm.

(P1) (Compact support) The support $S_{ij} := \operatorname{supp} w_{ij}$ holds that $d_i \sim 2^{-i}$, where $d_i := \max\{\operatorname{diam}(S_{ij}) : j \in \mathbb{Z}_{w(i)}\}$. For any $i \in \mathbb{N}$ and $j \in \mathbb{Z}_{\omega(i)}$, there are at most ρ (independent of i and j) numbers of $w_{ij'}+, j' \in \mathbb{Z}_{\omega(i)}$, such that $\operatorname{meas}(S_{ij} \cap S_{ij'}) \neq 0$.

(P2) (Orthogonality) For any $i, i' \in \mathbb{N}_0$,

$$\langle w_{ij}, w_{i'j'}\rangle_1 = \delta_{ii'}\delta_{jj'}, \quad j \in \mathbb{Z}_{\omega(i)}, j' \in \mathbb{Z}_{\omega(i')},$$

where $\delta_{ii'}$ is the Kronecker delta.

(P3) (Vanishing moment) For any $i \geq 1$ and $p \in \mathbb{Z}_{k-1}$,

$$\left\langle w_{ij}, t^p \right\rangle_1 = 0, \quad j \in \mathbb{Z}_{\omega(i)}.$$

This, with integration by parts, implies that $(w_{ij}, f) = 0$, where f is an arbitrary polynomial with order $\leq k-2$.

(P4) (Boundedness) There exists a constant $c > 0$ such that

$$\|w_{ij}\|_1 = 1 \quad \text{and} \quad \| w_{ij} \|_\infty \leq c2^{-i/2}, \quad i \in \mathbb{N}_0, j \in \mathbb{Z}_{\omega(i)}.$$

To end this section, we give the concrete multiscale linear basis functions which will be used in our numerical experiments. For linear case, $k = 2$, $\dim(\mathbb{W}_i) = 2^{i-1}$, for $i \geq 1$. Obviously, 0 is the only linear function which vanishes at both 0 and 1. That is $\mathbb{X}_0 = \{0\}$. The basis of the space $\mathbb{W}_1$ is given by (see [24])

$$w_{10}(t) = \begin{cases} t, & 0 \leq t < \frac{1}{2}, \\ 1-t, & \frac{1}{2} \leq t \leq 1. \end{cases}$$

Using $w_{10}(t)$ and Lemma 2.1, we can generate the bases of the spaces $\mathbb{W}_2, \mathbb{W}_3, \ldots, \mathbb{W}_n$ in turn, and then we obtain the basis of the approximation subspace $\mathbb{X}_n$. We illustrate in Fig. 1 the graph of the basis functions for $\mathbb{X}_6$.

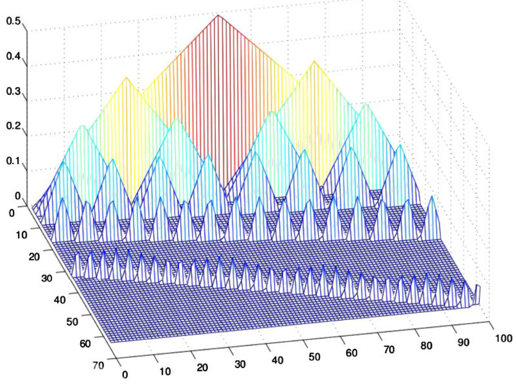

Figure 1 Multiscale orthonormal basis for $\mathbb{X}_6$

3 Fast multiscale Galerkin algorithm

In this section, we first present a Galerkin scheme associate with the multiscale basis introduced in last section for solving Eq. (1). Then a matrix truncation strategy is proposed, which leads to the fast multiscale Galerkin algorithm.

We begin with the variational (weak) formulation of the (1) subject to the boundary conditions $u(0) = u(1) = 0$, which reads: find $u \in H_0^1(I)$ such that

$$-\left(u', v'\right) + \theta\left(D^\alpha u, v\right) + \sigma(u, v) = (f, v), \quad \forall v \in H_0^1(I). \tag{2}$$

Making use of the basis of $\mathbb{X}_n$, the Galerkin scheme for solving Eq. (2) is to seek a vector $\mathbf{c}_n := [c_{ij} : (i,j) \in J_n]^\top$ such that

$$u_n(t) := \sum_{(i,j)\in J_n} c_{ij} w_{ij}(t) \in \mathbb{X}_n,$$

satisfies

$$\sum_{(i,j)\in J_n} \left[-\left(w'_{ij}, w'_{i'j'}\right) + \theta\left(D^\alpha w_{ij}, w_{i'j'}\right) + \sigma(w_{ij}, w_{i'j'})\right] c_{ij} = (f, w_{i'j'}), \quad \left(i', j'\right) \in J_n. \tag{3}$$

To distinguish the method from the traditional Galerkin method, the scheme (3) is called the multiscale Galerkin method due to the use of the multiscale basis $\{w_{ij}, (i,j) \in J_n\}$. Let

$$I_{i'j';ij} := \left(w'_{ij}, w'_{i'j'}\right), \qquad D_{i'j';ij} := \left(D^\alpha w_{ij}, w_{i'j'}\right), \qquad E_{i'j';ij} := (w_{ij}, w_{i'j'}).$$

We define the matrices

$$\mathbf{I}_n := \left[I_{i'j';ij} : \left(i', j'\right), (i,j) \in J_n\right], \qquad \mathbf{D}_n := \left[D_{i'j';ij} : \left(i', j'\right), (i,j) \in J_n\right],$$
$$\mathbf{E}_n := \left[E_{i'j';ij} : \left(i', j'\right), (i,j) \in J_n\right],$$

and vector $\mathbf{f}_n := [-(f, w_{i'j'}) : (i', j') \in J_n]^\top$. Using these notations, we write the linear system (3) in matrix form

$$(\mathbf{I}_n - \theta \mathbf{D}_n - \sigma \mathbf{E}_n)\mathbf{c}_n = \mathbf{f}_n. \tag{4}$$

From the properties (P1) and (P2), it is easily seen that both matrices $\mathbf{I}_n$ and $\mathbf{E}_n$ are sparse. In fact, $\mathbf{I}_n$ is an identity. However, $\mathbf{D}_n$ is a dense matrix because of the non-local property of the fractional order differential operator. Moreover, all the entries of $\mathbf{D}_n$ are defined by singular integrals

$$D_{i'j',ij} = \int_0^1 \left(\int_0^t \frac{w'_{ij}(s)}{(t-s)^\alpha}\, \mathrm{d}s\right) \cdot w_{i'j'}(t)\, \mathrm{d}t, \quad (i,j), \left(i', j'\right) \in J_n.$$

Therefore, it is computationally costly to establish such a matrix numerically, which becomes an obvious computational deficiency of the Galerkin method for solving such a kind of equations. Fortunately, we observe that the matrix $\mathbf{D}_n$ is numerically sparse under the multiscale orthonormal basis. That is, a large number of the entries are very small in

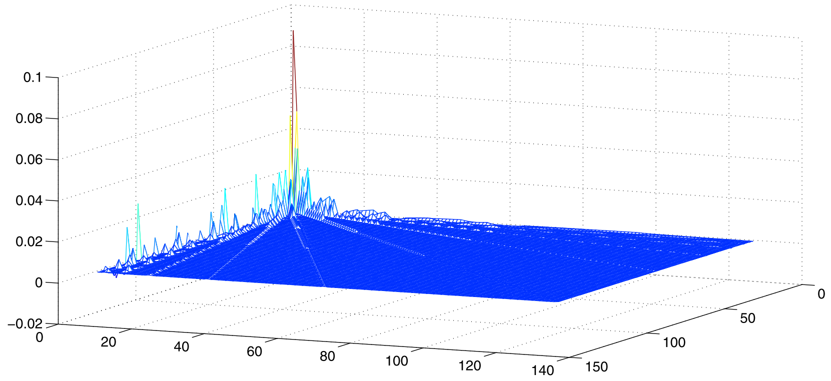

Figure 2 $\mathbf{D}_7$ with respect to the multiscale orthonormal basis

magnitude. To visualize this observation, we plot in Fig. 2 the matrix $\mathbf{D}_n$ with respect to the multiscale basis with $n = 7$ and $\alpha = \frac{1}{2}$.

According to the different scales of spaces, we partition matrix $\mathbf{D}_n$ into a block matrix $\mathbf{D}_n = [\mathbf{D}_{ii'} : i, i' \in \mathbb{Z}_{n+1}]$ with $\mathbf{D}_{ii'} = [D_{ij,i'j'} : j \in \mathbb{Z}_{w(i)}, j' \in \mathbb{Z}_{w(i')}]$. We can see from Fig. 2 that the absolute values of the entries lying outside the diagonals of the blocks are small, which is similar to the case of multiscale methods for solving the Fredholm integral equation [25–27]. This observation motivates us to "truncate" the small entries to zero by the same matrix truncation strategy as the references [25–27]. Specifically, For $(i,j),(i',j') \in J_n$, we define

$$\tilde{\mathbf{D}}_{ij,i'j'} = \begin{cases} \mathbf{D}_{ij,i'j'}, & \operatorname{dist}(S_{ij}, S_{i'j'}) \le \varepsilon^n_{ii'}, \\ 0, & \text{otherwise}, \end{cases} \tag{5}$$

where $\operatorname{dist}(S_{ij}, S_{i'j'})$ is the distance between S_{ij} and $S_{i'j'}$, and the truncation parameter $\varepsilon^n_{ii'}$ are chosen as [25]

$$\varepsilon^n_{ii'} := \max\bigl\{\mu 2^{-n+\lambda(n-i)+\lambda'(n-i')}, \rho\bigl(2^{-i} + 2^{-i'}\bigr)\bigr\}$$

for some positive constants μ, λ, λ' and $\rho > 1$. Then we obtain a truncation matrix

$$\tilde{\mathbf{D}}_n := \bigl[\tilde{\mathbf{D}}_{ii'}, i, i' \in \mathbb{Z}_{n+1}\bigr],$$

where

$$\tilde{\mathbf{D}}_{ii'} := \tilde{\mathbf{D}}\bigl(\varepsilon^n_{ii'}\bigr)_{ii'} = \bigl[\tilde{\mathbf{D}}_{ij,i'j'}, j \in \mathbb{Z}_{w(i)}, j' \in \mathbb{Z}_{w(i')}\bigr].$$

Replacing $\mathbf{D}_n$ in Eq. (4) by $\tilde{\mathbf{D}}_n$ leads to the fast multiscale Galerkin algorithm (FMGA): find $\tilde{\mathbf{c}}_n := [\tilde{c}_{ij}, (i,j) \in J_n]^\top$ such that

$$(\mathbf{I}_n - \theta\tilde{\mathbf{D}}_n - \sigma\mathbf{E}_n)\tilde{\mathbf{c}}_n = \mathbf{f}_n. \tag{6}$$

Solving $\tilde{\mathbf{c}}_n$ from Eq. (6), we then obtain the fast multiscale Galerkin solution

$$\tilde{u}_n(t) := \sum_{(i,j)\in J_n} \tilde{c}_{ij} w_{ij}(t).$$

Similar to the analysis of [26, 27] or Theorem 3.3 of [25], we have the following lemma about the computational complexity of the fast multiscale Galerkin scheme (6), which is measured by the number of nonzero entries of the truncation matrix $\tilde{\mathbf{D}}_n$.

Theorem 3.1 *For any $n \in \mathbb{N}$, we have*

$$\mathcal{N}(\tilde{\mathbf{D}}_n) = \mathcal{O}\bigl(n2^n\bigr) = \mathcal{O}\bigl(x(n)\log x(n)\bigr),$$

where $\mathcal{N}(M)$ denotes the number of nonzero entries of matrix M.

The above theorem shows that the truncation matrix $\tilde{\mathbf{D}}_n$ is sparse, which is very beneficial to the solution of the resulting large linear system by iterative method. More importantly, it greatly reduces the computational burden for calculating all the entries of matrix, in other words, it greatly saves the time for generating the coefficient matrix.

4 Numerical examples

In this section, we present some numerical illustrations for the solution of the boundary value problem of the Bagley–Torvik equation to show the accuracy and the efficiency of the proposed method. All the numerical calculations are implemented with Matlab 2012a in OS Windows 10 with 2.5 G CPU and 8 G memory.

Example 1 Consider the following Bagley–Torvik equation [17–19]:

$$\begin{aligned} &u''(t) + \theta D^\alpha u(t) + \sigma u(t) = f(t), \quad 0 \le t \le 1, \\ &u(0) = u(1) = 0, \end{aligned}$$

where $f(t) = (\lambda - 1)(\lambda t - \lambda + 2)t^{\lambda-3} + \theta \frac{(\lambda-1)!}{\Gamma(\lambda-\alpha)}(\frac{\lambda t}{\lambda-\alpha} - 1)t^{\lambda-\alpha-1} + \sigma t^{\lambda-1}(t-1)$. The exact solution of this equation is $u^*(t) = t^{\lambda-1}(t-1)$. We use the linear multiscale basis to discretize the equation. In order to illustrate the computational efficiency of the FMGA, we solve the equation by both the original scheme (4) (OMGM) and the truncated scheme (6) (FMGA). In all three examples, we choose $\mu = \rho = 2$, $\lambda = 1$ and $\lambda' = \frac{5}{6}$ in the truncation parameter $\varepsilon^n_{ii'}$. The numerical results for $\theta = 0.5$, $\sigma = 1$, $\lambda = 5$ and $\alpha = 0.5$ are reported in Tables 1 and 2. In the two tables, n, $x(n)$ and T $(\tilde{T})$ denote the level number of approximation space, the dimension of approximation space and the time of OMGM (FMGA) for generating the coefficient matrix, respectively. The notations $\|\cdot\|_1$ and $\|\cdot\|_2$ stand for the H^1_0-error and L^2-error, respectively, and the corresponding convergence orders (C.O.) are computed by the formulas

$$\text{C.O.} := \log_2 \frac{\|u^* - \tilde{u}_{n-1}\|_p}{\|u^* - \tilde{u}_n\|_p}, \qquad \text{C.O.} := \log_2 \frac{\|u^* - u_{n-1}\|_p}{\|u^* - u_n\|_p},$$

where $p = 1$ or 2; $\tilde{u}_n$ and u_n are the numerical solutions solved by FMGA and OMGM, respectively. It is seen from Table 1 that FMGA and OMGM have the same optimal convergence orders 1 and 2 in the H^1_0 norm and L^2 norm, respectively, and they have almost the same accuracy, which implies that our truncation strategy does not affect the overall accuracy. However, the computational time in Table 2 and Fig. 3 indicate that FMGA is remarkably faster than OMGM. To measure the degree of sparsity of the truncation matrix,

Table 1 Numerical results for Example 1 ($\theta = 0.5$, $\sigma = 1$, $\lambda = 5$, $\alpha = 0.5$)

n	$x(n)$	FMGA				OMGM			
		$\|u^* - \tilde{u}_n\|_1$	C.O.	$\|u^* - \tilde{u}_n\|_2$	C.O.	$\|u^* - u_n\|_1$	C.O.	$\|u^* - u_n\|_2$	C.O.
4	15	4.3753e-2		8.8248e-4		4.3753e-2		8.8244e-4	
5	31	2.1963e-2	0.9943	2.2097e-4	1.9977	2.1963e-2	0.9943	2.2089e-4	1.9982
6	63	1.0992e-2	0.9986	5.5325e-5	1.9978	1.0992e-2	0.9986	5.5278e-5	1.9986
7	127	5.4975e-3	0.9996	1.3886e-5	1.9943	5.4975e-3	0.9996	1.3867e-5	1.9950
8	255	2.7489e-3	0.9999	3.4821e-6	1.9956	2.7489e-3	0.9999	3.4757e-6	1.9963
9	511	1.3745e-3	1.0000	8.7666e-7	1.9899	1.3745e-3	1.0000	8.7453e-7	1.9907
10	1023	6.8724e-3	1.0000	2.2071e-7	1.9899	6.8724e-3	1.0000	2.1997e-7	1.9912

Table 2 Comparison of the computational time for FMGA and OMGM for Example 1

	n						
	4	5	6	7	8	9	10
C.R.	62.2%	44.3%	29.3%	18.5%	11.2%	6.69%	3.89%
$\tilde{T}(s)$	0.01	0.03	0.09	0.24	0.68	1.95	4.88
$T(s)$	0.01	0.04	0.17	0.69	2.82	11.5	47.4

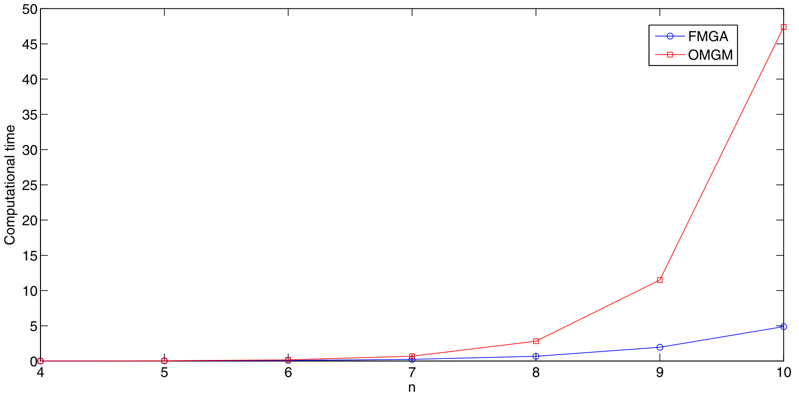

Figure 3 Growth of computational time for Example 1

we denote by C.R. in Table 2 the ratio of number of nonzero entries to the total number of entries of $\tilde{\mathbf{D}}_n$. We observe that the larger the dimension of $\tilde{\mathbf{D}}_n$, the smaller the ratio. For example, C.R. is only 0.0389 or 3.89% for $n = 10$, which means you can only calculate less than 4 out of 100 entries. To visualize the sparseness of the matrix, we plot in Fig. 4 the distribution of nonzero entries of $\mathbf{D}_{10}$ and $\tilde{\mathbf{D}}_{10}$, in which the nonzero entries are identified by a black region.

A comparison of the absolute error between our method (FMGA) and the methods in [17–19] is reported in Table 3, which reveals that the our results have superiority in accuracy.

Example 2 Consider the following Bagley–Torvik equation [17, 28]:

$$u''(t) + \theta D^{\alpha} u(t) = -1 - e^{t-1}, \quad 0 \le t \le 1,$$

$$u(0) = u(1) = 0.$$

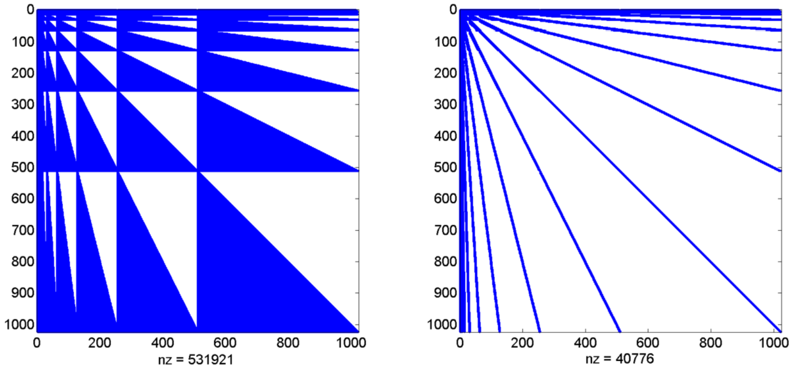

Figure 4 Distribution of nonzero entries of matrices $\mathbf{D}_n$ (left) and $\tilde{\mathbf{D}}_n$ (right) with $n = 10$

Table 3 Comparison of the absolute error with methods in [17–19] for Example 1

x	Our method	GD in [17]	NSGD in [17]	3-WSGD in [17]	Method in [18]	Method in [19]
0.125	2.56e-7	8.36e-5	3.05e-5	8.16e-6	2.00e-3	1.73e-4
0.250	1.06e-6	1.59e-4	5.80e-5	1.73e-5	4.08e-3	5.35e-4
0.375	2.19e-6	2.09e-4	7.31e-5	2.70e-5	5.83e-3	7.98e-4
0.500	5.03e-6	2.01e-4	6.24e-5	3.56e-5	6.85e-3	6.74e-4
0.625	8.82e-6	1.34e-4	1.60e-5	4.06e-5	6.81e-3	9.50e-5
0.750	1.11e-5	1.12e-5	6.18e-5	3.86e-5	5.57e-3	1.78e-3
0.875	1.27e-5	7.81e-5	1.51e-4	2.60e-5	3.26e-3	3.42e-3

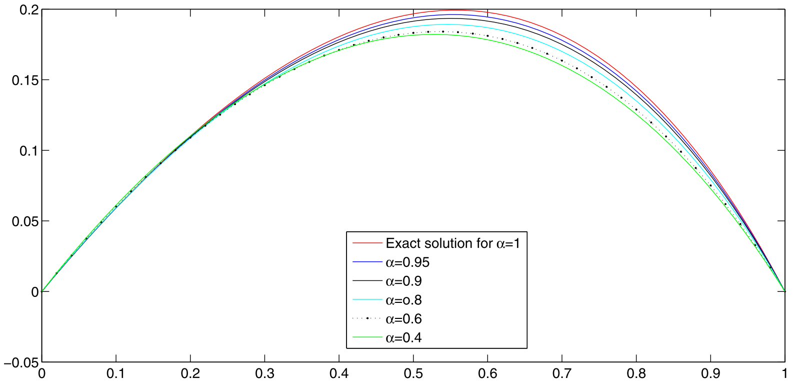

Figure 5 Exact and numerical solutions for Example 2 for different α

The exact solution for $\theta = -1$, $\alpha = 1$ is $u^*(t) = t(1 - e^{t-1})$. For general values of α, the exact solution is not known. We set up this example to test the effectiveness of our algorithm. Let $\theta = -1$, we solve the equation by FMGA for $\alpha = 0.4, 0.6, 0.8, 0.9, 0.95$. The corresponding numerical solutions are shown in Fig. 5. We observe that as α approaches 1, the solutions of the fractional order equation converge to the exact solution of the integer order ($\alpha = 1$) equation.

Example 3 As the last example we consider the equation [12, 17, 28]

$$u''(t) + D^{\alpha}u(t) = g(t) + t^{1-\alpha}p(t), \qquad 0 \le t \le 1,$$
$$u(0) = u(1) = 0,$$

where

$$g(t) = 20t^3 - \frac{174}{5}t^2 + \frac{456}{25}t - \frac{339}{125},$$
$$p(t) = \frac{120}{\Gamma(6-\alpha)}t^4 - \frac{348}{5\Gamma(5-\alpha)}t^3 + \frac{456}{25\Gamma(4-\alpha)}t^2 - \frac{339}{125\Gamma(3-\alpha)}t + \frac{27}{125\Gamma(2-\alpha)}.$$

The exact solution is

$$u^*(t) = t^5 - \frac{29}{10}t^4 + \frac{76}{25}t^3 - \frac{339}{250}t^2 + \frac{27}{125}t.$$

The numerical results obtained by FMGA are presented to show the accuracy, convergence and computing time in Table 4, Fig. 6 and Fig. 7. We also compare the absolute error of our method with that of the methods in [12, 17, 28] and listed in Table 5, where $n^* - 1$ and m are the number of the spline and Bessel functions used in the approximate solution in [17] and [12, 28], respectively. These results show that our algorithm has better accuracy with fewer basis functions ($x(3) = 7$).

Table 4 Numerical results for Example 3

n	$x(n)$	$\Vert u^* - \tilde{u}_n\Vert_1$	C.O.	$\Vert u^* - \tilde{u}_n\Vert_2$	C.O.	$\Vert u^* - \tilde{u}_n\Vert_\infty$	C.O.	C.R.	T
3	7	2.3740e-2		9.2810e-4		3.3646e-3		77.6%	<0.01
4	15	1.2346e-2	0.94328	2.4402e-4	1.9273	9.5997e-4	1.8094	62.2%	0.01
5	31	6.2333e-3	0.98597	6.2106e-5	1.9742	2.7120e-4	1.8236	44.3%	0.04
6	63	3.1242e-3	0.99651	1.5368e-5	2.0148	6.1689e-5	2.1363	29.3%	0.12
7	127	1.5630e-3	0.99913	3.8719e-6	1.9888	1.7795e-5	1.7935	18.5%	0.29
8	255	7.8164e-4	0.99978	9.9052e-7	1.9668	3.3084e-6	2.5501	11.2%	0.73
9	511	3.9084e-4	0.99994	2.8535e-7	1.7954	1.0062e-6	1.5944	6.69%	2.03

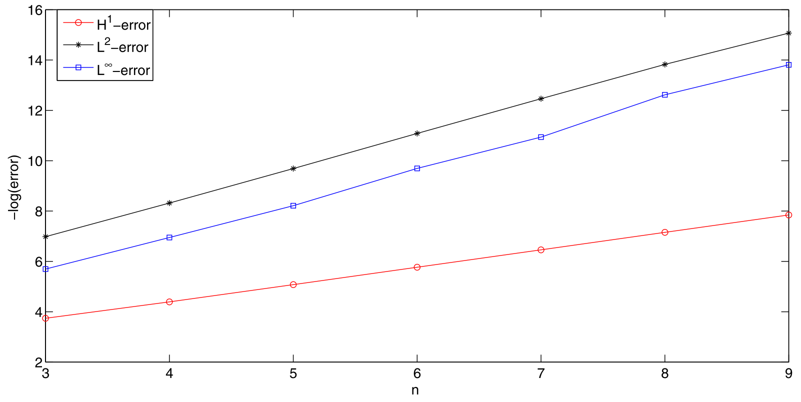

Figure 6 Errors of the numerical solution for Example 3 with $\alpha = 0.5$

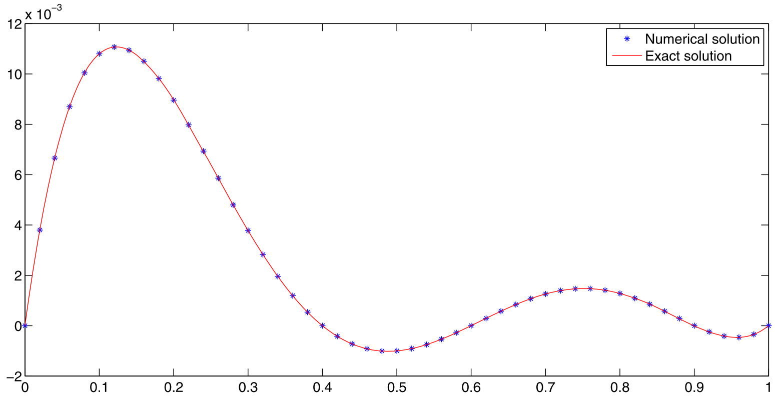

Figure 7 Exact and numerical solution for Example 3 with $\alpha = 0.5$, $n = 9$

Table 5 Comparison of the absolute error with the methods in [12, 17, 28] for Example 3

x	Our method	Method in [17]	Method in [12]	Method in [28]
	$n = 3$	$n^* = 10$	$m = 8$	$m = 8$
0.1	1.91e-3	2.00e-3	1.08e-2	3.59e-3
0.2	6.30e-4	1.74e-3	8.96e-2	1.58e-3
0.3	4.17e-4	1.53e-3	3.78e-2	1.79e-3
0.4	4.38e-4	1.32e-3	1.44e-7	1.63e-3
0.5	7.23e-5	1.11e-3	1.01e-3	1.16e-3
0.6	1.45e-4	8.92e-4	6.62e-8	5.84e-4
0.7	1.87e-4	6.53e-4	1.26e-3	1.27e-4
0.8	2.19e-4	4.19e-4	1.28e-4	1.19e-4
0.9	3.06e-4	2.02e-4	2.07e-8	5.54e-4

5 Conclusion

In this paper, a fast multiscale Galerkin algorithm based on a matrix truncation strategy for solving boundary value problem of the Caputo fractional Bagley–Torvik equation is presented. The algorithm reduces the computational complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$ (N is the dimension of the approximation space), which makes the algorithm efficient. Examples are presented to illustrate the performance of our algorithm by comparing it with the original Galerkin scheme (OMGM) from accuracy, convergence order and computing time. It shows that they have the same convergence orders and accuracy, but FMGA is much faster than OMGM. Numerical results are also compared with some recent methods, which demonstrate the effectiveness of our algorithm and its superiority in accuracy. The proposed algorithm still needs rigorous theoretical analysis, which constitutes the subject of our ongoing work.

Acknowledgements
Not applicable.

Funding
This work has been supported in part by the Natural Science Foundation of China under grant 11501106 and by the Natural Science Foundation of Guangdong Province under grant 2016A030313835 and 2018A030313258.

Abbreviations
Not applicable.

Availability of data and materials
Not applicable.

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors declare that he has no competing interests.

Consent for publication
Not applicable.

Authors' contributions
The author completed the paper, read and approved the final manuscript.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 20 February 2020 Accepted: 3 May 2020 Published online: 12 May 2020

References

1. Caputo, M., Mainardi, F.: Linear models of dissipation in anelastic solids. Riv. Nuovo Cimento **1**, 161–198 (1971)
2. Kilbas, A., Srivastava, H., Trujillo, J.: Theory and Applications of Fractional Differential Equations. Elsevier, Amsterdam (2006)
3. Metzler, R., Klafter, J.: The random walks guide to anomalous diffusion: a fractional dynamics approach. Phys. Rep. **339**, 1–77 (2000)
4. Oldham, K.B., Spanier, J.: The Fractional Calculus. Academic Press, New York (1974)
5. Freed, A., Diethelm, K.: Fractional calculus in biomechanics: a 3D viscoelastic model using regularized fractional-derivative kernels with application to the human calcaneal fat pad. Biomech. Model. Mechanobiol. **5**, 203–215 (2006)
6. Marin, M., Nicaise, S.: Existence and stability results for thermoelastic dipolar bodies with double porosity. Contin. Mech. Thermodyn. **28**(6), 1645–1657 (2016)
7. Marin, M., Vlase, S., Ellahi, R., Bhatti, M.M.: On the partition of energies for the backward in time problem of thermoelastic materials with a dipolar structure. Symmetry **11**(7), Article ID 863 (2019)
8. Marin, M., Bhatti, M.M.: Head-on collision between capillary-gravity solitary waves. Bound. Value Probl. **2020**(1), Article ID 12 (2020)
9. Marin, M., Ellahi, R., Chiril, A.: On solutions of Saint-Venant's problem for elastic dipolar bodies with voids. Carpath. J. Math. **33**(2), 219–232 (2017)
10. Alsuyuti, M.M., Doha, E.H., Ezz-Eldien, S.S., Bayoumi, B.I., Baleanu, D.: Modified Galerkin algorithm for solving multitype fractional differential equations. Math. Methods Appl. Sci. **42**(5), 1389–1412 (2019)
11. Groza, G., Khan, S., Pop, N.: Approximate solutions of boundary value problems for ODEs using Newton interpolating series. Carpath. J. Math. **25**(1), 73–81 (2009)
12. Yüzbasi, S.: Numerical solution of the Bagley–Torvik equation by the Bessel collocation method. Math. Methods Appl. Sci. **36**, 300–312 (2013)
13. El-Gamel, M., El-Hady, M.A.: Numerical solution of the Bagley–Torvik equation by Legendre-collocation method. SeMA J. **74**(4), 371–383 (2017)
14. Ray, S.S.: On Haar wavelet operational matrix of general order and its applications for numerical solution of fractional Bagley–Torvik equation. Appl. Math. Comput. **218**, 5239–5248 (2012)
15. Balaji, S., Hariharan, G.: An efficient operational matrix method for the numerical solutions of the fractional Bagley–Torvik equation using wavelets. J. Math. Chem. **57**(8), 1885–1901 (2019)
16. Stynes, M., Gracia, J.L.: A fifinite difference method for a two-point boundary value problem with a Caputo fractional derivative. IMA J. Numer. Anal. **35**(2), 698–721 (2015)
17. Zahra, W.K., Van Daele, M.: Discrete spline method for solving two point fractional Bagley–Torvik equation. Appl. Math. Comput. **296**, 42–56 (2017)
18. Zahra, W.K., Elkholy, S.M.: Cubic spline solution of fractional Bagley–Torvik equation. Electron. J. Math. Anal. Appl. **1**, 230–241 (2013)
19. Zahra, W.K., Elkholy, S.M.: The use of cubic splines in the numerical solution of fractional differential equations. Int. J. Math. Math. Sci. **2012**, Article ID 638026 (2012)
20. Karaaslana, M.F., Celiker, F., Kurulay, M.: Approximate solution of the Bagley–Torvik equation by hybridizable discontinuous Galerkin methods. Appl. Math. Comput. **285**, 51–58 (2016)
21. Mokhtary, P.: Numerical treatment of a well-posed Chebyshev tau method for Bagley–Torvik equation with high-order of accuracy. Numer. Algorithms **72**(4), 875–891 (2016)
22. Torvik, P.J., Bagley, R.L.: On the appearance of the fractional derivative in the behavior of real materials. J. Appl. Mech. **51**(2), 294–298 (1984)
23. Podlubny, I.: Fractional Differential Equations. Academic Press, San Diego (1999)
24. Chen, Z., Wu, B., Xu, Y.: Multilevel augmentation methods for differential equations. Adv. Comput. Math. **24**, 213–238 (2006)

25. Micchelli, C.A., Xu, Y., Zhao, Y.: Wavelet Galerkin methods for second-kind integral equations. J. Comput. Appl. Math. **86**, 251–270 (1997)
26. Chen, Z., Micchelli, C.A., Xu, Y.: Multiscale Methods for Fredholm Integral Equations. Cambridge University Press, Cambridge (2015)
27. Huang, M.: Wavelet Petrov–Galerkin algorithms for Fredholm integral equations of the second kind. Ph.D. thesis, Academy Scinece of China (2003)
28. Rehman, M.U., Khan, R.A.: A numerical method for solving boundary value problems for fractional differential equations. Appl. Math. Model. **36**, 894–907 (2012)